

解説

3. ハードウェアアルゴリズムの設計法



3.1 ハードウェアアルゴリズムと VLSI 設計†

須藤 常太** 中島 孝利** 吉村 寛**

1. はじめに

近年の集積回路技術および自動設計技術の進歩により、数 10 K ゲートの大規模な論理回路を高性能・低価格で実現できるようになってきた。このため、これまで主としてソフトウェアで実現されていたアルゴリズムを専用のハードウェア、すなわち VLSI で実現する方がシステムの性能/価格比を向上できるようになり、VLSI に適したアルゴリズム（これを通常ハードウェアアルゴリズムと呼んでいる）の研究が注目され始めてきた。本章ではハードウェアアルゴリズムを、LSI の特徴を発揮できる構成法、設計法まで含むと解釈し、まず VLSI 設計の特徴を述べ、我々が設計してきた VLSI の具体的な設計データ（論理・回路・レイアウト設計・テスト）をもとに、VLSI 設計の観点からみたハードウェアアルゴリズムに対する要望・問題点を概観してみる。

2. VLSI のインパクトと特徴

(1) VLSI のインパクト

論理機能を VLSI 化すると図-1 に示すようにシステムの性能/価格化を大幅に向上させることができる。すなわち、①微細化に伴って大規模の論理を 1 チップに搭載できる訳であるから、高機能化を図ることができ、また、高速化・低消費電力化も達成でき、②実装部品数の激減、装置の小型化等による低価格化を図れ、さらに③プリント板上での半田接続点数やプリント板間の接続配線数の減少による信頼度の向上を図ることが可能となる。したがって、もともと取り扱うデータ自身がデジタルで VLSI 化に適している情報処理装置や交換機だけでなく、各種伝送装置、さらにはオーディオ装置を始めとする本来アナログ信号を

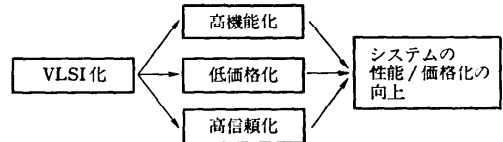


図-1 VLSI 化による効果

取り扱う民生用の各種機器も VLSI 化が急ピッチに進められており、今やエレクトロニクス装置は VLSI 無しでは考えられないほど大きなインパクトを与えている。

(2) VLSI 設計の問題点と解決策

前述したように VLSI のインパクトは計り知れないが、VLSI の実現には下記のような特有の各種問題がある。

① 配線領域の増大とそれに伴う性能低下

VLSI 設計では配線層数がプリント板と異って極めて少なく、設計上の大きな制限条件となっている。現状技術では金属 2 層が一般的であり、3 層がようやく実用に供されるようになってきているが、10 層以上が利用可能なプリント板と比べるべくもない。このため論理素子の占有面積よりも配線の占有面積がはるかに大きく、チップ面積の 7~8 割を占め、コストの増大を招く。さらに配線が長くなり、配線容量の増大による性能低下をも招く。

上記問題に対処するには局所的にはできる限り規則性の高い論理・回路構成を採り、また、大域的には内部バスを利用した効率的な配線手法をとることが重要である。さらに、ハードウェアの並列性を十分発揮させる構成が性能を大幅に向上させるのに有効である。

② 端子数の制限

VLSI の制約条件の 1 つに、入出力端子数の少ないことが上げられる。すなわち、VLSI のチップ面積は一辺の長さの 2 乗に比例して増加するが、チップに取り付けられる入出力端子数は辺長に比例するだけである。したがって VLSI では内部機能を複雑にできる

† Hardware Algorithm for VLSI Design by Tsuneta SUDO, Takayoshi NAKASHIMA Hiroshi YOSHIMURA (Atsugi Electrical Communication Laboratory, N. T. T.).

** NTT 厚木電気通信研究所

割には入出力端子数が少ない。

この問題に対しては、双方向バスを用いて入力信号と出力信号を同一ピンにアサインする方法や、データ信号をシフトレジスタにより直列化して時分割に入力する方法がよく用いられる。パイプライン乗算器やテストに多用されるスキャンレジスタはこの好例と言える。また時間的に変化しない制御信号を入力する必要のある VLSI、例えば、設定パラメータを変えるだけで各種用途に使える VLSI の場合、制御信号を外部より与えるかわりに ROM およびそのデコーダを内蔵し、デコーダ信号を外部ピンより与える手法によりピン数削減を図ることができる。

③ 試験の困難性

出来上がった VLSI が設計者の意図通り、電氣的・論理的条件を満足しない場合、誤り箇所を特定する必要がある。誤り箇所を特定するためのテストパターンは一般的には入出力端子のみから与えて解析できることが要求される。しかし論理 VLSI は一般的に順序回路を含むこと、また故障が単一縮退故障であるとは限らないこと、さらにはテストパターンがゲート数の2~3乗に比例することからテストパターン生成技術のみから VLSI の解析を容易にすることは困難である。一方測定器からの解決策もない訳ではない。すなわち電子ビームを VLSI に照射し、二次電子を測ることにより最上層配線の電位を調べることができる非接触テスタ（電子ビームテスタ）がある。これは将来有効

