

## リーフセル用配線システム

四宮 典子<sup>†</sup>、西垣 泰男<sup>‡</sup>、福井 正博<sup>†</sup>

<sup>†</sup>松下電器産業株式会社 半導体先行開発センター <sup>‡</sup>株式会社松下ソフトリサーチ  
570 守口市八雲中町 3-1-1  
{bekku, nisigaki, fukui}@vdrl.src.mei.co.jp

あらし ライブラリの開発コスト削減およびTime to market短縮のために、ライブラリ自動合成技術への要望が高まっている。本稿ではセル内部のトランジスタ間配線手法について述べる。

本手法は引きはがし再配線手法であり、従来より一般的に用いられる配線長、コンタクト数、設計規則違反の他に、スタンダードセル設計に特有なセル高さ制約に関するコストを評価指標として配線経路改善を行ない、同時にトランジスタ端子位置の最適化を行なう。本手法により人手並の配線結果を得ることができた。

キーワード 引きはがし再配線 スタンダードセル セル高さ制約 レイアウト自動合成

## A Routing System for Leaf Cell Design

Noriko Shinomiya<sup>†</sup>, Yasuo Nishigaki<sup>‡</sup>, Masahiro Fukui<sup>†</sup>

<sup>†</sup>Advanced LSI Technology Development Center, Matsushita Electric Industrial Co., Ltd.  
<sup>‡</sup>Matsushita Soft-Research, Inc.  
3-1-1 Yakumo-Nakamachi, Moriguchi, Osaka 570, Japan.  
{bekku, nisigaki, fukui}@vdrl.src.mei.co.jp

### Abstract

Due to the strong requirement to reduce the library development cost and its time-to-market, automatic library synthesis is getting an important technology. This paper proposes a transistor-level routing method for library cell layout.

We have improved conventional rip-up and rerouting method so as to be used for standard-cell design by considering cell height constraint, optimization of terminal positions of transistors. The routing results are comparable to manual layout.

key words rip-up and reroute, standard-cell, cell height constraint, layout synthesis

# 1. はじめに

ライブラリの開発コスト削減およびTime to market短縮のために、ライブラリ自動合成技術への要望が高まっている。

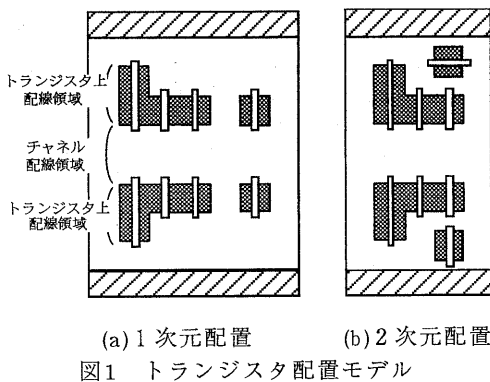
我々はスタンダードセル設計において人手並のレイアウト結果を得ることを目的とし、トランジスタの2次元配置モデルを用いたセルレイアウト自動合成システムの提案を行なって来た[1,2]。

本セル合成システムは、(1)トランジスタ配置、(2)トランジスタ間配線、(3)コンパクションの3つの処理からなる。本稿ではトランジスタ間配線について述べる。

従来のトランジスタ間配線技術では[3,4]が着目に値する。[3]は、P型トランジスタ列とN型トランジスタ列を平行に配置するトランジスタ1次元配置モデルを前提とし、同一トランジスタ列内のソース/ドレイン間の接続はトランジスタ上配線領域を用いて行なう。トランジスタ列上に収容しきれなかったネットと残りのネットに対しセル中央部においてチャンネル配線を行なう(図1(a))。

一方、我々が前提とする2次元配置モデルは人手設計並みの自由度を持ち、トランジスタの縦積みや、横向きトランジスタを扱うことができる(図1(b))。このモデルではレイアウト領域全体に広がったトランジスタ端子間を配線しなければならないのでエリア配線手法(迷路法や線分探索法など)の適用が考えられる。しかし迷路法や線分探索法で逐次配線を行なうと、既配線が障害となり後の配線が不可能となる問題がある。これを解決する方法として、引きはがし再配線手法が提案されている[4,5,6,7,8,9,10]。我々の配線システムも引きはがし再配線手法をベースとして開発を行なった。

