POLYPUBLIE

Une plateforme de mesure de caractéristiques électroniques pour les nanostructures

Olivier Duval

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

Date	2005
Type	Mémoire ou thèse / Dissertation or Thesis Mémoire de maîtrise/Master's thesis
URL de PolyPublie PolyPublie URL	https://publications.polymtl.ca/7364/
Conditions d'utilisation Terms of use	© Tous droits réservés / All rights reserved
Référence Citation	Duval, O. (2005). Une plateforme de mesure de caractéristiques électroniques pour les nanostructures [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/7364/

Détails de la thèse
Thesis details

Directeurs de recherche Advisors	Yvon Savaria
Programme Program	Non spécifié

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal
https://publications.polymtl.ca

NOTE TO USERS

This reproduction is the best copy available.

UMI®

UNIVERSITÉ DE MONTRÉAL

UNE PLATEFORME DE MESURE DE CARACTÉRISTIQUES ÉLECTRONIQUES POUR LES NANOSTRUCTURES

OLIVIER DUVAL
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(MICROÉLECTRONIQUE)
MAI 2005

© Olivier Duval, 2005.

Library and Archives Canada
Bibliothèque et Archives Canada

Published Heritage Branch
Direction du Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-494-01310-9
Our file Notre référence
ISBN: 0-494-01310-9

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:

UNE PLATEFORME DE MESURE DE CARACTÉRISTIQUES ÉLECTRONIQUES POUR LES NANOSTRUCTURES

présenté par: DUVAL Olivier

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

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

M LAURIN Jean-Jacques, PhD, président

M SAVARIA Yvon, PhD, membre et directeur de recherche

M KHOUAS Abdelhakim, PhD, membre

À Gordon Moore, prophète malgré lui.

REMERCIEMENTS

Merci à Marc-André Cantin, Louis-Pierre Lafrance, Jean-François Coulombe, Abdelaziz Trabelsi et Yamu Hu pour leur aide inestimable et soutenue tout au long de ma maîtrise.

Merci à Patrick Desjardins, Richard Martel et Alain Rochefort de m'avoir partagé une part de leurs connaissances immenses.

Merci à mon directeur Yvon Savaria pour avoir allumé en moi la flamme de la recherche scientifique. En gardant son enthousiasme malgré des difficultés et des échecs déchirants, il m'a grandement aidé à garder le cap tout au long de ce projet.

Merci à mes parents François et Francine, qui ne m'ont jamais empêché d'embarquer dans les projets les plus ambitieux.

Merci à mes amis Pierre-Alexandre et Guillaume, vous m'avez sauvé du pétrin en maintes occasions.

Merci à Christine, ton soutien m'a permis d'oublier mes angoisses et de garder le sourire.

RÉSUMÉ

Ce projet vise à réduire le pont séparant l'univers de la microélectronique de celui des nanotechnologies. La fabrication d'une plateforme microélectronique de mesure de caractéristiques électriques de nanostructures s'inscrit doublement dans cet objectif.

D'abord, la façon de lier la nanostructure sous test à la plateforme est un problème important. En effet, les méthodes de fabrication de plusieurs nanostructures sont actuellement incompatibles avec la fabrication de circuits électroniques conventionnels. Nous avons donc conçu la plateforme de test de façon à recevoir les nanostructures sur des électrodes fabriquées à même la couche métallique supérieure du circuit.

Une fois fabriquée, la plateforme permet la mesure de caractéristiques électriques de nanostructures dans le but d'établir des modèles de ces nanostructures.

Dans le cadre de ce projet, nous avons conçu et fabriqué une plateforme de test en utilisant la technologie TSMC CMOS 180nm. Cette plateforme de test n'a pas fonctionné comme les simulations le prévoyait à cause de problèmes de design. Pour la création de modèles de nanostructures, nous avons toutefois établi un modèle de transistor à base de nanotube de carbone.

Les résultats de la recherche que ce mémoire rapporte s'inscrivent dans un projet qui s'échelonnera sur plusieurs années et qui utilisera ces résultats pour, espérons-le, fabriquer un jour un nanodispositif fonctionnel, une tâche très ardue à l'heure actuelle.

ABSTRACT

This project aims at the reduction of the gap separating the microelectronic to nanotechnology worlds. The fabrication of a microelectronic platform for the measurement of electronic characteristics in nanostructures answers to this objective in two distinct ways.

First, linking nanostructures under test to a platform is not a simple task. Most nanooelectronic fabrication methods currently used are not compatible with the creation of microelectronic devices. We have designed and built this test platform to permit the inclusion of nanostructures on electrodes built on the top layer of the circuit. Secondly, by measuring electronic characteristics for a given nanostructure, we used these measurements to complete electronic models describing their behaviour.

For this project, we have designed and built a test platform using the TSMC CMOS 180nm technology. This platform did not work as predicted because of problems in the design of its high-level architecture. Nonetheless, we used prior experimental nanostructure measurements in order to create a model for a carbon nanotube based transistor.

The research results for this thesis are part of a project that spans over a long period of time. We can only hope that our successors will use the results of this research to create a useful nanodevice, which is very difficult for now.

TABLE DES MATIÈRES

DÉDICACE iv

REMERCIEMENTS v

RÉSUMÉ vi

ABSTRACT vii

TABLE DES MATIÈRES viii

LISTE DES FIGURES xii

LISTE DES TABLEAUX xv

LISTE DES NOTATIONS ET DES SYMBOLES xvi

LISTE DES ANNEXES xvii

INTRODUCTION 1

CHAPITRE 1 REVUE DE LITTÉRATURE 3

1.1 Nanostructures 3

1.1.1 CMOS à double grille 3

1.1.2 Dispositifs quantiques 5

1.1.2.1 Diodes à effet tunnel 5

1.1.2.2 Spintronics 9

1.1.3 Nanotubes de carbone 14

1.1.3.1 Fullerènes 14

1.1.3.2 Premiers nanotubes de carbone 15

1.1.3.3 Structures moléculaires de nanotubes 16

1.1.3.4 Propriétés électriques des SWCNTs 17
1.1.3.5 Propriétés électriques des MWNTs 18
1.1.3.6 Méthodes de fabrication de CNTs 19
1.1.3.7 Fabrication de CNTFETs 21
1.1.3.8 Placement des CNTs sur une paire d'électrodes . . 22
1.1.3.9 Tri des Nanotubes de Carbone 23
1.1.3.10 Exemple d'application: une porte logique en CNT 26
1.1.3.11 Applications des nanotubes 27
1.2 Mesures de précision 29
1.2.1 Mesure de délais 30
1.2.1.1 Télécommunications 30
1.2.1.2 VLSI 32
1.2.1.3 Autres sources sur des mesures intégrées à haute précision 36

CHAPITRE 2 PLATEFORME DE TEST 39
2.1 Mesures à effectuer 40
2.1.1 Mesure I-V 40
2.1.1.1 Précision et modèle recherchés 41
2.1.2 Mesure de délais 42
2.1.2.1 Précision recherchée 42
2.2 Système électronique de mesure et contrôle 43
2.2.1 Module de mesure DC 44
2.2.1.1 Porte de tansmission utilisée dans le module de mesure DC 46
2.2.2 Module de validation d'intégrité technologique 47
2.2.3 Module de JTAG 48
2.2.4 Module de mesure de délai 49

2.2.4.1 Module de mesure de délai proposé 49
2.2.4.2 Formules utilisées pour calculer le délai mesuré . . 52
2.2.4.3 Discussion sur la précision du module de mesure de délai 53
2.3 Intégration de nanostructures 54
2.3.1 Architectures utilisables 55
2.3.1.1 Plaque de silicium "nue" 55
2.3.1.2 Circuits intégrés configurables 56
2.3.1.3 Circuits intégrés à application spécifique 56
2.3.2 Intégration de CNTs à la plateforme de test 57
2.3.2.1 Choix de la méthode de fabrication 57
2.3.2.2 Choix de la méthode de placement 58
2.3.3 Conception de la surface d'électrodes de contact 59
2.3.3.1 Emplacement des électrodes de contact 60
2.3.3.2 Liaison des électrodes aux modules de mesure internes 61
2.4 Conclusions du chapitre 61

CHAPITRE 3 MODÉLISATION D'UN NANO-COMPOSANT 62
3.1 Modèles de transistors 62
3.2 Modèle et données expérimentales traités avec le programme 63
3.2.1 CNTFET de McEuen et al. 64
3.2.2 Modèle de FET de Curtice et Ettenberg modifié 65
3.2.3 Description des paramètres fixes du modèle de Curtice et Ettenberg 66
3.3 Programme de génération de paramètres pour le modèle C. & E. . 67
3.3.1 Pseudocode de l'algorithme génétique utilisé dans ce programme 67
3.3.2 Description du code du programme 68

3.4 Résultats 71
3.5 Conclusion du chapitre 73

CHAPITRE 4 CARACTÉRISATION PAR SIMULATION DES STRUCTURES DE TEST 74
4.1 Résultats de simulation du module DC 74
4.1.1 Incertitude structurelle du module DC 74
4.1.2 Incertitude du module DC causée par les aléas de fabrication 75
4.1.2.1 Définition des tests de coin 76
4.1.2.2 Description des tests de coin 77
4.1.3 Discussion des résultats de simulation DC 78
4.2 Simulation du module de mesure de délai 80
4.2.1 Simulation numérique du module de mesure de délai 80
4.2.2 Simulation analogique du module de mesure de délai 82
4.3 Conclusion des simulations 83

CHAPITRE 5 RÉSULTATS 86
5.1 Évaluation du procédé de fabrication 86
5.1.1 Électrodes de test 87
5.1.2 Structure de caractérisation hors règles de design 87
5.2 Mesures effectuées avec la plateforme 88
5.2.1 Montage du circuit de mesure sur carte 89
5.2.2 Mesures effectuées sur l'analyseur logique 89
5.3 Conclusion des mesures 90

CONCLUSION 91

RÉFÉRENCES 93

ANNEXES 99

LISTE DES FIGURES

Figure 1.1	MOSFET à double grille planaire	4
Figure 1.2	MOSFET à double grille non-planaire	5
Figure 1.3	Alignement des états quasi-liés dans une RTD: a. Maximum d'énergie pour l'effet tunnel b. État de résonance c. Minimum d'énergie pour l'effet tunnel	7
Figure 1.4	Courbe I-V d'une RTD	8
Figure 1.5	Une mémoire bistable à base de RTD	8
Figure 1.6	Topologie de mémoire FIFO-RTD en 3D	9
Figure 1.7	Dessin de masques d'une mémoire 128kB en MRAM	11
Figure 1.8	Schéma simplifié d'une cellule 1T1MTJ de IBM	11
Figure 1.9	Schéma simplifié d'une cellule 1T1MTJ de Motorola	12
Figure 1.10	Fonctionnement interne d'une cellule MTJ	12
Figure 1.11	Une molécule de fullerène	14
Figure 1.12	Senseur piézoélectrique en fullerène	15
Figure 1.13	Coupes de plans de graphite	16
Figure 1.14	Nanotubes à simple et multi-couche	17
Figure 1.15	Procédés de fabrication de nanotubes	20
Figure 1.16	Un des premiers modèles de CNTFETs	21
Figure 1.17	Modèle récent de CNTFET	22
Figure 1.18	Système de nanofabrication	24
Figure 1.19	Méthode d'effeuillage de MWNTs	25
Figure 1.20	Mesures effectuées en cours de filtration	26
Figure 1.21	Écrasement de MWCNT entre une pointe d'AFM et un substrat	27
Figure 1.22	Un inverseur logique à base de CNTFETs	28
Figure 1.23	Mesures de permittivité relative et de $\propto$conductivité	29

Figure 1.24	Une pointe d'AFM avec terminaison en CNT	30
Figure 1.25	Ligne de délai pour la synchronisation	31
Figure 1.26	Système de mesure de délais de Petre et al.	32
Figure 1.27	Montage de mesure de délai	33
Figure 1.28	Configuration de mesure de délai d'interconnexions	34
Figure 1.29	Oscillateur bouclé de Pera et al.	35
Figure 1.30	Détecteur de phase de Pera et al.	36
Figure 1.31	Mesure de délai de Sato et al.	38
Figure 2.1	courbes I-V d'un transistor CMOS et d'un fil de cuivre	41
Figure 2.2	Système électronique global	43
Figure 2.3	Miroir de courant du module de mesure DC	45
Figure 2.4	Porte de transmission utilisée dans le module	46
Figure 2.5	Le module d'autotest	47
Figure 2.6	Emplacement des sondes d'autotest	48
Figure 2.7	Schéma-bloc du module de mesure de délai	50
Figure 2.8	Oscillateur bouclé configurable	51
Figure 2.9	Dessin de masques des électrodes de contact	60
Figure 3.1	Transistor électrolytique de McEuen et al.	64
Figure 3.2	Courbe I-V du transistor de McEuen et al.	65
Figure 3.3	Courbe I-V expérimentale de McEuen vs modèle de Curtice et Ettenberg	72
Figure 4.1	Précision de la mesure du module DC	75
Figure 4.2	Test de coin de référence (R=1Ω)	78
Figure 4.3	Test de coin avec R=600Ω	78
Figure 4.4	Test de coin avec R=12kΩ	79
Figure 4.5	Test de coin avec R=200kΩ	79
Figure 4.6	Simulation numérique du module de mesure de délai	81

Figure 4.7 Circuit utilisé pour la simulation analogique du module de délai 82
Figure 4.8 Simulation analogique du module de mesure de délai 83
Figure 4.9 Circuit utilisé pour les tests de coin 85
Figure 5.1 Micrographe des électrodes fabriquées 87
Figure 5.2 Micrographe des électrodes fabriquées 88

LISTE DES TABLEAUX

Tableau 3.1 Définition des paramètres du modèle C. & E. modifié 66

Tableau 3.2 Paramètres du modèle du transistor de McEuen 73

LISTE DES NOTATIONS ET DES SYMBOLES

LISTE DES ACRONYMES

AFM	Atomic Force Microscope
CMOS	Complementary Metal Oxide Semiconductor
CNT	Carbon NanoTube
CFP	Ceramic Flat Pack
CVD	Chemical Vapor Deposition
dg-MOSFET	double-gated MOSFET
FED	Field Emission Display
FIFO-RTD	Élément de mémoire First In First Out en RTD
GHz	GigaHertz
IBM	International Business Machines Corporation
JTAG	Joint Test Action Group
LASER	Light Amplification by Stimulated Emission of Radiation
MBE	Molecular Beam Epitaxy
MOS	Metal-Oxide Semiconductor
MOSFET	Metal-Oxide Semiconductor Field-Effect Transistor
MRAM	Magnetic Random Access Memory
MWCNT	Multi-Wall CNT
NDR	Negative Differential Resistance
NEC	Nippon Electric Company, Ltd.
NVRAM	Non-Volatile Random Access Memory
RAM	Random Access Memory
RF	Radio Frequency
RTD	Resonant-Tunneling Diode
SWCNT	Single-Wall CNT
TSMC	Taiwan Semiconductor Manufacturing Corporation
VDD	Voltage Drain Drain

LISTE DES ANNEXES

ANNEXE I CODE VHDL DE LA PLATEFORME DE TEST 99

ANNEXE II CODE SCHEME DU MODÈLE DE CURTICE ET ETTENBERG 173

ANNEXE III ARTICLE DE JIN SUR LE MODULE DE CURTICE ET ETTENBERG 191

INTRODUCTION

La très grande majorité des circuits électroniques sont fabriqués avec une technologie CMOS. La robustesse, la rapidité de commutation, l'accessibilité de fabrication à grande échelle et la faible consommation énergétique des circuits CMOS ont poussé l'industrie des semiconducteurs à améliorer les performances de cette technologie des années 1970 jusqu'à nos jours. Notre société pourrait difficilement se passer d'ordinateurs, de téléphones cellulaires, d'équipements médicaux, qui demandent des améliorations constantes de cette technologie.

Cette forte demande a poussé la recherche, notamment dans le domaine de la microfabrication, à fournir des efforts gigantesques depuis plus de 40 ans. La réduction d'échelle continue alimente la prophétie annoncée en 1965 par Gordon Moore, cofondateur de Intel corp., voulant que la densité en transistors sur les circuits intégrés double à tous les 18 mois. Plusieurs obstacles de taille ont été surmontés contre toute attente durant ces décennies. Le prochain obstacle qui se dresse devant l'entreprise de réduction d'échelle du CMOS risque fort de ne jamais être surmonté: les dimensions des transistors s'approchent de la taille de l'atome de silicium.

C'est face à ce mur que la recherche de dispositifs nanoélectroniques a littéralement explosé depuis environ une décennie. Les premiers dispositifs commerciaux exploitant des nanostructures sont apparus sur le marché, mais un véritable dispositif nanoélectronique actif fabriqué en série tarde à venir.

La première approche pour la création de tels dispositifs consiste d'abord à créer un hybride entre les nouveaux nano-dispositifs et la technologie CMOS, dont les méthodes de fabrication sont éprouvées depuis longtemps. Une première approche consiste donc à fabriquer une plateforme de test sur laquelle on intègrera des nanostructures. Cette plateforme répond donc à un double objectif:

- Offrir une première hybridation entre le procédé CMOS et des nanotechnologies.
- Mesurer des caractéristiques électriques inconnues de ces nanostructures dans le but de créer et d'améliorer des modèles de nano-disopsitifs.

Ce document débute avec une revue de la littérature touchant la fabrication de nanostructures et des techniques de mesure électroniques intégrées avancées. Au deuxième chapitre, nous verrons l'architecture envisagée pour la fabrication de la plateforme de mesure de caractéristiques électroniques de nanostructures. Au chapitre 3, nous verrons un modèle de transistor à base de nanotube de carbone que nous avons perfectionné. Le chapitre 4 montre les simulations que nous avons effectuées lors de la conception de la plateforme de test et nos résultats expérimentaux sont présentés au 5^{ieme} chapitre.

CHAPITRE 1

REVUE DE LITTÉRATURE

1.1 Nanostructures

Ce projet vise l'intégration de nanostructures à un circuit électronique standard. Une revue de littérature importante est nécessaire afin de choisir un nombre limité de nanostructures jugées à la fois prometteuses au niveau des performances électroniques et dont l'intégration à un circuit CMOS standard est plausible. Cette section met en relief des nanostructures qui répondent à ces critères à différents niveaux en mettant un accent sur les nanotubes de carbone qui offrent, aux yeux de l'auteur, le meilleur tandem performance-intégrabilité.

1.1.1 CMOS à double grille

Un large éventail de variations à des procédés CMOS standard est actuellement disponible chez les fabriquants de semiconducteurs. Par exemple, une architecture basée sur des hétérojonctions bipolaires en AlGaAs/GaAs permet de créer des systèmes plus performants à haute fréquence. Il est donc possible d'utiliser la technologie sur silicium existante comme base pour créer des dispositifs alternatifs au transistor CMOS standard, chacune de ces alternatives ayant ses applications spécialisées.

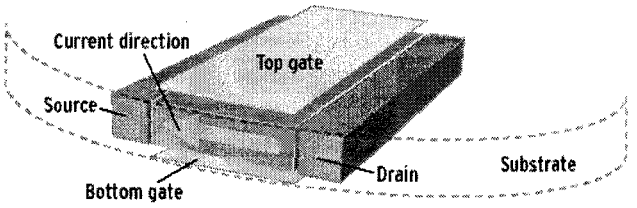
Nous nous intéressons ici à l'ajout d'une seconde grille à un transistor MOS conventionnel pour former des *dual-gated MOSFETs*. Des estimations qui datent de 1992 montrent que des dg-MOSFETs repoussent la limite de réduction d'échelle du

CMOS à 30 nm (Frank, 1992). Des simulations démontrent qu'un dg-MOSFET de type n réduit les effets de canaux courts, permettant ainsi une réduction d'échelle substantielle.

Il est aussi possible d'utiliser cette architecture à une échelle plus grande. En effet, Zhang (Zhang, 2003) a fabriqué un dispositif dont le canal mesure 0,35 μm. Ce dispositif est destiné à des applications pour lesquelles la tension de seuil change dynamiquement. La fabrication de ce dispositif montre sa faisabilité, mais la fabrication de plus petits dispositifs demeure un défi important.

Deux architectures de dg-MOSFETS sont proposées. L'architecture planaire (fig1.1) est similaire à un transistor CMOS standard, mais auquel on a ajouté une grille sous le canal. Bien qu'elle prenne avantage des procédés de fabrication existants, cette méthode amène des défis de fabrication importants. La fabrication de la grille du dessous et du diélectrique qui la jouxte est plutôt difficile. De plus, se connecter à la grille du dessous est difficile et réduit la densité d'intégration.

L'architecture non-planaire (fig 1.2) permet de contourner les problèmes relatifs aux dg-MOSFETS mentionnés ci-haut. Par contre, cette approche compromet l'uniformité dans l'épaisseur du canal, ce qui nuit aux performances du transistor. Cette approche complique aussi la fabrication des autres parties du transistor, car elle s'éloigne beaucoup des procédés de fabrication couramment utilisés.



Geppert, L., IEEE Spectrum, October 2002, p.33.

Figure 1.1 MOSFET à double grille planaire

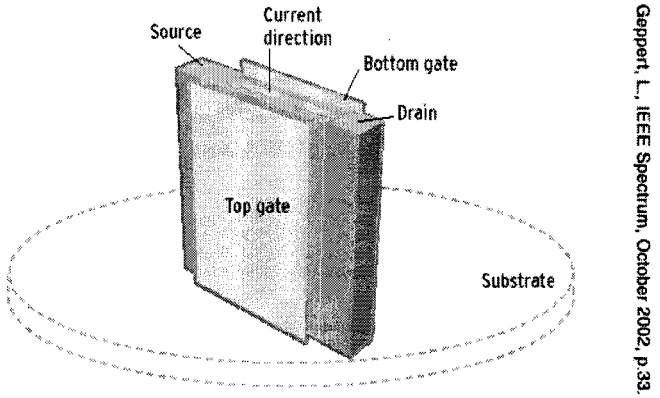


Figure 1.2 MOSFET à double grille non-planaire

Les transistors CMOS à double grille offrent donc des performances intéressantes et leur fabrication est relativement simple par rapport à d'autres nanostructures. Par contre, leur miniaturisation demandera des méthodes de fabrication qui résolveront les problèmes d'alignement des grilles inhérents à cette architecture.

1.1.2 Dispositifs quantiques

Dans cette section, nous décrivons la fabrication et le comportement de dispositifs nanoélectroniques prometteurs. Néanmoins, la fabrication de ces dispositifs est complexe au point où leur utilisation dans un circuit commercialisable apparaît peu probable à moyen terme.

1.1.2.1 Diodes à effet tunnel

La physique classique ne peut expliquer l'effet tunnel, il faut se tourner vers la physique quantique pour comprendre ce phénomène. Selon la physique classique, si une particule frappe une barrière de potentiel dont l'énergie V est plus élevée que sa propre énergie E, elle est repoussée. Sinon, elle peut passer l'obstacle qui

induit cette barrière. Selon la physique quantique, la particule a une probabilité non-nulle de passer l'obstacle même si $E < V$, mais cette probabilité est faible tant que E n'est pas près de V. Si la particule passe l'obstacle alors que $E < V$, on assiste à l'effet tunnel et la particule émerge de l'autre côté de l'obstacle avec la même énergie E.

La diode à effet tunnel résonant (RTD) est un dispositif qui exploite l'effet tunnel. Dans une façon populaire de les fabriquer, elles sont composées de minces couches de matériaux à large bande interdite, tel le $AlGaAs$. Divers composants moléculaires peuvent aussi se comporter comme des RTD. Les couches externes du dispositif servent de barrière d'énergie potentielle pour former des états quasi-liés dans la couche centrale. Pour permettre la fabrication des couches minces d'éléments à concentration contrôlée, on fait appel à des techniques de déposition épitaxiale telle que l'épitaxie par jets moléculaires (MBE) (Mathews, 1999).

La figure 1.3 illustre la structure et le fonctionnement de ce dispositif. En partant de la gauche, un électron tente de traverser les barrières énergétiques que constituent la diode pour continuer son chemin vers la droite du graphique. Les lignes pointillées dans ces graphiques illustrent les états quasi-liés selon trois niveaux de tension de polarisation dans le dispositif.

La figure 1.3b illustre la RTD en résonance[1] alors que les figures 1.3a et 1.3c montrent les limites supérieure et inférieure de tension de polarisation en deçà desquelles l'effet tunnel a peu de chances de se manifester. La figure 1.4 illustre les courbes I-V simulées et mesurées par Mathews et al. Lorsque la tension de polarisation est située entre 1 et 2 V, le RTD entre en résonance et offre une résistance différentielle négative (NDR).

[1]La tension de polarisation appliquée sur la RTD aligne l'état quasi-lié entre les barrières avec l'énergie des états fortement occupés adjacents

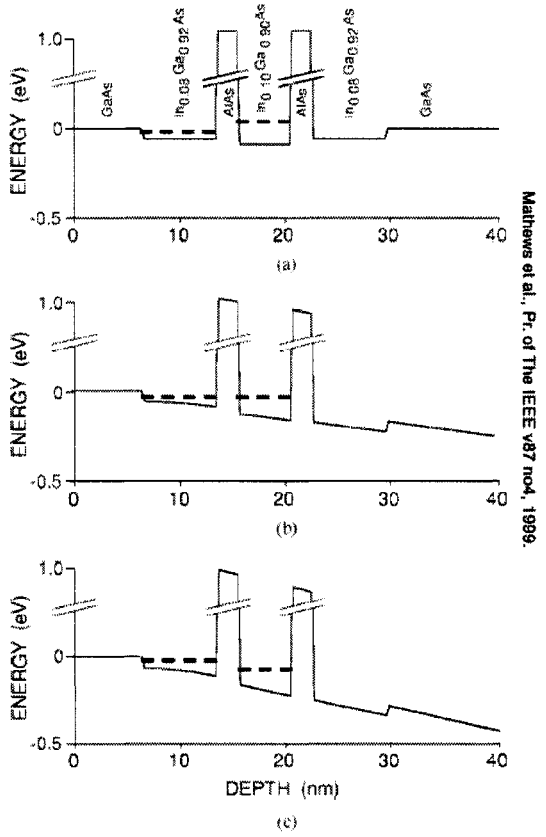


Figure 1.3 Alignement des états quasi-liés dans une RTD: a. Maximum d'énergie pour l'effet tunnel b. État de résonance c. Minimum d'énergie pour l'effet tunnel

Cette caractéristique des RTDs, combinée à leur faible capacité parasite et à leur forte densité de courant, fait de cette nanostructure un élément de circuit non-linéaire très rapide. Plusieurs applications électroniques ont été proposées et expérimentées pour exploiter les RTDs:

1. L'intégration de RTDs dans un circuit CMOS standard (Bergman, 1999) en améliore la performance en termes de vitesse, de consommation de puissance et de densité de dispositifs actifs. Des RTDs ont été utilisées dans plusieurs circuits à haute performance tels des oscillateurs, mélangeurs, multiplieurs de fréquence, détecteurs RF et comme étage d'entrée de circuits analogiques

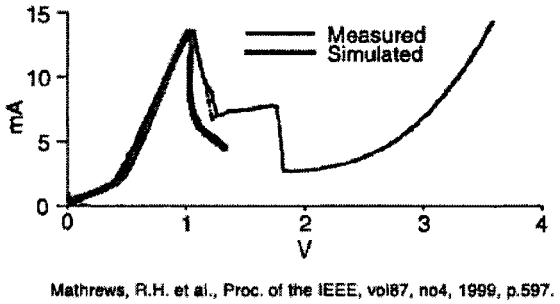


Figure 1.4 Courbe I-V d'une RTD

à haute vitesse.

2. On exploite la caractéristique de NDR pour former un dispositif constitué d'une paire de RTDs placées en série (fig.1.5) qui se comporte comme un bistable[2] appelée *RTDlatch* ou *gotopair*. Cette topologie ne requiert aucune

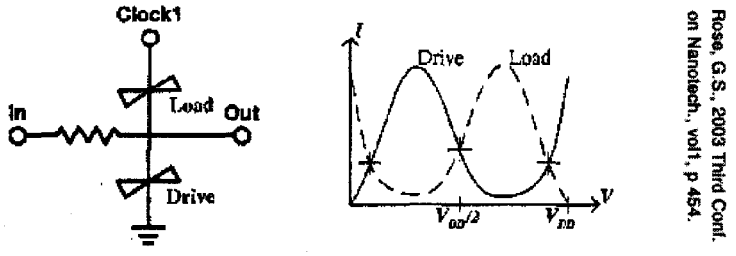


Figure 1.5 Une mémoire bistable à base de RTD

rétroaction comme pour un circuit CMOS et est plus compacte. Rose (Rose, 2003) propose une architecture de circuit mémoire en format FIFO exploitant des RTD moléculaires et n'ayant besoin d'aucun dispositif pour sélectionner les bits individuels dans la mémoire. Seul le dispositif moléculaire et des connexions sont nécessaires dans le noyau de cette architecture. Cette topologie

[2]Le terme bistable est employé ici comme un élément de mémoire logique, i.e. un latch.

facilite la régularisation géométrique et l'alignement des éléments mémoire par rapport au CMOS standard. On peut aussi envisager des topologies de mémoire en 3-D (fig.1.6), augmentant la densité et la performance des éléments mémoire.

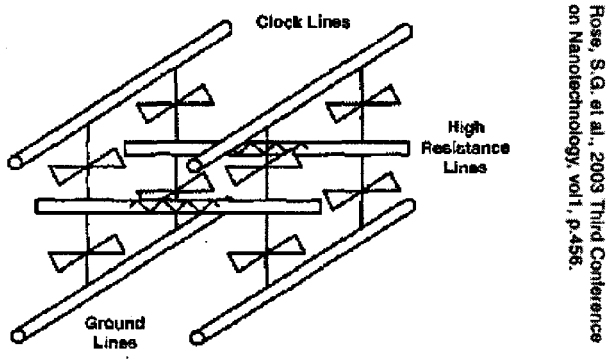


Figure 1.6 Topologie de mémoire FIFO-RTD en 3D

1.1.2.2 Spintronics

L'archivage et le traitement de données suit une tendance de miniaturisation depuis la première pierre gravée jusqu'aux éléments mémoire dynamiques, où un simple condensateur est utilisé pour conserver pendant un court laps de temps de l'information sous la forme de tension électrique. En suivant cette tendance, les nanostructures actuellement disponibles qui poussent le plus loin cette tendance sont les dispositifs spintroniques. C'est pour cette raison que nous nous sommes intéressés à ce champ de recherche qui recoupe plusieurs disciplines incluant la physique des matières condensées, l'étude des matériaux, l'optique et le génie électrique. Ce champ de recherche est divisé en deux approches:

1. Intégration d'éléments magnétiques dans un dispositif électronique pour créer

de nouvelles fonctionalités exploitant des propriétés uniques de matériaux ferromagnétiques. Cette approche se base sur le fait que le transport d'électrons de conduction dépend de leur spin pour intégrer un bit d'état magnétique à des dispositifs électroniques basés sur la charge. Parmi les applications de cette approche, on retrouve les têtes de lecture utilisant des senseurs de champs magnétique ou des éléments mémoire RAM magnétique.

2. Utilisation de populations d'électrons au spin polarisé, i.e. au moment magnétique individuel, pour créer de nouvelles méthodes de traitement de l'information telle l'informatique quantique. De cette façon il serait possible d'accélérer exponentiellement la vitesse de traitement de certains calculs parallélisables.

Si les applications exploitant la première approche sont fonctionelles et apportent des avantages concrets par rapport au CMOS, notamment au plan de la conception de mémoires, celles touchant la deuxième approche demeure inatteignable à court terme. La recherche dans ce domaine est actuellement stimulée par les résultats commercialisables à court terme de la première approche. Les mémoires magnétiques à accès aléatoires - MRAMs - offrent des avantages importants par rapport aux mémoires CMOS ou aux RAMs non-volatiles, les NVRAMs.
En effet, en utilisant le magnétisme plutôt que la charge électrique pour le stockage d'information, on peut créer des éléments mémoire non-volatiles. L'apport d'éléments spintroniques à ce domaine est qu'ils augmentent grandement la densité des éléments mémoire par rapport à des éléments mémoire actuellement utilisés comme les mémoires Flash.
Un premier prototype de MRAM à 128 kB a été présenté au "2003 Symposium on VLSI Circuits" (Bette, 2003) par une équipe de recherche regroupant *IBM* et *Infineon Technologies*. La figure 1.7 montre le dessin de masques de ce circuit CMOS 180 nm au centre duquel sont regroupés les éléments de mémoire. Ces éléments

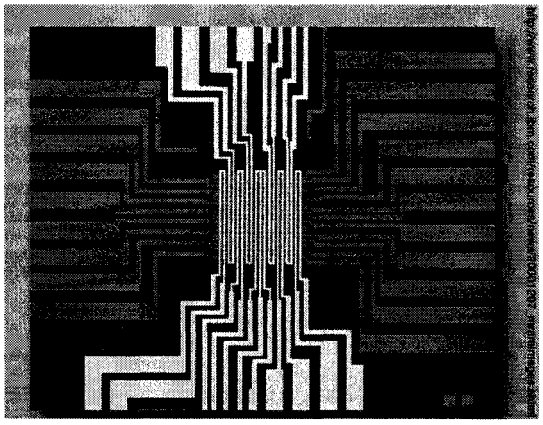


Figure 1.7 Dessin de masques d'une mémoire 128kB en MRAM

mémoire, les 1T1MTJ[3], occuppent chacun une superficie de $1.4\mu m^2$ (fig. 1.8).

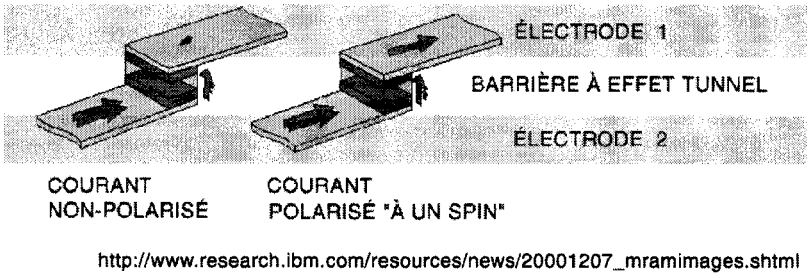


Figure 1.8 Schéma simplifié d'une cellule 1T1MTJ de IBM

Une équipe de recherche chez Motorola (Tehrani, 2003) a fabriqué une MRAM de 1-Mbit presqu'en même temps qu'IBM. Dans cette publication, Tehrani et al. décrivent le fonctionnement de leurs MRAMs à base de MTJs[4] comme étant des dispositifs qui changent de résistance selon leur état magnétique. La miniaturisation de ces dispositifs est rendue possible grâce à l'utilisation de matériaux

[3] 1-Transistor/1-Magnetic Tunnel Junction
[4] Magnetic Tunnel Junction

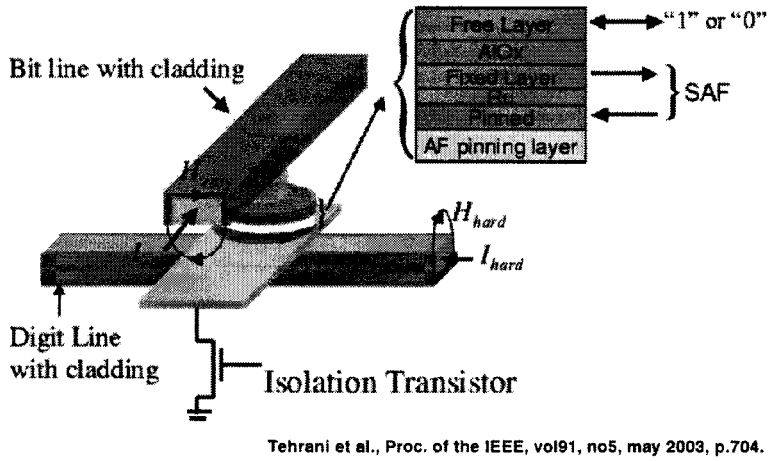


Figure 1.9 Schéma simplifié d'une cellule 1T1MTJ de Motorola

magnétorésistifs disposés en couches minces. La figure 1.10 illustre un modèle simplifié de densités d'états d'une MTJ. La MTJ est composée de deux couches magnétiques, FM1 et FM2, séparées par une barrière diélectrique.

