

設計者の協調的介入を受理するディレイ最適化自動設計システムと 高速RISCチップ設計への適用

寺井秀一 (*1) 源馬和寿 (*2) 長尾葉介 (*3) 佐藤康夫 (*4)

(*1) terai@bkc.ritsumei.ac.jp 立命館大学理工学部
(*2) kgemma@kanagawa.hitachi.co.jp 日立製作所汎用コンピュータ事業部
(*3) ynagao@kanagawa.hitachi.co.jp 日立製作所汎用コンピュータ事業部
(*4) ysatoh@ddc.hitachi.co.jp 日立製作所デバイス開発センタ

コンピュータの核となるプロセッサチップそのものは、性能の向上と、機能の充実を求めてたえず改良が行われている。これらのチップの設計にはデバイスの性能を極限まで引き出すことによって高性能化を実現することが要求され、場合によってはデバイス技術の開発と並行してチップ開発が進むこともある。以下では、120MHz、280万トランジスタを集積した高速、大規模RISCプロセッサ (HARP-1; Hitachi Advanced RISC Processors 1) の設計に際して開発したDAシステムについて、開発期間の短縮を狙いとす、論理とディレイの並行検証、インクリメンタルレイアウト方式、および、きめ細かいディレイ制御を可能とするために、ブロックの位置・形状、配線順序、配線層、配線幅等の項目をアクセプトし設計者の意図を反映したレイアウトを行うタイミングドリブン配置配線のアルゴリズムについて述べる。

Basic Concept of Cooperative Timing-driven Design Automation Technology for High-speed RISC Processor Design

Hidekazu Terai (*1), Kazutoshi Gemma (*2), Yohsuke Nagao (*3), Yasuo Satoh (*4)

(*1) terai@bkc.ritsumei.ac.jp : Ritsumeikan University
(*2) kgemma@kanagawa.hitachi.co.jp : General Purpose Computer Division, Hitachi Ltd.
(*3) ynagao@kanagawa.hitachi.co.jp : General Purpose Computer Division, Hitachi Ltd.
(*4) ysatoh@ddc.hitachi.co.jp : Device Development Center, Hitachi Ltd.

This paper outlines the concepts and the key technologies of design automation for developing high-performance microprocessor chips. The goals of the design automation system are twofold: (1) reducing the development period, (2) implementing high performance. Problems in design automation include concurrent verification of both logic and timing design, incremental physical design, and a cooperative timing-driven placement and a routing procedure which accept various designer's specifications. Solutions to these problems are presented.

1. はじめに

ASIC (Application Specific Integrated Circuit) の設計に関しては、近年、ワークステーションを利用した様々な設計支援システム (以下ではDA ; Design Automationシステムの用語を使用する) が商用ツールとして提供され、これらを利用することで、手軽に論理LSIを手中にすることが可能となってきている。

一方、コンピュータの核となるプロセッサチップそのものは、性能の向上と、機能の充実を求めてたえず改良が行われている [1,2,3,4]。この種のLSIは、高性能化の追求という観点から高速・大規模な論理をいかにチップへ集積するかが主要な課題となる。これらのチップの設計にはデバイスの性能を極限まで引き出すことによって高性能化を実現することが要求され、場合によってはデバイス技術の開発と並行してチップ開発が進むこともある。このため、チップの設計に当たっては、既存のDAツールのみを用いるだけでなく、対象となるチップの性能目標に対応してDAシステム自体のチューニングとカスタマイズが必要になってくる。以下では、120MHz、280万トランジスタを集積した高速、大規模RISCプロセッサ (HARP-1 ; Hitachi Advanced RISC Processors 1) [4] の設計に際して開発したDAシステムについて、開発目標と設計手法、設計者の意図を反映するタイミングドリブン配置配線のアルゴリズムを中心に述べる。

2. チップの概要

HARP-1はPA-RISC Architecture (Hewlett Packard Precision Architecture) を採用した高性能RISCプロセッサである。設計ルールは $0.5\mu\text{m}$ BiCMOS技術を用い、クロック周波数120MHzで動作する。プロセッサは、命令キャッシュ8Kバイト、データキャッシュ16Kバイト、浮動小数点演算器、2命令スーパースカラ制御を実行する論理等を内蔵している。トータルトランジスタ数は約2.8M、配線は4層 (タンゲステン1層、アルミニウム3層) でチップサイズは16.2x16.5mmである。

