



HAL
open science

Etude des méthodes de conception et des outils de CAO pour la synthèse des circuits intégrés analogiques

F. Chaahoub

► **To cite this version:**

F. Chaahoub. Etude des méthodes de conception et des outils de CAO pour la synthèse des circuits intégrés analogiques. Micro et nanotechnologies/Microélectronique. Institut National Polytechnique de Grenoble - INPG, 1999. Français. NNT: . tel-00163766

HAL Id: tel-00163766

<https://theses.hal.science/tel-00163766>

Submitted on 18 Jul 2007

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

ISBN 2-913329-37-3 (broché)

ISBN 2-913329-38-1 (électr)

N° attribué par la bibliothèque

//////////

THESE

pour obtenir le grade de

DOCTEUR de L'INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Discipline: Microélectronique

présentée et soutenue publiquement

par

Faouzi CHAAHOUB

Le 29 Septembre 1999

Titre :

=====

**ETUDE DES METHODES DE CONCEPTION ET DES
OUTILS DE C.A.O. POUR LA SYNTHESE DES CIRCUITS
INTEGRES ANALOGIQUES**

=====

Directeur de thèse: M. Alain GUYOT

JURY :

Président: Pr. Guy MAZARE (ENSIMAG - INPG)
Rapporteur: Pr. Daniel AUVERGNE (LIRMM Montpellier)
Pr. Michel DECLERCQ (EPFL Lausanne)
Examineur: M. Louis-Olivier DONZELLE (CNET)
M. Christian CORBEX (CNET)
M. Alain GUYOT (TIMA - INPG)
Invité: M. Bernd WUPPERMANN (Rockwell-Nice)

Thèse préparée au CNET Grenoble en collaboration avec le laboratoire TIMA/INPG

Remerciements

Le travail présenté dans cette thèse a été effectué au Centre National d'Etudes des Télécommunications (CNET) Grenoble, et plus précisément au sein du département Méthodes, Modélisation, Outils et Support (MOS) qui est rattaché au groupement Circuits Intégrés pour les Télécommunication (CIT). A cet égard, je remercie Messieurs Patrice SENN et Jacques LECOURVOISIER de m'avoir accepté dans leur département respectifs.

Tous mes remerciements à Monsieur le Professeur Guy MAZARE Directeur de l'ENSIMAG en tant que président du jury, ainsi qu'à Monsieur Daniel AUVERGNE, Professeur de l'université de Montpellier (LIRMM) qui fut mon professeur, et Monsieur Michel DECLERCQ, Professeur de l'université de Lausanne (EPFL), pour l'intérêt qu'ils portent à mon travail, et pour avoir bien voulu accepté d'être les rapporteurs de cette thèse.

Je tiens à exprimer ma reconnaissance à Monsieur Alain GUYOT, maître de conférences à l'ENSIMAG et responsable de l'équipe "Conception de Systèmes Intégrés" au laboratoire TIMA/INPG d'avoir accepté d'être mon directeur de thèse universitaire. Son expérience de la recherche, ses conseils et ces encouragements sont toujours très précieux.

Je voudrais également remercier Monsieur le Professeur Pierre Gentil, Responsable de la Formation Doctoral à l'INPG de ces encouragements qui mon permis de persister et de mener à bien cette thèse.

Je tiens tout particulièrement à exprimer ma profonde gratitude et mes remerciements les plus sincère à Louis-Olivier DONZELLE, ingénieur au CNET et Chef du projet CHIRVAN, qui a encadré ce travail, ainsi qu'à toute l'équipe CHIRVAN; Anne GERODOLLE et Christian CORBEX, leur participation dans le projet, leurs encouragements et leur disponibilité tout au long de ces années de recherche ont joué un rôle déterminant dans l'obtention des résultats présentés ici. Qu'ils trouvent ici l'expression de ma profonde reconnaissance et de mon amitié.

Je tiens également à remercier toute l'équipe de conception avec laquelle j'ai également travaillé, tout particulièrement Messieurs Patrice SENN, Bernard HENNION, Freddy BALESTRO, André ABRIAL... Leurs disponibilités et conseils tout au long de ces années de recherches ont aussi joué un rôle déterminant dans l'obtention des résultats présentés dans cette thèse. Qu'ils trouvent ici l'expression de ma profonde reconnaissance.

PREMIERE PARTIE

*Conception des Circuits Intégrés
Analogiques à Basse Tension d'Alimentation
et à Faible Consommation*

Introduction Générale

INTRODUCTION GENERALE

Le domaine de la micro-électronique a connu un développement technologique sans pareil ces dernières années. Ainsi sont aujourd'hui intégrés sur une seule puce des systèmes électroniques réalisés auparavant sous forme de cartes. Cette tendance à l'intégration et à la miniaturisation des circuits est portée par le développement "explosif" des applications multimédia, de télécommunications et automobiles. De tels systèmes comportent un nombre toujours croissant de modules pouvant appartenir à des domaines différents: des fonctions numériques, prédominantes, qui sont fondées sur des microprocesseurs ou micro-contrôleurs, des mémoires et des blocs DSP de traitement de signal ("Digital Signal Processing"), mais aussi des fonctions analogiques d'amplification et de filtrage qui se trouvent en particulier dans les circuits de conversion analogique/numérique (CAN) en entrée, et numérique/analogique (CNA) en sortie.

Durant ces dernières années, beaucoup d'efforts ont été consacrés à la réduction de la tension d'alimentation et à la réduction de la consommation des systèmes CMOS mixtes analogiques/numériques. Cela est dû principalement à la croissance de l'utilisation des systèmes portables alimentés par des batteries, mais provient aussi de la réduction de la taille des circuits intégrés. Les conséquences de ces réductions sont différentes selon la nature des circuits.

D'un côté, les circuits intégrés numériques à basse tension d'alimentation et à faible consommation, peuvent facilement atteindre des qualités excellentes du point de vue fonctionnement (par exemple une très grande résolution et un bon rapport signal sur bruit [1]). De plus, la taille de la partie numérique est réduite radicalement avec la minimisation des tailles des blocs fonctionnelles.

D'un autre côté, les circuits intégrés analogiques à basse tension d'alimentation et à faible consommation avec de bonnes fonctionnalités sont très difficiles à réaliser. Par exemple, la dynamique de sortie d'un amplificateur opérationnel diminue substantiellement en réduisant la tension d'alimentation. Par ailleurs, les circuits intégrés analogiques ne peuvent pas être conçus en utilisant des tailles minimales des transistors, pour des raisons de gain de l'amplificateur, offset, bruit, etc. La surface de la partie analogique ne peut donc pas être réduite radicalement avec la minimisation des tailles des fonctions à réaliser.

Bien que plusieurs parties analogiques puissent être remplacées par des parties numériques, la nécessité des circuits analogiques reste assez importante. Le monde réel étant analogique, il y aura en effet toujours besoin de convertisseurs pour convertir les données numériques des dispositifs électroniques en signal analogique (ex: signal audio) perceptible par l'homme. D'autre part, les circuits intégrés analogiques peuvent réaliser des fonctions à haute fréquence (Radio Fréquence) contrairement aux circuits numériques. Les senseurs, les transmetteurs, les récepteurs sont ainsi souvent des composants analogiques. Par conséquent, les circuits

analogiques qui ne peuvent pas être remplacés par des circuits numériques, posent un problème majeur pour la conception des systèmes à basse tension d'alimentation et à faible consommation.

L'objectif de cette thèse est tout d'abord de synthétiser les travaux effectués afin d'établir un bilan des acquis dans ce domaine, puis, à partir de cette synthèse, de dégager les limitations de la réalisation des circuits analogiques tant au niveau de la conception des circuits intégrés analogiques à basse tension d'alimentation et faible consommation, qu'au niveau de l'automatisation du dessin des masques des cellules analogiques ("layout"), et de proposer de nouvelles solutions.

Depuis dix ans, la conception des fonctions numériques a été fortement automatisée par le développement d'outils de conception assistée par ordinateur (CAO) très avancés. Citons par exemple, l'apparition de logiciels de synthèse qui permettent de générer le layout d'un circuit intégré constitué de portes logiques à partir d'une simple description de sa fonction. Or, la conception de circuits intégrés mixtes où les blocs analogiques et numériques ne peuvent pas être dissociés pour une étude précise du fonctionnement, reste particulièrement difficile, essentiellement du fait d'un manque d'outils semblables du côté analogique. Une transposition vers l'analogique des outils et des méthodologies qui ont fait le succès du numérique doit donc être effectué. En particulier, un des points clés à résoudre concerne l'automatisation de la phase du dessin du masque, le layout (Génération, Placement, Routage et Compaction).

La conception de cellules analogiques performantes est une tâche importante dans la réalisation des circuits et des systèmes de Télécommunications. Ces cellules (amplificateurs opérationnels, comparateurs, références de tension) constituent en effet les parties essentielles des convertisseurs analogiques/numériques ou numériques/analogiques, ainsi que des filtres à capacités commutées. Leur mise au point demeure un travail délicat, tant par le nombre important de spécifications à prendre en compte, que par la multitude de degrés de liberté électriques (largeurs et longueurs de grille de chaque transistor en particulier) ou physiques (implémentation des transistors, trace des interconnexions). Les paramètres de la technologie utilisée, ainsi que leur variation possible, interviennent également de manière beaucoup plus subtile qu'en conception de circuits numériques, pour garantir le fonctionnement attendu.

La conception assistée par ordinateur (CAO) des circuits VLSI était jusqu'à présent consacrée au domaine des circuits numériques, pour des raisons économiques évidentes. Les analogiciens ne disposent encore souvent que de simulateurs électriques (Hspice puis Eldo ou Saber, Spectre...) pour vérifier le comportement d'un circuit, et d'éditeurs de masques permettant de manipuler des rectangles élémentaires. La fin des années 80 et le début des années 90 a coïncidé avec l'apparition de nouveaux besoins en circuits analogiques, lié entre autres à l'introduction ou à l'expérimentation de nouveaux systèmes de transmission du son et de l'image (RNIS, GSM, CDMA, TVHD...). Ce phénomène, allié à l'introduction de technologies

et dispositifs permettant de réaliser des systèmes mixtes numériques/analogiques sur un seul circuit, a entraîné le besoin de nouveaux outils de CAO pour permettre une réaction plus rapide à des normes et des technologies encore souvent en évolution. Si certains vendeurs de logiciels (Cadence et Mentor-Graphic) annoncent des produits encore à venir dans ce domaine, force est de constater qu'actuellement l'essentiel des développements connus et reconnus est conduit par des universitaires (Berkeley, Carnegie Mellon, CSEM, Leuven, EPFL) ou de manière captive par des industriels (ATT, Philips, Rockwell...).

Cette thèse est divisée en deux parties. La première partie débute par un rappel des notions de base et par les considérations à prendre en compte lors de la conception des circuits intégrés analogiques à basse tension d'alimentation et à faible consommation (chapitre 1). Puis, elle se consacre à la conception des circuits intégrés analogiques, et plus précisément à l'étude de nouvelles techniques (méthodes ou structures) de conception des circuits intégrés analogiques à basse tension d'alimentation et à faible consommation (chapitre 2).

La deuxième partie s'intéresse à l'automatisation de toute la phase de layout des cellules analogiques et mixtes: Chirvan. L'étude qui a conduit à la réalisation de Chirvan, a commencé à la fin de l'année 1988. Il s'agissait de réaliser un logiciel permettant d'automatiser ou du moins d'accélérer le dessin des masques d'une cellule analogique. Nous présentons dans ce rapport notre contribution à l'automatisation du dessin des masques, c'est-à-dire l'étude des contraintes analogiques à respecter lors de la phase du dessin de masques (chapitre 3), et l'étude de l'implémentation d'une nouvelle méthode (algorithme) pour l'automatisation du placement qui respecte ces contraintes analogiques (chapitre 5). Enfin nous présentons l'environnement d'automatisation du dessin des masques qui a été développé au CNET Grenoble. Cet environnement est composé d'outils qui permettent d'automatiser la phase de layout des cellules analogiques et mixtes (Générateur, Routeur et Compacteur), et qui prennent en compte toutes les contraintes analogiques (chapitre 4).

Chapitre I

Chapitre 1

Considérations sur la Conception des Circuits Intégrés Analogiques à Basse Tension d’Alimentation et à Faible Consommation

1.1. INTRODUCTION

La contrainte de conception à faible consommation de puissance est restée pendant longtemps une contrainte mineure dans la conception des circuits intégrés, contrairement à d’autres performances comme la vitesse d’exécution ou la dynamique de sortie (DR dynamic range), considérés comme les buts primordiaux à atteindre dans la conception des circuits intégrés.

Cependant, durant ces dernières années, l’augmentation du nombre des équipements portables utilisant des batteries, dans le domaine des ordinateurs comme dans celui des systèmes de communication, et l’augmentation du nombre des blocs fonctionnels réalisés sur le même circuit intégré, ont forcé à réduire la consommation de puissance.

Dans la technologie moderne des circuits VLSI, la réduction de la consommation est souvent acquise par la réduction des tensions d’alimentation [2][20]. La réduction des tensions d’alimentation ne permet pourtant pas nécessairement la réduction de puissance de consommation dans le cas des circuits intégrés analogiques. Néanmoins, c’est en général vrai pour les circuits intégrés mixtes, où la majorité du circuit est numérique. La réduction de l’augmentation de la densité de puissance et la réduction du nombre des batteries employées incitent également à utiliser des tensions d’alimentation basses.

Dans les circuits intégrés analogiques, la réduction des tensions d’alimentation induit d’importantes modifications dans la topologie de ces circuits. Car la plupart des paramètres de

performance des circuits intégrés analogiques, comme la dynamique de sortie, la linéarité, le gain et la vitesse d'exécution, sont fortement influencés par la réduction de la tension d'alimentation dans un sens non-désirable.

Durant ces dernières années, de nombreuses recherches [2][12][16][21][20][22][23][24][34][35] ont été consacrés à résoudre les problèmes générés par des tensions d'alimentations basses, tant au niveau systèmes qu'au niveau des techniques de conception des circuits ou de la modélisation des composants.

Ce chapitre présentera d'abord les justifications de la réalisation des circuits intégrés dans les conditions de basses tensions d'alimentations et de faible consommation. Nous examinerons ensuite l'impact de ces conditions sur la conception des circuits mixtes numériques/analogiques et nous ferons état des limitations possibles, fondamentales ou pratiques. Pour comprendre l'effet de la réduction des tensions d'alimentations sur les composants de base (MOS,...), nous présenterons brièvement le modèle de transistor MOS, ses éléments parasites, ses différentes régions d'opération et ses caractéristiques électriques dans chaque région. Nos conclusions seront exposées au paragraphe 1.10.

1.2. JUSTIFICATION DE LA CONCEPTION DES CIRCUITS INTEGRES CMOS A BASSE TENSION D'ALIMENTATION ET A FAIBLE CONSOMMATION

Pourquoi est-il nécessaire de concevoir des circuits intégrés pour un fonctionnement à basse tension, et comment cette exigence est liée à la dissipation de puissance? les motivations résident en des facteurs liés à l'application de ces circuits ou concernant la demande du marché.

Étant donné que la technologie se dirige vers des tailles de caractéristiques en sous-microniques, l'épaisseur de l'oxyde des composants MOS décroît en conséquence. Pour éviter des ruptures de jonctions p-n, l'intensité du champ électrique doit être réduite. Comme la longueur du canal est réduite à des dimensions sous-microniques et que l'épaisseur grille-oxide n'est plus que de quelques nanomètres, la tension d'alimentation doit être réduite pour assurer la fiabilité des composants. Par exemple, un processus standard à 0.8 μ m peut résister à une tension d'alimentation maximum subie de environ 5volts, mais les composants conçus à l'aide d'une méthode CMOS à 0.5 μ m ne peuvent tolérer qu'environ 3.5volts. (ils peuvent cependant résister à des transitions de tensions plus importantes). Il est possible de développer un procédé de tailles caractéristiques de 0.5 μ m ou moins qui résistera à une alimentation de 5V, en accroissant le dopage du substrat. On obtient ainsi des régions de déplétion plus minces autour des diffusions du drain et de la source, et le risque de pincement du canal est tempéré. Si en

restreinte ensuite la taille caractéristique, la rupture de la jonction devient dominante et la tension d'alimentation doit être réduite. Cet aspect du problème apparaît clairement à travers le besoin de concevoir des systèmes capables d'opérer à des tensions d'alimentations plus basses que la tension typique de 5V. Il est anticipé dans un futur proche des dimensions très faibles (0.35 μm ; 0.25 μm ; 0.18 μm), et la tension d'alimentation maximale se réduira de ce qui est aujourd'hui 5V, à 3V et probablement au-dessous de cette tension d'alimentation, jusqu'à 1.8V et même 1.5 V [3][4].

En réalité, cette réduction de la tension d'alimentation est non seulement nécessaire mais aussi désirable. En conception de circuits numériques avec un niveau de commutation élevé, la puissance dissipée est essentiellement une puissance dynamique; Elle est associée à la charge ou décharge des capacités. Cette expression de la dissipation de puissance dynamique prend la forme générale suivante:

$$P = p.C.V^2.f_s \quad (1.1)$$

où p est le facteur d'activité (dépendant des traitements du signal) et f_s est la fréquence de commutation. Dans la plupart des circuits numériques, les tensions représentant les états "haut" et "bas" coïncident avec les rails d'alimentation de puissance, c'est-à-dire, V_{DD} et la masse. Par conséquent, le facteur V est égal à V_{DD} . Étant donné que la puissance est proportionnelle au carré de facteur de tension V , abaisser ce facteur de 5V à 3V induit une réduction de la dissipation de puissance par un facteur de 2.8 environ. Ce résultat est important, non seulement en terme de baisse de la dissipation de puissance total, mais aussi en terme de *densité* de la dissipation de puissance qui, comme processus sous-micronique permet l'intégration d'un nombre important de fonctions. Pour des raisons de fiabilité, la densité thermique doit être gardée faible. L'un des effets positifs de cette utilisation de tailles caractéristiques plus petites est que, les capacités parasites commutées contribuant à C sont plus petites, et par conséquent la dissipation de puissance l'est aussi.

Il pourrait sembler que la dissipation de puissance puisse être réduite à des valeurs tolérables, en réduisant simplement la tension d'alimentation. En pratique, des facteurs variés empêchent ce phénomène, ou au moins compliquent les échanges. Par exemple, une valeur abaissée de la tension d'alimentation a un impact sur le délai des cellules et par conséquent, réduit la vitesse de l'opération. Cette perte en vitesse peut être compensé si les tensions seuils sont aussi restreintes, ou si des architectures parallèles ou tuyautées sont utilisées [5][6]. L'équation (1.1) suggère cependant que la puissance totale peut être minimisé par d'autres moyens, par exemple en minimisant l'activité et les capacités parasites commutées. On peut réduire l'activité en divisant les algorithmes (ce qui requiert plusieurs cycles de calcul), et en

utilisant des blocs conducteurs d'événements, désactivés quand ils ne sont pas employés [7][8]. On peut réduire la capacité en rendant les éléments du processus numérique (architecture, cellules logiques, dessin de layout) aussi simples et petits que possible [5], et en utilisant la logique basée sur les branches [9][10]. Néanmoins, abaisser la valeur de la tension d'alimentation est encore le moyen le plus efficace de réduire la dissipation de la puissance dans les circuits numériques.

La question de l'optimisation de la puissance dissipée dans les circuits numériques dépasse le cadre de ce travail. Elle fournit cependant un éclairage nouveau sur l'une des motivations les plus importantes pour concevoir les circuits intégrés à basses tensions d'alimentation et à faible consommation. Que devient alors la dissipation des circuits analogiques ? Nous tenterons de répondre à cette question dans le paragraphe 1.4.

L'augmentation de la densité des composants sur la puce justifie aussi la conception des circuits intégrés à basse tension d'alimentation et à faible consommation. La puce de silicium ne peut dissiper qu'une quantité limitée de puissance par unité de surface. Puisque l'augmentation de la densité des composants permet l'implantation de plus de fonctions électroniques par unité de surface, la puissance consommée par fonction électronique doit se réduire pour éviter des problèmes de surchauffe.

La dernière raison est liée au fonctionnement des batteries d'alimentation des systèmes portables des équipements. Si l'on veut obtenir une fréquence d'opération acceptable pour une batterie, la tension d'alimentation et la puissance d'alimentation doivent être réduites.

1.3. IMPACT DE LA REDUCTION DE LA TENSION D'ALIMENTATION SUR LA CONCEPTION DES CIRCUITS INTEGRES MIXTE:

Réduire la tension d'alimentation exige une philosophie de conception différente et une connaissance de limitation de la technologie assez profonde. Les implications sont particulièrement remarquables dans la conception des circuits intégrés analogiques, où une variété des contraintes pourrait aboutir à un compromis plutôt qu'à un objectif voulu.

En réduisant les tensions d'alimentation, les tensions de références internes se réduisent. L'amplitude maximale du signal d'entrée est donc limitée. Pour maintenir un rapport signal-sur-bruit (SNR) élevé ou une dynamique de sortie (DR dynamic range) large, le bruit total doit aussi être restreint. Ceci est souvent achevé au détriment de la surface totale du silicium (larges capacités, composants à surface très large, et nécessité éventuelle d'ajout de circuits supplémentaires), ce qui ne mène pas forcément à une réduction de consommation de la puissance. La basse tension d'alimentation se traduit aussi directement par une réduction de la marge de variation du signal; les gains peuvent être écaillés pour limiter l'excursion du signal dans la région où le gain des amplificateurs est élevé.

Nous allons donc analyser en détail toutes les conséquences de la réduction des tensions d'alimentation dans les circuits et les systèmes mixtes, pour faire état des limitations fondamentales et pratiques.

1.4. LES LIMITATIONS DE LA CONCEPTION DES CIRCUITS INTEGRES ANALOGIQUES A BASSE TENSION D'ALIMENTATION ET A FAIBLE CONSOMMATION

1.4.1 Limite fondamentale

Dans les circuits intégrés analogiques, pour pouvoir obtenir un rapport signal-sur-bruit (SNR) acceptable, l'énergie du signal doit être supérieure à l'énergie thermique (bruit thermique), et fixe ainsi une limite absolue.

Cette condition peut être exprimée par la consommation de puissance nécessaire pour réaliser un système à 1 pole [12] [13]. Cette limite est définie par l'expression suivante:

$$P_{\min} = 8.f.k.T.SNR \quad (1.2)$$

où f est la fréquence d'opération (la bande passant requise) du signal, SNR est le rapport signal sur bruit, k désigne la constante de Boltzmann ($k=1.381 \cdot 10^{-23} \text{ J}^\circ\text{K}$) et T désigne la température de la résistance en $^\circ\text{K}$.

Cette limite ne dépend pas de la technologie. Elle est atteinte dans le cas d'un simple filtre passif RC, tandis que le meilleur filtre actif existant est encore deux fois d'ordre de magnitude au-dessus. Dans les circuits intégrés analogiques, la consommation de puissance est proportionnelle à la fréquence et au rapport signal-sur-bruit (SNR).

La consommation de puissance minimale est aussi proportionnelle au rapport de la tension d'alimentation et à l'amplitude du signal. Les circuits analogiques efficaces en terme de consommation doivent donc être conçus pour maximiser la dynamique de sortie. L'amplitude du signal en tension doit pouvoir osciller d'une extrémité de la tension d'alimentation à l'autre ("rail-to-rail") [14][15][16].

1.4.2 Les limites d'implémentation réelle des circuits intégrés analogiques et mixtes

Dans la pratique, d'autres limitations rentre en jeu. Ces limitations peuvent être des limitations technologiques, des limitations dues au choix de la tension d'alimentation des circuits, ou encore aux concepteurs eux-mêmes. Leur méthodologie de conception peut en effet

s'appuyer sur l'utilisation de cellules classiques (amplificateurs, comparateurs,...) inadaptées aux basses tensions et à la faible consommation. La liste suivante présente quelques exemples de ces éléments limitant, mais elle n'est pas exhaustive [13].

La première limitation pratique des circuits intégrés analogiques et la présence des capacités, En général, les capacités augmentent la consommation de la puissance nécessaire pour atteindre la bande passante requise. Le besoin de précision mène souvent à l'utilisation de dimension large pour les composants, et aboutit à une augmentation des capacités parasites (comme nous montrerons dans le paragraphe 1.6), et donc une augmentation consommation de la puissance. Ces capacités sont acceptables seulement si leur présence font réduire la puissance de bruit par la même quantité.

Quand les capacités de charges sont imposées (par exemple par les capacités parasites), le courant I nécessaire pour obtenir une bande passante donnée est inversement proportionnel au rapport de la transconductance sur le courant (g_m/I) du composant actif. Pour des valeurs petites du rapport (g_m/I) le transistor MOS opérant dans la région à forte inversion peut donc produire une augmentation en consommation de puissance.

Il est évident que les tensions de grille-source et de saturation d'un transistor MOS, et par conséquent la tension d'alimentation minimum d'un circuit, dépendent de paramètres spécifiques de conception, tels que les tensions de seuil et les niveaux de polarisation. Pour obtenir la tension d'alimentation la plus basse possible, on doit polariser les transistors MOS en régime de faible inversion, étant donné que ceci induit la plus petite tension de grille-source possible pour un transistor donné. Cependant, des fréquences relativement élevées ou des applications à haute vitesse de balayage (slew-rate) requièrent des transistors polarisés en forte inversion plutôt qu'en faible inversion. Ceci augmente la tension de grille-source du composant, et donc la tension d'alimentation minimum.

La présence d'autre source de bruit implique une augmentation dans la consommation de puissance. Ceci inclue le bruit en fréquence ($1/f$) (paragraphe 1.6.3.2) dans les composants et le bruit venant des alimentations ou généré sur la puce par d'autres blocs de ce circuit.

La puissance consommée dans les circuits d'alimentation (circuit de génération des tensions ou des courants de référence) est en général considérée comme une perte de puissance, elle doit être en principe minimisé. Cependant des montages de génération de référence inadaptés peuvent augmenter le bruit et donc nécessitent une augmentation proportionnelle en puissance. Par exemple, le courant de référence est plus bruyant s'il est obtenu en multipliant un faible courant.

Les moyens pour réduire les effets de toutes ces limitations peuvent être trouvés à tous les niveaux de la conception des circuits analogiques, du composant au système en passant par la nécessité de trouver d'autres techniques de conception plus intelligentes qui permettent de prendre en compte les contraintes de basses tensions d'alimentation et faible consommation.

1.5. CLASSIFICATION DES CIRCUITS CMOS A BASSE TENSION D'ALIMENTATION

Dans les systèmes de conception d'aujourd'hui, le terme basse tension est utilisé pour les circuits qui sont capables de fonctionner à des tensions d'alimentation comprises entre 1 et 5 volts. Ces basses tensions d'alimentation induisent le nombre maximum de tension grille-source et de tension de saturation qui peuvent être assemblés en série. Cependant, la tension d'alimentation en elle-même n'induit aucune information sur la topologie du circuit requise. Par exemple, la conception d'un amplificateur à 3 volts dans un procédé de tension-seuil d'environ 1 volt, permet la mise en série d'environ deux tensions de grille-source, alors que celle d'un amplificateur à 3 volts utilisant un procédé ayant une faible tension-seuil de 0,5 volts permet de mettre en série environ cinq tensions grille-source. Par conséquent, afin de pouvoir cataloguer les topologies des différents circuits, on doit faire une classification des basses tensions en termes de tension grille-source et de tension de saturation.

Dans le but de permettre au concepteur de prédire la faisabilité d'une application d'un circuit donné, ce travail donne la relation qui existe entre la basse tension d'alimentation et le nombre de tensions de grille-source et de tensions de saturation mises en série. Ici, le terme basse tension est utilisé pour des circuits capables d'opérer avec une tension d'alimentation égale à la mise en série de deux tensions grille-source et de deux tensions de saturation. Dès lors, on a la relation suivante:

$$v_{a \text{ lim, min}} = 2(v_{gs} + v_{dsat}) \quad (1.3)$$

où V_{gs} et V_{dsat} sont, respectivement, la tension de grille-source et la tension de saturation d'un transistor MOS.

Les circuits ne nécessitant qu'un minimum de tension d'alimentation, c'est à dire une tension égale à la somme d'une tension de grille-source et d'une tension de saturation, seront considérés comme circuits à extrêmement basse tension. Ce qui peut s'exprimer ainsi:

$$v_{a \text{ lim, min}} = v_{gs} + v_{dsat} \quad (1.4)$$

Il est important de remarquer que les circuits à extrêmement basse tension requièrent une tension d'alimentation minimum qui soit environ égale à la moitié de la tension d'alimentation nécessaire pour les circuits à basse tension.

1.6. PROPRIETES ELECTRIQUES DES TRANSISTORS MOS

Une des propriétés les plus importantes que possède le transistor MOS, quand il s'agit de concevoir un amplificateur à basse tension, est la tension grille-source, car elle détermine la tension d'alimentation minimum à laquelle l'amplificateur est capable d'opérer. La transconductance est associée à cette tension grille-source. Étant donné que le transistor MOS est un composant dépendant de la tension, la transconductance nécessaire détermine la tension grille-source du transistor. Dans ce paragraphe, nous traiterons de la tension grille-source et de la transconductance d'un transistor MOS. Par la suite, nous passerons en revue les propriétés d'un composant opérant en forte et faible inversion.

1.6.1. Notations utilisées

- C_{ox} : capacité d'oxyde par unité de surface
- i_d : courant drain-source du transistor MOS
- L : longueur du transistor MOS
- λ : paramètre de modulation de la longueur du canal
qui peut être considéré comme l'inverse de la tension d'Early
(par analogie au transistor bipolaire)
- μ_o : mobilité des porteurs
- ϕ_p : constante caractéristique du substrat
- V_g, V_s, V_d, V_b : Potentiels de grille, de source, de drain, de substrat
- V_{gs} : différence de potentiel entre grille et source
- V_{ds} : différence de potentiel entre drain et source
- V_{bs} : différence de potentiel entre substrat et source
- V_{tn}, V_{tp} : tension de seuil du transistor NMOS, PMOS
- W : largeur du transistor MOS

1.6.2. Modèle petits signaux

Pour définir le modèle petits signaux du transistor MOS, on utilise une linéarisation des équations du transistor autour de son point de polarisation. Le modèle dégagé dans ce paragraphe concerne la zone dite de saturation du transistor, plus fréquemment utilisée. Les composants petits signaux peuvent s'exprimer ainsi :

$$i_d = g_m v_{gs} + g_{ds} v_{ds} + g_{mb} v_{sb} \quad (1.5)$$

où $g_m = \left(\frac{\delta I_d}{\delta V_{gs}} \right)$, $g_{ds} = \left(\frac{\delta I_d}{\delta V_{ds}} \right) \equiv \frac{1}{r_{ds}}$, $g_{mb} = \left(\frac{\delta I_d}{\delta V_{sb}} \right)$

En dérivant l'expression du courant i_d , on trouve :

$$g_m = 2\sqrt{k'(1 + \lambda V_{ds})I_d} \approx 2\sqrt{k'I_d} \quad (1.6)$$

$$g_{mb} = \frac{-\gamma g_m / 2}{\sqrt{2\phi_p + V_{SB}}} \quad (1.7)$$

$$g_{ds} = \frac{\lambda}{1 + \lambda V_{DS}} I_D \approx \lambda I_D \quad (1.8)$$

Le paramètre k' est donné par l'expression suivante :

$$k' = \frac{\mu_n C_{ox}}{2} \frac{W}{L} \quad (1.9)$$

γ est un paramètre du composant donné par la relation : $\gamma = \frac{\sqrt{2\epsilon_s q N_{imp}}}{C_{ox}}$

L'autre partie importante du schéma en petits signaux concerne les capacités parasites. La figure 1.1 permet de localiser et de comprendre chacune des capacités parasites du transistor.

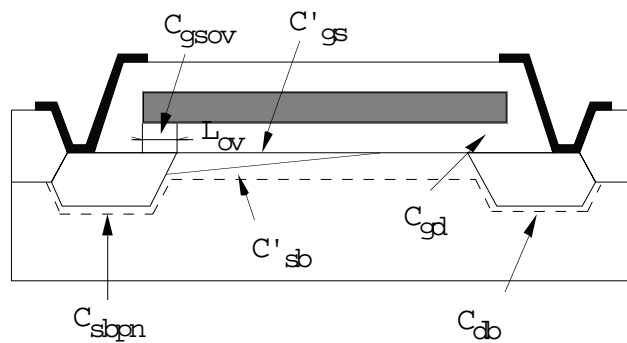


Fig. 1.1 : Capacités parasites du transistor MOS.

On distingue les capacités intrinsèques du transistor qui dépendent des dimensions du transistor (capacités de jonction,...) et les capacités extrinsèques qui dépendent du dessin du transistor (capacités de recouvrement,...). Les capacités les plus importantes sont les suivantes :

C_{gd} : capacité grille-drain : due au recouvrement du drain par la grille, elle est donc constante.

C_{gs} : capacité grille-source: elle comporte deux composantes et peut être considérée constante:

- C_{gsov} : capacité de recouvrement de la source par la grille,

- C'_{gs} : capacité grille-canal. Elle vaut environ $2/3C_{ox}$.

C_{sb} : capacité source-substrat : elle comporte deux composantes et varie comme une jonction pn:

- C_{sbpn} : capacité de jonction entre la diffusion de source et le substrat,
- C'_{sb} : 2/3 de la capacité de la région déplétée sous le canal.

C_{db} : capacité drain-substrat : capacité de jonction variable.

C_{gb} : capacité grille-substrat : généralement faible en saturation = $0.1 C_{ox}$.

Le circuit équivalent petits signaux est donné à la figure 1.2.

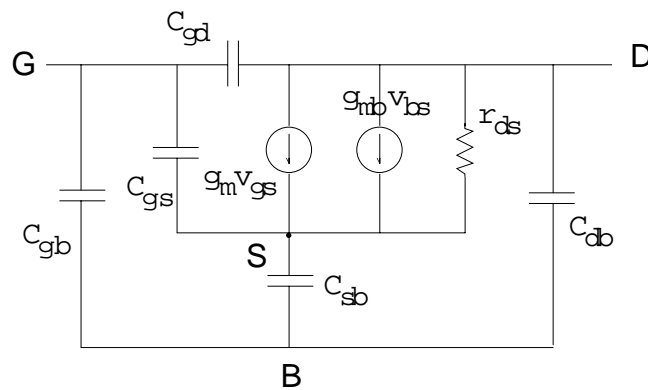


Fig. 1.2 : Modèle petits signaux du transistor MOS.

1.6.3. Le Bruit dans les transistors MOS

On distingue différentes sources de bruit dans un transistor MOS [17].

1.6.3.1. Bruit thermique

Le bruit thermique des résistances est le résultat du mouvement brownien des électrons dans un conducteur. Celui-ci produit aux bornes d'une résistance R une tension erratique $v(t)$ dont la distribution d'amplitude est gaussienne, de moyenne nulle, et dont la valeur quadratique moyenne $\langle v^2(t) \rangle$ (ou variance) mesurée dans une bande df est égale à :

$$\langle v^2 \rangle = 4kTRdf \quad (1.10)$$

où k désigne la constante de Boltzmann ($k=1.381 \cdot 10^{-23} \text{ J/}^\circ\text{K}$),

et T désigne la température de la résistance en $^\circ\text{K}$.

On peut établir en outre que ce résultat est indépendant de la fréquence (tant que celle-ci reste inférieure à une dizaine de GHz). On en conclut que le bruit thermique est un bruit blanc dont la densité spectrale bilatérale est uniforme et égale à $2kTR$ (V^2/Hz).

Dans le cas du canal du transistor en conduction, le bruit thermique peut être représenté par une source de tension connectée à la grille d'un transistor sans bruit. Si le composant est en

saturation, l'approximation $R=2/(3 \cdot g_m)$ peut être utilisé pour calculer la tension de bruit du générateur qui devient:

$$v_{nT}^2 = \frac{8}{3} kT g_m \cdot df \quad (1.11)$$

1.6.3.2. Bruit en 1/f

Le bruit en 1/f ("flicker noise") est un bruit dont la densité spectrale de puissance varie comme $f^{-\nu}$ où ν est proche de 1. C'est donc un bruit très gênant dans les applications basses fréquences. Il est attribué à des inhomogénéités, des disparités de l'interface Si/SiO₂, mais son origine est encore mal cernée. La valeur moyenne de ce bruit est également nulle.

Comme pour le bruit thermique, une modélisation possible de ce phénomène peut se faire par l'intermédiaire d'une source de tension connectée à la grille dont l'amplitude est donnée par la formule approximative suivante :

$$v_{nT}^2 = \frac{K}{C_{OX} WL} \frac{df}{f} \quad (1.12)$$

Où K est un paramètre qui dépend de la température et du processus de fabrication.

1.6.3.3. Bruit de grenaille

Le bruit dit "de grenaille" ("shot noise") est dû à la nature discrète du flux d'électrons. A tout courant continu I se superpose des charges individuelles de propagation aléatoire. W. Schottky a montré que la variance du bruit de grenaille est directement proportionnelle à la valeur moyenne du courant I. Cependant, lorsque le nombre de porteurs est élevé et que leur vitesse est faible, les interactions qui existent entre les charges ont tendance à réduire le caractère aléatoire de la propagation, et donc l'amplitude de ce bruit.

Dans le canal d'un transistor MOS, la densité de charge est généralement élevée et le champ électrique faible. Le bruit induit par le déplacement aléatoire de porteurs est donc mieux décrit par l'expression du bruit thermique.

1.6.4. Régions d'opération du transistor MOS et ces caractéristiques Courant-Tension [16]

Le transistor MOS peut fonctionner en trois régions bien distinctes selon les valeurs de ces tensions de référence qui sont la tension grille-source V_{gs} , la tension de seuil V_T et la tension drain-source V_{ds} . Ces régions d'opération sont:

- 1- région à forte inversion ou de saturation
- 2- région à faible inversion
- 3- région intermédiaire

1.6.4.1 Région à Forte Inversion (saturation)

Le transistor MOS est dit opérant en région à forte inversion si sa tension grille-source est plus grande que sa tension seuil. Dans cette région, le transistor est saturé quand:

$$v_{ds} > v_{gs} - v_T \quad (1.13)$$

où V_{ds} et V_T sont, respectivement, la tension grille-source et la tension seuil. La tension grille-source pour laquelle le transistor commence à saturer est dite tension de saturation, V_{dsat} . Dans la pratique de conception d'amplificateur opérationnel, presque tous les transistors sont polarisés dans la région de saturation, car ceci fournit le plus grand gain en tension pour un courant drain-source donné et pour des géométries de composant données.

Pour déterminer la tension grille-source totale d'un transistor MOS, on peut la diviser en deux parties, la tension seuil et la tension grille-source effective qui traverse le transistor. Dès lors, on a la relation suivante:

$$v_{gs} = v_T + v_{gs,eff} \quad (1.14)$$

Pour les circuits analogiques à basse tension, la plupart des transistors opèrent à la limite de la région de saturation. Dans ce cas, la relation entre le courant drain-source, I_{ds} , et la tension grille-source, V_{gs} , s'exprime de la façon suivante:

$$I_{ds} = \frac{1}{2} \mu C_{ox} \frac{W}{L} V_{gs,eff}^2 \quad (1.15)$$

où μ est la mobilité des porteurs de charge, C_{ox} est la capacité d'oxyde par unité de surface, V_{gs} est la tension grille-source et V_T est la tension seuil du composant. W et L sont, respectivement, la largeur et la longueur.

La transconductance est un paramètre clé du transistor MOS à petit signal. On peut la déterminer en calculant la dérivée partielle du courant drain-source du transistor par rapport à la tension grille-source. Si on utilise l'équation (1.15), on obtient:

$$g_m = \frac{\partial I_{ds}}{\partial V_{gs}} = \mu C_{ox} \frac{W}{L} V_{gs,eff} = \sqrt{2\mu C_{ox} \frac{W}{L} I_{ds}} \quad (1.16)$$

La transconductance g_m d'un transistor opérant en région à forte inversion peut également s'écrire de la manière suivante :

$$g_m = \frac{2I_{ds}}{V_{gs,eff}} \quad (1.17)$$

ce qui est immédiatement déduit de l'équation (1.16).

L'équation (1.16) montre que la transconductance d'un transistor MOS est déterminée par sa tension grille-source effective. Plus la tension grille-source effective est élevée, plus la transconductance est élevée. Cependant, si certains transistors nécessitent une transconductance g_m plus grande, il est possible que la tension grille-source effective ne puisse pas être élevée. Finalement, augmenter la tension grille-source revient à élever la tension d'alimentation. Dans ce cas-là, la transconductance g_m d'un transistor peut être élevé en augmentant le rapport W/L et le courant drain par le même facteur n . De cette façon, la transconductance croît avec le facteur n , alors que la tension grille-source reste constante.

1.6.5.2 Région à Faible Inversion

Le transistor MOS opère en région de faible inversion, ou sous-seuil, quand sa tension grille-source est au-dessous de sa tension seuil:

$$v_{ds} < v_{gs} - v_T \quad (1.18)$$

Dans cette région, le transistor sature quand:

$$V_{ds} > 3 \text{ à } 4 V_{th} \quad (1.19)$$

où V_{th} est la tension thermique kT/q , qui est environ égale à 25mV à la température ambiante. En général, la tension de saturation d'un transistor MOS opérant en faible inversion est plus basse que celle d'un composant opérant en forte inversion.

En région de saturation, la relation entre le courant drain et la tension grille-source d'un transistor MOS opérant en faible inversion s'exprime de la façon suivante [11] :

$$I_{ds} = I_s e^{\frac{V_{gs} - V_T}{nV_{th}}} \quad (1.20)$$

où n est le facteur de la pente en faible inversion et I_s est le courant spécifique, donné par l'équation suivante:

$$I_s = 2n\mu C_{ox} V_{th}^2 \frac{W}{L} \quad (1.21)$$

Les valeurs typiques de I_s s'étale entre 2nA et 200nA [11].

Le remaniement de l'équation (1.20) implique une tension grille-source effective exprimée par la formule suivante:

$$V_{gs,eff} = nV_{th} \ln \frac{I_{ds}}{I_s} \quad (1.22)$$

Cette tension grille-source effective a une valeur négative, car le courant drain-source est plus petit que le courant spécifique. Ceci induit que la tension grille-source d'un transistor opérant en faible inversion est plus petite que la tension grille-source d'un composant opérant en forte inversion. Par conséquent, un transistor polarisé sous un régime de faible inversion est plus adapté pour une opération à basse tension [18].

La transconductance d'un transistor MOS opérant en faible inversion est donnée par l'équation suivante:

$$g_m = \frac{I_{ds}}{nV_{th}} \quad (1.23)$$

On peut déduire de cette formule que le g_m d'un transistor MOS opérant en faible inversion ne dépend que du courant drain-source. Si le transistor nécessite une transconductance plus large, par exemple pour accomplir certaines performances à haute fréquence, le courant drain du transistor doit être augmenté. Cependant, si le courant drain-source est trop élevé, le transistor finit en régime de forte inversion. Bien que le transistor puisse être maintenu en faible inversion en accroissant son rapport W sur L , cela n'est pas toujours possible - la raison la plus fréquente

est la bande-passante - car l'augmentation de la taille du transistor implique l'augmentation des capacités parasites du composant.

1.6.6.3 Région à Inversion Modérée

La discussion ci-dessus peut suggérer qu'il y a une transition abrupte entre inversion faible et forte. Cependant, en pratique, la transition se fait doucement d'une région à l'autre, ce qu'on appelle une transition modérée. Par approximation, la région en inversion modérée étend les courants drain-source entre [18][19]:

$$\frac{1}{8}I_s < I_{ds} < 8I_s \quad (1.24)$$

Pour cette région d'opération, les équations analytiques simples ne sont pas valides. Cependant, il est conseillé d'utiliser des simulations par ordinateur, quand le transistor fonctionne dans cette région.

1.10. CONCLUSION

Nous avons pu montrer dans ce chapitre la nécessité de la conception de circuits intégrés à basse tension d'alimentation et faible consommation ainsi que la manière dont cette contrainte se traduit dans la conception des circuits intégrés numériques et dans la conception des circuits intégrés analogiques. Nous avons aussi présenté l'impact et les limitations de cette réduction de tension d'alimentation.

L'analyse, en premier ordre, montre que la consommation minimum de la puissance requise pour les circuits intégrés analogiques est quasiment indépendante de la réduction de la tension d'alimentation, contrairement aux circuits numériques, où la consommation de puissance dynamique décroît en fonction du carré de la tension d'alimentation.

Ceci s'explique par le fait que la consommation de puissance des circuits intégrés analogiques à température donnée est fixée par le rapport signal-sur-bruit SNR et la fréquence d'opération (ou bande passante requise).

D'autres analyses plus détaillées montrent que la consommation minimum de la puissance des circuits intégrés analogiques est proportionnelle au rapport entre la tension d'alimentation et l'excursion de l'amplitude de la tension du signal d'une extrémité à l'autre (rail-to-rail).

Donc, des circuits analogiques efficaces en terme de puissance de consommation doivent être conçus de telle sorte que l'excursion du signal soit la plus large possible, de préférence d'une extrémité de l'alimentation à l'autre (rail-to-rail),

La baisse de la tension d'alimentation a un impact énorme sur les capacités des circuits intégrés analogiques à manier le signal. La dynamique de sortie décroît dramatiquement, non seulement à cause de la baisse de tension du signal permis, mais aussi à cause de l'existence de tensions de bruits assez élevées, provoquées par de faibles courants. Si on veut avoir une dynamique de sortie maximale, le signal doit être aussi grand que possible. L'étage de sortie de l'amplificateur par exemple, doit posséder alors nécessairement une amplitude de la tension de sortie oscillante d'une extrémité de la tension d'alimentation à l'autre ("rail-to-rail").

Un moyen possible de maintenir une dynamique de sortie suffisante quand on réduit la tension d'alimentation sans dégrader la consommation de puissance des circuits intégrés analogiques, est d'utiliser la technique des circuits composites [34][35], qui sera étudiée dans le chapitre suivant.

Chapitre II

Chapitre 2

Etude et Réalisation des Circuits Intégrés Analogiques à Basse Tension d’Alimentation

2.1. INTRODUCTION

La réduction de la tension d’alimentation a sans doute mené à minimiser la consommation de puissance des cellules numériques, car la consommation moyenne de courant des circuits numériques CMOS est proportionnelle au carré de la tension d’alimentation [25][26]. La puissance dissipée par la circuiterie analogique ne diminue pourtant pas nécessairement quand on baisse la tension d’alimentation. Comme l’empilement traditionnel des transistors a été remplacé par les techniques folding, la valeur du courant d’alimentation circulant au travers des transistors a inévitablement été augmentée.

Désormais, pour diminuer la puissance dissipée dans les circuits analogiques à basse tension d’alimentation, le circuit doit rester aussi simple que possible, tout en maintenant les bonnes spécifications du circuit.

La réduction de la tension d’alimentation a un impact énorme sur la dynamique d’un amplificateur: du côté le plus haut de la tension, la dynamique est réduite à cause de la diminution d’amplitude du signal d’entrée; du côté le plus bas de la tension, elle est réduite du fait du bruit élevé de la tension dû à un faible courant d’alimentation.

Pour maximiser la dynamique de sortie, l’amplificateur à basse tension d’alimentation doit fonctionner avec un signal de tension ayant une amplitude étendue d’une extrémité de la tension d’alimentation à l’autre. Ceci mène à réfléchir à d’autres structures d’amplificateur. Les circuits classiques doivent être remplacés par de nouvelles configurations, plus adaptées aux basses tensions d’alimentation.

L'unité de gain en fréquence d'un amplificateur opérationnel est aussi grandement affectée par les conditions de basse tension d'alimentation et de faible consommation. Le faible courant d'alimentation va réduire dramatiquement la marge de phase lorsque la capacité de charge ne peut pas être réduite.

De plus, pour obtenir un gain à basse fréquence suffisant, l'amplificateur à faible tension d'alimentation nécessite souvent un étage de gain cascode, ce qui implique plus de structure de compensation en fréquence complexe. Dans un environnement à basse tension d'alimentation et à faible consommation, ces structures de compensation en fréquences doivent être efficaces du point de vue de la consommation de puissance.

Ce chapitre présente une nouvelle structure d'amplificateur opérationnel CMOS, adaptée aux basses tensions d'alimentation et aux faibles consommations. Cette structure est basée sur la nouvelle technique **des transistors composites**.

Ce travail rentre dans le cadre de la réalisation d'un convertisseur analogique-numérique (CAN 8 bits) à basse tension d'alimentation (3 volts) pour une application vidéo (TVHD) en technologie 0.5 μm (MASTAR). Dans un premier temps, nous présenterons le système dans lequel cette étude a été réalisée. Puis, dans un deuxième temps, nous montrerons les limitations intrinsèques des structures CMOS traditionnelles, et nous décrirons les nouvelles techniques proposées pour résoudre le problème posé par la conception à basse tension d'alimentation ainsi que leur implémentation. Finalement, nous présenterons la nouvelle structure d'amplificateur opérationnel CMOS, adaptée aux basses tensions d'alimentation, que nous avons réalisée, et nous donnerons les résultats de ses performances.

2.2 PRINCIPE DE LA CONVERSION

Lorsque l'on parle de conversion analogique-numérique, une seule étape nous vient souvent à l'esprit : la quantification. En effet, c'est cette transformation qui caractérise le mieux le passage d'un univers continu (ou plutôt qui nous semble continu) vers un univers discret. Cependant cette transformation a bien d'autres exigences que la quantification. Et c'est ainsi que les convertisseurs analogiques-numériques sont constitués d'une véritable chaîne de conversion dont la composition générale est donnée figure 2.1 [27].