Ce dispositif offre une résistance faible lorsque la magnétisation des couches

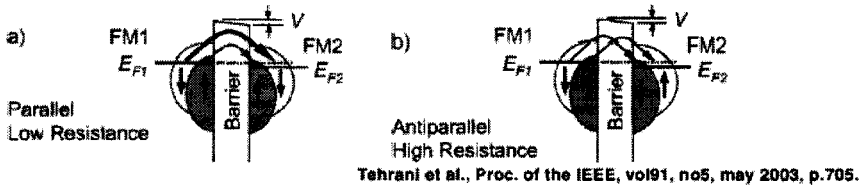


Figure 1.10 Fonctionnement interne d'une cellule MTJ

magnétiques est parallèle (fig 1.10a). Il offre au contraire une résistance élevée lorsque la magnétisation des couches externes est antiparallèle (fig 1.10b). Sur la figure, chacune des couches magnétiques est composée d'une bande minoritaire, représentée en gris et d'une bande majoritaire, représentée en blanc. Les flèches incluses dans ces couches montrent le sens de magnétisation. La flèche en gras

dans la figure 1.10a montre le fort courant d'électrons qui passent d'une bande magnétique à l'autre par effet tunnel quand la magnétisation des deux bandes externes est parallèle. L'intensité du courant provient des polarisations de spin dans les bandes magnétiques qui contribuent à rendre disponibles un plus grand nombre d'états de tunnel quantiques.

Les dispositifs les plus intéressants offrent un rapport courant ouvert/courant fermé élevé. Dans cette optique, on s'intéresse aux matériaux qui ont le plus grand "débalancement" entre les densités d'états des bandes minoritaire et majoritaire au niveau de Fermi. On peut définir ce déblancement en termes de polarisation de spin P:

$$P = (N\uparrow - N\downarrow)/(N\uparrow + N\downarrow)$$

où N↑ et N↓ représentent le nombre d'états spin-up et spin-down disponibles qui contribuent au courant de tunnel quantique. Rose et al. ont donc choisi du nickel, du fer et du cobalt, des matériaux ferromagnétiques à fort "débalancement" pour fabriquer leur MTJ.

En plus d'offrir une densité très intéressante, les cellules des MRAMs commutent en un temps inférieur à 5 ns. Des dispositifs à quelques MB sont attendus pour l'année 2005 sur le marché. Ces dispositifs pourraient provoquer une révolution entre autres dans le domaine de l'architecture des ordinateurs, rendant instantané le processus de "boot up". Les MRAMs ont un fort potentiel d'être la première nanotechnologie à avoir un impact concret sur le domaine de l'électronique.

1.1.3 Nanotubes de carbone

Le nanotube de carbone est une molécule dont le potentiel pour des applications nanoélectroniques est très tangible. Le grand nombre d'articles récents traitant du sujet en témoigne. La plateforme de test qui fait l'objet de ce mémoire vise d'abord la mesure de caractéristiques électriques de nanotubes de carbones. Nous examinerons cette molécule sous plusieurs angles sans en analyser les détails techniques. Nous suggérons au lecteur curieux d'en connaître plus de consulter l'imposante bibliographie de cette section.

1.1.3.1 Fullerènes

La découverte des nanotubes de carbone part de la recherche sur les fullerènes ou *buckyballs*. En 1985, Richard Smalley, de l'université Rice, découvre une molécule de carbone ressemblant à un dôme géodésique (fig.1.11). Cette molécule a suscité de

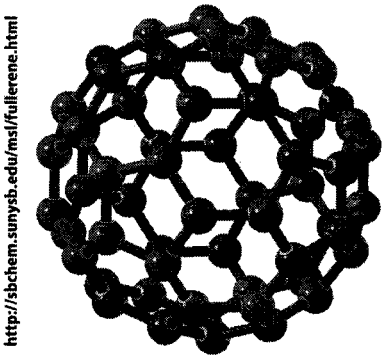


Figure 1.11 Une molécule de fullerène

grands espoirs d'applications pour former entre autres des agents d'administration de médicaments, des traceurs environnementaux et des supraconducteurs (Madou, 2002). L'équipe de Wudl (Wudl, 2002), à l'Université de Californie à Santa Bar-

bara, a réussi à stocker des molécules à l'intérieur de ces fullerènes pour en faire de véritables nano-contenants très stables.

Les applications des fullerènes en électronique sont rares, les nanotubes de carbone prenant une place beaucoup plus importante dans les recherches publiées. Une application intéressante qu'offrent les fullerènes exploite une propriété électromécanique surprenante de cette nanostructure (Joachim, 1998). Les auteurs ont placé le fullerène au bout d'une pointe d'AFM[5], (fig.1.12a). Ils ont relié ce montage à un système de lecture qui permet d' amplifier un signal généré par un effet piézoélectrique dans le fullerène. De cette façon, ils sont parvenus à mesurer des compressions mécaniques aussi faibles que 0.1 nm. Une compression aussi faible augmente l'intensité du courant mesuré de deux ordres de grandeur dans la molécule (fig.1.12b).

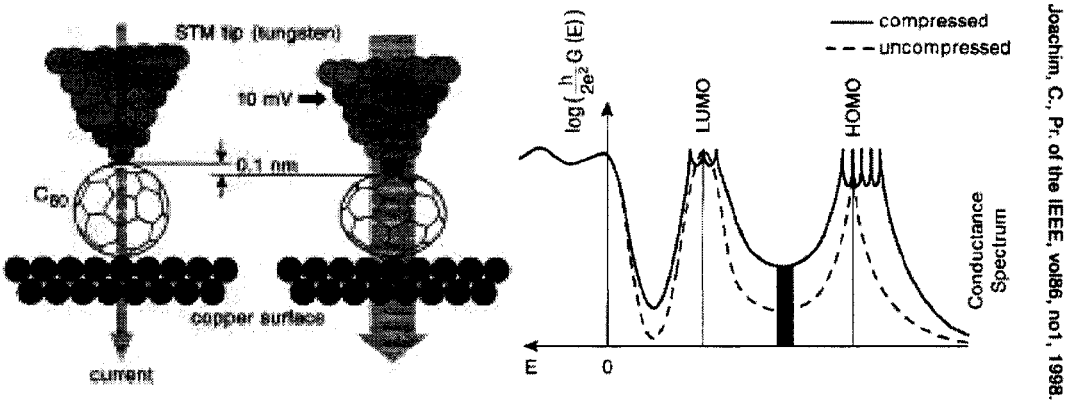


Figure 1.12 Senseur piézoélectrique en fullerène

1.1.3.2 Premiers nanotubes de carbone

En 1991, c'est en étudiant le processus de synthèse des fullerènes que Sumio Iijima (Iijima, 1991), chercheur chez NEC, Japon, observe le premier des nanotubes de

[5]Atomic Force Microscope

carbone. Leur structure hexagonale, symétrique comme les fullerènes, mais mince et très longue a ouvert la porte à une étude scientifique intense.

1.1.3.3 Structures moléculaires de nanotubes

La structure du nanotube de carbone peut être vue comme un enroulement d'une couche de graphite coupée par deux axes parallèles (fig.1.13) puis recollée pour former un long cylindre. La chiralité[6] du nanotube influence beaucoup ses propriétés électriques (Collins, 2000). En effet, certains nanotubes sont conducteurs et d'autres semiconducteurs. Les premier nanotubes de carbone observés par Iijima étaient composés de plusieurs couches concentriques de graphite. On appelle ces structures des "multi-walled carbon nanotubes" ou MWCNTs (fig.1.14b). Les "singled-walled" nanotubes - SWCNTs (fig.1.14a)- furent observés pour la première fois en 1993 par (Iijima, 1993) et (Bethune, 1993).

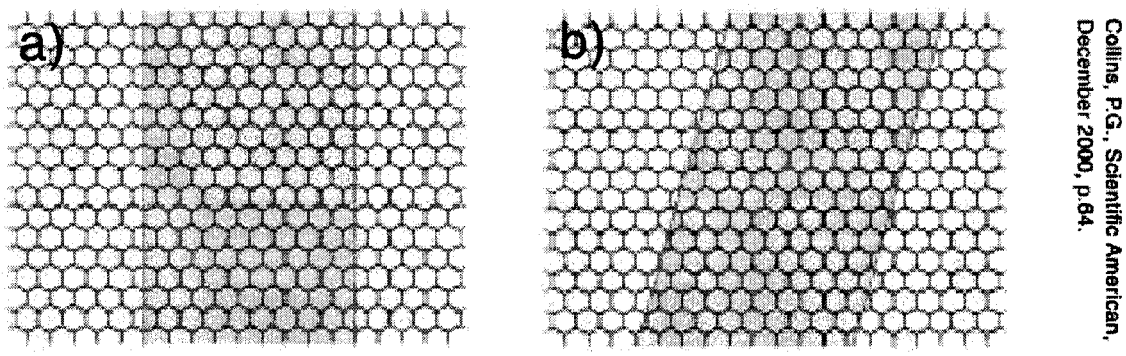


Figure 1.13 Coupes de plans de graphite

[6]L'angle auquel la coupe dans la feuille de graphite est effectuée

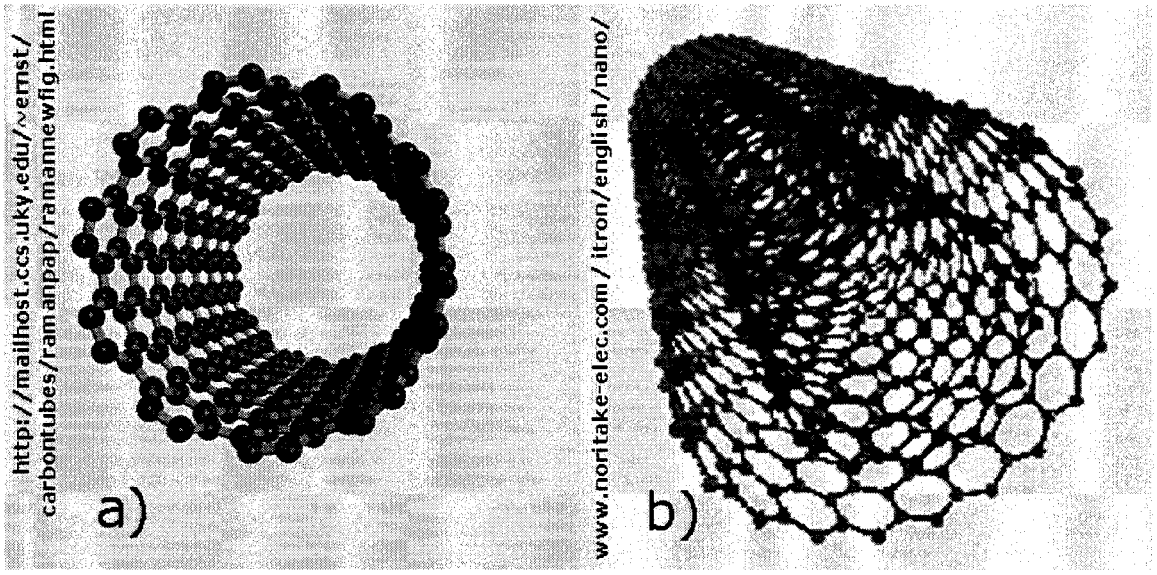


Figure 1.14 Nanotubes à simple et multi-couche

1.1.3.4 Propriétés électriques des SWCNTs

Les propriétés des nanotubes de carbone sont simplement phénoménales, même en les comparant à une technologie mature comme le CMOS. Collins (Collins, 2000) a regroupé cinq propriétés intéressantes pour des applications électroniques:

1. À cause des liaisons électroniques entre les atomes de carbone inhérentes à leur structure moléculaire, le transport électrique s'effectue en une seule dimension. Cela permet d'envisager que le transport dans les CNTs est ballistique[7]. Une conséquence de cette propriété est une réduction de la consommation de puissance par rapport à des conducteurs ohmiques comme le cuivre. Le comportement électrostatique des CNTs est différent de ce qu'on observe dans les dispositifs en silicium. Des champs électriques élevés sont atteignables avec de faibles tensions.

[7]Comme l'électron voyage dans un environnement à une seule dimension, la probabilité d'entraîner une collision entre 2 particules est faible. Dans les nanotubes de longueur inférieure à 1 μm, on estime qu'il n'y a aucune collision dans un CNT (Berger, 2001)

2. Tous les liens chimiques entre les atomes de carbone sont remplis. Cela annule le besoin pour une passivation de liens flottants comme dans les technologies à base de silicium. Une technologie à base de CNT pourrait n'avoir nul besoin de SiO_2 ou d'un autre diélectrique comme isolant. On pourrait le remplacer par une structure cristalline à forte constante diélectrique, ce qui permet, entre autres, de fabriquer des structures tridimensionelles.

3. Les liens covalents très forts des CNTs leur donnent une forte stablité mécanique et thermique ainsi qu'une protection contre l'électromigration[8]. Les CNTs peuvent soutenir des densités de courant atteignant $10^9 A/cm^2$ (Frank, 1998).

4. La dimension critique des CNTs, leur diamètre, est contrôlée par des procédés chimiques et non par des méthodes conventionnelles de fabrication.

5. En principe, il est possible de fabriquer des dispositifs actifs (transistors) et des interconnexions avec, respectivement, des CNTs semiconducteurs et métalliques.

1.1.3.5 Propriétés électriques des MWNTs

Les MWCNTs offrent des propriétés électriques moins impressionantes que les SWCNTs. Leur structure complexe peut être vue comme un assemblage concentrique de plusieurs SWCNTs ayant chacun des caractéristiques électriques et une chiralité différente. Par contre, il a été démontré qu'avec des conditions particulières (Bachtold, 1999), la conduction dans les MWCNTs est dominée par le

[8](Savaria, 1988) Électromigration, phénomène d'usure associé aux métallisations. L'électromigration est une conséquence du fait que le passage du courant d'un grain à l'autre tend à déplacer des atomes d'aluminium (ou autres conducteurs). L'électromigration est un obstacle à la miniaturisation de circuits, car la densité de courant augmente d'un facteur égal à la réduction d'échelle d'une technologie.

tube le plus en périphérie. Par contre, la largeur de la bande interdite étant inversement proportionnelle au diamètre du tube, seuls les plus petits MWCNTs pourraient fournir un rapport de commutation[9] satisfaisant à température de la pièce. Comme les SWCNTs sont plus petits que les plus petits MWCNTs, ils leur sont préférables pour fabriquer des dispositifs commutants comme des transistors.

1.1.3.6 Méthodes de fabrication de CNTs

Trois méthodes sont présentement connues pour produire des nanotubes de carbone avec un rendement satisfaisant:

1. Par choc électrique

   En 1992, Ebbesen et al. (Ebbesen, 1992) publient la première méthode pour fabriquer une grande quantié de nanotubes. On place deux tiges de graphite à une source électrique, on les place à quelques millimètres l'une de l'autre et on fait grimper le courant à 100 A. L'arc électrique créé vaporise le carbone dont une partie se recondense et forme des nanotubes (fig.1.15: gauche).

   - Rendement massique: 30%
   - Avantages: En ajoutant des catalyseurs métalliques, on obtient des MWNTs et des SWCNTs avec peu ou pas de défauts structurels.
   - Désavantages: Les tubes sont relativement courts ($< 50\ \mu$m) et se déposent à des positions et selon des orientations aléatoires.

2. Par déposition de vapeurs chimiques (CVD) En 1995, Endo et al. (Endo, 1995) placent un substrat dans un four à 600^oC et ajoutent lentement du

[9] rapport des courants quand le dispositif est ouvert et fermé

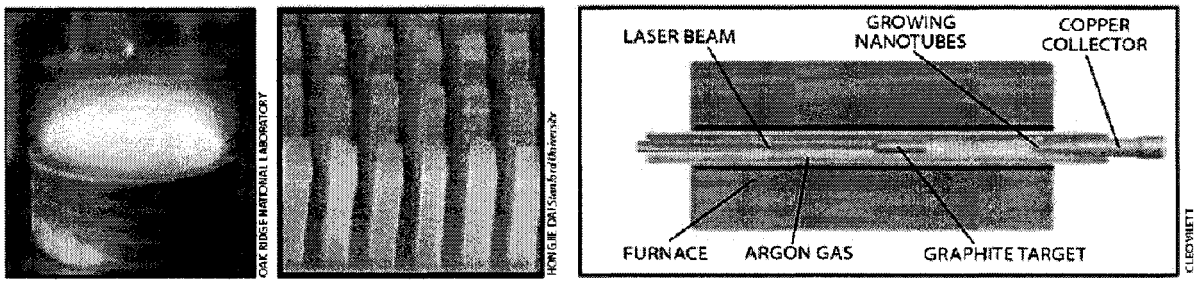


Figure 1.15 Procédés de fabrication de nanotubes

gaz porteur de carbone comme du méthane. Le gaz se décompose, libère des atomes de carbone, qui se recombinent et forment des nanotubes. Dai et al. (Dai, 1998) ont déposé un motif de catalyseurs sur le substrat (fig.1.15: centre) et travaillent actuellement à combiner cette croissance contrôlée avec des procédés CMOS standards. La très haute température du procédé de CVD rend par contre cet objectif très difficile à atteindre.

- Rendement massique: de 20 à 100%
- Avantages: La méthode CVD est la plus simple des trois méthodes pour fabriquer des dispositifs à plusieurs centaines de milliers de transistors comme le CMOS. Elle permet aussi de fabriquer des tubes très longs, condition nécessaire pour fabriquer des fibres utilisées dans des composites.
- Désavantages: La majorité des nanotubes fabriqués par CVD sont des MWNTs et peu sont exempts de défauts. Par conséquent, ces tubes ont une résistance mécanique en tension 10 fois plus faible que les tubes fabriqués à l'arc électrique.

3. Par impact LASER Smalley et al. (Scott, 2001), à l'Université Rice attaquent des tubes de graphite dans un four à $1200°C$ avec un LASER pulsé et forment ainsi les gaz porteurs de carbone à partir desquels les nanotubes se forment

(fig.1.15: droite). En variant les types de catalyseurs, ce groupe est parvenu à produire des quantités prodigieuses de SWCNTs.

- Rendement massique: inférieur à 70%
- Avantage: Production de SWCNTs au diamètre contrôlable par la température de réaction.
- Désavantages: Les lasers utilisés pour cette technique font exploser ses coûts de production.

1.1.3.7 Fabrication de CNTFETs

Les proprétés semiconductrices des CNTs poussent naturellement la recherche vers la fabrication de transistors à effet de champs à base de nanotubes de carbone ou CNTFETs. Les premiers CNTFETs furent fabriqués en 1998 (Martel, 1998; Tans, 1998). Le nanotube joue ici le rôle de canal de conduction entre deux électrodes métalliques: le drain et la source (fig.1.16). La grille, placée sous une couche de

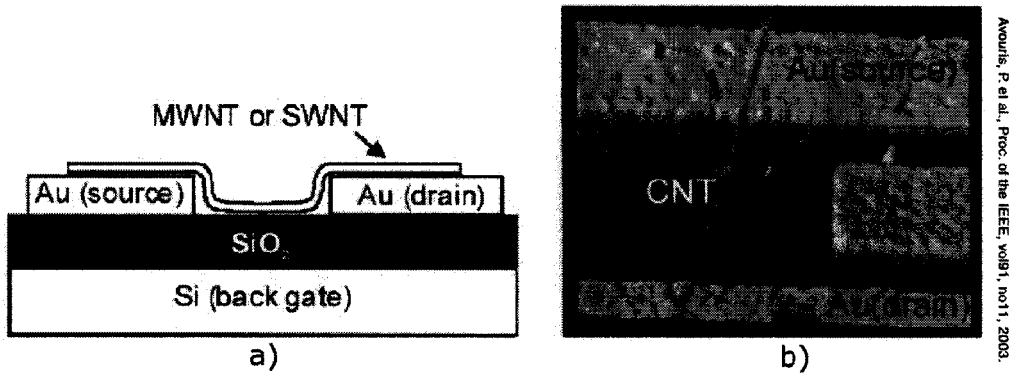


Figure 1.16 Un des premiers modèles de CNTFETs

SiO_2, est fabriquée en silicium hautement dopé. Les performances de ce premier transistor furent décevantes, notamment à cause des contacts entre le tube et les

électrodes. En effet, seules les forces de Van der Waal[10] retenaient le tube aux plaquettes d'or. En utilisant des électrodes en titane placées au dessus des nanotubes et soumises à un recuit thermique (fig.1.17), Martel et al.(Martel, 2001) ont amélioré les caractéristiques du transistor, notamment la transconductance.

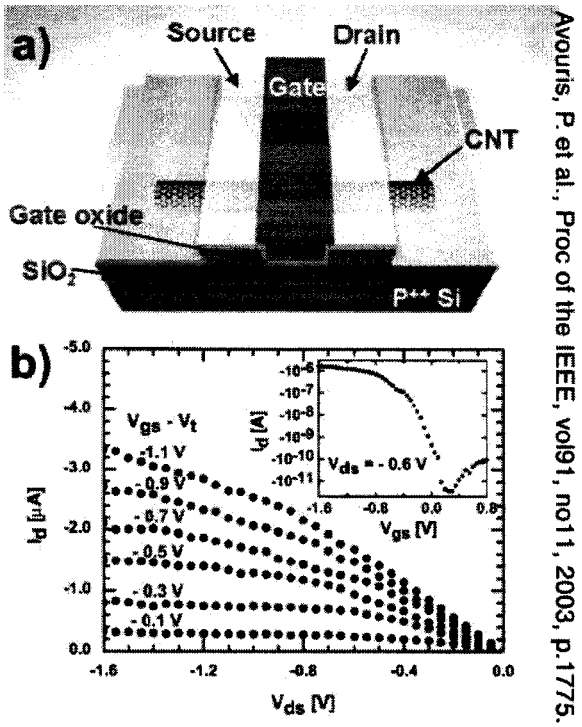


Figure 1.17 Modèle récent de CNTFET

1.1.3.8 Placement des CNTs sur une paire d'électrodes

Pour placer les CNTs sur les électrodes de contact, l'équipe du IBM research center effectue la gravure d'une matrice d'électrodes sur la surface de silicium. Ensuite, des nanotubes de carbone sont insérés en solution dans du dichloroéthane selon une densité contrôlée. Cette solution est versée sur la surface de test, puis on souffle

[10]On peut également dire une force d'attraction intermoléculaire entre le nanotube et la plaquette d'or. Cette force est suffisante pour garder le tube en place, mais ce lien n'offre pas un contact électrique adéquat.

le liquide avec un soufflet à gaz comprimé. Les forces de Van der Waal retiennent des nanotubes sur la surface. Ensuite, un AFM est utilisé pour déterminer la disposition des nanotubes. On se fie a la chance pour qu'un et un seul CNT aille se placer entre deux électrodes. Si tel est le cas, cette paire d'électrodes est reliée à des macroplaques qui serviront à lier les électrodes à des sondes extérieures . Les mesures sont effectuées par des machines de test externes.

Une alternative à cette méthode que l'on peut juger incertaine est d'utiliser des pointes d'AFM pour placer un nanotube à l'endroit désiré. Le problème avec cette approche est la difficulté de manipuler les CNTs. En effet, les tubes offrent une grande force d'attraction de Van der Waal, ce qui le rend "collant". Dans le cas qui nous concerne, il s'agit bien sûr de placer le CNT entre deux électrodes de contact, une cible très grande à l'échelle des nanotubes. C'est pourquoi certaines équipes de recherche arrivent à fabriquer des nanostructures avec cette méthode

En effet, l'équipe de Fukuda, Université de Nagoya, Japon (Dong, 2004), manipule des CNTs en trois dimensions pour caractériser leurs propriétés mécaniques. Avec une précision de 30 nm, cette méthode pourrait être utilisée pour placer des CNTs à l'endroit désiré avec une précision suffisante pour créer des CNTFETs. Jusqu'à ce jour, les publications de ce groupe ne font pas état de manipulation de SWNTs, seulement des MWNTs. De plus, le système de fabrication (fig.1.18) composé d'équipements nanorobotiques utilisé par cette équipe semble très coûteux et difficile à reproduire à court terme.

1.1.3.9 Tri des Nanotubes de Carbone

Les méthodes énoncées à la section 1.1.3.6 ont en commun le problème qu'elles créent un amas de nanotubes possédant toute une gamme de longueurs, de types et des défauts structurels. Les CNTs fabriqués par choc électrique sont recueillis dans une poudre de carbone contenant des impuretés qui doivent être ensuite

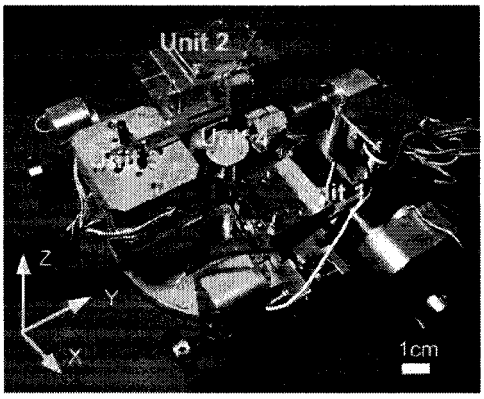


Figure 1.18 Système de nanofabrication

purifiées. La purification de ces nanotubes de carbone repose sur de nombreuses méthodes de filtration, centrifugation et des techniques de chromatographie pour séparer les nanotubes d'impuretés inhérentes à cette technique (Niyogi, 2000).
De plus, lorsqu'ils sont créés, les SWCNTs sont souvent regroupés en cordons, qui doivent être défaits pour que les SWCNT puissent être utilisés en tant que dispositif nanoélectronique. Les MWCNTs (sect 1.1.3.5) ont des propriétés électroniques moins intéressants que les SWNTs, mais en enlevant des couches aux MWNTs, il est possible de se rapprocher de dispositifs viables électroniquement.
Aucune méthode ne permet de contrôler la propriété de conduction des nanotubes lors de leur fabrication. Une approche intéressante consiste à utiliser un amalgame de CNTs de diverses formes comme matière première pour obtenir des CNTs de forme désirée. L'équipe du IBM Research Division a mis au point une méthode de séparation par claquage sélectif de MWNTs ou de cordons de SWCNTs (Collins, 2001) placés sur une paire d'électrodes. Comme la couche extérieure du MWCNT est la seule qui touche aux électrodes de stimulation, la majorité du courant injecté dans la nanostructure demeure dans cette couche. La figure 1.19a montre les échelons de courant injecté pour successivement brûler des couches de MWCNTs. Les figures 1.19b et 1.19c montrent les effeuillages partiels de MWCNTs résultants des stimulations de courant. Si l'on place un cordon de SWCNT sur

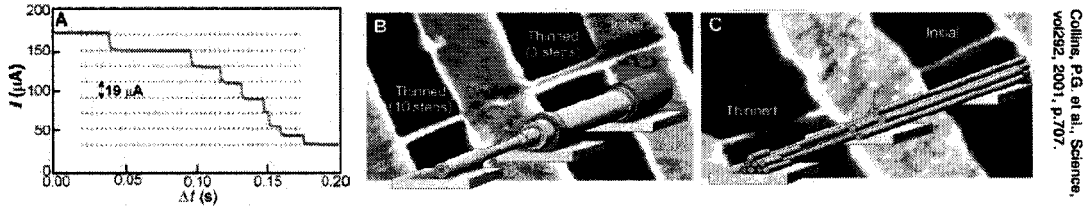


Figure 1.19 Méthode d'effeuillage de MWNTs

les électrodes plutôt qu'un MWCNT, on utilise une grille pour appauvrir le transport dans les CNTs semiconducteurs permettant ainsi d'injecter un fort courant dans les seuls SWCNT conducteurs. La figure 1.20A montre la conductance initiale du cordon de SWCNTs et la conductance finale sans SWCNTs métalliques. La figure 1.20B montre un cordon un peu plus gros et la figure 1.20C un cordon très large. Dans des cordes larges, certains SWCNTs semiconducteurs doivent être sacrifiés pour atteindre les conducteurs au centre du cordon. Les figures 1.20D et 1.20E montrent la structure à trois terminaux utilisée pour faire contact avec les dispositifs. Même si un cordon n'offre initalement aucun comportement semiconducteur 1.20F, les dispositifs purifiés ont de bonnes caractéristiques de CNTFET. L'équipe de l'Université Nagoya (sect 1.1.3.8) a démontré des méthodes de fabrication de CNTs à dimensions contrôlées en utilisant des méthodes mécaniques (Dong, 2004). Elle consiste à écraser le CNT entre une pointe d'AFM et un substrat (fig.1.21). Cette méthode permet plusieurs modifications de MWCNTs dont notamment l'effeuillage de couches du tube, qui n'est pas sans rappeler la méthode énoncée par (Collins, 2001). La précision atteinte par ce dernier est par contre nettement supérieure à la méthode mécanique de Dong et al. Des résultats futurs plus intéressants pourraient conduire à l'adoption des méthodes élaborées par ces chercheurs pour la fabrication de CNTFETs.

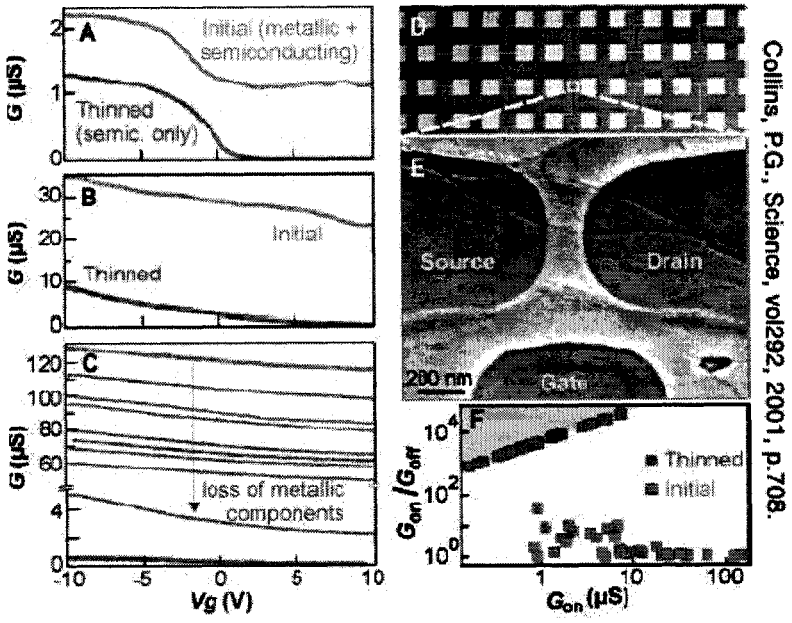
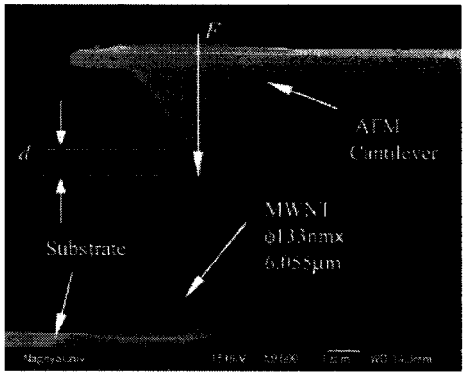


Figure 1.20 Mesures effectuées en cours de filtration

1.1.3.10 Exemple d'application: une porte logique en CNT

Lors de la fabrication de CNTFETs, les figures 1.16 et 1.17 illustrent qu'une seule grille était juxtaposée au nanotube. Ce dernier forme le canal de conduction du transistor. Il est toutefois possible d'utiliser plus d'une grille sur un seul nanotube pour former deux transistors - un n-FET et un p-FET - (fig.1.22 a et b) pour former un inverseur logique. En comparaison, les portes logiques en CMOS standard sont fabriquées avec des transistors ayant des canaux de conduction distincts. Cela cause un débalancement que des circuits logiques à base de CNTFETs pourraient réduire. Ce dispositif, fabriqué par l'équipe du IBM Research Division (Avouris, 2003), offre un gain en tension de près de 2 V/V (fig.1.22c). Ce dispositif n'ayant pas été l'objet d'une construction optimisant sa performance, on peut envisager l'apparition de circuits logiques à base de CNTFETs plus complexes et offrant une performance améliorée.



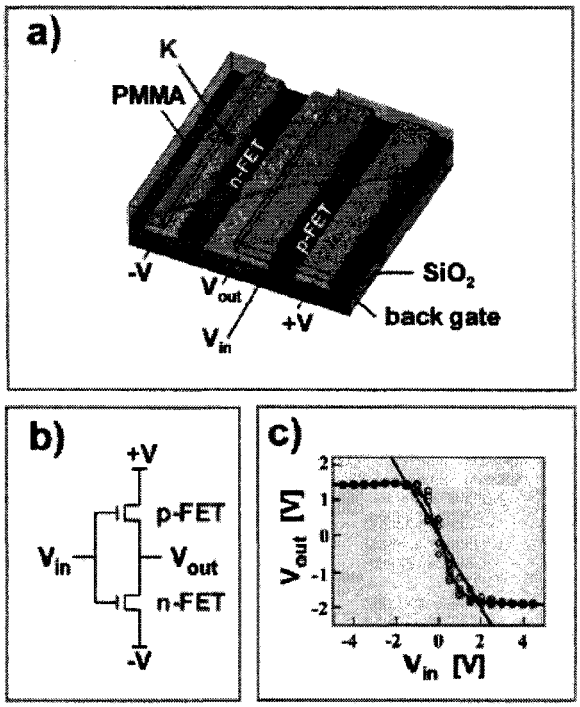
Dong, L. et al., IEEE Trans., on mechatronics vol9, no2, 2004, p.352.

Figure 1.21 Écrasement de MWCNT entre une pointe d'AFM et un substrat

1.1.3.11 Applications des nanotubes

Les technologies exploitant les nanotubes de carbone sont très jeunes, si on les compare au CMOS. Néanmoins, plusieurs idées sont présentement étudiées et certaines applications exploitent déjà des propriétés intéressantes des nanotubes de carbone.

- Les forts champs électriques générés par les nanotubes de carbone permettent de produire des écrans à émission de champs à basse consommation énergétique. Les méthodes de microfabrication sont ici plus simples que celles énoncées en 1.1.3.7 et 1.1.3.10, parce que les nanotubes sont utilisés dans un amalgame plus ou moins ordonné. Ren (Ren, 2003) a mis au point une pâte de nanotubes de carbone destinée à la conception de dispositifs à émission de champs (FED). Les compagnies Motorola, DuPont et Samsung intensifient la recherche pour des FEDs à base de CNTs et fabriquent actuellement leurs premiers prototypes d'écrans (Stuart, 2003). La consommation de puissance étant un obstacle majeur dans la fabrication d'écrans plasma, des disposi-



Avouris, P. et al., Proc. of the IEEE, vol91, no11, 2003, p.1781.

Figure 1.22 Un inverseur logique à base de CNTFETs

tifs moins énergivores ouvrent la porte d'un marché de quelques milliards de dollars.

- La conduction ballistique dans les nanotubes de carbone permet aux électrons de préserver leur spin quand ils se déplacent dans le tube (Collins, 2000). Des chercheurs tentent d'utiliser cette propriété dans des systèmes spintroniques qui commutent de 0 à 1 selon le spin de l'électron plutôt qu'en fonction de sa charge.

- Les CNTs, lorsqu'exposés à certains gaz comme le NH_3, changent de caractéristiques électriques. En effet, en variant la concentration de NH_3 autour de senseurs à base de CNTs, à température ambiante, (Moon, 2003) a mesuré des changements notables dans le courant électrique. Ong, (Ong, 2002) a mesuré des temps de réaction entre 45 s et 4 min face à des changements de concentration de O_2, NH_3 et CO_2 dans des senseurs de gaz à base

de CNTs. Ce sont la permittivité relative et la conductivité du CNT qui varient avec la concentration de gaz. Ces paramètres sont traduits en une inductance et une capacité qui sont mesurables avec des appareils de mesure précis. La figure 1.23 montre les mesures de permittivité relative $\epsilon_r^{'}$ et de conductivité $\propto \epsilon_r^{''}$[11] en fonction du temps pour des expositions successives au CO_2 et à l'air.

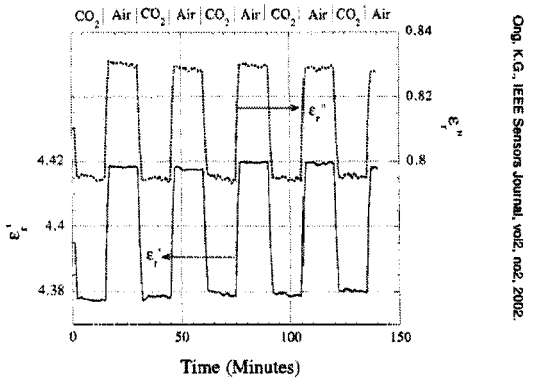


Figure 1.23 Mesures de permittivité relative et de $\propto$conductivité

- Les nanotubes de carbone sont aussi utilisés comme pointe pour des microscopes à force atomique. Les CNTs sont de 10 à 20 fois plus petits que les pointes de silicium les plus fines utilisées. Matsumoto (Matsumoto, 2001) est parvenu à faire croître des nanotubes directement sur la pointe en silicium de l'AFM (fig.1.24) par procédé CVD.

