



HAL
open science

Conception pour la faible consommation en technologies SOI 2D et 3D : application à l'arithmétique

S.J. Abou-Samra

► **To cite this version:**

S.J. Abou-Samra. Conception pour la faible consommation en technologies SOI 2D et 3D : application à l'arithmétique. Micro et nanotechnologies/Microélectronique. Institut National Polytechnique de Grenoble - INPG, 1998. Français. NNT: . tel-00002982

HAL Id: tel-00002982

<https://theses.hal.science/tel-00002982>

Submitted on 11 Jun 2003

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

THÈSE « label européen »

pour obtenir le grade de

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Discipline : **Microélectronique**

présentée et soutenue publiquement

par

ABOU-SAMRA Selim Joseph

le 18 Décembre 1998

Titre :

**CONCEPTION POUR LA FAIBLE CONSOMMATION EN
TECHNOLOGIES SOI 2D ET 3D : APPLICATION À L'ARITHMÉTIQUE**

Directeur de thèse : Alain Guyot

JURY

M. Bernard Courtois,	Président
Mme. Márta Rencz,	Examineur
M. Utz Baitinger,	Rapporteur
M. Alain Guyot,	Examineur
M. Christian Piguet,	Rapporteur

"Il faut imaginer Sisyphe heureux", Camus

REMERCIEMENTS

"Ciel! que lui vais-je dire, et par où commencer?", Phèdre, Racine

Je tiens à remercier Monsieur Bernard Courtois, Directeur de recherche au CNRS et directeur du laboratoire TIMA, qui a bien voulu me faire l'honneur de présider mon jury de thèse.

Je remercie aussi Monsieur Utz Baitinger, Professeur à l'Université de Stuttgart et directeur de l'IPVR, pour m'avoir accueilli durant six mois dans son équipe; que lui et Monsieur Christian Piguet, chargé de cours à l'Université de Neuchâtel et chef du département de micro-puissance au CSEM trouvent ici l'expression de ma reconnaissance pour avoir accepté d'être rapporteurs de mon travail.

Je tiens à témoigner ma reconnaissance à Madame Márta Rencz, Professeur à l'Université de Budapest, qui a accepté de faire partie de mon jury de thèse.

Je remercie particulièrement Monsieur Alain Guyot pour m'avoir accueilli dans son équipe et pour avoir dirigé ce travail.

Que tous ceux qui m'ont aidé durant ma thèse, que ce soit au laboratoire TIMA, à l'IPVR ou en dehors trouvent ici l'expression de ma reconnaissance. Ces remerciements vont tout particulièrement à Isabelle Essalhiene, Jens Arweiler, Sebastien Roux, Pier Alessandro Aisa, Marc Renaudin ainsi qu'à Bernd Stöhr, Markus Bühler et Patricia Sagmeister sans oublier Isabelle Saliou et Aïda Holtmeier.

Je remercie enfin tous les membres de ma famille pour m'avoir toujours soutenu au cours de mes études, et je souhaite leur dédier cette thèse.

TABLE DES MATIÈRES

Introduction.....	29
English Introduction	33
Chapitre premier : Faible consommation	37
I.1. Introduction	39
I.2. Hiérarchie de la consommation	39
I.3. Sources de dissipation	40
I.3.1. Dissipation dynamique	41
I.3.1.1. Genèse d'une transition	41
I.3.1.2. Redondance et activité.....	42
I.3.1.3. Exemple de réduction d'activité	43
I.3.1.4. Rendre l'information redondante.....	44
I.3.1.5. Compromis	45
I.3.2. Court circuit.....	45
I.3.3. Dissipation statique	45
I.4. Réduction d'échelle.....	46
I.4.1. Réduction à champ constant	46
I.4.2. Réduction à tension constante	47
I.4.3. Réduction latérale	47
I.4.4. Interconnexion.....	47
I.4.5. Limitations.....	47
I.5. Métrique et performance.....	48
I.5.1. Débit.....	48
I.5.2. Autonomie	49
I.5.3. Efficacité	49

I.6.	Cellules complexes.....	51
I.6.1.	Niveau porte logique.....	51
I.6.1.1.	Niveau transistor.....	52
I.6.2.	Protocole expérimental	52
I.6.2.1.	Circuits test - niveau porte	52
I.6.2.2.	Circuits test - niveau transistor	53
I.6.3.	Résultats et discussion.....	53
I.7.	Collision / Absorption / Propagation.....	55
I.8.	Résumé	59
I.9.	Summary	60
Chapitre second : SOI.....		61
II.1.	SOI.....	63
II.1.1.	Qu'est ce?.....	63
II.1.1.1.	SIMOX.....	63
II.1.1.2.	Collage et polissage.....	64
II.1.1.3.	Smart Cut®.....	64
II.1.2.	Pourquoi le SOI.....	65
II.1.3.	Polarisation canal.....	66
II.1.4.	DTMOS	67
II.1.5.	Grille en T	67
II.2.	SOI sur SOI.....	69
II.2.1.	Motivations	69
II.2.2.	Structure tridimensionnelle.....	69
II.2.2.1.	Couches disponibles.....	69
II.2.2.2.	Réalisation	69
II.2.3.	Cellules autoduales.....	72
II.3.	Outils de conception.....	73
II.3.1.	Design kits.....	73
II.3.2.	Bibliothèques de cellules.....	73
II.3.2.1.	Performances d'une cellule	75

II.4. Résumé	77
II.5. Summary	78
Chapitre troisième: Arithmétique.....	79
III.1. Introduction	81
III.2. Addition.....	81
III.2.1. Synthèse d'Additionneurs.....	82
III.2.1.1. Propagation et génération de groupe.....	82
III.2.2. Modèle complexité / Délai.....	83
III.2.2.1. Additionneur à propagation.....	83
III.2.2.2. Carry Select à deux étages	83
III.2.2.3. Additionneur de Brent & Kung.....	84
III.2.2.4. Additionneur de Sklansky.....	84
III.2.2.5. Kogge & Stone et Han & Carlson	84
III.2.3. Modèles d'activité.....	85
III.2.3.1. Additionneur à propagation.....	86
III.2.3.2. Autres architectures.....	88
III.2.3.3. Comparaison.....	90
III.3. Multiplication.....	90
III.3.1. Synthèse de multiplieurs	91
III.3.1.1. Braun.....	91
III.3.1.2. Wallace Dadda.....	92
III.3.1.3. Réducteurs 4 : 2.....	93
III.3.2. Évaluation de l'activité.....	94
III.3.2.1. Simulation Monte Carlo.....	94
III.3.2.2. Approche combinée HSPICE / VERILOG.....	94
III.3.3. Additionneurs pour la multiplication.....	96
III.4. Division.....	96
III.4.1. Introduction à la division rapide.....	96
III.4.2. La notation "borrow save".....	97
III.4.3. Le diviseur faible consommation.....	100

III.4.4. La conversion du quotient.....	102
III.4.4.1. Modélisation mathématique.....	103
III.4.4.2. Synthèse de layout.....	104
III.5. Résumé	106
III.6. Summary	107
Chapitre quatrième : Comparaison 2D vs. 3D.....	109
IV.1. Introduction	111
IV.2. Critères de comparaison	111
IV.2.1. Surface	111
IV.2.2. Capacité	111
IV.2.3. Délai	111
IV.2.4. Énergie par opération.....	111
IV.2.5. Produit énergie - délai	112
IV.3. Multiplieur 16x16.....	112
IV.4. Résultats et analyse.....	114
IV.5. Résumé	116
IV.6. Summary	117
Conclusion.....	119
English conclusion.....	123
Annexe : Bibliothèques de cellules	127
I.1. Introduction	129
I.1.1. Taille des cellules.....	129
I.1.2. Routage.....	129
I.1.3. Terminologie.....	129
I.2. Inverseur.....	130
I.3. Non-Et à deux entrées.....	132
I.4. Non-Ou à deux entrées.....	134

I.5. Et à deux entrées.....	136
I.6. Et - Ou Inversé.....	138
I.7. Majorité Inversée à 3 entrées.....	140
Bibliographie	143

ABRÉVIATIONS ET ACRONYMES

Back end : Côté fondeur

Bit : Chiffre binaire (Binary digit)

Bulk : Silicium Massif

CS : Cellule de réduction de 4 produits partiels en 2 dans un multiplieur (Carry Save)

CSA : Additionneur à anticipation de retenue (Carry Select Adder)

Design kit : Ensemble de fichiers de configuration d'un outil de CAO spécifiques à une technologie.

DIBL : Effet canal court (Drain Induced Barrier Lowering)

DRC : Vérification des règles de dessin (Design Rules Check)

DTMOS : MOS à seuil dynamiquement variable ou encore MOS hybride (Dynamic Threshold MOS)

EDMOS : Edge Defined MOS

EDP : Produit énergie - délai (Energy - Delay Product)

ELO : Epitaxial Lateral Overgrowth

EPO : Energie Par Opération

ERC : Vérification des règles électriques (Electrical Rules Check)

FA : Additionneur 1 bit (Full Adder)

Fanout : Sortance

Front end : Côté concepteur

Full custom design : Conception totalement personnalisée

HA : Demi additionneur (Half Adder)

HIPERLOGIC : Thousand MOPS per milliwatt **HI**gh **PER**formance **LOGIC** : Projet Européen ESPRIT IV numéro 20.023

Layout : Dessin des masques

LDD : Lightly Doped Drain

LOCOS : LOCal Oxydation of Silicon

LVS : Layout Versus Schematic

Netlist : Fichier contenant la liste des nœuds d'un circuit

Process : Ce mot est traduit de deux manières selon le contexte :

- Processus : suite d'étapes technologiques
- Procédé : méthode de fabrication

RCA : Additionneur à propagation de retenue (Ripple Carry Adder)

SIA : Semiconductor Industry Association

SIMOX : Separation by IMplanted OXYgen

SOI : Silicium sur isolant (Silicon On Insulator)

LISTE DES FIGURES

Figure 1.1 :	Divers niveaux d'abstraction.....	39
Figure 1.2 :	Mécanisme de dissipation dynamique.....	41
Figure 1.3 :	Charge et décharge de C_1	42
Figure 1.4 :	Taxinomie des transitions.....	42
Figure 1.5 :	Délai d'une porte logique en fonction de la tension d'alimentation..	48
Figure 1.6:	Énergie par opération en fonction de la tension d'alimentation.....	49
Figure 1.7 :	EDP en fonction de V_{dd}	50
Figure 1.8 :	Construction de cellules complexes	51
Figure 1.9a :	Complexité = 3.....	52
Figure 1.9b :	Complexité = 5.....	52
Figure 1.10 :	Porte Nand à n entrées.....	53
Figure 1.11 :	EPO et EDP en fonction de la profondeur logique.....	54
Figure 1.12 :	EPO/bit et EDP/bit en fonction du nombre de transistors en série..	54
Figure 1.13 :	Génération et propagation d'une transition redondante	55
Figure 1.14 :	Seuil de propagation.....	56
Figure 1.15 :	Amplitudes en fonction de V_{dd} pour diverses valeurs de C_1	57
Figure 1.16 :	Variation du seuil de propagation en fonction de la capacité	57
Figure 2.1 :	Principe du SIMOX.....	63
Figure 2.2 :	SOI par collage et polissage.....	64
Figure 2.3 :	Procédé Smart Cut®	65
Figure 2.4 :	Inverseurs en Silicium massif et en SOI.....	66
Figure 2.5 :	Transistor avec grille en T	67
Figure 2.6 :	Étapes de fabrication de la grille en T	68
Figure 2.7 :	Coupe schématique de la structure tridimensionnelle.....	70
Figure 2.8 :	Première couche de SOI	70
Figure 2.9 :	EDMOS type N.....	70
Figure 2.10 :	Ouverture et épitaxie avec inversion du dopage.....	71
Figure 2.11 :	Épitaxie latérale de la couche de Silicium supérieure.....	71
Figure 2.12 :	Planarisation du Silicium par polissage chimico-mécanique (CMP)..	71
Figure 2.13 :	Structure finale avant métallisation	72
Figure 2.14a :	Majorité auto-duale.....	73

Figure 2.14b :	Majorité autoduale alternative	73
Figure 2.15 :	Flux de conception d'une bibliothèque de cellules.....	74
Figure 2.16a :	$\overline{\text{Maj}}$	76
Figure 2.16b :	$\overline{\text{Maj}}$ alternative.....	76
Figure 3.0 :	Diagramme d'échange délai - énergie	81
Figure 3.1 :	Architecture générique	83
Figure 3.2 :	Un 2-CSA.....	84
Figure 3.3 :	Additionneur de Kogge et Stone (32 bits).....	85
Figure 3.4 :	Fenêtres d'activité dans une chaîne de propagation.....	86
Figure 3.5 :	Un autre exemple de fenêtres d'activité.....	86
Figure 3.6 :	Activité à un instant donné t_i	88
Figure 3.7 :	Multiplication en trois étapes	91
Figure 3.8 :	Multiplieur de Braun	92
Figure 3.9 :	Arbre de Wallace.....	92
Figure 3.10 :	Réducteurs 4 : 2.....	93
Figure 3.11 :	Simulation Monte Carlo améliorée.....	95
Figure 3.12 :	Enveloppe temporelle pour l'arbre de Wallace.....	96
Figure 3.13 :	Additionneur pour arbre de Wallace.....	96
Figure 3.14 :	Classification des algorithmes de division.....	97
Figure 3.15 :	Une ligne d'addition/soustraction hybride.....	98
Figure 3.16 :	Diagramme de Robertson de la division SRT en base 2	98
Figure 3.17 :	Étage du diviseur	99
Figure 3.18 :	Diviseur régulier en notation redondante.....	99
Figure 3.19 :	Chemin critique : une cellule de tête et deux cellules de queue.....	100
Figure 3.20 :	Partitionnement des cellules de queue.....	100
Figure 3.21 :	Cellule de tête	101
Figure 3.22 :	Cellule de queue avec q_j en dernier.....	101
Figure 3.23 :	Cellule de queue avec q_j en premier.....	102
Figure 3.24 :	Layout du diviseur basse consommation 16x16 bits.....	102
Figure 3.25 :	Exemple d'additionneur en temps constant.....	103
Figure 3.26 :	Découpage en blocs	104

Figure 3.27 :	Plan de masse en fonction de	105
Figure 3.28 :	Diviseur 16 bits avec conversion du quotient en temps constant.....	105
Figure 4.1 :	Layout du multiplieur 16x16 en technologie SOI 2D.....	113
Figure 4.2 :	Layout du multiplieur 16x16 en technologie SOI 3D.....	114

LISTE DES TABLEAUX

Tableau 1.1 : Exemple de code redondant.....	43
Tableau 1.2 : Distances de Hamming.....	43
Tableau 1.3 : Nombre de transitions.....	44
Tableau 1.4 : Codage redondant.....	44
Tableau 1.5 : Codage redondant.....	44
Tableau 1.6 : Réduction d'échelle.....	46
Tableau 2.1 : Cellules de base.....	75
Tableau 2.2 : Résultats pour la cellule $\overline{\text{Maj}}$	76
Tableau 3.1 : Erreurs relatives.....	87
Tableau 3.2 : Comparaison d'additionneurs.....	90
Tableau 3.3 : Architecture de multiplieur.....	93
Tableau 3.4 : Comparatif Monte-Carlo.....	94
Tableau 3.5 : Proportion de transitions redondantes.....	95
Tableau 4.1 : Arbre de réduction.....	112
Tableau 4.2 : Résultats comparatifs pour les multiplieurs.....	115
Tableau 4.3 : Résultats pour la cellule $\overline{\text{Maj}}$	115

INTRODUCTION

“Ein Irrtum wird dann erst zum Fehler, wenn man in ihm beharrt.”, E. Jünger

Le souci de diminuer l'énergie nécessaire pour effectuer une opération (ou une action) donnée n'est pas uniquement lié à la microélectronique. C'est une constante de l'histoire des technologies. Ainsi, à près de 5000 ans d'intervalle, la roue et le transistor s'inscrivent dans la même lignée.

Depuis l'invention du transistor en 1947 le nombre d'opérations par Joule effectuées par un circuit a augmenté de plusieurs ordres de grandeur. Cela est lié aux besoins croissants en outils portables (outils de communication ou simplement de travail).

La portabilité d'un outil est liée à son poids, et donc à la quantité d'énergie embarquée. C'est la raison pour laquelle nous focaliserons sur la réduction de l'énergie consommée et non pas de la puissance. La puissance est une moyenne dans le temps de l'énergie consommée, et de ce fait contient moins d'informations. D'ailleurs le terme “consommation de puissance” trop courant dans la littérature est abusif. Contrairement à l'énergie, la puissance ne peut pas être consommée. Comme illustration prenons l'exemple d'une voiture : elle consomme une certaine quantité de carburant (énergie chimique) par kilomètre qui est liée au nombre de chevaux vapeur (puissance), mais en aucun cas la consommation d'un véhicule ne se mesure en chevaux.

Dans le cadre du présent travail nous nous intéressons d'abord aux causes de la dissipation d'énergie ainsi qu'aux métriques associées à la mesure des performances. Nous définissons aussi des métriques de complexité des circuits intégrés qui permettent des comparaisons dans l'espace performance/complexité. Ensuite nous étudions le comportement des aléas dus aux collisions entre transitions.

Ce travail s'étant déroulé en étroite liaison avec le projet Européen ESPRIT IV HIPERLOGIC (# 20.023), la seconde partie sera consacrée aux technologies développées dans le cadre de ce projet ainsi qu'aux méthodologies et outils de conception spécifiques. Les technologies en question utilisent des transistors de 0,1 μ m de longueur de canal grâce à un procédé de grille en forme de **T**. Ces dispositifs sont intégrés soit sur un substrat de Silicium sur isolant (SOI) soit sont empilés selon la dimension verticale en deux couches de SOI (SOI 3D) : l'une pour les transistors de type N et l'autre pour les transistors de type P. Ces technologies sont développées à l'IMS (Institut pour la Microélectronique de Stuttgart - Allemagne).

Les opérateurs arithmétiques combinatoires ont été choisis comme cas d'étude pour l'évaluation de ces technologies. Ce choix est délibéré; il se justifie par la présence de compétences en arithmétique combinatoire au sein du groupe ainsi que par la richesse algorithmique. Les opérateurs choisis sont l'addition, la multiplication et la division. Le chapitre suivant sera consacré aux architectures de ces opérateurs et à l'évaluation de la consommation dans ces architectures en général.

Au dernier chapitre nous présenterons les résultats de la comparaison des technologies SOI et SOI 3D. Le véhicule de cette comparaison sera le multiplieur 16x16 bits. Il sera implanté à l'aide des outils présentés au second chapitre. Les critères de comparaison et de mesure des performances seront ceux étudiés au premier chapitre.

ENGLISH INTRODUCTION

"Apprendre c'est se contredire", C. Baudelaire

The reduction of the energy necessary to achieve a given operation (or action) is not only related to microelectronics. It is a constant in the history of technologies. At nearly 5000 years interval, the wheel and the transistor serve the same needs.

Since the invention of the transistor in 1947 the number of operations per Joule in a circuit has increased by several orders of magnitude. This is due to the increasing needs for portable tools (communication and computing tools).

The portability of a tool is inversely proportional to its weight, i.e. inversely to the quantity of available energy. This is the reason why it will be focused here on energy consumption reduction and not on power. Indeed, power is a time average of the energy, and consequently contains less information. Moreover, the current expression "power consumption" is improper. Contrary to energy, power cannot be consumed. As an example, a car consumes a certain quantity of fuel (chemical energy) per mile. This quantity is proportional to the number of horsepower. but under no circumstances the consumption of a car can be measured in horsepower.

In the first part of the present work, the performance metrics are presented. Also complexity metrics are defined; these allow to quantify the complexity of an integrated circuit. These scales are used for a performance/complexity space exploration. Then the phenomena of colliding transitions and glitches generation and propagation are investigated.

As this work is closely related to the European project ESPRIT IV HIPERLOGIC (# 20.023), the second part is devoted to the technologies that are developed in the frame of this project as well as to the specific design tools. These technologies make use of the 0.1 μ m channel length transistors with **T** shaped gates. These devices are either integrated on a Silicon On Insulator substrate (SOI), or stacked in the vertical dimension in two layers of SOI (3D SOI) : the N type transistors are fabricated in the first SOI layer while the P type ones are on the second SOI layer. These technologies are developed at the IMS (Institute for Microelectronics of Stuttgart - Germany).

Combinatorial arithmetic operators were selected as demonstrators for the evaluation of these technologies. This choice is deliberate. The chosen operators are : addition, multiplication and division. The next chapter is devoted to arithmetic operators architecture exploration and to the evaluation of the energy dissipation.

The comparisons results of the SOI and 3D-SOI are presented in the last chapter. The comparison vehicle is a 16x16 bits multiplier based on a 4:2 reduction tree. It is implemented with the tools presented in the second chapter. The comparison criteria are those discussed in the first chapter.

CHAPITRE PREMIER : FAIBLE CONSOMMATION

I.1. INTRODUCTION

Contrairement à d'autres technologies où l'énergie consommée est statique (bipolaire, AsGa etc...), la majeure partie de la consommation d'un circuit en CMOS est dynamique. Elle correspond pour la plus grande part à la charge/décharge de capacités. Un circuit CMOS ne consomme de façon significative que quand il est actif. La consommation statique d'un circuit CMOS peut parfois devenir importante, surtout en submicronique profond, mais d'une part, la réduction de cette composante est plutôt du ressort de la technologie, et d'autre part, nous nous intéressons ici à une classe de circuits (opérateurs arithmétiques combinatoires) très actifs dans lesquels la part de consommation statique est faible.

Dans ce premier chapitre nous commencerons par une présentation de l'impact que peuvent avoir les choix aux différents niveaux d'abstraction sur la consommation. Ensuite nous passerons en revue les différentes composantes de la dissipation d'énergie et montrerons comment la redondance de l'information peut réduire l'activité. La troisième partie traitera des différentes méthodes de réduction d'échelle et de leur influence sur la consommation, puis seront présentées dans la quatrième partie les différentes métriques de performance qui sont utilisées tout au long de cette thèse. La cinquième partie traitera des cellules complexes et de leur bon usage pour l'augmentation des performances du circuit. Finalement, sera abordé le problème des collisions entre transitions et des aléas générés.

I.2. HIÉRARCHIE DE LA CONSOMMATION

L'espace de conception des circuits intégrés peut-être hiérarchisé par niveaux d'abstraction. Une hiérarchisation possible est représentée sur la figure 1.1 [GaKu83, ChBr96].

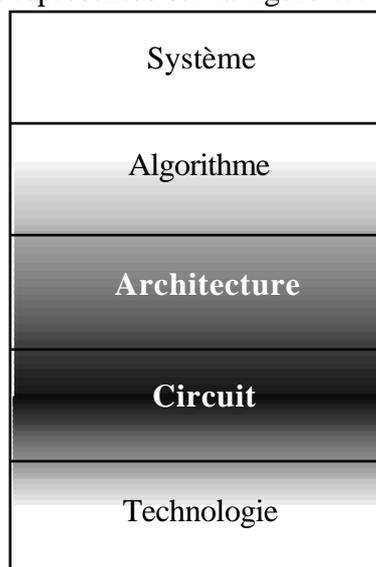


Figure 1.1 : Divers niveaux d'abstraction

Une réduction substantielle de la consommation ne peut être obtenue que si les efforts se portent conjointement sur plusieurs niveaux, sinon il y a le risque de perdre à l'étage inférieur ce qui a été gagné à l'étage supérieur. Ainsi, la frontière entre niveaux est plutôt floue. Voici

sommairement les actions qui peuvent être entreprises à chacun des niveaux afin de réduire la consommation :

- **Systeme** : Partitionnement, mise en veille.
- **Algorithme** : Complexité, granularité, parallélisme, régularité, représentation des données, synchronisation.
- **Architecture** : Parallélisme, régularité, jeu d'instruction, notations...
- **Circuit** : Famille logique, optimisation logique, taille des transistors...
- **Technologie** : Interconnexions, SOI, DTMOS, polarisation substrat, double V_t .

Le domaine exploré par cette thèse est centré sur les niveaux "circuit" et "architecture" avec une forte interaction avec la technologie (figure 1.1); les rares choix algorithmiques étant dictés par la seule vitesse. La réduction de la consommation au niveau système est donc ici totalement occultée.

- **Algorithme** : Les choix algorithmiques effectués dans cette thèse sont assez peu nombreux; et concernent surtout la division. Du fait de la grande diversité des algorithmes de division et de leurs performances variées, une approche exhaustive sortirait du cadre de ce travail. Notre choix s'est donc appuyé sur d'autres travaux [Mon95, Rou97, ErLa94]
- **Architecture** : En termes d'architecture, le parallélisme est massivement exploité. Il est facilité du fait de la régularité des circuits étudiés. Pour ce qui est de la division, bien que la base deux offre les meilleures performances, nous verrons qu'une base plus grande pourrait être mieux adaptée à la réduction de la consommation d'énergie.
- **Circuit** : Les axes explorés au niveau circuit sont multiples, allant du filtrage de transitions redondantes à l'usage des cellules complexes, en passant par le choix de la tension d'alimentation optimale. La grande constante est ici la famille logique : CMOS statique.
- **Technologie** : Diverses options technologiques ont pu être explorées du fait de la participation au projet européen HIPERLOGIC (Esprit IV) [HPL97, HPL98]. Une forte interaction avec les technologues a permis une prise en compte mutuelle des contraintes (choix des tensions de seuil - courant de fuite - couches d'interconnexion pour la technologie tridimensionnelle etc.)

Après ce rapide tour d'horizon des facteurs sur lesquels il sera possible d'intervenir, examinons maintenant les sources et mécanismes de dissipation d'énergie dans les circuits CMOS.

I.3. SOURCES DE DISSIPATION

Bien que pour des raisons de fiabilité il soit important de réduire la dissipation maximale, nous nous intéresserons uniquement à la réduction de la dissipation moyenne car c'est elle qui conditionne la "portabilité" d'un système. D'autre part, les actions entreprises pour réduire la dissipation moyenne le plus souvent réduisent aussi la dissipation maximale. Mais, comme nous pouvons le voir dans [BDPS98], il n'en est pas toujours ainsi.

La dissipation d'énergie d'un circuit en CMOS peut-être décomposée en trois composantes majeures :

- dynamique
- court-circuit
- statique.

La composante dynamique est largement dominante dans les circuits combinatoires à fort taux d'activité tels que les opérateurs arithmétiques. Cette section comporte trois parties consacrées chacune à l'une de ces composantes de la consommation.

I.3.1. DISSIPATION DYNAMIQUE

L'énergie dissipée par un bloc fonctionnel ainsi que son délai dépendent fortement des données. Par exemple, si un circuit combinatoire reçoit successivement le même vecteur en entrée, le délai et la consommation seront nuls. Néanmoins, les approches pour l'évaluation sont très différentes qu'il s'agisse de dissipation ou de délai. Pour le délai nous nous intéressons au pire cas, donné par le chemin le plus lent (chemin critique); pour la dissipation, c'est la moyenne qui compte. L'approche sera donc statistique. Il peut être intéressant de remarquer que dans le cas des circuits asynchrones, c'est également la valeur moyenne du délai qui importe, alors que la puissance maximale compte pour le dimensionnement des pistes d'alimentation [BDPS98].

Dans cette partie seront exposés les points suivants : pourquoi une variation de tension engendre-t'elle une dissipation d'énergie, à quoi est due une transition, transitions utiles et redondantes, et finalement comment la redondance de l'information peut-elle réduire l'activité?

I.3.1.1. GENÈSE D'UNE TRANSITION

De l'énergie est dissipée quand une tension change, entraînant la charge ou la décharge d'une capacité parasite C_1 .

Pour une transition 0 \rightarrow V_{dd} , l'interrupteur 1 est ouvert et l'interrupteur 2 est fermé (figure 1.2), et une énergie $E_{0 \rightarrow 1} = C_1 * V_{dd}^2$ est puisée dans l'alimentation V_{dd} . L'énergie $E_C = \frac{1}{2} C_1 * V_{dd}^2$ est stockée dans la capacité C_1 . L'autre moitié ($\frac{1}{2} C_1 * V_{dd}^2$) est dissipée.

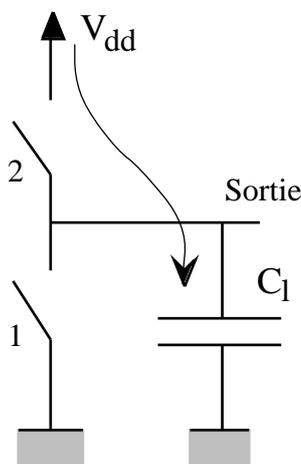


Figure 1.2 : Mécanisme de dissipation dynamique

Pour une transition $V_{dd} = 0$, l'interrupteur 2 est ouvert. Aucune énergie n'est puisée dans l'alimentation, mais l'énergie stockée dans C_1 est dissipée.

Pour illustrer ce phénomène, supposons que le courant I passant par les interrupteurs est constant (figure 1.3) - le résultat serait le même quelque soit le courant.

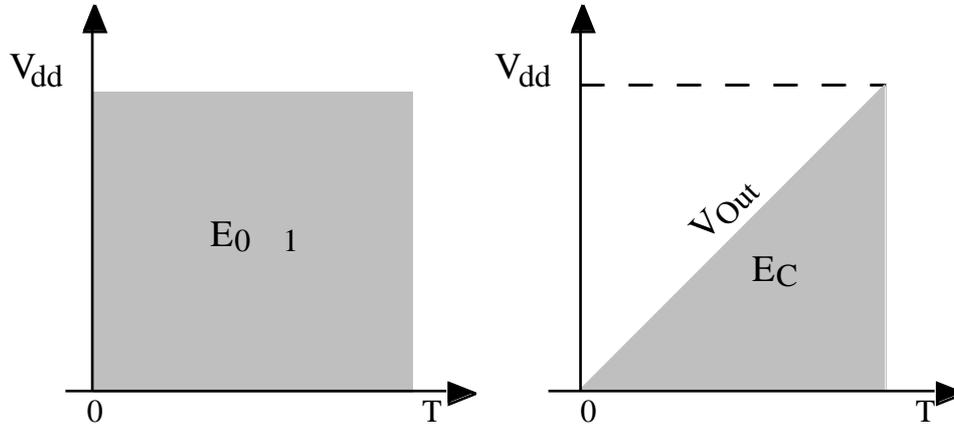


Figure 1.3 : Charge et décharge de C_1

Nous avons :

$$I = \frac{dq}{dt} ; \quad q = C_1 \cdot V_{out}. \quad (1.1)$$

L'énergie E_s stockée dans la capacité est :

$$E_s = \int_0^{V_{dd}} V_{out} \cdot I \, dt = \int_0^{V_{dd}} \frac{q}{C_1} \frac{dq}{dt} \, dt = \int_0^{V_{dd}} \frac{q}{C_1} \, dq \quad (1.2)$$

$$E_s = \frac{1}{2} C_1 * V_{dd}^2 \quad (1.3)$$

L'énergie totale dissipée E_d est :

$$E_d = \int_0^{V_{dd}} V_{dd} \cdot I \, dt = V_{dd} \int_0^{V_{dd}} C_1 \, dq \quad (1.4)$$

$$E_d = C_1 * V_{dd}^2$$

La moitié de l'énergie est perdue.

I.3.1.2. REDONDANCE ET ACTIVITÉ

Les transitions en sortie sont évidemment provoquées par des transitions des entrées. Dans une machine synchrone, toute l'activité dérive finalement des transitions de l'horloge. Voyons les différents types d'activité qui peuvent être générés.

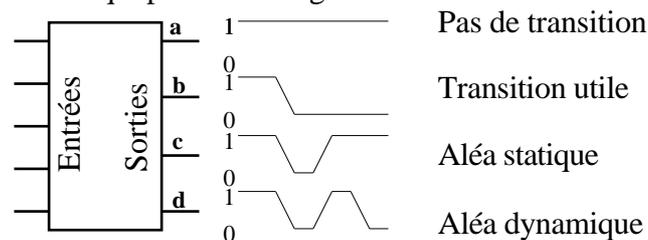


Figure 1.4 : Taxinomie des transitions

Dans cet exemple, **a** ne change pas, **b** effectue une transition utile, **c** effectue deux transitions redondantes et finalement **d** a deux transitions redondantes et une transition utile. Il est facile d'analyser les transitions utiles, car elles obéissent aux lois de l'algèbre booléenne. Les transitions redondantes sont dues à des chemins de délais différents depuis les entrées jusqu'à la même sortie, et sont donc plus difficiles à analyser et à minimiser.

Dans l'exemple précédent (figure 1.4) il y a sur les chronogrammes de sortie deux fois plus de transitions redondantes que de transitions utiles.

En CMOS statique, des transitions sont injectées dans un circuit par un changement des entrées. Nous appelons activité le nombre moyen de transitions injectées, divisé par le nombre total de changements pour toutes les paires de vecteurs possibles.

En CMOS dynamique, il y a une phase de précharge avant chaque phase d'évaluation. L'activité ne dépend donc pas des transitions en entrée, mais seulement des valeurs des entrées. Néanmoins, chaque valeur logique "0" entraîne 2 transitions : une pour la décharge et une autre pour la précharge suivante.

I.3.1.3. EXEMPLE DE RÉDUCTION D'ACTIVITÉ [GUAS97]

Supposons qu'au lieu de s'intéresser à la valeur des deux bits x et y {00, 01, 10, 11}, nous nous intéressions à leur somme $s = x+y$ {0, 1, 2}. Introduisons $o=x \oplus y$ et $a=x \cdot y$, o et a ne sont pas indépendants ($a \oplus o = x$), ils véhiculent donc moins d'informations que x et y .

Tableau 1.1 : Exemple de code redondant

x	y	$o=x \oplus y$	$a=x \cdot y$	$s=x+y$	$o + a$
0	0	0	0	0	0
0	1	1	0	1	1
1	0	1	0	1	1
1	1	0	1	2	2

Notons que $s = x+y = o+a$. Si x et y sont équiprobables, la probabilité des lignes du tableau 1.1 sera la même : 1/4.

Pour compter les transitions, le tableau 1.2 est construit avec les anciennes et les nouvelles valeurs de (x, y) , et le nombre de transitions entre elles (distance de Hamming).

Tableau 1.2 : Distances de Hamming

xy	00	01	10	11
00	0	1	1	2
01	1	0	2	1
10	1	2	0	1
11	2	1	1	0

Toutes les cases sont équiprobables. Le nombre moyen de transitions est $16/16 = 1$. Construisons maintenant la même table en recodant (x, y) par (o, a) (tableau 1.3).

