

VLSIブロック自動配置手法

三浦地平 茂垣真人 小澤時典
(日立製作所 中央研究所)

1. 緒言

L S I の集積度は年々増加し、設計に多くの人工数を要している。そのため 10,000ゲートを越える規模のVLSIのレイアウト設計では、設計の複雑さを避けるために、階層的アプローチ〔1, 2〕が採られている。このレイアウト階層は、通常、チップ・ブロック・セルという3つのレベルからなる。今まで、ブロック内配置配線及びブロック間配線を自動的に行うレイアウトシステムは幾つか開発され、報告されている〔1~3〕。しかし、これらのシステムではブロックの配置は人手に委ねられていた。

本論文ではブロック配置の自動化を目的としたブロック相対配置アルゴリズムCTOP (Combine and TOP down placement algorithm), ブロック間配線領域推定アルゴリズムWSE (Wiring Space Estimation algorithm) について述べる。さらに、これらのアルゴリズムを約19,000ゲートカスタム論理VLSI 2品種に適用した結果について述べる。

2. レイアウトモデル

本論文で対象とするカスタム論理VLSIのレイアウト階層構造を図1に示す。チップはブロックとI/Oバッファ列で構成される。各ブロックはセルの集合、又は、ROM, RAM等のマクロセルであり、通常、規模は500~1,000ゲートである。セル間を接続する配線には3層、すなわち、ポリシリコン層(PS), 金属2層(AL1, AL2)を用いている。ブロック端子は金属層のみを用いて接続される。

ブロック配置モデルを図2に示す。ブロック配置では詳細配線は行わないため、ここでのモデルは図1と比較してマクロなものとなっている。モデルの特徴を以下に示す。

- (1) チップは矩形である。
- (2) I/O端子はチップの4辺に配置される。
- (3) ブロックは矩形である。
- (4) ブロック間に生じる隙間を無効領域と呼ぶ。
- (5) 配線領域を垂直配線チャンネルと水平配線チャンネルに分割する。ここでチャンネルの形状は矩形でなくともよい。
- (6) ブロック端子はブロック領域内部に散在する。
- (7) ブロック内配線で使用されなかった領域を通過するブロック間AL2配線の量を推定するために、ブロック内に垂直フィードスルーを設定する。

3. ブロック配置評価基準

ブロック配置において中心となる評価基準は(A)チップ面積最小化である。ブロック領域はブロックの配置によらず一定と考え、(A)は(B)ブロック間領域面積最小化と同値である。このブロック間領域は図2に示したように配線領域と無効領域の2つに分けて考えることができる。ブロックの寸法が揃って

いる場合には、標準セルの配置手法〔3〕と同様に主に配線領域面積最小化を評価基準とすればよいが、大小のブロックが混在するときには、ブロックの寸法を考慮して配置を行わないと、配線領域は小さくなくても無効領域は大きくなる場合がある。この点を考慮して、ブロックの配置評価基準を（a）配線領域面積最小化、（b）無効領域面積最小化の2つとする。

4．ブロック自動配置処理の流れ

ブロック自動配置処理の流れを図3に示す。処理の入力は、各ブロックの縦、横の寸法とブロック間論理結線関係であり、出力は各ブロックのチップ上の絶対位置である。ここでブロックの絶対位置が必要な理由は次の点からである。我々の階層型自動レイアウトシステムにおけるブロック間配線手法は、大小の任意矩形ブロックが混在し、ブロック端子が内部に散在するレイアウトモデル（図1）に対処するため、迷路法を採用している〔1〕。そのため、あらかじめ配線用にブロック間をあけておく必要がある。処理は以下の2つの手順からなる。

STEP 1：ブロック相対配置

ここでは、各ブロックについて、その上下左右に隣接するブロックを決定する。相対配置の評価基準は3章で示した（a）配線領域面積最小化、（b）無効領域面積最小化である。これらの基準は相対配置の段階では直接評価できないので、図3に示すように、配線領域面積はブロック端子間の仮想配線長で近似し、ブロック間隙間面積を無効領域面積とし、評価する。