図1にチップの基本レイアウトを示す。図に示すとおり、チップは演算器等の規則論理部、RAM、I/Oバッファ、制御論理部の4種類の部分からなる。以下ではこれらの部分を「リージョン」と呼ぶ。リージョンはチップレベルでのフロアプランの単位でもある。各リージョンのうち、制御論理部はスタンダードセルからなるブロックの集合であり、残り3種類はマクロセルで実現している。

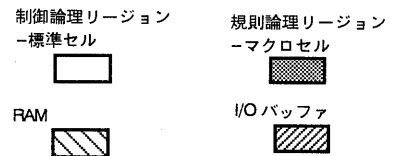
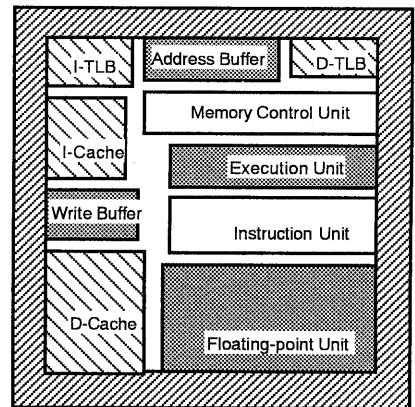


図1 チップの概要

3. DAシステムの目標

高性能チップを実現するには信号伝播ディレイ値の短縮が重要な課題となる。また、開発期間を短縮するためにはチップの作り直し回数を最小とする高品質の論理設計が必須である。本DAシステムでは次の

2点を重点開発目標とした。

目標1；開発期間を短縮するための設計管理手法

目標2；チップの高速動作を確保するためのディレイ制御設計方式

目標1を実現するためには、論理設計品質を高めることと、きめ細かな設計管理技術がキーポイントとなる。前者については、チップの各部分の論理設計方式を吟味し、以下の方策を採った。また、後者では「コンカレントデザイン」と「インクリメンタルデザイン」手法が効果的であると考え、DAシステムにインプリメントした。

(1) 論理設計方式

論理不良の多くが、命令制御部とメモリ制御部およびその関連の制御論理にあるとのこれまでの経験から、この部分には、機能レベル記述手法と自動論理合成システムを適用し、スタンダードセルベースの論理設計を行うことにより設計品質の確保と論理検証効率の向上を図った。マクロセル論理部はMOSトランスファ回路やダイナミック回路を多用するためゲート/スイッチレベルのシミュレータを用いて検証を行い、さらに、マクロセルリージョンのトランジスタレベル記述をゲートレベル記述に自動変換することにより、制御論理部と、関連するマクロセル論理部とをゲートレベルで高速に検証する環境を構築した。これにより、スタンダードセルリージョンとマクロセルリージョンとの混在状態でゲートレベル論理シミュレーションを実施することが容易となり、任意の論理ユニットを組み合わせて、スモールスタートで効率良い論理シミュレーションを実施することが可能となった。ユニット毎のトランジスタ回路レベルの動作シミュレーションに加えて、最終的にはチップ全体をゲートレベルで動作させる、徹底した論理シミュレーションを行った。これには、スーパーコンピュータをベースとした高速論理シミュレーションエンジンを用い、乱数TMPによる実チップ動作を確認することで、論理不良のほとんどをチップ製造前に摘出することができた。

(2) 論理・ディレイ検証の並行化

図2に開発したDAシステムによる設計フローを示す。ここでの特徴は、論理検証とディレイ検証のコンカレント処理を可能としていることである。

ディレイ検証サイクルの初期の段階では仮想配線長を用いたディレイ計算によりパスディレイを求める。論理設計が進むに従ってディレイ計算の精度を上げるため、自動配置配線を行い、各素子の配置位置と配線経路を決定した後、すべての経路の信号伝播遅延時間を高速に計算し、規定値以内か否かを確認する。ディレイ不良には通常、論理設計に起因するものと、レイアウト設計に起因するものがある。従って、論理を修正すべきかレイアウトを修正すべきかの切り分けが必要となる。設計の初期フェーズではインクリメンタル論理生成によ