Tableau 1.3 : Nombre de transitions

oa	00	10	10	11
00	0	1	1	2
10	1	0	0	1
10	1	0	1	1
11	2	1	1	0

Encore une fois, toutes les cases sont équiprobables. Le nombre moyen de transitions est maintenant de 12/16. Il y a un gain de 25% en activité.

Notons que par construction, le couple oa ne peut pas prendre la valeur 01 car $a \leq o$ (tableau 1.1).

I.3.1.4. RENDRE L'INFORMATION REDONDANTE

Supposons que les valeurs de x et de $y \in \{00, 01, 10, 11\}$ nous intéressent. Nous ajoutons un troisième bit i , et nous codons : $a = x \oplus i$ et $b = y \oplus i$. Il est aisé de retrouver x et y à partir de (i, a, b) , en effet : $x = a \oplus i$ et $y = b \oplus i$. Ce code est redondant car pour chaque valeur de (x, y) il y a deux couples (i, a, b) possibles.

Tableau 1.4 : Codage redondant

iab xy	00	01	10	11
iab	000	001	010	011
	111	110	101	100

La valeur de i est choisie de manière à réduire le nombre de transitions. Le tableau 1.5 donne le nouveau triplet (i, a, b) en fonction de la nouvelle valeur de (x, y) et de l'ancienne valeur de (i, a, b) .

Tableau 1.5 : Codage redondant

xy	00	01	10	11
000	000	001	010	100
001	000	001	101	011
010	000	110	010	011
011	111	001	010	011
100	000	110	101	100
101	111	001	101	100
110	111	110	010	100
111	111	110	101	011

Encore une fois, toutes les possibilités sont équiprobables. Le nombre minimal de transitions est 0, le maximal est 2 et la moyenne est de 24/32. Comme dans l'exemple précédent, l'activité est réduite de 25%.

Formellement, si tous les bits sont indépendants et équiprobables, l'activité moyenne est :

$$\frac{1}{2^n} \sum_{i=0}^n i C_n^i \quad (1.5)$$

si uniquement un maximum de $n/2$ bits est autorisé à changer, l'activité devient [StBu95] :

$$\frac{1}{2^n} \sum_{i=0}^{n/2} C_{n+1}^i \quad (1.6)$$

I.3.1.5. COMPROMIS

Dans les paragraphes précédents, des exemples sont fournis où la redondance de l'information est exploitée pour réduire l'activité, et par conséquent la consommation d'énergie. Mais afin d'aboutir à cette réduction, des portes logiques supplémentaires ont été ajoutées avec leur cortège de majorations du délai et de la consommation. Nous ne nous préoccupons pas de la surface ici... il faudra donc rechercher le compromis le plus favorable.

I.3.2. COURT CIRCUIT

Dans le paragraphe précédent, nous avons vu que la composante dynamique de l'énergie consommée lors d'une transition ne dépendait pas de la durée des rampes de montée et de descente. Néanmoins, des temps de transition (montée ou descente) non nuls entraînent une conduction simultanée des transistors N et P qui donne lieu à un courant de court-circuit entre V_{dd} et la masse. Ce courant existe tant que la tension d'entrée V_e satisfait l'inégalité suivante :

$$V_{tn} < V_e < V_{dd} - |V_{tp}| \quad (1.7)$$

Ce court-circuit n'existe pas en logique dynamique, car les phases de précharge et d'évaluation n'ont jamais lieu simultanément. C'est donc un problème spécifique au CMOS statique.

En supposant une rampe linéaire de durée t_r ainsi que des gains équivalents pour les transistors N et P : $\beta_p W_p = \beta_n W_n$, il est alors aisé de calculer le courant moyen ainsi que la composante de court-circuit E_{cc} de l'énergie consommée durant la transition :

$$E_{cc} = \frac{1}{2} t_r \left(V_{dd} - V_{tn} - |V_{tp}| \right)^3 \quad (1.8)$$

L'énergie de court-circuit peut-être minimisée en équilibrant les durées des transitions en entrée et en sortie [Vee84].

Il est important de noter que si $V_{dd} < V_{tn} + |V_{tp}|$, le courant de court-circuit est éliminé car les transistors N et P ne seront jamais conducteurs simultanément. Mais une telle réduction de la tension d'alimentation n'est pas sans impact sur les performances... De plus les sorties passent (et éventuellement restent) en état de haute impédance.

I.3.3. DISSIPATION STATIQUE

La composante statique de la dissipation dans les circuits en CMOS est la moins importante dans notre cas. Elle peut être décomposée en deux parties : le courant de fuite des diodes polarisées en inverse et le courant sous seuil. Le courant sous-seuil est dû à la diffusion des porteurs entre la source et le drain quand la tension de la grille est supérieure au point de faible inversion, et inférieure à V_t [ChBr96, Sze81]. La dissipation statique peut être réduite en polarisant le substrat pour les technologies Silicium massif, ou encore en utilisant deux tensions de seuil différentes.

I.4. RÉDUCTION D'ÉCHELLE

"Si parva licet componere magnis"¹, Virgile

Dans ce paragraphe nous examinons les effets de la réduction de la taille des dispositifs MOS sur le comportement des circuits. Trois approches seront considérées [WeEs94, RaPe97] :

- réduction à champ constant
- réduction à tension constante
- réduction latérale

Tableau 1.6 [WeEs94] : Réduction d'échelle

PARAMÈTRE	MODÈLE DE RÉDUCTION		
	Champ constant	Tension constante	Réduction latérale
Longueur (L)	1/K	1/K	1/K
Largeur (W)	1/K	1/K	1
Tension (V_{dd})	1/K	1	1
Épaisseur oxyde (t_{ox})	1/K	1/K	1
Courant ($I = (W \cdot V^2)/(L \cdot t_{ox})$)	1/K	K	K
Transconductance (g_m)	1	K	K
Profondeur de jonction (X_j)	1/K	1/K	1
Dopage substrat (N_A)	K	K	1
Champ électrique dans l'oxyde (E)	1	K	1
Zone de déplétion (d)	1/K	1/K	1
Capacité de charge ($C = W \cdot L / t_{ox}$)	1/K	1/K	1/K
Délai ($V \cdot C / I$)	1/K	1/K ²	1/K ²
INFLUENCE			
Puissance statique (P_s)	1/K ²	K	K
Puissance dynamique (P_d)	1/K ²	K	K
Énergie par opération	1/K ³	1/K	1/K
Surface (W.L)	1/K ²	1/K ²	1/K
Densité de puissance (V.I/A)	1	K ³	K ²
Densité de courant	K	K ³	K ²

I.4.1. RÉDUCTION À CHAMP CONSTANT

La théorie de réduction à champ constant dérive d'un modèle formulé par Dennard et alii en 1973 [Den73]. Elle consiste à conserver les caractéristiques électriques du transistor en réduisant tous ses paramètres par un facteur constant sans dimension K, ce qui résulte en une réduction de l'énergie par opération par un facteur K³ et une réduction du délai par un facteur K, alors que la densité de puissance (par unité de surface) reste constante (Tableau 1.6). Ce

¹S'il est permis de comparer les petites choses aux grandes

modèle de réduction est séduisant; il était peu utilisé dans le passé au profit de la réduction à tension constante. Il est maintenant d'actualité car avec l'avènement des technologies fortement submicroniques, la réduction à tension constante atteint ses limites.

I.4.2. RÉDUCTION À TENSION CONSTANTE

La réduction d'échelle à tension constante est souvent préférée, car elle permet à effort technologique identique un gain de vitesse bien plus important que la réduction à champ constant. L'effort technologique - du moins en termes d'investissement - est surtout déterminé par la lithographie (réduction de L et W). Le tableau 1.6 montre l'évolution des divers paramètres du dispositif selon ce modèle de réduction. Nous pouvons voir que le délai est divisé par K^2 au lieu de K pour une réduction à champ constant. L'inconvénient de cette méthode est la faible diminution de l'énergie par opération (seulement comme K) ce qui entraîne une augmentation inquiétante de la densité de puissance (comme K^3). Une réduction d'échelle de ce type doit donc obligatoirement s'accompagner d'une réduction de la dissipation à des niveaux d'abstraction plus élevés (architecture, algorithme...).

I.4.3. RÉDUCTION LATÉRALE

La réduction latérale consiste à ne réduire que la longueur du canal L , le plus souvent pour des raisons économiques. Comme il sera montré au chapitre suivant, celle-ci peut être faite indépendamment de la résolution de la lithographie permettant les mêmes gains de vitesse que la réduction à tension constante, mais une moins grande densité d'intégration.

I.4.4. INTERCONNEXION

Parallèlement à l'augmentation des performances intrinsèques des dispositifs, il faut aussi se soucier de leur interconnexion. En effet, plus la gravure est fine, plus le délai intrinsèque est faible et plus le délai dû aux interconnexions devient préoccupant. Il est donc intéressant de minimiser la somme de ces deux contributions. La finesse de gravure optimale actuellement prévue par la SIA [SIA97] est de 180nm. Cette valeur évolue au cours du temps, et en fonction des nouveaux matériaux (Cuivre, etc...); elle était d'environ 300nm en 1996 [Bor97]. L'intégration tridimensionnelle présentée au chapitre suivant [AAGC98, AAG98] est un moyen de réduire la longueur des interconnexions; son impact sur les performances sera évalué au cours du dernier chapitre.

I.4.5. LIMITATIONS

L'inconvénient de l'approximation au premier ordre tel que présentée dans ce paragraphe est surtout de donner l'impression qu'il est possible d'atteindre des dimensions nulles, des champs infinis et des tensions de seuil nulles. Mais la réduction des dispositifs MOS se heurte à des obstacles à la fois théoriques [ChBr96, IwMo97] et pratiques [SIA97]. Au delà du MOS, d'autres dispositifs sont envisageables, auxquels s'appliquent d'autres schémas de réduction d'échelle [Gau97, FDCT98, LeTo97, ToLe96]

I.5. MÉTRIQUE ET PERFORMANCE

Le mot “performance” est subjectif, et son emploi courant en microélectronique peut être trompeur. En effet, performance peut vouloir dire plusieurs choses selon l’objectif recherché : débit, énergie par opération ou bien énergie par débit. Ces problèmes de métrique ont été traités entre autres dans [CPi97a, CPi97b] par C. Pigué. Nous passerons ici en revue les métriques qui nous intéresseront pour la suite.

I.5.1. DÉBIT

Si la vitesse est la contrainte principale du circuit, alors la performance sera le délai sans aucune considération des autres paramètres (surface, dissipation, etc.). C’est souvent le cas des microprocesseurs destinés aux serveurs ou au calcul intensif; certains de ces systèmes devant même avoir recours à un refroidissement à eau...

Le délai d’une porte CMOS est une fonction décroissante de la tension d’alimentation V_{dd} . L’équation (1.9) donne une approximation au premier ordre du délai D en fonction de V_{dd} .

$$D \sim \frac{V_{\text{dd}}}{(V_{\text{dd}} - V_t)^2} \quad (1.9)$$

est une constante qui dépend de la technologie. Cette équation ne rend pas directement compte des effets canaux courts qui pour certains peuvent être inclus dans la tension de seuil. C’est le cas de l’effet DIBL (Drain Induced Barrier Lowering) où le drain agit comme une seconde grille, ce qui peut être traduit simplement par une réduction linéaire de la tension de seuil en fonction de V_{dd} . V_t s’écrit alors : $V_t = V_{t0} - k \cdot V_{\text{dd}}$ [SaNe91].

L’équation 1.9 montre qu’il est toujours possible de satisfaire la contrainte de délai simplement en augmentant la tension d’alimentation - néanmoins dans les limites de tolérance de la technologie (figure 1.5). Le délai est mesuré en nanosecondes (ns). Le débit - soit l’inverse du délai - sera donné en MHz car nous nous intéressons aux systèmes synchrones.

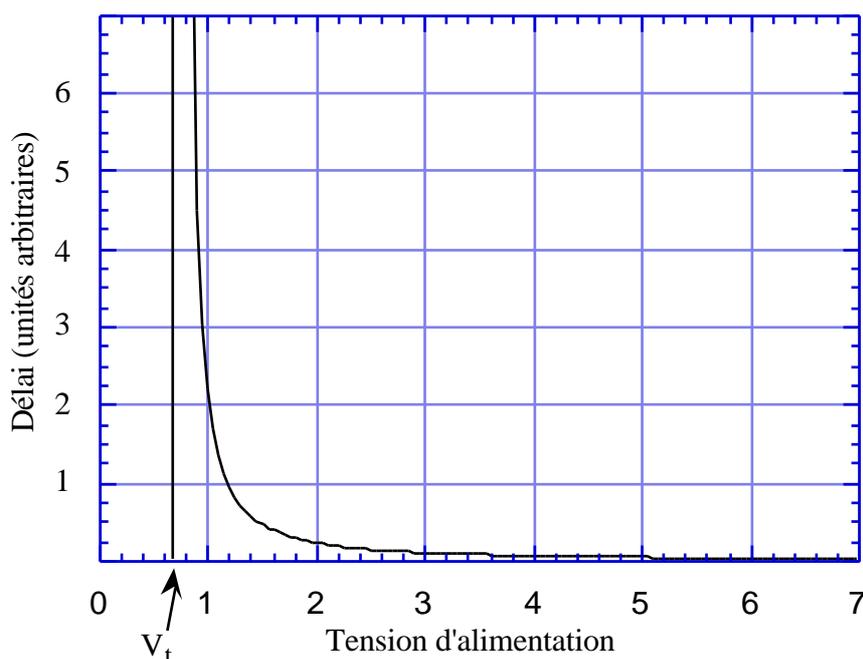


Figure 1.5 : Délai d’une porte logique en fonction de la tension d’alimentation

D'autres modèles de délai sont bien sûr nécessaires [ADT97] pour faire des prédictions plus précises.

I.5.2. AUTONOMIE

Si la seule contrainte imposée lors de la conception d'un circuit est l'autonomie (c'est à dire la durée de vie de la pile ou de la batterie) comme dans une montre ou un implant par exemple, la performance s'exprimera alors en terme d'énergie par opération (EPO). EPO est en fait la même chose que le produit puissance-délai (PDP), mais le terme "énergie par opération" est ici préféré car le délai (ou débit) n'intervient pas dans cette métrique. En effet, l'énergie nécessaire pour effectuer une opération est proportionnelle à V_{dd}^2 (disons $EPO = \alpha \cdot V_{dd}^2$), et la puissance dépend de la fréquence f :

$$P = \alpha \cdot V_{dd}^2 \cdot f = \alpha \cdot V_{dd}^2 / D; \text{ d'où, } P \times D = PDP = \alpha \cdot V_{dd}^2 = EPO \quad (1.10)$$

EPO est donc une fonction monotone de la tension d'alimentation V_{dd} (figure 1.6), ce qui veut dire que l'EPO peut être réduite à volonté, simplement en abaissant la tension d'alimentation du circuit. Cette approche est bien sûr incompatible avec l'augmentation du débit. Par ailleurs, il y a aussi des limites technologiques à la réduction de la tension d'alimentation. L'EPO est mesurée en picojoules (pJ).

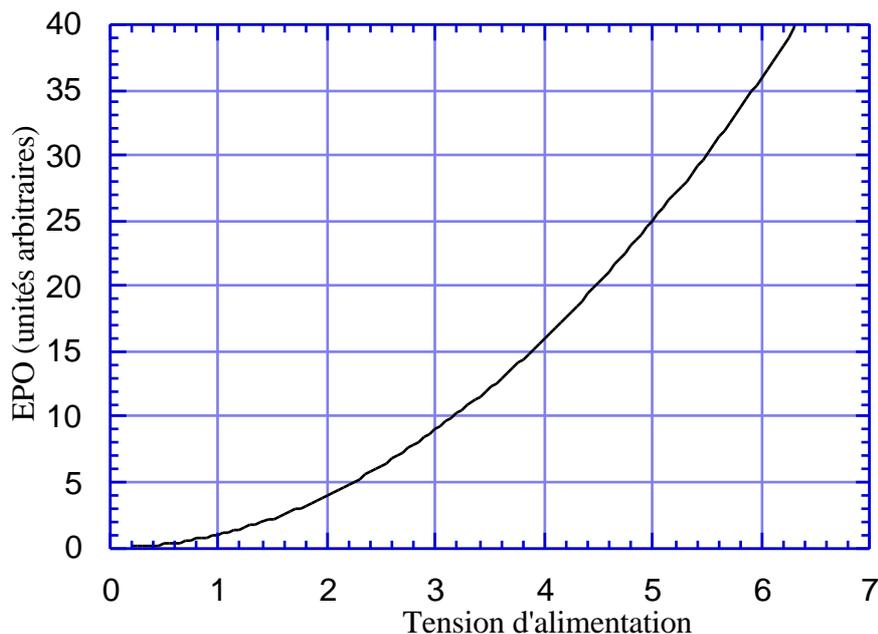


Figure 1.6: Énergie par opération en fonction de la tension d'alimentation

I.5.3. EFFICACITÉ

"Tous les moyens sont bons quand ils sont efficaces.", Jean-Paul Sartre, Les mains sales

Que se passe-t'il si les deux contraintes, autonomie et débit, doivent être satisfaites? Cela revient à minimiser simultanément l'EPO et le délai. Un compromis est alors nécessaire, car d'une part il faut diminuer V_{dd} pour réduire l'EPO (figure 1.6), et d'autre part il faut l'augmenter pour réduire le délai (figure 1.5). Ce compromis est le minimum du produit énergie-délai (EDP).

$$\text{EDP} = \text{EPO} \times \text{Délai} \sim \frac{V_{\text{dd}}^3}{(V_{\text{dd}} - V_t)^2} \quad (1.11)$$

En utilisant les modèles approchés d'énergie et de délai présentés dans les paragraphes précédents, il est aisé de résoudre l'équation suivante :

$$\frac{\partial \text{EDP}}{\partial V_{\text{dd}}} = 0 \quad (1.12)$$

La solution est $V_{\text{dd}} = 3V_t$. En fait, il s'agit d'une approximation au premier ordre, qui peut être affinée par des considérations d'ordre technologique. Cette tension optimale est plus faible en SOI qu'en Silicium massif par exemple [Col97]. Sur la figure 1.7, EDP est tracé en fonction de V_{dd} pour un oscillateur en anneau en technologie ATMEL $0,7\mu\text{m}$ avec $(|V_{\text{tn}}| + |V_{\text{tp}}|)/2 = 0,7\text{V}$. Le produit énergie-délai est mesuré en pJ/MHz.

Il existe d'autres métriques que celles présentées ici, comme la métrique "ET²" qui est le produit de l'énergie par opération et du carré du délai. Cette fonction permet d'éliminer V_{dd} et ainsi de comparer des circuits n'ayant pas la même tension d'alimentation. Mais elle suppose le modèle simple de délai de l'équation (1.9). La définition d'une métrique indépendante de la tension d'alimentation en sub-micronique profond implique de prendre en compte un modèle de délai plus complet.

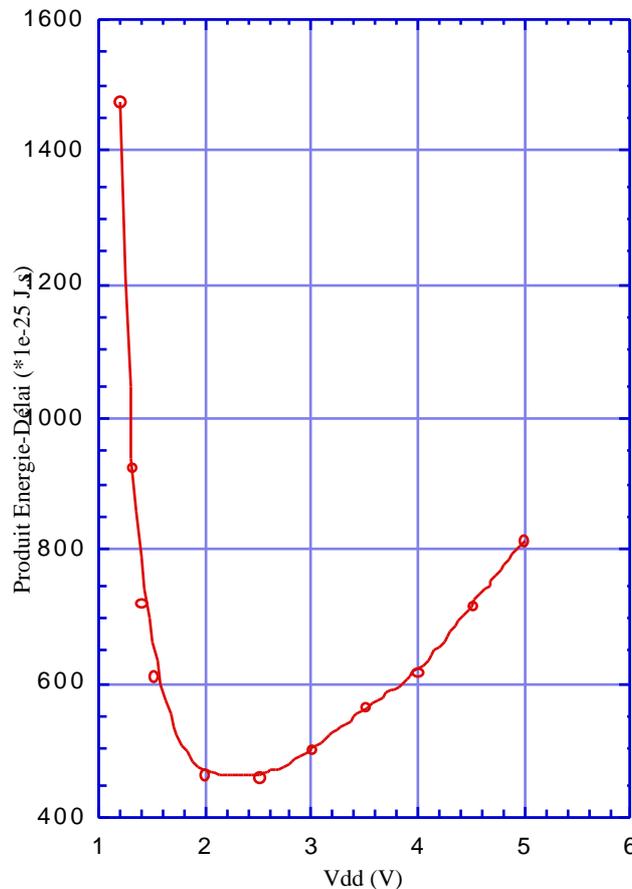


Figure 1.7 : EDP en fonction de V_{dd}

I.6. CELLULES COMPLEXES [ASG98]

De prime abord il est tentant de définir la complexité comme étant le nombre de transistors sans se soucier du niveau d'abstraction auquel nous nous plaçons. Cette définition a l'avantage d'être simple, mais elle est très insatisfaisante. En effet, si elle est vraie au niveau système, elle est fautive au niveau portes. L'implantation en cellules complexes d'une fonction donnée prend généralement moins de transistors que l'implantation de la même fonction en cellules simples (figure 1.8). La réduction du nombre de transistors a un double coût :

- un coût logique
- un coût en terme de performance

Le coût logique correspond à la perte des nœuds internes (sur la figure 1.8). Ce coût est purement formel et n'affecte pas le comportement du circuit. Pour être précis, le coût logique est d'un nœud par paire de transistors en moins. L'impact des cellules complexes sur la performance sera traité plus loin dans ce chapitre.



Figure 1.8 : Construction de cellules complexes

La décomposition des portes logiques pour en améliorer la vitesse ou la consommation est un sujet souvent abordé [Mas91, CPi95, MeRe79]. Diverses implantations d'une fonction donnée sont analysées, mais sans échelle de complexité. La tentative sera faite dans ce qui suit de définir une telle échelle.

La métrique de complexité dépend du niveau d'abstraction en question. Ici, seuls deux niveaux sont considérés : le niveau porte logique et le niveau transistor. En premier lieu, la tentative est faite de quantifier la complexité, ensuite le protocole expérimental mis en place est décrit, et enfin les résultats seront présentés et analysés.

I.6.1. NIVEAU PORTE LOGIQUE

Au niveau porte logique, la complexité d'un circuit est liée à la longueur du chemin critique. Il paraît donc adéquat de définir la complexité à ce niveau comme étant la profondeur logique nécessaire à la réalisation d'une fonction donnée. Dans ce cas, il s'agit d'une porte Nand (Non Et) à 16 entrées. Elle peut être implantée soit en une seule couche logique avec 16 transistors en série (correspondant à une complexité = 1), soit en trois couches comme sur la figure 1.9a (complexité = 3), soit encore en 5 couches (figure 1.9b) pour une complexité de 5. D'autres schémas de décomposition sont possibles, mais n'ont été retenus que ceux qui font appel à des portes identiques de manière à ne traiter qu'un problème à la fois. De cette façon, l'échelle de complexité au niveau porte ne dépend pas de l'implantation physique (niveau transistor) qui sera traitée au paragraphe suivant. Les performances peuvent maintenant être tracées sur cette échelle de complexité. Le protocole expérimental sera décrit plus loin. Il est important de noter que cette définition de la complexité dépend de la fonction logique.

I.6.1.1. NIVEAU TRANSISTOR

Au niveau transistor il est nécessaire d'avoir une granularité plus fine. A ce niveau, la complexité d'une porte ne doit pas dépendre de sa fonction logique. La complexité peut alors être considérée comme étant proportionnelle au nombre de transistors en série dans une porte quelconque. Les comparaisons deviennent ainsi aisées après normalisation. Sur cette échelle, tous les résultats seront normalisés par bit.

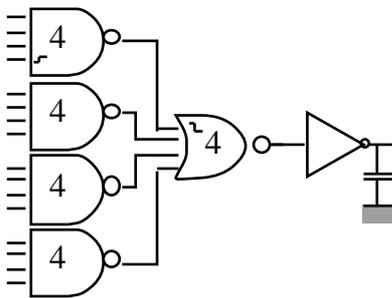


Figure 1.9a : Complexité = 3

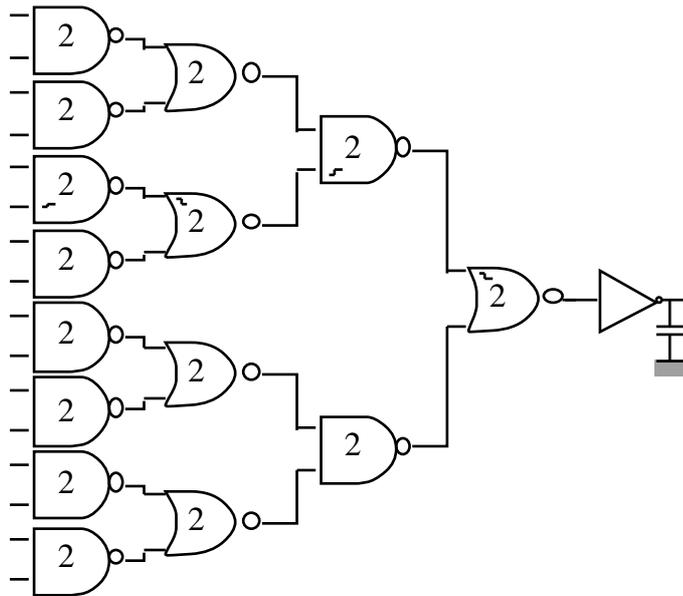


Figure 1.9b : Complexité = 5

I.6.2. PROTOCOLE EXPÉRIMENTAL

Ce paragraphe présente les circuits de test qui sont utilisés pour l'exploration de l'espace performance/complexité. Les résultats sont obtenus par simulations électriques HSPICE.

I.6.2.1. CIRCUITS TEST - NIVEAU PORTE

Le circuit de test utilisé est ici une porte Nand à 16 entrées. Il est commode de choisir un nombre d'entrées qui peut s'écrire sous la forme 2^{2n} , car ainsi la fonction peut être décomposée en un arbre équilibré avec des portes ayant toutes le même nombre d'entrées. Par exemple, il est possible de réaliser une Nand 16 avec soit uniquement des portes à 4 entrées (figure 1.9a), soit avec des portes à deux entrées (figure 1.9b) mais pas avec des portes à 8 entrées.

La capacité de sortie est la même dans tous les cas : $C_1 = 100\text{fF}$. Le délai est toujours mesuré dans le pire cas, c'est à dire que c'est toujours le transistor qui se trouve le plus loin de la sortie qui commute. Les entrées qui commutent sont marquées par le signe τ sur les figures 1.9a et 1.9b.

L'EPO est calculée en prenant le produit du courant moyen passant par V_{dd} durant la transition (i), de V_{dd} et du délai de la porte ().

$$\text{EPO} = i \cdot V_{\text{dd}} \cdot \tau \quad (1.13)$$

I.6.2.2. CIRCUITS TEST - NIVEAU TRANSISTOR

Comme l'échelle est ici le nombre de transistors en série, la fonction logique effectuée n'a pas d'importance. Nous considérons une porte Nand avec un nombre d'entrées variant de deux à huit (figure 1.10). Étant donné que toutes les mesures sont données par bit, la capacité de sortie est proportionnelle au nombre d'entrées : $C_1 = n \cdot 50\text{fF}$ avec n le nombre d'entrées. Le délai mesuré est aussi un pire cas.

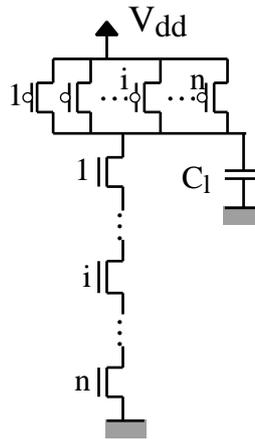


Figure 1.10 : Porte Nand à n entrées

Des simulations électriques ont été effectuées pour deux versions de la même technologie : Silicium massif et SOI. Cette technologie sera décrite au cours du chapitre suivant; les résultats obtenus sont résumés dans le paragraphe ci-dessous.

I.6.3. RÉSULTATS ET DISCUSSION

Les versions Silicium massif (Bulk) et Silicium sur isolant (SOI) de la même technologie CMOS sont comparées en utilisant les circuits test et les métriques présentées précédemment.

La figure 1.11 montre l'énergie par opération (EPO) et le produit énergie-délai (EDP) en fonction de la profondeur logique pour une porte Nand à 16 entrées chargeant 100fF.

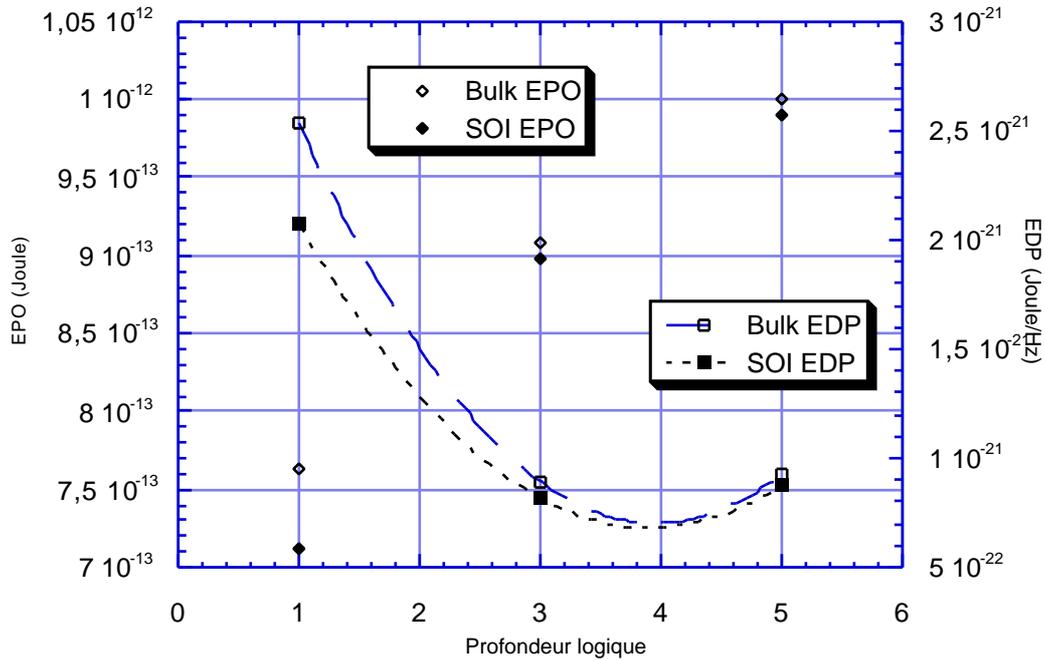


Figure 1.11 : EPO et EDP en fonction de la profondeur logique

La figure 1.12 montre les EPO et EDP normalisés par bit en fonction du nombre de transistors en série.

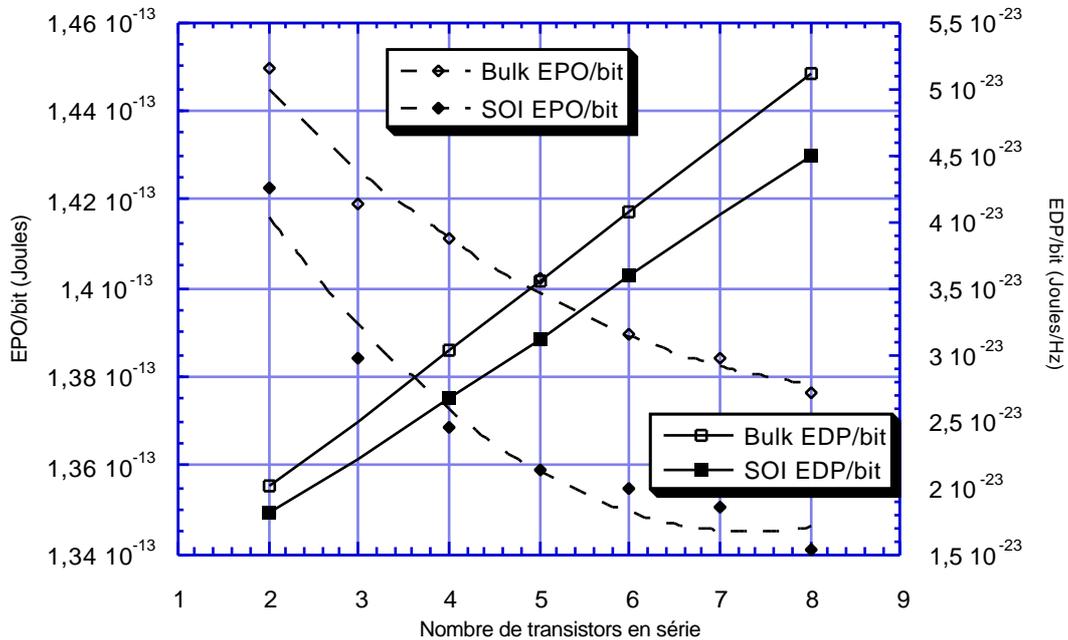


Figure 1.12 : EPO/bit et EDP/bit en fonction du nombre de transistors en série

Voici quelques commentaires et directives de conception issues de ces tests :

- La différence relativement faible entre le Silicium massif et le Silicium sur isolant s'explique ici par l'utilisation de la même capacité de charge dans les deux cas. Dans la réalité cette capacité sera plus faible en Silicium sur isolant, ce qui accentuerait les différences.

- Il y a une décomposition optimale en terme de produit énergie-délai (EDP) pour les grandes fonctions. Cet optimum n'existe pas pour l'EPO (figure 1.11). Dans ce cas, les implantations avec des profondeurs logiques de 3 et de 5 donnent environ le même EDP; il faut bien sûr choisir celle qui donne l'EPO la plus faible. La profondeur logique minimisant l'EDP dépend de la fonction étudiée et de la technologie.

- Sur la figure 1.12, la différence en termes d'EDP/bit entre SOI et Silicium massif augmente avec le nombre de transistors en série. Ce qui signifie que le SOI favorise les portes plus complexes. Cette différence croissante est essentiellement due aux très faibles capacités de jonctions en SOI qui permettent une commutation plus rapide d'un plus grand nombre de dispositifs en série. Les courbes EPO/bit sont parallèles pour le SOI et le Si massif avec une valeur plus faible pour le SOI.