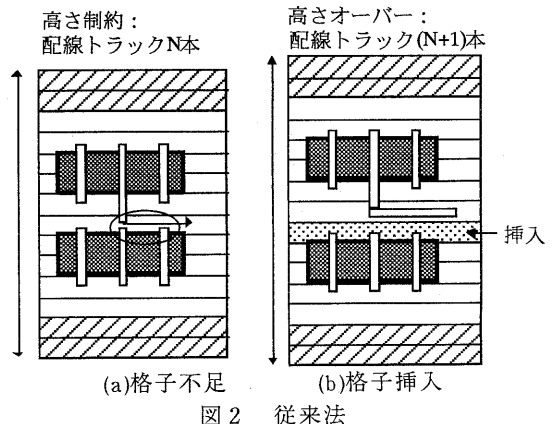
トランジスタ配置が配線処理に与えるデータは、オブジェクト(トランジスタ、端子)の配置情報で



ある。オブジェクトの相互間隔は、同領域を通過する配線本数を見積ることにより与える。ここで、オブジェクト間隔を正確に見積ることは困難であり、人手設計では、配線をしながらオブジェクト間隔を調整するといった試行錯誤が繰り返される。

オブジェクト間の配線本数を少なく見積もった場合、オブジェクト間隔が狭くなり、格子不足による配線失敗を生じる。これを解決するために[4]は、新たな格子を挿入してレイアウト領域を広げる方法を提案している。たとえば、スタンダードセルのレイアウト設計は、セル高さが与えられる(セル高さ制約)が、図2(a)は指定のセル高さ(配線トラックN本分相当)でトランジスタを配置した場合で、セル中央付近で格子不足による配線失敗を生じる例を示す。この場合に[4]は図2(b)のように、格子を1つ挿入して配線を完了する。レイアウト領域の拡大により、このままではセル高さ違反を生じるので、高さ制約を満足するために、別の場所の格子を1つ削除するなどの対策を行なう。しかしながらこのような方法では、セル全体にとって最適な格子線挿入位置を求めることは難しく、また格子の挿入削除により処理が複雑になる。

この問題に対する我々のアプローチは、まずオブジェクト間の格子本数をやや多めに設定して配線経路の自由度を確保し、配線完了率を上げる。同時に、セル高さ制約を守るために、各格子列に割当てることができる配線本数である「容量」を定義し、各格子列への現在の割当て配線本数が容量を越える「容量違反」を引きはがし再配線により解消することにより、セル高さ制約を満たしながら全体で最適な配線結果を得るものである。



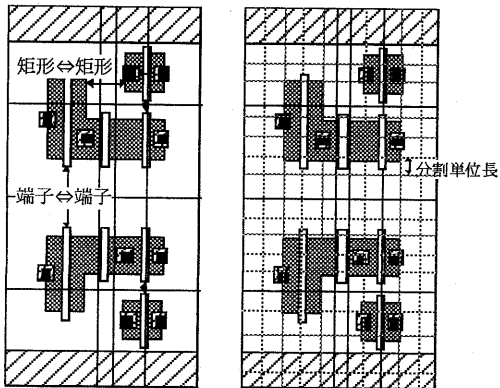
## 2. レイアウトモデル

### 2.1 セル・アーキテクチャ

セル高さは予め与えられる。セルの横幅は自由である。信号配線はポリシリコン層(Poly)と第1金属層(AL1)を使用し、電源配線はAL1でセルの上下端に置く。各層における配線方向に制約はない。ソース/ドレイン端子は、コンタクトを拡散領域に最大限設ける形状(非サリサイドプロセス)と、1コンタクト形状(サリサイドプロセス)があり、プロセスにより選択する。外部端子には、セル間配置配線ツールに適した配置制約(一列、交互など)を与えることができる。なお、電源位置の変更や配線層(拡散層や第2金属層など)の追加などのセル・アーキテクチャ変更への対応も容易に行なえる。

