

Title	VLSIの自動レイアウト設計に関する研究
Author(s)	寺井, 正幸
Citation	大阪大学, 1988, 博士論文
Version Type	VoR
URL	<a href="https://hdl.handle.net/11094/2237">https://hdl.handle.net/11094/2237</a>
rights	
Note	

*Osaka University Knowledge Archive : OUKA*

<https://ir.library.osaka-u.ac.jp/>

Osaka University

VLSIの自動レイアウト設計  
に関する研究

1988年

寺 井 正 幸

## 内 容 梗 概

本論文は、筆者が昭和53年三菱電機株式会社に入社以来、LSI研究所及びカスタムLSI設計技術開発センターで行って来たVLSIの自動レイアウト設計に関する研究成果をまとめたものである。本研究は、VLSI用の高性能の自動レイアウト設計システムを構築することを目的としたもので、セミカスタムVLSIに対しては、種々の相異なるチップ構造を持つゲートアレイ向きの配線手法とトランジスタアレイ向きのレイアウト自動生成手法を中心に、フルカスタムVLSIに対しては、マクロセルのレイアウト設計のためのシンボリックレイアウト手法を中心に考察するものである。本論文は、全9章から構成される。

第1章では、VLSIの自動レイアウト設計に関するこれまでの研究と課題について述べ、本研究の目的を明らかにすると共に、研究内容と成果について概説する。

第2章では、バイポーラやMOS等のデバイス技術を用いて実現された種々の相異なるチップ構造を持つゲートアレイ向き配線手法について考察する。まず、ゲートアレイのチップ構造について述べ、次に配線手法の適用対象となるチップモデル、配線径路を記述するデータ構造、及び配線手法について述べる。更に、この配線手法の適用結果について述べ、本手法は特定のチップ構造を持つゲートアレイ専用の配線手法と比べ、性能が劣らないことを示す。

第3章では、ゲートアレイの配線設計におけるグローバル配線の手法について考察する。ここでは、配線の迂回が少なくなるように、グローバル配線における信号線の処理順序を決定し、これに基づき配線の混雑度を均一化するようにグローバル配線を行う算法を考察する。本算法は、配線の迂回を減少させ、配線の局所的混雑を緩和して配線不能が生じる可能性を低くするという点で、有効であることを示す。

第4章では、ゲートアレイの配線設計における端子割当の手法について考察する。矩形の水平チャネル（配線領域）内の配線において、幹線（配線径路の水平方向成分）間のサイクリックな上下制約がある時、配線が困難になることがよく知られている。端子割当問題は、このような幹線間のサイクリックな上下制約を除去するために、信号線を論理的に等価な端子間でどのように割り当てるかを決定する問題である。この問題に対し、幹線間のサイクリックな上下制約の数を正確に数えながら端子割当を行う算法を提案する。適用結果により本章で提案する算法が有効であることを示す。

第5章では、ゲートアレイの詳細配線設計における幹線分割の手法について考察する。チャンネル内の配線において、幹線間の上下制約の連鎖がある場合、幹線の折り曲げ（ドッグレッグ）により配線に使われる水平トラック（水平方向配線格子線）数を減少させることができる。幹線分割問題とは、チャンネル内の配線において使用される水平トラック数を減少させるために幹線の折り曲げ位置を決定する問題である。この問題に対し、幹線間の上下制約の連鎖の長さが増加せず、かつ配線を行うための水平トラック数に関して定義されるある関数 $W$ の値が減少するように、幹線分割を行う算法を提案する。適用結果から本章で提案する算法が有効であることを示す。

第6章では、ゲートアレイのセル配置が定まった段階において、チップ全体の配線に必要な水平トラック数を予測する問題について考察する。まず、各信号線の配線の幹線分割回数に制限を加えた時、 $N$ セル列間の配線に必要なかつ十分な水平トラック数がある範囲内にあることを理論的に証明し、次に自動配線プログラムで実際に配線したときに使用される水平トラック数がほぼ上記の範囲の上限に等しいことを示す。

第7章では、トランジスタアレイのレイアウト手法について考察する。まず、CMOSトランジスタペアが敷き詰められたトランジスタアレイの構造を述べ、次いでRAM/ROMを含むCMOS VLSIを高密度に設計するためのレイアウト手法を考察する。更に、適用結果から、この手法がRAM/ROMを含むVLSIに対して特に有効であることを示す。

第8章では、フルカスタムVLSIのマクロセル（機能ブロック）のトランジスタ単位でのレイアウト設計を行うためのシンボリックレイアウト手法について考察する。シンボリックレイアウト手法とは、トランジスタ、コンタクト、配線などのレイアウト設計の基本要素に対してシンボルを定め、これらのシンボルの相対的な位置関係によってレイアウトを表現し、それらの間のコンパクションを行って最終的なマスクパターンを生成する手法である。本章では、マクロセルのレイアウト設計を短期間に、かつ高密度に行うことができ、しかもレイアウト結果が製造プロセスの進歩に伴うパターン設計ルールの変更に充分追従できるようなシンボルの定義方法とシンボルの配置手法、及びコンパクション手法について考察する。適用結果から本章で提案する手法が有効であることを示す。

第9章では、本研究で得られた成果を要約し、今後に残された課題について述べる。

# VLSIの自動レイアウト設計に関する研究

## 目 次

第1章	緒 論	1
1.1	VLSIの自動レイアウト設計に関する研究の概要	1
1.2	研究の目的と本論文の構成	3
第2章	相異なるチップ構造を持つゲートアレイ向き配線手法	7
2.1	緒 言	7
2.2	ゲートアレイのチップ構造	7
2.2.1	CMOSゲートアレイのチップ構造	8
2.2.2	バイポーラゲートアレイのチップ構造	8
2.3	チップモデル	11
2.3.1	チップモデルに関する考察	11
2.3.2	チップモデル	12
2.4	入力データと配線径路データ	13
2.4.1	入力データ	14
2.4.2	配線径路データ	14
2.5	配線手法	15
2.5.1	グローバル配線	15
2.5.2	端子割当, 幹線分割とチャンネル配線	18
2.6	実験結果	18
2.7	結 言	25
第3章	グローバル配線手法	27
3.1	緒 言	27
3.2	グローバル配線の算法	27
3.3	評価関数 $\Delta L(W_i)$ と $X(P_{ij})$ の意味	29
3.4	実験結果と考察	31
3.5	結 言	35
第4章	端子割当手法	37
4.1	緒 言	37
4.2	問題の定式化	37
4.3	端子割当の算法	41

4.4	算法の高速化	43
4.5	実験結果と考察	48
4.6	結 言	51
第5章	幹線分割手法	53
5.1	緒 言	53
5.2	問題の定式化と関数 $W(T)$ の定義	54
5.3	幹線分割の算法	59
5.4	算法の高速化	61
5.5	実験結果と考察	66
5.6	結 言	69
第6章	ゲートアレイにおける水平トラックの必要総数の予測問題	71
6.1	緒 言	71
6.2	問題の定式化	71
6.3	配線要求と最大交さ数	73
6.4	配線に必要な水平トラックの総数の予測	76
6.5	実験結果と考察	82
6.6	結 言	85
第7章	トランジスタアレイにおけるレイアウト手法	87
7.1	緒 言	87
7.2	チップ構造	88
7.3	レイアウトモデル	89
7.3.1	レイアウトモデルに対する考察	89
7.3.2	レイアウトモデル	91
7.4	レイアウトの手法	92
7.5	ROM/RAMの構成方法	93
7.6	実験結果と考察	95
7.7	結 言	96
第8章	シンボリックレイアウト手法と評価	99
8.1	緒 言	99
8.2	シンボリックレイアウトの概要	99
8.3	シンボリックレイアウト図におけるシンボル配置手法	101
8.4	コンパクションの手法	104
8.5	適用結果と性能改善	110

8.5.1	16ビットマイクロコントローラの論理構成	110
8.5.2	16ビットマイクロコントローラのレイアウト設計結果	111
8.5.3	性能改善	116
8.6	結 言	122
第9章	結 論	124
謝 辞		127
研究業績目録		128

## 第1章 緒 論

本章ではVLSIの自動レイアウト設計に関するこれまでの研究と課題について述べ、本研究の目的を明らかにすると共に、研究内容と成果について概説する。

### 1.1 VLSIの自動レイアウト設計に関する研究の概要

微細加工技術の進歩によって、VLSIの集積度は毎年1.5～2倍の割合で上昇を続けており、近い将来には、100万素子を超える論理VLSIの出現が確実視されている。カスタムVLSIに対しては、その集積度の増大と多品種少量化とに伴って、高性能の自動レイアウト設計（パターン設計）技術の要求がますます高まりつつある。カスタムVLSIは、表1.1のようにレイアウト設計の専用化の度合によってセミカスタムVLSIとフルカスタムVLSIに大別される。

セミカスタムVLSIの内、最も広く用いられているゲートアレイ方式では、必要とする素子を組み込み拡散工程迄を完了したウェハ（ウェハ上の個々のチップをマスタスライスと呼ぶ）を予め作っておき、素子間の配線のマスクパターンのみを品種毎にレイアウト設計し、ウェハ上に配線パターンを形成してVLSIを作成する。拡散工程迄を完了したウェハは全品種に対し共通であるため、短期間に安価に設計製造できる利点がある。ゲートアレイのチップ上には、図1.1のようにトランジスタまたはゲートの列（セル列）と矩形の配線領域（配線チャンネル）が交互に並んでいる。ゲートアレイのレイアウト設計において使われる既設計の基本回路パターンを論理セルと呼ぶ。論理設計に従い必要な全論理セルのセル列上での位置を決めるのが配置設計であり、論理セルの信号端子間を結ぶ配線パターンを生成するのが配線設計である。ゲートアレイでは、自動配置配線が主要な設計自動化技術となる。ゲートアレイの配置配線プログラムは多数発表されている（例えば、文献(1)、(2)の配置配線プログラム）。

最近、更に高集積化を達成するためトランジスタをVLSIチップ全面に敷き詰めた方式のゲートアレイ（トランジスタアレイ）が提案されている<sup>(3)、(4)</sup>。トランジスタアレイにおいても、自動配置配線が主要な設計自動化技術となるが、これに対する配置配線プログラムについて記述した文献は少ない<sup>(5)</sup>。

一方、フルカスタムVLSIは、製造プロセスに必要な全マスクパターンを専用化して設計するもので、高性能、高集積のVLSIが実現できる。このフルカスタムVLSIにおけるマクロセル（機能ブロック）のレイアウト設計におい



表1.1 カスタムVLSI設計方式とCMOS製造プロセス

CMOS製造プロセス		カスタムVLSI設計方式	
		セミカスタム設計方式 (ゲートアレイ、 トランジスタアレイ)	フルカスタム設計方式
トランジスタ 構築工程	アイランド フィールド アイソレーション チャネルドープ ポリシリコン ソースドレイン コンタクト	共通マスク (全品種で共通)	専用マスク (品種毎に異なる)
配線工程	第1層アルミ配線 スルーホール 第2層アルミ配線		

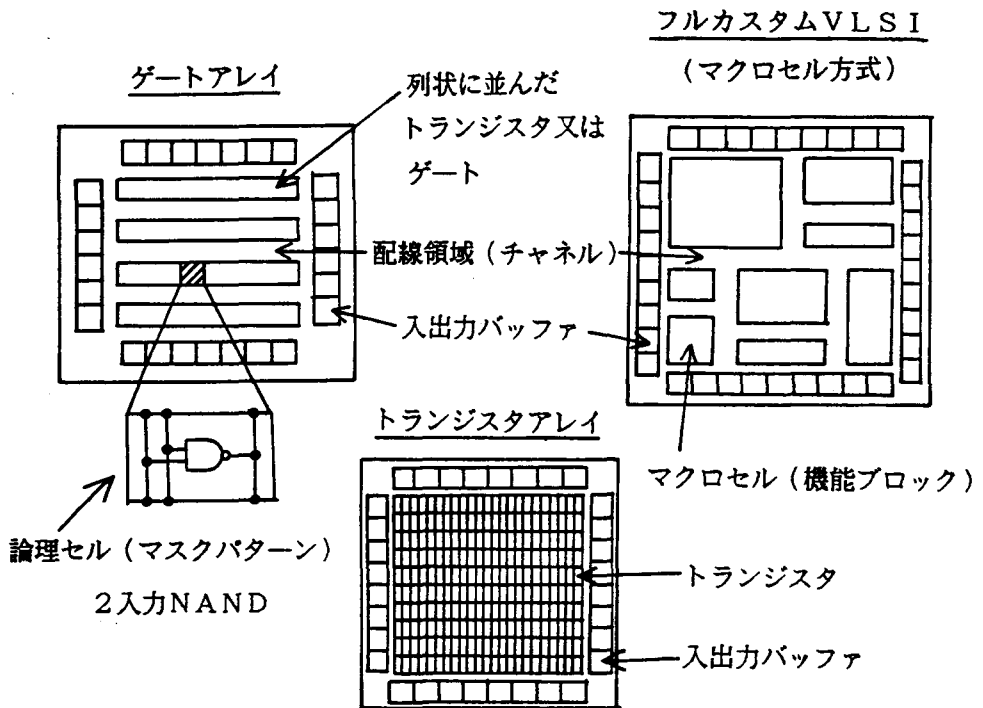


図1.1 カスタムVLSIのチップ構造

て、設計品質の向上に極めて大きく寄与するのがシンボリックレイアウト手法である。シンボリックレイアウト手法とは、トランジスタ、コンタクト、配線などのレイアウト設計の基本要素に対してシンボルを定め、これらのシンボルの相対的な位置関係によってレイアウトを表現し、それらの間のコンパクションを行って最終的なマスクパターンを生成する手法である。シンボリックレイアウトシステムとしては、STICKS<sup>(6)</sup>、CABBAGE<sup>(7)</sup>、SLIM<sup>(8)</sup>等が著名である。

## 1.2 研究の目的と本論文の構成

VLSI用の高性能な自動レイアウト設計システムを構築するために、ゲートアレイの配線手法、トランジスタアレイのレイアウト手法やシンボリックレイアウト手法に対し特に下記のような問題点を解決することが必要である。

(1) ゲートアレイの配線手法はデバイス依存性が強く、各デバイス専用の配線手法が多い。従って、従来の配線手法はチップ構造が相異なるゲートアレイに適用することは困難である。

(2) ゲートアレイでは、配線領域の大きさがあらかじめ固定されているため、配線設計において配線不能信号が発生することが多い。このため、高い集積度が達成できない。

(3) トランジスタアレイにおいて、RAM/ROMを含む回路を全自動で、高密度にレイアウト設計することは難しい。

(4) シンボリックレイアウト手法によりマクロセルを短期間にかつ高密度にレイアウト設計し、しかもレイアウト結果がパターン設計ルールの微細化と共に容易に更新できるようにすることは難しい。

そこで、本研究ではゲートアレイの配線手法、トランジスタアレイのレイアウト手法やシンボリックレイアウト手法に対し独自の新しい手法を考案して上記の問題点を解決し、VLSI用の高性能な自動レイアウト設計システムを構築することを目標とする。本研究では、まずゲートアレイに対して、種々の相異なるチップ構造を持つゲートアレイ向きで、高い配線率を達成するための配線手法を考案し、これを実際のゲートアレイの配線設計に適用してその有効性を確認した。次に、トランジスタアレイに対して、RAM/ROMを含む回路の高密度なレイアウトを自動生成する手法を考案し、実際の適用結果からその有効性を確認した。一方、フルカスタムVLSIに対しては、マクロセルのレイアウト設計を短期間に、高密度に、かつパターン設計ルールと独立に行うためのシンボリックレイアウト手法を考案し、実際のVLSIのレイアウト設計に適用してその有効性

を確認した。

ここで章を追って具体的に本論文の内容を説明する。

第2章では、バイポーラやMOS等のデバイス技術を用いて実現された種々の相異なるチップ構造を持つゲートアレイ向き配線手法について考察する。まず、ゲートアレイのチップ構造について述べ、次に配線手法の適用対象となるチップモデル、配線径路を記述するデータ構造、及び配線手法について述べる。更に、この配線手法の適用結果について述べ、本手法は特定のチップ構造を持つゲートアレイ専用の配線手法と比べ、性能が劣らないことを示す。

第3章では、ゲートアレイの配線設計におけるグローバル配線の手法について考察する。ここでは、配線の迂回が少なくなるように、グローバル配線における信号線の処理順序を決定し、これに基づき配線の混雑度を均一化するようにグローバル配線を行う算法を考察する。本算法は、配線の迂回を減少させ、配線の局所的混雑を緩和して配線不能が生じる可能性を低くするという点で、有効であることを示す。

第4章では、ゲートアレイの配線設計における端子割当の手法について考察する。矩形の水平チャネル（配線領域）内の配線において、幹線（配線径路の水平方向成分）間のサイクリックな上下制約がある時、配線が困難になることがよく知られている。端子割当問題は、このような幹線間のサイクリックな上下制約を除去するために、信号線を論理的に等価な端子間でどのように割り当てるかを決定する問題である。この問題に対し、まず初期端子割当を行い、次に幹線間のサイクリックな上下制約の数を正確に数えながら端子割当を改善することにより、最終的な端子割当を決定する算法を提案する。更に、この算法の高速化のために、端子割当の変更によりサイクリックな上下制約の数が減少するか否かの判定が一部の幹線間の上下制約に着目するだけで可能であることを理論的に証明する。適用結果により本章で提案する算法が有効であることを示す。

第5章では、ゲートアレイの詳細配線設計における幹線分割の手法について考察する。チャネル内の配線において、幹線間の上下制約の連鎖がある場合、幹線の折り曲げ（ドッグレッグ）により配線に使われる水平トラック（水平方向配線格子線）数を減少させることができる。幹線分割問題とは、チャネル内の配線において使用される水平トラック数を減少させるために幹線の折り曲げ位置を決定する問題である。この問題に対し、幹線間の上下制約の連鎖の長さが増加せず、かつ配線を行うための水平トラック数に関して定義されるある関数 $W$ の値が減少するように、幹線分割を行う算法を提案する。更に、この算法の高速化のため

に、幹線分割により幹線間の上下制約の連鎖の長さが増加しないための必要十分条件と、関数 $W$ の値の計算方法について理論的考察を行う。適用結果から本章で提案する算法が有効であることを示す。

第6章では、ゲートアレイのセル配置が定まった段階において、チップ全体の配線に必要な水平トラック数を予測する問題について考察する。まず、各信号線の配線の幹線分割回数に制限を加えた時、 $N$ セル列間の配線に必要なかつ十分な水平トラック数がある範囲内にあることを理論的に証明し、次に自動配線プログラムで実際に配線したときに使用される水平トラック数がほぼ上記の範囲の上限に等しいことを示す。

第7章では、トランジスタアレイのレイアウト手法について考察する。まず、CMOSトランジスタペアが敷き詰められたトランジスタアレイの構造を述べ、次いでRAM/ROMを含むCMOS VLSIを高密度に設計するためのレイアウト手法を考察する。更に、適用結果から本手法がRAM/ROMを含むVLSIに対して特に有効であることを示す。

第8章では、フルカスタムVLSIやマイクロプロセッサ等のマクロセル（機能ブロック）のトランジスタ単位でのレイアウト設計を行うためのシンボリックレイアウト手法について考察する。本章では、マクロセルのレイアウト設計を短期間に、かつ高密度に行うことができ、しかもレイアウト結果が製造プロセスの進歩に伴うパターン設計ルールの変更に充分追従できるようなシンボルの定義方法とシンボルの配置手法、及びコンパクションの手法について考察する。更に、このシンボリックレイアウト手法を実際に16ビットマイクロコントローラのレイアウト設計に適用した結果を述べ、本手法が有効であることを示す。

第9章では、本研究で得られた成果を要約し、今後に残された課題について述べる。

### 参考文献

- (1) K.A.Chen, M.Feuer, K.H.Khokhani, N.Nan and S.Schmidt: "The chip layout problem: an automatic wiring procedure", Proc. 14th DA Conf., pp. 298-302(1977).
- (2) H.Shiraishi and F.Hirose: "Efficient placement and routing techniques for master slice LSI", Proc. 17th DA Conf., pp. 458-464 (1980).
- (3) H.Fukuda, H.Yoshimura and T.Adachi: "A CMOS pair-transistor array

- masterslice", 1982 Symposium on VLSI Technology, pp. 16-17(1982).
- (4) Y.Kuramitsu, K.Sato, Y.Akasaka and I.Ohkura: "A 540k-transistor CMOS variable-track masterslice", Proc. 1986 CICC, pp. 572-575(1986).
  - (5) C.P.Hsu, R.A.Perry, S.C.Evans, J.Tang and J.Y.Liu: "Automatic layout of channelless gate array", Proc. 1986 CICC, pp. 281-284(1986).
  - (6) J.D.Williams: "STICKS-A graphical compiler for high level LSI design", Proc. 1978 Nat. Comput. Conf., pp. 289-295(1978).
  - (7) M.Y.Hsueh and D.O.Pederson: "Computer-aided layout of LSI circuit building-blocks", Proc. 1979 ISCAS, pp. 474-477(1979).
  - (8) A.E.Dunlop: "SLIM-The translation of symbolic layouts into mask data", Proc. 17th DA Conf., pp. 595-602(1980).

## 第2章 相異なるチップ構造を持つゲートアレイ向き配線手法

### 2.1 緒 言

ゲートアレイは動作速度の高速化，低消費電力化などの多様な要求に対応してバイポーラやMOSなどの種々のデバイス技術を用いて実現され，また，ワイヤボンダ技術やVLSIの入出力ピン数の増加に対応するためのフリップチップ技術を用いて実装されている。このため，種々の相異なるチップ構造を持つゲートアレイがある<sup>(1)~(5)</sup>。ゲートアレイの配線手法が多数報告されているが，このうち，ある特定のチップ構造を持つゲートアレイ向きの配線手法は，通常そのチップ構造の特徴を生かして効率よく配線するように作られている<sup>(6)・(7)</sup>。しかし，異なったチップ構造を持つ何種類かのゲートアレイに対しそれぞれ別の配線手法を考えるならば，これらを基に何種類かのプログラムを作成する必要がある，プログラムの作成作業量が膨大となる。一方，種々の相異なるチップ構造を持つゲートアレイ向きの配線手法は，その適用対象となるチップモデルが複雑になるため，特定のチップ構造を持つゲートアレイ向きの配線手法に比べて，それを基にプログラムを作成した時のステートメント数や使用される記憶領域が大きくなり，配線率や処理速度などの性能が劣ることがある。これらの欠点を除いた種々の相異なるチップ構造を持つゲートアレイ向きの配線手法を考案すれば，上記の問題は解決される。

本章では，まず種々の相異なるチップ構造を持つゲートアレイ向きの配線手法の適用対象となるチップモデルについて考察し，次に配線手法の性能を高めるための入力データや配線径路のためのデータ構造を述べる。更に，この配線手法を種々の相異なるチップ構造を持つゲートアレイに適用した結果と特定のチップ構造を持つゲートアレイ向きの配線手法<sup>(6)・(8)</sup>の適用結果とを比較し，本手法の有効性を示す。

### 2.2 ゲートアレイのチップ構造

ゲートアレイのチップ構造はそのデバイスや実装方法により異なる。まず，具体的な例として，大倉ら<sup>(1)</sup>が考案したセル構造のCMOSゲートアレイおよび中屋ら<sup>(2)</sup>が考案したセル構造のバイポーラゲートアレイに対し，ワイヤボンダ技術およびフリップチップ技術をそれぞれ適用した場合のチップ構造について注目する。

### 2.2.1 CMOSゲートアレイのチップ構造

CMOSゲートアレイのチップ構造を図2.1に示す。チップは周辺領域と内部領域に分けられる。周辺領域はボンディングパッドと入出力バッファ回路を含む入出力バッファセルからなり、内部領域はセル列と配線領域である水平チャンネルと垂直チャンネルからなる。配線は2層からなり、第1層の水平格子線上に水平配線、第2層の垂直格子線上に垂直配線を置き、これらをビアホールにより連結する。チップ中央には電源、グランド配線を置き、セル列へ給電する。セル列内にはPおよびN型トランジスタのペアが酸化膜で分離されることなく図2.2のように並んでおり、PおよびN型トランジスタの列を構成している。相連続する一つ以上のトランジスタペアをセル列内で結線することにより論理セルを構成する。このため、論理セルは論理ゲートやフリップフロップ等の機能をもつ種々の大きさのものがああり、これが配置配線設計時の実装単位となる。図2.2に3入力NANDゲートの論理セルの構成例を示す。論理セルは第1層の配線パターンだけから構成され、そのすべての端子は垂直格子線上に一つずつ存在する。このため、すべての端子は上下両方向から第2層の垂直配線と接続でき、また、論理セルとして使用されなかったトランジスタペア上や論理セル内の端子の存在しない垂直格子線上は貫通トラックとなり第2層の垂直配線が貫通できる。

### 2.2.2 バイポーラゲートアレイのチップ構造

バイポーラゲートアレイのチップ構造を図2.3に示す。チップはセル列と水平チャンネルおよび垂直チャンネルからなる。フリップチップ技術<sup>(9)</sup>を用いてチップを実装するため配線は3層からなり、第1層の水平格子線上に水平配線、第2層の垂直格子線上に垂直配線、第3層には電源、グランド配線およびフリップチップボンディングのための半田バンプと入出力バッファセル間の配線を置き、これらをビアホールにより連結する。第3層の配線パターンは品種によらず固定される。第3層に電源、グランド配線が置かれているチップ中央部分やレファレンス回路<sup>(2)</sup>の部分は垂直チャンネルとして用いられる。半田バンプは水平チャンネル上の第3層に設けられるため、半田バンプの設けられている水平チャンネルは他の水平チャンネルより幅が広がっている。セル列内にはゲートセルと入出力バッファセルが並んでいる。ゲートセルに種々の論理機能をもたせ、入力トランジスタの使用率を高めるため、ゲートセルは図2.4のように2ゲートのペアからなる。セル列内のすべてのゲートセルは同一の大きさで、これが配置配線設計時の実装単位となる。ゲートセルや入出力バッファセルのすべての端子は垂直格子線上に一

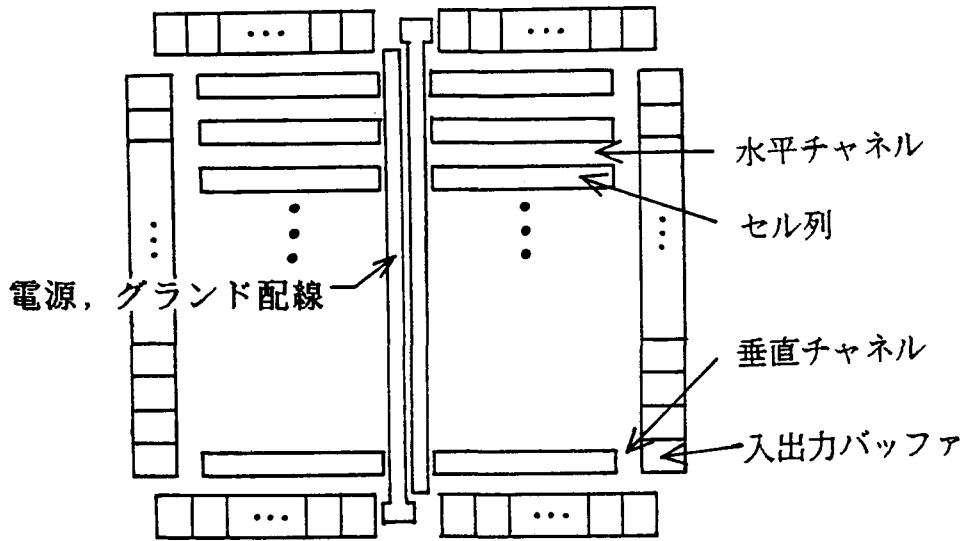


図2.1 CMOSゲートアレイのチップ構造

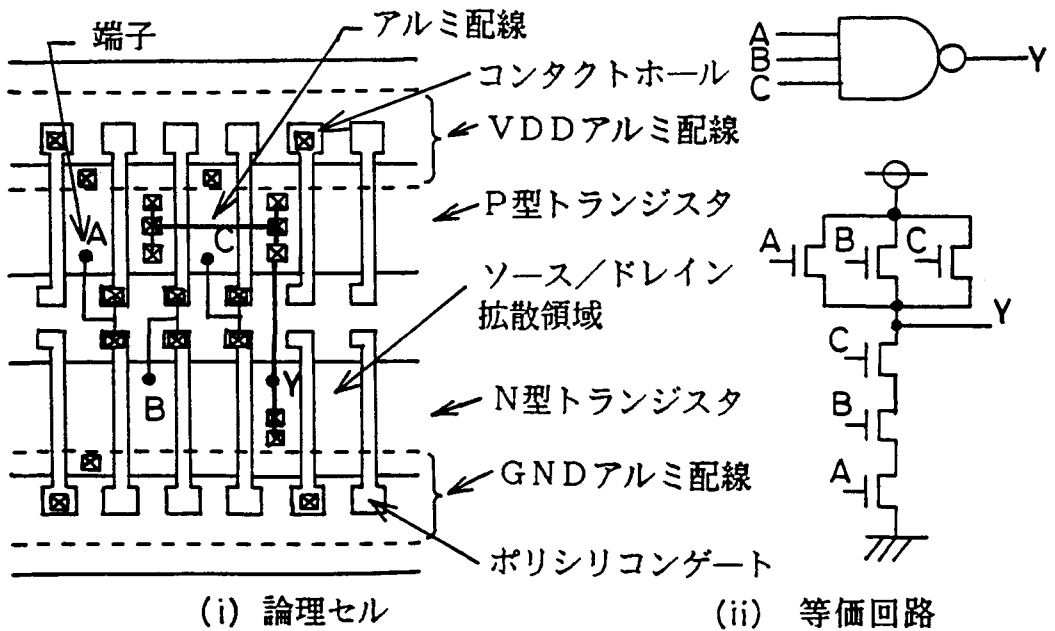
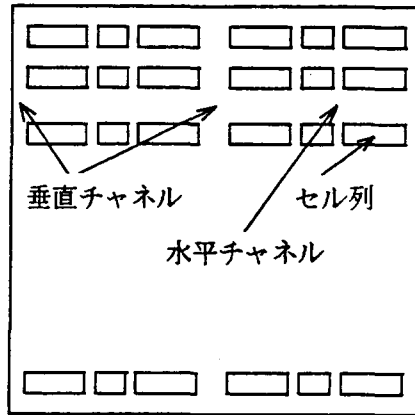


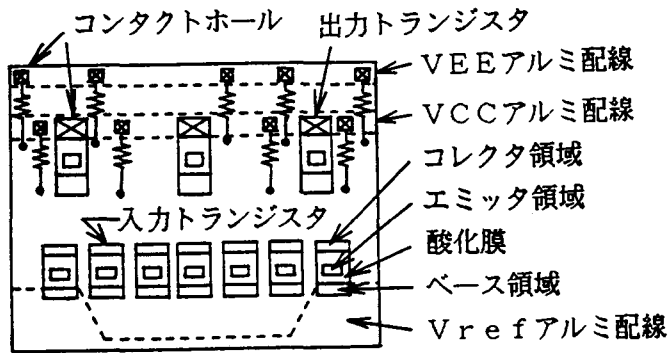
図2.2 CMOSゲートアレイの論理セルの例



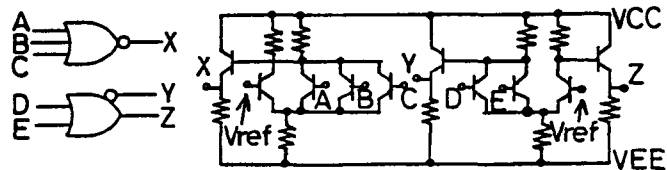


ゲートセルと入出力バッファセルがセル列内に並んでいる。

図2.3 バイポーラゲートアレイのチップ構造



(i) ゲートセル



(ii) 3入力NORと2入力OR/NORとして使われたゲートセルの等価回路

図2.4 バイポーラゲートアレイのゲートセル

つずつ存在するが、セル列内で第1層と第3層の電源、グランド配線がビアホールにより接続される。このため、ゲートセルや入出力バッファセルの端子は上下両方向から第2層の垂直配線と接続できる場合といずれか一方だけからしか接続できない場合があり、また、使用されなかったゲートセルや入出力バッファセル上およびゲートセルや入出力バッファセル内の端子の存在しない垂直格子線上を第2層の垂直配線が貫通できる場合とできない場合がある。

### 2.3 チップモデル

本節では、種々の相異なるチップ構造を持つゲートアレイ向きの配線手法の適用対象となるチップモデルについて述べる。この配線手法の性能が、特定のチップ構造を持つゲートアレイ向きの配線手法の性能に比べて劣らないようにするため、チップモデルをできるだけ単純化する。

#### 2.3.1 チップモデルに関する考察

2.2節に示したように、ゲートアレイはそのデバイスや実装技術により論理ゲートの機能をもつセルの形状や構成法、セル列や配線領域の形状、セルの端子と配線の接続方法が大幅に異なる。種々の相異なるチップ構造を持つゲートアレイ向きの配線手法の適用対象となるチップモデルを単純化するための考察を以下に行う。

配線手法において、種々の相異なるチップ構造のゲートアレイを扱うためには、チップ上の素子拡散パタンの繰返しのマスクレベルの最小単位（たとえばCMOSゲートアレイにおけるトランジスタベア、バイポーラゲートアレイにおけるゲートセルを指す）と配置配線設計時の実装上の最小単位（たとえばCMOSゲートアレイにおける論理セル、バイポーラゲートアレイにおけるゲートセルを指す）を区別することが必要である。したがって、本チップモデルにおいては前者を基本セル領域、後者をゲートセルと呼び区別し、ゲートセルは相連続する基本セル領域をセル列内で結線したもので、その論理機能により大きさが異なる。

次に、ゲートアレイではしばしばセル列が垂直チャネルや電源、グランド配線により分割される。セル列を分割する垂直チャネルがある場合、通常、水平チャネルと垂直チャネルがT字型に交差しているものとして扱うため、チャネルの配線順序が複雑となる<sup>(10)</sup>。セル列上の端子が垂直配線とだけ接続される場合には、セル列を分割する垂直チャネルをセル列内の貫通トラックの集合（貫通領域と呼ぶ）として扱えば、配線処理能力が低下せずチャネルの配線順序が単純とな

る。このため、本チップモデルではチップ上にはセル列が上から下へ一列に並び、セル列上の端子は垂直配線とだけ接続されるものとし、セル列内には基本セル領域のほかに、貫通領域、セル列を貫通する電源、グランド配線のためセル間の配線を置くことが禁止されている禁止領域を設ける。また、セル列内に入出力バッファセルを構成できるバッファセル領域も設ける。

配線設計の内のグローバル配線において、セル列内の各領域に面したチャンネル部分、チャンネルの交差点および貫通トラックをもつ領域をチャンネル要素と呼び、これを頂点对応させ、隣接する二つのチャンネル要素に対応する頂点对を辺で結んだグラフ（チャンネルグラフ）上で径路を探索する方法がよく用いられている<sup>(8)</sup>、<sup>(11)</sup>。この場合、グラフとチップモデルとの対応が単純でグラフの接続関係が規則正しいならば、この単純さを利用して効率よく径路探索できる<sup>(11)</sup>。そこで、本チップモデルではチップ上には凹凸のない同一の大きさのセル列が一列に並び、各セル列内の上記の領域の並ぶ順序はすべてのセル列内で同一であると制限する。このとき、グローバル配線におけるチャンネルグラフは後述のようにチップモデルとの対応が単純で接続関係も規則正しくなり、径路探索が効率よくできる。また、セル列が凹凸のないことから各チャンネルは凹凸のない長方形となり配線設計のうちのチャンネル配線が単純となる。

### 2.3.2 チップモデル

前節の考察の下に、配線手法の適用対象となるチップモデルを図2.5のように定める。すなわち、チップは周辺入出力バッファセルからなる周辺領域と同一の大きさの長方形のセル列と配線領域からなる内部領域に分けられる。配線領域はセル列間の水平チャンネルとチップ両脇にのみ許されている垂直チャンネルに分けられる。配線手法では、セルの端子間の配線を2層配線方式により行う。すなわち、第1層の水平格子線上に水平配線、第2層の垂直格子線上に垂直配線を置き、これらをビアホールにより連結する。各水平（垂直）チャンネル内の水平（垂直）格子線の数はチャンネルごとに異なるものとする。

各セル列内にはそれぞれ幾種類かの大きさをもつ基本セル領域、バッファセル領域、貫通領域および禁止領域がそれぞれ複数個並んでいるが、その並ぶ順序はすべてのセル列に対し同一でなければならない。ゲートセル（入出力バッファセル）は相連続する基本セル領域（一つのバッファセル領域）をセル列内で結線したもので、これが配置配線設計時の実装単位となる。ゲートセルはその論理機能によって種々の幅をもつ。ゲートセルや入出力バッファセルの端子は垂直格子線

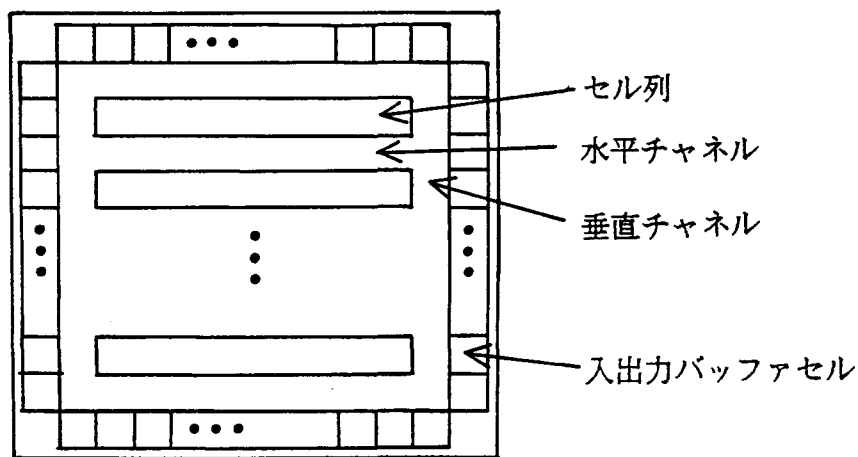


図2.5 チップモデル

上に一つずつ存在し、端子は第2層の垂直配線と上下両方向から接続できるかまたはいずれか一方方向だけからしか接続できないかを端子ごとに指定できる。ゲートセルやバッファセルとして使用されない基本セル領域上や入出力バッファセル領域上およびゲートセルや入出力バッファセル内の端子が存在しない垂直格子線を第2層の垂直配線が通過可能か否かを垂直格子線ごとに指定できる。これらは後述のセル構造データに記述される。

以上のようにチップモデルを定めることにより、この単純なチップモデルを扱う配線手法が種々の相異なるチップ構造を持つゲートアレイに適用可能となる。実際、2.2節で述べたCMOSゲートアレイやバイポーラゲートアレイのみならず、文献(3)、(4)のDSA MOSゲートアレイや文献(5)の3入力ECL回路を基本回路として用いたバイポーラゲートアレイに対しても、この配線手法は適用可能となる。

#### 2.4 入力データと配線経路データ

本節では、2.3.2節で述べたチップモデルを扱う配線手法の入力データと配線経路データについて述べる。本配線手法では配線を行いやすいように、これらの構造を工夫する。

### 2.4.1 入力データ

2.3.2節のチップモデルを扱う配線手法に入力するデータとして、対象とするゲートアレイのチップモデルを記述したデータ、チップ上のセル（ゲートセル，入出力バッファセルを総称してセルと呼ぶ）の端子間の論理接続データのほかにセル構造データを考える。

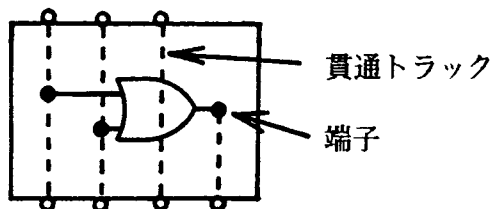


図2.6 ゲートセルの例

セル構造データでは、すべての種類の論理機能のセルに対し、セル内のすべての端子と貫通トラックのクラス名と位置を記述している。とくに、端子に対しては上下いずれの方向から垂直配線と接続できるかも記述している。ここで、セル内の全端子を論理的に等価な端子のクラスに分類し、各端子にそのクラス名を与えるものとする。また、すべての貫通トラックに同一のクラス名を与えるものとする。セル構造データの導入により本配線手法が種々の相異なるセル構造を持つゲートアレイを取扱い可能となる。

[例2.1] 図2.6のゲートセルは2入力ORの論理機能をもつ。二つの入力端子は論理的に等価であるためこれらに同一のクラス名 $C_1$ を与え、出力端子と貫通トラックにはそれぞれ別々のクラス名 $C_0$ 、 $C_F$ を与える。（例終）

次に、論理接続データでは各信号ネットにつながるセルの端子をクラス名で記述し、配線設計において、セル構造データを参照して同一クラス名の端子集合の中の最も適切な端子を信号ネットの端子として選ぶようにする。ここで、信号ネットとは等電位となるように結線すべきセルの端子の集合またはその配線である。このようにして、配線設計のうちの後述の端子割当を種々の相異なるセルに対しても効率よく行うことができ、高い配線率が期待される。

### 2.4.2 配線径路データ

効率よく配線を行うため、配線径路を記述するデータ構造として幹線テーブルと端子テーブルを考える。以下では、セル列の貫通トラックをその上の一つの格子点で代表させ、この格子点もゲートセルや入出力バッファセルの端子と同様に端子として扱う。また、水平（垂直）チャンネル内の各水平（垂直）配線を相連続する2端子間の水平（垂直）配線に分解したとき、そのそれぞれを2端子間の幹線と呼び、各垂直（水平）配線を支線と呼ぶ。チップ上のセル列数を $\gamma$ としたとき、 $\gamma+3$ 個の幹線テーブルと $\gamma$ 個の端子テーブルがある。各水平（垂直）チャ