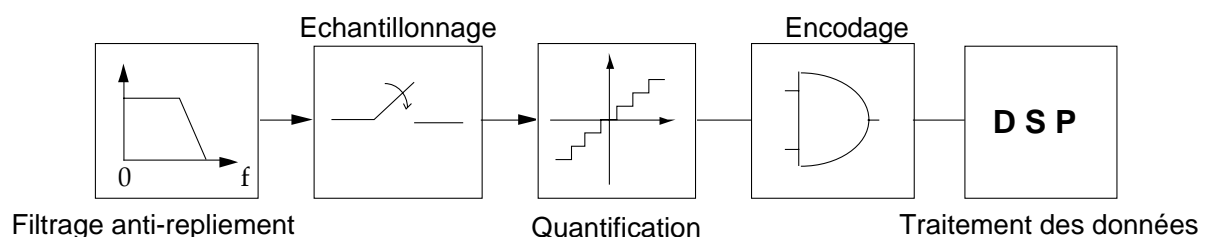


Fig. 2.1: Chaîne de Conversion Analogique-Numérique.

La première étape se compose d'un filtre passe bas qui permet de limiter la bande passante du signal d'entrée. Ceci permettra d'éviter les repliements de spectre lors de l'échantillonnage.

Le rôle du bloc suivant est d'échantillonner le signal à une fréquence donnée puis de maintenir constant le niveau de sortie pour permettre le traitement de l'information.

Le quantificateur dispose, lui, d'un ensemble de niveaux discrets. En fonction de l'échantillon que lui transmet le bloqueur, il lui associe le niveau discret le plus proche.

On trouve ensuite un encodeur dont la tâche essentielle est de coder le flot d'informations issues du quantificateur, dans un langage qui soit compréhensible par les circuits numériques.

La dernière étape est constituée d'un Digital Signal Processing : circuits numériques de mise en forme du signal avec différents algorithmes, suivant le type de quantificateur utilisé.

2.3. DESCRIPTION GENERALE DU SYSTEME

Le but de ce circuit est la mise en forme d'un signal vidéo analogique avant traitement numérique.

Avant d'être traité numériquement, le signal vidéo doit subir un certain nombre de modifications, résumées à la figure 2.2:

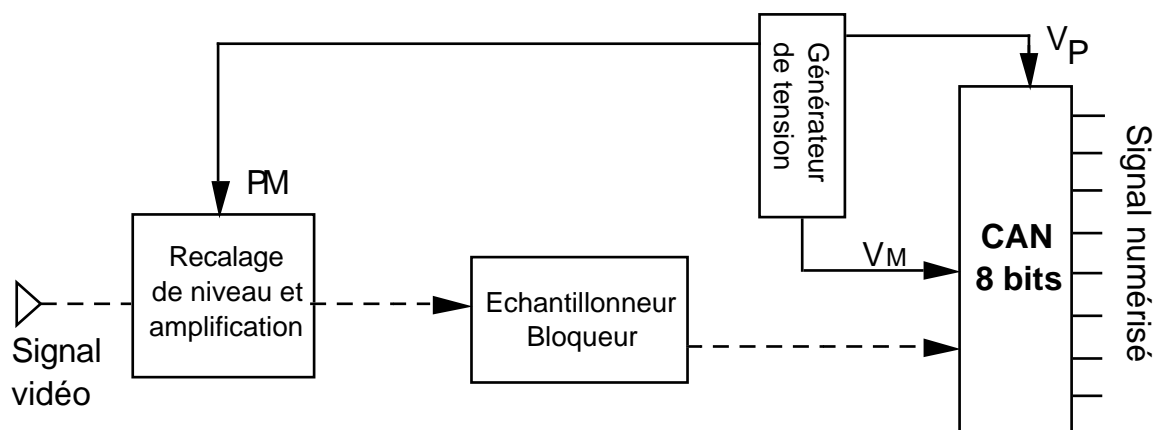


Fig. 2.2 : Synoptique.

La forme vidéo est représentée à la figure 2.3; sa bande passante est comprise dans l'intervalle [0 - 5 MHz].

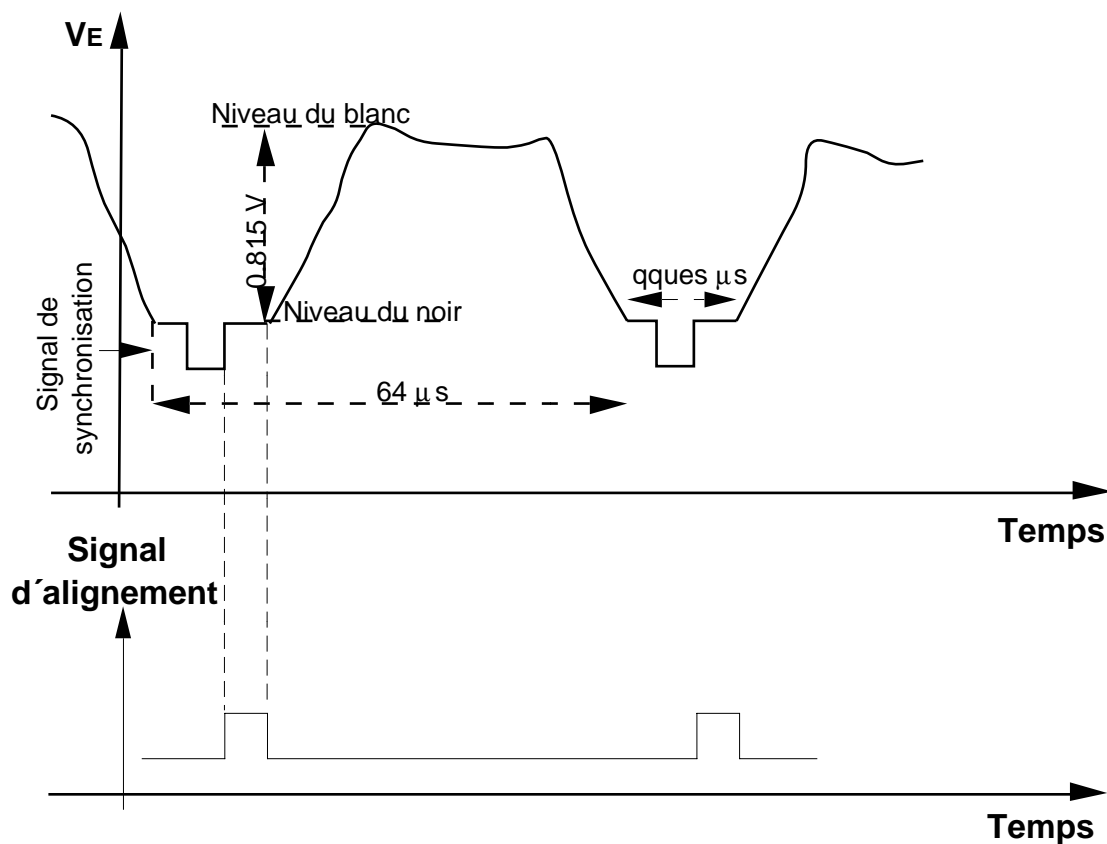


Fig. 2.3 : Signal vidéo.

Détaillons de manière plus approfondie les différents blocs.

Circuit d'alignement de niveau et d'amplification ("Clamping circuit"):

Le bloc dessiné à la figure 2.4 a deux fonctions :

- *une fonction d'alignement de niveau* : il s'agit de recalculer le niveau du noir du signal vidéo à un niveau de tension noté V_{CLAMP} . Pour $V_{CLAMP} = 1.843 \text{ V}$, le signal vidéo est centré sur la plage d'entrée du convertisseur [1.5 V ; 3 V] dont le niveau moyen est 2.25 V (Fig. 2.5).

Le circuit se compose de :

- * un pont résistif qui fixe la tension V_{CLAMP} à partir d'une alimentation externe,
- * un interrupteur réalisé par un transistor MOS en commutation et piloté par un signal externe synchronisé sur l'impulsion de ligne du signal vidéo (Fig. 2.3),
- * un condensateur en entrée (100 pF) qui filtre la composante continue du signal vidéo,
- * un condensateur de charge (1 pF) qui impose la nouvelle tension de noir V_{CLAMP} .

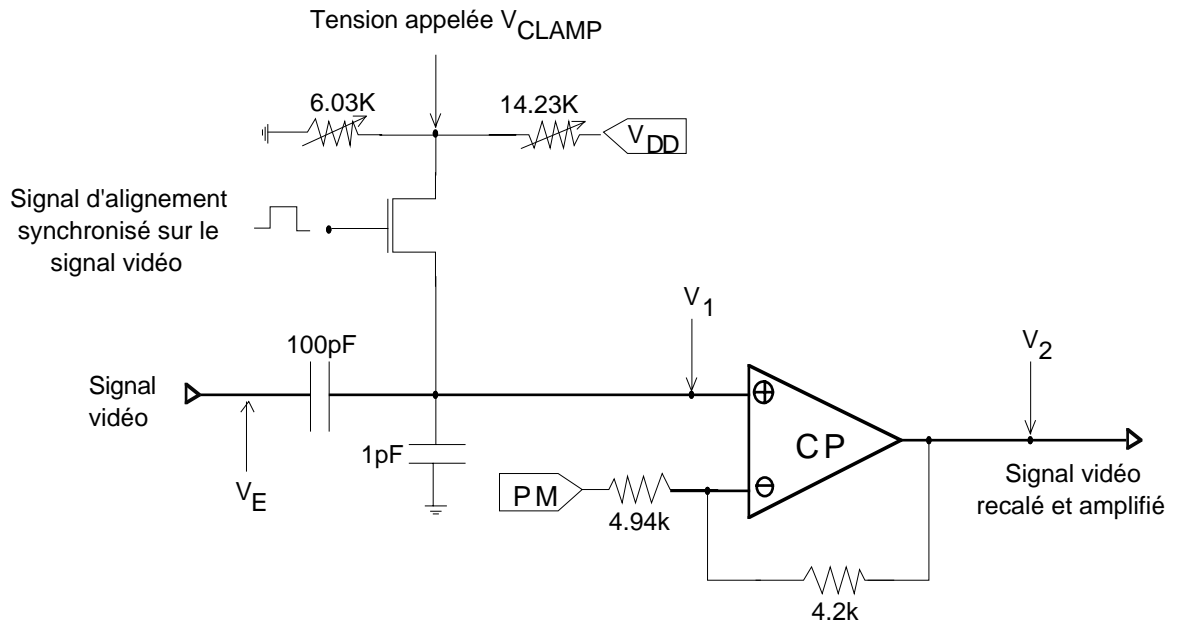


Fig. 2.4 : Circuit d'alignement de niveau et d'amplification.

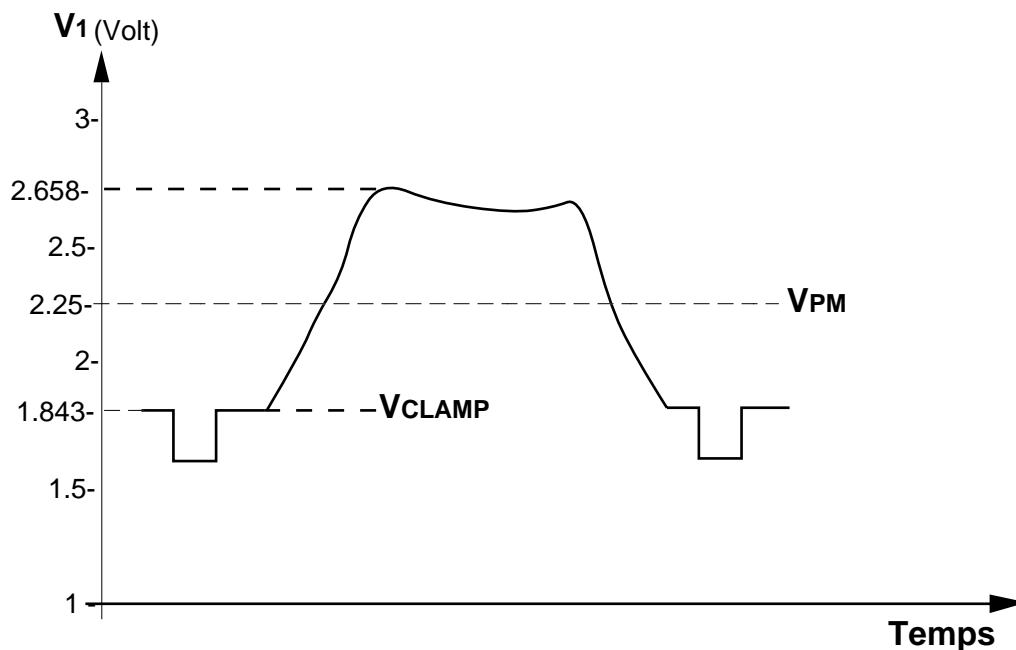


Fig. 2.5 : Recalage du signal vidéo.

- *une fonction d'amplification* : une fois le signal vidéo recalé par rapport à la tension V_{CLAMP} , il s'agit d'amplifier le signal vidéo pour qu'il corresponde à la dynamique totale du convertisseur en entrée. L'amplitude du signal vidéo (initialement égale à 0.815 V) doit être de 1.5 V : le gain de l'amplificateur opérationnel est donné par le rapport $\frac{1,5}{0,815} \approx 1,8$ (Fig. 2.6). Le circuit est composé d'un amplificateur opérationnel utilisé dans un montage amplificateur sans inversion.

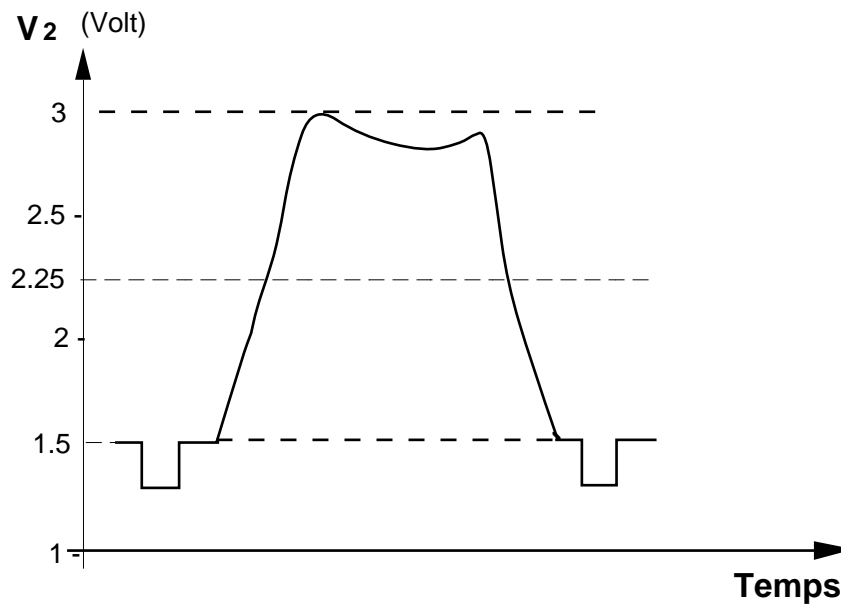


Fig. 2.6 : Amplificateur du signal vidéo.

Echantionneur-bloqueur ou circuit d'échantillonnage ("Sampler-and-hold circuit"):

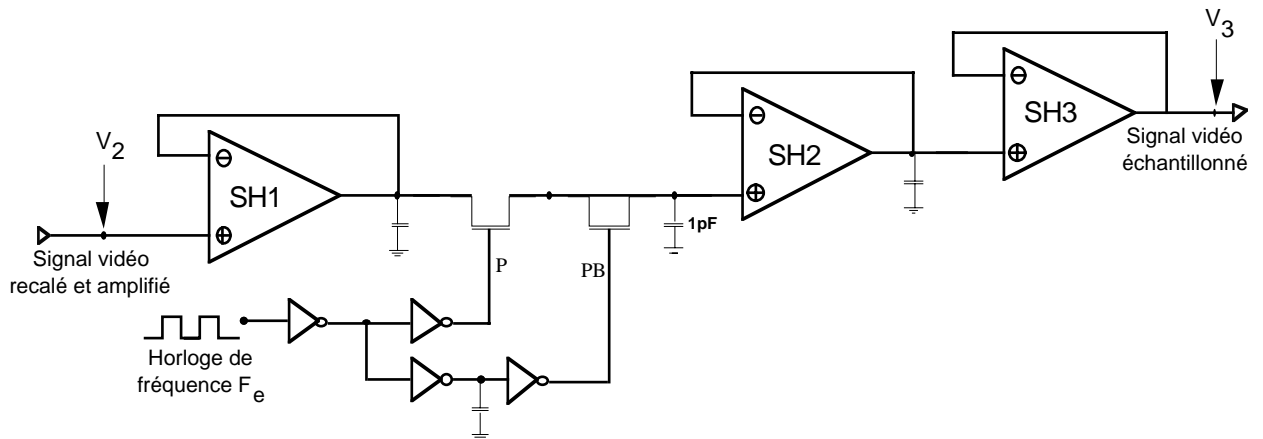


Fig. 2.7 : Echantionneur-bloqueur.

Un échantillonneur-bloqueur est un circuit dont la fonction est d'acquérir une information analogique et de la mémoriser pendant un intervalle de temps donné (Fig. 2.8).

Les deux éléments centraux de l'échantillonneur-bloqueur sont constitués par:

- un interrupteur commandé par une horloge externe dont la fréquence est appelée fréquence d'échantillonnage ($F_e = 12.5 \text{ MHz}$),
- un condensateur dont la capacité-mémoire est de 1 pF .

En amont de ces deux éléments, on trouve un amplificateur opérationnel (SH1) utilisé en montage suiveur ; il joue le rôle d'adaptateur d'impédance. Le signal analogique est présent en permanence à l'entrée de ce montage.

En aval de ces deux éléments, on trouve deux amplificateurs opérationnels (SH2 et SH3) utilisés en montage suiveur qui fournissent en sortie le signal échantillonné sous faible impédance. L'adjonction de l'amplificateur opérationnel SH2 permet de diminuer le couplage parasite entre la sortie de l'amplificateur opérationnel SH3 et la capacité mémoire. L'amplificateur opérationnel SH3 du deuxième montage contient un étage de sortie permettant de commander une importante capacité de charge (25 pF).

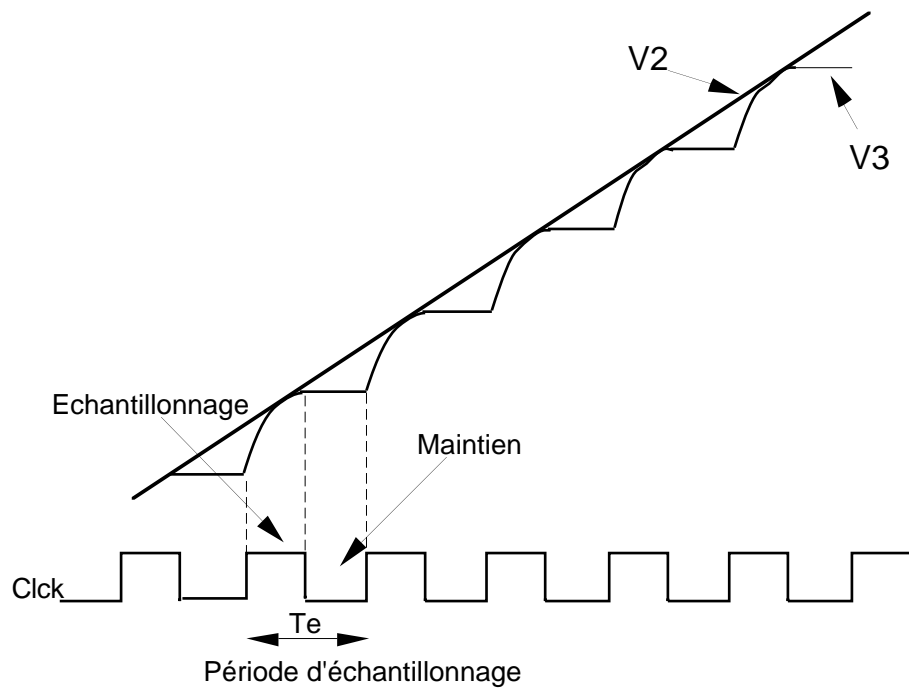


Fig. 2.8 : Echantillonnage d'un signal en rampe.

Principe de fonctionnement : lorsque l'interrupteur est fermé, la capacité-mémoire est chargée à une valeur de la tension analogique à cet instant ; lorsque l'interrupteur repasse en position ouverte, la charge électrique contenue dans le condensateur constitue l'échantillon de tension prélevé.

Rappelons pour mémoire, la condition sur la fréquence d'échantillonnage imposée par le théorème de SHANNON:

$$F_e \geq 2 \cdot F_{\max} \quad (2.1)$$

où F_{\max} : fréquence maximale du signal vidéo ($\Rightarrow F_c \geq 10 \text{ MHz}$).

Cette condition permet d'éviter une perte d'information due au recouvrement des spectres.

Générateur de tension:

Le circuit générateur de tension a pour but de fournir des tensions continues stabilisées en température ; elles sont générées à partir d'une tension d'alimentation externe V_{DD} et d'une source de tension stabilisée en température, réalisées à l'aide de transistors bipolaires parasites de la structure CMOS.

Trois tensions continues sont générées : V_P , V_M et PM :

$$* \begin{cases} V_P = 3V \\ V_M = 1.5V \end{cases} \quad \text{sont les tensions d'alimentation du CAN,}$$

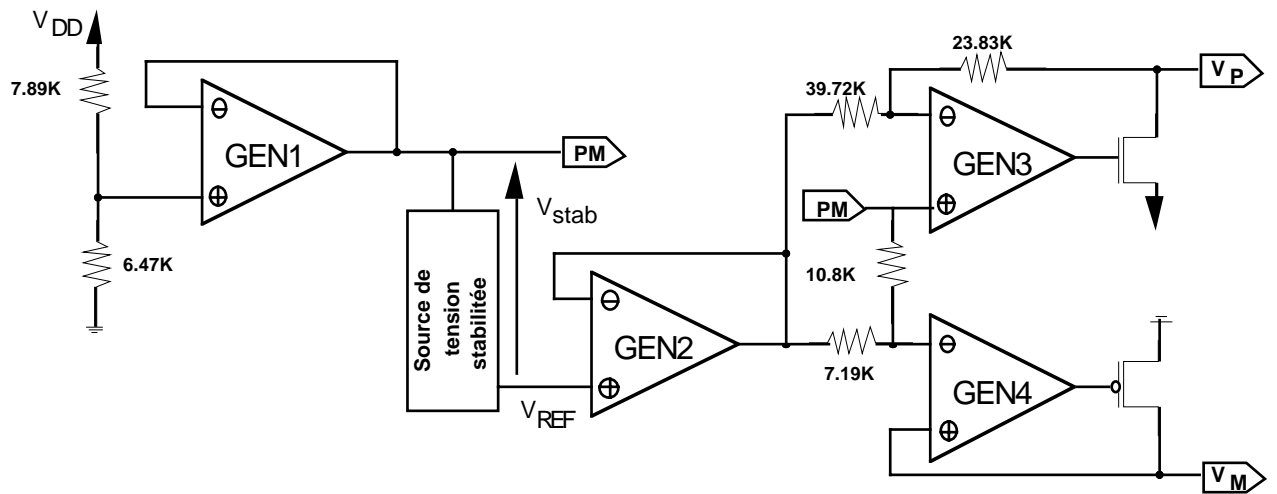
$$* PM = \frac{V_P + V_M}{2} = 2.25V \text{ est le point milieu de la dynamique d'entrée du CAN.}$$

Cette tension est utilisée dans le circuit d'alignement de niveau pour recalibrer le signal vidéo par rapport à la dynamique du CAN ; elle correspond aussi au point milieu de la dynamique du circuit d'échantillonnage.

La tension PM est créée à l'aide d'un pont diviseur résistif et d'un montage suiveur ; ce dernier utilisant l'amplificateur opérationnel GEN1, joue le rôle d'adaptateur d'impédance.

A partir de la tension stabilisée en température V_{REF} , nous générons les tensions V_P et V_M . La tension d'entrée du montage suiveur utilisant l'amplificateur opérationnel GEN2 est égale à V_{REF} ; ce montage suiveur joue le rôle d'adaptateur d'impédance. V_P est générée à partir d'un montage amplificateur avec (GEN3), tandis que V_M est générée à l'aide d'un pont résistif et d'un montage suiveur (GEN4) adaptateur d'impédance.

GEN3 et GEN4 sont suivis chacun d'un transistor jouant le rôle de source suiveuse afin de pouvoir délivrer du courant.



avec $V_{DD} = 5 \text{ V}$; $V_{REF} = 1.3 \text{ V}$; $V_P = 3 \text{ V}$
 $PM = 2.25 \text{ V}$; $V_{stab} = 1.2 \text{ V}$; $V_M = 1.5 \text{ V}$

Fig. 2.9 : Générateur de tension.

Convertisseur analogique-numérique noté CAN (“Analog-Digital Converter ADC”) [28]:

Le CAN 8 bits est une variante de CAN parallèle (“flash ADC”) appelée CAN semi-parallèle ; ces types de convertisseur ont pour principal avantage une grande rapidité de conversion.

Le principe d’un CAN parallèle n bits est le suivant: la tension à convertir est comparée au même instant avec (2^n-1) tensions étalon réalisées avec un réseau de résistances. A partir des signaux fournis par les comparateurs, une logique de codage fournit la valeur de la tension sous forme binaire. Comme cette technique est très coûteuse en surface et en consommation (elle nécessite (2^n-1) comparateurs), des techniques dérivées sont apparues: le CAN semi-parallèle utilise $2.(2^n-1)$ comparateurs.

Les comparateurs utilisés dans le CAN 8 bits sont des comparateurs logiques à auto-zéro ; ils sont au nombre de $2.(2^{\frac{8}{2}} - 1) = 30$.

La figure 2.10 montre l’architecture d’un convertisseur flash ADC à 3 bits donné comme exemple explicatif.

Quand la valeur du signal continue, arrive sur l’entrée de l’échantillonneur-bloqueur (S/H), elle est échantillonnée et gardée constante durant une période de temps. Les comparateurs comparent le signal échantillonné avec la tension de référence générée par la mise en série des résistances, et le code métrique résultant de la sortie du comparateur, est codé en représentation binaire numérique.

Ainsi, le circuit échantillonneur-bloqueur (S/H), échantillonne une nouvelle valeur de la tension d'entrée, et toute la procédure de conversion est répétée encore une fois pour ce nouvel échantillon.

Trois fonctions clés sont réalisées durant ce processus: l'échantillonnage, la quantification, et la génération de tension de référence.

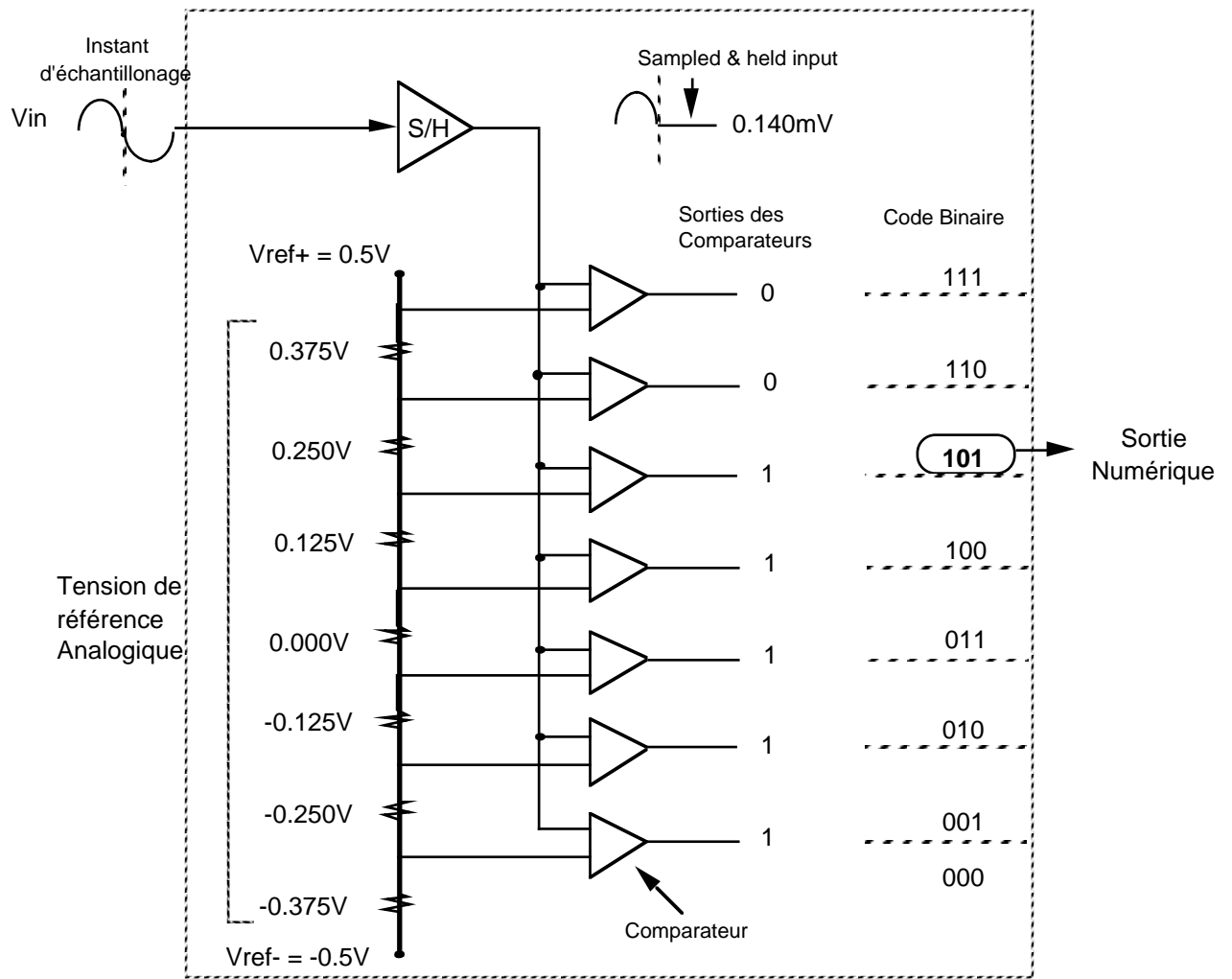


Fig. 2.10 : Architecture d'un convertisseur Flash ADC à 3 bits.

Le schéma d'amplificateur opérationnel utilisé pour CP, SH1 et SH2 est donné à la figure 2.11.

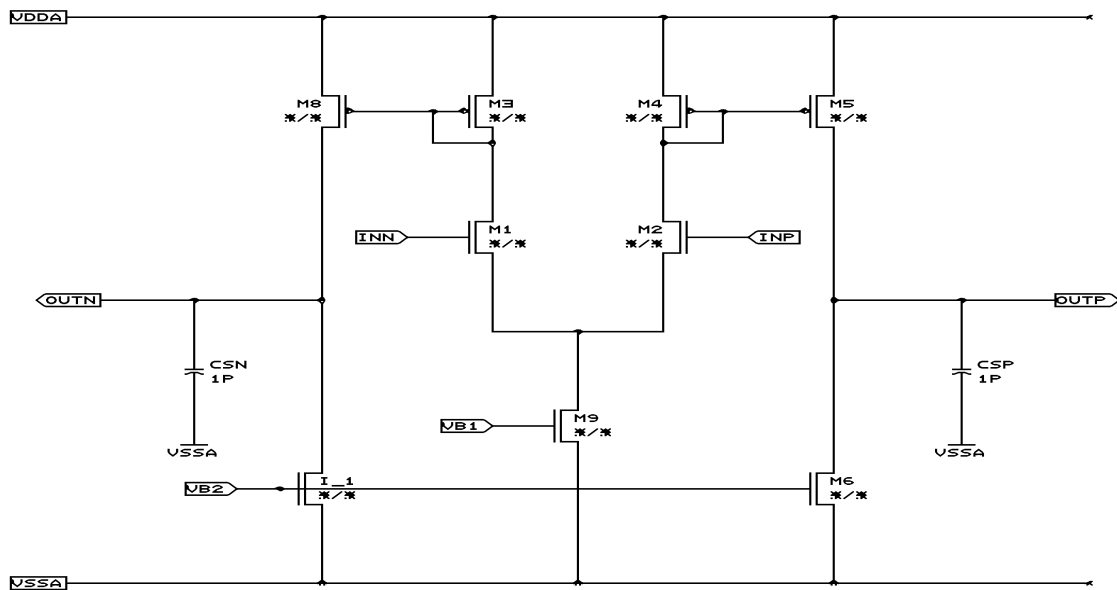


Fig. 2.13 : Amplificateur opérationnel CMOS MIRROR_OTA.

2.4. JUSTIFICATION DES PERFORMANCES DEMANDEES

Pour les amplificateurs opérationnels CMOS (SH1, SH2, et SH3) utilisés dans le circuit d'échantillonnage, nous allons montrer le lien entre les caractéristiques nominales exigées pour ces amplificateurs et les contraintes imposées par le contexte.

Gain minimal requis:

Le CAN 8 bits pour une dynamique de référence de 1.5 V possède une résolution q égale à :

$$q = \frac{1.5}{2^8} \approx 6mV \quad (2.2)$$

La résolution q se définit comme la plus petite variation de tension que le convertisseur peut coder ; elle est aussi appelée **niveau élémentaire de quantification** ou **quantum**.

Rappelons quelques caractéristiques du montage suiveur (figure 2.14).

La relation entre le signal de sortie et le signal d'entrée s'exprime de la façon suivante:

$$\frac{V_S}{V_E} = \frac{A_{VD}}{1 + A_{VD}} = \frac{1}{1 + \frac{1}{A_{VD}}} \quad (2.3)$$

où V_E : tension du signal d'entrée,
 V_S : tension du signal de sortie,
 A_{VD} : gain différentiel statique ($f = 0$) en boucle ouverte de l'amplificateur opérationnel ($A_{VD} = \text{cte}$).

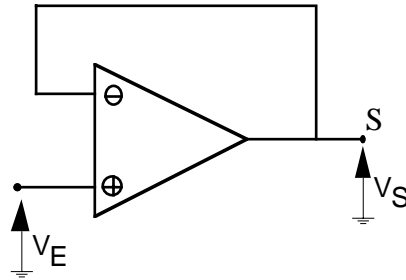


Fig. 2.14 : Montage suiveur.

Dans le cas réel, il apparaît que le gain A_{VD} n'est pas constant sur toute la dynamique de sortie ; il présente en effet des erreurs de non-linéarité qui engendrent une dépendance de A_{VD} en fonction de V_{DIFF} .

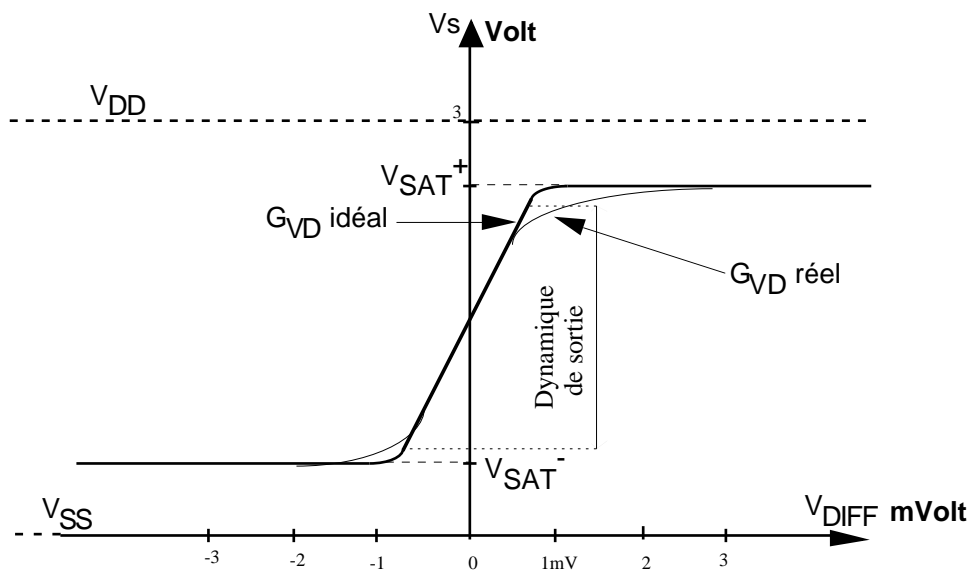


Fig. 2.15 : Courbe de transfert en boucle ouverte.

Dès lors, on a la relation suivante:

$$\frac{V_S}{V_E} = \frac{1}{1 + 1/A_{VD} \cdot (V_{DIFF})} \quad (2.4)$$

Afin de minimiser ces non-linéarités en montage suiveur, il faut avoir un A_{VD} théorique le plus élevé possible (l'idéal étant un gain infini, puisque dans ces conditions le rapport $\frac{V_S}{V_E} \approx 1$ ne dépend plus de V_{DIFF}). Comme un gain infini est impossible à réaliser, le gain minimal requis est déterminé d'après la résolution du CAN 8 bits:

$$\frac{1}{A_{VD}} < \frac{1}{2} \cdot q \Leftrightarrow \begin{cases} A_{VD} > 341 \\ A_{VD} > 50.7db \end{cases} \quad (2.5)$$

$\frac{1}{2} \cdot q = 3 \text{ mV}$ désigne le demi-quantum ou le demi bit du poids le plus faible ("half LSB") du CAN 8 bits.

Temps d'établissement maximal requis:

La limite de cette caractéristique est imposée par la fréquence d'échantillonnage F_e de l'échantillonneur-bloqueur:

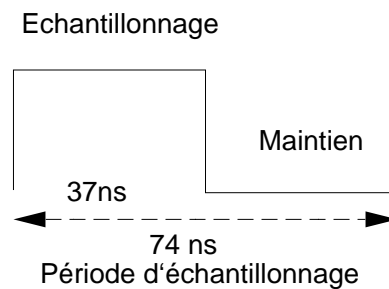


Fig. 2.16 : Décomposition de la période d'échantillonnage.

Le signal en sortie des montages suiveurs doit pouvoir s'établir dans un intervalle de temps qui est inférieur à une demi-période d'échantillonnage; on en déduit le temps d'établissement maximal:

$$T_e < 37 \text{ ns pour un échelon en sortie de } 1.5 \text{ V} \quad (2.6)$$

La tolérance maximale admise (Fig. 2.16) est choisie à partir de la résolution du CAN ; on la choisit égale à un quantum:

$$\text{Tol.} = 6 \text{ mV} \quad (2.7)$$

Par définition, le temps d'établissement est partagé entre le temps d'amortissement et le temps de montée (Fig. 2.16) . Nous donnons ici un ordre de grandeur pour l'un et l'autre.

Pour un système classique du deuxième ordre (fonction de transfert avec un pôle dominant et un pôle non dominant), le **temps d'amortissement** optimal est assuré pour une marge de phase comprise entre 60° et 70° ; de plus la pseudo-période T des oscillations amorties (caractérisant le temps d'amortissement) est reliée à la fréquence de transition par la relation suivante:

$$T \approx \frac{1}{\sqrt{\omega_t \cdot \omega_{pnd}}} \approx \frac{1}{2\pi \cdot \sqrt{f_t \cdot f_{pnd}}} \quad (2.8)$$

où f_t : fréquence de transition,
 f_{pnd} : fréquence du pôle non-dominant.

Pour avoir un temps d'amortissement (donc un temps d'établissement) le plus court possible, la pseudo-période T doit être minimisée et la fréquence de transition f_t doit être maximisée.

Nous rappelons que le **temps de montée** est inversement proportionnel à la vitesse de balayage limite de l'amplificateur opérationnel.

On prend généralement l'approximation suivante:

$$\text{Temps de montée} \approx \text{Temps d'amortissement} \approx \frac{\text{Temps d'établissement}}{2} \approx 18ns \quad (2.9)$$

On en déduit la vitesse de balayage limite:

$$SR = \frac{\text{Amplitude de l'échelon}}{\text{Temps de montée}} \approx \frac{1.5V}{18ns} \approx 83 \text{ V}/\mu s \quad (2.10)$$

D'autre part, il faut prendre la précaution suivante pour l'amplificateur opérationnel SH1 qui est amené à transmettre un signal analogique sinusoïdal dans la bande de fréquence 0-5 MHz ; afin d'éviter une limitation non linéaire en régime harmonique grand signal, la vitesse de balayage limite doit vérifier l'inégalité :

$$SR > 2\pi \cdot f \cdot V \quad (2.11)$$

où f : fréquence maximale du signal,
 v : dynamique maximale (crête à crête) du signal.

Pour $f = 5$ MHz et $v = 1.5$ V, on obtient:

$$SR > 47.V/\mu s \quad (2.12)$$

On peut raisonnablement fixer f_t et f_{pnd} , respectivement à 50 MHz et 200 MHz, ce qui donne d'après la formule (2.8) une pseudo-période $T = 2$ ns ; cela satisfait largement, après simulation, la contrainte d'un temps d'amortissement ≤ 18 ns.

Les performances électriques demandées sont résumées ci-dessous:

$$\begin{cases} A_{VD} \geq 51 \text{ dB} ; & F_t \geq 50 \text{ MHz} ; & SR \geq 83 \text{ V}/\mu s \\ T_e \leq 37 \text{ ns} & \text{pour un chelon de } 1.5 \text{ V} \end{cases} \quad (2.13)$$

Pour $V_{DD} = 5$ V et $V_{SS} = 0$ V .

Après avoir passé en revue la description générale du système et le contexte de la conception de ce circuit, nous présenterons les différents problèmes rencontrés lors de l'implémentation du circuit, ainsi que les limitations des structures habituelles. Nous concluons ce chapitre par la présentation de la méthode résolvant le problème de la conception des circuits à basses tensions d'alimentation et à faible consommation. Nous appliquerons cette méthode à l'amplificateur opérationnel CMOS (SH3), amplificateur qui posait un problème au niveau de la conception du convertisseur 8 bits à basses tensions d'alimentation (3 Volts). Cette conclusion constitue l'aboutissement de nos recherches sur la conception des circuits à basses tensions d'alimentation et à faible consommation.

2.5. LIMITATION DES STRUCTURES CMOS TRADITIONNELLES

2.5.1. Etage de gain simple:

Considérons l'étage de gain CMOS "classique" représenté à la figure 2.17 :

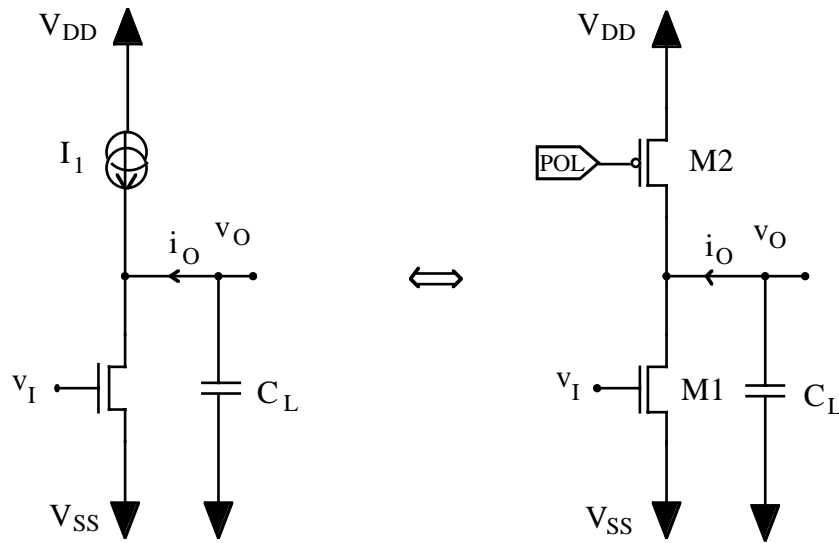


Fig. 2.17 : Etage de gain CMOS.

Détaillons les caractéristiques principales en régime de petits signaux.

Les transistors M1 et M2 étant dans la région de saturation, les paramètres de petit signal peuvent s'exprimer de la façon suivante:

$$g_m = \sqrt{2\mu C_{ox} \frac{W}{L} I_D} \quad (2.14)$$

$$g_{ds} = \lambda I_D \quad (2.15)$$

Pour tout transistor MOS, on peut écrire l'équation suivante:

$$i_d = g_m \cdot v_{gs} + g_o \cdot v_{ds} \quad (2.16)$$

Dans ce circuit (Fig. 2.17), la source de courant est connectée au drain du transistor MOS, donc $i_d = 0$ et l'équation (2.16) devient:

$$0 = g_m \cdot v_{gs} + g_o \cdot v_{ds} \quad (2.17)$$

En substituant aux tensions v_{gs} et v_{ds} les tensions v_i et v_o ($v_i = v_{gs}$ et $v_o = v_{ds}$), on obtient :

$$\frac{v_o}{v_i} = A_o = -\frac{g_m}{g_o} = -g_m \cdot r_{out} \quad (2.18)$$

avec:

$$r_{out} = \frac{1}{g_o} \quad (2.19)$$

où g_m : transconductance du transistor d'entrée M1,

$r_{out} = (r_{ds})_{M1} // (r_{ds})_{M2}$: résistance de sortie de l'étage de gain.

Le gain-dc A_o est donc le produit de la transconductance avec l'impédance de sortie du circuit, ce qui explique que la réalisation de la fonction de transfert du signal se fait en deux étapes: tout d'abord, la tension d'entrée est transformée en courant via la transconductance g_m , puis le courant est transformé en tension de sortie via l'impédance de sortie r_{out} de l'étage.

Le gain-dc A_o peut s'exprimer aussi en fonction du courant de drain I_D et du rapport W/L .

$$A_o = -\frac{g_{m1}}{g_{ds1} + g_{ds2}} = -\frac{\sqrt{2\mu_1 C_{ox} \left(\frac{W}{L}\right)_1}}{\sqrt{I_D}(\lambda_n + \lambda_p)} \quad (2.20)$$

On en déduit que le gain-dc est inversement proportionnel à la racine carré du courant drain, car la transconductance du transistor dépend de la racine carré du courant drain (Eq.(2.14)), et la conductance de sortie dépend du courant drain (Eq.(1.15)).

Analyse du circuit en haute fréquence.

En haute fréquence, on doit prendre en compte les effets des différentes capacités présentes dans le circuit, comme le montre le schéma équivalent du circuit (fig. 2.18).

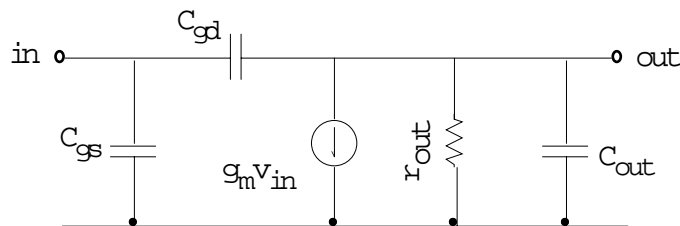


Fig. 2.18 : Schéma équivalent de l'étage de gain avec la capacité de charge.

La réponse en fréquence d'un étage de gain CMOS classique est calculée d'après le circuit équivalent (figure 2.18).

C_{gs} représente la capacité totale existant entre la grille et la source, C_{gd} la capacité totale existant entre la grille et le drain et C_{out} la capacité totale connectée au noeud de sortie. $C_{out} = C_{ds} + C_{load}$ où C_{ds} est la capacité drain source et C_{load} la capacité de charge.

Dès lors, la fonction de transfert, est caractérisée par 1 pôle et 1 zéro:

$$\frac{v_o}{v_i} = -g_m \cdot r_{out} \frac{1 - sC_{gd}/g_m}{1 + s(C_{out} - C_{gd})r_{out}} \quad (2.21)$$

Le diagramme de bode de cette fonction de transfert est donné à la figure 2.19

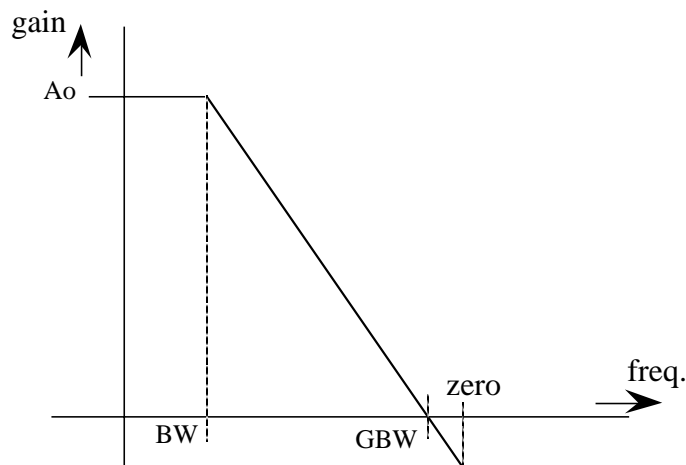


Fig. 2.19 : Diagramme de bode de la fonction de transfert de l'étage de gain.

La bande passante du circuit est donc:

$$BW = \frac{1}{r_{out} C_{tot}} \quad (2.22)$$

où: $C_{tot} = C_{out} + C_{gd}$.