STEP 2：配線領域推定

ブロック間の論理結線とSTEP 1の配置結果に基づき、ブロック間に配線チャンネルを生成し、接続すべきブロック端子間のマクロ配線経路（配線が通過するチャンネル）を決定し、各配線チャンネル幅を推定する。次にブロック間をチャンネル幅だけ広げることによりブロックの絶対配置を決定する。ここでの評価基準は（c）推定誤差最小である。すなわち、後の自動配線結果に未配線が少なく、無効領域が少ないほど質の良い絶対配置ということになる。

5．ブロック自動配置アルゴリズム

本章ではブロック相対配置アルゴリズムCTOP、配線領域推定アルゴリズムWSEについて述べる。CTOPはコンバイン処理とトップダウン配置処理からなる。ここでコンバインとは、4章で示した相対配置の評価基準、仮想配線長最小、無効領域面積最小を考慮するために新たに定義した操作である。まずコンバインの概念について説明する。

5.1 コンバインの概念

図4に基づき、コンバインの概念を以下に説明する。

定義1．超ブロック

ブロック、又は縦、横寸法をもつブロックの集合を超ブロックと定義する。

定義2．コンバイン

2つの超ブロックI、Jを縦、又は横に接合させ、それらの外接矩形を求めることにより、新たに超ブロックKを構成できる。この操作をコンバインと定義し、" $K = I \odot J$ "で表わす。ここでDはコンバイン方向を示し、横方向にコンバインした場合はX、縦方向の場合はYである。

定義3．コンバイン度

“超ブロック” $I \odot J$ が与えられたとき， I ， J ， D に関するコンバイン度 $P(I \odot J)$ を次式で定義する。

$$P(I \odot J) = S(I \odot J) + C(I, J) \cdot S(I \odot J) \cdot a$$

$$S(I \odot J) = \frac{A(I) + A(J)}{A(I \odot J)} \quad (\leq 1)$$

$$C(I, J) = \frac{2 \cdot T_c(I, J)}{2 \cdot T_c(I, J) + T_d(I, J)} \quad (\leq 1)$$

$$T_c(I, J) = \sum_{n1 \leftarrow N1} \frac{1}{|n1| \#}$$

$$T_d(I, J) = \sum_{n2 \leftarrow N2} \frac{1}{|n2| \#}$$

- I, J : 超ブロック
- D : コンバイン方向
- S : スペース効率
- C : 結合度
- T_c : I, J の合接
- T_d : I, J の離接
- $A(I)$: I の面積
- $N1$: I と J に共通に接続するネットの集合
- $N2$: I と J のどちらか一方に接続するネットの集合
- $|n| \#$: ネット n に接続するブロック端子数
- a : 調整パラメータ

上式において， $S(I \odot J)$ はスペース効率を表わし， I と J の面積の和を $I \odot J$ の面積で割った値である。 $S(I \odot J)$ が大きい場合には， I と J を隣接して配置することは無効領域面積を小さくすることにつながる。

一方， $C(I, J)$ は結合度を表わし， I と J の合接を I と J の離接で割り，それを正規化した値である。上記の合接，離接の定義式では，多点ネットの場合を考慮している。 $C(I, J)$ が大きい場合には， I と J を隣接して配置するこ

とは仮想配線長を小さくすることにつながる。しかし、ここで、無効領域が仮想配線長に悪影響を及ぼすことを考慮すべきである。次の条件が成り立つ場合には、一般的に、 I_1 と J_1 間の仮想配線長は I_2 と J_2 間の仮想配線長より長いと考えられる。

- 〔条件〕 $\bigcirc A(I_1) = A(I_2)$, $A(J_1) = A(J_2)$
 $\bigcirc S(I_1 \odot J_1) < S(I_2 \odot J_2)$
 $\bigcirc C(I_1, J_1) = C(I_2, J_2)$

従って、 $C(I, J)$ と $S(I \odot J)$ が大きくなる I と J を隣接して配置すると、仮想配線長を小さくできることになる。

以上の点から、コンバイン度 $P(I \odot J)$ は無効領域面積と仮想配線長を考慮した式となっており、 $P(I \odot J)$ が大きくなる I と J を隣接して配置すると、無効領域面積と仮想配線長を小さくできると考える。ここで a は調整パラメータであり、最適値は実験的に求める。