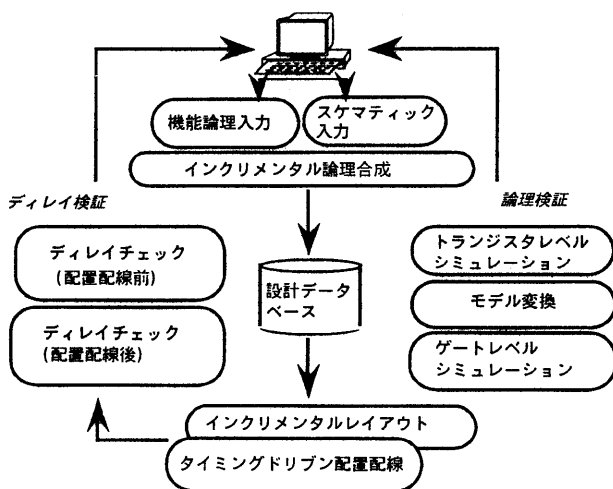


図2 DAシステムと設計フロー

り、論理ファイルの変更で対処するが、論理検証が進んだ段階では設計の手戻りを少なくするために、ディレイ考慮自動レイアウト技術がもつインクリメンタル配置配線機能によって実装の部分変更を行なう。以下に、コンカレント検証における設計管理の考え方を示す。

①ディレイ検証の精度を上げるには、配置配線の実装情報が必要である。このためレイアウトが終了した後にディレイ検証を行うことは必須である。②一方、チップ全体の配置配線には数日オーダーの期間が必要である。③このため、論理検証、ディレイ検証およびレイアウトの3つのチームを構成し、それらが設計データベースをパイプライン的にアクセスすることにより全体として検証期間の短縮を図る。

これを円滑に行なうには設計データベースに対するバージョン管理が重要である。図3を用いてコンカレント検証の流れを説明する。まず、同じバージョンのファイルに（第Nバージョン）に対して論理検証とレイアウトを並行させる。【このとき、ディレイ検証チームは、その時点で論理検証チームがアクセスしているバージョンの1つ前のバージョン（第N-1バージョン）のファイルを用いてディレイ検証を行っている。】第Nバージョンのレイアウトが終了した後、引き続きディレイ検証フェーズ（レイアウト情報が付加されている）にはいる。一方、論理検証は第N+1バージョンファイルに対して継続する。この過程で発生したECN（要設計変更項目）すなわち、第N+1バージョンの論理検証ECNおよび、第Nバージョンのディレイ検証ECNは設計管理者が吟味しそれらが矛盾なく収束するようにチェックした後、第N+2バージョンのファイルとしてデータベースの更新を行う。以上の作業を逐次繰り返す。

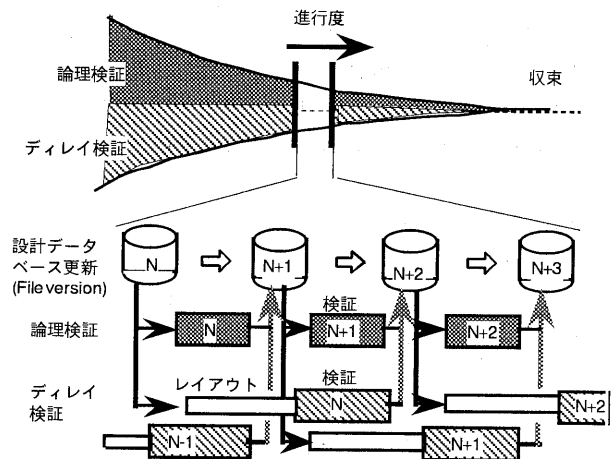


図3 論理とディレイの並行検証

(3) インクリメンタル（追加型）レイアウト設計

論理検証とディレイ検証をコンカレントに進める手法では、それぞれの不良対策が相互に及ぼし合う影響を極力押さえることが、設計の収束を速め、設計期間を短縮する上で重要である。特にレイアウト設計に起因するディレイ不良を修正する場合、論理そのものには影響を及ぼさず、かつ、問題のないレイアウト部分をフィックスした状態で必要箇所のみを修正できる部分変更機能が望ましい。本システムは、設計変更に関係しない部分については設計済みのレイアウト結果を保存し、変更対象部分のみを再レイアウトするインクリメンタルレイアウト機能をもつ。インクリメンタルレイアウト機能はセルの配置、セル間およびリージョン間の配線それぞれに対して作用することができる。

