

## 解説



### 3. レイアウト設計における CAD

#### 3.4 シンボリック・レイアウト設計手法†

田丸 喜一郎†

##### 1. まえがき

VLSI のレイアウト設計において自動配置配線手法と並んでシンボリック・レイアウト設計手法が注目を集めている。従来から、レイアウト設計は単調で設計誤りも多く発生する工程であり、LSI の設計期間の多くを占める工程でもある。自動配置配線手法は、このようなレイアウト設計の工程をすべて CAD 化したもので、数多くのシステムが開発され広く実用化されている。しかし、自動配置配線手法では手設計と比較すると集積密度の点で劣るため、適用が困難な分野も多い。シンボリック・レイアウト設計手法は、自動配置配線手法と比べると人手はかかるが、より手設計に近いレイアウトが得られるため、自動配置配線手法の適用が困難な分野の多くに適用することが可能である。

本稿では、シンボリック・レイアウト設計手法の概要、シンボリック・レイアウト設計のための各種支援ツール、自動圧縮のアルゴリズムなどについて概説する。

##### 2. シンボリック・レイアウト設計手法

シンボリック・レイアウト設計手法は、トランジスタ、コンタクト、配線といった回路の構成要素をシンボルを用いて記述し、レイアウトを行う手法である。大別すると、シンボルで記述したレイアウト図（シンボリック図）での設計ルールを設定し、設計者がその設計ルールに従ってシンボリック図を作成した後、単純に変換して最終的なマスク・パターンを得る方法<sup>7)~10)</sup>と、設計者がラフなシンボリック図を作成し、CAD システムで自動的に設計ルールに従って圧縮を行い最終的なマスク・パターンを得る方法がある<sup>1)~6)</sup>。

シンボリック・レイアウト設計手法を用いると、自動圧縮を行わない場合でも、マスク・パターンを直接記述する場合に比べると

- (1) 記述量が減少する
- (2) 図面倍率が低下し回路全体の見通しがよくなる
- (3) 設計ルールが単純になる

などの理由によりレイアウト設計の期間はかなり短縮される。

自動圧縮を行う場合には、更に、

- (1) 設計ルールを意識しなくてよい
- (2) 修正、変更が容易である（回路の追加、削除の際に、スペースを空けたり、スペースを詰めたりする必要がない）

ため、設計期間は大幅に短縮される。

図-1 にレイアウト設計における各種設計手法の関係を示す。自動配置配線手法は、論理/トランジスタ回路図からマスク・パターン図へ CAD システムが変換する。シンボリック・レイアウト設計手法では、自動圧縮を行う場合には、人手でラフなシンボリック図を作成し、それからマスク・パターン図へ CAD システムが変換する。自動圧縮を行わない場合は、人手で設計ルールに従ったシンボリック図を作成し、それからマスク・パターン図への変換だけを CAD システムが行う。

##### 3. シンボリック・レイアウト設計のための支援ツール

シンボリック・レイアウト設計手法のための支援ツールには、シンボリック図の作成を支援するツール、シンボリック図から最終的なマスク・パターンを生成するツール、シンボリック図の検証を支援するツールがある。

###### 3.1 シンボリック図のエディティング

対話形式でシンボリック図の作成を支援するツール

† Symbolic Layout Design Method by Kiichiro TAMARU (Semiconductor Device Engineering Laboratory, TOSHIBA Corporation).

†† (株)東芝半導体技術研究所

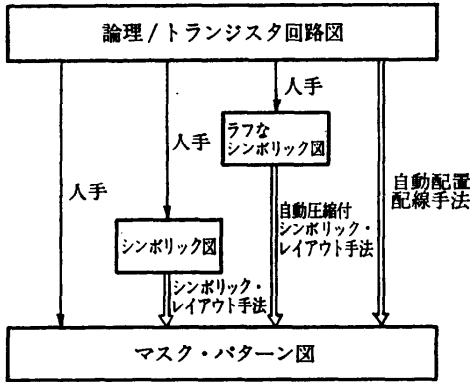


図-1 各種設計手法によるレイアウト設計手順

が、シンボリック図エディタである。レイアウトの要素をシンボルで取り扱う点は、論理/トランジスタ回路図エディタに類似するが、要素の位置関係に意味があり、また、配線層等を意識する必要がある点は、マスク・パターン図エディタに類似する。したがってシンボリック図のエディタでは両者の機能が必要で