- Aussi bien pour le SOI que pour le Si massif, l'EPO/bit décroît avec la complexité (nombre de transistors en série) alors que l'EDP/bit augmente (figure 1.12). Donc, quelque soit la technologie utilisée, si le souci majeur est la durée de vie des batteries alors les cellules complexes devraient être utilisées autant que possible. D'autre part, si c'est l'EDP qui doit être minimisé alors les fonctions complexes devraient être décomposées et implantées à l'aide de cellules plus simples.

I.7. COLLISION / ABSORPTION / PROPAGATION

Dans ce paragraphe, les mécanismes d'absorption et de propagation des aléas sont analysés. Le concept de "Seuil de propagation" est introduit [ASG97b]. Les analyses sont faites d'après les résultats obtenus par simulations HSPICE d'un circuit de test représentatif.

Quand les transitions aux entrées d'une porte logique sont séparées par un délai t_{diff} , une transition complète ou incomplète peu se produire à la sortie (figure 1.13). Cette transition peut être soit propagée soit absorbée selon sa largeur, et selon le délai t_{diff} de la porte suivante et sa sortance.

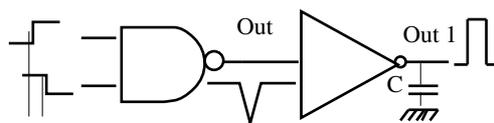


Figure 1.13 : Génération et propagation d'une transition redondante

La figure 1.14 représente les variations d'amplitude des tensions aux points "Out" et "Out 1" de la figure 1.13 en fonction du délai différentiel t_{diff} entre les deux entrées de la porte Nand. Comme il apparaît sur ce graphe, la variation d'amplitude est monotone au point "Out", mais un seuil apparaît au point "Out 1". Dans ce qui suit, nous appelons un "seuil de propagation" (G_{th}) la valeur de t_{diff} pour laquelle l'amplitude maximale du pic $V_{out1} = V_{dd}/2$.

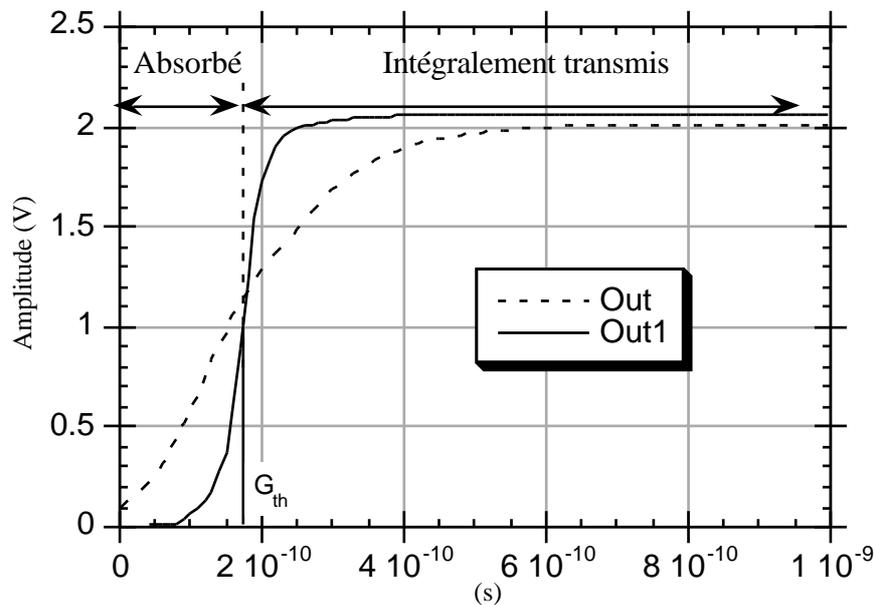


Figure 1.14 : Seuil de propagation

Des simulations HSPICE ont été effectuées afin de mesurer les variations de ce seuil en fonction des autres paramètres du circuit de test. Cette caractérisation est faite en fonction des paramètres suivants :

- : Le délai entre les transitions en entrée
- : Le délai de la porte chargée par la porte Nand
- C_1 : La capacité de charge

Les transistors de type N ont tous la taille minimale, la taille des transistors de type P est choisie de façon à ce que les temps de montée et de descente soient identiques. Les rampes en entrée sont toutes à 100ps. En effet, aucun changement significatif n'a été observé pour des rampes plus lentes.

La figure 1.15 montre la variation de l'amplitude des tensions en "Out" et "Out 1" en fonction de C_1 et pour différentes valeurs de la capacité de charge C_1 .

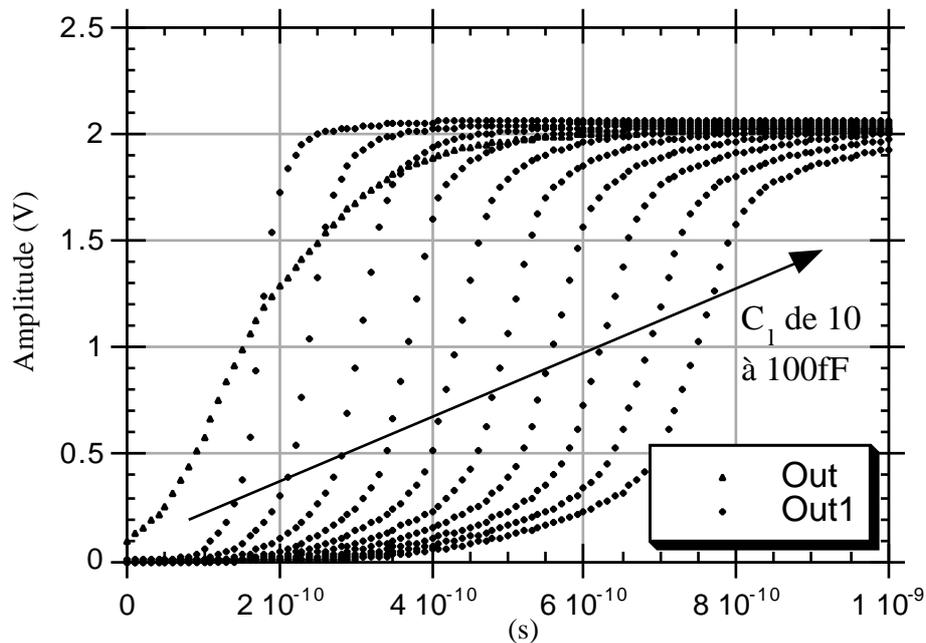


Figure 1.15 : Amplitudes maximales en fonction de C_1 pour diverses valeurs de C_1

Les valeurs de la capacité de charge vont ici de 10 à 100fF. Pour des valeurs plus grandes, l'effet de seuil s'atténue et la courbe $V(\text{Out1})$ devient similaire à $V(\text{Out})$.

Comme il a été dit précédemment, le seuil de propagation (G_{th}) est défini comme étant la valeur de C_1 pour laquelle $V_{\text{Out1}} = V_{\text{dd}}/2$. La figure 1.16 montre que la variation de G_{th} est linéaire avec la capacité de charge C_1 .

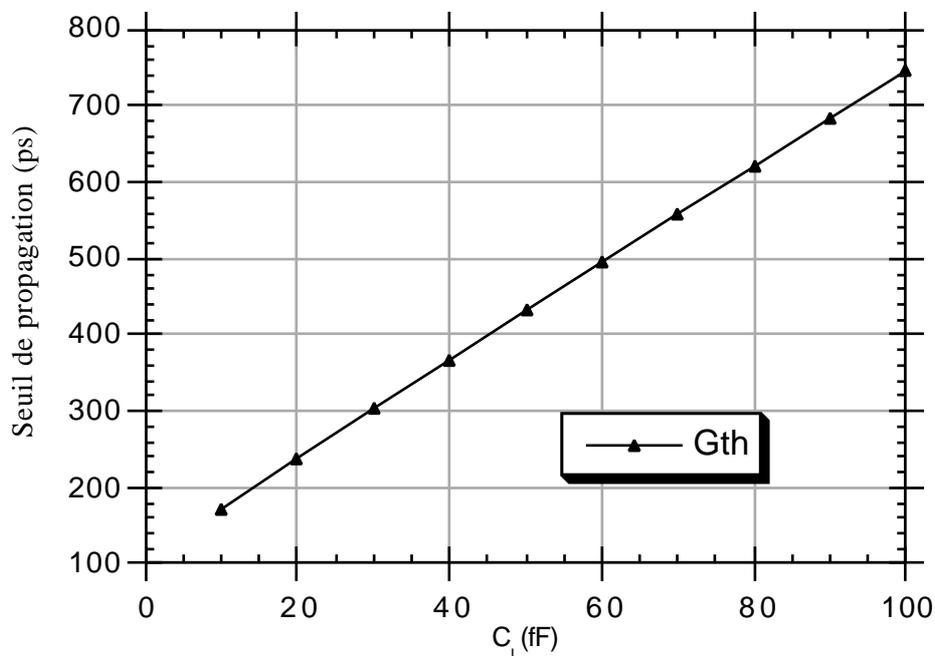


Figure 1.16 : Variation du seuil de propagation en fonction de la capacité de charge

Finalemment, le comportement de G_{th} en fonction de τ , le délai de la porte suivante, (figure 1.13) est aussi linéaire. La pente de cette droite est environ deux; cela se traduit simplement par le fait qu'un aléa généré par un délai différentiel 2τ sera "absorbé" ou "filtré". Cette pente dépend de la technologie et fixe au concepteur une marge pour l'équilibrage des chemins.

I.8. RÉSUMÉ

Dans ce premier chapitre nous avons commencé par une présentation de l'impact que peuvent avoir les choix aux différents niveaux d'abstraction sur la consommation, afin de définir le champ d'action du présent travail. Ensuite nous avons passé en revue les différentes composantes de la dissipation d'énergie :

- dissipation dynamique
- dissipation de court-circuit
- dissipation statique

Étant donné que la dissipation dynamique, due aux charge/décharge des capacités lors des commutations est dominante en CMOS, nous avons montré comment réduire le nombre de ces commutations en rendant l'information redondante. Plusieurs modèles de réduction d'échelle ont ensuite été présentés, et leur influence sur la dissipation discutée. Au paragraphe 5, les différentes métriques de performances utilisées tout au long de cette thèse sont détaillées. Il s'agit du débit, de l'autonomie et de ce que nous avons appelé l'efficacité.

- Le débit est ici proportionnel à la fréquence, car nous ne considérons que des circuits combinatoires synchrones.
- L'autonomie : l'augmentation de l'autonomie d'un circuit consiste à réduire le plus possible l'énergie nécessaire pour effectuer une opération donnée (EPO), souvent au détriment du débit.
- L'efficacité : ce que nous appelons efficacité ici, n'est autre qu'un compromis entre le débit et l'autonomie. C'est le produit énergie - délai ou EDP (exprimé en pJ/MHz).

Le paragraphe 6 traite de la conception utilisant des cellules complexes. Afin de mesurer l'impact des cellules complexes sur les performances, la complexité a d'abord été quantifiée, et des échelles ont été définies. Ces échelles ont permis de comparer les performances de deux versions d'une même technologie (SOI et Si massif) dans l'espace performance/complexité.

Finalement, le dernier paragraphe a abordé le problème des collisions entre transitions, et du filtrage des aléas ainsi générés. La notion de seuil de propagation d'un aléa a aussi été introduite [ASG97b].

I.9. SUMMARY

This first chapter started by a review of the impact the decisions may have on the dissipation depending on the considered level of abstraction. This enabled us to delimit the field of this work. Then, the different components of energy dissipation were listed :

- dynamic dissipation
- short-circuit dissipation
- static dissipation

Given that dynamic dissipation (capacitance charge and discharge) is predominant in static CMOS, it was shown how to reduce signal activity by introducing information redundancy. Scaling schemes were then presented, and their influence on energy dissipation discussed. In the fifth paragraph, the different figures of merit that are used all through this work were detailed. These are : throughput, battery life and what is called here efficiency.

- The throughput is here proportional to the frequency as we consider only combinatorial synchronous circuits.
- Battery life : increasing the battery life for a given design, consists in reducing the energy necessary to achieve a given operation (EPO).
- Efficiency : what is called efficiency here, is nothing but a compromise between throughput and battery life : this is the energy - delay product (EDP).

The sixth paragraph dealt with complex cells design. First, complexity metrics were defined, then the performances of a bulk and a SOI versions of the same technology were compared on these scales.

Finally, the last paragraph addressed the problems of colliding transitions, glitch filtering and propagation. The concept of “glitch threshold” was also introduced [ASG97b].

CHAPITRE SECOND : SOI

II.1. SOI

II.1.1. QU'EST CE?

"Inutile d'interroger le Ciel, il a réponse à tout"

SOI : Silicon - On - Insulator, soit Silicium sur isolant. L'idée de réaliser des dispositifs à base de semiconducteurs mécaniquement isolés entre eux par un substrat isolant est assez ancienne. La première trace remonte à un brevet déposé par Linienfield datant de 1926 [Col97]. On peut donc dire que le premier MOSFET était en SOI. Les dispositifs de Linienfield n'ont pu être réalisés pour des raisons techniques, ensuite l'apparition du bipolaire [Sho76] les a occultés pour un certain temps.

Les circuits intégrés CMOS sont actuellement fabriqués sur Silicium massif de manière quasi-exclusive, et ce pour des raisons simples de disponibilité et de coût. Les procédés SOI pour des applications grand public sont juste émergents (IBM et SAMSUNG).

L'épaisseur d'un substrat de Silicium est d'environ 800 μm , alors que seul le premier micromètre est utilisé. Des interactions indésirables se produisent entre les transistors et le substrat (latchup, capacités parasites etc...). L'importance de ces effets croît à mesure que la taille des dispositifs diminue. L'utilisation d'un substrat SOI permet la réalisation de structures MOS quasi-idéales. L'isolation diélectrique complète élimine ou atténue la plupart des effets parasites rencontrés en Silicium massif.

Il existe une grande variété de méthodes qui permettent d'obtenir un film mince de Silicium monocristallin. Nous en citons les plus répandues :

II.1.1.1. SIMOX

SIMOX ou Séparation par IMplantation d'OXYgène. Le principe consiste à implanter de l'Oxygène en profondeur, puis à former une couche d'oxyde enterrée après recuit (figure 2.1).

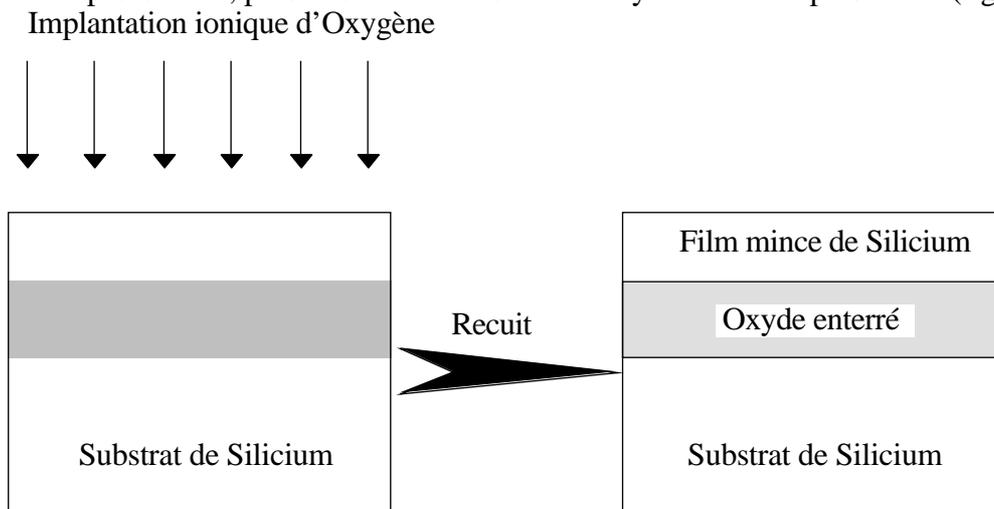


Figure 2.1 : Principe du SIMOX

L'inconvénient principal du SIMOX est le nombre élevé de défauts cristallins induits par l'implantation des ions d'Oxygène qui sont assez lourds.

II.1.1.2. COLLAGE ET POLISSAGE

Un autre procédé de fabrication de plaquettes SOI consiste à coller ensemble deux plaquettes oxydées, puis à en sacrifier une en la polissant de manière à ne laisser qu'un film mince de Silicium comme indiqué sur la figure 2.2

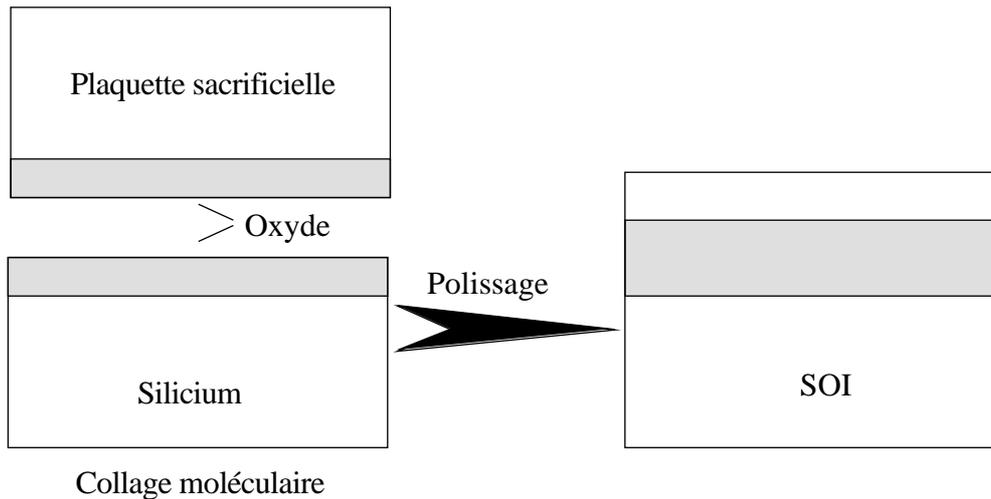


Figure 2.2 : SOI par collage et polissage

L'avantage de cette méthode est qu'elle offre un film de Silicium de très bonne qualité. Les inconvénients sont d'une part financiers : le prix élevé dû en partie au sacrifice d'une plaquette et d'autre part techniques : la difficulté à contrôler l'épaisseur du film de Silicium.

II.1.1.3. SMART CUT®

Le procédé Smart Cut® [Bru95, Bru96] consiste à former des micro-bulles dans une plaquette de Silicium par implantation d'Hydrogène, puis à coller cette plaquette sur une autre. Ensuite, un recuit permet de propager des cassures entre les microbulles, et ainsi de séparer les plaquettes en laissant un film mince de Silicium sur celle qui n'a pas été implantée (figure 2.3).

Les avantages de ce procédé sont multiples :

- les films de Silicium produits sont de bonne qualité, car les ions Hydrogène étant petits, l'implantation induit peu de défauts
- l'épaisseur de la couche de SOI est aisément contrôlable - elle est fonction de l'énergie de l'implantation
- le coût : pour l'obtention de N plaquettes, $N+1$ plaquettes sont nécessaires contrairement à $2N$ pour le procédé précédent.

Il existe un grand nombre d'autres méthodes d'obtention de substrats SOI, comme la recristallisation par laser et l'épitaxie locale par exemple. Pour le détail de ces procédés on pourra consulter [Col97, Cri95, Fur85].

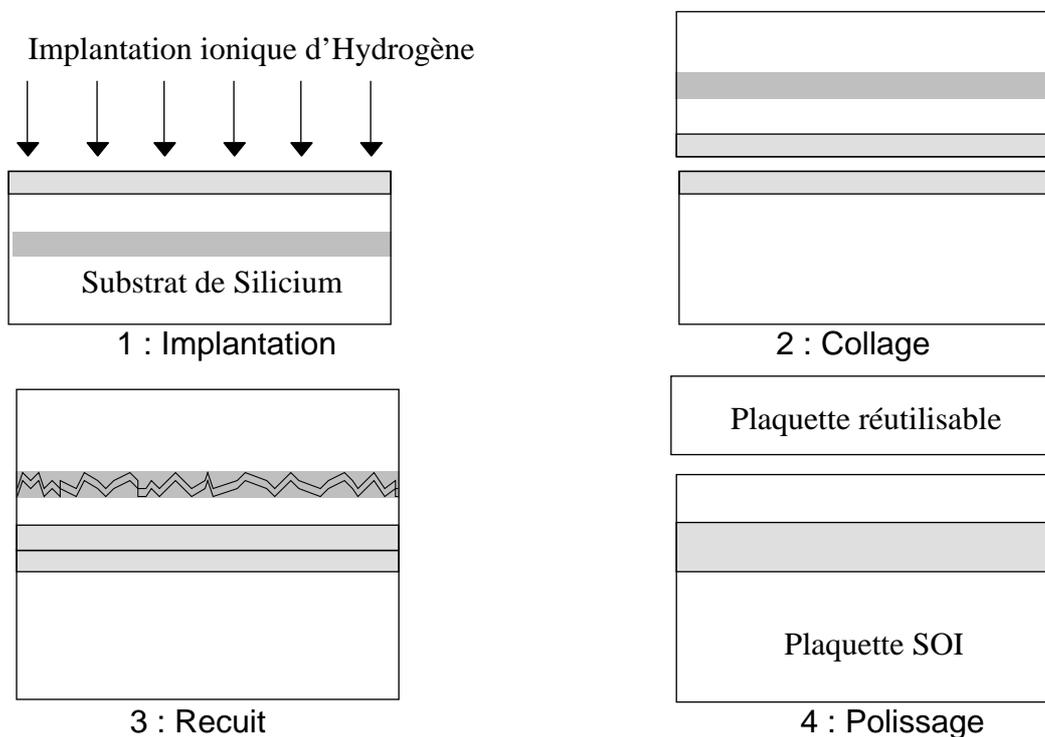


Figure 2.3 : Procédé Smart Cut®

II.1.2. POURQUOI LE SOI

“La réponse est le malheur de la question”

Nous avons vu au paragraphe précédent qu’il est maintenant possible de disposer de plaquettes SOI offrant un film mince de Silicium de grande qualité. Nous allons ici aborder les avantages du SOI sur le Silicium massif, et voir *pourquoi* le SOI est intéressant.

Il n’est ici question que de SOI-film mince. Un film est dit mince si son épaisseur e vérifie

$$e < 2 \cdot \sqrt{\frac{\epsilon_{\text{Si}} \cdot \Phi_F}{qN_a}} \quad (2.1)$$

où Φ_F est le potentiel de Fermi.

Dans l’équation 2.1, nous pouvons aisément reconnaître l’expression de l’extension de la zone de déplétion. Dans le cas d’un film mince, nous parlons aussi de FD MOSFET (Fully Depleted MOSFET) le Silicium est déplété sur toute l’épaisseur.

La présence d’une couche isolante élimine les effets parasites et aussi simplifie le procédé de fabrication : les caissons et les couches enterrées n’ont plus lieu d’être, de même que les LOCOS (LOCAl Oxydation of Silicon). La surface d’un circuit sera donc plus faible en SOI qu’en Silicium massif, entraînant aussi une réduction de la longueur des interconnexions (figure 2.4)

Le nombre d’étapes technologiques est réduit d’environ 30%; une manière de voir consiste à considérer qu’en SOI une partie du coût de fabrication est transférée en amont vers la plaquette.

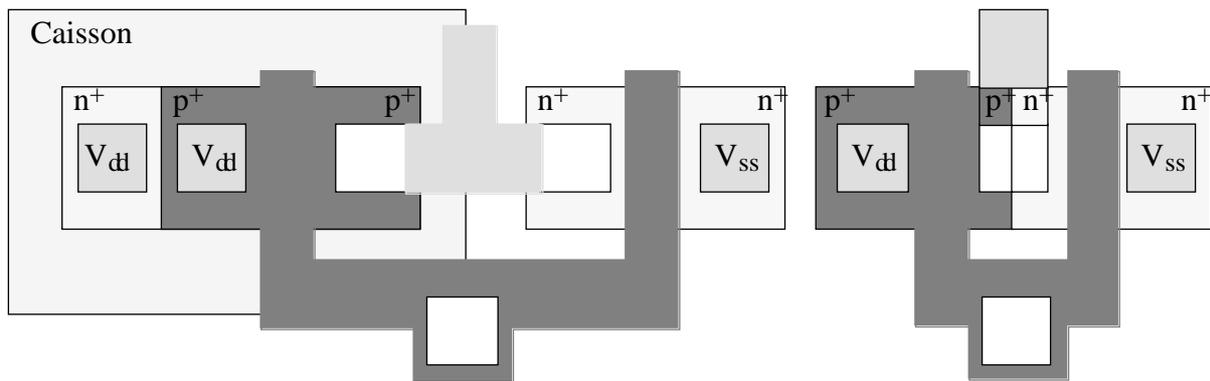


Figure 2.4 : Inverseurs en Silicium massif et en SOI

La version SOI d'un circuit aura un délai τ_{SOI} égal à environ $2/3$ le délai de la version Silicium massif τ_{massif} pour une même tension d'alimentation V_{dd} .

$$\tau_{SOI} \approx (2/3) \cdot \tau_{massif} \quad (2.2)$$

Cette augmentation de la vitesse est la traduction de la diminution des capacités parasites par un facteur équivalent.

$$C_{SOI} \approx (2/3) \cdot C_{massif} \quad (2.3)$$

Ainsi, il est maintenant possible de calculer une approximation du gain en terme de produit énergie-délai (EDP)

$$\frac{EDP_{SOI}}{EDP_{massif}} = \frac{\frac{1}{2} \cdot C_{SOI} \cdot V^2 \cdot \tau_{SOI}^2}{\frac{1}{2} \cdot C_{massif} \cdot V^2 \cdot \tau_{massif}^2} = \frac{\frac{1}{2} \cdot \frac{2}{3} \cdot C_{massif} \cdot V^2 \cdot (\frac{2}{3})^2 \cdot \tau_{massif}^2}{\frac{1}{2} \cdot C_{massif} \cdot V^2 \cdot \tau_{massif}^2} = \frac{8}{27} < 30\% \quad (2.4)$$

Le gain théorique est un facteur 3. Dans la pratique il est plutôt de l'ordre de 5, car le faible courant de fuite en SOI permet des tensions de seuil plus faibles, et donc une tension d'alimentation plus faible à délai constant.

Ces approximations sont données pour des circuits migrés de Silicium massif en SOI tels quels, et reflètent uniquement le gain apporté par la technologie. Le gain global peut-être bien plus important grâce à des méthodes de conception plus spécifiques qui permettent de mieux tirer partie des propriétés du SOI, comme par exemple l'utilisation des cellules complexes décrite au paragraphe I.6.

Le SOI apparaît donc comme un excellent candidat pour les applications faible tension - faible consommation.

II.1.3. POLARISATION CANAL

La présence d'un substrat flottant sous le canal des transistors donne lieu à un certain nombre de problèmes en mode transitoire, hystérèse etc... Tous ces effets trouvent leur origine dans la charge/décharge du substrat flottant par des courants venant de la source ou du drain ou dans le couplage capacitif entre la grille et le substrat. Voici une liste des principaux effets [Col97] :

- courant de fuite transitoire dans les transistors de passage qui peut affecter la circuiterie des SRAM et DRAM [PFSK95]

- l'apparition d'un pic de courant quand un créneau est appliqué à la grille (couplage capacitif entre la grille et le substrat flottant).
- variations du courant de drain en fonction de l'histoire du transistor
- propagation irrégulière des signaux (délai dépendant de la fréquence...)
- dégradation des états logiques en logique dynamique

Les effets cités ci-dessus sont surtout importants pour les dispositifs partiellement déplétés. Ils peuvent être éliminés en polarisant le canal [IHOW97]. Cette polarisation n'est d'ailleurs pas nécessaire pour les dispositifs entièrement déplétés - du moins pour ce qui concerne la conception de circuits numériques en logique CMOS statique.

II.1.4. DTMOS

En SOI il est possible de réaliser des transistors hybrides bipolaire-MOS plus communément appelés DTMOS pour Dynamic Threshold MOS (ou MOS à seuil dynamique). Cela consiste à contrôler l'effet bipolaire parasite en ajoutant un contact entre la grille et le canal. Le substrat qui était flottant devient la base du transistor bipolaire, alors que la source et le drain deviennent respectivement l'émetteur et le collecteur. Le problème majeur auquel se heurte la réalisation de ces dispositifs est la difficulté de réalisation d'un contact sur le canal du transistor.

II.1.5. GRILLE EN T

"Pas besoin de grille en fer, c'est les Autres", J. P. Sartre

Les transistors utilisés dans le cadre de ce travail sont réalisés à l'aide du procédé EDMOS (Edge-Defined MOS) développé à l'IMS (Institut pour la Microélectronique de Stuttgart - Allemagne) dans le cadre du projet européen Esprit IV HIPERLOGIC [HPL97, HPL98, http]. Ce procédé permet l'obtention de transistors dont la longueur de canal est indépendante de la résolution de la lithographie. L'idée principale de ce procédé nanométrique est le remplacement de la lithographie par le dépôt de couches minces. Ainsi, certaines dimensions habituellement définies par lithographie le sont maintenant par épaisseur de couches [DABD96], permettant l'obtention de canaux plus courts. La figure 2.5 présente une vue en coupe de la structure finale.

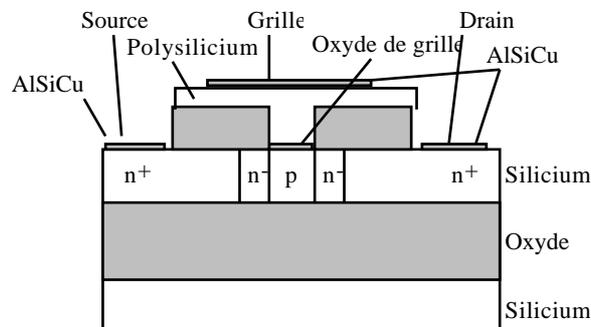


Figure 2.5 : Transistor avec grille en T.

Les étapes technologiques sont décrites dans les figures 2.6a à 2.6f.

Une vue en coupe de la structure finale est montrée sur la figure 2.5. Ces transistors peuvent être utilisés quasiment toujours avec les dimensions minima car ils présentent intrinsèquement un rapport W/L élevé.

Figure 2.6a : Sur la couche de Silicium sur isolant, une "marche" d'oxyde est formée. Aucune dimension critique du point de vue de la lithogravure n'intervient ici.



Figure 2.6b : Une couche de nitrure de Silicium est déposée sur la marche d'oxyde.

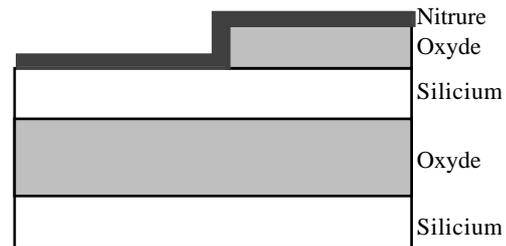


Figure 2.6c : La couche de nitrure est gravée isotropiquement.

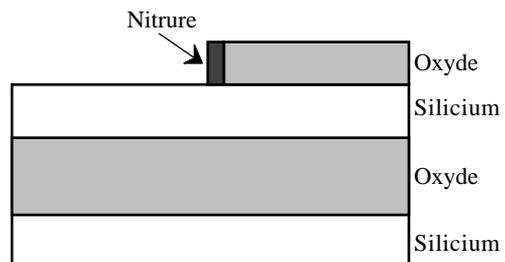


Figure 2.6d : Ensuite intervient une gravure sélective qui permet d'enlever l'oxyde et de ne laisser que le mince espaceur en nitrure. C'est lui qui déterminera la longueur du canal. A ce stade interviennent les dopages LDD qui sont auto-alignés sur cet espaceur.

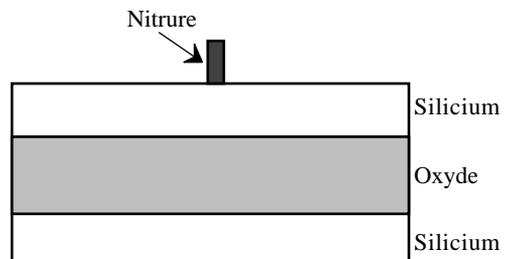


Figure 2.6e : oxydation puis planarisation

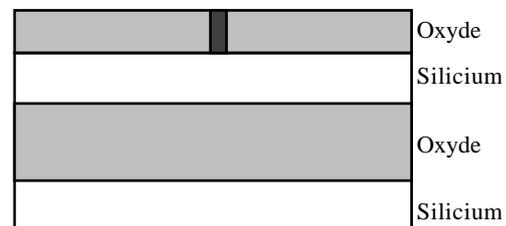
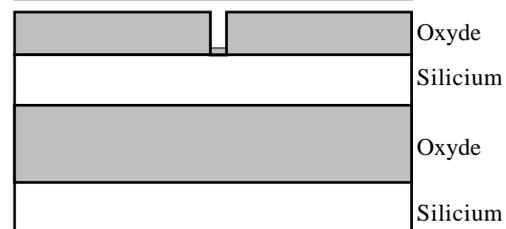


Figure 2.6f : Gravure sélective; mais cette fois c'est le nitrure qui est enlevé, laissant une mince tranchée. S'ensuit alors la formation de l'oxyde de grille et le comblement de la tranchée.



II.2. SOI SUR SOI

II.2.1. MOTIVATIONS

Dans ce paragraphe nous présentons une technologie tridimensionnelle CMOS-SOI utilisant les grilles en **T** décrites précédemment. Cette technologie consiste en deux couches de transistors - Silicium sur isolant sur Silicium sur isolant [ASDA97]. Elle est développée à l'IMS (Institut de Microélectronique de Stuttgart - Allemagne) dans le cadre du même projet européen [HPL97, HPL98, http].

Avec l'intégration tridimensionnelle la densité est augmentée et par conséquent les longueurs des interconnexions sont réduites. Sachant que les capacités dues aux interconnexions sont prédominantes en submicronique profond [Bor97, Dax97, SIA97] - ce qui est d'autant plus vrai en SOI car les capacités intrinsèques des dispositifs sont plus faibles que dans leurs équivalents en Silicium massif - nous pouvons nous attendre à ce que l'intégration tridimensionnelle apporte des gains substantiels en énergie par opération. Mais il convient d'abord d'adapter à cette technologie les méthodologies de conception ainsi que les outils.

II.2.2. STRUCTURE TRIDIMENSIONNELLE

II.2.2.1. COUCHES DISPONIBLES

La structure est formée par deux couches de transistors avec une couche de polysilicium au milieu et une autre en dessus. Deux niveaux de métallisation sont disponibles pour les routages local et global. Les deux couches de Silicium sont connectées entre elles par une "prise" en Silicium car la couche supérieure est obtenue par épitaxie latérale à partir de la couche inférieure qui sert à donner la maille cristalline.