### 2.2 格子モデル

配線領域は矩形であり、 $n$ 層の配線層に対応可能である。配線領域を格子に分割する。格子の大きさは均一である必要はないが、配線層面に垂直な方向から見たとき、各層の格子は一致する。レイアウト領域内には配線禁止領域および端子が存在する。配線層間の接続はコンタクトを介して行なう。我々は、オブジェクトの配置情報(非格子データ)から、配線用格子データを以下の手順で作成する。(1)異電位間の格子分割(図3(a))、(2)間隔の広い格子の細分割(図3(b))。(1)で、最低限必要な格子を作成する。(2)では、分割単位長を制御して、レイアウト領域の格子数を変化させる。格子不足による配線失敗が生じた場合、分割単位長を短くして格子数を増やし配線をやり直す。格子間隔が狭いとき、隣合う配線(コンタクト)間でデザイン・ルール違反が生じるが、コンパクションによりこれを解消する。



(a)異電位間の格子分割 (b)広い格子の細分割

図3 格子分割

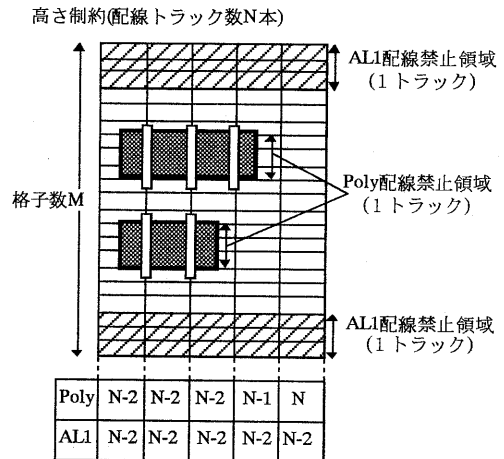
### 2.3 容量および容量違反

セル高さ方向の格子の集合である格子列において、配線を割当てることができるトラック数を「容量」と定義する。セル高さからその格子列を占める配線禁止領域高さを引いた値を図4に示す配線トラック幅で割ったものである。

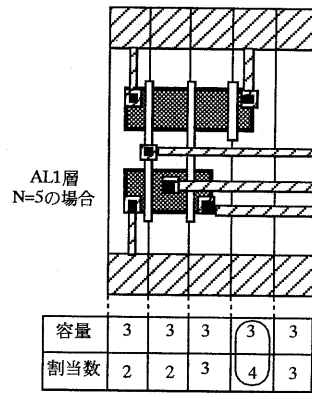
格子列への割当て配線本数が「容量」より大きい状態を「容量違反」と呼ぶ。容量違反が起きるとセル高さをオーバーする。図5に容量と容量違反の例を示す。



図4 配線トラック幅



(a)格子列毎の容量



(b)容量違反の例

図5 容量と容量違反

### 3. 本システムの特徴

#### 3.1 セル高さ制約

セル高さ制約を満足する配線を行なう。

#### 3.2 ソース/ドレイン端子位置最適化

サリサイド技術等を用いた場合、1コンタクトのソース/ドレイン端子が実現でき、トランジスタ上配線が可能となる。セル面積削減のため周囲の配線状況に応じて端子位置を最適に決定する。

#### 3.3 ゲート間配線

性能の点から、同ネットのP型トランジスタとN型トランジスタのゲート間配線をなるべくPoly層を用いてできる限りセルの中央付近で配線する。

## 4. アルゴリズム

図6に配線アルゴリズムのフローを示す。初期配線と引きはがし再配線の2フェーズに別れる。初期配線の後、見積り配線長、設計規則違反数、容量違反数などの評価指標に基づいて引きはがし再配線を行ない経路改善を行なう。配線経路探索には迷路法[11]を用いる。

#### 4.1 初期配線

初期配線では、ネットの配線経路を他のネットの配線経路に無関係に独立して求める。このとき配線やコンタクトの重なりといった設計規則違反およびセル高さオーバーを意味する容量違反を許して配線経路を求める。なお、配線禁止領域への配線経路割り当ては認めない。初期配線の配線経路コストを以下に示す。これらのコストは一般的に用いられるもの[11]と同様である。

$$\text{Cost} = a \times (\text{配線長}) + b \times (\text{コンタクト数})$$

ここに、 $a, b$ はパラメータ

まず、配線すべきサブネットの集合を求める。サブネットとは、同電位である接続された端子と配線の集合である。複数のサブネットが存在するとき、2つのサブネット対に分解してその間を配線する2サブネット配線と複数のサブネットを一度に配線する多サブネット配線の2つのモードを持つ。本配線システムは多サブネット配線を可能な限り続ける。多サブネット配線では、まず格子数が最大のサブネットを始点とし、残るサブネット全てを終点とする。次に、始点から全ての終点を包含する最小矩形で定義するターゲット矩形までの配線経路を求める。この時すでに、始点がターゲット矩形上にあれば、2サブネット配線