ある。

図-2(a)にシンボリック図エディタで入力されたラフなシンボリック図の例を示す。

### 3.2 シンボリック図の圧縮

入力されたラフなシンボリック図を、設計ルールに従って圧縮し最終的なマスク・パターンに自動変換するツールが自動圧縮プログラムである。圧縮のアルゴリズムについては4章で述べるが、圧縮の1例を図-2に示す。(a)のラフなシンボリック図の回路を横方向、縦方向に圧縮し、結果として(d)のマスク・パターンが得られる。

### 3.3 シンボリック図の検証

自動圧縮を行わないシンボリック・レイアウト設計手法では、シンボリック図の設計ルールに関する検査が必要である。設計ルールの検査は、シンボリック図をマスク・パターンに変換した後にすることもできるが、シンボリック図の段階で行えばシンボル間の単純な設計ルールで検査できるため、大幅に簡単化される。

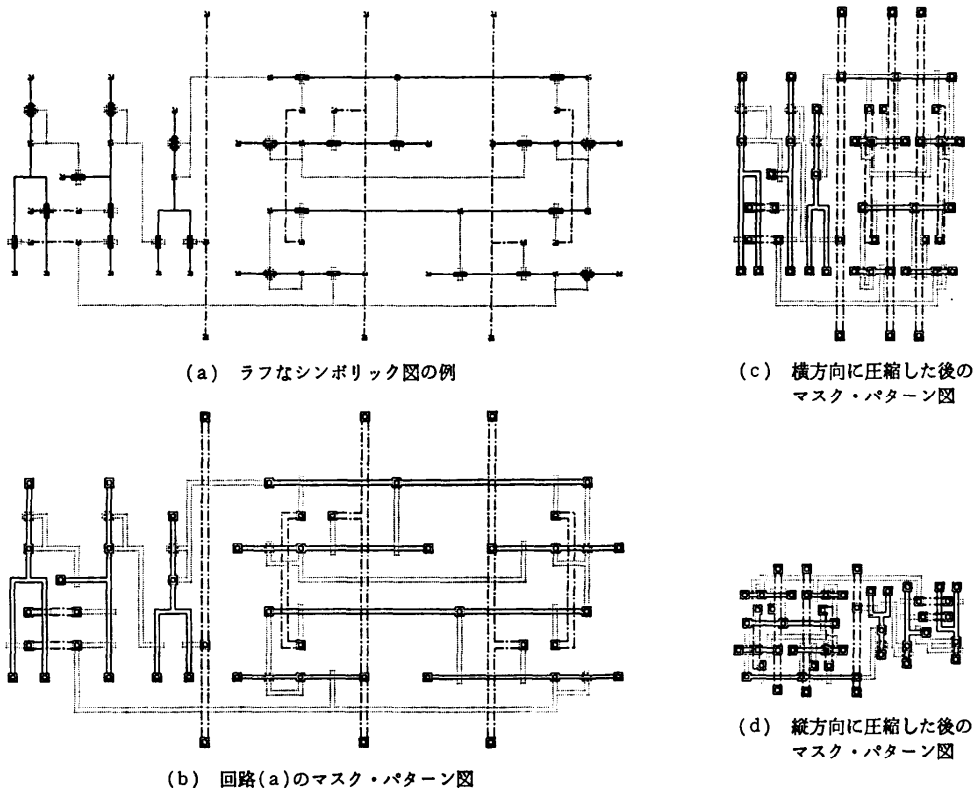


図-2

自動圧縮を行うシンボリック・レイアウト設計手法では、圧縮されたシンボリック図の設計ルール検査は必要である。設計ルールに従って圧縮が行われるからである。

シンボリック図に関する検証には、他に、論理/トランジスタ回路図との対応関係の検証がある。自動配置配線手法と異なりシンボリック・レイアウト設計手法では、論理/トランジスタ回路図からシンボリックによるレイアウト図への変換は人手で行われるため、両者の対応関係の検査が必要となる。このような検査は、シンボリック図から回路情報を抽出し、シミュレーションや回路の比較により行われる。シンボリック図からの回路情報の抽出は、マスク・パターンからの場合と比較すると、トランジスタ、コンタクトといった回路の構成要素がそれぞれ1つのシンボルで記述されているため大幅に簡単化される。