Un paramètre très important de petit signal est la fréquence de transition f_T ("Unity gain frequency"), appelée aussi produit gain-band ("gain-bandwidth product"). C'est la fréquence pour laquelle le gain s'annule (0 dB). Ce paramètre peut s'exprimer de la manière suivante:

$$GBW = A_o BW = \frac{g_m}{C_{tot}} \quad (2.23)$$

Les limites fondamentales d'un étage de gain simple sont la bande passante et le gain-dc. Il est souvent désirable de concevoir un circuit qui a en même temps un gain-dc élevé et un produit gain-bande aussi assez grand. Il est possible d'améliorer le gain sans dégrader la bande passante.

La technique très simple qui consiste à cascader des étages de gain pour augmenter le gain total de l'amplificateur est très nocive pour le produit gain-bande, car elle introduit systématiquement à chaque étage de nouveaux pôles secondaires (proches les uns des autres) dans la fonction de transfert ; en effet, on est obligé en boucle fermée d'utiliser des techniques de compensation en fréquence (séparation des pôles) qui ont pour conséquence de diminuer la fréquence de transition pour une marge de phase donnée.

On a les relations de proportionnalité suivantes [29] :

$$A_o \begin{cases} \propto L \\ \propto \frac{1}{\sqrt{I_{DS}}} \end{cases} \quad \text{et} \quad GBW \begin{cases} \propto \sqrt{\frac{W}{L}} \\ \propto \sqrt{I_{DS}} \end{cases} \quad (2.24)$$

Le gain-dc est donc proportionnel à la longueur du canal L et inversement proportionnel au courant drain I_D , tandis que le produit gain-bande, GBW, est proportionnel au courant drain I_D et inversement proportionnel à L. Si on augmente l'effet de la tension grille-source en réduisant en même temps la longueur du canal L, on augmente le produit gain-bande GBW, tout en réduisant le gain-dc A_o par le même facteur. Ce qui signifie que, pour une approximation de première ordre, le produit gain par gain-bande, $A_o GBW$, est une valeur indépendante du courant drain I_D et de la longueur du canal L.

$$A_o GBW = \text{Constant} \quad (2.25)$$

Le produit constant $A_o GBW$, ne dépend que de certains paramètres de la technologie utilisée.

Dès lors, on se rend compte qu'il est difficile d'augmenter le gain A_o sans diminuer le produit gain-bande GBW; la figure 2.20 illustre ce problème.

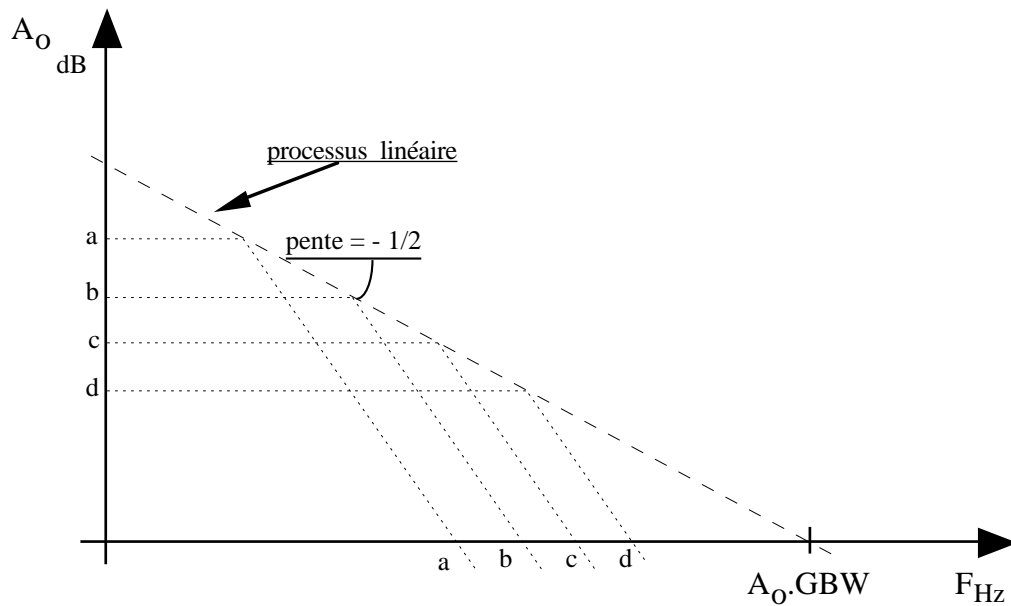


Fig. 2.20 : Réponse en fréquence d'un étage de gain CMOS classique (processus linéaire et ligne équivalente pour l'étage simple).

Remarque : Un autre inconvénient de l'étage de gain de la figure 2.17 provient de sa capacité Miller c_M (capacité grille-drain c_{gd1} du transistor M1, ramenée en entrée par " Effet Miller") élevée ; en effet, le drain du transistor M1 est un nœud à fort gain :

$$c_M = (1 + A_o) \cdot c_{gd1} \Rightarrow c_M \approx \left(1 + \frac{g_m \cdot r_{ds}}{2}\right) \cdot c_{gd1} \quad (2.26)$$

2.5.2. Etage de gain de circuit cascode:

Une amélioration de ce problème est proposée en utilisant un circuit cascode (Fig. 2.21). Ce qui permet d'augmenter le gain sans diminuer le produit gain-bande, par la mise en série de deux transistors. On obtient un étage de gain cascodé qui possède une transconductance effective $g_{m\text{eff}}$ similaire à la transconductance g_m de l'étage de gain classique. La figure 2.21 explique ce phénomène.

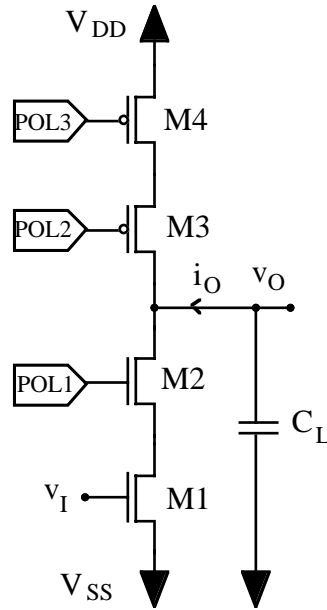


Fig. 2.21 : Etage de gain cascodé CMOS.

Nous allons montrer que le gain-dc peut s'exprimer en fonction de la transconductance effective, g_{meff} , et de l'impédance de sortie, R_{out} . Nous allons aussi montrer que pour les basses fréquences, la structure du circuit cascode a pour effet d'augmenter l'impédance de sortie avec peu d'effet sur la transconductance. Pour les hautes fréquences, l'effet de l'étage cascode est presque négligeable.

La transconductance effective est donnée par l'équation suivante:

$$\frac{\Delta I}{\Delta V_i} = g_{m1} \frac{g_{m2} r_{o1} + \frac{r_{o1}}{r_{o2}}}{g_{m2} r_{o1} + \frac{r_{o1}}{r_{o2}} + 1} = g_{meff} \quad (2.27)$$

Le gain intrinsèque de cet étage est alors :

$$A_o = g_{meff} \cdot R_{out} \quad (2.28)$$

L'impédance de sortie peut donc être calculée facilement. Elle est fortement augmentée en comparaison avec celle d'un étage de gain simple :

$$R_{out} = (g_{m2} r_{o2} + 1) r_{o1} + r_{o2} \quad (2.29)$$

Le gain-dc est par conséquent égale à:

$$A_o = g_{m1}r_{o1}(g_{m2}r_{o2} + 1) \quad (2.30)$$

Cette expression montre que le gain d'un étage cascode est approximativement égal au carré du gain d'un étage simple:

$$A_{ocascade} \cong g_{m1}r_{o1}g_{m2}r_{o2} \cong A_{osimple}^2 \quad (2.31)$$

Le transistor cascode M2 ne perturbe pas le produit gain-bande :

$$GBW = \frac{g_{meff}}{C_{load}} \quad (2.32)$$

On obtient un étage de gain cascodé (Fig. 2.21) qui possède une transconductance effective $g_{m,eff}$ similaire à la transconductance g_m de l'étage de gain classique, ce qui donne:

$$GBW_{cascode} \cong GBW_{simple} \quad (2.33)$$

La mise en cascode de deux transistors permet d'augmenter le gain-dc sans sacrifier la vitesse (i.e. même produit gain-bande GBW). Comme le montre la figure 2.22, la courbe du processus linéaire pour un étage cascode est de pente égale à -2/3, tandis que celle d'un étage simple est égale à -1/2 . Pour une même valeur de produit gain-bande , on a un gain-dc supérieur a celui d'un étage de gain simple. La courbe (d) représentant la fonction de transfert du montage cascode illustre bien ce phénomène; le gain-dc est égal au double de celui d'un étage de gain simple.

Nous avons montré qu'il est possible d'augmenter le gain en gardant le produit gain-bande assez élevé. Ceci est vérifié jusqu'à une certaine limite. Cette limite est déterminée par l'équation suivante [29]:

$$\sqrt{A_{ocascade}} GBW_{cascode} = \text{constant} \quad (2.34)$$

Dans l'idéal, on voudrait que le gain de l'amplificateur opérationnel soit le plus élevé possible, ce qui correspond à la courbe de pente égale à -1. Il est impossible d'atteindre cet objectif simplement par la mise en cascode de transistors.

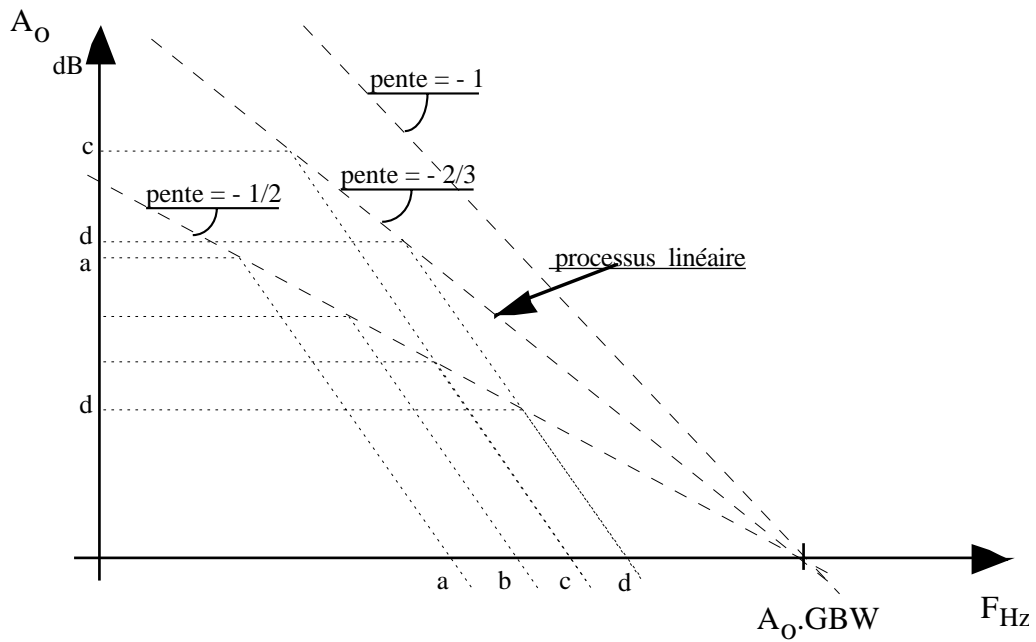


Fig. 2.22 : Processus linéaire et ligne équivalente pour l'étage cascode.

En conclusion, ce nouvel étage de gain permet d'élever à la puissance 2 le gain intrinsèque sans perturber le produit gain-bande (les pôles dus aux transistors cascodes M2 et M3, étant très éloignés du pôle principal). Cependant, ces améliorations se font au prix d'une dégradation significative de la dynamique de sortie (4 transistors à polariser en zone saturée).

De nouvelles structures sont analysées dans le paragraphe suivant ; elles permettent d'augmenter encore le gain intrinsèque de l'étage pour la même dynamique de sortie (voir circuit composite).

2.6. NOUVELLES STRUCTURES

2.6.1. Circuit Cascode Régulé [30][31][32]

2.6.1.1. Principe général

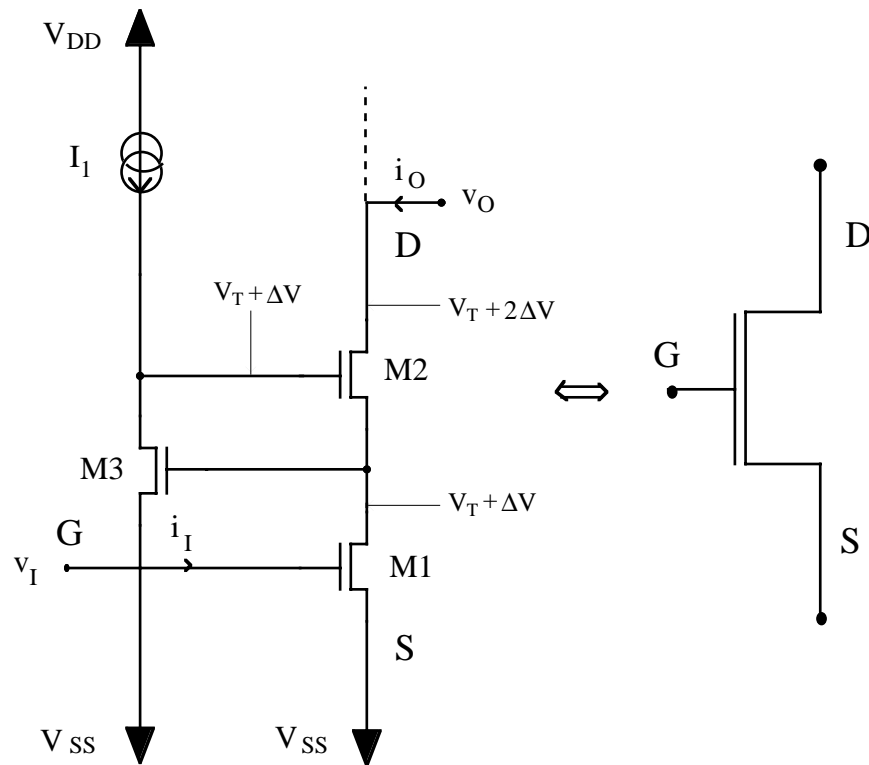


Fig. 2.23 : Circuit cascode régulé.

Le circuit cascode régulé est composé de deux branches (Fig. 2.23) :

- la **branche principale** constituée de deux transistors (\Leftrightarrow un circuit cascode). M1 est le transistor d'entrée et M2 est le transistor cascode ;

- la **branche de contre-réaction** constituée du transistor M3 et de la source de courant I_1 ; elle réalise un étage d'amplification.

Etudions le principe de fonctionnement du circuit cascode régulé :

Le transistor d'entrée M1 convertit une tension v_I en un courant i_O ; pour garantir une résistance de sortie très élevée, la tension drain-source du transistor M1 est gardée constante,

grâce à une boucle de contre-réaction constituée par l'étage de gain (transistor M3 + source de courant) et le transistor cascode M2.

Etudions l'effet d'une petite variation de tension positive sur la grille du transistor d'entrée :

$$\begin{aligned} \Delta v_I \uparrow : \\ \Rightarrow (\Delta v_D)_{M1} = (\Delta v_G)_{M3} \downarrow \Rightarrow (\Delta v_D)_{M3} = (\Delta v_G)_{M2} \uparrow, \\ \Rightarrow (\Delta v_S)_{M2} = (\Delta v_D)_{M1} \uparrow, \end{aligned}$$

où $v_I = V_I + v_i$: composante continue + composante variable,

$(\Delta v_D)_{M1}$: petite variation de tension sur le drain du transistor M1,

$(\Delta v_G)_{M3}$: petite variation de tension sur la grille du transistor M3,

etc...

On s'aperçoit que la tension de drain du transistor M1, $(v_D)_{M1}$, est maintenue constante.

On peut aussi raisonner en disant que la transconductance du transistor cascode M2 (g_{m2}) est multipliée par le gain de contre-réaction (M3, I1) (Eq. 2.41).

Remarque : Dans le but de se rapprocher d'un transconducteur parfait ($\Delta i_O = f(\Delta v_I)$), au lieu de $\Delta i_O = f(\Delta v_I, \Delta v_O)$, on cherche à avoir une résistance de sortie infinie ; ceci est réalisé lorsque la tension de drain du transistor d'entrée est gardée constante quelles que soient les variations imposées sur la grille du transistor d'entrée (posons $v_I = \text{cte}$, si l'on fait varier la tension de sortie v_O , le courant de sortie i_O reste inchangé car la tension de drain du transistor d'entrée est constante).

Par rapport au circuit cascode "classique" (Fig. 2.18), le circuit cascode régulé présente l'avantage d'assurer une tension de drain plus stable (grâce à une augmentation de la transconductance du transistor cascode M2). Le circuit cascode constituait déjà, en lui-même, une amélioration par rapport à un transistor d'entrée unique en "protégeant" le drain du transistor d'entrée (en effet, le transistor cascode permet de diminuer les variations de tension sur le drain du transistor d'entrée).

Détaillons quelques caractéristiques du circuit cascode régulé.

Dynamique de sortie (en régime statique) (Fig. 2.22):

On se placera dans l'hypothèse où tous les transistors du circuit cascode régulé fonctionnent en zone saturée avec un régime de forte inversion ; ce qui implique des conditions minimales sur les tensions V_{GS} et V_{DS} pour un transistor MOS canal N:

$$\begin{cases} V_{GS} = V_T + \Delta V \\ V_{DS} = V_{DS_{sat}} = V_{GS} - V_T = \Delta V \end{cases} \quad (2.35)$$

où V_T est la tension de seuil. Les ordres de grandeur sont : $V_T = 0.7 \text{ V}$ et $\Delta V = 0.2 \text{ V}$.

On désigne par V_{Omin} l'excursion de tension minimale en sortie qui est nécessaire pour que tous les transistors soient saturés.

Or, la tension de sortie V_O est égale à :

$$V_O = (V_{DS})_{M1} + (V_{DS})_{M2} \quad (2.36)$$

Une contrainte supplémentaire est appliquée sur la tension $(V_{DS})_{M1}$ puisque $(V_{DS})_{M1}$ doit être égale à $(V_{GS})_{M3}$; d'où :

$$V_O = (V_{GS})_{M3} + (V_{DS})_{M2} \quad (2.37)$$

En appliquant les conditions de conduction et de saturation (Eq. (5.11)), on obtient :

$$V_{Omin} = V_T + 2\Delta V \quad (2.38)$$

Remarque : Dans le cas d'un circuit cascode, $V_{Omin} = 2\Delta V$; cependant, le circuit cascode régulé présente l'avantage de pouvoir fonctionner même lorsque certains de ses transistors ne sont plus saturés ; dans ces conditions, on arrive à une tension V_{Omin} plus faible (comparable à un circuit cascode) au détriment de la résistance de sortie (qui diminue fortement mais reste comparable à celle d'un circuit cascode).

Résistance de sortie (en régime de petits signaux):

En supposant le courant de sortie constant, la résistance de sortie à basse fréquence est donnée par la formule :

$$r_{out} = \frac{g_{m_2} g_{m_3}}{g_{ds_1} g_{ds_2} \cdot (g_{ds_3} + g_{ds_i})} \quad (2.39)$$

ou encore
$$r_{out} = \frac{g'_{m_2}}{g_{ds_1} g_{ds_2}} \quad (2.40)$$

où g_{m_2} : transconductance du transistor M2,

$g'_{m_2} = g_{m_2} \cdot \frac{g_{m_3}}{g_{ds_3} + g_{ds_i}}$: transconductance équivalente ramenée sur le transistor M2,

g_{ds_2} : conductance drain-source du transistor M2,

g_{ds_i} : conductance drain-source de la source de courant,

etc....

On obtient la formule approximative suivante :

$$r_{out} \approx \frac{(g_m)^2 \cdot (r_{ds})^3}{2} \quad (2.41)$$

où $g_{ds_3} + g_{ds_i} \approx 2 g_{ds}$

Capacité Miller (“Miller capacitance”) (en régime de petits signaux) :

Par rapport au circuit cascode, le circuit cascode régulé diminue la capacité Miller ; en effet, le gain sur le drain du transistor M1 est divisé par le gain de la branche de contre-réaction (I1,M3) :

$$c_M = (1 + A_o) \cdot c_{gd_1} \Rightarrow c_M \approx \left(1 + \frac{g_{m_1} \cdot (g_{ds_3} + g_{ds_i})}{g_{m_2} g_{m_3}} \right) \cdot c_{gd_1} \quad (2.42)$$

Cet étage de gain cascode réglé possède une transconductance $g_{m,eff}$ qui est presque égale à la transconductance g_m de l'étage de gain classique (Fig. 2.17). Par contre, la résistance de sortie est bien supérieure :

$$r_{out} \approx \frac{(g_m)^2 \cdot (r_{ds})^3}{4} \quad (2.44)$$

Le gain intrinsèque de cet étage est donc :

$$A_o = g_{m,eff} \cdot r_{out} \approx \frac{(g_m \cdot r_{ds})^3}{4} \quad (2.45)$$

On peut aussi exprimer A_o de la manière suivante :

$$A_o = \frac{(g_m \cdot r_{ds})^2}{2} \cdot A_{add} \quad (2.46)$$

$$\text{où } A_{add} = \frac{g_{m_3}}{(g_{ds_3} + g_{ds_i})} = \frac{g_{m_3} \cdot r_{ds}}{2} \quad (2.47)$$

La fonction de transfert peut être approximativement donnée, en négligeant les pôles secondaires, par la formule :

$$\overline{A}_v = -\frac{(g_m \cdot r_{ds})^3}{4} \frac{1}{1 + j \frac{f}{f_{pd}}} \quad (2.48)$$

où $f_{pd} \approx \frac{(g_{ds})^3}{4(g_m)^2 \cdot C_L}$: fréquence du pôle dominant.

Le produit gain-bande reste inchangé :

$$GBW = \frac{g_{m,eff}}{2\pi \cdot C_L} \approx \frac{g_m}{2\pi \cdot C_L} \quad (2.49)$$

Ce nouvel étage possède un gain équivalent à 3 étages de gain classiques cascades, sans en avoir les inconvénients en régime variable ; en effet, il se comporte en haute fréquence comme un étage de gain classique avec un seul pôle (les pôles secondaires introduits sont suffisamment éloignés du pôle principal pour ne pas perturber la fréquence de transition).

2.6.1.3. Miroir de courant cascode régulé (“regulated cascode current mirror”)

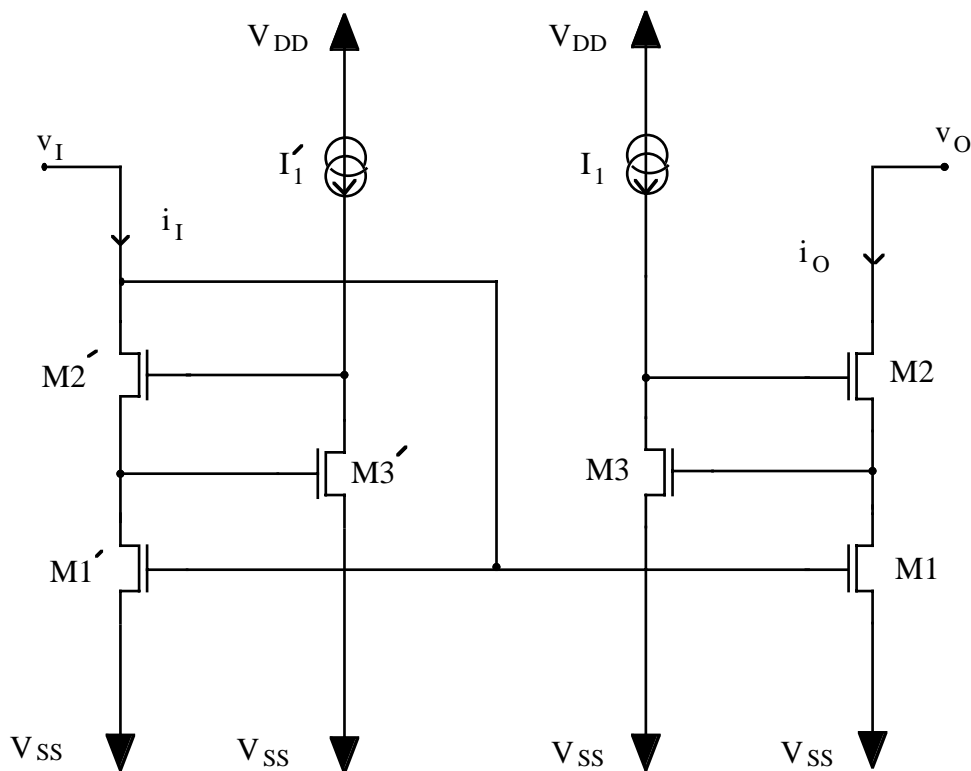


Fig. 2.24 : Miroir de courant cascode régulé.

Ce miroir de courant utilisant des circuits cascode régulé a pour principal avantage par rapport aux structures classiques (miroir simple, miroir de Wilson, miroir cascode[33]) d’avoir une résistance de sortie en petits signaux très élevés :

$$r_{out} \approx \frac{(g_m)^2 \cdot (r_{ds})^3}{2} \quad (2.50)$$

Elle réalise un décaleur de niveau : le courant circulant dans la branche est constant et égal à I_1 . Dans ces conditions, la tension $v_{GS} = v_{DS}$ du transistor M4 doit rester constante quelles que soient les variations imposées sur sa source :

$$(\Delta v_D)_{M1} = (\Delta v_S)_{M4} \uparrow \Rightarrow (\Delta v_G)_{M4} \uparrow \text{ tel que } (v_{GS})_{M4} = \text{cte}$$

Le circuit composite a pour principal avantage, par rapport au circuit cascode régulé (Eq. 2.38), d'avoir une dynamique de sortie bien meilleure ; en reprenant les mêmes conventions (Eq. 2.35) que précédemment, on a, pour le circuit composite (Fig. 2.25) :

$$V_O = (V_{DS})_{M1} + (V_{DS})_{M2} \Rightarrow V_{O_{\min}} = 2\Delta V \quad (2.51)$$

Comme pour le circuit cascode régulé, un fonctionnement avec certains transistors non saturés permet d'obtenir une valeur de $V_{O_{\min}}$ encore plus faible, mais au détriment de la résistance de sortie (en petits signaux). De plus, ce nouveau circuit conserve les propriétés du circuit cascode régulé énoncées précédemment.

L'idée principale de cette méthode, est que pour la conception des circuits à basse tension d'alimentation où les structures sont limitées, au niveau du gain ou au niveau de la dynamique de sortie de l'amplificateur, on peut remplacer un transistor ou un couple de transistors de cette structure montés en cascode par un ensemble de transistors en structure composite [34][35] (Fig.2.26, Fig. 2.27), ce qui permet d'augmenter le gain de l'amplificateur et qui surtout, donne une excursion de sortie très large, aidant ainsi à réduire la tension d'alimentation, sans perdre de l'efficacité du circuit. On note que ces transistors composites sont en général de petites tailles (W et L). L'application de cette technique est assez délicate dans le sens où il faut bien savoir dans quel cas son application peut être profitable et efficace, sans générer une consommation de puissance importante.

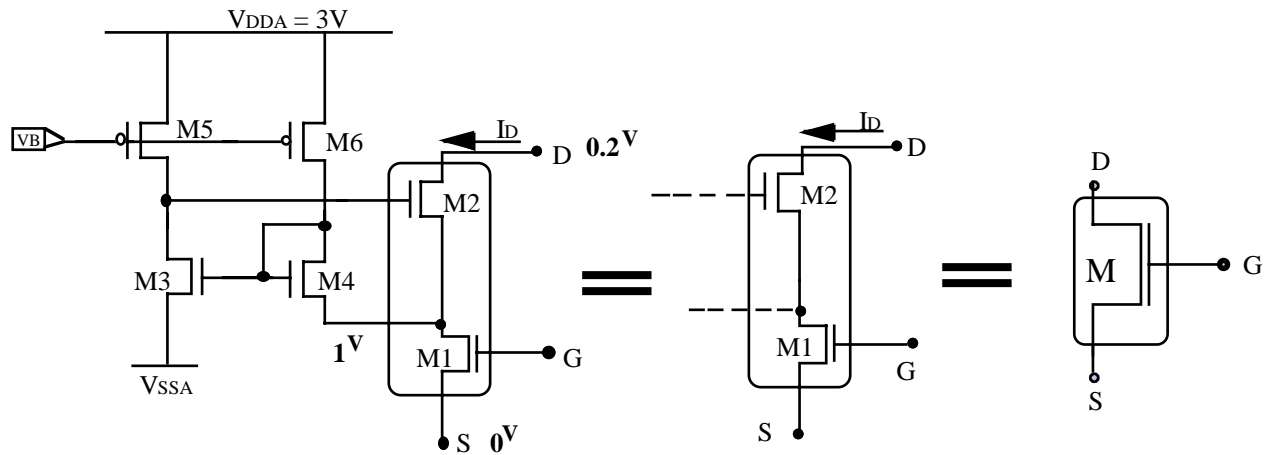


Fig. 2.26 : Circuit Composite NMOS.

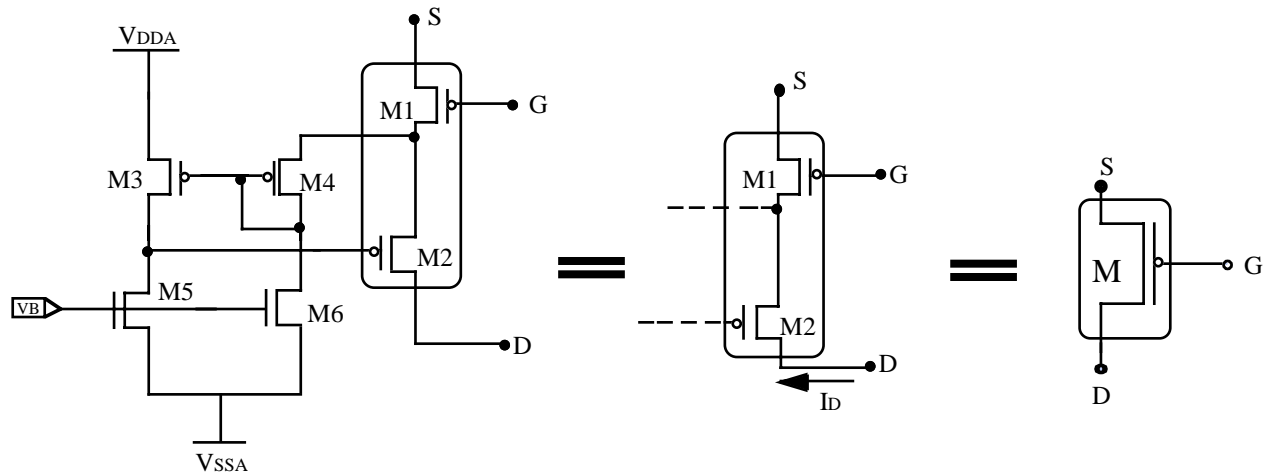


Fig. 2.27 : Circuit Composite PMOS.

Remarque : Une variante très proche du circuit composite a été développée [35][36][37][38]. Elle présente l'avantage d'être auto-polarisée (pas de polarisation supplémentaire).

2.7. APPLICATION DE LA METHODE DES TRANSISTORS COMPOSITES

La technique des transistors composites a été appliquée à un cas très concret d'amplificateur opérationnel CMOS pour lequel l'utilisation d'une nouvelle technologie ne

permettait plus de satisfaire le cahier des charges initial, surtout en ce qui concernait la contrainte de la dynamique de sortie de l'amplificateur. La figure 2.28 représente le schéma de l'amplificateur opérationnel SH3 qui a été utilisé dans le montage échantillonneur-bloqueur (Fig. 2.8).

Ce schéma SH3 appartient à la famille "FOLDED_OTA", avec une paire différentielle d'entrée à transistors PMOS (\Rightarrow une tension de bruit plus faible). Le miroir de courant simple a été remplacé par un miroir de courant de Wilson amélioré, transistors (M8-M9-M10-M11). De plus, un étage de sortie (étage de puissance) de type classe AB a été rajouté ; cela permet d'avoir une faible impédance de sortie et donc de pouvoir commander une capacité de charge relativement élevée (25pF). Enfin, deux étages suiveurs, I_3-I_4 et I_6-I_7, sont utilisés en translateur de tension pour polariser les grilles des transistors de sortie I_2 et I_5, de façon à faire passer un courant de repos minimal en classe A dans la branche de sortie.

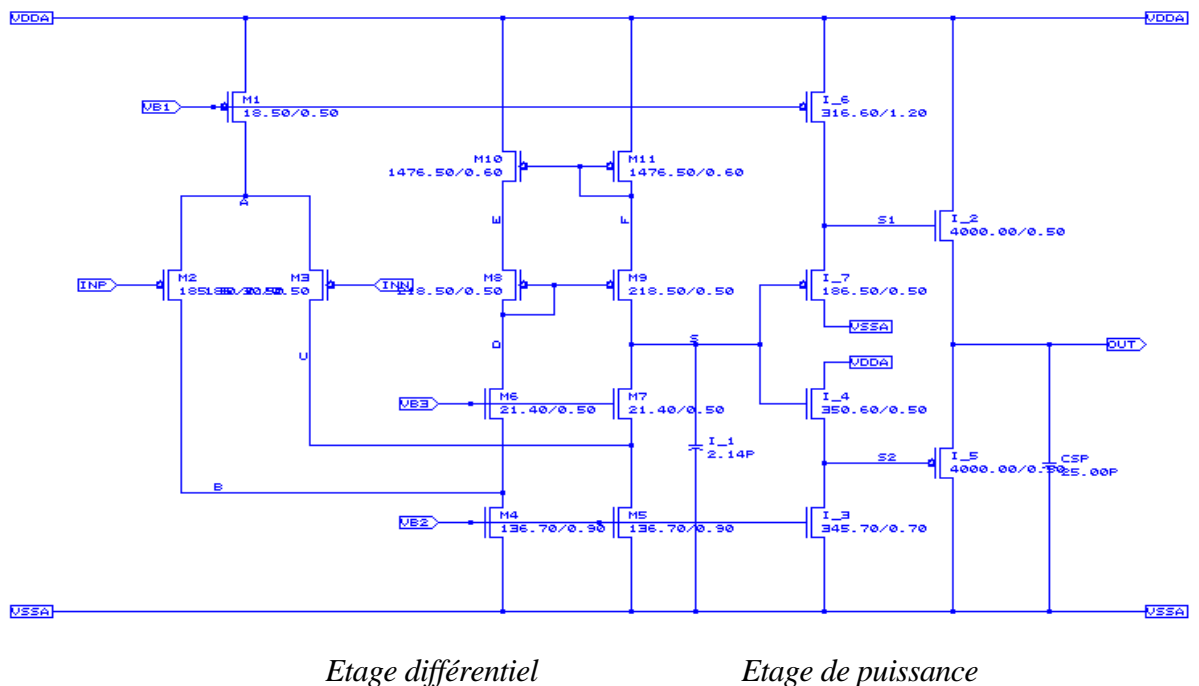


Fig. 2.28 : Schéma initial SH3, amplificateur opérationnel classe AB.

Initialement, ce schéma a été conçu et optimisé pour la technologie Cns1t (longueur de canal $L_{\min} = 1 \mu\text{m}$ et $V_{DD} = 5 \text{ V}$ (cf. annexe A4), pour une capacité de charge $C_L = 25 \text{ pF}$ et en fonction du cahier des charges suivant :

$$\left\{ \begin{array}{l} A_{vd} \geq 51 \text{ dB} \\ F_t \geq 50 \text{ MHz} \\ M_\phi \geq 55 \text{ deg} \\ DYN_{in} \geq 1.5 \text{ V} \\ DYN_{out} \geq 1.5 \text{ V} \\ SR \geq 83 \text{ V} / \mu\text{s} \\ T_{set} \leq 25 \text{ ns} \text{ pour un chelon de } 1.5 \text{ V} \\ P_{DC} \leq 20 \text{ mW} \\ SURF = \min \end{array} \right. \quad (2.52)$$

Rappelons que l'amplificateur opérationnel SH3 est utilisé dans son contexte, en boucle fermée, et sert d'interface entre un échantillonneur-bloqueur et un CAN (8 bits) de capacité d'entrée 25 pF et de dynamique d'entrée 1.5 V ; dans ces conditions, l'amplificateur SH3 doit pouvoir transmettre un signal d'amplitude 1.5 V (ce qui explique les valeurs de DYN_{in} et DYN_{out} du cahier des charges). Les valeurs demandées pour T_{set} , SR, F_t et G_{vd} ont été explicitées dans la description générale du système. Ce convertisseur analogique-numérique est utilisé dans un circuit d'acquisition vidéo mixte analogique-numérique pour une application visiophonie [39], qui a été présentée en détail précédemment dans le paragraphe de description du système.

La première version du circuit réalisée sous une tension d'alimentation de 5 volts et en technologie 1µm a obtenu les succès attendus.

Lors de la réalisation du même circuit, mais dans les conditions de basse tension d'alimentation (3 Volts), beaucoup de problèmes sont apparus, au niveau de l'optimisation du schéma à l'aide de l'optimiseur Opart [40]. L'amplificateur opérationnel SH3 a atteint difficilement une dynamique de sortie supérieur à 0.8 Volts avec la structure habituelle CMOS "FOLDED_OTA" classe AB. Le tableau 2.1 présente les résultats de mesure et de simulation du même amplificateur SH3 dans les deux différents cas de tension d'alimentation (5 V et 3 V).

Résultats de mesures de circuits dans les conditions suivantes: $C_{load} = 25 \text{ pF}$, $Temp = 27 \text{ }^\circ\text{C}$		
Tensions d'alimentation	5 Volts	3 Volts
Gain-dc	55.38 db	61.43 db
Dynamique de sortie	2.1 V	0.8 V
Fréquence de transition	46 MHz	81 MHz
Marge de Phase	77.47 deg	62.2 deg

Slew-Rate	0.105 V/ns	0.1 V/ns
Courant totale consommé	2.1 mA	4.67 mA
CMRR	87.9 db	89.5 db

Tableau 2.1 : Tableau comparatif des résultats de l'amplificateur à deux valeurs différentes de tension d'alimentation.

La dynamique de sortie de l'amplificateur opérationnel SH3 (Fig. 2.28) au condition suivante (longueur de canal $L_{\min} = 0.5 \mu\text{m}$ et $V_{DD} = 3 \text{ V}$) est :

$$DYN_{out_{\max}} = 0.8 \text{ V} \quad (2.53)$$

Une analyse statique permet de comprendre ce résultat :

$$DYN_{out} = (V_{DD} - V_{SS}) - ((V_{DS})_{M16} + (V_{DS})_{M17}) \quad (2.54)$$

Or les grilles des transistors I_2 et I_5 dépendent chacune de la branche précédente. Prenons comme exemple le transistor I_2 ; si on pose ΔV , la tension drain-source minimale nécessaire pour saturer le transistor I_6, on a $(V_G)_{I_2} = V_{DD} - \Delta V$. Pour que le transistor I_2 conduise, on doit avoir $(V_{GS})_{I_2_{\min}} = V_T + \Delta V$; on en déduit la tension de polarisation maximale pour la source du transistor I_2 :

$$(V_S)_{I_2} = V_{DD} - (V_T + 2\Delta V) \quad (2.55)$$

En raisonnant de la même manière, on trouve :

$$(V_S)_{I_5} = V_T + 2\Delta V + V_{SS} \quad (2.56)$$

On en déduit la dynamique de sortie maximale :

$$DYN_{out_{\max}} = (V_{DD} - V_{SS}) - (2V_T + 4\Delta V) \quad (2.57)$$

Pour $V_{DD} - V_{SS} = 3V$, $V_T = 0.7V$ et $\Delta V = 0.2V$, on retrouve les résultats de l'optimisation (2.40).

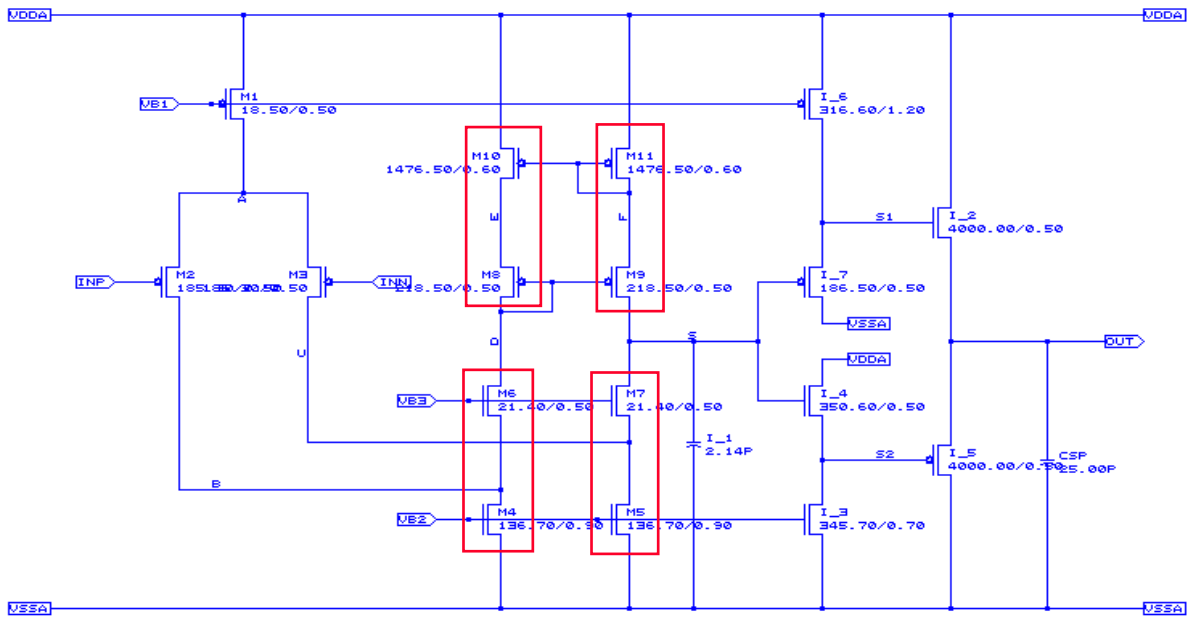
Pour résoudre ce problème, un nouveau schéma d'amplificateur opérationnel CMOS a été conçu (Fig. 2.31); il contient plusieurs circuits composites :

- Premièrement, on a enlevé l'étage de puissance puisque c'est un des facteurs de limitation de la dynamique de sortie de l'amplificateur opérationnel.

- Deuxièmement, les deux circuits cascades repliés M4-M6 et M5-M7 (Fig. 2.28), ont été modifiés ; ils ont été transformés en structure de transistors composites NMOS (Fig. 2.26) en rajoutant les deux autres étages: l'étage de gain, formé des transistors MCP1 et MCN1, et l'étage de décaleur de niveau, formé des transistors MCP2 et MCN2.

- Troisièmement, les deux circuits cascades repliés M8-M10 et M9-M11 (Fig. 2.28), ont été modifiés ; ils ont été complétés et transformés en structure de transistors composites PMOS (Fig. 2.27) en rajoutant les deux autres étages: l'étage de gain, formé des transistors MCP1 et MCN1, et l'étage de décaleur de niveau, formé des transistors MCP2 et MCN2.

Le schéma de l'amplificateur opérationnel CMOS "FOLDED_OTA" Classe AB de la figure 2.29, montre que SH3 a finalement été complètement transformé en une nouvelle structure d'amplificateur opérationnel CMOS Composite (Fig. 2.31), plus adaptée aux conditions de basse tension d'alimentation et de faible consommation. Cette nouvelle structure est purement différentielle; elle est constituée d'un étage d'entrée (paire différentielle PMOS), et de deux étages de sortie symétriques en structure de transistors composites.



Etage différentiel

Etage de puissance

Fig. 2.30 : Schéma initial SH3, Amplificateur Opérationnel Classe AB.

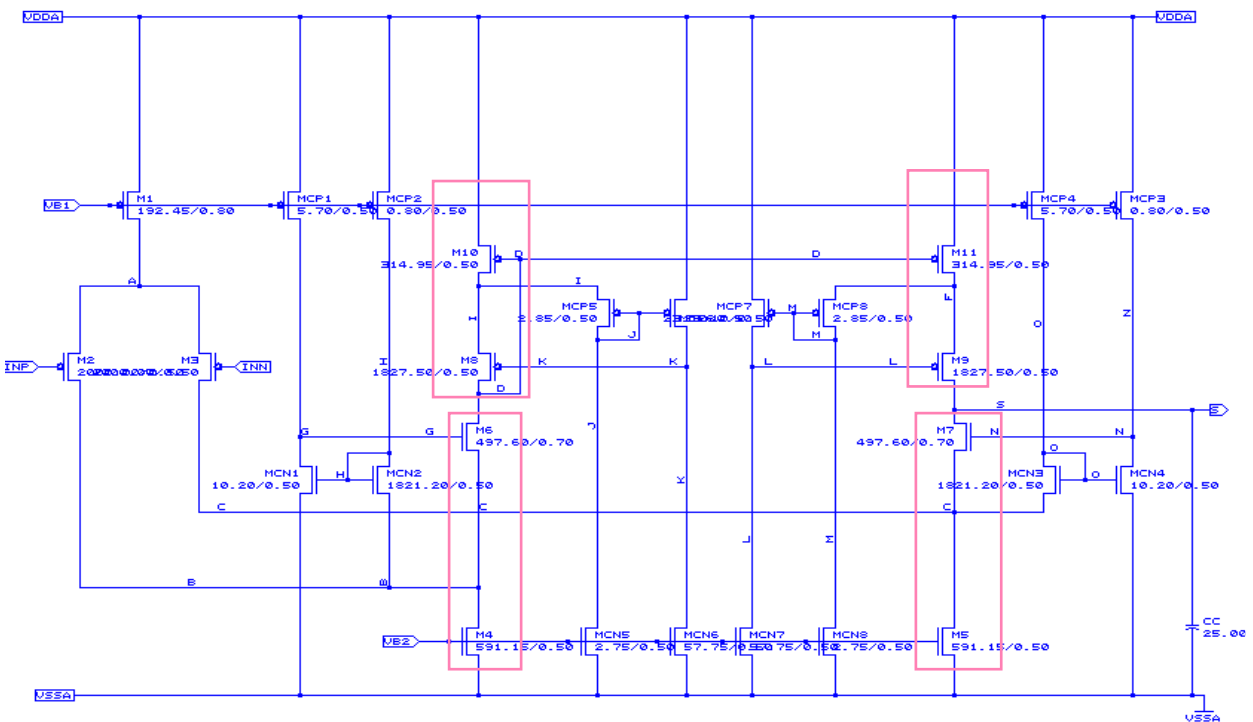


Fig. 2.31 : Schéma de l'amplificateur opérationnel à Transistors Composite final.

Après optimisation de ce nouveau schéma, on obtient des performances qui satisfont entièrement au cahier des charges initial. La plus grande amélioration obtenue concerne la dynamique de sortie :

$$DYN_{out_{max}} = 2.12 \text{ V} \quad (2.58)$$

alors qu'initialement, la dynamique était limitée à 0.8 V.

Une analyse statique permet de comprendre ce résultat :

$$DYN_{out} = (V_{DD} - V_{SS}) - \left((V_{DS})_{M5} + (V_{DS})_{M7} + (V_{DS})_{M9} + (V_{DS})_{M11} \right) \quad (2.59)$$

$$\Rightarrow DYN_{out} = (V_{DD} - V_{SS}) - 4\Delta V \quad (2.60)$$

Pour $V_{DD} - V_{SS} = 3 \text{ V}$ et $\Delta V = 0.3 \text{ V}$, on retrouve les résultats de l'optimisation.

En ce qui concerne les cibles transitoires (vitesse de balayage et temps d'établissement), l'utilisation de circuits composites permet d'éviter l'emploi d'un étage de sortie classe AB. Analysons ce résultat :

$$\text{le gain} \left(A_{vd} \propto \frac{1}{\sqrt{I_{DS}(M1)}} \right) \text{ et la vitesse de balayage } SR \propto I_{DS}(M5, M11)$$

où $I_{DS}(M1)$: courant circulant dans la paire différentielle d'entrée,

$I_{DS}(M5, M11)$: courant circulant dans la branche de sortie.

Dans ce type d'amplificateur, on peut faire l'approximation suivant : $I_{DS}(M1) \approx I_{DS}(M5, M11)$; on s'aperçoit ainsi qu'une augmentation de la vitesse de balayage entraîne automatiquement une diminution du gain.

Pour le schéma de la figure 2.30 sans étage de sortie classe AB, son gain intrinsèque $A_{vd} \propto (g_m \cdot r_{ds})^2$ n'était pas assez élevé ; en effet, satisfaire au cahier des charges pour la vitesse de balayage et le temps d'établissement, sous-entendait une diminution de la valeur du gain qui ne satisfaisait plus à son tour au cahier des charges (à savoir 51 dB) ; cela a nécessité l'utilisation d'un étage de sortie classe AB.

Avec le schéma de la figure 2.31, l'utilisation des circuits composites permet d'obtenir un gain intrinsèque plus élevé $A_{vd} \propto (g_m \cdot r_{ds})^3$; il est donc possible de diminuer le gain tout en satisfaisant au cahier des charges pour les cibles transitoires.

L'utilisation de circuits composites a dégradé, sans conséquence, deux caractéristiques :

- La **marge de phase** : $M_\phi = 65.3^\circ$ au lieu des 77.47° générés par la structure SH3 (Fig. 2.30) sous une tension d'alimentation de 5 volts. Néanmoins, elle reste toujours supérieure à ce qui est demandé dans le cahier des charges (Eq. 2.52). En effet, le circuit composite génère des pôles secondaires supplémentaires que nous avons volontairement négligés dans l'élaboration des formules mais qui influencent pourtant la marge de phase.