La première couche de Silicium est aussi obtenue par épitaxie locale. La prise pour l'épitaxie peut être utilisée comme connexion vers la masse économisant le routage d'un rail d'alimentation. La distribution de l'une des tensions d'alimentation par le substrat a été anciennement exploitée par RCA (technologie COSMOS) et par le CNET (technologie MDMOS).

Il y a deux couches de polysilicium, une pour chaque niveau de transistors. Le contact poly-poly est une étape technologique critique indispensable pour la réalisation de circuits CMOS. Les méthodologies de conception sont adaptées de manière à tirer profit au mieux des diverses couches d'interconnexions disponibles (figure 2.7).

En comparaison avec la technologie planaire, les règles de dessins supplémentaires sont dues à la présence d'un niveau de Silicium intermédiaire : en effet, il faut veiller à ce que les divers contacts se fassent en dehors de l'îlot de Silicium de type P.

II.2.2.2. RÉALISATION

Ce paragraphe décrit les principales étapes technologiques pour la réalisation de la structure tridimensionnelle de la figure 2.7.

Le but de la première étape est l'obtention de la couche inférieure de Silicium sur isolant. Ce qui peut être fait soit en utilisant un substrat SOI comme ceux décrits aux paragraphe II.1.1, soit

par épitaxie locale (figure 2.8). Cette deuxième méthode est préférée car elle permet de rendre le processus de fabrication symétrique pour les deux types de transistors N et P. Les niveaux de gris (originellement les couleurs) représentent le dopage.

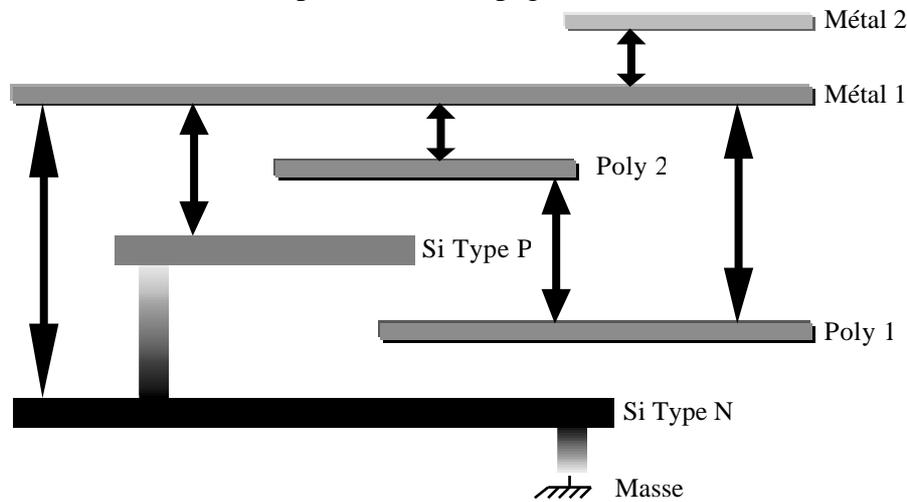


Figure 2.7 : Coupe schématique de la structure tridimensionnelle; les flèches représentent les contacts possibles

Les transistors de type N sont ensuite fabriqués dans cette couche. Le processus de fabrication des transistors avec la grille en **T** est décrit au paragraphe II.1.6. Le résultat de cette étape est illustré par la figure 2.9.

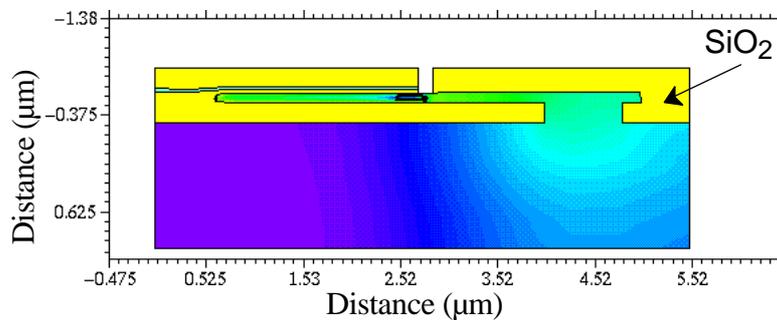


Figure 2.8 : Première couche de SOI

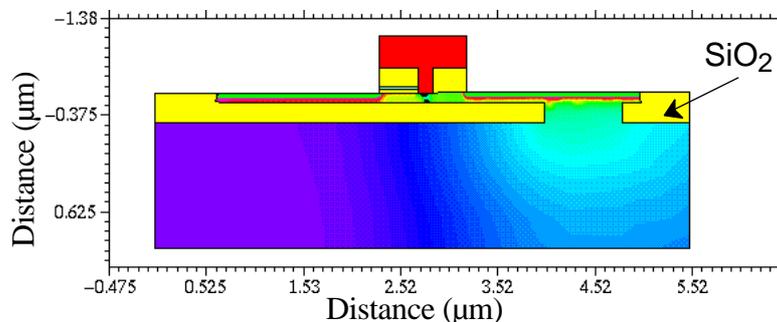


Figure 2.9 : EDMOS type N

Les étapes qui suivent (figures 2.10 à 2.12) sont une répétition du processus de fabrication des transistors de type N pour les transistors de type P : dégagement d'une alvéole remplie par épitaxie, planarisation et grille en **T**.

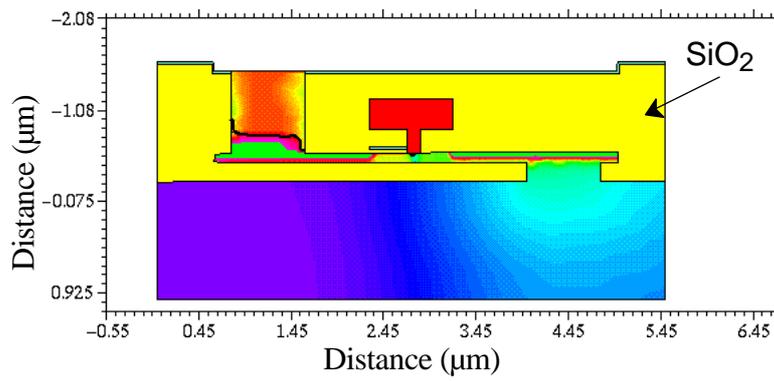


Figure 2.10 : Ouverture et épitaxie avec inversion du dopage

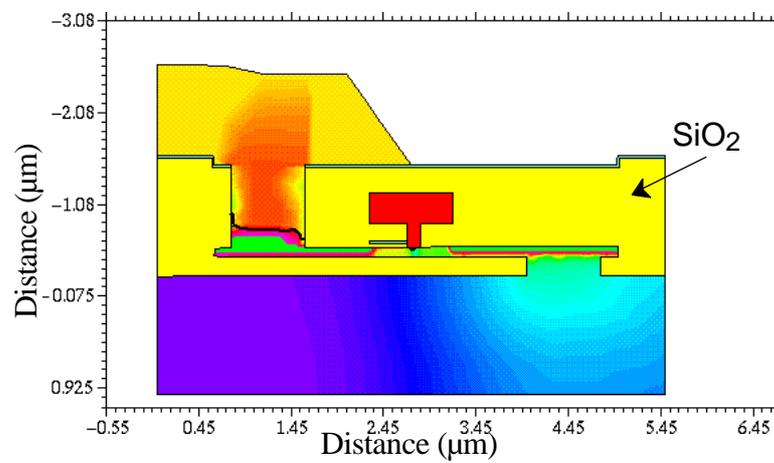


Figure 2.11 : Épitaxie latérale de la couche de Silicium supérieure

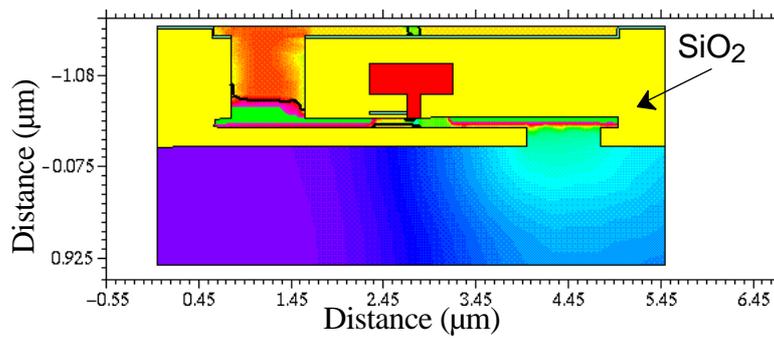


Figure 2.12 : Planarisation du Silicium par polissage chimico-mécanique (CMP)

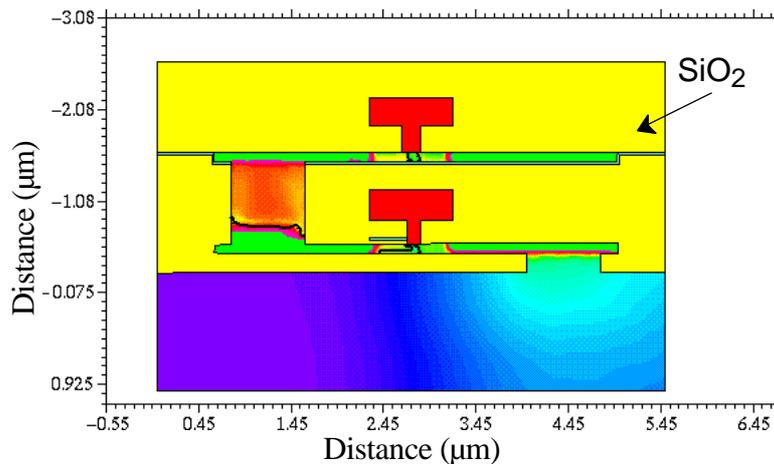


Figure 2.13 : Structure finale avant métallisation

Deux couches de métallisation viennent ensuite s'ajouter à cette structure, le substrat de départ servant de connexion vers la masse.

II.2.3. CELLULES AUTODUALES

"Je les connais par les cinq sens et quelques autres", G. Apollinaire

Ce type d'intégration tridimensionnelle introduit des contraintes de conception supplémentaires qui empêchent d'atteindre un facteur deux en terme d'augmentation de la densité. La limitation fondamentale tient simplement au fait que la surface minimale occupée est celle du réseau de transistors de type P qui sont plus larges que les transistors de type N. Cette limite peut être atteinte dans certains cas, où les topologies des réseaux N et P sont identiques. Dans ce cas les cellules sont dites autoduales.

Le dual F^D d'une fonction booléenne F est obtenu en interchangeant les opérateurs "et" et "ou" entre eux. L'autodualité est atteinte quand $F = F^D$. Prenons l'exemple de la fonction "majorité de 3 variables" $Maj(a, b, c)$

$$\begin{aligned}
 Maj^D(a, b, c) &= (ab + ac + bc)^D \\
 &= (a + b)(a + c)(b + c) \\
 &= \dots = (ab + ac + bc) \\
 &= Maj(a, b, c)
 \end{aligned}$$

Une fois la propriété d'autodualité établie, les réseaux N et P peuvent alors être rendus identiques [PeGl91]. Deux topologies sont possibles (figures 2.14a et 2.14b).

La topologie choisie est bien sûr la plus rapide, c'est à dire celle qui comporte le moins de transistors en série (figure 2.14a). Les grilles des transistors N et P sont connectées à l'aide du contact poly-poly comme sur la figure 2.7. Le gain en densité atteint dans ce cas 40%. La fonction majorité a été intégrée dans les bibliothèques de cellules 2D et 3D qui seront présentées au paragraphe II.3.2.

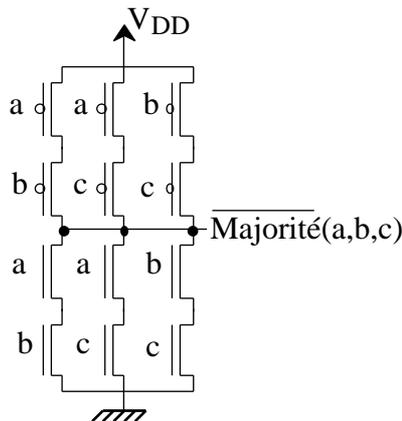


Figure 2.14a : Majorité autoduale

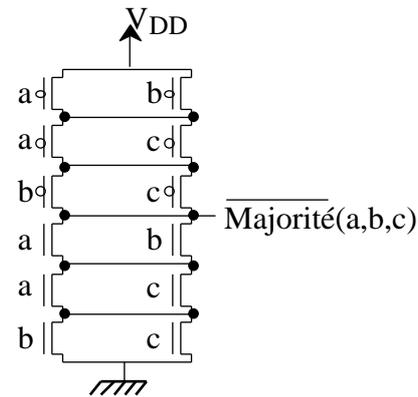


Figure 2.14b : Majorité autoduale alternative

II.3. OUTILS DE CONCEPTION

Ce paragraphe présente les outils que nous avons développés pour la conception de circuits utilisant les technologies décrites, à savoir SOI planeaire et SOI tridimensionnelle.

II.3.1. DESIGN KITS

Nous avons développé des design kits pour ces technologies dans l'environnement de conception CADENCE DFII version 97a. Un design kit est composé de deux parties distinctes : l'une pour la conception "full custom" et l'autre pour les bibliothèques de cellules standard. Chacune de ces parties comporte des fonctionnalités "front end" et "back end".

- **Full custom front end** inclut : symboles (MOS, résistance, capacité), interface SPICE et fichiers de configuration.
- **Cellules standard front end** inclut : symboles des cellules, modèles VERILOG, (délai, capacité, rétro annotation et sortance) utilisés pour la synthèse logique et la simulation post-synthèse.
- **Full custom back end** inclut : couches, numéros GDSII, DRC, ERC, extraction, LVS et cellules paramétrées (P-cells).
- **Cellules standard back end** inclut : représentation abstraite pour le placement routage automatique et rétro annotation VERILOG.

Les outils développés ne prétendent pas être complets, mais simplement permettent de conduire des études comparatives afin de mesurer les conséquences de l'intégration tridimensionnelle sur les performances des circuits. Ils comportent pour chacune des deux technologies : DRC, ERC, LVS, placement/routage automatique des cellules standard et modèles VERILOG. Toutes les simulations électriques sont effectuées avec HSPICE.

II.3.2. BIBLIOTHÈQUES DE CELLULES

Le développement des bibliothèques de cellules a été conduit en parallèle pour les deux technologies SOI et SOI sur SOI. Le flot de conception est illustré sur la figure 2.15. Les bibliothèques contiennent uniquement quelques cellules de base, car l'objectif est limité à la

conception d'opérateurs arithmétiques combinatoires. Le gain en surface est d'environ 30%. Il est plutôt faible, car dans l'approche cellules standard, la largeur des cellules est déterminée par le nombre de plots d'entrées/sorties qui est le même pour une cellule donnée quelque soit la technologie.

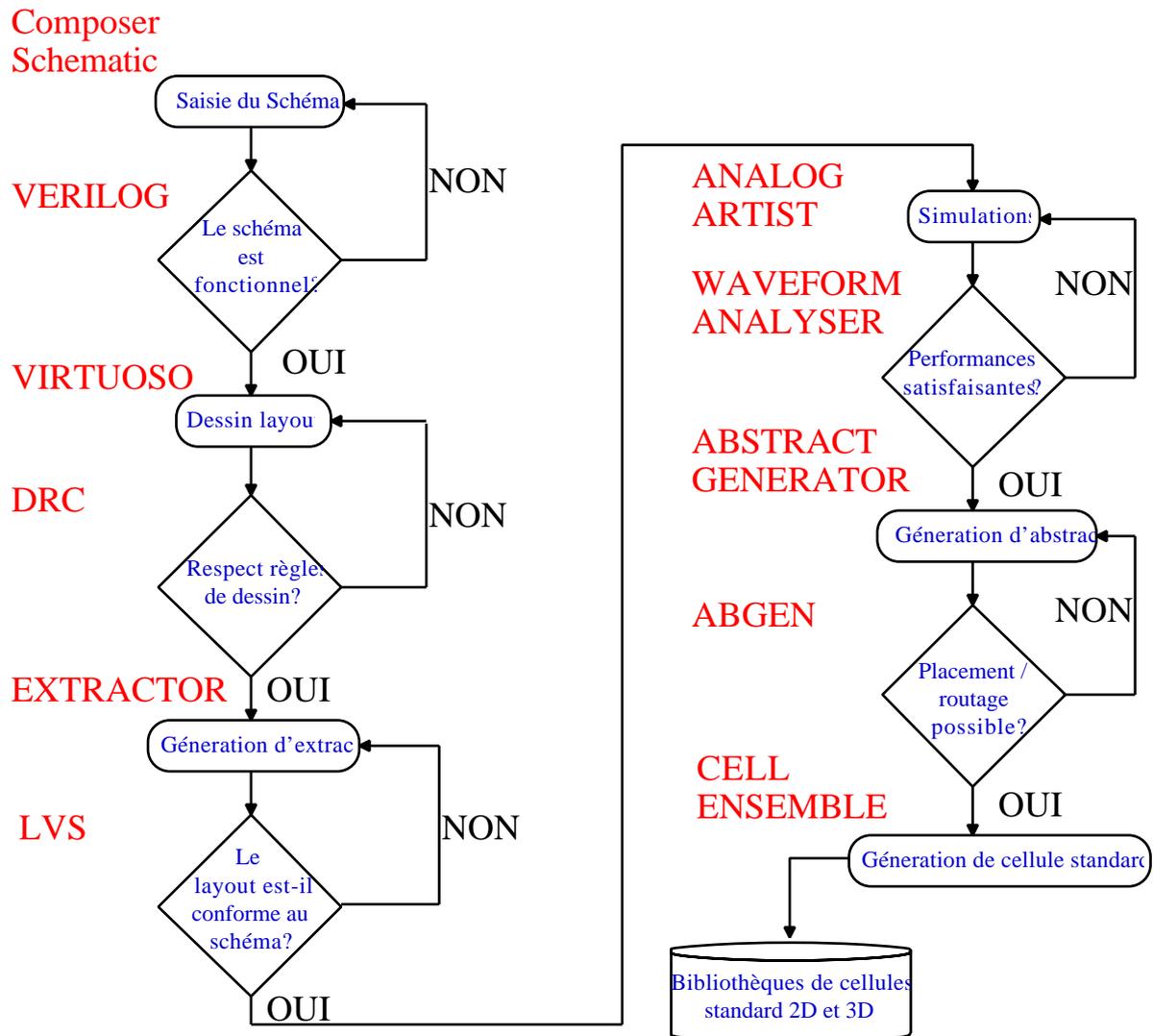
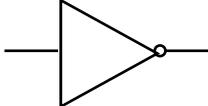
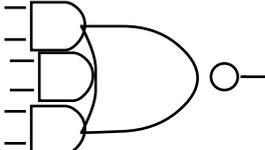


Figure 2.15 : Flux de conception d'une bibliothèque de cellules

Les cellules de base intégrées dans les bibliothèques sont données par le tableau 2.1

Tableau 2.1 : Cellules de base

Inverseur	
Non-et à deux entrées	
Et à deux entrées	
Non-ou à deux entrées	
Ou à deux entrées	
Et-Ou Inversé (AOI)	
Ou exclusif à deux entrées	
Majorité inversée de trois variables	

Les caractéristiques de chacune de ces cellules dans les deux technologies sont données en fin de volume dans l'annexe.

II.3.2.1. PERFORMANCES D'UNE CELLULE

La cellule $\overline{\text{Maj}}$ qui a été intégrée dans les bibliothèques de cellules est fréquemment utilisée dans la réduction des produits partiels de la multiplication. Les dessins des masques en 2D et en 3D sont donnés respectivement figures 2.16a et 2.16b. On remarquera l'absence du rail GND pour la version tridimensionnelle, le substrat servant de masse.

Le tableau 2.2 montre des résultats comparatifs pour la cellule $\overline{\text{Maj}}$. Il est intéressant de noter que le gain en surface est plus important pour cette cellule autoduale que pour les autres cellules.

Le paramètre "capacité totale" est obtenu en additionnant toutes les capacités parasites données par l'extraction. Celles-ci incluent les capacités intrinsèques et les capacités des interconnexions. Étant donné que les capacités intrinsèques sont les mêmes en SOI et en SOI sur SOI, les gains obtenus sont à mettre uniquement sur le compte de la réduction des longueurs des interconnexions.

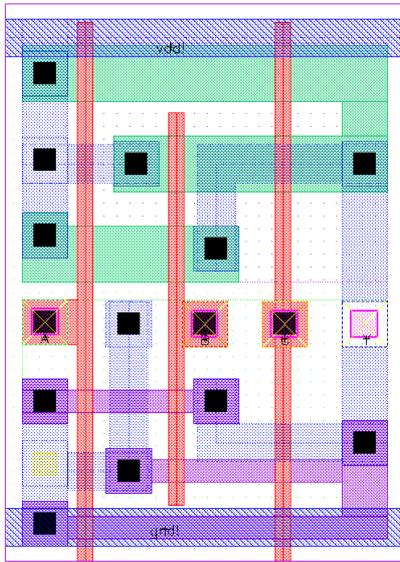


Figure 2.16a : Maj en SOI

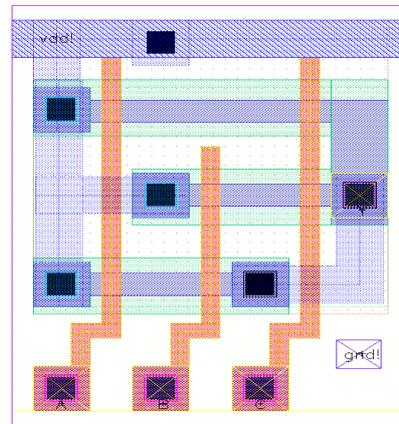


Figure 2.16b : Maj en SOI sur SOI

Tableau 2.2 : Résultats pour la cellule Maj

	SOI	SOI sur SOI	Gain
Surface (μm^2)	588	352.8	40 %
Capacité totale (fF)	229.4	194.8	15 %
Délai (ps)	385	276	28.3 %

Dans ce paragraphe nous avons présenté les outils de conception développés dans le cadre du projet européen ESPRIT IV HIPERLOGIC. Un flot de conception complet est disponible pour les deux technologies; il permet à partir d'une description haut-niveau de générer le dessin des masques du circuit. Il faut aussi ajouter que ces bibliothèques s'interfacent aussi avec d'autres outils de synthèse [StPr96] et de caractérisation [BDB98] développés à l'IPVR (Institut für Parallele und Verteilte Höchstleistungsrechner) à Stuttgart en Allemagne.

II.4. RÉSUMÉ

Dans ce chapitre nous avons détaillé les technologies utilisées ainsi que les outils de conception et les bibliothèques de cellules que nous avons développés dans le cadre de ce travail.

Dans la première partie nous avons passé en revue diverses méthodes d'obtention de substrats de Silicium sur isolant et avons présenté le procédé de fabrication des transistors EDMOS. Dans ce procédé la longueur du canal est de 100nm. Elle est définie indépendamment de la résolution de la lithographie par dépôt et gravure de couches minces. Une vue en coupe est présentée sur la figure 2.5.

La seconde partie a été consacrée à une technologie spécifique où les transistors de type P et de type N sont empilés en deux couches de Silicium sur isolant. Nous appelons cette technologie SOI sur SOI; son avantage est surtout la faible longueur des interconnexions, mais de nouvelles contraintes de conception sont introduites; de nouvelles méthodologies de conception ont donc été développées.

Des outils de conception et des bibliothèques de cellules spécifiques ont aussi été développés; ils ont fait l'objet de la troisième partie.

II.5. SUMMARY

This chapter detailed the technologies that were used as well as the design tools and cell libraries developed in the context of this work.

In the first part, an overview of SOI substrates fabrication methods was given, and the EDMOS process was presented. In this process the channel length is 100nm. It is lithography independent, and is obtained by suitable thin films deposition and etching techniques. A cross sectional view was presented in figure 2.5.

The second part was devoted to a specific technology where the N type and P type transistors are stacked in two layers of SOI. The main advantage of this technology is the reduced length of the interconnections. But new design constraints were introduced, and thus new design methodologies and design tools had to be developed.

The specific tools and cell libraries that we developed were detailed in the third part.

CHAPITRE TROISIÈME : ARITHMÉTIQUE

III.1. INTRODUCTION

Ce chapitre est consacré aux opérateurs arithmétiques combinatoires, et plus précisément aux opérateurs arithmétiques à faible consommation d'énergie par opération (EPO). Il est toujours possible de réduire la consommation d'une partie opérative donnée en échange d'une dégradation de ses performances comme cela est illustré dans la figure 3.0. Cependant, ne voulant pas sacrifier le débit, nous nous attachons dans ce qui suit à explorer les architectures d'opérateurs arithmétiques combinatoires les plus performantes, et à réduire leur consommation à délai constant.

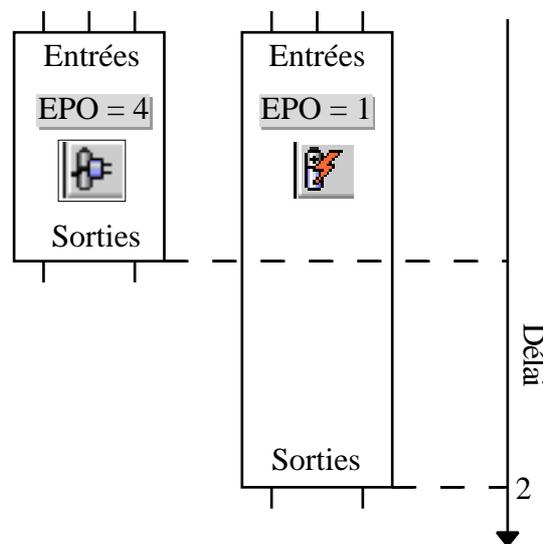


Figure 3.0 : Diagramme d'échange délai - énergie

Ce chapitre est divisé en trois parties consacrées chacune à un opérateur arithmétique : addition, multiplication et division.

III.2. ADDITION

Dans cette partie l'objectif n'est pas d'améliorer les additionneurs les plus rapides, mais plutôt de donner un compromis entre complexité, vitesse et consommation : d'abord, l'opérateur ¹ introduit par Brent et Kung [BrKu82] est rappelé. Il est ensuite utilisé pour concevoir des additionneurs "standard". Plusieurs architectures bien connues présentant différents compromis vitesse/délai sont décrites dans un formalisme unifié. En mélangeant différents styles, il est possible d'obtenir la complexité minimale pour un délai total donné. Finalement, un modèle d'activité est développé pour l'additionneur à propagation et étendu à d'autres architectures. Le problème de l'adaptation des additionneurs à la multiplication et la division sera traité dans les parties correspondantes à ces opérateurs.

¹ Cet opérateur est souvent appelé α ou β , mais le symbole α est typographiquement plus adapté pour une cellule qui effectue la réduction de quatre bits en deux.

III.2.1. SYNTHÈSE D'ADDITIONNEURS

Dans une addition combinatoire de deux nombres $A = \sum_{i=0}^{n-1} a_i * 2^i$ et $B = \sum_{i=0}^{n-1} b_i * 2^i$, à chaque position i , la retenue suivante c_{i+1} est soit générée, c'est à dire $c_{i+1} = 1$ soit tuée c'est à dire $c_{i+1} = 0$ soit propagée c'est à dire $c_{i+1} = c_i$ en fonction des valeurs de a_i et b_i . La propagation est aussi appelée transmission. Trois signaux peuvent donc être définis, un pour chaque cas : $g_i = a_i \cdot b_i$, $p_i = a_i \oplus b_i$ et $k_i = \overline{a_i} \cdot \overline{b_i}$, mais un minimum de deux est nécessaire. Nous retiendrons le couple (p_i, g_i) pour des raisons de simplicité des équations logiques.

L'addition peut être décomposée en trois étapes. D'abord, les signaux (g_i, p_i) sont calculés $\forall i \in \{0 .. n-1\}$. A partir de ces signaux nous obtenons les retenues c_i . Finalement l'ensemble des bits de sortie $S = \sum_{i=0}^{n-1} s_i * 2^i$ est obtenu par $s_i = p_i \oplus c_i$. Nous supposons que la retenue entrante c_0 est nulle et que nous voulons une retenue sortante. Les autres cas pouvant être obtenus simplement par décalage d'indices.

III.2.1.1. PROPAGATION ET GÉNÉRATION DE GROUPE

Notons P_i^j la propagation de groupe et G_i^j la génération de groupe, avec $n-1 \geq i \geq j \geq 0$.

P_i^j signifie que la retenue se propage de la position j jusqu'à la position i , c'est à dire c_{i+1} est égal à c_j . $P_i^j = \bigwedge_{n=i}^j p_n$.

G_i^j signifie qu'une retenue est générée quelque part entre j et i et propagée jusqu'à la position i . D'où $c_{i+1} = 1$. $G_i^j = g_i \vee \bigwedge_{n=i}^j (P_i^{n-1} \cdot g_n)$. Notons que l'expression des P_i^j est plus simple que celle des G_i^j .

Clairement, nous avons $P_i^i = p_i = a_i \oplus b_i$, $G_i^i = g_i = a_i \cdot b_i$, $P_i^j \cdot G_i^j = 0$ et $c_{i+1} = G_i^0$ (en supposant que $c_0 = 0$).

Notons aussi PG_i^j le couple (P_i^j, G_i^j) . Pour tout k tel que $n-1 \geq i \geq k \geq j \geq 0$, PG_i^j peut être calculé à partir de PG_i^k et de PG_{k-1}^j comme suit :

$$PG_i^j = (P_i^j, G_i^j) = (P_i^k \cdot P_{k-1}^j, G_i^k \cdot P_i^k \vee G_{k-1}^j)$$

Nous notons l'opérateur tel que $PG_i^j = PG_i^k \cdot PG_{k-1}^j$. Dans les figures, l'opérateur sera représenté par la cellule $\begin{array}{|c|} \hline \oplus \\ \hline \end{array}$.

Il est aisé de montrer que :

- ① est associatif : $PG_i^j = (PG_i^k \cdot PG_{k-1}^j) \cdot PG_{j-1}^k = PG_i^k \cdot (PG_{k-1}^j \cdot PG_{j-1}^k)$
- ② est non commutatif $PG_i^j \neq PG_{k-1}^j \cdot PG_i^k$
- ③ est idempotent $PG_i^j = PG_i^k \cdot PG_i^j$ avec $i-1 \geq k-1 \geq j$
- ④ est non décroissant.

Tout couple PG_i^j requiert $(i-j-1)$ cellules pour être calculé à partir des entrées. Les résultats intermédiaires des cellules peuvent être réutilisés, réduisant ainsi le nombre total de cellules tout en augmentant la sortance de certaines.

Une addition est exécutée en trois étapes (figure 3.1) :

- D'abord le calcul des (p_i, g_i) à partir des entrées $a_i \cdot b_i$,

- s'ensuit le calcul des G_0^i
- puis finalement, $s_i = p_i \quad c_i = p_i \quad G_0^{i-1}$.

Les première et troisième étapes sont exécutées en temps constant (indépendant de n). Elles sont implicites dans les figures. Étant donné l'associativité et la non commutativité de l'opérateur \oplus , toute structure d'arbre de cellules convient au calcul des G_0^i [GBB94, Zim96] du moment que :

- G_0^i est connecté à tous les (p_j, g_j) , j allant de 0 à i
- il n'y a pas d'inversion dans l'ordre des entrées

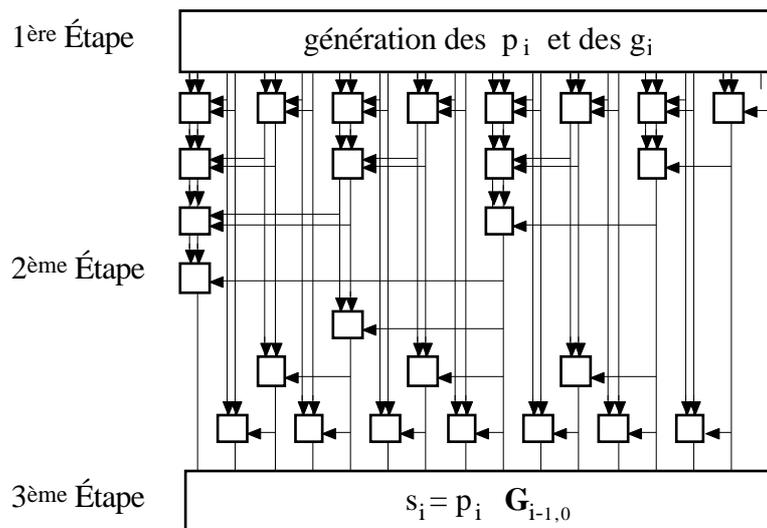


Figure 3.1 : Architecture générique

Il est important de noter que la sortie P_i^j d'une cellule qui n'est pas connectée à l'entrée de droite d'une autre cellule est inutile. La dernière cellule en bas de chaque colonne peut donc être simplifiée.

III.2.2. MODÈLE COMPLEXITÉ / DÉLAI

Examinons maintenant certaines architectures bien connues [GBB94], leur délai (nombre de cellules sur le chemin critique), et leur coût (nombre total de cellules). Dans cette partie nous considérons que "l'enveloppe temporelle" désirée de l'additionneur est rectangulaire, autrement dit que les entrées sont toutes disponibles au même moment, et que les sorties sont toutes désirées au même moment. Il n'en est pas toujours ainsi. Certains cas d'arrivées dans le désordre seront traités plus loin dans le cadre de la multiplication et de la division.

III.2.2.1. ADDITIONNEUR À PROPAGATION

Le délai de l'additionneur à propagation de retenue (RCA) et son coût sont en $O(n-1)$. Il est inefficace et obtenu par un simple aboutement de cellules.

III.2.2.2. CARRY SELECT À DEUX ÉTAGES

L'additionneur carry-select à 2 étages ou 2-CSA (figure 3.2) est basé sur le précédent, tronqué en blocs de tailles variables. Son coût est en $O(2n)$ et son délai en $O(\lceil \sqrt{2n} \rceil)$; plus précisé-

ment, avec exactement k cellules sur le chemin critique, un additionneur peut accommoder jusqu'à $1 + \sum_{i=1}^k i = \frac{1}{2} k(k+1)$ bits.