1.2 Mesures de précision

La seconde partie de cette revue de littérature consiste à brosser un tableau de développements récents se rapportant aux dispositifs de mesures à haute précision.

[11] Mesure directement proportionnelle à la conductivité

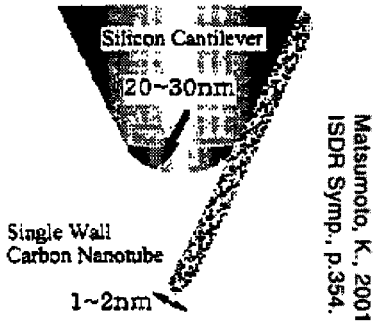


Figure 1.24 Une pointe d'AFM avec terminaison en CNT

Cette étude montre les bases théoriques justifiant les choix architecturaux pour la plateforme de test que nous présenterons au chapitre 2.

1.2.1 Mesure de délais

Le délai est défini comme étant le temps requis par un signal pour traverser un échantillon de test.

La mesure de délai est un sujet très vaste qui s'étend aux domaines des communications optiques et électriques, aux mesures de délais d'interconnexion "on-chip" et à des mesures expérimentales. Dans le cas qui nous intéresse, nous cherchons à mesurer le délai dans un nanotube de carbone ou dans un dispositif nanoélectronique à base de nanotube de carbone.

1.2.1.1 Télécommunications

La synchronisation de différents circuits électroniques est un problème intimement relié à la conception d'un réseau de communication. Les débits de données augmentant, le niveau de précision de la mesure des délais de propagation des signaux

doit être de plus en plus élevé.

Petre (Petre, 2002) fait état d'une méthode de mesure de délai qui permet de mesurer le temps requis pour transférer une donnée entre deux circuits intégrés. Pour ce problème, il est nécessaire de mesurer précisément, rapidement et avec un minimum de ressources, le délai dans quelques lignes à délai. Ces lignes à délai sont des circuits numériques configurables (fig. 1.25) qui offrent deux chemins à délais distincts. Chacun de ces chemins ne peut activer qu'un tampon de sortie à la fois. Ces lignes à délai sont utilisées pour générer des signaux de test desquels on mesurera ensuite le délai. La mesure de délai proprement dite est effectuée avec un système basé sur un oscillateur bouclé et un compteur (fig. 1.26). La technique de mesure consiste à mettre le compteur à zéro, puis à l'activer avec le signal "obs" qui signifie qu'on observe la fréquence d'horloge. Quand on remet le signal "obs" à zéro, le nombre d'oscillations dans la boucle est comptabilisé.

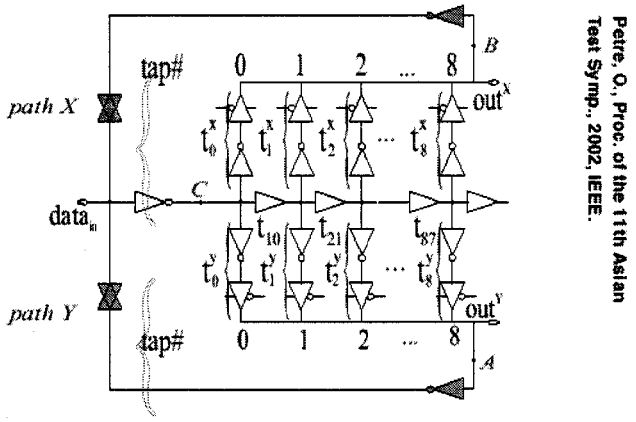


Figure 1.25 Ligne de délai pour la synchronisation

Cette approche est très intéressante pour le cas qui nous intéresse. En effet, la grande précision offerte par la répétition de la mesure permet d'accroître la performance des circuits électroniques numériques standard à une précision que même des circuits analogiques faits sur mesure et très performants ne peuvent atteindre. Par contre, en utilisant un multiplexeur standard, on inclut une différence

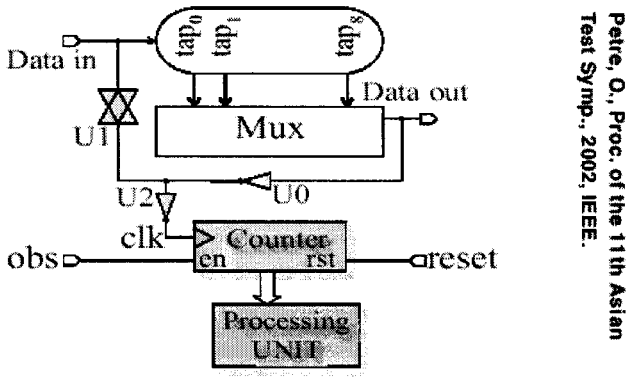


Figure 1.26 Système de mesure de délais de Petre et al.

de délai entre les chemins de propagation. Cette différence de délai, relativement imprévisible, demeure une source élevée d'incertitude sur la mesure finale. De plus, le fait que l'on mesure ici un temps absolu et non relatif à une référence fiable introduit de nombreuses sources d'incertitude. En effet l'auteur mentionne des sources d'erreur telles que la tempétrature et des variations de tension d'alimentation. Ces sources d'erreur variables diminuent la précision des mesures et forcent l'auteur à considérer plusieurs de ses variables en tant que variables aléatoires. On pourrait ajouter aux sources d'erreurs soulevées par Petre et al. les variations de tailles de transistors inhérentes aux imperfections que même les technique de fabrication de circuits intégrés les plus précises conservent. Nous utiliserons une partie des résultats des travaux de Petre et al. en améliorant la précision des mesures à la section 2.2.4.1.

1.2.1.2 VLSI

Lors de la conception de circuits intégrés dédiés, il est impensable d'obtenir une haute performance sans connaître précisément les délais de propagation propres aux composants utilisés. Les délais dans les circuits logiques MOS sont dus aux effets combinés des résistances et des capacités parasites (Savaria, 1988). On peut séparer

les sources de ces délais en deux groupes: les transistors et les interconnexions.

Soumyanath (Soumyanath, 1999) mesure le délai dans le domaine du temps entre des interconnexions dans un circuit intégré selon une technique basée dans le domaine du temps. Le circuit intégré contient plusieurs matrices de test de paires d'interconnexions (fig. 1.28) dont on veut mesurer le comportement. Un signal de test est propagé dans ces structures, puis échantillonné à l'extérieur du circuit. Un montage très soigné est requis pour réduire les sources d'erreur: des plots à faible inductance, un amplificateur à 20 GHz et d'autres composants précis.

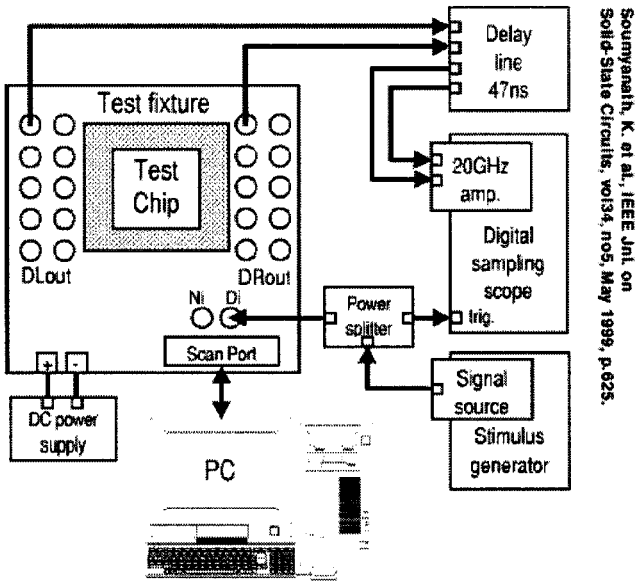


Figure 1.27 Montage de mesure de délai

L'intérêt de cette technique par rapport à celle de Petre est que la mesure est ici analogique, ce qui permet d'obtenir plus d'information sur le signal que le délai entre deux interconnexions. En effet, on peut mesurer le bruit induit par le passage d'un signal près de l'interconnexion que l'on mesure. Le fait d'utiliser une mesure extérieure au circuit réduit néanmoins la performance de la mesure de délai par

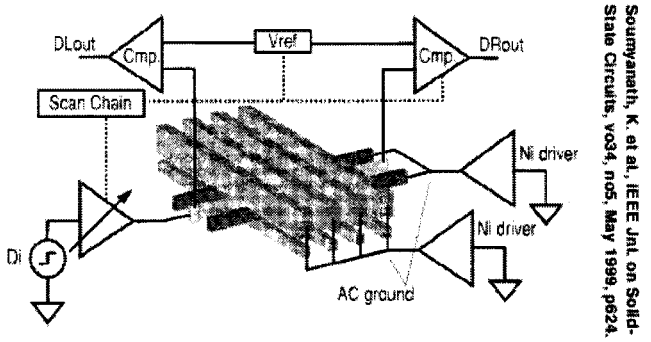


Figure 1.28 Configuration de mesure de délai d'interconnexions

la faiblesse relative de la largeur de bande des plots d'entrée/sortie de tout circuit intégré. Ceci est vrai même si le circuit a des plots à faible inductance.
Les auteurs sont parvenus à mesurer des délais de l'ordre de 15 ps avec cette technique. Le montage de mesure qu'ils ont utilisé (fig. 1.28) pour ce faire est hautement spécialisé aux mesures d'interconnexions dans le circuit de test. La méthode de mesure de délai est intrinsèquement liée à la forme des matrices de mesure d'interconnexions (fig. 1.28) et non à des mesures de dispositifs sous test à deux ou à trois broches. De plus, comme les mesures de bruit ne sont pas dans le spectrre d'intérêt de ce projet, la technique proposée par Soumyanath et al. ne sera pas utilisée.

Pera (Pera, 1997) montre deux techniques de mesure de délai dont l'une est semblable à celle de Petre (Petre, 2002) pour mesurer le délai entre des interconnexions avec une précision atteignant 100 ps. Cette publication, parue il y a 7 ans, avec une technologie de 1.5μm, moins performante en termes de temps de commutation que celle possible avec des technologies actuelles comme le CMOS 130nm. Il est donc permis de croire qu'une réactualisation des circuits de Pera à une technologie plus récente pourrait améliorer cette performance. L'auteur utilise un oscillateur bouclé (fig. 1.29) dont la boucle contient un échantillon sous test. La fréquence d'oscillation de la boucle est échantillonée et ralentie par des diviseurs de fréquence

et ensuite expédiée à l'extérieur du circuit intégré.

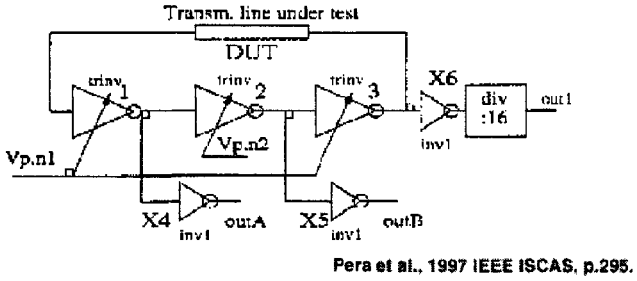


Figure 1.29 Oscillateur bouclé de Pera et al.

Ralentir ainsi la fréquence d'oscillation de la boucle est un bon moyen de contourner les capacités parasites des plots de sortie, mais cela réduit la précision offerte par la mesure. Ce choix de design permet aussi de procéder à des mesures de délais sans consommer l'espace requis par un module intégré de mesure de délai.

En ce qui concerne notre plateforme de mesure, il est plus intéressant d'effectuer les mesures sur le circuit qu'à l'extérieur. En effet, seule la précision des mesures nous importe.

La seconde méthode utilisée par Pera pour mesurer des délais est un détecteur de phase appliqué à la mesure de délai (fig. 1.30). Ce module sert pour mesurer les délais de propagation propres aux montées ou descentes de tension. À la figure 1.30, on peut observer les sorties out_P et out_N qui représentent respectivement les temps de montée et de descente du signal.

De tels détails de mesures sont certes très intéressants. Par contre, les caractéristiques électroniques des nanostructures sous test étant très peu connues à l'heure actuelle, il est préférable de ne pas effectuer des mesures que l'on peut juger un peu trop pointues pour le moment. La mesure des temps de montée ou de descente du signal pourrait constituer un ajout intéressant à une version prochaine la plateforme de test.

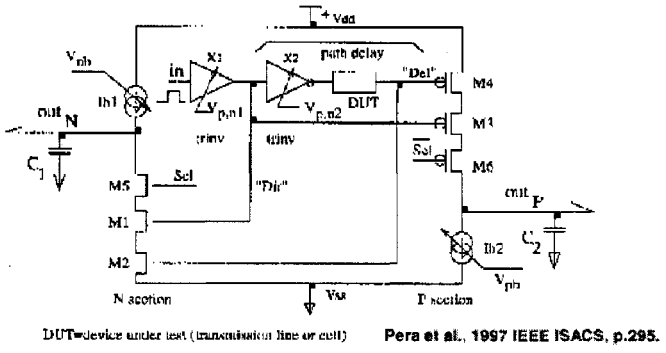


Figure 1.30 Détecteur de phase de Pera et al.

1.2.1.3 Autres sources sur des mesures intégrées à haute précision

Plusieurs autres articles ont été étudiés lors de la conception des circuits constituant la plateforme de test. Les articles présentés plus tôt dans cette section en ont beaucoup influencé la conception. Nous présentons ici d'autres publications qui, après une lecture approfondie, n'ont pas été retenues comme des avenues intéressantes pour le problème qui nous intéresse. Il est toutefois important de nous positionner par rapport à ces résultats de recherche touchant des domaines connexes au nôtre.

Brock (Brock, 1995) présente un système de mesure de délais de portes logiques offrant une précision sous la picoseconde. Cette performance est très impressionante pour un circuit datant d'une dizaine d'années. Malheureusement, les portes logiques dont ce système mesure le délai aussi finement sont fabriquées en technologie RFSQ[12]. Cette technologie, basée sur des jonctions de Josephson, tire avantage des délais réduits que l'on observe lorsqu'un canal est utilisé en supraconduction. Évidemment, les moyens techniques énormes nécéssités par ce genre de montage ne nous sont pas accessibles. Il nous sera donc impossible de recréer les conditions préalables à la mesure de délais aussi fins que ceux présents dans cet article.
Par contre, on remarque que pour atteindre une précision de mesure élevée, les au-

[12]Rapid Flux Single Quantum

teurs ont procédé à une calibration très poussée du système de mesure. Les efforts que les auteurs ont mis à parfaire cette technique sont justifiables par le fait que la précision de la mesure dépend presqu'exclusivement de cette calibration. La lecture de cet article confirme nos présomptions que la calibration est une considération de conception très importante d'un instrument de mesure de haute précision.

Laquai (Laquai, 1995) montre une structure de test pour la mesure de signaux internes dans des circuits VLSI CMOS. Cet instrument est composé de sondes qui échantillonent des signaux analogiques à l'intérieur de circuits intégrés. La mesure, effectuée par exemple sur un noeud d'intérêt dans un circuit intégré, échantillone le comportement à ce noeud à plusieurs pendant plusieurs oscillations et retourne un courant DC. La valeur de ce courant DC est proportionnelle à la valeur maximale de courant des échantillons mesurés. Si le signal mesuré respecte plusieurs conditions dont la périodicité et une phase constante, ce courant permet d'extraire beaucoup d'information du signal d'entrée.
Cette méthode est certes intéressante et a permis de mesurer des délais de propagation de l'ordre de 200ps en utilisant une technologie CMOS 0.5 μm. Nous recherchons par contre des résultats plus précis. De plus, cette méthode préconise l'utilisation d'une source contrôlée comme référence pour la mesure de délai. Bien entendu, cela est nécessaire pour montrer la performance du circuit à un moment donné, mais, comme cette mesure est directe plutôt que relative, elle ne permet pas de réduire plusieurs sources d'imprécision variables comme par exemple la température.

Sato (Sato, 2001) suggère une méthode de mesure intégrée de délai et de bruit. Le circuit, qui a été fabriqué avec une technologie CMOS 0.35 μm, permet d'en caractériser les interconnexions. L'analyse de signal que propose cet auteur passe par l'analyse du bruit et à partir de l'information que cela procure, il examine l'effet du bruit sur le délai de propagation. Une partie importante de ce papier traite donc

de la méthode de mesure de hauteur de pic de bruit qui permet de mesurer le bruit de diaphonie [13].

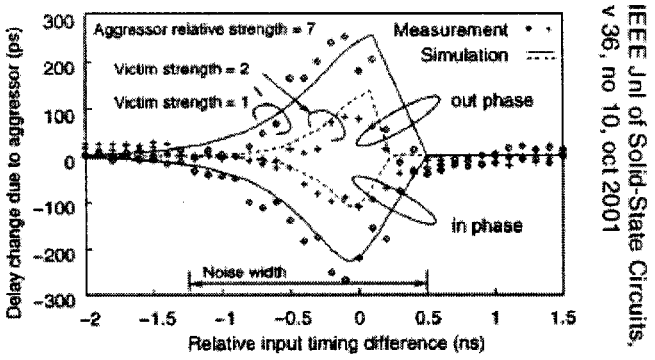


Figure 1.31 Mesure de délai de Sato et al.

La partie qui traite des modifications de délai causées par le bruit ne montre pas la précision des mesures effectuées, mais montre une mesure (fig. 1.31) située entre -250 ps et 250 ps. Ces mesures sont très intéressantes en ce qui concerne le bruit de diaphonie, mais comme nous avons visé la mesure de délais, cet article n'est pas directement pertinent pour nous. Une version subséquente de la plateforme qui pourrait mesurer les bruits de diaphonie entre deux lignes fabriquées en nano-composants, par exemple, bénéficierait certainement de l'information contenue dans ce papier.

[13] cross-talk

CHAPITRE 2

PLATEFORME DE TEST

Le premier chapitre permettait de montrer des méthodes de mesure et des nanostructures utiles pour la fabrication d'un circuit nanoélectronique. Comme mentionné en introduction de ce mémoire, l'objectif primaire de ce projet est d'ouvrir la voie à l'amélioration de circuits intégrés commercialement viables en utilisant des dispositifs nanoélectroniques novateurs. Nous avons vu qu'il existe actuellement certaines méthodes de fabrication et de mesure de ces dispositifs. Il reste par contre beaucoup de chemin à faire avant de pouvoir remplacer les technologies actuellement dominées par des dispositifs à base de silicium par ces nouvelles nanostructures.

C'est dans cette optique que la plateforme de mesure devient un choix intéressant. En effet, étant donné que plusieurs caractéristiques électroniques de ces nanostructures sont actuellement inconnues, il est difficile de modéliser leur comportement adéquatement. Cette plateforme permettra notamment de mesurer le comportement de ces nanostructures soumises à des signaux à haute fréquence.

Ce chapitre traite des détails de design de cette plateforme et des raisons qui nous ont amenées à choisir cette architecture. L'architecture de la plateforme dépend en grande partie du choix des caractéristiques à mesurer et de la précision envisagée (section 2.1). La plateforme repose sur un système électronique standard (section 2.2), utilisé pour effectuer les mesures proprement dites et offrir une interface de contrôle à l'usager de la plateforme. Finalement, les nanostructures sous test sont directement intégrées à la plateforme selon des méthodes expliquées à la section 2.3.

2.1 Mesures à effectuer

La plateforme de test vise la mesure de caractéristiques électroniques de nanostructures. Dans cette section, nous expliquons en détail chacune de ces caractéristiques et le niveau de précision recherché pour ces mesures.

2.1.1 Mesure I-V

La courbe I-V à courant continu (DC) fournit l'information nécessaire pour connaître l'impédance d'un dispositif électronique. Cette mesure est souvent utilisée pour évaluer les zones de conduction d'un transistor par rapport à une tension de grille et à la polarisation de ses électrodes[1]. À titre d'exemple, on peut observer la courbe I-V d'un fil conducteur de 1kΩ et celle d'un transistor CMOS standard à la figure 2.1. Dans notre cas, nous stimulons la nanostructure sous test avec une source de courant et nous mesurons la tension drain-source correspondante.
Lorsqu'une équipe de chercheurs réussit à fabriquer un nouveau dispositif nanoélectronique, la mesure de la courbe I-V est communément utilisée dans les publications. Par conséquent, cette mesure peut aisément servir de base de comparaison entre plusieurs dispositifs nanoélectroniques. Puisque la performance du dispositif dépend souvent de l'habileté des chercheurs à manipuler des structures à l'échelle nanométrique et même d'une certaine dose de chance, il est important de se doter de telles bases de comparaison.

[1]Grille, drain, source et substrat

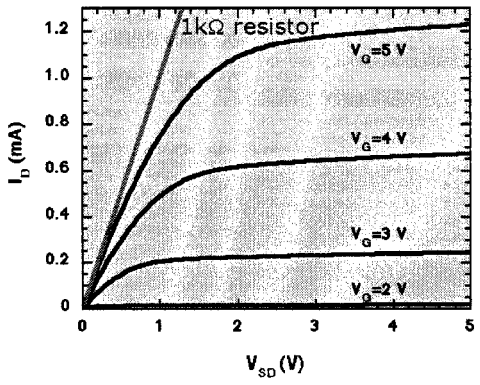


Figure 2.1 courbes I-V d'un transistor CMOS et d'un fil de cuivre

2.1.1.1 Précision et modèle recherchés

Les caractéristiques électriques des nanostructures sous test n'étant pas entièrement connues à ce jour, la mesure des caractéristique I-V doit supporter un large éventail de courants et de tensions. Comme nous utilisons un circuit CMOS 180nm, la tension d'opération maximale est relativement faible. Les tensions d'opération normales sont de 1.5 V dans notre cas. Il est toutefois possible d'augmenter ces tensions jusqu'à 3V sans risquer d'endommager les circuits. Il serait bien entendu souhaitable d'avoir une plus grande plage de tension de mesure, mais des mesures de tension effectuées sur des nanotubes de carbone ont démontré que la plage de tension drain-source typique demeure à l'intérieur de ces limites.

Comme les nanotubes de carbone peuvent soutenir des densités de courant de l'ordre de $10^9 A/cm^2$ (sect 1.1.3.4), nous cherchons à stimuler une grande plage de courants. Nous estimons le courant de stimulation maximal requis avec l'inéquation suivante:

$$\frac{Courant_{MAX}}{Section_{TUBE}} < 10^9 A/cm^2$$

Où la $Section_{TUBE}$ minimale est de $3.14 * 10^{-14}$ cm^2.

Nous devrions donc nous limiter à des courants de stimulation sous la barre des 31μA. Nous avons fixé la limite inférieure de courant de stimulation à 100 pA, car à des niveaux plus bas, il devient difficile de garantir une précision suffisante.
Les lectures effectuées avec le module DC doivent conserver une précision de l'ordre de 0.1%. C'est à dire que pour l'échelle de mesure de 0 à 1.5V choisie, la précision est de l'ordre du milliVolt.

2.1.2 Mesure de délais

Le délai est défini comme le temps requis par un signal pour traverser un échantillon de test. La réduction du délai de propagation dans des dispositifs actifs intégrés se traduit généralement par une augmentation de la performance des circuits qui les utilisent. Comme nous l'avons montré au premier chapitre, plusieurs méthodes sont actuellement utilisées pour mesurer le délai de propagation à l'intérieur de dispositifs microélectroniques. À la sous-section suivante, nous situons la précision que nous visons pour ces mesures.

2.1.2.1 Précision recherchée

Dans le cas des nanotubes de carbone, le délai de propagation est actuellement inconnu, mais la propriété de conduction ballistique (sect. 1.1.3.4) des CNTs permet de croire que ce délai est très faible. Nous visons donc à pousser la performance des circuits intégrés existants afin de mesurer des délais très courts avec précision (sect.1.2.1), concrètement nous visons à mesurer des délais de l'ordre de 1 ps.
Les nanostructures et les électrodes de contact sont modélisées comme étant des boîtes noires qui causent des délais sur un signal qui les parcourent sans causer de

perturbations suffisament significatives pour empêcher des mesures valides. Nous traiterons plus en détails des modèles utilisés pour la mesure de délais au chapitre 3.

2.2 Système électronique de mesure et contrôle

Le système électronique comprend toutes les fonctionnalités nécessaires pour le contrôle et pour effectuer les tests et mesures sur les nanostructures intégrées. La figure 2.2 offre une vue schématique de la fonctionalité globale du circuit: le circuit comprend trois parties: les modules I-V, les modules qui mesurent des délais et les modules d'autotest. L'accès à ces modules de test est effectué avec un module JTAG qui est décrit à la section 2.2.3.

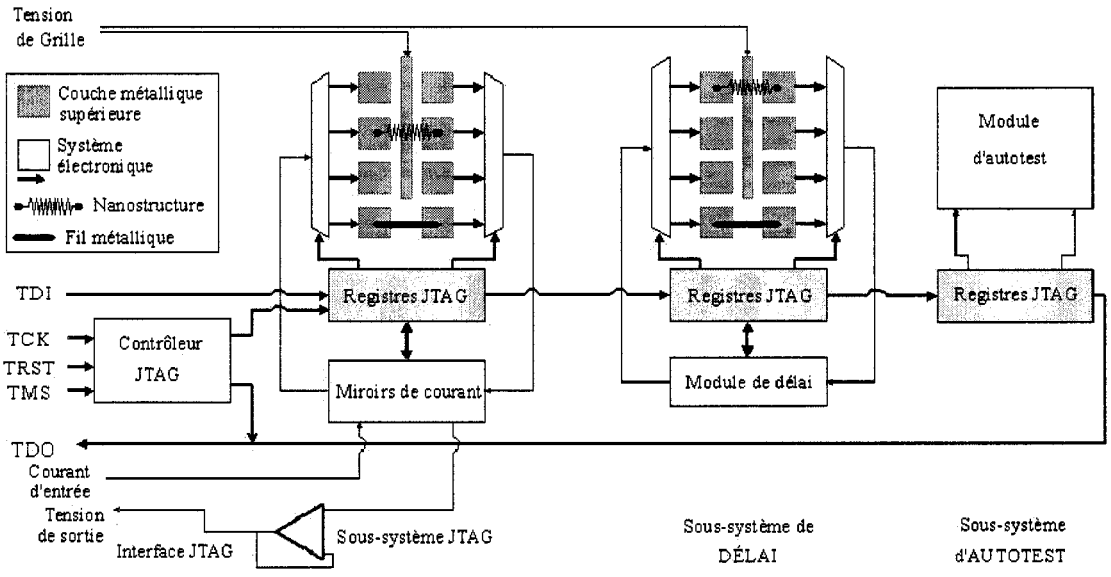


Figure 2.2 Système électronique global

2.2.1 Module de mesure DC

Le module de mesure DC est composé en grande partie d'un ensemble de miroirs de courant en cascode, chacun de ces miroirs étant connecté à une paire d'électrodes. Le miroir de courant prend en entrée un courant provenant de sources extérieures et offre un courant stable et contrôlé en sortie. Les copies du courant d'entrée peuvent être appliquées à des nanostructures sous test. La caractéristique I-V_{DS} est obtenue en mesurant la tension entre les électrodes dans la plage de tension de 0 à 1.5 V permise par la technologie CMOS 180 nm. L'électrode de grille est appliquée à tous les dispositifs à partir d'une broche d'entrée /sortie. Puisqu'aucun composant actif ne contrôle cette tension, une plus grande plage de tension est admissible: typiquement de -6 à 6 V.

La sortie des miroirs de courant est ensuite reliée à un amplificateur opérationnel interne branché en configuration "suiveur". Cet amplificateur, dont la sortie est branchée à la broche de sortie de tension est illustré à la figure 2.2. Son objectif est d'adapter l'impédance de la structure de test intégrée avec celle du long fil reliant la nanostructure sous test et l'équipement de mesure externe.

On utilise une source de courant externe au lieu d'une source intégrée même si cela requiert plus de broches pour des courants de contrôle. Cette approche offre une plage dynamique de courants de test très importante. Des sources de courant intégrées au circuit consommeraient également plus d'espace et du temps de conception. Il y a donc des parties du design pour lesquelles l'utilisation d'equipements externes performants est préférable à essayer de répliquer leur fonctionnalité sur le circuit.

Un avantage important de l'architecture intégrée par rapport à l'utilisation de sondes (sect 1.1.3.8) est la protection contre des décharges électrostatiques offertes par les broches analogiques d'entrée de courant. Come une seule paire d'électrodes peut être sélectionnée à la fois, la nanostructure sous test est sélectionnée en utilisant

des portes de transmission placées entre VDD et les dispositifs sous test (fig. 2.3). Dans la version actuelle de la plateforme, 15 paires d'électrodes sont disponibles pour l'usager. Une 16^{ieme} paire reliée à une connexion métallique est utilisée pour des fins de calibration. La figure 2.3 montre aussi une paire d'électrodes du module de mesure DC.

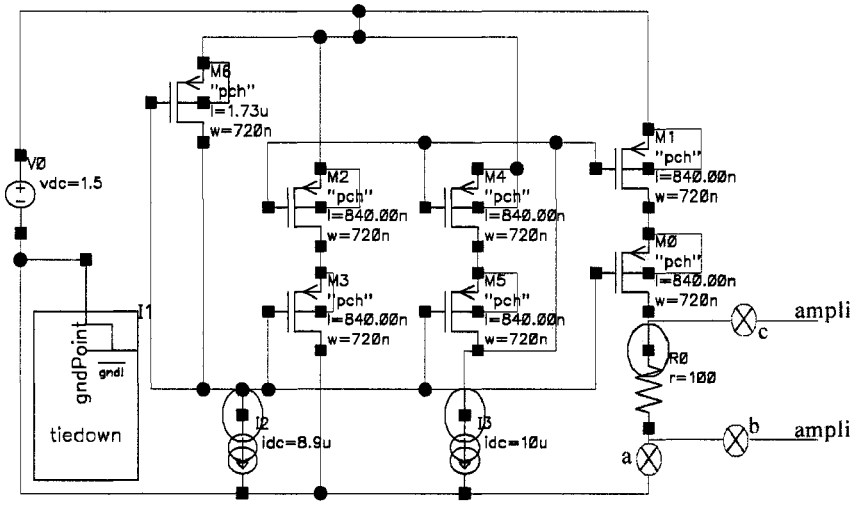


Figure 2.3 Miroir de courant du module de mesure DC

Ce circuit a deux modes d'utilisation: mesure de tension et mesure de courant. Le mode "mesure de courant" est accessible en activant les portes de transmissions "a" et "c" d'une branche du module. Ainsi, le courant soumis à l'entrée est transmis dans l'échantillon sous test de cette branche et c'est la chute de tension qui est lue à la sortie sur la branche "c". Bien entendu, seule une branche peut être sélectionnée à la fois, ce qui bloque l'accès en écriture aux branches non-sollicitées. Un amplificateur opérationnel intégré est utilisé pour offrir une adaptation d'impédance entre le circuit interne et le monde extérieur. Cet amplificateur occupe une surface énorme sur le circuit et offre une performance amplement suffisante pour nos besoins. Nous avons simulé le fonctionnement du module de mesure de délai entier et nos résultats sont disponibles au chapitre 4.

En ce qui concerne la deuxième méthode de mesure du circuit, elle permet plutôt de relier 2 électrodes de test à des plots de sortie du circuit. Pour ce faire, on active les portes b et c d'une branche du module. Encore une fois, une seule branche peut être utilisée à la fois. Nous verrons des utilisations de ces méthodes de mesure au chapitre 4 qui traite des simulations que nous avons effectuées lors de la conception de ce circuit.
Le miroir de courant est constitué de transistors disposés en cascode dont les tailles sont affichées à la figure 4.9.

Cette topologie ne permet pas une très grande chute de tension dans les échantillons testés. Comme notre circuit est dédié à la mesure I-V, cela n'est pas un problème, car on contrôle le courant d'entrée, ce qui nous permet de l'ajuster pour réduire la chute de tension occasionée par l'échantillon sous test.

2.2.1.1 Porte de tansmission utilisée dans le module de mesure DC

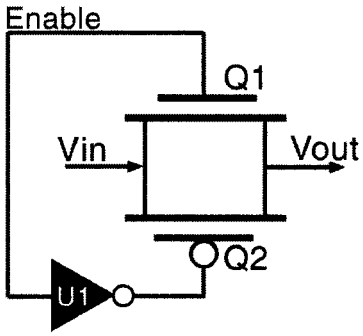


Figure 2.4 Porte de transmission utilisée dans le module

Les portes de transmission que nous avons utilisé dans ce module ne proviennent pas de bibliothèques de cellules. Nous avons déterminé la taille optimale des transistors qui la composent pour conserver le signal qui s'y propage le plus intact possible. Aussi, à la figure 2.4, on voit les transistors Q1 et Q2 qui ont tous deux une

longueur de 720 nm. Le canal de Q2 a une largeur de 3.82 μm alors que celui de Q1 est de 720 nm. C'est en effectuant des simulations avec le logiciel Cadence que nous sommes arrivés à ces tailles qui nous semblent optimales. Ces simulations consistent à fixer la taille de Q2 et à faire varier la largeur de Q1 de 180 nm à 18 μm dans le but d'offrir une fonction de transfert la plus proche possible de "1". La porte U1 est un inverseur de taille minimale "INVD0".

2.2.2 Module de validation d'intégrité technologique

Le module d'autotest permet de mesurer d'éventuels dommages causés aux circuits électroniques que constitue la plateforme par des manipulations telle que la lithographie utilisée pour créer les nano-dispositifs. L'impact que ces manipulations auront sur les métaux et les composants actifs situés sous la surface du circuit est inconnu. Par exemple, des problèmes pourraient émerger dans les composants actifs suite à une exposition à une température excessive et il est important d'en mesurer l'impact sur la fonctionnalité de la plateforme. Le module d'autotest est constitué d'un réseau d'inverseurs placés sous la région des électrodes de test (fig.2.5). La dégradation de couches de métal ou de régions actives occasionerait

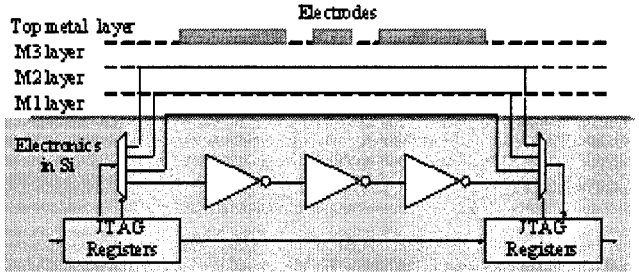


Figure 2.5 Le module d'autotest

un bris de la chaîne d'inverseurs. Il est donc possible de mesurer plusieurs points de mesure indépendamment. Chacun des points de test a été placé à une position

jugée critique (fig.2.6). Les critères de sélection de ces endroits sont déterminés par la proximité des zones où les électrodes ont été déployées. Ces zones sont regroupées en deux groupes distincts: sous des fenêtres dans la couche de passivation ou non. Si des bris localisés sont occasionnés par des manipulations du circuit, il sera possible, grâce aux sondes d'autotest, de connaître l'emplacement précis de ces bris.

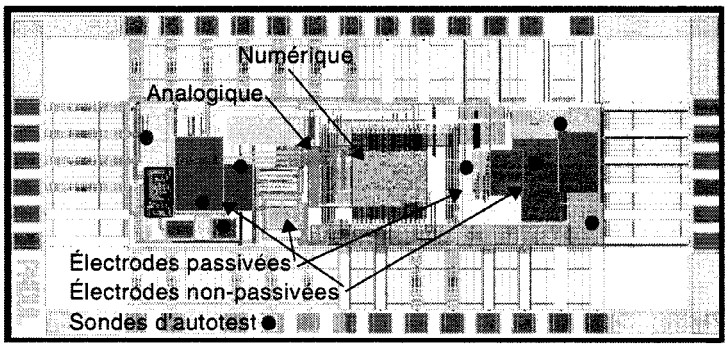


Figure 2.6 Emplacement des sondes d'autotest

2.2.3 Module de JTAG

L'architecture supérieure de la plateforme prend une entrée sérielle et adresse des données à tous les blocs de la plateforme. Cette architecture et conforme à la norme IEEE1149.1 ou JTAG. Ce choix de design se justifie par le faible nombre de pins d'entrée au circuit nécessaire pour permettre un accès à beaucoup de modules à l'intérieur du circuit.