- La **dissipation de puissance** : $P_{DC} = 15.93mW$ au lieu de 10.5 mW (pour la structure SH3, Fig. 2.30 sous 5 volts), ou même 14 mW (pour la structure SH3, Fig. 2.30 sous 3volts). Néanmoins, elle reste toujours inférieure à ce qui est demandé dans le cahier des charges (Eq. 2.52). En effet, chaque circuit composite introduit 4 transistors supplémentaires par rapport à la structure classique, ce qui naturellement augmente la dissipation de puissance. Mais les dimensions des transistors supplémentaires sont très faibles, ce qui permet de ne pas générer une consommation de puissance trop importante. On a donc pu respecter la marge donnée par le cahier des charges initial.

Le tableau 2.2 permet de comparer les performances électriques des schémas de la figure 2.30 et de la figure 2.31, obtenus après optimisation, en réponse au cahier des charges (Eq. (2.39)) et pour la technologie Cc05 ($L_{min} = 0.5 \mu m$ et $V_{DD} = 3 V$). Ces résultats ont été obtenus à l'aide de l'optimiseur Opart.

Résultats de mesures de circuits dans les conditions suivantes: $V_{supply} = 3 \text{ Volts}$, $C_{load} = 25 \text{ pF}$, $Temp = 27 \text{ }^\circ\text{C}$		
Amplificateurs Opérationnels	Class-AB	Amp-Op à Transistors Composites
Gain-dc	61.43 db	67.3 db
Dynamique de sortie	0.8 V	2.12 V
Fréquence de transition	81 MHz	116 MHz
Marge de Phase	62.2 deg	65.3 deg
Slew-Rate	0.1 V/ns	0.16 V/ns
Courant total consommé	4,67 mA	5.31 mA
Temps d'établissement	25 ns	25 ns

Distortion à (10 kHz)	-50 db	-80 db
Surface totale	45158 μm^2	34026 μm^2

Tableau 2.2 : Tableau comparatif des performances des deux schémas d'amplificateur soumis aux mêmes conditions de conception.

Le tableau 2.2 ne permet pas de conclure sur l'apport du circuit composite pour la fréquence de transition ; en effet, les deux valeurs de la fréquence de transition sont données pour une marge de phase différente.

2.8. CONCLUSION

Les résultats obtenus en insérant des circuits composites dans le schéma traditionnel de la figure 2.30 sont extrêmement intéressants: en augmentant de manière importante le gain intrinsèque de l'étage d'amplification, les circuits composites permettent de s'affranchir d'un étage de sortie classe AB en source suiveuse, étage qui était responsable d'une faible dynamique de sortie. Faisons une synthèse des résultats obtenus :

Schéma traditionnel (Fig. 2.30)
(sans étage de sortie)

Schéma amélioré (Fig. 2. 31)
(circuit composite)

$$A_{vd} \approx (g_m \cdot r_{ds})^2 \text{ (d'après Eq. 2.32)} \quad \text{et} \quad A_{vd} \approx (g_m \cdot r_{ds})^3 \text{ (d'après Eq. 2.45).}$$

Puisque $A_{vd} \propto \frac{1}{\sqrt{I_{DS}}}$, on a:

$$(g_m \cdot r_{ds})^2 \propto \frac{1}{I_{DS}^{1/2}} \quad \text{et} \quad (g_m \cdot r_{ds})^2 \propto \frac{1}{I_{DS}^{1/3}}$$

où I_{DS} : le courant dans la branche de sortie.

On constate que, pour le schéma amélioré, le gain diminue moins lorsque le courant augmente. A gain équivalent, le courant dans la branche de sortie est plus élevé pour le schéma de la figure 2.31 que pour le schéma de la figure 2.30 (sans étage de sortie).

Pour avoir une vitesse de balayage équivalente ($SR \propto I_{DS}$), on doit rajouter un étage de sortie classe AB en source suiveuse (Fig. 2.30).

Dans ce chapitre, nous avons présenté les raisons qui ont conduit à la conception électrique du nouveau schéma (Fig. 2.31), puis nous avons interprété les résultats de simulation obtenus. Bien que l'idée d'augmenter le gain intrinsèque d'un étage d'amplification ne soit pas nouvelle [30], il semble que ses applications deviennent de plus en plus nécessaires aujourd'hui, avec l'arrivée de nouvelles technologies utilisant des tensions d'alimentation inférieures à 5 Volts. Cette étude a permis de générer une méthode de conception de nouvelle structure, plus adaptées aux basses tensions d'alimentation et aux faibles consommations. Cette méthode a eu un intérêt assez important dans le monde de la conception des circuits intégrés analogiques, puisque, suite à sa présentation à la conférence ESSCIRC'95 [34], elle a reçu le prix de la meilleure publication de la conférence.

Le circuit composite peut aussi être utilisé comme générateur de courant d'une paire différentielle d'entrée ; le générateur de courant ainsi réalisé possède une résistance bien supérieure à un générateur de courant classique utilisant un transistor unique. Dans le cas d'un amplificateur opérationnel, ceci permet d'améliorer le rapport de réjection en mode commun (*CMRR*).

Il apparaît cependant que l'utilisation de circuit composite soit limitée vers les hautes fréquences ; en effet, le circuit composite introduit des noeuds supplémentaires donc des pôles secondaires que l'on a pu négliger dans notre application. Par contre, pour des fréquences supérieures, la marge de phase risque de ne plus être suffisante pour assurer la stabilité du montage.

DEUXIEME PARTIE

*Méthodologie d'Automatisation du Dessin
des Masques des Circuits Intégrés
Analogiques et Mixtes: CHIRVAN*

Chapitre III

Chapitre 3

Contraintes de Conception des Circuits Intégrés Analogiques

3.1. INTRODUCTION

L'automatisation du dessin des masques des circuits intégrés analogiques s'est considérablement améliorée ces dernières années, en dépit de la complexité croissante des circuits analogiques et mixtes (analogiques/numériques).

Durant les années 80 et le début des années 90, de nombreuses recherches ont été consacrées au développement des outils de génération automatique (ou assistée) du layout des circuits intégrés analogiques à l'aide d'algorithmes et de logiciels appropriés. Ces recherches ont été conduites essentiellement par des universitaires. Citons par exemple les outils (KOAN et ANAGRAM) [43][48][53][63][96] développés par l'université de Carnegie Mellon, l'outil (SALIM,...) [42][92][93] [94][95] développé par l'EPFL, les logiciels et technique de placement et de routage développés par l'université de Berkeley [45][60][70], ceux développés par l'université de Leuven [85][86][87][88], et finalement les outils (ELDO, OPART, CHIRVAN) [40][46][47][55] [56][57][56][73] développés par le CNET. Toutes ces travaux représentent l'état de l'art en la matière. Ces études ont essayé de faire intervenir la plupart des contraintes analogiques pour réaliser des circuits intégrés analogiques de bonnes qualités. Néanmoins, les contraintes analogiques ne sont pas toutes connues en détail par les développeurs de logiciel, et souvent difficilement gérable par les algorithmes proposés. Ceci explique en partie l'absence de leur utilisation par les industriels ainsi que l'absence de

commercialisation par les vendeurs des logiciels de C.A.O comme Cadence ou Mentor-Graphic...

L'étude faite en première partie nous a permis de mieux comprendre les contraintes analogiques et leur effet sur les performances des circuits intégrés analogiques. Ce chapitre va donc se consacrer à bien définir ces contraintes.

Le comportement électrique des circuits intégrés analogiques est fortement influencé par divers phénomènes parasites. Il est impossible de fournir une liste complète de tous les effets parasites entrant en ligne de compte. Car d'une part, ces effets dépendent du circuit à réaliser, et d'autre part, ils sont de natures diverses (effets parasites internes à un transistor, dus à la proximité d'éléments bruyants, dus à des phénomènes électrothermiques...). Néanmoins, nous allons donner une liste des phénomènes parasites les plus répandus, ceux susceptibles d'être les plus souvent rencontrés. Cet inventaire nous permettra ensuite de déduire les contraintes à prendre en compte lors de la phase du dessin des masques, et tout particulièrement lors des étapes de placement, de routage et de compaction.

3.2. PRINCIPAUX PHENOMENES PARASITES RENCONTRES DANS UN CIRCUIT INTEGRE ANALOGIQUE

3.2.1. Capacités parasites

On trouve des capacités parasites à divers niveaux du layout. Tout d'abord, il existe des capacités parasites internes aux cellules élémentaires. Par exemple, si l'on observe le layout des transistors présentés à la figure 3.1, on constate que certaines pistes de métal, appartenant au drain ou à la source, croisent les pistes de polysilicium de grille. Etant séparées par un oxyde, ces deux couches technologiques, l'une en métal et l'autre en polysilicium, introduisent de faibles capacités parasites entre les noeuds du drain ou de la source, et le noeud de la grille.

De même, pour une technologie à deux ou plusieurs niveaux de métallisation, le croisement de deux pistes métalliques, elles aussi séparées par un oxyde, entraîne un couplage capacitif des noeuds impliqués. Le phénomène a lieu de la même façon pour deux pistes d'une même couche technologique qui se suivent en étant proches.

Enfin, il existe des capacités parasites de jonctions classiques (transistors bipolaires) et des capacités situées entre les pistes de routage (en métal ou en polysilicium) et le substrat (exemple : la capacité dite de grille pour le transistor MOS). Le modèle de transistors MOS représenté à la figure 3.2, montre ces capacités parasites générées par le transistor lui même, et représentées par le modèle de SCHICHMAN et HODEGS.

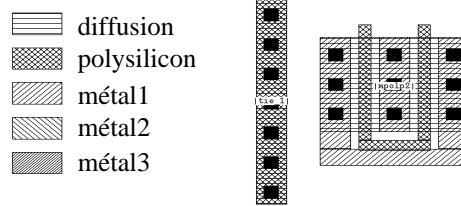
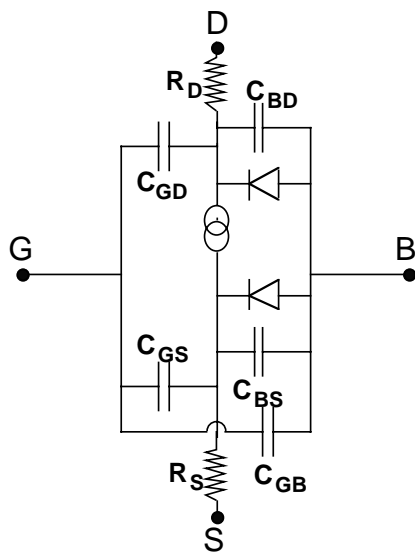
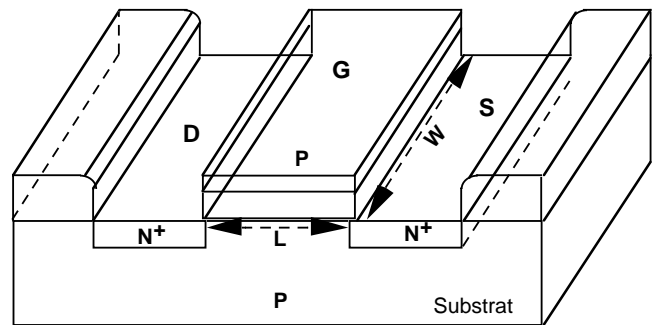


Fig. 3.1 : Layout du transistor MOS



MODELE DE SCHICHMAN et HODEGS



TRANSISTOR MOS

Fig. 3.2 : Représentation des différents éléments parasites internes du MOS.

3.2.2. Résistances parasites

Comme dans le cas des capacités, on trouve des résistances parasites internes aux composants eux-mêmes. Ce sont d'une part les résistances d'accès à la base, à l'émetteur ou au collecteur (transistor bipolaire), d'autre part, les résistances d'accès à la grille, au drain ou à la source (transistor MOS). De plus, le métal et le polysilicium (parfois utilisés comme piste de routage) ont une résistivité non négligeable, qui se traduit, si les pistes sont longues (exemple : les pistes d'alimentations), par des résistances dont il faut tenir compte dans le schéma électrique final.

3.2.3. Inductances parasites

Les pistes de métallisation internes aux transistors ou utilisées pour le routage conduisent à des inductances parasites. Ce phénomène est généralement négligeable mais il peut prendre de l'importance, notamment pour les circuits intégrés travaillant à haute fréquence (en technologie CMOS-analogique, de l'ordre de 50 à 60 MHz; en technologie Bipolaire ou BICMOS, de l'ordre de 1 à 2 GHz, pour les circuits des GSM par exemple) et (ou) travaillant dans un environnement électromagnétique fortement perturbé.

3.2.4. Composants parasites actifs

Dans tout transistor MOS intégré, on rencontre des transistors bipolaires parasites. Ils sont à l'origine des phénomènes de "**latch up**", comme le montre la figure 3.3 [49]. De même, dans une technologie bipolaire, la présence de pistes métalliques et de diffusions de même type, séparées par l'oxyde de champ, engendre une structure MOS parasite. Dans un layout de circuit intégré analogique complet, ces éléments parasites actifs se connectent entre eux par l'intermédiaire des éléments parasites passifs présentés précédemment, en particulier par le substrat. On met ainsi en évidence des réseaux parasites qui, sous certaines conditions de polarisation, se mettent en conduction et perturbent fortement le fonctionnement du circuit [49][50].

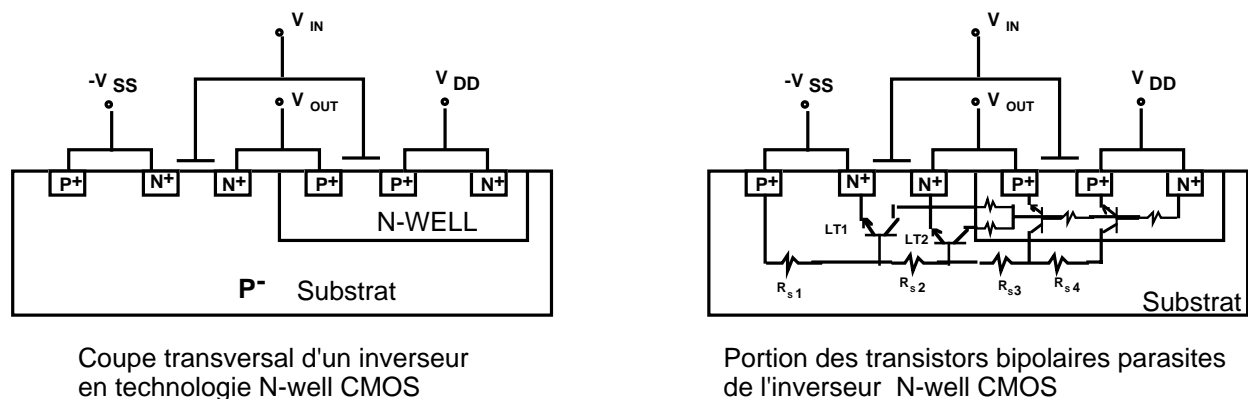


Fig. 3.3 : Représentation des phénomènes de "**latch up**" des transistors MOS.

3.2.5. Appairage de composants (Matching)

L'une des contraintes les plus importantes à considérer dans la conception des circuits analogiques, est l'appairage des composants.

Les circuits électriques analogiques sont souvent basés sur la connexion de réseaux devant être parfaitement symétriques. L'exemple classique est celui d'un circuit différentiel. L'obtention d'un appairage parfait entre les branches du circuit est l'une des principales difficultés rencontrées lors du layout.

L'appairage, c'est l'uniformité du dopage des transistors pendant la phase de fabrication (à la fonderie). Il est difficile de l'obtenir pour toute la plaque de silicium. C'est pourquoi on accorde une attention particulière aux composants qui nécessitent cette uniformité (les paires différentielles, les transistors de charges d'un amplificateur différentiel...) .

Cette contrainte analogique du layout concerne les composants et l'appairage des parasites. Les variations inévitables, présentes dans tous les processus menés jusqu'ici, viennent du faible non-appairage des caractéristiques électriques de deux composants identiques. Si ce non-appairage est assez grand, il peut avoir un effet sur les performances des circuits en introduisant des problèmes électriques comme l'offset.

Les trois facteurs majeurs du layout qui peuvent effectuer un bon appairage (matching) des composants identiques sont : la **Forme**, l'**Orientation** et la **Séparation**.

La création des composants utilisant des géométries identiques améliore l'appairage et assure que les mêmes objets soient sujet à la même distorsion géométrique.

3.2.6. Interactions électrothermiques

Les caractéristiques électriques des composants intégrés sont fortement influencées par la température. Si la puissance dissipée par certains circuits intégrés est relativement faible, elle est forte dans les circuits intégrés en technologie Bipolaires et BICMOS, en raison de forts gradients thermiques apparaissant en surface de la puce de silicium, et surtout à basse tension (3 Volts). Ainsi pour minimiser les interactions électrothermiques, les composants doivent être placés judicieusement. On dispose par exemple, les composants vulnérables à l'effet de la température (les paires différentielles, les transistors de charges,...) loin des composants qui dissipent une forte température (les sources de courants,...). Certains circuits intégrés à haute densité d'intégration subissent eux aussi des effets de température, mais de faible intensité. Le nombre élevé de ces derniers entraîne une puissance moyenne dissipée par les circuits intégrés non négligeable.

3.2.7. Incertitude sur les paramètres de la technologie

Le processus technologique n'est jamais parfaitement stable. Ceci se traduit par une incertitude, parfois non négligeable, sur la valeur des caractéristiques électriques des composants intégrés, et par des différences de comportement entre deux éléments identiques se

trouvant sur la même puce. C'est particulièrement gênant lors de l'appairage de composants et de sous-circuits. Nous ne sommes malheureusement pas maître de ce phénomène parasite au niveau layout. Il ne peut être pris en compte que de manière statistique.

Le changement de technologie a aussi un effet fâcheux, puisqu'il nous pousse toujours à créer de nouveaux générateurs (transistors MOS, paires différentielles, résistances, capacités...). Les anciens générateurs ne sont plus adaptés à ces nouvelles technologies, ce qui est très lourd à gérer.

3.3. CONTRAINTES A APPLIQUER LORS DE LA CONCEPTION D'UN CIRCUIT INTEGRE ANALOGIQUE

3.3.1. Contraintes sur les performances attendues du circuit intégré analogique

Les contraintes de performances sont toujours données par les concepteurs eux-mêmes suivant un cahier des charges bien défini. Ces contraintes sont obtenues grâce à la simulation et l'optimisation des circuits électriques (schémas électriques) choisis par les concepteurs pour remplir une fonction donnée. Exemple de fonctions : "transmission de signal, conversion (analogique/numérique, numérique/analogique), convertisseur delta-sigma, amplification, échantillonnage..."

Ces contraintes de performances sont primordiales. Elles sont le but de la réalisation d'un circuit intégré, et sont prises comme base de conception. Il faut donc que les logiciels de conception gardent toujours en vue ces objectifs, pendant toutes les phases de la conception et du dessin du masque, (placement, routage, et compaction). Les performances du circuit ne doivent, à aucunes de ces phases, subir des pertes ou être influencées. Dans le cas contraire la fonctionnalité du circuit sera remise en cause.

Un bon outil d'automatisation de dessin du layout respecte toujours ces contraintes de performances. Cet outil doit prendre en compte les contraintes de conception des circuits intégrés analogiques, énumérées précédemment.

Chaque circuit analogique a des contraintes de performances spécifiques à sa fonctionnalité. Sachant que les amplificateurs sont les éléments de base des circuits analogiques, nous donnons, à titre d'exemple, les contraintes de performances qui leur sont liées et dont nous avons donné les détails dans l'annexe A1:

- le gain de l'amplificateur,
- la marge de phase (déphasage entre le signal de sortie et le signal d'entrée),

Finalement, si tous les circuits différentiels étaient complètement symétriques, c'est-à-dire, si chaque composant avait son symétrique, le problème de layout serait beaucoup plus simple. Dans ce cas, il suffirait de dessiner seulement la moitié du circuit différentiel. L'autre moitié pourrait être formée en pliant la première moitié du layout par rapport à l'axe de symétrie global du circuit.

Malheureusement, dans la plupart des circuits analogiques, si une partie est symétrique, l'autre est complètement asymétrique, (exemple: l'amplificateur folded-cascod-OTA, figure 3.5). Cette asymétrie entraîne l'impossibilité d'appliquer le raisonnement précédemment développé.

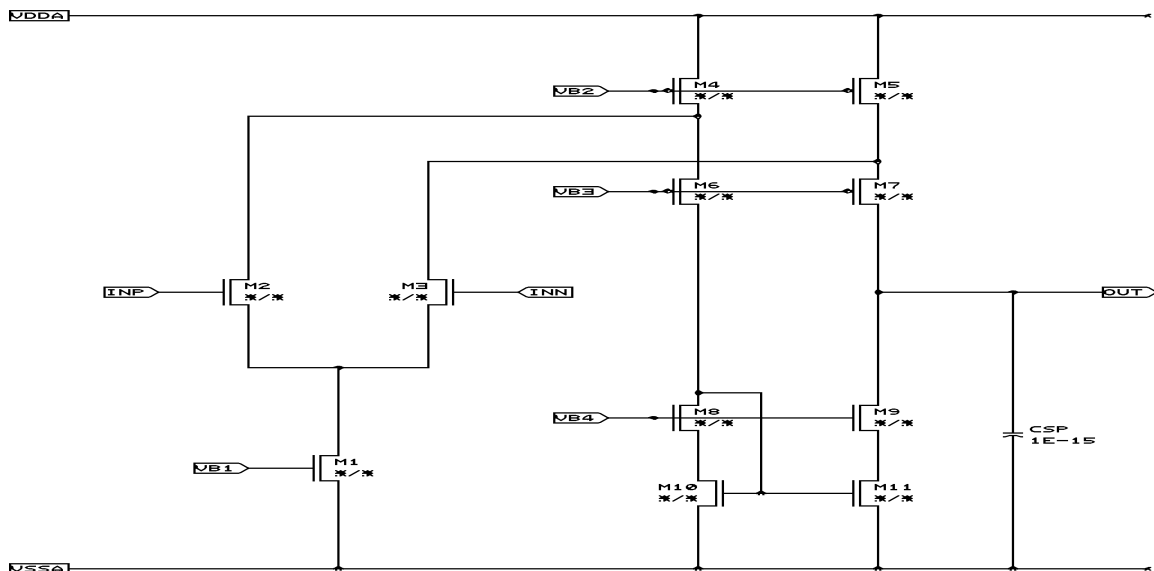


Fig. 3.5 : Amplificateur folded-cascod-OTA.

3.3.2.2. Minimisation de la longueur des interconnexions

Sur le layout, un noeud du schéma électrique se traduit par une piste, soit en métal, soit en polysilicium, connectant tous les equipotentielles "pins" des cellules élémentaires contenant le noeud. Plus la longueur de la piste est longue, plus sa résistance et sa capacité parasite sont élevées. La minimisation des interconnexions est donc une contrainte à considérer.

Un des procédés pour minimiser les longueurs d'interconnexions, consiste à **fusionner** tous les transistors qui peuvent l'être, c'est-à-dire les transistors possédant un noeud commun et une diffusion semblable. Ceci permet de réduire à la fois les capacités parasites générées par les fils de connexions entre les transistors, et la surface totale du circuit.

3.3.2.2. Minimisation de la surface totale du circuit

Une des contraintes classiques de tous circuits intégrés, qu'il s'agisse de circuits intégrés analogiques ou numériques, est la minimisation de la surface totale du circuit. Cette contrainte est davantage liée à des problèmes économiques (coût de fabrication), plutôt qu'à des problèmes électriques ou fonctionnels. Il est donc important pour tout logiciel de conception de prendre en

compte cette contrainte sans pour autant entraver celles à caractères fonctionnelles ou électriques, c'est à dire les plus importantes.

3.3.2.3. *Mise hors-service des réseaux parasites*

Eviter la mise en conduction des réseaux parasites dus à la connexion, généralement par le substrat, des composants parasites actifs et passifs, demande parfois la mise en place d'isolement basés sur la polarisation en inverse de la jonction PN. D'autre part, une bonne polarisation de substrat, grâce à la présence de contacts de "bulk" régulièrement répartis, est requise. Il s'agit donc de mettre en place des **prises** d'alimentation du substrat.

3.3.3. Contraintes de Placement

3.3.3.1. *Symétrie*

La première et la plus importante des contraintes analogiques est bien sûr la contrainte de symétrie. Quand un concepteur réalise un circuit analogique différentiel, il suppose que le layout va introduire des effets parasites identiques dans les deux moitiés du circuit, celles-ci étant symétriques. Rappelons que chaque composant et chaque fil d'interconnexion génèrent des résistances et des capacités parasites. L'absence de placement symétrique et de bon appairage de ces effets parasites, peut induire une augmentation de la tension d'offset, ainsi qu'une dégradation du rapport de réjection d'alimentation ("power supply rejection ratio" noté PSRR) [51].

Un placement et un routage symétriques par rapport à un axe de symétrie globale sont la meilleure façon d'équilibrer ces effets parasites. Un placement symétrique réduit aussi la sensibilité des circuits analogiques au gradient de la température. Ceci est particulièrement intéressant pour les circuits bipolaires et BICMOS, très vulnérables à l'effet de la température.

3.3.3.2. *L'appairage (Matching)*

L'appairage de composants ou de branches d'un circuit intégré analogique est difficile à obtenir pour deux raisons. La première vient de l'incertitude sur les paramètres de la technologie. Le concepteur n'a aucun pouvoir sur ce phénomène parasite si ce n'est de positionner les composants à appairer aussi proche que possible les uns des autres en espérant que l'uniformité des dopages sur la surface considérée sera bonne. La deuxième raison provient du dessin des masques qui doit être identique pour tous les éléments à appairer. Elle est maîtrisable par le concepteur à condition qu'il prenne certaines précautions. D'une part, les layouts des cellules élémentaires à appairer doivent avoir la même forme, et être orientées selon

la même direction lors du placement. D'autre part, les éléments d'environnement qui conditionnent les phénomènes parasites pouvant influencer le comportement électrique du circuit doivent être éliminés ou identiques d'une cellule à l'autre. Ceci se traduit au niveau du placement par des bruits important. Enfin, au niveau du routage, l'appairage se manifeste par une attention toute particulière accordée à l'équilibrage des éléments parasites (capacités et résistances) présents sur chaque noeud des branches du circuit considéré. Par exemple, si pour un noeud n1, la piste métallique P1 le matérialisant croise N fois une autre piste P, la piste P2 matérialisant n2, noeud symétrie de n1 dans le schéma électrique, devra croiser, elle aussi N fois la piste P pour équilibrer les capacités parasites. Enfin les longueurs de P1, P2 devront être aussi proches que possible pour équilibrer les résistance parasites.

3.3.3.3. Influence électrothermique

L'influence électrothermique sur les circuits analogiques est une contrainte qui reste plus ou moins réservée à des circuits réalisés en technologie Bipolaire ou BICMOS. En effet, les transistors bipolaires sont particulièrement sensible à l'effet de la température. Si deux transistors sont placés dans un gradient de température non approprié, la différence de température peut produire un effet de non-appairage. Donc à défaut d'une adéquation thermique dans un circuit différentiel, cela peut introduire des oscillations de signal [52]. Un placement symétrique, peut réduire la sensibilité de ces circuits analogiques au gradient de la température.

3.3.3.4. Elargissement de la surface de routage

Un placement automatique doit toujours prendre en compte la phase de routage qui va suivre. Autrement dit, le placeur doit laisser un espace assez grand entre les composants, pour que les fils de routage puissent passer pour connecter les transistors entre eux sans difficulté. Cet espace est plus ou moins grand, selon le logiciel de routage qui va suivre. Pour le routeur de canal, l'espace de routage entre les composants est moins contraignant que pour le routeur de labyrinthe, car celui-là peut compacter ou décompacter le canal après routage de chaque canal. Le routeur de labyrinthe route sans déplacer les transistors, c'est pour cela qu'il a davantage besoin de place de routage pour pouvoir respecter les autres contraintes comme par exemple, la mise en place des fils symétriques d'un côté et de l'autre de l'axe de symétrie, pour les circuits qui ont une structure symétrique (amplis différentielles). Par contre le grand avantage qu'il a sur le routeur de canal, c'est que les fils de routage peuvent passer sur les transistors, si le concepteur le permet. Cet espace peut être aussi très important si un compacteur réduit ensuite la surface du circuit (en respectant toujours les contraintes analogiques prises en compte par le routeur).

3.3.3.5. *Eloignement des sorties différentielles*

Pour les circuits différentiels possédant aussi des sorties différentielles, il faut toujours éloigner le plus possible les sorties les unes des autres. Ceci pour éviter un couplage capacitif possible qui peut avoir une incidence sur les tensions de sortie, principalement lorsque ces tensions sont en opposition de phase, causant ainsi un offset indésirable.

Topologiquement il faut éloigner les sorties différentielles d'à peu près une dizaine de micromètres les unes des autres, et ceci quel que soit le niveau de métal utilisé pour le routage.

3.3.4. Contraintes de Routage

3.3.4.1. *Symétrie*

Cette contrainte est aussi très importante pendant la phase de routage, pour les mêmes raisons que celles que nous avons décrites dans le paragraphe de symétrie pour le placement (§ 3.3.3.1).

Dans le chapitre 4, nous expliquerons, comment le routeur tient compte de cette contrainte même si parfois le circuit n'est pas complètement symétrique.

3.3.4.2. *Minimisation de la longueur des interconnexions*

Sachant que chaque fil de routage génère des résistances et des capacités parasites, plus au moins importantes en fonction de sa résistivité et de sa longueur, la contrainte de la minimisation de la longueur d'interconnexion est nécessaire pour réduire les effets capacitifs et résistifs générés par les fils, et ceci pendant la phase de routage comme pendant la phase de compaction. Il faut également router avec des fils moins résistifs (comme métal1, métal2...), plutôt qu'avec des fils plus résistifs (comme le polysilicium).

3.3.4.2. *Découplage des noeuds sensibles*

Un noeud bruyant (exemple : la piste d'alimentation d'un circuit intégré utilisé en commutation, la piste de l'horloge d'un sous-ensemble logique, une piste de sortie branchée sur une charge inductive...) ne doit en aucun cas se retrouver couplé à un noeud sensible (exemple : l'entrée d'un amplificateur). Il est donc essentiel que les pistes concernées ne se croisent ni, si possible, se côtoient. De même, les composants élémentaires impliqués doivent être éloignés les uns des autres.

3.3.4.3. Contrainte sur la largeur des fils (fils à fort flux de courants)

Les largeurs des fils minimales sont données automatiquement par les règles de dessin, et ceci pour chaque technologie. Mais il y a des fils qui nécessitent des largeurs supérieures à celles données par les règles de dessin, ceci pour des raisons qui leurs sont propres, ce que nous expliquerons dans ce paragraphe.

Les fils d'alimentations (V_{DD} , V_{SS}) du circuit engendrent un flux de courant continu très élevé. Il est donc indispensable d'utiliser des largeurs de fils assez importantes.

D'autres branches de circuit font également l'objet d'un flux de courant relativement important, mais ce dernier n'est pas toujours continu. Il faudrait néanmoins élargir moyennement ces fils de routage, pour éviter l'effet de "fusible"(coupure des fils) qui peut survenir après un long fonctionnement. Les noeuds qui subissent un flux de courant élevé sont, par exemple, les noeuds de sortie d'amplificateur ayant une capacité de charge très élevée.

3.3.4.4. Minimisation des capacités parasites (“Cross-talk, Cross-over”)

Le layout peut aussi introduire un couplage de signal inattendu, ou cross-talk, dans un circuit qui pourrait injecter un bruit électrique indésirable ou éventuellement une distorsion de la stabilité à travers une branche de contre réaction (“feed-back”). Ce couplage pourrait être introduit par une combinaison d'effet capacitif, résistif ou thermique [53].

Deux conducteurs peuvent avoir un couplage capacitif dans les deux cas suivant: si les deux conducteurs sont sur des fils différents et se croisent (cross-over), ou si les deux conducteurs ont le même niveau de métal ou un niveau différent, mais restent proches l'un de l'autre (cross-talk). Ces deux cas sont illustrés sur la figure 3.8 (a).

Dans le cas du croisement, le couplage capacitif est proportionnel à la surface de croisement des fils; tandis que dans le cas des fils adjacents, la capacité est proportionnelle à la longueur des surfaces qui se font face. Cependant, ces couplages peuvent être réduits par minimisation des croisements ou par minimisation des longueurs des fils parallèles à signaux incompatibles.

Si la mise en parallèle de deux fils à signaux incompatibles est inévitable, un fil neutre (par exemple, un fil de masse ou une ligne de référence) peut être placé entre eux comme un couplage de protection. La figure 3.8 (b) illustre ce type de protection.

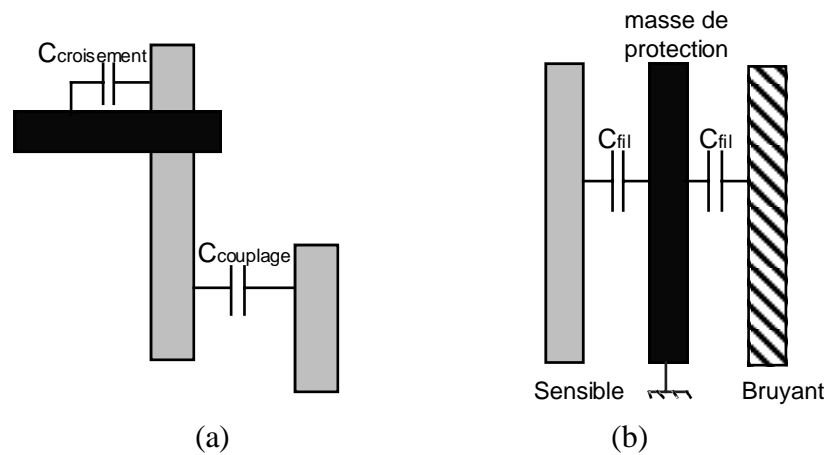


Fig. 3.8 : Layout de l'effet de couplage. (a) couplage capacitife entre les fils de routage.
 (b) réduction du couplage capacitif par utilisation d'un fil de masse de protection.

3.3.5. Contraintes de Compaction

En ce qui concerne le logiciel de compaction, le problème des contraintes est davantage lié au respect des contraintes prises en compte par les différents outils de dessin de masques travaillant en amont (comme le placeur et le routeur), qu'à de nouvelles contraintes. Néanmoins, il y a des contraintes spécifiques au compacteur, qui lui donne son originalité, en particulier la minimisation de la surface totale du circuit, et la réduction des longueurs des fils d'interconnexions.

3.3.5.1. Minimisation de la surface totale du circuit

La principale contrainte du compacteur est bien sûr la réduction de la surface totale du circuit. Ceci est nécessaire comme nous l'avons dit plus haut, pour diminuer le coût de fabrication, plutôt que pour résoudre des problèmes de fonctionnalité des circuits intégrés analogiques.

La compaction est surtout nécessaire après un routage avec le routeur de labyrinthe, puisque ce routeur trace les interconnexions entre les transistors et les autres composants, sans les déplacer.

On sait que chaque composant dans le layout est un objet composé (transistors MOS, paires différentielles, résistances, capacités...), généré par un outil automatique de génération des composants spécifiques à chaque technologie et respectant les règles de dessin par construction. Il est donc inutile de prendre en compte les règles de dessin à l'intérieur des composants. La minimisation de la surface du circuit, se réduit à la minimisation de l'espace extérieur des composants, et à la minimisation des fils d'interconnexions.

Cette contrainte de minimisation de la surface totale du circuit doit se faire tout en respectant les autres contraintes prises en compte auparavant, à savoir: la symétrie, l'appariage, la minimisation des capacités parasites (Cross-talk, Cross-over), etc...

3.4. Conclusion

Nous venons de présenter une liste exhaustive des principaux phénomènes parasites et leur traduction en terme de contraintes. La liste de ces contraintes n'est pas exhaustive car selon le circuit intégré à concevoir, d'autres effets très particuliers peuvent être rencontrés.

Les contraintes sont liées soit au dessin des cellules élémentaire, soit au placement, soit au routage. Nous avons principalement mis l'accent sur les contraintes de l'étape de placement, et les étapes de routage et de compaction. Car ces contraintes nous ont permis par la suite de réaliser un outil de dessin de masque automatique (Génération, Placement, Routage, Compaction) basé sur ces contraintes analogiques: CHIRVAN.

Ces contraintes seront encore davantage développées dans les chapitre 4 et 5, et les détails de leur implémentation dans les outils automatiques de placement, routage et compaction y seront présentés.

Aux contraintes de minimisation des phénomènes parasites s'ajoutent la contrainte classique de minimisation de la surface totale de silicium occupée, contrainte directement liée au coût de fabrication. Ces contraintes sont les mêmes pour tous circuits intégrés, qu'ils soient digitaux ou analogiques.

Chapitre IV

Chapitre 4

Environnement de Dessin des Masques (layout) des Circuits Intégrés Mixtes Analogiques/Numériques

CHIRVAN

4.1. INTRODUCTION

L'étape de dessin des masques, si elle est réalisée manuellement, est sans doute la phase la plus longue et la plus fastidieuse de la conception des circuits intégrés. Il s'agit en effet de réaliser tous les motifs des masques qui seront utilisés lors de la fabrication du circuit. Or, pour assurer le bon fonctionnement de celui-ci, un certain nombre de règles technologiques (typiquement une quarantaine) concernant les dimensions et les espacements de ces motifs doivent être respectées. Pour corser le tout, le concepteur doit de plus s'assurer que les parasites introduits par son dessin ne dégradent pas trop les performances du circuit. Enfin, pour des raisons de coût, il doit aussi faire en sorte que le circuit réalisé soit le plus compact possible de façon à économiser au maximum la quantité de silicium requise. Une étape de vérification après coup reste malgré tout nécessaire pour s'assurer que toutes les règles de dessin ont bien été respectées et que les parasites introduits n'auront pas d'influence critique sur le fonctionnement.

Sachant que le dessin d'un amplificateur opérationnel comportant 50 transistors nécessite environ deux semaines de travail, on imagine le temps qu'il faut pour dessiner les circuits logiques comportant plusieurs milliers de transistors. D'où l'intérêt d'automatiser cette phase de la conception. En effet, outre un gain de temps considérable lors du dessin du circuit, cela permet aussi de supprimer l'étape de vérification et les multiples allers-retours qui étaient jusqu'alors nécessaires avant d'obtenir un layout correct. Tout ceci entraînant, bien sûr, un énorme gain de productivité.

Comme nous l'avons dit précédemment, la plus grande part des outils de layout automatique existant concerne les circuits intégrés numériques. L'électronique numérique est en effet le seul domaine pour lequel la recherche a débouché sur des logiciels utilisés dans l'industrie. Le développement de ces logiciels a été facilité par l'introduction d'une certaine standardisation dans les circuits logiques. Cette standardisation consiste à réutiliser au maximum, au sein d'un même circuit, un certain nombre de structures prédéfinies. Elle permet donc de simplifier la tâche du concepteur en limitant ses degrés de liberté. Cette simplification se fait au détriment de la taille du circuit et de ses performances. Un compromis doit donc être trouvé entre la diminution du temps de conception du circuit et l'augmentation de sa taille. Différents degrés de standardisation sont donc possibles, chacun correspondant à un style de layout différent. Sur les six styles existant en numérique [54], seuls quatre permettent une génération automatique du layout.

En ce qui concerne l'analogique, une telle standardisation n'est pas envisageable. En effet, les tailles des transistors utilisés au sein d'un même circuit varient beaucoup plus et il est quasiment impossible de réaliser des bibliothèques de blocs fonctionnels: en effet, même s'ils réalisent la même fonctionnalité, ceux-ci doivent être redimensionnés à chaque fois, voire même complètement reconçus, pour satisfaire aux spécifications du circuit (par exemple, l'architecture d'une source de courant ainsi que les dimensions de ses transistors changent en fonction du courant qu'elle doit délivrer et de la résistance de sortie qu'elle doit avoir). Il est donc particulièrement difficile de trouver des simplifications dans la manière de procéder pour réaliser le dessin des masques d'un circuit intégré analogique.

Celui-ci comporte en général trois étapes: la création des layouts de toutes les cellules élémentaires du circuit, le placement de ces cellules les unes par rapport aux autres de manière optimisée, et enfin la réalisation des interconnexions. Cette méthodologie serait proche de certains types de dessin des masques en numérique si les phases d'optimisation ne faisaient pas intervenir les contraintes d'appairage des composants et la prise en compte des phénomènes de bruits ou des capacités parasites de routage, indispensables au bon fonctionnement final du circuit. Le principal problème devient alors non pas d'arriver à connecter entre elles toutes les cellules du circuit (il n'y en a heureusement que quelques centaines au maximum), mais bien d'assurer que celui-ci fonctionnera correctement à la fin, même si cela doit entraîner une petite perte de place. Nous sommes passés d'un problème d'ordre plutôt quantitatif à un problème plus qualitatif, qu'il est d'autant plus difficile de traiter de manière informatique. C'est pourquoi, à notre connaissance, il n'existe actuellement que peu de systèmes d'automatisation du layout en analogique.

Dans ce chapitre, nous allons donc présenter l'outil Chirvan qui a été développé au CNET-Grenoble pour résoudre le problème de l'automatisation du layout des circuits analogiques.

4.2. METHODOLOGIE DE CONCEPTION DES CIRCUITS INTEGRES ANALOGIQUES AU CNET GRENOBLE

La conception des circuits analogiques débute par la saisie du schéma électrique du circuit que le concepteur a choisi pour réaliser les fonctions nécessaires pour répondre aux spécifications du cahier des charges.

Une fois le schéma saisi, un utilitaire permet de produire automatiquement le fichier de simulation correspondant (netlist), qui peut être exploité par un simulateur électrique ELDO, ainsi que son optimiseur OPART. Le but de cette simulation est de vérifier que la cellule que l'on va implanter respecte bien les spécifications attendues. Il faut associer à cette netlist d'autres fichiers nécessaires au bon fonctionnement de Eldo/Opарт, à savoir le fichier contenant les paramètres technologiques et le fichier des spécifications du cahier des charges. Cette phase de synthèse électrique (Fig. 4.1), permet d'optimiser les dimensions des transistors du schéma électrique ainsi que le circuit, de telle façon que les fonctions demandées soient réalisées, en respectant les spécifications. A la fin de cette phase, le schéma électrique ainsi obtenu peut servir pour l'implantation physique.

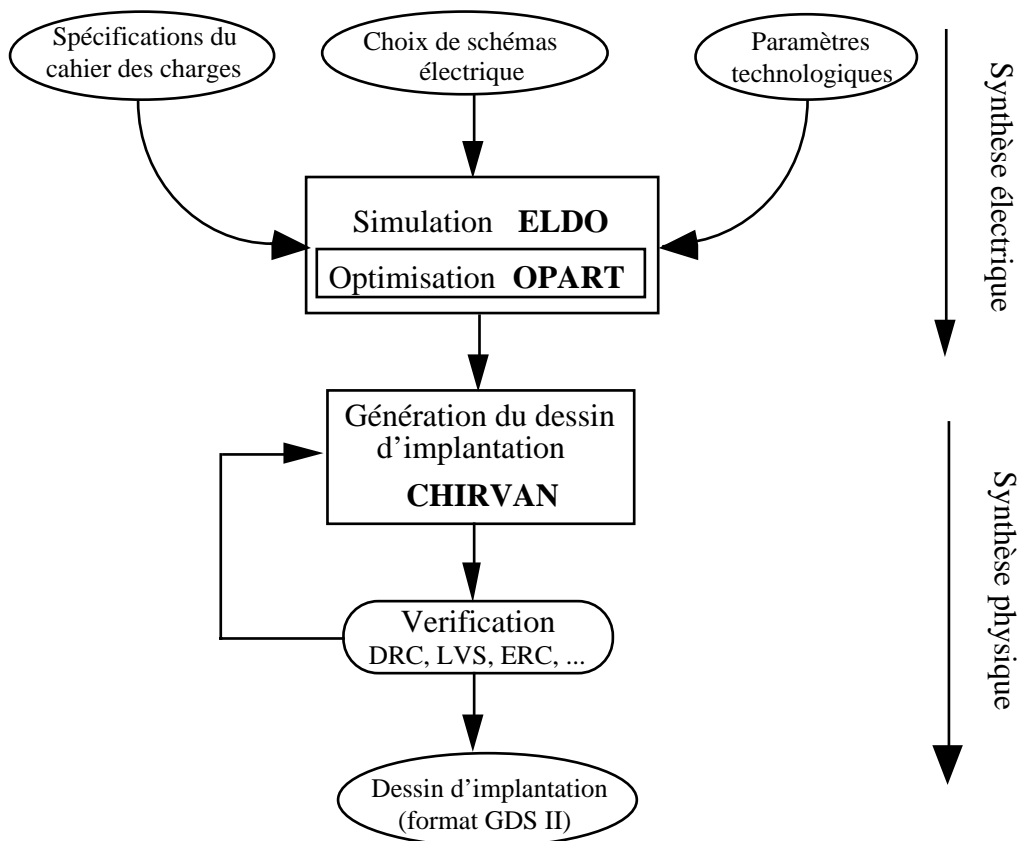


Fig. 4.1 : Description de la chaîne de conception des circuits intégrés au CNET-Grenoble

ELDO [55][56] est un simulateur électrique développé au CNET-Grenoble. Il est basé sur des simulations numériques itératives. Eldo est actuellement un outil connu mondialement par les concepteur analogiciens, commercialisé par la société Mentor Graphic.

OPART [40][57] est l'optimiseur d'Eldo. Il optimise automatiquement les dimensions des paramètres variables du schéma électrique choisi, en fonction du cahier des charges et des paramètres technologiques définis initialement. Son principe est de réaliser, entre chaque simulation, une mesure des performances électriques et d'en déduire à l'aide de méthodes mathématiques d'optimisation, de nouvelles valeurs pour les paramètres variables du schéma.

La phase de synthèse physique (Fig. 4.1) consiste en l'implémentation du layout du circuit intégré. L'environnement utilisé est l'outil CHIRVAN (§ 4.3), développé au CNET par notre équipe, et contient plusieurs étapes (Fig. 4.2): génération automatique du layout des composants élémentaires (transistors, capacités, résistances.....) et de certains blocs de base (paire différentielle, source de courant...); placement de ces composants; routage et compaction automatique.

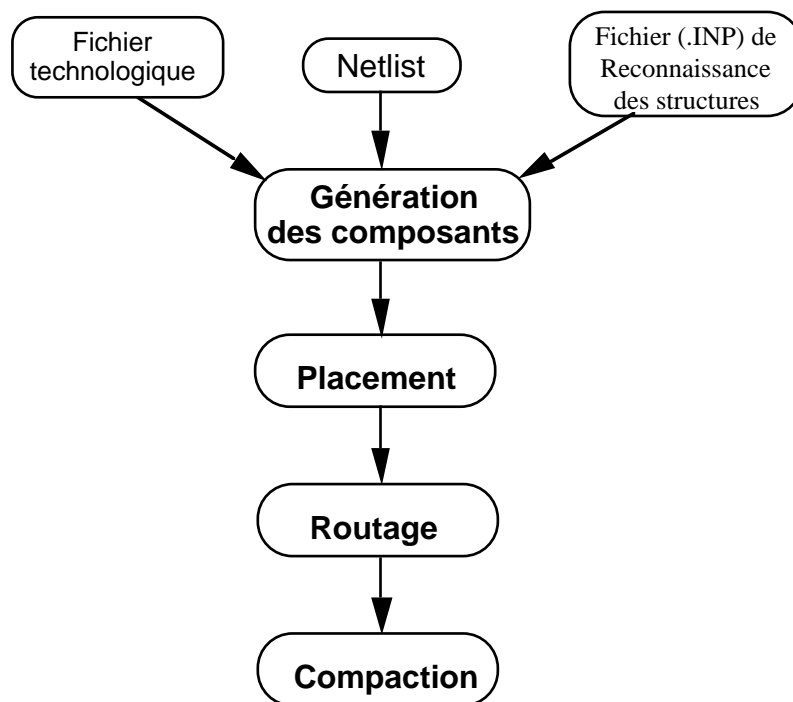


Fig. 4.2 : Chaîne d'implémentation du Layout des circuits intégrés au CNET-Grenoble CHIRVAN.

Le layout de la cellule étant achevé, il s'agit de vérifier les règles de dessin DRC (Design Rule Checking) selon différents critères (ex: vérification physique LVS (Layout Versus Schematic), ERC, ESD....). Puis, une extraction et une nouvelle simulation du circuit sont réalisées afin de vérifier que le layout final permet toujours la réalisation des fonctions demandées, en respectant les spécifications du cahier.

Stratégie de Conception Hiérarchique des Circuits Intégrés Mixtes

La conception des systèmes mixtes analogiques/numériques est simplifiée par la méthodologie de conception hiérarchique, qui permet de réduire la complexité du système en sous-blocs, à leur tour divisés en plusieurs cellules, etc. Une hiérarchie constituée de plusieurs niveaux de conception est ainsi obtenue. Les différents niveaux ainsi formés sont présentés à la figure 4.3 : niveau système ou niveau modules (CAN, CNA, PLL...), niveau circuits (comparateur, amplificateur opérationnel, générateur de tension ou de courant, switch...), et niveau composants (transistors, capacités, résistances...). Chaque fonction hiérarchisée peut être réalisée par différents types architecturaux ou topologiques. Cependant, une telle hiérarchie n'enlève rien à la difficulté de réalisation d'un outil de synthèse automatique de circuits analogiques (par exemple, l'outil Synopsys pour la synthèse des circuits numériques), étant donné qu'une même fonction (le gain par exemple) peut être obtenue par des éléments de "niveaux hiérarchiques" différents (un transistor, un étage, un amplificateur opérationnel...) et que les interactions entre ces différents niveaux influent très fortement sur les caractéristiques globales du circuit (bruit, bande passante, gain...)