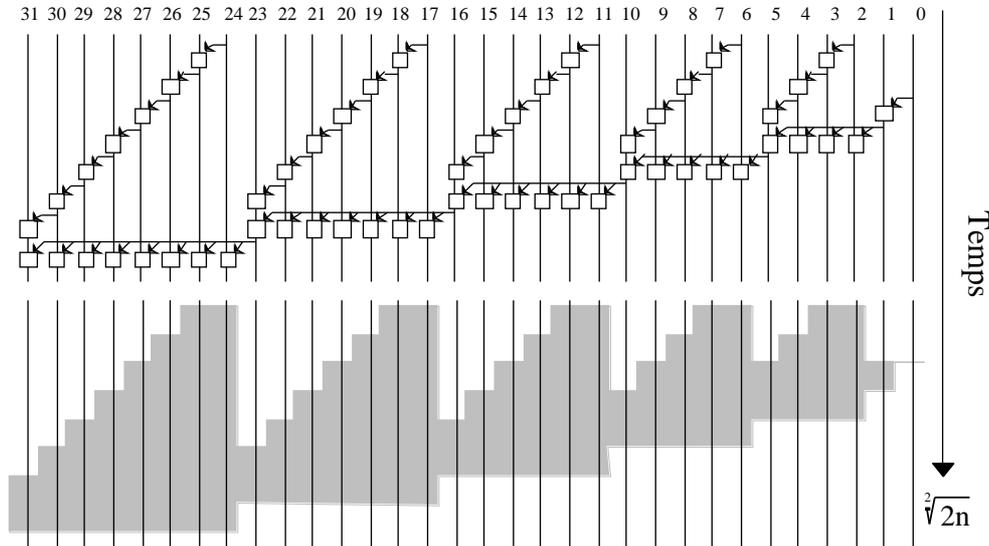


Figure 3.2 : Un 2-CSA

La figure 3.2 montre aussi l'enveloppe temporelle du 2-CSA. Le délai (en nombre de cellules) est égal à la hauteur du rectangle circonscrit.

III.2.2.3. ADDITIONNEUR DE BRENT & KUNG

L'additionneur de Brent et Kung [BrKu82] est basé sur des arbres binaires de cellules. Son coût est en $O(2n)$, et son délai en $O(\lceil \log_2(n) \rceil)$. Un arbre binaire produit tous les G_i^0 pour chaque i de la forme 2^j-1 , puis un autre arbre donne les G_i^0 restants.

III.2.2.4. ADDITIONNEUR DE SKLANSKY

Il est montré que l'additionneur de Sklansky [Skl60] est l'architecture la plus rapide. Son coût est en $O\left(\frac{n \log_2(n)}{2}\right)$, et son délai en $O(\lceil \log_2(n) \rceil)$. L'inconvénient majeur est que la sortance sur le chemin critique augmente exponentiellement avec le nombre de bits, par conséquent les tailles des transistors doivent être ajustées.

III.2.2.5. KOGGE & STONE ET HAN & CARLSON

Le bit le plus significatif de l'additionneur de Brent et Kung ainsi que de celui de Sklansky est obtenu par un arbre binaire équilibré en temps $\log_2(n)$. En répliquant cet arbre pour toutes les autres positions, nous obtenons l'additionneur de Kogge et Stone [KoSt73] (figure 3.3). La sortance est alors réduite à deux, au prix d'un plus grand nombre de cellules qui devient $O(n(\log_2(n) - 1) + 1)$ cellules. Comme pour l'additionneur de Sklansky, le délai est en $O(\lceil \log_2(n) \rceil)$.

Afin de réduire le nombre de cellules du Kogge et Stone, Han et Carlson [HaCa87] proposent de calculer uniquement les positions impaires, et ensuite d'ajouter une couche de

cellules pour calculer les positions paires à partir des impaires. Le délai augmente légèrement à $O\lceil \log_2(n) \rceil + 1$, alors que la complexité est divisée par deux.

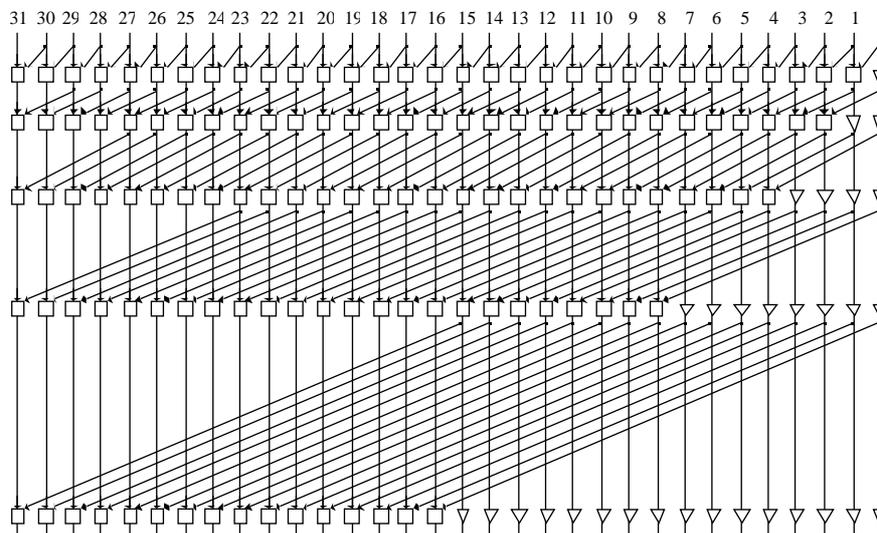


Figure 3.3 : Additionneur de Kogge et Stone (32 bits)

III.2.3. MODÈLES D'ACTIVITÉ [ASG97]

Dans cette partie nous développons un modèle de l'activité dans un additionneur à propagation de retenue (RCA); ce modèle ne tient pas compte de l'atténuation des transitions redondantes (§ I.7).

Nous appelons "activité" le nombre total de transitions pour toutes les paires de vecteurs d'entrée divisé par le nombre de paires de vecteurs d'entrée possibles.

Dans un additionneur à propagation, quand toutes les entrées sont disponibles simultanément, l'activité est surtout due à la propagation de la retenue à travers une chaîne de $p_i = 1$. Soit $T(n, k)$ le nombre de chaînes de k "1" consécutifs dans un mot de longueur n : Il est évident que :

- $T(n, 0) = 0$ (pas de chaîne de "longueur zéro")
- $T(n, n) = 1$ (c'est à dire 111...11)
- $T(n, n - 1) = 2$ (c'est à dire 2 possibilités : 011...111 or 11...1110).

Calculons maintenant le terme général $T(n, k)$ pour $0 < k < n$. Sachant que les extrémités de mot ainsi que la valeur 0 agissent comme séparateurs de chaînes, nous distinguons deux cas. Quand la chaîne touche l'une des deux extrémités du mot de n bits, il y a $2^{n-(k+1)}$ valeurs différentes pour les $n - (k + 1)$ bits en dehors de la chaîne :

$$\begin{array}{cc} \underline{11\dots10} & \underline{011\dots01} \\ k+1 & n-(k+1) \end{array}$$

Il y a $n - (k+2)$ possibilités pour que la chaîne se trouve au milieu du mot et, pour chaque position, il y a $2^{n-(k+2)}$ valeurs des $n - (k + 2)$ bits restants.

$$\begin{array}{cc} \underline{01\dots10} & \underline{011\dots01} \\ k+2 & n-(k+2) \end{array}$$

$$\text{Donc } T(n,k) = 2^{n-k} \left(1 + \frac{n-k-1}{4} \right) \text{ pour } 0 < k < n, \quad T(n,0) = 0 \text{ et } T(n,n) = 1$$

Introduisons maintenant la notion de “fenêtre d’activité”. La fenêtre d’activité d’une cellule est le créneau de temps pendant lequel sa sortie peut changer d’état. Par exemple, dans une chaîne de propagation de retenue (figure 3.4), la fenêtre d’activité de la $i^{\text{ème}}$ cellule est $[i, i^*]$ (où i est le délai d’une cellule) : l’ $i^{\text{ème}}$ cellule peut être active du temps i au temps i^* .

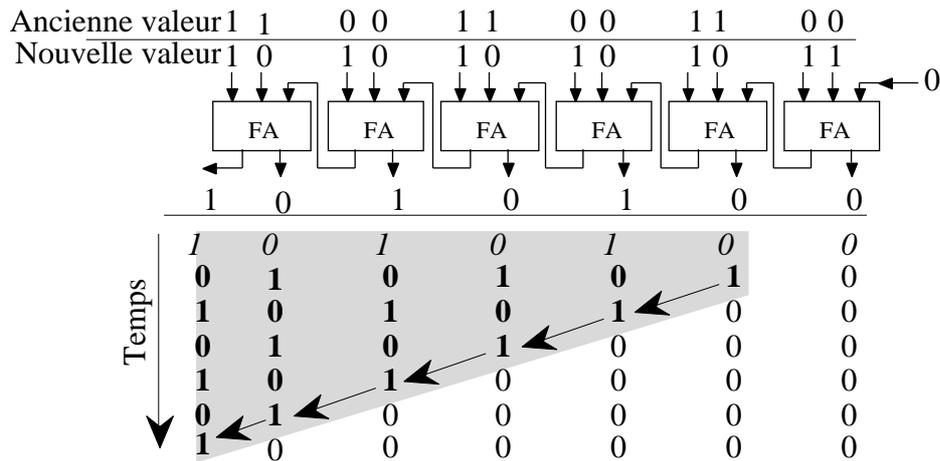


Figure 3.4 : Fenêtres d’activité dans une chaîne de propagation.

De façon générale, les fenêtres d’activité ne s’ajoutent pas. En supposant que les entrées sont disponibles au même moment, les sorties des cellules 1 et 2 (figure 3.5) peuvent uniquement changer d’état à $[1, 2]$, et la sortie de la cellule 3 peut uniquement changer à $[2, 3]$. Mais la cellule 4 ne peut être active que 1^* après le changement de l’une de ces entrées, soit à $[3, 4]$ ou bien à $[3, 4]$. En ne prenant en compte que la différence de délai entre les entrées de la cellule 4 (c’est à dire sa profondeur logique), nous surestimerions l’activité.

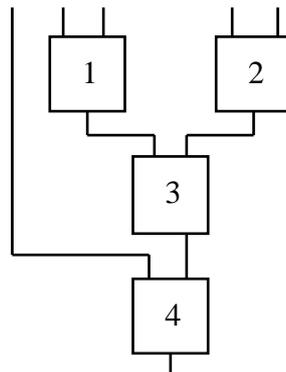


Figure 3.5 : Un autre exemple de fenêtres d’activité.

III.2.3.1. ADDITIONNEUR À PROPAGATION

Dans le cas d’un additionneur à propagation (RCA), aucune des sorties n’est obtenue par un arbre binaire, aussi les fenêtres d’activités des cellules se confondent avec la profondeur logique. Cette remarque ne s’applique pas aux autres architectures où les sorties sont obtenues par des arbres binaires, comme par exemple l’additionneur de Kogge et Stone.

L'activité engendrée par la propagation sur k positions est proportionnelle à $k^2/2$ [MoPa96].
D'où l'activité moyenne :

$$A(n) = \frac{1}{2^n} \sum_{k=0}^n \frac{k^2}{2} \cdot T(n,k). \quad (3.1)$$

Rappelons quelques formules utiles [Kre93] :

$$\begin{aligned} \sum_{i=0}^n i \cdot 2^{-i} &= 2 - \frac{n}{2^n}, & \sum_{i=0}^n i^2 \cdot 2^{-i} &= 6 - \frac{n^2}{2^n} \\ \sum_{i=0}^n i^3 \cdot 2^{-i} &= 26 - \frac{n^3}{2^n} \end{aligned} \quad (3.2)$$

qui permettent de simplifier l'expression de l'activité :

$$A = \frac{3n-4}{4} - \frac{3n^2}{2^{n+3}} \sum_{k=0}^n \frac{3n-4}{4}. \quad (3.3)$$

Avec ces formules nous pouvons aisément vérifier que la moyenne des longueurs est bien $n/2$

$$\frac{1}{2^n} \sum_{k=0}^n k \cdot T(n,k) = \frac{n}{2} - \frac{3n}{2^{n+2}} \sum_{k=0}^n \frac{n}{2} \quad (3.4)$$

Dans la suite, nous prendrons en compte uniquement les termes du premier ordre, c'est à dire $A = \frac{3n-4}{4}$. Le tableau 3.1 montre les erreurs relatives dues à cette approximation pour 8, 16, 32 et 64 bits.

Tableau 3.1 : Erreurs relatives

# de bits	Activité (%)	Délai (%)	η (%)
8	9,3750	2,34	8,2759
16	0,15	0,02	0,04
32	8,94e-06	5,59e-07	4,26e-06
64	8,33e-15	2,60e-16	2,58e-06

Étant donné l'équiprobabilité des vecteurs d'entrée, le nombre moyen de transitions utiles dans un RCA est égal à la moitié du nombre total de cellules. L'activité totale A peut ainsi être divisée en 2 parties :

$$A = A_{\text{utile}} + A_{\text{redondante}}$$

D'où le rapport η de l'activité redondante sur l'activité totale :

$$\eta = \frac{A_{\text{redondante}}}{A} = \frac{A - A_{\text{utile}}}{A} = \frac{n-4}{3n-4} \quad (3.5)$$

Pour de grandes valeurs de n , $\eta \approx 1/3$. Ce résultat qui conforte les analyses par BDD avec un modèle de délai unitaire est cohérent avec [LMJ95].

En adoptant cette approche, il devient aussi possible de déterminer l'activité à un instant donné t_i (figure 3.6).

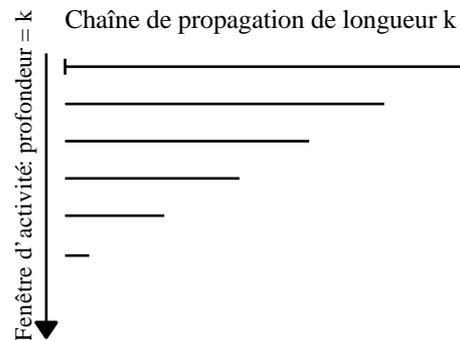


Figure 3.6 : Activité à un instant donné t_i .

L'activité à l'instant t_1 est donnée par : $A(t_1) = \frac{1}{2^n} \cdot \sum_{k=1}^n k.T(n,k)$. Les chaînes de propagation de longueur k qui subsistent au temps t_2 sont celles de longueur $k+1$ à t_1 , d'où : $A(t_2) = \frac{1}{2^n} \cdot \sum_{k=1}^{n-1} k.T(n,k+1)$. La somme commence à $n-1$ car dans un mot de longueur n , il n'y a pas de chaîne de longueur supérieure à n .

Plus généralement, l'activité à l'instant t_i est donnée par :

$$A(t_i) = \frac{1}{2^n} \cdot \sum_{k=1}^{n-i} k.T(n,k+i) = 2^{-i} \cdot \frac{n-i}{2} + 2 \quad (3.6)$$

III.2.3.2. AUTRES ARCHITECTURES

"Née du désir d'un seul, l'architecture est vite le bien de tous", J.-P. Michel

Le modèle décrit précédemment est étendu aux additionneurs qui peuvent être obtenus par association de chaînes de propagation, comme par exemple l'additionneur à anticipation de retenue (2-CSA).

Dans le calcul de η , l'activité utile A_{utile} est prise comme étant la moitié du nombre de cellules, étant donné que "0" et "1" sont équiprobables.

CSA À DEUX NIVEAUX

Le 2-CSA est en fait un RCA tronqué en blocs. Pour n bits, la longueur de ces blocs varie entre 1 et $\sqrt{2n} - 1$. Donc, un 2-CSA de n bits, peut être vu comme $\sqrt{2n}$ RCAs de longueur variant de 1 à $\sqrt{2n} - 1$ (premier niveau) plus une rangée de cellules qui forment le second niveau.

PREMIER NIVEAU

L'activité du premier niveau est donnée par la somme des activités de chaînes de propagation indépendantes.

$$\begin{aligned}
A_{1^{\text{er}} \text{ niveau}} &= \frac{1}{2^n} \cdot \sum_i^{\sqrt{2n}} [i \cdot A(i)] \\
&= \frac{1}{2^n} \cdot \sum_i^{\sqrt{2n}} i \cdot \frac{3i-4}{4} \\
A_{1^{\text{er}} \text{ niveau}} &= \frac{\sqrt{2n}(4n-3) - 6.n}{8}
\end{aligned} \tag{3.7}$$

SECOND NIVEAU

Le second niveau du 2-CSA est ici modélisé par une chaîne de propagation de longueur $\sqrt{2n}$, dans laquelle chaque cellule à la position k est dupliquée k fois. Cette approche néglige en fait l'activité au second niveau qui est due aux transitions redondantes du premier niveau. L'activité d'une telle chaîne, peut être déduite de celle du RCA en dupliquant k fois la $k^{\text{ième}}$ cellule.

$$\begin{aligned}
A_{2^{\text{nd}} \text{ niveau}} &= \frac{1}{2^n} \cdot \sum_{k=1}^{\sqrt{2n}} k \cdot \frac{k^2}{2} \cdot T(n,k) \\
A_{2^{\text{nd}} \text{ niveau}} &= \frac{6 \cdot \sqrt{2n} - 57}{8}
\end{aligned} \tag{3.8}$$

ACTIVITÉ TOTALE DU 2-CSA

L'activité totale du 2-CSA est la somme des activités des premier et second niveaux :

$$\begin{aligned}
A_{\text{Total}} &= A_{1^{\text{er}} \text{ niveau}} + A_{2^{\text{nd}} \text{ niveau}} \\
A_{\text{Total}} &= \frac{\sqrt{2n}(4n+3) - 6n - 57}{8}
\end{aligned} \tag{3.9}$$

Toujours dans l'hypothèse que l'activité utile est égale à la moitié du nombre de cellules, le pourcentage de transitions redondantes peut être calculé comme suit :

$$\begin{aligned}
\eta_{2\text{CSA}} &= \frac{A - A_{\text{utile}}}{A} \\
\eta_{2\text{CSA}} &= \frac{(\sqrt{2n}(4n+7) - 14.n - 57)}{\sqrt{2n}(4n+3) - 6n - 57}
\end{aligned} \tag{3.10}$$

ADDITIONNEUR KOGGE & STONE

Chaque bit de l'additionneur de Kogge et Stone est obtenu par un arbre binaire équilibré, donc chaque cellule peut voir sa sortie changer au plus une fois par cycle d'horloge. Il n'y a pas de transitions redondantes.

La propagation de retenue est le résultat d'un ET logique, donc son activité décroît très rapidement avec la profondeur logique (comme 2^{-i}), mais la probabilité de transition pour la génération de retenue est quasiment constante ($1/2$). Ces considérations nous permettent d'approximer l'activité de l'additionneur de Kogge et Stone par la moitié du nombre des cellules qui le composent :

$$A_{\text{K\&S}} = \frac{n}{4} \log_2 n, \text{ et } \eta_{\text{K\&S}} = 0 \quad (3.11)$$

III.2.3.3. COMPARAISON

Le tableau 3.2 résume les résultats obtenus pour différentes architectures d'additionneurs. La limite de validité de ces formules est surtout donnée par l'hypothèse de départ qui consiste à considérer un délai unitaire pour les cellules, en ne tenant compte ni du routage ni de la sortie. Ces équations sont donc à manipuler avec moult précautions. Ces indications ont surtout une valeur comparative.

Tableau 3.2 [TVG95] : Comparaison d'additionneurs

Type	# de cellules ∇	Délai (cellules ∇)	Sortance Max.	Activité Utile
RCA	$n - 1$	$n - 1$	2	$n / 2$
2-CSA	$\lceil 2n - \sqrt{2n} \rceil$	$\lceil \sqrt{2n} \rceil$	$\lceil \sqrt{2n} \rceil$	$\lceil n - \sqrt{n/2} \rceil$
3-CSA	$5/2 n - 3 \log_2(n/2)$	$\lceil \sqrt[3]{6n} \rceil$	$\lceil \sqrt[3]{6n} \rceil$	N.A.
Brent-Kung	$\lceil 2n - \log_2(n) \rceil$	$\lceil \log_2(n) \rceil$	$\lceil 2 \log_2(n) - 2 \rceil$	N.A.
Sklansky	$\lceil n/2 \log_2(n) \rceil$	$\lceil \log_2(n) \rceil$	$n/2$	$\lceil n/4 \log_2(n) \rceil$
Kogge & Stone	$\lceil n \log_2(n) - n + 1 \rceil$	$\lceil \log_2(n) \rceil$	2	$\lceil n \log_2(n) - n + 1 \rceil / 2$
Han & Carlson	$\lceil n \log_2(n) - n + 1 \rceil / 2$	$\lceil \log_2(n) \rceil + 1$	2	$\lceil n \log_2(n) - n + 1 \rceil / 4$

III.3. MULTIPLICATION

"Toute évaluation est le produit de la valeur de l'estimé par la sphère d'appréciation de l'estimateur", A.S.

L'ère préhistorique où la multiplication était réalisée par logiciel ou micro-programmation est révolue; cette opération est aujourd'hui une instruction courante pour tout microprocesseur. Il convient donc de particulièrement soigner son implantation matérielle [JM2]. Il y a aussi une ère moins préhistorique, mais non moins révolue où le compromis *surface / temps d'exécution* était dominant quant au choix de l'architecture. Nous nous intéresserons ici à d'autres contraintes plus contemporaines liées au nombre de transitions et à la dissipation d'énergie (et au temps d'exécution bien sûr), mais non pas à la surface.

Supposons que nous cherchions à calculer le produit $P = A * B$ où $A = \sum_{i=0}^{n-1} a_i * 2^i$ et $B = \sum_{j=0}^{n-1} b_j * 2^j$, les chiffres $a_i, b_i \in \{0, 1\}$.

Le produit $P = A * B$ est $\left(\sum_{i=0}^{n-1} a_i * 2^i \right) * \left(\sum_{j=0}^{n-1} b_j * 2^j \right)$ ou par simple distributivité $P = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} (a_i * b_j * 2^{i+j})$. $A < 2^n$ et $B < 2^n$ implique $P < 2^{2n}$. P peut donc s'écrire sur $2n$ bits. Multiplier revient à réduire la somme pondérée de n^2 bits ($a_i * b_j$) en une somme pondérée de $2n$ bits. Cette réduction se fait en deux étapes : D'abord les n^2 bits sont réduits à deux nombres de $2n$ bits, ensuite ces nombres sont additionnés (figure 3.7).

La première étape peut s'effectuer en temps $O(n)$, $O(\lceil \sqrt{n} \rceil)$, $O(\lceil \log_{3/2}(n) \rceil)$, $O(\lceil \log_2(n) \rceil)$ selon l'architecture. L'addition a déjà été abordée au paragraphe précédant (les additionneurs spécifiques à la multiplication seront abordés en fin de section). En ce qui concerne le délai, la réduction des produits partiels prend environ les 2/3 du temps, et l'addition finale 1/3. Dans le paragraphe suivant nous présentons différentes façons de réduire les produits partiels.

Calcul des n^2 produits partiels $(a_i \ b_j) * 2^{i+j}$
reduction des n^2 bits en 2 nombres de $2n$ -bit
addition des 2 nombres de $2n$ -bit

Figure 3.7 : Multiplication en trois étapes

III.3.1. SYNTHÈSE DE MULTIPLIEURS

Dans ce paragraphe nous présentons différentes architectures pour la réduction des produits partiels, puis nous discutons de l'évaluation de la consommation dans ces différentes architectures.

III.3.1.1. BRAUN [BRA63]

L'idée de base du multiplieur de Braun est de réduire les produits partiels à l'aide d'additionneurs 1 bit (cellule FA) qui sont en fait des réducteurs 3 : 2 (ou 3 donne 2). Son avantage est sa régularité : en effet, la matrice peut-être obtenue par simple aboutement de cellules FA (figure 3.8). Ses principaux inconvénients sont :

- son délai : le délai de cette architecture est linéaire avec la taille des opérandes
- sa consommation : il y a un grand nombre de chemins de propagation qui génèrent beaucoup d'activité redondante. Cette question sera approfondie dans la section suivante.

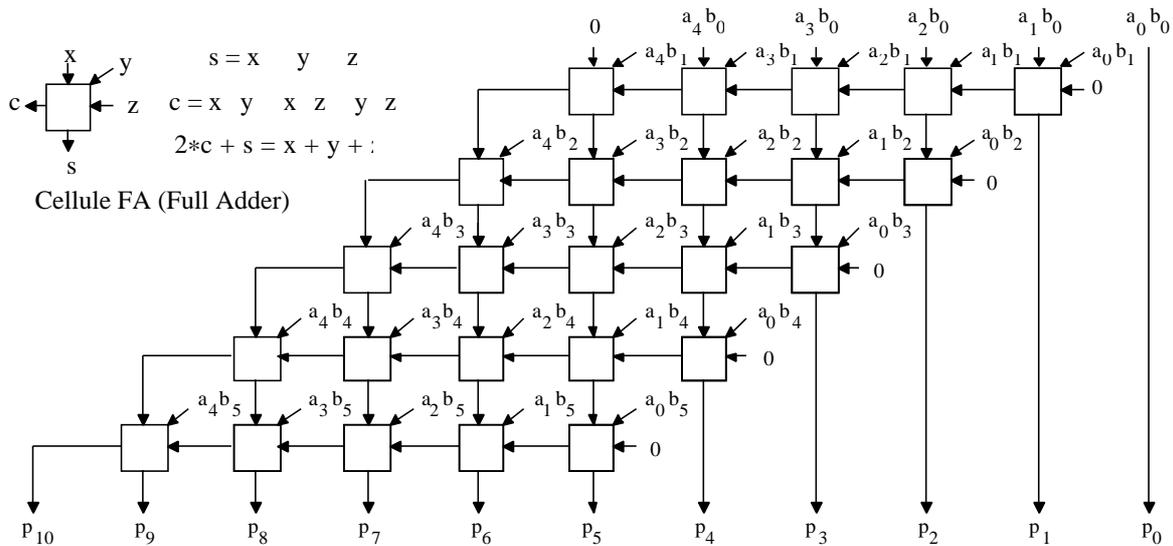
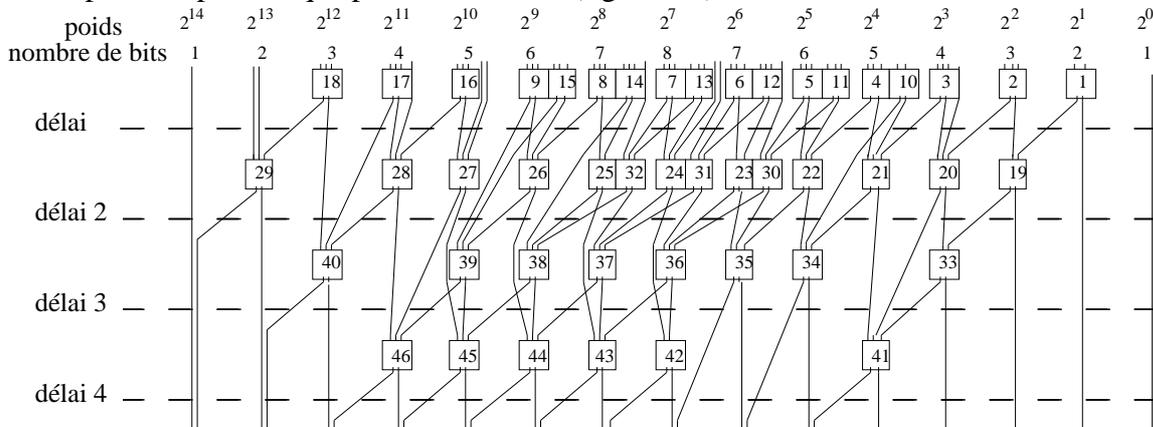


Figure 3.8 : Multiplieur de Braun

III.3.1.2. WALLACE DADDA [DAD65]

L'idée principale de l'arbre de Wallace est de tirer avantage du fait que les produits partiels sont disponibles au même moment, et ainsi de les réduire en parallèle [JM2]. Au lieu d'additionner une somme, une retenue et un produit partiel avec chaque cellule FA, on additionne autant de produits partiels que possible à la fois (figure 3.9)



Vers addition rapide (10 bits)

Figure 3.9 : Arbre de Wallace

Le délai de cette architecture est proportionnel au logarithme de la taille des opérandes. Plus précisément, au logarithme base 3/2 - car trois bits d'un étage sont réduits à 2 à l'étage suivant (cellule FA). Son coût est en $O(n \cdot \log(n))$.

En 1965, L. Dadda [Dad65] proposa une amélioration à l'arbre de Wallace. On peut trouver des explications détaillées concernant les méthodes de construction des arbres de réduction dans [JM2, AGcours]. Sommairement, l'amélioration consiste à ne pas prendre les produits partiels systématiquement au plus tôt, mais à équilibrer le nombre de bits à réduire par poids de façon à gagner du temps.

III.3.1.3. RÉDUCTEURS 4 : 2

Au lieu d'utiliser des cellules FA, il est possible de prendre des cellules ayant un plus grand nombre d'entrées. Les réducteurs 4 : 2 sont des cellules avec 4 entrées et une retenue entrante et 2 sorties et une retenue sortante. Une réalisation possible est représentée sur la figure 3.10. Il existe un grand nombre d'implantations : pour plus de détails voir [AGcours].

L'utilisation de réducteurs 4 : 2 (aussi appelés cellules CS) permet de diminuer le nombre d'étapes nécessaires à la réduction des produits partiels. Si ces cellules sont assemblées en arbre binaire équilibré, le nombre d'étapes est de $\text{Log}_2(n)$. Quand le nombre de bits à réduire est inférieur à 4, nous utilisons une cellule FA (ou HA).

Sur la figure 3.10, l'une des entrées du multiplexeur est inversée. Ce qui permet d'une part de simplifier l'équation du multiplexeur, et d'autre part de gagner le délai d'un inverseur sur le chemin critique en utilisant une porte Nand au lieu d'une porte And.

Le multiplieur 16x16 bits implanté dans le dernier chapitre utilise la cellule 4 : 2 de la figure 3.10. L'architecture adoptée est illustrée par le tableau 3.3.

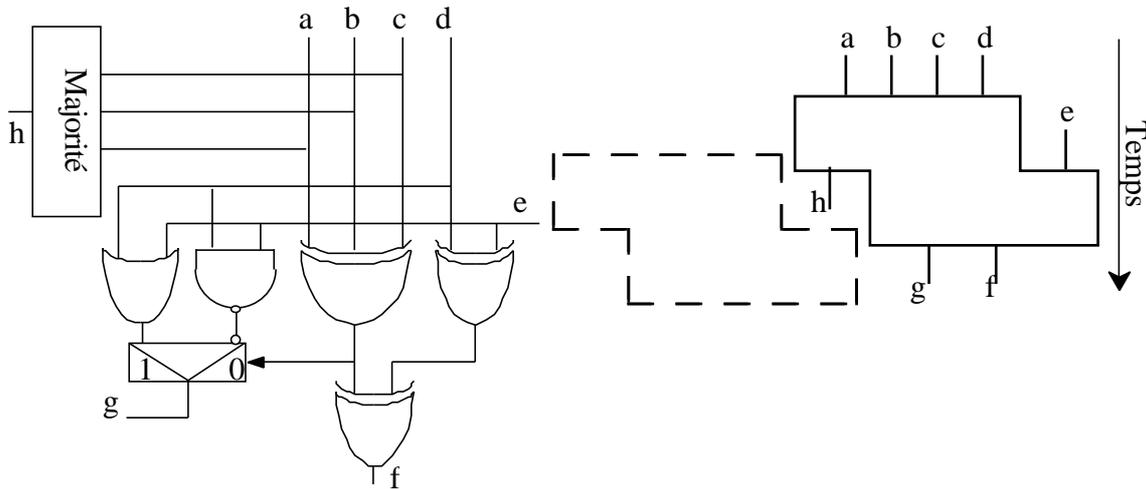


Figure 3.10 : Réducteurs 4 : 2

Tableau 3.3 : Architecture de multiplieur

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	# de bits à réduire	
0	0	0	1	1	1	2	2	2	2	3	3	3	4	4	3	3	2	2	2	2	2	2	1	1	1	0	0	0	0	0	# de bits de retenues horizontales	
		FA	FA	CS	FA	FA	HA																									
						FA	CS																									
1	3	2	4	3	5	5	4	5	7	7	6	8	8	8	8	7	7	8	8	7	6	5	4	3	4	2	3	2	1	1	# de bits à réduire	
0	0	0	0	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	0	0	0	0	0	# de bits de retenues horizontales	
			FA		CS																											
1	3	3	2	5	3	3	2	4	4	4	4	4	4	4	4	4	4	4	4	3	4	3	2	2	2	2	2	1	1	1	# de bits à réduire	
0	0	0	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	# de bits de retenues horizontales	
		FA	FA	FA	CS	FA	FA	FA	CS																							
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	1	1	

III.3.2. ÉVALUATION DE L'ACTIVITÉ

L'évaluation de l'activité dans un multiplieur de manière analytique (implicitement exhaustive) se heurte à une complexité sans commune mesure avec l'additionneur. Cette complexité est principalement due au grand nombre de signaux interdépendants. D'autre part, les hypothèses simplificatrices nécessaires ajouteraient une telle incertitude sur les résultats, qu'ils seraient inutilisables. Le simulateur PowerMill [DHNT95] aurait pu donner satisfaction dans ce cas, mais il n'est pas disponible. D'autres solutions doivent donc être envisagées.

Il est clair que les résultats les plus précis seront obtenus par simulation électrique, mais de telles simulations sont trop coûteuses en temps pour être effectuées sur un grand nombre de vecteurs (approche statistique). En revanche, une simulation logique permet d'appliquer un grand nombre de vecteurs sans pour autant pouvoir mesurer précisément la consommation d'énergie.

III.3.2.1. SIMULATION MONTE CARLO

La méthode Monte Carlo [BNYT93] est une méthode statistique d'approximation d'une variable donnée par mesures (ou simulations) successives. La confiance qui peut être accordée à une telle approximation dépend du nombre de simulations et du nombre de nœuds [Kre93]. L'intervalle de confiance que nous considérons est de 0.1% (tableau 3.4)

Tableau 3.4 : Comparatif Monte-Carlo

Architecture	Nombre minimum de vecteurs	Nombre effectif
Braun	7463	8000
Wallace	4943	8000
Arbre binaire (réducteurs 4 : 2)	3786	8000

Pour plus de détails sur la méthode Monte Carlo, on pourra consulter [Pap84], [Naj92] et [Kre93].