インクリメンタルセル配置においては、図4に示すように、設計変更が発生していないブロックに属するセルは配置位置変更をせず、設計変更が発生したブロックに属するセルのみを該当ブロックが占めている領域のなかで再配置する。ブロック内のセルの再配置には、セルの配置をできるだけ保存し、セルの削除後のスペースの詰め合わせおよび追加セルの挿入を行う機能をもつ。

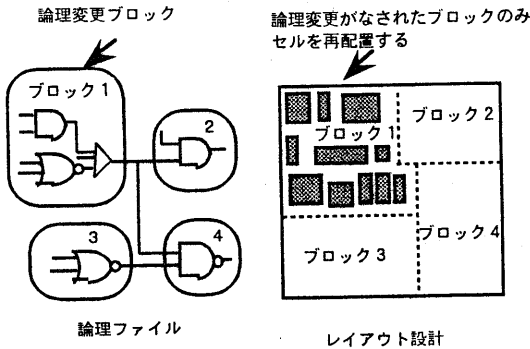


図4 インクリメンタルレイアウト（配置）

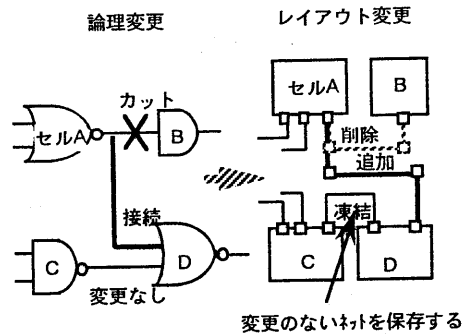


図5 インクリメンタルレイアウト（配線）

インクリメンタル自動配線は設計変更後のネットリストを対象とする。すなわち、図5のように設計変更の対象となったネットに対し、パターンの削除および追加を行う。初期設計時の自動配線では、配線率の向上のため既配線を移動しながら配線するが、インクリメンタル自動配線では既配線は保存する。

図6にインクリメンタルレイアウト手法を用いたディレイ対策過程でのディレイ分布の一例を示す。(A)の部分はレイアウト改善によりディレイ対策の効果があつた部分である。(E)の部分は改善の前と後のいずれも目標ディレイは満足している。(B)の部分は目標ディレイ値は未達であるが、レイアウト改善の効果が認められる。(C)(D)の部分は逆にディレイ値が大きくなっている。(E)の部分固定しながら(B)(C)(D)に対してインクリメンタルにディレイ対策をくりかえすことにより、ディレイ設計を完了させる。

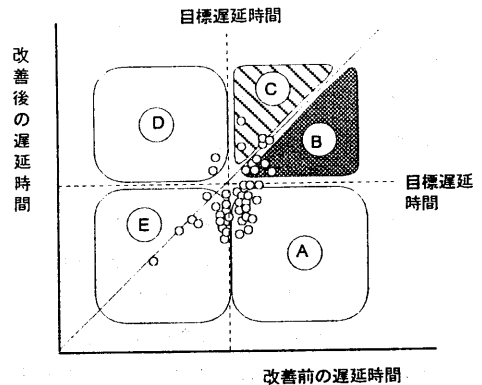


図6 インクリメンタルレイアウトの遅延時間改善効果

目標2、すなわち高性能チップを実現するための方策としては、きめ細かいディレイ制御設計が不可欠である。これには、「パスディレイ」を短縮することと「クロックスキュー」を小さくすることが必須となる。パスディレイに対してもっとも支配的な項目は配線の長さである。HARP-1チップでは以下に述べる配線戦略（配線の詳細なprocedureは4章で述べる）によってパスディレイの低減を図った。4層のメ

タル配線層の中で最上層の配線抵抗は他の層に比較し低く設計してある。配線幅には図7(a)に示すように基本配線格子幅(通常配線幅)に対し、3倍、5倍と、合計3種類の幅を選択することができる。

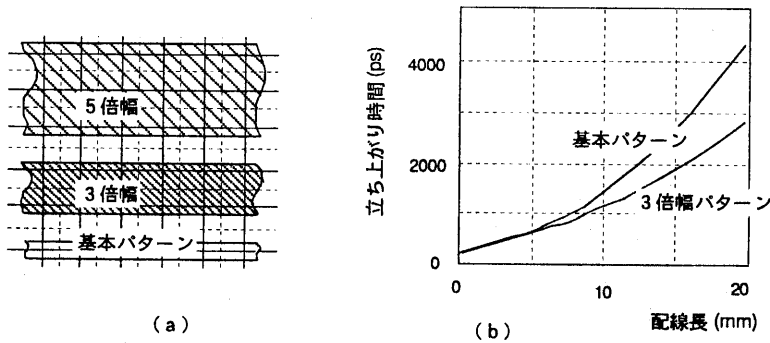


図7 配線幅による遅延低減効果

これは、クリティカルパスに対して、幅を太くした配線を行なうことによって配線抵抗を下げ、ディレイ値を低減させる狙いもつ。図7(b)に、配線幅の違いによるディレイ低減効果を示す。