に使用されると考えられるが、測定速度、効率化の点でこれからの技術であろう。したがって、機能・論理設計段階より、あらかじめ試験を容易にする VLSI 構成面からのアプローチを探ることが必須となる。すなわち、規則性が高く試験効率の高いハード構成法や試験用の付加ハードウェアをチップ内に持ち込む手法である。後者の手法は大別して以上の二つがある。

(i) スキャン方式

試験のための補助回路であるスキャンレジスタを組み込み、等価的に外部端子数を増加させる手法。

(ii) 自己診断方式

チップ上にテストパターン発生器等試験用回路を内蔵させ、試験モードを設定することにより自己診断を可能とする方法。

④ 設計工数の増大

VLSI の設計においては数 10 K~数 100 K 個のトランジスタが要求機能どおりに接続され（論理設計）、トランジスタ性能や配線容量に起因する信号遅延、雑音等が十分に考慮されて要求速度を満たし（回路設計）、さらにプロセスから決まるデザインルールを犯すことなく数 100 K 本もの配線が 1 本の接続ミスなく配線され（パターン設計）なければならない。これら設計工数は膨大であり、数十人年にも及ぶ。さらに、上記設計において 1 箇所でも誤っていると、再び VLSI を製造する必要があり、時間とコストを大幅に消費することになる。したがって無びゅう性を保証

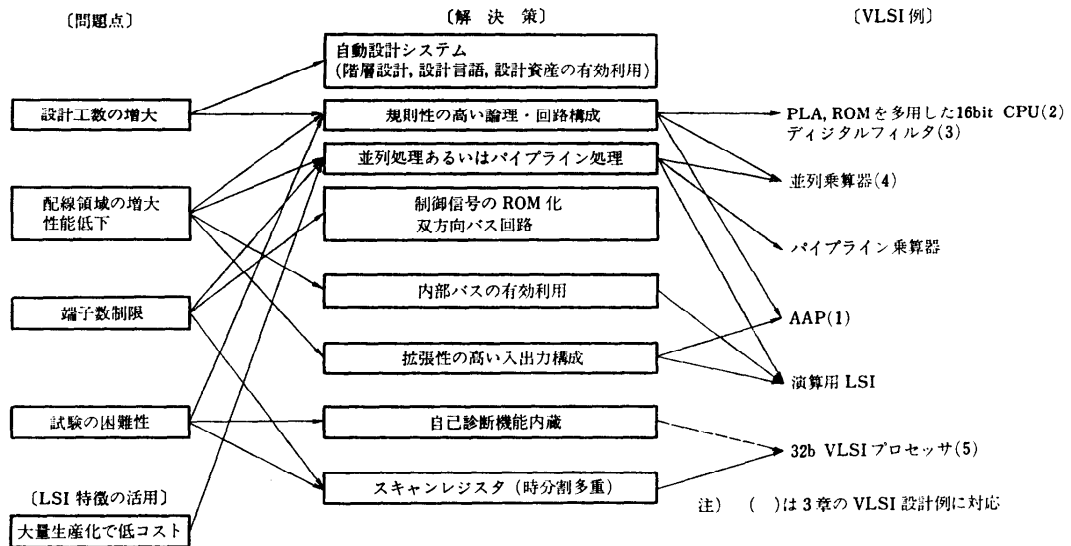


図-2 VLSI 設計上の問題点とその解決策

し、かつ設計工数を短縮するには自動設計システムを駆使することが必須となるが、ハードウェアアルゴリズムの面から見ると規則性の高い論理・回路構成を採ることも重要である。

また LSI の特徴を積極的に活かすことも考える必要がある。その1つに、大量生産にのれば低コスト化が達成できるという LSI の特徴を活かすため、単一品種の LSI を多数個使用し性能を上げようとする考え方がある。アレイに代表される構成であるが、これは超高性能な演算用プロセッサを低価格で提供することを可能にするだけでなく、従来のプロセッサに不適であった画像のような2次元データの高効率処理に新しい展開をもたらすものでもある。

以上述べた VLSI 設計上の問題点とその解決策および VLSI 特徴の活用策をまとめて図-2 に示す。以下これら解決策の適用を試みた VLSI の設計例を述べる。

3. 各種 VLSI の設計とハードウェアアルゴリズム

通研では各種論理 VLSI の開発を行ってきた。これらの中から5例を選び前章で述べた LSI の特徴活用、LSI の問題点解決の観点から振り返り、設計時点でどのような方策を採ったか、またそれがどのような効果を生んだか等についてまとめてみる。なお図-2 に問題点、特徴活用の観点、目的、方策と LSI 例との関連についてまとめた。

(1) 規則性・拡張性の高いアレイ論理 (アレイプロセッサ用 LSI)

VLSI のインパクトが最も大きいシステムの1つ

は、高並列計算機である。これまではハードウェアが膨大となりほとんど実現不能であったような大規模システムが多数の VLSI を使用することにより比較的容易にかつコンパクトに実現されるからである。特に通研で開発した2次元アレイプロセッサ AAP-1 (Adaptive Array Processor)¹⁾ は、設計・テストの容易な同一の規則的構成の LSI を適当数並べて構成されており、その典型である。