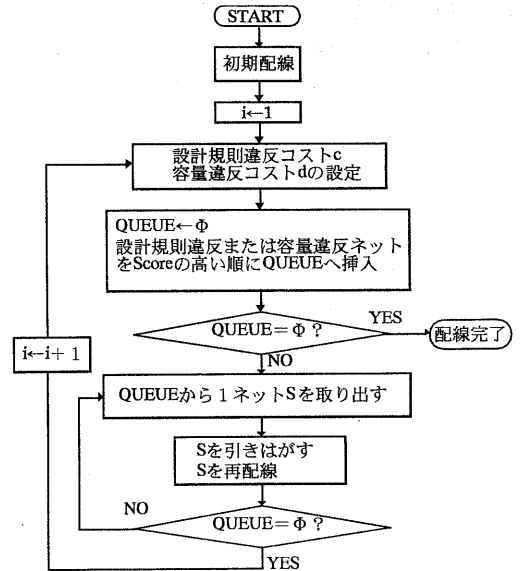


図6 配線アルゴリズム・フロー

を行なう。2サブネット配線は、マンハッタン距離が小さいサブネット対から順に配線するもので、格子数が多いサブネットを始点とする。

#### 4.2 配線引きはがし

設計規則違反または容量違反を生じているネットを引きはがす。ネットの得点Scoreを以下で求め、得点の高い順に引きはがし再配線を行なう。

$$\begin{aligned} \text{Score} = & a \times (\text{見積り配線長}) \\ & + b \times (\text{設計規則違反個数}) \\ & + c \times (\text{容量違反個数}) \end{aligned}$$

ここに、 $a, b, c$ はパラメータ、見積り配線長はネットの端子を囲む最小矩形の1/2周囲長

$a$ は、見積り配線長が長いほど可能な配線経路の数が多いと考えられ、引きはがしても他の配線経路を見つけ易いと考えられるためである。

$b$ は設計規則違反が多いほど $c$ は容量違反が多いほど引きはがされやすくするためのものである。

#### 4.3 再配線

再配線における配線経路コストは、初期配線のコストに、設計規則違反コストおよび容量違反コストを加えたものである。

$$\begin{aligned} \text{Cost} = & a \times (\text{配線長}) \\ & + b \times (\text{コンタクト数}) \\ & + c \times (\text{設計規則違反個数}) \\ & + d \times (\text{容量違反数}) \end{aligned}$$

ここに、 $a, b, c, d$ はパラメータ

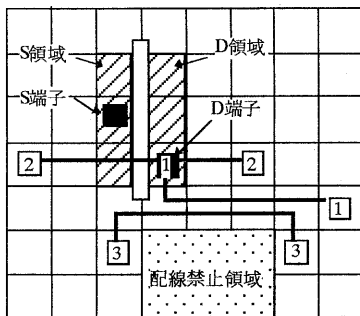
設計規則違反コストは、[10]と同様に設計規則違反を解消するために、引きはがし再配線の繰り返しが進むにつれて $c$ を大きくする。

また、容量違反コストは本システム独自のものであり、セル高さ制約を満たすために、引きはがし再配線の繰り返し数 $i$ が大きくなるにつれて $d$ を大きくする。現在、 $d = \alpha^i$  ( $\alpha$ は定数)としている。

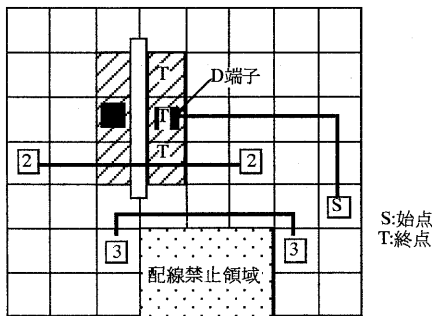
#### 4.4 ソース/ドレイン端子位置最適化

配線経路探索において、配線とソース/ドレイン(S/Dと呼ぶ)端子との重なりを許すものとする。

まず、初期配線において、他のネットと独立してS/D端子の初期位置を求める(図8(a))。続く引きはがし再配線において、S/D端子につながる配線が引きはがされた場合、S/D領域全体をS/D端子の候補として再配線経路を求め、この経路とS/D領域が接する場所にS/D端子を移動する(図8(b))。このように、S/D端子位置の移動も含めた引きはがし再配線を行なう。