HARP-1チップでは、図8に示すようにクロックをメッシュ状の配線としクロック供給元からフリップフロップまでの距離を一定の誤差内で合わせるようにした。すなわちチップ周辺部に配置されたクロックアンプから、スタンダードセルリージョンにメッシュ状に配線を準備し、内部に配置されている最終段クロックドライバに接続する。この接続は自動配線により最終段クロックドライバからもっとも近いメッシュ配線に接続する。これをクロックフック配線と呼ぶ。

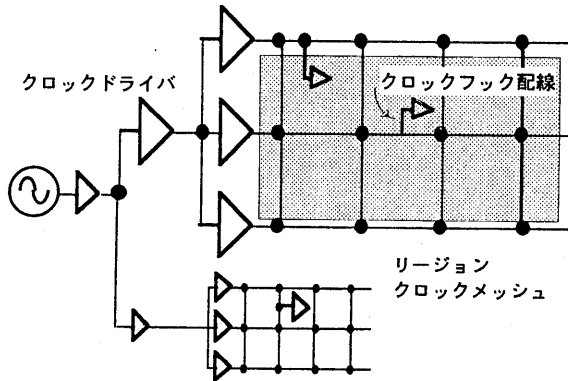


図8 クロックスキュー制御配線

メッシュ配線は電源配線とペアでスタンダードセル部に用意しておくが、その間隔はメッシュ内のすべての最終段クロックドライバが一定距離以内で接続できるようになっている。クロックフック配線は最短配線とする必要があり、配線迂回を回避するため一般信号の配線より前に行う。

4. タイミング指向レイアウト方式

HARP-1の開発では、論理設計が完全に閉じていない状態から、レイアウト設計を開始し、設計を収束させて行く戦略を採った。このために、設計の初期段階では指示項目を少なく、設計の進行に伴ってよりきめ細かい指示を与えることでレイアウト設計を進めてゆく。

(1) 論理構造保存配置手法

高性能チップのディレイ設計を容易にするためには、論理設計の単位とディレイ管理の単位を一致させることがよい。そのため、論理結合度が強い同一機能ブロック中のセルは、レイアウト時には論理構造を保存し近隣に配置する。また、ブロック間のネットのディレイを管理するためには、配置時のブロックの相対位置関係が重要になる。このため、自動配置の機能として、以下の2項目の指示を受け入れるようにした。

- (i) ブロックサイズの制御；ブロックサイズの上限を制約することでブロック内部ディレイを保証する。
- (ii) ブロック相対配置、形状の指示；これにより、論理設計者が意図したとおりの論理構造を保存した配置を行なう。

セルはブロック（論理設計者が取り扱う500～1000ゲートに分割した論理機能のまとめり）を単位として設計者の指定に従った相対位置関係を遵守しつつ、同一ブロックに属するセルは地続きで自動配置される。配置アルゴリズムはすでに発表した2次元クラスタリング配置[5]をベースに、上記の設計者の指示をアクセプト可能なように拡張した。設計者によるブロック相対配置位置と形状の指示は、図9に示すように、リージョンを格子状に分割したマクロ格子番号を用いて行なう。クラスタを部分リージョンへ割り付ける際に、相対位置が指定されているブロックを含むクラスタを、指定されたマクロ格子を含む部分リージョンに置く。また、マクロ格子を必要な形に陣取ることによってブロックの形状を指定する。これにより相対位置と形状を制御することができる。図10は制御論理部のリージョンで、ブロックの相対配置指示を行なった場合と行なわない場合のディレイ分布を示したものである。この図は設計者の意志を取り入れた配置の効果を示している。