En détail, le module JTAG est utilisé pour programmer la plateforme avec des paramètres de tests dans chacun des modules. Par exemple, on peut programmer laquelle des branches du miroir de courant sera activée ou encore la durée du signal "activation" (Fig. 2.7). Une fois les paramètres de test entrés dans les modules de mesure, on peut procéder au test lui-même. On utilise aussi le module JTAG pour lire les résultats de tests une fois qu'ils sont terminés.

Bien entendu, un registre à décalage relativement simple pourrait offrir les mêmes avantages qu'une topologie JTAG dans le cas qui nous intéresse. Par contre, étant donné que ce projet se veut une plateforme de mesure, il est intéressant d'offrir à l'usager un accès standard au circuit, d'où l'intérêt de suivre une norme établie. Nous n'allons pas discuter des détails des exigences de la norme JTAG, ceux-ci étant très clairement énoncées dans les divers documents que la société IEEE met à la disposition de tous.

2.2.4 Module de mesure de délai

Le module de mesure de délai constitue une partie importante de la plateforme de mesure. Nous exposons notre évaluation des techniques existantes puis nos choix d'architectures adaptés à notre problème.

2.2.4.1 Module de mesure de délai proposé

À la lumière de l'analyse des méthodes de mesure de délai existantes, nous proposons une méthode adaptée à la plateforme de test qui vise le test expérimental et la caractérisation. Il n'est donc pas nécessaire d'optimiser surface et puissance consommée dont d'autres applications de mesure de délai pourraient avoir besoin. Nous posons donc la mesure de délais les plus fins possible comme priorité. La technique de mesure de délai à l'aide d'un oscillateur bouclé (sect 1.2.1.1) étant très prometteuse, nous l'avons choisie pour notre design. Par contre, nous ajoutons un élément important: le procédé de caractérisation en entier est effectué directement sur le circuit.
Ce choix de conception permet une mesure directe de la féquence et, par conséquent, cela permet de garder une précision très élevée. Une structure de calibration permet

aussi de réduire l'incertitude causée par des variations de délais entre des instances différentes d'une porte logique qui sont supposément identiques. Les mesures sont effectuées en plaçant la structure sous test sur des électrodes qui font partie d'un chemin de la boucle d'oscillation, ce qui augmente le délai dans la ligne et permet de mesurer la fréquence résultante sur un compteur. Le temps pendant lequel ce compteur est activé est aussi très finement ajusté pour permettre une mesure de fréquence très précise. Une description détaillée de la méthode de caractérisation est fournie à la section 2.2.4.3. L'unité de mesure proposée peut être séparée en trois blocs: l'oscillateur bouclé, l'horloge de référence et le compteur de fréquence, comme l'illustre la figure 2.7.

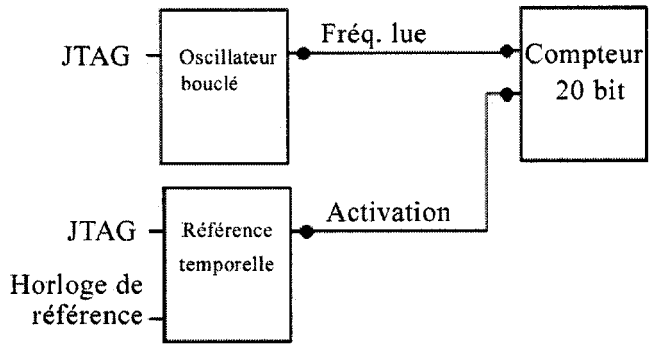


Figure 2.7 Schéma-bloc du module de mesure de délai

1. L'objectif de l'oscillateur bouclé est de transformer le délai et les différences de délai de l'ordre de la picoseconde produits par la nanostructure sous test en un signal de test "Fréq lue" qui oscille à une fréquence inférieure à 500 MHz. La figure 2.8 montre un des 8 chemins de l'oscillateur bouclé qui peuvent être utilisés pour permettre une oscillation dans la boucle.

   Une paire d'électrodes, où un dispositif sous test est placé, est représentée par le symbole d'une résistance dans le figure 2.8. Les divers chemins possibles de la ligne de délai passent à travers une structure de commutation qui laisse

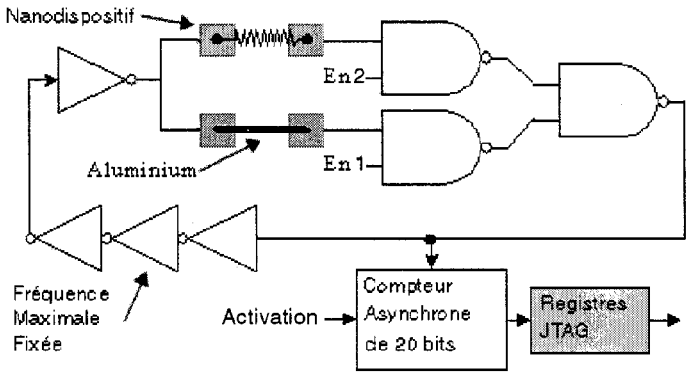


Figure 2.8 Oscillateur bouclé configurable

l'usager choisir laquelle des 8 paires d'électrodes sera utilisée. Nous n'utilisons pas un multiplexeur classique, parce que l'incertitude de délais entre deux chemins est inférieure à celle causée par l'élément logique présenté à la figure 2.8.

2. Un deuxième module, "Référence temporelle", est composé d'un compteur programmable. Son entrée "Horloge de référence" sert de base de temps pour la référence temporelle. Cette entrée peut être dérivée d'un cristal, par exemple. Le signal de sortie de ce module, "Activation" (fig 2.7), est une impulsion dont la longueur est un multiple programmable de celle de l'entrée "Horloge de référence".

3. Le troisième module est un compteur asynchrone de 20 bits qui s'incrémente sur les fronts montants du signal de test "Fréq. lue" quand le signal "Activation" est actif. L'architecture de compteur asynchrone a été choisie à cause de la très grande fréquence qu'elle peut atteindre, qui est de 500 MHz pour l'unité de mesure de délais proposée dans ce mémoire. Ce compteur, d'une taille de 20 bits, permet de compter plus de 1,000,000 de périodes de l'oscillateur. Utiliser une architecture de compteur asynchrone demande une période de stabilisation inférieure à 1 ns par étage après que le compteur

ait fini d'incrémenter. Notre compteur de 20 bits requiert donc 20 ns pour se stabiliser. La plateforme de test ne permet pas la lecture du compteur pendant cette période.

2.2.4.2 Formules utilisées pour calculer le délai mesuré

On utilise la formule suivante pour extraire le délai de propagation dans la nanostructure sous test:

$$Dboucle = \frac{1}{2}\frac{Ndiv}{2 * Fclkref}\frac{1}{Compte} \tag{2.1}$$

Où

$Ndiv$ est le facteur de division de la fréquence chargé dans le module "Référence temporelle" (Fig. 2.7),

$Fclkref$ est la fréquence du signal "Horloge de Référence" (Fig. 2.7),

$Compte$ est le nombre de périodes d'oscillation dans la boucle comptées par le compteur asynchrone.

Le premier facteur "$\frac{1}{2}$" dans l'équation précédente est dû au fait que la période mesurée a le double de la durée du délai dans la boucle. Le second facteur $\frac{1}{2}$ représente le fait que le signal "Activation" produit par le module "Référence temporelle" (Fig. 2.7) n'est actif que lorsqu'il est à un niveau logique positif.

Le délai de propagation mesuré dans la boucle de référence est ensuite soustrait à celui de la boucle de l'échantillon sous test. Le délai résultant est celui qui est causé par la nanostructure sous test.

$$Delai_nano = Dboucle_nano - Dboucle_ref \tag{2.2}$$

2.2.4.3 Discussion sur la précision du module de mesure de délai

Comme la précision du module de mesure de délai doit être la plus grande possible, il importe de discuter de ses limites.
La mesure de délais est basée sur la différence de délai de propagation entre deux chemins à parcourir dans un oscillateur bouclé. Toutes les différences de délais entre ces deux chemins qui ne sont pas causées par l'échantillon sous test sont donc des sources d'erreur. Il importe donc d'éliminer ces sources d'erreurs le plus possible pour augmenter la précision de la mesure.

Une observation détaillée de la figure 2.8 permet d'établir la plus grande source d'érreur de ce circuit. Il s'agit de la différence de délai entre les deux portes "NAND" au centre du circuit. Ces portes logiques sont crées á partir de transistors CMOS dont la taille fixée par le concepteur demeure variable à cause d'aléas de fabrication. Bien entendu, comme les portes sont disposées côte-à-côte sur le circuit, il serait très pessimiste d'envisager une variation maximale sur les délais de propagation entre ces portes. En effet, les variations de tailles des transistors sont moindres lorsques ces derniers sont disposés à proximité l'un de l'autre.

Dans le cas du circuit de la figure 2.8, nous avons utilisé des portes "NAN2D4" de la bibliothéque CMOSP18 de la fonderie TSMC. Comme l'indique la feuille de spécifications de la fonderie pour cette porte logique, la différence de délais entre deux portes distinctes peut atteindre 38 ps. À la connaissance de l'auteur, aucune simulation ne peut prévoir exactement la différence de tailles de transistors en fonction de leur position sur un circuit intégré. Il est donc difficile de prévoir à quel point le délai dans les élements de la boucle d'oscillation sera modifié par les aléas de fabrication.

Une amélioration possible de la structure du circuit de la figure 2.8 demande une

analyse analogique de ce circuit numérique. En effet, en utilisant des transistors de grande taille et en créant des

Pour améliorer encore plus la précision de la mesure, une version subséquente de la plateforme de test pourrait bénéficier de l'utilisation de fusibles placés entre les électrodes de contact. Ainsi, lors des premières utilisations de ce module, on pourrait caractériser les délais de propagation de toutes les branches de la boucle d'oscillation. Une fois cette caractérisation effectuée, on pourrait utiliser une ablation LASER localisée pour détruire le fusible. Ensuite, il suffirait d'ajouter l'échantillon sur les électrodes et de procéder à la mesure de délais dans léchantillon. Les seules sources d'erreurs restantes après ces manipulations demeureraient au niveau des interfaces entre le nanodispositif et les électrodes de contact. Si on considère ces interfaces comme partie prenante du dispositif sous test, alors on obtiendrait une mesure très précise.

2.3 Intégration de nanostructures

La dernière partie de ce chapitre concerne l'intégration des nanostructures à la plateforme de test. Une des plus grandes difficultés de l'approche que constitue ce projet consiste à relier des nanostructures à un circuit intégré commercial en CMOS. Les méthodes de fabrication et de manipulation des nanostructures limitent nos choix d'intégration à un faible nombre de nanostructures. Comme nous en avons discuté brièvement en introduction du chapitre 1, ce sont les nanotubes de carbone qui ont remporté la palme de la nanostructure que nous souhaitons intégrer.

Cette section montre les choix de design que nous avons faits en termes d'architecture et de méthodes d'intégration de nanotubes de carbone à la plateforme de test.

2.3.1 Architectures utilisables

La plateforme de test a été fabriquée avec une technologie TSMC CMOS 180 nm, ce choix n'est pas le seul qui nous est offert. Dans cette section, nous justifions l'utilisation de cette technologie.
Les ressources technologiques dans le domaine de l'électronique sont très variées. Il importe d'en explorer les possibilités pour choisir la technologie la plus en mesure de répondre aux besoins de mesure de la plateforme.

2.3.1.1 Plaque de silicium "nue"

Plusieurs méthodes de mesure de nanostructures prévoient l'intégration de nanostructures sur une plaque de silicium avec gravures de conducteurs. On parle ici d'une application sans circuit électronique actif intégré. Des électrodes gravées sur la plaque (sect 1.1.3.8) contiennent des embouts macroscopiques utilisés pour faire contact avec des éléments de mesure à haute précision externes. La simplicité et les résultats de test de nanostructures prouvés (sections 1.1.3.7 et 1.1.3.9) par cette méthode la rende très attirante. Par contre, le lien entre la plaque et la sonde induit des inductances et des capacités parasites qui filtrent des signaux dont la fréquence dépasse quelques dizaines de MHz. Comme nous avons établi que les requis de ce projet comprennent la mesure de délais de propagation de l'ordre de quelques picosecondes, les signaux à des fréquences élevées ne peuvent pas être mesurés sur une architecture comme celle-ci. Nous l'avons donc rejeté.

2.3.1.2 Circuits intégrés configurables

Le choix d'une architecture électronique passe souvent par des circuits intégrés configurables et reconfigurables, tels des FPGAs. Même si des circuits à haute performance sont possibles sous cette architecture, ce choix doit être écarté d'emblée, car ces architectures ne permettent pas d'intégrer de nanostructures au circuit. En effet, nous verrons à la section 2.3.3 que des électrodes sont nécessaires pour offrir un lien adéquat entre les nanotubes et le circuit sous test. De telles électrodes ne peuvent être fabriquées dans un FPGA.

2.3.1.3 Circuits intégrés à application spécifique

Étant donné que les autres technologies ne permettent pas de répondre aux requis de ce projet, il a fallu nous tourner vers une architecture ASIC. Cette architecture permet la fabrication de circuits de test à très haute performance et l'intégration de nanostructures sur un même circuit, sans avoir à faire des liens avec l'extérieur. On protège ainsi l'intégrité des signaux, ce qui permet d'effectuer des mesures très précises. Nous discuterons plus en détail sur les techniques pour l'intégration des nanostructures à la plateforme à la section 2.3.2.

Le *Groupe de Recherche en Microélectronique* a une entente avec la CMC[2] pour fabriquer à faible coût des circuits intégrés CMOS à la fonderie TSMC[3]. Étant donné que notre circuit doit mesurer des signaux à très haute fréquence, il est préférable d'utiliser la technologie la plus récente possible pour en réduire les temps de propagation de signaux et de commutation des portes. Au moment où la plateforme a été conçue, la technologie CMOS 180 nm correspondait à cette exigence.

[2]Corporation de Microélectronique du Canada, http://www.cmc.ca/
[3]Taiwan Semiconductor Manufacturing Corporation

2.3.2 Intégration de CNTs à la plateforme de test

La majorité des équipes de recherche qui fabriquent des CNTFETs intègrent des nanotubes de carbone à une plaque de silicium "nue" (sect. 2.3.1.1). Comme nous avons rejeté cette approche, il importe de proposer une méthode alternative d'intégration de nanotubes de carbone à notre plateforme. Cette section montre et justifie les méthodes de fabrication et de manipulation que nous préconisons.

2.3.2.1 Choix de la méthode de fabrication

Examinons d'abord les méthodes de fabrication des nanotubes de carbone. L'installation des tubes dépend de la méthode de fabrication utilisée (sect. 1.1.3.6) et du choix d'architecture: sur circuit intégré ou sur plaquette de silicium.

- La méthode de CVD permet de faire croître des nanotubes de carbone à des endroits déterminés. Cette propriété permet de faciliter l'intégration de CNTs ainsi créés. Malheureusement, les températures supérieures à 600^oC nécessaires pour mener à bien cette méthode la rendent inopérante dans le cadre de notre plateforme de test qui serait vraisemblablement détruite sous cette chaleur.

- La méthode de fabrication de nanotubes par impact laser se produit dans un four à 1200^oC. On doit donc aussi écarter cette méthode pour notre plateforme de test.

- La seule méthode que nous pouvons utiliser est donc la décharge électrique. Cette méthode de fabrication de CNTs ne permet pas une croissance localisée

comme les deux autres méthodes, ce qui complique en certains points leur intégration.

2.3.2.2 Choix de la méthode de placement

Une fois les CNTs créés par décharge électrique et filtrés (sect. 1.1.3.9), le problème de placement de ces tubes sur les électrodes de contact demeure entier. Une méthode de placement usitée (Avouris, 2003) consiste à déposer les CNTs créés par décharge électrique à des positions et selon des orientations aléatoires (sect. 1.1.3.6). Cette technique doit être modifiée en certains points pour correspondre au contexte spécifique de notre plateforme:

- Limitations d'espace:
  L'espace requis pour fabriquer les macroplaques de contact avec les sondes de test est tout simplement trop élevé pour l'espace disponible sur le circuit intégré. De plus, pour avoir une probabilité suffisament élevée de placer un nanotube de carbone sur une paire d'électrodes par cette méthode, il faut graver un nombre élevé de paires d'électrodes, ce qui occupe une grande surface de silicium.

- Un problème soulevé par le dispersement aléatoire de nanotubes de carbone est le nombre limité d'exemplaires de la plateforme de test à "gaspiller" lorsqu'aucun nanotube n'a été placé adéquatement sur une paire d'électrodes.

- Comme mentionné à la section 2.3.1.1, le contact entre la macroplaque métallique et la sonde induit des capacités parasites qui empêchent les mesures à haute fréquence.

À la lumière de la dernière observation, il pourrait être tentant de conserver le lien macroplaques-sondes pour effectuer des mesures DC et utiliser la plateforme de test pour effectuer les mesures à haute fréquence. Malheureusement, les limitations d'espace mentionnées plus haut réduisent beaucoup les probabilités de succès de cette méthode. Nous concevons donc la surface d'électrodes en fonction d'une intégration des CNTs utilisant une pointe d'AFM comme expliqué à la section 1.1.3.8. Cette approche est risquée en grande partie à cause des difficultés occasionées par les manipulations des AFM. Dans le cas où nous échouons à cette tâche, nous considérerons la méthode de dispersion aléatoire utilisée par l'équipe du IBM Research Division, malgré les faibles probabilités de succès de cette méthode.

Richard Martel, professeur à l'Université de Montréal et pionnier des CNTFET [(Avouris, 2003), (Martel, 1998), (Martel, 2001)], recherche actuellement une méthode pour améliorer le placement de CNTs sur des électrodes de contact. Il est permis de croire que des versions subséquentes de la plateforme pourraient grandement bénéficier des résultats de ses travaux.

2.3.3 Conception de la surface d'électrodes de contact

L'architecture que nous proposons est basée sur une technologie ASIC et prévoit l'utilisation d'AFM pour disposer les CNT sur des électrodes. Nous utilisons l'étage supérieur de métallisation de la plateforme pour former ces électrodes. Pour la technologie CMOS 180 nm TSMC que nous utilisons, il s'agit du 6^{ieme} étage.

Les électrodes de contact ont une forme rectangulaire de 10μm par 20μm. La figure 2.9 montre l'emplacement des électrodes de contact sur la plateforme. Ces électrodes sont placées le plus près possible des circuits de stimulation qui le joux-

tent, au dessus de la figure. Ces électrodes ne sont pas sous une fenêtre sans passivation. En effet, seule la moitié des électrodes a été placée sous une fenêtre de passivation. Sur la figure 2.9, on voit une partie de deux de ces fenêtres en gris foncé en haut à droite. Nous avons pris soin de garder une distance minimale de 50 μm entre les électrodes et tout composant électronique actif.

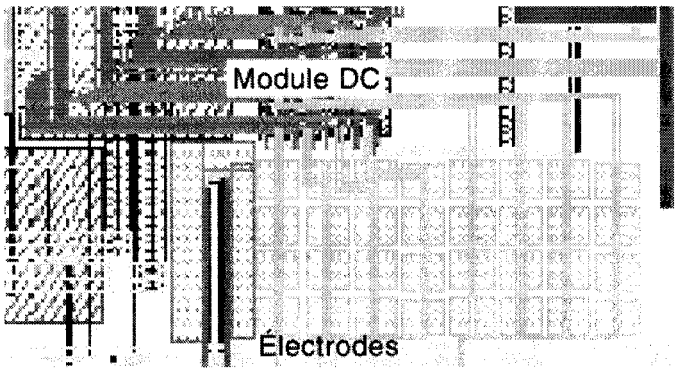


Figure 2.9 Dessin de masques des électrodes de contact

2.3.3.1 Emplacement des électrodes de contact

Les électrodes de contact doivent être placées à des endroits stratégiques de la plateforme de test. En effet, pour accéder à ces électrodes, des ouvertures dans la couche de passivation sont nécessaires. Ces ouvertures représentent un danger pour les circuits actifs qui composent les modules de mesure internes. Il faut donc éviter de placer ces électrodes directement au dessus ou trop près de ces circuits actifs. Par contre, si on les place trop loin, on risque de perdre de la précision dans les mesures à cause des grandes impédances des fils qui relieront les électrodes aux modules de mesure internes. L'objectif est ici de trouver un compromis entre ces préoccupations.

2.3.3.2 Liaison des électrodes aux modules de mesure internes

Une fois les électrodes placées, il faut les relier adéquatement aux modules de mesure internes. Pour ce faire, il faut d'abord choisir des chemins balancés entre toutes les électrodes pour ne pas occasioner de variations d'impédances entre les différents exemplaires de paires d'électrodes. En effet, rappelons que les mesures que nous effectuons sont relatives à une référence canonique, comme un fil d'aluminium. Il est donc impératif que les différents chemins qui conduisent aux électrodes amènent un minimum d'incertitude

Il importe également de choisir quelles électrodes doivent être branchées ensemble. En effet, Il y a beaucoup plus de paires d'électrodes que d'entrées dans les modules de mesure. Par contre, il faut considérer qu'une manipulation erronée lors du placement de nanotubes peut rendre inutile une paire d'électrodes, par exemple en causant un court-circuit entre les électrodes. Dans un tel cas, toutes les électrodes reliées à cette première paire deviendraient inutilisables à leur tour. On a donc lié les électrodes en groupe de quatre paires. Sur la figure 2.9, on voit bien ces groupes de quatre paires d'électrodes.

2.4 Conclusions du chapitre

Dans ce chapitre, nous avons montré les différentes parties de la plateforme de test. Nous avons aussi examiné plusieurs méthodes de mesure disponibles et choisi les plus intéressantes pour répondre aux requis de notre projet. Au prochain chapitre, nous montrerons, entre autres, les simulations que nous avons effectuées lors de la conception de cette plateforme.

CHAPITRE 3

MODÉLISATION D'UN NANO-COMPOSANT

Un des buts principaux de la plateforme de test consiste à mesurer des caractéristiques électroniques de nanostructures. Ces caractéristiques pourront ensuite servir de base pour établir des modèles de nanodispositifs électroniques. Bien entendu, plusieurs données expérimentales demeureront longtemps inconnues. Ainsi, pour guider la conception d'équipements de mesure et des procédures de test, il est primordial de prodéder à certaines simulations qui serviront ensuite de base de comparaison avec les résultats expérimentaux. Dans ce chapitre, nous verrons une méthode d'adaptation d'un modèle de transistor conventionnel au comportement de CNTFETs.

Cette méthode consiste à utiliser un modèle de transistor conventionnel et d'en modifier les paramètres pour l'adapter à des résultats expérimentaux de CNTFETs publiés. Ces paramètres ont été extraits avec un programme exploitant des algorithmes génétiques et que l'auteur de ce mémoire a entièrement codé.

3.1 Modèles de transistors

Les logiciels de simulation de transistors conventionnels dans des environnements tels HSpice et Spectre utilisent des modèles mathématiques représentant leur comportement selon plusieurs paramètres. Un modèle utilisé actuellement pour représenter les transistors CMOS conventionnels est le BSIM3. Ce modèle offre l'avantage de représenter de nombreux comportements précis de ces transistors tels les courants de fuite et les effets de canaux courts. Étant donné que la connaissance du com-

portement des transistors à base de nanotubes de carbone demeure très faible à ce jour, il est préférable d'utiliser des modèles moins spécialisés pour ces structures. Comme aucun modèle de type BSIM3 n'est encore disponible pour représenter les nano-dispositifs à base de nanotubes de carbone, il est souhaitable de créer des modèles de ce type afin de prévoir le comportement de ces nanostructures. Jin (Jin, 2004) propose d'utiliser un modèle d'abord présenté par Curtice et Ettenberg (Curtice, 1985) pour modéliser des transistors composés en partie de GaAs[1]. Dans son article, en préparation, il propose des paramètres à implanter dans ce modèle pour produire des courbes I-V correspondant au comportement expérimental de CNTFETs. En effet, il est parvenu à déterminer les paramètres optimaux à inclure dans le modèle de C. & E. pour créer une courbe I-V très proche de celle d'un CNTFET expérimental.

3.2 Modèle et données expérimentales traités avec le programme

Les paramètres que Jin propose (sect 3.1) servent à caractériser le comportement de certains CNTFETs fabriqués et mesurés. Nous confirmons ici le bien fondé de l'utilisation du modèle de C. & E. en déterminant les paramètres à utiliser pour créer le modèle d'un autre CNTFET. De plus, nos résultats s'ajoutent à ceux de Jin pour amorcer une bibliothèque de modèles de CNTFETs qui pourront ensuite être utilisés par des concepteurs de systèmes électroniques hybrides CMOS/carbone. Dans la prochaine section, nous allons donc discuter du CNTFET particulier dont nous modélisons le comportement.

[1] arséniure de gallium

3.2.1 CNTFET de McEuen et al.

Le CNTFET que nous modélisons est un transistor à grille électrolytique dont le canal est un nanotube de carbone à simple couche. Pour ce faire, nous utilisons les résultats expérimentaux de McEuen (Mceuen, 2002). Ce transistor baigne dans une solution de NaCl et offre une grille électrolytique en chlorure d'argent (figure 3.1).

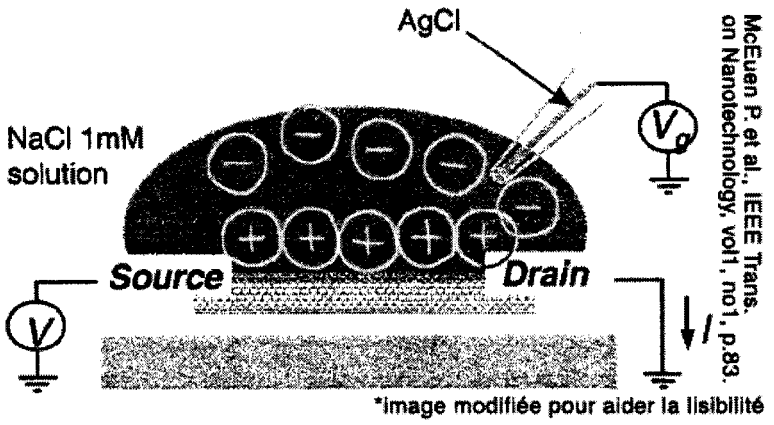


Figure 3.1 Transistor électrolytique de McEuen et al.

La courbe I-V de ce CNTFET est affichée à la figure 3.2. Ce transistor P est bien entendu très difficile à reproduire en grand nombre dans le contexte d'un circuit intégré à grande échelle. Néanmoins, ce dispositif expérimental offre une transconductance $dI/dVg = 20\mu A/V$ pour une tension de grille $Vg = -0.9V$. Considérant que le canal peut être aussi mince que 1-2 nm, on peut normaliser sa transconductance par unité de largeur à 10 mS / μm. En comparant cette caractéristique avec celle d'un transistor n-CMOS 0.8 μm de taille minimale en saturation (Allan, 2001), de l'ordre de 0.2 mS/μm, ce dispositif a bien les caractéristiques de conduction des CNTFETs énoncées à la section 1.1.3.4.

Nous utilisons donc les résultats expérimentaux de McEuen et al. pour confirmer

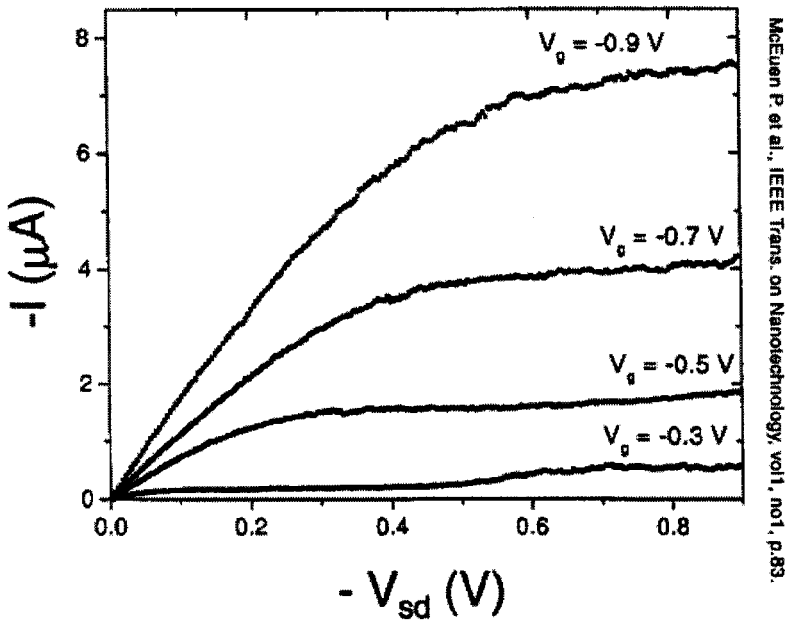


Figure 3.2 Courbe I-V du transistor de McEuen et al.

que l'utilisation du modèle de Curtice et Ettenberg correspond au comportement des CNTFETs.

3.2.2 Modèle de FET de Curtice et Ettenberg modifié

L'équation que nous utilisons pour modéliser le CNTFET est la même que celle présentée par Jin (Jin, 2004):

$$Ids(\alpha, \beta, \gamma, \lambda, b, v'_{t0}, v_{gexp}) =$$

$$\frac{\beta}{1+b(v_{gs}-v'_{t0})}[(v_{gs} - (v'_{t0} + \gamma v_{ds}))^2]^{v_{gexp}/2} * tanh(\alpha v_{ds})(1 + \lambda v_{ds})$$

La modification que Jin a apportée à l'équation originale de C. & E. en améliore l'exactitude pour modéliser les CNTFETs. Nous invitons le lecteur soucieux de vérifier les détails de ces modifications à lire la référence disponible en annexe de ce

mémoire. Par contre, afin de bien comprendre la fonction présentée ci-haut, nous décrivons à la section suivante ses paramètres fixes, c'est-à-dire α, β, γ, λ, b, v'_{t0} et v_{gexp}.

3.2.3 Description des paramètres fixes du modèle de Curtice et Ettenberg

Les paramètres de la fonction de C. & E. modifiée que nous utilisons sont les suivants (Andersson, 1991), (Jin, 2004) :

Symbole	Paramètre	Unité
α	Constante de tanh	V^{-1}
β	Coefficient de transconductance	A/V^2
γ	Coefficient de tension de seuil	nil
λ	Coefficient de modification de longueur de canal	V^{-1}
b	Paramètre du profil de dopage	V^{-1}
v'_{t0}	Tension de seuil généralisée	V
v_{gexp}	Exposant variable	nil

Tableau 3.1 Définition des paramètres du modèle C. & E. modifié

Il est évidemment possible que certains des éléments de ce modèle ne représentent pas la réalité d'un dispositif comme un CNTFET. Par exemple, le paramètre "b" représentant le profil du dopage n'a pas de lien direct avec un dispositif sans dopant comme un CNTFET. Il pourrait donc sembler judicieux de retirer ce paramètre du modèle[2]. Cela aurait comme avantage de réduire le temps de calcul du programme, car il aurait moins de paramètres à optimiser.

Nous allons tout de même conserver ce paramètre, car il pourrait modéliser certains phénomènes inconnus dans un dispositif tel un CNTFET. De plus, comme avec notre modèle, il n'y a que 7 paramètres, la recherche d'une combinaison optimale

[2]ou de le fixer à une valeur neutre, ce qui a le même résultat dans notre cas

avec notre programme n'est pas aussi complexe que pour un modèle comme le BSIM3. Ce modèle ayant plus de 50 paramètres, il demanderait une stratégie différente

3.3 Programme de génération de paramètres pour le modèle C. & E.

Nous avons codé un algorithme génétique pour déterminer les paramètres optimaux pour recréer les courbes présentées à la figure 3.2. Ce programme est codé selon l'interprète Gambit 3.0 du langage Scheme, une implémentation de Lisp. Le code est disponible en annexe de ce mémoire. Nous justifions l'utilisation d'un algorithme génétique par le fait que nous connaissons très peu le comportement des données et cette approche permet d'atteindre une solution optimale sans connaissance préalable des données. Regardons plus en détails l'adaptation de cet algorithme que nous avons effectuée.

3.3.1 Pseudocode de l'algorithme génétique utilisé dans ce programme

Algorithme *génétique*

(* Appliqué à un modèle de CNTFET *)

1. Créer N cellules qui constituent le bassin génétique initial. Chaque cellule comporte 7 gènes, i.e. les paramètres du modèle de C. & E modifié par Jin: $\alpha, \beta, \gamma, \lambda, b, v'_{t0}$ et v_{gexp}
2. **tant que** aucune cellule n'obtient un critère $< K_1$ **ou** après K_2 itérations
3. **faire** Éliminer une partie des cellules dont le critère est le moins optimal.
4. "Accoupler" les cellules restantes: **tant que** le bassin génétique contient plus qu'une cellule
5. Retirer 2 cellules au hasard du bassin: le "père" et la "mère".

6. Prendre les i premiers gènes du "père", où i $\in \mathbb{N}$, aléatoire à distribution continue, 0<=i<=6.
7. Prendre les $7 - i$ derniers gènes de la "mère".
8. Concaténer ces 7 gènes pour créer une cellule "enfant".
9. Déposer les parents et l'enfant dans le bassin de sortie.
10. Faire "muter" quelques cellules prises au hasard: modifier au hasard 1 gène choisi au hasard dans chaque cellule mutée.
11. Éliminer tous les "clones" sauf 1: les cellules trop semblables sont des clones.
12. Ajouter assez de cellules pour atteindre le nombre N initial de cellules.
13. Mesurer le critère pour toutes les cellules: générer une courbe I-V avec ces paramètres et calculer la différence au carré de cette courbe avec la référence expérimentale de McEuen.
14. **retourner** la cellule ayant le critère le plus bas

Des changements manuels ont dû être effectués sur cet algorithme pour atteindre la solution. Ces changements ont consisté, entre autres, à ajuster les constantes K_1 et K_2 à mesure que nous observions le comportement général du programme avec ces données. Nous discuterons plus en détail de cette méthode à la section suivante.

3.3.2 Description du code du programme

Cette section est un guide pour aider à la compréhension du programme de génération de paramètres au modèle C. & E. présenté en annexe. Ce programme est codé en Scheme, un langage Lisp fonctionnel. Nous décrivons d'abord les fonctions principales du programme :

La fonction genetic-search est la partie principale du programme. Son argument max-iter est le nombre maximum d'itérations, K_2 dans le pseudocode (sect 3.3.1). Elle retourne un bassin des meilleures cellules obtenues à la fin des itérations. Le fonctionnement de cette fonction est un calque du pseudocode. Notons que la liste "perf" est utilisée pour afficher périodiquement la progression des performances.

La fonction mate-population effectue l'accouplement des cellules tel que décrit au point 4 du pseudocode (sect 3.3.1).

La fonction mutate-population effectue la mutation des cellules tel que décrite au point 10 du pseudocode.

La fonction kill-clones élimine les cellules trop semblables tel que décrit au point 11 du pseudocode.

La fonction create-cell crée une cellule dont les gènes sont pris selon une distribution gaussienne centrée sur les paramètres du modèle de C. & E. de transistors pris comme référence.

Le fichier ivcnt800.scm représente les courbes I-V du CNTFET de McEuen. Nous utilisons ces données comme référence pour mesurer l'erreur sur nos résultats.

La fonction eval-params mesure le critère d'une combinaison de paramètres par différence au carré avec la courbe de référence de McEuen.

Lors d'une utilisation normale du programme, l'utilisateur n'a qu'à poser les paramètres K_1 et K_2, puis à démarrer l'interprétation du code. Pour des recherches complexes, par exemple lorsque la population comprend 1000 cellules et qu'on itère plus de 100 fois, il est préférable de compiler le Scheme en C. Cela diminue les

temps de calculs d'environ un ordre de grandeur. Cela cause par contre certains inconvénients: l'accès à plusieurs variables internes est bloqué et il devient impossible de modifier le programme alors qu'il est en marche.