図-3 は、AAP-1 の構成概念図である。高並列演算部のアレイ部は、1ビットの要素プロセッサ (PE) の 256×256 の規則的な配列からなり、多数の同一 LSI (AAP-LSI) で構成されている。各 PE は制御部から配られる共通の命令のもとで動作する (SIMD 型マシン)。65,536 (=256×256) 個の PE の並列動作により、AAP-1 は種々の2次元データ処理に高い処理能力を発揮する。

[AAP-LSI の基本構成]

AAP-1 の PE は、ゲート規模がわずか百数十ゲートの1ビットプロセッサであり、1チップに数十個以上搭載することができる。LSI の高集積度と設計・テストの容易性とを確保するには、PE アレイの規則的な配列構成をそのままレイアウトに反映するのが最も自然である。そこで、チップのレイアウト構成は、同一の細胞状 PE を基盤の目状に配列するセルラ・アレイ構成としている。この構成のチップでは、細胞内の配線はあまり長くない。配線の引き回しが、ほとんど細胞内に限られるからである。そこで、負荷駆動能力では若干劣るものの設計当時 (約5年前) 最も高い集積度の得られた n E/D MOS (Al 1 層, 3μm ルール) を採用した。しかし、それでもいかに高集積度を

得て、より多くの PE を搭載したチップを短時間に設計するかが、システムを経済的に実現する上でのポイントである。この意味でチップのレイアウト設計は極めて重要である。

[セルラ・アレイの概略レイアウト]

セルラ・アレイ構成のチップのレイアウトは極めて単純で、矩形にまとめあげた PE を基盤の目状に並べればよい。しかし、AAP-1 の場合、このように単純な並べ方では信号線や電源線の配線領域が非常に大きくなってしまふ。これは、電源線と50本を越える各 PE 共通の制御信号線をチップ全体に張りめぐら

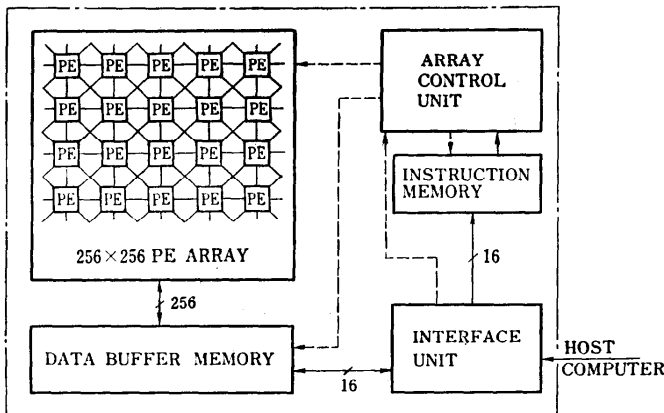


図-3 AAP-1 の構成図

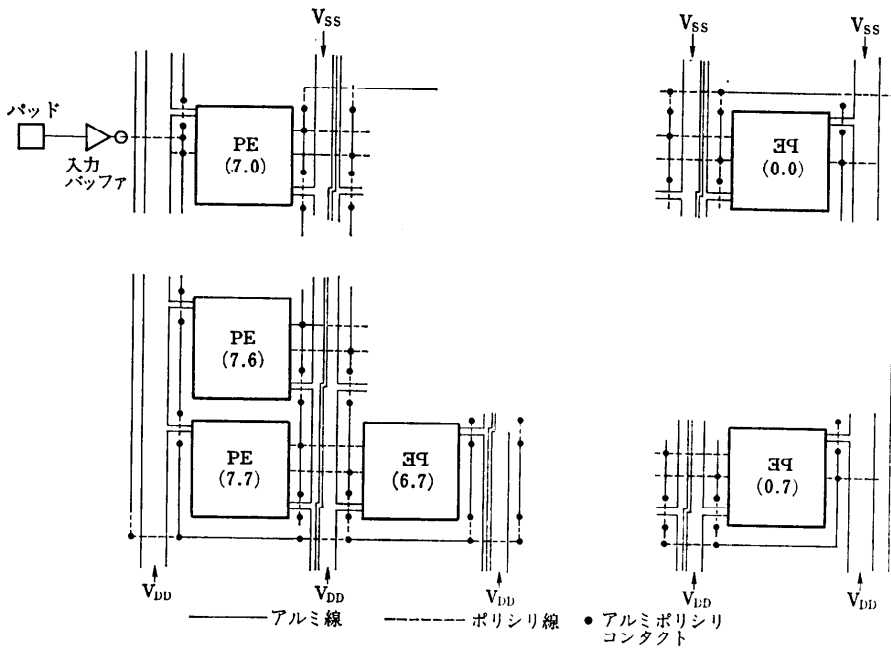


図-4 セルラ・アレイの概略レイアウト

さねばならないからである。そこで、AAP-1 ではチップ・レイアウトに若干の工夫を加えている。図-4 にセルラ・アレイの概略レイアウトを示す。図から明らかかなように、ミラーをかけた PE(7.6) とそうでない PE を交互に配置している。これにより、1本の制御信号線で左右両側の PE に信号を供給し、制御信号線の配線領域を半減している。また、電源線についてはパッドから離れるにしたがい電流値が規則的に減少する点に着目し、電流値に応じて線幅をステップ状に狭くして電源の配線領域を半減させている。