ネルに対してそれぞれ一つの幹線テーブルがあり、各幹線テーブルには対応するチャンネル内のすべての2端子間の幹線に対し信号ネット識別名、幹線の置かれるトラックの番号および幹線の左右(上下)端点の水平(垂直)格子座標値を記述する。ここで水平(垂直)チャンネルにおける水平(垂直)格子線をトラックと呼ぶ。また、各セル列に対してそれぞれ一つの端子テーブルがあり、各端子テーブルには対応するセル列上のすべての端子に対し水平格子座標値、クラス名、端子に接続する支線の接続方向および信号ネット識別名を記述する。

[例2.2] 図2.6のゲートセル間を接続する信号ネット(識別名:  $\omega$ )の図2.7のような配線径路は幹線テーブルと端子テーブルに表2.1と表2.2のように記述される。

## 2.5 配線手法

本節では、2.3.2節のチップモデルを扱う配線手法において、グローバル配線と個々のチャンネル内での配線の2段階からなる配線手法<sup>(7)</sup>を用いた場合、幹線テーブルと端子テーブルに配線径路をどのように記述していくかを述べ、このデータ構造が上記の配線手法に適していることを示す。

配線手法のフローを図2.8に示す。入力処理では入力データを読み取り、チップ上のすべての端子に対し、端子テーブルに水平格子座標値とクラス名を記述する。ゲートセルおよび入出力バッファセルの端子に対しては、同一クラス名をもつ端子集合の中の各端子に信号ネット端子を仮に割り当て、端子に接続する支線の信号ネット識別名を端子テーブルに記述する。次にグローバル配線を行い、一番下の水平チャンネルから順に各水平チャンネルに対し、端子割当、幹線分割とチャンネル配線を行う。一番上の水平チャンネル内の配線を行った後に、チップ左右脇の2つの垂直チャンネル内の配線を水平チャンネル内の配線と同様に行う。以下では、グローバル配線、端子割当、幹線分割およびチャンネル配線について述べる。

### 2.5.1 グローバル配線

グローバル配線では、各信号ネットの配線径路を図2.9のグラフ  $G_c = [V, E]$  上に文献(8)の手法で割り当てる。ここで、 $V$ はチャンネル要素に対応する頂点集合で、 $E$ は隣接する二つのチャンネル要素に対応する頂点对を結ぶ辺集合である。次に、これらの径路をチップモデル上の幹線と支線に分け、すべての幹線に対し幹線テーブル上にトラックの番号を除く全項目を記述し、すべての支線に対し端子テーブル上に端子と支線との接続方向を、貫通トラック上の支線に対して

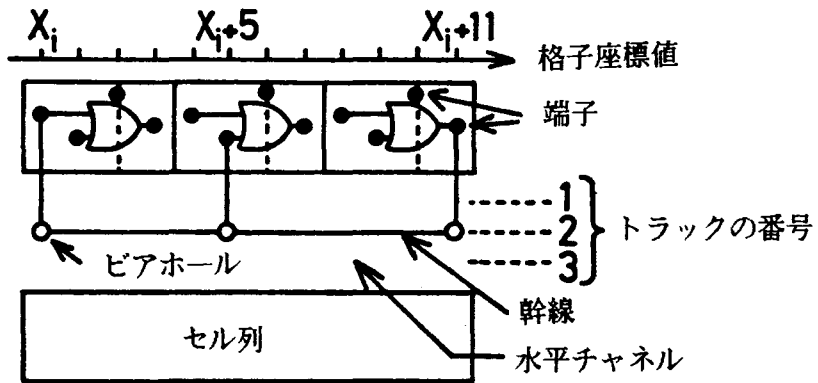


図2.7 信号ネット $\omega$ の配線径路

表2.1 図2.7 の水平チャネルに対する幹線テーブル

信号ネット 識別名	トラック の番号	左端点の水平 格子座標値	右端点の水平 格子座標値
$\omega$	2	$X_i$	$X_i + 5$
$\omega$	2	$X_i + 5$	$X_i + 11$

表2.2 図2.7 の上側のセル列に対する端子テーブル

水平格子座標値	端子の クラス名	支線との 接続方向	信号ネット 識別名
$X_i$	$C_1$	下方向	$\omega$
$X_i + 5$	$C_1$	下方向	$\omega$
$X_i + 11$	$C_0$	下方向	$\omega$

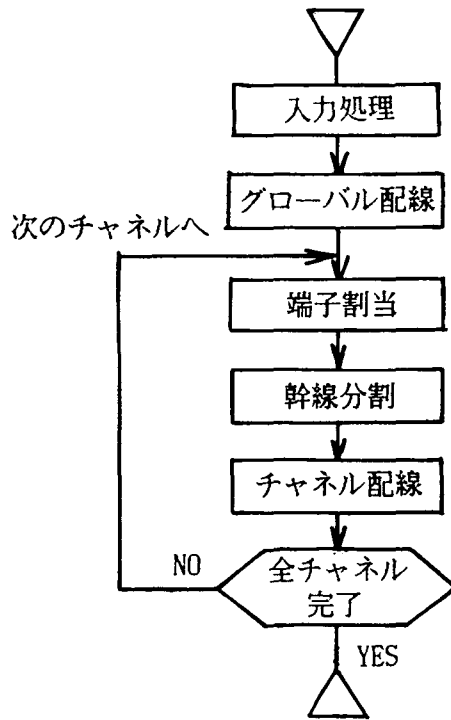


図2.8 配線手法のフロー図

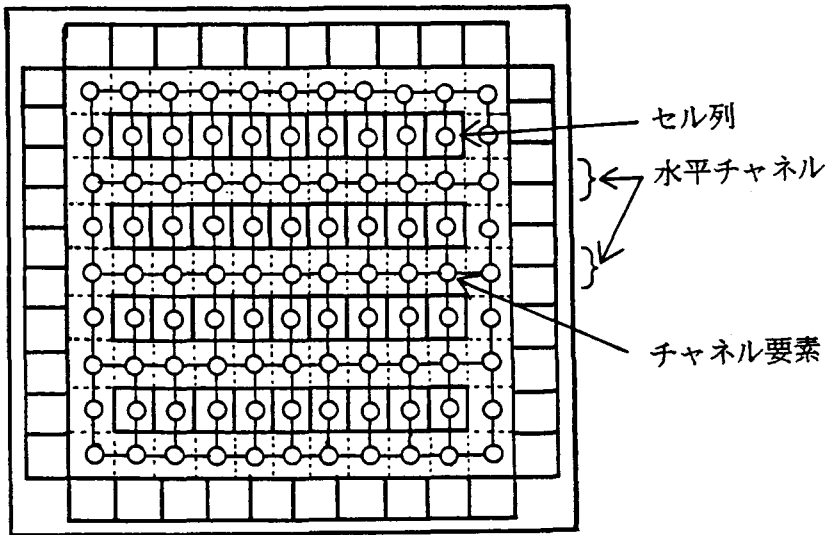


図2.9 グローバル配線におけるチャンネルグラフG。



はその信号ネット識別名も記述する。

### 2.5.2 端子割当，幹線分割とチャンネル配線

対向する2セル列間の水平チャンネルに対する幹線間の上下制約グラフ<sup>(6)</sup>、<sup>(12)</sup>にサイクルがあるときサイクルを解消しなければ配線できない。このため端子割当において，端子テーブル上でクラス名が等しくかつ同一ゲートセルまたは同一領域内にある2端子に対し，それらに接続する支線を置換することにより上下制約グラフ中のサイクルを除去する<sup>(12)</sup>。更に幹線分割では，配線に使われるトラック数を減少させるために，幹線テーブル上で1本の幹線を2本の幹線に分割する<sup>(13)</sup>。幹線分割では幹線の分割点だけを決め，幹線をどのトラック上に置くかはチャンネル配線で決定する。次にチャンネル配線では，“dogleg channel router”<sup>(14)</sup>と同様の手法を用いてチャンネル内のすべての幹線を，幹線間および支線間の重なりがないようにトラック上に配置する。チャンネル配線の結果は幹線テーブル上のトラックの番号の項目に記述する。

以上のように，端子テーブルは端子割当における支線の置換が容易な構造に，幹線テーブルは幹線分割およびチャンネル配線が容易な構造になっている。

## 2.6 実験結果

前節までに述べた手法に基づいて，2.3.2節に示したチップモデルを扱える配線プログラムGALOPを作成した。本節では，GALOPを2.2節で示したCMOSゲートアレイおよびバイポーラゲートアレイに適用した結果について述べる。次に，GALOPをDSA MOSゲートアレイ専用の配線プログラムMARS-IIと比較した実験結果について述べる。

CMOSゲートアレイでは，チップ上の垂直方向に並んだセル列数，トランジスタペア数，水平チャンネル内のトラックの総数はそれぞれ20, 8,000, 336である。バイポーラゲートアレイでは，チップ上の垂直方向に並んだセル列数，ゲートセル数，入出力バッファセル数，水平チャンネル内のトラックの総数はそれぞれ28, 1,008, 280, 482である。GALOPをこれらのゲートアレイに適用した結果を表2.3に示し，表2.3のデータAとCに対する配線結果のパターン図を図2.10と図2.11に示す。使用した計算機はMELCOM-COSMO900II (3.5 MIPS)である。表2.3のように，GALOPはチップ構造の大幅に異なるゲートアレイに対して，共に短い処理時間で高い配線率を達成した。これはチップモデルをできるだけ単純にし，配線径路を記述するデータを配線手法に適した構

表2.3 GALOPの配線結果

ゲートアレイ	CMOS ゲートアレイ		バイポーラ ゲートアレイ	
	A	B	C	D
論理接続データ 信号ネット数	1,385	952	1,276	1,487
使用されたゲートセル数	1,451	890	768	830
使用された 基本セル領域数	6,502	5,568	768	830
使用された 入出力バッファセル数	38	64	155	163
配線率 (%)	100	100	100	100
水平チャンネル内の使用 されたトラックの総数	293	286	402	435
処理時間 (分)	19.9	13.5	5.6	6.9

表2.4 GALOPとMARS-M IIの配線結果

論理接続データ	E		F	
	GALOP	MARS-M II	GALOP	MARS-M II
プログラム				
信号ネット数	615		635	
使用されたゲートセル数	616		614	
使用された入出力バッファセル数	66		64	
使用された記憶領域	750 kバイト	740 kバイト	714 kバイト	710 kバイト
配線率 (%)	100	100	100	100
水平チャンネル内の使用された トラックの総数	157	158	135	135
総配線長	53,173	55,355	41,655	43,116
ビアホール数	2,380	2,408	1,986	2,013
処理時間 (分)	2.3	3.0	1.7	2.9

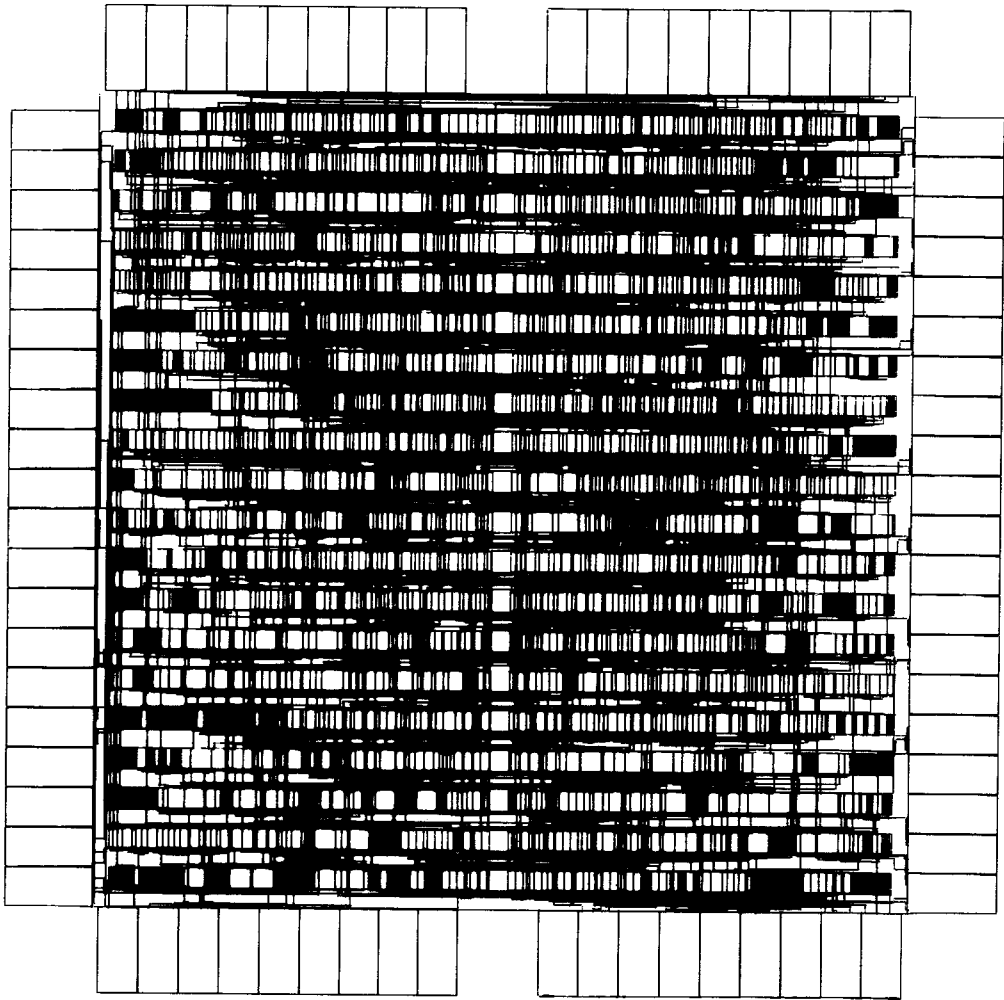


図2.10 データAの配線パターン図

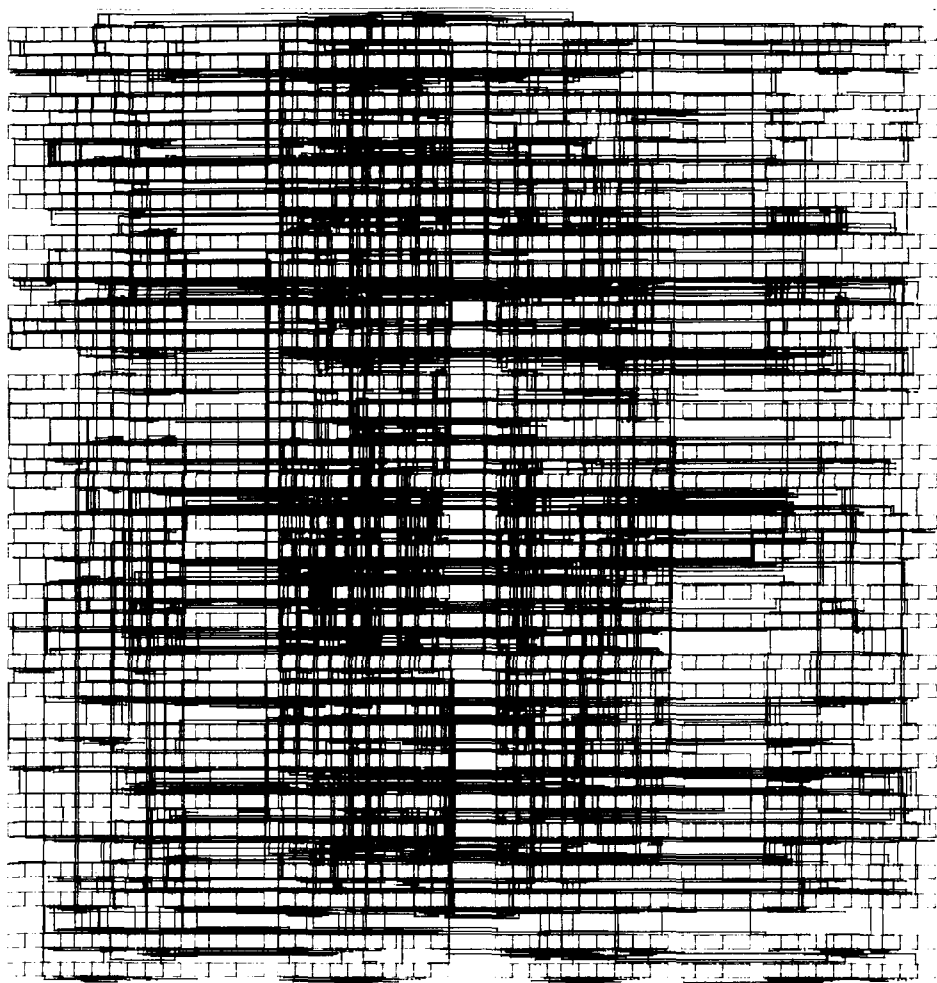


図2.11 データCの配線パターン図

造にして配線を効率よく行ったことによるものと考えられる。

次にGALOPとMARS-M IIを比較した結果について述べる。実験に用いたMARS-M IIの配線手法は性能を高めるため、文献(6)の配線手法に文献(8)のグローバル配線手法を組み込んだものである。GALOPとMARS-M IIはともにFORTRAN言語で記述され、MELCOM-COSMO900 IIで実行可能である。GALOPとMARS-M IIのプログラム規模はそれぞれ約22,000, 18,000ステートメントで、このプログラム規模の差は種々の相異なるセル構造を扱うため、GALOPの入力処理がMARS-M IIの入力処理より複雑になったことによる。

GALOPとMARS-M IIを、拡散自己整合形 (Diffusion Self-Aligned, DSA) MOSトランジスタを用いたDSA MOSゲートアレイ<sup>(3)・(4)</sup>に適用した結果を表2.4に示す。なお、DSA MOSゲートアレイのチップ構造の例を図2.12に示す。配線領域は2層で、セル列内には図2.13に示すゲートセルと1本の垂直配線を通すことができる貫通トラックが交互に並んでいる。ゲートセルは3入力NORゲート、2入力NORゲートまたはインバータとして用いられ、これが配置配線設計時の実装単位となる。また、ゲートセルの端子は上下両方向から第2層の垂直配線と接続でき、ゲートセルを2入力NOR又はインバータとして使用する場合や未使用の場合にはゲートセル内の端子の存在しない垂直格子線上を第2層の垂直配線が貫通できる。なお、チップ上の垂直方向に並んだセル列数、ゲートセル数、水平チャンネル内のトラックの総数はそれぞれ12, 624, 211である。表2.4より明らかのように、GALOPとMARS-M IIでは、プログラム実行時に使用される記憶領域がほぼ同程度であり、処理時間及び配線結果における配線率、使用トラック数、総配線長、ビアホール数等の配線処理能力もほぼ同等である。これは、GALOPでは2.3.2節に示した単純なチップモデル上で配線を行い、そのため配線径路を記述するデータ構造および配線手法をMARS-M IIとほぼ同様に単純にできたことによると考えられる。実際、たとえばチャンネル配線プログラムはMARS-M IIのものをそのままGALOPに用いることができた。なお、表2.4でのGALOPとMARS-M IIの処理時間の差は、配線径路データをGALOPでは主記憶装置上に、MARS-M IIでは外部記憶装置上にもつことによるものである。また、種々の相異なるセル構造を扱うためGALOPでのみ必要となるセル構造データは、DSA MOSゲートアレイの場合、インバータ、2入力NORおよび3入力NORの3種類の論理機能のゲートセルと周辺入出力バッファセルに対し、それぞれ端子と貫通トラッ

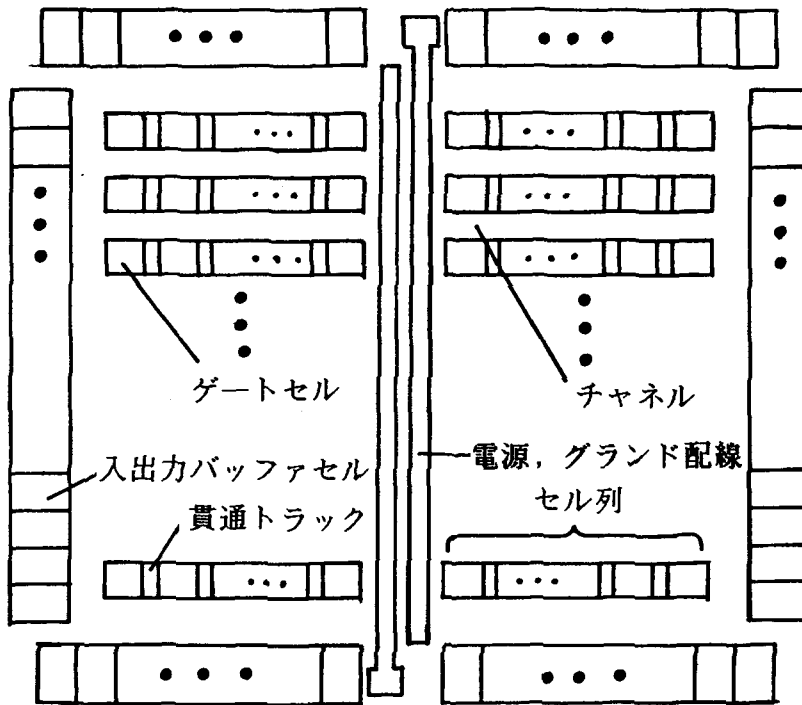
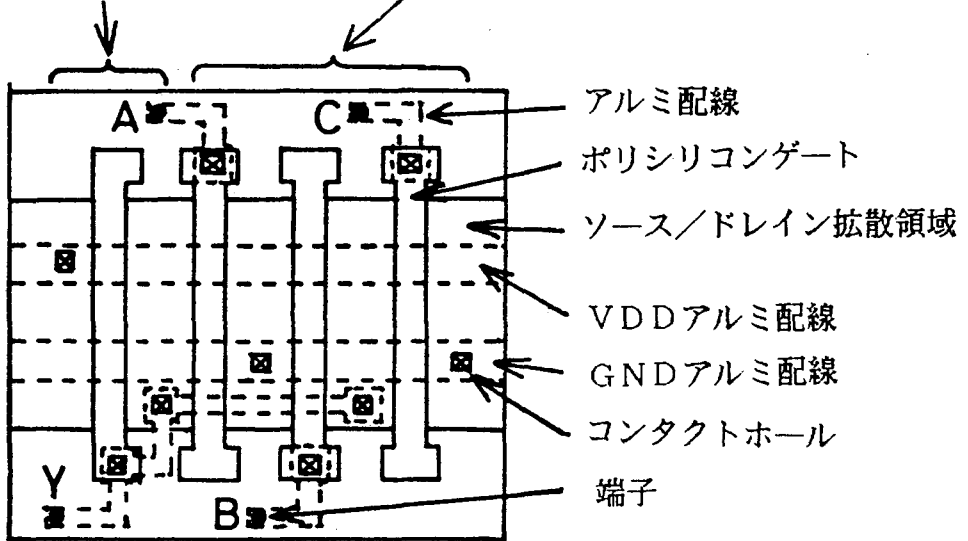
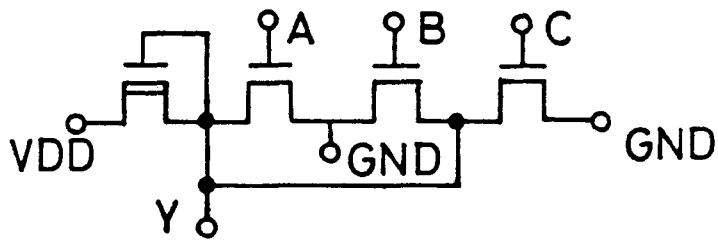


図2.12 DSA MOSゲートアレイのチップ構造

MOS負荷トランジスタ DSA MOSトランジスタ

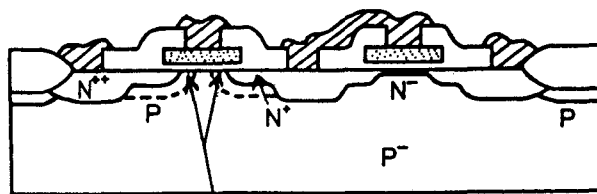


(a) ゲートセル



(b) 等価回路

DSA MOSトランジスタ MOS負荷トランジスタ



実効チャネル領域

(c) DSA MOSトランジスタの構造

図2.13 DSA MOSゲートアレイのゲートセル

クのクラス名と位置等を記述した高々400バイトの簡単なデータとなる。

以上の実験から、GALOPは種々の相異なるチップ構造のゲートアレイに適用可能であり、GALOPのプログラム規模や使用する記憶領域の大きさはMARS-M IIのそれらとほぼ等しく、かつ配線率や処理時間などの性能もほぼ同等であることを実証した。したがって、本章で述べた種々の相異なるチップ構造を持つゲートアレイ向きの配線手法は有効であることが確かめられた。

## 2.7 結 言

ゲートアレイはそのデバイスや実装方法により種々の相異なるチップ構造を持つ。本章では、これらの種々の相異なるチップ構造を持つゲートアレイ向きの配線手法について考察した。まず、種々の相異なるチップ構造を持つゲートアレイ向きの配線手法の適用対象となるチップモデルについて考察し、次に配線手法の性能を高めるため、その入力データや配線径路を記述するデータ構造について考察した。更に、この配線手法が実際のCMOS、バイポーラ及びDSA MOSのゲートアレイに適用可能であり、特定のチップ構造を持つゲートアレイ向きの配線手法と比較して性能が劣らないことを実証した。

なお、本章の内容は、文献(15)に基づいている。

### 参考文献

- (1) I. Ohkura, T. Noguchi, K. Sakashita, H. Ishida, T. Ichiyama and T. Enomoto: "Gate isolation—a novel basic cell configuration for CMOS gate arrays", Proc. 1982 CICC, pp. 307-310(1982).
- (2) M. Nakaya, S. Kato, K. Tsukamoto, H. Sakurai, T. Kondo and Y. Horiba: "A bipolar 2500-gate subnanosecond masterslice LSI", IEEE J. Solid-State Circuits, vol. SC-16, no. 5, pp. 558-562(1981).
- (3) O. Tomisawa, K. Anami, M. Nakaya, M. Ohmori, I. Ohkura and T. Nakano: "A 920 gate DSA MOS masterslice", IEEE J. Solid-State Circuits, vol. SC-13, no. 5, pp. 536-541(1978).
- (4) I. Ohkura, O. Tomisawa, M. Nakaya, Y. Ohbayashi and T. Nakano: "A multilevel metallized DSA MOS masterslice", IEEE J. Solid-State Circuits, vol. SC-14, no. 4, pp. 764-766(1979).
- (5) 堀場, 加藤, 茅野, 金田, 村井: "高速ECLゲートアレイ", 三菱電機技報, vol. 54, no. 12, pp. 829-833(1980).



- (6) H.Kanada, K.Okazaki, M.Tachibana, R.Kato and S.Murai: "Channel-order router-a new routing technique for a masterslice LSI", J. Digital Syst., vol. 4, no. 4, pp. 427-441(1981).
- (7) H.Shiraishi and F.Hirose: "Efficient placement and routing techniques for master slice LSI", Proc. 17th DA Conf., pp. 458-464 (1980).
- (8) 寺井, 村井, 田中, 金田, 藤原, 樹下: " マスタスライス方式LSIにおけるチャネル割当の一手法", 情報処理学会電子装置設計技術研究会資料, 1-2(1979).
- (9) M.Kohara, H.Shibata, S.Nakao and H.Nakata: "High-thermal conduction package technology for flip chip devices", Proc. 33rd Electronic Components Conf., pp. 140-144(1983).
- (10)K.Sato and T.Nagai: "A method of specifying the relative locations between blocks in a routing program for building block LSI", Proc. 1979 ISCAS, pp. 673-676(1979).
- (11)原田, 福井, 築山, 白川, 尾崎: " マスタスライス方式LSIの一グローバル配線手法", 信学技報, CAS82-80(1982).
- (12)M.Terai: "A method of improving the terminal assignment in the channel routing for gate arrays", IEEE Trans. Comput.-Aided Des. Integrated Circuits & Syst., vol. CAD-4, no. 3, pp. 329-336(1985).
- (13)寺井, 佐藤: " チャネル配線法における幹線分割の一手法", 信学論(A), vol. J68-A, no. 11, pp. 1184-1192(1985).
- (14)G.Persky, D.N.Deutsh and D.G.Schweikert: "LTX-A minicomputer-based system for automated LSI layout", J. Des. Autom. & Fault-Tolerant Comput., vol. 1, no. 3, pp. 217-255(1977).
- (15)寺井, 野田, 佐藤, 八原: " 種々のチップ構造のゲートアレイに適用可能な配線プログラム", 情報処理学会論文誌, vol. 25, no. 3, pp. 357-364 (1984).

## 第3章 グローバル配線手法

### 3.1 緒言

プリント基板の配線等で使用されている迷路法<sup>(1)</sup>や線分探索法<sup>(2)</sup>、<sup>(3)</sup>は端子対毎に配線径路を決めていくため、配線結果の良否がその配線順序に大きく左右される。また、迷路法や線分探索法を大規模なVLSIに適用すると処理時間が問題となる。これらの問題点を解決するため、配線設計をグローバル配線とチャンネル配線の2段階に分けて行う<sup>(4)</sup>、<sup>(5)</sup>。

本章では、このグローバル配線の手法を考察する。グローバル配線の手法については、既に配線混雑度を評価しながら配線する手法<sup>(5)</sup>等が報告されている。本章で提案する手法は、文献(5)の手法を基本として、更にグローバル配線における信号ネットの処理順序を評価する関数を導入したものである。ここでは、配線の迂回が少なくなるようにグローバル配線における信号ネットの処理順序を決定し、これに基づき配線の混雑度を均一化するようにグローバル配線を行う手法を考察する。更に、この手法を900ゲートのECLゲートアレイに適用した結果を示す。

### 3.2 グローバル配線の算法

2.5.1節で述べたように、グローバル配線では図2.9のチャンネルグラフ $G_c = [V, E]$ 上で信号ネットの径路を決定する。ここで、 $V$ はチャンネル要素に対応する頂点の集合であり、 $E$ は隣接する2つのチャンネル要素に対応する頂点对を結ぶ辺の集合で、 $E$ の各辺 $e_k$  ( $k = 1, 2, \dots, K$ )には配線の通過可能数を表わす容量 $C_k$ が与えられている。但し、 $K$ は辺の総数である。

ゲートアレイにおける配線設計では、チャンネルの幅があらかじめ固定されているため配線不能が発生する。このため、ゲートアレイのグローバル配線での目標を、配線の局所的混雑を少なくし配線率の向上をはかることと、迂回径路を割り当てられる信号ネットを少なくし総配線長を短くすることとする。

グローバル配線では、まず各信号ネット $W_i$  ( $i = 1, 2, \dots, S$ )に対し、複数個の同一の径路長を持つ径路の候補 $\{P_{i1}, P_{i2}, \dots, P_{in}\}$ を見つける。但し、 $S$ は全信号ネット数である。この時、以下に定義する信号ネットの処理順序を示す評価関数 $\Delta L(W_i)$ の値を計算する。

$$\Delta L(W_i) = L(d_i) - L(P_{ij}) \quad (3.1)$$

但し、 $L(d_i)$ は容量 $C_k$ の値が大きい辺(つまり水平チャンネルと垂直チャンネル

) だけを通る信号ネット  $W_i$  の迂回径路  $d_i$  の径路長で、 $L(P_{ij})$  はある径路の候補  $P_{ij}$  の径路長である。

信号ネットの径路の候補を以下のように見出す。一般に  $m$  個の端子を接続する信号ネットに対し、最小木アルゴリズム<sup>(6)</sup>を用いて  $m$  個の端子間の接続を、マンハッタン距離<sup>(6)</sup>の和が最小となるような  $m-1$  個の端子対 (2端子の組) 間の接続に分解する。次に各端子対間の  $G_c$  上での径路を見出すことにより、図3.1のように信号ネットの径路の候補が決定される。図3.1の例では端子対  $(t_2, t_3)$  間の径路を見出す時に、配線長を短くするために、端子  $t_2$  に接続する既に求めた径路上の  $G_c$  の頂点と端子  $t_3$  の間の径路を求めている。

次に、 $\Delta L(W_i)$  の値が大きい信号ネットから順に、容量制限式(3.2)を満たすように径路の候補を最終的なグローバル配線の径路として決定していく。

$$\sum_{j=1}^S g_{kj} + \alpha \leq C_k \quad (k=1, 2, \dots, K) \quad (3.2)$$

但し、グローバル配線のある途中の段階で、既に径路決定された信号ネット  $W_j$  の径路が辺  $e_k$  を通過する場合  $g_{kj} = 1$  とし、それ以外の場合は  $g_{kj} = 0$  とする。また、 $\alpha$  は外部から指定するパラメータで、水平チャネルと垂直チャネルの部分の容量制限式では  $\alpha = 1$  又は  $2$  とし、それ以外の部分の容量制限式では  $\alpha =$

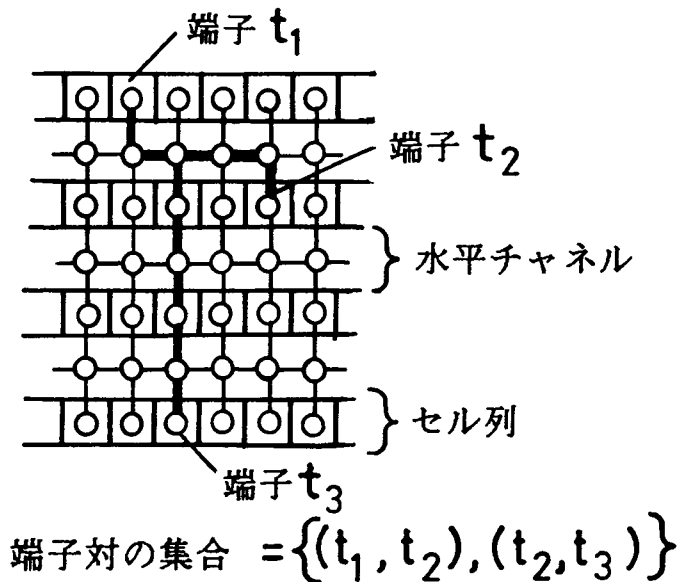


図3.1 信号ネットの径路の候補の例

0とする。 $\alpha = 1$ 又は2と指定する理由は、チャンネル配線において実際に必要とするトラック数は、幹線間の上下制約関係<sup>(7)</sup>等のために、式(3.2)の $\sum g_{kj}$ より大きくなるからである。以下に、グローバル配線の算法を示す。

[グローバル配線の算法]

ステップ1：すべての信号ネットに対し、チャンネルグラフ $G_c$ 上で複数個の同一径路長を持つ径路の候補 $\{P_{i1}, P_{i2}, \dots, P_{in}\}$ を見出し、評価関数 $\Delta L(W_i)$ の値を計算する。

ステップ2：すべての信号ネットの径路がステップ4で決定されたならば終了。そうでない場合、ステップ3へ行く。

ステップ3：径路未決定の信号ネットの内、 $\Delta L(W_i)$ の値が最も大きい信号ネット $W_i$ を選ぶ。もし、 $W_i$ の径路の候補の内のどれを最終的な径路としても容量制限式(3.2)を満足しないならば、 $W_i$ に対しこの時点での容量制限式(3.2)を満足するような複数個の同一径路長の径路の候補 $\{P_{i1}, P_{i2}, \dots, P_{in}\}$ を決定しなおし、 $\Delta L(W_i)$ を変更する。その後、ステップ3へ行く。そうでない場合はステップ4へ行く。

ステップ4：ステップ3で選択した $W_i$ に対し、容量制限式(3.2)を満足する径路の候補の中で下記の関数 $X(P_{ij})$ が最小の $P_{ij}$ を $W_i$ の最終的な径路として決定し、ステップ2へ行く。

$G_c$ の頂点 $v_k$ がセル列上の領域に対応する頂点である時、 $x(v_k)$ を次のように定義する。

$$x(v_k) = \sum_{m=1}^S x_{km}$$

これ以外の頂点に対し、 $x(v_k) = 0$ とする。ここで、信号ネット $W_m$ の最終的な径路がステップ4で未だ決定されていなく、かつ $W_m$ のある径路の候補 $P_{mj}$ が頂点 $v_k$ に対応するセル列上の貫通トラックを通過する時、 $x_{km} = 1$ とし、それ以外の時、 $x_{km} = 0$ とする。信号ネット $W_i$ のある径路の候補 $P_{ij}$ に対し、 $X(P_{ij})$ を次のように定義する。

$$X(P_{ij}) = \sum x(v_k) \quad (3.3)$$

ここで、 $\sum$ は $P_{ij}$ が通過する全頂点についての和を示す。

### 3.3 評価関数 $\Delta L(W_i)$ と $X(P_{ij})$ の意味

式(3.1)の評価関数 $\Delta L(W_i)$ の意味について述べる。 $\Delta L(W_i)$ は図

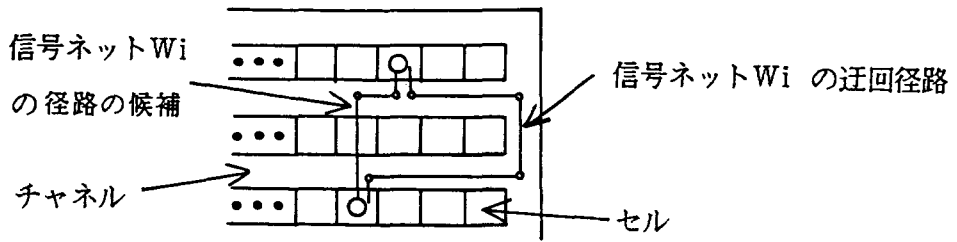
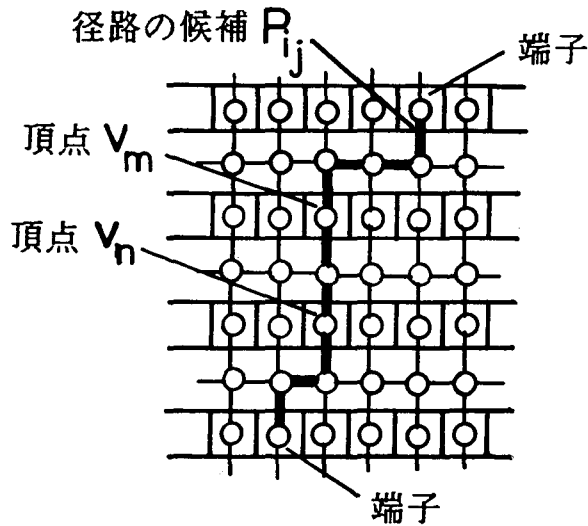


図3.2 評価関数  $\Delta L(W_i)$



$$X(P_{ij}) = x(v_m) + x(v_n)$$

図3.3 関数  $X(P_{ij})$

3.2に示すように、信号ネット  $W_i$  の径路の候補の径路長と、容量  $C_k$  の大きい水平チャンネルやチップ両脇の垂直チャンネルだけを通る迂回径路の径路長の差を表わす。従って、 $\Delta L(W_i)$  の値が大きい径路の候補はセル列上の貫通トラックを有効に使った径路の候補である。こうして、 $\Delta L(W_i)$  が大きい信号ネットから順にグローバル配線を行うことによりセル列上の貫通トラックを有効に使ったグローバル配線結果が得られ、総配線長が短くなることが期待できる。

次に、式(3.3)の関数  $X(P_{ij})$  の意味について述べる。 $X(P_{ij})$  の例を図3.3に示す。 $X(P_{ij})$  の値が小さい径路の候補とは、 $W_i$  以外の信号ネットの

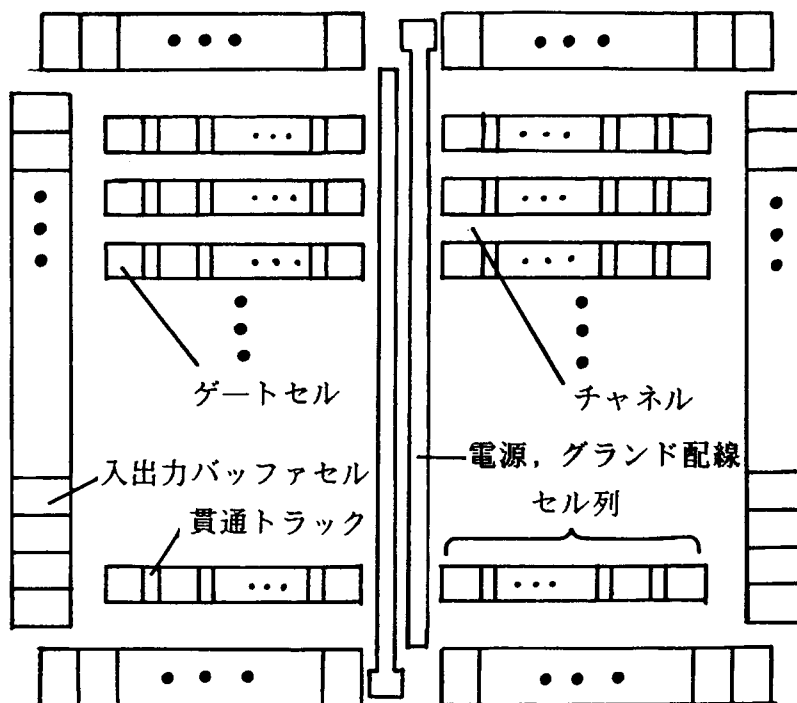


図3.4 900ゲートのECLゲートアレイのチップ構造

径路としては使われる可能性の低いセル列上の貫通トラックを使う径路である。従って、信号ネット $W_i$ の径路の候補 $\{P_{i1}, P_{i2}, \dots, P_{in}\}$ の中で、 $X(P_{ij})$ が最小の $P_{ij}$ を最終的な径路として決定することによりセル列上の貫通トラックを有効に使ったグローバル配線結果が得られ、総配線長が短くなることが期待できる。

### 3.4 実験結果と考察

本節では、以上で述べたグローバル配線算法を900ゲートのECLゲートアレイに適用した結果について述べる。本手法に基づくグローバル配線のプログラムをFORTRANで作成した。プログラムは約4,400ステートメントからなる。本プログラムを第2.6節で述べたMARS-II配線プログラムに組み込み、これを900ゲートのECLゲートアレイ<sup>(8)</sup>に適用した。