Qu'il soit interprété ou compilé, le programme produit une cellule optimale pour chaque itération. Le critère obtenu pour chaque cellule est affiché pour des fins de statistiques. En effet, on peut constater que le programme est dans un minimum local de solution lorsqu'il ne trouve plus de meilleure solution pendant un grand nombre d'itérations. Cette situation montre que le nombre d'itérations est suffisant et force l'utilisateur à changer d'autres prémisses pour faire progresser la solution. Les prémisses que nous avons fait varier pour arriver à une solution sont les suivantes:

- la probabilité de mutation d'une cellule,
- l'écart-type des distributions de probabilités utilisées pour créer les gènes,
- la proportion d'élimination effectuée par la fonction "reduce-pool", à l'étape 3 du pseudocode (sect 3.3.1),
- la taille de la population,
- la méthode de mesure du critère: par différence des carrés ou par division,
- la loi de probabilité utilisée pour générer les gènes: continue ou gaussienne,
- le niveau acceptable de similarité de deux gènes dans la fonction "very-similar?" utilisée par "kill-clones", à l'étape 11 du pseudocode.

Nous n'entrerons évidemment pas dans les détails impliquant l'historique de modification de ces paramètres. La valeur de ces paramètres du programme a été

déterminée par inspection et nous pourrions automatiser le choix de ces paramètres. Cela permettrait d'améliorer le champ d'action du programme à un large spectre de dispositifs électroniques. Nous reviendrons sur ce sujet en conclusion du chapitre. La dernière partie du programme, très simple, consiste à produire une simulation DC du dispositif modélisé par les paramètres inclus dans la cellule optimale. Les résultats de cette simulation sont sauvegardés sous forme de points d'un graphique représentant la courbe I-V de ce dispositif ainsi que de la courbe de référence de McEuen. On affiche ensuite cette courbe avec le logiciel "gnuplot"[3]. La courbe résultante finale est affichée à la prochaine section.

3.4 Résultats

Nous avons jugé par inspection que la courbe I-V affichée à la figure 3.3, produite par le programme était conforme à nos attentes. Ces résultats ont été obtenus après 50 utilisation successives du programme pendant lesquelles 200 itérations étaient effectuées. Pendant chacune de ces utilisations du programme, on a considéré un groupe de gènes initiaux différent et on a généré un bassin de 1000 individus.

Cette courbe a été produite avec la cellule offrant les paramètres finaux obtenus par la procédure d'optimisation. Comme le point de départ de notre programme était d'utiliser les paramètres d'un transistor p, nous comparons ces paramètres au tableau 3.2.

À la lumière de ces résultats, on peut affirmer que le modèle de Curtice et Ettenberg (Curtice, 1985) modifié par Jin (Jin, 2004) permet de décrire adéquatement le comportement DC du transistor à base de nanotube de carbone de McEuen (Mceuen, 2002). Sur la base de cette observation, il est permis d'espérer que ce

[3] 4^{ieme} distribution, version disponible notamment sur Linux, Fedora Core 1

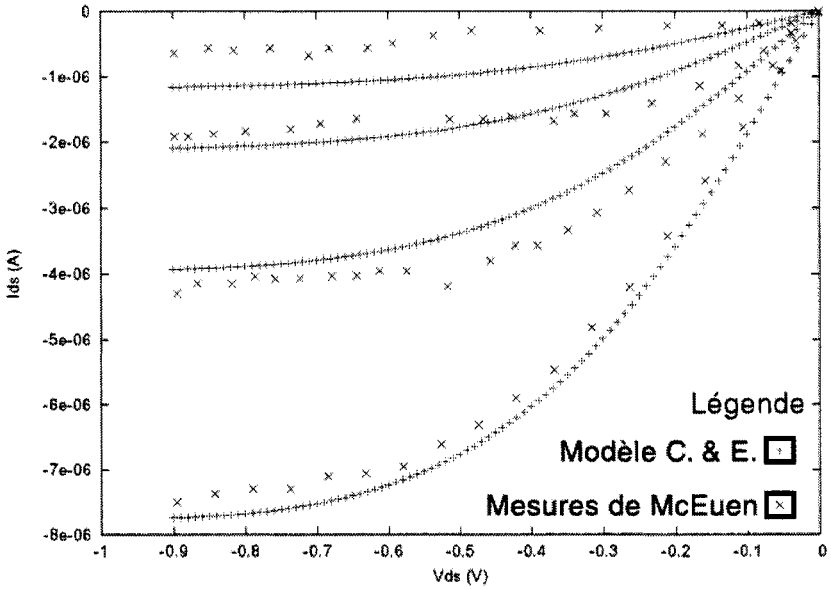


Figure 3.3 Courbe I-V expérimentale de McEuen vs modèle de Curtice et Ettenberg

modèle pourrait servir pour une grande partie des CNTFETs.

Par contre, étant donné que les résultats expérimentaux de McEuen n'illustrent qu'une plage de courants de grille et de drain presque limitée à la zone active, il est primordial d'étendre le modèle à d'autres zones d'activités du CNTFET. Il est de notre avis que le modèle devra alors être enrichi de la même façon que le BSIM3 pour bien représenter le comportement du dispositif. En effet, dans ce modèle, on utilise des équations et des paramètres différents pour plusieurs tranches de V_{DS}.

De plus, les résultats que nous avons obtenus pour le transistor de McEuen pourraient probablement être améliorés en automatisant la méthode d'assignation de prémisses décrite à la section 3.3.2. L'utilisation de très grandes populations et de simulations très longues, par exemple, 100k individus pendant 10k itérations pourrait aussi améliorer la réponse. Le temps de calcul énorme qu'une telle solution

Paramètre	McEuen	pCMOS
α	2.32	2.56
β	0.021	1.319
γ	-0.61	0.061
λ	-3.29	-0.07
b	0.147	-1.026
v'_{t0}	0.186	-0.07
$v_{gexp}/2$	-4.28	1.06

Tableau 3.2 Paramètres du modèle du transistor de McEuen

demande la rend par contre difficile à réaliser. En effet, même lorsque compilé en "C", le programme effectuait un calcul typique[4] en 2 heures sur un Pentium 4 à 1.5 GHz. Même avec une croissance linéaire de l'algorithme, un tel calcul nécéssiterait 14 mois selon ces conditions.

3.5 Conclusion du chapitre

Le modèle de Curtice et Ettenberg utilisé par Jin (Jin, 2004) représente adéquatement le comportement du CNTFET électrolytique de McEuen. Les paramètres que nous avons dégagés montrent aussi qu'il est possible d'adapter ce modèle à plusieurs dispositifs. On pourrait améliorer encore la performance de ce modèle en l'appliquant par section de Vds comme dans certains modèles Spice. Lors de l'application du modèle dans un logiciel de conception, une telle approche pourrait être envisageable. Cela correspondrait à un cas particulier de nos résultats qu'il serait intéressant d'étudier.

[4]avec 1000 individus et 200 itérations

CHAPITRE 4

CARACTÉRISATION PAR SIMULATION DES STRUCTURES DE TEST

Ce chapitre montre les résultats de simulations effectuées avec les logiciels *Cadence Analog Artist* et *ModelSim Design Analyzer*. Ces simulations permettent de vérifier le fonctionnement de tous les modules de la plateforme de test. De plus, les résultats de simulations montrent l'incertitude sur les mesures effectuées avec chacun de ces modules.

Ce chapitre compte deux sections qui montrent respectivement les résultats de simulations du module de mesure DC et du module de mesure de délai.

4.1 Résultats de simulation du module DC

Les simulations du module DC permettent de montrer les incertitudes causées par la structure du module et par des aléas de fabrication. La section suivante traitera de l'incertitude sur la mesure DC causée par la structure du module DC.

4.1.1 Incertitude structurelle du module DC

Pris en entier, le module de mesure DC offre une erreur de mesure maximale de 0.1% de la source de courant à la broche de tension de sortie. La figure 4.1 illustre une simulation du module de mesure DC de résistances dans la gamme 100Ω-1MΩ, utilisées pour émuler des nanostructures sous test. Nous effectuons ici le rapport entre les tensions mesurées et idéales pour mettre en évidence les différences entre

ces deux ensembles de données. Les courbes illustrent donc ce rapport de mesure VDSrel défini par:

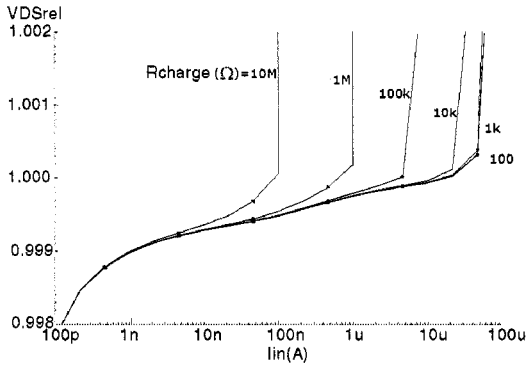


Figure 4.1 Précision de la mesure du module DC

$$VDSrel = Vout_lu/Vout_ideal \tag{4.1}$$

Où

Vout_lu est la tension mesurée avec le module de mesure DC.

Vout_ideal est la tension aux bornes de la nanostructure sous test sans le module de mesure DC.

La courbe montre que dans la plage d'intérêt de 100 pA à 100 μA, la déviation de VDSrel de sa valeur idéale demeure inférieure à 0.1%.

4.1.2 Incertitude du module DC causée par les aléas de fabrication

Comme les méthodes de fabrication de circuits intégrés ne peuvent garantir que la taille des transistors produits sera exactement la même que celle prévue par le

concepteur, on doit effectuer des simulations qui montrent cette incertitude.

4.1.2.1 Définition des tests de coin

La simulation précédente ne tient pas compte des variations des conductances des transistors causées par des erreurs de procédé de fabrication. Nous avons donc effectué des tests de coin pour démontrer que notre circuit fonctionne malgré ces variations.

Pour ces simulations, on utilise le circuit illustré à la figure 4.9[1]. Ce circuit est composé de deux parties distinctes: à gauche, le miroir de courant et à droite, un amplificateur opérationnel en configuration suiveur. Pour cette simulation, les portes de transmission sont activées en configuration "mesure I-V". Les portes de transmission "Switch 1" et "Switch 3" sont donc activées et la porte "Switch 2" est désactivée.

Dans ce cas, l'entrée du circuit est la source de courant "Iin", i.e. la source "I3". La sortie du circuit est indiqué par l'étiquette "Vout" située à droite du schéma. Les entrées "Ipol" et "Vpol" sont disponibles sur des broches du circuit. Ces entrées permettent de calibrer respectivement le courant de polarisation du miroir de courant et de l'amplificateur opérationnel utilisé à la sortie du circuit. On émule la nanostructure sous test avec une résistance variable, "R0" sur le schéma.

Pour effectuer les tests de coin, on modifie simplement le modèle des transistors utilisés lors de différentes simulations. On choisit donc des temps de commutation courts ou longs pour les transistors "P" et "N". On obtient ainsi quatre configurations possible: "SS", "SF", "FS" et "FF". Le code "FS" représente des temps de

[1]Ce circuit est placé en fin de chapitre à cause de sa grande taille.

commutation rapide des transistors "N" et lent des transistors "P", ce qui correspond à des courants élevés et faibles. Ces quatre cas représentent donc les quatre coins d'un ensemble de variations possibles du procédé de fabrication. Un cinquième test correspondant au centre du domaine de variations des procédés, "TT", représente un test typique, effectué sans se soucier des variations de procédés.

4.1.2.2 Description des tests de coin

Les figures suivantes montrent les résultats des simulations effectuées avec les tests de coin décrits à la section précédente. Ces simulations ont été effectuées en utilisant un courant de polarisation de 10 μA et une tension de polarisation de 1.2V. Nous avons procédé à une simulation pour plusieurs valeurs de résistance de test: 1 Ω, 600Ω, 12kΩ et 200kΩ. Chacun des résultats est présenté avec une échelle linéaire et logarithmique pour les courants de simulations. L'échelle logarithmique permet d'illustrer le comportement du module de mesure avec des faibles courants et l'échele linéaire permet de montrer la courbe I-V d'une résistance de façon plus intuitive: un trait oblique!

Comme expliqué à la section 2.1.1, les mesures effectuées avec le module de mesure DC sont relatives. En effet, on doit soustraire à la tension mesurée la tension de référence obtenue avec la branche reliée à l'intérieur du circuit. Ici, cette tension de référence est illustrée avec les simulations sur une résistance de 1Ω. Les figures 4.3, 4.4 et 4.5 montrent les résultats de simulations desquels les tensions de référence ont déjà été soustraites pour différentes valeurs de résistances représentant les nanostructures sous test.

On a choisi ces valeurs de résistances pour illustrer le comportement du module de mesure DC sur une grande gamme de résistances. Bien entendu, comme on ne connaît pas à l'avance la résistance des nanostructures que l'on souhaite mesurer,

il est important d'être en mesure de mesurer un grande gamme de résistances.

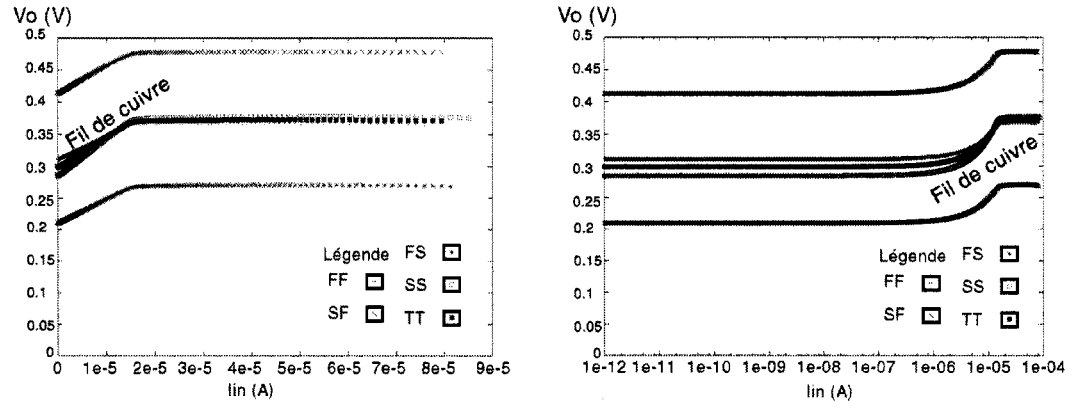


Figure 4.2 Test de coin de référence (R=1Ω)

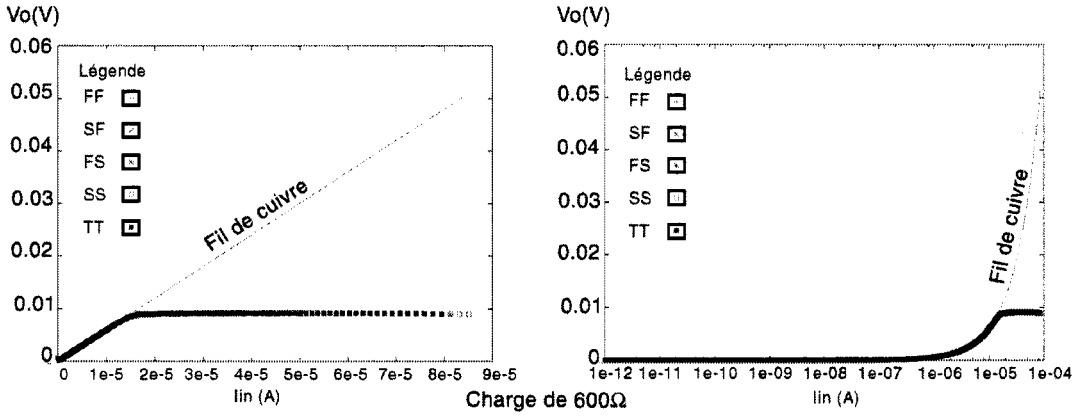


Figure 4.3 Test de coin avec R=600Ω

4.1.3 Discussion des résultats de simulation DC

Les résultats de simulations illustrés à la section précédente montrent bien les conséquences des variations de procédés sur la mesure effectuée. En effet, sans faire de mesure relative, la précision des mesures serait tellementfaible que le module de mesure en deviendrait inutilisable. De plus, dans les simulations linéaires, on voit la limite de mesure du module de mesure DC. Par exemple, la figure 4.3 montre une limite de courant d'entrée d'environ 14μA. Des courants d'entrée supérieurs à

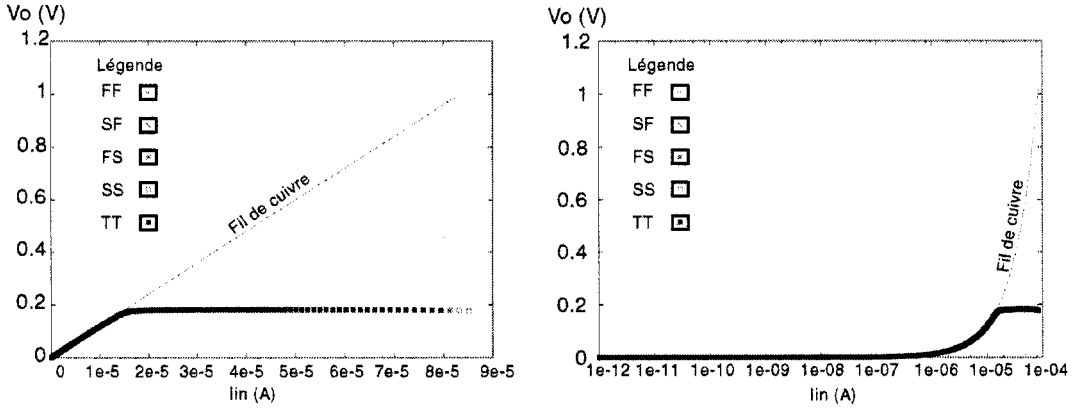


Figure 4.4 Test de coin avec R=12kΩ

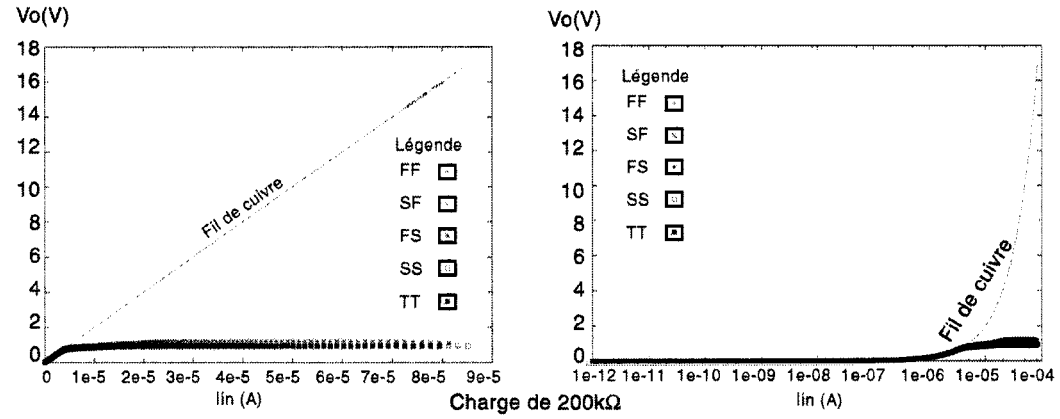


Figure 4.5 Test de coin avec R=200kΩ

cette valeur ne génèrent pas des tensions adéquates à la sortie du module de mesure DC. La même observation peut être faite avec les figures 4.4 et 4.5.

On explique en partie cette incertitude avec le passage dans le miroir de courant dont les branches ne peuvent être balancées parfaitement. La taille des transistors dans chacune des branches doit être exactement la même si l'on souhaite avoir un miroir complètement balancé.

Par contre, à l'intérieur des balises constituées par ces limites, le circuit de mesure DC fonctionne très bien, comme en témoigne la courbe représentant une résistance

idéale, courbe que suivent les mesures dans les zones de fonctionnement du module de mesure.

Les limites de courant imposées à l'entrée ne sont évidemment pas souhaitables et devraient être réduites dans les versions prochaines de la plateforme de test. On pourrait aussi souhaiter obtenir des mesures d'une alimentation à l'autre plutôt que selon la plage de tension relativement faible qu'on observe à la figure 4.3.

4.2 Simulation du module de mesure de délai

Les simulations entourant le module de mesure de délai montrent que les circuits répondent aux exigences énoncées au chapitre 2. En effet, on peut mesurer des délais de propagation de l'ordre de 1 ps avec ce module. On présente dans cette section deux simulations. La première est une simulation numérique qui permet de montrer l'exactitude du modèle avec des délais typiques dans les portes numériques. La seconde est une simulation analogique qui permet de valider le comportement de la partie "interface" du module de mesure de délai.

4.2.1 Simulation numérique du module de mesure de délai

Cette simulation permet de valider l'exactitude de notre modèle pour une mesure de délai. Elle suppose que l'échantillon ayant un délai de propagation de 1 ps plus élevé qu'un fil conducteur d'une longueur de $\simeq$ 1 μm. Comme ce délai est très faible, nous l'avons pris comme nul dans notre modèle. Ainsi, la simulation démontre le nombre de cycles mesurés par le compteur asynchrone du module de mesure pour chacun de ces modèles.

Les résultats de simulations donnent 513212 cycles pour le circuit qui passe par



Figure 4.6 Simulation numérique du module de mesure de délai

le nanotube émulé et 513734 cycles pour celui qui passe dans le fil conducteur. Cette simulation dure exactement une milliseconde, telle que définie dans le module de référence de temps. Les fréquences correspondant aux simulations sont donc de 513.212 MHz et de 513.734 MHz pour la simulation du nanotube et du fil de conducteur. Les délais de propagation correspondant à ces simulations sont donc de 974.256ps et de 973.266ps respectivement.La mesure effectuée correspond à calculer la différence entre ces mesures, ce qui donne 1.01ps dans ce cas. Comme nous recherchions une mesure de 1.0 ps, l'erreur de mesure est ici de 1%, un résultat très acceptable.

Notre méthode de mesure réduit considérablement les incertitudes causées par des variations de température. En effet, il suffit de procéder aux deux mesures à 1 minute d'intervalle ou moins pour que la température dans la zone d'intérêt demeure presque constante.

Il est à noter que cette simulation comprend tous les signaux de contrôle du module JTAG, ces signaux ne sont pas pertinents dans la mesure de délai, nous omettons donc intentionnellement de les présenter ici.

Cette simulation ne tient pas compte des aléas de fabrication intrinsèques aux circuits intégrés. Les outils mis à notre disposition nous permettent de procéder à des simulations qui démontrent les conséquences de ces aléas. Par exemple,

une simulation de Monte Carlo pourrait nous permettre de faire varier les délais de toutes les portes logiques qui constituent le module de mesure de délai pour trouver le délai maximal et minimal de chacun des chemins critiques. Par contre, comme nous effectuons une mesure différentielle, tous les élements communs aux deux mesures sont annulés. Nous ne ferons pas varier des éléments qui seront annulés par la suite.

4.2.2 Simulation analogique du module de mesure de délai

Comme le délai présenté dans la simulation numérique a été créé à partir de simulations en VHDL, il est important de valider notre modèle avec un circuit analogique, jugé plus près de la réalité. Nous avons donc remplacé le nanotube de carbone sous test par un simple circuit RC. Nous avons effectué des tests en branchant ce circuit à une paire d'inverseurs et en variant la valeur de "R". Ainsi, sur la figure 4.8, on voit l'onde carrée à 500MHz utilisée en entrée et la réponse du circuit. Le condensateur utilisé dans le circuit RC est de 10fF.

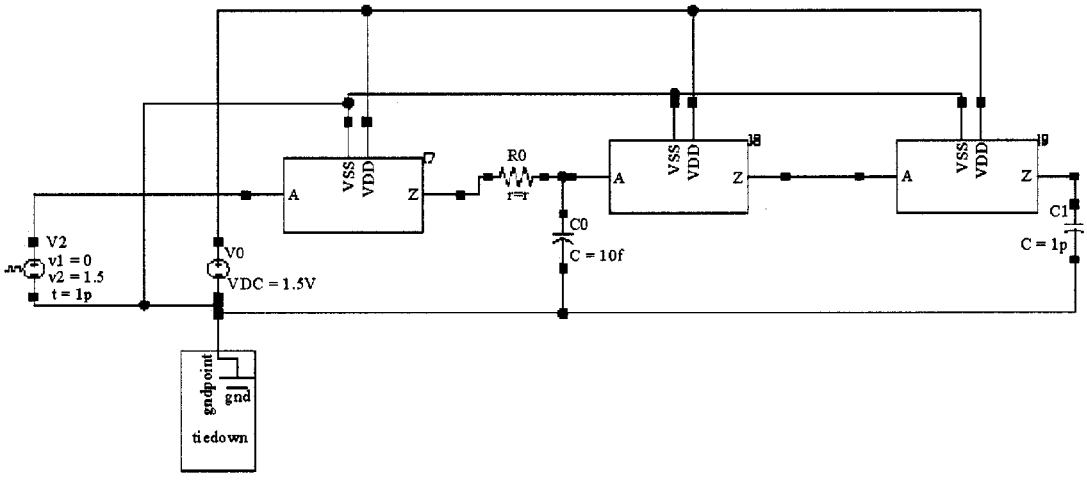


Figure 4.7 Circuit utilisé pour la simulation analogique du module de délai

À la lumière des résultats obtenus avec cette simulation, on remarque que pour des résistances inférieures à 50k, la réponse de notre modèle est suffisament rapide pour effectuer une lecture correcte du signal. Le délai de propagation d'une telle structure s'évalue à environ $5\tau = 2.5$ ns. Comme le signal d'entrée de notre simulation oscille à 500 MHz, il est normal qu'une telle structure n'offre pas la chance au signal d'atteindre VDD. Par contre, pour tous les autres cas, étant donné l'intégrité du signal sortant du circuit, on peut affirmer que l'inverseur qui alimente le nanotube procure un courant suffisant pour permettre le passage du signal numérique à l'intérieur du modèle de nanotube. Cette simulation nous permet d'envisager qu'un branchement effectué avec un nanotube de carbone permettra aussi le passage adéquat du signal de stimulation.

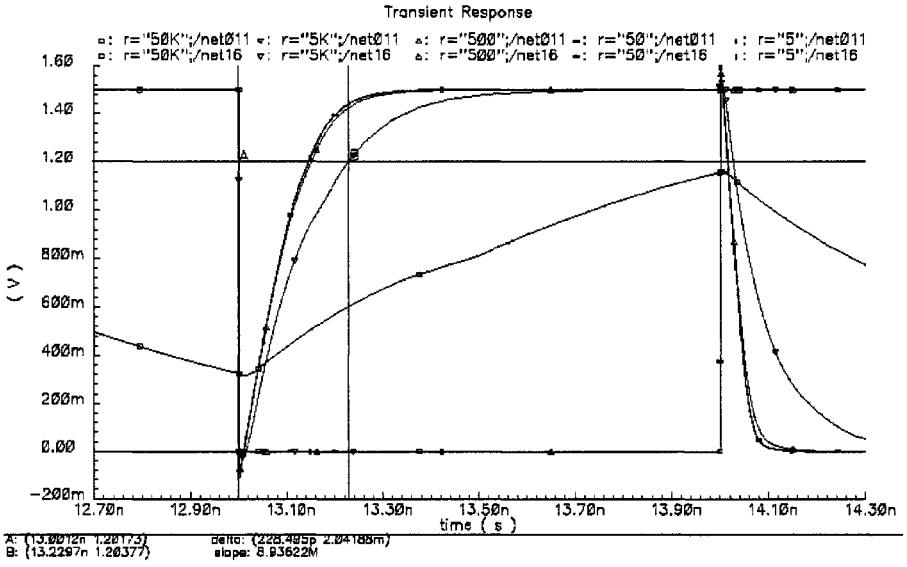


Figure 4.8 Simulation analogique du module de mesure de délai

4.3 Conclusion des simulations

Ce chapitre permet de démontrer les bases de la conception de notre plateforme de test. Nous avons effectué des tests permettant de prévoir les changements de

comportement causés par des variations de procédés de fabrication. Nous avons démontré aux sections 4.1.3 et 4.2.1 qu'en faisant des mesures relatives, ces changements de comportement deviennent négligeables. Au chapitre suivant, nous verrons les mesures effectuées sur la plateforme.

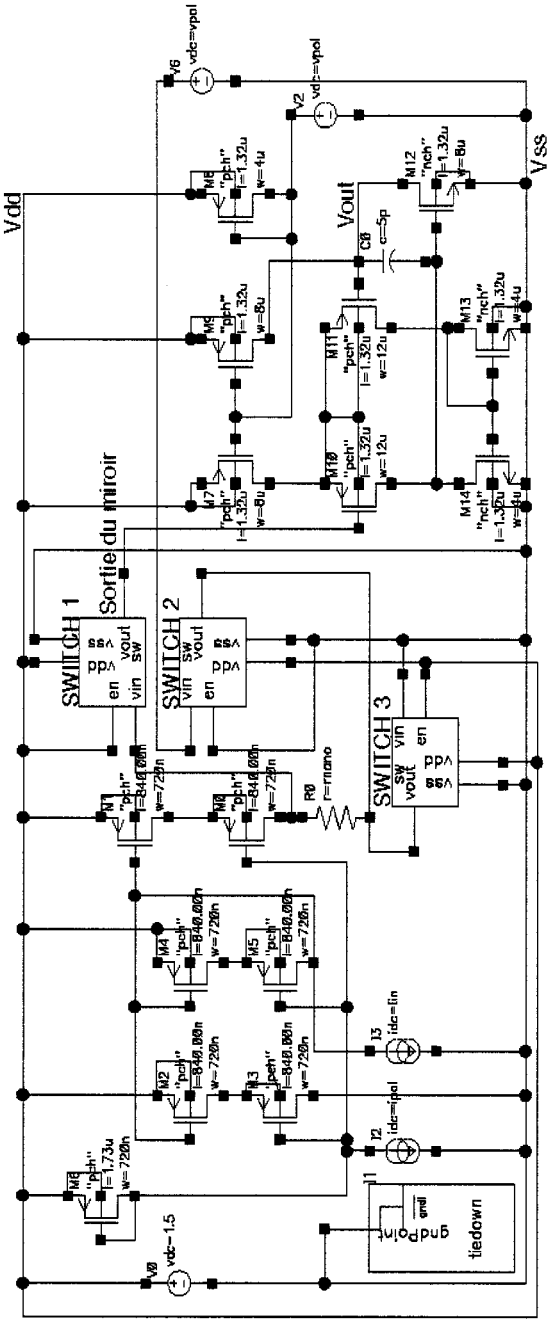


Figure 4.9 Circuit utilisé pour les tests de coin

CHAPITRE 5

RÉSULTATS

La plateforme de test n'a pas fonctionné comme prévu. En effet, il était impossible de générer quelque résultat que ce soit à partir de l'architecture supérieure de contrôle. Comme cette dernière était au coeur du fonctionnement du circuit, nous n'avons pas pu faire de mesures avec la plateforme. Ce chapitre se concentrera donc sur l'information utile que nous avons pu extraire de la plateforme ainsi qu'aux mesures que nous avons prises pour tenter de la faire fonctionner.

5.1 Évaluation du procédé de fabrication

Les méthodes de microfabrication de la compagnie TSMC étant gardés comme secret industriel, il est difficile de prévoir la composition exacte des surfaces de contact utilisées en tant qu'électrodes. De plus, la méthode de gravure n'étant pas connue, il est difficile de prévoir le résultat de manipulations qui ne respectent pas les règles de design de la fonderie. Comme nous l'avons énoncé à la section 2.3.1.1, la fabrication des électrodes de contact sur la plateforme de test requiert des manipulations dangereuses. Nous avons donc fabriqué le circuit selon la méthode énoncée à la section 2.3.1.1 et nous avons obtenu la confirmation que la méthode fonctionne bien.

5.1.1 Électrodes de test

On peut observer sur le micrographe (figure 5.1) que les électrodes ont bel et bien la forme désirée et que la couche d'isolant qui constitue leur base est essentiellement intacte. On peut même la qualifier de lisse. Cela implique nécessairement que la fonderie procède à une gravure sèche[1] plutôt que d'utiliser un acide, par exemple. L'éventualité qu'une gravure par acide soit utilisée aurait rendu la tâche de fabrication des électrodes plutôt ardue, car les interstices non-métalliques entre les électrodes sont susceptibles de fondre en leur présence. Cela aurait évidemment pour conséquence de détruire la plateforme entière.

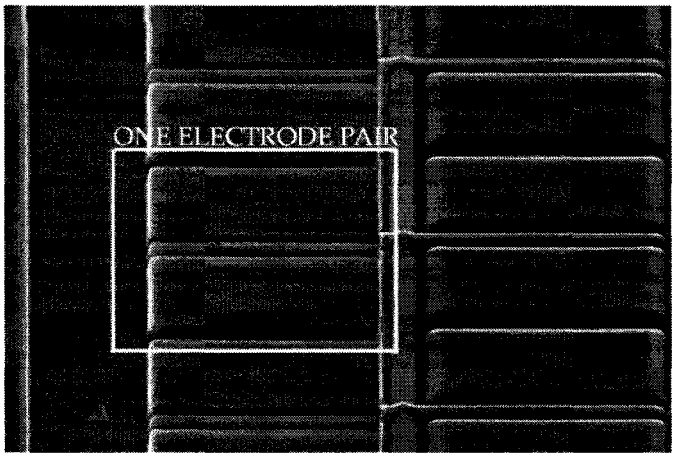


Figure 5.1 Micrographe des électrodes fabriquées

5.1.2 Structure de caractérisation hors règles de design

Un autre groupe de structure a été fabriqué sur la surface de la plateforme pour sonder les limites de taille des structures dessinées sur la 6^{ieme} couche métallique. Le micrographe suivant (figure 5.2) montre ces structures. Même si en deçà des limites

[1] dry etching: par plasma, LASER ou autre

imposées par les règles de design, les coins des carrés sont arrondis et des court-circuits apparaissent entre des structures rapprochées, il est intéressant de constater la répétitivité de la forme obtenue. Cela implique qu'avec certaines manipulations post fonderie, il serait possible d'utiliser de telles structures pour fabriquer des électrodes de formes différentes dans des versions subséquentes de la plateforme de test.

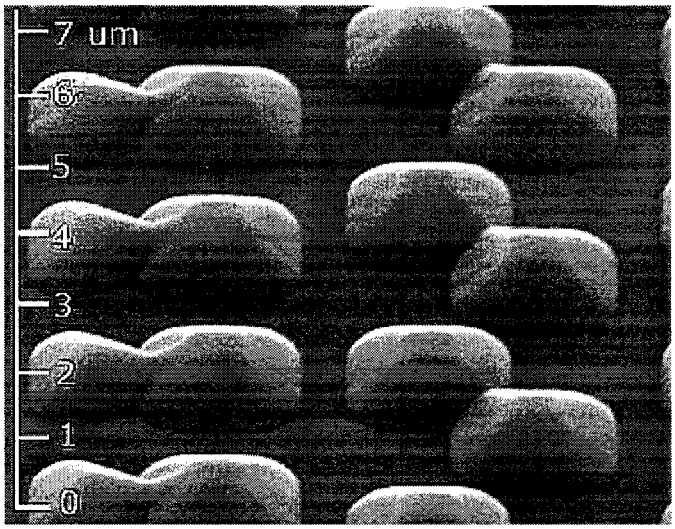


Figure 5.2 Micrographe des électrodes fabriquées

5.2 Mesures effectuées avec la plateforme

La fonction principale de la plateforme de test n'étant pas opérationelle, nous décrirons ici tout ce qui a été tenté pour la faire fonctionner. Les détails des vecteurs de test sont explicités en annexe de ce mémoire. Nous montrerons dans cette section les mesures électriques effectuées au laboratoire du GRM[2].

[2]Groupe de Recherche en Microélectronique

5.2.1 Montage du circuit de mesure sur carte

La plateforme de test est un circuit intégré à application spécifique monté dans un boîtier 80 CFP[3]. Voici donc, étape par étape la méthode utilisée pour effectuer le montage de son circuit de mesure:

1. Coupe des bouts des broches du boîtier.
2. À l'aide de pincettes, insertion du boîtier dans une cassette en polymère fabriquée sur mesure pour ce circuit.
3. À l'aide de pincettes, déposition d'élastomères sur les traces métalliques de la carte de test devant faire contact avec les broches de la plateforme.
4. Insertion de la cassette sur la carte de test.

La carte utilisée est spécialement conçue pour relier les broches du boîtier de la plateforme aux entrées/sorties du testeur logique.