このようにシンボリック図の検証は、マスク・パターン図の検証と比較すると、容易にかつ短期間に行える。これも、シンボリック・レイアウト設計手法の利点の1つである。

4. 圧縮のアルゴリズム

ラフなシンボリック図から設計ルールに従って最終的なマスク・パターンを生成するための代表的な圧縮のアルゴリズムを紹介する。

シンボリック図の圧縮は、レイアウトの要素の中で伸縮可能な配線を伸縮することにより行われる。このとき、縦方向と横方向の両方を同時に行うことは困難であり、縦方向と横方向に分けて圧縮を行

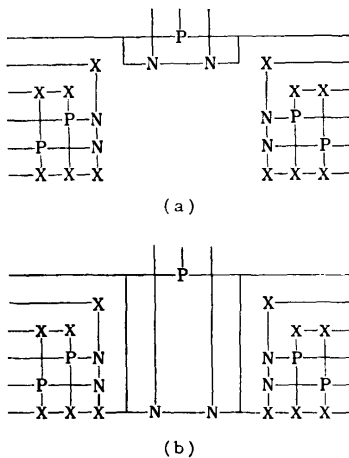


図-3 STICKS のアルゴリズムによる圧縮の例

い、それらを繰り返すことで最終的なマスク・パターンを生成する点がどのアルゴリズムでも共通している。

4.1 STICKS のアルゴリズム<sup>1)</sup>

STICKS のアルゴリズムは、レイアウトの各要素を設計ルールに従って最小間隔で順に積めることにより圧縮を行う。縦方向の圧縮は水平方向の要素（水平方向の配線、トランジスタ、コンタクトなど）を一番底にある要素から順に積み上げることにより行う。図-3(a)の例では、左右の一連のコンタクトと中央の2個のN形トランジスタが最初に配置される。次に、その上に位置する水平方向の要素が最小間隔で配置される。これを繰り返すことで図-3(b)のレイアウトが得られる。

4.2 Cabbage のアルゴリズム<sup>2)</sup>

Cabbage のアルゴリズムは有向グラフを利用してクリティカル・パス（レイアウトの圧縮限度を決める要素をたどる経路）を見つけ、これを基準に各要素を再配置することで圧縮を行う。以下、その手順を簡単に示す。

- (1) レイアウトの各要素を水平方向と垂直方向の座標に従って分類する。
- (2) 横方向の圧縮の際に水平方向の位置関係調べするのに必要な要素（水平成分）と縦方向の圧縮の際に垂直方向の位置関係を調べるのに必要な要素（垂直

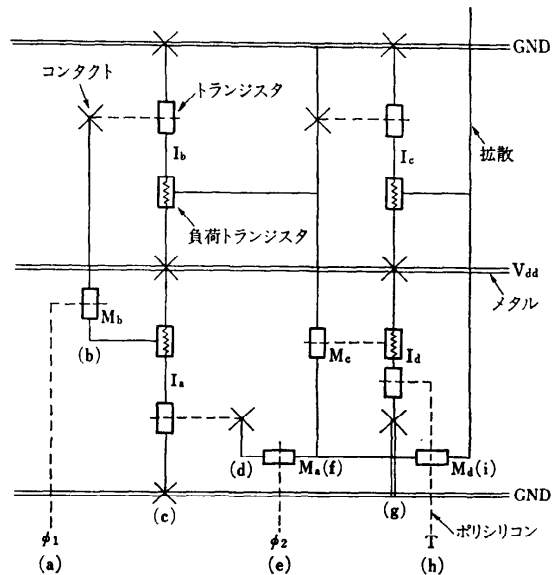


図-4 シンボリック図の例

成分)に分類する。

(3) 水平成分と垂直成分をそれぞれ中心座標、位置関係、電気的接続関係によりグループ分けする。図-4 の例では水平成分は(a)~(i)の9つのグループに分けられる。例えば、インパター  $I_a$  と  $I_b$  を構成する4つのトランジスタ、3つのコンタクト、それらをつなぐ拡散配線が1つのグループとなる。