III.3.2.2. APPROCHE COMBINÉE HSPICE / VERILOG

Nous adoptons une approche "Monte Carlo améliorée" qui consiste à combiner deux simulateurs : l'un électrique (HSPICE) et l'autre logique (VERILOG). La combinaison se fait de la manière suivante :

- un intervalle de confiance est défini, donnant le nombre de vecteurs à appliquer [Kre93]
- l'architecture est décrite en VERILOG
- les cellules élémentaires sont simulées avec HSPICE pour toutes les combinaisons de paires de vecteurs possibles; à chaque transition est associée une énergie (1024 simulation pour la cellule CS à 5 entrées). Nous avons automatisé cette procédure. Les résultats sont tabulés dans un fichier. Les délais sont aussi mesurés depuis chaque entrée vers chaque sortie afin d'affiner les modèles VERILOG des cellules

- l'architecture désirée est ensuite simulée avec VERILOG pour un grand nombre de vecteurs garantissant l'intervalle de confiance fixé. Tout les nœuds sont enregistrés dans un fichier
- un programme en C nous permet alors d'analyser ce fichier; connaissant l'énergie consommée par chaque transition, nous pouvons en déduire la consommation globale
- un autre programme en C nous permet de compter les transitions redondantes pour chaque nœud.

Cette méthode est assez similaire à celle utilisée par l'outil HEAT (Hierarchical Energy Analysis Tool" présenté dans [SaPa96].

Le processus de simulation est illustré par la figure 3.11. Les outils développés concernent la génération automatique de tous les stimuli électriques, l'automatisation des simulations électriques et le traitement des fichiers VERILOG.

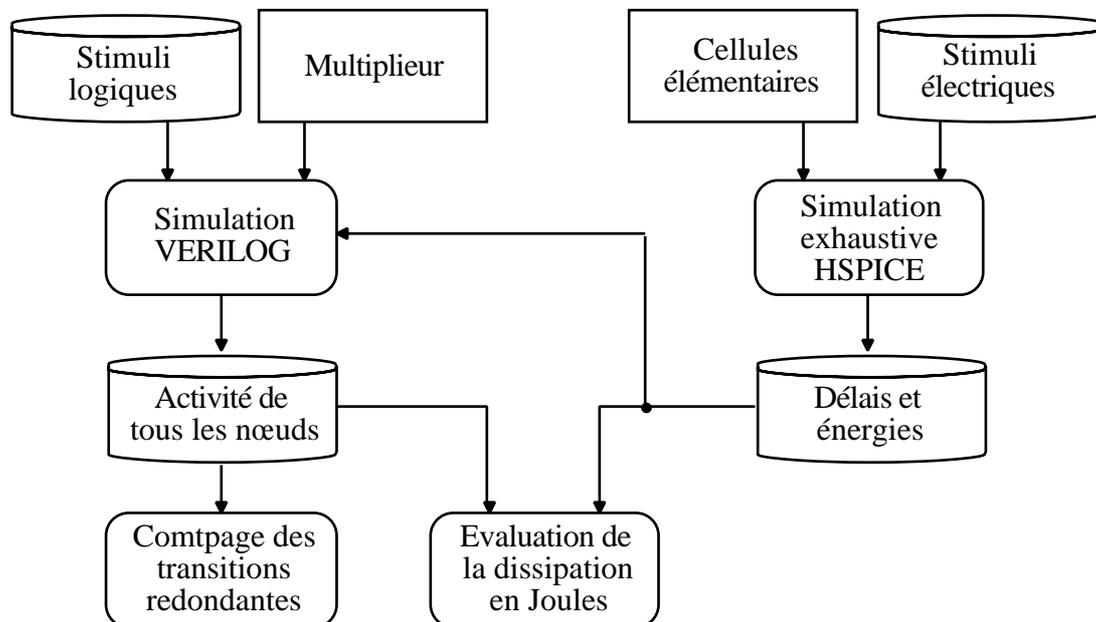


Figure 3.11 : Simulation Monte Carlo améliorée

Le tableau 3.5 résume les résultats obtenus par cette méthode pour différentes architectures de multiplieurs 8x8 bits. La colonne "Énergie totale" représente l'énergie consommée durant toute la simulation, soit pour les 8000 vecteurs.

Tableau 3.5 : Proportion de transitions redondantes

Architecture	Transitions redondantes (%)	Énergie totale (nJ)	Énergie par opération (pJ)
Braun [Bra63]	54,6	129	16,14
Wallace-Dadda [Dad65]	30,2	101	12,6
Arbre binaire	24,8	82,2	10,3

Nous retenons de ces simulations que, de manière générale, les arbres équilibrés génèrent moins de transitions redondantes. Cette stratégie sera mise en œuvre pour la réalisation du multiplieur 16x16 bits qui sera implanté au chapitre suivant.

III.3.3. ADDITIONNEURS POUR LA MULTIPLICATION

L'arbre de réduction des n^2 produits partiels génère deux nombres de $2n$ -bits. La dernière étape de la multiplication consiste à additionner ces deux nombres sans détériorer les performances du multiplieur; c'est ici que les additionneurs rapides interviennent encore.

La figure 3.12 donne les délais simulés pour un arbre de Wallace 16 par 16 bits; les sorties de ces arbres ne sont pas disponibles au même moment. $a_0 \quad b_0$ arrive en premier. Les derniers sont vers le milieu. L'enveloppe des délais n'est ni lisse ni symétrique.

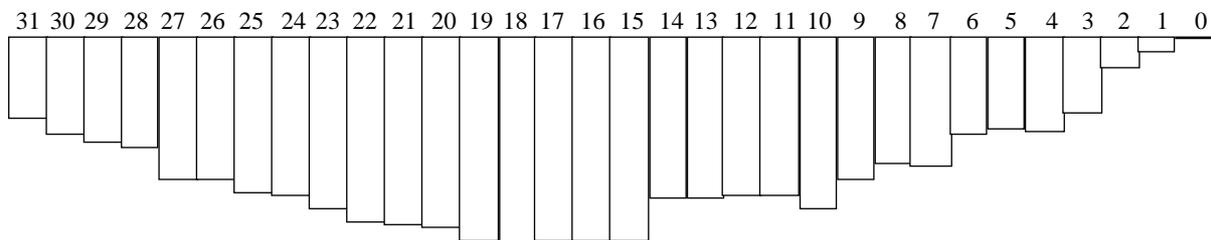


Figure 3.12 : Enveloppe temporelle pour l'arbre de Wallace

Néanmoins, un additionneur épousant cette enveloppe de délai est donné dans la figure 3.13.

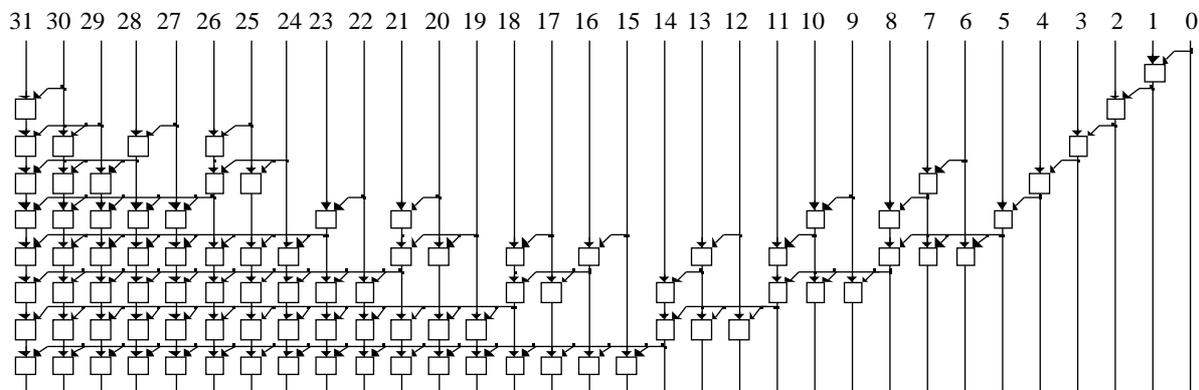


Figure 3.13 : Additionneur pour arbre de Wallace

III.4. DIVISION

III.4.1. INTRODUCTION À LA DIVISION RAPIDE

“*Divide ut imperes*” disait Philippe de Macédoine. Le déclin des diverses monarchies nous fait dire que la division est une opération bien difficile...

Selon [GoZu81] la division est environ 10 fois moins fréquente que les autres opérations élémentaires dans les calculs courants. Il existe un grand nombre d'algorithmes de division; ceux-ci sont classifiés dans la figure 3.14.

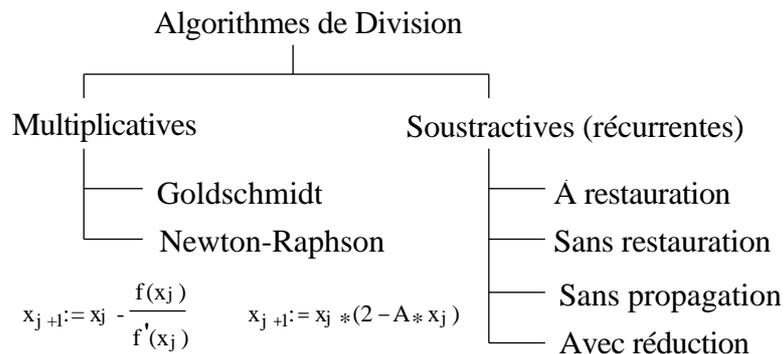


Figure 3.14 : Classification des algorithmes de division

Dans ce cas nous nous intéresserons à la division rapide, l'architecture choisie est récurrente sans propagation, et la base utilisée est 2.

Dans la suite, nous appellerons $Q = \sum_{i=0}^{n-1} q_i 2^{-i}$, le quotient; $A = \sum_{i=0}^{n-1} a_i 2^{-i} = a_0, a_1 a_2 \dots a_{n-1}$, le dividende et $D = \sum_{i=0}^{n-1} d_i 2^{-i}$, le diviseur normalisé ($d_0=1$). Le calcul à effectuer est : $Q = \frac{A}{D}$; la récurrence conduisant à ce résultat est :

$$Q_{i+1} = Q_i + q_{i+1} * 2^{-i-1}$$

$$R_{i+1} = R_i - q_{i+1} * D * 2^{-i-1} \quad \text{avec } q_i \text{ choisi tel que } R_i \geq 0$$

$$Q_0 = 0$$

$$R_0 = A$$

Comme le montrent les équations ci-dessus, la division par récurrence est constituée de n lignes d'addition/soustraction identiques. De ce fait, la rapidité du circuit dépend de manière significative du type d'opérateur utilisé. La division sans propagation de retenue qui a l'avantage de faire passer le délai de $o(n^2)$ à $o(n)$, nécessite l'emploi de cellules d'additionneur/ soustracteur modifiées.

Afin de supprimer la propagation de retenue dans les lignes du diviseur, une notation redondante est utilisée (notation "borrow save").

III.4.2. LA NOTATION "BORROW SAVE"

Dans le cas présent, la représentation des nombres en notation redondante est la suivante :

$$R = R^+ - R^- = \sum_{i=0}^{n-1} r_i^+ 2^{-i} = \sum_{i=0}^{n-1} (r_i^+ - r_i^-) 2^{-i}$$

avec : $r_i^+, r_i^- \in \{0,1\}$ et $r_i \in \{-1,0,1\}$

On a donc maintenant à effectuer une addition hybride $S = A + B$, avec $a_i, s_i \in \{-1, 0, 1\}$ et $b_i \in \{0, 1\}$ (figure 3.15).

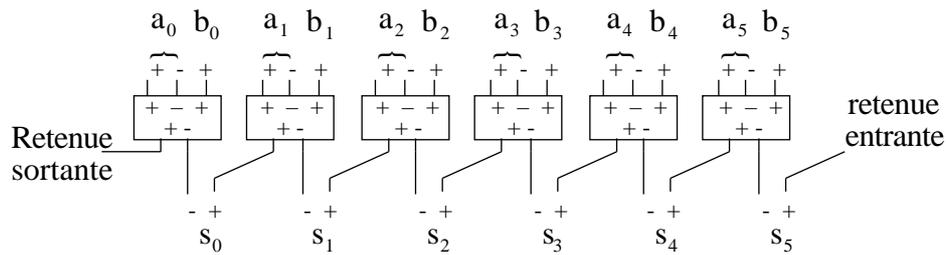


Figure 3.15 : Une ligne d'addition/soustraction hybride

Le diviseur obtenu est constitué comme tous les diviseurs récurrents d'une suite d'addition-neurs/soustracteurs, mais du fait de l'utilisation d'une notation redondante, il est suivi d'un convertisseur de "borrow save" vers standard. Cette conversion est effectuée ici en temps constant grâce à l'arrivée des bits du quotient dans le désordre (poids forts d'abord). L'architecture du convertisseur en temps constant est détaillée plus loin dans ce chapitre au paragraphe III.4.4.

La division redondante la plus répandue est la division SRT découverte indépendamment à la même époque par Sweeney, Robertson [ROB58] et Tocher[TOC58].

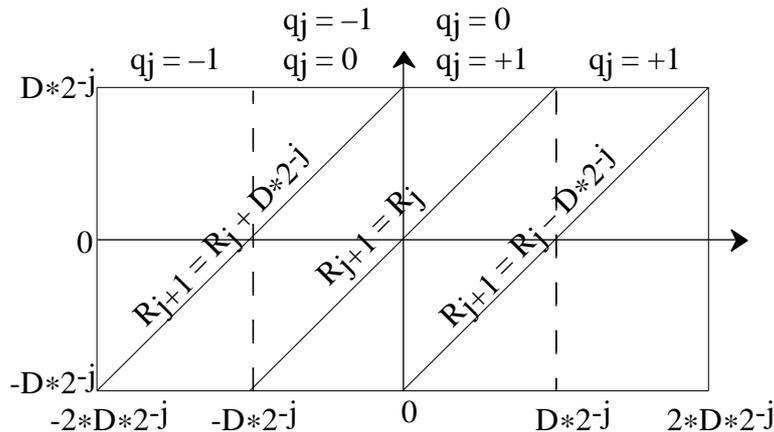


Figure 3.16 : Diagramme de Robertson de la division SRT en base 2

Le reste partiel obtenu à la $j^{\text{ème}}$ itération est borné : $-2 * D * 2^{-j} \leq R_j \leq 2 * D * 2^{-j}$.

La difficulté majeure est la sélection de l'opération à exécuter (choix de q_j); en effet, il faut être capable de replacer R_j dans le diagramme de Robertson (figure 3.16). Cependant, aux vues de la norme et de la notation choisies, il suffit d'examiner les trois premiers chiffres de R_j .

D est normalisé comme suit : $1 \leq D < 2$, donc $-4 < R_j * 2^j < 4$.

La condition de convergence étant : $|R_j| \leq 2 * D * 2^{-j}$ et $|R_{j+1}| \leq 2 * D * 2^{-j-1}$, l'examen de r_{-2} , r_{-1} , r_0 ($R_j = r_{-2}r_{-1}r_0$, $r_1r_2 \dots r_n$) suffit au choix de q_{j+1} .

$$\begin{array}{l}
 \left. \begin{array}{l}
 r_{-2} \ r_{-1} \ r_0 \\
 -1 \ x \ x \\
 0 \ -1 \ x \\
 0 \ 0 \ -1
 \end{array} \right\} -2 \cdot D \cdot 2^{-j} \quad R_j < 0 \quad R_{j+1} = R_j + D \cdot 2^{-j} \\
 \left. \begin{array}{l}
 0 \ 0 \ 0 \\
 0 \ 0 \ +1 \\
 0 \ +1 \ x \\
 +1 \ x \ x
 \end{array} \right\} -2^{-j} < R_j < +2^{-j} \quad R_{j+1} = R_j \\
 \left. \begin{array}{l}
 0 \ 0 \ 0 \\
 0 \ 0 \ +1 \\
 0 \ +1 \ x \\
 +1 \ x \ x
 \end{array} \right\} 0 < R_j \quad 2 \cdot D \cdot 2^{-j} \quad R_{j+1} = R_j - D \cdot 2^{-j}
 \end{array}$$

On introduit un estimateur de $R_j \cdot 2^j$, $\hat{R} = 4 \cdot r_{-2} + 2 \cdot r_{-1} + r_0$; $-1 < \hat{R} - R_j \cdot 2^j < 1$.

Un étage de diviseur peut alors être représenté selon la figure 3.17, les entrées sont R_j (R en notation redondante) et D (en conventionnel), les sorties sont R_{j+1} (S en redondant) et D (en conventionnel). L'opération effectuée est une addition si \hat{R} est négatif, une soustraction si \hat{R} est positif et une conservation si \hat{R} est nul.

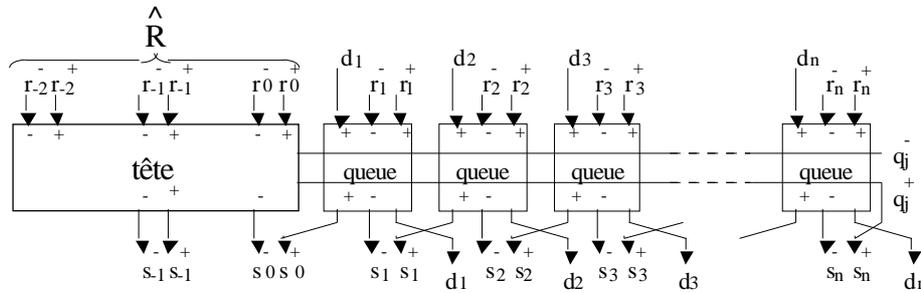


Figure 3.17 : Étage du diviseur

Le diviseur régulier complet en notation redondante est schématisé sur la figure 3.18.

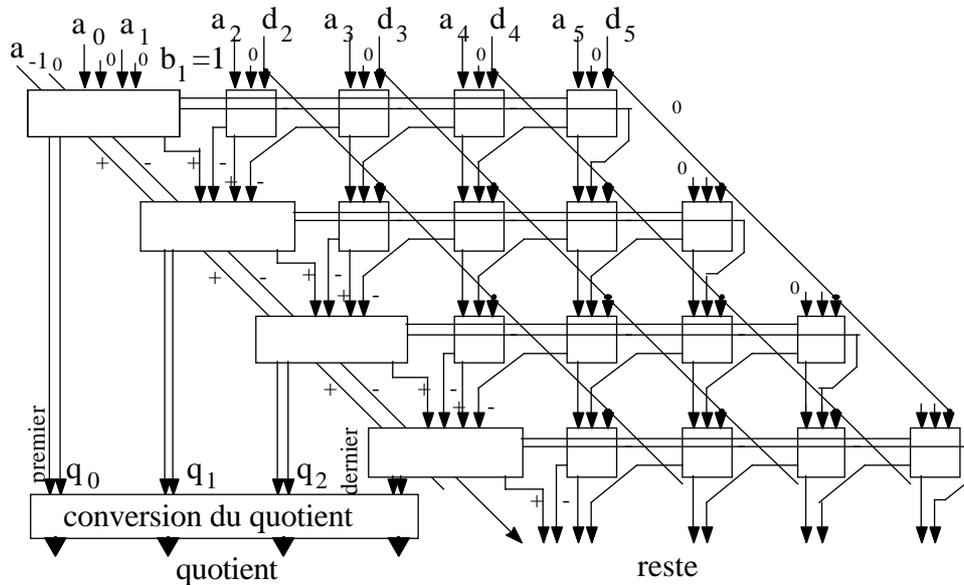


Figure 3.18 : Diviseur régulier en notation redondante

Le rôle de la cellule de tête est non seulement de déterminer l'opération à exécuter (addition, soustraction ou rien) mais encore d'effectuer cette opération sur les chiffres de tête et de recoder le résultat pour éliminer le chiffre de poids fort s_{-2} ; alors que le rôle des cellules de queue est

d'exécuter l'opération (addition, soustraction ou rien) sur les chiffres de queue sans retenue propagée et de transmettre D décalé vers les poids faibles. Le chemin critique du diviseur est illustré sur la figure 3.19.

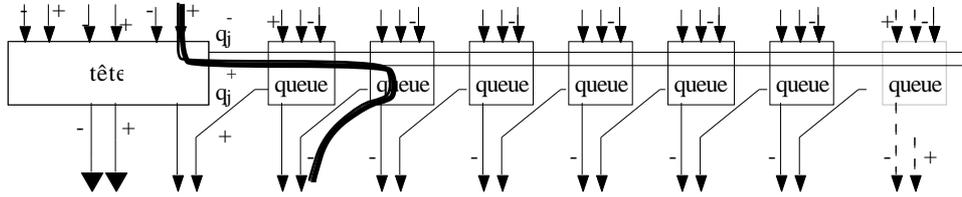


Figure 3.19 : Chemin critique : une cellule de tête et deux cellules de queue

III.4.3. LE DIVISEUR FAIBLE CONSOMMATION

Un diviseur est donc une superposition d'autant de tranches qu'il y a de bits de quotient (figure 3.18). Dû à leurs complexités, le délai des cellules de tête est plus important que celui des cellules de queue. Cette différence est d'autant plus importante que la base est grande. Ces différences de reconvergence entre les bits du quotient et ceux du reste partiel génèrent un grand nombre de transitions redondantes. La suppression de ces transitions réduirait la consommation globale du diviseur par un facteur d'environ 2. Mais il n'est malheureusement pas possible d'équilibrer les délais simplement en changeant la taille des transistors.

La régularité du diviseur cellulaire nous permet de partitionner les cellules de queue en deux blocs identiques chacun alimenté à $V_{dd} / 2$ comme indiqué sur la figure 3.20. Cette architecture nécessite des convertisseurs de niveaux logiques.

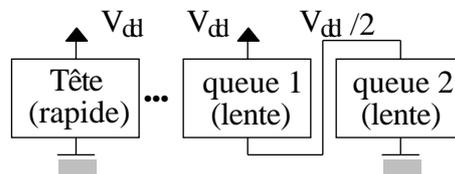


Figure 3.20 : Partitionnement des cellules de queue

Peuvent être alimentées de cette manière uniquement les cellules de queue qui sont suffisamment éloignées dans le temps du chemin critique, de sorte que l'augmentation de délai n'influe pas sur le délai global du diviseur. Le pourcentage de cellules de queue pouvant être alimentées ainsi croît avec la précision du diviseur; il devient supérieur à 25% pour un diviseur 53 bits.

Un système de régulation maintient la tension intermédiaire proche de $V_{dd} / 2$ même si le courant moyen à travers les blocs queue 1 et queue 2 sont différents. Ainsi la diminution de la consommation d'énergie varie entre 50% (un bloc est inactif) et 75% (même activité pour les deux blocs). La présence d'une petite capacité au point $V_{dd} / 2$ permet d'atténuer les différences de courant instantané entre les blocs et fournit un réservoir de charges.

En dehors de la réduction "mécanique" de l'énergie dissipée due à la réduction de la tension d'alimentation, le nombre de transitions redondantes est significativement diminué grâce à l'allongement du délai des cellules.

Le problème se rapporte en quelque sorte à celui du pavage d'un plan dont l'une des dimensions est le délai et l'autre les variables. La forme des pavés dans cet espace est déterminée par des simulations électriques HSPICE. La figure 3.21 montre la cellule de tête vue dans cet es-

pace. Les figures 3.22 et 3.23 montrent deux différentes implantations des cellules de queue alimentées à V_{dd} ou à $V_{dd} / 2$. La hauteur de la cellule pour une sortie donnée est proportionnelle au délai maximal entre une entrée quelconque et cette sortie.

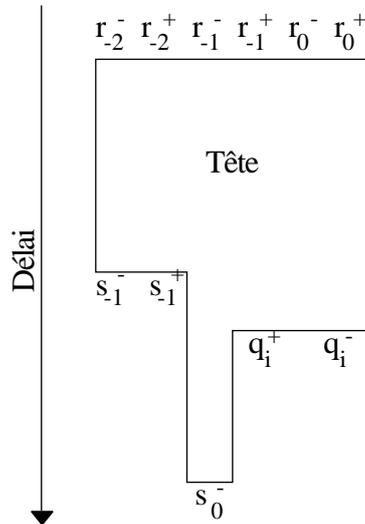


Figure 3.21 : Cellule de tête

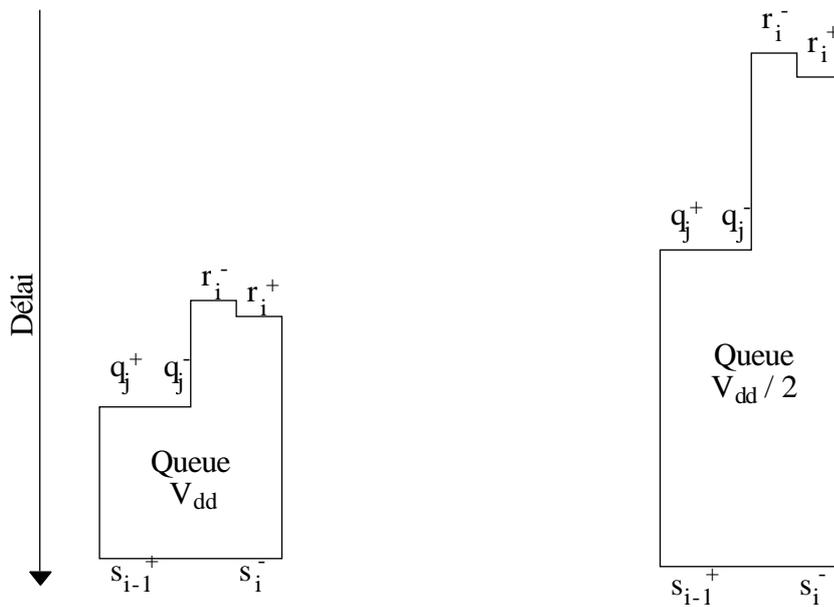
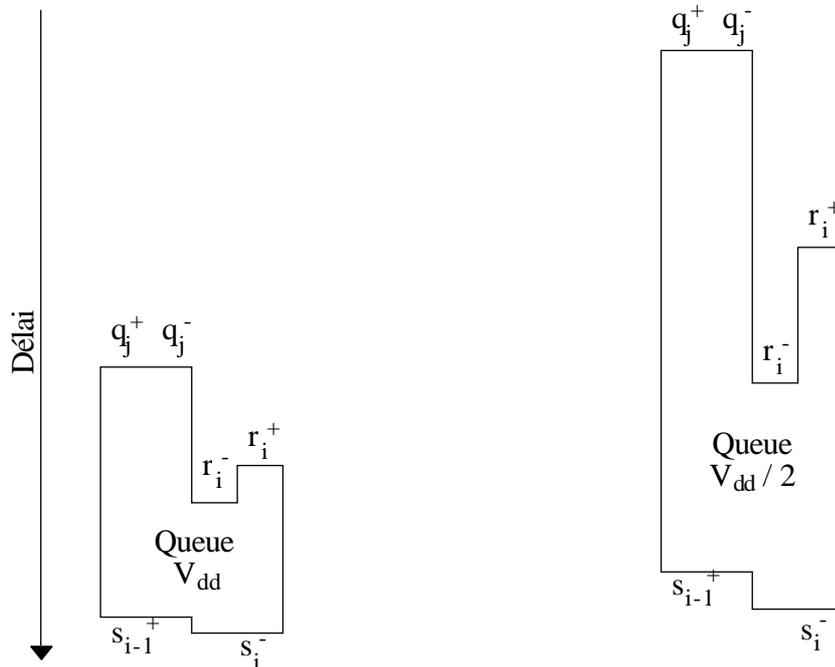


Figure 3.22 : Cellule de queue avec q_j en dernier

Figure 3.23 : Cellule de queue avec q_j en premier

Ces cellules ont été dessinées automatiquement à l'aide de l'outil de synthèse de layout de Cadence (LAS) avec la technologie ATMEL $0,7\mu\text{m}$. Elles ont ensuite été placées manuellement afin de paver au mieux l'espace bits / délai. Nous obtenons le layout de la figure 3.24. Du fait de l'utilisation de LAS, cette réalisation est loin d'être optimale en surface.



Figure 3.24 : Layout du diviseur basse consommation 16x16 bits

Pour cette implantation le choix de la base s s'est avéré crucial. Afin de pouvoir remplacer plus de cellules, une base plus grande s'impose car nous aurons un écart de délai plus important entre les cellules de tête et de queue. Le choix d'une plus grande base s'accompagnera d'une légère augmentation du délai global, mais cette étude reste à faire.

III.4.4. LA CONVERSION DU QUOTIENT

Nous abordons maintenant le problème de la conversion du quotient de la notation redondante en notation standard [GAAH97]. Il est possible de faire cette conversion en temps constant car les bits du mot à convertir arrivent dans l'ordre, poids forts en tête. Nous utilisons cette information, ainsi que les règles de construction d'additionneurs à l'aide de cellules qui ont été énoncées au début de ce chapitre.

L'additionneur de la figure 3.25 convient quand la $i^{\text{ème}}$ entrée arrive au moins i fois le délai d'une cellule avant le dernier bit (pente = 1). Le délai total de cette addition est celui d'une cellule. Le principal inconvénient de cet additionneur est sa complexité qui est de $\frac{1}{2} (n-1)^2$ cellules.

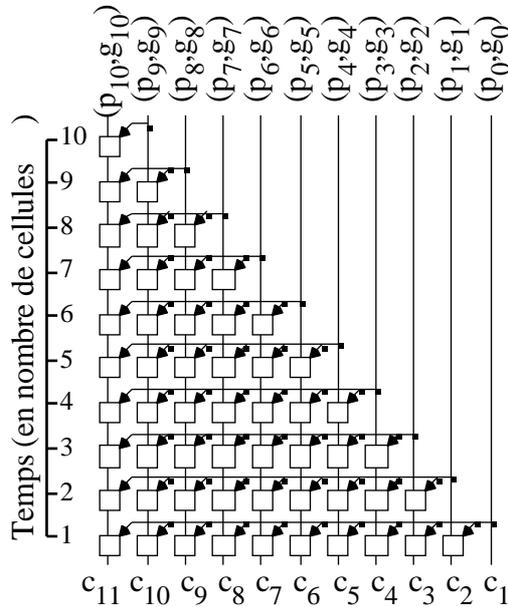


Figure 3.25 : Exemple d'additionneur en temps constant

Soit α le rapport entre la cadence d'arrivée et le délai d'une cellule. Si $\alpha > 1$ (ce qui est le cas du diviseur), la complexité de l'additionneur peut être réduite sans sacrifier le débit [ErLa87, GAAH97].

L'additionneur présenté sur la figure 3.25 est en fait similaire à un additionneur à anticipation de retenue "carry select" (CSA). Dans le cas d'un CSA conventionnel, deux résultats sont anticipés à l'aide de deux additionneurs : une retenue entrante à 1 pour l'un et à 0 pour l'autre. Comme dans un CSA, nous utilisons des blocs de taille croissante.

III.4.4.1. MODÉLISATION MATHÉMATIQUE

Appelons b_j la taille du $j^{\text{ème}}$ bloc et exprimons la récursivement :

$$b_j = b_{j-1}(1 + \alpha) - 1 \quad j \geq 2$$

avec la valeur initiale $b_1 = 1$.

$$b_j = (1 + \alpha)^{j-1} - \sum_{k=0}^{j-2} (1 + \alpha)^k$$

$$b_j = \frac{-1}{\alpha} (1 + \alpha)^{j-1} + \frac{1}{\alpha}$$

Le nombre de cellules du $j^{\text{ème}}$ bloc est $j * b_j$.

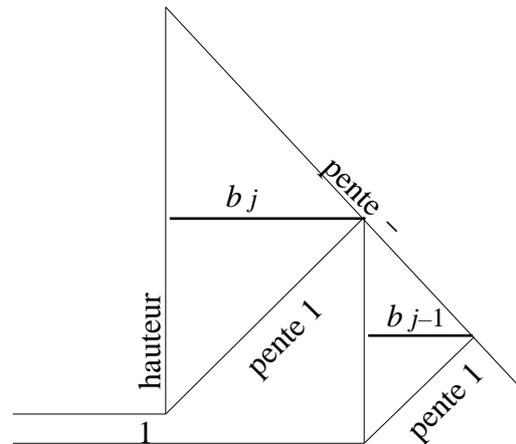


Figure 3.26 : Découpage en blocs

III.4.4.2. SYNTHÈSE DE LAYOUT

Les synthèses logique et topologique ont lieu en même temps. Sachant que dans le $j^{\text{ème}}$ bloc en partant de la droite, chaque colonne contient j cellules avec une propagation verticale et une avec une propagation horizontale, l'algorithme de génération détermine le numéro de bloc de chaque colonne.

Cet algorithme suppose un délai unitaire pour tout les blocs et ne prend pas en compte les capacités de routage. Ces effets peuvent être aisément considérés en intégrant le modèle de délai approprié dans l'algorithme.

```

i, j, b : integer ;      : real ;
j := 1 ;                { numéro de bloc
b := 1 ; { taille de bloc
for i = 1 to n-1 do { pour tout les bits
  if b = j then begin { il n'y a plus de hauteur
    b := i *      +1 { initialisation de la hauteur
    j := j + 1 { bloc suivant
  end else b := b -1 { moins de hauteur
endfor

```

k est le plus grand entier plus petit ou égal à k .

La figure 3.27 donne le plan de masse pour $n = 54$ (mantisse d'un flottant IEEE) pour différentes valeurs de k .

Le cas où les entrées arrivent plus vite que le délai d'une cellule n'est pas abordé car la solution est triviale : elle consiste à augmenter le délai global de l'additionneur en effectuant l'opération en autant de délai de cellules qu'il est nécessaire.

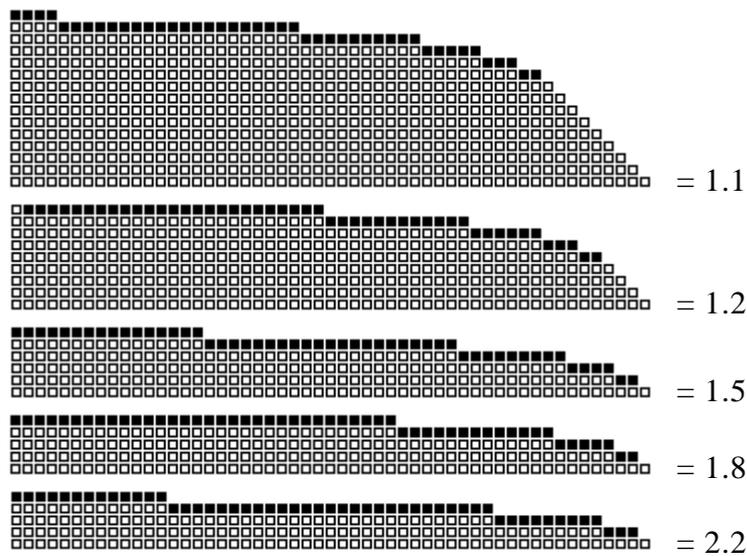


Figure 3.27 : Plan de masse en fonction de

Cet algorithme de génération d'additionneurs est incorporé dans GenOptim [HMH94, GAAH97], un outil de génération d'opérateurs développé au LIP6 (Paris) par A. Houelle, N. Vaucher et H. Mehrez. La figure 3.28 montre le layout d'un diviseur utilisant cette architecture de conversion du quotient.

