



**HAL**  
open science

## Analyse d'architectures de multiplieurs en vue de la basse consommation

Mariem Slimani, Philippe Matherat

► **To cite this version:**

Mariem Slimani, Philippe Matherat. Analyse d'architectures de multiplieurs en vue de la basse consommation. Journées Nationales du Réseau Doctoral en Microélectronique, Jun 2010, Montpellier, France. hal-00491095

**HAL Id: hal-00491095**

**<https://hal.science/hal-00491095>**

Submitted on 11 Jun 2010

**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.

# Analyse d'architectures de multiplieurs en vue de la basse consommation

Mariem Slimani, Philippe Matherat  
 CNRS UMR 5141, LTCI  
 Institut Télécom - Télécom ParisTech - COMELEC  
 346, rue Barrault F-75634 Paris Cedex 13, France

E-mail: (mariem.slimani, philippe.matherat)@telecom-paristech.fr

## Résumé

Ce papier présente une étude des performances de différentes architectures de multiplieurs. Cette étude comporte, outre une comparaison des multiplieurs combinatoires entre eux et celles séquentiels entre eux, une comparaison combinatoire/séquentielle rarement présentée dans les travaux précédents. Les résultats d'analyse sont obtenus par l'outil de simulation Encounter du MentorGraphics. L'accent est mis sur les performances en matière de consommation.

## 1. Introduction

Devant l'émergence des systèmes de plus en plus complexes qui demandent des calculs performants à des cadences de plus en plus rapides, la consommation est devenu un facteur indispensable à prendre en compte dans la conception des circuits VLSI. La dissipation est égale à la consommation. C'est un facteur limitant la vitesse des opérations dans les systèmes de traitement du signal numérique. Le multiplieur a représenté et représente jusqu'à présent le cœur de plusieurs travaux de recherches dont certaines cherchent à trouver des nouvelles architectures plus performantes [1] et d'autres à optimiser celles déjà existantes [2]. Dans ce papier, une analyse des performances de différentes architectures de multiplieurs est présentée. Il y a plusieurs architectures de multiplieurs dont on distingue deux classes : les multiplieurs combinatoires et les multiplieurs séquentiels. A la différence d'autres travaux qui ne comparent que les performances d'architectures combinatoires ou celles séquentielles, le présent travail présente une étude combinatoire/séquentielle. Cette étude n'était pas possible en rassemblant les résultats des autres travaux puisque ils travaillent sur des technologies et des tensions d'alimentation différentes. Trois architectures combinatoires et trois architectures séquentielles seront présentés et implémentés dans ce papier en vue de comparer leurs performances.

## 2. Architectures de multiplieurs standard

Le principe de multiplication de 2 nombres binaires de  $n$  bits,  $A$  et  $B$ , est illustré dans la fig.1 avec  $n = 4$ . Cette

opération se résume en une somme de produits partiels dont chacun d'eux est constitué du multiplicande décalé d'un nombre de positions égales au poids du bit du multiplicateur, si ce dernier n'est pas nul.

			$A_3$	$A_2$	$A_1$	$A_0$	Multiplicande $A$
		$\times$	$B_3$	$B_2$	$B_1$	$B_0$	Multiplicateur $B$
			$A_3B_0$	$A_2B_0$	$A_1B_0$	$A_0B_0$	Produit partiel $A \times B_0$
			$A_3B_1$	$A_2B_1$	$A_1B_1$	$A_0B_1$	$A \times B_1$ décalé de 1 rang
			$A_3B_2$	$A_2B_2$	$A_1B_2$	$A_0B_2$	$A \times B_2$ décalé de 2 rangs
			$A_3B_3$	$A_2B_3$	$A_1B_3$	$A_0B_3$	$A \times B_3$ décalé de 3 rangs
$P_7$	$P_6$	$P_5$	$P_4$	$P_3$	$P_2$	$P_1$	$P_0$
							Produit $P = A \times B$

FIG. 1. Principe de multiplication de 2 nombres binaires.