5.2.2 Mesures effectuées sur l'analyseur logique

Les mesures effectuées sur cet appareil se sont toutes avérées infructueuses. En effet, on doit se rappeler l'architecture supérieure de la plateforme énoncée à la section 2.2: le JTAG. Les tests effectués dans cette section font appel au JTAG pour accéder aux fonctions internes de la plateforme. Voici en détail la méthode utilisée:

1. Insertion de la carte de test sur le testeur logique

[3]Ceramic Flat Pack

2. Branchement d'une source de tension aux bornes de la carte.

3. Mise sous tension de la source de tension et du testeur logique

4. Vérification que le courant consommé à la source est d'environ 1 mA[4,5].

5. Envoi des vecteurs de test et prise de mesure automatique des résultats.

Comme les résultats exposés en annexe le montrent, le JTAG n'a répondu à aucun test. En effet, même le plus simple des tests, un "BYPASS", ne parvenait pas à faire fonctionner le JTAG. Étant donné que cette structure n'a pas fourni les résultats escomptés, le reste de la plateforme ne peut être accédé et nos tests sur l'IMS ne peuvent être poussés plus loin.

5.3 Conclusion des mesures

Nous avons effectué des tests et pris des mesures correspondant aux méthodes énumérées dans ce chapitre sans obtenir le moindre signal à la sortie du circuit. Cette situation est évidemment très frustrante et l'apanage de nombreux concepteurs ASIC. Elle conduit à de longs et pénibles moments pour tenter de résoudre un problème impossible à régler.

[4]S'il est de l'ordre du μA, le boitier est désenligné dans la cassette.
[5]S'il est bien supérieur à 1 mA, il y a court circuit.

CONCLUSION

De nombreuses conclusions peuvent être tirées des quelques années de travail passées à étudier, concevoir et fabriquer un circuit pour effectuer des expérimentations dans le domaine de la nanoélectronique.

Tout d'abord, à plusieurs reprises pendant ce projet, de nombreux intervenants provenant du milieu des nanotechnologies semblaient pour le moins sceptiques quant à la faisabilité de notre entreprise d'intégration nano/microélectronique. En réduisant au minimum la difficulté d'intégration de nanostructures à un circuit microélectroniques par la création d'électrodes de contact, nous croyons avoir facilité cette intégration.

La plateforme de test en elle-même avait une bonne conception: robuste, précise, compacte. Si l'on doit chercher un coupable pour l'échec de cette conception, on doit bien sûr mentionner le manque d'expérience du concepteur de la plateforme. Concevoir un circuit microélectronique mixte dédié comporte des difficultés énormes qu'un concepteur novice a de la difficulté à surmonter sans l'aide constante d'un mentor intéressé.

Un des objectifs du projet consistait à créer des modèles pour des caractéristiques électriques de nanotubes de carbone. Nous avons atteint cet objectif en modélisant grâce au modèle de Curtice et Ettenberg, que nous avons modifié pour permettre la modélisation de transistors à base de nanotubes de carbone. Ce modèle pourra servir de point de référence pour des mesures de nanostructures effectuées dans le futur. Il pourra aussi servir à créer des systèmes comprenant plusieurs transistors à base de nanotubes de carbone. Ces systèmes pourraient exploiter les propriétés électroniques uniques de ce matériau révolutionnaire.

En conclusion, l'objectif de ce projet était colossal pour un projet de maîtrise en microélectronique. Sa conclusion en est donc peu surprenante. Heureusement, d'autres chercheurs reprennent le flambeau de l'intégration de nanostructures sur des circuits microélectroniques. Le temps aidera ces chercheurs, car de nouvelles méthodes de fabrication permettront d'intégrer plus facilement ces nanostructures à des circuits conventionnels.

En termes de conclusion finale, nous prévoyons qu'à mesure que le canal des transistors CMOS s'écourtera, la liste de chercheurs en nanoélectronique grandira proportionellement. Les ressources mises à leur disposition seront également beaucoup plus grandes qu'actuellement.

RÉFÉRENCES

ALLAN, HOLLBERG, 2001, "CMOS Device Modeling", *http://www.ece.utexas.edu/~holberg/lecture_notes/bk3.pdf*

ANDERSSON, M., 1991, "GaAs MESFET Modeling for the Digital Integrated Circuit Simulation", *Licentiate Thesis, Helsinki University of Technology, Otaniemi, Finland, 81 p.*

AVOURIS, P., APPENZELLER, J., MARTEL, R., WIND, S.J., 2003, "Carbon Nanotube Electronics", *Proceedings of the IEEE, vol 91, no 11, november 2003, pp. 1772-1784*

BACHTOLD, A. ET AL., 1999, "Aharonov Bohm oscillations in carbon nanotubes", *Nature, vol 397, pp 673-675*

BERGER, C., YI, Y., WANG, W.A., DEHEER, W.A., 2001, "Multiwalled carbon nanotubes are ballistic conductors at room temperature", *Applied Physics A, vol 74, pp. 363-365*

BERGMAN, J.I., CHANG, J., JOO, Y., MATINPOUR, B., LASKAR, J., JOKERST, N.M., BROOKE, M.A., BRAR, B., BEAM, I., 1999, "RTD/CMOS Nanoelectronic Circuits: Thin-Film Inp-Based Resonant Tunneling Diodes Integrated with CMOS Circuits", *IEEE Electron Device Letters, vol. 20, no 3, pp. 119-122*

BETHUNE, D.S., KIANG, C.H., DEVRIES, M.S., GORMAN, G., SAVOY, R., VAZQUEZ, J., BEYERS, R., 1993, "Cobalt-Catalyzed Growth of Carbon Nanotubes with Single Atomic Layer Walls", *Nature, vol 363, pp 605-607*

BETTE, A., DEBROSSE, J., GOGL, D., HOENIGSCHMID, H., ROBERTAZZI, R., ARNDT, C., BRAUN, D., CASAROTTO, D., HAVRELUK, R., LAMMERS,

S., OBERMAIER, W., REOHR, W., VIEHMANN, H., GALLAGHER, W.J., MULLER, G., "A High-Speed 128Kbit MRAM Core for Future Universal Memory Applications" *VLSI Circuits, 2003. Digest of Technical Papers. 2003 Symposium on , June 2003, pp. 217-220*

BROCK, D.K., MARTINET, S.S., BOCKO, M.F., 1995, "On-Chip Picosecond Delay Measurement of RFSQ Digital Logic Gates" *IEEE Trans. on Superconductivity, Vol.5, No. 2, pp. 2844-2848*

BURKE, P.J., 2003, "An RF Circuit Model for Carbon Nanotubes" *IEEE Trans. on Nanotechnology, Vol.2, Iss.1, p.55-58.*

COLLINS, P.G., AVOURIS, P., 2000, "Nanotubes for Electronics", *Scientific American, December 2000, pp 62-69*

COLLINS, P.G., AVOURIS, P., 2001, "Engineering Carbon Nanotubes and Nanotube Circuits Using Electrical Breakdown", *Science, 292, p.706*

CURTICE, W.R., ETTENBERG, M., "N-FET, a New Software Tool for Large-Signal GaAs FET Circuit Design", *RCA Review, vol 46, Sept.1985, pp. 321-339*

DAI, H., FRANKLIN, N., HAN, J., 1998, "Exploiting the Properties of Carbon Nanotubes For Nanolithography", *Applied Physics Letters, vol 73, p 1508*

DONG ,L., ARAI, F., FUKUDA, T., 2004, "Destructive construction of Nanostructures With Carbon Nanotubes Through Nanorobotic Manipulations", *IEEE tran. on Mechatronics, vol. 9, no. 2, pp. 350-357*

DUVAL, O., SAVARIA, Y., 2004, "An On-Chip Delay Measurement Module For Nanostructure Characterization", *Proc. of the 2004 IEEE Int'l. Symp. on Circuits and Systems*

EBBESEN, T.W., AJAYAN, P.M., 1992, "Large Scale Synthesis of Carbon Nanotubes", *Nature, vol 358, 220, pp 220-222*

ENDO, M. ET AL., 1995, "Purolytic carbon nanotubes from vapor-grown carbon fibres.", *Carbon 33, pp 873-881*

FRANK, D.J., LAUX, S.E., FISCHETTI, M.V., 1992, "Monte Carlo Simulation of a 30 nm Dual-Gate MOSFET: How Short Can Si Go?", *Proc. of the 1992 Int'l Electron Devices Meeting, Technical Digest., pp.553-556*

FRANK, S., PONCHARAL, P., WANG, Z.L., DEHEER ,W.A., 1998, "Carbon Nanotube Quantum Resistors", *Science, vol 280, pp 1744-1746*

IIJIMA, S., 1991, "Helical Microtubules of Graphite carbon", *Nature, vol 354, pp 56-58*

IIJIMA, S., ICHIHASHI, T., 1993, "Single-Shell Carbon Nanotubes of 1 nm Diameter", *Nature, vol 363, pp 603-605*

JIN, Z., DUVAL, O., SAVARIA, Y., LAURIN, J.-J., 2004, "Parametric study of an empirical model for IV-Curve of Carbon Nanotube Field Effect Transistor", *To be published*

JOACHIM, C., GIMZEWSKI, J.K., 1998, "A Nanoscale Single-Molecule Amplifier and its Consequences", *Proc. of the IEEE, vol. 86, no 1, pp. 184-190*

LAQUAI, B., RICHTER, H., HOEFFLINGER, B., 1995, "A Test Structure for the Measurement of Fast Internal Signals in CMOS VLSI Circuits", *Proc of the 1995 IEEE Conf. On Microelectronic Test Structures, vol 8, pp. 45-50*

MADOU, M.J., 2002, "Fundamentals of microfabrication: the science of miniaturization", *CRC Press, Boca Raton, Florida, USA, 723 p.*

MARTEL, R., WONG, H.-S. P., AVOURIS, P., 1998, "Single and multi-wall carbon nanotube field effect transistors", *Applied Physics Letters, vol 73, pp 2447-2449*

MARTEL, R., DERYCKE, V., LAVOIE, C., APPENZELLER, J., CHEN, K., TERSOFF, J., AVOURIS, P., 2001, "Ambipolar electric transport in semiconducting single-wall carbon nanotubes", *Physics, Review Letters, vol 87, pp 256 805/1-256 805/4*

MATHEWS, R.H., SAGE, J.P., SOLLNER, G., CALAWA, S.D., CHEN, C.-L., MAHONEY, L.J., MAKI, P.A., MOLVAR, K.M., 1999, "A new RTD/FET Logic Family", *Proc. of the IEEE, vol. 87, no. 4, pp. 596-605*

MATSUMOTO, K., GOTOH, Y., 2001, "A Wireless, Passive Carbon Nanotube-based Gas Sensor", *Proc. of the 2001 Int'l Semiconductor Device Research Symp., pp. 354 - 357*

MCEUEN, P., FUHRER, M.S., PARK, H., "Single-Walled Carbon Nanotube Electronics", *IEEE Transactions on Nanotechnology, Vol.1, No.1, March 2002, pp. 78-85*

MOON, S.I., LEE, D.J. ,JANG, Y.T., KIM, S.W., LEE, Y.H., JU, B.K., 2003 "Evaluation of Carbon Nanotubes as Gas Sensing Materials in Micro Gas Sensors", *Proc. of the 2003 Int'l Conf. on Solide State Sensors, Actuators and Microsystems, pp. 127-130*

NIYOGI, S., HU, H., HAMON, M.A., BHOWMIK, P., ZHAO, B., ROZENZHAK, S.M., CHEN, J., ITKIS, M.E., MEIER, M.S., HADDON, R.C., 2001, "Chromatic Purification of Soluble Single-Walled Carbon Nanotubes (s-SWNTs)", *J. Am. Chem., 123, pp. 733-734*

ONG, K.G., ZENG, K., GRIMES, C.A., 2003, "A Wireless, Passive Carbon Nanotube-based Gas Sensor", *IEEE Sensors Journal, vol. 2, no. 2, pp. 82-88*

PERA, F., SAVARIA, Y., BOIS, G., 1997, "Time Delay Methods for Integrated Transmission Lines and High Speed Cell Characterization", *Proc. of the 1997 IEEE Int'l. Symp. on Circuits and Systems, p.293-296*

PETRE, O., KERKHOFF, H.G., 2002, "On-Chip Tap-Delay Measurements for a Digital Delay-Line Used in High-Speed Inter-Chip Data Communications", *Proceedings of the 11th Asian Test Symposium, IEEE.*

REN, H., DENG, S.Z., CHEN, J., XU, N.S., 2003, "Carbon-nanotube based paste for field emission display application", *Proc. of the 2003 Int'l Vacuum Microelectronics Conf., p. 149*

ROSE, G.S., STAN, M.R., 2003, "Memory Arrays Based on Molecular RTD Devices", *Proc. of the 2003 Third Conference on Nanotechnology, vol.1, pp. 453-456*

SATO, T., SYLVESTER, D., CAO, Y., HU, C., 2001, "Accurate In Situ Measurement of Peak Noise and Delay Change Induced by Interconnect Coupling" *IEEE Journal of Solid-State Circuits, Vol.36, No. 10, pp. 1587-1591*

SAVARIA, Y., 1988, "Conception et vérification des circuits VLSI", *Éditions de l'École Polytechnique de Montréal, 1988, 398 p.*

SCOTT, C.D., AREPALLI, S., NIKOLSEV, P., SMALLEY, R.E., 2001, "Growth mechanisms for single-wall carbon nanotubes in a laser ablation process", *Applied Physics A, vol 72, p 573*

SCOTT, C.D., AREPALLI, S., NIKOLSEV, P., SMALLEY, R.E., 2002, "Growth mechanisms for single-wall carbon nanotubes in a laser ablation process", *Applied Physics A, Materials Science & Processing, vol. 74, p 11*

SOUMYANATH, K., BORKAR, S., ZHOU, C., BLOECHEL, B.A., 1999, "Accurate On-Chip Interconnect Evaluation: A Time-Domain Technique", *IEEE Journal of Solid-State Circuits, 34, 5, pp.623-631*

STUART, C., 2003, "Nanotubes are bidding for star billing on big screens", *Small Times, http://www.smalltime.com, September 12th*

TANS, S., VERSHUEREN, S., DEKKER, C., 1998, "Room-temperature transistor based on single carbon nanotube", *Nature, vol 393, pp 49-52*

TEHRANI, S., SLAUGHTER, J.M., DEHERRERA, M., ENGEL, B.N., RIZZO, N.D., SALTER, J., DURLAM, M., DAVE, R.W., JANESKY, J., BUTCHER, B., SMITH, K., GRYNKEWICH, K., 2003, "Magnetoresistive Random Access Memory Using Magnetic Tunnel Junctions", *Proceedings of the IEEE, vol. 91, no. 5, pp. 703-714*

WUDL, F. J., 2002, "Fullerene Materials", *Journal of Material Chemistry, 7, pp. 1959-1963*

ZHANG, M., LIN, X., HUANG, R., HAN, R., CHAN, M., 2003, "A self-aligned, electrically separable double-gate MOS transistor technology for dynamic threshold voltage application", *IEEE Transactions on Electron Devices, vol. 50, iss. 11, pp. 2297 - 2299*

ANNEXE I

CODE VHDL DE LA PLATEFORME DE TEST

ANNEXE 1: code VHDL de la plateforme

```
==========================================
Titre: testable_scannable_projet
Description: composant structurelle qui l'entite superieur du jtag

file: testable_scannable_projet.vhd
==========================================
Creation
date
1 fevrier 03 Olivier Duval
==========================================
library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
entity testable_scannable_projet is
port (
 pin JTAG de base
from_ech: in std_logic_vector(7 downto 0);
to_ech: out std_logic_vector(7 downto 0);
tck : in std_logic;
tms : in std_logic;
tdi : in std_logic;
tdo : out std_logic;
```

```
reset_n : in std_logic;
selftest_in : in std_logic_vector (size_ctrl-1 downto 0);
selftest_out : out std_logic_vector (size_ctrl-1 downto 0);
pin pour module analogique
sam_out : out std_logic_vector(15 downto 0);
scope_out : out std_logic_vector(15 downto 0);
pin pour module delai
enable : in std_logic;
clk_ref :in std_logic;
end_test: out std_logic);
end testable_scannable_projet;
 purpose: description structurelle
architecture struct of testable_scannable_projet is
-

Declaration des composants utilises

-
component tap_ctrl
port ( tck : in std_logic;
tms : in std_logic;
reset_n : in std_logic;
shiftIR : out std_logic;
clockIR : out std_logic;
updateIR : out std_logic;
shiftDR : out std_logic;
clockDR : out std_logic;
updateDR : out std_logic);
end component;
component ir
```

```
port (shiftIR : in std_logic;
clockIR : in std_logic;
updateIR : in std_logic;
reset_n : in std_logic;
scan_in : in std_logic;
scan_out : out std_logic;
data_in : in std_logic_vector (11 downto 0);
data_out : out std_logic_vector (11 downto 0));
end component;
component br_cell bypass register cell
port ( clockDR : in std_logic;
shiftDR : in std_logic;
scan_in : in std_logic;
scan_out : out std_logic);
end component;
component mux_4_1
PORT (a : in std_logic;
b : in std_logic;
c : in std_logic;
d : in std_logic;
g : in std_logic_vector (1 DOWNTO 0);
z : out std_logic);
END component;
component mux_2_1
PORT (a : in std_logic;
b : in std_logic;
g1 : in std_logic;
z : out std_logic);
END component;
component scannable_self composant du module autotest
```

```
generic (size_ctrl : integer :=8;
size_dr : integer :=8);
port (
reset_n : in std_logic;
fonction JTAG
shiftDR : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
mode_jtag : in std_logic_vector(1 downto
selftest_in : in std_logic_vector (size_ctrl-1 downto 0);
selftest_out : out std_logic_vector (size_ctrl-1 downto 0);
scan_in : in std_logic;
scan_out : out std_logic);
end component;
component scannable_projet composant du module analogique
generic (size_ctrl : integer := 8);
port (
sam_out : out std_logic_vector(15 downto 0);
scope_out : out std_logic_vector(15 downto 0);
reset_n : in std_logic;
fonction JTAG
shiftDR : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
mode_jtag : in std_logic_vector (1 downto 0);
scan_in : in std_logic;
scan_out : out std_logic);
end component;
component scannable_delai composant du module delai
generic (size_ctrl : integer :=23;
```

```
size_dr : integer :=20);
port (
 pin du component externe sauf jtag
from_ech: in std_logic_vector(7 downto 0);
to_ech: in std_logic_vector(7 downto 0);
enable : in std_logic;
clk_ref : in std_logic;
reset_n : in std_logic;
end_test: out std_logic;
fonction JTAG
shiftDR : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
mode_jtag : in std_logic_vector(1 downto 0);
scan_in : in std_logic;
scan_out : out std_logic);
end component;
```

Declaration des signaux des interconnexions

```
signaux JTAG
signal shiftIR : std_logic;
signal clockIR : std_logic;
signal updateIR : std_logic;
signal shiftDR : std_logic;
signal clockDR : std_logic;
signal updateDR : std_logic;
```

```
signaux du instruction register et bypass register
signal scan_in : std_logic;
signal scan_out : std_logic;
signal ir_scan_out : std_logic;
signal br_scan_out : std_logic;
signal ir_data_in : std_logic_vector (11 downto 0);
signal ir_data_out : std_logic_vector (11 downto 0);
signal d'interconnexion controle par le contenu du IR
signal mode_jtag_test1 : std_logic_vector (1 downto 0);
signal mode_jtag_test2 : std_logic_vector (1 downto 0);
signal mode_jtag_test3 : std_logic_vector (1 downto 0);
signal g : std_logic_vector (1 downto 0); pour mux output
signal no_test: std_logic_vector (1 downto 0); pour mux input
signaux d'interconnexion temporaire
signal temp_scan_out : std_logic;
signal pro_scan_out : std_logic; sortie serielle du module analogique
signal del_scan_out : std_logic; sortie serielle du module de delai
signal sel_scan_out : std_logic; sortie serielle du module selftest
-


Debut de l'architecture


-
begin  struct
process servant a faire que TDO change au front descendant
tdo_output : process (tck)
begin  process tdo_output
 activities triggered by rising edge of clock
if tck'event and tck = '0' then
tdo <= scan_out;
```

```
end if;
end process tdo_output;
assignation de signaux pour les multiplexeur de controle
scan_in <= tdi;
no_test <= ir_data_out (5 downto 4); no du test
g <= ir_data_out (5 downto 4); pour mux output
multiplexeur qui selectionne le test ou l'on veut acceder
possibilite de 4 test
le test 4 est reserve au bypass register
11 signifie que le test est desactive
choix_test : process (ir_data_out, no_test)
begin  process choix_test
case no_test is
when "00" =>
mode_jtag_test1 <= ir_data_out (3 downto 2);
mode_jtag_test2 <= "11"; desactive le test
mode_jtag_test3 <= "11"; desactive le test
when "01" =>
mode_jtag_test1 <= "11"; desactive le test
mode_jtag_test2 <= ir_data_out (3 downto 2);
mode_jtag_test3 <= "11"; desactive le test
when "10" =>
mode_jtag_test1 <= "11"; desactive le test
mode_jtag_test2 <= "11"; desactive le test
mode_jtag_test3 <= ir_data_out (3 downto 2);
when "11" =>
mode_jtag_test1 <= "11"; desactive le test
mode_jtag_test2 <= "11"; desactive le test
mode_jtag_test3 <= "11"; desactive le test
when others =>
```

```
mode_jtag_test1 ⇐ "11"; desactive le test
mode_jtag_test2 ⇐ "11"; desactive le test
mode_jtag_test3 ⇐ "11"; desactive le test
end case;
end process choix_test;
G and mode_jtag sont determine par le contenu de IR
U_tc1 : tap_ctrl port map (tms ⇒ tms,
tck ⇒ tck,
reset_n ⇒ reset_n,
shiftIR ⇒ shiftIR,
clockIR ⇒ clockIR,
updateIR ⇒ updateIR,
shiftDR ⇒ shiftDR,
clockDR ⇒ clockDR,
updateDR ⇒ updateDR);
U_ir1 : ir port map ( shiftIR ⇒ shiftIR,
clockIR ⇒ clockIR,
updateIR ⇒ updateIR,
reset_n ⇒ reset_n,
scan_in ⇒ scan_in,
scan_out ⇒ ir_scan_out,
data_in ⇒ ir_data_in,
data_out ⇒ ir_data_out);
U_br1 : br_cell port map (clockDR ⇒ clockDR,
shiftDR ⇒ shiftDR,
scan_in ⇒ scan_in,
scan_out ⇒ br_scan_out);
U_pro : scannable_projet port map (
sam_out ⇒ sam_out,
scope_out ⇒ scope_out,
.
```

```
reset_n ⇒ reset_n,
shiftDR ⇒ shiftDR,
clockDR ⇒ clockDR,
updateDR ⇒ updateDR,
mode_jtag ⇒ mode_jtag_test1,
scan_in ⇒ scan_in,
scan_out ⇒ pro_scan_out);
U_delai : scannable_delai port map(
 pin du component externe sauf jtag
from_ech⇒from_ech,
to_ech⇒to_ech,
enable ⇒enable,
clk_ref ⇒ clk_ref,
reset_n ⇒ reset_n,
end_test ⇒ end_test,
pin JTAG
shiftDR ⇒ shiftDR,
clockDR ⇒ clockDR,
updateDR ⇒ updateDR,
mode_jtag ⇒ mode_jtag_test2,
scan_in ⇒ scan_in,
scan_out ⇒ del_scan_out);
U_self : scannable_self port map(
reset_n ⇒ reset_n ,
shiftDR ⇒ shiftDR,
clockDR ⇒ clockDR,
updateDR ⇒ updateDR,
mode_jtag ⇒ mode_jtag_test3,
selftest_in ⇒ selftest_in,
selftest_out ⇒ selftest_out,
```

```
scan_in ⇒ scan_in,
scan_out ⇒ sel_scan_out);
U_mux1 : mux_4_1 port map (pro_scan_out,
del_scan_out,
sel_scan_out,
br_scan_out,
g, signal 2 bits de commande
temp_scan_out); sortie
U_mux2 : mux_2_1 port map (temp_scan_out,
ir_scan_out,
shiftIR, signal de commande
scan_out); sortie
end struct;
```

```
Data Register P.M.Campbell Nov. 21, 1991

This is a data register which is composed of a chain of
boundary-scan/data register cells. The data register
is described generically to allow any size register to
be generated.

Note that the shift-register loads data into the data
register MSB and shifts data out of the LSB. We assume
that the data register is defined as 'X DOWNTO Y'.


modifie par Olivier Duval Hiver2003

library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
ENTITY dr IS
 le generic permet de specifier la taille de chaque chaine
generic (size_dr : integer range 0 to 32);
PORT (reset_n : in std_logic;
enable_reg : in std_logic;
shiftDR : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
mode_jtag : in std_logic;
scan_in : in std_logic;
scan_out : out std_logic;
data_in : in std_logic_vector (size_dr-1 downto 0);
```

```
data_out : out std_logic_vector (size_dr-1 downto 0));
END dr;
ARCHITECTURE structural OF dr IS
COMPONENT dr_cell PORT (reset_n : in std_logic;
enable_reg : in std_logic;
mode_jtag : in std_logic;
data_in : in std_logic;
shiftDR : in std_logic;
scan_in : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
data_out : out std_logic;
scan_out : out std_logic);
END COMPONENT;
SIGNAL temp_scan : std_logic_vector(data_in'RANGE);
BEGIN
a : FOR i IN data_in'LOW TO data_in'HIGH GENERATE
genration de la 1ere cellule de la chaine
b : IF (i = data_in'HIGH) GENERATE
c : dr_cell PORT MAP (reset_n,
enable_reg,
mode_jtag,
data_in(i),
shiftDR,
scan_in,
clockDR,
updateDR,
data_out(i),
temp_scan(i));
END GENERATE;
```

```
generation des cellules au milieu de la chaine
d : IF ((i ¡ data_in'HIGH) AND (i ¿ 0)) GENERATE
e : dr_cell PORT MAP (reset_n,
enable_reg,
mode_jtag,
data_in(i),
shiftDR,
temp_scan(i+1),
clockDR,
updateDR,
data_out(i),
temp_scan(i));
END GENERATE;
genration de la derniere cellule de la chaine
f : IF (i = 0) GENERATE
g : dr_cell PORT MAP (reset_n,
enable_reg,
mode_jtag,
data_in(i),
shiftDR,
temp_scan(i+1),
clockDR,
updateDR,
data_out(i),
temp_scan(i));
END GENERATE;
END GENERATE;
scan_out <= temp_scan(0);
END structural;
```

```
Data Register Cell P.M.Campbell Nov. 17, 1991

This is a boundary-scan/data register cell.

The schematic for this cell is on page 10-18, fig. 10-16,
of IEEE Std 1149.1-1990.

modifie par Olivier Duval, Hiver2003

library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
ENTITY dr_cell IS
PORT (reset_n : in std_logic;
enable_reg : in std_logic;
mode_jtag : in std_logic;
data_in : in std_logic;
shiftDR : in std_logic;
scan_in : in std_logic;
clockDR : in std_logic;
updateDR : IN std_logic;
data_out : out std_logic;
scan_out : OUT std_logic);
END dr_cell;
ARCHITECTURE behavioral OF dr_cell IS
SIGNAL ff1, ff2 : std_logic;
BEGIN
 process modifiant le contenu du 1er registre JTAG
 le 1er registre JTAG est branche en serie avec la chaine
```

```
 JTAG
dff1 : PROCESS (clockDR,shiftDR,data_in,scan_in,enable_reg,reset_n)
BEGIN
if (reset_n = '0') then
ff1 <= '0'; assigne 0 lors du reset
elsif enable_reg = '1' then
IF (clockDR = '1' and clockDR'EVENT) THEN
IF shiftDR = '0' THEN
ff1 <= data_in; mode normal
ELSE
ff1 <= scan_in; mode JTAG
END IF;
END IF;
end if;
END PROCESS dff1;
process modificant le deuxieme registre JTAG. Ce registre
est branche en parallele avec le circuit
dff2 : PROCESS (updateDR,ff1,reset_n)
BEGIN
if (reset_n = '0') then
ff2 <= '0'; assigne 0 lors du reset
elsif (updateDR = '1') THEN
ff2 <= ff1; load valeur en parallele
END IF;
END PROCESS dff2;
data_out <= data_in WHEN (mode_jtag = '0') ELSE ff2; bypass si
mode normal
scan_out <= ff1; sortie serielle assigne au
premier registre
END behavioral;
```

Instruction Register Cell P.M.Campbell Dec. 6, 1991

This is an instruction register cell.

The schematic for this cell is on page 6-4, fig. 6-1,
of IEEE Std 1149.1-1990.

modifie par Olivier Duval, hiver2003

```
library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
ENTITY ir_cell IS
PORT (shiftIR : in std_logic;
data_in : in std_logic;
scan_in : in std_logic;
clockIR : in std_logic;
updateIR : in std_logic;
reset_bar : in std_logic;
reset_value : in std_logic;
data_out : out std_logic;
scan_out : out std_logic);
END ir_cell;
ARCHITECTURE behavioral OF ir_cell IS
SIGNAL ff1 : std_logic;
BEGIN
Process assignant une valeur au registre serielle
dff : PROCESS (clockIR,
```

```
shiftIR,
data_in,
scan_in,
reset_bar,
reset_value)
BEGIN
if (reset_bar = '0') THEN
ff1 <= reset_value; attribue valeur 1 au reset
elsif (clockIR = '1' and clockIR'EVENT) THEN
IF shiftIR = '0' THEN
ff1 <= data_in; assignation en mode normal
ELSE
ff1 <= scan_in; assignation en mode JTAG
END IF;
END IF;
END PROCESS dff;
scan_out <= ff1;
Process assignant une valeur au registre parallele
dffr : PROCESS (updateIR,
reset_bar,
reset_value)
BEGIN
IF (reset_bar = '0') THEN
data_out <= reset_value; assignation de la valeur 1 au reset
ELSIF ((updateIR = '1') AND updateIR'EVENT) THEN
data_out <= ff1;
END IF;
END PROCESS dffr;
END behavioral;
```

Instruction Register P.M.Campbell Nov. 21, 1991

This is an instruction register which is composed of a
chain of instruction register cells. The instruction
register is described generically to allow any size register
to be generated. Note that the register must contain at
least two stages and the last two stages must always load
a binary '01' value. When the reset_bar signal is asserted,
each IR cell loads the value passed as the reset_value
parameter. This allows the BYPASS instruction or IDCODE
value to be loaded when the Test-Logic-Reset state is
entered.

Note that the shift-register loads data into the instruction
register MSB and shifts data out of the LSB. We assume
that the instruction register is defined as 'X DOWNTO Y'.

modife par Olivier Duval Hiver2003

```
library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
ENTITY ir IS
PORT (shiftIR : in std_logic;
clockIR : in std_logic;
updateIR : in std_logic;
reset_n : in std_logic;
scan_in : in std_logic;
```

```
scan_out : out std_logic;
data_in : in std_logic_vector (11 downto 0);
data_out : out std_logic_vector (11 downto 0));
END ir;
ARCHITECTURE structural OF ir IS
COMPONENT ir_cell PORT (shiftIR : in std_logic;
data_in : in std_logic;
scan_in : in std_logic;
clockIR : in std_logic;
updateIR : in std_logic;
reset_bar : in std_logic;
reset_value : in std_logic;
data_out : out std_logic;
scan_out : out std_logic);
end component;
SIGNAL temp_scan : std_logic_vector (data_in'RANGE);
SIGNAL vdd : std_logic;
SIGNAL gnd : std_logic;
BEGIN
vdd <= '1';
gnd <= '0';
a : FOR i IN data_in'LOW TO data_in'HIGH GENERATE
generation de la 1ere cellule de l'instruction register
b : IF (i = data_in'HIGH) GENERATE
c : ir_cell PORT MAP (shiftIR,
data_in(i),
scan_in,
clockIR,
updateIR,
reset_n,
```

```
vdd,
data_out(i),
temp_scan(i));
END GENERATE;
generation des cellules du milieu de l'instruction register
d : IF ((i ¡ data_in'HIGH) AND (i ¿ 1)) GENERATE
e : ir_cell PORT MAP (shiftIR,
data_in(i),
temp_scan(i+1),
clockIR,
updateIR,
reset_n,
vdd,
data_out(i),
temp_scan(i));
END GENERATE;
generation de l'avant derniere cellule de l'instruction register
f : IF ((i = 1) AND (data_in'HIGH ¿ 1)) GENERATE
h : ir_cell PORT MAP (shiftIR,
gnd, set a 0, voir std IEEE_1149.1
temp_scan(i+1),
clockIR,
updateIR,
reset_n,
vdd,
data_out(i),
temp_scan(i));
END GENERATE;
generation de la derniere cellule de l'instruction register
j : IF (i = 0) GENERATE
```

```
k : ir_cell PORT MAP (shiftIR,
vdd, set a 1, voir std IEEE_1149.1
temp_scan(i+1),
clockIR,
updateIR,
reset_n,
vdd,
data_out(i),
temp_scan(i));
END GENERATE;
END GENERATE ;
scan_out <= temp_scan(0);
END structural;
```

```
Bypass Register Cell P.M.Campbell Nov. 17, 1991

This is a bypass register cell.

The schematic for this cell is on page 9-1, fig. 9-1,
of IEEE Std 1149.1-1990.
modifie par Olivier Duval Hiver2003

library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
ENTITY br_cell IS
PORT (clockDR : in std_logic;
shiftDR : in std_logic;
scan_in : in std_logic;
scan_out : out std_logic);
END br_cell;
ARCHITECTURE behavioral OF br_cell IS
SIGNAL temp : std_logic;
BEGIN
temp <= (shiftDR AND scan_in);
 Process synchrone
 latch la valeur de scan_in dans scan_out
lorsque clockDR=1
dff : PROCESS (clockDR,temp)
BEGIN
IF (clockDR = '1') THEN scan_out <= temp; END IF;
END PROCESS dff;
END behavioral;
```

```
==========================================

Title: scannable_delai
Description: composant structurelle qui rajoute 2 chaines JTAG
 au module de delai.
 1 chaine pour les input : ctrl
 1 chaine pour les output : data

file: scannable_delai.vhd
==========================================

Creation
date Author
1 fevrier 03 Olviier Duval
==========================================

library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
entity scannable_delai is
taille par defaut du registre de ctrl
generic (size_ctrl : integer :=22;
taille par defaut du registre de data
size_dr : integer :=20);
port (
 pin du component externe sauf jtag
from_ech: in std_logic_vector(7 downto 0);
to_ech: in std_logic_vector(7 downto 0);
enable : in std_logic;
clk_ref : in std_logic;
```

```
reset_n : in std_logic;
end_test: out std_logic;
fonction JTAG
shiftDR : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
mode_jtag : in std_logic_vector(1 downto 0);
scan_in : in std_logic;
scan_out : out std_logic);
end scannable_delai;
 purpose: description structurelle du module de delai
architecture struct of scannable_delai is
component module_width20
port(
 signaux d'entrees/sorties du module delai
from_ech: in std_logic_vector(7 downto 0);
to_ech: out std_logic_vector(7 downto 0);
reset_n : in std_logic;
start : in std_logic;
sam : in std_logic_vector( 2 downto 0);
en : in std_logic;
N : out std_logic_vector ( 19 downto 0);
clk_ref : in std_logic ;
reg_load : in std_logic_vector (17 downto 0);
end_test : out std_logic);
end component;
component dr cellule JTAG elementaire pour un data
generic (size_dr : integer range 1 to 32);
port (reset_n : in std_logic;
enable_reg : in std_logic;
```

```
shiftDR : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
mode_jtag : in std_logic;
scan_in : in std_logic;
scan_out : out std_logic;
data_in : in std_logic_vector (size_dr-1 downto 0);
data_out : out std_logic_vector (size_dr-1 downto 0)
);
end component;
signaux de mapping entre registre JTAG et module delai
signal dr_data_in : std_logic_vector (size_dr-1 downto 0);
signal dr_data_out : std_logic_vector (size_dr-1 downto 0);
signal ctrl_data_in : std_logic_vector (size_ctrl-1 downto 0);
signal ctrl_data_out : std_logic_vector (size_ctrl-1 downto 0);
signaux de controle du multiplexeur de chaine de registre
signal enable_reg_dr : std_logic;
signal enable_reg_ctrl : std_logic;
signal scan_out_dr : std_logic;
signal scan_out_ctrl : std_logic;
signaux d'initialisation a 1
SIGNAL vdd : std_logic;
begin  struct
vdd <= '1';
process qui determine dans quelle registre on va lire ou ecrire
choix_reg : process (mode_jtag,scan_out_dr,scan_out_ctrl)
begin
multiplexeur qui selectionne le registre ou l'on veut access
possibilite de 4 registres
case mode_jtag is
```

```
when "00" ⇒
enable_reg_dr ⇐ '0';
enable_reg_ctrl ⇐ '1'; registre de controle selctionne
scan_out ⇐ scan_out_ctrl;
when "01" ⇒
enable_reg_dr ⇐ '1'; registre de data selctionne
enable_reg_ctrl ⇐ '0';
scan_out ⇐ scan_out_dr;
when "10" ⇒
enable_reg_dr ⇐ '0';
enable_reg_ctrl ⇐ '0';
scan_out ⇐ '0';
when "11" ⇒
enable_reg_dr ⇐ '0';
enable_reg_ctrl ⇐ '0';
scan_out ⇐ '0';
when others ⇒
enable_reg_dr ⇐ '0';
enable_reg_ctrl ⇐ '0';
scan_out ⇐ '0';
end case;
end process choix_reg;
Mapping des composants
U_delai1 : module_width20
port map (from_ech⇒ from_ech,
to_ech⇒to_ech,
reset_n ⇒ reset_n,
start ⇒enable,
sam ⇒ ctrl_data_out (2 downto 0),
en ⇒ ctrl_data_out (21),
```

```
N ⇒ dr_data_in,
clk_ref ⇒ clk_ref,
reg_load ⇒ctrl_data_out (20 downto 3),
end_test ⇒ end_test);
U_dr_1 : dr
generic map (size_dr ⇒ size_dr) chaine du data register
port map (reset_n ⇒ reset_n,
enable_reg ⇒ enable_reg_dr,
shiftDR ⇒ shiftDr,
clockDR ⇒ clockDr,
updateDR ⇒ updateDR,
mode_jtag ⇒ vdd,
scan_in ⇒ scan_in,
scan_out ⇒ scan_out_dr,
data_in ⇒ dr_data_in,
data_out ⇒ dr_data_out);
U_ctrl_1 : dr
generic map (size_dr ⇒ size_ctrl) chaine du ctrl register
port map (reset_n ⇒ reset_n,
enable_reg ⇒ enable_reg_ctrl,
shiftDR ⇒ shiftDr,
clockDR ⇒ clockDr,
updateDR ⇒ updateDR,
mode_jtag ⇒ vdd,
scan_in ⇒ scan_in,
scan_out ⇒ scan_out_ctrl,
data_in ⇒ ctrl_data_in,
data_out ⇒ ctrl_data_out);
end struct;
```

```
==========================================

Title: tap_controller
 (test_access_port controller)
Description: composant rtl qui elabore la machine a etat JTAG


file: tap_ctrl.vhd
==========================================

Creation
date Author
1 fevrier 03 Olivier Duval
==========================================

library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
entity tap_ctrl is
port (
 pin JTAG de base
tck : in std_logic;
tms : in std_logic;
reset_n : in std_logic;
shiftIR : out std_logic; indique que shift le inst reg
clockIR : out std_logic; indique hold register passe
updateIR : out std_logic;
shiftDR : out std_logic; indique que l'on shift DR
clockDR : out std_logic; indique hold register passe
updateDR : out std_logic);
```

```
END tap_ctrl;
architecture behavior of tap_ctrl is
assignation des etats
conforme aux standards IEEE1149.1
constant test_logic_reset : std_logic_vector(3 downto 0) := "0000";
constant run_test_idle : std_logic_vector(3 downto 0) := "0001";
constant select_dr_scan : std_logic_vector(3 downto 0) := "0010";
constant select_ir_scan : std_logic_vector(3 downto 0) := "0011";
constant capture_ir_scan : std_logic_vector(3 downto 0) := "0100";
constant shift_ir : std_logic_vector(3 downto 0) := "0101";
constant exit1_ir : std_logic_vector(3 downto 0) := "0110";
constant pause_ir : std_logic_vector(3 downto 0) := "0111";
constant exit2_ir : std_logic_vector(3 downto 0) := "1000";
constant update_ir : std_logic_vector(3 downto 0) := "1001";
constant capture_dr_scan : std_logic_vector(3 downto 0) := "1010";
constant shift_dr : std_logic_vector(3 downto 0) := "1011";
constant exit1_dr : std_logic_vector(3 downto 0) := "1100";
constant pause_dr : std_logic_vector(3 downto 0) := "1101";
constant exit2_dr : std_logic_vector(3 downto 0) := "1110";
constant update_dr : std_logic_vector(3 downto 0) := "1111";
signaux d'etats
signal jtag_state : std_logic_vector(3 downto 0);
signal next_jtag_state : std_logic_vector (3 downto 0);
begin
process servant aux changement d'etat au coup d'horloge
synchrone : process (tck,reset_n)
begin  process synchrone
if reset_n = '0' then
jtag_state <= test_logic_reset;
elsif tck'event and tck = '1' then
```

```
jtag_state ⇐ next_jtag_state;
end if;
end process synchrone;
process servvant a definir les sorties en fonctions de l'etat present
assignation : process (jtag_state,tck)
begin
IF jtag_state = shift_ir THEN
shiftIR ⇐ '1';
ELSE
shiftIR ⇐ '0';
END IF;
IF jtag_state = shift_dr THEN
shiftDR ⇐ '1';
ELSE
shiftDR ⇐ '0';
END IF;
if tck = '0' AND (jtag_state = capture_ir_scan
or jtag_state = shift_ir) then
clockIR ⇐ '0';
else
clockIR ⇐ '1';
end if;
if tck = '0' AND (jtag_state = capture_dr_scan
or jtag_state = shift_dr) then
clockDR ⇐ '0';
else
clockDR ⇐ '1';
end if;
if tck = '0' AND jtag_state = update_ir then
updateIR ⇐ '1';
```

```
else
updateIR ⇐ '0';
end if;
if tck = '0' AND jtag_state = update_dr then
updateDR ⇐ '1';
else
updateDR ⇐ '0';
end if;
end process assignation;
-
process servant a trouver le prochain etat en fonctiOon de
l'entree TMS
-
asynchrone : process (tms,jtag_state)
begin
case jtag_state is
when test_logic_reset ⇒
if tms = '1' then
next_jtag_state ⇐ test_logic_reset;
else
next_jtag_state ⇐ run_test_idle;tms = 0
end if;
when run_test_idle ⇒
if tms = '1' then
next_jtag_state ⇐ select_dr_scan;
else
next_jtag_state ⇐ run_test_idle; tms = 0
end if;
when select_dr_scan ⇒
if tms = '1' then
```