5.2 CTOPアルゴリズム

図5にCTOPの処理の流れを示す。CTOPは4つのステップ、初期コンバイン、4単位コンバイン、同方向コンバイン、トップダウン配置からなる。

STEP 1: 初期コンバイン (図6参照)

このステップでは、共通ネットをもつ超ブロック I, J についてコンバイン度 $P(I \odot J)$ を計算し、この値が最大となる I, J, D を求めて、超ブロック $I \odot J$ を作成する処理を、全体が1つの超ブロックになるまで繰り返す。この処理を制限なしで行うと、処理の最終段階では X, Y 各方向のみに成長した巨大な超ブロックができ、大きな無効領域を残す原因となる。そのため、超ブロックの寸法に一時的に上限値を設けて、段階的にコンバインを行う。

コンバイン処理の過程はラベル付き2進木で表される。ここで、木の節点は超ブロックに、枝は超ブロック間の親子関係に対応する。

STEP 2: 4単位コンバイン (図7参照)

このステップでは、STEP 1で構成されたコンバイン木を変更することにより無効領域を削減する。大幅な変更は仮想配線長を増加させるため、局所的な変更を行う。コンバイン木の根から葉に向かって以下の処理を繰り返す。各超ブロック K について、祖先が K となり、かつ、お互いに親子関係のない4つの超ブロックを選択する。次に、それらについて、すべてのコンバインを求め、縦、横寸法が初期の K のものよりも小さく、かつ、無効領域が最小なコンバインを選択する。ここで、4つの超ブロックをコンバインする仕方は52通りある。

STEP 3: 同方向コンバイン (図8参照)

このステップでは、コンバインの方向は変更せずに、すなわち、超ブロックの寸法は変更せずに、コンバイン木を変更することにより、仮想配線長を短縮する。以下の処理をボトムアップに行う。STEP 2で再構成されたコンバイン木において、すべての節点が同じラベルをもつ部分木を求める。次に、STEP 1と同様にして、その葉を結合度 $C(I, J)$ のみに従い再コンバインする。

STEP 4 : トップダウン配置 (図9参照)

STEP 3で再構成されたコンバイン木の根から葉に向かって以下の処理を繰り返す。子をもつ超ブロックについて、それらの子を、仮想配線長が短くなるように、コンバイン方向に従って相対配置する。

5.3 WSEアルゴリズム

図10にWSEの処理の流れを示す。WSEは3つのステップ、チャンネル作成、チャンネル探索、チャンネル拡張からなる。

STEP 1 : チャンネル作成

隣接したブロック間に幅0の配線チャンネルを作成する。又、ブロック内フィードスルーを通過配線本数の制限をもつチャンネルとみなし、追加する。

STEP 2 : チャンネル探索

各ネットについて、接続すべきブロック端子間のマクロ配線径路（通過する配線チャンネル）を決定し、マクロ配線で使用したチャンネルの範囲を混雑度分布表に記入する。ここで混雑度分布表とは、各チャンネル上の点について、そこを通過する配線本数を示した表である。処理手順を図11に基づき、以下に説明する。

各ネットNについて次の処理を行う。

(1) 目標点の設定

Nにつながるブロック端子の重心点を求め、この点に最も近いチャンネルを求め、目標チャンネルとする。又、重心点に最も近い目標チャンネル上の点を目標点とする。

(2) ブロック端子からの探索

Nにつながる各ブロック端子から出発し、チャンネル間の隣接関係を探索し、目標チャンネルに到達する径路を決定する。ここで、注目チャンネルに隣接する複数の候補チャンネルの中で次のチャンネルを選択する時の評価関数は目標点までの距離であり、これが最小となるチャンネルを次の注目チャンネルとして選択する。ここでの距離はフィードスルー長を加算しない、又、目標点までの径路が迂回を必要とする場合、迂回長を加算することを特徴とする。1つのブロック端子からの探索が終了すると、その探索で使用されたチャンネルは、次の探索のために、新たに目標チャンネルとして追加される。