〔PE のレイアウト〕

PE は、350 素子 (論理部) + 900 素子 (レジスタ・ファイル部) からなり、人手でも比較的容易に設計可能な規模である。また、繰り返しの単位なので、PE の高密度化はチップ全体の集積度を上げることにつながる。そこで、PE のレイアウトは人手により念入りに設計し高密度化をはかっている。

〔レイアウト結果〕

試作したチップの写真を図-5 に示す。規則性を活かしたむだのない配置・配線により 8×8 個の PE (約 81,000 素子) を 10.0×9.4mm² のチップに実現している。

〔設計工数〕

本チップは規則性が高いため、高い集積度が得られ

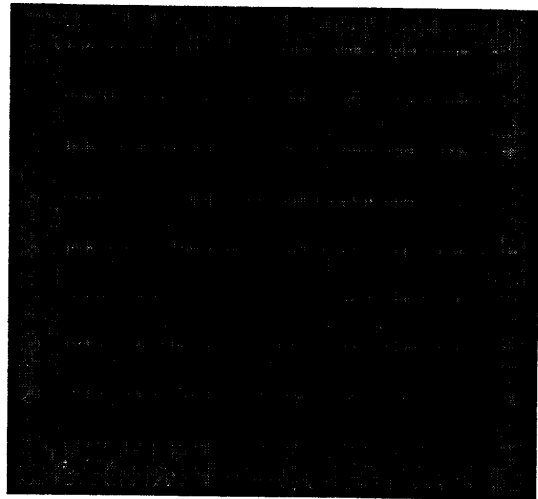


図-5 AAP-LSI のチップ写真

るだけでなく、設計が極めて容易である。実際、人手設計にもかかわらずレイアウトにわずか 5 月程度しか要していない。また、その他論理設計・テストパターン発生等の工数も小さく全体でも 1 年以下である。この値は、同規模の論理 VLSI (DA を用いてレイアウト工数を低減した場合) に比べて、数分の 1 以下である。

試作した LSI の諸元を表-1 に示す。また、本 LSI の使用により、256×256 の大規模な PE アレイをわずか 0.33mm² の容積に実現した AAP-1 システムの写真を図-6 に示す。なお、AAP-1 の性能はこれまで主に進めてきた LSI-CAD への応用において 100 MIPS 程度であることを確認している。また他の画像処理、パターン認識においても非常に高速な処理が可能で、ハードウェア提供側からも、応用側からも VLSI に適したプロセッサということができよう。

(2) レギュラロジック (16 ビット CPU)

論理 VLSI 設計において ROM, PLA 等のレギュラロジックを多用するケースが増えてきている。これは、設計のターンアラウンドタイム短縮が主な狙いである。ROM, PLA は規則的構造を有するため、論理設計からテストに至る一連の処理が簡単化でき、設計工数を削減することができる。問題としては、性能、

面積が一般にランダムロジックより劣る点があげられる。

16 ビット 1 チップ CPU に対して、ランダムロジック主体の場合と PLA 等を積極的に利用した場合の 2 通りを設計し、比較した例について述べる²⁾。具体的には 2 層 Al 配線・2μm CMOS 技術をベースとし、PLA 等のレギュラロジックを大幅に導入した構成と、ランダムロジックをスタンダードセルで実現する構成を比較している。本検討の主な特長は以下の通

表-1 AAP-LSI の諸元

搭載 PE 数	8×8
デバイス	NE/D MOS
素子数 (/チップ)	81200
集積密度 (素子/mm ²)	PE: 2100
(素子/mm ²)	chip: 860
サイクル・タイム(ns)	250
消費電力 (W)	1.6
パッド数	141

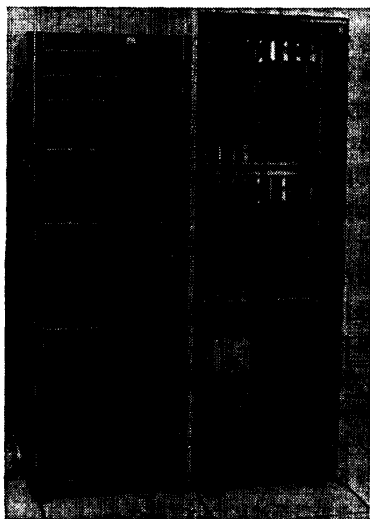
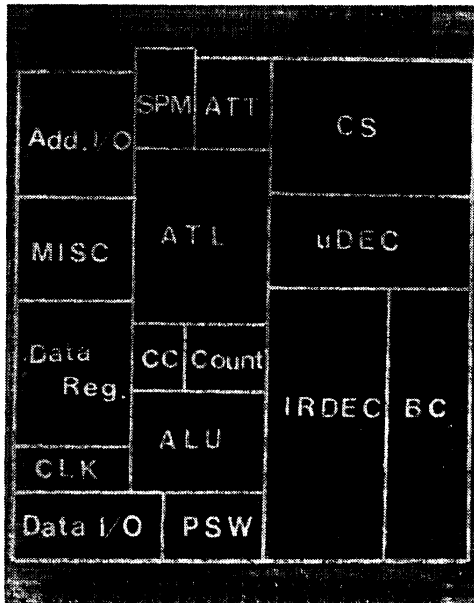
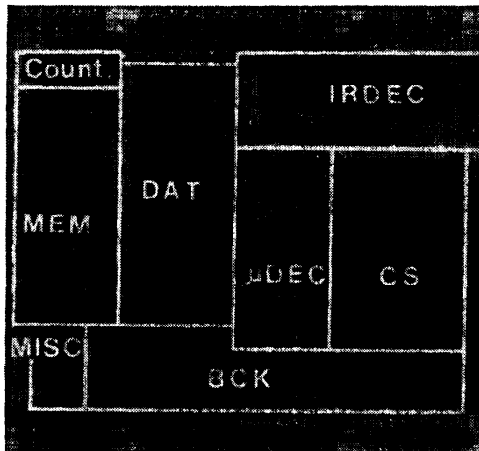