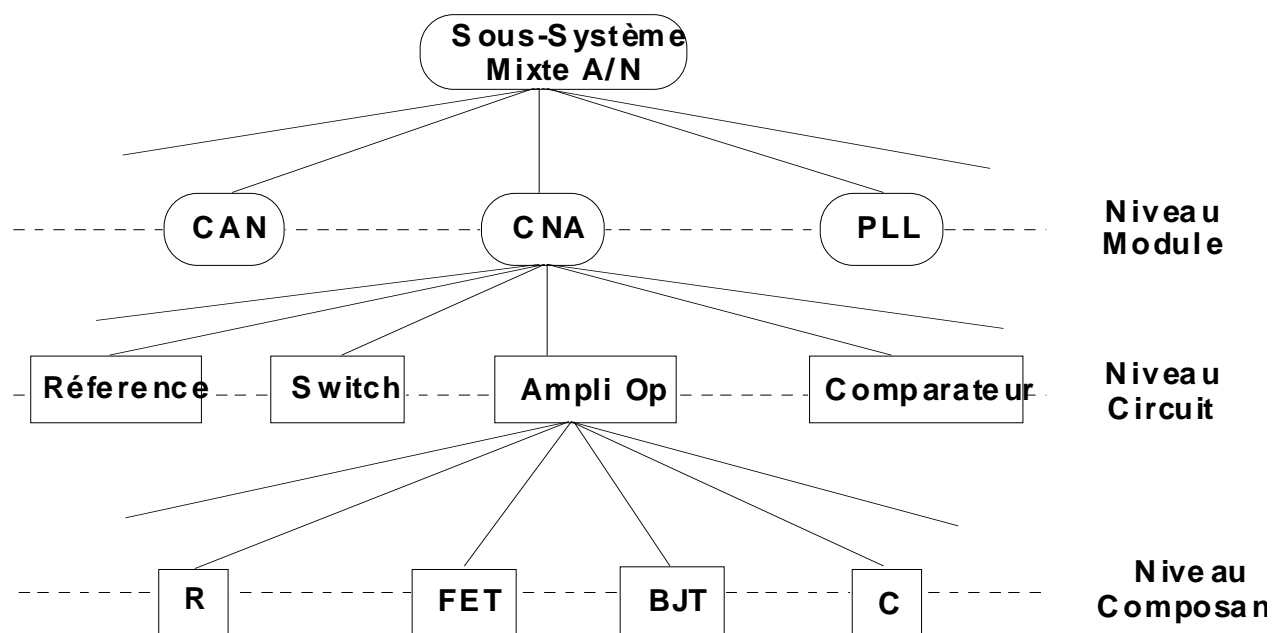


Fig. 4.3 : Les niveaux hiérarchiques de la conception des circuits mixtes A/N.

4.3. LE LOGICIEL CHIRVAN.

Le logiciel CHIRVAN développé au CNET Grenoble est destiné à accélérer le processus de synthèse du layout des circuits intégrés analogiques ou mixtes. Il est organisé en plusieurs modules:

- ◆ Éditeur de schémas électriques.
- ◆ Éditeur de dessin de masque (layout)
- ◆ Générateur automatique du layout des composants
- ◆ Placement interactif des composants
- ◆ Routeur automatique des cellules analogiques (routeur de Labyrinthe)
- ◆ Routeur automatique entre blocs analogiques (routeur de Canal)
- ◆ Compacteur automatique.

Tous ces modules sont intégrés dans un seul environnement que nous avons appelé CHIRVAN (Fig. 4.4).

Chirvan possède deux fonctions principales : la saisie de schémas électriques et le dessin des masques. A chacune de ces fonctions correspond une fenêtre ayant ses caractéristiques propres: menus, modes de fonctionnement et méthodes d'affichage. Ces deux interfaces sont orientées objet. C'est-à-dire que toutes les entités graphiques qu'elles contiennent réagissent aux "clics souris" selon leur nature. Cela permet un apprentissage plus rapide des commandes et une économie de menus.

La première application permet au concepteur d'introduire le schéma électrique de son circuit sous forme graphique. Elle est adaptée à des circuits d'environ une quarantaine de composants. Les interconnexions sont réalisées explicitement par le concepteur en dessinant les fils reliant les composants. Un certain nombre d'opérations est offert pour dessiner le circuit sous un aspect agréable : rotation, symétrie, alignement de plusieurs composants.

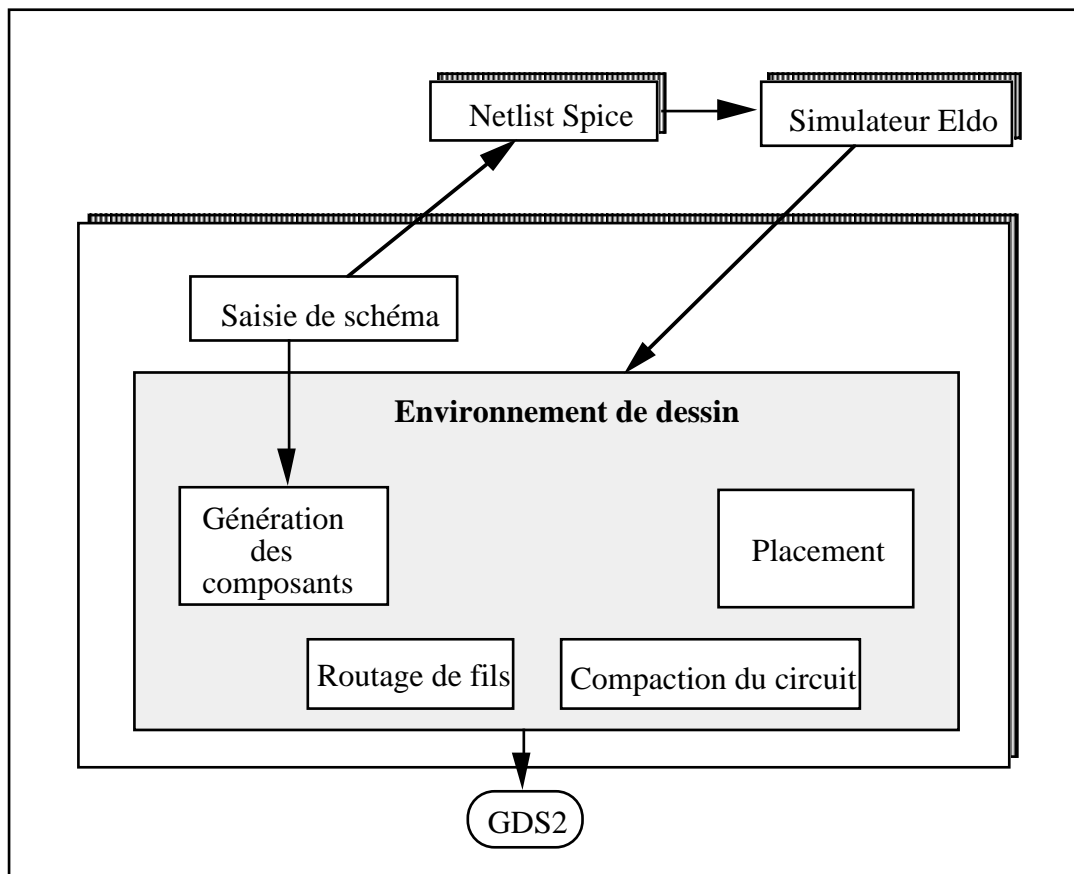


Fig. 4.4 : Structure des tâches dans CHIRVAN

Les masques sont ensuite conçus de façon constructive à partir du schéma ainsi validé. Le dessin des composants est réalisé automatiquement par des générateurs. Un générateur est un programme capable de dessiner, de manière optimale, les masques d'un composant donné, en tenant compte de ses paramètres électriques, des règles de dessin de la technologie et de la surface à occuper.

Le placement des composants reste encore interactif. Celui-ci, basé sur une analyse de contraintes électriques et topologiques à respecter. Il dépend complètement de l'expertise des concepteurs.

Le routage des composants entre eux est réalisé soit d'une façon manuelle, soit par le routeur automatique "le routeur de labyrinthe" (cf. § 4.4) qui permet de router automatiquement les cellules analogiques en tenant compte de toutes les contraintes analogiques.

Le routage des blocs entre eux (routage "top level") est réalisé par un routeur de canal (Chirvan routeur).

Etant donné que nous utilisons un routeur de surface pour le routage interne à la cellule (le routeur de labyrinthe) le placement des composants est réalisé de telle façon que l'espace de routage soit suffisamment large. D'où la nécessité d'exploiter un compacteur pour réduire la

surface totale du circuit. Nous avons donc réalisé un compacteur qui est en cohérence avec le routeur, c'est à dire, qui respecte les contraintes analogiques et géométriques déjà prises en compte par le routeur.

En résumé, Chirvan est un outil complet de synthèse du layout des circuits intégrés analogiques et mixtes. Il permet en outre, de réduire considérablement le temps passé à dessiner les layout de ces circuits. Il est aussi un outil indépendant de la technologie employée car celle-ci est un paramètre d'entrée pour l'outil.

Les interfaces graphiques du logiciel CHIRVAN ont été construites sur une bibliothèque d'objets graphiques, que nous avons développée en commun avec l'ENST. Cette bibliothèque, écrite en langage Le-Lisp, a été transférée à la société ILOG qui la commercialise sous la marque Aida-2D [58]. Il nous semble important, pour comprendre certains détails d'implémentation de CHIRVAN (Fig. 4.5), d'évoquer les principes généraux et les bonnes propriétés d'extensibilité d'Aida-2D. Celles-ci proviennent essentiellement de l'approche orientée objet qui a été retenue dans son implémentation. Pour une description plus détaillée d'Aida-2D, nous suggérons la lecture de l'annexe A2.

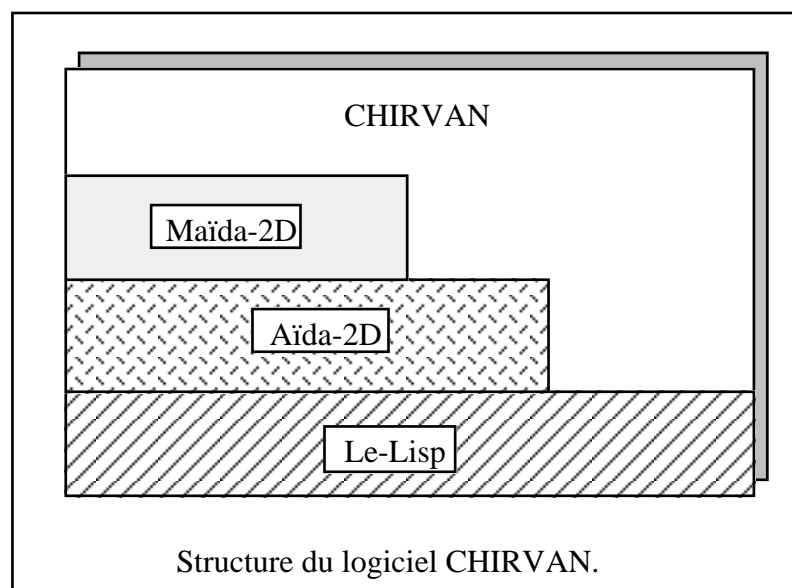


Fig. 4.5 : Structure du logiciel CHIRVAN, environnement de programmation.

4.4. GENERATION DES COMPOSANTS

La première étape de l'automatisation du layout des circuits intégrés analogiques est la génération automatique des composants. Cette étape consiste à générer les masques des

composants (transistor, résistance, capacité, ...) à partir de leur représentation au niveau du schéma électrique et de la netlist. Le générateur de Chirvan dispose d'une liste assez complète de générateurs (NMOS, PMOS, MOS de charge, paire différentielle, source de courant, miroir de courant, BJT, résistance, capacité, prises caissons et substrats, ...).

4.4.1. Le MOS

Il existe un générateur de MOS replié (Fig. 4.6a). Les paramètres d'entrée sont le type du transistor, sa longueur et largeur de grille et le facteur de repliement (ou nombre de transistors élémentaires à interconnecter). Les trois premières données sont recherchées automatiquement dans la netlist du schéma et ne peuvent être modifiées. Une valeur est suggérée pour le facteur de repliement, mais peut être corrigée interactivement dans le formulaire proposé à l'utilisateur.

Selon le type du transistor (N ou P) (Fig. 4.6b), la nature de la diffusion est automatiquement déterminée, ainsi que l'existence du caisson.

Le générateur calcule automatiquement les différents masques composites en respectant les règles de dessin minimales de la technologie:

- Le caisson séparé si le transistor est de type P ;
- La diffusion ;
- Les grilles en poly ainsi que leur interconnexion par une barre de poly perpendiculaire ;
- Les connexions entre les drains élémentaires ainsi qu'entre les sources élémentaires, qui correspondent aux niveaux métal1 et contact ;
- La surgravure est prise en compte dans le calcul de la diffusion en fonction du nombre de repliements.

Chaque rectangle de poly ou de métal connaît l'équipotentielle à laquelle il est attaché dans le schéma électrique. Drain et source sont considérés comme étant symétriques.

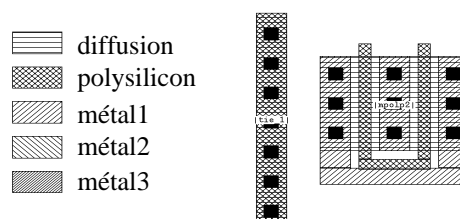
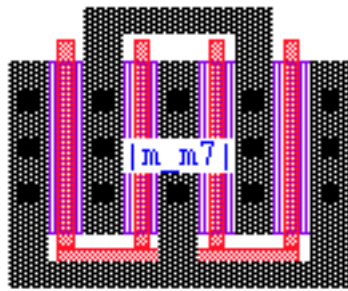
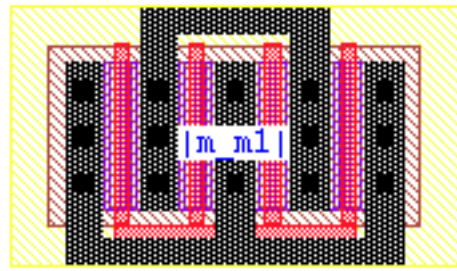


Fig. 4.6a : Layout d'un transistor MOS.



Transistor NMOS



Transistor PMOS

Fig. 4.6b : Layout des transistors NMOS et PMOS.

4.4.2. Le MOS de Charge

Il s'agit d'une variante du précédent où la grille du transistor est connectée à une des deux diffusions. Le générateur vérifie que cette connexion est effectivement présente dans la netlist avant de la générer.

La surgravure est prise en compte dans le calcul de la diffusion en fonction du nombre de repliements.

4.4.3. La Paire Différentielle

Il existe différents types de générateurs de paire différentielle. Par exemple, celui où les deux MOS sont repliés de manière identique et sont placés côte à côte, ou celui où les deux MOS sont interdigités.

Les paramètres d'entrée sont le type des transistors, la longueur et la largeur de grille communes et le facteur de repliement. Les trois premières données sont recherchées automatiquement dans la netlist du schéma et ne peuvent être modifiées. Une valeur pour le facteur de repliement est proposée mais peut être corrigée interactivement dans le formulaire. Le générateur vérifie également qu'il s'agit bien d'une paire différentielle (transistors identiques et connectique correspondante).

Selon le type des transistors (N ou P), la nature de la diffusion est automatiquement déterminée, ainsi que l'existence du caisson.

Chaque rectangle de poly ou de métal connaît l'équipotentielle à laquelle il est attaché dans le schéma électrique. Les deux transistors sont considérés comme étant symétriques.

4.4.4. Le Miroir de Courant

Il n'existe qu'un seul générateur de miroir de courant: les deux MOS ont les grilles placées dans le prolongement l'une de l'autre. Il est également possible de réaliser un miroir où les transistors sont côte à côte, par fusion (voir paragraphe suivant) d'un transistor normal et d'un MOS de charge.

Les paramètres d'entrée sont le type des transistors, les longueurs et la largeur commune de grille et les 2 facteurs de repliement. Une valeur pour les facteurs de repliement est proposée mais peut être corrigée interactivement dans le formulaire. Le générateur vérifie également qu'il s'agit bien d'un miroir de courant (même type et connectique correspondante).

Selon le type des transistors (N ou P), la nature de la diffusion est automatiquement déterminée, ainsi que l'existence du caisson.

Chaque rectangle de poly ou de métal1 connaît l'équipotentielle à laquelle il est attaché dans le schéma électrique. Etant donné le schéma électrique correspondant, il n'y a aucune symétrie.

4.4.5. La Fusion de deux Transistors ou Groupes de Transistors

Il n'existe qu'un seul type de fusion, celui des diffusions (Fig. 4.7). La fusion des grilles est remplacée par leur connexion manuelle en Poly. La fusion est une opération récursive.

Les paramètres d'entrée sont les transistors sélectionnés, ainsi que leur position relative au moment de l'opération. La fusion calcule automatiquement la position finale des deux MOS en tenant compte des règles de dessin minimums. Cette position est obtenue par une translation perpendiculaire à l'axe commun des grilles des transistors.

Avant de réaliser cette opération, le programme vérifie qu'elle est licite: même type de transistors, même orientation des grilles, connexion effectivement présente dans le schéma.

Après fusion, les éventuelles ambiguïtés (symétries) qui peuvent être levées le sont.

N.B. : La fusion est récursive. On peut également détruire une fusion (il s'agit alors d'une fission), dans ce cas les symétries ou ambiguïtés sont restaurées.

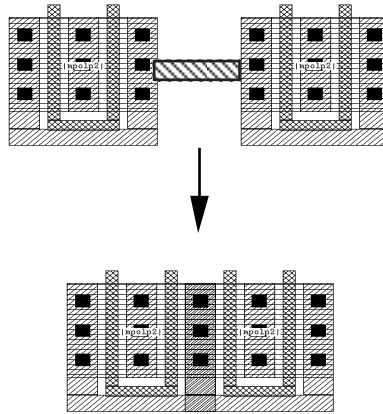


Fig. 4.7 : Fusion de deux transistors.

4.4.6. Les Prises Caissons et Substrats

Les prises sont générées à la demande à partir d'un menu particulier. En fonction du type de prise choisi, l'unique paramètre d'entrée à fournir est la longueur de prise à réaliser. La largeur de la prise générée est minimale. Il est possible de fusionner des prises pour en obtenir de plus larges.

Le générateur calcule automatiquement les différents masques composites en respectant les règles de dessin minimales de la technologie:

- Le caisson s'il s'agit d'une prise caisson ;
- La diffusion du bon type ;
- Une couche de métal1 connectée à la diffusion par des contacts ;
- Une couche de métal2 connectée au métal1 par des vias intercalés entre les contacts.

4.4.6. La Résistance

Il n'existe qu'un seul générateur de résistance, correspondant à la topologie "en serpent" qui est généralement utilisée.

Les paramètres d'entrée sont la valeur de la résistance qui est automatiquement recherchée dans la netlist, ainsi qu'un certain nombre de paramètres spécifiés par l'utilisateur à travers le formulaire qui lui est proposé :

-
- Le niveau de poly choisi ;
 - La largeur utilisée (on vérifie automatiquement que celle-ci est compatible avec la règle de dessin correspondante) ;
 - Le nombre de coudes dans le serpent.

Le générateur calcule automatiquement les différents masques composites en respectant les règles de dessin de la technologie :

- Le serpent dans le niveau de poly choisi ;
- Le contact à chacune des deux extrémités ainsi que le métal1 entourant ce contact ;
- La surgravure des polys est prise en compte.

4.4.7. Les Capacités

Il existe deux générateurs de capacités: les capacités apparaissant dans le schéma, dites capacités fines, ayant une valeur bien déterminée, et les capacités de découplage, dites capacités gaufrées, qui s'adaptent davantage à la surface disponible. Les capacités sont réalisées par superposition de deux plateaux en poly1 et poly2.

a. Capacité fine

Les capacités déterminées dans le schéma sont générées comme pour une résistance ou un transistor. Le formulaire qui apparaît comporte les paramètres suivants:

- Nom et valeur de la capacité ;
- Largeur (w) et hauteur (h) de la capacité ;
- Sélection du mode avec ou sans contacts (Fig. 4.8 et 4.9).

Le principe de génération est le suivant: puisque la valeur est fixée, seule l'une des deux dimensions sera libre (ou les deux), l'autre étant calculée. Si les deux valeurs sont libres, la capacité sera carrée. Laisser libre une dimension consiste à lui donner une valeur nulle. Les dimensions correspondent à celles des deux plaques de poly et ne concernent pas l'encombrement total de la capacité. Dans le mode "sans contacts", seuls les deux plateaux de poly sont générés.

On prend en compte les règles de débordement poly1/poly2, poly/contact, métal1/contact. La surgravure entre aussi dans le calcul des dimensions mais non la capacité périmétrique.

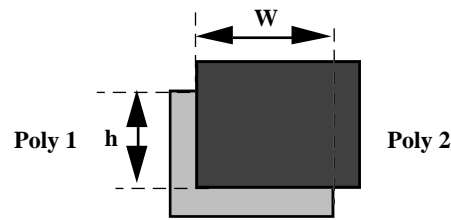


Fig. 4.8 : Capacité fine sans contacts.

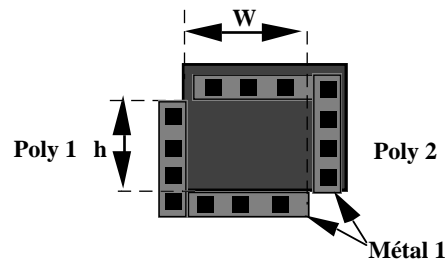


Fig. 4.9 : Capacité fine avec contacts.

b. La Capacité Gaufrée

Les capacités de couplage sont déterminées par leur encombrement, la valeur de la capacité étant estimée à partir des dimensions. Les paramètres de génération sont les deux dimensions w et h de la capacité. La capacité est réalisée par la mise en parallèle de plusieurs capacités élémentaires d'une longueur minimale (environ 30μ , interconnexions comprises). Ces capacités sont interconnectées par des rails en métal1. Le calcul de la capacité est actuellement très approximatif.

4.5. ROUTAGE

4.5.1 Méthodologie

Deux méthodes différentes sont généralement proposées pour résoudre le problème de routage (le routeur de canal et le routeur de labyrinthe) [59].

Le routage de canal est efficace du point de vue du temps de routage; il est aussi bien adapté au routage de circuits denses, puisqu'il permet un routage de plusieurs noeuds simultanément dans le même canal. De plus, une compaction ou décompaction locale, peut être exécutée après le routage de chaque canal. La phase de compaction globale n'est donc pas nécessaire.

Cependant, la méthode de routage de canal nécessite que la surface de routage soit divisée sous forme de canaux de routage, ce qui implique une arborescence de rectangles ou pseudo-rectangles, qui peut être difficile à créer dans le cas des géométries complexes; par exemple, dans la conception des circuits analogiques, les tailles et les formes des composants des cellules varient considérablement. La compaction du canal peut s'avérer difficile à achever dans ces géométries et éventuellement une compaction globale deviendrait nécessaire.

Ces considérations, ajoutées au fait que le nombre des équipotentielles des cellules analogiques est assez réduit, conduit à choisir le routeur de surface [60] pour le routage des cellules analogiques.

Étant donné qu'un routeur de labyrinthe ne déplace pas les composants, une étape de compaction est nécessaire. Le développement des deux outils doit être cohérent, c'est à dire qu'une contrainte respectée par le routeur doit également être respectée par le compacteur. Par conséquent, la description de la contrainte ne doit être faite qu'une fois pour les deux outils. Pour cela, les contraintes de performances sont d'abord traduites en contraintes géométriques: minimisation de la longueur de la piste métallique (spécialement pour des couches de métal résistives comme le polysilicium), minimisation du nombre de vias, absence de superposition entre les noeuds, distance entre les noeuds, symétries.

Le layout physique peut alors se scinder en quatre étapes principales:

- Un placement initial suffisamment large (automatique ou manuel) est exécuté pour prévoir de l'espace pour le routage.

- Le routeur minimise la longueur de la piste métallique et le nombre de vias, prenant en compte le non-recouvrement des noeuds et les symétries. Le routage sur les transistors peut être permis ou interdit.

- Les pistes métalliques sont alors élargies, soit localement soit globalement, dans le but d'abaisser la résistance des noeuds.

- Finalement, les compacteurs minimisent la surface de la cellule et les longueurs des pistes métalliques, tout en respectant les contraintes du routeur. Le nombre de vias reste inchangé durant cette étape.

La minimisation de la longueur des pistes métalliques et du nombre de vias, s'exécute selon une fonction de coût de la forme:

$$\sum c_i l_i + \sum v_j n_j \quad (4.1)$$

où

c_i	est le coût linéaire de la couche métallique i ,
l_i	la longueur totale de la piste métallique de la couche i ,
v_j	le coût d'un via de type j ,
n_j	le nombre de vias de type j .

Les c_i ne sont pas tous identiques (par exemple, le c_i est plus élevé pour le polysilicium que pour le métal). Une piste de routage entre deux zones de polysilicium peut alors être routée en métal, si la distance est assez longue; c'est à dire, plus longue que:

$$2.v_{\text{métal-poly}}/(c_{\text{poly}} - c_{\text{métal}}) \quad (4.2)$$

4.5.2. Description des Contraintes Géométriques

Les contraintes géométriques sont utilisées par le routeur et le compacteur, et résulte d'une analyse des contraintes de performances qui peut être établie sommairement comme suit:

- liste des largeurs des fils de routage (noeud, largeur);
- paramètres (c_i, v_j) de la fonction de coût correspondant à chaque fil de routage et à chaque type de vias;
- listes des couples (noeud1, noeud2) qui doivent respecter les contraintes de non-recouvrement;
- liste des distances minimums (noeud1, noeud2, dist) à respecter entre les noeuds pour éviter une capacité de couplage excessive.
- liste des symétries (pour l'appairage des parasites); chaque symétrie est décrite comme une liste de couples de composants symétriques, une liste de couples de noeuds symétriques et l'orientation de leur axe de symétrie (vertical ou horizontal).

Notons que les deux outils utilisent la fonction de coût mais pour différents propos. La fonction de coût du routeur est utilisée pour guider l'outil dans une recherche partiellement discrète d'espace. Certaines contraintes analogiques résultent de choix topologiques discrets (par exemple, les fils à un seul niveau de routage ou à plusieurs niveaux), d'autres sont continus (la longueur de la piste métallique). En changeant les coefficients c_i et v_j mentionnés ci-dessus, le type de comportement désiré peut être induit. Par contraste, une fois que le circuit a été routé, tous les choix topologiques sont établis et doivent être respectés par le compacteur. La fonction de coût du compacteur est simplement utilisée pour minimiser la résistivité "réelle" de la piste métallique.

4.5.3. Algorithme de Routage

Les méthodes basées sur des grilles sont mal adaptées aux cas de géométries complexes. Plusieurs méthodes ont été proposées pour le routage des circuits analogiques [61][62][63][64][94][95]. La méthode décrite ici dérive du routeur d'expansion de tuiles (tile-expansion router). Cependant, cette technique a été adaptée pour permettre le routage à un seul ou à

plusieurs niveaux de métal. Les trois étapes de l’algorithme de routage décrites ci-dessous, sont la construction des structures de corner-stitching[65], l’ordonnancement des noeuds et le routage des fils individuellement, incluant la mise à jour des structures de corner-stitching.

4.5.3.1 Construction de la représentation du Corner-Stitching

Dans une représentation de layout standard, seule la surface occupée par les composants ou les pistes métalliques est représentée. Dans notre système, nous utilisons une librairie graphique à deux dimensions (“2D”) et une structure d’arbre à quatre voisinages (“quad-tree”). Cependant, pour le routage, une bonne représentation de la surface vide est requise. Pour cela, une structure de corner-stitching correspondant à la surface de routage est attribuée à chaque fil de routage (généralement du polysilicium ou du métal). La surface de routage correspond soit à la surface totale de la cellule soit à une partie rectangulaire de la cellule. Une structure de corner-stitching (que nous noterons C-S) est un ensemble de rectangles -ou de tuiles- respectant le non-recouvrement, reliés entre eux; chaque tuile consiste en une origine (x,y), quatre pointeurs vers des tuiles voisines particulières (Fig. 4.10), et une liste d’attributs. L’usage des pointeurs et la non-redondance d’information font la mise à jour de la structure (en insérant ou supprimant un rectangle, en fusionnant deux tuiles...) très rapidement. Cette structure permet également une recherche assez rapide ($O(\sqrt{N})$) où N est le nombre de tuiles) des tuiles contenant un point, ou de la liste de tuiles croisant un rectangle donné [66].

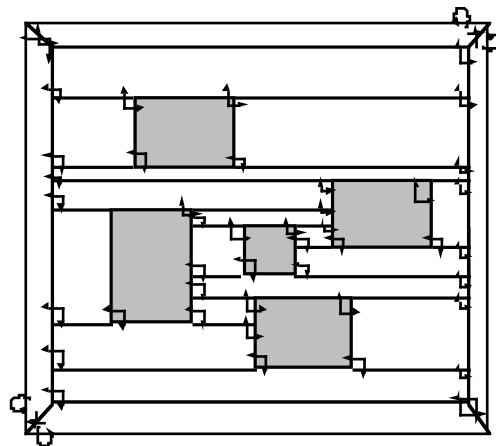


Fig. 4.10 : Une représentation de corner-stitching est un ensemble de tuiles rectangulaires reliées entre elles.

Une liste d'attributs est donnée pour chaque tuile. Ces attributs fournissent une information utile pour le routeur telle que le(s) composant(s) et/ou le(s) piste(s) métallique(s) auxquels le rectangle original appartient. Une liste vide signifie que le rectangle peut être utilisé pour le routage.

La construction de la C-S commence avec une structure initiale de cinq tuiles (Fig. 4.11) correspondant à une surface de routage vide. Les rectangles basiques du layout, élargis de la moitié de la distance minimum sont alors insérés, et les tuiles résultantes reçoivent un attribut d'après la nature du rectangle du layout. Si deux rectangles se chevauchent (Fig. 4.12), l'attribut de l'intersection résultante est l'union des deux ensembles d'attributs. Notons que les rectangles d'une couche de métal donné peuvent représenter des tuiles du C-S de couches différentes si une règle existe entre les deux couches; par exemple, une tuile correspondant à un rectangle de diffusion peut être insérée à l'intérieur d'une surface de polysilicium C-S.

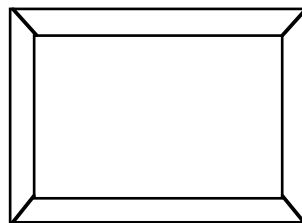


Fig. 4.11 : Le corner-stitching initial, représentant une surface vide. Les quatre tuiles environnantes ne correspondent pas à la surface de routage, mais permettent aux algorithmes de recherche de bien fonctionner à la frontière du circuit (ou des blocs).

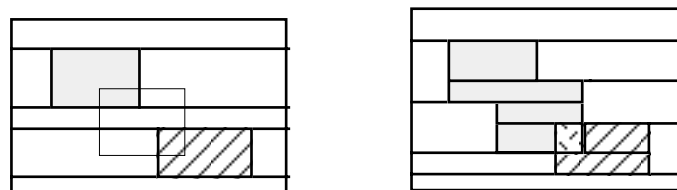


Fig. 4.12 : Insertion d'un nouveau rectangle dans la structure de corner-stitching. Le nouveau rectangle a les mêmes attributs (représentés en gris) qu'une tuile existant déjà. Les attributs des nouvelles tuiles sont l'union des nouveaux attributs (résultant de l'insertion du rectangle) avec les attributs des anciennes tuiles. Les rectangles qui ont les mêmes attributs peuvent fusionner.

Cette approche consiste à insérer chaque rectangle d'un composant donné plutôt qu'à insérer un rectangle unique du composant. Bien que davantage de mémoire soit consommée, ceci offre deux avantages majeurs: premièrement, les composants non-rectangulaires peuvent être traités correctement (Fig. 4.13); deuxièmement, les zones métalliques d'un composant

peuvent être considérées comme des fils ordinaires, ce qui réduit le nombre de traitements spécifiques.

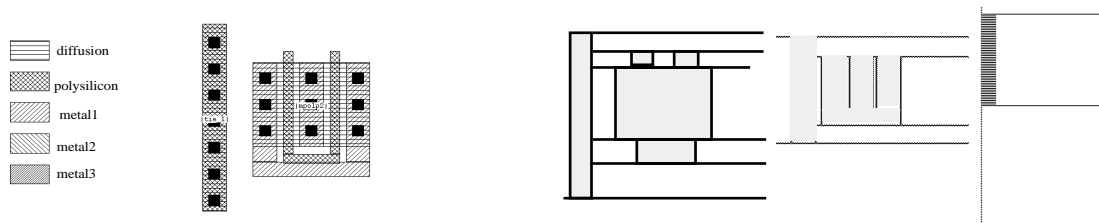


Fig. 4.13 : Une structure de layout consistant en un transistor NMOS et un contact du Well, et les tuiles correspondantes (de droite à gauche, dans le polysilicium de C-S, le métal1, le métal2).

4.5.3.2. Classement des Noeuds

Une fois que la surface de routage a été construite, les noeuds (ou equipotentielles) sont routés un à un. La détermination du prochain noeud à router s’accomplit de la façon suivante:

- Avant le dessin des fils de routage, la source et le drain des transistors peuvent, en général, être échangés, exception faite pour les transistors déjà fusionnés avec une prise (“body-tie”) ou un autre composant. Un noeud qui ne contient que des zones ambiguës (avec source/drain non encore distinguées) ne peut pas être routé en premier;
- De tous les noeuds routables, il faut d’abord router ceux qui ont le plus grand nombre de contraintes de non-recouvrement;
- Des précédents, il faut d’abord router celui qui a le meilleur rang dans la liste de priorité définie par l’utilisateur;
- S’ils ne reste que des noeuds ambigus (ce qui n’est généralement pas le cas, étant donné que les ambiguïtés sont progressivement résolues durant le routage des premiers noeuds), il faut assigner arbitrairement des zones de l’un des transistors ambigus et procéder comme décrit précédemment. Ceci est rarement le cas dans le routage d’un circuit complet, étant donné qu’au moins les terminaux d’entrée et de sortie (I/O) pin du circuit ont un noeud non-ambigu, mais cela peut apparaître dans le routage local.

4.5.3.3. Routage des noeuds

Le routage de chaque noeud est dérivé de l’algorithme de Lee-Moore [67][68] avec A* améliorant [69], et de l’algorithme d’expansion de tuiles [62][63]. Chaque connexion se fait donc en deux phases: expansion d’un noeud (appelé source) jusqu’à ce qu’un des noeuds cible soit atteint; puis, tracé rétrograde de la cible jusqu’à la source. Un noeud est une collection de

rectangles connectés, appartenant soit à un composant soit à un fil déjà dessiné. Par contraste avec l’algorithme initial, il n’y a pas de graphe de routage explicitement prioritaire. A la place, la description de la surface de routage est mise à jour de façon dynamique durant la phase d’expansion, en vue d’éviter de trop petits rectangles qui ne peuvent pas être utilisés pour le routage.

a. Expansion

Le principe d’expansion A* est basé sur le calcul pour chaque noeud, de deux coûts: le coût réel correspondant au calcul de la fonction de coût le long du chemin réel de la source au noeud, et le coût estimé du chemin optimisé ignorant les obstacles du noeud à la cible. La fonction de coût estimée entre le noeud et un élément de la cible est calculée à l’aide de la formule suivante:

$$\min_i (c_i \cdot D + \sum v_j \cdot n_j) \quad (4.3)$$

où D est la distance,
 i est un des fils de routage;
 n_j est le nombre de via de type j nécessaire pour router le fil i.

Par exemple, pour connecter un noeud de polysilicium à un noeud de métal1, en utilisant du métal2, nous avons besoin d’un via de poly/métal1 et de deux vias de métal1/métal2. Ce coût estimé relativement compliqué est rendu nécessaire par le choix d’une fonction de coût hétérogène. Par la suite, le coût total représente la somme des coûts réels et estimés.

L’expansion crée des objets du type “noeud”, qui sont placés à l’intérieur de deux collections, respectivement appelées A (atteinte) et E (étendue). Le type “noeud” est défini par les critères suivants:

tuile: la tuile sur laquelle le noeud est basé (un rectangle dans une couche donnée);

coût-r: le coût réel, dont la définition est donnée ci-dessus;

coût-e: le coût estimé (pour atteindre la cible);

entrée: la zone où le fil doit pénétrer la tuile: il s’agit soit d’un segment si la tuile et son parent appartiennent à la même couche, soit d’un rectangle (l’intersection de la tuile et de son parent) s’ils appartiennent à différentes couches. A ceci, il faut ajouter le “point d’entrée” (le point d’entrée le plus proche du point d’entrée parent) qui est utilisé pour calculer les distances dans l’évaluation du coût.

parent: le noeud parent (description de l’arbre d’expansion); le parent peut être dans la même couche ou dans une couche différente.

Une fois que le noeud à router a été déterminé, l’expansion commence avec le noeud non-ambigu. Un ensemble de noeuds initiaux est formé par les tuiles de noeud source. Ces noeuds

ont tous un coût réel nul, et forment la liste initiale A. L'algorithme d'expansion est alors défini comme suit:

il faut choisir dans A le noeud qui a le coût total minimum;

si le noeud est un noeud cible, **alors** procéder au tracé (rétrograde) ou final;

sinon, il faut placer le noeud dans E, en le déplaçant de A;

l'étendre dans une troisième direction: par exemple, pour chaque couche connectable à la couche courante, créer les tuiles vides les plus larges dans chaque direction et créer le noeud correspondant (Fig. 4.14) si la jonction est assez large selon la largeur minimum de la couche; mettre ces noeuds dans R;

continuer jusqu'à ce que la cible soit atteinte ou que l'expansion ne soit pas possible (cible inaccessible).

b. Tracé retour ("Backtracing")

Une fois que la cible a été atteinte, le fil est tracé. Pour chaque noeud, l'entrée est précisément déterminée, les vias sont placés de telle façon que la longueur du chemin soit minimisée dans la couche la plus résistive et que le nombre de jogs soit aussi minimisé. Parallèlement, le fil réel est tracé dans la fenêtre d'éditeur de layout.

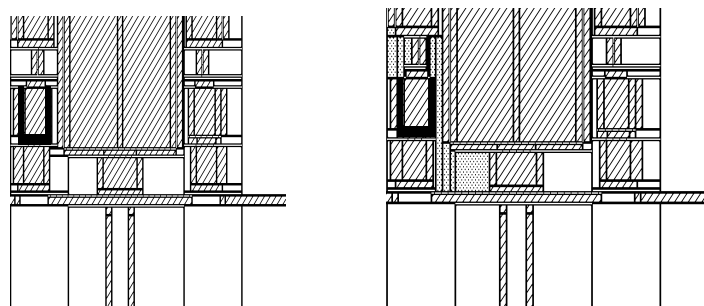
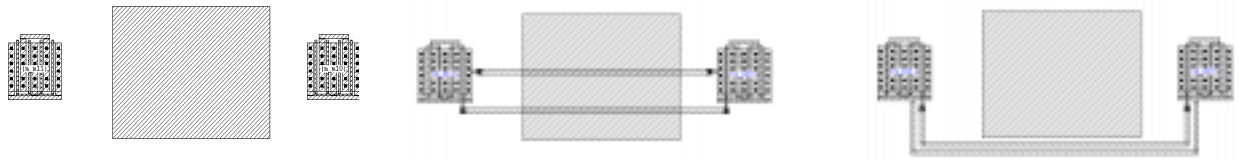


Fig. 4.14 : La phase d'expansion modifie le C-S dans le but d'éviter les tuiles superficiels qui ne peuvent pas être utilisées pour le routage. Le C-S correspondant au métal peut être observé. La surface noire représente le noeud source, la surface hachée les obstacles, la surface blanche les tuiles vides, et la surface grise les tuiles correspondant au noeud.

4.5.3.4. Routage à un seul niveau et à plusieurs niveaux de couche métallique

La figure 4.15 illustre l'influence de la fonction coût sur le choix entre un routage à un seul niveau ou à plusieurs niveaux de couche de métal. Deux transistors sont séparés par un obstacle de métal. Ils doivent être connectés ensemble via la grille et la source. A la figure 15(b), les coûts de vias sont bas, et le routage suit le chemin le plus court à travers l'obstacle utilisant le

métal2; à la figure 15(c), les coût de vias ont été augmentés, et le routage contourne l'obstacle, utilisant du métal1. Cependant, les grilles sont connectées à l'aide du métal1, à cause du coût élevé du polysilicium.



(a) Placement initial. (b) Routage utilisant une fonction de coût standard. (c) Routage avec coût de via élevé.

Fig. 4.15 : Routage à un seul niveau et à plusieurs niveaux de couches métalliques.

4.5.3.5. Attribution de noeud Ambigu

Quand un noeud est routé, une attention particulière doit être portée à ce que l'ambiguïté soit bien résolue: dans certains transistors repliés "folded transistors", une des zones du drain et de la source peut être plus difficile à connecter que d'autres. L'algorithme simple décrit ci-dessus peut conduire à un résultat non-optimal (Fig. 4.16).

Le problème peut être résolu en ajoutant un "coût d'accès" à la zone externe au drain et à la source. Ce coût permet au routeur de favoriser la zone interne. Le coût d'accès est posé comme le produit du coût linéaire du métal1 avec la hauteur du transistor. Une fois l'ambiguïté résolue, le coût d'accès est remis à zéro.

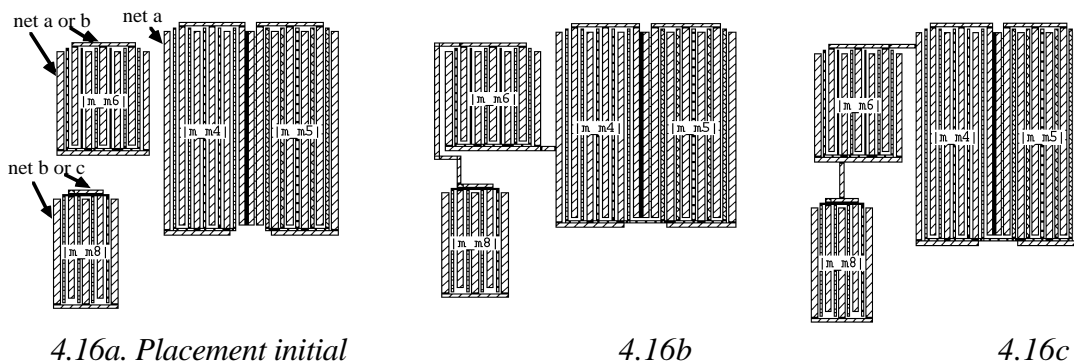


Fig. 4.16 : La longueur minimum du noeud (a) conduit à un routage non-optimal du noeud (b) (4.16b). Si nous prenons en compte le coût d'accès, cela conduit à un meilleur résultat (4.16c). Dans les deux cas, le noeud (a) est routé le premier.

4.5.3.6. *Non-recouvrement*

Avant le routage du noeud, des tuiles virtuelles sont temporairement insérées dans chaque couche de métal, au-dessus ou au-dessous de toutes les tuiles appartenant à un noeud ayant des contraintes de non-recouvrement avec le noeud routé. Les obstacles sont alors enlevés avant le routage du prochain noeud.

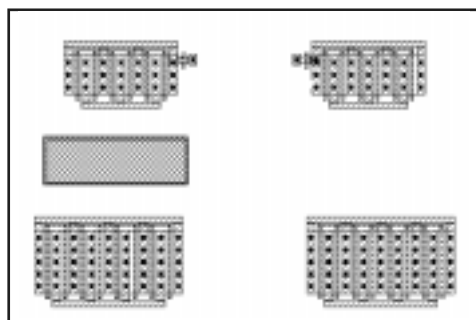
D'autre part, les noeuds ayant des contraintes de non-recouvrement sont routés en premier, dans le but de limiter les congestions.

4.5.3.7. *Symétries*

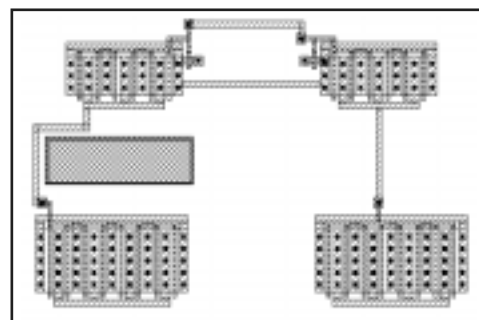
Le routeur peut manipuler des symétries, même imparfaites (Fig. 4.17). Si l'axe de symétrie est spécifié, une liste des correspondances (noeud 1, noeud 2) est extraite de la netlist et du placement du composant; c'est une liste de noeuds quasi-symétriques, c'est à dire partiellement symétriques (sur une partie du circuit). Dans ce cas, le routage se déroule en deux phases:

Dans la première phase, des obstacles virtuels symétriques aux obstacles réels (composant, fil de routage...) sont créés durant la description de la surface de routage. Chaque noeud est routé comme décrit ci-dessus dans la surface de routage réduite. Cependant, quand un fil est dessiné, un fil symétrique est dessiné si nécessaire et, dans tous les cas, un obstacle symétrique virtuel est créé, pour permettre à l'espace de routage de rester symétrique durant le routage. Lorsque les noeuds symétriques sont dessinés, des valeurs de propriétés spéciales sont attachées à des paires de segments symétriques dans le but d'informer en détails le compacteur sur les symétries qui doivent être maintenues. Si tous les noeuds n'ont pas été routés lors de la première phase, les objets virtuels sont enlevés, et le routage de tous les noeuds est complété, ignorant les symétries.

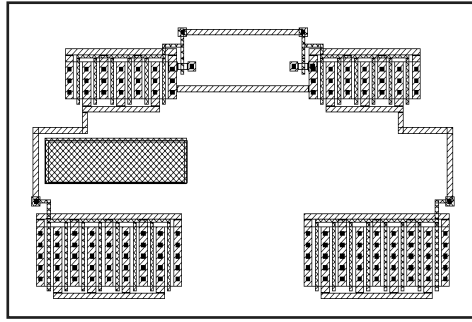
Notons que la description de l'occupation de la surface en termes d'attributs de tuiles facilite le placement et l'enlèvement des masques: l'enlèvement d'un masque consiste en la suppression de l'attribut correspondant de toutes les tuiles qui le possèdent.



(a) *Placement initial.*



(b) *Routage sans obstacle virtuel.*



(c) Routage avec obstacle virtuel.

Fig. 4.17 : Routage de géométrie quasi-symétrique. Dans la situation initiale (a), un obstacle non-symétrique sépare les composants de gauche. Le routage minimum amène à un résultat asymétrique (b); la présence de l'obstacle virtuel permet un routage symétrique (c).

4.6. COMPACTION

4.6.1. But

Le but du compacteur est de minimiser la taille du circuit (silicium) et de minimiser la longueur des pistes métalliques, en produisant un layout le plus compact possible et en prenant en compte les contraintes analogiques. En ce qui concerne la prise en compte des contraintes, nous supposons que les effets parasites inacceptables sur les performances du circuit sont évités.

Minimiser la taille du circuit aussi bien que la longueur des fils est clairement un problème d'optimisation qui peut être résolu par des algorithmes classiques tant qu'ils peuvent être traduits en une minimisation d'une fonction de coût significative. Une telle fonction de coût est généralement la somme pondérée de la taille du circuit et des longueurs des différentes branches de fils, c'est à dire:

minimiser $(\alpha \cdot \text{taille du circuit} + \beta \cdot \sum \Theta_i \cdot (\text{longueur des segments})_i)$

sous contrainte de l'ensemble des règles de dessin et des contraintes analogiques
 α et β étant les poids utilisés pour la taille et les longueurs, respectivement.

Θ_i étant la résistivité linéaire du segment de fil i .

Les contraintes du problème de minimisation résultent de la traduction des règles de dessin et des contraintes géométriques décrites ci-dessus. Ces contraintes sont d'égalités et d'inégalités linéaires entre les coordonnées des objets.

4.6.2. Méthodologie

a. Compaction mono-dimensionnelle

Pour maintenir la complexité du problème à un niveau acceptable, la compaction est accomplie en alternant les étapes de compaction à une seule dimension. A chaque étape qui compacte le circuit le long de la direction X ou Y, le graphe de contrainte est construit, à partir des relations de voisinage entre les couples d'objets qui peuvent être déplacés. La fonction de coût associée est alors minimisée amenant à de nouvelles positions des objets. Cette étape se répète alors le long de la direction orthogonale (Fig. 4.18).