STEP 3 : チャンネル拡張

混雑度分布表に合わせて、ブロック間を広げ、ブロックの絶対座標を決定する。まず、垂直配線チャンネルとブロックの隣接関係、垂直配線チャンネルの混雑度分布に基づき、ブロックのX座標及びチップX寸法を決定し、次にY方向について同様の処理を行う。

6. 実験結果

本章では、CTOP、WSEアルゴリズムを約19,000ゲート規模のカスタム論理VLSI 2品種に適用した結果を述べる。表1に実験結果を示す。正規化面積〔3〕は、2品種とも、ブロック配置を手で行った場合と同程度であった。又、CTOPとWSEを合わせた処理時間(M200H・CPU)は約20秒であった。チップC1について、CTOPの結果を図12に、WSEの結果を図13に、自動配置配線結果を図14に示す。

以上の実験は、すべて、同じブロック内配置配線プログラムを用いて行った。

7. 結言

(1) VLSIの階層化レイアウト設計におけるブロック配置の自動化を目的として、ブロック相対配置アルゴリズムCTOP，配線領域推定アルゴリズムWSEを開発した。CTOPでは、ブロック間のスペース効率と結合度を組み合わせたコンパイン度を用いて配置を行う。

(2) CTOP，WSEアルゴリズムを約19,000ゲート規模のカスタム論理VLSI 2品種に適用した。チップ面積は、2品種とも、ブロック配置を手で行った場合と同程度であった。又、CTOPとWSEを合わせた処理時間(M200H・CPU)は約20秒であった。

参考文献

- [1] H. Terai, et al.; "Automatic Placement and Routing program for Logic LSI Design", Proc. of Conf. ICCG, September 1982.
- [2] T. Adachi, et al.; "Hierarchical Top-down Layout Method for VLSI chip", Proc. of 19th DA Conf., June 1982.
- [3] T. Kozawa, et al.; "Automatic Placement Algorithms for High Packing Density VLSI", Proc. of 20th DA Conf., June 1983.

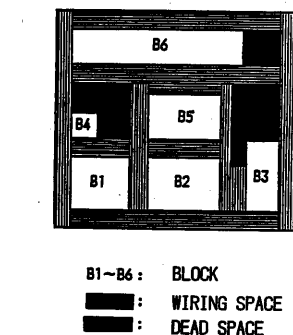
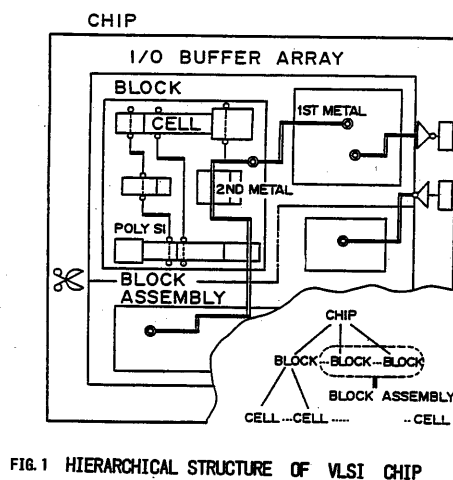