図-6 AAP-1 システム図



(a) ランダムロジック中心の構成のチップ



(b) レギュラロジック中心の構成のチップ

図-7 16 ビット CPU のチップ外観

表-2 16ビットCPU設計結果

	構成素子数		占有面積 (mm ²)		速度性能	
	ランダムロジック中心の構成	レギュラロジック中心の構成	ランダムロジック中心の構成	レギュラロジック中心の構成	ランダムロジック中心の構成	レギュラロジック中心の構成
データパス部	8,420	6,410	8.29	4.75	13.0ns (キャリ伝播速度)	14.8ns
命令デコーダ部	3,096	14,498	4.77	3.54	16.3ns	15.1ns
CPU全体	55,620	66,180	53.96	40.71	-	-

りである。

- (i) PLA の最適利用
- (ii) PLA の高速化
- (iii) PLA の高密度化

(i)については、この例ではインストラクションデコーダ等2種のデコーダを含む制御部に PLA を適用している。これは、一般に制御部の方が演算部に比して PLA 化に適することによる。すなわち、制御部の方が一般に論理深度が小さく、そのため PLA 化した場合の積項数が少なくてすむことによる³⁾。

(ii)については、信号の動作振幅を小さくした回路⁴⁾を考案し、約50%の性能改善を達成している。本回路構成は積項数が数百の大規模 PLA において特に効果を発揮する。

(iii)については、DA ツール⁵⁾により積項数の最小化を徹底するとともに、レイアウト面積最小化を狙ったフォールディングを行っている。フォールディングは入出力線および積項線の重複化を行うものであり、PLA を最大25%まで圧縮することができる。すでに DA ツールも用意されており、本検討では約50%の縮小を達成した。DA ツールを用いた実験により元の PLA 容量とフォールディングした容量の関係についても明らかにされている³⁾。

以上の項目を盛りこみ、論理・レイアウト設計を行った。レイアウト設計においては、ROM, PLA の形状が固定的であることから生じる無駄領域が問題となった。これに対しては、スタンダードセルブロック形状の自由度が大きいことを活用し、ROM, PLA の周辺部にスタンダードセルブロックを配置し、かつ形状の調整を行うことにより対処した。

図-7 にチップ写真を、表-2 に設計結果を示す。PLA 化を進めたチップの制御部については、ほぼ同一の性能、面積を達成している。

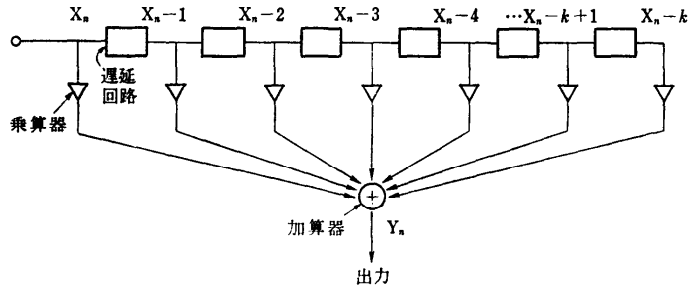


図-8 非巡回形デジタルフィルタの構成

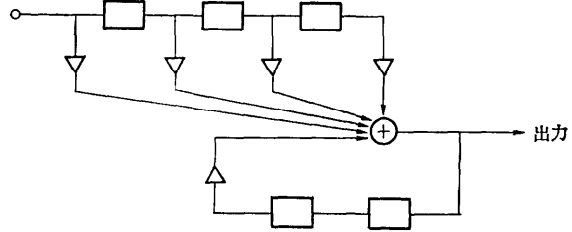


図-9 巡回形デジタルフィルタの一構成

以上、レギュラロジックを導入しても性能、面積の点で大きな損失にならないことを一例で示した。他にランダムロジックを採用している MC 6800 に対して、種々制御方式を適用し、性能、面積、規格化率を比較した例がある⁶⁾。複数個の PLA で制御部を設計した場合の結果は、上記とほぼ同じである。今後論理 VLSI の集積度がますます増大するにあたり、設計ターンアラウンドタイムの点やまたデバックや仕様変更への柔軟な対応からレギュラロジックの大幅導入は必須と思われる。これに向けた設計手法の一層の明確化が重要である。

(3) 規則性の高い回路構成 (デジタルフィルタの構成)