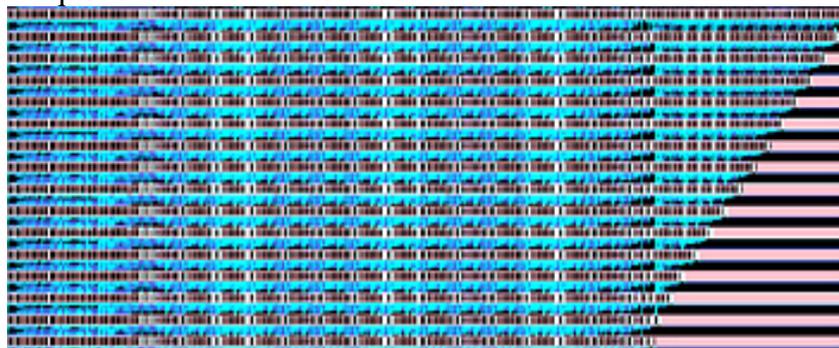


Figure 3.28 : Diviseur 16 bits avec conversion du quotient en temps constant (à droite)

III.5. RÉSUMÉ

Ce chapitre a été consacré aux architectures d'opérateurs arithmétiques pour la faible consommation d'énergie. nous avons abordé l'addition, la multiplication et la division.

Dans le cas de l'addition, un modèle analytique de l'activité a été développé pour l'additionneur à propagation de retenue (RCA). Ce modèle fut ensuite étendu à d'autres architectures.

Le développement de modèles analytiques pour l'activité dans les multiplieurs nécessite trop d'hypothèses simplificatrices qui rendraient ces prédictions inexploitable. L'approche adoptée ici a été statistique. Nous avons utilisé une méthode Monte Carlo modifiée qui consistait à combiner deux simulateurs :

- l'un logique (VERILOG) donc rapide mais peu précis
- l'autre électrique (HSPICE) très précis mais lent

Le problème d'architectures d'additionneurs rapides adaptés à la multiplication fut aussi abordé.

En ce qui concerne la division, notre choix s'est porté sur un algorithme récurrent de division rapide en base deux. La méthode choisie pour réduire la consommation de ce diviseur consistait à réduire la tension d'alimentation des cellules de queue en dehors du chemin critique en les mettant en série deux par deux entre V_{DD} et la masse. Seules les cellules suffisamment éloignées du chemin critique pouvaient être alimentées de la sorte; leur proportion augmente avec la précision du diviseur. Néanmoins, il s'est avéré que l'impact du choix de la base sur la proportion de cellules pouvant être alimentées en basse tension est déterminant. Une base 4 ou 8 donnerait de bien meilleurs résultats grâce à la plus grande différence qui existerait alors entre les cellules de tête et de queue.

Étant donné que les bits du quotient arrivent dans l'ordre - poids forts d'abord, nous avons suggéré une méthode de synthèse d'additionneurs permettant la conversion du quotient de la notation redondante en notation standard en temps constant. Un algorithme d'optimisation en surface de ces additionneurs a aussi été proposé.

III.6. SUMMARY

This chapter was devoted to low energy consumption arithmetic operators architecture. The operations tackled were addition, multiplication and division.

In the case of addition, an analytical activity model was developed for the ripple carry adder. This model was then extended to other adders.

Developing analytical activity models for multipliers requires too many simplifying hypotheses that would lead to irrelevant predictions. The adopted approach was statistical. An enhanced Monte Carlo method was applied. It consisted in combining two simulators :

- one logical simulator (VERILOG) : fast but not accurate
- and one electrical simulator (HSPICE) : very accurate but slow

The question of fast adders architecture for multiplication was also addressed.

As far as division was concerned, a fast radix-2 recurrent division algorithm was chosen. The energy reduction scheme adopted consisted in reducing the supply voltage of the tail cells that were not on the critical path. This reduction was achieved by putting them in series two by two between V_{DD} and ground. This method could be applied only to the cells that were sufficiently far from the critical path; their rate increases with the accuracy of the divider. Nevertheless, it was found that the choice of the radix has a dramatic impact on the proportion of cells that could use a low supply. A radix 4 or 8 would have given better results, as the difference in terms of delay between the head and the tail increases with the radix.

As the bits of the quotient were delivered at a fixed rate starting by the MSBs, an adders synthesis method allowing quotient conversion from redundant to standard notation in constant time was presented. Also, an algorithm for the area optimisation of these converters was proposed.

CHAPITRE QUATRIÈME : COMPARAISON 2D vs. 3D

IV.1. INTRODUCTION

Ce dernier chapitre est consacré à la comparaison des technologies bi- et tri-dimensionnelles afin d'évaluer les gains apportés par ce type d'intégration. Dans un premier temps, les critères de comparaison retenus sont exposés, ensuite sont présentés les circuits utilisés pour cette comparaison et enfin le dernier paragraphe contient les résultats et l'analyse de cette comparaison.

IV.2. CRITÈRES DE COMPARAISON

Ce paragraphe rappelle brièvement les critères de comparaison. Pour une description détaillée on pourra se reporter au § I.5.

IV.2.1. SURFACE

S'agissant d'intégration tridimensionnelle, la surface est le premier critère de comparaison qui vient à l'esprit. Ce n'est toutefois pas le plus important ici. Son influence est indirecte. Elle affecte la longueur des interconnexions ainsi que les capacités parasites, et par là, les caractéristiques électriques. L'unité de mesure de la surface sera ici le mm^2 . Les technologies ne sont pas encore suffisamment avancées pour que l'utilisation de l'hectare soit nécessaire.

IV.2.2. CAPACITÉ

Le critère de comparaison "capacité" permet d'apprécier dans quelle mesure les capacités dues aux interconnexions sont réduites. La mesure de ce paramètre s'est faite ici en additionnant toutes les capacités de tous les nœuds d'un circuit. Cette opération est effectuée à l'aide d'un programme traitant la "netlist" finale. La valeur finale comprend aussi bien les capacités intrinsèques des dispositifs que les capacités de routage. Les capacités intrinsèques étant les mêmes dans les deux cas (2D et 3D), la mesure de ce paramètre donne une idée précise de la réduction des interconnexions. La capacité des circuits sera donnée en pF.

IV.2.3. DÉLAI

Le délai du circuit se confond ici avec son débit car nous nous intéressons aux circuits combinatoires synchrones. Une amélioration du délai peut-être obtenue en augmentant la tension d'alimentation (dans la limite autorisée par la technologie); il n'y a pas de compromis possible car il s'agit d'une fonction monotone. Le délai sera donné en nanosecondes (ns).

IV.2.4. ÉNERGIE PAR OPÉRATION

L'énergie par opération EPO (ou bien le produit puissance - délai) mesure la quantité d'énergie nécessaire pour effectuer une opération donnée, indépendamment du temps d'exécution de cette opération. Il est toujours possible d'échanger du délai contre de l'EPO, mais il n'y a pas de minimum de l'EPO pour un délai donné; c'est, tout comme le délai, une fonction monotone. L'EPO est obtenue en intégrant le courant pendant la durée de l'opération; elle sera mesurée en picojoules.

IV.2.5. PRODUIT ÉNERGIE - DÉLAI

Le produit énergie - délai (EDP) représente un compromis entre deux contraintes incompatibles, le délai d'une part, et l'EPO d'autre part. Nous le mesurons en pJ/MHz.

IV.3. MULTIPLIEUR 16X16

Dans ce paragraphe nous présentons le multiplieur 16x16 bits qui a été implanté dans les deux technologies. L'architecture utilisée est une de celles présentées au paragraphe III.3. Nous avons retenu une architecture en arbres binaires à base de réducteurs 4:2 afin de minimiser les transitions redondantes et d'améliorer le délai. Cette architecture est représentée dans le tableau 4.1. : CS représente un réducteur 4:2, FA un additionneur 1 bit et HA un demi additionneur. La génération des produits partiels et l'additionneur final ne sont pas représentés sur le tableau, mais sont bien sûr présents sur les layouts finaux.

La réalisation de ce multiplieur a été faite en cellules standard, à l'aide des outils et des bibliothèques présentés au § II.3. Les layouts sont reproduits sur les figures 4.1 et 4.2.

Tableau 4.1 : Arbre de réduction

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	# de bits à réduire	
0	0	0	1	1	1	2	2	2	2	3	3	3	4	4	3	3	2	2	2	2	2	2	1	1	1	0	0	0	0	0	# de bits de retenues horizontales	
		FA	FA	CS	FA	FA	HA																									
1	3	2	4	3	5	5	4	5	7	7	6	8	8	8	8	7	7	8	8	7	6	5	4	3	4	2	3	2	1	1	# de bits à réduire	
0	0	0	0	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1	1	1	0	0	0	0	0	0	# de bits de retenues horizontales	
		FA		CS	HA	FA	HA																									
1	3	3	2	5	3	3	2	4	4	4	4	4	4	4	4	4	4	4	3	4	3	2	2	2	2	2	1	1	1	# de bits à réduire		
0	0	0	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	# de bits de retenues horizontales	
	FA	FA	FA	CS	FA	FA	FA	CS	FA																							
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	1	1	

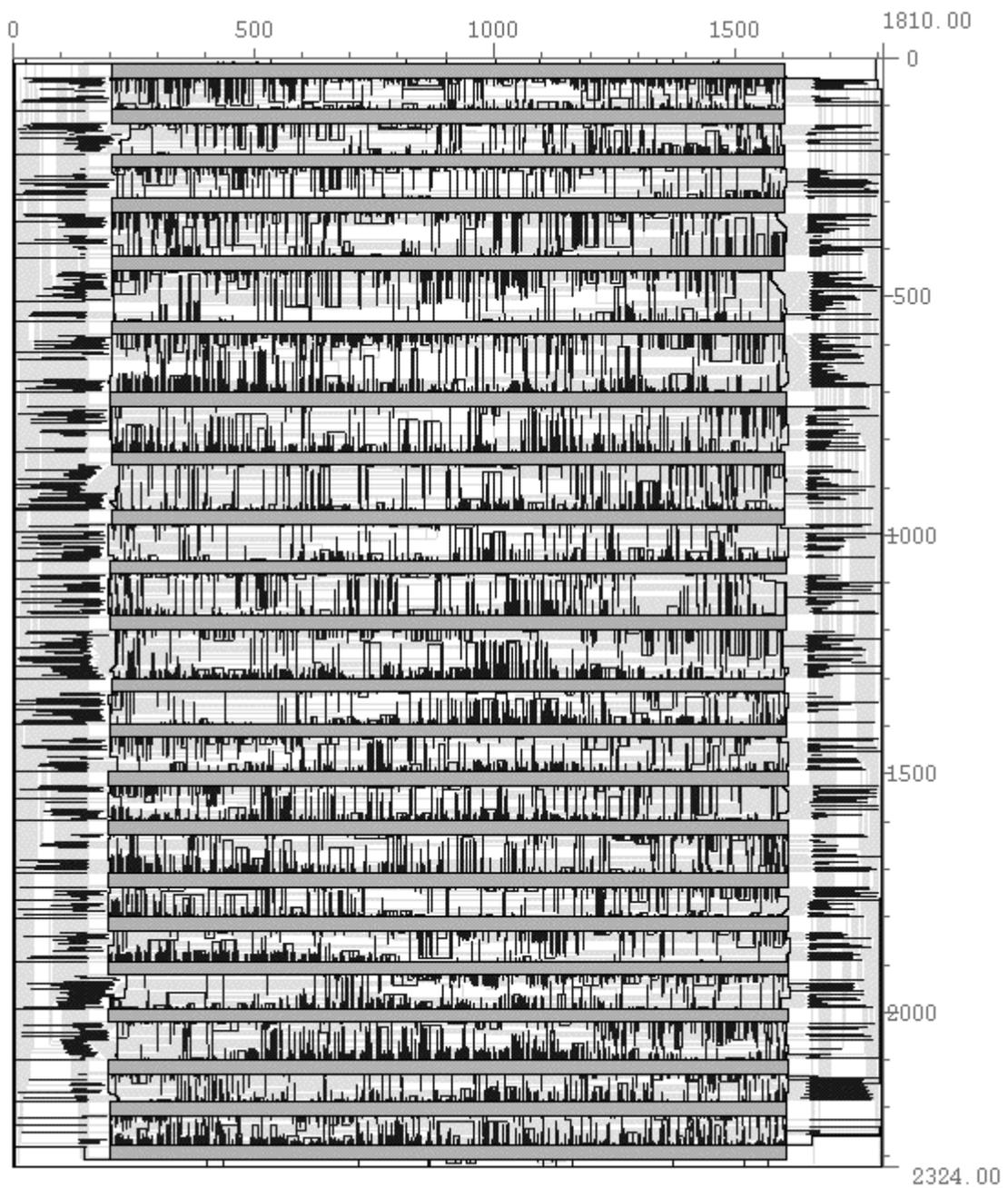


Figure 4.1 : Layout du multiplieur 16x16 en technologie SOI 2D

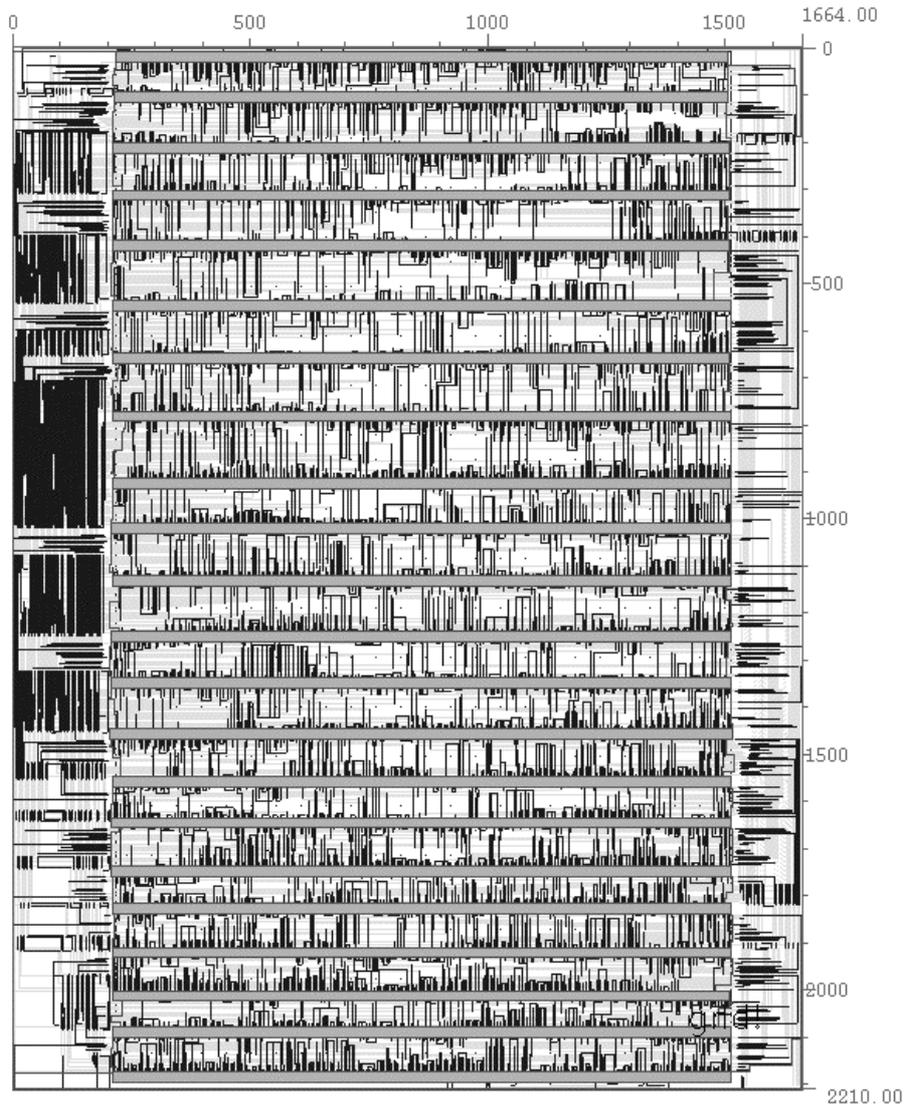


Figure 4.2 : Layout du multiplieur 16x16 en technologie SOI 3D

IV.4. RÉSULTATS ET ANALYSE

"A wise scepticism is the first attribute of a good critic.", Lowell

Les résultats des simulations des multiplieurs sont résumés dans ce paragraphe. Les circuits sont réalisés en cellules standard et le routage est fait automatiquement à l'aide de "Cell Ensemble". Les mesures des caractéristiques électriques (Délai, EPO et EDP) sont obtenues par simulation électrique (HSPICE) des layouts incluant les éléments parasites. Le même ensemble de vecteurs est utilisé dans les deux cas. La valeur de ces résultats est plus comparative qu'absolue car la lenteur de la simulation électrique ne permet pas une approche véritablement statistique et encore moins exhaustive.

Le tableau 4.2 montre les résultats ainsi que le gain obtenu en 3D-SOI par rapport à 2D-SOI. Le gain est calculé comme suit (eq. 4.1) :

$$\text{Gain} = (2D - 3D)/2D \quad (4.1)$$

Tableau 4.2 : Résultats comparatifs pour les multiplieurs

	2D SOI	3D SOI	Gain
Surface (mm ²)	4,21	3,67	12,8 %
Capacité totale (pF)	262,7	188,0	28,4 %
Délai (ns)	12,3	9,6	21,9 %
Énergie/Opération (pJ)	556	387	30,4 %
Énergie x Délai (pJ/MHz)	6,8	3,7	45,6 %

Pour plus de clarté, le tableau 4.3 rappelle le tableau 2.2 du § II.3.2.1. qui résume les résultats obtenus pour une seule cellule : la majorité autoduale.

Tableau 4.3 : Résultats pour la cellule $\overline{\text{Maj}}$

	SOI	SOI sur SOI	Gain
Surface (µm ²)	588	352,8	40 %
Capacité totale (fF)	229,4	194,8	15 %
Délai (ps)	385	276	28,3 %

Le faible gain en surface (12,8 %) pour le multiplieur comparé à un gain de 30 % pour l'ensemble des cellules standard et 40 % pour le cas spécifique de la majorité autoduale est dû au fait que la place occupée par les canaux de routage des cellules est la même en 2D et 3D. Une implantation en "full custom" permettrait une réduction drastique de cette surface dédiée au routage, ainsi le gain pour un circuit complet serait proche du gain obtenu dans le cas d'une cellule.

S'agissant de la capacité totale, le gain est plus important pour un circuit que pour une seule cellule car d'une part, les capacités intrinsèques sont les mêmes dans les deux cas (2D et 3D), et que d'autre part, celles-ci occupent une part relativement plus importante dans une cellule que dans un circuit. Une part importante du gain résulte donc de la réduction de la longueur des interconnexions obtenue par l'intégration tridimensionnelle.

La diminution du délai s'explique par la réduction des capacités; il en est de même pour l'EPO. Ces deux quantités sont des fonctions linéaires de la capacité, elles se trouvent donc améliorées dans les mêmes proportions que celle-ci.

Le produit énergie - délai cumule exactement les gains de délai et d'EPO; en effet :

$$(1 - \text{Gain}_{\text{délai}}) \cdot (1 - \text{Gain}_{\text{EPO}}) = (1 - \text{Gain}_{\text{EDP}}) = (0,781 * 0,694) = 0,544 \quad (4.2)$$

$$\text{D'où} \quad \text{Gain}_{\text{EDP}} = 1 - 0,544 = 0,456 = 45,6\% \quad (4.3)$$

Les améliorations obtenues sont significatives, mais dépendent toutefois de la mesure utilisée. Les sources d'améliorations sont la réduction des interconnexions et l'adaptation des méthodologies de conception à ce type d'intégration (cellules autoduales).

Pour finir, nous pouvons nous demander s'il s'agit réellement d'une intégration tridimensionnelle? En effet, l'intégration à chaque niveau de Silicium est incomplète et toutes les possibilités d'interconnexions des diverses couches ne sont pas disponibles. C'est donc entre 2D et 3D. Pourrait-on l'appeler 2,5D?

IV.5. RÉSUMÉ

Ce chapitre a été consacré à la présentation des résultats et à la comparaison des deux technologies SOI et SOI sur SOI afin d'évaluer les gains apportés par l'intégration multi-niveaux de Silicium.

Les principales sources de gain sont la réduction de la longueur des interconnexions et l'adaptation des méthodologies de conception. L'implantation en cellules standard du multiplieur est loin d'être optimale du point de vue du routage, et une approche "full custom" s'accompagnerait de nettes améliorations des performances.

IV.6. SUMMARY

This chapter was dedicated to the presentation of comparative results for the SOI and SOI on SOI technologies, thus allowing an accurate evaluation of the gains resulting from this type of multi Silicon levels integration.

The main sources of gain are the reduction of the interconnections length and the adapted design methodologies. From the routing point of view, the standard cell implementation of the multiplier is far from optimal, and a full custom approach would have lead to larger performance improvements.

CONCLUSION

"L'idée qu'un texte peut être définitif relève de la religion ou de la fatigue", Borges

Dans le cadre de ce travail nous nous sommes intéressés aux opérateurs arithmétiques à faible consommation d'énergie d'une part, et aux technologies 0,1 μ m CMOS-SOI 2D et 3D d'autre part. Ces technologies sont développées à l'IMS (Institut pour la Microélectronique de Stuttgart - Allemagne) dans le cadre du projet Européen ESPRIT IV HIPERLOGIC (# 20.023) [HPL97, HPL98, http], et les opérateurs arithmétiques ont été choisis comme circuits de démonstration pour ces technologies. Ces sujets se sont insérés dans un thème de recherche plus large concernant les méthodologies de conception pour la faible consommation.

Dans le premier chapitre nous avons passé en revue les différentes composantes de la dissipation d'énergie :

- dissipation dynamique
- dissipation de court-circuit
- dissipation statique

Étant donné que la dissipation dynamique, due aux charge/décharge des capacités lors des commutations est dominante en CMOS, nous avons montré comment réduire le nombre de ces commutations en rendant l'information redondante. Plusieurs modèles de réduction d'échelle ont ensuite été présentés, et leur influence sur la dissipation discutée. Ensuite le problème des métriques de performances des circuits a été abordé. Ces métriques sont :

- le débit
- l'autonomie qui est dans ce cas liée à l'énergie par opération (EPO)
- l'efficacité ou le produit énergie - délai (EDP).

La question de la conception utilisant des cellules complexes fut aussi étudiée. Afin de mesurer l'impact des cellules complexes sur les performances, la complexité a d'abord été quantifiée, et des échelles définies. Ces échelles permettent de comparer les performances de deux versions d'une même technologie (ici SOI et Si massif) dans l'espace performance/complexité.

Le premier chapitre s'est terminé par une étude des collisions entre transitions, et du filtrage des aléas ainsi générés. La notion de seuil de propagation d'un aléa a été introduite.

Au second chapitre nous avons présenté les technologies utilisées ainsi que les outils de conception et les bibliothèques de cellules développés.

Tout d'abord les diverses méthodes d'obtention de substrats de Silicium sur isolant ont été passées en revue, ensuite le procédé de fabrication des transistors EDMOS a été présenté. Dans ce procédé la longueur du canal est de 100nm. Elle est définie indépendamment de la résolution de la lithographie par dépôt et gravure de couches minces.

La partie suivante était consacrée à une technologie spécifique où les transistors de type P et de type N sont empilés en deux couches de Silicium sur isolant. Nous appelons cette technologie SOI sur SOI (ou aussi SOI 3D); son avantage est surtout la faible longueur des interconnexions; mais de nouvelles contraintes de conception sont introduites. De nouvelles méthodologies de conception ont donc été développées ainsi que des outils de conception et des bibliothèques de cellules.

Le chapitre suivant a été consacré aux architectures d'opérateurs arithmétiques combinatoires à faible consommation d'énergie. Les opérations abordées sont l'addition, la multiplication et la division.

Dans le cas de l'addition, nous avons développé un modèle analytique de l'activité pour l'additionneur à propagation de retenue. Ce modèle fut ensuite étendu à d'autres architectures.

Le développement de modèles analytiques pour l'activité dans les multiplieurs nécessite trop d'hypothèses simplificatrices qui rendraient ces prédictions inexploitables. L'approche adoptée ici a été statistique. Nous avons utilisé une méthode Monte Carlo modifiée qui consiste à combiner deux simulateurs :

- l'un logique (VERILOG) donc rapide mais peu précis
- l'autre électrique (HSPICE) très précis mais lent

Le problème d'architectures d'additionneurs rapides adaptés à la multiplication a aussi été abordé.

En ce qui concerne la division, notre choix s'est porté sur un algorithme récurrent de division rapide en base deux. La méthode choisie pour réduire la consommation de ce diviseur consistait à réduire la tension d'alimentation des cellules de queue en dehors du chemin critique en les mettant en série deux par deux entre V_{DD} et la masse. Seules les cellules qui sont suffisamment éloignées du chemin critique peuvent être alimentées de la sorte; leur proportion augmente avec la précision du diviseur. Néanmoins, il s'est avéré que l'impact du choix de la base sur la proportion de cellules pouvant être alimentées en basse tension était déterminant. Une base 4 ou 8 donnerait de bien meilleurs résultats grâce à la plus grande différence de délai qui existerait alors entre les cellules de tête et de queue.

Étant donné que les bits du quotient arrivent dans l'ordre (poids forts d'abord), nous avons suggéré une méthode de synthèse d'additionneurs permettant la conversion du quotient de la notation redondante en notation standard en temps constant. Un algorithme d'optimisation en surface de ces convertisseurs a aussi été proposé.

Enfin, au dernier chapitre nous avons présenté les résultats de la comparaison des deux technologies SOI et SOI sur SOI afin d'évaluer les gains apportés par l'intégration multi-niveaux de Silicium.

Il s'est avéré que les principales sources de gain sont la réduction de la longueur des interconnexions et l'adaptation des méthodologies de conception.

Le véhicule de test choisi pour cette étude comparative est un multiplieur 16x16 bits à base de réducteurs 4:2. L'implantation en cellules standard du multiplieur est loin d'être optimale du point de vue du routage, et une approche "full custom" permettrait de nettes améliorations des performances.



ENGLISH CONCLUSION

This work dealt with low energy consumption combinatorial arithmetic operators on one hand, and with 2D and 3D CMOS SOI technologies on the other hand. These technologies are developed at the IMS (Institute for Microelectronics of Stuttgart - Germany) in the frame of the European project ESPRIT IV HIPERLOGIC (# 20.023) [HPL97, HPL98, [http](#)]. Arithmetic operators were chosen as benchmarks for these technologies. These subjects are part of a wider research topic concerning design methodologies for low energy circuits in general.

The first chapter started by a review of the impact the decisions may have on the dissipation depending on the considered level of abstraction. This enabled to delimit the field of this work. Then, the different components of energy dissipation were explained :

- dynamic dissipation
- short-circuit dissipation
- static dissipation

Given that dynamic dissipation (capacitance charge and discharge) is dominant in static CMOS, it was shown how to reduce signal activity by introducing information redundancy. Scaling schemes were also presented, and their influence on energy dissipation discussed. Then, the different figures of merit were detailed. These are :

- the throughput
- the battery life : which is inversely proportional to the energy per operation here
- the efficiency or the energy - delay product (EDP).

The problem of the use of complex cells was also addressed. For that purpose, complexity scales were defined. These scales allowed quantitative comparisons in the performance/complexity space.

Finally, the first chapter addressed the problems of colliding transitions, glitch filtering and propagation. The concept of “glitch threshold” was also introduced.

The second chapter was dedicated to the technologies that were used as well as to the design tools and cell libraries developed in the frame of this work.

In the first part, an overview of SOI substrates fabrication methods was given, and the EDMOS process was presented. In this process the channel length is 100nm. It is lithography independent, and is obtained by suitable thin films deposition and etching techniques.

The second part was devoted to a specific technology where the N type and P type transistors are stacked in two layers of SOI. The main advantage of this technology is the reduced length of the interconnections. But new design constraints were introduced, and thus new design methodologies and design tools had to be developed.

The next chapter was devoted to low energy consumption arithmetic operators architecture. The operations tackled were addition, multiplication and division.

In the case of addition, an analytical activity model was developed for the ripple carry adder. This model was then extended to other adders.

Developing analytical activity models for multipliers requires too many simplifying hypotheses that would have lead to irrelevant predictions. Thus, the adopted approach was

statistical. An enhanced Monte Carlo method was applied. It consisted in combining two simulators :

- one logical simulator (VERILOG) : fast but not accurate
- and one electrical simulator (HSPICE) : very accurate but slow

The question of fast adders architecture for multiplication was also addressed.

As far as division is concerned, a fast radix-2 recurrent division algorithm was chosen. The energy reduction scheme adopted consisted in reducing the supply voltage of the tail cells outside the critical path. This reduction was achieved by putting them in series two by two between V_{DD} and ground. This method could be applied only to the cells that were sufficiently far from the critical path; their rate increases with the accuracy of the divider. Nevertheless, it was found that the choice of the radix has a dramatic impact on the proportion of cells that could use a low supply. A radix 4 or 8 would have given better results, as the difference in terms of delay between the head and the tail increases with the radix.

As the bits of the quotient are delivered at a fixed rate starting by the MSBs, an adders synthesis method allowing quotient conversion from redundant to standard notation in constant time was presented. Also, an algorithm for the area optimisation of these converters was proposed.

The last chapter presented comparative results for the SOI and SOI on SOI technologies, thus allowing an accurate evaluation of the gains resulting from this type of multi Silicon levels integration.

It was shown that the main source of gain is the reduction of the interconnections length. The standard cell implementation of the multiplier was far from optimal from the routing point of view, and a full custom approach would have lead to larger performance improvements.



ANNEXE : BIBLIOTHÈQUES DE CELLULES

I.1. INTRODUCTION

Dans cette annexe sont détaillées les données concernant les principales cellules des bibliothèques. Les propriétés géométriques, électriques et logiques sont données pour chacune des cellules dans les deux technologies.

I.1.1. TAILLE DES CELLULES

Les cellules standard devant avoir toutes la même largeur, cette dimension est donnée par le pire cas - la cellule la plus grande. Une fois cette contrainte satisfaite, les cellules peuvent être juxtaposées tout en assurant la continuité des lignes d'alimentation.

La longueur des cellules est définie par le nombre de plots d'entrées/sorties. Elle est donnée par la formule suivante :

$$\text{Longueur} = N_{\text{plot}} \cdot (\text{Taille}_{\text{plot}} + \text{Espace}_{\text{plot}}) = N_{\text{plot}} \cdot (2.4 + 1.8) \mu\text{m}$$

La longueur des cellules est la même dans les deux technologies, car elle ne dépend que du nombre de plots. Les contraintes sur le positionnement des plots permettent d'assurer le routage automatique.

I.1.2. ROUTAGE

Le routage est automatique. La couche métal1 est utilisée horizontalement et métal2 verticalement, ce qui implique que les cellules doivent offrir des transparences horizontale à métal1 et verticale à métal2.

I.1.3. TERMINOLOGIE

I.1.3.1. VARIABLES GÉOMÉTRIQUES

- **Longueur** : Fixe pour une cellule donnée; dépend du nombre d'entrées/sorties
- **Largeur** : Fixe pour une bibliothèque donnée
- **W_n** : Largeur de transistors de type N
- **W_p** : Largeur de transistors de type P
- **Nombre** : Nombre total de transistors dans la cellule

I.1.3.2. PARAMÈTRES DYNAMIQUES

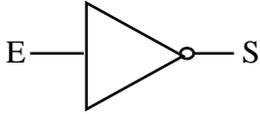
Dans un souci de concision, les paramètres dynamiques sont ici résumés par : t_m , t_d et t_{moy} qui sont respectivement le temps de montée, le temps de descente et le délai moyen de la cellule. Ces paramètres sont en fait des moyennes calculées à partir des résultats collectés pour toutes les transitions possibles.

I.1.3.3. PARAMÈTRES LOGIQUES

C'est tout simplement la table de vérité de la fonction logique implantée.