## 2.1. Les Multiplieurs Combinatoires

### 2.1.1. Multiplieur à réseau cellulaire (Array Multiplier).

La fig.2 présente un multiplieur à réseau cellulaire de 2 nombres à 4 bits non signés,  $A$  et  $B$ . Il est constitué d'un ensemble de cellules d'additionneurs (FA) placés suivant un réseau d'interconnexion régulier. Chaque étage ajoute un nouveau produit partiel à l'accumulateur, dont le premier bit calculé est mis à la sortie et les bits restants se décalent à gauche pour être calculés avec l'étage suivant. Le temps de calcul d'un tel réseau est proportionnel au plus long chemin du graphe. En général, pour un réseau de taille  $n$ , ce temps de calcul est égal au temps de traversée de  $2n-2$  cellules élémentaires [3].

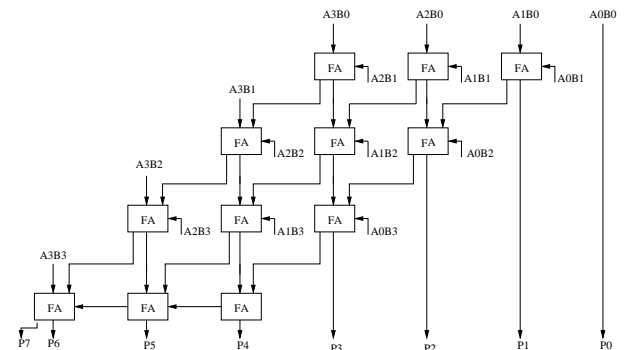


FIG. 2. Multiplieur à Réseau Cellulaire

### 2.1.2. Multiplieur de Wallace.

Le principe du multiplieur de Wallace (MW) de 2 nombres de 4 bits est illustré dans la fig.3. Dans cette architecture, tous les bits de tous les produits partiels dans chaque colonne sont additionnés ensemble. Chaque étage est formé donc par un ensemble de compteurs en parallèles qui réduit la matrice obtenue par l'étage précédent jusqu'à ce qu'on obtient une matrice à 2 lignes [4](figure 3(d)). Le produit final est le résultat de l'addition de ces deux lignes.

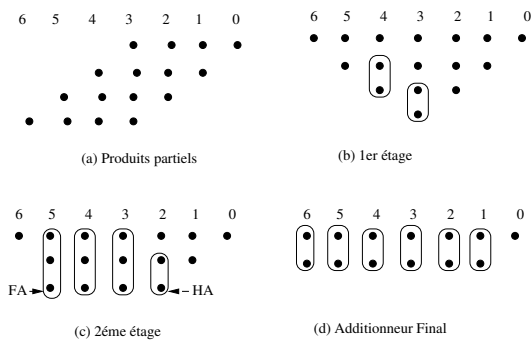


FIG. 3. Principe du Multiplieur de Wallace

### 2.1.3. Multiplieur CSM "Carry Save Multiplier".

Le multiplieur CSM est constitué d'une cellule élémentaire appelée "CSA (Carry Save Adder)" illustrée dans la fig.4(a). Cette cellule prend trois entrées binaires A0,A1,A2 et fournit deux sorties également binaires S (en référence à la somme) et C (en référence à la retenue (carry))[3]. Le multiplieur CSM consiste à sommer les vecteurs  $W_i$ , qui n'est autre que le  $i^{\text{ème}}$  produit partiel (fig.1) [5].

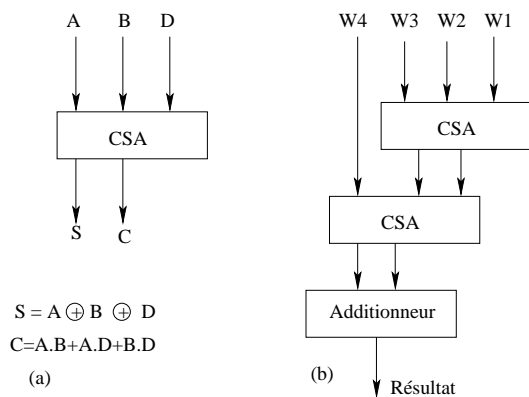


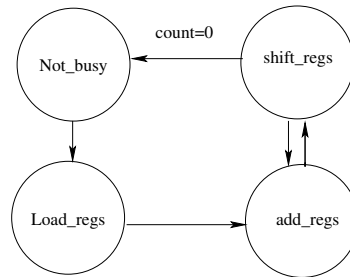
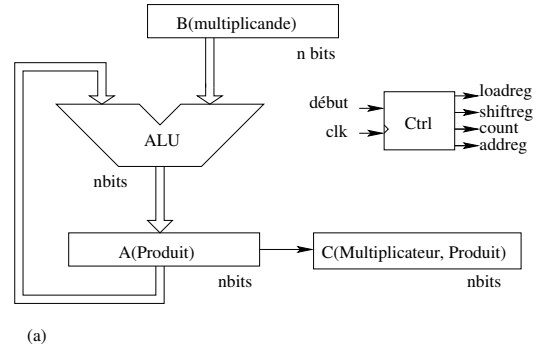
FIG. 4. Architecture du multiplieur CSM