実験に用いたECLゲートアレイのチップ構造を図3.4に示す。チップ上の周

辺領域には入出力バッファセルが並び、内部領域にはセル列が垂直方向に15列と配線領域（チャンネル）と電源、グランド配線が図3.4のように並んでいる。各セル列にはゲートセル60個と1本の垂直配線を通すことができる貫通トラックが交互に並んでいる。ゲートセルの端子間の配線はセル列間の2層のチャンネル内で行われる。ゲートセルは図3.5のように3入力NORの論理機能を持ち、上下両辺に電気的に等価な端子を持っているため、ゲートセルを2入力NOR又はインバータとして使用する時や未使用の時に空き端子が生じ、この上を貫通トラックとして用いることができる。

図3.6はECLゲートアレイ41品種に対して、各信号ネット毎に擬似スタイナー木に基づいて計算した配線長の総和と配線プログラムMARS-M IIによる総配線長の関係を示したものである。スタイナー木<sup>(6)</sup>に基づいて計算した配線長は最適解を与えるが、実用的な時間内で計算することは難かしいので、ここでは最適解に近い擬似スタイナー木を計算し、これを配線長の評価基準にしている。

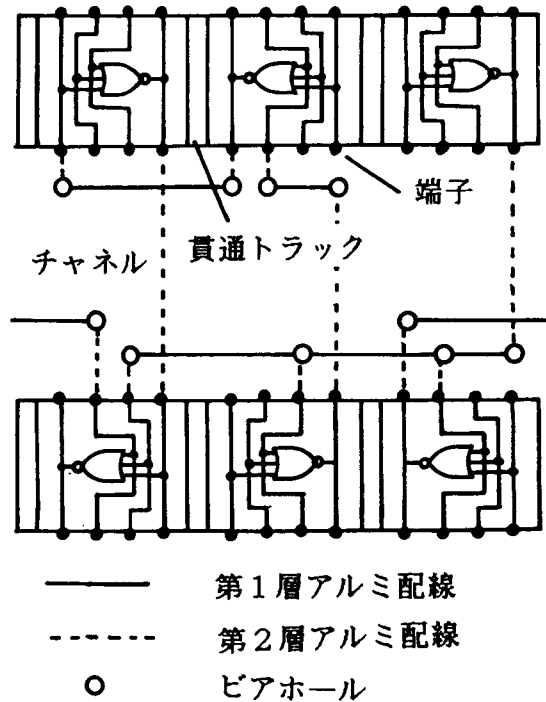


図3.5 ゲートセル間の配線

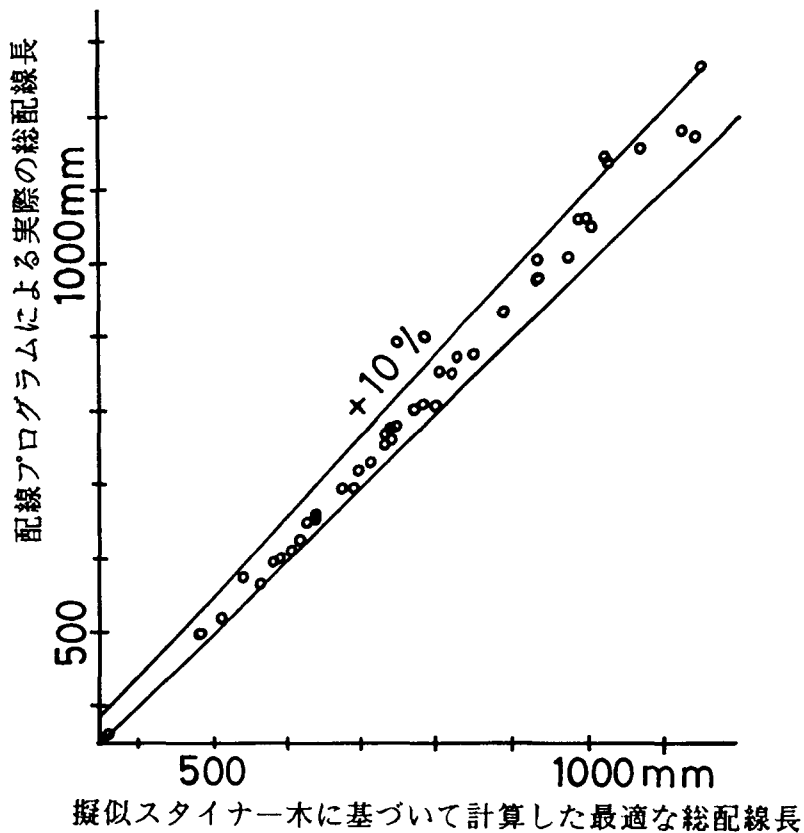
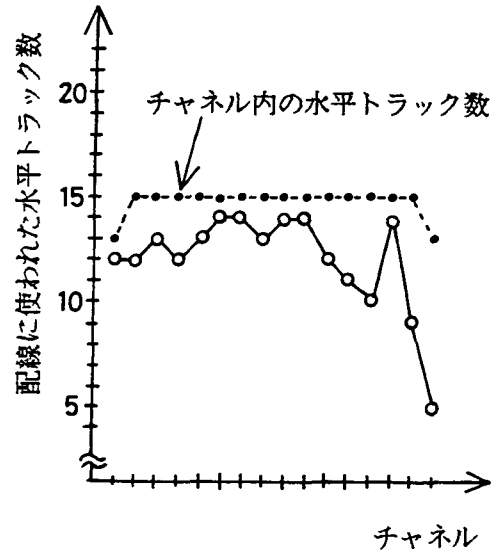
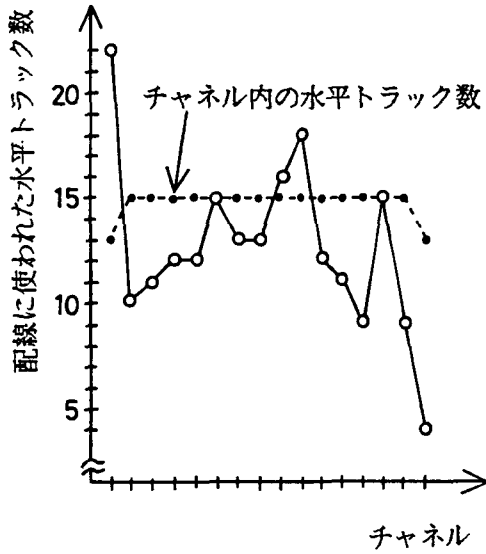


図3.6 ECLゲートアレイにおける擬似スタイナー木に基づいて計算した最適な総配線長と配線プログラムによる総配線長

図3.6に示すように、実際の総配線長は擬似スタイナー木に基づき計算した総配線長より高々10%増加しているだけである。総配線長を短くするのに最も重要な役割を持つのが配線設計中のグローバル配線であることを考慮すると、本手法に基づくグローバル配線は配線の迂回が少なくなるように配線径路を決定していることが図3.6から明らかになる。

図3.7は、740個のゲートセルを使用し、信号ネット数が779であるデータに対しグローバル配線の配線混雑度を均一化する機能の有効性を示したものである。グローバル配線では、各チャンネル内の配線混雑度の均一化のため容量制限式(3.2)を考慮している。図3.7では、容量制限式(3.2)を考慮せずグローバル





( i ) 容量制限式を考慮しない場合      ( ii ) 容量制限式を考慮した場合  
 図3.7 配線混雑度の均一化の機能の評価



図3.8 配線結果のプロット図

配線を行った時と、 $\alpha = 2$ とした容量制限式(3.2)を考慮してグローバル配線を行った時の配線に使用された水平トラック数を各水平チャンネル毎に示した。使用水平トラック数がチャンネル内に用意された水平トラック数を越えた場合、配線不能となる。図3.7からグローバル配線の配線混雑度を均一化する機能が有効に働いていることが分かる。

図3.8は図3.7のデータに対する配線結果である。0.7MIPSの計算機MELCOM-7700によるMARS-MIIの処理時間はグローバル配線が8分、全配線設計が31分であった。

### 3.5 結 言

本章では、ゲートアレイの配線設計におけるグローバル配線に対し、配線の迂回が少なくなるように信号ネットの処理順序を決定し、これに基づき配線の混雑度を均一化するようにグローバル配線を行う手法を考察した。適用結果により、本手法が配線の迂回を減少させ、かつ配線の局所混雑を緩和して配線不能が生じる可能性を低くする点で、有効であることを示した。

なお、本章の内容は文献(9)、(10)に基づいている。

### 参考文献

- (1) C.Y.Lee: "An algorithm for path connections and its applications", IRE Trans., vol. EC-16, no. 3, pp. 346-365(1961).
- (2) K.Mikami and K.Tabuchi: "A computer program for optimal routing of printed circuit conductors", IFIP Congress 68, pp. 1475-1478(1968).
- (3) D.W.Hightower: "A solution to line routing problems on the continuous plane", Proc. 6th DA Workshop, pp. 1-24(1969).
- (4) A.Hashimoto and J.Stevens: "Wire routing by optimizing channel assignment with large apertures", Proc. 8th DA Workshop, pp. 155-169 (1971).
- (5) 杉山, 可児: "マルチチップLSI基板配線の一算法", 信学論(C), vol. 58-C, no. 4, pp. 201-208(1975).
- (6) M.A.Breuer: Design Automation of Digital Systems, Englewood Cliffs, New Jersey: Prentice-Hall, Inc.(1972).
- (7) G.Persky, D.N.Deutsch and D.G.Schweikert: "LTX-A minicomputer-based system for automated LSI layout", J. Des. Autom. & Fault-Tolerant

Comput., vol. 1, no. 3, pp. 217-255(1977).

- (8) 堀場, 加藤, 茅野, 金田, 村井: "高速ECLゲートアレイ", 三菱電機技報, vol. 54, no. 12, pp. 829-833(1980).
- (9) 寺井, 村井, 田中, 金田, 藤原, 樹下: "マスタスライス方式LSIにおけるチャンネル割当の一手法", 情報処理学会電子装置設計技術研究会資料, 1-2(1979).
- (10) C.Tanaka, S.Murai, H.Tsuji, T.Yahara, K.Okazaki, M.Terai, R.Katoh and M.Tachibana: "An integrated computer aided design system for gate array masterslices: Part 2 the layout design system MARS-M3", Proc. 18th DA Conf., pp. 812-819(1981).

## 第4章 端子割当手法

### 4.1 緒言

ゲートアレイの配線設計におけるチャンネル配線では、対向する2セル列間の矩形の水平チャンネル内で配線を行う。この時、幹線（配線の水平方向の線分）間の上下制約グラフにサイクルが含まれる場合は、幹線分割によりサイクルを解消しなければ配線できないことがよく知られている<sup>(1)</sup>。しかし、幹線分割を行えばビアホールが増えるため信号ネットの抵抗が増大し、また一般に配線に使用される水平トラックの数が増大する。

本章では、ゲートアレイの配線設計における端子割当の算法について考察する。端子割当問題は、上下制約グラフにおけるサイクルを除去するために、信号ネットを論理的に等価な端子間でどのように割当ててくるかを決定する問題である。この問題に対し、まず初期端子割当を行い、次にこれを改善することにより最終的な端子割当を得る算法を提案する。この端子割当改善においては、論理的に等価な2端子間で信号ネットの割当を繰り返し交換することにより上下制約グラフ中のサイクルの数を減少させる算法を適用する。更に、算法の高速化のために、端子割当を変更することにより上下制約グラフ中のサイクルの数が減少するか否かの判定が、その部分グラフに着目するだけで可能であることを理論的に証明する。更に、900ゲートのECLゲートアレイに本算法を適用した結果について述べる。

### 4.2 問題の定式化

端子割当問題に対し、ある初期端子割当を反復改善することにより最終的な端子割当を得る方法について考察する。このため、本節では、まずある端子割当が得られた状態における配線要求と上下制約グラフの定義を行い、次に端子割当改善問題の定式化を行う。

対向する2セル列の上側セル列、下側セル列上であり、左から第 $i$ 番目の垂直格子線上にあるゲートセルの端子をそれぞれ $t_i$ 、 $b_i$  ( $i=1, 2, \dots, M$ ;  $M$ は1セル列上の端子数を示す)と表わす。 $D(t_i)$ 、 $D(b_i)$ はそれぞれ端子 $t_i$ 、 $b_i$ に割り当てられた信号ネット番号を表わす。すなわち、全信号ネット数を $S$ として、 $D(t_i)$ 、 $D(b_i) \in \{0, 1, \dots, S\}$ であり、 $0$ は空き端子を表わす。 $0$ 以外の同じ信号ネット番号を割り当てられた端子は等電位となるように、2セル列間のチャンネル内において結線される。チャンネルは2層からなり、

第1層の水平トラック上にセル列に平行な配線（幹線），第2層の垂直格子線上にセル列に垂直な配線を置き，これらをビアホールにより連結する。この時，配線要求Dは  $\{t_1, t_2, \dots, t_M, b_1, b_2, \dots, b_M\}$  から  $\{0, 1, \dots, S\}$  への写像であり，次のように表わす。

$$D = \begin{bmatrix} D(t_1), D(t_2), \dots, D(t_M) \\ D(b_1), D(b_2), \dots, D(b_M) \end{bmatrix}$$

本章で扱う上下制約グラフとしては2種類の上下制約グラフ，即ち，各頂点が信号ネットに1対1対応する上下制約グラフと，各頂点が2端子サブネット (2-terminal subnet) <sup>(1)</sup> に1対1対応する上下制約グラフが考えられる。ここで，2端子サブネットとは，信号ネットの端子をそのx座標値に基づき順序づけし，その信号ネットを連続する隣接2端子のペアに分解することにより得られる信号ネットの一部である。

後述の例4.2に示すように，端子割当改善問題に対しては，たとえチャネル配線の手法として2端子サブネットに対する幹線を水平トラック上へ配置していく "dogleg channel router" <sup>(1), (2)</sup> と同様のチャネル配線の手法を用いるとしても，前者の上下制約グラフを使って得られる結果は，後者の上下制約グラフを使って得られる結果より，よい結果となることが多い。それ故，上下制約グラフを次のように定義する。

[定義4.1] 配線要求Dに対し，有向グラフ  $G(D) = [V, E]$  を上下制約グラフと呼ぶ。ここで， $V = \{1, 2, \dots, S\}$  の各頂点  $i$  は信号ネット番号  $i$  の幹線を表わし，信号ネット  $i$  から信号ネット  $j$  への有向辺  $e_k = (i, j) \in E$  であるのはある  $n$  で  $D(t_n) = i, D(b_n) = j, i \neq j$  の時かつその時に限る。但し， $|\{n \mid D(t_n) = i, D(b_n) = j, i \neq j, i \cdot j \neq 0\}| = m > 1$  ならば， $G(D)$  には  $m$  本の同方向に並列な辺  $e_{k_1} = (i, j), e_{k_2} = (i, j) \dots, e_{k_m} = (i, j)$  が存在する。 (定義終)

$V$  の空でない部分集合  $U$  によって生成される  $G(D)$  の部分グラフとは， $U$  を頂点集合とし，両端点を  $U$  に持つすべての辺を辺集合とするようなグラフであると定義する。

上下制約グラフ  $G(D)$  の辺の順序列  $L = (e_{k_1}, e_{k_2}, \dots, e_{k_n})$  (ただし  $e_{k_i} = (v_{i-1}, v_i), i = 1, 2, \dots, n$ ) が同一頂点を2度以上通らない時， $L$  のなす部分グラフを  $v_0$  から  $v_n$  に至る有向道といい，更に  $v_0 = v_n$  である時，サイクルという。対向する2セル列上のある端子  $x_i, x_j$  ( $x = t$  又は  $x = b$ ) が論理的に等価である時， $x_i \sim x_j$  と書く。例えば，関係  $\sim$  は  $NO$

Rゲートのような論理ゲートの入力端子の間で成り立つ。 $x_i \sim x_j$ とすると、 $\sim$ は同値関係である。配線要求Dに対し、同値関係 $\sim$ の同じ同値類に属する2つ端子に割り当てられている信号ネット番号を交換しても、回路の論理機能は影響されない。端子割当問題における初期端子割当を配線要求Dと表わし、同値関係 $\sim$ が成り立つ端子間で割り当てられた信号ネット番号を任意に置換して得られる配線要求全体の集合(D自身を含む)を $\psi(D)$ で表わす。また、Dの上下制約グラフ $G(D)$ の独立なサイクル数を $\mu(D)$ とする。この時、本章で考察する端子割当改善問題は次のように定式化される。

〔問題〕 配線要求Dとその各端子間の同値関係 $\sim$ が与えられた時、 $\mu(D')$ を最小にする $D' \in \psi(D)$ を求めよ。

〔例4.1〕 図4.1(a)に示す配線要求D

$$D = \begin{bmatrix} 1, 4, 2, 2, 3, 1 \\ 3, 2, 1, 3, 4, 4 \end{bmatrix}$$

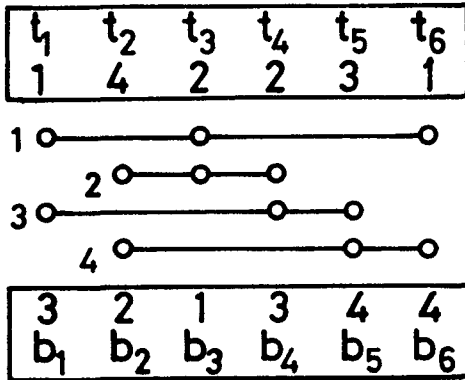
とその全端子の同値関係 $\sim$ による直和分割

$$\{t_1, t_2\}, \{t_3\}, \{t_4, t_5\}, \{t_6\}$$

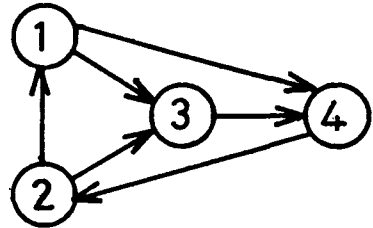
$$\{b_1\}, \{b_2, b_3\}, \{b_4\}, \{b_5, b_6\}$$

が与えられた時、上下制約グラフ $G(D)$ は図4.1(b)に示され、 $\mu(D) = 3$ である。Dにおいて同値関係 $\sim$ にある2端子 $t_1$ と $t_2$ 、 $t_4$ と $t_5$ 及び $b_2$ と $b_3$ に割り当てられた信号ネット番号を置換することにより、図4.2(a)の配線要求 $D' \in \psi(D)$ が得られる。上下制約グラフ $G(D')$ は図4.2(b)に示され、 $\mu(D') = 0$ であるから、端子割当改善問題はDに対して $D'$ を求める問題である。

〔例4.2〕 例4.1の配線要求D、同値関係 $\sim$ 及び上下制約グラフ $G(D)$ に対して、端子割当を改善することによって $G(D)$ におけるすべてのサイクルは除去され、図4.2(a)に示す配線要求 $D'$ が得られる。配線要求 $D'$ に対して、4本の水平トラック(水平格子線)と9個のビアホールを使って配線を行うことができる。一方、配線要求Dに対し、2端子サブネット間の上下制約を表わすグラフは図4.3(b)のようになる。このグラフが端子割当改善問題に用いられると仮定する。配線要求Dに対し、図4.3(b)のグラフはサイクルを持たないので、端子割当は改善されず図4.3(a)に示すように6本の水平トラックと15個のビアホールを使って配線は行われる。それ故、端子割当改善問題に対しては、定義4.1で定義された上下制約グラフを使うことが望ましい。

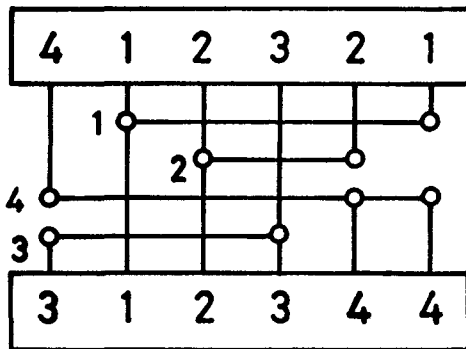


(a)

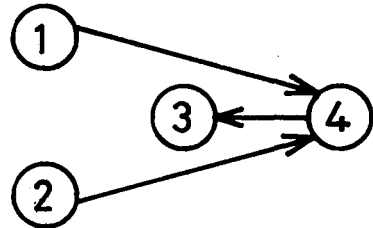


(b)

図4.1 (a) 配線要求D, (b) 上下制約グラフG(D)



(a)



(b)

図4.2 (a) 配線要求D', (b) 上下制約グラフG(D')

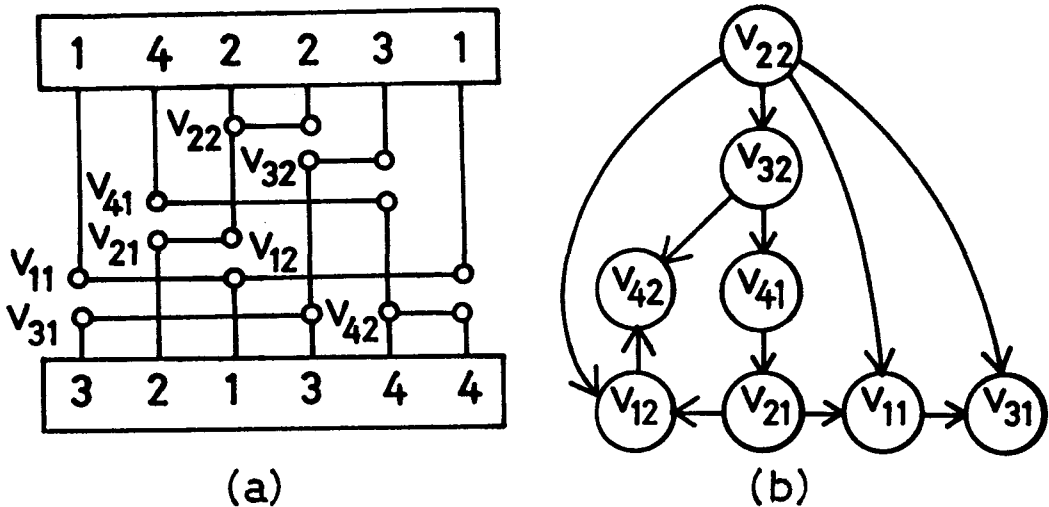


図4.3 (a) 配線要求 $D$ , (b) 2端子サブネット間の上下制約グラフ

### 4.3 端子割当の算法

端子割当問題は扱いにくい組合せ問題であり、多項式時間で最適解を求めることは絶望的である。従ってヒューリスティックな算法が実用上有用となる。本節では、逐次改善により端子割当を得る算法を提案する。ここでは、配線要求 $D$ に次の交換操作を可能な限り反復適用することにより、 $\mu$ が極小となる解を得る算法を考察する。

[定義4.2] 配線要求 $D$ に対し、2端子 $x_i, x_j$  ( $x = t$  又は  $x = b$ ) に割り当てられた信号ネット番号を置換した配線要求を $\sigma_{x_i, x_j}(D)$ で表わす。この時、端子 $x_i$ と $x_j$ に対する交換操作とは、 $D$ を $\sigma_{x_i, x_j}(D)$ に変更する操作であると定義する。 $x_i \sim x_j$ かつ $\mu(D) > \mu(\sigma_{x_i, x_j}(D))$ であるとき、端子 $x_i$ と $x_j$ に対し交換操作は適用可能であるという。 (定義終)

配線要求 $D$ に対する端子割当改善の算法を次に示す。

[端子割当改善の算法]

ステップ1:  $i \leftarrow 1$

ステップ2: 端子 $x_i$  ( $x = t$  又は  $x = b$ ) に対し交換操作が適用可能な $x_j$



が存在するならば、 $x_i$ と $x_j$ に交換操作を適用し、ステップ3へ行く。存在しないならば、ステップ3へ行く。

ステップ3 :  $i < M$ ならば、 $i \leftarrow i + 1$ としてステップ2へもどる。 $i = M$ かつDのすべての端子に交換操作が適用不可能であった時、終了する。それ以外の時はステップ1へもどる。

[例4.3] 例4.1の配線要求Dと同値関係 $\sim$ に対し $t_1 \sim t_2$ ,  $\mu(D) = 3$ ,  $\mu(\sigma_{t_1 t_2}(D)) = 2$ であるから、2端子 $t_1$ と $t_2$ に対し交換操作が適用可能である。 $t_1$ と $t_2$ に交換操作を適用後の配線要求 $\sigma_{t_1 t_2}(D)$ とその制約グラフ $G(\sigma_{t_1 t_2}(D))$ を図4.4に示す。図4.1(a)の配線要求Dに端子割当改善の算法を適用すると、 $t_1$ と $t_2$ ,  $b_2$ と $b_3$ 及び $t_4$ と $t_5$ に、この順に交換操作が適用され、上下制約グラフ中のすべてのサイクルが除去され図4.2(a)の配線要求D'が得られる。 (例終)

次に $\mu(D)$ の算出方法について考える。上下制約グラフ $G(D)$ の強連結成分を $G_{(1)}, G_{(2)}, \dots, G_{(n)}$ と表わす。 $G(D)$ のすべてのサイクルは $G(D)$ の強連結成分 $G_{(1)}, G_{(2)}, \dots, G_{(n)}$ のいずれかの部分グラフである。強連結成分 $G_{(i)} = [V_i, E_i]$ には独立なサイクルが $|E_i| - |V_i| + 1$ 個あることが文献(3)に証明されている。従って、 $G(D)$ の独立なサイクル数 $\mu(D)$ は次式のようにして算出される。

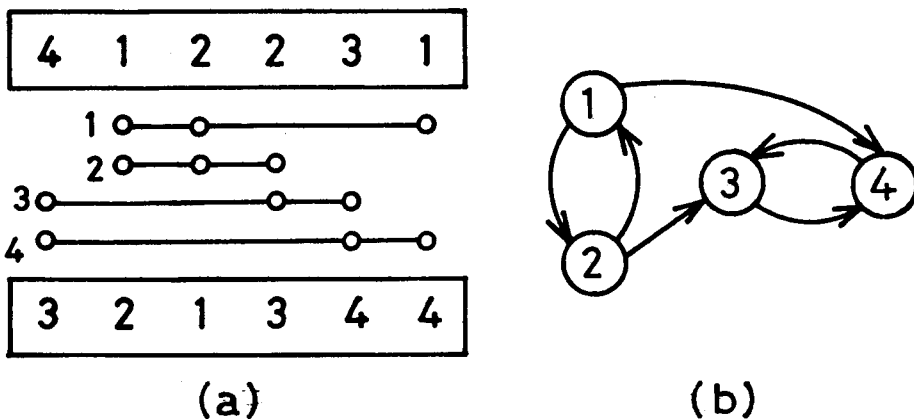


図4.4 (a) 配線要求 $\sigma_{t_1 t_2}(D)$ , (b) 上下制約グラフ $G(\sigma_{t_1 t_2}(D))$

$$\mu(D) = \sum_{i=1}^n (|E_i| - |V_i| + 1) \quad (4.1)$$

次に、本算法の手数について考察する。本算法が有限ステップで終了することを保証する次の補題4.1が成り立つ。

[補題4.1] 端子割当改善の算法は、算法のステップ3からステップ1へ高々 $\mu_0 - 1$ 回もどると終了する。但し、算法適用前の配線要求を $D_0$ とすれば、 $\mu_0 = \mu(D_0)$ である。

(証明) 本算法においてステップ3からステップ1へもどるのは、ステップ2において $x_1, x_2, \dots, x_M$  ( $x = t$  又は  $x = b$ ) のある2端子 $x_i$ と $x_j$ に対し交換操作が適用可能で、 $\mu$ が減少した時かつその時に限る。また $\mu(D) = 0$ なる配線要求 $D$ のいかなる2端子にも交換操作は適用不可能である。故に、本算法はステップ3からステップ1へ高々 $\mu_0 - 1$ 回もどると終了する。(証明終)

配線要求 $D$ とその各端子間の同値関係 $\sim$ が与えられた時、上下制約グラフ $G(D)$ の隣接構造(adjacency structure)<sup>(4)</sup>の作成に要する手数は $O(M)$ である。深さ優先の探索(depth-first search)を用いた分割アルゴリズム<sup>(4)</sup>によれば $G(D)$ を強連結成分に分割するのに要する手数は高々 $O(|V| + |E|)$ であるから、式(4.1)を用いて $O(|V| + |E|)$ の手数で $\mu(D)$ を算出できる。 $\mu(\sigma_{x_i, j}(D))$ の算出の手数も同様であるから、 $x_i \sim x_j$ なる2端子 $x_i$ と $x_j$ に交換操作が適用可能か否かが $O(M + |V| + |E|)$ の手数で判定できる。故に、本算法の手数は高々

$$\begin{aligned} & O(\mu_0 \cdot 2M \cdot N(M + |V| + |E|)) \\ & \leq O(\mu_0 \cdot 2M \cdot N(M + 2M + M)) \\ & \leq O(\mu_0 \cdot N \cdot M^2) \end{aligned}$$

である。

但し、 $N$ は同値関係 $\sim$ の各同値類における端子の数の最大値を表わす。従って、次の定理4.1が成り立つ。

[定理4.1] 端子割当改善の算法に要する手数は高々 $O(\mu_0 \cdot N \cdot M^2)$ である。

#### 4.4 算法の高速化

前節で提案した端子割当改善のアルゴリズムの実行を高速化するために、本節では配線要求 $D$ の $x_i \sim x_j$ なる2端子 $x_i$ と $x_j$  ( $x = t$  又は  $x = b$ ) に対し

交換操作が適用可能か否かの判定方法について述べる。まず2端子 $x_i$ と $x_j$ に割り当てられた信号ネット番号を置換した後の上下制約グラフ $G(\sigma_{x_i, x_j}(D))$ について述べ、次に交換操作に関する定理を示す。

一般に、上下制約グラフ $G(D) = [V, E]$ に対して $G(\sigma_{x_i, x_j}(D)) = [V, E']$ は図4.5のように

$$E' = E \cup \{e_{i|E|+1}, e_{i|E|+2}\} - \{e_p, e_q\} \quad (4.2)$$

$$e_p = (D(t_i), D(b_i)), \quad e_q = (D(t_j), D(b_j))$$

$$e_{i|E|+1} = (D(t_i), D(b_j)), \quad e_{i|E|+2} = (D(t_j), D(b_i))$$

と表わされる。なお、定義4.1に示すように、 $t_i, b_i, t_j, b_j$ のいくつかに同一信号ネット番号が与えられているため式(4.2)の有向辺の両端点が同一頂点となる辺や、空き端子のため式(4.2)の有向辺の端点が信号ネット0となる辺は $E$ や $E'$ に含めない。

[定理4.2] 配線要求 $D$ の $x_i \sim x_j$ なる2端子 $x_i$ と $x_j$  ( $x = t$ 又は $x = b$ ) に対し、上下制約グラフ $G(D)$ における辺 $e_p = (D(t_i), D(b_i))$ ,  $e_q = (D(t_j), D(b_j))$ が $G(D)$ のどの強連結成分にも含まれないならば、 $x_i$ と $x_j$ に対し交換操作が適用不可能である。

(証明)  $G(D)$ における辺 $e_p, e_q$ が $G(D)$ のどの強連結成分にも含まれないならば、 $G(D)$ におけるどのサイクルもこれらの辺を通らない。それ故、 $G(D)$ の部分グラフ $[V, E - \{e_p, e_q\}]$ における独立なサイクルの数は $\mu(D)$ に等しい。従って、 $G(D)$ と $G(\sigma_{x_i, x_j}(D))$ の関係式(4.2)より $\mu(D) \leq \mu(\sigma_{x_i, x_j}(D))$ となり、 $x_i$ と $x_j$ に対し交換操作が適用不可能である。

(証明終)

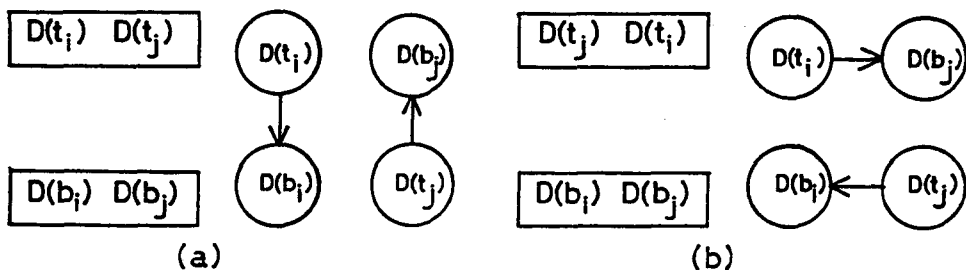


図4.5 (a) 上下制約グラフ $G(D)$ と (b) 上下制約グラフ $G(\sigma_{x_i, x_j}(D))$

$G(D)$ の強連結成分を $G_{(1)} = [V_1, E_1]$ ,  $G_{(2)} = [V_2, E_2]$ ,  
 $\dots$ ,  $G_{(n)} = [V_n, E_n]$ と表わす。以下では,  $x_i \sim x_j$ なる2端子 $x_i$ ,  
 $x_j$ に対し, 定理4.2の条件が満たされないとき, 即ち, ある $G_{(k)} = [V_k,$   
 $E_k]$ が辺 $e_p = (D(t_i), D(b_i))$ を含む時に交換操作が適用可能か否かを判  
 定する方法について述べる。以下で述べる補題4.1—4.4では与えられた配線要  
 求 $D$ の $x_i \sim x_j$ である2端子 $x_i$ と $x_j$ に対し,  $G_{(k)} = [V_k, E_k]$ が辺  
 $e_p$ を含む時, 交換操作が適用可能であるかどうかを判定する方法を示す。頂点  
 $D(t_j)$ 又は $D(b_j)$ が $V_k$ に含まれる場合と共に $V_k$ に含まれない場合の2つ  
 の場合に分けて考察する。

(i)  $D(t_j) \in V_k$ かつ/又は $D(b_j) \in V_k$ の場合

[補題4.1]  $D(t_j) \in V_k$ かつ/又は $D(b_j) \in V_k$ とする。2端子 $x_i$   
 と $x_j$ に対し交換操作が適用可能であるための必要十分条件は $G(\sigma_{x_i, x_j}(D))$ の  
 $V_k$ によって生成される部分グラフの独立なサイクル数が $G_{(k)}$ の独立なサイク  
 ル数より少ないことである。

(証明)  $V_r$ によって生成される $G(\sigma_{x_i, x_j}(D))$ の部分グラフを $G'_{(r)}$ と  
 表わす。 $D(t_i), D(b_i), D(t_j) \in V_k$ かつ/又は $D(t_i), D(b_i),$   
 $D(b_j) \in V_k$ であるから, 式(4.2)より各 $G'_{(r)}$  ( $1 \leq r \leq n, r \neq k$ )は  
 強連結である。ある部分グラフ $G'_{(r)}$  ( $1 \leq r \leq n, r \neq k$ )が $G(\sigma_{x_i, x_j}(D))$   
 の強連結成分でないとして仮定する。この時,  $V_r \subset U \subseteq V$ なる $G(\sigma_{x_i, x_j}(D))$   
 の強連結成分 $[U, F]$ が存在し, 辺集合 $F$ は式(4.2)の辺 $e_{1 \in E_{1+1}} = (D(t_i),$   
 $D(b_j))$ 又は辺 $e_{1 \in E_{1+2}} = (D(t_j), D(b_i))$ を含む。このことは $G(D)$ に  
 おいて次の条件の1つが満たされることを示す。

(1)  $V_r$ のある頂点から頂点 $D(t_i)$ へ至る有向道かつ, 頂点 $D(b_j)$ から  
 $V_r$ のある頂点に至る有向道がある。

(2)  $V_r$ のある頂点から頂点 $D(t_i)$ へ至る有向道, 頂点 $D(b_j)$ から頂点  
 $D(t_j)$ へ至る有向道, かつ頂点 $D(b_i)$ から $V_r$ のある頂点へ至る有向道があ  
 る。

(3)  $V_r$ のある頂点から頂点 $D(t_j)$ へ至る有向道, かつ頂点 $D(b_i)$ から  
 $V_r$ のある頂点へ至る有向道がある。

(4)  $V_r$ のある頂点から頂点 $D(t_j)$ へ至る有向道, 頂点 $D(b_i)$ から頂点  
 $D(t_i)$ へ至る有向道, かつ $D(b_j)$ から $V_r$ のある頂点へ至る有向道がある。

この時,  $D(t_i), D(b_i), D(t_j) \in V_k$ 又は $D(t_i), D(b_i), D(b_j) \in V_k$ であるから,  $V_r$ の頂点が $V_k$ の頂点と $G(D)$ において同じ強連結

成分に属する。これは $G_{(k)}$ と $G_{(r)}$ が $G(D)$ の相異なる強連結成分であることに矛盾する。故に、各 $G'_{(r)}$  ( $1 \leq r \leq n$ ,  $r \neq k$ )は、 $G(\sigma_{x_i, x_j}(D))$ の強連結成分である。明らかに、各 $G'_{(r)}$  ( $1 \leq r \leq n$ ,  $r \neq k$ )の独立なサイクル数は $G_{(r)}$ のそれと同じである。よって、補題4.1が成り立つ。(証明終)

(ii)  $D(t_i) \notin V_k$ かつ $D(b_j) \notin V_k$ の場合

[補題4.2]  $D(t_i) \neq D(b_j)$ ,  $D(t_i) \notin V_k$ かつ $D(b_j) \notin V_k$ とする。頂点 $D(t_i)$ との間に有向道が存在する $G(D)$ のすべての頂点の集合を $W$ とする。但し、 $D(t_i) \in W$ とする。交換操作が $x_i$ と $x_j$ に対し適用可能であるための必要十分条件は $G(\sigma_{x_i, x_j}(D))$ の $W$ によって生成される部分グラフの独立なサイクル数が $G(D)$ の $W$ によって生成される部分グラフの独立なサイクル数より少ないことである。特に、 $D(t_i) \notin W$ ,  $D(b_j) \notin W$ ならば交換操作が $x_i$ と $x_j$ に適用可能である。

(証明)  $W$ によって生成される $G(D)$ の部分グラフを $G_w$ と表わし、 $W$ によって生成される $G(\sigma_{x_i, x_j}(D))$ の部分グラフを $G'_w$ と表わす。また、 $V_r$ によって生成される $G(\sigma_{x_i, x_j}(D))$ の部分グラフを $G'_{(r)}$ と表わす。考えられる2つの場合がある。

(1) 頂点 $D(t_i)$ と $D(b_j)$ が $G(D)$ の相異なる強連結成分に含まれる場合

各 $G'_{(r)}$  ( $1 \leq r \leq n$ ,  $V_r \not\subseteq W$ )は、 $G(\sigma_{x_i, x_j}(D))$ の強連結成分である。このことは、性質4.1の証明と同様にして背理法により証明される。明らかに、各 $G'_{(r)}$  ( $1 \leq r \leq n$ ,  $V_r \not\subseteq W$ )の独立なサイクル数は $G_{(r)}$ の独立なサイクル数に等しい。こうして、交換操作が $x_i$ と $x_j$ に適用可能であるのは、 $G'_w$ の独立なサイクル数が、 $G_w$ の独立なサイクル数より少ない時、かつその時に限る。式(4.2)の辺 $e_p$ は $G_w$ のあるサイクルに含まれるから、特に $D(t_i) \notin W$ かつ $D(b_j) \notin W$ ならば、 $G'_w$ の独立なサイクル数は $G_w$ の独立なサイクル数より少ない。それ故、 $x_i$ と $x_j$ に対し交換操作が適用可能である。

(2) 頂点 $D(t_i)$ と $D(b_j)$ が $G(D)$ の同一の強連結成分 $G_{(m)} = [V_m, E_m]$ に含まれる場合

$V_m \subset W$ の時は、この証明の(1)の場合と同様にして補題4.2が成り立つことを証明できる。従って、 $V_m \not\subset W$ と仮定する。 $G(\sigma_{x_i, x_j}(D))$ の $V_k \cup V_m$ によって生成される部分グラフと各 $G'_{(r)}$  ( $1 \leq r \leq n$ ,  $r \neq k$ ,  $r \neq m$ )は $G(\sigma_{x_i, x_j}(D))$ の強連結成分である。これは、補題4.1の証明と同様にして背理法によって証明できる。各 $G'_{(r)}$  ( $1 \leq r \leq n$ ,  $r \neq k$ ,  $r \neq m$ )の独立なサイクル数は $G_{(r)}$ の独立なサイクル数に等しい。 $G(\sigma_{x_i, x_j}(D))$ の $V_k \cup V_m$ によ

て生成される部分グラフの独立なサイクル数は  $|E_k \cup E_m| - |V_k \cup V_m| + 1$  である。それ故、式(4.1)より  $\mu(\sigma_{x_i, x_j}(D)) = \mu(D) - 1$  となり交換操作は  $x_i$  と  $x_j$  に対し適用可能である。更に、式(4.2)の辺  $e_p$  は  $G_m$  のあるサイクルに含まれ、 $V_m \subset W$  であるから、 $G'_m$  の独立なサイクル数は  $G_m$  の独立なサイクル数より少ない。よって、補題4.2は成り立つ。 (証明終)

補題4.1と4.2の特殊な場合として次の補題4.3と4.4が成り立つ。

[補題4.3] 補題4.2で定義した集合  $W$  に対し、 $D(t_i) \in W - V_k$ ,  $D(b_j) \in W - V_k$  とする。この時、もし頂点  $D(t_i)$  と  $D(b_j)$  が  $G(D)$  の同じ強連結成分に属するならば、 $x_i$  と  $x_j$  に対し交換操作は適用不可能である。

(証明) 集合  $W$  によって生成される  $G(D)$  の部分グラフに含まれ、かつ  $G(D)$  のどのサイクルにも含まれないすべての辺の集合を  $Z$  とする。グラフ  $[V, E - Z]$ ,  $[V, E' - Z]$  を考える。但し、 $G(\sigma_{x_i, x_j}(D)) = [V, E']$  とする。グラフ  $[V, E - Z]$  と  $[V, E' - Z]$  における独立なサイクルの数をそれぞれ  $\mu_z(D)$ ,  $\mu_z(\sigma_{x_i, x_j}(D))$  と表わす。明らかに、 $\mu(D) = \mu_z(D)$  である。補題4.2の証明の(2)の場合から  $\mu_z(\sigma_{x_i, x_j}(D)) = \mu_z(D) - 1$  である。また、 $G(\sigma_{x_i, x_j}(D))$  において、 $\mu(\sigma_{x_i, x_j}(D))$  個の独立なサイクルのどの集合も  $Z$  のある辺を含むサイクルを含むので、 $\mu(\sigma_{x_i, x_j}(D)) \geq \mu_z(\sigma_{x_i, x_j}(D)) + 1$  である。それ故、

$$\mu(D) = \mu_z(\sigma_{x_i, x_j}(D)) + 1 \leq \mu(\sigma_{x_i, x_j}(D))$$

となる。こうして補題4.3は成り立つ。

(証明終)

[補題4.4] 次の(1)–(3)の1つが成り立つならば、 $x_i$  と  $x_j$  に対し交換操作は適用不可能である。

- (1)  $D(t_i) = D(t_j)$
- (2)  $D(b_i) = D(b_j)$
- (3)  $D(t_i) = D(b_j) \neq 0$

次の(4)–(7)の1つが成り立つならば、 $x_i$  と  $x_j$  に対し交換操作は適用可能である。

- (4)  $D(t_i) = D(b_j) = 0$
- (5)  $D(b_i) = D(t_j)$ ,  $D(t_i) = D(b_j)$
- (6)  $D(b_i) = D(t_j)$ ,  $D(b_j) \notin V_k$
- (7)  $D(t_i) = D(b_j)$ ,  $D(t_j) \notin V_k$

補題4.4の証明は明らかであるから省略する。

提案した端子割当改善の算法の実行を高速化するために、算法のステップ2に

において配線要求 $D$ の2端子 $x_i$ と $x_j$  ( $x = t$ 又は $x = b$ )に対し交換操作が適用可能か否か判定する時に、定理4.2と補題4.1—4.4を使う。補題4.1と4.2を用いると端子割当改善の算法の実際の処理時間がかなり短縮されることを説明する。

補題4.1の条件が満たされている時、 $G_{(k)}$ の独立なサイクル数と $V_k$ によって生成される $G(\sigma_{x_i, j}(D))$ の部分グラフの独立なサイクル数を算出し、これらを比較することにより、 $x_i$ と $x_j$ に対し交換操作が適用可能か否か判定できる。 $G(D)$ が多くの強連結成分からなる場合、この手法の処理時間は、単に $\mu(D)$ と $\mu(\sigma_{x_i, j}(D))$ を算出して、これらを比較することにより $x_i$ と $x_j$ に対し交換操作が適用可能か否か判定する手法の処理時間と比較して、かなり小さくなる。同様に、 $G(D)$ が多くの連結成分からなる時、補題4.2を用いると、実際の処理時間が短縮される。

[例4.4] 図4.6(a)の配線要求 $D$ を考える。 $t_3 \sim t_4$ と仮定し、図4.6(b)に示す部分グラフが上下制約グラフ $G(D)$ の連結成分であると仮定する。図4.6(b)に示す $G_{(k)} = [V_k, E_k]$ は $G(D)$ の強連結成分である。交換操作が2端子 $t_3$ と $t_4$ に対し適用可能かどうかを判定する方法を説明する。 $G_{(k)}$ は辺 $(D(t_3), D(b_3))$ を含み、 $D(t_4) \notin V_k$ 、かつ、 $D(b_4) \notin V_k$ であるから、補題4.2の条件は満たされる。頂点 $D(t_3)$ に対し、補題4.2で定義した集合 $W$ は $\{1, 2, 3, 4\}$ となる。次に、式(4.2)を用いて、 $W$ によって生成される $G(\sigma_{t_3, 4}(D))$ の部分グラフを作る。 $W$ によって生成される $G(D)$ の部分グラフは1個の独立なサイクルを持ち、 $W$ によって生成される $G(\sigma_{t_3, 4}(D))$ の部分グラフは図4.6(c)に示すようにサイクルを持たないので、補題4.2によって交換操作は $t_3$ と $t_4$ に対し適用可能である。この例から分かるように、補題4.2を用いると、 $G(D)$ の部分グラフに注目するだけで交換操作が適用可能か否か判定でき、提案した端子割当改善の算法の実際の処理時間が短縮される。(例終)

#### 4.5 実験結果と考察

本節では、端子割当の算法を第3章で述べた900ゲートのECLゲートアレイに適用した実験結果について述べる。本章で考察した端子割当の算法のプログラムをFORTRANで記述した。本プログラムは約2,400ステートメントであり、その内の大部分が端子割当改善のプログラムである。端子割当のプログラムをGALOP配線プログラム<sup>(5)</sup>に組み込み、GALOPを3.5MIPSの計算機MELCOM-COSMO900II上で実行した。図2.8に示したように、こ

$t_1$	$t_2$	$t_3$	$t_4$	$t_5$	$t_6$	...
2	3	1	4	4	5	...