FIG. 2 BLOCK PLACEMENT MODEL

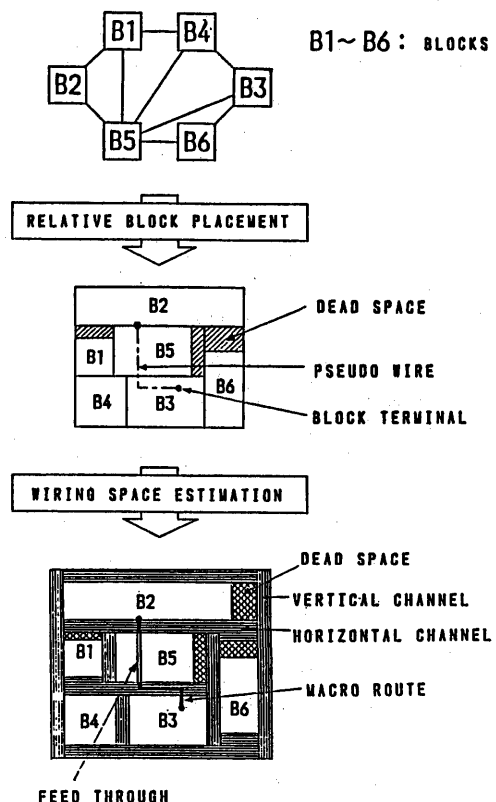


FIG. 3 AUTOMATIC BLOCK PLACEMENT FLOW

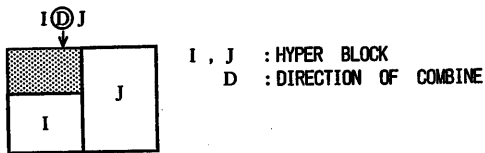


FIG. 4 DEFINITION OF COMBINE VALUE

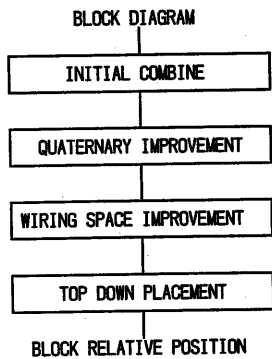


FIG. 5 RELATIVE BLOCK PLACEMENT FLOW

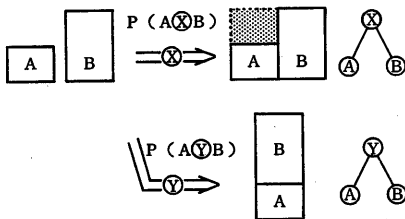


FIG. 6 INITIAL COMBINE PROCESS

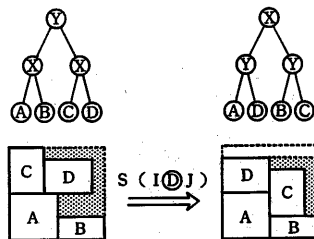


FIG. 7 QUATERNARY IMPROVEMENT PROCESS

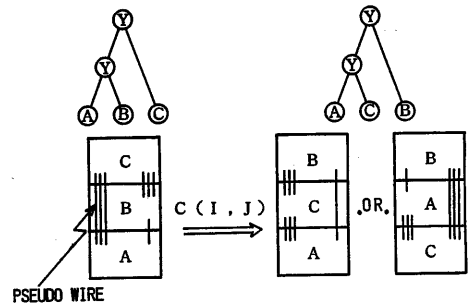


FIG. 8 WIRING SPACE IMPROVEMENT PROCESS

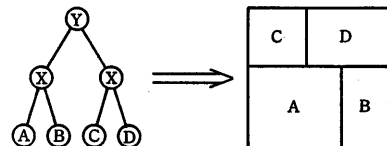


FIG. 9 TOP DOWN PLACEMENT PROCESS

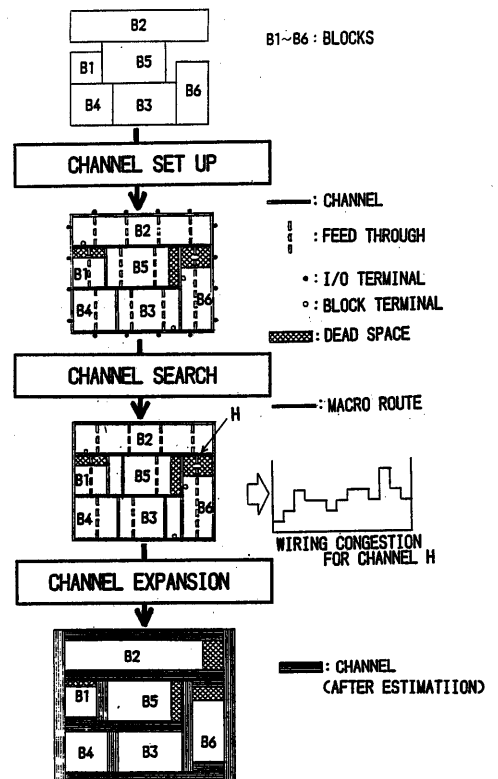