デジタル化された音声、画像信号の処理においてデジタルフィルタが多用される。ここでは VLSI 化に適した設計手法について述べる。

図-8 に位相線形な非巡回形デジタルフィルタの

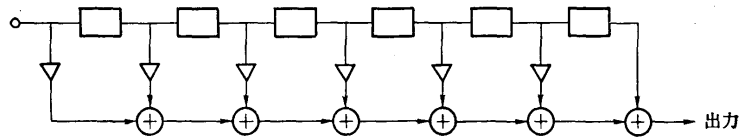
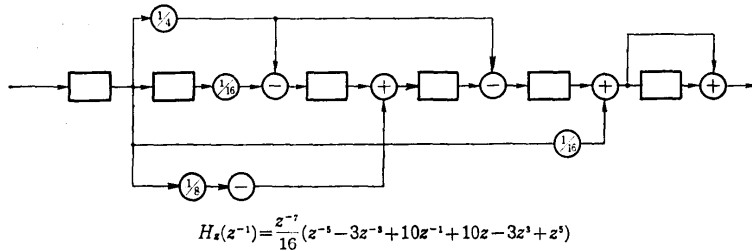


図-10 非巡回形デジタルフィルタの別構成



$$H_z(z^{-1}) = \frac{z^{-7}}{16}(z^{-1} - 3z^{-2} + 10z^{-3} + 10z^{-4} - 3z^{-5} + z^{-6})$$

図-11 乗算器を用いない非巡回形デジタルフィルタの構成

基本構成を示す。デジタルフィルタの特性は差分方程式で表現され、その回路は差分方程式の演算要素である遅延、乗算、加算をそれぞれフリップフロップ、乗算器、加算器で構成することにより実現できる。位相が線形であることが要求されない場合、図-9に示す巡回形フィルタが適している。この構成は過去の出力結果をフィードバックすることにより部品点数の大幅な削減を図ることができる。

さて、位相線形なデジタルフィルタを VLSI 化することを考える。VLSI 化において第1に考慮すべき点は配線占有面積の低減である。例えば図-8の回路をそのまま実現すると加算器のまわりに配線が集中し、面積の増大を招く。そのため図-10のように加算器を分散させ、配線の集中を避ける方法が有効である。第2に考慮すべき点はハードウェア量の削減である。図-10のように基本的なデジタルフィルタの構成では乗算器がハードウェア量の大半を占めるので乗算係数を2の中乗りに選び、ビットシフト加算あるいは減算で代用する方法が効果的である。この手法では語長に留意して設計することが重要である。図-11に加算器および減算器で実現したローパスフィルタの構成および伝達関数を示す。本構成は配線の集中もおおきく規則性の高い LSI 化に適した構成といえる。

レイアウト設計自動化の面から考察すると、現在よく用いられているスタンダードセル方式では、各セルがセル間の接続強度により配置されるので信号の流れに則して配置される保証がない。そのため配線長のバラツキや占有面積の増大を招く。したがってデジタルフィルタのように論理構成が規則的な回路に対して

は、信号の流れ、ビット構成を考慮した自動レイアウト手法が必要となる。

(4) 論理設計とレイアウト (32ビット並列乗算器)

高速演算を実現するため、並列乗算器の開発が進んでいるが、並列化に伴いハードウェア量が多くなるため LSI 化においても高密度化を達成できる論理構成、レイアウト設計が増々重要となる。並列乗算方式には、部分積を単純加算するアレイ方式や、ゲート数・論理段数の削減を狙ったブース法に基づく Wallance 木方式等がある。ここでは以上の2方式による32ビット並列乗算器をとりあげ、チップ面積、演算時間の比較から LSI 化における問題点を検討する⁷⁾。まず両方式の概略構成を図-12に示す。アレイ方式では図-12(a)の基本セルが同図(b)の様に桁シフトされる形で接続される。また、Wallance 木方式はブースの2次アルゴリズムを用い図-12(c), (d)の基本セルを用い、同図(e)の様に部分積生成部およびキャリセイブ加算部を構成し、キャリ先見部を介して積を出力するようになっている。これら両方式に対して $2\mu\text{m}$ ルールの SST-2 プロセス⁸⁾による2段シリーズゲート LCML 回路で32ビット並列乗算器を設計した。レイアウト設計では、アレイ方式は規則性が高く、セル品種が少ないことから人手レイアウトを、Wallance 木方式は配線が不規則で、セル品種が多いのでスタンダードセルを用いた自動レイアウトを採用した。表-3に両方式によるゲート数とクリティカルパス論理段数、さらにレイアウト設計結果として、チップ面積、集積密度、配線エリア占有率を示す。論理構成が規則的