(a) S/D端子初期位置決定



(b) 端子も含めた引きはがし再配線

図8 S/D端子位置最適化

#### 4.5 ゲート間配線

セル中央側にあるゲート端子からの配線経路が優先して求まるように、セル中央から遠いゲート

端子に予め高いコスト(基底コストと呼ぶ)を付加し、探索を遅らせる。

図7において、 $tp1, tp2, tn1$ のゲートが同一ネットとする。 $tp1$ のゲートを始点(S)、残りのゲートを終点(T)とする。 $tp1$ の中央から離れたゲート端子Sに基底コストを与えることにより、配線長最小の点線で示す経路よりもセル中央の経路が選ばれる。基底コストの値としては、セル中央にあるゲートの始点集合と終点集合との間の最小マンハッタン距離を用いる。

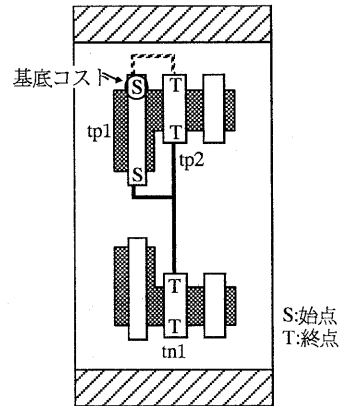


図7 基底コスト

### 5. 実験結果

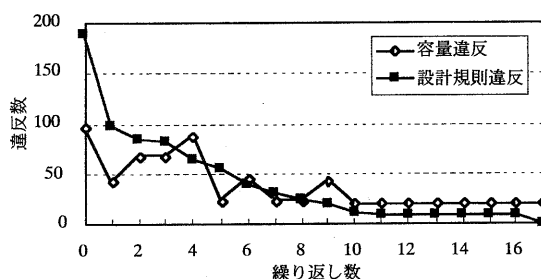
人手配置結果を用いて本手法により配線を行なった。人手配線結果と本手法による自動配線結果をコンタクト数と総配線長に関し比較した結果を表1に示す。コンタクト数はほぼ人手並みである。配線長に関しては、自動配線結果はコンパクション前の結果であり、最終的にはやや表の値よりも増加すると思われるが、ほぼ人手並みの結果が得られたと言える。

表1 人手との比較

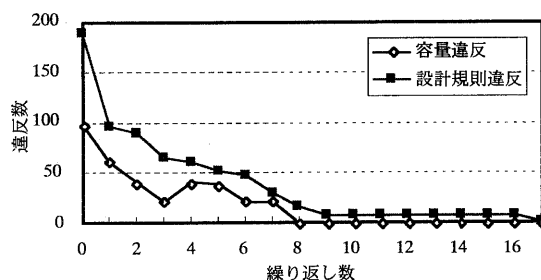
セル名	# Trs	# contacts (人手)	# contacts (自動)	total length (人手)	total length (自動)	CPU Time (sec)
AND2	6	3	3	55	54	2.3
BUF8	9	4	4	61	64	5.1
OR4	10	5	5	60	52	3.0
CMBI	18	9	8	142	134	10.1
NOR6	20	12	10	182	168	15.1
NAND	25	8	14	221	200	73.3
LTCH	26	14	11	193	175	59.2
FF1	36	22	22	323	336	392.8
DF1	56	40	30	476	449	150.5

本実験では、セル高さ制約をPoly層の $N=19$ 、A L1層の $N=16$ とした。セルFF1で行なった実験について説明する。水平方向の格子数は全層で同一の $M=33$ であった。まず、容量違反コストを与えずに配線した場合の引きはがし再配線の繰り返しに伴う設計規則違反数および容量違反数の変化の様子を図8(a)に示す。設計規則違反の解消が進むにつれて容量違反数も減少する傾向にあるが、設計規則違反数が0になり配線処理を終えた時点でも、まだ容量違反が残っていることがわかる。