## 2.2. Les multiplieurs séquentiels

Les multiplieurs séquentiels sont tous basés sur l'algorithme de multiplication par addition décalage inspiré par la méthode de résolution à la main ou on additionne le multiplicande A à lui même B fois tel que B représente le multiplicateur. Dans ce qui suit, on va présenter 3 versions de ces multiplieurs.

### 2.2.1. Multiplieur à Séquenceur.

La structure interne du Multiplieur à Séquenceur (MS) est présentée dans la fig.5 (a). Elle est constituée de 3 registres à décalages, d'un additionneur et d'un contrôleur qui contient la machine d'états qui va séquencer l'exécution de l'algorithme. Cette machine est décrite dans la fig.5 (b). Pour effectuer une multiplication n bits, ce multiplieur a besoin de  $2n+1$  cycles d'horloge.



(b)

FIG. 5. Architecture du multiplieur à Séquenceur

### 2.2.2. Multiplieur de ROBERTSON.

Le principe de ce multiplieur consiste à tester successivement chaque bit du multiplicateur. Si ce dernier est égal à 1, le multiplicande est additionné à la somme cumulée dans le produit. La première version de ce multiplieur est présentée dans la fig.6. Ici, le registre du multiplicande se décale à gauche à chaque étape afin d'aligner le multiplicande avec le produit, le registre du multiplicande doit être donc de taille  $2n$ . Le décalage à droite du multiplicateur prépare le bit suivant pour examiner l'itération suivante. Ce multiplieur effectue une opération de multiplication à chaque  $n+1$  cycles d'horloges.

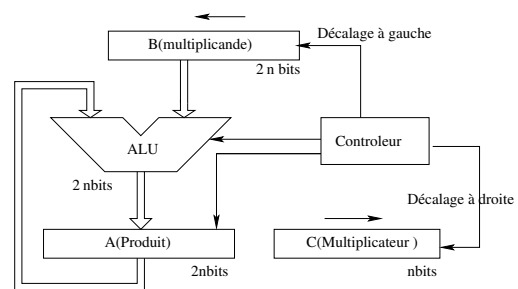


FIG. 6. Architecture du multiplieur de Robertson(1<sup>ère</sup> version)

La deuxième version de ce multiplieur est illustrée dans la fig.7. Dans cette architecture, au lieu de décaler la multiplicande à gauche, le produit est décalé à droite. Ainsi, on a besoin juste d'un additionneur n bits qui ajoute la multiplicande à la moitié gauche du produit. Au début de la multiplication, la moitié droite du produit est chargé par le multiplieur. Cette version nécessite aussi n+1 cycles d'horloges pour effectuer une opération.

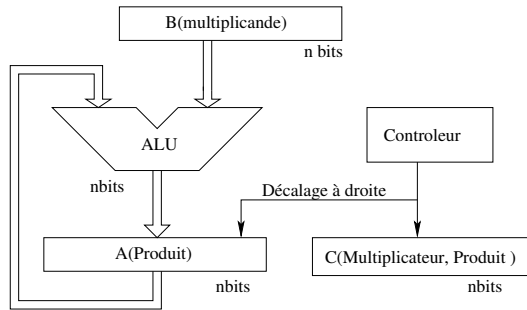


FIG. 7. Architecture du multiplieur de Robertson(2<sup>ème</sup> version)

### 3. Présentation et analyse des résultats

#### 3.1. Détails d'implémentation

Les multiplieurs décrits précédemment sont implémentés en suivant le diagramme de conception présenté ci-dessous. Ils sont synthésés en utilisant la bibliothèque standard de STmicroelectronics en technologie 65nm avec une tension d'alimentation  $V_{dd} = 1.2V$ .