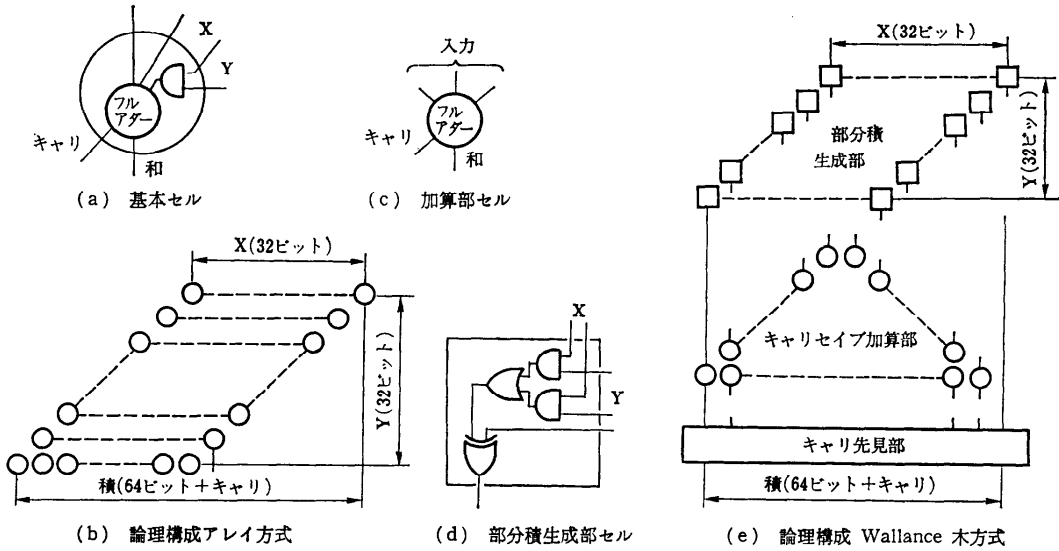


図-12 アレイ方式と Wallace 木方式

表-3 アレイ方式と Wallace 木方式の比較

比較項目	アレイ方式	キャリセイブ方式
ゲート数	14,300	11,000
クリティカルパス論理段数	136	82
チップ面積 (mm ²) (比)	7.806 × 4.32 (1)	8.544 × 8.766 (2.22)
集積密度 (ゲート/mm ²)	424	147
配線エリア占有率	40%	76%
クリティカルパス総配線長	26.6 mm	269 mm
クリティカルパス平均配線長	0.196 mm	3.28 mm
クリティカルパス平均配線遅延	Δt_w	16.8 Δt_w

なアレイ方式はゲート数が多いにもかかわらず配線エリアが少ないため、Wallace 木方式に比べ約 2 倍の高集積化を達成した。なお、設計工数は両方式ともほぼ同じであった。

演算時間を決定するクリティカルパスについては、アレイ方式のクリティカルパス平均配線付加によるゲート遅延増を Δt_w とすれば、Wallace 木方式のクリティカルパス平均配線付加によるゲート遅延増は配線長が長いこと 16.8 Δt_w となる。アレイ方式、Wallace 木方式の演算時間 t_{AA}, t_{AC} は基本ゲート遅延を t_{pd} とすると次式で求まる。

$$t_{AA} = (t_{pd} + \Delta t_w) \cdot 136$$

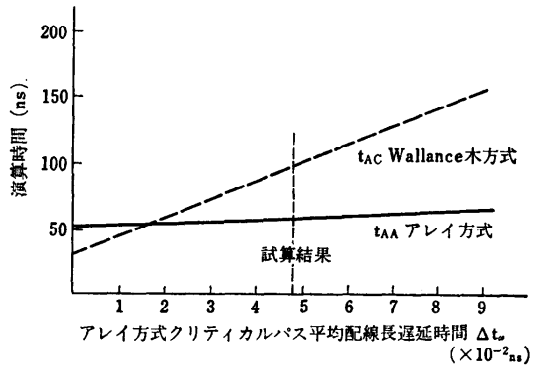


図-13 両方式の演算時間の比較

$$t_{AC} = (t_{pd} + 16.8 \Delta t_w) \cdot 82$$

本プロセスでの基本ゲート遅延 t_{pd} は 0.39 ns であり、 Δt_w をパラメータとした両方式の演算時間を図-13 に示す。

Δt_w が 20 ps 以上ではアレイ方式のほうが演算時間が速くなることが判る。今回のプロセス、および回路定数では $\Delta t_w = 48$ ps であり、アレイ方式は Wallace 木方式に比べ、論理段数は約 1.7 倍であるが、演算時間では約 1.6 倍高速化できる事が明らかになった。

以上のように、Wallace 木方式の並列乗算器はゲート数が少ないので汎用の SSI や MSI を使用して並列乗算器を達成する場合は有効であるが、1 チップ VLSI 化を考えた場合、配線が不規則なため、チ

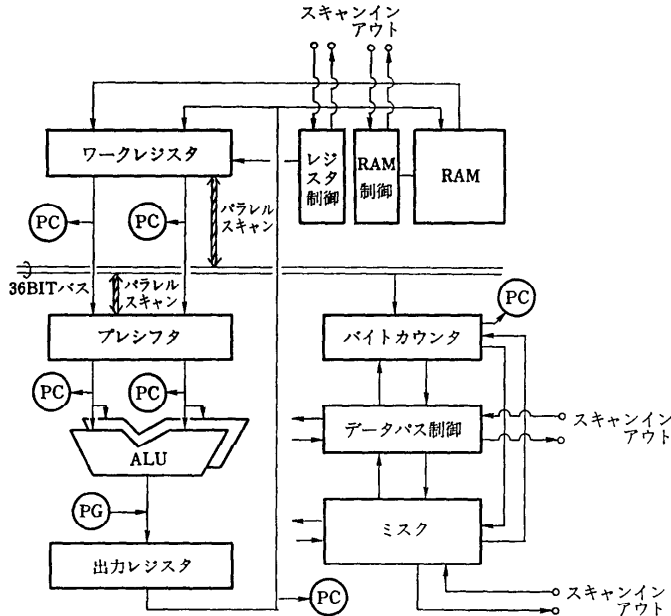


図 14 32ビットプロセッサのブロック図

ップサイズの増大をまねくことが明らかとなった。一方、アレイ方式では配線も含めて極めて規則性が高いセルレイアウトができるため、高集積化が図れる。さらに演算速度に関しては、ゲート遅延が極めて速く、配線遅延が相対的に大きくみえるプロセス、回路条件の場合、信号配線長が短いアレイ方式のほうが高速になるという結果をもたらす。