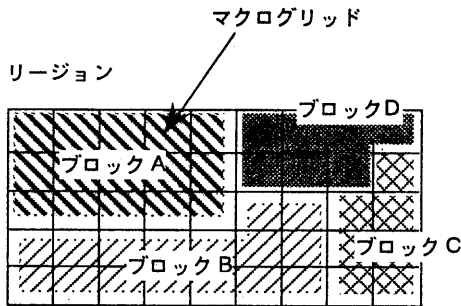


図9 ブロック配置時のマクロ格子

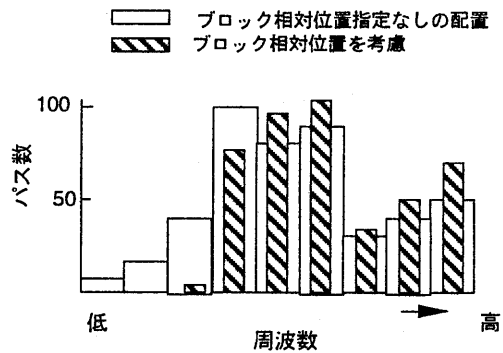


図10 配置によるディレイ分布の違い

(2) ディレイ最適化配線方式

自動配線が対象とするネットは、(1)クロックネット、(2)論理信号ネット、の2種類である。

クロックネットについては配線スキューを最小にすることが重要であり、その実現方法についてはすでに述べた。以下では、論理信号の配線について述べる。

論理信号ネットの内、一般に、リージョン間配線は距離が長いことディレイ性能を支配する大きな要因となる。パスディレイを制御しながら自動配線を行う際に、設計者の指示として下記の項目を受け入れ

るようにした。

(i) 幅広配線、層指定配線；配線抵抗の制御によるクリティカルパスの優先レイアウトを行なう。

(ii) 配線経路の割付；クリティカルパスへのマクロセル上フィードスルーの優先割付けにより迂回配線を抑止する。

幅広配線と層指定配線を比較した場合、配線幅を広くすると配線チャンネルを多く必要とするため、配線幅の選択はチップサイズとのトレードオフを考慮する必要がある。使用する配線チャンネルの量の点で層指定配線の方が有利であり、配線長が長いネットの配線に対しては、層指定配線を行った方が高密度実装が可能となる。HARP-1チップの設計では配線幅の広い配線の数を見積もるためにリージョン間全ネット数の20%以下とした。自動配線において、配線処理を行なう順序は大きな意味をもつ。早い順序で処理されるネットは配線領域が混雑していないときに経路が探索されるため、最短経路に近い配線が得られる。従って、レイアウトがクリティカルなネットは配線順序を指定することで配線長の短縮を図る。幅広配線、層指定配線を行なうため「矩形探索迷路法」と呼ぶ新しいアルゴリズムを開発した。本アルゴリズムは、配線可能領域を矩形領域に分割しその矩形領域単位で経路を探索するものであり、処理が高速で、配線幅や層の制御が容易等の利点を持つ。図11を用いて矩形探索迷路法アルゴリズム説明する。

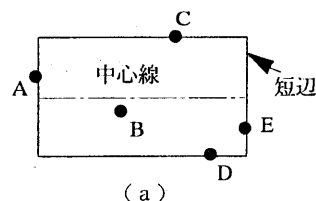
•Step 1: 探索開始点の決定

(1) 1つのネットを構成する全ての端子を包含する最小の矩形を求める。

(2) 矩形の短辺に直交して中心線を引く。

(3) 短辺上の点で中心線に最も近い点を探索開始点とする。

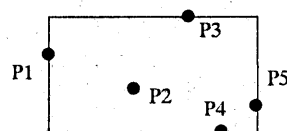
図11(a)では点Aが探索開始点となる。



(a)

•Step 2: 端子の探索順序の決定

マンハッタン距離 $K_i = |P_1 - P_i|$ ($i=2, 3, 4, \dots$) を求め、その値が小さい順に探索順序とする。図11(b)の例では $P_1 \rightarrow P_2 \rightarrow P_3 \rightarrow P_4 \rightarrow P_5$ となる。以下では、この順に従って矩形探索を行う。



(b)

•Step 3: 矩形探索その1 (P1→P2)