3	1	2	5	2	6	...
$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	$b_6$	...

(a)

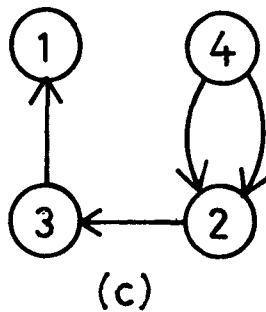
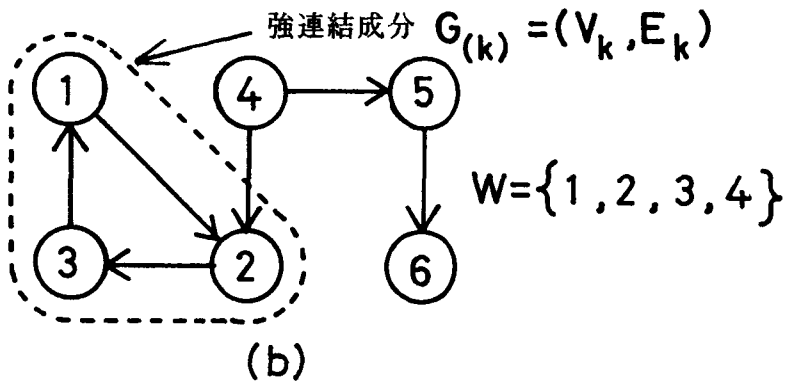


図4.6 補題4.2の適用の説明図，(a) 配線要求D，(b) 上下制約グラフ  $G(D)$ の1つの連結成分，(c)  $W$ によって生成される  $G(\sigma_{1,3,4}(D))$ の部分グラフ



の配線プログラムではグローバル配線の後に、配線長を短くすることを目標とした信号ネットのゲートセル端子への単純な初期割当，提案した算法に基づく端子割当改善，及び“dogleg channel router”<sup>(1)・(2)</sup>と同様の手法に基づいたチャネル配線を，一番下の水平チャネルから順に各水平チャネルに対し行う。このため，端子割当では処理対象の水平チャネルの下側のセル列上のゲートセル端子の中で，既に配線が完了した下側の水平チャネル内の配線とつながる端子は端子割当の対象から除外する。

本プログラムをECLゲートアレイのある回路に適用した結果を表4.1に示す。ここで，チップ上の対向する2セル列間のチャネルに下から順に1，2，…，14と番号を付け， $\mu > 0$ であるチャネルだけを表4.1に記述した。この回路ではチップ上の884ゲートセルを使用している。図4.7は第9番目のチャネルの上下制約グラフの3つの独立なサイクルをもつ強連結成分である。表4.2に示すように，本算法はこの上下制約グラフの全サイクルを解消し，その結果，GALOP配線プログラムはすべての信号ネットを自動配線した。また，表4.1に示すように，上下制約グラフは多くの強連結成分，連結成分からなるため，4.4節で述べた定理4.2や補題4.1—4.4を使うと処理時間がかなり短縮された。この実験結果から，次の2点が実証された。

(1) 実験に用いたECLゲートアレイのように，LSIチップ上の各ゲートセルの多くの端子が論理的に等価である場合には，端子割当改善の算法はかなり短い処理時間で上下制約グラフのサイクルをうまく除去する。

(2) 上下制約グラフが多くの連結成分を持つ時，定理4.2と補題4.1—4.4を用いると算法の実際の処理時間がかなり短縮される。

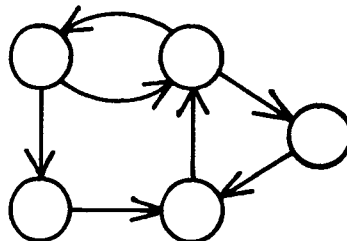


図4.7 上下制約グラフの強連結成分

表4.1 端子割当改善の算法のECLゲートアレイへの適用結果

チャネル 番号	信号 ネット数	算 法 適 用 前				算法 適用後 の $\mu$	処理時間 (秒)	
		E	$\mu$	n	連結成分 の数		(A)	(B)
1	129	78	2	127	54	0	0.5	1.9
3	113	60	2	109	58	0	0.8	28.4
4	82	26	1	81	57	0	0.2	0.5
5	113	69	4	109	51	0	0.9	29.5
6	118	54	1	117	65	0	0.2	0.5
7	97	32	1	96	67	0	0.1	0.7
8	106	58	3	103	57	0	0.7	26.8
9	122	69	11	108	67	0	1.5	61.0
10	111	43	1	110	71	0	0.3	1.4
11	111	54	3	108	63	0	0.8	27.5
12	112	55	3	108	62	0	0.9	28.3
14	114	78	5	107	46	0	0.8	30.5

|E| : 上下制約グラフの辺の数

$\mu$  : 上下制約グラフの独立なサイクルの数

n : 上下制約グラフの強連結成分の数

(A) の場合には定理4.2と補題4.1-4.4を適用したが、(B) の場合にはこれらを適用しなかった。

#### 4.6 結 言

本章では、端子割当問題に対し、まず初期端子割当を行い、次に端子割当を逐次改善することにより最終的な端子割当を決定する算法を提案した。この内の端子割当改善に対し、論理的に等価な2端子間で信号ネットの割当を繰り返し交換することにより上下制約グラフ中のサイクルを減少させる算法を考察した。次に、この算法の高速化のために、端子割当を変更することにより上下制約グラフ中のサイクルの数が減少するか否かの判定が、その部分グラフに着目するだけで

可能であることを理論的に証明した。更に，適用結果により本章で提案した算法が有効であることを実証した。

なお，本章の内容は，文献(6)に基づいている。

#### 参考文献

- (1) G.Persky, D.N.Deutsch and D.G.Schweikert: "LTX-A minicomputer-based system for automatic LSI layout", J. Des. Autom. & Fault-Tolerant Comput., vol. 1, no. 3, pp. 217-255(1977).
- (2) D.N.Deutsch: "A dogleg channel router", Proc. 13th DA Conf., pp. 425-433(1976).
- (3) C.Berge: Graphs and Hypergraphs, Amsterdam, The Netherlands: North-Holland(1973).
- (4) R.E.Tarjan: "Depth-first search and linear graph algorithms", SIAM J. Comput., vol. 1, no. 2, pp. 146-160(1972).
- (5) 寺井, 野田, 佐藤, 八原: "種々のチップ構造のゲートアレイに適用可能な配線プログラム", 情報処理学会論文誌, vol. 25, no. 3, pp. 357-364 (1984).
- (6) M.Terai: "A method of improving the terminal assignment in the channel routing for gate arrays", IEEE Trans. Comput.-Aided Des. Integrated Circuits & Syst., vol. CAD-4, no. 3, pp. 329-336(1985).

## 第5章 幹線分割手法

### 5.1 緒 言

VLSIの配線設計におけるチャンネル配線とは、上下辺上に位置固定の端子を持つ矩形の水平チャンネル内の配線をいう。Hashimoto-Stevenec<sup>(1)</sup>以来、多くのチャンネル配線の手法が提案されている<sup>(2)-(6)</sup>。これらの内、文献(1)-(3)、(6)で提案された手法では、まず全配線の幹線(配線径路の水平方向成分)の水平方向の区間を決め、次にそれらの全幹線を水平トラック上に配置することによりチャンネルの配線を行う。このため、幹線相互間の上下制約グラフに長い有向道が含まれる場合には、有向道を構成する幹線の1つを折り曲げる(幹線分割することにより、配線に使用される水平トラック数を減少させることができる。最近、幹線の水平方向の区間をあらかじめ決めないでチャンネル配線を行う手法が提案されているが<sup>(4)・(5)</sup>、特に文献(5)ではいわゆる"difficult channel"<sup>(2)</sup>を最大幹線密度(density)<sup>(2)</sup>に等しい水平トラック数で配線できたと報告している。一方、文献(4)の手法により"difficult channel"を配線すると、文献(2)、(3)、(6)の手法で配線する場合より数倍から10倍以上の処理時間を要し、幹線分割(ドッグレッグ)<sup>(2)</sup>の数も2倍から数倍になり、さらに文献(5)の手法は文献(4)の手法より多くの処理時間を要すると報告している。実際のVLSIのチャンネル配線では、文献(2)の手法でも最大幹線密度に等しいか又はそれに近い水平トラック数で配線できることが多い。従って、通常は文献(2)、(3)、(6)の手法で高速にチャンネル配線を行い、配線に使用された水平トラック数と最大幹線密度の差が大きいチャンネルに対してだけ、再度、幹線分割とチャンネル配線を行う方が、処理時間と配線結果におけるドッグレッグの数を考慮した場合、より実用的と考えられる。

そこで本章では、チャンネル配線は文献(2)、(3)、(6)のように幹線の水平方向の区間を決めた後、これらの全幹線を水平トラック上に配置することにより行われると仮定し、その下で幹線の水平方向の区間を決める段階における幹線分割問題について考察する。幹線分割問題とは、チャンネル配線において使用される水平トラック数を減少させるために、幹線の折り曲げ位置を決定する問題である。幹線分割がチャンネル配線において使用される水平トラック数を減少させる効果を持つかどうかは、どの幹線をどの点で幹線分割するかにより決まる<sup>(7)</sup>。しかし、どの幹線をどの点で幹線分割すれば有効であるかについて述べた文献は少なく、それを判別する実用的な方法はまだ報告されていない。例えば、文献(7)で

は、上下制約グラフ中のある有向辺連鎖に着目した場合、それを構成する幹線の内のどの幹線をどの点で幹線分割するのが有効である可能性が高いかについて述べているが、どの有向辺連鎖を構成する幹線を幹線分割すれば有効であるかについては述べていない。

本章では、まずチャンネル配線の全幹線の集合 $T$ に対し、ある関数 $W(T)$ を定義し、 $W(T)$ の値が $T$ の全幹線を水平トラック上に配置してチャンネル配線を行うのに十分な水平トラック数であることを証明する。次に幹線分割問題に対し、幹線相互間の上下制約グラフ中の有向辺連鎖の長さが増加せず、かつ $W(T)$ の値が減少するように、幹線分割を行う算法を提案する。更に提案した算法を8,000ゲートCMOSゲートアレイの配線設計に適用した結果について述べる。

## 5.2 問題の定式化と関数 $W(T)$ の定義

本節では、本章で扱う幹線分割問題を正確に定式化するための用語の定義と関数 $W(T)$ の定義を行う。

チャンネルは図5.1に示すように長方形の配線領域で、その上下辺上には位置が固定されたセルの端子が並んでいる。上下辺上の全端子は垂直方向配線格子線上にある。チャンネルの左右辺は、通常、他のチャンネルと境を接しているため、左右辺上にはその位置が固定されていない端子を置く。左右辺上の端子の位置はチャンネル配線により決定される。チャンネルの4辺上の各端子には、信号ネット番号

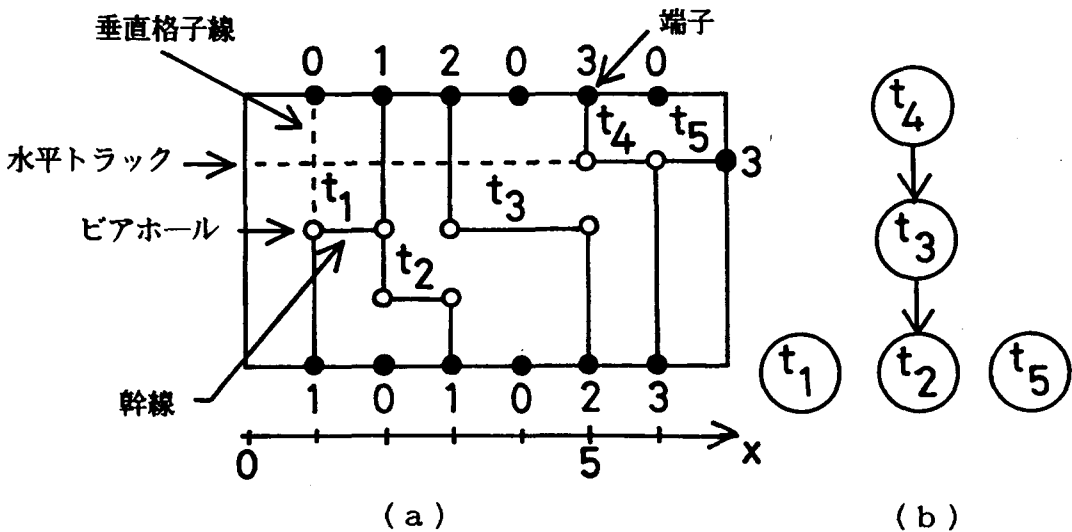


図5.1 (a) 配線要求と幹線集合 $T$  (b) 上下制約グラフ $G(T)$

0, 1, 2, 3, …の内の1つの番号が割り当てられている。番号0を持つ端子は空き端子である。配線要求はチャンネルの4辺上の端子とその信号ネット番号により与えられる。配線要求が与えられた時、チャンネル配線では0以外の同じ信号ネット番号  $i$  を持つ全端子をチャンネル内で結線する。チャンネルは2層からなり、第1層に水平方向の配線を第2層に垂直方向の配線を置き、ビアホールによって同一信号ネットの2つの層の配線が結合される。

チャンネルの上下辺に平行な  $x$  軸を考える。本章で扱うチャンネル配線の過程は次の2つのステップからなる<sup>(7)</sup>。

(1) 各信号ネットに対して、幹線の  $x$  方向の区間（例えば図5.1の信号ネット1に対して  $x$  方向の区間  $[1, 2]$  と  $[2, 3]$  の2本の幹線）を決める。

(2) 全幹線を幹線間及び垂直方向配線間が重ならないように水平トラック上に配置する。

ある信号ネットに対する2端子サブネット (2-terminal subnet)<sup>(2)</sup> は、その信号ネットの端子をそれらの  $x$  座標値に基づいて順序づけし、元の信号ネットを連続する隣接2端子のペアに分解することにより得られる。文献(2), (3), (6)のチャンネル配線の算法は、全信号ネットに対する2端子サブネットにそれぞれ1本の幹線を割り当て、これらの全幹線を水平トラック上に配置する手法であることから、与えられた配線要求の全信号ネットに対する2端子サブネットにそれぞれ1本の幹線を割り当てた時の全幹線の集合を  $T$ 、 $T$  の全幹線を水平トラック上へ配置してチャンネル配線を行うのに必要十分な最小の水平トラック数を  $M(T)$  とし、本章で扱う幹線分割問題を次のように定式化する。

[問題]  $T$  が与えられた時、 $M(T')$  を最小にするような、 $T$  のいくつかの幹線を幹線分割した時の全幹線の集合  $T'$  を求めよ。 (問題終)

この場合、チャンネル配線では幹線分割問題により得られた幹線集合  $T'$  の全幹線を水平トラック上に配置する。

以下では、幹線分割を行っているある時点での幹線集合  $T = \{t_1, t_2, \dots, t_N\}$ 、即ち与えられた配線要求の全信号ネットに対する2端子サブネットにそれぞれ1本の幹線を割り当てた時又はその内のいくつかの幹線を幹線分割した時の全幹線の集合を考える。幹線  $t_i \in T$  の  $x$  方向の区間を  $[x_L(t_i), x_R(t_i)]$  と表わす。 $U(t_i), L(t_i)$  をそれぞれ  $t_i$  の両端点の内、垂直方向の配線を介して上辺、下辺上の端子と接続する端点の  $x$  座標値の集合とし、 $P(t_i)$  を  $t_i$  の両端点の内、垂直方向配線を介して上下辺上の端子と接続せず、左右辺上の端子とも接続しない端点（幹線分割によって生じた端点）の  $x$  座標値の集

合とする。 $U(t_i) \cap L(t_j)$ は幹線 $t_j$ の両端点の内、垂直方向の配線を介して上下両辺上の端子と接続する端点のx座標値の集合を表わし、配線可能であるためには、少なくともこのx座標値の点で別の幹線 $t_i$ を幹線分割しないことが必要である。即ち、配線可能であるためには、少なくともすべての相異なる幹線 $t_i, t_j \in T$ に対し、

$$P(t_i) \cap U(t_j) \cap L(t_j) = \phi$$

である必要がある。同一x座標値の点でいくつかの幹線を幹線分割することは、通常、配線に使用される水平トラック数を減らす効果を持たないことが多い。そこで、上下制約グラフの定義を簡単にするため、すべての相異なる2つの幹線 $t_i, t_j \in T$ に対し、

$$P(t_i) \cap P(t_j) = \phi$$

と制限する。この制限の下で行われる幹線分割は5.5節の実験結果で示すように有効である。まず、上下制約グラフを次のように定義する。

[定義5.1] 幹線集合 $T$ に対し、次の有向グラフ $G(T) = [V, E]$ を上下制約グラフと呼ぶ。頂点集合 $V$ の各頂点は $T$ の幹線と一対一に対応し(従って以下では $V = \{t_1, t_2, \dots, t_N\}$ と表わす)、有向辺 $e_k = (t_i, t_j)$ が有向辺集合 $E = \{e_1, e_2, \dots, e_E\}$ の要素であるのは、 $t_i$ と $t_j$ が相異なる信号ネットの幹線であり、かつ、

$$(U(t_i) \cap L(t_j)) \cup (U(t_i) \cap P(t_j)) \cup (P(t_i) \cap L(t_j)) \neq \phi$$

即ち、幹線 $t_i$ を幹線 $t_j$ より上の水平トラックに配置しなければ配線できない時かつその時に限る。(定義終)

上下制約グラフ中のサイクルはあらかじめ文献(8)の手法により除去され、本章で扱う上下制約グラフはサイクルを持たないと仮定する。

次に、幹線集合 $T$ に対して関数 $W(T)$ を定義する。上下制約グラフ $G(T)$ における頂点 $t_i$ から出る有向道の最大の長さを $r_T(t_i)$ で表わす。但し、頂点 $t_i$ を始点とする有向辺がない時、 $r_T(t_i) = 0$ とする。 $G(T)$ の全頂点に対する $r_T$ の値は深さ優先探索(depth-first search)の手法<sup>(9)</sup>を用いることにより、高々 $O(|V| + |E|)$ の手数で算出される。幹線集合 $T$ の $r_T$ による直和分割 $\{T_0, T_1, \dots, T_h\}$ を次のように定義する。

$$h = \max_{t_i \in T} r_T(t_i)$$

$$T_i = \{t_k \mid r_T(t_k) = i, t_k \in T\}$$

Tの最大幹線密度と関数W(T)を次のように定義する。

[定義5.2] 幹線集合Tに対し, x座標値がxである垂直格子線を横切るか又はその上に端点を持つTの幹線の総数をC(T, x)と表わす。但し, 相異なる幹線 $t_i, t_j \in T$ が $x_R(t_i) = x_L(t_j) = x$ かつ $t_i$ と $t_j$ が同一信号ネットの幹線であるならば, C(T, x)において $t_i$ と $t_j$ は同一幹線とみなす。この時, Tの最大幹線密度D(T)を

$$D(T) = \max_x C(T, x)$$

と定義する。また,  $C(T, x) = D(T)$ である垂直格子線のx座標値の集合をTのスパンといい, S(T)と表わす。D(T<sub>i</sub>), S(T<sub>i</sub>)も同様に定義する。更に関数W(T)を次式で定義する。

$$W(T) = \sum_{i=0}^h D(T_i) \quad (\text{定義終})$$

{T<sub>0</sub>, T<sub>1</sub>, ..., T<sub>h</sub>}はTの直和分割であることから, 次の補題5.1が成り立つ。

[補題5.1] 与えられた配線要求の幹線集合Tに対し, 次式が成り立つ。

$$W(T) \geq D(T) \quad (\text{補題終})$$

[例5.1] 図5.1(a)の配線要求の全ネットに対する2端子サブネットにそれぞれ1本の幹線を割り当てた時の全幹線の集合Tは次のようになる。

$$T = \{t_1, t_2, t_3, t_4, t_5\}$$

上下制約グラフG(T)を図5.1(b)に示す。G(T)の各頂点に対し,  $r_T(t_1) = 0$ ,  $r_T(t_2) = 0$ ,  $r_T(t_3) = 1$ ,  $r_T(t_4) = 2$ ,  $r_T(t_5) = 0$ である。従って, Tは $r_T$ により{T<sub>0</sub>, T<sub>1</sub>, T<sub>2</sub>}に直和分割され, T<sub>0</sub> = {t<sub>1</sub>, t<sub>2</sub>, t<sub>5</sub>}, T<sub>1</sub> = {t<sub>3</sub>}, T<sub>2</sub> = {t<sub>4</sub>}である。また, 最大幹線密度, スパン及びW(T)は次のようになる。

$$D(T) = 2, \quad S(T) = \{3, 5\}.$$

$$D(T_0) = 1, \quad S(T_0) = \{1, 2, 3, 6\}.$$

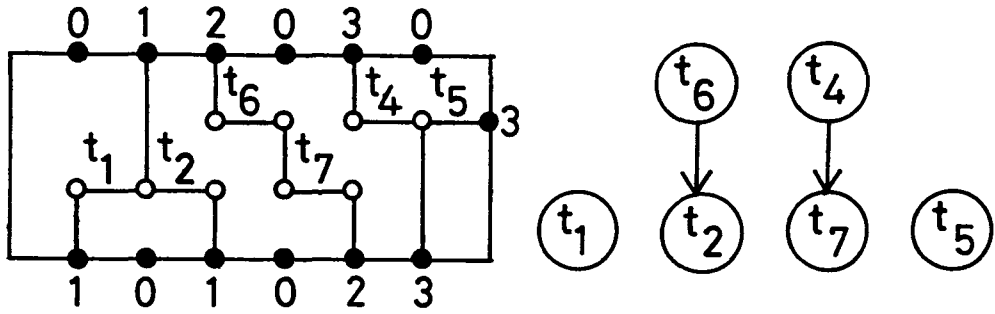
$$D(T_1) = 1, \quad S(T_1) = \{3, 4, 5\}.$$

$$D(T_2) = 1, \quad S(T_2) = \{5, 6\}.$$

$$W(T) = D(T_0) + D(T_1) + D(T_2) = 3.$$

次に幹線 $t_3 \in T$ を図5.2(a)のように幹線分割した時の全幹線の集合をT'とする。図5.1(a)の配線要求を配線するためには2本以上の水平トラックが必要であり, M(T) = 3, M(T') = 2である。従って, 幹線分割問題はTに対しT'を求め





(a)  $T$ の幹線  $t_3$ を分割した後の幹線集合  $T'$   
 (b) 上下制約グラフ  $G(T')$

る問題である。

(例終)

関数  $W(T)$  について次の定理が成り立つ。

[定理5.1] 与えられた配線要求の幹線集合  $T$  に対し、高々  $W(T)$  本の水平トラックがあれば、 $T$  の全幹線がチャンネル配線可能である。

(証明)  $T$  の全幹線を、幹線間に重なりがなくかつ上下制約グラフ  $G(T) = [V, E]$  で表現された幹線間の上下制約関係 (即ち、有向辺  $(t_i, t_j) \in E$  ならば幹線  $t_i$  を幹線  $t_j$  より上の水平トラック上に配置しなければならないという関係) をすべて満たすように、水平トラック上に配置すれば、チャンネル配線は完了する。 $T_i$  の全幹線を幹線間に重なりがないように水平トラック上に配置するのに必要十分な水平トラック数は  $D(T_i)$  本である<sup>(1)</sup> ことから、チャンネルの下から順に  $D(T_0)$  本の水平トラック上に  $T_0$  の全幹線を配置し、次にその上の  $D(T_1)$  本の水平トラック上に  $T_1$  の全幹線を配置し、以下同様にして、最後にチャンネルの最も上の  $D(T_h)$  本の水平トラック上に  $T_h$  の全幹線を配置するという方法で、 $T$  の全幹線を幹線間に重なりがないように合計  $W(T)$  本の水平トラック上に配置する。相異なる幹線  $t_i, t_j \in T$  に対して、 $(t_i, t_j) \in E$  ならば、 $r_T(t_i) > r_T(t_j)$  であるから、この幹線の水平トラック上への配置においては  $t_i$  は  $t_j$  よりも上の水平トラック上に配置されている。この幹線の配置は上下制約グラフで表現された幹線間の上下制約関係をすべて満たしてい

る。故に、定理5.1は成り立つ。

(証明終)

### 5.3 幹線分割の算法

本節では、幹線分割問題に対する算法を提案する。

与えられた配線要求の幹線集合 $T$ に対し、 $M(T)$ を実用的な処理時間で算出する方法は見つかっていないが、補題5.1と定理5.1より $M(T)$ は次式を満たす。

$$D(T) \leq M(T) \leq W(T)$$

幹線分割により $D(T)$ を小さくできないので、 $M(T)$ が $D(T)$ に等しい場合は幹線分割により $M(T)$ を小さくすることができない。一方、 $M(T)$ が $D(T)$ より大きい場合、幹線分割を行い $W(T)$ を小さくすることにより結果として $M(T)$ も小さくなることが考えられる。そこで、本章では幹線分割問題を $W(T)$ を最適化する問題として扱い、幹線分割問題に対し、 $T$ の任意の幹線を後で定義する分割条件A、B、Cを満たすように反復的に幹線分割する算法を提案する。幹線集合 $T$ に対し、幹線 $t_i \in T$ を $x$ 座標値が $x_D$ の点で2本の幹線 $t_{i1}$ 、 $t_{i2}$ に幹線分割した時の全幹線の集合を $T'$ とする。但し、幹線の分割点はチャンネル内にあり、 $x_L(t_i) - \delta \leq x_D \leq x_R(t_i) + \delta$  ( $\delta$ は定数である)を満たすものとする。 $T$ と $T'$ の $r_T$ と $r_{T'}$ による直和分割をそれぞれ $\{T_0, T_1, \dots, T_h\}$ 、 $\{T'_0, T'_1, \dots, T'_h\}$ として、分割条件を次のように定義する。

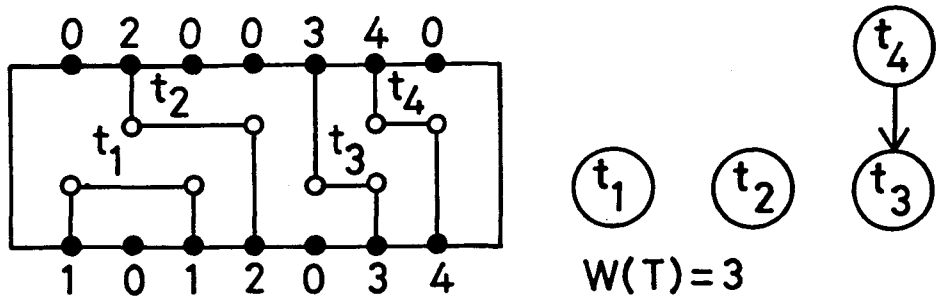
[分割条件A] (1)  $W(T) > W(T')$ 、又は(2)  $W(T) = W(T')$ 、かつ、ある $j$  ( $0 \leq j \leq h$ ) に対し、 $D(T_j) = D(T'_j)$ 、 $S(T_j) \supset S(T'_j)$ かつ $|T_j| > |T'_j|$ 。

[分割条件B]  $G(T')$ はサイクルを持たない。

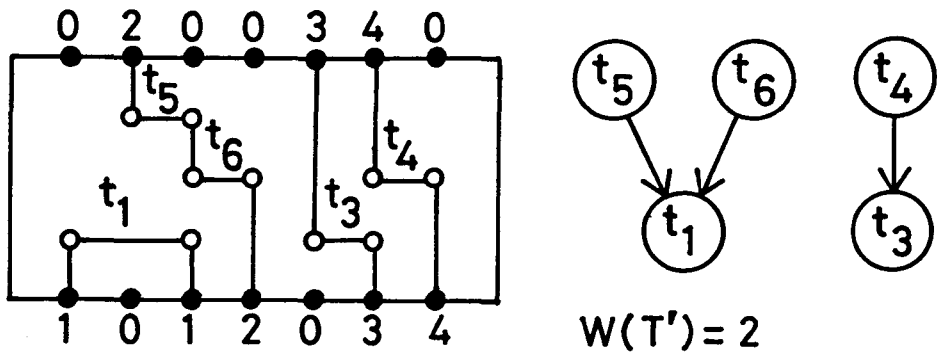
[分割条件C] すべての $t_k \in T$  ( $k \neq i$ ) に対し、 $r_T(t_k) \geq r_{T'}(t_k)$ 。更に $r_T(t_i) \geq r_{T'}(t_{i1})$ 、 $r_T(t_i) \geq r_{T'}(t_{i2})$ 。(条件終)

分割条件A(2)及びCを設定した理由について考察する。後述の例5.3で示すように、同一の $j$ に対して分割条件A(2)を満たす幹線分割を繰り返すことにより結果として $S(T'_j) = \phi$ となれば $W(T) > W(T')$ となるから、分割条件A(2)を設定した。また、分割条件A、Bだけを考慮するならば、例えば図5.3(a)の幹線 $t_2$ を図5.3(b)のように幹線分割した場合のように $W(T) > W(T')$ であるが、実際には配線に使用される水平トラック数を減少させる効果を持たない幹線分割が行われることがある。このことから、分割条件Cを設定した。

[例5.2] 図5.1(a)の幹線集合 $T$ に対し、幹線 $t_3$ を図5.2(a)のように $t_6$ と $t_7$ に幹線分割した幹線集合を



(a) 幹線集合 $T$



(b)  $T$ の幹線 $t_2$ を分割した後の幹線集合 $T'$

図5.3 有効でない幹線分割の例

$T' = \{t_1, t_2, t_4, t_5, t_6, t_7\}$ とする。上下制約グラフ $G(T')$ を図5.2(b)に示す。 $r_{T'}(t_1) = 0$ ,  $r_{T'}(t_2) = 0$ ,  $r_{T'}(t_4) = 1$ ,  $r_{T'}(t_5) = 0$ ,  $r_{T'}(t_6) = 1$ 及び $r_{T'}(t_7) = 0$ である。従って、 $T'$ の $r_{T'}$ による直和分割は $\{T'_0, T'_1\}$ で、 $T'_0 = \{t_1, t_2, t_5, t_7\}$ ,  $T'_1 = \{t_4, t_6\}$ ,  $W(T') = 2$ となる。また、 $G(T')$ はサイクルを持たないことから、この幹線分割は分割条件A, BかつCを満たす。実際、 $T$ に提案した幹線分割の算法を適用すると $T'$ が得られる。(例終)

## 5.4 算法の高速化

本節では、提案した幹線分割の算法の実際の処理時間を短縮するために、ある幹線分割が分割条件BかつCを満たすための必要十分条件を考察する。

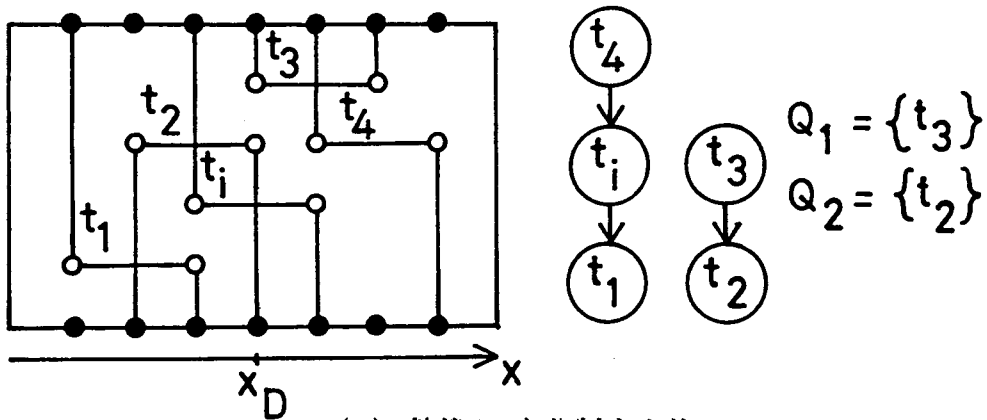
以下では、幹線集合 $T$ に対し、幹線 $t_i \in T$ を図5.4のように $x$ 座標値が $x_D$ の点で2本の幹線 $t_{i1}$ ,  $t_{i2}$ に幹線分割した時の全幹線の集合を $T'$ とする。上下制約グラフ $G(T)$ における頂点 $t_k$ の祖先<sup>(10)</sup>である頂点の集合( $t_k$ 自身は含まれない)を $\Gamma_T(t_k)$ で表わす。 $G(T)$ の頂点集合 $V$ の部分集合 $Q_1(x_D)$ ,  $Q_2(x_D)$ ,  $Z_1$ ,  $Z_2$ を次のように定義すると、 $Z_1$ ,  $Z_2$ ,  $\Gamma_T(t_i)$ ,  $\{t_i\}$ は $V$ の直和分割となる。

$$\begin{aligned} Q_1(x_D) &= \{t_k \mid x_D \in U(t_k), t_k \in T\} \\ Q_2(x_D) &= \{t_k \mid x_D \in L(t_k), t_k \in T\} \quad (5.1) \\ Z_1 &= Q_1(x_D) \cup \left( \bigcup_{t_k \in Q_1(x_D)} \Gamma_T(t_k) \right) - (\{t_i\} \cup \Gamma_T(t_i)) \\ Z_2 &= V - (\{t_i\} \cup \Gamma_T(t_i) \cup Q_1(x_D) \\ &\quad \cup \left( \bigcup_{t_k \in Q_1(x_D)} \Gamma_T(t_k) \right)) \end{aligned}$$

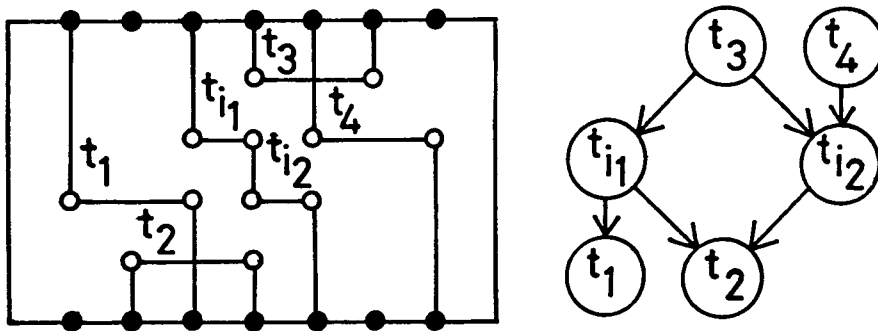
上下制約グラフ $G(T) = [V, E]$ と $G(T') = [V', E']$ の間に次式が成り立つ(図5.4参照)。但し、 $G(T)$ において、頂点 $t_i$ につながるすべての有向辺の集合を $\text{Inc}(t_i)$ と表わす。

$$\begin{aligned} V' &= V \cup \{t_{i1}, t_{i2}\} - \{t_i\} \\ E' &= E - \text{Inc}(t_i) \\ &\quad \cup \left\{ \begin{aligned} &U \{ (t_{i1}, t_k) \mid ((t_i, t_k) \in E, \\ &x_L(t_i) \cap (x_L(t_k) \cup x_R(t_k)) \neq \phi) \text{ 又は } t_k \in Q_2(x_D) \} \\ &U \{ (t_k, t_{i1}) \mid ((t_k, t_i) \in E, \\ &x_L(t_i) \cap (x_L(t_k) \cup x_R(t_k)) \neq \phi) \text{ 又は } t_k \in Q_1(x_D) \} \\ &U \{ (t_{i2}, t_k) \mid ((t_i, t_k) \in E, \\ &x_R(t_i) \cap (x_L(t_k) \cup x_R(t_k)) \neq \phi) \text{ 又は } t_k \in Q_2(x_D) \} \\ &U \{ (t_k, t_{i2}) \mid ((t_k, t_i) \in E, \\ &x_R(t_i) \cap (x_L(t_k) \cup x_R(t_k)) \neq \phi) \text{ 又は } t_k \in Q_1(x_D) \} \end{aligned} \right\} \quad (5.2) \end{aligned}$$

次の補題5.2, 5.3, 5.4で上下制約グラフの頂点の $r_T$ と $r_{T'}$ の関係を示し、これらを用いて、 $T$ と $T'$ に対し分割条件BかつCが成り立つための必要十分条件を与える定理5.2を証明する。



(a) 幹線  $t_i$  を分割する前



(b) 幹線  $t_i$  を分割した後

図5.4 幹線分割と上下制約グラフの例

[補題5.2] 上下制約グラフ  $G(T')$  がサイクルを持たないとする。この時、すべての  $t_k \in Z_2$  に対し、 $r_T(t_k) = r_{T'}(t_k)$  である。

(証明)  $Z_2$  の定義と式(5.2)より、 $G(T)$  と  $G(T')$  において、共に  $Z_2$  のある頂点を始点とし、終点は  $Z_2$  に含まれない有向辺はない。式(5.2)より、 $Z_2$  によって生成される  $G(T)$  の部分グラフと  $Z_2$  によって生成される  $G(T')$  の部分グラフは同形である。故に補題5.2は成り立つ。 (証明終)

[補題5.3] 上下制約グラフ  $G(T')$  がサイクルを持たないとする。ある  $t_f \in Q_2(x_D)$  に対し、 $r_T(t_i) \leq r_T(t_f)$  の時、 $r_{T'}(t_{i1}) = r_{T'}(t_{i2}) = r_T(t_f) + 1$  である。これ以外の場合は、 $r_T(t_i) = r_{T'}(t_{i1})$ 、 $r_T(t_i) \geq r_{T'}(t_{i2})$  又は  $r_T(t_i) \geq r_{T'}(t_{i1})$ 、 $r_T(t_i) = r_{T'}(t_{i2})$  である。