(4) 各グループ間の最小間隔を設計ルールに従って計算する。このとき、各グループを構成する要素のマスク・パターンごとに直接的に位置関係を有するグ

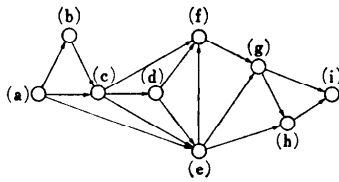


図-5 図-4 の回路の有向グラフ

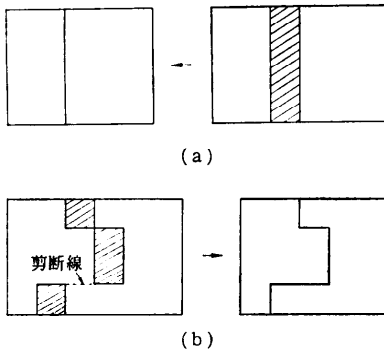


図-6 Compression Ridge の例

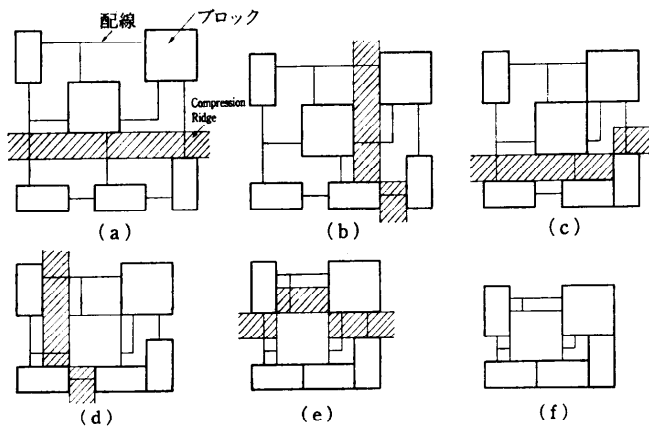


図-7 Compression Ridge による圧縮の例

ループ間について最小間隔を計算する。図-4 の例では、例えば、グループ a については、グループ b, c, e との最小間隔を計算する。

(5) 各グループの関係から有向グラフを作成する。グラフのノードがグループを表し、辺はその両端のグループに一定の間隔が必要であることを表す。図-5 は図-4 のレイアウトに対する有向グラフである。

(6) 有向グラフからレイアウトの寸法を決定するクリティカル・パスを求める。図-5 の例では、例えば、グループ (g) と (i) の間の最小間隔が、(g) と (h) 及び (h) と (i) の間の最小間隔の和より大きいと、(a)→(b)→(c)→(d)→(e)→(f)→(g)→(i) の経路がクリティカル・パスとして求まる。

(7) クリティカル・パス上で各グループを移動し、グループ間の距離が最小間隔になるように再配置する。

(3)~(7)の手順を縦方向、横方向について繰り返す、圧縮を行う。

### 4.3 Compression Ridge のアルゴリズム<sup>3),4)</sup>

Compression Ridge のアルゴリズムは、シンボリック図をそのままマスク・パターンに変換した後、このマスク・パターンから余分なスペースを取り除くことによって圧縮を行う。ここで、以下の3つの条件を満足する余分なスペースを Compression Ridge と呼び、これを縦方向、横方向について繰り返し取り除くアルゴリズムである。

(1) 対象となる図形の両端の辺(縦方向の圧縮の際には左右の辺、横方向の圧縮の際には上下の辺)に接している。このとき、切断線でつながれたものでもよい。図-6(a)は単純な場合、(b)は切断線を含む場合の例である。

(2) 水平な配線と重なっていない(垂直な配線とは重なってよい)。

(3) トランジスタ、コンタクトなど配線以外の要素と重なっていない。

図-7(a)~(f)は、Compression Ridge を縦方向、横方向について繰り返し取り除くことによって圧縮が行われる様子を示している。

### 4.4 圧縮のアルゴリズムの比較

前述の3つのアルゴリズムを比較すると、最終的なマスク・パターンの寸法は、STICKS と Cabbage に比べ Compression Ridge では多少大きい寸法となる。これ

は、Compression Ridge のアルゴリズムでは、マスク・パターンに変換した後余分なスペースを取り除くため、重ねて配線が可能な配線を重ねることができないためである。この傾向は、配線の多層化が進むと顕著になる。STICKS と Cabbage のアルゴリズムでは、処理量はアルゴリズムが単純な STICKS が少ないが、最終的なマスク・パターンの品質の点では、Cabbage のアルゴリズムが有利である。例えば、STICKS のアルゴリズムでは抵抗の高い拡散配線が無意味に伸ばされる場合があり、そのような場合には対話機能による修正が必要となる。一方、Cabbage のアルゴリズムでは、有向グラフを作成する際付加情報を加えることができるためクリティカル・パス上でのグループの再配置の際に最終マスク・パターンの品質をある程度考慮することができる。