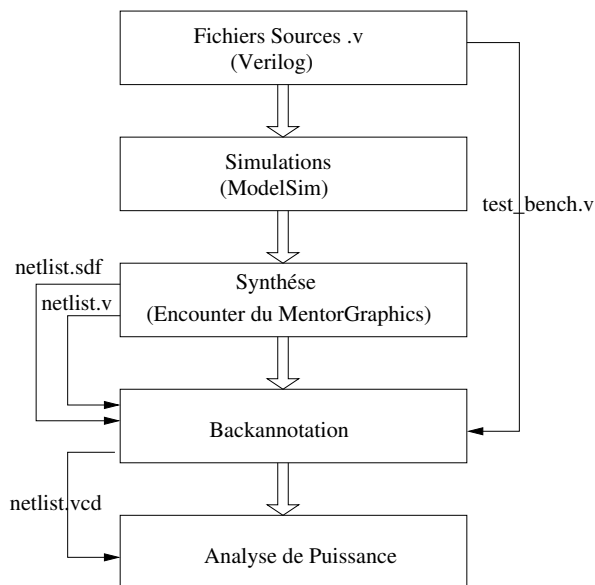


FIG. 8. Diagramme de conception

### 3.2. Résultats de simulations

Les différents résultats de simulation sont présentés dans le tableau ci-dessous dans lequel F désigne la fréquence d'horloge,  $F(\text{mul/s})$  correspond à la fréquence des multiplications, E est l'énergie consommée par une multiplication et  $E/E1$  est le rapport de l'énergie avec une énergie de référence qui correspond dans ce cas à la consommation la plus petite de tous les multiplieurs.

Algorithme	Combinatoires			Séquentiels		
	MRC	MW	CSM	MS	MR1	MR2
Surface( $\mu m^2$ )	661.4	657.28	866.8	804.4	940.16	648.4
$P_T(\mu w)$	564.2	602.5	1252.5	382.9	506.6	223.8
$F(\text{mul/s})$	F	F	F	F/18	F/9	F/9
E(fJ)	564.2	602.5	1252.5	6893.6	4560.1	2014.2
E/E1	1	1.06	2.21	12.21	8.08	3.569

TAB. 1. Résultats de simulation

Au vue de ces résultats, nous remarquons que pour les multiplieurs combinatoires, il n'y a pas une grande différence entre le multiplieur à réseau cellulaire MRC et le multiplieur de Wallace MW. En effet, le multiplieur MRC présente une meilleure consommation d'énergie en dépit d'une légère augmentation de surface par rapport au MW et inversement. Ainsi, pour une application à basse consommation, le multiplieur MRC est proposé. Pour les multiplieurs séquentiels, nous pouvons constater une augmentation de la consommation d'énergie d'un facteur de 4 dans le meilleur cas. Le multiplieur MR2 est le meilleur multiplieur séquentiel. Il présente une consommation d'énergie 3 fois inférieure aux deux autres et une occupation surfacique optimale par rapport à tous les multiplieurs. Ceci se justifie par le fait qu'il utilise moins de ressources matériels ( Registres..).

Pour une étude combinatoire/séquentiel, la fig. 9 et la fig. 10 présentent respectivement la consommation d'énergie et la surface occupée en fonction du nombre de bits pour le meilleur multiplieur combinatoire (MRC) et celui séquentiel (MR2).