(証明) 式(5.2)と $G(T)$ ,  $G(T')$ がサイクルを持たないことから, すべての  $t_k \in \Gamma_T(t_i) \cup Z_1$  に対し,  $(t_i, t_k) \notin E$ ,  $(t_{i1}, t_k) \notin E'$ ,  $(t_{i2}, t_k) \notin E'$  である。式(5.2)より  $t_k \in Z_2 - Q_2(x_D)$  に対し  $(t_{i1}, t_k) \in E'$  又は  $(t_{i2}, t_k) \in E'$  であるのは,  $(t_i, t_k) \in E$  である時かつその時に限る。また,  $G(T)$ ,  $G(T')$ がサイクルを持たないことから  $Q_2(x_D) \subseteq Z_2$  であり, すべての  $t_f \in Q_2(x_D)$  に対し,  $(t_i, t_f) \in E$  か否かに拘らず  $(t_{i1}, t_f) \in E'$  かつ  $(t_{i2}, t_f) \in E'$  である。従って, 補題5.2より補題5.3は成り立つ。 (証明終)

[補題5.4] 上下制約グラフ $G(T')$ はサイクルを持たないとする。この時, すべての  $t_k \in Z_1$  に対し,  $r_T(t_k) \leq r_{T'}(t_k)$  である。但し, すべての  $t_k \in Z_1$  に対し  $r_T(t_k) = r_{T'}(t_k)$  であるのは, すべての  $t_u \in Q_1(x_D) \cap Z_1$ ,  $t_f \in Q_2(x_D)$  に対し, 次の条件(i)又は(ii) ( $Q_2(x_D) = \phi$  の時は  $r_T(t_i) < r_T(t_u)$ ) が成り立つ時, かつその時に限る。

(i)  $r_T(t_i) \leq r_T(t_f)$ ,  $r_T(t_f) + 1 < r_T(t_u)$

(ii)  $r_T(t_f) < r_T(t_i) < r_T(t_u)$

(証明) すべての  $t_k \in Z_1$ ,  $t_m \in \Gamma_T(t_i)$  に対し,  $(t_k, t_m) \notin E$  であり, 式(5.2)より  $(t_k, t_m) \notin E'$  である。式(5.2)より  $t_k \in Z_1$ ,  $t_m \in Z_2$  に対し,  $(t_k, t_m) \in E'$  であるのは  $(t_k, t_m) \in E$  である時かつその時に限る。 $t_k \in Z_1$  に対し  $(t_k, t_{i1}) \in E'$  又は  $(t_k, t_{i2}) \in E'$  であるのは  $t_k \in Q_1(x_D)$  の時かつその時に限り,  $t_k \in Z_1$  ならば  $(t_k, t_i) \notin E$  である。式(5.2)より  $Z_1$  によって生成される  $G(T)$  の部分グラフと  $Z_1$  によって生成される  $G(T')$  の部分グラフは同形である。従って, 補題5.2よりすべての  $t_k \in Z_1$  に対し  $r_T(t_k) \leq r_{T'}(t_k)$  である。すべての  $t_k \in Z_1$  に対し等号が成り立つのは, すべての  $t_u \in Q_1(x_D) \cap Z_1$  に対し  $r_T(t_u) > r_{T'}(t_{i1})$  かつ  $r_T(t_u) > r_{T'}(t_{i2})$  の時かつその時に限る。これは, 補題5.3より補題5.4の条件(i), (ii)と等価である。 (証明終)

[定理5.2] 幹線集合 $T$ に対し, 幹線  $t_i \in T$  を  $x$  座標値が  $x_D$  の点で, 2本の幹線  $t_{i1}$ ,  $t_{i2}$  に幹線分割した時の全幹線の集合を  $T'$  とする。 $T$  と  $T'$  が分割条件BかつCを満たすための必要かつ十分条件は式(5.1)で定義される  $Q_1(x_D)$ ,  $Q_2(x_D)$  に対し, 次の条件(i)かつ(ii)を満たすことである。

(i) すべての  $t_u \in Q_1(x_D)$  に対し  $r_T(t_i) < r_T(t_u)$

(ii) すべての  $t_f \in Q_2(x_D)$  に対し  $r_T(t_i) > r_T(t_f)$

(証明) 条件(i)又は(ii)が満たされないと仮定すれば, 補題5.3,

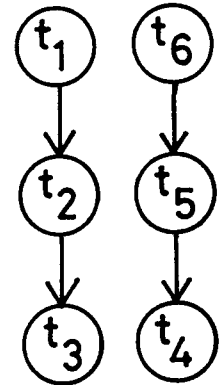
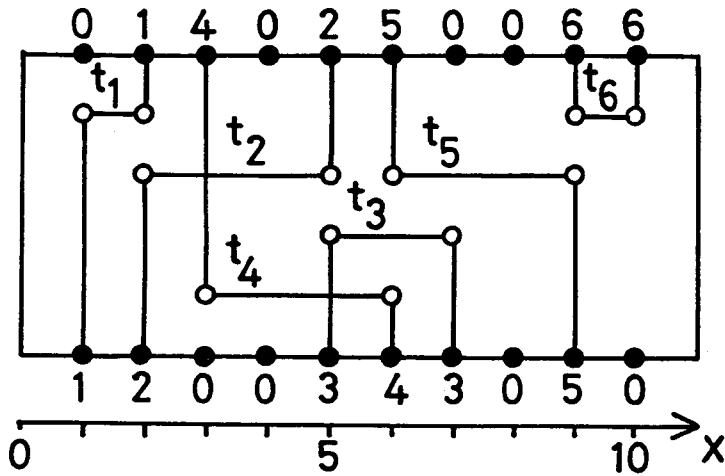
5.4より分割条件Cが満たされない。

逆に、条件(i)かつ(ii)が満たされる仮定する。式(5.2)より $V - \{t_i\}$ によって生成される $G(T)$ の部分グラフと $V' - \{t_{i1}, t_{i2}\}$ によって生成される $G(T')$ の部分グラフは同形である。 $G(T)$ がサイクルを持たないことから、 $G(T')$ がサイクルを持つとすると、そのサイクルはある $t_u \in Q_1(x_D)$ 、 $t_f \in Q_2(x_D)$ に対する有向辺 $(t_u, t_{i1})$ 、 $(t_u, t_{i2})$ 、 $(t_{i1}, t_f)$ 、 $(t_{i2}, t_f)$ の内のいくつかを含む。この時、式(5.2)より $G(T)$ においても $t_i$ から出て $t_u$ に至る有向道、 $t_f$ から出て $t_i$ に至る有向道又は $t_f$ から出て $t_u$ に至る有向道があり、条件(i)又は(ii)と矛盾する。故に、分割条件Bは満たされる。次に、すべての $t_n \in V - (\{t_i\} \cup \Gamma_T(t_i))$ に対し、補題5.2、5.4より $r_T(t_n) = r_{T'}(t_n)$ であり、補題5.3より $r_T(t_i) \geq r_{T'}(t_{i1})$ 、 $r_T(t_i) \geq r_{T'}(t_{i2})$ である。 $t_k \in \Gamma_T(t_i)$ 、 $t_n \in V - (\{t_i\} \cup \Gamma_T(t_i))$ に対し、式(5.2)より $(t_k, t_n) \in E'$ であるのは $(t_k, t_n) \in E$ である時かつその時に限る。式(5.2)より $\Gamma_T(t_i)$ によって生成される $G(T)$ の部分グラフは $\Gamma_T(t_i)$ によって生成される $G(T')$ の部分グラフと同形である。すべての $t_k \in \Gamma_T(t_i)$ に対し $r_T(t_k) > r_{T'}(t_i)$ である。従って、すべての $t_k \in \Gamma_T(t_i)$ に対し $r_T(t_k) \geq r_{T'}(t_k)$ である。故に、分割条件Cは満たされる。(証明終)

提案した幹線分割の算法において、幹線 $t_i \in T$ を $x$ 座標値が $x_D$ の点で幹線分割する場合、分割条件BかつCが満たされるか否かの判定に定理5.2を用いる。この判定を行うには、上下制約グラフ $G(T)$ を作成し、 $\{t_i\} \cup Q_1(x_D) \cup Q_2(x_D)$ の全頂点の $r_T$ を算出すればよいだけである。このため、定理5.2を用いれば、分割条件BかつCが満たされるか否かその定義通り判定する場合に比べて、幹線分割の算法の実際の処理時間が短縮されることが期待できる。次に上下制約グラフの各頂点の $r_T$ と $r_{T'}$ の関係について述べる。各頂点の $r_T$ と $r_{T'}$ は、幹線分割が分割条件Aを満たすか否かの判定に必要である。分割条件BかつCを満たす場合、補題5.2と5.4より $G(T)$ と $G(T')$ において、 $V - (\{t_i\} \cup \Gamma_T(t_i))$ の各頂点に対する $r_T$ と $r_{T'}$ は等しい。従って、 $G(T)$ の全頂点の $r_T$ を算出した後、 $G(T')$ の全頂点の $r_{T'}$ を算出する際、 $\{t_{i1}, t_{i2}\} \cup \Gamma_T(t_i)$ の全頂点の $r_{T'}$ だけを算出すればよい。

[例5.3] 図5.5(a)の配線要求の全信号ネットに対する2端子サブネットにそれぞれ1本の幹線を割り当てた時の全幹線の集合 $T$ は次のようになる。

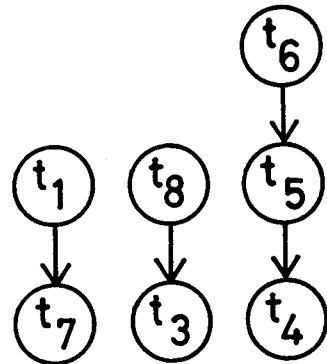
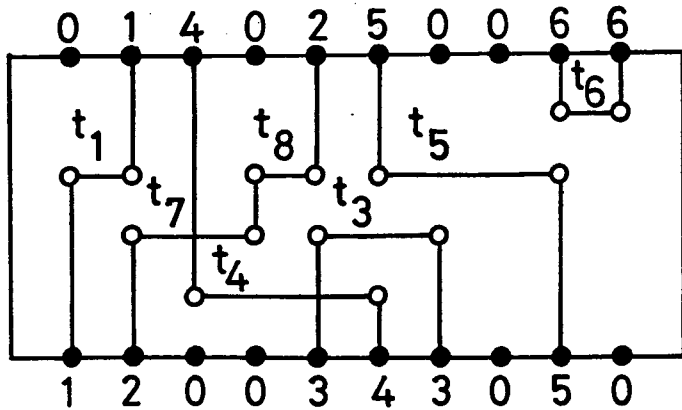
$$T = \{t_1, t_2, t_3, t_4, t_5, t_6\}$$



(a)

(b)

図5.5 (a) 配線要求と幹線集合T (b) 上下制約グラフG(T)



(a)

(b)

図5.6 (a) Tの幹線 $t_2$ を分割した後の幹線集合 $T'$   
(b) 上下制約グラフ $G(T')$



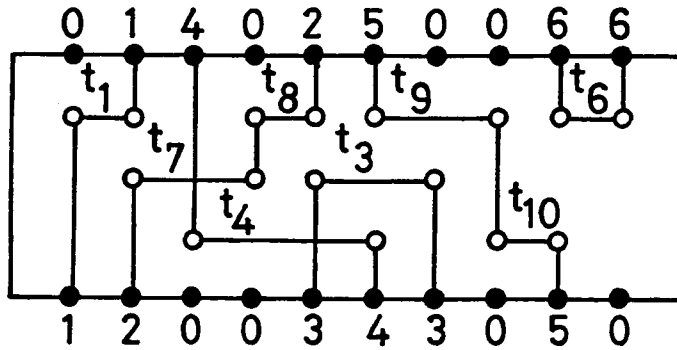


図5.7 幹線分割の算法を適用した後の幹線集合

上下制約グラフ  $G(T)$  を図5.5(b)に示す。幹線  $t_2 \in T$  を  $x$  座標値が3の点で幹線分割すると、 $r_T(t_4) < r_T(t_2)$  であるから定理5.2により分割条件B又はCが満たされない。幹線  $t_2$  を図5.6(a)のように  $x$  座標値が4の点で  $t_7$  と  $t_8$  に幹線分割して得られる幹線集合を

$$T' = \{t_1, t_3, t_4, t_5, t_6, t_7, t_8\}$$

とすると、定理5.2より、この幹線分割は分割条件BかつCを満たす。次に、 $G(T)$  において、 $r_T(t_1) = 2$ 、 $r_T(t_2) = 1$ 、 $r_T(t_3) = 0$ 、 $r_T(t_4) = 0$ 、 $r_T(t_5) = 1$ 、 $r_T(t_6) = 2$  である。上下制約グラフ  $G(T')$  は式(5.2)より図5.6(b)のようになる。 $\Gamma_T(t_2) = \{t_1\}$  であるから、頂点  $t_3$ 、 $t_4$ 、 $t_5$ 、 $t_6$  に対しては、その  $r_T$  と  $r_{T'}$  は同じ値である。頂点  $t_1$ 、 $t_7$ 、 $t_8$  に対して  $r_{T'}$  を計算すると  $r_{T'}(t_1) = 1$ 、 $r_{T'}(t_7) = 0$ 、 $r_{T'}(t_8) = 1$  となる。定義5.2より  $W(T) = 4$ 、 $W(T') = 4$  となるが、 $T_2 = \{t_1, t_6\}$ 、 $D(T_2) = 1$ 、 $S(T_2) = \{1, 2, 9, 10\}$  に対し、 $T'_2 = \{t_6\}$ 、 $D(T'_2) = 1$ 、 $S(T'_2) = \{9, 10\}$  であるから、分割条件A(2)を満たす。従って、この幹線分割は分割条件A、BかつCを満たす。提案した幹線分割の算法を図5.5(a)の幹線集合  $T$  に適用すると、幹線  $t_2$  と  $t_5$  がそれぞれ  $x$  座標値4と8の点で幹線分割され、図5.7のように3本のトラックを用いたチャネル配線(最適解)が可能となる。(例終)

### 5.5 実験結果と考察

本節では、提案した幹線分割の算法を文献(5)の配線要求と8,000ゲートCMOSゲートアレイ<sup>(11)</sup>の配線要求に適用した計算機実行の結果について述べる。

実験に用いた8,000ゲートCMOSゲートアレイは図2.1に示したチップ構造を持ち、チップ上の垂直方向に並んだセル列数は28で、各セル列にはP及びN型トランジスタのペア<sup>(11)</sup>が864ずつ並んでいる。

提案した幹線分割の算法のプログラムはFORTRANで記述され、その規模は約3,000ステートメントである。幹線分割のプログラムを第2章で述べたGALOP配線プログラム<sup>(12)</sup>に組み込んだ。この実験において、GALOPのチャンネル配線ルーチンの機能上の制約から幹線分割後の各幹線 $t_i$ に対し $|P(t_i)| \leq 1$ と制限し、5.3で述べた定数 $\delta$ を-1とした。また、チャンネル配線において、4つの"range"パラメータ<sup>(2)</sup>値と8通りの"routing sequence"<sup>(2)</sup>の32通りのパラメータの組み合わせに対しチャンネル配線を試み、この内で使用された水平トラック数が最も少ない配線結果を採用した。

文献(5)の配線要求(配線要求A)と2品種の8,000ゲートCMOSゲートアレイのGALOP配線プログラムによる配線結果から抽出した4つのチャンネル(配線要求B, C, D, E)に対し、GALOPのチャンネル配線ルーチンだけを適用した結果と、幹線分割プログラムとGALOPのチャンネル配線とルーチンを適用した結果を表5.1に示す。この2品種において、チャンネル配線で使用される水平トラック数と最大幹線密度の差が3以上であるチャンネルは表5.1の配線要求B, C, D, Eだけであつた。実験に使用した計算機はMELCOM-COSMO900II(3.5MIPS)である。幹線分割プログラムは表5.1のすべての配線要求に対し、実用的な処理時間で $W(T)$ の値を幹線分割により減少させ、その結果チャンネル配線に使用される水平トラック数は減少した。配線要求Aに対し、幹線分割プログラムとGALOPのチャンネル配線ルーチンを適用した後の配線結果を図5.8に示す。図5.8の配線結果は最小の水平トラック数での配線結果ではない。図5.8では信号ネット1と10の幹線がそれぞれx座標値7と8の点で幹線分割され、6本の水平トラックを用いてチャンネル配線されている。このように幹線分割を行った場合、5本の水平トラックではチャンネル配線できない。一方、文献(5)の配線結果では信号ネット1と10の幹線がそれぞれx座標値8と7の点で幹線分割され、5本の水平トラックを用いてチャンネル配線されている。これは、本章で提案した幹線分割の算法では繰り返し幹線分割を行う時の幹線を選ぶ順序及び分割する点を選ぶ順序について考慮していないことによる。

以上の実験から、チャンネル配線で使用される水平トラック数と最大幹線密度の差が大きい配線要求に対して、チャンネル配線で幹線の水平方向の区間を決める時に提案した算法で幹線分割を行えば、実用的な処理時間でチャンネル配線に使用さ

表5.1 実験結果

配線要求	幹線数	最大幹線密度 D (T)	W (T)	チャンネル配線のみを行った場合		幹線分割とチャンネル配線を行った場合			
				使用トラック数	処理時間 (秒)	幹線分割された幹線数	幹線分割後の W(T)	使用トラック数	幹線分割の処理時間 (秒)
A	14	4	11	11	0.7	2	7	6	0.4
B	366	25	40	29	50.8	24	29	26	181.8
C	324	27	42	30	57.4	14	36	28	220.5
D	331	27	43	32	75.6	22	36	30	288.9
E	238	21	32	25	74.7	12	28	23	104.2

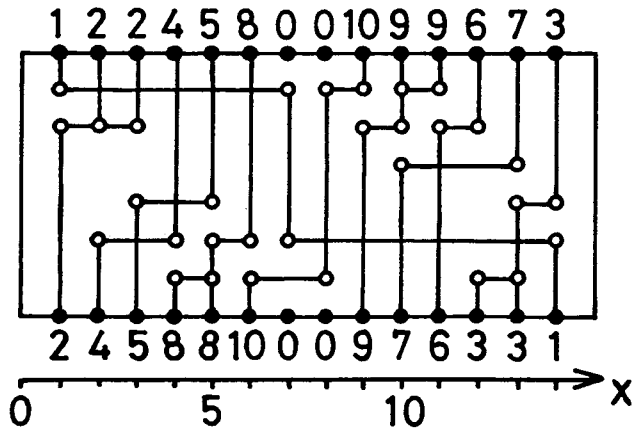


図5.8 配線要求Aの配線結果

れる水平トラック数が減少することが実証された。

## 5.6 結 言

本章では、チャンネル配線の全信号ネットの幹線の集合 $T$ に対し関数 $W(T)$ を定義し、 $W(T)$ の値が $T$ の全幹線を水平トラック上に配置してチャンネル配線を行うために十分な水平トラック数であることを証明した。次に幹線分割問題に対し、幹線相互間の上下制約グラフ中の有向道の長さが増加せず、かつ関数 $W(T)$ の値が減少するように幹線分割を行う算法を提案した。更に、8,000ゲートCMOSゲートアレイの配線設計に提案した幹線分割算法を適用し、チャンネル配線で使用される水平トラック数と最大幹線密度の差が大きい場合には、本算法が有効であることを確認した。現在、GALOP配線プログラムでは配線に使用された水平トラック数と最大幹線密度の差が大きいチャンネルに対してだけ、再度、提案した算法に基づく幹線分割とチャンネル配線を行うようになっており、実際にこの幹線分割の算法は配線に使用される水平トラック数の減少に役立っている。

提案した算法では繰り返し幹線分割を行う時の幹線を選ぶ順序及び分割する点を選ぶ順序については考慮していない。実際、表5.1の配線要求Aに対しこれらの順序を変えて幹線分割を行うと、信号ネット1と10の幹線がそれぞれ $x$ 座標値8と7の点で幹線分割され、5本の水平トラックでチャンネル配線できた。よりよい解を得るために、これらの順序を考慮する問題が残されている。

なお、本章の内容は文献(13)に基づいている。

## 参考文献

- (1) A.Hashimoto and J.Stevens: "Wire routing by optimizing channel assignment within large apertures", Proc. 8th DA Workshop, pp. 155-169(1971).
- (2) D.N.Deutsch: "A dogleg channel router", Proc. 13th DA Conf., pp. 425-433(1976).
- (3) T.Yoshimura and E.S.Kuh: "Efficient algorithms for channel routing" IEEE Trans. Comput.-Aided Des. Integrated Circuits & Syst., vol. CAD-1, no. 1, pp. 25-35(1982).
- (4) R.L.Rivest and C.M.Fiduccia: "A "greedy" channel router", Proc. 19th DA Conf., pp. 418-424(1982).
- (5) M.Burstein and R.Pelavin: "Hierarchical channel router", Proc. 20th

- DA Conf., pp. 591-597(1983).
- (6) T.Yoshimura: "An efficient channel router", Proc. 21st DA Conf., pp. 38-44(1984).
- (7) K.Sato, H.Shimoyama, T.Nagai, M.Ozaki and T.Yahara: "A "grid-free" channel router", Proc. 17th DA Conf., pp. 22-31(1980).
- (8) M.Terai: "A method of improving the terminal assignment in the channel routing for gate arrays", IEEE Trans. Comput.-Aided Des. Integrated Circuits & Syst., vol. CAD-4, no. 3, pp. 329-336(1985).
- (9) R.E.Tarjan: "Depth-first search and linear graph algorithms", SIAM J. Comput., vol. 1, no. 2, pp. 146-160(1972).
- (10)尾崎, 白川, 翁長: グラフ理論, コロナ社(昭和50).
- (11)I.Ohkura, T.Noguchi, K.Sakashita, H.Ishida, T.Ichiyama and T.Enomoto: "Gate isolation-A novel basic cell configuration for CMOS gate arrays", Proc. 1982 CICC, pp. 307-310(1982).
- (12)寺井, 野田, 佐藤, 八原: "種々のチップ構造のゲートアレイに適用可能な配線プログラム", 情報処理学会論文誌, vol. 25, no. 3, pp. 357-364(1984).
- (13)寺井, 佐藤: "チャネル配線法における幹線分割の一手法" 信学論(A), vol. J68-A, no. 11, pp. 1184-1192(1985).

## 第6章 ゲートアレイにおける水平トラックの必要総数の予測問題

### 6.1 緒言

ゲートアレイでは配線領域が前もって固定されるため、特定の領域に配線が集中して配線不能となる信号ネットが発生するという問題が生ずる。ゲートアレイのセル配置が定まった段階において、チップ全体の配線に必要な水平トラック数を正確に予測できれば、配線を行う前に配線可能かどうか判定でき、これを基にセル配置を改善することにより配線不能となる信号ネットの発生を減らすことが可能となる。

最近、2セル列間の水平チャネルの最小間隔配線について種々の成果<sup>(1)-(3)</sup>が得られているが、これをチップ全体の配線に必要な水平トラック数の予測問題に適用するには、全信号ネットのグローバル配線を先に行わなければならない、処理時間がかかる点で実用的でない。また、文献(4)では垂直格子線を何本でも挿入できるという仮定の下で、Nセル列間の配線を行うために必要かつ十分な水平トラック数が最大交さ数M又はM+1本であることが証明され、この水平トラック数で配線を行うアルゴリズムが示されている。しかし、一般にこの最小の水平トラック数で配線を行うと幹線分割を何回も行う必要があるため垂直配線の数が多くなり、実際には実現不可能な配線結果となる。

本章では、ゲートアレイのセル配置が定まった段階において、チップ全体の配線に必要な水平トラック数を予測する問題について考察する。まず図6.1に示すチップモデルのゲートアレイにおいて、Nセル列間の配線要求Dを定義する。次に各信号ネットの幹線分割回数に制限を加えた場合において、配線要求Dを実現するために必要かつ十分な水平トラック数について考察する。更に、624ゲートのDSA MOSゲートアレイに関する計算機実験の結果について述べる。

### 6.2 問題の定式化

ゲートアレイにおける配線問題を以下のように定式化する。

(1) チップ上にはセルの端子を列状に規則正しく配列したセル列が図6.1のように並んでいる。セル列上のすべての端子には整数 $s$  ( $s$ は0, 1, ..., Sのいずれかの整数; Sは全信号ネット数)が与えられている。0以外の番号 $s$ の全端子は電氣的に同電位となるように、同じ信号ネット $W_s$ により結線される。番号0の端子を空き端子と呼ぶ。

(2) セル列間の配線領域を水平チャネルと呼ぶ。水平チャネルを二つの層に分

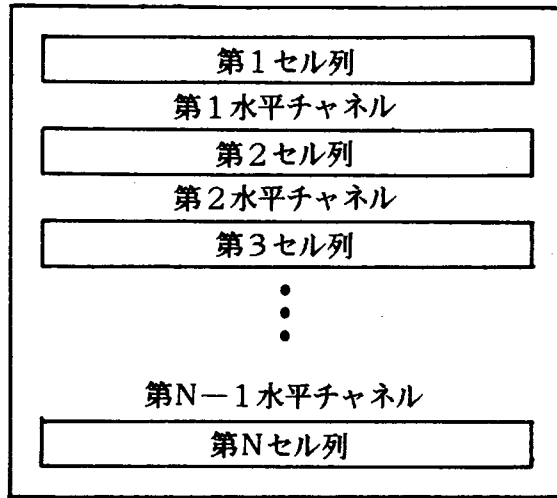


図6.1 水平トラックの必要総数の予測問題におけるチップモデル

け，第1層の水平格子線（水平トラック）上に水平配線，第2層の垂直格子線上に垂直配線を置き，これらをビアホールにより連結する2層配線方式を仮定する。

(3) セル列上の端子には上下両方向に信号ネットが接続できる。また，空き端子の上を信号ネットが通過できる。

(4) チップ上のセル列の数を  $N$ ，1セル列上の端子の数を  $P$  とすると，配線要求  $D$  を次のように表わすことができる。

$$D = [T_1, T_2, \dots, T_N, L, R]$$

ここで， $T_i$  は第  $i$  セル列上の順序付けされた端子集合で，

$$T_i = (t_{i1}, t_{i2}, \dots, t_{iP}), t_{ij} \in \{0, 1, \dots, S\}$$

と示される。また， $L, R$  はそれぞれチップの左辺または右辺の端子に接続しなければならない信号ネットの集合である。

(5) 配線要求  $D$  で端子  $t_{1j}, t_{2j}, \dots, t_{Nj}$  は同一垂直格子線上にある。この垂直格子線を  $v_j$  と表わす。

本章では，次の仮定の下で配線における水平トラックの必要総数の予測問題の理論的考察を行う。

(i) 垂直格子線上の全端子が空き端子であるものを空き垂直格子線と呼ぶ。各垂直格子線の間空き垂直格子線を何本でもそう入できる。

(ii) チップ左右辺への接続はチップ左右脇の垂直チャネルを用いて適宜行われるものとし、ここでは考慮しない。

(iii) ある垂直格子線  $v_j$  上の端子だけを結線する信号ネットで、水平トラックを用いなくて  $v_j$  上だけで配線が完了するネット（次節の定義6.2の同値関係  $\sim$  が成立する端子だけを結線する信号ネット）はあらかじめ  $v_j$  上だけで配線が行われるものとし、以下では考察の対象としない。

以下では、この仮定の下で、更に幹線分割回数に制限を加え、配線に必要なかつ十分な水平トラック数について考察する。実際のゲートアレイでは、空き端子が多く発生し、しかも同一セル列または隣接する2セル列上の端子を結線する信号ネットが全信号ネットの大半を占めるようになる。この場合には、実際の配線を行う時に、空き垂直格子線の挿入はほとんど必要がなく、仮定(i)の下で得られた結果は実用的である。このことを6.5節で実証する。

### 6.3 配線要求と最大交さ数

本節では、配線要求  $D$  の最大通過ネット数  $C_M$  と最大交さ数  $M$  を定義する。

〔定義6.1〕 配線要求  $D$  に対し、垂直格子線  $v_j$  上の端子に接続しているか、あるいは、 $v_j$  を横切ることなしには配線できない信号ネットの集合を  $C(j)$  とする。このとき、配線要求  $D$  の最大通過ネット数  $C_M$  を

$$C_M = \max_j |C(j)| \quad (1 \leq j \leq P)$$

と定義する。

〔定義6.2〕 配線要求  $D$  に対し、垂直格子線  $v_j$  上にある2端子  $t_{ij}, t_{kj}$  が次の(i)及び(ii)を満たすとき、 $t_{ij} \sim t_{kj}$  である。

(i)  $t_{ij} = t_{kj} > 0$

(ii)  $i \leq h \leq k$  又は  $k \leq h \leq i$  なるすべての  $h$  に対し、 $t_{hj} = 0$  又は  $t_{hj} = t_{ij}$  が成り立つ。 (定義終)

2項関係  $\sim$  は同値関係である。垂直格子線  $v_j$  上の端子  $t_{1j}, t_{2j}, \dots, t_{Nj}$  の内、空き端子でない全端子の集合  $E_j$  (但し、前節の仮定(iii)の信号ネットを構成する端子は  $E_j$  より除く) を同値関係  $\sim$  によって同値類に直和分割したとき、この同値類全体の集合 ( $E_j$  の  $\sim$  による商集合) を  $E_j / \sim$  で表わす。

〔例6.1〕 垂直格子線  $v_j$  上の同一番号を持つ2端子が  $v_j$  上だけで結線できるとき、同値関係  $\sim$  が成立する。例えば図6.2の例では、 $t_{2j} \sim t_{3j}$  が成立するが、 $t_{1j} \sim t_{4j}$  は成立しない。従って、 $E_j = \{t_{1j}, t_{2j}, t_{3j}, t_{4j}\}$  は図6.2のように三つの同値類に分けられ、 $|E_j / \sim| = 3$  となる。



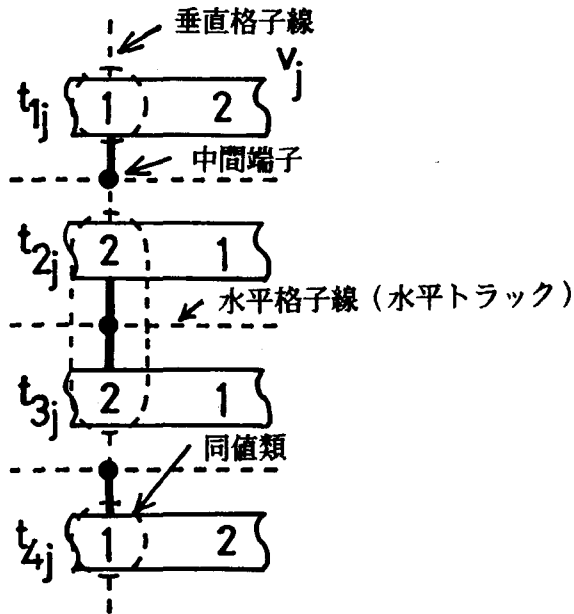


図6.2 同値関係～

〔定義6.3〕 配線要求Dに対し，垂直格子線 $v_j$ 上の端子に接続せず，しかも $v_j$ を横切らなければならない信号ネットの集合を $A(j)$ とするとき， $M(j)$ を

$$M(j) = |A(j)| + |E_j / \sim|$$

と定義し，配線要求Dの最大交さ数Mを

$$M = \max M(j) \quad (1 \leq j \leq P)$$

と定義する。

(定義終)

なお，文献(4)では端子列も水平トラック上に置き，端子には上下左右から信号ネットを接続できるものとした場合の配線に必要なかつ十分な水平トラック数について議論している。この場合には，垂直格子線 $v_j$ 上の空き端子でない各端子に対し，それを置くために必ず1本の水平トラックが必要となるので，最大交さ数Mを  $\max ( |A(j)| + [v_j \text{上の空き端子以外の端子数}] )$  と定義している。

〔補題6.1〕 配線要求Dに対し，最大通過ネット数 $C_M$ と最大交さ数Mの間には次の関係が成り立つ。

$$C_M \leq M \leq C_M + N - 2$$

(証明)  $C(j)$ ， $A(j)$ ， $E_j / \sim$ の定義より

$$0 \leq |C(j)| - |A(j)| \leq |E_j / \sim| \leq N$$

である。しかも、 $|E_j / \sim| = N$ のとき、 $|C(j)| - |A(j)| \geq 2$ であり、 $|C(j)| - |A(j)| \leq 1$ のとき、 $|E_j / \sim| \leq 1$ であるので、

$$0 \leq |E_j / \sim| + |A(j)| - |C(j)| \leq N - 2$$

である。故に、

$$C_M \leq M \leq C_M + N - 2$$

である。ここで、 $M = C_M + N - 2$ となるのは、 $C(j) = C_M$ なる $v_j$ 上で、

$t_{1j} = t_{3j} = \dots > 0$ 、 $t_{2j} = t_{4j} = \dots > 0$  ( $t_{1j} \neq t_{2j}$ ) であるとき、かつそのときに限る。 (証明終)

[例6.2] 図6.3に示す配線要求 $D = [T_1, T_2, T_3, T_4, L, R]$

$$T_1 = (3, 4, 4, 0, 5)$$

$$T_2 = (2, 1, 2, 5, 1) \quad L = \{W_3\}$$

$$T_3 = (1, 2, 1, 0, 2) \quad R = \{W_5\}$$

$$T_4 = (3, 3, 4, 4, 4)$$

が与えられているとき、 $|E_j / \sim| = 4$ 、 $A(j) = \phi$  ( $j = 1, 2, 3, 5$ )、 $|E_4 / \sim| = 2$ 、 $A(4) = \{W_1, W_2\}$ であるので、 $C_M$ 及び $M$ は次のようになる。

$$C(1) = \{W_1, W_2, W_3\} \quad M(1) = 4$$

$$C(2) = \{W_1, W_2, W_3, W_4\} \quad M(2) = 4$$

$$C(3) = \{W_1, W_2, W_4\} \quad M(3) = 4$$

$$C(4) = \{W_1, W_2, W_4, W_5\} \quad M(4) = 4$$

$$C(5) = \{W_1, W_2, W_4, W_5\} \quad M(5) = 4$$

$$C_M = \max |C(j)| = 4 \quad M = \max M(j) = 4 \quad (1 \leq j \leq 5)$$

なお、図6.4に結線要求 $D$ を $M$ 本の水平トラックで実現した例を示す。(例終)

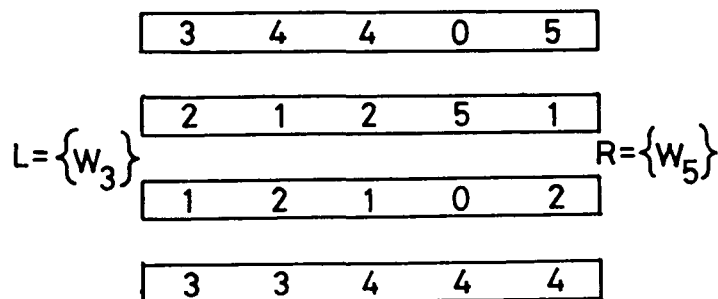


図6.3 配線要求 $D$ の例

### 6.4 配線に必要な水平トラックの総数の予測

文献(4)では、配線要求Dに対する最小間隔配線について次の定理6.1が証明されている。

[定理6.1] 配線要求Dを実現するために必要かつ十分な水平トラック数はM又はM+1本である。ここで、 $N \leq M$ であるとする。 (定理終)

定理6.1は本章でのレイアウトモデルについても成り立つことが次のようにして確かめられる。M本の水平トラックを各水平チャンネル内に任意に置く。各垂直格子線 $v_j$ に対して、 $E_j/\sim$ の各同値類の代表元に対応する信号ネットの中間端子<sup>(4)</sup>(intermediate terminal<sup>(2)</sup>)を $v_j$ と水平トラックの交点上に置き、各中間端子とそれに対応する $v_j$ 上の端子を $v_j$ 上で接続する(図6.2参照)。この中間端子を文献(4)での端子と考えれば、定理6.1は文献(4)と同様に証明できる。

定理6.1で示した水平トラック数で配線要求Dを実現すると、例えば図6.4の信号ネット $W_1$ のように幹線分割が何回も行われ、垂直配線の数が多くなり、空き垂直トラックを多数挿入しなければならない。その結果、実際には実現不可能な配線結果となる。幹線分割を1回行うごとに1本の幹線が2本に分けられるので(図6.5参照)、ある信号ネットに対し幹線分割をn回行うと、その信号ネットはn+1本の幹線を用いて結線される。

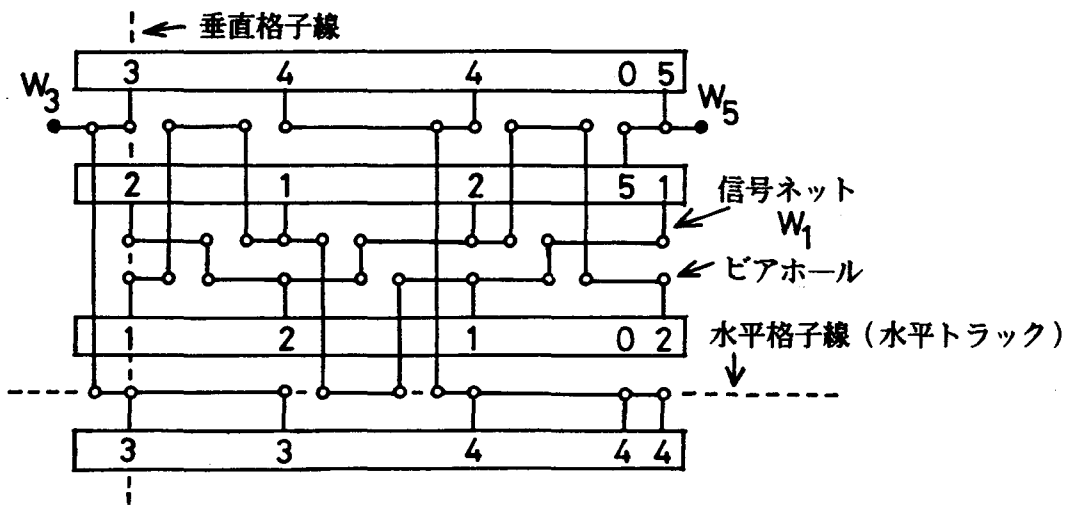
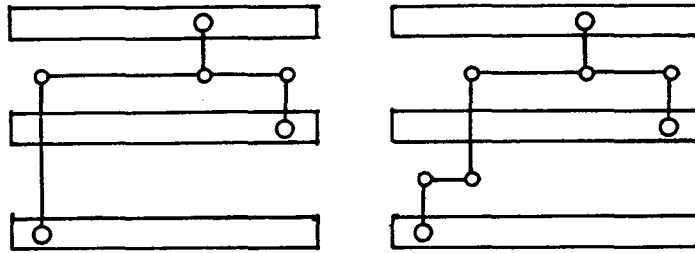


図6.4 最小の水平トラック数での配線結果



(a) 幹線分割なし (b) 幹線分割1回

図6.5 幹線分割

本節では、挿入される空き垂直格子線の数減らし、実現可能な配線結果となるように、各信号ネットの幹線分割の回数に対し制限を加える。VLSIの自動配線設計で広く用いられているチャネル配線の手法<sup>(5)-(7)</sup>では、一般に端子の位置でしか幹線分割を行わない。即ち、 $n$ 個の端子から構成される信号ネットに対し、高々 $n-2$ 回の幹線分割しか行わない。このため、 $n$ 個の端子から構成される信号ネットの幹線分割回数を高々 $n-2+\alpha$  ( $\alpha$ は0か小さな整数値)回に制限することは現実的である。以下では、 $\alpha=2$ とし、各信号ネットに対し、高々信号ネットを構成する端子数の回数だけしか幹線分割を行わないという制限を加える。このとき、図6.4の例で示したように一般に定理6.1で示した水平トラック数での配線は不可能となる。

〔定理6.2〕 配線要求 $D$ に対し、各信号ネットの幹線分割回数とその信号ネットを構成する端子数以下であるという制限の下で配線する時、高々 $C_M+N-1$ 本の水平トラックを用いて配線可能である。

(証明)  $C_M+N-1$ 本の水平トラックを用いて配線できることを示す。配線要求 $D$ に対し、各垂直格子線の間空き垂直格子線を $N-1$ 本挿入した配線要求 $D'=[T'_1, T'_2, \dots, T'_N, L, R]$

$$T'_i = (\underbrace{0, \dots, 0}_{N-1}, t_{i1}, \underbrace{0, \dots, 0}_{N-1}, t_{i2}, \dots, t_{iP-1}, \underbrace{0, \dots, 0}_{N-1}, t_{iP})$$

を考える。更に端子集合 $T''_2$ を次のように定義し、図6.6のように第1水平チャネル内に置く。

$$T''_2 = (t_{N1}, t_{N-11}, \dots, t_{21}, 0, t_{N2}, t_{N-12}, \dots, t_{22}, 0, \dots, t_{NP}, t_{N-1P}, \dots, t_{2P}, 0)$$