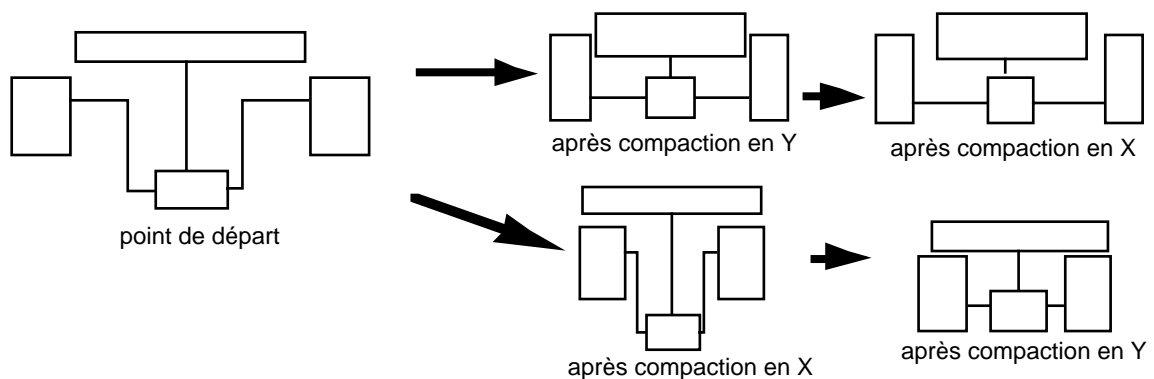


Fig. 4.18 : Différents résultats des compactations XY et YX.

Etant donné que le problème de compaction tend à être très contraint, les règles diagonales peuvent être ignorées temporairement pour permettre plus de flexibilité dans une direction de compaction. Dans un tel cas, les règles de diagonales sont prises en compte dans l'autre direction (Fig. 4.19).

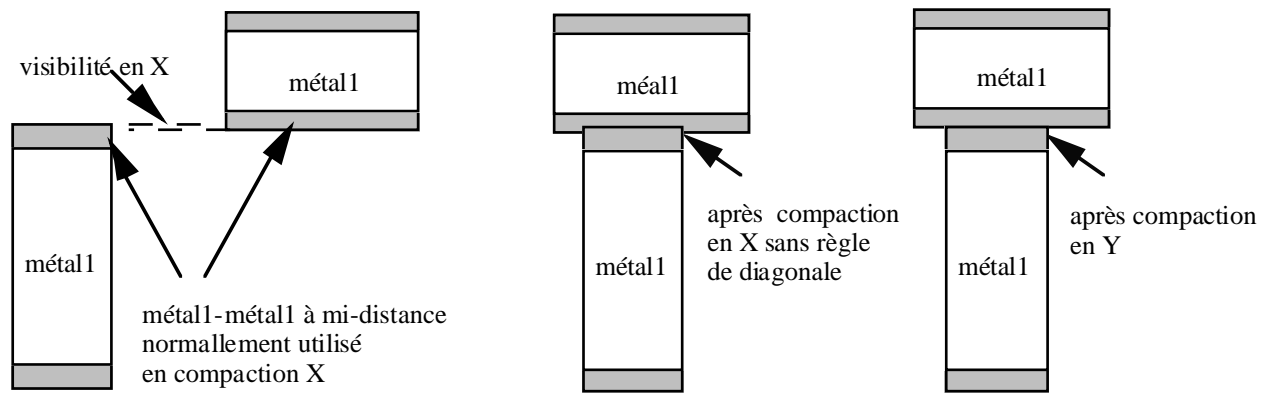


Fig. 4.19 : Relaxation des règles de diagonales.

b. Compaction Partielle

Comme le souhaitent souvent les concepteurs, le compacteur permet une compaction partielle de deux façons: il est possible soit de demander au compacteur de travailler sur une surface prédéfinie du circuit, soit de prendre en compte seulement les fils de routage pour la compaction. Dans le premier cas, tous les objets à l'intérieur de la surface choisie peuvent se déplacer en maintenant leurs connections avec les objets traversant la limite de la surface. Dans le second cas, les composants sont maintenus à des positions fixées et seule la minimisation des longueurs de fils est réalisée.

4.6.3. Description d'Objet

Chaque composant (transistors MOS, paires différentielles, résistances...) est généré par un programme générateur de composant qui prend en compte la technologie utilisée; donc les règles de dessin sont respectées par construction. La forme géométrique du composant ne change pas durant la compaction et il est par conséquent inutile de prendre en compte ces règles de dessin "internes" durant cette étape. Néanmoins, son contenu n'est pas représenté par la boîte de limitation ("bounding box") de chaque couche, mais il est plutôt la combinaison de tous les masques dont il est composé. De cette façon, il est possible de traiter des formes irrégulières, produites quand les composants fusionnent (ex.: transistors m10, m1 et m11 à la figure 4.27).

Les segments de fils métalliques sont représentés par les coordonnées de leur centre. Jusqu'à maintenant, seuls les segments horizontaux et verticaux ont été considérés. Pour une compaction à une dimension, une catégorie de segments peut être déplacée (ex.: les rectangles verticaux pour une compaction le long de la direction horizontale), et les autres doivent

s'adapter aux nouvelles positions des positions précédentes en adaptant leur position et leur longueur, en prêtant particulièrement attention aux inversions et disparitions de segments.

4.6.4. Contraintes de Descriptions

La liste des différents types de contraintes est donnée ci-dessous.

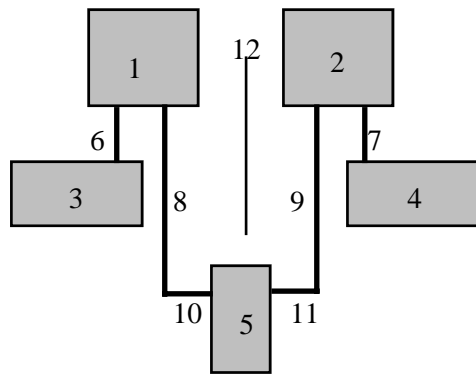
- Règles de Dessin: toutes les règles de dessin sont prises en compte, exceptées celles définissant les détails internes des composants. Elles expriment soit une distance minimum entre deux objets de deux couches données, soit une distance maximum entre deux objets tels qu'un transistor et une body-tie appliquant une polarisation du Well.

- Contraintes de Couplage de noeuds: certaines règles de distance minimum peuvent aussi être imposées (par le routeur par exemple) entre un couple de noeuds même si ces derniers sont routés avec différentes couches métalliques. Ces contraintes sont d'inégalités linéaires.

- Règles de Fusion: les règles fixant les positions relatives de certains objets. Elles consistent, par exemple, en la fusion de plusieurs transistors par leur zone de diffusion dans le but de minimiser certaine capacité de source/drain. Quand la direction de compaction est parallèle à la grille, les règles de fusion des diffusions sont traduites en règles de glissement (bornes supérieure et inférieure) pour maintenir la surface de recouvrement maximum des régions de diffusion. Quand la direction n'est pas parallèle, les règles de fusion sont d'égalités simples linéaires. Les règles de groupement d'objets (ex.: sous-bloc) sont aussi des égalités.

- Contraintes de Non-recouvrement: pendant le routage, l'utilisateur peut exiger que certaines couches de routage ne recouvrent pas les composants. Ces contraintes sont prises en compte dans le processus de compaction, de la façon suivante: pour chaque couche interdite, un rectangle de protection virtuel est créé dans chaque composant; ces rectangles sont alors traités comme des masques ordinaires.

- Règles de Symétries: elles sont traduites en égalités linéaires entre les coordonnées d'un objet physique et les coordonnées d'un pseudo-objet représentant l'axe de symétrie (Fig. 4.20).



contraintes de symétries avec un axe de symétrie vertical
 $y1 = y2$
 $y3 = y4$
 $y10 = y11$
 $x1 + x2 = x3 + x4 = 2.x12 = x6 + x7 = x8 + x9$
 $x5 = x12$

Fig. 4.20 : Exemple de contraintes de symétrie.

- Contraintes Discrètes: les contraintes de non-croisement de noeuds et de séparation de région équipotentielle méritent un traitement spécial. Le routeur est capable de minimiser le nombre de croisements entre deux noeuds mais ceci n'assure pas une situation de non-croisement. Si nous utilisons les contraintes pour le couplage de noeuds décrites ci-dessus, le compacteur n'ajoute jamais de croisements: ce qui assure que la situation n'empire pas.

Les règles de séparation entre des régions équipotentielles (en particulier entre les well) sont normalement ignorées car elles ne peuvent pas être traduites en équations linéaires (littéralement parlant, une règle telle que "la distance doit être nulle ou plus grande que x micromètres" devrait être traduite en logique ou en prédicat). Néanmoins, l'utilisateur peut demander qu'elles soient mises en vigueur, c'est à dire que la distance d'espace minimum entre les régions équipotentielles de non-recouvrement soit respectée (avec un risque d'expansion de la taille du circuit).

4.6.5. Construction des Contraintes

Le graphe des contraintes est construit en trois étapes. Premièrement, les contraintes de connectivité sont introduites: elles sont traduites en contraintes de limites supérieures ou inférieures. Puis les objets (composants, segments de fils, plots d'entrée et de sortie) sont éclatés en une structure "quad-tree". Les fonctions de recherche de l'arbre "quad-tree" sont utilisées pour scanner les couples de rectangles individuels pour lesquels des règles d'espace minimum sont appliquées. Ceci mène à un graphe plutôt dense où les redondants sont enlevés par la suite. Finalement, les contraintes géométriques analogiques (groupement d'objets, symétries, non-croisement de noeud, etc...) sont ajoutées.

Concernant les segments de fils, il est important de savoir si les segments de déplacement équipotentielles ont la permission ou non de se croiser entre eux. La réponse dépend de l'algorithme d'optimisation utilisé (voir ci-dessous). Il n'y a pas d'insertion de jog automatique: ceci doit être accompli manuellement.

Les connections de segments avec des composants ou des contacts, résultent de la même sorte de règles de glissement, comme les règles de fusion décrites précédemment. Ces règles peuvent être plus ou moins strictes (Fig. 4.21) selon les souhaits de l'utilisateur.

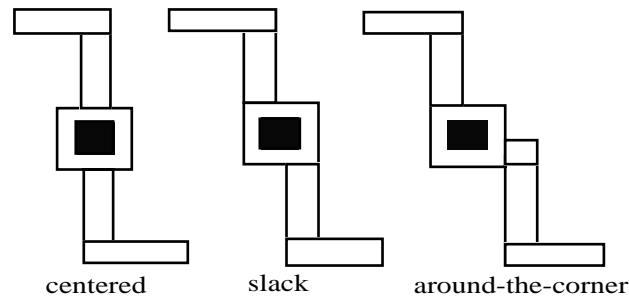


Fig. 4.21 : Différents choix de connections de contact dans le sens X.

4.6.6. Algorithmes d'Optimisation

Deux algorithmes d'optimisation sont utilisés, les deux étant capables de manipuler les contraintes exprimées comme des inégalités ou des égalités linéaires.

La taille à une dimension du circuit est une fonction linéaire de la position de l'objet le plus éloigné. La résistance globale d'un fil peut être représentée comme une fonction linéaire des coordonnées seulement si les segments qui se suivent sur une équipotentielle n'ont pas la permission de s'entrecroiser; dans ce cas la valeur absolue pondéré de la différence entre les coordonnées peut être remplacée par la différence pondéré elle-même, en supposant qu'une contrainte supplémentaire interdit que cette différence soit négative (Fig. 4.22). La fonction de coût est alors linéaire. Ceci produit un problème sur-contraint compte tenu des besoins du concepteur, mais d'un autre côté, cela conduit à résoudre un Programme Linéaire (fonction de coût linéaire et contraintes linéaires) qui peut, par exemple, être accompli de façon efficace par l'algorithme du Simplex.

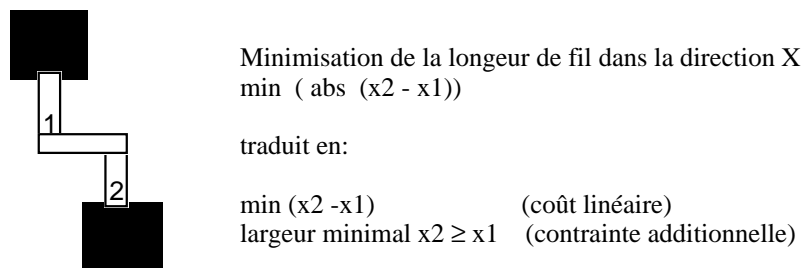


Fig. 4.22 : Minimisation de la longueur de fil utilisant un programme linéaire.

A l'inverse, si nous avons besoin d'accomplir une compaction maximum, nous ne pouvons plus maintenir une telle fonction de coût linéaire. Il est possible de calculer cette fonction comme la somme de la taille du circuit et des carrés pondérés des différences des coordonnées. Ceci est un problème moins contraignant, qui peut être résolu par la technique du Gradient Projeté proposée par Rosen en 1960. Le coût, cependant, n'est désormais plus une longueur (nous étudierons les conséquences que cela entraîne plus tard).

Dans les deux cas, l'algorithme doit commencer à partir d'une solution faisable, c'est à dire d'une solution respectant toutes les contraintes. Si le layout initial ne correspond pas à une telle situation, les coordonnées de l'objet ne peuvent pas être utilisées comme point de départ et de nouvelles coordonnées de départ sont calculées. Ceci consomme du temps mais ne fait pas une différence significative entre les deux algorithmes et peut être évité dans de nombreux cas (spécialement lorsqu'une compaction successive en boucle est exécutée).

Les coordonnées calculées par de tels algorithmes peuvent être n'importe quel nombre réel et par conséquent ne sont pas des multiples du cadrant de grille. Il a été démontré dans [70] que la solution finale respectant ces arrondis de coordonnées peut être simplement dérivée de la solution optimisée sans qu'aucune contrainte ne soit violée. Nous donnons ci-dessous les détails de ce processus.

Deux Algorithmes d'Optimisation Complémentaires:

L'algorithme du Simplex

Cet algorithme est l'algorithme de résolution des programmes linéaires le plus répandu. A partir d'une solution réalisable, l'optimum est recherché le long du chemin longeant les limites du domaine faisable (ce qui revient à suivre des sous-espaces définis par l'intersection des contraintes) (Fig. 4.23). Cette méthode est basée sur le fait que la fonction de coût étant linéaire, l'optimum est localisé sur un arête du polyèdre définissant le domaine faisable.

Le désavantage principal de cet algorithme est que la fonction de coût naturelle (une somme de valeurs absolues) doit être considérablement transformée pour obtenir une fonction de coût linéaire compatible.

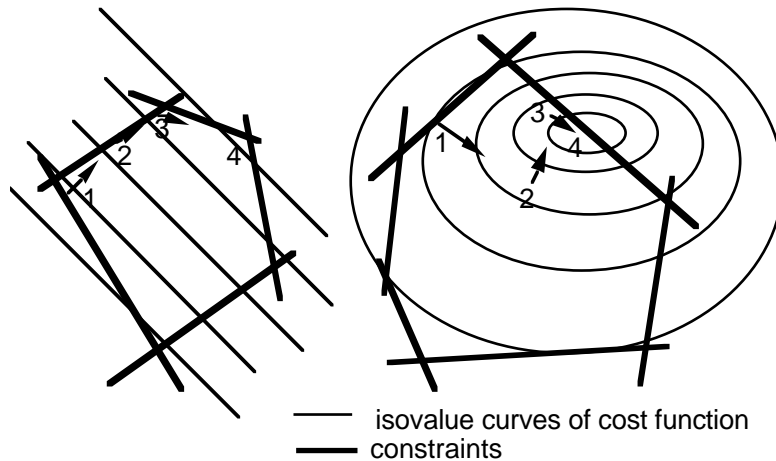


Fig. 4.23 : Comparaison de chemins d'optimisation pour Simplex (à gauche) et pour des algorithmes de Gradient Projeté (à droite).

Un autre désavantage concerne le fait que très souvent la valeur optimale de notre fonction de coût peut être atteinte via plusieurs solutions différentes qui sont essentiellement équivalentes. En pratique, les solutions obtenues utilisant notre version du Simplex ne contiennent pas de décrochement des fils (cette caractéristique peut être utilisée pour “nettoyer” la solution faisable) (Fig. 4.24).

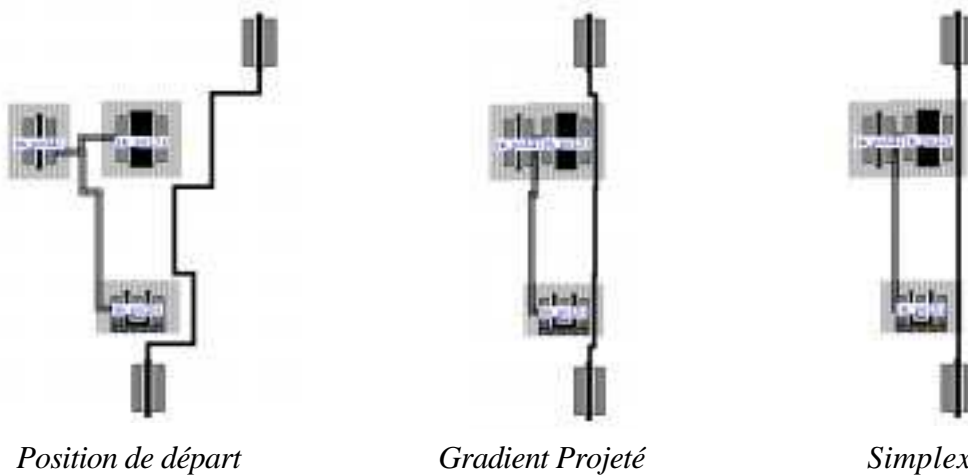


Fig. 4.24 : Comparaison de deux algorithmes appliqués à une structure simplifiée.

Des résultats théoriques obtenus dans les années 80 montrent que dans la plupart des cas concrets le temps de résolution est acceptable (quelques minutes pour une station de travail de puissance moyenne) jusqu'à plusieurs milliers de contraintes et plusieurs centaines de variables.

L'algorithme de Gradient Projeté

Cet algorithme est plus ou moins un mélange de l'algorithme de gradient (classe de descente la plus raide) et de l'idée du chemin de Simplex: la méthode de descente raide est utilisée, et quand il atteint une ou plusieurs contraintes, le chemin est obligé de suivre les limites du domaine. Ce choix est imposé par la fonction de coût non-linéaire qui peut avoir un optimum à l'intérieur du domaine (ce qui est rarement le cas en pratique). Déplacer la solution le long des limites est difficile étant donné que cela implique le calcul des projections du gradient sur l'intersection de plusieurs hyperplans. Cependant, des procédures d'analyse numérique efficaces ont été proposées en littérature pour résoudre ce problème [71]. Cet algorithme visant à résoudre des problèmes plus compliqués, il est davantage consommateur de temps que le Simplex.

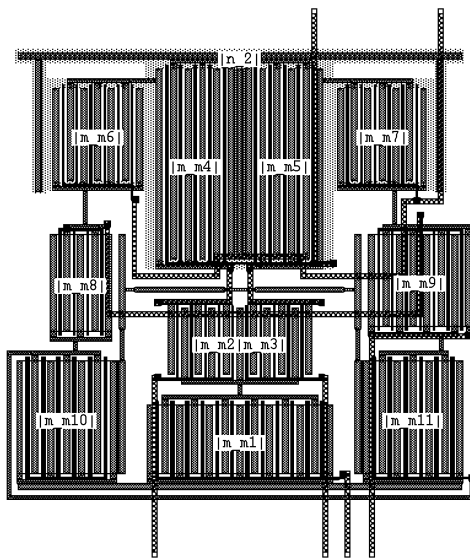
Finalement, il apparaît que l'algorithme de Gradient Projeté produit une solution optimum unique mais qui n'est pas la plus propre (c'est à dire minimisant le nombre de décrochement). Par contre, le problème est sur-contraint pour l'algorithme du Simplex. Notre expérience pragmatiques montre que résoudre le problème premièrement par l'algorithme de Gradient Projeté, permet le maximum de mouvements des éléments dans le bloc, et ensuite on peut appliquer l'algorithme du Simplex sur les fils uniquement pour "nettoyer" les décrochements inutiles.

4.6.7. Traitement du layout

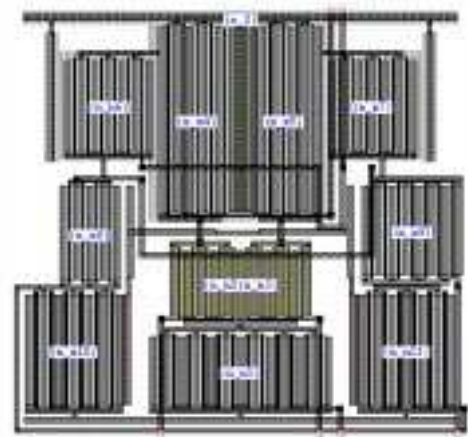
Le travail d'optimisation produit une liste de coordonnées associées aux nouvelles positions des objets déplacés. La procédure suivante, pour obtenir le nouveau layout, est un appel automatique des fonctions interactives de l'outil, et chaque objet est déplacé de sa position initiale à sa nouvelle position, en accomplissant tous les aboutements nécessaires et les arrondis de coordonnées. Réutiliser cette part interactive pour notre propos a permis de concentrer le développement du compacteur sur les contraintes de construction et sur l'optimisation numérique, et assure une cohérence de ces résultats avec des résultats manuels. Les fonctions de déplacements interactives accomplissent déjà la plupart des travaux de "nettoyage" décrits en [72] et sont simplement réutilisées.

4.7. RESULTATS EXPERIMENTAUX

Le tableau 1 présente la taille des problèmes sur des exemples réels (Fig. 4.25). Les temps CPU incluent trois compactations à une dimension (deux utilisant l'algorithme PG et une utilisant l'algorithme Simplex) à partir d'une solution non-faisable.



Départ



Arrivée

Fig. 4.25 : Compaction de l'amplificateur opérationnel Folded cascode à symétrie partielle.

Circuit	Nombre de variables		Nombre de contraintes		Temps de CPU (SUN sparc 20)
	X	Y	X	Y	
Cascoden (Fig. 4.25)	79	77	419	339	86 s
Op-amp cascodifn	127	105	604	566	214 s
Op-amp (Fig. 4.27)	196	168	1056	855	20 min

Tableau 1 : Tableau comparatif.

Pour satisfaire aux requêtes de l'utilisateur en termes de flexibilité, il est possible de ne router qu'une partie des noeuds, ou une partie du circuit: une surface de routage peut être spécifiée graphiquement par un rectangle délimitant la surface de routage. La compaction peut aussi être accomplie localement; la minimisation de la longueur de fil peut être achevée sans déplacer les composants.

La fonction de coût peut être aisément modifiée, de même que les priorités de noeuds et l'algorithme de compaction. Ces installations sont aujourd'hui largement utilisées par notre compagnie pour le dessin des cellules analogiques [73].

Utilisation de CHIRVAN dans la réalisation du layout automatique d'un circuit réel

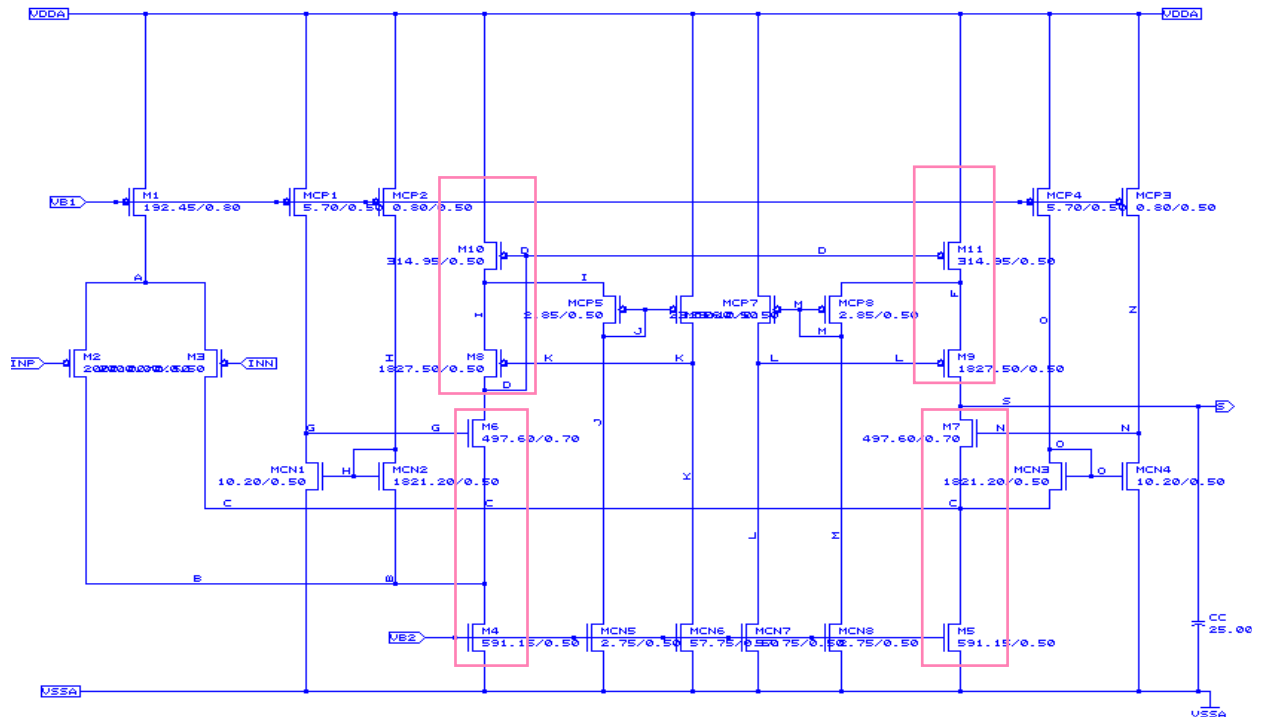


Fig. 4.26 : Schéma de l'amplificateur opérationnel à transistors composites.

Les figures suivantes (Fig. 4.27) illustre l'utilisation de l'outil dans un cas réel. Le circuit est un amplificateur opérationnel différentiel transcomposite que nous avons conçu pour intégrer un convertisseur analogique/numérique CAN à basse tension d'alimentation et à faible consommation (chapitre 2). La structure de l'amplificateur a été construite à l'aide d'une technique de transistors Composites (Fig. 4.26). Tout d'abord, un placement est exécuté grossièrement par l'utilisateur, avec suffisamment d'espace pour faciliter le travail du routeur. Le routage est alors complété et finalement, la compaction est mise à exécution dans la direction Y puis dans la direction X, en respectant les contraintes diagonales. Le layout de cette cellule a été réalisé automatiquement sur Chirvan; le processus de sa création a duré environ 60 min au lieu de six jours, dans le cas de layout manuel. Ce circuit est totalement symétrique ("fully differential"); il est donc très sensible aux contraintes de symétrie et d'appairage. De ce point de vue, Chirvan a atteint les objectifs recherchés puisqu'il respecte ces contraintes analogiques, comme le montre le layout de la cellule. Le routage et la compaction des fils sont en effet symétriques. Le circuit a été fabriqué en technologie CMOS à 0,5 μm 3 volt [34]. D'autres circuits ont été aussi réalisés au CNET Grenoble grâce à cet outil.

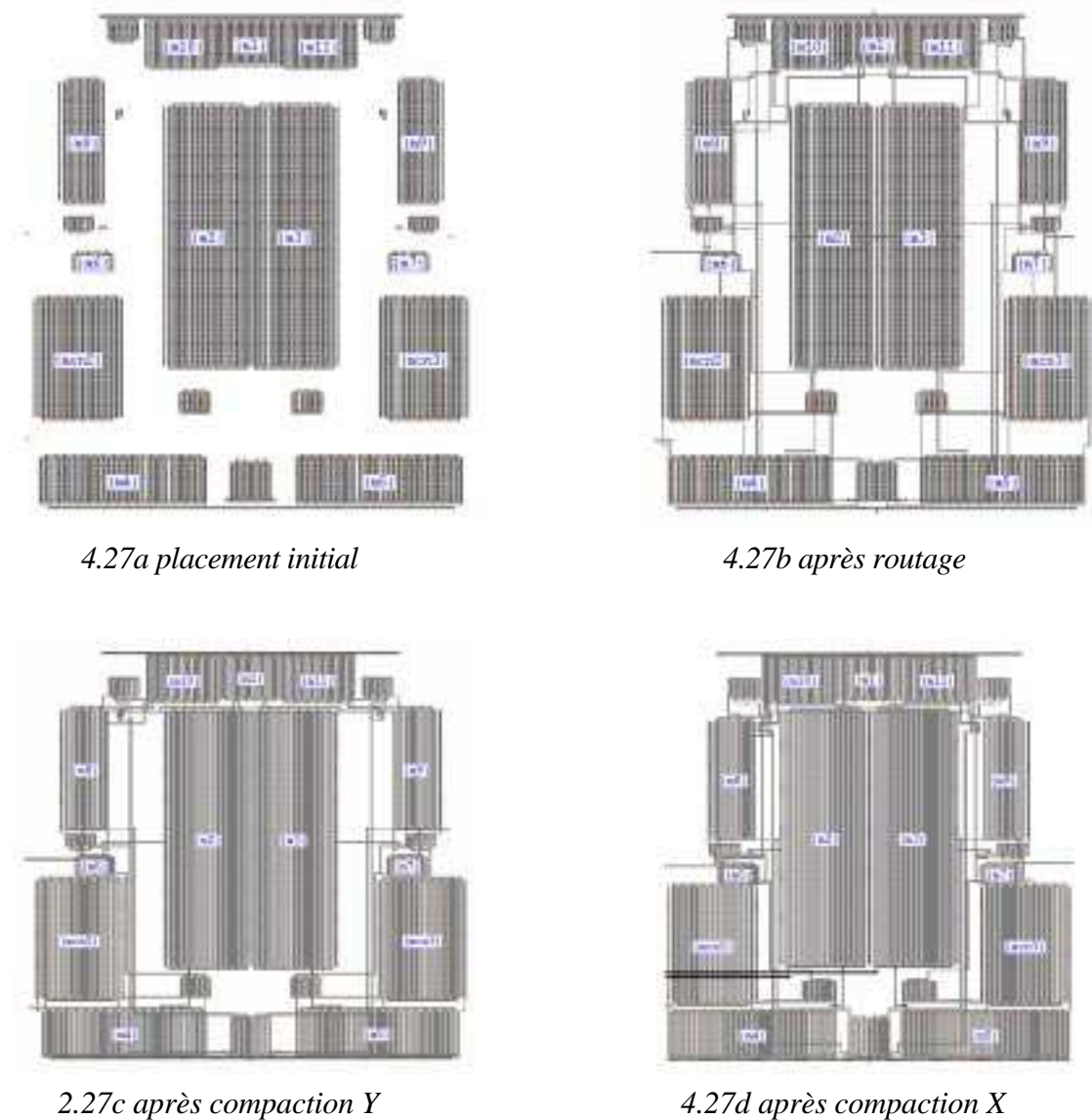


Fig. 4.27 : Processus automatique du layout de l'amplificateur opérationnel à transistors composites.

4.8. CONCLUSION

Une approche pragmatique du problème du dessin du layout de circuit analogique a été présentée. En considérant le fait que nombre de circuits sont encore aujourd'hui dessinés manuellement, et que toutes les contraintes ne sont pas explicitement statuées, un outil interactif a d'abord été créé. Cet outil a évolué vers une plus grande automatisation, mais en même temps a assuré que les installations interactives ne soient pas perdues. Nous considérons cela comme le seul moyen pour que cette tendance vers une plus grande automatisation soit acceptée par l'utilisateur final.

Pour obtenir une harmonieuse cohabitation des dessins manuel et automatique, l'outil Chirvan et ses différentes composantes (le Générateur, le Placeur, le Routeur et le Compacteur) ont été créés de telle façon qu'à n'importe quelle étape du processus de dessin du masque, l'utilisateur soit maître de la situation. Il a le choix d'utiliser ces outils automatiquement, ou d'intervenir manuellement pour changer partiellement le placement ou le routage, l'outil se chargeant de la partie restante du circuit. Il peut aussi arrêter l'outil et rajouter d'autres contraintes, et laisser l'outil automatique continuer son travail, à partir de cette intervention. L'expertise de l'utilisateur reste toujours dominante. L'outil est capable de trouver une solution correcte et acceptée par l'utilisateur. La flexibilité de pouvoir intervenir ou guider l'outil vers une solution optimale est un des points forts de Chirvan.

La cohérence entre ces outils est assurée par trois points: premièrement, ces outils doivent partager la même description des contraintes à respecter ; deuxièmement, le routeur doit être capable de traiter avec les résultats du placeur, et le compacteur doit aussi être capable de traiter avec les résultats du routeur; et finalement, le compacteur ne doit pas défaire ce que le routeur a fait.

Ce travail s'inscrit dans la tendance à automatiser progressivement les fonctions répétitives et ennuyeuses, et offre un nombre croissant d'installations automatiques, tout en maintenant la possibilité d'interaction directe avec l'utilisateur à n'importe quelle étape du processus de dessin. Nous avons donc présenté dans ce chapitre, les différents outils d'automatisation du layout qui composent l'outil Chirvan à l'exception de l'outil du placement automatique qui fera l'objet du chapitre 5, et l'outil de routage des blocs entre eux ("top level") (Chirvan routeur).

Chapitre V

Chapitre 5

Automatisation du Placement des Circuits Intégrés Analogiques

5.1. INTRODUCTION

Ce chapitre traite du problème de l'automatisation du placement des circuits intégrés analogiques à haute performance. Le placement est une phase cruciale pour éviter la dégradation des performances du layout des circuits analogiques. En effet, elle influence directement tous les phénomènes parasites du layout exposés au chapitre 3. C'est pendant l'étape de placement que la distance entre les composants appairés ainsi que leur degré d'appairage sont déterminés. Le placement détermine aussi le profil thermique du circuit. De plus, la phase de placement influence largement les valeurs des interconnexions parasites, même si leurs valeurs finales sont déterminées durant la phase de routage, car leurs valeurs minimales effectives sont fixées par la configuration du composant, elle-même déterminée durant la phase de placement. Un algorithme de placement basé sur les contraintes de performance doit, par conséquent, prendre simultanément en compte tous ces effets de dégradation des performances.

Nous présenterons tout d'abord les problèmes rencontrés lors du placement ainsi que les diverses contraintes analogiques que cette étape doit respecter afin d'atteindre les spécifications du cahier des charges (§ 5.2). Puis, nous donnerons un aperçu général de l'outil de placement (§ 5.3). Afin de pouvoir choisir la méthode de placement qui sera la plus adaptée à la diversité des contraintes analogiques, nous présenterons brièvement les différentes techniques de placement et nous les comparerons entre-elles (§ 5.4).

A partir de ces comparaisons, nous sélectionnerons le meilleur algorithme d'optimisation du placement des cellules analogiques, et son implémentation sera discutée au paragraphe 5.5. Les paragraphes 5.7 et 5.8 s'intéresseront plus particulièrement à certains points importants concernant l'implémentation du placement, le modèle de placement, la manipulation des contraintes analogiques, le déplacement des éléments et la fonction de coût. Enfin, nous présenterons nos conclusions au paragraphe 5.11.

5.2. FORMULATION DU PROBLEME

Le placement de composants doit être basé sur une analyse des contraintes électriques et topologiques à respecter, afin de minimiser à la fois, la surface de silicium occupée et la valeur des éléments parasites et donc, de fournir en un temps minimum une solution finale optimale. Les contraintes de performances sont toujours données par les concepteurs eux-mêmes suivant un cahier des charges bien défini. Ces contraintes sont obtenues grâce à la simulation et l'optimisation des circuits électriques (schémas électriques) choisis par les concepteurs pour remplir une fonction donnée. Le problème peut donc être décrit de la façon suivante: après avoir saisi le schéma d'un circuit défini comme un ensemble d'une quarantaine de composants (transistors, capacités et résistances), (exemple: amplificateur opérationnel, comparateur), et après avoir produit le fichier de simulation correspondant (netlist) des terminaux d'interconnexion fixés sur le bord des composants et sur la périphérie du circuit lui-même, il s'agit de sélectionner pour chaque composant une forme géométrique optimale, et de générer un programme capable de dessiner, de manière optimale, cette forme, en tenant compte des règles de dessin de la technologie et de la contrainte de minimisation de la surface totale de silicium occupée, afin de pouvoir router le circuit ultérieurement. Il faut ajouter à cette définition de base les contraintes et les objectifs suivants:

* **Contraintes de Symétrie**

Dans les circuits analogiques à haute performance, il est donc souvent exigé que des groupes de composants soient placés de façon symétrique, en respectant un ou plusieurs axes de symétrie. Un placement symétrique permet un routage symétrique et entraîne un appairage des parasites. On peut formuler les contraintes de symétrie en termes de *couples*, *composants auto-symétriques* et *groupes symétriques*. Deux composants placés symétriquement par rapport à l'axe de symétrie forment un couple. Un composant auto-symétrique est un composant placé sur l'axe de symétrie. Un groupe symétrique est un ensemble de couples et de composants auto-symétriques qui partagent le même axe de symétrie. Le groupe symétrique présenté à la figure 5.1 est constitué des couples (M1, M2) et (M3, M4), et du composant auto-symétrique M5. On peut définir plusieurs groupes symétriques pour un même circuit. Ces différentes définitions se traduisent au niveau du placement par les démarches suivantes:

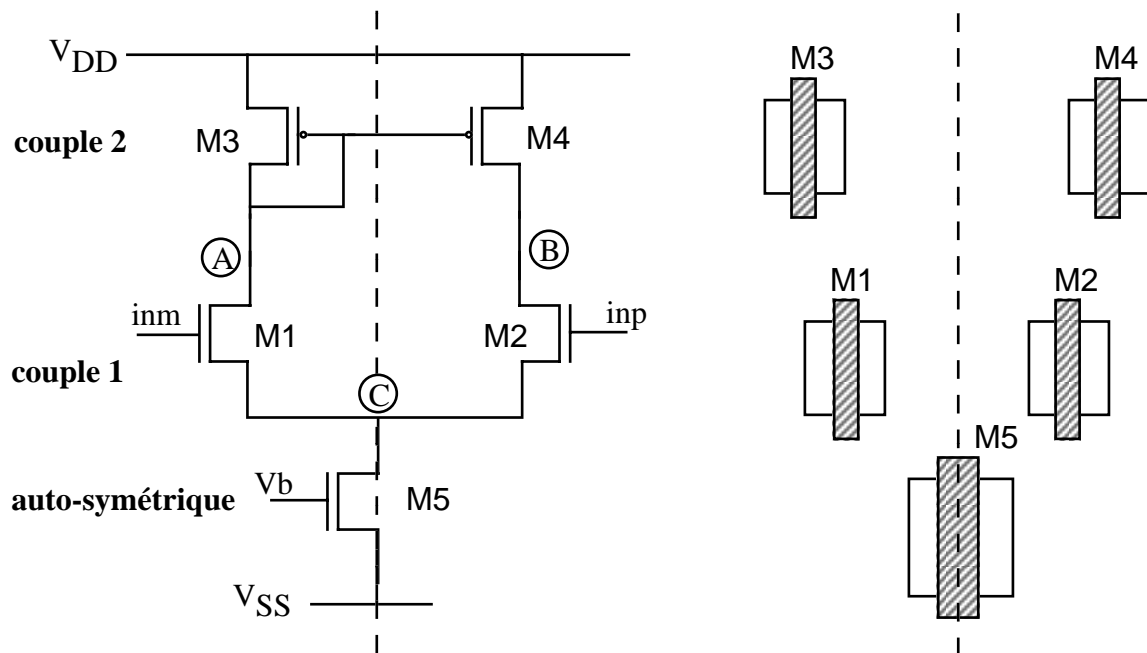


Fig. 5.1 : Fragment d'un circuit et son layout correspondant, illustrant les contraintes de symétrie.

- Deux composants définis comme un couple doivent être placés symétriquement par rapport à leur axe de symétrie et doivent posséder des formes identiques et des orientations en miroir.
- Un composant spécifié comme étant auto-symétrique doit être placé sur l'axe de symétrie.
- Les couples et les composants auto-symétriques appartenant au même groupe de symétrie doivent partager le même axe de symétrie.

* Contraintes d'appairage (Matching)

Les circuits intégrés analogiques sont souvent basés sur la connexion de réseaux devant être parfaitement symétriques. On peut spécifier les contraintes d'appairage en définissant un groupe d'appairage. Un groupe d'appairage est un ensemble de deux ou plusieurs composants pour lesquels un rapport approprié des caractéristiques des composants est requis. Le cas le plus simple et le plus commun de groupe d'appairage correspond à une paire de composants identiques. Un cas plus complexe de groupe d'appairage est présenté à la figure 5.2. On peut définir un nombre quelconque de groupes d'appairage dans un circuit analogique. Au niveau du placement, la définition d'un groupe d'appairage entraîne comme conséquences:

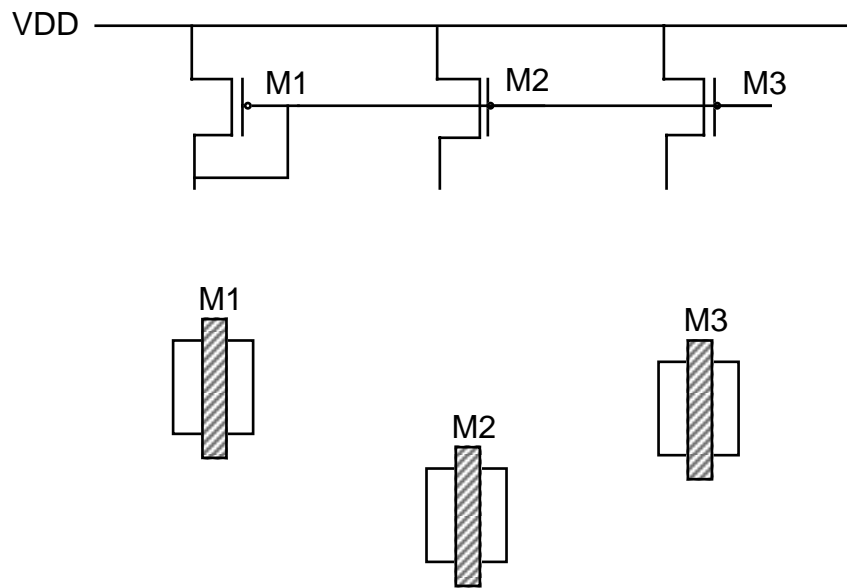


Fig. 5.2: Fragment d'un circuit et son layout correspondant, illustrant les contraintes d'appariement.

- Tous les composants qui appartiennent au même groupe d'appariement doivent être orientés selon la même direction.
- Si tous les composants d'un groupe d'appariement sont identiques (rapport 1 pour 1), ils doivent être implémentés avec des dessins de masque identiques (formes identiques). S'ils ont un autre rapport, ils doivent être constitués de cellules élémentaires identiques, conformément à ce rapport.
- L'outil de placement doit déterminer les positions et, par conséquent, il doit aussi déterminer la distance entre les composants appariés de telle façon que les contraintes de performance du circuit soient obtenues. Etant donné qu'il n'est pas toujours possible pour le layout de circuit analogique de satisfaire en même temps à toutes les conditions de symétrie, de placer directement les composants appariés les uns à côté des autres et d'obtenir un layout compact acceptable, le degré d'appariement d'une paire de composants doit être sélectionné en vue de son influence sur la performance du circuit.

* **Contraintes de dégradation des Performances**

Comme nous l'avons montré au chapitre 3, les performances du circuit sont influencées par les effets parasites du layout, dont voici les trois principaux:

- **les parasites d'interconnexion**

Un algorithme de placement basé sur les performances, doit créer un placement qui permette au routeur d'effectuer les interconnexions en respectant les contraintes de

performances. Bien que les valeurs réelles des capacités et des résistances d'interconnexion soient déterminées durant la phase de routage, leurs valeurs minimums effectives sont fixées durant le placement. Il est par conséquent capital de prendre en compte pendant le placement l'estimation de la dégradation de performance induite par les parasites d'interconnexion.

– **non-appairage des composants**

La distance entre les composants à appairer doit être choisie en fonction de son influence sur les performances du circuit.

– **les effets thermiques**

La présence de composant dissipant de la puissance dans un circuit induit une distribution de la température à travers le placement. Etant donné que les caractéristiques électriques des composants sont influencées par la température locale, les composants appairés doivent être placés de telle façon que la dégradation de performance provoquée par leurs différences de température, reste dans les limites des spécifications. Il s'agit donc de placer les composants sur la même ligne isotherme, de manière à respecter une symétrie axiale par rapport à la source de chaleur. Toutefois, il n'est pas toujours possible de placer les composants appairés exactement en milieu isotherme, tout en satisfaisant dans le même temps aux autres contraintes.

* **Contraintes géométriques**

Les blocs générés par un outil générateur de layout au niveau du circuit font souvent partie d'un système très large. Pour minimiser la dégradation de performance au niveau du système, un outil de floorplanning (outil de représentation et de manipulation du plan directeur) où le concepteur lui même prédétermine un rapport de cadrage tel que les hauteurs et/ou les positions des terminaux soient fixées. Ces contraintes géométriques supplémentaires doivent également être prises en compte durant la phase de placement.

5.3. VUE GENERALE DE L'OUTIL DE PLACEMENT

L'architecture du programme de placement est illustrée à la figure 5.3. Les données d'entrée de l'outil sont la netlist du circuit après calibrage ainsi que la liste des spécifications des performances du circuit que celui-ci doit obtenir (ex: marge de phase > 60°), et le fichier des paramètres de la technologie utilisée.

La première étape d'exécution du programme de placement consiste en un certain nombre de simulations numériques réalisé par le simulateur Eldo et son optimiseur Opart. Ces deux outils génèrent des fichiers de sortie, dont deux sont les plus importants pour les outils de layout automatique. Ces fichiers sont le fichier (.OPZ) qui contient les informations concernant

les performances du circuit (gain, marge de phase, dynamique de sortie, temps d'établissement, ...), et le fichier (.INP) qui contient les informations concernant la structure du circuit (branche de courant, paire différentielle, source de courant, branches symétriques,...) (cf. annexe A5). Ces informations, ainsi que la netlist du circuit, sont alors utilisées comme données d'entrée pour un ensemble de générateurs de modules qui construit une liste de formes géométriques pour chaque composant et pour le placeur automatique.

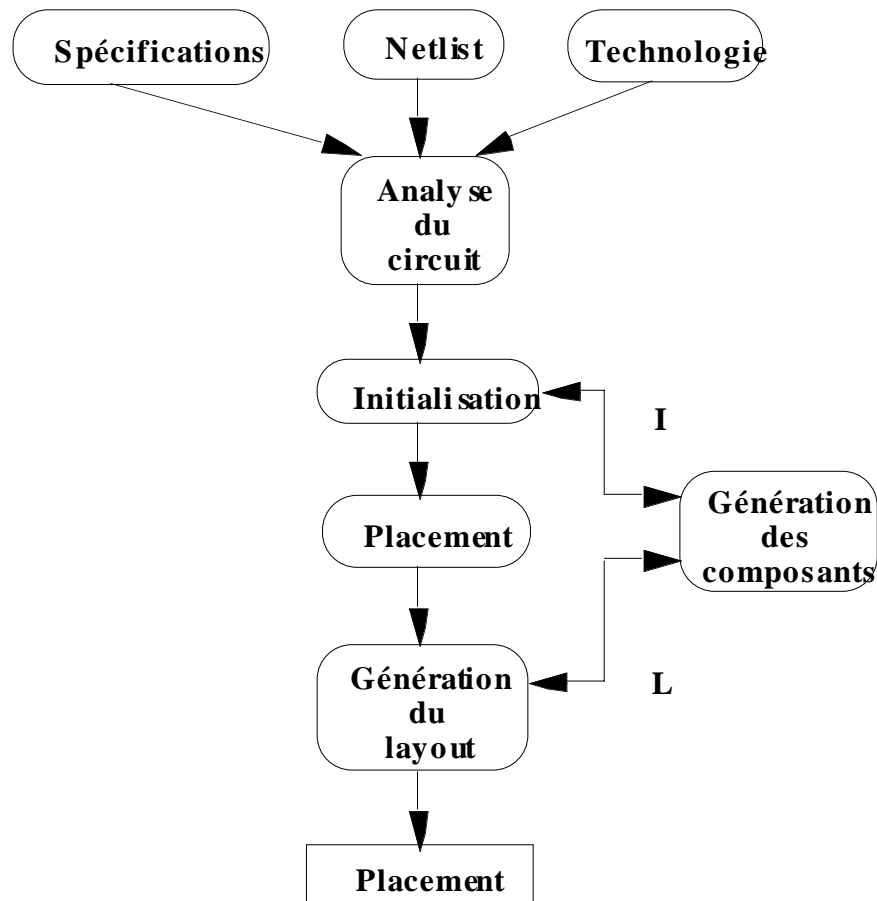


Fig. 5.3 : Méthodologie de placement automatique du layout analogique.

Nous verrons plus tard comment l'algorithme de recuit simulé peut générer le placement réel, en prenant en compte toutes les contraintes et objectifs identifiés au paragraphe précédent.

Après l'optimisation, les générateurs de modules interviennent une fois de plus (cette fois-ci en mode layout) pour créer un nouveau layout des formes géométriques sélectionnés, ce qui achève le layout final. La sortie du programme est alors le placement final du layout, ainsi que les informations concernant la dégradation des performances dans ce layout final.