next_jtag_state ⇐ select_ir_scan;

else

next_jtag_state ⇐ capture_dr_scan; tms = 0

end if;

when select_ir_scan ⇒

if tms = '1' then

next_jtag_state ⇐ test_logic_reset;

else

next_jtag_state ⇐ capture_ir_scan; tms = 0

end if;

DESCRIPTION des etats associes au traitement du INstruction RESGITER

-

when capture_ir_scan ⇒

if tms = '1' then

next_jtag_state ⇐ exit1_ir;

else

next_jtag_state ⇐ shift_ir; tms = 0

end if;

when shift_ir ⇒

if tms = '1' then

next_jtag_state ⇐ exit1_ir;

else

next_jtag_state ⇐ shift_ir; tms = 0

end if;

when exit1_ir ⇒

if tms = '1' then

next_jtag_state ⇐ update_ir;

```
else
next_jtag_state ⇐ pause_ir;  tms = 0
end if;
when pause_ir ⇒
if tms = '1' then
next_jtag_state ⇐ exit2_ir;
else
next_jtag_state ⇐ pause_ir;  tms = 0
end if;
when exit2_ir ⇒
if tms = '1' then
next_jtag_state ⇐ update_ir;
else
next_jtag_state ⇐ shift_ir;  tms = 0
end if;
when update_ir ⇒
if tms = '1' then
next_jtag_state ⇐ select_dr_scan;
else
next_jtag_state ⇐ run_test_idle; tms = 0
end if;
```

DESCRIPTION des etats associes au traitement des registres de DATA

```
-
when capture_dr_scan ⇒
if tms = '1' then
next_jtag_state ⇐ exit1_dr;
else
```

```
next_jtag_state ⇐ shift_dr;  tms = 0
end if;
when shift_dr ⇒
if tms = '1' then
next_jtag_state ⇐ exit1_dr;
else
next_jtag_state ⇐ shift_dr;  tms = 0
end if;
when exit1_dr ⇒
if tms = '1' then
next_jtag_state ⇐ update_dr;
else
next_jtag_state ⇐ pause_dr;  tms = 0
end if;
when pause_dr ⇒
if tms = '1' then
next_jtag_state ⇐ exit2_dr;
else
next_jtag_state ⇐ pause_dr;  tms = 0
end if;
when exit2_dr ⇒
if tms = '1' then
next_jtag_state ⇐ update_dr;
else
next_jtag_state ⇐ shift_dr;  tms = 0
end if;
when update_dr ⇒
if tms = '1' then
next_jtag_state ⇐ select_dr_scan;
else
```

```
next_jtag_state ⇐ run_test_idle; tms = 0
end if;
when others ⇒
next_jtag_state ⇐ test_logic_reset;
end case;
end process asynchrone;
end behavior;
```

```
==========================================

Title: compteur
 (test_access_port controller)
Description: compteur pour le timer de test


file: compteur.vhd
==========================================

Creation
date Author
1 fevrier 03 Olivier Duval
==========================================

library ieee;
library work;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
entity compteur is
port ( clk,sload,clk_enable : in std_logic;
D : in std_logic_vector(31 downto 0); mettre le compteur
pres du registre de timer
timer_end : out std_logic);
end compteur;
architecture archi of compteur is
signal tmp : std_logic_vector(31 downto 0);
begin  behav
 purpose: total
 type: memoryless
```

```
 in
 purpose: the clocked process
 type: memorizing
 inputs: clk, mode
 outputs:
Compteur : process (clk)
begin
if(clk'event and clk='1') then
if sload='1' then
tmp⇐D;
timer_end⇐'0';
else
if clk_enable ='1' then
if tmp/= "00000000000000000000000000000000" then
tmp⇐tmp-1;
timer_end⇐'0';
else
timer_end⇐'1';
end if;
end if;
end if;
end if;
end process compteur;
end archi ;
```

```
==========================================

Title: scannable_self
Description: composant structurelle qui rajoute 2 chaines JTAG
 au module de auto_test
 1 chaine pour les input : ctrl
 1 chaine pour les output : data

file: scannable_self.vhd
==========================================

Creation
date Author
1 fevrier 03 Olivier Duval
==========================================

library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
entity scannable_self is
taille par defaut du registre de ctrl
generic (size_ctrl : integer :=8;
taille par defaut du registre de data
size_dr : integer :=8);
port (
reset_n : in std_logic;
fonction JTAG
shiftDR : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
```

```
mode_jtag : in std_logic_vector(1 downto 0);
selftest_in : in std_logic_vector (size_ctrl-1 downto 0);
selftest_out : out std_logic_vector (size_ctrl-1 downto 0);
scan_in : in std_logic;
scan_out : out std_logic);
end scannable_self;
 purpose: description structurelle du module selft_test
architecture struct of scannable_self is
component selftest
port ( ctrl : in std_logic_vector(7 downto 0);
data_out : out std_logic_vector (7 downto 0));
end component;
component dr
generic (size_dr : integer range 1 to 32);
port (reset_n : in std_logic;
enable_reg : in std_logic;
shiftDR : in std_logic;
clockDR : in std_logic;
updateDR : in std_logic;
mode_jtag : in std_logic;
scan_in : in std_logic;
scan_out : out std_logic;
data_in : in std_logic_vector (size_dr-1 downto 0);
data_out : out std_logic_vector (size_dr-1 downto 0)
);
end component;
signaux de mapping entre registre JTAG et module delai
signal dr_data_in : std_logic_vector (size_dr-1 downto 0);
signal dr_data_out : std_logic_vector (size_dr-1 downto 0);
signaux de controle du multiplexeur de chaine de registre
```

```
signal enable_reg_dr : std_logic;
signal enable_reg_ctrl : std_logic;
signal scan_out_dr : std_logic;
signal scan_out_ctrl : std_logic;
signaux d'initialisation a 1
signal vdd : std_logic;
begin  struct
vdd <= '1';
process qui determine dans quelle registre on va lire ou ecrire
choix_reg : process (mode_jtag,scan_out_dr,scan_out_ctrl)
begin
multiplexeur qui selectionne le registre ou l'on veut access
possibilite de 4 registres
case mode_jtag is
when "00" =>
enable_reg_dr <= '0';
enable_reg_ctrl <= '1'; registre de controle selctionne
scan_out <= scan_out_ctrl;
when "01" =>
enable_reg_dr <= '1'; registre de data selctionne
enable_reg_ctrl <= '0';
scan_out <= scan_out_dr;
when "10" =>
enable_reg_dr <= '0';
enable_reg_ctrl <= '0';
scan_out <= '0';
when "11" =>
enable_reg_dr <= '0';
enable_reg_ctrl <= '0';
scan_out <= '0';
```

```
when others ⇒
enable_reg_dr ⇐ '0';
enable_reg_ctrl ⇐ '0';
scan_out ⇐ '0';
end case;
end process choix_reg;
Mapping des composants
 Not used anymore, the connection to analog part is made
 through i/o pins
 U_delai1 : selftest
 port map (ctrl ⇒ ctrl_data_out (7 downto 0),
 data_out ⇒ dr_data_in);
U_dr_1 : dr chaine du data register
generic map (size_dr ⇒ size_dr)
port map (reset_n ⇒ reset_n,
enable_reg ⇒ enable_reg_dr,
shiftDR ⇒ shiftDr,
clockDR ⇒ clockDr,
updateDR ⇒ updateDR,
mode_jtag ⇒ vdd,
scan_in ⇒ scan_in,
scan_out ⇒ scan_out_dr,
data_in ⇒ selftest_in,
data_out ⇒ dr_data_out);
U_ctrl_1 : dr chaine du ctrl register
generic map (size_dr ⇒ size_ctrl)
port map (reset_n ⇒ reset_n,
enable_reg ⇒ enable_reg_ctrl,
shiftDR ⇒ shiftDr,
clockDR ⇒ clockDr,
```

```
updateDR ⇒ updateDR,
mode_jtag ⇒ vdd,
scan_in ⇒ scan_in,
scan_out ⇒ scan_out_ctrl,
data_in ⇒ ctrl_data_in,
data_out ⇒ selftest_out);
end struct;
-
```

Multiplexeur 2:1

Il s'agit d'un multiplexeur 2:1
La sortie est l'entree 1 si ctrl(g) = 00
La sortie est l'entree 2 si ctrl(g) = 01
La sortie est l'entree 3 si ctrl(g) = 10
La sortie est l'entree 4 si ctrl(g) = 11

```
-
library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
ENTITY mux_4_1 IS
PORT (a : in std_logic;
b : in std_logic;
c : in std_logic;
d : in std_logic;
g : in std_logic_vector (1 DOWNTO 0);
z : out std_logic);
END mux_4_1;
```

```
ARCHITECTURE behavioral OF mux_4_1 IS
BEGIN
PROCESS (a, b, c, d, g)
BEGIN
CASE g IS
WHEN "00" ⇒ z ⇐ a; sortie z=a si in =00
WHEN "01" ⇒ z ⇐ b; sortie z=b si in =01
WHEN "10" ⇒ z ⇐ c; sortie z=c si in =10
WHEN "11" ⇒ z ⇐ d; sortie z=d si in =11
when others ⇒ z ⇐ a;
END CASE;
END PROCESS;
END behavioral;
```

-

Multiplexeur 2:1

Il s'agit d'un multiplexeur 2:1. La sortie est l'entre 1 si ctrl(g) = 0
et la sortie est l'entree 2 si ctrl(g) = 1

-

```
library IEEE;
library WORK;
use IEEE.std_logic_1164.all;
ENTITY mux_2_1 IS
PORT (a : in std_logic;
b : in std_logic;
g1 : in std_logic;
z : out std_logic);
```

```
END mux_2_1;
ARCHITECTURE behavioral OF mux_2_1 IS
BEGIN
PROCESS (a, b, g1)
BEGIN
IF (g1 = '0') THEN z ⇐ a; sortie z=a si in =0
ELSIF (g1 = '1') THEN z ⇐ b; sortie z=b si in =1
END IF;
END PROCESS;
END behavioral;
```

```
-
dcModule.vhd
Ce module controle les parametres de test du module DC
Auteur: Olivier Duval 10 fevrier 2003
-
library ieee;
library WORK;
use ieee.std_logic_1164.all;
entity anal_unit is
port ( clk : in std_logic;
ech : in std_logic_vector(2 downto 0);
enable : in std_logic;
controle : in std_logic_vector(15 downto 0);
mode : in std_logic_vector(2 downto 0);
timer_end : in std_logic; mettre le compteur
pres du registre de timer
adc_out: in std_logic_vector(7 downto 0);
adc_end: in std_logic;
end_test: out std_logic;
vout : out std_logic_vector(7 downto 0));
end anal_unit;
architecture behav of anal_unit is
constant idle : std_logic_vector(1 downto 0) := "00";
constant testing : std_logic_vector(1 downto 0) := "01";
constant digitize : std_logic_vector(1 downto 0) := "10";
constant read : std_logic_vector(1 downto 0) := "11";
signal state : std_logic_vector(1 downto 0);
```

```
signal next_state : std_logic_vector (1 downto 0);
begin  behav
 purpose: total
 type: memoryless
 in
 purpose: the clocked process
 type: memorizing
 inputs: clk, mode
 outputs:
Clock : process (clk, mode)
begin  process Clock
 activities triggered by asynchronous reset (active low)
if mode = "111" then
vout <= (others => '0');
end_test <= '0';
state <= idle;
 activities triggered by rising edge of clock
elsif clk'event and clk = '1' then
state <= next_state;
end if;
end process Clock;
state_machine : process (enable, timer_end, adc_end, state )
begin  process state_machine
case state is
when idle =>
if enable = '1' then
next_state <= testing;
else
next_state <= idle;
end if;
```

```
end_test<='0';
vout<= (others => '0');
when testing =>
if timer_end = '1' then
next_state <= digitize;
else
next_state <= testing;
end if;
end_test<='0';
vout<= (others => '0');
when digitize =>
if adc_end = '1' then
next_state <= read;
else
next_state <= digitize;
end if;
end_test<='0';
vout<= controle(7 downto 0) and controle(15 downto 8);
when read =>
if enable = '0' then
next_state <= idle;
else
next_state <= read;
end if;
end_test<='1';
vout<= adc_out;
when others =>
if enable = '1' then
next_state <= testing;
else
```

```
next_state ⇐ idle;
end if;
end_test⇐'0';
vout⇐ (others ⇒ '0');
end case;
end process state_machine;
end behav ;
```

```
==========================================

TITLE : BANC D'ESSAI DU CODE SYNTHETISABLE DE LA NANOPLATEFORME
DESCRIPTION :Description VHDL des vecteurs de test "testbench"
de la nano-plateforme synthetisable.
Tous les signaux fonctionnels de la nanoplateforme
y sont exposes et testes.
FILE : plateforme_tb.vhd
==========================================

CREATION
DATE AUTEURS
22mars2003 Olivier Duval

==========================================

library IEEE;
library WORK;
use ieee.std_logic_1164.all;
entity nano_plateforme_tb is
end nano_plateforme_tb;
architecture behav of nano_plateforme_tb is
component testable_scannable_projet
port (
rst_n : in std_logic;
tck : in std_logic;
tms : in std_logic;
tdi : in std_logic;
```

```
tdo : out std_logic;
clk_ref : in std_logic;
enable : in std_logic;
sam_out : out std_logic_vector(15 downto 0);
scope_out : out std_logic_vector(15 downto 0);
end_test: out std_logic
);
end component;
constant CLK_PERIOD : time := 10 ns;
signal tck : std_logic := '1';
signal rst_n : std_logic;
signal enable : std_logic :='0';
signal tms : std_logic := '0';
signal tdi : std_logic := '0';
signal tdo : std_logic := '0';
signal clk_ref : std_logic;
signal sam_out : std_logic_vector(15 downto 0);
signal scope_out : std_logic_vector(15 downto 0);
signal end_test: std_logic;
signal mode : std_logic_vector(2 downto 0) := "000";
begin
tck <= not tck after (CLK_PERIOD/2);
DUT: testable_scannable_projet
port map (
tck=>tck,
tms=>tms,
tdi=>tdi,
tdo=>tdo,
```

```
rst_n=>rst_n,
sam_out=>sam_out,
scope_out=>scope_out,
enable=>enable,
clk_ref=>clk_ref,
end_test=>end_test
);
-
 PROCESS tdi_setup
 signaux affectes : tdi, rst_n
-
do_tdi_setup : process
begin
rst_n <= '1'; wait for CLK_PERIOD;
rst_n <= '0'; wait for CLK_PERIOD;
rst_n <= '1'; wait for CLK_PERIOD;
wait for 9*CLK_PERIOD;
-********************** MODULE ANALOGIQUE
tdi <= '1'; wait for CLK_PERIOD; (0) Default value, not used
tdi <= '0'; wait for CLK_PERIOD; (1) Default value, not used
tdi <= '0'; wait for CLK_PERIOD; (2) Numero du test dans le
tdi <= '0'; wait for CLK_PERIOD; (3) module pointe
tdi <= '0'; wait for CLK_PERIOD; (4) Numero du registre/module
tdi <= '0'; wait for CLK_PERIOD; (5) pointe (00:analogique)
tdi <= '1'; wait for CLK_PERIOD;  Not used
tdi <= '1'; wait for CLK_PERIOD;  Not used
tdi <= '1'; wait for CLK_PERIOD;  Not used
tdi <= '1'; wait for CLK_PERIOD;  Not used
```

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

Charges dans ir [instruction register]

[module select] g := (5)&(4) = "00" [reg./mod. analogique]

[test select] := (3)&(2) = "00"

⇒ mux_4_1 ⇐ pro_scan_out [scan out mod. analog.]

mode_jtag_test1 [analog.] ⇐ ir_data_out (3)&(2);

mode_jtag_test2 [delai] ⇐ "11" [OFF];

mode_jtag_test3 [self test] ⇐ "11" [OFF];

⇒ Dans scan_proj :

enable_reg_ctrl ⇐ '1';

scan_out ⇐ scan_out_ctrl;

tdi ⇐ '0';wait for 6*CLK_PERIOD;

tdi@0

tdi ⇐ '1'; wait for CLK_PERIOD; (0)

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD; (3)

tdi ⇐ '1'; wait for CLK_PERIOD; (4)

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD; (7)

Charges dans ctrl_data_out du mod. analog.

sam [in] ⇒ ctrl_data_out(3 downto 0)

scope [in] ⇒ ctrl_data_out(7 downto 4)

tdi ⇐ '0';wait for 6*CLK_PERIOD;

tdi@0

-********** MODULE DELAI 1/2

tdi ⇐ '1'; wait for CLK_PERIOD; (0) Default value, not used

tdi ⇐ '0'; wait for CLK_PERIOD; (1) Default value, not used

tdi ⇐ '0'; wait for CLK_PERIOD; (2) Numero du test dans le

tdi ⇐ '0'; wait for CLK_PERIOD; (3) module pointe

tdi ⇐ '1'; wait for CLK_PERIOD; (4) Numero du registre/module

tdi ⇐ '0'; wait for CLK_PERIOD; (5) pointe (01:delai)

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

Charges dans ir [instruction register]

[module select] g := (5)&(4) = "01" [reg./mod. delai]

[test select] := (3)&(2) = "00"

⇒ mux_4_1 ⇐ del_scan_out [scan out mod. delai]

mode_jtag_test1 [analog.] ⇐ "11" [OFF];

mode_jtag_test2 [delai] ⇐ ir_data_out (3)&(2);

mode_jtag_test3 [self test] ⇐ "11" [OFF];

⇒ Dans scan_del :

enable_reg_dr ⇐ '0';

enable_reg_ctrl ⇐ '1';

scan_out ⇐ scan_out_ctrl;

tdi ⇐ '0';wait for 6*CLK_PERIOD;

tdi@0

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

Charges dans ctrl_data_out du mod. delai

sam [in] ⇒ ctrl_data_out(21 downto 0)

en [in] ⇒ ctrl_data_out(22)

tdi ⇐ '0';wait for 6*CLK_PERIOD;

tdi@0

-********** MODULE DELAI 2/2 tdi ⇐ '1'; wait for CLK_PERIOD;

(0) Default value, not used

tdi ⇐ '0'; wait for CLK_PERIOD; (1) Default value, not used

tdi ⇐ '1'; wait for CLK_PERIOD; (2) Numero du test dans le

tdi ⇐ '0'; wait for CLK_PERIOD; (3) module pointe

tdi ⇐ '1'; wait for CLK_PERIOD; (4) Numero du registre/module

tdi ⇐ '0'; wait for CLK_PERIOD; (5) pointe (01:delai)

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

Charges dans ir [instruction register]

[module select] g := (5)&(4) = "01" [reg./mod. delai]

[test select] := (3)&(2) = "01"

⇒ mux_4_1 ⇐ del_scan_out [scan out mod. delai]

mode_jtag_test1 [analog.] ⇐ "11" [OFF];

mode_jtag_test2 [delai] ⇐ ir_data_out (3)&(2);

mode_jtag_test3 [self test] ⇐ "11" [OFF];

⇒ Dans scan_del :

enable_reg_dr ⇐ '1';

enable_reg_ctrl ⇐ '0';

scan_out ⇐ scan_out_dr;

tdi ⇐ '0';wait for 6*CLK_PERIOD;

tdi@0

tdi ⇐ '1'; wait for 21*CLK_PERIOD;

Vider data_reg

tdi ⇐ '0';wait for 8*CLK_PERIOD;

tdi@0

-********** MODULE SELF-TEST 1/2

tdi ⇐ '1'; wait for CLK_PERIOD; (0) Default value, not used

tdi ⇐ '0'; wait for CLK_PERIOD; (1) Default value, not used

tdi ⇐ '0'; wait for CLK_PERIOD; (2) Numero du test dans le

tdi ⇐ '0'; wait for CLK_PERIOD; (3) module pointe

tdi ⇐ '0'; wait for CLK_PERIOD; (4) Numero du registre/module

tdi ⇐ '1'; wait for CLK_PERIOD; (5) pointe (10:self-test)

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

Charges dans ir [instruction register]

[module select] g := (5)&(4) = "10" [reg./mod. self-test]

[test select] := (3)&(2) = "00"

⇒ mux_4_1 ⇐ del_scan_out [scan out mod. delai]

mode_jtag_test1 [analog.] ⇐ "11" [OFF];

mode_jtag_test2 [delai] ⇐ "11" [OFF];

mode_jtag_test3 [self test] ⇐ ir_data_out (3)&(2);

⇒ Dans scan_del :

enable_reg_dr ⇐ '0';

enable_reg_ctrl ⇐ '1';

scan_out ⇐ scan_out_ctrl;

tdi ⇐ '0';wait for 7*CLK_PERIOD;

tdi@0

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '0'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

Charges dans ctrl_data_out du mod. self-test

ctrl [in] ⇒ ctrl_data_out(7 downto 0)

tdi ⇐ '0';wait for 6*CLK_PERIOD;

tdi@0

-********** MODULE SELF-TEST 2/2

tdi ⇐ '1'; wait for CLK_PERIOD; (0) Default value, not used

tdi ⇐ '0'; wait for CLK_PERIOD; (1) Default value, not used

tdi ⇐ '1'; wait for CLK_PERIOD; (2) Numero du test dans le

tdi ⇐ '0'; wait for CLK_PERIOD; (3) module pointe

tdi ⇐ '0'; wait for CLK_PERIOD; (4) Numero du registre/module

tdi ⇐ '1'; wait for CLK_PERIOD; (5) pointe (10:self-test)

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

Charges dans ir [instruction register]

[module select] g := (5)&(4) = "10" [reg./mod. self-test]

[test select] := (3)&(2) = "01"

⇒ mux_4_1 ⇐ del_scan_out [scan out mod. delai]

mode_jtag_test1 [analog.] ⇐ "11" [OFF];

mode_jtag_test2 [delai] ⇐ "11" [OFF];

mode_jtag_test3 [self test] ⇐ ir_data_out (3)&(2);

⇒ Dans scan_del :

enable_reg_dr ⇐ '1';

enable_reg_ctrl ⇐ '0';

scan_out ⇐ scan_out_dr;

tdi ⇐ '0';wait for 7*CLK_PERIOD;

tdi@0

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

tdi ⇐ '1'; wait for CLK_PERIOD;

Charges dans ctrl_data_out du mod. self-test

ctrl [in] ⇒ ctrl_data_out(7 downto 0)

tdi ⇐ '0';wait for 6*CLK_PERIOD;

tdi@0

-********** MODULE BYPASS

tdi ⇐ '1'; wait for CLK_PERIOD; (0) Default value, not used

tdi ⇐ '0'; wait for CLK_PERIOD; (1) Default value, not used

tdi ⇐ '1'; wait for CLK_PERIOD; (2) Numero du test dans le

tdi ⇐ '0'; wait for CLK_PERIOD; (3) module pointe

tdi ⇐ '1'; wait for CLK_PERIOD; (4) Numero du registre/module

tdi ⇐ '1'; wait for CLK_PERIOD; (5) pointe (11:Aucun, Bypass)

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

tdi ⇐ '1'; wait for CLK_PERIOD; Not used

Charges dans ir [instruction register]

[module select] g := (5)&(4) = "11" [Bypass]

[test select] := (3)&(2) = "00"

⇒ mux_4_1 ⇐ br_scan_out [scan out mod. bypass]

```
mode_jtag_test1 [analog.] <= "11" [OFF];
mode_jtag_test2 [delai] <= "11" [OFF];
mode_jtag_test3 [self test] <= "11" [OFF];

tdi <= '0';wait for 7*CLK_PERIOD;
tdi@0
tdi <= '1'; wait for CLK_PERIOD;
tdi <= '0'; wait for CLK_PERIOD;
tdi <= '1'; wait for CLK_PERIOD;
tdi <= '1'; wait for CLK_PERIOD;
tdi <= '0'; wait for CLK_PERIOD;
tdi <= '0'; wait for CLK_PERIOD;
tdi <= '1'; wait for CLK_PERIOD;
tdi <= '0'; wait for CLK_PERIOD;
tdi <= '1'; wait for CLK_PERIOD;
tdi <= '0'; wait for CLK_PERIOD;
tdi <= '1'; wait for CLK_PERIOD;
tdi <= '1'; wait for CLK_PERIOD;
tdi <= '0'; wait for CLK_PERIOD;
tdi <= '0'; wait for CLK_PERIOD;
tdi <= '1'; wait for CLK_PERIOD;
tdi <= '0'; wait for CLK_PERIOD;
tdi se reflete directement sur tdo

tdi <= '0';wait for 6*CLK_PERIOD;
tdi@0
-**********************
tdi <= '0';wait for 50*CLK_PERIOD;
```

```
 tdi@0
end process do_tdi_setup;

 Etat dans lequel mene la lecture du bit tms
do_tms_setup : process
begin
wait for 2*CLK_PERIOD;
tms <= '1'; wait for 5*CLK_PERIOD;
 Test-logic-reset
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 Run-test-idle
**********************************************
 ATTENTION
 Code relatif au module analogique

**********************************************
tms <= '1'; wait for 1*CLK_PERIOD;
 select-DR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-IR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** IR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
```

```
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 Shift_IR (avant dernier bit passe)
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 Exit1_IR
 Le dernier bit de donnee est passe
tms <= '1'; wait for 1*CLK_PERIOD;
 Pause_IR
tms <= '1'; wait for 1*CLK_PERIOD;
 Exit2_IR
tms <= '1'; wait for 1*CLK_PERIOD;
 update-IR
tms <= '0'; wait for 1*CLK_PERIOD;
 select-DR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
 capture-DR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-DR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** DR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
```

```
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-DR (avant dernier bit passe)
tms <= '1'; wait for 1*CLK_PERIOD; tms <= '1'; wait for 1*CLK_PERIOD;  Le dernier
bit de donnee est passe
 Exit1_DR
**********************************************
 ATTENTION
 Code relatif au module de mesure
 de delai.
 1
**********************************************
tms <= '1'; wait for 1*CLK_PERIOD; tms <= '1'; wait for 1*CLK_PERIOD;
 select-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 capture-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-IR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** IR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
```

```
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 Exit1-IR
 Le dernier bit de donnee est passee
tms <= '1'; wait for 1*CLK_PERIOD;
 Pause-IR
tms <= '1'; wait for 1*CLK_PERIOD;
 Update-IR
tms <= '1'; wait for 1*CLK_PERIOD;
 select-DR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-DR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** DR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
```

```
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 Exit1_DR
 Le dernier bit de donnee est passe
**********************************************
 ATTENTION
 Code relatif au module de mesure
 de delai.
 2
**********************************************
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 select-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 capture-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
```

```
shift-IR
tms <= '0'; wait for 1*CLK_PERIOD;
Le premier bit de donnee est passe ** IR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
Exit1-IR
Le dernier bit de donnee est passee
tms <= '1'; wait for 1*CLK_PERIOD;
Pause-IR
tms <= '1'; wait for 1*CLK_PERIOD;
Update-IR
tms <= '1'; wait for 1*CLK_PERIOD;
select-DR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
shift-DR
tms <= '0'; wait for 1*CLK_PERIOD;
```

Le premier bit de donnee est passe ** DR

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '0'; wait for 1*CLK_PERIOD;

tms ⇐ '1'; wait for 1*CLK_PERIOD;

tms ⇐ '1'; wait for 1*CLK_PERIOD;

Exit1_DR

Le dernier bit de donnee est passe

****

ATTENTION

Code relatif au module d'auto-test (self-test)

```
1
*********************************************
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 select-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 capture-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-IR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** IR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 Exit1-IR
 Le dernier bit de donnee est passee
tms <= '1'; wait for 1*CLK_PERIOD;
 Pause-IR
```

```
tms <= '1'; wait for 1*CLK_PERIOD;
 Update-IR
tms <= '1'; wait for 1*CLK_PERIOD;
 select-DR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-DR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** DR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 Exit1_DR
 Le dernier bit de donnee est passe
**********************************************
 ATTENTION
 Code relatif au module d'auto-test (self-test)
 2
**********************************************
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 select-IR-scan
```

```
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 capture-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-IR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** IR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 Exit1-IR
 Le dernier bit de donnee est passee
tms <= '1'; wait for 1*CLK_PERIOD;
 Pause-IR
tms <= '1'; wait for 1*CLK_PERIOD;
 Update-IR
tms <= '1'; wait for 1*CLK_PERIOD;
 select-DR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
```

```
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-DR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** DR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 Exit1_DR
 Le dernier bit de donnee est passe
**********************************************
 ATTENTION
 Code relatif au module bypass

**********************************************
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 select-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 capture-IR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-IR
```

```
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** IR
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 Exit1-IR
 Le dernier bit de donnee est passee
tms <= '1'; wait for 1*CLK_PERIOD;
 Pause-IR
tms <= '1'; wait for 1*CLK_PERIOD;
 Update-IR
tms <= '1'; wait for 1*CLK_PERIOD;
 select-DR-scan
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
 shift-DR
tms <= '0'; wait for 1*CLK_PERIOD;
 Le premier bit de donnee est passe ** DR
```

```
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '0'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
tms <= '1'; wait for 1*CLK_PERIOD;
 Exit1_DR
 Le dernier bit de donnee est passe
tms <= '1'; wait for 52*CLK_PERIOD;
end process do_tms_setup;
end behav;
configuration cfg_nano_plateforme_tb of nano_plateforme_tb is
for behav
end for;
end cfg_nano_plateforme_tb;
```