2セル列間の配線要求 $[T'_1, T''_2, L, R]$ の最大通過ネット数 $C'_M$

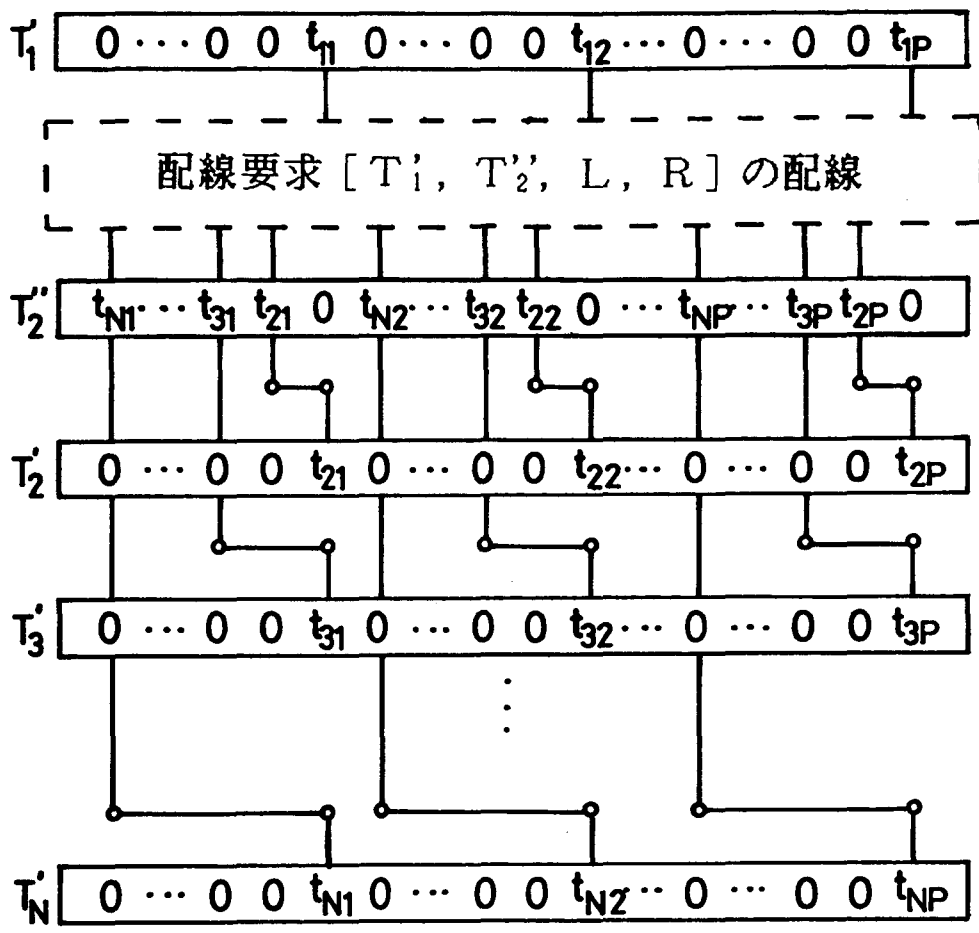


図6.6 配線要求D'

は  $C'_M \leq C_M$  である。なお、 $T''_2$  で  $t_{Nj}, t_{N-1j}, \dots, t_{2j}, 0$  の順序は必ずしも上記のとおりでなくてもよく、この場合  $C'_M = C_M$  となるのは、ある垂直格子線  $v_j$  に対し  $|C(j) \cap C(j+1)| = C_M$  のとき、かつそのときに限る。また配線要求  $[T'_1, T''_2, L, R]$  において幹線の上下制約関係<sup>(1), (2)</sup>がないので、文献(1), (2)より、これを配線するために必要かつ十分な水平トラック数は  $C'_M$  本であり、しかも幹線分割を行わないで配線できる。従って、配線要求  $[T'_1, T''_2, L, R]$  は高々  $C_M$  本の水平トラックで幹線分割を行わないで配線できる。

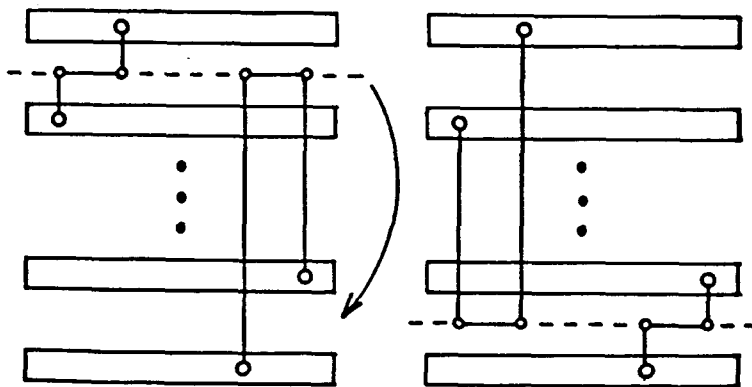
この配線結果において、端子集合  $T''_2$  の各端子  $t_{ij}$  ( $2 \leq i \leq N$ ) に対し、

この端子を通過して第  $i-1$  水平チャンネルまで垂直格子線上をまっすぐ配線し、第  $i-1$  水平チャンネルを通過して第  $i$  セル列上の端子  $t_{i,j}$  とつなぐ操作を行う（図 6.6 参照）。この配線のためには、各水平チャンネルに高々 1 本の水平トラックがあれば十分である。（証明終）

各垂直格子線の間空き垂直格子線を  $N$  本挿入し、 $C_M + N$  本の水平トラックを使用して定理 6.2 の証明と同様の配線を行った場合、図 6.7 のように第 1 水平チャンネル内の同一水平トラック上に置かれた幹線の組をそれぞれ別の水平チャンネル内へ移動できる。従って、次の定理 6.3 が成り立つ。

〔定理 6.3〕 配線要求  $D$  に対し、各信号ネットの幹線分割回数はその信号ネットを構成する端子数以下であるという制限の下で配線する。このとき、全体で高々  $C_M + N$  本の水平トラックを使用すれば、これらの水平トラックはそれぞれの水平チャンネル内にあっても配線可能である。（定理終）

定理 6.2 の幹線分割回数に対する制限の下で、配線要求  $D$  を実現するために必要かつ十分な水平トラック数  $G$  は、 $M \leq G \leq C_M + N - 1$  を満たすことを証明し（補題 6.1 と文献 (4) の配線に必要かつ十分なトラック数が  $M + 1$  となる条件から、〔定理 6.1 の水平トラック数〕  $< C_M + N - 1$  である）、次に水平トラックを  $C_M + N$  本使用すれば、これらの水平トラックがそれぞれの水平チャンネル内にあってもよいことを示した。



(a) 幹線の移動前 (b) 幹線の移動後

図 6.7 幹線の移動

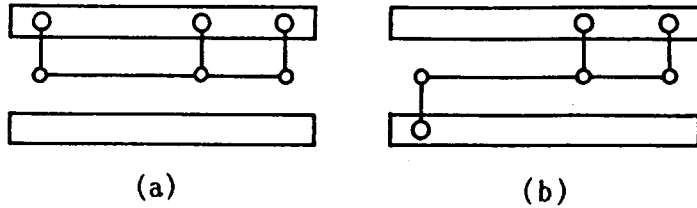


図6.8 (a) 1セル列ネットと(b) 2セル列ネット

以下で、必要な空き垂直格子線の数に言及しておく。6.2節で述べたように、現実の配線では空き垂直格子線を挿入する代わりに、既存の垂直格子線を使用することになる。図6.1のチップ構造では、実際には、同一セル列または隣接する2セル列上の端子を結線する信号ネットが全信号ネットの大半を占めることから、以下では、配線要求Dの信号ネットが次に示す1セル列ネットと2セル列ネットだけからなる場合に、定理6.3で示した配線を行うにはどれだけの空き垂直格子線があれば十分かを考察する。

まず、図6.8に示すように、信号ネットを構成する端子がすべて同一セル列上にある信号ネットを1セル列ネットと呼び、信号ネットを構成する端子がすべて隣接する2セル列上にある信号ネットを2セル列ネットと呼ぶことにする。ここで、第*i*セル列上の端子を結線する1セル列ネットは第*i*水平チャネル内で（但し、*i* = *N*のときは第*N* - 1水平チャネル内で）配線し、第*i*セル列と第*i* + 1セル列上の端子を結線する2セル列ネットは第*i*水平チャネル内で配線すると仮定する（図6.8参照）。

〔定義6.4〕 配線要求Dの各信号ネットに対し、一つの水平チャネルが配線領域として割り当てられているとき、第*i*水平チャネルを配線領域として割り当てられている信号ネットに対し、垂直格子線  $v_j$  の端子に接続しているか、あるいは、 $v_j$  を横切らなければならない信号ネットの集合を  $C_i(j)$  とする。このとき、第*i*水平チャネルの最大通過ネット数  $C_{M_i}$  を

$$C_{M_i} = \max |C_i(j)| \quad (0 \leq j \leq P)$$

と定義する。

(定義終)

このとき、 $C_M$  と  $C_{M_i}$  の間には次の関係式が成り立つ。

$$C_M \leq \sum_{i=1}^{N-1} C_{M_i} \quad (\text{以下、} \sum_{i=1}^{N-1} C_{M_i} \text{を} \Sigma C_{M_i} \text{と略記する})$$

(1)  $C_M = \sum C_{M_i}$  の場合

Nセル列間の配線要求Dの配線は、水平チャネルをはさんで対向する2セル列間の配線を繰り返して行くと考えられる。定理6.2より、それぞれの2セル列間の配線は $C_{M_i} + 1$ 本の水平トラックで配線できるので、次の定理6.4が導かれる。

[定理6.4] 配線要求Dの信号ネットが1セル列ネットと2セル列ネットだけからなり、 $C_M = \sum C_{M_i}$ であると仮定する。このとき、高々 $C_M + N - 1$ 本の水平トラックを使用し、各垂直格子線の間空き垂直格子線を高々1本挿入すれば、定理6.3の幹線分割回数に対する制限を満たして配線要求Dを配線することが可能である。 (定理終)

(2)  $C_M < \sum C_{M_i}$  の場合

$C_M = \sum C_{M_i}$ となるように、何本かの信号ネット(この信号ネット集合をQとする)に図6.9のような迂回した径路を割り当てれば(図6.9(b)の信号ネットは第i水平チャネルが配線領域として割り当てられている場合である)、高々 $C_M + N$ 本の水平トラックを使用し、定理6.3の幹線分割回数に対する制限を満たして配線要求Dを配線できる。このことは定理6.3により保証されている。このとき、定理6.4の場合より高々 $\sum_{W_i \in Q} [$ 信号ネット $W_i$ を構成する端子数 $]$ 本だけ空き垂直格子線を多く挿入すれば十分である。

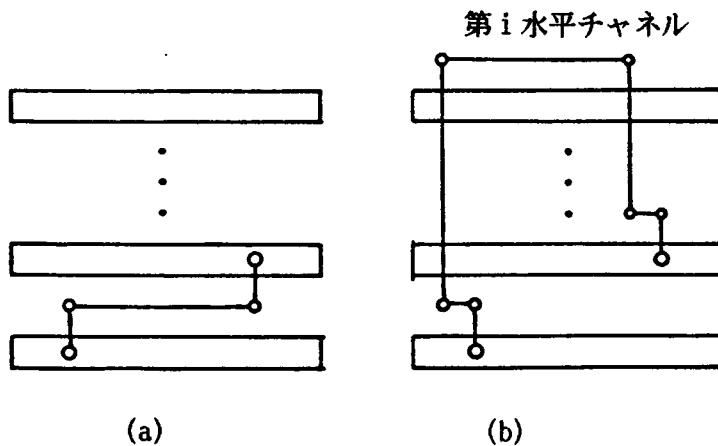


図6.9 (a) 迂回しない配線径路と(b) 迂回した配線径路



## 6.5 実験結果と考察

本節では、図2.12に示した624ゲートのDSA MOSゲートアレイについて計算機実験を行った結果について述べ、配線プログラムを実行したときに使用される水平トラック数が、定理6.3及び定理6.4で示したように、ほぼ $C_H + N$ 本に等しいことを実証する。

実験には、第2章に示したMARS-M IIの配線プログラムを用いた。624ゲートのDSA MOSゲートアレイをMARS-M IIの配線プログラムで配線した結果を表6.1に示す。これらの回路では、チップ上のゲートセルの配置設計も自動配置プログラム<sup>(8)</sup>により行った。

表6.1によれば、配線プログラムによる使用水平トラック数が $C_H + N$ 本より多くなっている。配線結果の詳細な検討から、この主な原因は次の2点で配線プログラムの性能不足によるものと考えられる。

(i) グローバル配線処理で、各水平チャンネル内の配線の局所的混雑を減らし、チップ全体の使用水平トラック数を減少させる径路探索能力が不足している。

(ii) チャンネル配線処理では、幹線の上下制約グラフ中にサイクルや長い有向道がない場合、端子との接続位置だけでしか幹線分割を行わない。

そこで、配線プログラムによる配線結果に次の4種類の簡単な人手修正を加えた。その結果、使用水平トラック数は表6.1のようにほぼ $C_H + N$ 本となった。

(1) 1セル列ネットの反転を行う(図6.10参照)。

(2) 幹線分割を行う(図6.5参照)。

(3) 配線径路が迂回した径路をとるように改める(図6.9参照)。

(4) 配線径路が迂回しているとき、迂回しない径路に改める(図6.9参照)。

人手修正後のデータAの配線パターン図を図6.11に示す。表6.1によれば、配線に使用される水平トラック数は、人手修正前後ともに、定理6.1で示した水平トラック数の $M$ 又は $M + 1$ 本と掛け離れている。しかし、人手修正後は定理6.3で示した $C_H + N$ 本の水平トラックでの配線が実現できている。これは次の理由に因ると考えられる。表6.1に示すように、実験に用いたデータでは1セル列ネットと2セル列ネットが全信号ネットの内の大半を占め、かつ空き端子がほぼ使用端子数程度存在している。使用端子数と等しい個数の空き端子の存在は各垂直格子線の間空き垂直格子線を1本挿入することとほぼ等価である。従って、定理6.3、6.4より配線プログラムと人手修正により $C_H + N$ 本の水平トラックでの配線が実現できたと考えられる。

表6.1 配線プログラムによる実験結果

デ ー タ	テ"タA	テ"タB	テ"タC	テ"タD
信号 ネット 数 S	585	618	576	566
(1セル列ネット数+ 2セル列ネット数) / S	83%	84%	83%	84%
未配線信号ネット数	1	1	0	0
使用端子数	1,794	1,958	1,773	1,749
空き端子数	1,590	1,426	1,611	1,635
信号ネットが通過した空き 端子数	359	482	361	351
最大通過ネット数 C <sub>M</sub>	99	118	106	102
最大交さ数 M	99	118	107	102
C <sub>M</sub> + N	113	132	120	116
配線プログラムによる使用 水平トラック数	121	139	124	119
人手修正信号ネット数	16	19	20	13
人手修正後の使用水平トラ ック数	113	131	119	114

(注1) 入出力バッファセルの列もセル列と考え、セル列数Nを14とした。使用端子数、空き端子数は第2, 3, …, 13セル列上のゲートセル端子についての総数である。但し、セル列上の貫通トラックも空き端子として数えた。

(注2) 未配線信号ネットは人手修正により配線した。

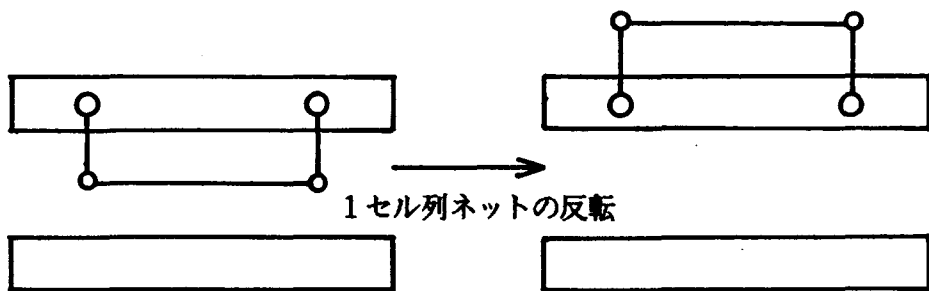


図6.10 1セル列ネットの反転

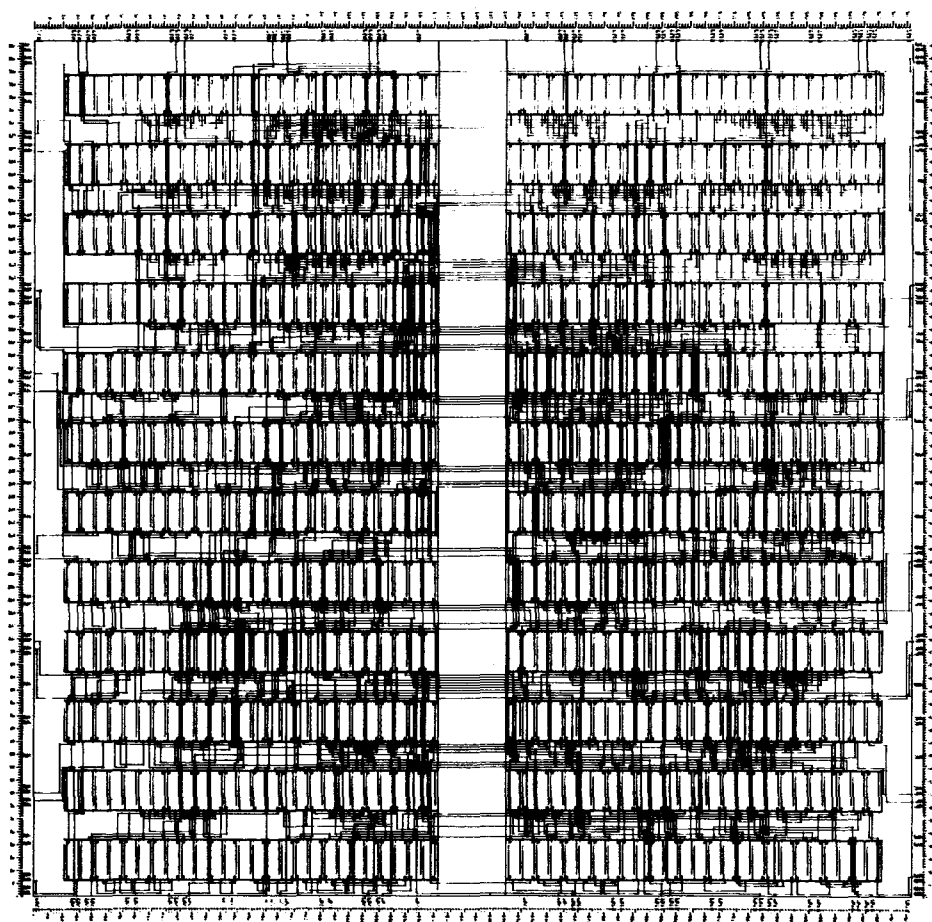


図6.11 人手修正後のデータAの配線パターン図

図6.1のチップ構造のゲートアレイの場合，総配線長の意味で最適な配置を行えば，表6.1のデータのように1セル列ネットと2セル列ネットが全信号ネットの大半を占めることになる。この場合，定理6.3及び6.4で示したように空き端子がほぼ使用端子数程度存在すれば，実際の配線プログラムで配線するのに必要な水平トラック数をほぼ $C_M + N$ 本と予測できることが，以上の実験から示された。

## 6.6 結 言

本章では，ゲートアレイのセル配置が決まった段階において，チップ全体の配線に必要な水平トラック数を予測する問題について考察した，まず，各信号ネットの配線の幹線分割回数に制限を加えた時， $N$ セル列間の配線に必要なかつ十分な水平トラック数 $G$ が $M \leq G \leq C_M + N$ であることを証明した，次に，配線プログラムMARS-M IIで実際にゲートアレイを配線した時に使用される水平トラック数がほぼ $C_M + N$ 本に等しいことを示した。

なお，本章の内容は文献(9)，(10)に基づいている。

## 参考文献

- (1) 河本，梶谷，金田，下山：“2セル列の2層最小間隔配線について”，信学技報，CST78-33(1978)。
- (2) T.Kawamoto and Y.Kajitani：“The minimum width routing of a 2-row 2-layer polycell-layout”，Proc. 16th DA Conf., pp. 290-296(1979)。
- (3) 浅野，北橋，田中，堀野，天野：“ビルディングブロック方式LSIの配線の実現可能性について”，信学論(A)，vol. 56-A，no. 9，pp. 489-496(1973)。
- (4) 河本，梶谷：“N端子列の2層最小間隔配線について”，信学技報，CAS79-143(1980)。
- (5) D.N.Deutsch：“A dogleg channel router”，Proc. 13th DA Workshop，pp. 425-433(1976)。
- (6) G.Persky，D.N.Deutsch and D.G.Schweikert：“LTX-A minicomputer-based system for automated LSI layout”，J. Des. Autom. & Fault-Tolerant Comput.，vol. 1，no. 3，pp. 217-255(1977)。
- (7) T.Yoshimura and E.S.Kuh：“Efficient algorithms for channel routing”，IEEE Trans. Comput.-Aided Des. Integrated Circuits & Syst.，vol.

CAD-1, no. 1, pp. 25-35(1982).

- (8) C.Tanaka, S.Murai, H.Tsuji, T.Yahara, K.Okazaki, M.Terai, R.Katoh and M.Tachibana: "An integrated computer aided design for gate array masterslices: Part 2 the layout design system MARS-M3", Proc. 18th DA Conf., pp. 812-819(1981).
- (9) 寺井, 金田, 八原: "マスタスライス方式レイアウトにおける横トラック数の考察", 信学論(D), vol. J64-D, no. 8, pp. 697-704(1981).
- (10) M.Terai, H.Kanada, K.Sato and T.Yahara: "A consideration of the number of horizontal grids used in the routing of a masterslice layout", Proc. 19th DA Conf., pp. 121-128(1982).

## 第7章 トランジスタアレイにおけるレイアウト手法

### 7.1 緒 言

近年、VLSIの微細加工技術の発達に伴い、VLSIチップ上に搭載可能なゲート数が飛躍的に増大している。このようなVLSIの集積度の増大と共に、ゲートアレイではRAM/ROM等のメモリを実現する必要性も高まっている。

ゲートアレイでは、メモリを実現するために、これまでに主として次のような方法がとられてきた。

(1) メモリを構成するための構成要素(32ワード×1ビットのRAM/ROM単位セル、デコーダ、センスアンプ、ドライバ等)を通常の論理セル(ゲートアレイにおける配置配線の単位)と同様に配置配線して構成する方法<sup>(1)</sup>。

(2) メモリ専用の領域をあらかじめマスタスライス上に設けておく方法<sup>(2)</sup>。

(1)の方法ではメモリをレイアウトするための特別な処理は必要としないものの、メモリ部の規則性を利用することができず、集積度が上がらないという欠点がある。また、(2)の方法では、あらかじめ人手設計しておくメモリ専用の領域では高い集積度が得られる反面、メモリ専用の領域に空き領域が生じないようにするためには、メモリ専用の領域のサイズを変えたマスタスライスを何種類も設計しておく必要がある。

これらに対して、チップ全面にトランジスタを敷き詰めたトランジスタアレイが提案されてきている<sup>(3)-(7)</sup>。トランジスタアレイでは、敷き詰められたトランジスタのうちの任意の部分を使用してメモリや論理素子を構成し、それ以外の未使用トランジスタ部分で配線を行う。このため、どのようなサイズのメモリも空き領域を生じることなく自由に構成でき、比較的高い集積度が得られるという利点がある。更にトランジスタアレイでは、配線領域の位置と大きさがあらかじめ固定されないので、配線の局所混雑による配線不能を減少させることができ、その結果、ゲートアレイより高い集積度を得ることができる。

本章では、トランジスタアレイに対して、その特長を活かしたレイアウト手法について考察する。以下では、まずトランジスタアレイのチップ構造を説明する。次に、これをレイアウトする際のモデル、及びRAM/ROMを含む回路を高密度に設計するためのレイアウト手法について考察する。更に、20,000ゲートのトランジスタアレイにこのレイアウト手法を適用した実験結果について述べる。

## 7.2 チップ構造

トランジスタアレイのチップ構造を図7.1に示す。チップは内部領域と周辺領域に分かれる。周辺領域には入出力バッファセルが並んでおり、内部領域にはP型及びN型トランジスタ列が交互に並んでいる。上下に隣接するP型、N型トランジスタの組（トランジスタペア）がCMOS回路を構成する際の基本単位となり、連続する複数のトランジスタペアで論理セルが構成される。例えば、3入力NANDの論理セルはトランジスタペア4個で構成される（図2.2参照）。各P型（N型）トランジスタ列は12本の水平トラック分の高さを持ち、各トランジ

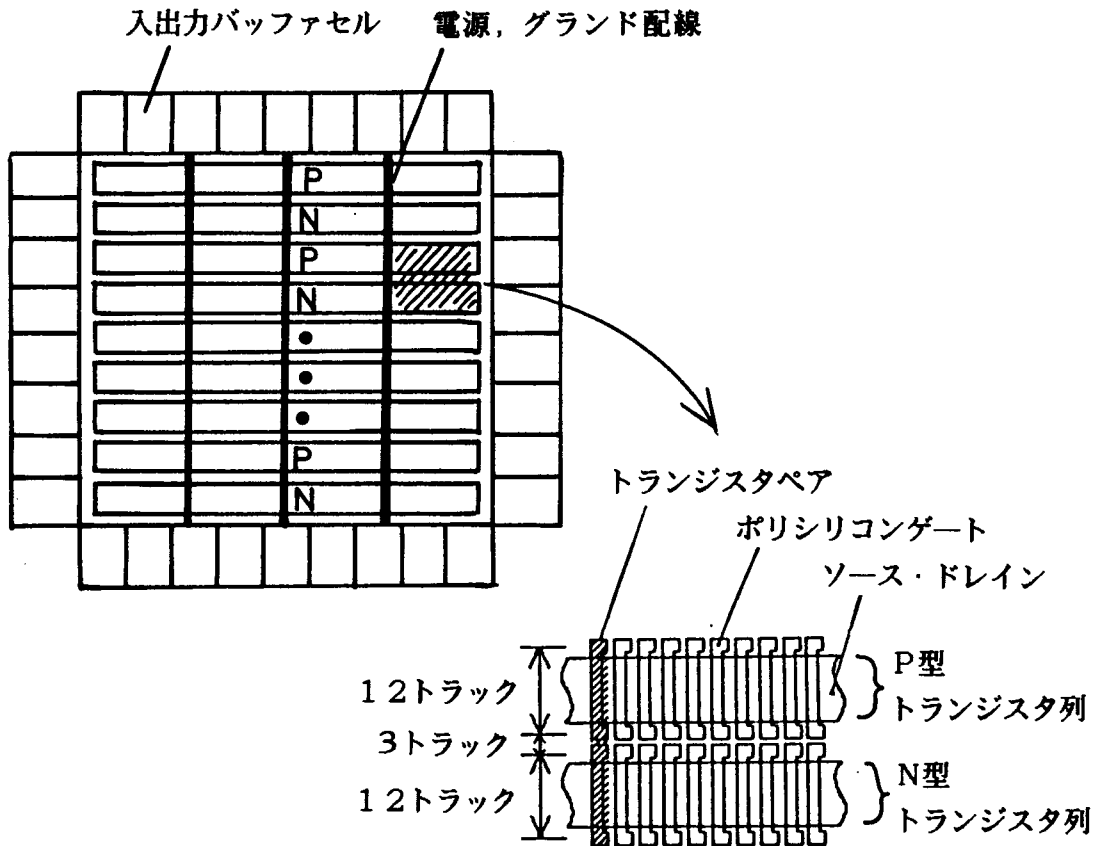


図7.1 トランジスタアレイのチップ構造

スタ列間には3本の水平トラックが設けられている。配線は2層で行われ、水平配線は第1層に、垂直配線は第2層に置かれる。

電源、グランド配線は第2層配線によりチップ内の3箇所にも垂直方向に通っているが、各トランジスタ列中にはあらかじめ電源、グランド配線は設けられていない。P型、N型トランジスタペアを用いて論理セルを構成する際に電源、グランド配線を設ける。

### 7.3 レイアウトモデル

ゲートアレイでは、論理素子が形成される領域（セル列）と配線領域とがあらかじめ固定されており、このため、局所的な配線集中により配線不能が発生することがある。これに対して、トランジスタアレイでは敷きつめられたトランジスタの任意の部分を利用して論理素子を構成し、それ以外の未使用のトランジスタの部分で配線を行う。従って、レイアウト設計の際に配線領域の面積を変えることができ、配線の局所的混雑の緩和を図ることができるという特長がある。また、RAM/ROMは、その内部に配線領域を設けずに形成できるので、高い集積度が得られるという利点もある。本節ではトランジスタアレイのこれらの特長を活かしたレイアウトを行うためのモデルについて考察する。

#### 7.3.1 レイアウトモデルに対する考察

まず、用語を定義する。水平方向に連続したP型、N型トランジスタペアを幾つか使用して構成される論理素子（例えば、NANDゲート、AND-ORゲート、フリップフロップ、シフトレジスタ等）を論理セルと呼ぶ。論理セルはP型、N型トランジスタのペアと同一の高さを持ち、水平方向の幅はその論理機能毎に異なる。論理セルを組合せて実現されるMSIレベルの論理機能ブロック（例えば、RAM/ROM、カウンタ、ALU等）を単にブロックと呼ぶ。ブロックはその高さ、幅、形状は任意である。論理セルやブロックの内部のレイアウトパターンはあらかじめ人手により設計されている。

一般にレイアウトの方法を配置配線の単位の面から分類すると次のようになる。

(1) ブロックのみを配置配線の単位とする方法。

(2) 論理セルのみを配置配線の単位とする方法。即ち、RAM/ROMのブロックも32ワード×1ビットのRAM/ROM単位セルとデコーダ等の論理セルを用いて構成する方法。



(1)の方法をトランジスタアレイに適用した場合、各ブロック内のレイアウトはトランジスタレベルで最適化されているため、高い集積度が得られるという利点がある。その反面、次の(a)、(b)の欠点を持つ。

(a) ブロックはその性能や面積が最適となるように設計されているため、その形状や大きさがさまざまに異なる。このため、ブロック間の配線を行うための配線領域の形状が複雑となる上に、ブロック上もブロック内のマスクパターンと重ならない範囲で配線を通す必要がある。従って、(1)の方法の配線設計では、高速でかつ高い配線率を達成できるチャンネル配線の手法<sup>(8)・(9)</sup>の適用が困難となり、迷路法<sup>(10)</sup>や線分探索法<sup>(11)・(12)</sup>の使用が避けられない。これらの配線手法をVLSIのような大規模な配線設計に用いると処理時間が大きくなる。その上、これらの配線手法を用いると、信号ネットの配線順序により大きく配線率が変わるため、一般に人手介入による配線設計の試行錯誤は避けられなくなる。従って、全自動による短期間のレイアウト設計が不可能となる。

(b) ゲートアレイでは、セル列内に電源、グランド配線が水平方向に置かれているため、論理セルをセル列上に並べると各論理セル内の電源、グランド配線が自然につながるようになっていく。しかし、ブロックはその形状や大きさがさまざまに異なるため、論理セルのように規則正しく配置することはできない。このため、ブロックに給電するための電源、グランド配線はブロック間の配線のための配線領域を使って行わなければならない、複雑な配線パターンとなる。電源、グランド配線の幅は通常信号ネットの配線の幅の約4倍であるため、電源、グランド配線のために多くの配線領域を必要とし、必ずしも高い集積度が得られるとは限らない。

一方、(2)の方法では、(a)、(b)の問題は生じないが、RAM/ROMを含む大規模な回路ではRAM/ROMの部分の集積度が(1)の方法より劣るという欠点がある。

短期間に設計製造ができるというセミカスタムVLSIの特長をトランジスタアレイにおいても実現するためには、全自動で短期間にレイアウト設計できることが重要である。このため、前述の(2)のレイアウトの方法を採用する。以下では、この方法に基づいたレイアウトモデルについて考察する。更に、RAM/ROMを含むVLSIを高密度にレイアウト設計するための手法についても考察する。

### 7.3.2 レイアウトモデル

従来のゲートアレイと同様に論理セルをチップの左端から右端まで横一列に配置する。この時、論理セルが配置されるP型及びN型トランジスタペア列をセル列と呼び、セル列とセル列の間のトランジスタ列を配線領域（水平チャンネル）として配線に用いる（図7.2参照）。各チャンネル内の水平トラックの本数をチャンネルの容量と呼ぶ。図7.3に示すように、各水平チャンネルの容量をトランジスタ列の高さを単位として変更することが可能である。

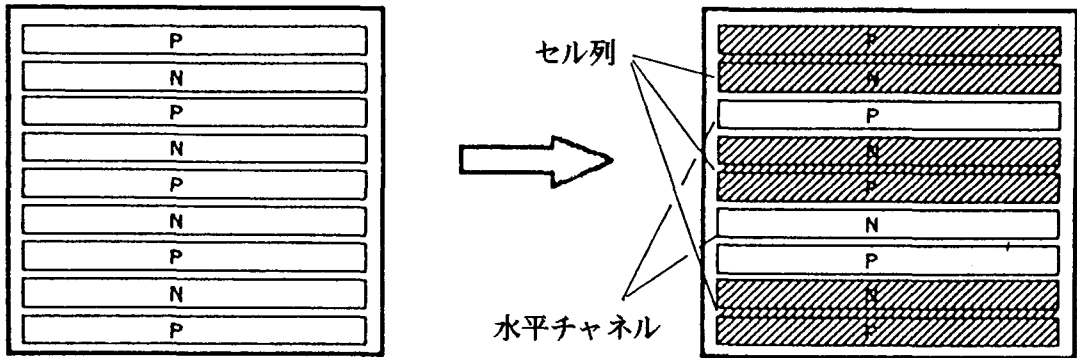


図7.2 セル列と水平チャンネル

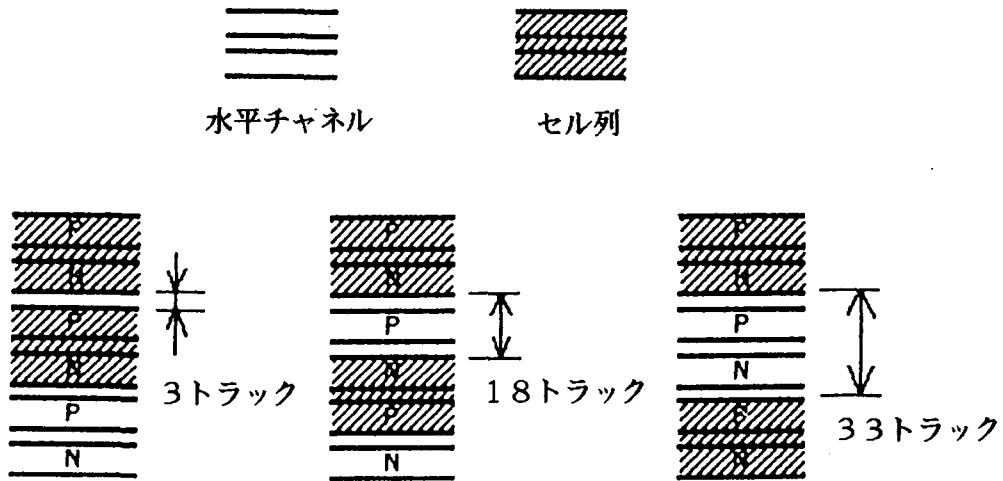


図7.3 チャンネル容量の変更

ROM/RAMは、大きさ、形状の決まったブロックではなく、RAM/ROM単位セルと、デコーダ等の論理セルを配置配線して構成される。例えば128ワード×1ビットのRAMは32ワード×1ビットの単位セルを4個とデコーダ等の論理セルから構成される。

#### 7.4 レイアウトの手法

ゲートアレイのレイアウト設計は配置と配線の二段階からなるが、トランジスタアレイのレイアウト設計では、配置配線以外にセル列数の決定と、チップ上に敷きつめられているP型及びN型トランジスタ列をセル列と水平チャンネルにどのように割り当てるかの決定（水平チャンネルの容量の決定）が必要である。

以下では、レイアウトの手法を5つのステップに分け、各ステップの処理内容について述べる。

##### (1) セル列数の決定

すべての論理セルを配置するために必要なP型、N型トランジスタペアの総数に、垂直配線がセル列上を通過するためのP型、N型トランジスタペアの領域を加え、この値を1セル列中のP型、N型トランジスタペア数で割った結果をセル列数とする。この段階では、各セル列間の水平チャンネルの容量はすべて均一にする。

##### (2) 配置

論理セルをセル列上へ配置する。配置では、総配線長最小を目的として、初期配置と反復配置改善を行う<sup>(13)</sup>。この時、RAM/ROM単位セルに関しては、アドレスラインが垂直に配線できるように7.5節で述べる手法で配置する。

##### (3) 水平チャンネルの容量決定

配線の局所的集中による配線不能が発生しないように、各水平チャンネルの容量を決定する。ここでは、まず各水平チャンネルの必要な容量を見積もるため、全信号ネットの概略配線径路を求める。この時、第3章に示した手法を用いるが、容量制限式(3.2)を考慮しない。次に、各水平チャンネルの容量と最大幹線密度（定義5.2の定義式に基づいて計算した最も混雑した箇所を通過する水平配線の本数）との比が、全水平チャンネルについて可能な限り均一になるように、チップ上に敷きつめられているP型及びN型トランジスタ列のセル列と水平チャンネルへの割り当てを変更することにより、各水平チャンネルの容量を変更する。更にセル列内のトランジスタ列の上下の順がP型、N型かあるいはN型、P型かに応じて、論理セルをx軸に関して反転させる。

トランジスタ列の高さを単位として木目細かく水平チャネルの容量を変更できるようにするため、図7.1の各トランジスタをx軸に関して対称になるように設計した。こうして、局所的な配線混雑による配線不能の発生が減少する。

#### (4) 配線

この段階では第3章で示した手法によるグローバル配線、第4章で示した算法による端子割当、第5章に示した算法による幹線分割、及び文献(8)の手法によるチャネル内の詳細配線を行う。

#### (5) 電源、グランド配線及びROMデータの書き込み

セル列として使用されたP型、N型トランジスタペア列上を水平方向に貫通するような電源、グランド配線パターンを配置する。また、ROM単位セル上に0、1のデータ（ROMデータ）の書き込みを行う。

### 7.5 RAM/ROMの構成方法

RAM/ROMはRAM/ROM単位セルとデコーダ等の論理セルを配置配線して構成する。本節では、RAM/ROMを高密度に構成するための手法について述べる。

データベース中に、ユーザにとって必要と思われる最大サイズのワード、ビットの構成を持つRAM/ROMのネットリストをあらかじめ準備しておく。RAM/ROMネットリストは32ワード×1ビットのRAM/ROM単位セル及びデコーダ等の論理セルを用いて記述されている。次にユーザの指定したRAM/ROMのワード、ビットの構成に応じて、データベース中に準備されたRAM/ROMネットリストの必要部分だけを抽出する。更に、アドレスラインが垂直に配線できるように、RAM/ROM単位セルを縦に揃えて並べて、図7.4のようにNワード×1ビット（Nは必要とするワード数以上の最小の32の倍数とする）のRAM/ROMモジュールを構成し、このモジュールを可能な限り横に並べてチップ上に所望のRAM/ROMを構成する。このようにしてRAM/ROMを構成するためには、図7.4のように、RAM/ROM単位セルを配置したセル列間の水平チャネル内に3本の水平トラックが必要かつ十分である。このため、図7.1のトランジスタアレイにおいて、各トランジスタ列が3本の水平トラック分の間隔を空けて並んでいる。こうして、高密度なRAM/ROMが構成できる。

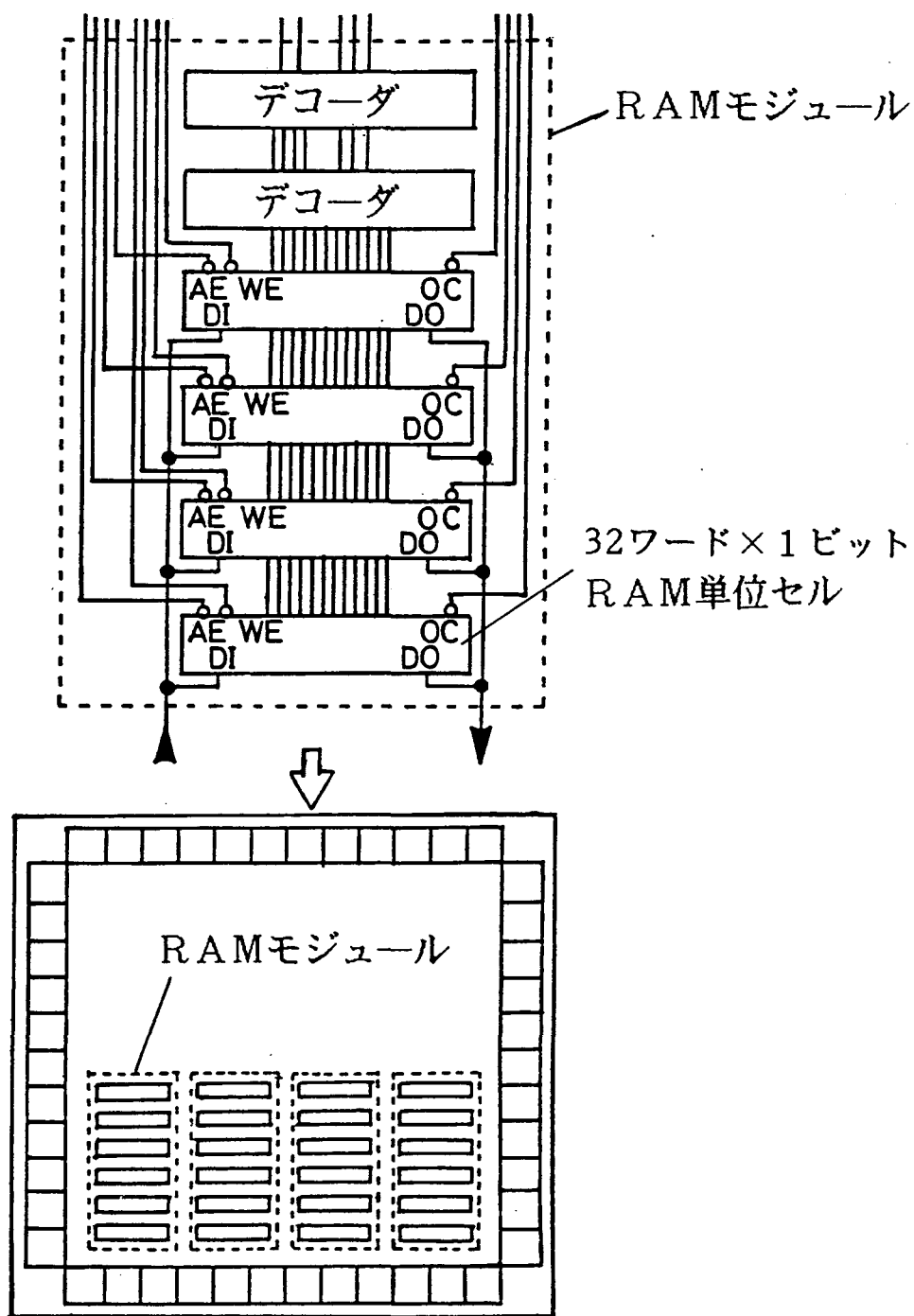


図7.4 高密度なRAM/ROMの実現法

## 7.6 実験結果と考察

本章で述べたレイアウト手法をトランジスタアレイに適用した実験結果を表7.1に示す。使用した計算機はIBM3084(7MIPS)である。表7.1におけるデータAは、32ワード×16ビットのRAM、512ワード×16ビットのROM、及び4,600ゲートのロジックからなる回路で、これを20,000ゲートのトランジスタアレイ上にレイアウトした結果のチップ写真を図7.5に示す。20,000ゲートのトランジスタアレイ上には、P型、N型トランジスタをそれぞれ142,128個敷きつめてある。図7.5において、RAM/ROMがチップの下方に規則正しくレイアウトされている。この結果から、提案した手法はRAM/ROMを含む大規模な回路を実用的な時間でレイアウトできることが明らかとなった。