5.4. PRESENTATION DES DIFFERENTS ALGORITHMES UTILISES POUR RESOUDRE LE PROBLEME DE PLACEMENT

Le nombre des contraintes identifiées aux paragraphes précédents ainsi que leur diversité explique la complexité du problème de placement en analogique. Le problème majeur, à savoir la disposition d'un ensemble de blocs connectés sur une surface de dessin, a été étudié de façon intensive dans le contexte de placement relatifs aux circuits VLSI ou pour résoudre des problèmes de mise en forme (floorplanning). Ces problèmes de placement sont connus pour être NP-complets[74]. A cause de cette complexité, il est nécessaire pour les résoudre, d'utiliser les algorithmes heuristiques. Il serait trop long de présenter ici tous les algorithmes de placement utilisés dans le passé. Le lecteur pourra se référer aux travaux effectués dans [75][76] et aux indications qu'on y trouve. Dans ce chapitre, nous donnerons un aperçu des classes principales d'algorithmes, comprenant une rapide description de leur principe de fonctionnement et une présentation de leurs principales caractéristiques. Cette vue d'ensemble nous permettra de justifier notre choix du recuit simulé comme algorithme de base pour le placement basé sur les performances analogiques.

Remarque

Un problème est dit de complexité P s'il existe un algorithme polynomial (demandant un nombre d'opérations de l'ordre de n^α où n est la taille du problème et α un entier) pour le résoudre. A défaut de connaître un tel algorithme, il se peut que l'on puisse vérifier en un temps polynomial qu'une solution donnée par un tiers fournit la réponse au problème. Ce dernier est alors qualifié de NP ("non-deterministic polynomial"). Le problème d'optimisation classique du "voyageur de commerce" est de classe NP. Notons que tout problème polynomial est aussi un problème NP. Savoir si la réciproque est vraie ou fausse est encore à l'état de recherche. Enfin, parmi les problèmes de classe NP, on distingue ceux qui sont complexes en ce sens que si l'on découvre un algorithme permettant d'en résoudre un en un temps polynomial, on sera à même de résoudre tous les autres en un temps lui aussi polynomial. On qualifie de tels problèmes, de NP-complets. En pratique, un problème de complexité NP demande un temps de résolution augmentant exponentiellement avec la taille du problème.

5.4.1. Placement Constructif

Cette méthode vise à construire progressivement le plan directeur du circuit à l'aide de l'information de connectivité [77]. Il s'agit au départ de choisir un module et de le placer sur la surface de dessin. A chaque étape suivante, un nouveau module est choisi et placé au meilleur emplacement disponible, c'est à dire en respectant certaines contraintes et en recherchant une

surface minimale du layout. L'évolution du système est dirigée soit par des règles d'experts soit par le concepteur lui-même, qui doit alors mettre en oeuvre ses connaissances dans le domaine. Cette technique n'est pas très performante si l'espace des possibilités est grand et si on exige de cette approche des opérations complexes.

5.4.2. Placement Dirigé par les Forces

Les algorithmes de placement dirigé par la force sont riches en diversité et diffèrent fortement en ce qui concerne les détails d'implémentation [77]. En général, ils traitent le problème de placement comme un problème de mécanique classique de système et de corps attachés à des ressorts. Les blocs connectés entre eux par des noeuds exercent des forces attractives les uns sur les autres. L'amplitude de ces forces est directement proportionnelle à la distance entre les blocs. La configuration idéale du placement des blocs est celle pour laquelle le système atteint l'équilibre. En utilisant cette analogie, nous pouvons donc considérer le problème de placement comme un problème de mécanique classique et les différentes méthodes utilisées en mécanique classique peuvent lui être appliquées. Certaines de ces méthodes sont constructives, d'autres sont basées sur l'amélioration itérative.

5.4.3. Placement par Partition

L'algorithme de placement basé sur la décomposition (partition) génère un placement en subdivisant le circuit à plusieurs reprises de telle façon que le nombre de noeuds coupés par la partition soit minimisé. Simultanément, la surface de dessin disponible est décomposée alternativement dans les sens horizontal et vertical et chaque sous-circuit est assigné à une division de surface du dessin. Ce processus est répété jusqu'à ce que chaque sous-circuit ne soit plus constitué que de cellules de bases et qu'il ait une place unique sur la surface de dessin. La plupart des méthodes de placement basées sur les partitions ("min-cut algorithms") utilisent des formes modifiées des heuristiques de partition de Kernighan-lin [78] ou de Fiduccia-Mattheyses [79] .

5.4.4. Optimisation Quadratique

Les techniques d'optimisation quadratiques génèrent un placement ayant une longueur de noeuds minimum en résolvant le problème de la minimisation quadratique. Les modules sont représentés par des points qui doivent être placés sur la surface de dessin. Pour modéliser un noeud, on utilise des modèles de graphe-complet. Si la norme de distance euclidienne (quadratique) est utilisée pour modéliser la longueur de noeud, le problème de la minimisation

de la longueur totale de noeud peut être résolu comme un problème de minimisation quadratique.

5.4.5. Placement Génétique

La technique du placement génétique utilise une méthode itérative permettant de sortir des minimums locaux de la fonction de coût, qui imite les lois de la sélection naturelle rencontrée en biologie génétique, en tant qu'elle progresse vers une solution de placement optimale. Un groupe initial de configurations de placement est tout d'abord généré aléatoirement. Ce groupe initial constitue la population de départ. Cette population s'améliore itérativement en utilisant une procédure qui imite le processus naturel de l'évolution. Le coût de chaque placement étant calculé, les individus de "mauvaise qualité" (coût trop élevé) sont éliminés puis remplacés par de nouveaux placements construits à partir de parents de "bonne qualité" choisis parmi les individus non préalablement éliminés. Plus la qualité d'un individu est élevée, plus la probabilité qu'il soit sélectionné comme parent est grande. Par souci de simplicité, la population est gardée constante. Le processus itératif est ensuite répété jusqu'à "mutation". En effet, pour éviter de s'enfermer dans un minimum local, certains individus de mauvaise qualité sont acceptés, ou des défauts sont introduits parmi divers individus en appliquant aux parents un certain nombre d'opérateurs génétiques (crossover, mutation et inversion) pour créer de nouveaux individus que nous appelons les "offspring". Les offspring sont ensuite évalués et on forme une nouvelle génération en sélectionnant certains des parents et des offspring sur leurs critères de qualité. L'individu de la population finale qui possède la plus grande qualité représente la meilleure solution de placement.

5.4.6. Recuit Simulé

Le problème du placement est par nature un problème d'optimisation combinatoire, et de surcroît de complexité NP. La recherche de la combinaison optimum se fait par la recherche du minimum d'une fonction de coût (ou énergie) modélisant la qualité du placement. La méthode du recuit simulé permet de s'approcher très fortement de cette méthode en un temps qui peut être polynomial. Le problème d'optimisation combinatoire est présenté par Van Laarhoven dans [80] comme une paire (R, C) où R est le groupe de configurations (encore appelé "espace de configuration") fini (ou infini mais dénombrable), et C une fonction de coût, $C : R \rightarrow \mathfrak{R}$, qui attribue un nombre réel à chaque configuration. Nous supposons, sans perte de caractères généraux, que C est définie de telle façon qu'à sa valeur la plus basse corresponde la meilleure configuration correspondante (en termes de critères d'optimisation). Résoudre un problème

d'optimisation combinatoire revient à trouver une configuration pour laquelle C a une valeur minimale, c'est à dire une configuration i_0 tel que:

$$C_{opt} = C(i_o) = \min_{i \in RC(i)} \quad (5.1)$$

où C_{opt} désigne le coût optimum (minimum).

La méthode du recuit simulé repose sur l'analogie existant entre les problèmes d'optimisation combinatoire et la mécanique statistique régissant la physique des matières condensées. Elle peut être décrite en utilisant le parallèle physique avec le processus de refroidissement des métaux lors de leur traitement thermique. Ce processus se présente de la façon suivante: si la température initiale est suffisamment élevée pour permettre l'exploration d'un grand nombre de configurations sans s'enfermer, dès le départ, dans un minimum local, et si le refroidissement s'effectue assez lentement pour ne pas figer le placement dans une configuration désordonnée (trempe) et pour ne pas demander trop d'itérations à chaque palier pour atteindre l'équilibre thermodynamique, les particules s'arrangent entre elles à l'état cristallin désiré. L'énergie du système d'atomes est alors minimale. La théorie de Boltzmann montre qu'un solide est à l'état d'équilibre thermodynamique si la probabilité de sa distribution d'énergie (états possibles) est égale à :

$$P\{energie = E\} = \frac{1}{Z(T)} \exp\left(-\frac{E}{k_B T}\right) \quad (5.2)$$

où $Z(T)$ est un facteur de normalisation dépendant de la température T et k_B la constante de Boltzmann. On peut déduire de l'équation (5.2) que la probabilité d'avoir un état énergétique bas augmente lorsque la température diminue.

Métropolis et ses confrères proposent dans [81] un algorithme qui simule l'évolution vers l'équilibre thermique d'un solide pour une valeur de T fixée. L'état solide, caractérisé par la disposition de ses particules, est modifié à plusieurs reprises par l'application d'un petit déplacement aléatoire sur une particule choisie aléatoirement. Si la différence d'énergie ΔE entre l'ancien état et le nouvel état est négative, le mouvement est immédiatement accepté car il a conduit à une amélioration de la qualité du placement. Le processus se poursuit avec le nouvel état. Si la différence d'énergie ΔE est positive ou égale à zéro, la probabilité pour que le mouvement soit accepté est égale à : $\exp(-\Delta E / k_B T)$. On appelle cette règle d'acceptation le critère de Métropolis. Si ce critère est appliqué, après un nombre important de perturbations, le système évolue vers un état d'équilibre thermique caractérisé par une répartition de l'énergie donnée par l'équation (5.2). L'algorithme de Métropolis peut donc être utilisé pour simuler

l'évolution d'un solide vers l'équilibre thermique. En appliquant cet algorithme à des valeurs de températures de plus en plus basses, on peut simuler le processus de recuit d'un solide.

Il existe une analogie entre l'algorithme du recuit simulé et le processus de refroidissement des métaux lors de leurs traitement thermiques, développé précédemment. L'état du solide représente les configurations du problème d'optimisation. Le paramètre de contrôle de l'algorithme peut être assimilé à une température tandis que sa fonction de coût C , associée à une configuration particulière, prend le rôle de l'énergie E qui doit diminuer au fur et à mesure que la température décroît. Un paramètre de contrôle T est introduit pour jouer le rôle de la température. L'algorithme peut maintenant être décrit de la façon suivante: tout d'abord, on donne au paramètre de contrôle T une valeur élevée et on génère une séquence de configuration à l'aide de l'algorithme de Métropolis. A partir de la configuration courante i , on choisit une nouvelle configuration j à l'aide d'un mécanisme de génération, c'est à dire d'une prescription qui génère une transition d'une génération à une autre par une petite perturbation. Soit ΔC_{ij} la différence de coût entre les deux configurations correspondant à $C(i) - C(j)$, la nouvelle configuration est acceptée avec une probabilité de 1 si $\Delta C_{ij} \leq 0$, et avec la probabilité $\exp(-\Delta C_{ij}/T)$ si $\Delta C_{ij} > 0$. Le processus se poursuit jusqu'à ce que l'équilibre soit atteint, c'est à dire jusqu'à ce que la distribution de probabilité des configurations approche la distribution de Boltzmann, alors donnée par l'équation suivante:

$$P\{\text{configuration} = i\} = \frac{1}{Q(T)} \exp\left(-\frac{C(i)}{T}\right) \quad (5.3)$$

où $Q(T)$ est une constante de normalisation dépendant du paramètre de contrôle T .

Le paramètre de contrôle est alors abaissé par paliers, de telle façon qu'à chaque palier, la système approche l'équilibre thermodynamique, c'est à dire que la distribution de probabilité des configurations soit quasiment stable, comme décrit précédemment. Le placement a atteint une configuration stable lorsque son coût moyen $\langle E \rangle$ ne varie pas d'un palier à un autre. Le placement est alors arrêté (en général après trois paliers consécutifs sans changement de qualité) et on considère la configuration *gelée* comme la solution du problème d'optimisation.

5.4.7. Discussion

A partir de la description du problème donnée au paragraphe 5.2, nous pouvons dégager les caractéristiques souhaitées pour l'algorithme de placement analogique:

* **1.** La plupart des composants des circuits intégrés analogiques peuvent être dessinés de différentes façons. Cependant, l'algorithme doit être capable de sélectionner simultanément la position, l'orientation et la forme.

* **2.** Les composants analogiques peuvent avoir des formes rectilignes arbitraires. Chaque terminal du composant peut aussi avoir une forme rectiligne arbitraire. Cependant, les terminaux des composants ne peuvent pas être réduits à des points en layout analogique.

* **3.** La plupart des circuits analogiques sont de tailles modérées. La complexité moyenne du problème de placement pour un circuit analogique est de 20 à 30 composants. Il est important de souligner que l'efficacité de l'algorithme n'est pas aussi capitale qu'elle l'est pour des problèmes de placement de circuits numériques à haute densité.

* **4.** La taille des composants rencontrés en circuits analogiques varie en fonction de l'ordre de magnitude. Par conséquent, l'algorithme doit pouvoir s'adapter à des blocs de tailles très variées.

* **5.** Les différentes contraintes de symétrie et d'appairage fréquemment rencontrés en circuits analogiques, requièrent un contrôle approprié sur les positions des composants et leurs orientations. De plus, le fait de manipuler des formes géométriques complexes, déformables et ayant des rapports de taille importants, impose un grand nombre de contraintes géométriques sur l'ensemble du layout (ex: hauteur de bloc fixée au préalable, rapport de forme du layout prédéterminé, plots d'entrée et de sortie fixés à l'avance). Il est alors nécessaire d'utiliser une technique d'optimisation de placement flexible, qui permette de contraindre arbitrairement chaque aspect du problème de placement.

* **6.** L'objectif le plus important du placement est de garantir que la dégradation de performance induite par le layout reste dans les limites des spécifications du circuit. Par conséquent, les décisions concernant le placement doivent être basées sur une évaluation précise de la dégradation de performance, ce qui demande une connaissance détaillée des positions et orientations de tous les composants, simultanément et à tout moment. Les algorithmes constructifs et les algorithmes basés sur les partitions travaillent de façon séquentielle, à partir d'une information incomplète et sont donc mal adaptés aux problèmes de placement analogique, d'autant qu'ils n'ont pas la possibilité de manier un grand nombre de contraintes.

Si nous considérons ces différents critères pour sélectionner un algorithme de placement analogique, les algorithmes d'optimisation itérative (l'algorithme de recuit simulé, noté ici RS et

l'algorithme de placement génétique, noté PG) apparaissent comme les méthodes les plus prometteuses. Ces deux algorithmes offrent la possibilité de soumettre des contraintes arbitrairement sur la génération des nouvelles solutions de placement. Dans le cas de RS, cela peut se faire en effectuant avec précaution un dessin de l'ensemble des mouvements; dans celui de PG, en utilisant des opérateurs de crossover, de mutation ou d'inversion. Ces deux algorithmes sont basés sur une fonction de coût. La fonction de coût qui se traduit par la qualité (ajustement) des solutions de placements intermédiaires, peut être utilisée pour implémenter les contraintes de performances analogiques et les contraintes géométriques. RS et PG opèrent simultanément sur la solution entière, ce qui est essentiel pour une évaluation précise des éléments parasites. Toutes ces caractéristiques nous amène à conclure que RS et PG constituent les algorithmes les plus habilités à résoudre les problèmes de placement des circuits analogiques.

L'une des caractéristiques que nous pouvons utiliser pour différencier l'algorithme de recuit simulé du placement génétique est leur contrôle respectif sur l'interactivité. PG opère sur un groupe de solutions simultanément. Ce qui offre la possibilité à l'utilisateur d'insérer (de suggérer) des solutions candidates dans ce groupe. Même si la solution suggérée n'est pas acceptée comme la solution optimale, elle continuera à exister dans le groupe pour quelques générations et les meilleurs aspects de la solution influenceront la solution finale à travers le mécanisme du chevauchement ("crossover"). Cela fait de PG un algorithme idéal pour les problèmes de placement lorsque l'intervention de l'utilisateur est largement requise (ex: les algorithmes de floorplanning).

Notre outil de placement analogique est destiné à opérer dans un environnement de synthèse physique de dessin du layout automatique, CHIRVAN. Par conséquent, l'avantage principal de PG est secondaire pour notre problème. A cela, il faut ajouter que RS, contrairement à PG, a fait l'objet de recherches intensives. La plupart de ces recherches peuvent être réutilisées pour notre problème de placement analogique. Ceci nous permet de nous concentrer davantage sur les aspects spécifiquement analogiques du problème, tels que l'implémentation des différentes contraintes et l'évaluation de la dégradation des performances induite par le layout. Nous avons donc sélectionné RS comme étant l'algorithme d'optimisation le plus adapté à notre outil de placement. Le reste de ce chapitre se consacrera à l'explication des différents aspects de l'implémentation de RS pour le placement au niveau d'un circuit analogique.

5.5. IMPLEMENTATION DE LA METHODE DU RECUIT SIMULE POUR LE PLACEMENT DES CELLULES ANALOGIQUES

La méthode du recuit simulé, présentée par un organigramme à la figure 5.4, se déroule de la façon suivante: tout d'abord, les entités sont placées aléatoirement dans une fenêtre correspondant à la surface maximum admise par le layout, puis la température initiale T_0 est calculée. L'outil sélectionne aléatoirement un mouvement, et l'applique aux objets concernés. Le coût du nouveau placement obtenu (C_{courant}) et l'écart ($\Delta C = C_{\text{courant}} - C_{\text{précédent}}$) par rapport au coût du placement précédent sont alors calculés. Si le coût baisse ($\Delta C < 0$), le mouvement est immédiatement accepté car il a conduit à une amélioration de la qualité du placement. Au contraire, si le coût augmente ($\Delta C \geq 0$), l'ordinateur génère aléatoirement un nombre P dans $[0, 1]$ représentant la probabilité pour que le mouvement soit accepté. D'après la théorie de Boltzman, la probabilité pour que le système se trouve dans l'état actuel est égale à $\exp(-\Delta C/T)$. Notons que par construction, la température T du recuit a la même dimension que le coût C . Ainsi, si $P \leq \exp(-\Delta C/T)$, le système est effectivement dans la configuration courante, et le mouvement est accepté. A l'inverse, si $P > \exp(-\Delta C/T)$, le mouvement est refusé, et le système est replacé dans son état précédent. Ce processus est reconduit jusqu'à obtention de l'équilibre thermodynamique, équilibre pour lequel la probabilité de trouver le placement dans sa configuration optimale est assez basse. Le "gel" est atteint et le coût du placement courant a une très forte chance d'être proche d'un des minimums de la fonction d'énergie. Ce processus est décrit à la figure 5.4. Pour pouvoir appliquer l'algorithme du recuit simulé au problème du placement basé sur les performances analogiques, nous devons définir les différentes configurations du problème, le mécanisme de génération des mouvements et la fonction de coût. Définir la configuration du problème revient à trouver une représentation convenable de l'évolution du placement et de tous les objets qui ont une relation avec le problème du placement. Le paragraphe 5.5.1. se consacrera à cette discussion. Le mécanisme de génération est implémenté par un ensemble de mouvements destinés à perturber le système, ce qui sera présenté au paragraphe 5.7. Enfin, la fonction de coût qui traduit la qualité de la solution de placement fera l'objet du paragraphe 5.8.

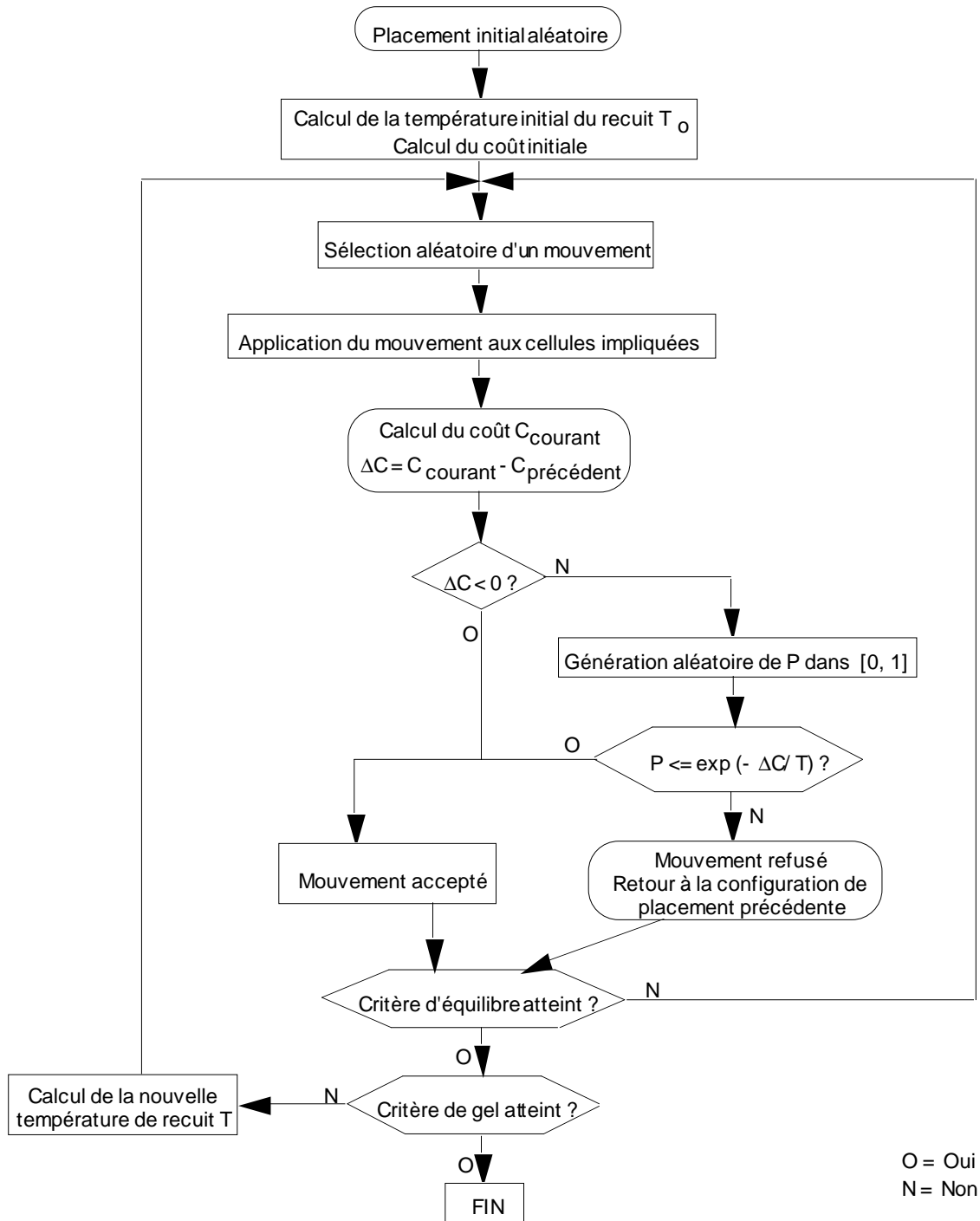


Fig. 5.4 : Algorithme du Recuit Simulé.

5.5.1. Représentation du Placement

On peut aborder de deux façons différentes la représentation des solutions de placement généré par la méthode du recuit simulé: le style Plat (“Flat style”) ou le style en Tranches

(“Slicing style”). Le choix d’un style particulier de placement a des implications sur l’ensemble des mouvements et sur la fonction de coût de l’algorithme de placement.

5.5.1.3. Représentation en Style Plat

La représentation d’un placement en style Plat, aussi appelé style de Gellat-Jepsen [82], est déterminée par les coordonnées absolues de tous les composants. Un outil de recuit manipule le placement en échangeant les coordonnées des composants. Etant donné qu’il n’y a pas de restrictions concernant les positions des composants, les superpositions sont permises facilitant ainsi le passage d’une configuration à une autre. En fin de placement, ce recouvrement illégal doit être réduit à zéro en insérant un terme de pénalité de recouvrement dans la fonction de coût.

5.5.1.2. Représentation en Style en Tranches

La structure d’un placement en style en tranches [83], est déterminée par la position relative de tous ses composants de telle façon qu’ils se respectent entre eux. Son établissement est obtenu grâce à l’utilisation d’une structure en Tranches (“Slicing structure”). Une telle représentation est obtenue en décomposant à plusieurs reprises la surface du layout en tranches verticales et/ou horizontales, comme le montre la figure 5.5(a). La surface du layout est décomposée en autant de partitions qu’il y a de composants dans le circuit, et chaque composant est assigné à une partition. Cette structure en tranches peut commodément être représentée par une structure arborescente (“Slicing tree”) (Fig. 5.5(b)). Dans cet arbre, les signes (*) et (+) sont deux opérateurs d’opérandes, symbolisant respectivement une coupure verticale et une coupure horizontale. Un outil de recuit peut améliorer les mouvements d’un objet en opérant directement sur l’arbre, par exemple en inversant deux opérandes ou opérateurs voisins.

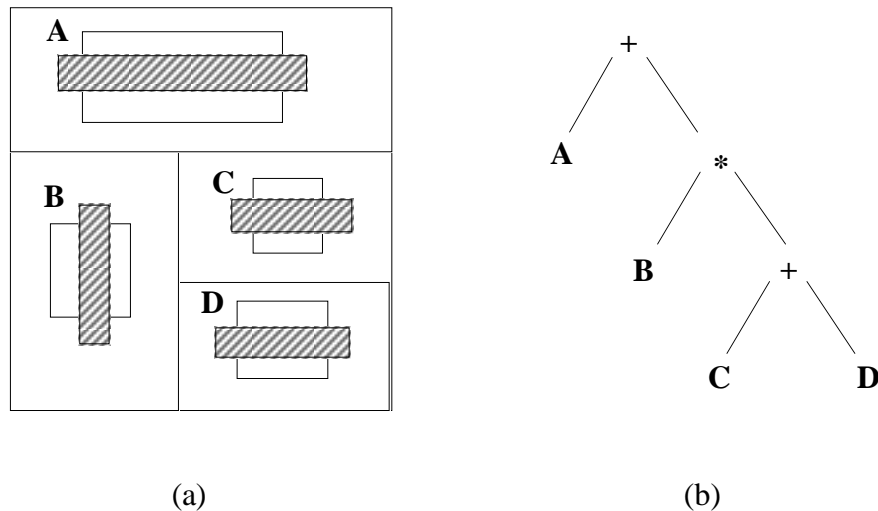


Fig. 5.5 : Représentation du placement en style en Tranches.

5.5.1.3. Discussion

Les deux types de placements dont nous venons de parler ont leurs avantages et leurs inconvénients.

Comme nous l'avons souligné au paragraphe 5.2, le placement symétrique est très important pour le layout de circuit analogique à haute performance. L'utilisation d'une représentation de placement plat permet à l'outil de recuit d'opérer directement sur les coordonnées réelles des composants. Ce qui permet d'implémenter les contraintes importantes de symétrie et d'auto-symétrie directement sur l'ensemble des mouvements comme nous le montrerons au paragraphe 5.7. Les outils de placement en style en tranches doivent implémenter les contraintes de symétrie globales dans la fonction de coût à travers l'utilisation d'axes de symétrie virtuels [84], ce qui est une solution moins efficace.

La connaissance des coordonnées réelles des composants est également nécessaire pour estimer précisément la valeur des parasites du layout et pour calculer la dégradation de performance résultante. Ces coordonnées réelles sont disponibles immédiatement lorsqu'un style de placement plat est utilisé. Si on utilise une représentation en style en tranches, la position réelle des composants doit être tracée en coordonnées réelles avant que la dégradation de performances ne puisse être calculée. Ce qui implique encore une perte de temps CPU, sans compter qu'il s'agit d'une solution moins élégante.

L'avantage principale d'un outil de placement en style en tranches devient évident quand il s'agit de l'utiliser au sein d'un système de layout numérique qui utilise le principe de routage par canaux (channel routing). Dans ce cas, la structure arborescente ("slicing tree") détermine l'ordre dans lequel positionner les canaux de routage, et évite ainsi les conflits. De plus, il ne peut pas y avoir de problèmes d'espace pour les fils de routage, puisque les canaux de routage peuvent être ajustés (compactés ou écartés) facilement pour s'adapter à la quantité requise d'espace routeur. Cependant, ces caractéristiques offrent peu d'avantages quand elles sont

utilisées dans un contexte de layout analogique. Comme nous l'avons vu au chapitre 4, le routeur de canaux est un mauvais choix pour le layout analogique. En conséquence, un routeur de surface doit être utilisé et les avantages principaux du style en tranches sur le style plat sont perdus.

Le désavantage de la représentation de placement plat est que les composants peuvent se chevaucher en solutions intermédiaires et que le placeur doit réduire ce recouvrement illégal à zéro en fin de recuit. Les placeurs en style en tranches évitent les problèmes de recouvrement, d'où une plus grande efficacité. Cependant, ce désavantage peut être tourné en avantage s'il est utilisé pour explorer les possibilités bénéfiques de fusionnement (d'aboutement) de certains composants.

Le dernier désavantage d'un placeur en style en tranches est qu'il restreint l'ensemble des topologies du layout accessibles. L'utilisation d'une représentation de placement plat permet d'explorer toutes les configurations de placement, et non seulement celles représentées par des structures en tranches, ce qui induit un layout plus dense, spécialement pour les placements de composants de tailles très variées.

Tous ces facteurs combinés nous permettent de désigner les représentations de placement plat comme le type de structure le plus adapté au placement analogique.

5.6. MANIPULER LES CONTRAINTES ANALOGIQUES EN RECUIT SIMULE

Jusqu'à maintenant, nous avons parlé des éléments fondamentaux du placement par recuit simulé. Au point où nous en sommes, une question importante se pose: comment et où implémenter les différentes contraintes analogiques identifiées au paragraphe 5.2 ? Ces contraintes ont pour effet de créer un sous-ensemble illégal à l'intérieur de l'ensemble complet des possibilités de placement. Par exemple, un placement où la dégradation de performances induite par l'effet combiné de tous les parasites d'interconnexion excède une ou plusieurs spécifications de performances, viole une contrainte de performance et constitue donc une solution illégale. Un autre exemple consiste en un placement où deux composants appairés ont des orientations différentes. On peut éviter ces solutions illégales par deux approches différentes.

La première approche consiste à dessiner l'ensemble des mouvements de telle façon qu'aucun placement inacceptable ne puisse être obtenu. Dans le cas de deux transistors appairés par exemple, il s'agit de les orienter selon la même direction dans la solution initiale et de déplacer ces orientations simultanément, de telle façon que les orientations soient unidirectionnelles à tout moment durant le processus de recuit et par conséquent aussi dans la solution finale. L'avantage de cette démarche est qu'il n'y a pas de temps CPU perdu à évaluer

le placement et considéré comme inacceptable. De plus, nous pouvons garantir que ces contraintes soient obtenues par construction. La difficulté de cette première approche réside dans le fait que l'ensemble des mouvements est plus compliqué.

La seconde approche consiste à imposer une pénalité sur les violations de contraintes dans la fonction de coût. Pour que la contrainte soit atteinte, ce terme de pénalité doit être réduit à zéro par le mécanisme de recuit. Dans le cas de deux transistors appairés, si les orientations des transistors ont la même direction, le terme de pénalité est égal à zéro, sinon il est différent de zéro. Une conséquence importante de cette approche est qu'elle ne garantit plus que la contrainte soit obtenue par construction. Le mécanisme de recuit simulé tente de minimiser la valeur totale de la fonction de coût, ce qui ne signifie pas nécessairement que chaque terme individuel sera réduit à zéro. Implémenter une contrainte dans la fonction de coût implique que cette contrainte soit échangée contre d'autres contraintes compétitives et que ces contraintes ne soient réduites à zéro seulement si le résultat final est alors le meilleur qui puisse être obtenu.

La conclusion qui ressort de la discussion précédente est que les contraintes obligatoires, c'est à dire les contraintes qui doivent absolument être obtenues, sont implémentées de façon plus satisfaisantes si elles sont considérées comme des restrictions de l'ensemble des mouvements. Malheureusement, certaines contraintes obligatoires sont difficiles à maintenir par construction. Si c'est le cas, elles doivent être implémentées comme termes de pénalité dans la fonction de coût et il faudra veiller tout particulièrement à ce que ces contraintes soient réduites à zéro dans le résultat final (par exemple en leur donnant un poids important). Les différentes contraintes seront implémentées de la façon suivante dans l'outil de placement de Chirvan:

* Contraintes de Symétrie

La symétrie est considérée comme une contrainte obligatoire. Si l'utilisateur désigne un certain nombre de composants comme étant symétriques et/ou auto - symétriques, les composants doivent être symétriques dans le placement résultant. Par conséquent, les contraintes de symétrie sont manipulées comme des restrictions dans l'ensemble des mouvements. Les groupes de composants symétriques sont déplacés simultanément de telle façon que leur symétrie soit préservée à tout moment durant l'optimisation ainsi que dans le résultat final.

* Contraintes d'Appairage

Si l'utilisateur désigne un groupe de composants comme étant un groupe d'appairage, cela aura deux conséquences: premièrement, les orientations et formes seront identiques dans le résultat final; deuxièmement, la distance entre les composants sera optimisée en vue de son influence sur les performances du circuit. La première spécification est implémentée comme une contrainte d'orientation identique/ forme identique sur l'ensemble des mouvements. La seconde est obtenue en incluant la distance entre les composants dans le groupe des effets parasites pour

lesquels la dégradation des caractéristiques de performances est calculée et incluse dans la fonction de coût du placement.

De cette façon, l'utilisateur peut désigner une paire de composants comme étant appairés sans spécifier le degré d'appairage. Les composants appairés sont toujours générés avec des formes et des orientations identiques mais c'est à l'outil de placement de déterminer les positions et donc les distances entre les composants appairés de telle façon que les contraintes de performances soient obtenues. Etant donné qu'il n'est pas toujours possible dans un layout de circuit analogique d'obtenir en même temps toutes les spécifications de symétrie, de disposer tous les composants appairés directement les uns à côté des autres et d'obtenir un layout compact acceptable, le degré d'appairage est sélectionné en vue de son influence sur la performance du circuit.

* Contraintes de Performances

La spécification la plus importante d'un outil de placement basé sur les performances analogiques, est d'assurer que la dégradation de performances induite par les effets des différents parasites du layout soit contenue à l'intérieur des spécifications du circuit. Cependant, les contraintes de performances ne peuvent pas être simplement traduites en restrictions sur les coordonnées et/ou sur les orientations des composants. Elles doivent être évaluées à partir d'une solution de placement intermédiaire. Par conséquent, il est impossible de maintenir les contraintes de performances par construction, et ces contraintes doivent être implémentées par des termes de pénalité dans la fonction de coût. Une attention particulière doit être portée à garantir que les violations de contraintes de performances soient réduites à zéro dans le résultat final. Ce que nous développerons au paragraphe 5.8.

* Contraintes Géométriques

Les contraintes géométriques peuvent être considérées comme contraintes obligatoires ou comme contraintes facultatives. C'est à l'utilisateur de le décider. Par exemple, si la hauteur d'un placement doit être plus petite qu'une certaine valeur pour pouvoir être insérer à l'intérieur d'un placement au niveau du système, la contrainte de hauteur minimum est une contrainte obligatoire et elle est implémentée dans l'ensemble des mouvements. Une autre situation où les contraintes géométriques obligatoires s'imposent, concerne le placement résultant quand il est utilisé dans un système d'assemblage de cellules de layout standards: la hauteur et les positions des terminaux des tensions d'alimentation sont fixées, et donc obligées. Dans le cas où les contraintes géométriques sont spécifiées comme des cibles d'optimisation, elles sont implémentées dans la fonction de coût. Un exemple de ce type est le rapport de forme ciblé qui peut être déterminé pour un placement.

* Contrainte de Superposition

La contrainte de superposition est une contrainte obligatoire: si le placement final contient des superpositions illégales, il est rejeté. Cependant, cette contrainte n'est pas implémentée comme une restriction dans l'ensemble des mouvements. Implémenter des contraintes de recouvrement dans l'ensemble des mouvements impliquerait que les placements comportant des modules de recouvrement ne soient jamais considérés. Une telle restriction rendrait impossible la détection de situations où le recouvrement est bénéfique, pour des raisons de densité ou de performances. Par conséquent, chaque situation de recouvrement doit être considérée individuellement et le recouvrement se gère mieux dans la fonction de coût.

Il apparaît donc, après examen des différentes stratégies exposées ci-dessus, que la fonction de coût du recuit contient des termes de pénalité qui doivent être réduits à zéro pour obtenir un résultat utilisable. Pour s'assurer que ces termes de pénalité soient effectivement réduits à zéro, nous devons utiliser des techniques spéciales, comme nous l'expliquerons au paragraphe 5.8.

5.7. FONCTION DE COUT

La recherche d'un placement optimum est conduite par la fonction de coût de l'algorithme du recuit simulé. La fonction de coût est conçue pour minimiser la superficie du placement final, pour réduire le recouvrement illégal entre les composants à zéro et pour faire respecter le rapport de forme et les contraintes de performances. La fonction de coût C est calculée pour chaque résultat de placement intermédiaire et résulte de la somme de quatre termes:

$$C = \alpha C_{Surface} + \beta C_{Rapport\ de\ Forme} + \gamma C_{Re\ couvrem\ ent} + \delta C_{Deg.\ de\ Perf} \quad (5.4)$$

où :

- $C_{Surface}$ est le Coût de Surface:

Ce terme est conçu pour minimiser la surface du layout ; il est égal à la surface du rectangle englobant ("bounding box") du placement intermédiaire. Ce terme est inclut dans la fonction de coût pour éviter un placement trop large.

- $C_{Rapport\ de\ Forme}$ est le Coût de Rapport de Forme:

Ce terme est utilisé pour amener le rapport de forme du placement final à la valeur spécifiée par l'utilisateur. Sa valeur est donnée par l'écart entre le rapport de forme du placement intermédiaire et le rapport de forme spécifié par l'utilisateur:

$$C_{\text{Rapport de Forme}} = |\text{Rapport de Forme} - \text{Rapport de Forme Désiré}| \quad (5.5)$$

- $C_{\text{Recouvrement}}$ est le **Coût de Recouvrement**:

Etant donné que nous utilisons une représentation de placement plat (“flat placement”), les composants sont autorisés à se superposer au cours de l’algorithme de recuit simulé. Le terme de recouvrement dans la fonction de coût est utilisé pour réduire à zéro le recouvrement des composants entre eux dans le résultat final. Le terme est obtenu en additionnant la totalité des recouvrements illégaux présents dans le placement intermédiaire:

$$C_{\text{recouvrement}} = \sum_{i=1}^n \sum_{j=i+1}^n (\text{surface de recouvrement})_{ij} \quad (5.6)$$

où n est le nombre de composants dans le circuit, et la surface de recouvrement est la surface de recouvrement entre les composants i et j .

- $C_{\text{Deg de Perf}}$ est le **Coût de Performances**:

Ce terme est utilisé pour garder la dégradation de performance induite par les effets des différents parasites du layout à l’intérieur des limites spécifiées par l’utilisateur. Sa valeur est nulle si toutes les caractéristiques de performances sont dans les marges des spécifications, et proportionnelle au montant des violations si elles ne le sont pas. Le calcul de ce terme de coût de performances est important pour l’approche du placement basé sur les performances. Il serait trop long de présenter ici les différentes méthodes de calcul de ce terme. Le lecteur pourra se référer aux travaux excellents effectués dans [85][86][87][88].

Les coefficients de pondération α , β , γ et δ , sont utilisés pour ajuster dynamiquement l’importance de chaque terme pendant la phase d’optimisation du placement. Au début de l’étape d’optimisation, quand la configuration générale du placement est déterminée, les termes du rapport de forme et de performances dominent la fonction coût. Vers la fin de l’optimisation, quand les positions finales des composants sont optimisées sans changement majeur de la configuration, le poids du terme de recouvrement doit augmenter pour assurer qu’aucun recouvrement illégal n’est présent dans la solution finale. Pour accomplir cette optimisation, les coefficients de pondération varient entre une valeur minimum et une valeur maximum. Après chaque boucle, les poids relatifs des termes de rapport de forme et de performances diminuent linéairement de leur valeur maximum vers leur valeur minimum, tandis que le poids du terme de recouvrement augmente de sa valeur minimum vers sa valeur maximum.

5.8. CONCLUSION

L'étape de placement est très importante pour le layout de circuit analogique. Tous les parasites du layout qui dégradent la performance d'un circuit (les parasites d'interconnexion, les composants appairés et les effets thermiques...) sont, soit fixés, soit largement influencés par le placement du circuit. Pour garder la dégradation de performances induite par le layout à l'intérieur des marges fixées par l'utilisateur, un algorithme de placement automatique doit prendre en compte simultanément tous ces effets. Par ailleurs, un certain nombre de contraintes topologiques, comme la contrainte de symétrie ou les contraintes du rapport de cadrage (aspect ratio), doivent être mises en vigueur. L'intérêt de ce chapitre est d'avoir présenté un algorithme de placement basé sur les performances qui prend en compte tous les effets parasites du layout.

Après avoir énuméré les aspects caractéristiques des différents algorithmes de placement possibles, nous avons explicité les raisons de notre choix de l'algorithme de recuit simulé comme le meilleur algorithme d'optimisation de placement automatique. Puis, nous avons montré comment l'algorithme de recuit simulé prend en compte toutes les contraintes analogiques nécessaires pour la phase de placement, tout en gardant la dégradation de performance induite par le layout dans les limites des spécifications.

Cet algorithme n'a pas pu être implémenté expérimentalement dans Chirvan pour des raisons de discontinuité de programme de recherche Chirvan par France Telecom et puis de mon départ dans l'industrie chez Motorola puis Rockwell. Néanmoins nos recherches dans ce domaine ne se sont pas arrêtées là; elle ont trouvé une prolongation au sein d'une équipe de recherche à Rockwell Semiconductor system aux Etats-Unis. Les résultats de ces recherches et l'implémentation de cet algorithme de placement ont été très encourageant même s'ils restent à l'état de développement (non encore utilisés par les concepteurs).

Conclusion Générale

CONCLUSION GENERALE

La place des circuits intégrés analogiques à hautes performances a pris une importance croissante ces dernières années dans la conception des systèmes mixtes analogiques/numériques.

La réalisation de tels circuits souffre de difficultés principalement dues à la réduction de la tension d'alimentation et à la réduction de la consommation, qui sont conduites par la prolifération des systèmes portables alimentés par des batteries, mais pâtit aussi du manque d'outils de C.A.O permettant d'automatiser la phase de layout qui est assez laborieuse et prend beaucoup de temps.

Nous avons donc orienté nos recherches vers ces deux aspects du problème de la conception des circuits intégrés analogiques.

Un des blocs analogiques les plus importants est l'amplificateur opérationnel. Il a trouvé son chemin dans de nombreuses applications, comme les filtres à capacité commuté ("switched capacitor filter"), amplificateur de signal, filtres, amplificateurs de charges, buffer d'entrée ou de sortie ("input or output buffer"), et dans beaucoup d'autres applications. Dans le but de garder une compétitivité comparable à celle de la conception des circuits numériques, la réalisation d'amplificateur opérationnel à haute performance, compact, à basse tension d'alimentation et à faible consommation, est un des très grands défis à relever aujourd'hui, dans le domaine de la conception des circuits intégrés analogiques.

La réduction progressive des tensions d'alimentation, liée aux nouvelles technologies (aujourd'hui de 5 à 3 volts, demain 1 volt), risque de révolutionner la conception des circuits intégrés analogiques. La norme $V_{dd}=5V$, qui existe depuis plus de vingt ans, sera progressivement remplacée. Dans ce nouveau contexte, un certain nombre de schémas (structures) traditionnels ne sont plus utilisables pour une tension d'alimentation inférieure ou égale à 3V. Les recherches s'orientent donc vers la conception de nouvelles structures plus adaptées aux faibles tensions d'alimentation et qui s'insèrent à l'intérieur d'un schéma classique d'amplificateur opérationnel. La première partie de notre thèse s'est orientée vers l'étude et l'application d'une nouvelle structure CMOS adaptée aux faibles tensions d'alimentation (i.e., le circuit composite) qui a donné des résultats très prometteurs pour $V_{dd}=3V$. En introduisant plusieurs circuits composites dans un schéma traditionnel, nous avons fortement augmenté le gain intrinsèque du montage initial. Ceci a permis, aussi, de satisfaire la dynamique de sortie et les cibles transitoires du cahier des charges, qui constituaient une limitation pour des tensions d'alimentation faibles.

Dans la deuxième partie de cette thèse nous nous sommes intéressés au problème d'automatisation du flot de conception du layout des circuits intégrés analogiques et mixtes en tenant compte de leur particularité. Sachant que les performances des circuits intégrés

analogiques dépendent fondamentalement des phénomènes parasites générés par leur layout, la manière de dessiner les masques de ces circuits devrait respecter les contraintes analogiques que nous avons étudiées et listées en détail dans le chapitre 3. Nous avons aussi expliqué comment intégrer ces différentes contraintes dans tous les outils d'automatisation du dessin des masques (Générateur, Placeur, Routeur, Compacteur).

Nous avons présenté dans le chapitre 4 les travaux qui ont été réalisés par notre équipe au CNET Grenoble, pour l'automatisation de ce flot: la génération automatique du layout des composants, le routage automatique de ces composants entre eux grâce au routeur de labyrinthe qui intègre toutes les contraintes analogiques, le routage des cellules entre elles grâce au routeur de canal (routage "top level"), et enfin le compacteur qui est nécessaire pour le routeur de labyrinthe et qui lui aussi respecte les contraintes analogiques. Nous avons expliqué comment ces différents modules intègrent les contraintes analogiques.

Enfin nous avons tenté de résoudre le problème du placement des circuits intégrés analogiques, qui est l'une des étapes les plus difficiles à automatiser puisque c'est pendant cette phase que la majorité des parasites sont définis. Le placement des composants influence largement les valeurs des interconnexions parasites, même si leurs valeurs finales sont déterminées durant la phase de routage, car leurs valeurs minimales effectives sont fixées par la configuration du composant, elle-même déterminée durant la phase de placement. Cette étape nécessite encore l'expertise des concepteurs. Nous avons étudié les différents algorithmes de placement possibles, et nous avons choisi l'algorithme de recuit simulé comme l'algorithme le plus fiable pour l'optimisation du placement des cellules analogiques. Nous avons montré comment il intègre les contraintes analogiques. Cet algorithme n'a pu être implémenté expérimentalement dans Chirvan pour les raisons que nous avons mentionnées au chapitre 5. Néanmoins nos efforts dans ce domaine ne se sont pas arrêtés là. Nous avons continué ces recherches au sein d'une équipe de Rockwell aux Etats-Unis. Les résultats de nos recherches et l'implémentation de cet algorithme de placement ont été très encourageants même s'ils restent à l'état de développement (non encore utilisés par les concepteurs).

L'aboutissement de cette thèse est, premièrement, la création d'une nouvelle méthode de conception des circuits intégrés analogiques, plus précisément la génération d'une technique de conception de nouvelle structure, plus adaptée aux basses tensions d'alimentation et aux faibles consommations, deuxièmement, notre contribution à l'automatisation de la phase du layout des circuits intégrés analogiques, à savoir l'étude détaillée des contraintes analogiques à prendre en compte dans tout outil d'automatisation du layout (générateur, placeur, routeur, compacteur), ainsi que notre participation au développement de Chirvan en aidant à sa mise au point, en l'utilisant, en proposant des améliorations, et surtout en consacrant tous nos efforts à l'étape de placement.

Références

REFERENCES

- [1] A. Matsuzawa, "Low-Voltage and Low-Power Circuit Design for Mixed Analog/Digital Systems in Portable Equipment", *IEEE Journal Solid-State Circuits*, vol. SC-29, April 1994, pp. 470-480.
- [2] A.L. COBAN, P.E. ALLEN, X.Shi, "Low-Voltage analog IC design in CMOS technology" *IEEE Trans. Circuits and Systems*, Vol. 42, pp. 955-958. Nov. 1995
- [3] K. Shimohigashi, K. Seki, "Low-Voltage ULSI Design", *IEEE Journal Solid-State Circuits*, vol. SC-28, April 1993, pp. 408-412.
- [4] M. Nagata, "Limitations, Innovations, Challenges of Circuits and Devices into a Half Micrometer and Beyond", *IEEE Journal of Solid-State Circuits*, vol. SC-27, April 1992, pp. 465-472.
- [5] A. P. Chandrakasan, S. Sheng and R. W. Brodersen, "Low-power CMOS digital design", *IEEE J. of Solid-State Circuits*, vol. SC-27, No.4 April 1992, pp. 473-484.
- [6] D. Liu and C. Svensson, "Trading speed for low power by choice of supply and threshold voltages", *IEEE Journal of Solid-State Circuits*, vol. SC-28, No.1 Jan. 1993,
- [7] D. A. Patterson and C. H. Sequin, "A VLSI RISC", *IEEE Computer*, pp. 8-21. September 1982.
- [8] C. Piguet, "Binary-decision and RISC-like machine for semicustom design", *Microprocessor and Microsystems*, Vol. 14, No 4, pp. 231-240, May 1990.
- [9] J-M. Masgonty, "Branch-based digital cell libraries", *Proc. of EURO-ASIC* May 1991.
- [10] H. J. 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, No.5 pp. 468-473, August 1984.
- [11] E.A. Vittoz, "Low-power Low-Voltage Limitations and Prospects in Analog Design", in *Analog Circuit Design* edited by R.J. v.d. Plassche, W. Sansen, J.H. Huijsing, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995, p.p. 3-16.
- [12] Enz, C.C.; Vittoz E.A., "CMOS Low-power analog circuit design", Edited by R. Cavin and W. Liu, *IEEE Service Center*, Piscataway, NJ, 1996, Chapter 1.2, pp. 79-133,
- [13] E.A. Vittoz, "Low-power Design: Ways to approach the limits" *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 4-18, May 1994.
- [14] R. Castello and P.R. Gray, "Performance limitation in switched-capacitor filters" *IEEE Trans. Circuits, Syst.*, vol. CAS-32, pp. 865-876, Sept. 1985.