I.2. INVERSEUR

I.2.1. SYMBOLE

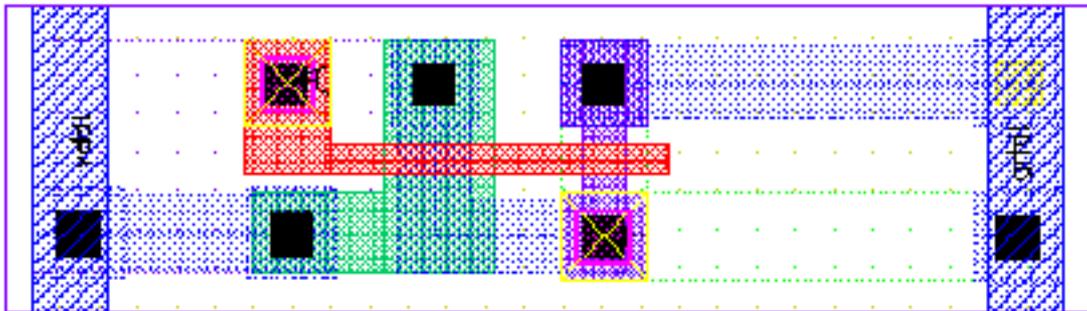


I.2.2. DONNÉES

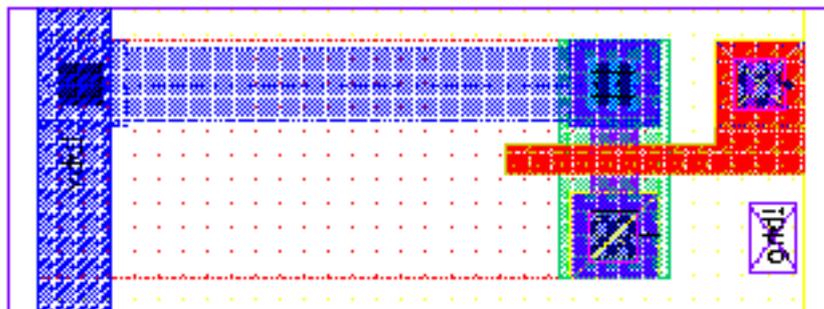
Propriété	Paramètre	Unité	SOI 2D	SOI 3D	Gain
Variables géométriques	Longueur	µm	8,4	8,4	
	Largeur	µm	28	21	
	W_n	µm	1,2	1,2	
	W_p	µm	3	3	
	Nombre	-	2	2	
Paramètres dynamiques	t_m	ps	204	174	14,7%
	t_d	ps	204	168	17,6%
		ps	105	73	30,5%
Table de vérité		E	S		
		0	1		
		1	0		

I.2.3. RÉALISATION

I.2.3.1. 2D



I.2.3.2. 3D



I.3. NON-ET À DEUX ENTRÉES

I.3.1. SYMBOLE



I.3.2. DONNÉES

Propriété	Paramètre	Unité	SOI 2D	SOI 3D	Gain
Variables géométriques	Longueur	µm	12,6	12,6	
	Largeur	µm	28	21	
	W_n	µm	2,4	2,4	
	W_p	µm	3	3	
	Nombre	-	4	4	
Paramètres dynamiques	t_m	ps	253	260	-2,8%
	t_d	ps	233	245	-5,1%
		ps	121	120	0,8%
Table de vérité		A	B	S	
		0	0	1	
		0	1	1	
		1	0	1	
		1	1	0	

I.4. NON-OU À DEUX ENTRÉES

I.4.1. SYMBOLE

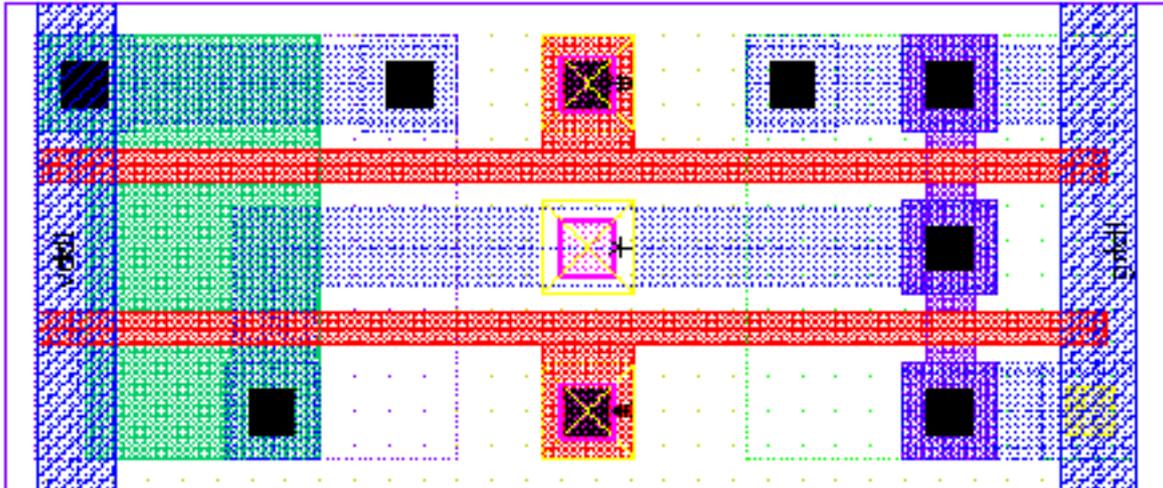


I.4.2. DONNÉES

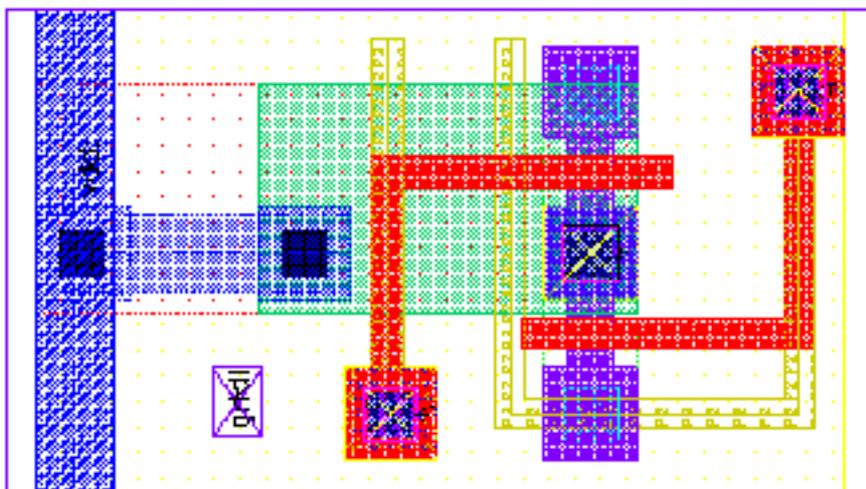
Propriété	Paramètre	Unité	SOI 2D	SOI 3D	Gain
Variables géométriques	Longueur	μm	12,6	12,6	
	Largeur	μm	28	21	
	W_n	μm	1,2	1,2	
	W_p	μm	6	6	
	Nombre	-	4	4	
Paramètres dynamiques	t_m	ps	401	293	26,9%
	t_d	ps	362	257	29,0%
		ps	179	157	12,3%
Table de vérité		A	B	S	
		0	0	1	
		0	1	0	
		1	0	0	
		1	1	0	

I.4.3. RÉALISATION

I.4.3.1. 2D



I.4.3.2. 3D



I.5. ET À DEUX ENTRÉES

I.5.1. SYMBOLE

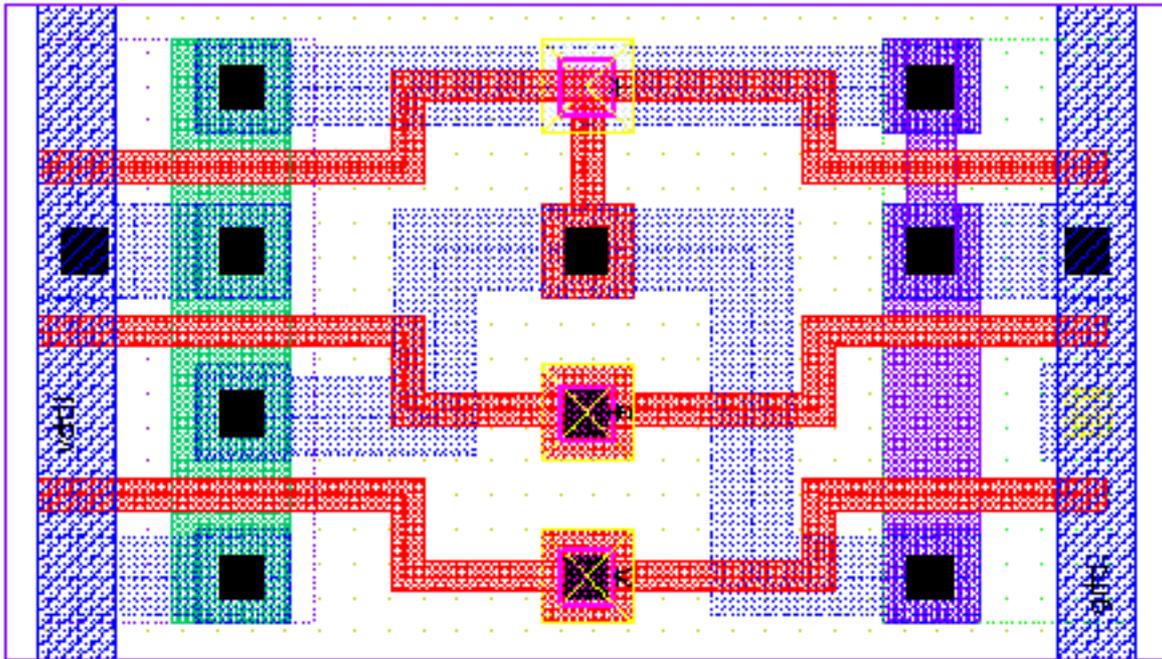


I.5.2. DONNÉES

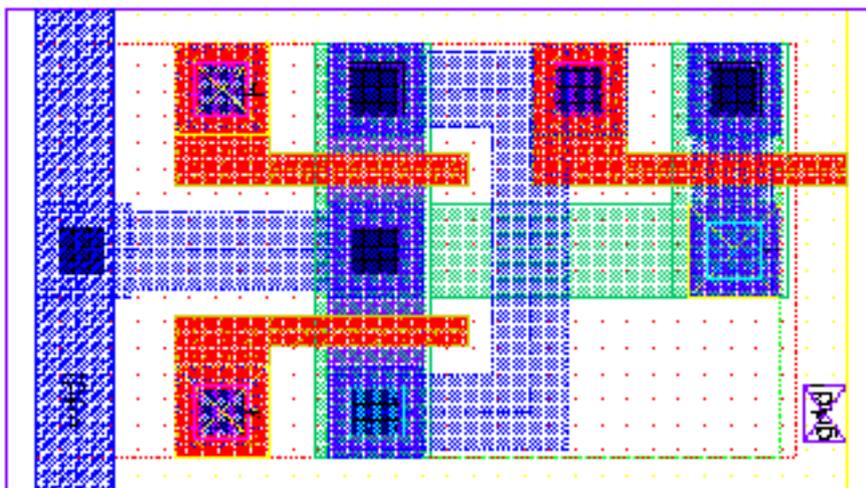
Propriété	Paramètre	Unité	SOI 2D	SOI 3D	Gain
Variables géométriques	Longueur	µm	16,8	16,8	
	Largeur	µm	28	21	
	W_n	µm	1,2/2,4	1,2/2,4	
	W_p	µm	3	3	
	Nombre	-	6	6	
Paramètres dynamiques	t_m	ps	251	245	2,4%
	t_d	ps	249	242	2,8%
		ps	271	269	0,7%
Table de vérité		A	B	S	
		0	0	0	
		0	1	0	
		1	0	0	
		1	1	1	

I.5.3. RÉALISATION

I.5.3.1. 2D

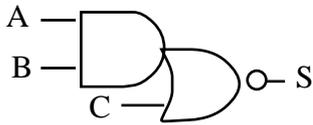


I.5.3.2. 3D



I.6. ET - OU INVERSÉ

I.6.1. SYMBOLE

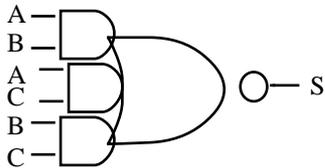


I.6.2. DONNÉES

Propriété	Paramètre	Unité	SOI 2D	SOI 3D	Gain
Variables géométriques	Longueur	μm	16,8	16,8	
	Largeur	μm	28	21	
	W_n	μm	1,2/2,4	1.2/2,4	
	W_p	μm	6	6	
	Nombre	-	6	6	
Paramètres dynamiques	t_m	ps	538	471	12,5%
	t_d	ps	544	497	8,6%
		ps	252	218	13,5%
Table de vérité		AB	C	S	
		00	0	1	
		00	1	0	
		01	0	1	
		01	1	0	
		10	0	1	
		10	1	0	
		11	0	0	
		11	1	0	

I.7. MAJORITÉ INVERSÉE À 3 ENTRÉES

I.7.1. SYMBOLE

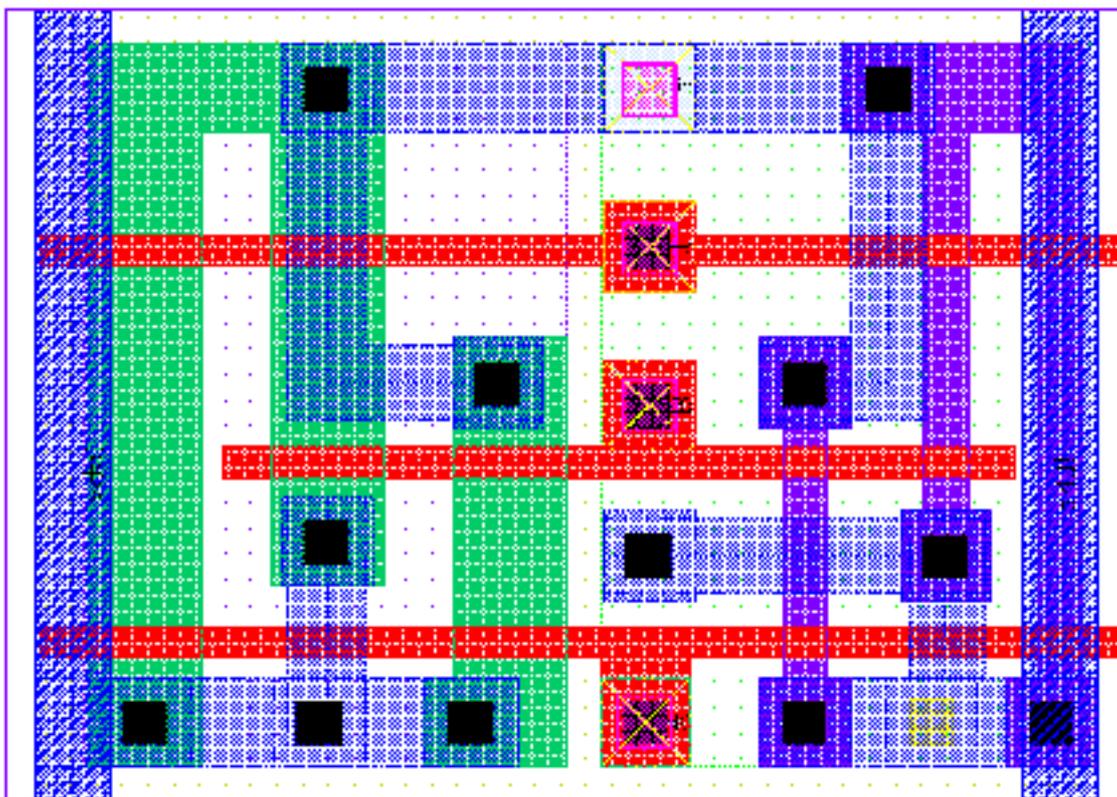


I.7.2. DONNÉES

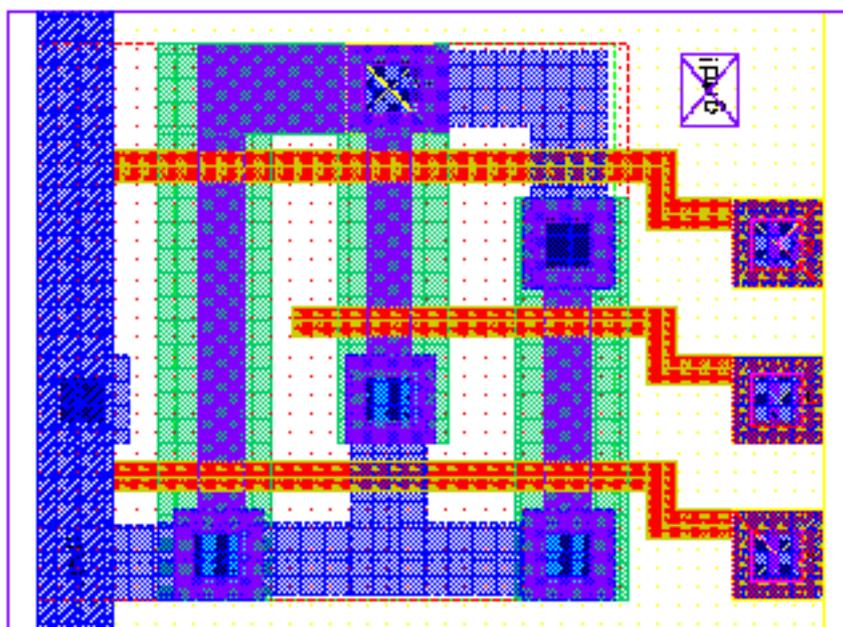
Propriété	Paramètre	Unité	SOI 2D	SOI 3D	Gain
Variables géométriques	Longueur	μm	21	16,8	
	Largeur	μm	28	21	
	W_n	μm	1,2	1,2	
	W_p	μm	3	3	
	Nombre	-	12	12	
Paramètres dynamiques	t_m	ps	994	828	16,7%
	t_d	ps	969	784	19,1%
		ps	425	349	17,9%
Table de vérité		AB	C	S	
		00	0	0	
		00	1	0	
		01	0	0	
		01	1	1	
		10	0	0	
		10	1	1	
		11	0	1	
	11	1	1		

I.7.3. RÉALISATION

I.7.3.1. 2D



I.7.3.2. 3D



BIBLIOGRAPHIE

- [AAG98] S.J. Abou-Samra, J. Arweiler* and A. Guyot, “*Low Power SOI CMOS Multipliers : 2D vs. 3D*”, In proc. of the 24th European Solid State CIRcuits Conference (ESSCIRC'98), The Hague, The Netherlands, September 22-24 1998, pp. 196-199 (* Technische Hochschule Darmstadt, Germany)
- [AAGC98] S.J. Abou-Samra, P. A. Aisa*, A. Guyot and B. Courtois, “*3D CMOS SOI for High Performance Computing*”, In proc. of the 1998 International Symposium on Low-Power Electronics and Design (ISLPED), Monterey, CA, August 10-12, 1998, pp. 54-58 (* University of Bologna, Bologna, Italy)
- [ADT97] D. Auvergne, J. M. Daga and S. Turgis, “*Power and Delay Macro-Modelling for Submicronic CMOS Process : Application to Low Power Design*”, Microelectronics Engineering, Vol. 39, Elsevier, Dec. 1997, pp. 209-233
- [AGcours] Alain Guyot, “*Arithmétique des ordinateurs*”, Cours en 2^{ième} année de l'École Nationale Supérieure de Mathématiques Appliquées de Grenoble (ENSIMAG)
- [ASDA97] S.J. Abou-Samra, V. Dudek, F. Ayache, A. Guyot, B. Courtois and B. Höfflinger, “*Designing With 3D SOI CMOS*”, In proc. of the 8th International Symposium on SOI Technology and Devices (ECS'97), Paris, France, September 1997, pp. 384-388.
- [ASG97] S.J. Abou-Samra and A. Guyot, “*Analytical Modelling of Spurious Transitions in Adder Circuits*”, In proc. of the 7th International Workshop Power and Timing Modeling Optimization and Simulation (PATMOS'97), Louvain la Neuve, Belgium, September 1997, pp. 151-160
- [ASG97b] S.J. Abou-Samra et A. Guyot, “*Glitch Threshold*”, In proc. of the Journées d'études Faible Tension Faible Consommation (FTFC'97), Paris, Nov. 1997
- [ASG98] S.J. Abou-Samra and A. Guyot, “*Performance/Complexity Space Exploration : Bulk vs. SOI*”, In proc. of the 8th International Workshop Power and Timing Modeling Optimization and Simulation (PATMOS'98), Copenhagen, Denmark, October 1998, pp. 307-315
- [ASGL97] S.J. Abou-Samra, A. Guyot and B. Laurent*, “*Spurious Transitions in Adder Circuits: Analytical Modelling and Simulation*”, In proc. of the 9th IFIP International Conference on VLSI (VLSI'97), Gramado, RS, Brazil, August 1997, pp. 377-388 (* CSI-INPG, Grenoble)
- [BDB98] Markus Bühler, D. Dallmann, U. G. Baitinger, “*Switching Activity Analysis Using a Set Theoretical Approach*”, In proc. of the GI/ITG/GME Workshop 1998, March 9-11 1998, Paderborn, Germany, ISBN 3-931466-35-3

- [BDPS98] D. T. Blaauw, A. Dharchoudhury, R. Panda, S. Sirichotiyakul, C. Oh and T. Edwards, “*Emerging Power Management Tools for Processor Design*”, In proc. of the 1998 International Symposium on Low-Power Electronics and Design (ISLPED), Monterey, CA, August 10-12, 1998, pp. 143-148
- [BNYT93] R. Burch, F. Najm, P. Yang, T. Trick : “*A Monte Carlo Approach for Power Estimation*”, IEEE Transactions on VLSI Systems, 1993.
- [Bor97] J. Borel, “*LP/LV circuits : state of the art and prospects*”, Microelectronics Engineering, Vol. 39, Elsevier, Dec. 1997, pp. 1-7
- [Bra63] E. L. Braun, “*Digital Computer Design*”, New-York Academic, 1963
- [BrKu82] R. P. Brent, H. T. Kung, “*A Regular Layout for Parallel Adders*”, IEEE Transactions on Computers, Vol. C31, pp 261 - 264, March 1982.
- [Bru95] M. Bruel, Electronics Letters, VOL. 31, 1995, p. 1201
- [Bru96] M. Bruel, Nuclear Instruments and Methods in Physics Research B, VOL. 108, 1996, p. 313
- [ChBr96] A. Chandarakasan, R. W. Brodersen, “*Low Power Digital CMOS Design*”, Kluwer Academic Publishers, Boston, 1995, ISBN : 0-7923-9576-X
- [Col97] J.-P. Collinge, “*Silicon-On Insulator Technology : Materials to VLSI*”, 2nd Edition, Kluwer, 1997, ISBN : 0-7923-8007-X
- [CPi95] C. Piguet et al., “*Low Power Design of a Standard Cell Library*”, Low-Voltage Low-Power Workshop during ESSCIRC’95, Lille, France, Sept. 22 1995
- [CPi97a] C. Piguet, “*Circuit and Logic Level Design*”, in Low Power Design in Deep Submicron Electronics, ed. by W. Nebel and J. Mermet, Kluwer, 1997, ISBN 0-7923-4569-X, pp. 105-133
- [CPi97b] C. Piguet, “*Low-Power and Low-Voltage CMOS Digital Design*”, Microelectronics Engineering, Vol. 39, Elsevier, Dec. 1997, pp. 179-208
- [Cri95] S. Cristoloveanu and S. S. Li, “*Electrical Characterization of Silicon-On-Insulator Materials and Devices*”, Kluwer, 1995, ISBN : 0-7923-9548-4
- [DABD96] V. Dudek, W. Appel, L. Beer, G. Digele, B. Höfflinger, “*Lithography-Independent Nanometer Silicon MOSFET’s on Insulator*”, IEEE Transactions on Electron Devices ED-43(10), 1996, pp.1626-1632
- [Dad65] L. Dadda, “*Some Schemes for Parallel Multipliers*”, Alta Frequenza, Vol. 34, March 1965, reprinted in E. E. Swartzlander, Vol. 1, IEEE Computer Society Press Tutorial, Los Alamitos, CA, 1990, pp. 349-356

- [Dax97] M. Dax, "*Inspection, Measurement and Test*", Semiconductor International, March 1997, p. 64.
- [Den73] R. H. Dennard et al., Semiconductor Silicon Electrochemical Society, (H. R. Huff and R. R. Burgess, eds.), 1973
- [DHNT95] A.C. Deng, X. Huang, S. Napper, J. Tuan, J. Benkoski, "*Simulation Algorithms, Power Estimation and Diagnostics in PowerMill*", In proc. of the 5th International Workshop Power and Timing Modelling Optimization and Simulation (Patmos), Kaiserslautern - Germany, 1995.
- [ErLa87] M.D. Ercegovac, T. Lang, "*On-the-fly Conversion of Redundant into Conventional Representations*", IEEE Trans. on Computers, Vol.36, no.7, 1987
- [ErLa94] M.D. Ercegovac, T. Lang, "*Division and Square Root, Digit Recurrence Algorithms and Implementations*", Kluwer Academic Publishers, 1994
- [FDCT98] T. J. Fountain, M. J. B. Duff, D. G. Crawley, C. D. Tomlinson and C. D. Moffat, "*The Use of Nanoelectronic Devices in Highly Parallel Computing Systems*", IEEE Tran. on VLSI Systems, VOL. 6, NO. 1, March 1998, pp. 31-38
- [Fur85] S. Furukawa, "*Silicon-On-Insulator Its Technology and Applications*", Kluwer, 1985, ISBN : 90-277-1940-3
- [GAAH97] A. Guyot, S.J. Abou-Samra, M. Aberbour*, A. Houelle*, H. Mehrez* and N. Vaucher*, "*Modelling and Synthesis of Optimal Adders under Left-to-Right Input Arrival*", In proc. of the IFIP International Workshop on Logic and Architecture Synthesis (IWLAS'97), Grenoble, France, December 1997, pp. 23-28 (* LIP6, Université Pierre & Marie Curie, Paris)
- [Gau97] J. Gautier, "*Beyond CMOS : quantum devices*" Microelectronics Engineering, Vol. 39, Elsevier, Dec. 1997, pp. 263-272
- [GaKu83] D. D. Gajski, R. H. Kuhn, "*New VLSI Tools*", IEEE Computer 16, Dec. 1983, pp. 11-14
- [GBB94] A. Guyot, M. Belrhiti, G. Bosco, "*Adders Synthesis*", in proc. of the IFIP Workshop on Logic and Architecture Synthesis, Grenoble, Dec.1994
- [GoZu81] J. B. Gosling, J. H. P. Zurawski, "*Design of High-Speed Digital Divider units*", IEEE Transaction on Computers, Vol. C-30, N°9, September 1981
- [GuAS97] A. Guyot, S.J. Abou-Samra, "*Modelling Power Consumption in Arithmetic Operators*", Low Power, Low Voltage Integrated Circuits: Technology and Design, Microelectronics Engineering, An International Journal of Semiconductor Manufacturing Technology, Vol. 39, Elsevier, Dec. 1997, pp. 245-255

- [HPL97] S.J. Abou-Samra, A. Guyot and B. Courtois, "*HIPERLOGIC First Annual Report*", Technical report for the EC ESPRIT IV Thousand MOPS per Milliwatt High PERFORMANCE LOGIC project (20023), May 1997
- [HPL98] S.J. Abou-Samra, A. Guyot and B. Courtois, "*HIPERLOGIC Second Annual Report*", Technical report for the EC ESPRIT IV Thousand MOPS per Milliwatt High PERFORMANCE LOGIC project (20023), May 1998
- [http] <http://tima-cmp.imag.fr/~selim>
- [HaCa87] T. Han, D. A. Carlson, "*Fast Area-Efficient VLSI Adders*", proc of the 8th Symposium on Computer Arithmetic, pp. 49-56, May 1987
- [HMOV94] A. Houelle, H. Mehrez, N. Vaucher, "*On portable Macro-cell generators using the fully 754-IEEE standard*", In proc. of the ICSPAT 94, Dallas, USA, October 1994.
- [IHOW97] T. Ikeda, H. Higuchi, Y. Ohkura and S. Wakahara, "*Time Constant of Switching-off Current in Partially Depleted SOI MOSFETs with Body Tie*", In proc. of the 8th International Symposium on SOI Technology and Devices (ECS'97), Paris, France, September 1997, pp. 251-258
- [IwMo97] H. Iwai and H. S. Momose, "*Technology towards low power / low voltage and scaling of MOSFETs*", Microelectronics Engineering, Vol. 39, Elsevier, Dec. 1997, pp. 7-30
- [JM2] J. M. Muller, "*Arithmétique des Ordinateurs, Opérateurs et Fonctions élémentaires*", Masson, Paris, 1989, ISBN : 2-225-81689-1
- [KoSt73] P. M. Kogge and H. S. Stone, "*A Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations*", IEEE Trans. Comput., vol. 22, no. 8, pp. 783-791, Aug. 1973.
- [Kre93] E. Kreyszig, "*Advanced Engineering Mathematics*", WILEY, New York, 1993
- [LeTo97] C. S. Lent and P. D. Tougaw, "*A Device architecture for computing with quantum dots*", in proc. of IEEE, VOL. 85, Apr. 97, pp. 541-557
- [LMJ95] J. Leijteen, J. van Meerbergen, J. Jess, "*Analysis and Reduction of Glitches in Synchronous Networks*", in proc. of the ED&TC, Paris, March 1995.
- [Mas91] J.-M. Masgonty et al., "*Technology and Power-Supply -Independent Cell-Library*", IEEE CICC'91, San Diego, CA., USA, May 12-15 1991
- [MeRe79] C. Mead, M. Renn, "*Cost and Performance of VLSI Computing Structures*", IEEE JSSC-14, April 1979, pp. 455-462

- [Mon95] L. A. Montalvo, “*Systèmes de Numération pour la Conception de Diviseurs Rapides*”, Thèse, INPG, Mars 1995
- [MoPa96] L. Montalvo and K.K. Parhi, “*Estimation of Average Energy Consumption of Ripple-Carry Adder Based on Average Length Carry Chains*”, in proc. 11th Design of Integrated Circuits and Systems Conference (DCIS’96), Barcelona, Spain, November 1996.
- [Naj92] F. Najm, “*Transition Density, A New Measure of Activity in Digital Circuits*”, IEEE Transactions on Computer Aided Design, 1992
- [Pap84] A. Papoulis, “*Probability, Random Variables, and Stochastic Processes*”, McGraw-Hill, 2nd ed., New York, 1984
- [PeGl91] N. Permingeat et D. Glaude, “*Algèbre de Boole, Théorie, Méthodes de calcul, Applications*”, Masson, Paris, 1991, ISBN : 2-225-81503-8
- [PFSK95] M. Pelella, J.G. Fossum, D. Suh, S. Krishnan and K.A. Jenkins, in proc. of the IEEE Intl. SOI Conference, 1995, p. 8
- [RaPe97] J. M. Rabaey and M. Pedram, “*Low Power Design Methodologies*”, Kluwer Academic Publishers, Boston, 1997
- [ROB58] J.E Robertson, “*A new class of digital division methods*”, IEEE Tran. Electronic Computers. Vol. EC-7, pp218-222 Sept 1958.
- [Rou97] S. Roux, “*Etude et Implementation d’un Diviseur Rapide Basse Consommation*”, rapport de DEA, DEA Microélectronique, UJF, 1997
- [SaPa96] J. Satyanarayana, K. Prahi, “*HEAT : Hierarchical Energy Analysis Tool*”, In proc. of the DAC, 1996, pp. 9-14
- [SaNe91] T. Sakurai, R. Newton, “*A Simple MOSFET Model for Circuit Analysis*”, IEEE Transactions on Electron Devices, VOL. 38 NO. 4, APRIL 1991, p. 887.
- [Sho76] W. Shockley, “*The path to the conception of the junction transistor*”, IEEE Trans. on Electron Devices, VOL. 23, NO. 7, July 1976, pp. 597
- [SIA97] Semiconductor Industry Association, “*The National Technology Roadmap for Semiconductors Technology Needs*”, 1997
- [Skl60] J. Sklansky, “*Conditional Sum Addition Logic*”, IRE Transaction EC-9(2), June 1960, pp 226-231.
- [Sze81] S. Sze, “*Semiconductor Physics and Devices*”, John Wiley & Sons, 1981
- [StBu95] M. Stan and W. Burleson, “*Bus-Invert Coding for Low Power I/O*”, IEEE Trans on VLSI system, March 1995.

- [StPr96] Bernd Stöhr, Roland Preussmann, “*Ein graphischer Editor zur Beschreibung von komplexen Steuerwerken*”, 2. Workshop Boolesche Probleme, 19-20 September 1996, Freiberg/Sachsen
- [TOC58] K.D. Tocher, “*Techniques of multiplication and division for automatic binary computers*”, Quart. J. Mech. Appl. Math., Vol. 11, pt. 3, pp 364-384, 1958.
- [ToLe96] P. D. Tougaw and C. S. Lent, “*Dynamic behaviour of quantum cellular automata*”, J. Appl. phys., VOL. 80, NO. 8, 1996, pp. 4722-4736
- [TVG95] V.Tchoumatchenko, T. Vassileva and A. Guyot, “*Timing Modelling for Adders Optimisation*”, In proc. of the 5th International Workshop Power and Timing Modeling Optimization and Simulation (PATMOS), Germany, 1995.
- [Vee84] H. J. M. Veendrick, “*Short-Circuit Dissipation of Static CMOS Circuitry and Its Impact on the Design of Buffer Circuits*”, IEEE Journal of Solid-State Circuits, Vol. SC-19, pp. 468-473, August 1984
- [WeEs94] N.H.E Weste, K. Eshraghian, “*Principles of CMOS Design*”, 2nd ed., Addison-Wesley Publishing Compagny, 1994
- [Zim96] R. Zimmermann, “*Non-Heuristic Optimization and Synthesis of Parallel Prefix Adders*”, In proc. of the International Workshop on Logic and Architecture Synthesis (IWLAS), Grenoble, Dec. 1996

RÉSUMÉ en français

Dans le cadre du présent travail nous nous sommes d'abord intéressés aux causes de la dissipation d'énergie dans les circuits intégrés ainsi qu'aux métriques associées à la mesure des performances. Ensuite les technologies utilisées ont été présentées; il s'agit des technologies bidimensionnelle et tridimensionnelle SOI 100nm grille en **T**. La version tridimensionnelle est composée de deux couches de transistors tel que le type P soit au dessus du type N. Des méthodologies de conception ainsi que des bibliothèques de cellules standard ont dû être développées pour ces technologies. Finalement, des architectures pour l'arithmétique combinatoire faible consommation ont été modélisées, évaluées et dessinées avec les technologies citées précédemment. Les opérations abordées sont l'addition, la multiplication et la division.

TITRE en anglais

LOW ENERGY INTEGRATED CIRCUIT DESIGN IN 2D AND 3D SOI TECHNOLOGIES : APPLIED TO ARITHMETICS

RÉSUMÉ en anglais

This work discussed first the causes of energy dissipation in integrated circuits and presented the performances metrics associated with this dissipation. Then were described the technologies that were used; these are the two dimensional and three dimensional **T** gate 100nm SOI technologies. The three dimensional technology consists in two layers of transistors where the P-channel devices are integrated on top of the N-channel ones. Design methodologies and standard-cell libraries were developed. Finally, low energy combinatorial arithmetic operators architectures were evaluated, modelled and implemented in these technologies. The target operations were addition, multiplication and division.

DISCIPLINE

MICROÉLECTRONIQUE

MOTS-CLÉS

Faible consommation. CMOS. SOI. SOI 3D. Silicium sur isolant.

Arithmétique combinatoire. Additionneur. Multiplieur. Diviseur. Bibliothèque de cellules.

INTITULÉ ET ADRESSE DU LABORATOIRE :

Laboratoire TIMA

46, avenue Félix Viallet

F38031 Grenoble cedex

<http://tima-cmp.imag.fr/>

ISBN 2-913329-14-4 Format papier

ISBN 2-913329-15-2 Format électronique