表7.1におけるデータBは、128ワード×16ビットのRAM2個と4,500ゲートのロジックからなる32ビットRALUの回路である。128ワード×16ビットのRAM2個は約6,000ゲートのロジックと同一のトランジスタ数であるため、データBは全体で約10,500ゲートの規模となり、8,000ゲートのゲートアレイ上に

表7.1 実験結果

データ	P型、N型トランジスタペアの総数	メモリ	ロジック(ゲート)	信号ネット数	処理時間(分)		配線不能信号ネット数
					セル列数決定と配置	チャネル容量決定と配線	
A	38,209	32W×16bのRAM 512W×16bのROM	4,600	3,959	91	38	0
B	32,787	128W×16bのRAM2個	4,500	2,499	106	32	0

はレイアウトできない。しかし、このデータBを8,000ゲートのゲートアレイと同一のチップサイズのトランジスタアレイ上に提案したレイアウト手法により全自動でレイアウトできた。このことから、提案したレイアウト手法を用いるとRAM/ROMを含む回路を高密度にレイアウトできることが明らかになった。

## 7.7 結 言

本章では、トランジスタアレイのレイアウト手法について考察した。まず、CMOSトランジスタペアが敷き詰められたトランジスタアレイのチップ構造を述べ、次にRAM/ROMを含むCMOS VLSIを高密度に設計するためのレイアウト手法を提案した。更に20,000ゲートのトランジスタアレイに提案したレ

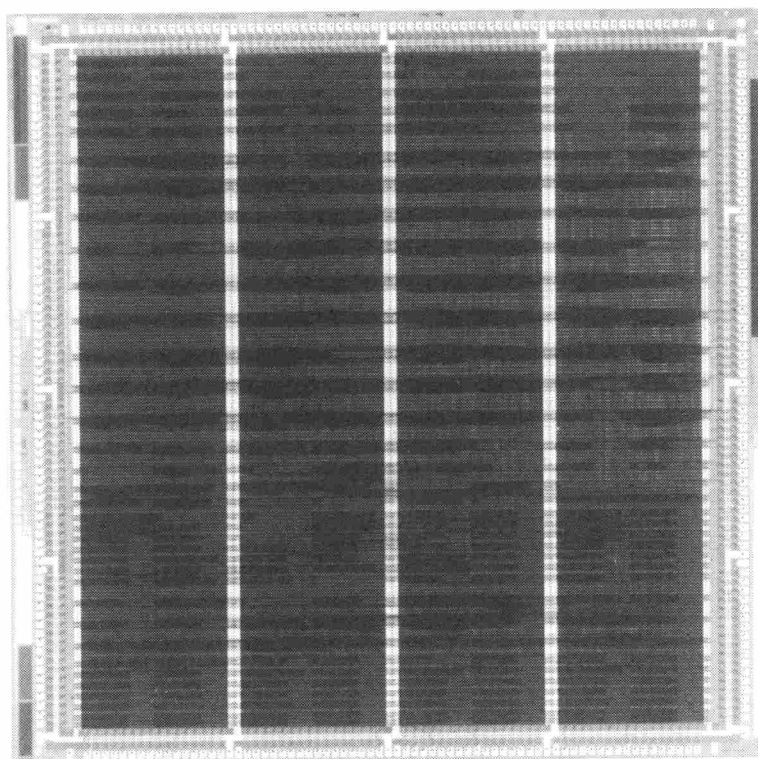


図7.5 20,000ゲートのトランジスタアレイのチップ写真

イアウト手法を適用した結果から、この手法がRAM/ROMを含むVLSIに対して特に有効であることを示した。

提案したレイアウト手法により、更に高密度なレイアウトを行うには、トランジスタアレイのチップ上のトランジスタのサイズを小さくし、チャンネル容量の木目細かい調整を行うことが有効である<sup>(7)・(14)</sup>。これを実現するためのチップ構造とレイアウト手法に関する研究が今後に残された課題である。

なお、本章の内容は文献(15)に基づいている。

### 参考文献

- (1) M.Ueda, K.Sakashita, R.Yonezu, T.Fujimura, T.Arakawa, S.Asai and Y.Kuramitsu: "A 1.5 $\mu$  CMOS gate array with configurable ROM and RAM", 1985 IEEE ISSCC Digest of Technical Papers, pp. 126-127(1985).
- (2) N.Miyahara, S.Hamaguchi, K.Ishikawa, S.Horiguchi and M.Aoki: "A composite CMOS gate array with 4K RAM and 128K ROM", Proc. 1985 CICC, pp. 248-251(1985).
- (3) H.Fukuda, H.Yoshimura and T.Adachi: "A CMOS pair-transistor array masterslice", 1982 Symposium on VLSI Technology, pp. 16-17(1982).
- (4) R.Lipp: "Advanced architecture(channel-less) dual layer metal CMOS gate array", Proc. 1983 CICC, pp. 71-73(1983).
- (5) H.Takahashi, S.Sato, G.Goto and T.Nakamura: "A 240K Transistor CMOS array with flexible allocation of memory and channels", 1985 IEEE ISSCC Digest of Technical Papers, pp. 124-125(1985).
- (6) A.Hui, A.Wong, C.Dell'oca, D.Wong and R.Szeto: "A 4.1K gate double metal HCMOS sea of gates array", Proc. 1985 CICC, pp. 15-17(1985).
- (7) T.Arakawa, M.Ueda, Y.Saito, T.Fujimura, S.Asai, M.Terai, Y.Akasaka and Y.Kuramitsu: "A Basic-Cell buffer 440K transistor CMOS masterslice", 1986 IEEE ISSCC Digest of Technical Papers, pp. 78-79 (1986).
- (8) D.N.Deutsch: "A dogleg channel router", Proc. 13th DA Conf., pp.425-433(1976).
- (9) T.Yoshimura and E.S.Kuh: "Efficient algorithms for channel routing", IEEE Trans. Comput-Aided Des. Integrated Circuits & Syst., vol. CAD-1, no. 1, pp. 25-35(1982).



- (10) C.Y.Lee: "An algorithm for path connections and its applications", IRE Trans., vol. EC-16, no. 3, pp. 346-365(1961).
- (11) K.Mikami and K.Tabuchi: "A computer program for optimal routing of printed circuit conductors", IFIP Congress 68, pp. 1475-1478(1968).
- (12) D.W.Hightower: "A solution to line routing problems on the continuous plane", Proc. 6th DA Workshop, pp. 1-24(1969).
- (13) C.Tanaka, S.Murai, H.Tsuji, T.Yahara, K.Okazaki, M.Terai, R.Katoh and M.Tachibana: "An integrated computer aided design system for gate array masterslices: Part 2 the layout design system MARS-M3", Proc. 18th DA Conf., pp. 812-819(1981).
- (14) Y.Kuramitsu, M.Ueda, T.Arakawa, M.Terai and S.Asai: "A 540K-Transistor CMOS variable-track masterslice", IEEE J. Solid-State Circuits. vol. SC-22. no. 2, pp. 198-201(1987).
- (15) 野田, 藤野, 寺井, 味岡, 佐藤: "トランジスタ敷き詰め式ゲートアレイのレイアウトシステム", 情報処理学会設計自動化研究会報告, 34-7(1986).

## 第8章 シンボリックレイアウト手法と評価

### 8.1 緒言

VLSIの集積度の向上に伴い、VLSIのマスクパターンをパターン設計ルールを考慮しつつ直接人手で描画することは極めて困難となり、非常に多くの時間を必要とするようになってきている。このため、高性能の自動レイアウト設計（パターン設計）の要求がますます高まりつつある。フルカスタムVLSIに対しては、マクロセル方式による設計が行われている<sup>(1)</sup>。マクロセル方式では、機能モジュール（例えば、マイクロプロセッサのROM、RAM、プログラムカウンタ、インストラクションレジスタ、インストラクションデコーダ等）の物理的なマスクパターンであるマクロセルを設計し、これらのマクロセルの間をアバットメントや自動配線により接続することによりVLSIを実現する。マクロセル内のトランジスタレベルのレイアウト設計において、設計期間の短縮、誤まりのない設計の実現、及びパターン設計ルールの微細化に伴うレイアウト更新の容易化に極めて大きく寄与するのがシンボリックレイアウト手法である。シンボリックレイアウト手法とは、トランジスタ、コンタクト、配線などのレイアウト設計の基本要素に対してシンボルを定め、これらのシンボルの相対的な位置関係によってレイアウトを表現し、それらの間のコンパクションを行って最終的なマスクパターンに変換する手法である。

本章では、マクロセルのレイアウト設計を短期間に、かつ高密度に行うことができ、しかもレイアウト結果が製造プロセスの進歩に伴うパターン設計ルールの変更に充分追従できるようなシンボルの定義方法とシンボルの配置手法、及びコンパクション手法について考察する。更に、このシンボリックレイアウト手法を実際に16ビットマイクロコントローラのレイアウト設計に適用した結果を述べ、本手法が有効であることを示す。

### 8.2 シンボリックレイアウトの概要

シンボリックレイアウト手法の流れの概要を図8.1に示す。この手法は、入力・編集、接続照合、マスクパターン生成及び回路パラメータ抽出の段階に分かれる。

シンボリックレイアウト手法の概要を以下に説明する。まず、設計者は、計算機と接続されたカラーグラフィック端末上で、スティック図<sup>(2)</sup>と呼ばれるシンボリックレイアウト図の入力・編集を会話的に行う。ここで、シンボリックレイ

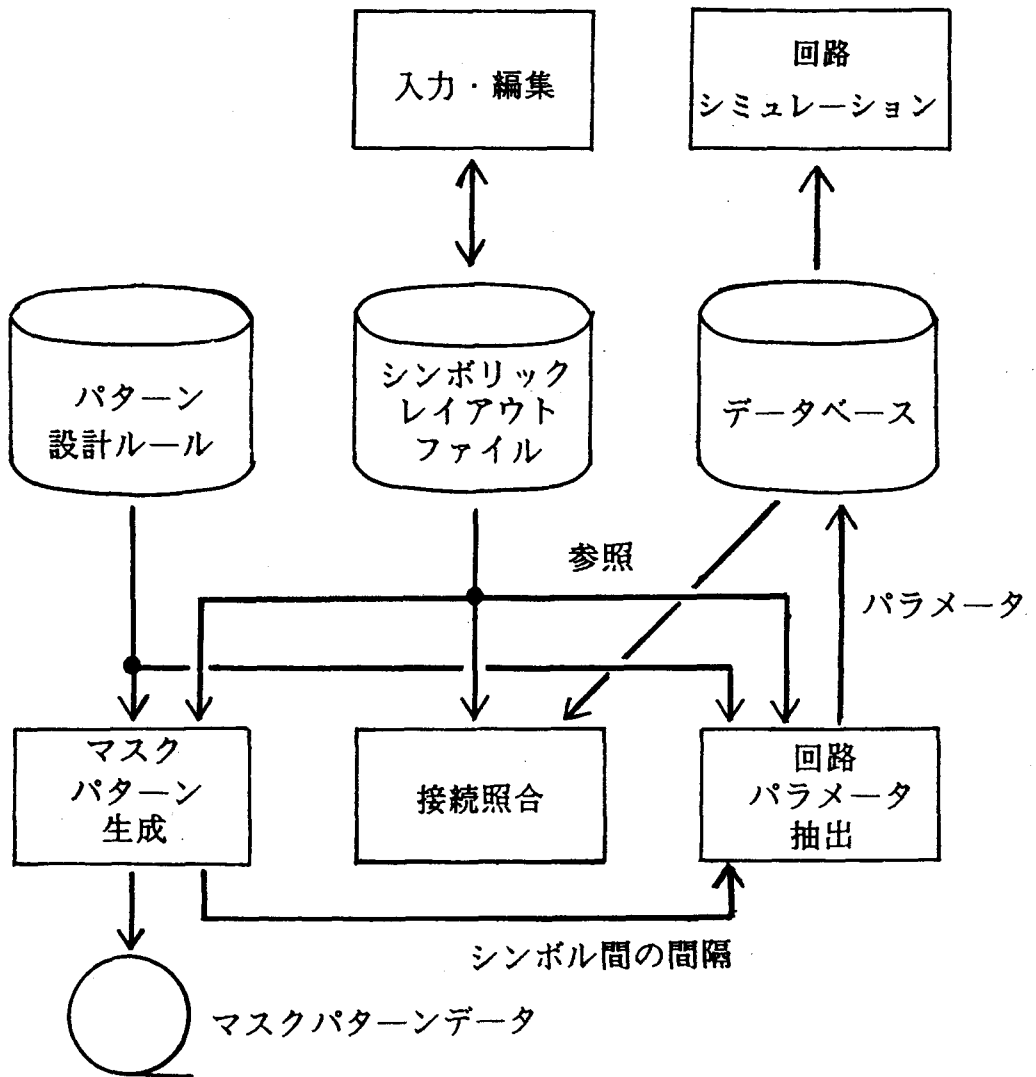


図8.1 シンボリックレイアウト手法の構成

アウト図とは、トランジスタ、コンタクト、配線などレイアウト設計の基本要素に対してそれぞれシンボルを定め、これらのシンボルの相対的な位置関係によって表現したマクロセルのレイアウト図のことである。この段階では、入力されたシンボリックレイアウト図におけるシンボル間の実際の幾何学的な間隔（距離）は確定していない。接続照合<sup>(3)</sup>ではシンボリックレイアウト図より抽出されるトランジスタ間の接続情報と予めデータベースに格納されている参照用のトラン

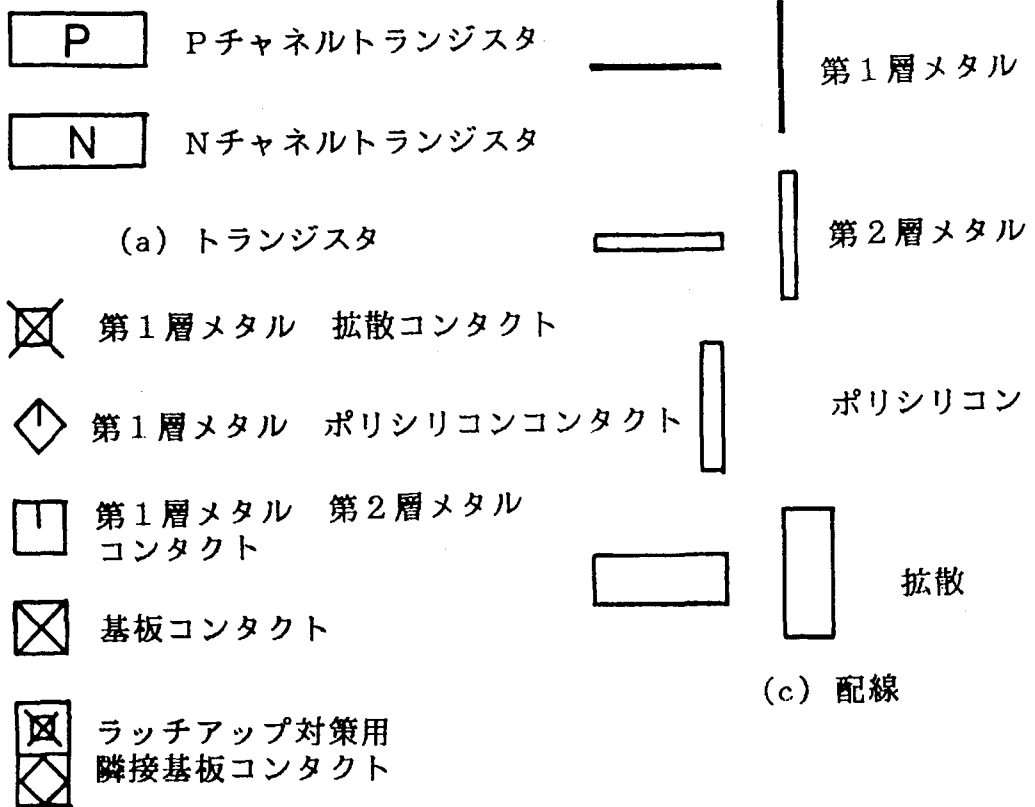
ジスタ間の接続情報を比較照合する。パターン設計ルールファイルには、トランジスタのチャンネル幅、チャンネル長等のサイズ、コンタクトのサイズ、配線幅、各層のパターン間に必要な最小限の間隔等の物理的な大きさやパターン設計ルールに関する情報が記述されている。マスクパターン生成では、パターン設計ルールファイルを入力してシンボリックレイアウト図のシンボル間のコンパクションを行い、最終的なマスクパターンを作成する。この時点で初めてマクロセル内のシンボル間の実際の幾何学的な間隔（距離）が確定する。この位置情報を回路パラメータ抽出の際に用いる。回路パラメータ抽出では、信号毎の配線容量及び設計されたマクロセルの大きさとその外部ピンの位置の情報をレイアウト結果から抽出し、データベースに格納する。配線容量の情報は信号遅延を考慮した回路シミュレーション及び論理シミュレーションを行うために用いられる。マクロセルの大きさとその外部ピンの位置はセル間の自動配置配線を行うために用いられる。

### 8.3 シンボリックレイアウト図におけるシンボル配置手法

本節では、シンボリックレイアウト図におけるシンボル配置手法について考察する。本手法では、格子点上に図8.2に示したシンボルを配置することによってシンボリックレイアウト図を描くといういわゆるvirtual grid方式<sup>(4)</sup>を採用する。その理由は、以下に述べるように、入力・編集時のグラフィック・コマンドに対する応答や、コンパクションと回路パラメータ抽出の処理が高速になるためである。

格子点上にシンボルを配置することによって記述したシンボリックレイアウト図はmatrix data structure<sup>(4)</sup>に基づいた内部データ構造 $M(i, j, k)$ に格納できる。図8.3に示すように、要素 $M(i, j, k)$ は、タイプ $k$ のシンボルがシンボリックレイアウト図の格子点の配列の第 $i$ 行、第 $j$ 列の格子点上に配置されているか否かを表わす。ここで、例えば、 $k=1$ は第1層メタル配線、 $k=2$ は第2層メタル配線、 $k=3$ はトランジスタを示す。配線のシンボルの場合には、各配線シンボルに特定の正整数を与え、この値を要素 $M(i, j, k)$ に格納する。このデータ構造の特長は、隣接する格子点に関する情報をデータ構造中でも隣接した箇所に格納することと、データ構造 $M(i, j, k)$ の添字を指定するだけで格子点上に置かれているシンボルに直接アクセスできることである。このため、このデータ構造を用いれば、入力・編集のグラフィック・コマンドに対する応答や、コンパクションと回路パラメータ抽出の処理が速くなる。

更に、設計時間を短縮し、処理速度を高速化するために、シンボリックレイア



(d) ポリゴン

図8.2 シンボリックレイアウト手法におけるシンボル

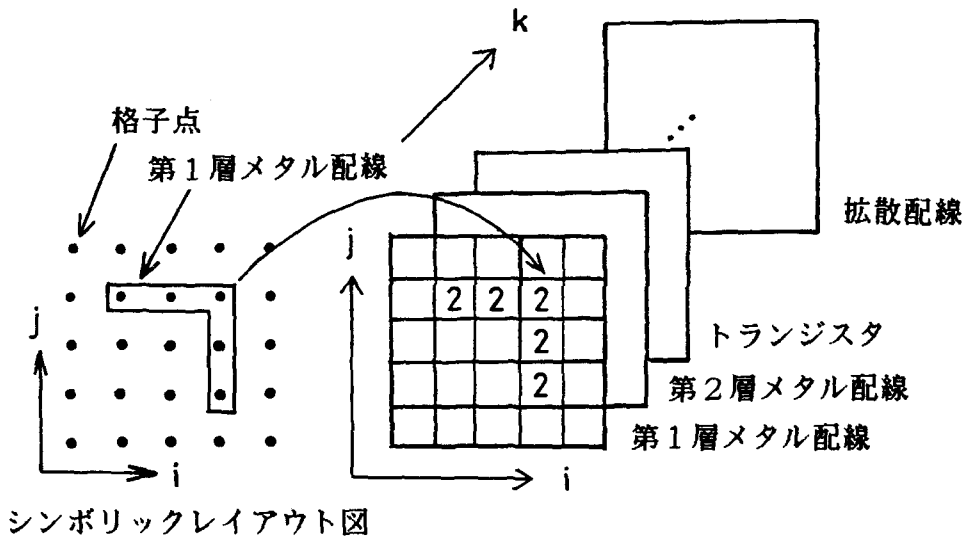


図8.3 シンボリックレイアウト図を格納する内部データ構造  $M(i, j, k)$

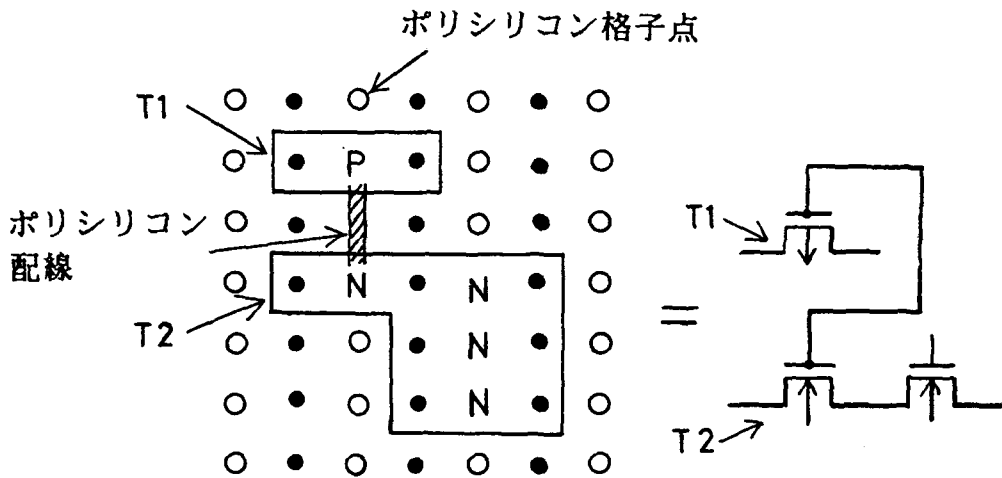


図8.4 トランジスタシンボル

ウト図においてゲートマトリックス<sup>(5)</sup>と類似した規則性のある構造を持たせる。ゲートマトリックスとの共通点は以下の通りである。

(1) ポリシリコン配線やトランジスタのゲートは1列おきの垂直格子点列(図8.4の○印の格子点で、これをポリシリコン格子点と呼ぶ)上のみ配置可能である。ポリシリコン以外のメタル、拡散配線は任意の格子点上に水平、垂直両方向に可能である。但し、斜め配線は許さない。

(2) トランジスタはポリシリコン格子点上のポリシリコンパタン及びその左右両側の隣接する格子点上の拡散パターンよりなる(図8.4のトランジスタT1参照)。トランジスタのシンボルを回転、反転しては置けない。図8.4に示すように垂直方向に隣接する複数のトランジスタはチャンネル幅の大きい1つのトランジスタとみなされる。水平方向に隣接するトランジスタはソース/ドレイン領域を共有するトランジスタの連なりとみなされる(図8.4のトランジスタT2参照)。

高い集積度を達成でき、広範囲なVLSIのレイアウト設計に適用可能なように、ゲートマトリックスに次の拡張を行った。

(1) 配線層として、第1層メタル、P拡散層、N拡散層及びポリシリコン層の他に、第2層メタルも取り扱える。

(2) 格子点間の間隔は固定でなく、一様でもない。これは、マスクパターン生成のコンパクションにより決定される。

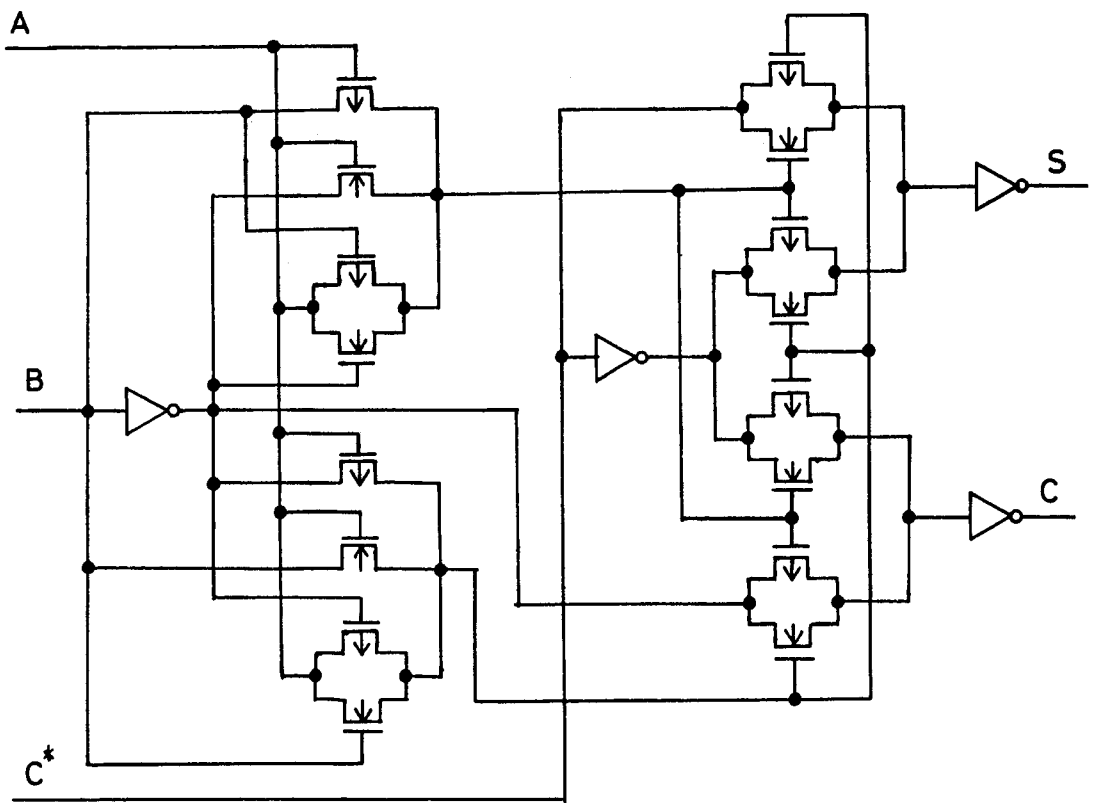
(3) チャンネル長及びチャンネル幅の異なるトランジスタを何種類でも定義できる。

(4) 拡散配線のシンボルはP拡散配線又はN拡散配線を表わすために使われる。拡散配線がP型かN型かは、拡散配線がPチャンネルトランジスタにつながるかNチャンネルトランジスタにつながるかに着目することにより、自動的に識別される。同様に、第1層メタル・拡散コンタクトのシンボルも第1層メタル・P拡散コンタクト又は第1層メタル・N拡散コンタクトを表わすために用いられる。これも自動的に区別される。

図8.5はMGXを用いて設計された全加算器のシンボリックレイアウト設計の例である。

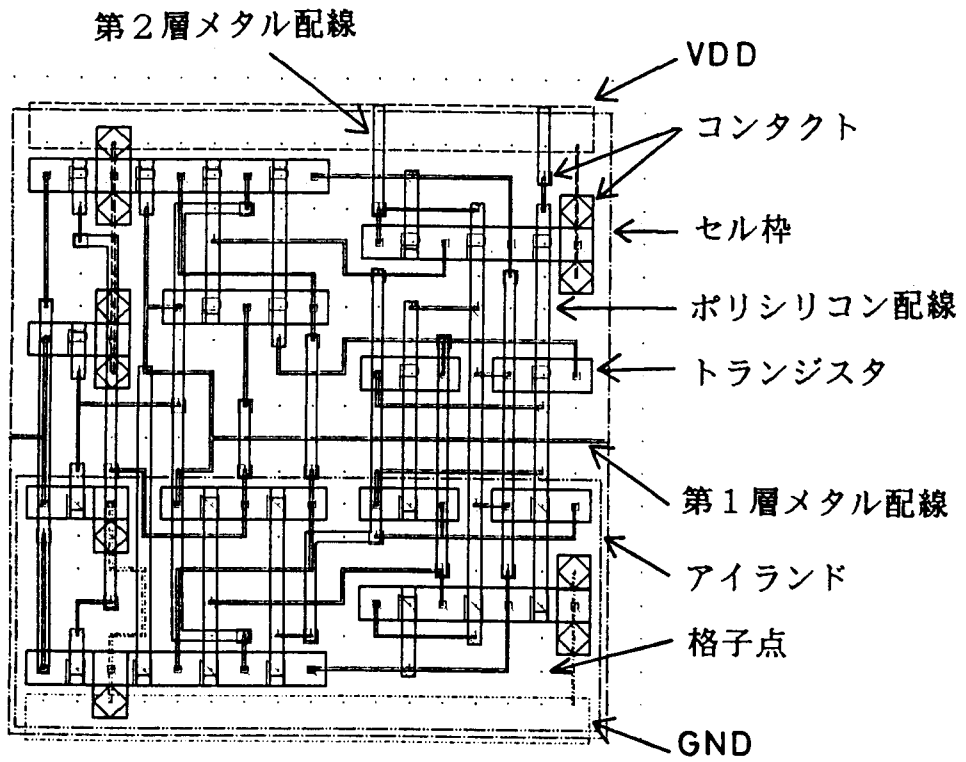
#### 8.4 コンパクションの手法

マスクパターン生成では、パターン設計ルールを入力して、シンボリックレイアウト図におけるシンボル間のコンパクションを行い、最終的なマスクパターンを作成する。シンボリックレイアウト図において、トランジスタのゲートサイ

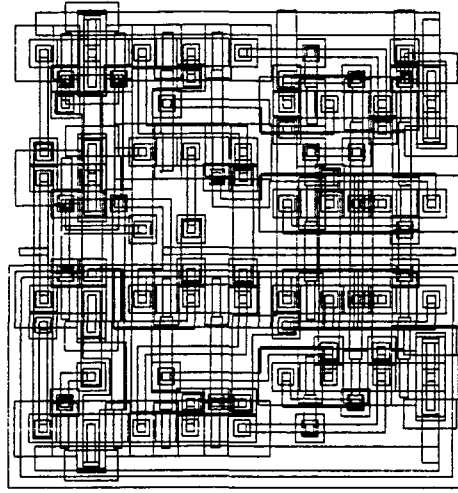


(a) 論理回路図





(b) シンボリックレイアウト図



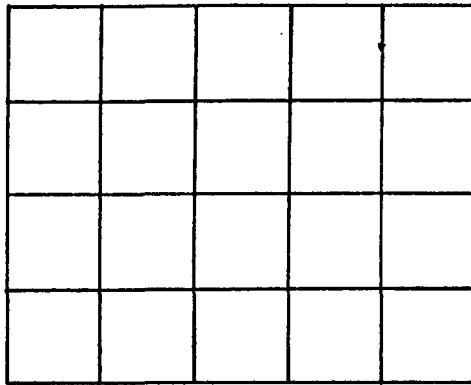
(c) マスクパターンデータ

図8.5 シンボリックレイアウト手法により設計した全加算器

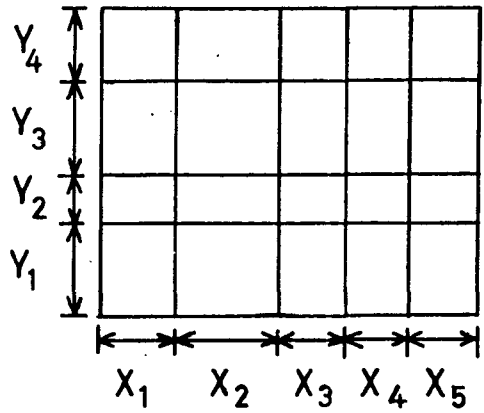
ズ、コンタクトのサイズ、配線の幅及びVDD、GNDのポリゴンのサイズ等は設計者によりパターン設計ルールとして指定された値に固定されるが、それ以外の配線の長さやシンボル間の間隔は可変である。これらの物理的寸法はコンパクションによって決定される。

従来、多くのコンパクションの手法が提案されているが、主な手法は、constraint graph法<sup>(6)-(8)</sup>、shear-line法<sup>(7),(8)</sup>及びvirtual grid法<sup>(4),(8)</sup>である。この内、virtual grid法では、すべてのシンボルは移動可能な格子点上にあり、隣接する格子線上のシンボル間の最小許容間隔値を考慮することにより、図8.6のように各格子線間の間隔 $X_1, X_2, \dots, X_5, Y_1, Y_2, \dots, Y_4$ を決定する。この処理はX方向とY方向に独立に行う。一般に、virtual grid法は他の手法に比べて、処理速度は速いが、同一格子線上に配置されたシンボルは同時に動くので、マクロセルのサイズが大きくなることがある<sup>(8)</sup>。しかし、ここでは次の2つの理由により、コンパクション処理にvirtual grid法を採用する。

(1) virtual grid法を用いた場合、コンパクション処理はシンボリック図の格子線の間隔を決めるだけで、コンパクション処理後も格子線の平行性は保たれる。従って、virtual grid法を採用した場合、回路パラメータ抽出ではマスクパ



(a) コンパクション前



(b) コンパクション後

図8.6 virtual grid法によるコンパクション

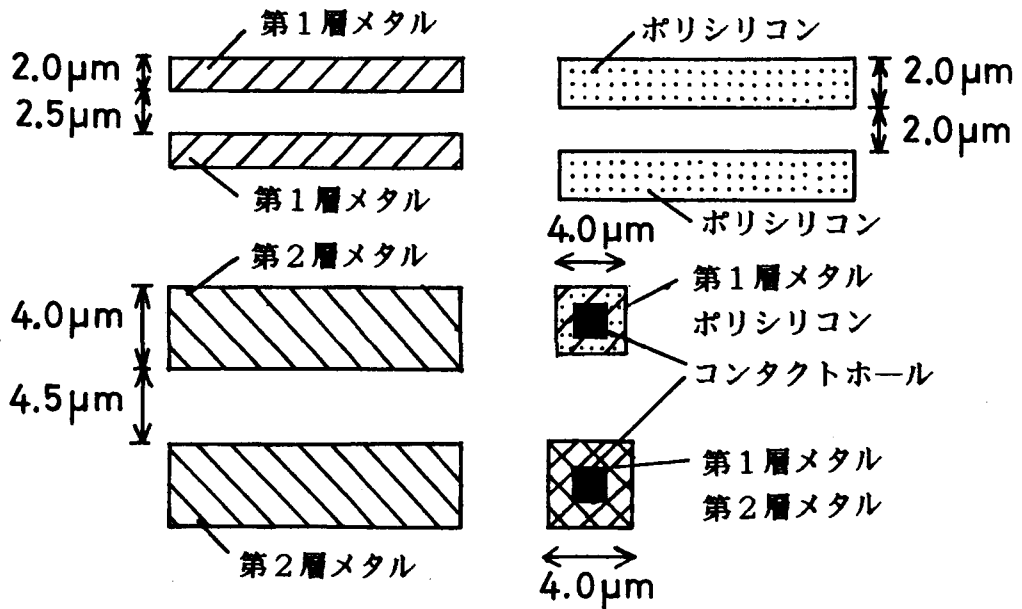


図8.7 2 μmパターン設計ルール例

ターンデータを直接解析する必要はなく、シンボリックレイアウト図とコンパクション処理により決定した格子間隔値から回路パラメータを計算できる。このため、マスクパターンデータに対するトランジスタの認識等の図形演算が不必要となり、回路パラメータ抽出の処理速度は速くなる。

(2) virtual grid法は、他の2つの方法より単純であり、そのコンパクション後の結果が予想しやすい。このため、設計者がコンパクション後の結果を予想し、コンパクションにとってよりよい入力データ（シンボリックレイアウト図）を作成することにより、レイアウト結果を実用上支障ない程度にまで縮小することができる。

上記(2)を、簡単な例を挙げて説明する。図8.7の $2\mu\text{m}$ パターン設計ルールを使って図8.8に示すレイアウト設計を行う場合を考える。第2層メタル配線の中心線間は $8.5\mu\text{m}$ の間隔が保持されなければならない。どの2つの第2層メタル配線のシンボルも隣接する垂直格子線上に置かれていない場合には、コンパクションによって決められる垂直格子線間隔はすべて $6.5\mu\text{m}$ 以下である。こ

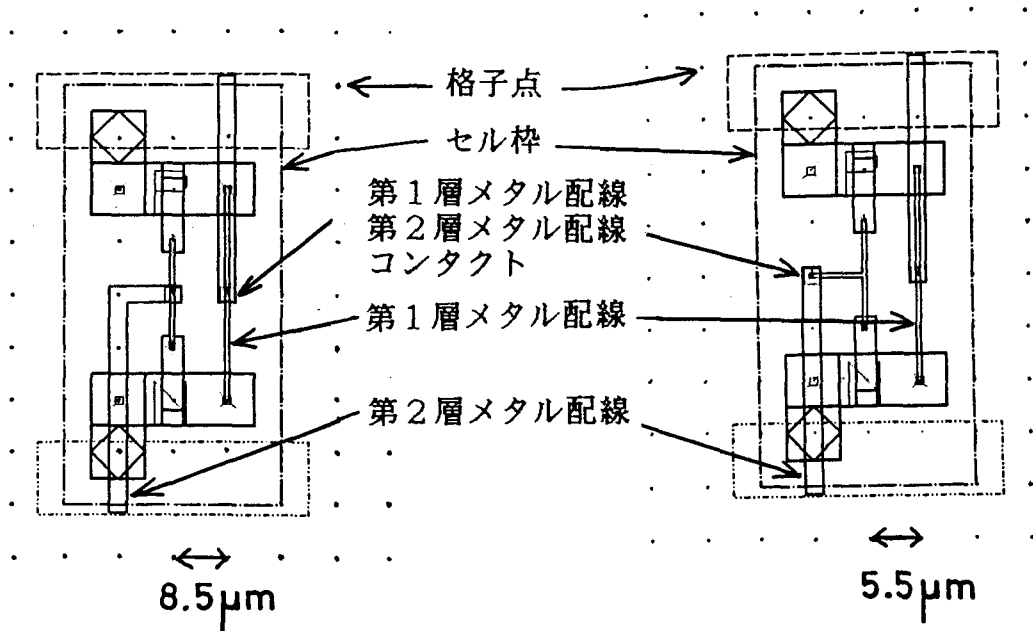


図8.8 2つの第2層メタル配線を隣接する格子線上に置くとレイアウト結果が大きくなる例

ここで、 $6.5\mu\text{m}$ の間隔は、2つのコンタクトの中心間の最小許容間隔である。実際のシンボリックレイアウト設計においては、どの2つの第2層メタル配線も隣接する格子線上には置いてはいけないという制約が課される。大部分の場合、レイアウト設計に使う第2層メタル配線の本数は少ないので、この制約を課しても図8.5(b)で示されたセル枠内の格子点の総数は増えない。こうして、この制約を課すことにより、図8.8に示すように、レイアウト結果は更に小さくなる。

## 8.5 適用結果と性能改善

前節までに述べたシンボリックレイアウト手法に基づき、大型計算機IBM 3084と接続されたカラーグラフィック端末上で実行可能なシンボリックレイアウトシステムMGXを開発した。MGXを使って、67,400トランジスタを含む16ビットマイクロコントローラを設計し、良好な結果が得られた<sup>(9)</sup>。

本節では、まず提案したシンボリックレイアウト手法の適用結果について述べ、次にこの適用結果に基づいて行った性能改善の内容について述べる。

### 8.5.1 16ビットマイクロコントローラの論理構成

16ビットマイクロコントローラの論理構成を図8.9に示す。すなわち、マイクロコントローラはI/Oポート部、RAM、マイクロプログラムROM、データバス部、データバス制御部、タイマー部、UART部 (Universal Asynchronous Receiver and Transmitter)、割り込み制御部等から構成されており、これらが16ビット幅のデータバスと24ビット幅のアドレスバスにより接続されている。

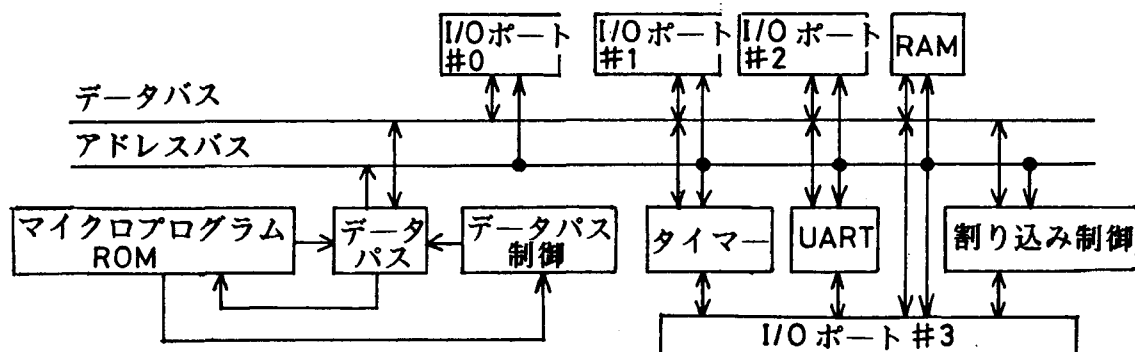


図8.9 16ビットマイクロコントローラの論理構成

### 8.5.2 16ビットマイクロコントローラのレイアウト設計結果

2 $\mu$ mルールのコモス2層配線技術を用いて16ビットマイクロコントローラのレイアウト設計を行った。16ビットマイクロコントローラチップは表8.1の14個のマクロセルに分けられる。各マクロセルの規則化率、レイアウト手法、トランジスタ数を表8.1に示す。レイアウト設計に当たっては、まず各マクロセルの面積、各マクロセルの縦横比、データフロー及びコントロールのフローを考慮して、図8.10のようなチップフロアプランを作成した。次にこの結果に基づき、規則化率の高いRAMとマイクロプログラムROMは高集積化をはかるため人手設計し、それ以外のマクロセルはMGXを用いてレイアウト設計した。その後、これらのマクロセルを配置配線した。