-
- [15] G. Groenewold, "Optimal dynamic range integrators" IEEE Trans. Circuits, Syst., vol.39, pp. 614-627, Aug. 1992.
- [16] Ron Hogervorst & Johan Huijsing, "Design of LowVoltage Low-power CMOS Operational Amplifier Cells", Kluwer Academic Publishers, 1996.
- [17] Kenneth R. LAKER, Willy M.C. SANSEN, "Design of Analog Integrated Circuits and Systems". Mac Graw Hill, 1994.
- [18] K. Bult, "Analog CMOS square-law circuits" Ph.D. dissertation, University of Twente, Enschede, The Netherlands, 1988.
- [19] R.J. Wiegerink, "Analysis and Synthesis of MOS Translinear Circuits", Ph.D. dissertation, University of Twente, Enschede, The Netherlands, 1992.
- [20] E.A. Vittoz, "Future of Analog in VLSI Environment," Proc. IEEE Int. Symp. Circuits Syst., pp. 1372-1375, May 1990.
- [21] E.A. Vittoz, "The Design of High-Performance Analog Circuits on Digital CMOS Chips," IEEE Journal of Solid-State Circuits, vol. SC-20, pp. 657-665, June 1985.
- [22] E.A. Vittoz and O. Neyroud, "A low-voltage CMOS bandgap reference," IEEE Journal of Solid-State Circuits, vol. SC-14, pp. 573-577, June 1979.
- [23] E.A. Vittoz and J. Fellrath, "A MOS analog integrated circuits based on weak inversion operation," IEEE Journal of Solid-State Circuits, vol. SC-12, pp. 224-231, June 1977.
- [24] C. C. Enz, "Low-power log-domain continuous-time filters: an introduction," Proc. Low-Power-Low-Voltage Workshop of the European Solid-State Circuits Conf. (ESSCIRC'95), Sept. 1995.
- [25] C. Mead, L. Conway, "Introduction to VLSI Systems", Addison-Wesley Publishing Company, USA, 1980.
- [26] A.P. Chandrakasan, S. Sheng and R.W. Brodersen, "A Low-Power Circuit Chipset for a Portable Multimedia I/O Terminal", IEEE Journal Solid-State Circuits, vol. SC-29, December 1994, pp. 1415-1428.
- [27] D. MORCHE. "Conception de codeurs sigma-delta en technologie CMOS pour la conversion analogique numérique haute résolution". Thèse: Institut National Polytechnique de Grenoble, 1994, 220 pages.
- [28] Behzad RAZAVI "Principles of Data Conversion System Design". IEEE Press. 1995, 256 pages.
- [29] M. ISMAIL, T. FIEZ. "ANALOG VLSI signal and information processing". Mac Graw Hill, 1994, pp. 25-27 .
- [30] B. J. Hosticka. "Improvement of the gain of MOS amplifiers". IEEE J. Solid-State Circuits, Vol. SC-14, No. 6, Décembre 1979, pp. 1111-1114.
- [31] E. SACKINGER, W. GUGGENBUHL. "A high-swing, high-impedance MOS cascode circuit". IEEE J. Solid-State Circuits, Vol. 25, No. 1, Février 1990, pp.289-298.

-
- [32] H.C. YANG, D.J. ALLSTOT. "An active-feedback cascode current source". IEEE Trans. on Circuits and Systems, Vol.37, No.5 Mai 1990, pp. 644-646.
- [33] P.R. GRAY, R.G. MEYER. "Analysis and Design of Analog Integrated Circuits". John Wiley & Sons, 1984, pp. 709-718.
- [34] F. CHAAHOUB, C. CORBEX, L. O. DONZELLE, A. GERODOLLE, and B. HENNION. "New CMOS Op Amp Design Dedicated For Low Supply Voltage". Proc. IEEE ESSCIRC'95, Lille, September 1995, pp. 298-301.
- [35] A.L. COBAN, P.E. ALLEN. "A 1.75V rail-to-rail CMOS op amp". Proc. IEEE ISCAS'94, Vol. 5, 1994, pp. 497-500.
- [36] M. ISMAIL, T. FIEZ. "ANALOG VLSI Signal and Information Processing". Mac Graw Hill, 1994, pp. 87-112.
- [37] J. HUIJSING, R. VAN DER PLASSCHE, W. SANSEN. "ANALOG CIRCUIT DESIGN operational amplifiers, analog to digital convertors, analog computer aided design". Kluwer academic publishers, 1993, 460 pages.
- [38] K. BULT, G.J.G.M. GEELLEN. "A Fast-settling CMOS op amp for SC circuits with 90-db DC Gain". IEEE J. Solid-State Circuits, Vol. 25, No 6. Décembre 1990, pp. 1379-1384.
- [39] D. MORCHE, F. BALESTRO, P. SENN. "Convertisseurs analogique-numérique CMOS à haute résolution pour les circuits VLSI audio". L'écho des RECHERCHES, No. 153, 3ème trimestre 1993, pp. 35-50.
- [40] B. HENNION, J. LECOURVOISIER. "OPART, un logiciel d' optimisation de circuits électroniques". L'écho des RECHERCHES, No. 149, 3ème trimestre 1992, pp. 56-59
- [41] J. Rijmenants, J. B. Litisios, T. R. Schwarz and M. G. R. Degrauwe, "ILAC: Automated Layout tool for Analog CMOS Circuits". IEEE Journal of Solid-State Circuits, vol. 24, n° 24, pp. 417-425, April 1989.
- [42] M. Kayal, S. Piguet, M. Declercq and B. Hochet, "SALIM: A Layout Generation Tool for Analog ICs", IEEE Journal of Solid-State Circuits, pp. 751-754, May 1988
- [43] J. M. Cohn, D. J. Garrod, R. A. Rutenbar, L. R. Carley, "KOAN/ANAGRAM II: New Tools for device-level Analog Placement and Routing", IEEE Journal of Solid-State Circuits, March 1991.
- [44] H. Mathias, L. Hébrard, J. Berger-Toussan, G. Jacquemod, F. Gaffiot and M. Le Helley, "Automatic Layout Generation for CMOS Analog Transistors", IEEE Custom Integrated Circuits Conference, vol. 26, n° 3, pp. 330-342, March 1994.
- [45] U. Choudhury and A. Sangiovanni-Vincentelli, "Constraint-based channel routing for analog and mixed analog/digital circuits", IEEE Transactions on computer-aided design of integrated circuits and systems, vol. 12, n° 4, pp. 497-510, April 1993.

-
- [46] L.O. Donzelle and P. F. Dubois, "A new approach to layout of custom analog cells", In Proc. EDAC, Paris, pp.480-483, February 1991.
- [47] L.O. Donzelle, P. F. Dubois, B. Hennion, J. Parissis, P. Senn, "A constraint based approach to automatic design of analog cell", In Proc. 28th DAC, San Francisco, pp.506-509, June 1991.
- [48] J. M. Cohn, D. J. Garrod, R. A. Rutenbar, L. R. Carley, "Analog device-level Layout Automation". New York: Kluwer Academic, 1994.
- [49] R. R. Trautman, "Latchup in CMOS technology". New York: Kluwer Academic, 1986.
- [50] T. Ohzone and H. Iwata, "Transient latchup characteristics in n-Well CMOS" IEEE Transactions on Electron Devices, vol. 39, n° 8, pp. 1870-1875, August 1992.
- [51] R. Kling and P. Banerjee, "A new standard cell placement package using simulated evolution". In Proc. 24th ACM/IEEE Design Automation Conference, 1985.
- [52] A. Grebne. "Bipolar and MOS integrated circuit design" Wiley Inter-Science, 1984.
- [53] J. M. Cohn, D. J. Garrod, R. A. Rutenbar, L. R. Carley, "Techniques for Simultaneous Placement and Routing of Custom Analog Cells in Koan/ANAGRAM II". In Proc. IEEE International Conf. on CAD, November 1991
- [54] T. Ohtsuki, "Layout Design and Verification" Advances in CAD
- [55] B. HENNION, P. SENN. "A new Algorithm for Third Generation Circuit Simulators : the One-Step Relaxation Method". Proc. 22nd IEEE DAC, Las Vegas, June 1985, pp. 137-143.
- [56] B. HENNION, P. SENN. "Simulation électrique de circuits MOS à grande densité d'intégration, exemple du simulateur électrique ELDO". L'écho des RECHERCHES, No. 140, 4ème trimestre 1987, pp. 47-58.
- [57] B. HENNION, D. CLAVELIER. "Analog IC design using a new optimizer, OPART". Proc. 3rd PATMOS'93, La Grande Motte, Octobre 1993, pp. 213-222.
- [58] ILOG, "Aida version 1.65 : Manuel de Référence", 1991.
- [59] N. Sherwani, "Algorithms for VLSI Physical Design Automation" Kluwer Academic Pub., 1993.
- [60] E. Malavasi and A. Sangiovanni-Vincentelli, "Area Routing for Analog Layout", IEEE Transactions on computer-Aided Design Vol. 12, No. 8, August 1993.
- [61] W. Heyns, W. Sansen, H. Beke, "A Line-expansion Algorithm for the General Routing Problem with a guaranteed Solution". ACM, pp.243-249, 1980.
- [62] A. Margarino, A. Romano, A. de gloria, F. Curatelli, P. Antognetti, "A Tile-Expansion Router". IEEE Transactions on computer-Aided Design Vol. Cas-6, No. 1, January 1987.

-
- [63] J. M. Cohn, D. J. Garrod, R. A. Rutenbar, L. R. Carley, "KOAN/ANAGRAM II: New tools for device-level analog placement and routing", IEEE Custom Integrated Circuits Conference, vol. 26, n° 3, pp. 330-342, May 1990.
- [64] Chia-Chun Tsai, Sao-Jie Chen, Zu-Shiung Feng, "An H-V Alternating Router". IEEE Transactions on computer-Aided Design Vol. 11, No. 8, August 1992.
- [65] J.K. Outsterhout, "Corner Stitching: A data-structuring Technique for VLSI Layout Tools". IEEE Transactions on computer-Aided Design Vol. Cas-3, No 1, January 1984.
- [66] M. A. Shand, "Algorithms for Corner Stitched Data-Structures". *Algorithmica* (1987) 2: 61-80.
- [67] C. Y. Lee, "An Algorithm for Path connections and its application". IRE Transactions on Electronic Computers, 1961.
- [68] E. W. Dijkstra, "A Note on Two Problems in connection with Graphs". *Numerische Mathematik*, Vol. 1, pp. 269-271, 1959.
- [69] N. Nilsson, "Problem-Solving Methods in Artificial Intelligence". McGraw-Hill, 1971.
- [70] E. Felt, E. Charabon, M. E. Malavasi, A. Sangiovanni-Vincentelli, "an efficient methodology for symbolic compaction of analog IC's with multiple symmetry constraints". In Proc. 1986 IEEE International Conference on CAD, Nov. 1992.
- [71] M. Minoux, "Programmation mathématique". Collection technique et scientifique des télécommunications, Ed. Dunod, 1983.
- [72] M. Kawakita, T. Watanabe, "Analog Layout Compaction with a clean-up Function". The Transactions of the IEICE, Vol. E 71, No 12, pp 1243-1252, December 1988.
- [73] F. CHAAHOUB, A. GERODOLLE, L. O. DONZELLE, C. CORBEX, "Coherent Behavior of a Router and Compactor in Analog Circuit Design". Proc. IEEE DATE'98, Paris, Fevrier 1998, Designer Track, pp. 257-264.
- [74] S. Sahni, A. Bhatt, "The Complexity of Design Automation Problems" Proc. IEEE/ACM Design Automation Conf., pp. 402-411, June 1980.
- [75] K. Shahookar and P Mazumder, "VLSI Cell Placement Techniques", ACM Computing Surveys, Vol. 23, No. 2, June 1991.
- [76] N. Sherwani, Algorithms for VLSI Physical Design Automation, second edition, chapter 3, Kluwer Academic Publishers, Boston, Dordrecht, London, 1995.
- [77] M. Hanan, J. Kurtzberg, Design Automation of Digital Systems. M. Breuer, Ed., Prentice Hall, Englewood Cliffs, N. J., Chapter 5, pp. 213-282. 1972.
- [78] B. Kernighan, S. Lin, "An Efficient Heuristic Procedure for partitioning Graphs", Bell Systems Technical Journal, Vol. 49, No.2, pp. 291-308, 1970.

-
- [79] C. Fiduccia, R. Mattheyses, "A Linear-Time Heuristic for Improving Network Partitions", Proc. IEEE/ACM Design Automation Conference, pp. 175-181, June 1982.
- [80] P. van Laarhoven, E. Aarts, Simulated Annealing: Theory and Applications, Kluwer Academic Publishers, 1987.
- [81] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, E. Teller, "Equation of State Calculation by Fast Computing Machines", Journal of Chem. Physics, Vol. 21, pp. 1087-1092, 1953.
- [82] D. W. Jepsen and C. D. Gelatt Jr, "Macro placement by Monte Carlo Annealing", Proc. IEEE int. Conf. on Computer Design, pp. 495-498, Nov. 1984.
- [83] D. Wong, C. Liu, "A new algorithm for floorplan design", Proc. IEEE/ACM Design Automation Conf., 1986.
- [84] E. Malavasi, E. Charbon, G. Jusuf, R. Totaro, A. Sangiovanni-Vincentelli, "Virtual Symmetry Axes for the Layout of Analog IC's Proc. ICVC, pp. 195-198, October 1991.
- [85] K. Lampaert, G. Gielen, W. Sansen, "Performance-Driven Placement of Analog Circuits". Proc. IEEE European Solid-State Circuits Conf., pp. 156-159, September 1994.
- [86] K. Lampaert, G. Gielen, W. Sansen, "Direct Performance-Driven Placement of Mismatch-Sensitive Analog Circuits". Proc. ACM/IEEE Design Automation Conf., pp. 445-449, June 1995.
- [87] K. Lampaert, G. Gielen, W. Sansen, "A Performance-Driven Placement Tool for Analog Integrated Circuits". IEEE Journal of Solid-State Circuits, vol. SC-30, n° 7, pp. 773-780, July 1995.
- [88] K. Lampaert, G. Gielen, W. Sansen, "Thermally Constrained Placement of Analog and Smart Power Integrated Circuits". Proc. IEEE European Solid-State Circuits Conf., Sep. 1996.
- [89] A. Papoulis, Probability, Random Variables and Stochastic Processes (third edition), McGraw-Hill International Editions, 1991.
- [90] R. Otten, L. van Ginneken, "Floorplan Design using Simulated Annealing", Proc. IEEE Intl. Conf. on Computer-Aided Design, pp. 96-98, November 1984.
- [91] F. Catthoor, H. de Man, J. Vandewalle, "SAMURAI: a general and efficient simulated-annealing schedule with fully adaptive annealing parameters", Integration, the VLSI journal, Vol. 6, pp. 147-178, 1988.
- [92] M. Kayal, S. Piguet, M. Declercq and B. Hochet : SALIM : A Layout Generation Tool For CMOS Analog ICs", Proc. CICC, Rochester, May 1988, pp. 7.5.1-7.5.4
- .

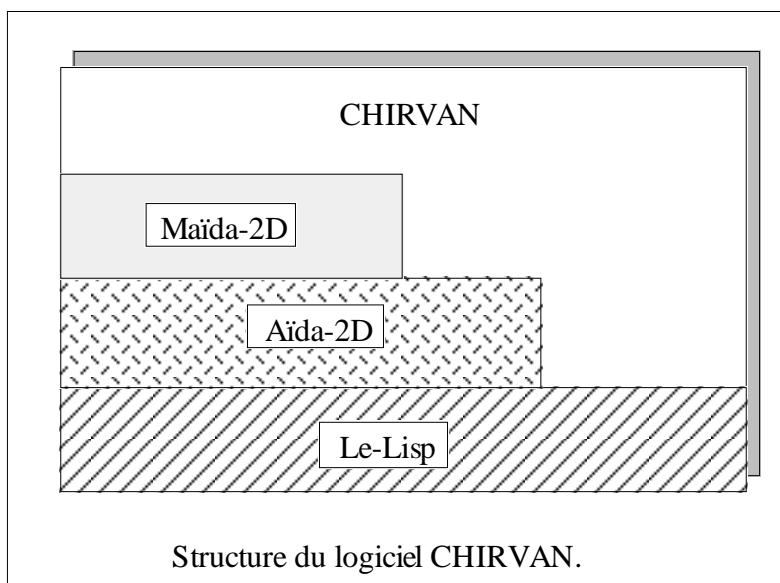
-
- [93] M. Kayal, S. Piguet, M. Declercq and B. Hochet : “An Interactive Layout Generation Tool For CMOS Analog ICs”, Proc. ISCAS, Helsinki, June 1988, pp. 2431-2434 .
 - [94] S. Piguet, F. Rahali, M. Declercq and M. Kayal : “An Analog-oriented Routing Tool For CMOS Analog Integrated Circuits”, Proc. ESSCIRC89, Vienna, Sept. 1989, pp. 80-83 .
 - [95] S. Piguet, F. Rahali, M. Kayal, E. Zysman and M. Declercq : “A New Routing Methode for Full Custom Analog ICs”, Proc. CICC, Boston, May. 1990,
 - [96] D. J. Garrod, R. A. Rutenbar, L. R. Carley, “Automatic Layout of Custom Analog Cells in ANAGRAM”. In Proc. IEEE Iternational Conf. on CAD, November 1988

Annexes

ANNEXE 1

A.1. OUTILS DE DEVELOPPEMENT

La structure du logiciel CHIRVAN est modélisée sur la figure suivante :



Ce sont alors ces outils : Le-Lisp, Äida2D et Maïda2D que nous allons utiliser.

Le langage utilisé dans CHIRVAN est constitué de trois niveaux. Le **Le-Lisp** est le langage de base, qui a été étendu par l'ajout de **Äida**, qui régit toutes les fonctionnalités de l'environnement X.Windows utilisé (clic-souris ...), puis **Maïda-2D**, pour tout ce qui est objet graphique.

Maïda-2D repose sur le bitmap virtuel de Le-Lisp et ses fonctions de dessin élémentaire. Elle utilise de manière intensive la programmation orientée objet, sous forme vectorielle.

A1.2. LE LISP

Une approche objet

A l'heure actuelle, la majorité des langages dits évolués offrent des possibilités intéressantes par la définition et l'utilisation de fonctions et de procédures. Ce sont typiquement Fortran, Pascal et C.

Il est aussi possible d'opérer, sur n'importe quelle variable de n'importe quel type, une transformation quelconque. Mais cette grande liberté cache des inconvénients inhérents à leur définition.

Considérons donc, à titre d'exemple, deux éléments graphiques : une porte et un tiroir.

En supposant qu'une procédure Ouvre() a été définie et qu'elle puisse indifféremment s'appliquer à la porte et au tiroir, on constate que la transformation à appliquer à chacun des deux objets est profondément différente, puisque dans un cas il s'agit d'une rotation, et dans l'autre d'une translation.

Dans les langages dits évolués il est donc nécessaire de prévoir tous les cas que peut rencontrer la procédure Ouvre() et de les inclure dans son code source. Le programme ainsi fait devient rapidement illisible et d'une maintenance difficile.

Une solution à ce problème est de définir une procédure Ouvre() pour la porte et une autre pour le tiroir. Mais pour peu que le nombre d'objets que l'on peut potentiellement ouvrir soit grand, on se trouve submergé par des procédures aux noms quasiment identiques.

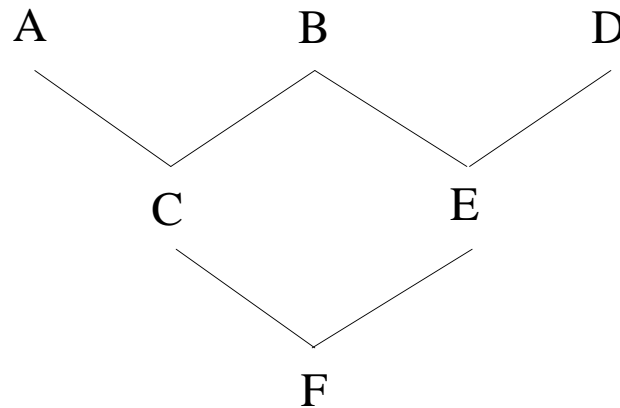
Il est donc nécessaire de créer une nouvelle structure de données plus performante qui pallie ce problème: ce sont les **objets**.

Les langages orientés objets ont donc fait le choix d'autoriser un nom unique pour chaque procédure, mais avec la subtile différence que la procédure Ouvre() d'un objet n'est accessible que par lui et qu'elle lui soit propre. Il existe autant de procédures Ouvre() qu'il n'y a d'objets.

Un objet est une structure de données qui rassemble des variables et des méthodes (nom donné à toutes les fonctions et procédures de l'objet), les méthodes n'ayant accès qu'aux variables internes de l'objet. L'utilisation d'un objet se fait en général à travers les méthodes car, le plus souvent, elles seules sont accessibles depuis le reste du programme (sous entendu l'extérieur de l'objet). Ce phénomène consistant à masquer une partie des données se nomme encapsulation.

Les objets ont également une propriété intéressante: l'**héritage**. Cette notion a été définie par analogie avec l'héritage de caractères génétiques entre individus. Il en est à peu près de même entre objets.

La relation de filiation se fait au moment de la définition de l'objet fils, qui peut ainsi hériter de plusieurs objets en même temps: c'est l'héritage multiple. Ainsi un objet C peut hériter d'un objet A et d'un objet B, et un objet F de C et de E.



Exemple d'héritage entre objets.

La figure précédente propose un schéma type d'héritage entre objets. Au moment de l'héritage l'objet fils acquiert les propriétés (données ou méthodes) que son père objet veut bien lui léguer (ceci dépend évidemment de l'encapsulation des données du père).

Ainsi, deux concepts sont importants dans les langages orientés objet: **classe** et **héritage**.

Une classe modélise un ensemble d'objets ayant une même structure et un même comportement. Plutôt que d'appliquer une méthode à un objet, on dit qu'on lui envoie le message correspondant. Ceci consiste à rechercher la fonction à appliquer à partir de la classe de l'objet à manipuler.

Il est possible d'envoyer un message unique à une liste d'objets hétérogènes, le langage se chargeant de retrouver la partie de code à exécuter pour chaque élément.

L'héritage conduit à la factorisation naturelle du code de diverses classes différentes.

On applique ce type de langage en CAO VLSI car :

- les objets ont un certain nombre de fonctionnalités semblables.
- Il existe de grands types d'objets graphiques qui ont des multitudes de sous-types.
- les réactions d'un objet graphique aux actions de l'utilisateur dépendent souvent de son type.

Approche pratique

Le Le-Lisp est une version du langage LISP réalisée par l'INRIA, nous avons été amenés à utiliser la version 15.25. Le langage est en principe interprété, ce qui permet un développement confortable du logiciel. Cependant, il est aussi possible de compiler le code. De cette façon, nous obtiendrons du code performant.

Ce langage permet la programmation orientée objets. La structure hiérarchique des types dans Le-Lisp est utilisée pour encoder les objets, les méthodes d'accès sont des fonctions Lisp attachées aux objets. L'héritage existe, donc les sous-objets héritent des champs et méthodes de l'objet père. Un objet sera défini de la façon suivante :

```
(defclass <nouveau-objet> ({objet-père})
  champ1
  champ2
  ...
)
```

et une méthode sera définie :

```
(defmethod ({objet} nom-methode (champs-ent) (champs-sort)) (<paramètres>)
  ... code de la méthode ...
)
```

champs-ent est une liste des champs de l'objet qui seront utilisés dans la méthode, une copie locale de ces champs sera effectuée. *Champs-sor* est la liste des champs antérieurs de l'objet qui garderont leurs valeurs en sortant de la méthode.

Supposons *inst1* une instance de l'objet de type *type1*, nous pouvons appeler ses méthodes de deux façons :

```
({type1}:nom-methode inst1 <paramètres>)
```

ou bien on peut envoyer un message, ce qui permettra de remonter l'arbre d'héritage,

```
(send 'nom-methode inst1 <paramètres>)
```

Les champs d'un objet sont accessibles au moyen des méthodes. Les noms des méthodes sont ceux des champs.

Exemples

-- La déclaration d'une classe s'effectue par la fonction **defclass**

```
(defclass 2points ()
  (x 0)
  (y 0)
)

(defclass 3points ({2points})
  (z 0)
)
```

-- La définition d'une méthode

```
(defmethod ({2points} surface (x y)) ()
  (* x y)
)

(defmethod ({3points} volume (x y z)) ()
  (* x y z)
)
```

-- Il existe deux primitives de création d'instances : **new** et **omakeq**

```
(new '{2points})
(omakeq {2points})

(let ((classe '{3points}))
  (new classe)
)
```

-- L'envoi de message à un objet

```
(let ((points (new '{2point})))
  (send 'x points 10)
  (send 'y points 20)
  (send 'surface points)
)
```

A1.3. AïDA - 2D

Un éditeur de masque de circuits intégrés VLSI est un logiciel de dessin en deux dimensions "2D" comme par exemple le Macdraw pour Macintosh, mais la finesse des objets à manipuler en font un outil beaucoup plus sophistiqué.

Contrairement à une simple interface graphique, un outil de dessin complexe ne peut reposer directement sur le système des coordonnées de l'écran. La manipulation d'objets impose à ceux-ci d'être décrit dans un espace de type affine, puis projetés à l'écran en fonction du facteur d'échelle. L'approche orientée objet a pour but de structurer plus clairement les concepts déjà utilisés.

Le but final de la CAO est de représenter des objets réels avec une dimension réelle. Toute grandeur doit donc être exprimée dans l'unité de la base d'objets, elle même modifiable par l'utilisateur.

Aïda permet de construire des interfaces graphiques. C'est un langage de programmation fonctionnelle et objet permettant de programmer et de manipuler des objets graphiques.

Il comprend une bibliothèque de plus de 100 classes de composants graphiques pré définis simples (boutons poussoirs, compteurs) ou complexes (éditeurs de graphes, de courbes ...).

Tout composant graphique est nommé *application*. Les applications sont créées au moyen des fonctions Aïda-2D. Diverses applications peuvent être groupées pour créer des applications plus complexes.

La vocation des composants Aïda-2D est avant tout de décrire des panneaux d'interface et de fournir des moyens d'interaction entre une application et un utilisateur. En conséquence, les mécanismes proposés par Aïda-2D pour réaliser le ré affichage et la sélection d'objets graphiques sont très satisfaisant pour des quantités d'objets restant en deçà de quelques milliers. La définition de comportements complexes (éditeur de texte, jauge graphique, éditeur spécialisés, ...) est très simplement réalisable. Par contre, les opérations de zoom graphique sur des images n'existe pas en standard.

Au-delà de quelques milliers d'objets, le ré affichage d'une partie de ces composants comme la sélection d'un objet dans un de ces composants risquent de devenir un peu lent. Il faudra alors leur préférer des composants Maïda-2D.

A1.4. MAÏDA - 2D

C'est un ensemble de primitives graphiques 2 dimensions écrites en langage objet Lisp.

On utilisera principalement Maïda-2D quand les fonctionnalités suivantes sont nécessaires :

- zoom graphique
- affichage de grand nombre d'objets
- affichage multivues
- masquage fréquent de certains objets.

L'utilisation de Maïda-2D se fait au travers d'Aïda-2D : on utilisera en effet Aïda-2D pour décrire l'ensemble de l'interface de l'application. Il est ainsi possible de considérer Maïda-2D comme afficheur de l'éditeur de graphe.

Tout graphique dans Maïda-2D est un ensemble d'éléments géométriques. Les éléments du graphique sont quant à eux rangés en mémoire dans des données de structure particulières pour pouvoir les manipuler d'une façon optimale.

Tout graphique manipulé par Maïda-2D est représenté par une *cellule* de la classe {cell} qui décrit à la fois la structure d'image du graphique et qui gère la structure de données qui lui est associée.

Une instance de la classe {cell} représente donc un graphique qui est caractérisée par son nom, ses dispositifs d'affichage, ses segments et ses éléments.

Les sous-ensembles d'une cellule sont appelés des *segments*.. Ce sont des instances de la classe {segment}.

Par défaut, il existe les deux segments suivants dans une cellule:

- Le segment Drawing qui contient tous les éléments de la cellule.
- Le segment Sélection qui contient les éléments sélectionnés.

Maïda-2D fournit un ensemble de fonctions qui permettent de manipuler cette structure de données. Elle permettent entre autres d'ajouter et d'enlever un élément; d'accéder rapidement aux éléments se trouvant sous un point donné ou encore inclus dans une région.

Les éléments géométriques constitutifs de tout graphique sont des instances de la classe {lego}. Il en existe deux catégories :

-
- les legos simples: par exemple des instances de la classe {rec} et qui sont des rectangles
 - les instances de cellule: ce sont des legos qui "contiennent" eux-mêmes d'autres legos. Une même cellule peut être instanciée plusieurs fois. Le processus est récursif, autrement dit une cellule contenant des instances peut elle-même être instanciée.

ANNEXE A 2

Cette annexe contient plusieurs fichiers de simulation utilisés ou générés lors de la conception du convertisseur analogiques/numériques. plus particulièrement la simulation et l'optimisation de l'amplificateur opérationnel à transistors composites Fig. A3.1.

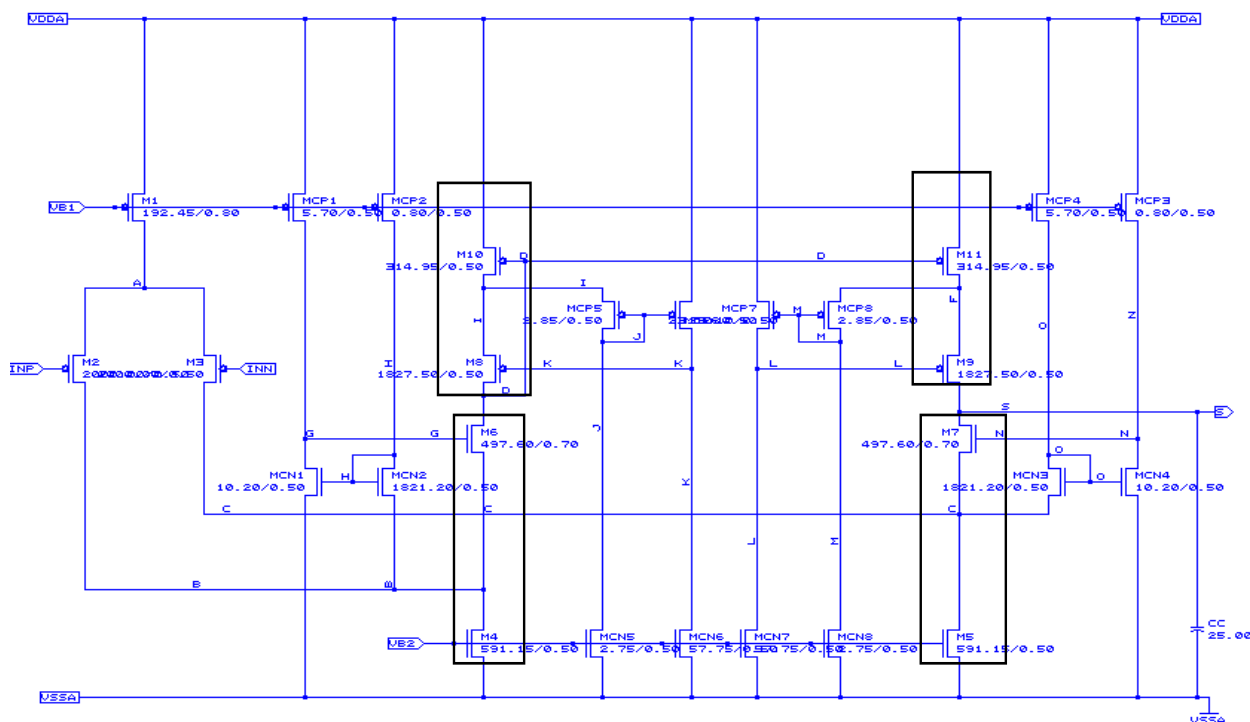


Fig. A3.1 : Schéma de l'amplificateur opérationnel à Transistors Composite final.

Nous allons listé ci-dessous les différents fichiers nécessaires pour le simulateur Eldo ou généré par l'optimiseur Opart:

1) Fichier d'entrée : transcomposite.cir

Le fichier d'entrée au simulateur ELDO est le fichier (transcomposit.cir), il est composé de la netlist de départ (description électrique du schéma initial dimensionné), les commandes de simulation et les spécifications cibles.

Feb 2 14:39 1995 /user/ss30-3/chaahoub/CC05/eldo/transcomposite.cir

```

Transcomposit.cir(schematic)
*
* Main Circuit Netlist:
* Block : Transcomposit
* Library: LibDemo
* Last Time Saved: Jan 5 11:49:10 1995
*
* Model parameter
*****
*
.INCLUDE /user/ss1001/gdt/techno/cc05.dir/cc05.typ
*
*
CC1 S VSSA 25p
MCN4 N O VSSA N3MAS W=51.8u L=0.5u
MCN1 VSSA H G VSSA N3MAS W=51.8u L=0.5u
MCN2 H H B VSSA N3MAS W=914.8u L=0.5u
MCN3 C O O VSSA N3MAS W=914.8u L=0.5u
MCN5 J VB2 VSSA VSSA N3MAS W=1.4u L=0.5u
MCN3 C O O VSSA N3MAS W=914.8u L=0.5u
M7 C N S VSSA N3MAS W=59.2u L=0.5u
M6 D G B VSSA N3MAS W=59.2u L=0.5u
MCN8 M VB2 VSSA VSSA N3MAS W=1.4u L=0.5u
MCN7 L VB2 VSSA VSSA N3MAS W=71.3u L=0.5u
MCN6 K VB2 VSSA VSSA N3MAS W=71.3u L=0.5u
M5 C VB2 VSSA VSSA N3MAS W=721.1u L=0.5u
M4 B VB2 VSSA VSSA N3MAS W=721.1u L=0.5u
M8 D K I VDDA P3MAS W=457u L=0.55u
M9 F L S VDDA P3MAS W=457u L=0.55u
MCP5 I J J VDDA P3MAS W=3.6u L=0.5u
MCP8 M M F VDDA P3MAS W=3.6u L=0.5u
MCP6 K J VDDA VDDA P3MAS W=48.75u L=0.5u
MCP7 VDDA M L VDDA P3MAS W=48.75u L=0.5u
MCP1 G VB1 VDDA VDDA P3MAS W=27.9u L=0.5u
MCP4 O VB1 VDDA VDDA P3MAS W=27.9u L=0.5u
MCP2 H VB1 VDDA VDDA P3MAS W=0.8u L=0.5u
MCP3 N VB1 VDDA VDDA P3MAS W=0.8u L=0.5u
M11 F D VDDA VDDA P3MAS W=294.35u L=0.7u
M10 VDDA D I VDDA P3MAS W=294.35u L=0.7u
M3 A INN C VDDA P3MAS W=1987.8u L=0.5u
M2 B INP A VDDA P3MAS W=1987.8u L=0.5u
M1 A VB1 VDDA VDDA P3MAS W=235.55u L=0.65u

* Bias Definition
*****
*
VDDA VDDA 0 3.000000
VSSA VSSA 0 0.000000
VB1 VB1 0 1.679441
VB2 VB2 0 0.738579
*
*Simulation Commands
*****
*.lfix m_i_2 m_i_3 m_i_4 m_i_5 m_i_6 m_i_7
.lfix mcp1 mcp2 mcp3 mcp4 mcn1 mcn2 mcn3 mcn4
.lfix mcp5 mcp6 mcp7 mcp8 mcn5 mcn6 mcn7 mcn8
*.fix vb1 vb2 m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
*.fix mcn1 mcn2 mcn3 mcn4 mcn5 mcn6 mcn7 mcn8
*.fix mcp1 mcp2 mcp3 mcp4 mcp5 mcp6 mcp7 mcp8

.OPTIONS EPS=1E-7 datasheet

.AC DEC 10 1 1E9

```

```
* VEP INP 0 1.5 AC pwl ( 0 1 100n 1 101n 2 280n 2 281n 1 480n 1 )
* VEM INN 0 1.5
```

```
VEP INP 0 1.5 AC pwl ( 0 1 100n 1 101n 1.8 280n 1.8 281n 1 480n 1 )
VEM INN 0 1.5
```

```
.PLOT AC VDB(S) VP (S)
```

```
.optimize ac
*surface
+ftcible=50e6
+gancible=50
+ mpcible=50
+ puicible=20m
+ wmax=2000u
+v(s)=1.5
+dynout=1.5
+ acout=s
+sleware=0.15e9
```

```
.loop INN s
.tran 10n 480n
.plot tran v(INP) v(s)
optimize tran
*+sleware(v(s) , vth=1.4)>0.15g
```

```
+yval(v(s) , 280n) > 1.798
+yval(v(s) , 280n) < 1.802
+yval(v(s) , 480n) < 1.002
*+ val_when(time, fabs(v(out)-yval(v(out) , 280n) )>0.003, 280n, 100n)- 100n < 25n
+ val_when(time, fabs(v(s)-yval(v(s) , 280n) )>0.002, 280n, 100n)- 100n < 25n
*+ val_when(time, fabs(v(out)-yval(v(out) , 480n) )>0.003, 480n, 280n)- 280n < 25n
+ val_when(time, fabs(v(s)-yval(v(s) , 480n) )>0.002, 480n, 280n)- 280n < 25n
```

2) Fichier de sortie : transcomposite.opz

Apres la simulation et l'optimization de l'amplificateur operationnel, le resultat est donne dans le fichier (transcomposite.opz) :

```
Feb 2 14:39 1995 /user/ss30-3/chaahoub/CC05/eldo/transcomposite.opz
```

```
M1 A VB1 VDDA VDDA MPA W=2.355500E+02U L=5.000000E-01U
M2 B INP A VDDA MPA W=1.987800E+03U L=5.000000E-01U
M3 C INN A VDDA MPA W=1.987800E+03U L=5.000000E-01U
M4 B VB2 VSSA VSSA MNA W=7.211000E+02U L=5.000000E-01U
M5 C VB2 VSSA VSSA MNA W=7.211000E+02U L=5.000000E-01U
M6 D G B VSSA MNA W=5.920000E+01U L=5.000000E-01U
M7 OUT N C VSSA MNA W=5.920000E+01U L=5.000000E-01U
M8 D K I VDDA MPA W=4.570000E+02U L=5.500000E-01U
M9 OUT L F VDDA MPA W=4.570000E+02U L=5.500000E-01U
M10 I D VDDA VDDA MPA W=2.943500E+02U L=7.000000E-01U
M11 F D VDDA VDDA MPA W=2.943500E+02U L=7.000000E-01U
MCN1 G H VSSA VSSA MNA W=5.180000E+01U L=5.000000E-01U
MCN2 H H B VSSA MNA W=9.148000E+02U L=5.000000E-01U
```

```

MCP1 G VB1 VDDA VDDA MPA W=2.790000E+01U L=5.000000E-01U
MCP2 H VB1 VDDA VDDA MPA W=8.000000E-01U L=5.000000E-01U
MCN4 N O VSSA VSSA MNA W=5.180000E+01U L=5.000000E-01U
MCN3 O O C VSSA MNA W=9.148000E+02U L=5.000000E-01U
MCP4 N VB1 VDDA VDDA MPA W=2.790000E+01U L=5.000000E-01U
MCP3 O VB1 VDDA VDDA MPA W=8.000000E-01U L=5.000000E-01U
MCN5 J VB2 VSSA VSSA MNA W=1.400000E+00U L=5.000000E-01U
MCN6 K VB2 VSSA VSSA MNA W=7.130000E+01U L=5.000000E-01U
MCP5 J J I VDDA MPA W=3.600000E+00U L=5.000000E-01U
MCP6 K J VDDA VDDA MPA W=4.875000E+01U L=5.000000E-01U
MCN8 M VB2 VSSA VSSA MNA W=1.400000E+00U L=5.000000E-01U
MCN7 L VB2 VSSA VSSA MNA W=7.130000E+01U L=5.000000E-01U
MCP8 M M F VDDA MPA W=3.600000E+00U L=5.000000E-01U
MCP7 L M VDDA VDDA MPA W=4.875000E+01U L=5.000000E-01U
VDDA VDDA 0 3.000000
VSSA VSSA 0 0.000000
VB1 VB1 0 1.678789
VB2 VB2 0 0.739059
CL OUT VSSA 2.500000E+01P
C_CC S VSSA 2.500000E+01P
* AREA = 3.402682E+04 microns( 2
* POWER = 15.930585 mW target < 20.00000 mW
* V(S) = 1.442775 Volts
* YVAL(V(S) , 280N) = 2.246020
* YVAL(V(S) , 280N) = 2.246020
* YVAL(V(S) , 480N) = 0.750600
*
* VAL_WHEN (TIME , FABS (YVAL(V(S) , 280N) ) > 0.002 , 280N , 100N) - 100N =
2.413024E-08
* VAL_WHEN (TIME , FABS (YVAL(V(S) , 480N) ) > 0.002 , 480N , 280N) - 280N =
2.508433E-08
*
* GAIN IN LOW FREQUENCY = 67.310291 dB target > 50.00000 dB
* TRANSITION FREQUENCY = 1.159998E+08 Hz target > 5.00000E+07 Hz
* PHASE MARGIN = 65.307569 deg target > 50.00000 deg
* GAIN MARGIN = 13.961592 dB target > 1.00000 dB
* INPUT DYNAMIC = 2.101325 V target > 1.80000 V
* CMG = 2.127818E+00 dB
* CMRR = 6.343811E+01 dB
* OUTPUT DYNAMIC = 2.1217033 V target > 1.500000 V
* CLOSED-LOOP DISTORTION = -8.102903E+01 dB
* GAIN+ = 6.045649E+01 dB
* GAINMI = 6.131029E+01 dB
* GAIN- = 6.054849E+01 dB
* SLEWRATE ON OUTPUT = 0.164524 V/ns V target > 0.15 V/ns

```

3) Fichier de sortie : transcomposite.imp.

Avant l'optimisation proprement dite, OPART génère un fichier de sortie .IMP contenant de nombreuses informations propres au schéma au schéma électrique. Ces informations sont exploitées par l'optimiseur Opart lui-même et par le placeur de Chirvan.

```

(ALIMENTATIONS (( VDD VDDA ) ( VSS VSSA )))
(CHEMINS_DE_COURANT ((( VDDA A B VSSA ) ( M_M1 M_M2 M_M4 ))
((( VDDA A C VSSA ) ( M_M1 M_M3 M_M5 )))
((( VDDA G VSSA ) ( M_MCP1 M_MCN1 )))
((( VDDA H B VSSA ) ( M_MCP2 M_MCN2 M_M4 )))
((( VDDA I D B VSSA ) ( M_M10 M_M8 M_M6 M_M4 )))
((( VDDA I J VSSA ) ( M_M10 M_MCP5 M_MCN5 )))

```

```

((( VDDA K VSSA ) ( M_MCP6 M_MCN6 ))
((( VDDA L VSSA ) ( M_MCP7 M_MCN7 ))
((( VDDA F M VSSA ) ( M_M11 M_MCP8 M_MCN8 ))
((( VDDA F S C VSSA ) ( M_M11 M_M9 M_M7 M_M5 ))
((( VDDA O C VSSA ) ( M_MCP4 M_MCN3 M_M5 ))
((( VDDA N VSSA ) ( M_MCP3 M_MCN4 ))
)
))
( PAIRE_DIFFERENTIELLE ( M_M2 M_3 ))
( NOEUDS_DE_SORTIES ( S ))
(( PAIRES_SYMETRIQUES ( SYMMETRIC_OBJECTS ( M_M1 M_M2 M_M4 ) ( M_M1 M_M3
M_M5 ))
(( M_MCP1 M_MCN1 ) ( M_MCP3 M_MCN4 ))
(( M_MCP2 M_MCN2 M_M4 ) ( M_MCP4 M_MCN3 M_M5 ))
(( M_M10 M_M8 M_M6 M_M4 ) ( M_M11 M_M9 M_M7 M_M5 ))
(( M_M10 M_MCP5 M_MCN5 ) ( M_M11 M_MCP8 M_MCN8 ))
(( M_MCP6 M_MCN6 ) ( M_MCP7 M_MCN7 ))
))
( NOEUDS_SYMMETRIQUES (( VDDA VDDA ) ( C B ) ( A A ) ( G N ) ( K L ) ( H O ) ( F I )
( J M ) ( VSSA VSSA ))
)
( NOEUDS_A_FORT_GAIN ( S ))
( NOEUDS_A_PLUS_FORT_GAIN S )

```

Table de Matieres

Introduction GENERALE	6
1.1. Introduction	10
1.2. Justification de la conception des Circuits Intégrés CMOS à Basse Tension d’Alimentation et à Faible consommation	11
1.3. Impact de la réduction de la tension d’alimentation sur la conception des circuits intégrés mixte:	13
1.4. Les limitations de la conception des circuits intégrés analogiques à basse tension d’alimentation et à faible consommation	14
1.4.1 Limite fondamentale	14
1.4.2 Les limites d’implémentation réelle des circuits intégrés analogiques et mixtes	14
1.5. Classification des circuits CMOS à basse tension d’alimentation	16
1.6. Propriétés électriques des transistors MOS	17
1.6.1. Notations utilisées	17
1.6.2. Modèle petits signaux	17
1.6.3. Le Bruit dans les transistors MOS	19
1.6.4. Régions d’opération du transistor MOS et ces caractéristiques Courant-Tension [16]	21
1.10. Conclusion	24
2.1. Introduction	27
2.2 Principe de la conversion	28
2.3. Description générale du système	29
2.4. Justification des performances demandées	38
2.5. LIMITATION DES STRUCTURES CMOS TRADITIONNELLES	42
2.5.1. Etage de gain simple:	42
2.5.2. Etage de gain de circuit cascode:	47
2.6. NOUVELLES STRUCTURES	51
2.6.1. Circuit Cascode Régulé [30][31][32]	51
2.6.2. Circuit Composite (Méthode des transistors composites) [34][35]	58
2.7. APPLICATION DE LA METHODE DES TRANSISTORS COMPOSITES	60
2.8. Conclusion	68
3.1. Introduction	72

3.2. Principaux phénomènes parasites rencontrés dans un circuit intégré	
analogique	73
3.2.1. Capacités parasites	73
3.2.2. Résistances parasites	74
3.2.3. Inductances parasites	75
3.2.4. Composants parasites actifs	75
3.2.5. Appairage de composants (Matching)	75
3.2.6. Interactions électrothermiques	76
3.2.7. Incertitude sur les paramètres de la technologie	76
3.3. Contraintes à appliquer lors de la conception d'un circuit intégré	
analogique	77
3.3.1. Contraintes sur les performances attendues du circuit intégré analogique	77
3.3.2. Contraintes sur le circuit final	78
3.3.3. Contraintes de Placement	80
3.3.4. Contraintes de Routage	82
3.3.5. Contraintes de Compaction	84
3.4. Conclusion	85
4.1. Introduction	87
4.2. Méthodologie de conception des circuits intégrés analogiques au CNET	
Grenoble	89
4.3. Le logiciel CHIRVAN.	91
4.4. Génération des composants	94
4.4.1. Le MOS	95
4.4.2. Le MOS de Charge	96
4.4.3. La Paire Différentielle	96
4.4.4. Le Miroir de Courant	97
4.4.5. La Fusion de deux Transistors ou Groupes de Transistors	97
4.4.6. Les Prises Caissons et Substrats	98
4.4.6. La Résistance	98
4.4.7. Les Capacités	99
4.5. Routage	100
4.5.1 Méthodologie	100
4.5.2. Description des Contraintes Géométriques	102
4.5.3. Algorithme de Routage	102
4.6. Compaction	110
4.6.1. But	110
4.6.2. Méthodologie	111
4.6.3. Description d'Objet	112

4.6.4. Contraintes de Descriptions	113
4.6.5. Construction des Contraintes	114
4.6.6. Algorithmes d'Optimisation	115
4.6.7. Traitement du layout	118
4.7. Resultats Experimentaux	118
4.8. Conclusion	121
5.1. Introduction	124
5.2. Formulation du problème	125
5.3. Vue générale de l'outil de Placement	128
5.4. Présentation des différents algorithmes utilises pour résoudre le problème de placement	130
5.4.1. Placement Constructif	130
5.4.2. Placement Dirigé par les Forces	131
5.4.3. Placement par Partition	131
5.4.4. Optimisation Quadratique	131
5.4.5. Placement Génétique	132
5.4.6. Recuit Simulé	132
5.4.7. Discussion	134
5.5. Implementation de la Méthode du Recuit Simulé pour le placement des cellules analogiques	137
5.5.1. Représentation du Placement	138
5.6. Manipuler les contraintes analogiques en recuit simulé	141
5.7. Fonction de coût	144
5.8. Conclusion	146
Conclusion GENERALE	148
Références	151
Annexe 1	159
A.1. OUTILS DE DEVELOPPEMENT	159
A1.2. Le Lisp	159
Une approche objet	160
Approche pratique	162
Exemples	163
A1.3. Aïda - 2D	164
A1.4. Maïda - 2D	165