次に、容量違反コストを与えた実験では、図8(b)に示すように17回の繰り返しで、両違反を無くすことができた。この時の配線結果を図9に示す。



(a) 設計規則違反コスト有、容量違反コスト無



(b) 設計規則違反コスト有、容量違反コスト有  
図8 繰り返し数と違反数の関係(セルFF1)

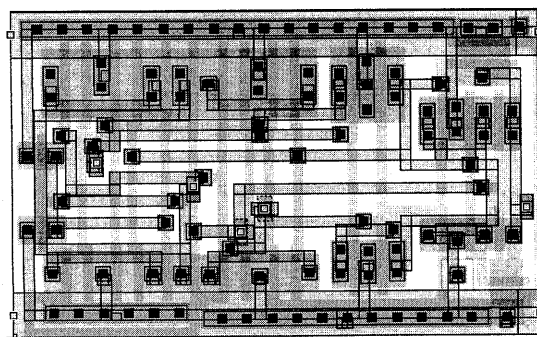
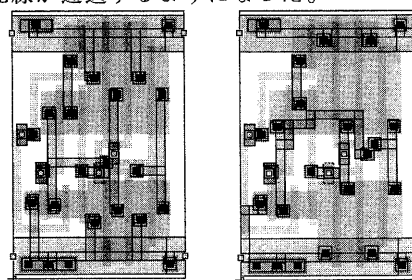


図9 配線結果(セルFF1)

図10はS/D端子位置を最適化しない場合(a)とした場合(b)を示す。最適化した場合は、配線長が短くなる位置に端子が移動し、トランジスタ上に金属配線が通過するようになった。



(a)最適化なし (b)最適化あり

図10 端子位置最適化

## 6. 結論

本稿では、セル高さ制約に関するコストを評価指標とする引きはがし再配線手法を提案した。また、ソース/ドレイン端子位置の最適化も行うようにした。計算機実験の結果、人手並の良好な結果を得ることができた。

## 参考文献

- [1] M. Fukui, N. Shinomiya and T. Akino, "A New Layout Synthesis for Leaf Cell Design," *Proc. ASP-DAC'95*, pp. 259-264. (1995).
- [2] S. Saika, M. Fukui, N. Shinomiya and T. Akino, "A Two-dimensional Transistor Placement for Cell Synthesis," *Proc. ASP-DAC'97* pp. 557-562. (1997).
- [3] C. L. Ong, J. T. Li and C. Y. Lo, "GENAC: An Automatic Cell Synthesis Tool," *Proc. 26th Design Automation Conference*, pp. 239-244. (1989).
- [4] C. J. POIRIER, "Excellerator: Custom CMOS Leaf Cell Layout Generator," *IEEE Trans. on COMPUTER-AIDED DESIGN. VOL8 NO. 7*, pp. 744-755. (1989).
- [5] H. Shin and A. S. Vincentelh, "MIGHTY: A Rip-Up and Reroute Detailed Router," *Proc. ICCAD*, Nov. pp. 2-5. (1986).
- [6] E. Rosenberg, "A New Iterative Supply/Demand Router with Rip-Up Capability for Printed Circuit Boards," *Proc. 24th Design Automation Conference*, pp. 721-726 (1987).
- [7] Y. L. Lin, U. C. Hsu and F. S. Tsai, "SILK: A Simulated Evolution Router," *IEEE Trans. on Computer-Aided Design*, vol.8 no. 10, October, pp. 1108-1114. (1989).
- [8] P. S. Tzeng and C. H. Sequin, "Codar: A Congestion-Directed General Area Router," *Proc. ICCAD*, pp. 30-33. (1988).
- [9] K. Kawamura, T. Shindo, T. Shibuya, H. Miwatari and Y. Ohki, "Touch and Cross Router," *Proc. ICCAD* pp. 56-59. (1990).
- [10] 羽根, 油井, 島本, 白川, 西口, "引き剥し再配線手法を用いた分散処理型多層VLSI配線システム," *信学技報, C AS-91-18*, pp. 29-36. (1991).
- [11] F. Rubin, "The Lee Path Connection Algorithm," *IEEE Trans. on COMPUTERS*, vol. c-23, no.9 Sep. PP. 907-914. (1974).