ANNEXE II

CODE SCHEME DU MODÈLE DE CURTICE ET ETTENBERG

ANNEXE 2 : Code source du programme du chapitre 3

```
;;
; cntfet_model.scm ;
; ;
; Auteur: Olivier Duval ;
; ;
; Description: Programme utilisé pour déterminer les paramètres optimaux d'un modèle
de transistor ;
; en utilisant un algorithme génétique. ;
; Date: 15/09/2004 ;
; ;
;;


;Algorithme de mise en ordre
(include "/usr/share/umbscheme/slib/sort.scm")


;Déclarations d'optimisation de compilation
(declare
(standardbindings)
(extendedbindings)
(block)
(inline)
```

```
(inlininglimit 1000))

;Fonctions utilitaires
(define pi 3.14159265358)
(define randomlocal (lambda ()
(if (< (randomreal) 0.5)
(* 10000 (randomreal))
(* 10000 (randomreal)))))

(define (tanh x)
(/ ( (exp x) (exp (* 1 x)))
(+ (exp x) (exp (* 1 x)))))

;Fonction qui crée une instance d'un transistor
(define (makeids alpha beta lamda vPto gamma b vgexp)
(define (ids vd vg vs vb)
(let ((vds ( vd vs))
(vgs ( vg vs))
(vsb ( vs vb)))
(* (expt (expt ( vgs (+ vPto (* gamma vds))) 2)
vgexp )
(/ beta (+ 1 (* b ( vgs vPto))))
(tanh (* alpha vds))
(+ 1 (* lamda vds)))))
ids)

;Fonction utilisée pour créer un transistor à partir
```

```
;des exemples offerts dans l'article de Jin
(define (makeidsfromlibrary column)
(cond ((= column 1)
(makeids 1.93 0.0227 0.0125 1.86 0.056 0.0512 0.96))
((= column 2)
(makeids 1.93 0.0227 0.0125 1.86 0.056 0.0512 0.96))
((= column 3)
(makeids 2.56 1.319 0.59 0.07 0.061 1.026 1.06))
((= column 4)
(makeids 15.21 7.144 1.067 1.015 0.263 0.033 7.98))
(else
(makeids 2.18 1.482 1.726 0.203 0.116 0.777 0.67))))


; Fonction de création d'un vecteur de valeurs partant de from
; et incrémentant de inc jusqu'à "to"
(define (makeinterval from to inc)
(if (>= from to)
'()
(cons from (makeinterval (+ from inc) to inc))))


; Fonction d'affichage d'un nombre quelconque d'arguments
(define (out . args)
(display args))


; Courbes de références utilisées pour fins de comparaison
; La courbe composée de X dans les résultats est "ivcnt800.scm"
(load "/home/olivier/scheme/projetspice/ivcnt.scm")
(load "/home/olivier/scheme/projetspice/ivcnt800.scm")
```

```
; Fonction importante:
; Évaluation des paramètres d'une cellule selon une méthode
; qui retourne un critère de qualité d'une combinaison de paramètres
(define (evalparams alpha beta lamda vPto gamma b vgexp)
(define currentmodel (makeids alpha beta lamda vPto
gamma b vgexp))
(define listref4 (lambda (input) (listref input 4)))
(define getcnt (lambda (table) (map listref4 table)))
(define getpoints (lambda (list)
(map reverse
(map cdr (map reverse list)))))
(define (listtocurrentmodel inputlist)
(currentmodel
(listref inputlist 0)
(listref inputlist 1)
(listref inputlist 2)
(listref inputlist 3)))
(define sqdiff (lambda (ref test) (expt ( ref test) 2)))
(define ratiocriterium (lambda (ref test)
(cond
((and (< (abs test) 1e15)
(< (abs ref) 1e15))
0)
((< (abs ref) 1e15)
(abs (/ test 1e15)))
((< (abs test) 1e15)
(abs (/ ( ref test) ref)))
((< (abs ref) (abs test))
```

```
(abs (/ test ref)))
(else (abs (/ ref test))))))
(abs (apply + (map sqdiff
(getcnt ivmceuen)
(map listtocurrentmodel
(getpoints ivmceuen))))))

; Fonction d'évaluation de cellule qui distribue les paramètres
; de la cellule à la fonction "evalparams"
(define (evalcell cell)
(evalparams (listref cell 0)
(listref cell 1)
(listref cell 2)
(listref cell 3)
(listref cell 4)
(listref cell 5)
(listref cell 6)))

; Fonction utilisée par randomgauss pour créer un nombre aléatoire
(define gauss01 (lambda ()
(let* ((x1 ( (* 2.0 (randomreal)) 1))
(x2 ( (* 2.0 (randomreal)) 1))
(w (+ (* x1 x1) (* x2 x2))))
(if (>= w 1.0)
(gauss01)
(* x1 (sqrt (/ (* 2.0 (log w)) w)))))))

; Fonction qui crée un nombre réel pris selon une
```

```
; distribution gaussienne entre low et high
(define randomgauss (lambda (m s)
(+ m (* s (gauss01)))))

; Fonction qui crée un nombre réel pris selon une
; distribution continue entre low et high
(define randomcontinuous (lambda (low high)
(+ low (* (randomreal) ( high low)))))

; Fonction de création d'un gène qui spécifie la loi de
; probabilité utilisée
(define (creategene law el1 el2)
(law el1 el2))

; Fonction de création de cellule au hasard. Les cellules sont
; prises selon une distribution gaussienne.
(define createcell (lambda ()
(list
(creategene randomgauss 2.32 1) ;alpha
(creategene randomgauss 0.021 0.2) ;beta
(creategene randomgauss 0.61 1) ;lamda
(creategene randomgauss 3.29 1) ;vPto
(creategene randomgauss 0.147 0.2) ;gamma
(creategene randomgauss 0.186 0.2) ;b
(creategene randomgauss 4.28 1)))) ;vgexp

; Création d'une liste dont les éléments sont d'abord évalués
```

```
; avec "eval" puis ajoutés à la liste
(define (createlist element total eval)
(define (iterate current)
(if (>= current total)
'()
(cons (eval element current)
(iterate (+ 1 current)))))
(iterate 0))

;Fonctions utilitaires pour manipuler des listes
(define identity (lambda (y z) (y) ))
(define identitynp (lambda (y z) y))
(define listreffn (lambda (y z) (listref y z)))

; Création d'une liste de "nb" instances "d'éléments"
(define (makepool element nb)
(createlist element nb identity))

; Création d'une liste composée de l'index et de l'élément sur
; lequel il pointe dans la liste "element"
(define indexing (lambda (element index)
(list index (listref element index))))

; Fonction qui définit le test de qualité utilisé pour comparer
; deux cellules
```

```
(define smaller (lambda (a b)
(if (< (car (cdr a)) (car (cdr b)))
#t
#f)))

; Fonction qui recherche l'index des meilleures cellules de la liste
(define (getindexes locallist)
(map car (sort locallist smaller)))

; Fonction qui retourne le sousensemble de la population constitué
; des cellules pointées par la liste des "indexes"
(define (retrievecells indexes pool)
(define (iterate current)
(if (>= current (length indexes))
'()
(cons (listref pool (listref indexes current))
(iterate (+ 1 current)))))
(iterate 0))

; Création d'une nouvelle population constituée des
; k premiers éléments de la liste "locallist"
(define (reducepool k locallist)
(createlist (getindexes locallist) k listreffn))

; Définition de la taille de la population
(define numberofcells 1000)
```

```
; Fonction qui enlève la partie la moins intéressante de la population
(define (evalpool pool)
(retrievecells
(reducepool
(* numberofcells 0.4)
(createlist (map evalcell pool)
(length (map evalcell pool))
indexing ))
pool))


; Fonction qui attend un entier et retourne un nombre entier
; aléatoire compris entre 0 et "a"
(define getsepar (lambda (a) (randominteger a)))


; Fonction qui retourne un nombre entier aléatoire compris entre
; 0 et "a" et différent de "b"
(define getsepardiff (lambda (a b)
(let ((y (getsepar a)))
(if (not (= y b))
y
(getsepardiff a b)))))
; fonction qui retire l'élément "index" de la liste "lst"
(define (rmlistelem lst index)
(cond ((= index 0) (cdr lst))
(else (cons (car lst)
(rmlistelem (cdr lst)
( index 1))))))
```

```
; fonction qui change l'élément "index" de la liste "lst"
; en lui donnant la valeur newelem
(define (chlistelem lst index newelem)
(cond ((= index 0)
(cons newelem (cdr lst)))
(else
(cons (car lst)
(chlistelem (cdr lst)
( index 1)
newelem)))))
; Fonction "d'accouplement": à partir des cellules père et mère,
; on crée un enfant qui possède une partie des gènes de chacun des parents
(define (mate father mother)
(define (iterate current total half)
(if (>= current total)
'()
(cons (if (>= current half)
(listref mother current)
(listref father current))
(iterate (+ 1 current) total half))))
(iterate 0
(length father)
(getsepar (length father))))


; Fonction qui décrémente un entier positif "a"
(define (decrnatural a)
(if (= a 0)
a
( a 1)))
```

```
; Fonction d'accouplement: applique la fonction "mate" à toute la population
(define (matepopulation initialpopulation)
(define (matesingles population)
(if (< (length population) 2)
population
;procéder à l'accouplement
(let* ((father (getsepar (length population)))
(mother (getsepardiff
(length population) father)))
(cons
(mate (listref population father)
(listref population mother))
(matesingles (rmlistelem
(rmlistelem population
father)
(decrnatural mother)))))))
(append initialpopulation
(matesingles initialpopulation)))

; Fonction de mutation appliquée à la cellule "child"
(define (mutate child)
(define localindex (getsepar (length child)))
(chlistelem child
localindex
(listref (createcell) localindex)))

; Application de la fonction de mutation à certaines cellules
```

```
; prises au hasard dans toute la population
(define (mutatepopulation population)
(define (iterate current)
(if (>= current (length population))
'()
(cons (if (= 0 (randominteger 50))
(mutate (listref population current))
(listref population current))
(iterate (+ 1 current)))))
(iterate 0))


; Test de similitude entre 2 gènes a et b
(define (verysimilar? a b)
(cond ((and (> 1e15 (abs a)) (> 1e15 (abs b)))
#t)
((> 1e15 (abs (+ a b)))
#f)
((< (/ (abs ( a b)) (+ a b)) 0.1)
#t)
(else
#f)))


; Test d'équivalence entre deux cellules: si tous les gènes
; sont semblables, les cellules le sont aussi
(define (eqvcells? cell1 cell2)
(define (iterate current)
(if (>= current (length cell1))
#t
```

```
(if (not (verysimilar? (listref cell1 current)
(listref cell2 current)))
#f
(iterate (+ 1 current)))))
(iterate 0))


; Test sur une liste: retourne #t si elem en fait partie,
; #f autrement
(define (partofthelist? list elem)
(let ((lengthlist (length list)))
(define (iterate current)
(if (>= current lengthlist)
#f
(if (eqvcells? (listref list current) elem)
#t
(iterate (+ 1 current)))))
(iterate 0)))


; Éliminer les cellules trop semblables d'une population
(define (killclones list)
(if (eqv? '() list)
'()
(if (partofthelist? (cdr list) (car list))
(killclones (cdr list))
(cons (car list) (killclones (cdr list))))))


; Création d'une population de "nbofnewcomers" cellules
(define newbreed (lambda (nbofnewcomers)
```

```
(makepool createcell nbofnewcomers)))


; Ajouter des cellules prises au hasard
(define (migratepopulation population)
(append population
(newbreed ( numberofcells (length population)))))


(define (geneticsearch maxiter)
(define (iterate current perf pool)
(if (and (< current maxiter)
(if (eqv? 0 perf)
#t
(> (cdr (listref perf ( current 1))) 9e12)))
(begin
(if (= 0 (modulo current 10)) (pp current))
(if (= 0 (modulo current 1))
(if (eqv? 0 perf)
(begin
;;(pp perf)
(set! perf
(list (cons current
(listref
(sort
(map evalcell pool)
<) 0)))))
(begin
(set!
perf (append
```

```
perf (list
(cons current
(listref
(sort
(map evalcell pool)
<) 0)))))
;;(pp perf)
)))
(iterate (+ 1 current)
; Fonction utilisée pour fin de statistiques
perf
; Ajouter des cellules prises au hasard
(migratepopulation
; Éliminer les cellules trop semblables
(killclones
; Modifier 1 gène pris au hasard dans certaines cellules
; prises au hasard
(mutatepopulation
; Prend les gènes de 2 cellules prises au hasard
; pour en créer une troisième
(matepopulation
; Enlever la partie la moins importante de la population
(evalpool pool)))))))
(begin
(pp perf)
; Effectuer le dernier ajustement sur la population
(mutatepopulation
(matepopulation
(evalpool pool))))))
; Initialisation du procédé
```

```
(iterate 0 0 (makepool createcell numberofcells)))

(define (iterate current pool perf)
(if (= 0 (modulo current 1))
(cons perf (cons current
(listref (sort (map evalcell pool) <)
0)))))

(iterate 0 (makepool createcell numberofcells) '())

; Un exemple d'utilisation du programme: "rel" est la
; population résultante de tout le procédé
(define rel (geneticsearch 300))

; Fonction qui extrait la meilleure cellule de "pool"
(define (getbestcell pool)
(retrievecells
(reducepool 1
(createlist (map evalcell pool)
(length (map evalcell pool))
indexing ))
pool))

; Exemple d'affichage du gène alpha de la meilleure cellule obtenue
(pp (listref (getbestcell rel) 0))
```

```
; Assignation de la meilleure cellule à "outmode"
(define outmod (listref (getbestcell re1) 0))


; Création d'un modèle de transistor à partir de la meilleure cellule
(define outmodel (makeids
(listref outmod 0)
(listref outmod 1)
(listref outmod 2)
(listref outmod 3)
(listref outmod 4)
(listref outmod 5)
(listref outmod 6)))


; Évaluation du modèle défini cihaut dans une simulation DC
; qui fait varier les tensions de grille et de drain
(define sweepboth
(apply append
(map (lambda (p1)
(map (lambda (p2)
(cons p2 (outmodel p2 p1 0 0)))
(makeinterval 0.9 0 0.01)))
(makeinterval 0.9 0.29 0.2))))


; (define sweep
; (map (lambda (p1)
; (cons p1 (outmodel p1 0.9 0 0)))
; (makeinterval 0.9 0 0.01)))
```

```
; (define sweepvds
; (map (lambda (p1)
; (cons p1 (model1 p1 3 0 0)))
; (makeinterval 2 20 0.1)))


; Affichage de la liste "sweep"
(define (printsweep sweep)
(foreach
(lambda (e)
(out (car e) " " (cdr e) "\n"))
sweep))


; Sauvegarde du résultat de simulation dans un fichier
(withoutputtofile
"/home/olivier/scheme/projetspice/cntoli1.tmp"
(lambda () (printsweep sweepboth)))
```

ANNEXE III

ARTICLE DE JIN SUR LE MODULE DE CURTICE ET ETTENBERG

PARAMETRIC STUDY OF AN EMPIRICAL MODEL FOR IV-CURVE OF CARBON NANOTUBE FIELD EFFECT TRANSISTOR

Zhongfang Jin, Olivier Duval, Yvon Savaria, and Jean-Jacques Laurin
zhongfang.jin@polymtl.ca, olivier.duval@polymtl.ca, savaria@vlsi.polymtl.ca and jean-jacques.laurin@polymtl.ca
Ecole Polytechnique de Montreal, Montreal, QC, Canada

***Abstract*—In this paper, an empirical model of 7 parameters in the literature is used to approximate the IV-curve of carbon nanotube field effect transistor (CNTFET). Compared to conventional silicon-based field effect transistors (FETs), an extra parameter, flat-band voltage, representing quantum mechanical effects of CNTFETs, is merged with one of 7 modal parameters, voltage threshold. The empirical model is slightly revised so as to obtain drain currents of real values. An example is then used to illustrate the sensitivity of each modal parameter. By implementing nonlinear least square techniques, a Matlab routine is compiled and a set of modal parameters are extracted for measured IV-curve of a p-type CNTFET. A good agreement is achieved between the measured IV-curve and the one based on the empirical model.**

For comparison, two sets of modal parameters are also extracted from simulated IV-curves for a double-gated silicon-based n-type FET with Purdue University NanoMOS and for a top-gated p-type FET with IBM 130-nm CMOS technology. Numerical results illustrate that the modal parameters vary a lot among conventional silicon-based FETs and CNTFETs. These modal parameters not only help us to understand the performance difference among conventional silicon-based FETs and CNTFETs, but also provide a useful way for the evaluation of the performance improvement among different versions of CNTFETs in the current stages, where high-level models such as BSIM3 FET models are impossible to be constructed now due to many challenges in recent development of CNTFETs.

Keyword: ***Empirical model; Carbon nanotube*** **(CNT);** ***Field effect transistor*** **(FET);** ***IV-curve;***

I. INTRODUCTION

Recently, carbon nantube (CNT) is extensively investigated because such kind of nanoscale material demonstrated lot of advantages over conventional material. Determined by its chemical structure, CNT can behavior either as a conductance or as a semiconductor, which makes such nanoscale material useable to manufacture future generations of nanoelectronic devices, when the conventional silicon-based technology meets its limits. Several authors [1-4] reported the realization of a single carbon nanotube field effect transistor (CNTFETs). By considering quantum effects in these CNTFETs, CV- and IV- curve modeling was also investigated [5]. However, due to many challenges in the current stages of development, high-level models such as BSIM3 FET model are impossible to be constructed now. Therefore, empirical models are still very useful for the performance evaluation of CNTFETs.

This paper focuses on IV-curve modeling of CNTFET. In section II, an empirical IV-curve model in [6] is selected for CNTFET. An extra modal parameter, flat-band voltage [5], representing quantum mechanical effects in CNTFETs, is merged with one of seven modal parameters, voltage threshold. Mathematical adjustment is also performed so as to obtain drain currents of real values. Based on the empirical model, a Matlab routine is complied with nonlinear least square techniques. Before testing the empirical model, parametric sweeps are studied in section III to illustrate the sensitivity of each modal parameter. In sections IV and V, numerical simulations are performed to demonstrate the validity of the empirical model and to illustrate the performance difference among conventional FETs and CNTFETs. Last section gives out our conclusions.

II. EMPIRICAL IV-CURVE MODEL

In this section, an empirical hyperbolic tangent model, originally proposed by Curtice and Ettenberg [6] for the empirical current-voltage description of GaAs metal-semiconductor-field-effect-transistors (MESFETs), is selected to approximate the IV-curve of a CNTFET. Original equation (4.18) in [7] described the initial expression of the empirical hyperbolic tangent model with four parameters α, β, λ and v_{t0}. For completeness, it is rewritten below.

$$I_{ds}(\alpha,\beta,\lambda,v_{t0}) = \beta(v_{gs}-v_{t0})^2\tanh(\alpha v_{ds})(1+\lambda v_{ds}) \quad (1)$$

As discussed in [7], two modifications were performed to improve the accuracy of the empirical model. The modified empirical model is expressed by original equation (4.19) in [7] with 7 modal parameters, and is given by

$$I_{ds}(\alpha,\beta,\gamma,\lambda,b,v_{t0},v_{g\exp}) = \frac{\beta}{1+b(v_{gs}-v_{t0})}(v_{gs}-(v_{t0}+\gamma v_{ds}))^{v_{g\exp}} * \tanh(\alpha v_{ds})(1+\lambda v_{ds}) \quad (2)$$

Here, both of the two modifications were already included in the above equation. Therefore, Eq.2 is slightly different from the original equation (4.19) in [7].

For CNTFET, an extra flat-band voltage v_{fb} [5] exists due to the interactions between metal electrodes and CNT. Thus, the voltage difference v'_{gs} between external gate and source metal electrodes is not equal to the voltage difference v_{gs} between two positions at a CNT near gate and source metal electrodes as given by the following equation:

$$v'_{gs} = v_{gs} + v_{fb} \quad (3)$$

It needs to mention that v'_{gs} is the measured gate voltage applied at the external gate and source metal electrodes, while one of 7 modal parameters in the empirical model refers to the internal gate voltage v_{gs}. Equation 3 can be rewritten as:

$$v_{gs} = v'_{gs} - v_{fb} \tag{4}$$

Combining Eqs.2 and 4, an IV-curve model of 8 parameters is then obtained for CNTFETs:

$$\begin{aligned} & I_{ds}(\alpha,\beta,\gamma,\lambda,b,v_{t0},v_{g\exp},v_{fb}) \\ &= \frac{\beta}{1+b(v'_{gs}-v_{fb}-v_{t0})}\left(v'_{gs}-v_{fb}-(v_{t0}+\gamma v_{ds})\right)^{v_{g\exp}} \\ &* \tanh(\alpha v_{ds})(1+\lambda v_{ds}) \\ &= \frac{\beta}{1+b(v'_{gs}-(v_{t0}+v_{fb}))}\left(v'_{gs}-((v_{t0}+v_{fb})+\gamma v_{ds})\right)^{v_{g\exp}} \\ &* \tanh(\alpha v_{ds})(1+\lambda v_{ds}) \end{aligned} \tag{5}$$

Here, the flat-band voltage v_{fb} can be merged with the original threshold voltage v_{t0} by introducing a general threshold voltage v'_{t0} defined as:

$$v'_{t0} = v_{t0} + v_{fb} \tag{6}$$

Combining Eq.6 with Eq.5, the final empirical model is described as:

$$\begin{aligned} & I_{ds}(\alpha,\beta,\gamma,\lambda,b,v'_{t0},v_{g\exp}) \\ &= \frac{\beta}{1+b(v_{gs}-v'_{t0})}\left(v_{gs}-(v'_{t0}+\gamma v_{ds})\right)^{v_{g\exp}} \\ &* \tanh(\alpha v_{ds})(1+\lambda v_{ds}) \end{aligned} \tag{7}$$

On certain conditions, complex values of drain currents are obtained by using Eq.7. Therefore, Eq.7 is slightly revised as Eq.8, where a square function guarantees the drain currents of real values.

$$\begin{aligned} & I_{ds}(\alpha,\beta,\gamma,\lambda,b,v'_{t0},v_{g\exp}) \\ &= \frac{\beta}{1+b(v_{gs}-v'_{t0})}\left[\left(v_{gs}-(v'_{t0}+\gamma v_{ds})\right)^2\right]^{v_{g\exp}/2} \\ &* \tanh(\alpha v_{ds})(1+\lambda v_{ds}) \end{aligned} \tag{8}$$

III. PARAMETRIC STUDY OF EMPIRICAL IV-CURVE MODEL

Before applying the empirical model to the CNTFETs, parameter sensitivity is investigated in this section. The original set of parameters listed in the 2nd column of Table A is a n-type FET. Since a CNTFET initially is a p-type FET, in this section, an example of a p-type FET is studied by changing the symbol of one modal parameter v_{t0} as shown in the 3rd column of Table A. The parameters in the 4th, 5th and 6th columns list the extracted modal parameters of FETs and will be discussed in sections IV and V.

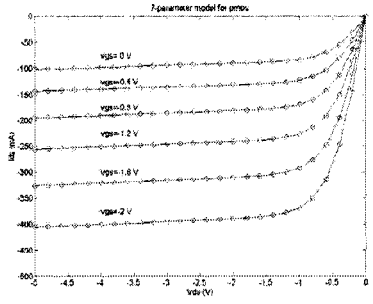
Table A: Model parameters of conventional FETs and CNTFETs

Modal parameters	Conventional FETs				CNTFETs
	n-type	p-type (Fig.1)	pMOS (Fig.6)	nMOS (Fig.8)	p-type (Fig.3)
α	1.93	1.93	2.56	15.21	2.18
β	0.0227	0.0227	1.319	7.144	1.482
λ	0.0125	0.0125	-0.590	-1.067	-1.726
$v_{t0}(v_{fb})$	-1.86	1.86	-0.070	-1.015	0.203
γ	-0.056	-0.056	0.061	-0.263	0.116
b	0.0512	0.0512	-1.026	0.033	-0.777
$v_{gexp}/2$	0.96	0.96	1.06	7.98	0.67
Note	Table 4.2 in Ref.[7]		Extracted modal parameters		

By using Eq.8, a group of data is calculated and IV-curve is plotted in Fig.1a as a reference for parametric study. In the following, the gate voltage is fixed at –1.2 V while each of 7 parameters vary relatively from 0.5, 0.75, 1.0, 1.25 and 1.5. Figures 1b, 1c, 1d, 1e, 1f, 1g and 1h show the impact of the variation of modal parameters α, β, λ, v_{t0}, γ, b and v_{gexp} on IV-curve, respectively.

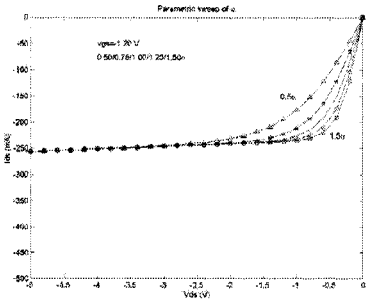
Figure 1b indicates that the slope of IV-curve is sensitive to the value of α. Figure 1c shows that the profile of IV-curve is linearly varied with the value of β, while figures 1e and 1h show the profile of IV-curve is dramatically and nonlinearly changed with other two modal parameters: threshold voltage v_{t0} and exponent parameter v_{gexp}, respectively.

Compared to the four parameters above α, β, v_{t0} and v_{gexp}, the other three parameters λ, γ and b behavior as tuning parameters. The profile of IV-curve is slightly changed with the variation of these three parameters. Therefore, α, v'_{t0} and v_{gexp} are three important modal parameters.

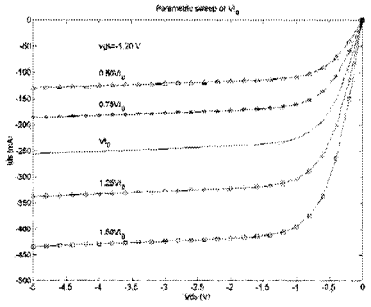


(a) reference

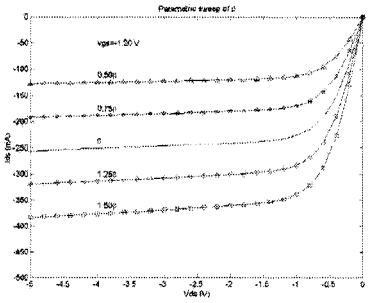
Figure 1: Parametric study of a conventional p-type FET



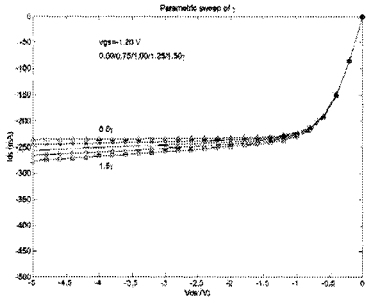
(b) α (slope parameter)



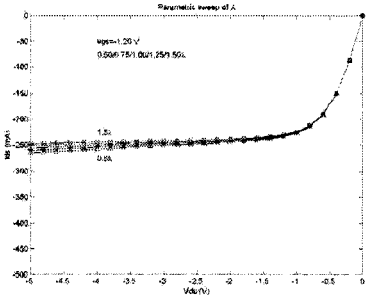
(e) v_{t0} (voltage threshold)



(c) β (scaling parameter)

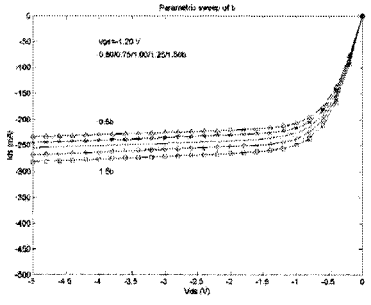


(f) γ (tuning parameter)



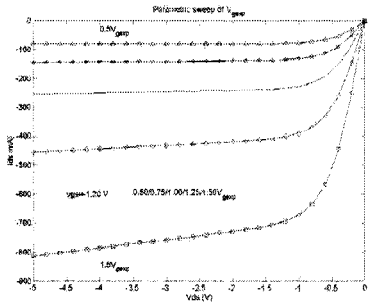
(d) λ (tuning parameter)

Figure 1: Parametric study of a conventional p-type FET (continued)



(g) b (tuning parameter)

Figure 1: Parametric study of a conventional p-type FET (continued)



(h) v_{gexp} (exponent parameter)

Figure 1: Parametric study of a conventional p-type FET (continued)

IV. MODAL PARAMETERS FOR CNTFET

In this section, the empirical model discussed in section II is used to approximate the IV-curve of a top-gated p-type CNTFET reported in [2]. Figure 2 shows 2D cross section of the top-gated p-type CNTFET with a gate length of 260 nm and the gate oxide thickness of 15 nm.

Based on Eq.8, and by using nonlinear least square technique, a Matlab routine is compiled. Based on the measured IV-curve in [2], 7 modal parameters are extracted for the top-gated p-type CNTFET and listed in the 6th column of Table A. Figure 3 shows the measured IV-curve and the one based on the model. A good agreement is achieved between these two curves. Compared to a maximum current of 3.25 μA in the whole IV-curve region of interest, the maximum difference of drain current between model and measurement is below 0.10 μA.

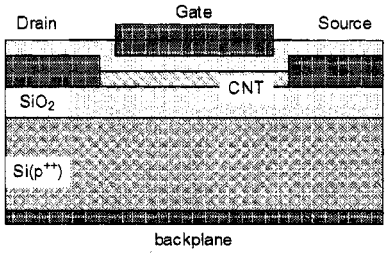


Figure 2: 2D cross section of a top-gated p-type CNTFET (not to scale)

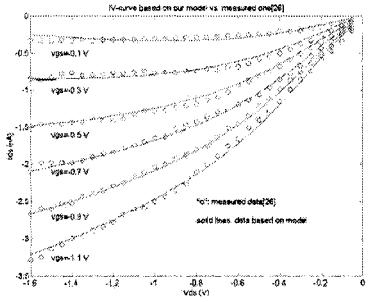


Figure 3: Comparison of IV-curve based on model and the measured one of CNTFET in [2]

V. MODAL PARAMETERS FOR CONVENTIONAL SILICON-BASED FETS

For comparison, modal parameters are also extracted from simulated IV-curves for two conventional silicon-based FETs: a top-gated p-type FET with IBM 130-nm CMOS technology [8], and a double-gated silicon-based n-type FET with NanoMOS 2.0 from Purdue University [9,10].

5.1 A TOP-GATED P-TYPE FET WITH IBM 130-NM CMOS TECHNOLOGY

For a top-gated p-type FET with 130-nm CMOS technology, a gate length of 120 nm and a gate width of 120 nm are used in our simulations. These parameters refer to a minimum FET from libraries in Cadence [11]. Based on libraries in Cadence, BSIM3 FET models were first constructed in ADS [12]. Then, IV-curve was obtained by performing circuit simulation in ADS.

By using our Matlab routine as mentioned in section IV, 7 modal parameters are extracted for the top-gated p-type FET and are listed in the 4th column of Table A. Figure 4 shows the simulated IV-curve in ADS and the calculated one based on the model. A good agreement is achieved between these two curves. Compared to a maximum current of 1.85 mA in the whole region of interest, the maximum difference of drain current between model and simulation is below 0.08 mA.

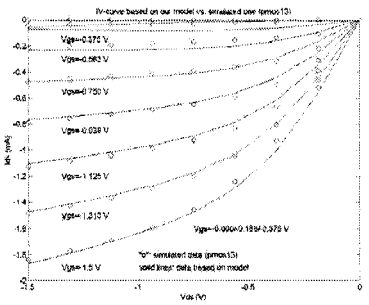


Figure 4: Comparison of IV-curve based on model and the simulated one of a p-type FET with CMOS 130-nm technology

5.2 A DOUBLE-GATED SILICON-BASED N-TYPE FET WITH NANOMOS

Besides testing simulated IV-curve with IBM 130-nm CMOS technology, we also test simulated IV-curve with Purdue University NanoMOS2.0 [10]. Figure 5 shows 2D cross section of a double-gated silicon-based n-type FET. The key geometrical and electrical parameters are listed in Table B. The simulated IV-curve was obtained from NanoMOS2.0.

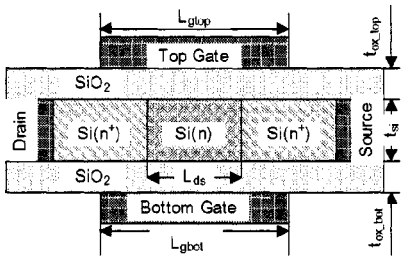


Figure 5: 2D cross section of a double-gated silicon-based n-type FET (not to scale)

Table B: Key geometrical and electrical parameters in Fig.5

geometrical parameters		electrical parameters	
t_{si}	3 nm	ε_{r_si}	11.7
t_{ox_top}	1.5 nm	$\varepsilon_{r_ox_top}$	3.9
t_{ox_bot}	1.5 nm	$\varepsilon_{r_ox_bot}$	3.9
L_{gtop}	10 nm		
L_{gbot}	10 nm	n^+	10^{20}/cm^3
L_{gd}	5 nm	n	10^{10}/cm^3

Similarly, by using the Matlab routine discussed in section IV, 7 modal parameters were extracted for the double-gated silicon-based n-type FET as shown in the 5th of Table A. Figure 6 shows the simulated IV-curve from NanMOS2.0 and the one based on the model. A good agreement is achieved between these two curves. Compared to a maximum current of 650 μA/μm in the whole region of interest, the maximum difference of drain current between model and simulation is below 40 μA/μm.

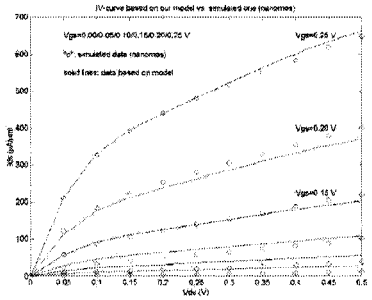


Figure 6: Comparison of IV-curve based on model and the simulated one of a double-gated silicon-based n-type FET

VI. CONCLUSIONS

In this paper, an empirical model of 7 parameters in the literature was selected to approximate the IV-curve of CNTFET. Compared to conventional FETs, an extra parameter, flat-band voltage, representing quantum mechanical effects of CNTFETs, was merged with one of seven modal parameters, voltage threshold. In order to obtain drain currents of real values, the original empirical model was slightly revised. An example was used to illustrate the parameter sensitivity of the empirical model. For a specific set of modal parameters, the profile of IV-curve is dramatically changed with four modal parameters α, β, v_{t0} and v_{gexp}, while it is slightly changed with the other three parameters λ, γ and b.

By using our Matlab routine, a set of modal parameters were first extracted for the measured IV-curve of a p-type CNTFET in [2]. Then, two sets of modal parameters were extracted from simulated IV-curves for a top-gated p-type FET with IBM 130-nm CMOS technology, and for a double-gated silicon-based n-type FET with NanoMOS (Purdue Univeristy), respectively. Good agreements were achieved between the measured/simulated IV-curves and those based on the empirical model. As expected, the empirical model works well for both conventional FETs and CNTFETs. Numerical results illustrate that the modal parameters vary a lot among conventional FETs and CNTFETs. These modal parameters also help us to understand the performance

difference among conventional FETs and CNTFETs. Moreover, they provide a useful method to evaluate the performance improvement among different versions of CNTFETs, especially in the current early stages, where high-level models such as BSIM3 FET models are impossible to be constructed due to many challenges in the recent development of CNTFETs.

REFERENCES

[1] R. Martel, H.-S. P. Wong, K. Chan, and P. Avouris, "Carbon nanotube field-effect transistors for logic applications," in *Proc. IEDM 2001*, 2001, pp. 159-161.

[2] S.Wind, J. Appenzeller, R. Martel, V. Derycke, and P. Avouris, "Vertical scaling of single-wall carbon nanotubes transistors using top gate electrodes," *Appl. Phys. Lett.*, vol. 80, p. 3817, 2002.

[3] Marty, L.; Naud, C.; Chaumont, M.; Bonnot, A.M.; Fournier, T.; Bouchiat, V., "Self-assembled single wall carbon nanotube field effect transistors," *Nanotechnology, 2003. IEEE-NANO 2003.* 2003 Third IEEE Conference on, vol.1, 12-14 Aug. 2003, pp.240 - 243 vol.2

[4] *Nihey, F.; Hongo, H.; Ochiai, Y.; Yudasaka, M.; Iijima, S.,* "High performance carbon-nanotube field-effect transistors," Microprocesses and Nanotechnology Conference, 2003. Digest of Papers. 2003 International, 29-31 Oct. 2003, pp.72 - 73

[5] *Raychowdhury, A.; Mukhopadhyay, S.; Roy, K.;* "Modeling of ballistic carbon nanotube field effect transistors for efficient circuit simulation," *Computer Aided Design, 2003.* ICCAD-2003. International Conference on, 9-13 Nov. 2003 Pages: 487 - 490

[6] W.R. Curtice and M. Ettenberg, "N-FET, a New Software Tool for Large-Signal GaAs FET Circuit Design," *RCA Review*, vol.46, Sept.1985, pp.321-339.

[7] Ravender Goyal, "Monolithic Microwave Integrated Circuits: Technology & Design," pp.244-261, *Artech House, Inc.*, 1989.

[8] www-3.ibm.com, "Foundry Technologies 130-nm CMOS and RF CMOS," Sept. 2003.

[9] Ren Zhibin *et al.*, "Ballistic Nanotransistor: A Simulation Study," *IEDM Technical Digest*, San Francisco, CA, Dec.11-13, 2000.

[10] *www.ece.purdue.edu/celab/*, NanoMOS 2.0

[11] *www.cadence.com*, Cadence manuals, 2004.

[12] *www.agilent.com*, ADS manuals, 2004.