## 5. む す び

本稿では、シンボリック・レイアウト設計手法について自動圧縮を中心に述べた。自動圧縮を行わないシンボリック・レイアウト設計手法は、数多くのシステムが開発され広く実用化されている。マスク・パターンの寸法は、直接人手でマスク・パターン図を作成する場合に比べ0~10%程度増加するが、設計期間は、1/2~1/4に短縮される。一方、自動圧縮を行うシンボリック設計手法も、多くのシステムが開発され実用化の域に達しつつある。マスク・パターンの寸法は同様に数~20%程度増加するが上記の場合に比べより設計期間の短縮が行える。また、異なる設計ルールに対しても、同一のシンボリック図から処理しなおすだけでマスク・パターンを作成することが可能である。圧縮のための計算時間は、レイアウトの要素数 $n$ に対してほぼ $n^{1.5}$ に比例する値となっており、実用的な適用範囲は数千要素程度までであるが、階層的な手法を用いれば大規模な回路にも適用できる。

今後、自動配置配線手法の適用が困難な分野だけでなく、自動配置配線手法で用いられる大規模セルのレイアウト設計等広い適用が期待される。

## 参 考 文 献

- 1) Williams, J. D.: STICKS—A Graphical Compiler for High Level LSI Design, Proc. of 1978 National Computer Conference, pp. 289-295 (June 1978).
- 2) Cho, Y. E., Korenjack, A. J. and Stockton, D. E.: FLOSS: An Approach to Automated

- Layout for High-Volume Design, Proc. of 14th Design Automation Conference, pp. 138-141 (June 1977).
- 3) Dunlop, A. E.: SLIP: Symbolic Layout of Integrated Circuits with Compaction, Computer-Aided Design, Vol. 10, No. 6, pp. 387-391 (Nov. 1978).
- 4) Idem: SLIM—The Translation of Symbolic Layouts into Mask Data, Proc. of 17th Design Automation Conference, pp. 595-602 (June 1980).
- 5) Hsueh, M. Y. and Pederson, D. O.: Computer-Aided Layout of LSI Circuit Building-Blocks, Proc. of International Symposium on Circuits and Systems, pp. 474-477 (July 1979).
- 6) Weste, N.: Virtual Grid Symbolic Layout, Proc. of 18th Design Automation Conference, pp. 225-233 (June 1981).
- 7) Larsen, R. P.: Versatile Mask Generation Techniques for Custom Microelectronic Device, Proc. of 15th Design Automation Conference, pp. 193-198 (June 1978).
- 8) Gibson, D. and Nance, S.: SLIC—Symbolic Layout of Integrated Circuits, Proc. of 13th Design Automation Conference, pp. 434-440 (June 1976).
- 9) Idem: Symbolic System for Circuit Layout and Checking, Proc. of International Symposium on Circuits and Systems, pp. 436-440 (Apr. 1977).
- 10) Clary, D., Kirk, R. and Sapiro, S.: SIDS—A Symbolic Interactive Design Systems, Proc. of 17th Design Automation Conference, pp. 292-295 (June 1980).
- 11) Daram, B.: PLANAR Layout System, Proc. of COMPCON, Spring 78, pp. 136-138 (Feb. 1978).
- 12) Infante, B., Brocken, D., McCalla, B., Yamayoshi, S. and Cohen, E.: An Interactive Graphics System for the Design of Integrated Circuits, *ibid.*, pp. 182-187 (June 1978).
- 13) Jennings, R. M.: ICDS: A Symbolic Layout and Mask Verification System for VLSI, Proc. of Custom Integrated Circuits Conference, pp. 101-102 (May 1980).
- 14) Cranswick, A. P., Bather, K.: MICROCELL—A User Orientated Custom Design System for VLSI, Proc. of International Conference on Circuits and Computers, pp. 590-593 (Oct. 1980).
- 15) Hardage, K.: ASAP: Advanced Symbolic Artwork Preparation, LAMBDA, Fourth Quarter, pp. 32-37 (1980).
- 16) Lopez, A. D. and Low, H. S.: A Dense Gate Matrix Layout Method for MOS VLSI, IEEE

- Trans. on Electron Devices, Vol. ED-27, No. 8, pp. 1671-1675 (1980).
- 17) Hage, C. J. and Krueger, J. C.: An Efficient Symbolic Layout and Design Rule Checking System for Gate Arrays, Proc. of International Conference on Circuits and Computers, pp. 1040-1043 (Oct. 1980).
- 18) Weste, N. H.: MULGA—An Interactive Symbolic Layout System for the Design of Integrated Circuits, The Bell System Technical Journal, pp. 823-857 (July 1981).

(昭和 59 年 8 月 31 日 受付)

---