(5) テスト・診断容易性 (32ビット VLSI プロセッサ)

20K ゲート規模の 32ビット VLSI プロセッサ⁹⁾の実現にあたり、スキャンレジスタの設置によりテスト容易化を図った。スキャンレジスタを用いたテストの問題として、シリーズに接続した F/F にテストパターンを書き込む、あるいは F/F の内容を読み出すために必要なクロック数が増大し、テストシーケンスが長大になってしまうことがあげられる。この問題を回避するために、直列に接続する F/F 数を適当に制限し、各スキャンレジスタはテスト用制御回路によって指定する方式 (アドレス・スキャン方式) を採用した。

またテスト用ハードウェア量を抑制するために、データ系レジスタのように外部バスに接続されたレジスタは、テスト時にそのレジスタを指定してバスより直接データを書き込み・読出しができるパラレルスキ

ャン方式とした。

以上によりテスト用ハードウェアの増加を 10% 程度に抑え、故障検出率を大幅に向上させた。また本チップを含むシステムの高信頼化を図るために VLSI 内でのパリティの持ち回り、および ALU の 2 重化を行い、スキャン結果と合わせて障害検出端子として出力させた。図-14 に 32ビット VLSI プロセッサのブロック構成を示す。なお、チップ内数箇所には診断用内部パッドを設け、故障診断時に論理ゲート内部の波形も観測可能とした。このチップに対してはこれらの方策により比較的容易にチップの診断・試験を行うことができたが、チップの本質的構成や搭載機能に応じた方策を考えることも大切である。

(6) まとめ

ハードウェアアルゴリズムは、ハー

ドウェア、すなわち LSI で実現されるものであり、LSI の特質の十分な理解のうえで思考されねばならない。(1)で述べたアレイプロセッサ用 LSI は最初からこのような立場にたって検討されており、設計や診断・テストの容易性、コスト低下策などが工夫されていると同時に、2次元データの高速処理に本質的に向いた構成となっている。これはあるアルゴリズム専用のハードウェアというよりは、2次元データ処理全般に適したハードウェアである。動作も、アレイ全面のデータの一括処理だけでなく、シストリックアレイのような動作も可能である。(2)~(5)は特にある応用分野に限定したアルゴリズムの LSI 化というよりも、LSI の特徴の活用、問題点の克服という観点から、ハードウェアアルゴリズムを考える時どのような事項に留意せねばならないかについて具体的設計例をもとに概観した。しかし、ここで述べた事項も LSI 技術の発展に応じて変わってくるべきものもあり、新しいハードウェアアルゴリズムを思考するにあたっては常にこのあたりの状況を把握しておくことが必要である。

4. あとがき

従来のプロセッサはハードウェアの高価な時代の発想をもとに作られており、具体的なアルゴリズムの実

現はそのプロセッサ上のソフトウェアによって実現されてきた。柔軟ではあるが低速の従来の方法に対して、低コストの LSI をふんだんに使用することで、ハードウェア上にアルゴリズムを直接実現し、高性能なシステムの実現が期待される時代となった。LSI の特質をはっきりと理解し、ハードウェアアルゴリズムの新たな展開を図ることが大切である。

謝辞 LSI 設計データの提供、討論をいただいた厚木通研集積応用研究室および論理回路研究室の永谷三義、松広一良、石野文明、近藤利夫、深見健之助、平田道広の諸氏に深謝いたします。

参 考 文 献

- 1) Kondo, T. et al.: A Large Scale Cellular Array Processor, AAP-1, ACM CSC '85, pp. 100-111 (Mar. 1985).
- 2) Kasai, et al.: An Integrated Modular and Standard Cell IC Design Method, ISSCC '84, pp. 12-13 (1984).
- 3) 松広: VLSI のブロック面積推定法, 信学技報, Vol. 83, No. 264 (1984).
- 4) 深見他: CMOS PLA 高速化の検討, 信学会全大, No. 392 (1983).
- 5) 宮下他: PLA 設計支援システム PLACAD 研実報, Vol. 32, No. 6, pp. 1339-1352 (1983).
- 6) Obrebska, M.: Algorithm Transformations Improving Control Part Implementation, ICCD '83, pp. 307-310 (1983).
- 7) 宇敷他: 32×32 乗算器構成法の検討, 57 年度信学全大 No. 387.
- 8) 山内他: SST-2 による LCML ゲートの特性, 5 年度信学全大 No. 334.
- 9) 吉村他: 32 ビットプロセッサの VLSI 設計, 昭和 60 年度信学全大 S4-6.

(昭和 60 年 4 月 17 日受付)