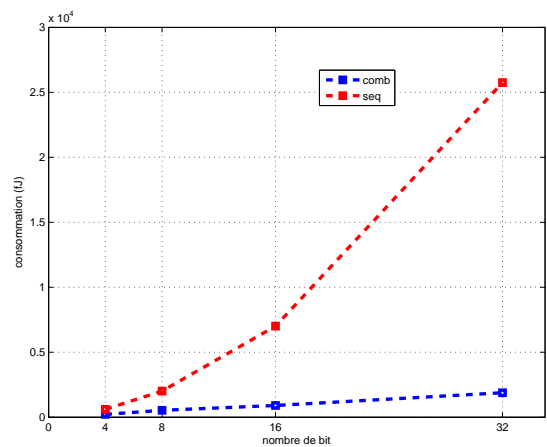


FIG. 9. Variation de la consommation d'Énergie en fonction du nombre de bits

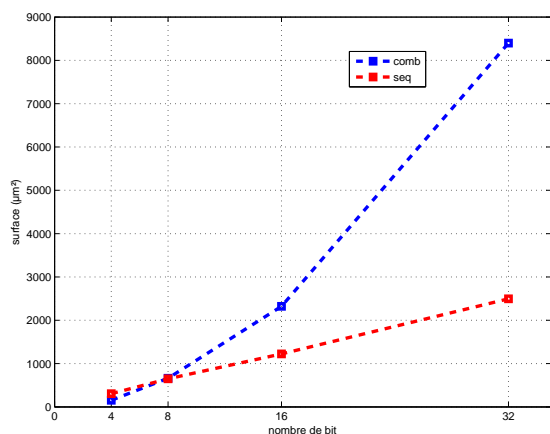


FIG. 10. Variation de la surface occupée en fonction du nombre de bits

Nous remarquons que pour un nombre de bits inférieur ou égale à 8, les multiplieurs séquentiels n'ont aucun intérêt puisque le multiplieur combinatoire présente la même surface avec une consommation 4 fois moins. D'après la figure présentée ci-dessous, on remarque que plus le nombre de bits augmente, plus la surface du combinatoire augmente, mais plus également la consommation du séquentiel augmente. Le choix entre les deux dépend des contraintes et des spécifications de l'application. Pour des modules complexes qui peuvent intégrer des modules séquentiels et d'autres combinatoire, nous proposons de faire un partitionnement combinatoire/séquentiel à base d'algorithmes d'optimisation permettant de trouver la solution optimale (quel module en séquentiel et quel module en combinatoire) en vue de satisfaire les contraintes de l'application.

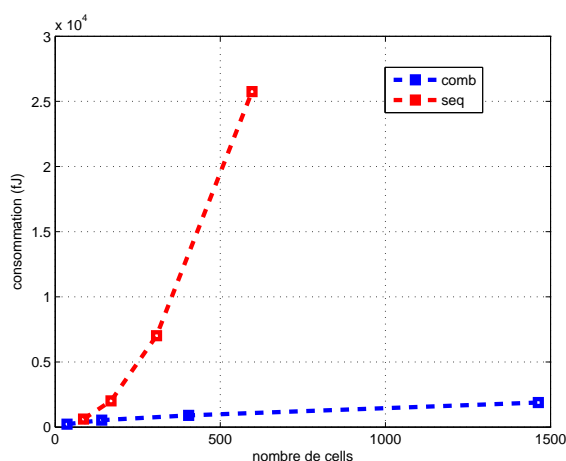


FIG. 11. Comparaison des performances Séquentiel/Combinatoire

## 4. Conclusion

Cet article présente une étude de performances de différentes architectures de multiplieurs. Les résultats de simulations montrent que les multiplieurs séquentiels consomment beaucoup plus que les multiplieurs combinatoires alors qu'ils présentent une surface beaucoup plus petite. Ainsi, le choix de l'un ou de l'autre de ces multiplieurs dépend essentiellement des contraintes (surfaces, consommation) de l'application.

En considérant que l'utilisation des modules séquentiels est une première étape vers des implémentations de plus en plus logicielles, on peut associer une dissipation plus élevée à une implémentation plus logicielle. C'est la recherche d'une formulation plus quantitative de cette remarque qui guide notre motivation en cherchant à l'étendre à des architectures plus variées.

## Références

- [1] P. Ramanathan P. thangapandian and S. Agrawal. High speed multiplier design using decomposition logic. *Serbian journal of electrical engineering*, 6(1) :33–42, May 2009.
- [2] P.Cyril P.V. Rao and S.Ravi. Vlsi design and analysis of multipliers for low power. *Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2009.
- [3] J.-M. Muller. *Arithmétique des ordinateurs*. Masson, Paris, 1989.
- [4] P.S.Patil P.G.Parate and S.Subbaraman. Asic implementation of 4 bit multipliers. *First International Conference on Emerging Trends in Engineering and Technology*, 2008.
- [5] N.Masmoudi L.SAMET, M.W.KHARRAT and L.KAMOUN. Intégration de multiplieurs arithmétiques : Architectures, simulations et comparaison. *2 ème conférence Maghrébine d'Automatique d'Electrotechnique et d'Electronique Industrielle*.