FIG. 10 WIRING SPACE ESTIMATION FLOW

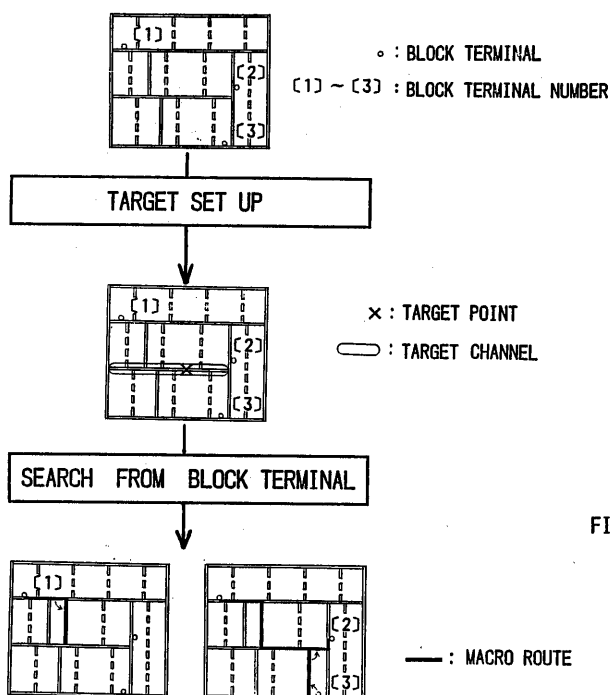


FIG. 11 CHANNEL SEARCH PROCESS

TABLE. 1 EXPERIMENTAL RESULT

ITEM		PLACEMENT METHOD	
		MANUAL	AUTO
CHIP AREA (NA) *1	C1	2 8	2 8
	C2	3 3	3 1
EXECUTION TIME *2	C1	—	1 9
	C2	—	2 2

$$*1): NA = \frac{(\text{CWIRING PITCH})^2}{\text{NUMBER OF TRANSISTOR}}$$

*2): M200H - CPU - SEC

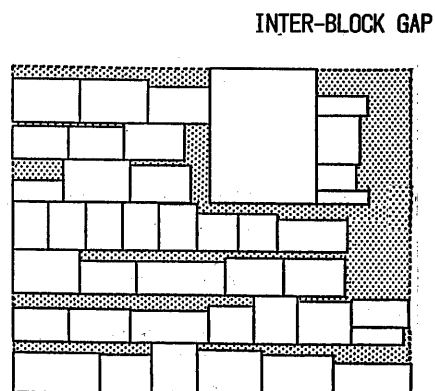


FIG. 12 EXAMPLE OF RELATIVE BLOCK PLACEMENT

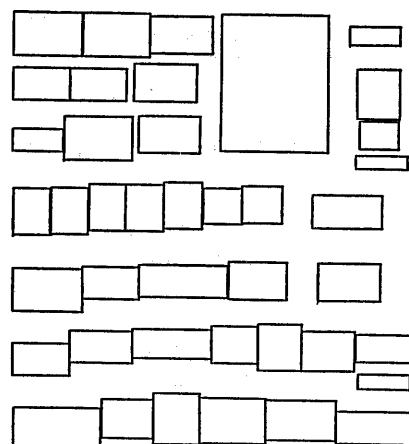


FIG. 13 AFTER WIRING SPACE ESTIMATION

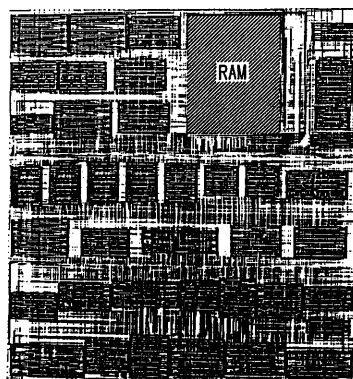


FIG. 14 DRAWING PATTERN AFTER AUTOMATIC PLACEMENT AND ROUTING