(1) P1より端子の層の矩形を出す(L1)。

(2) L1に交差する矩形を求める(L2)。

(3) L2に直交する矩形をP2に向かって発生する(L3~L5)。

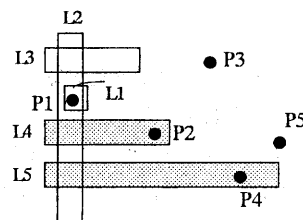
(4) 以下の3ケースで該当するものを採用する。

ケース1 P1-P2間が矩形で結べればその矩形をルートとする。

ケース2 P1-P2間は結べないが、他の端子と結べた場合は、その矩形もルートとする。

ケース3 P1-P2間および、他の端子のいずれとも結べない場合はP2に最も近づく矩形をルートとする。

図11(c)の例ではケース1と2が該当するので矩形L4、L5を選ぶ。



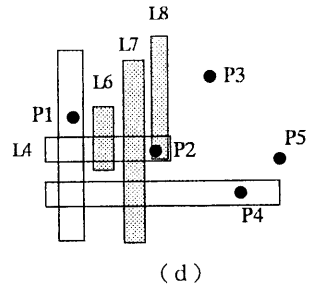
(c)

図11 矩形探索迷路法

•Step 4: 矩形探索その2 (P2→P3)

図11(d)(e)参照。

- (1) L4に直交する矩形をP3に向かって発生する(L6~L8)。
- (2) この場合は、前記ケース3が該当するので矩形L8を選ぶ。
- (3) L8に直交する矩形をP3に向かって発生する(L9~L10)。
- (4) この場合は、ケース1に該当するので矩形L9を選ぶ。



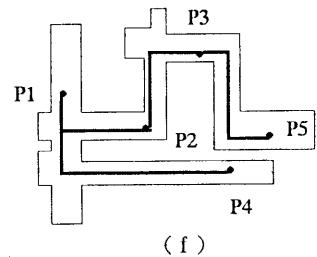
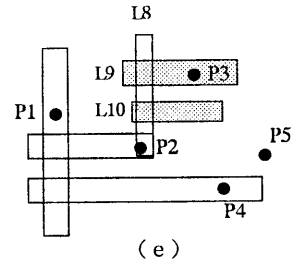
•Step 5: 矩形探索その3 (P3→P5)

ステップ3、4に同じ。

•Step 6: 配線経路の決定

求まった矩形ルートの中で最短の配線経路(エレメント、スルーホール)を求める。図11(f)参照。

矩形探索迷路法は幅をもった矩形単位に配線するアルゴリズムのため、Step 3, 4の探索時に、矩形の幅が3倍もしくは5倍以上という制約を設けることにより幅広配線を実現することができる。例えば、図12のように、端子SとTの間を幅dの配線をしたとき、端子の引出し部分は端子の大きさ以上の矩形を使用し、w以上の幅のみの経路を探索する。そして、見つかった経路の内部に幅dの配線パターンをひく。層指定配線の場合も同様で指定された層に存在する矩形領域のみを探索の対象とすることで実現できる。



5. むすび

開発期間の短縮と、高性能チップの実現を狙いとするDAシステムについて述べた。論理とディレイの並行検証、および、インクリメンタルレイアウトにより開発期間の短縮を、ブロックの位置・形状、配線順序、配線層、配線幅等の項目をアクセプトし設計者の意図を反映したレイアウトを行うことにより、きめ細かなディレイ制御が可能となった。本システムをRISCチップ設計に適用し有効性を確認した。

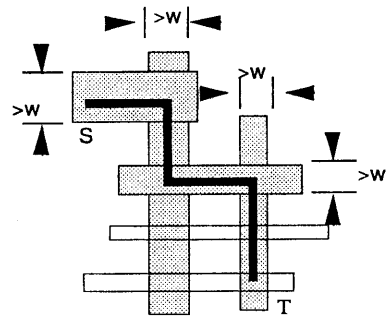


図12 幅広配線

参考文献

1. D. Alpert and D. Avnon, "Architecture of the Pentium Microprocessor," IEEE Micro, Hot Chips IV, pp.11-21, June 1993
2. E. McLellan, "The Alpha Architecture and 21064 Processor," IEEE Micro, Hot Chips IV, pp.36-47, June 1993.
3. T. Asprey, et al., "Performance Features of the PA7100

- Miroprocessor," IEEE Micro, Hot Chips IV, pp.22-35, June 1993

4. K. Matsubara, "A 120 MHz BiCMOS Superscalar PA-RISC Processor," HOT Chips V, pp.8.1.1-8.1.9, August 10, 1993

- 5 Y. Ogawa, et al., "Efficient Placement Algorithm Optimizing Delay for High-speed ECL Masterslice LSI's", Proc. 23rd DAC, pp.404-410, June 1986