MGXは大型計算機上で他のバッチジョブと時分割で実行されるため、入力・編集のグラフィック・コマンドに対する応答が遅い上に、規模の大きいセルの入力・編集を行うとコマンド応答時間が更に長くなる。この理由のため、マクロセルの設計の際には、まずMGXで高々40トランジスタ程度の比較的小さな規模の基本セルを設計し、次に、これらの基本セルをセルのアバットメントの機能を有した自動配線プログラムMILD<sup>(10)</sup>を使用して、アバットメント及び自動配線を階層的に繰り返すことによりマクロセルを構成した。

表8.2はマクロセルのレイアウト設計結果を示している。表8.2の基本セルの平均集積度をみると、データバス制御部の基本セルは他のマクロセルに比べ、かなり平均集積度が高くなっているが、これはデータバス制御部の回路が採用したPLA構成に適していたことによる。しかし反面、データバス制御部においては、基本セル間の配線本数が多いため、マクロセル全体の集積度は低くなった。なお、MGXを用いて基本セルを設計すると約1,300トランジスタ/ $\text{mm}^2$ の集積度が得られ、これは人手設計の場合の約70%の集積度である。なお、チップ全体の集積度はRAM、マイクロプログラムROMを含めた場合1,070トランジスタ/ $\text{mm}^2$ で、RAM、マイクロプログラムROMを含まない場合は588トランジスタ/ $\text{mm}^2$ であった。

表8.2に示す設計工数とは、データベースへのマクロセルの論理図入力、マクロセルのレイアウト設計とその検証に要した工数である。マクロセル間の配線に要した工数も含めると、本LSIチップの設計に要した工数は14 man-monthsであった。MGXを用いると、基本セルの設計に要する時間は人手設計の場合の約1/3.5に短縮され、一人一日当たり約9トランジスタの割合で基本セルを設計できた<sup>(9)</sup>。図8.11に本マイクロコントローラのチップ写真を示す。チップサイズ

表8.1 各マクロセルの規則化率，レイアウト手法とトランジスタ数

マクロセル	規則化率	レイアウト手法	トランジスタ数
R A M	489.0	人手レイアウト	26,092
マイクロプログラムROM	47.0	人手レイアウト	10,400
データバス	8.1	MGXとMILD	5,934
データバス制御	2.0	MGXとMILD	1,897
タイマ	2.9	MGXとMILD	3,171
I/Oポート #0	11.0	MGXとMILD	1,101
I/Oポート #1	4.1	MGXとMILD	1,286
I/Oポート #2	11.0	MGXとMILD	1,154
I/Oポート #3	3.7	MGXとMILD	1,237
割り込み制御	2.6	MGXとMILD	2,206
UART	2.7	MGXとMILD	3,528
クロックジェネレータ	6.5	MGXとMILD	451
リセット制御	1.0	MGXとMILD	67
チップ制御	1.0	MGXとMILD	328

$$\text{規則化率} = \frac{\text{基本セルの総数}}{\text{実際にレイアウトした基本セルの数}}$$

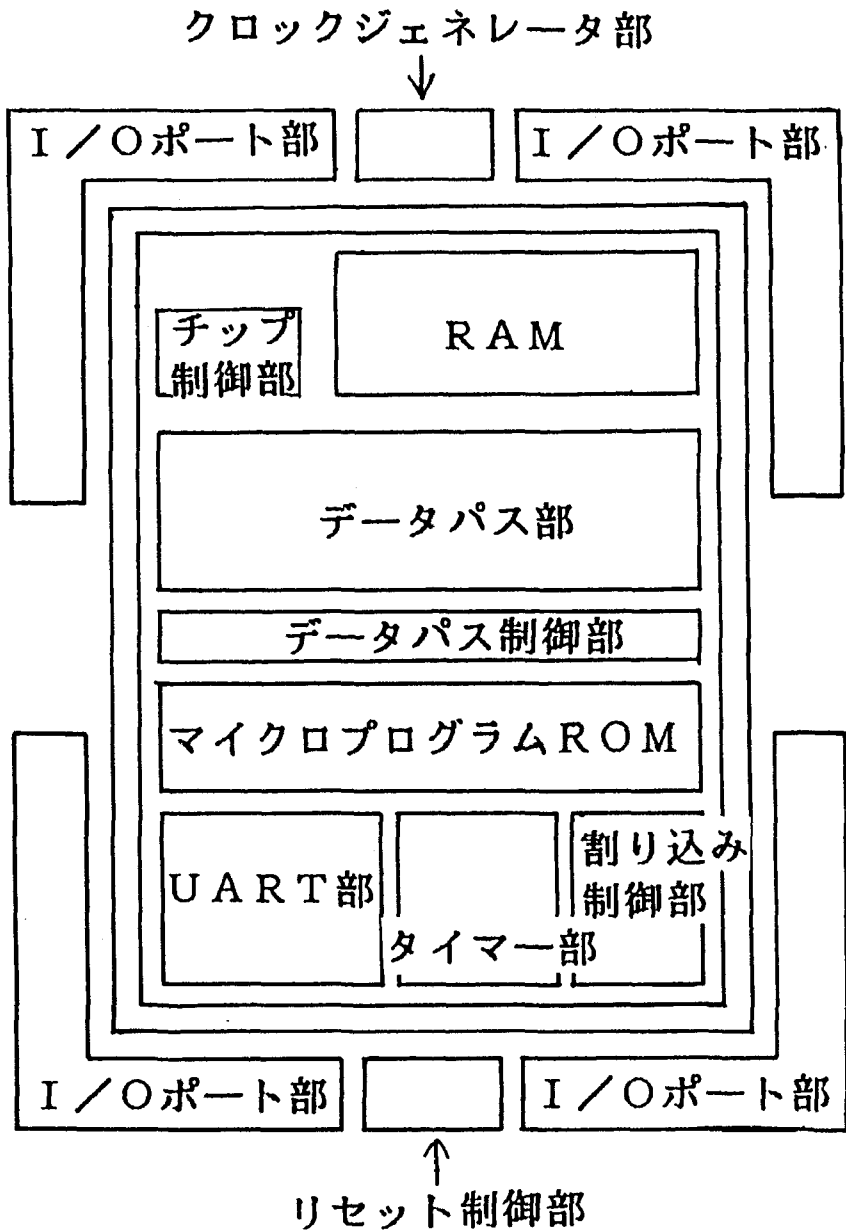


図8.10 16ビットマイクロコントローラのチップフロアプラン



表8.2 マクロセルのレイアウト設計結果

マクロセル	セルサイズ ( $\mu\text{m}^2$ )	マクロセルの 集積度 (トランジスタ/ $\text{mm}^2$ )	基本セルの 集積度の平均 (トランジスタ/ $\text{mm}^2$ )	設計工数 (man- months)
RAM	3,868 × 1,488	4,538	—	1.0
マイクロ プログラム ROM	4,531 × 1,083	2,124	—	2.0
データバス	5,014 × 1,168	1,013	1,202	1.5
データバス 制御	4,524 × 422	993	2,200	1.5
タイマー	1,479 × 1,867	1,148	1,350	2.0
割り込み 制御	995 × 1,766	1,256	1,348	1.8
UART	2,244 × 2,071	760	1,320	1.2

表8.3 図8.14(a)-(c)の第1層メタルに関するパターン設計ルールの記事

マスクパターン	マスクパターン	最小許容間隔
第1層メタル配線シンボルの第1層メタルパターン	第1層メタル配線シンボルの第1層メタルパターン	2.5 $\mu\text{m}$
第1層メタル配線シンボルの第1層メタルパターン	太い第1層メタル配線シンボルの第1層メタルパターン	3.0 $\mu\text{m}$
第1層メタル配線シンボルの第1層メタルパターン	第1層メタル・ポリシリコンコンタクトシンボルの第1層メタルパターン	2.0 $\mu\text{m}$

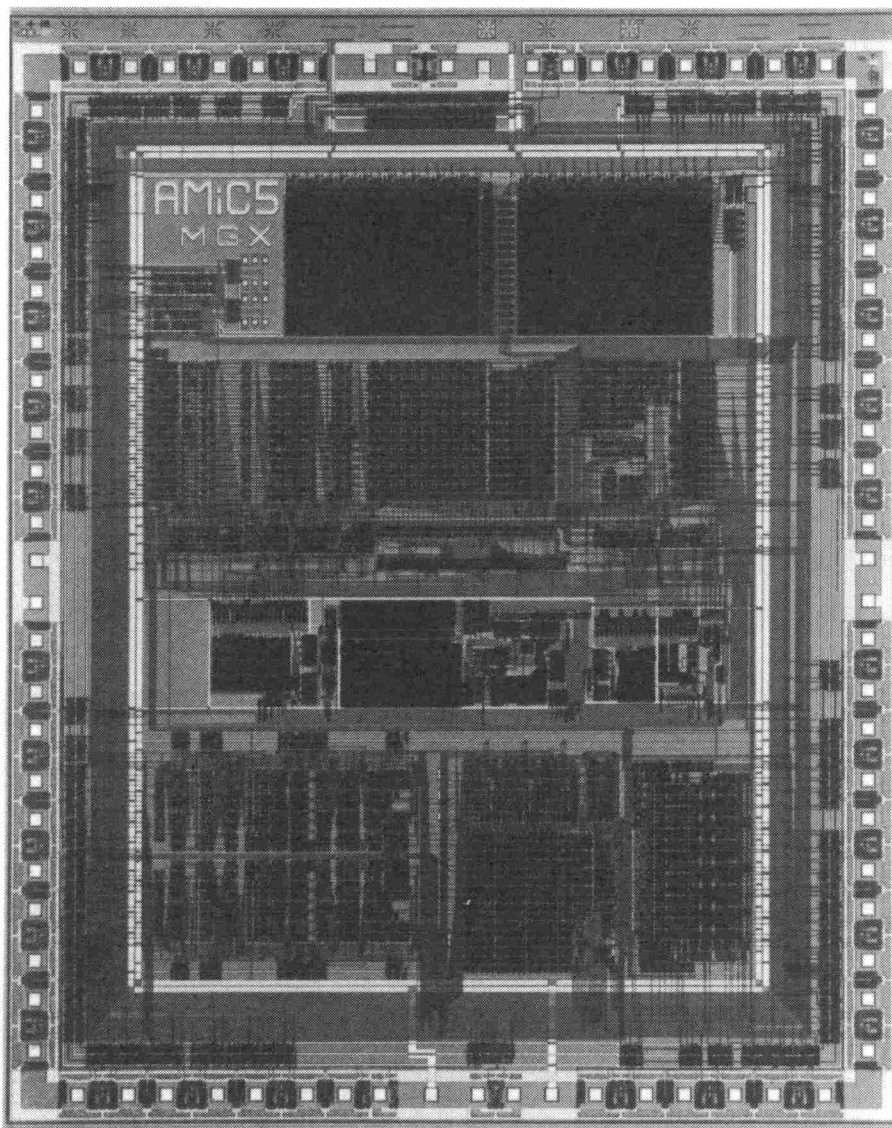


図8.11 16ビットマイクロコントローラのチップ写真

は $7.50 \times 9.26 \text{mm}^2$ である。

なお、MGXの処理は非常に高速であり、例えば、50トランジスタ程度の規模の基本セルを設計する場合、7.5MIPSの大型計算機上でマスクパターン生成の処理時間が約5秒、接続照合と回路パラメータ抽出の処理時間が約5秒であった。

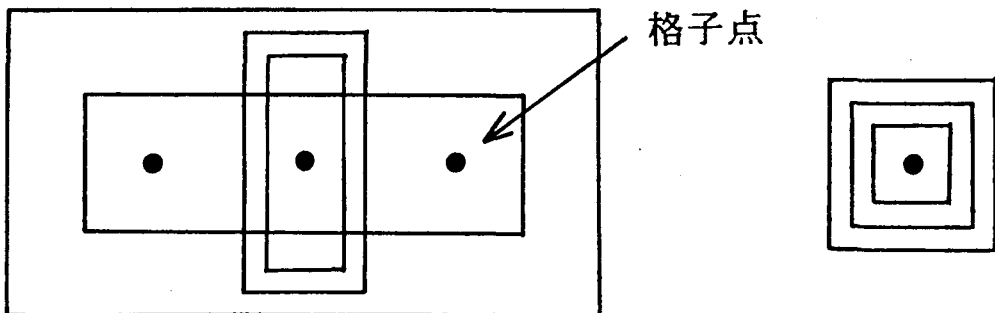
これらの適用結果から、提案したシンボリックレイアウト手法を用いるとマクロセルのレイアウト設計を短期間に、かつ高密度に行うことができることを確認した。

### 8.5.3 性能改善

第8.5.2節で述べた適用結果に基づき、レイアウト設計結果の設計品質を更に向上させるために、提案したシンボリックレイアウト手法に以下の2つの性能改善を行った。

#### (A) パターン設計ルールに依存しないレイアウト設計

VLSI製造技術の進歩と共にパターン設計ルールが変化しても適用可能な手法とするために、任意の数のマスクレイヤが定義でき、任意のマスクレイヤ上の種々の幅の配線のシンボル、及び任意の2つのマスクレイヤ上のパターン間を接続するコンタクトのシンボルを定義できるようにした。トランジスタやコンタクトなどのシンボルのマスクパターンは図8.12に示すように適当なマスクレイヤ上の互いに重なった長方形の図形の集合として表現する。更に、パターン設計ルールの記述法に自由度を持たせた。例えば、第1層メタルの配線シンボルを2種類以上定義している場合には、表8.3のように、第1層メタルの配線シンボルの任



(a) トランジスタ

(b) コンタクト

図8.12 シンボルに対応するマスクパターン

意の組合せに対しそれぞれ別の最小許容間隔値を指定できるようにした。

以上のようなシンボリックレイアウト手法は、VLSI製造プロセスの進歩と共にマスクの数が増加しても適用可能である。例えば、改良後の手法はCMOSメタル3層配線技術を用いたパターン設計ルールも扱える。図8.13は、各トランジスタ毎に、チャンネル領域がチャンネルドーピングプロセスによりドーパされる製造プロセスを示す。改良後の手法はこのような製造プロセスに対しても適用可能で、この場合、チャンネルドーパのマスクレイヤを定義し、このレイヤ上の長方形の図形をトランジスタのシンボルのマスクパターンに追加すればよい。更に、VLSI製造技術の進歩と共にパターン設計ルールがさらに複雑になっても、改良後の手法は適用可能である。図8.14は複雑なパターン設計ルールの例である。図8.14(a), (b), (c)のパターン設計ルールを表8.3のように記述する。改良後の手法はこれらの図8.14(a), (b), (c)のパターン設計ルールを扱うことはできるが、図8.14(d)のように、ある配線が他の配線の上に置かれているか否かによって最小許容間隔が変わるようなパターン設計ルールは扱えない。

#### (B) 高い集積度を達成するシンボル配置手法

処理の高速化のため、シンボリックレイアウト図では格子点上にシンボルを配置するが、高い集積度を達成するため、ゲートマトリックス構造という制限を取り除く。すなわち、シンボリックレイアウト図において、次の2点の改善を行った。

(1) ポリシリコン配線もメタルや拡散の配線と同様に任意の格子点上に水平、垂直両方向に可能とする。

(2) トランジスタ、コンタクトのシンボルも、任意の格子点上に回転（回転角度は90度の倍数に制限）及び反転して配置可能とする。

改良後のシンボリックレイアウト手法を用いて、一般に広く使われている論理素子であるDフリップフロップをレイアウト設計した。図8.15, 8.16はそれぞれ改良前と改良後の手法を使って設計したDフリップフロップのシンボリックレイアウト図である。図8.16では、ポリシリコン配線を折り曲げ、トランジスタのシンボルを回転して配置した。図8.17はそのマスクパターンである。Dフリップフロップのレイアウト設計において、改良の結果として集積度は約39%増大した。

この実験結果から、この改良後のシンボリックレイアウト手法は非常に高い集積度を達成することを確認した。しかし、改良後の手法によるレイアウト結果においても、依然として次の2つの理由により人手設計の結果よりも集積度が低い。

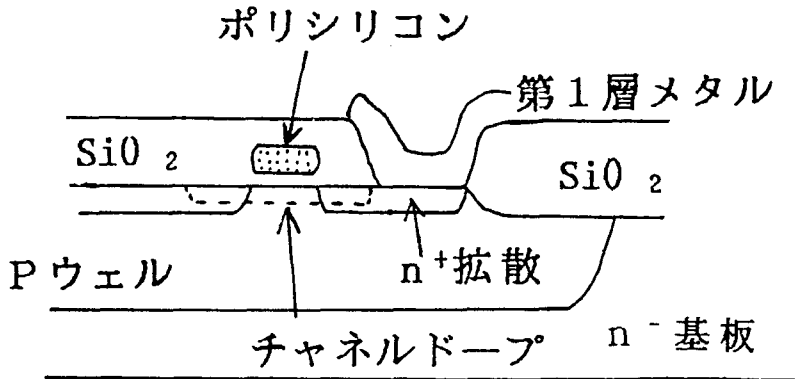


図8.13 各トランジスタのチャンネル領域がチャンネルドーピングプロセスによりドーピングされる製造プロセス

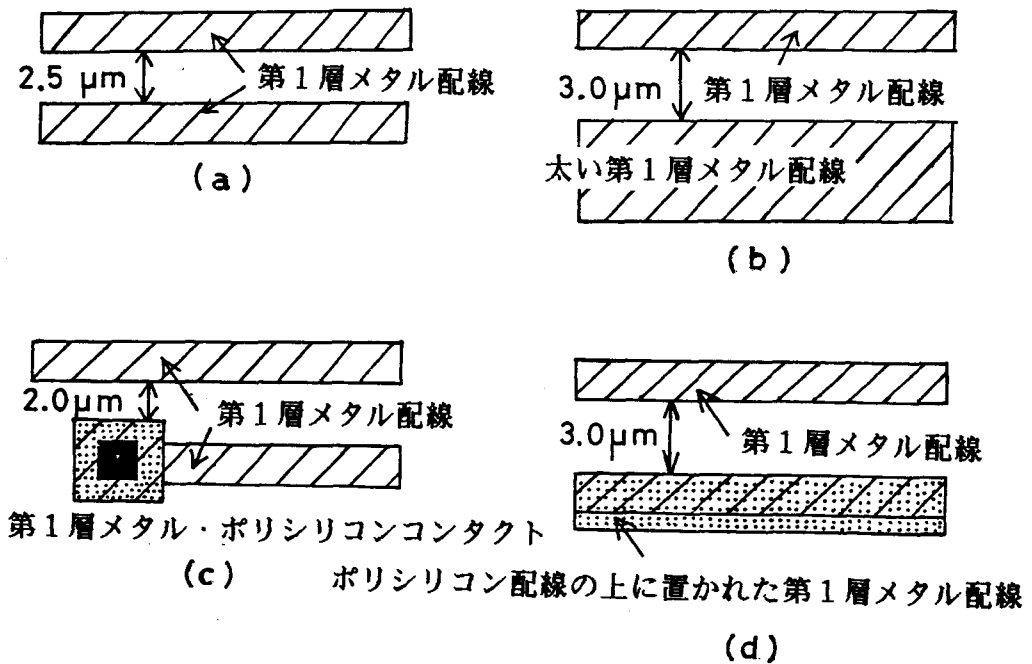


図8.14 複雑なパターン設計ルール例

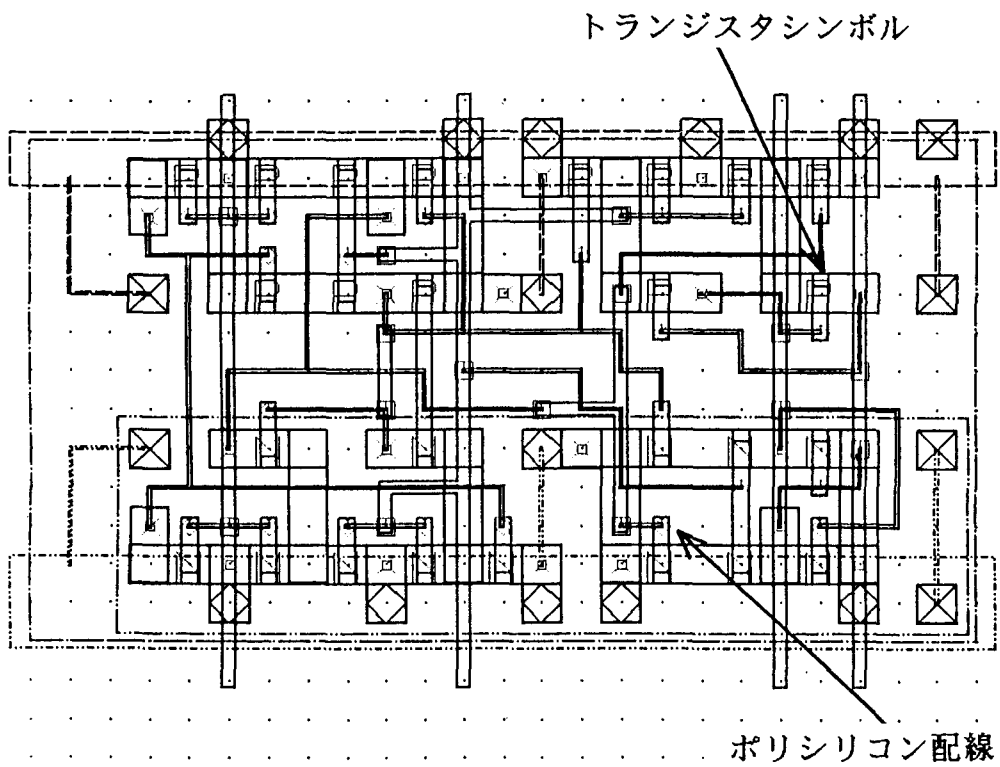


図8.15 改良前の手法によるシンボリックレイアウト図

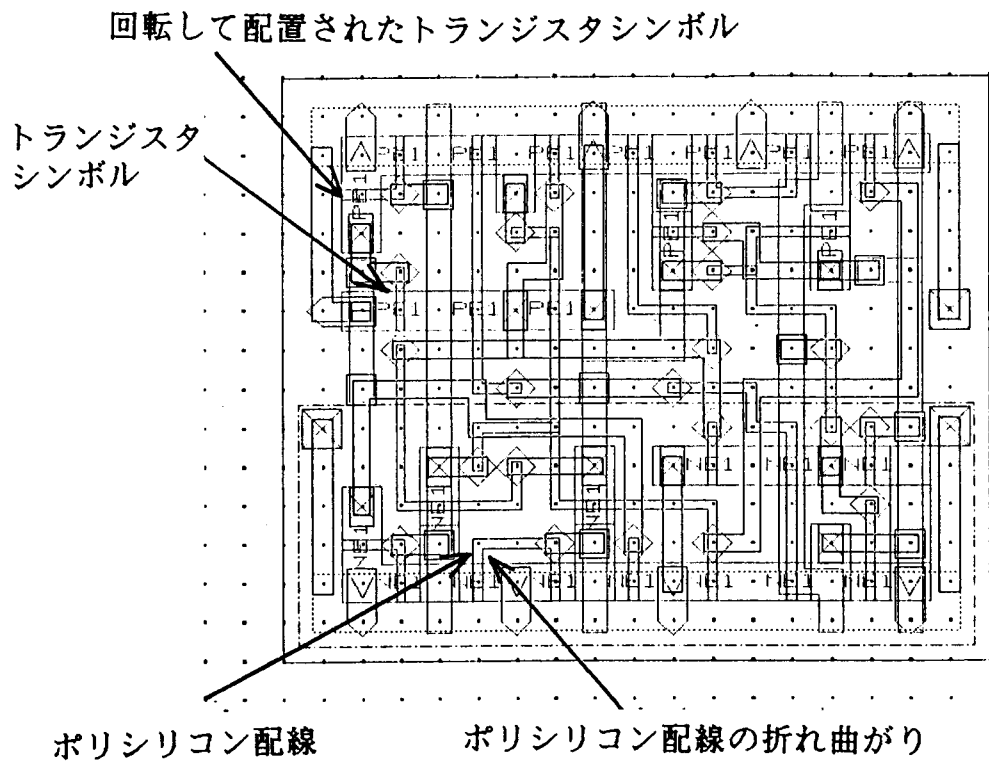
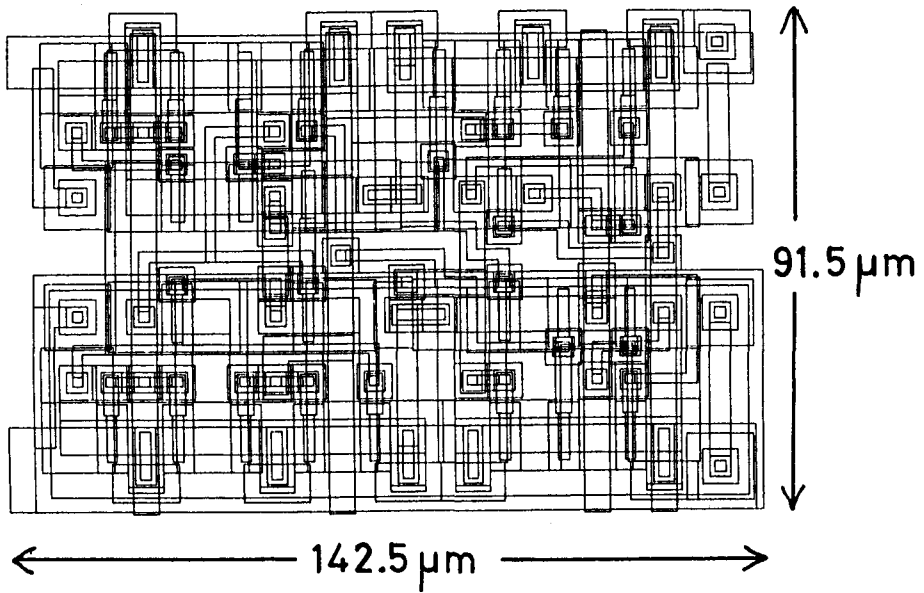
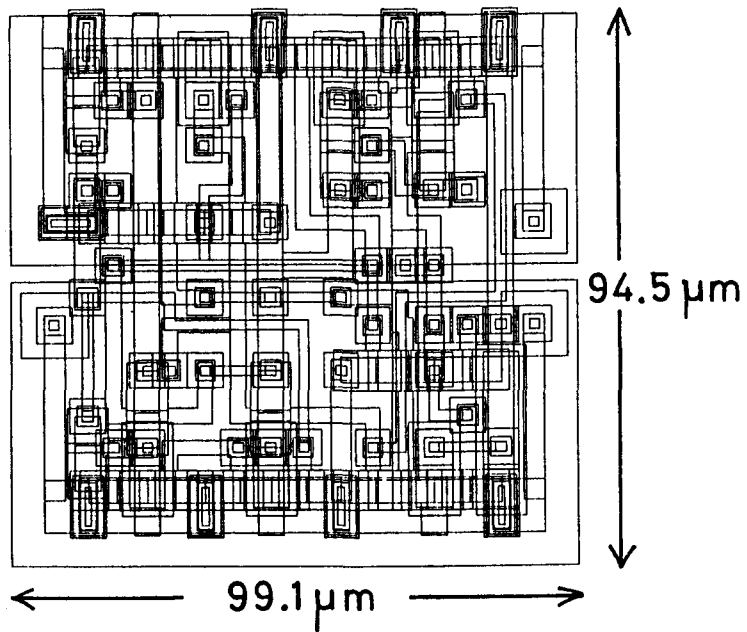


図8.16 改良後の手法によるシンボリックレイアウト図



(a)



(b)

図8.17 (a) 改良前の手法によるマスクパターン  
(b) 改良後の手法によるマスクパターン



(1) シンボリックレイアウト図において、すべてのシンボルは移動可能な格子点上にあるため、同一格子線上に置かれたすべてのシンボルはコンパクションで同時に移動しなければならない。

(2) 斜め配線は許されない。

## 8.6 結 言

本章では、フルカスタムVLSIやマイクロプロセッサ等のマクロセルをトランジスタレベルでレイアウト設計するためのシンボリックレイアウト手法について考察した。ここでは、マクロセルのレイアウト設計を短期間にかつ高密度に行うことができ、しかもレイアウト結果が製造プロセスの進歩に伴うパターン設計ルールの変更に充分追従できるようなシンボルの定義方法とシンボルの配置手法、及びコンパクション手法について考察した。更にこのシンボリックレイアウト手法を実際に16ビットマイクロコントローラのレイアウト設計に適用した結果を示し、本手法が有効であることを示した。

なお、本章の内容は文献(11)に基づいている。

## 参考文献

- (1) J.Korematsu, T.Tokuda, Y.Shimazu, Y.Saitoh, O.Tomisawa, and T.Enomoto: "A macrocell approach for microcomputer design", Proc. 1984 ICCD, pp. 280-283(1984).
- (2) J.D.Williams: "STICKS-A graphical compiler for high level LSI design", Proc. 1978 Nat. Comput. Conf., pp. 289-295(1978).
- (3) K.Sato, M.Terai, T.Noda, and M.Ozaki: "A method of connectivity checking for MOS circuits considering logical equivalence", Proc. 1985 ISCAS, pp. 1285-1288(1985).
- (4) N.Weste: "Virtual grid symbolic layout", Proc. 18th DA Conf., pp. 225-233(1981).
- (5) A.D.Lopez and H.S.Law: "A dense gate matrix layout method for MOS VLSI", IEEE Trans. Electron Devices, vol. ED-27, pp. 1671-1675 (1980).
- (6) M.Y.Hsueh and D.O.Pederson: "Computer-aided layout of LSI circuit building-blocks", Proc. 1979 ISCAS, pp. 474-477(1979).
- (7) A.E.Dunlop: "SLIM-The translation of symbolic layouts into mask

- data", Proc. 17th DA Conf., pp. 595-602(1980).
- (8) Y.E.Cho: "A subjective review of compaction.", Proc. 22nd DA Conf., pp. 396-404(1985).
- (9) T.Umeki, Y.Saito, H.Nakagawa and T.Enomoto: "A Prototype micro-controller design by symbolic layout approach", Proc. 1985 ICCD, pp. 63-67(1985).
- (10) K.Sato, T.Nagai, M.Tachibana, H.Shimoyama, M.Ozaki and T.Yahara: "MILD-A cell-based layout system for MOS-LSI." Proc. 18th DA Conf., pp. 828-836(1981).
- (11) M.Terai, Y.Ajioka, T.Noda, M.Ozaki, T.Umeki and K.Sato: "Symbolic layout system: Application results and functional improvements", IEEE Trans. Comput.-Aided Des. Integrated Circuits & Syst., pp. 346-354(1987).

## 第9章 結 論

本論文はVLSIのレイアウト設計の内、特に、種々の相異なるチップ構造を持つゲートアレイ向き配線手法、高い配線率を達成するためのゲートアレイの自動配線手法、トランジスタアレイにおけるレイアウト手法及びカスタムVLSIやマイクロプロセッサのマクロセルのレイアウト設計のためのシンボリックレイアウト手法に関する研究成果をまとめたものである。本章では、本研究を通じて得られた成果を要約すると共に今後の課題や問題点を指摘する。

本論文により得た成果は次の通りである。

(1) CMOSゲートアレイやECLゲートアレイ等の様々のチップ構造を持つゲートアレイ向きの配線手法、及びそれに適したデータ構造を考案した(第2章)。

(2) 相異なるチップ構造を持つゲートアレイ向きの配線手法を2,600ゲートCMOSゲートアレイと2,500ゲートのECLゲートアレイに適用し、100%の配線率を達成した。更に、この配線手法と特定のチップ構造を持つゲートアレイ専用の配線手法を624ゲートのDSA MOSゲートアレイに適用した結果において、2つの配線手法の計算機上で使用する記憶領域の大きさ、配線率及び処理時間はほぼ同等であった。これらの結果から本配線手法が有効であることを示した(第2章)。

(3) ゲートアレイの配線設計におけるグローバル配線に対し、各信号ネットの迂回径路による配線長の増大に関して定義される評価関数 $\Delta L(W_i)$ の値が大きい信号ネットから順に、配線の混雑度を均一化するように、グローバル配線を行う手法を考案した(第3章)。

(4) このグローバル配線手法を900ゲートのECLゲートアレイに適用した結果から、上記の手法が配線の迂回を少なくし、配線の局所的混雑を緩和して配線不能を少なくする点で優れていることを示した(第3章)。

(5) ゲートアレイの配線設計における端子割当に対し、まず初期端子割当を行い、次に幹線間のサイクリックな上下制約の数を正確に数えながら端子割当を改善することにより、最終的な端子割当を決定する算法を考案した(第4章)。

(6) この端子割当算法の高速化のために、端子割当変更によりサイクリックな上下制約の数が減少するか否かの判定が一部の幹線間の上下制約に着目するだけで可能であることを理論的に証明した(第4章)。

(7) この端子割当算法を900ゲートのECLゲートアレイに適用した結果、

すべての幹線間のサイクリックな上下制約を短い処理時間で除去できた。この適用結果から、本算法の有効性を示した（第4章）。

(8) ゲートアレイの詳細配線設計における幹線分割に対し、幹線間の上下制約の連鎖の長さが増加せず、かつ水平チャンネル内の配線を行うための水平トラック数に関して定義される評価関数 $W(T)$ の値が減少するように、幹線分割を行う算法を考案した（第5章）。

(9) この幹線分割算法を高速化するために、幹線分割により幹線間の上下制約の連鎖の長さが増加しないための必要十分条件を示し、更に関数 $W(T)$ の値の計算方法について理論的考察を行った（第5章）。

(10) 本幹線分割算法を8,000ゲートのCMOSゲートアレイの水平チャンネル内の詳細配線設計に適用した結果、実用的な処理時間でチャンネル内の配線に使われる水平トラック数を減らすことができた。この適用結果から本算法が優れていることを示した（第5章）。

(11) ゲートアレイのセル配置が定まった段階において、チップ全体の配線に必要な水平トラック数を予測する問題について考察し、各信号ネットの配線の幹線分割回数に制限を加えた時、 $N$ セル列間の配線に必要なかつ十分な水平トラック数がある範囲内にあることを理論的に証明した（第6章）。

(12) 624ゲートのDSA MOSゲートアレイの自動配線において、使用される水平トラック数がほぼ上記の範囲の上限に等しいことを示した（第6章）。

(13) RAM/ROMを含む回路をトランジスタアレイ上に高密度にレイアウトするために、まずセルの配置を行い、次に配線混雑度を予測してセル列間の水平チャンネルの容量の調整を行った後に、配線を行うトランジスタアレイ向きのレイアウト手法を考案した（第7章）。

(14) このトランジスタアレイ向きレイアウト手法を20,000ゲートのCMOSトランジスタアレイに適用した結果から、この手法がRAM/ROMを含むVLSIに対して特に有効であることを示した（第7章）。

(15) フルカスタムVLSIやマイクロプロセッサ等のマクロセル（機能ブロック）のレイアウト設計のためのシンボリックレイアウト手法に対し、レイアウト設計を短期間にかつ高密度に行うことができ、更にレイアウト結果が製造プロセスの進歩に伴うパターン設計ルールの変更に充分追従できるようにするためのシンボルの定義方法、シンボル配置手法、及びコンパクション手法を考案した（第8章）。

(16) このシンボリックレイアウト手法を16ビットマイクロコントローラのレ

レイアウト設計に適用した結果、人手レイアウト設計の場合の約1/3.5の設計期間で、人手レイアウト設計の場合の約70%の集積度を達成できた(第8章)。

本研究において得られた成果に基づき、VLSI用の高性能の自動レイアウトシステムを構築した。この結果、種々の相異なるチップ構造を持つゲートアレイの配線設計に対し高い配線率を達成することが可能となり、トランジスタアレイに関しては高密度なレイアウトを自動生成することが可能となった。更に、フルカスタムVLSIやマイクロプロセッサのマクロセルのレイアウト設計をシンボリックレイアウト手法により短期間に、高密度に、かつパターン設計ルールと独立に行うことが可能となった。

しかしながら、VLSIの微細加工技術やデバイス製造技術は現在も急速に進展しつつあり、このため、新しいデバイス製造技術を用いた新規なセル構造のVLSI向けのレイアウト手法の研究や、更に集積度の増大に伴い必要となるVLSIのレイアウト結果の対話型修正の手法の研究が今後の課題である。

## 謝 辞

本論文を結ぶにあたり、終始懇切なる御指導と御鞭撻を賜った大阪大学工学部電子工学教室白川功教授に衷心より感謝の意を表します。

更に、本研究をまとめる過程で種々の御指導と御鞭撻を頂いた大阪大学工学部電子工学教室児玉慎三教授、寺田浩詔教授、浜口智尋教授に深く感謝致します。

本研究の機会を与えられ、研究遂行にあたって終始御指導、御鞭撻を頂いた三菱電機株式会社専務取締役岡久雄博士、三菱電機株式会社カスタムLSI設計技術開発センター長蒲生容仁博士、三菱電機株式会社LSI研究所長柴山恭一博士、三菱電機セミコンダクタソフトウェア株式会社北伊丹事業所長八原俊彦博士、三菱電機株式会社カスタムLSI設計技術開発センターCAD技術開発第二部第1グループマネージャー佐藤興二博士に厚く御礼申し上げます。

本研究の遂行にあたり御指導と御教示を頂いた三菱電機中央研究所部長田中千代治博士、同カスタムLSI設計技術開発センター部長村井真一氏、グループマネージャー岡崎芳博士、同伊丹製作所主幹金田一博士および御協力頂いた方々に心から感謝致します。

## 研究業績目録

### I. 学会論文誌発表論文

- (1) 寺井, 金田, 八原: " マスタスライス方式レイアウトにおける横トラック数の考察", 信学論(D), vol. J64-D, no. 8, pp. 697-704(1981).
- (2) 寺井, 野田, 佐藤, 八原: " 種々のチップ構造のゲートアレイに適用可能な配線プログラム", 情報処理学会論文誌, vol. 25, no. 3, pp. 357-364 (1984).
- (3) M.Terai: "A method of improving the terminal assignment in the channel routing for gate arrays", IEEE Trans. Comput.-Aided Des. Integrated Circuits & Syst., vol. CAD-4, no. 3, pp. 329-336(1985).
- (4) 寺井, 佐藤: " チャネル配線法における幹線分割の一手法", 信学論(A), vol. J68-A, no. 11, pp. 1184-1192(1985).
- (5) M.Terai, Y.Ajioka, T.Noda, M.Ozaki, T.Umeki and K.Sato: "Symbolic layout system: Application results and functional improvements", IEEE Trans. Comput.- Aided Des. Integrated Circuits & Syst., pp. 346-354(1987).
- (6) Y.Kuramitsu, M.Ueda, T.Arakawa, M.Terai and S.Asai: "A 540K-Transistor CMOS variable-track masterslice", IEEE J. Solid-State Circuits, vol. SC-22, no. 2, pp. 198-201(1987).

### II. 研究会等発表論文 (査読付)

- (1) M.Terai, H.Kanada, K.Sato and T.Yahara: "A consideration of the number of horizontal grids used in the routing of a masterslice layout", Proc. 19th DA Conf., pp. 121-128(1982).
- (2) C.Tanaka, S.Murai, H.Tsuji, T.Yahara, K.Okazaki, M.Terai, R.Katoh and M.Tachibana: "An integrated computer aided design system for gate array masterslices: Part 2 the layout design system MARS-M3", Proc. 18th DA Conf., pp. 812-819(1981).
- (3) K.Sato, M.Terai, T.Noda, and M.Ozaki: "A method of connectivity checking for MOS circuits considering logical equivalence", Proc. 1985 ISCAS, pp. 1285-1288(1985).

### III. その他研究会等発表論文

- (1) 寺井, 村井, 田中, 金田, 藤原, 樹下: " マスタスライス方式LSIにおけるチャンネル割当の一手法", 情報処理学会電子装置設計技術研究会資料, 1-2(1979).
- (2) 寺井, 金田, 河本, 梶谷: " マスタスライス方式レイアウトにおける横トラック数について", 信学技報, CAS80-53(1980).
- (3) 寺井, 下山, 岡崎, 柿沼, 辻, 村井: " マスタスライスLSIの配置配線プログラム", 電気学会電子デバイス・システム・制御合同研究会, EDD-81-9(1981).
- (4) 寺井, 野田, 佐藤, 八原: " 種々のチップ構造のマスタスライスLSIに適用可能な配線プログラム", 昭57信学総全大, 361(1982).
- (5) 野田, 寺井, 佐藤, 八原: " 多様なデバイスのゲートアレイに適用可能な配線プログラム", 信学技報, CAS82-79(1982).
- (6) 寺井, 野田, 佐藤, 八原: " 種々のチップ構造のゲートアレイに適用可能な配線プログラム", 昭和58年度電子通信学会情報・システム部門全国大会, S1-5(1983).
- (7) 寺井, 佐藤: " ゲートアレイの配線における端子割当改善の一手法", 昭58信学総全大, 384(1983).
- (8) 寺井, 佐藤: " ゲートアレイのチャンネル配線における端子割当改善の一手法", 信学技報, CAS83-208(1984).
- (9) 寺井, 佐藤: " チャンネル配線法における幹線分割の一手法", 昭60信学総全大, 416(1985).
- (10) 寺井, 味岡, 野田, 尾崎, 佐藤: " シンボリックレイアウトシステム一適用結果と機能改善一", 信学技報, CAS85-160(1986).
- (11) 藤野, 寺井, 野田, 味岡, 佐藤: " ゲートアレイ配置配線プログラムのトランジスタ敷き詰め式ゲートアレイへの適用", 昭61信学総全大, 394(1986).
- (12) 野田, 藤野, 寺井, 味岡, 佐藤: " トランジスタ敷き詰め式ゲートアレイのレイアウトシステム", 情報処理学会設計自動化研究会報告, 34-7(1986).