計算機アーキテクチャ 45－5
（1982. 6. 1）

マクロ変換手法による汎用計算機のＬＳＩ化

今村貞良，荻原拓治，中村俊一郎，寺井正幸
（三菱電機）

1．はじめに

ショットキーＴＴＬ（Ｔｒａｎｓｉｓｔｅｒ Ｔｒａｎｓｉｓｔｅｒ Ｌｏｇｉｃ）のＭＳＩ（Ｍｉｄｄｌｅ Ｓｃａｌｅ Ｉｎｔｅｇｒａｔｅｄ Ｃｉｒｃｕｉｔ）およびＳＳＩ（Ｓｍａｌｌ Ｓｃａｌｅ Ｉｎｔｅｇｒａｔｅｄ Ｃｉｒｃｕｉｔ）の論理素子で構成された汎用計算機を，マクロ変換手法を用いてＥＣＬ（Ｅｍｉｔｔｅｒ Ｃｏｕｐｌｅｄ Ｌｏｇｉｃ）のＬＳＩ化を行った実例を報告する．

このＬＳＩ化の目的は，短期間にＬＳＩを開発し，高密度化，コンパクト化を行い，性能向上を図ることにあった。以下にＬＳＩ化の手法および適用結果の概要について述べる．

2．ＬＳＩの仕様

約１００品種のＬＳＩを開発する必要があったため，ＬＳＩはマスタスライス方式のＥＣＬゲートアレーを採用した。基本回路を図１．に示す。

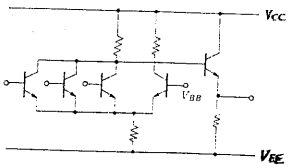

図1. ECLゲートアレー基本回路

ＬＳＩチップの集積度および入出力信号数は，ＭＳＩ／ＳＳＩの搭載されたカード１枚をほぼ１チップに対応させるために，それぞれ約１０００ゲート／チップおよび１１６ピン／チップを選んだ。ここで言うゲート数は実際のＥＣＬ回路数を意味しており，ワイヤードロジックを使用した場合はさらにファンクション数は増大する．ＬＳＩのパッケージは１２４ピンのリードレスパッケージを採用し，高密度実装を行うためＬＳＩを４個搭載したセラミックモジュール実装方式を採用した．

表１．にＥＣＬゲートアレーの概略仕様を，図２．にセラミックモジュール実装方式の外観を示す．

表1. ECLゲートアレー概略仕様

項　　目	仕　　様
基　本　回　路	ECL
ゲ　ー　ト　数／チ　ッ　プ	998 {内部ゲート 900 出力バッファゲート 98
入出力ピン数／チップ	124 {信号 116 V_{CC}・V_{EE} 8
電　源　電　圧	−2.8 V 単一電源
消　費　電　力／チ　ッ　プ	Typ. 2.0 W

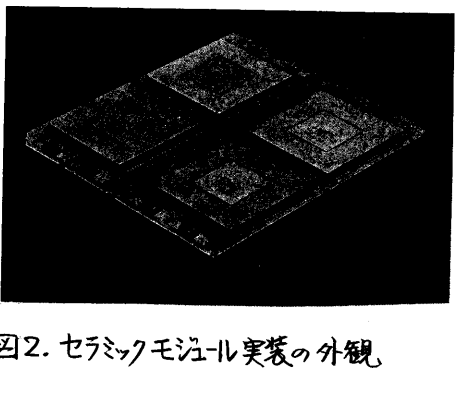

図2. セラミックモジュール実装の外観

3．LSI化設計

まずLSI論理の設計は，TTLのMSI／SSIの論理をゲート数および入出力信号数の制約のもとに分割を行い，TTL（＝NANDゲート）からECL（＝NORゲート）への変換を行う．分割指定は人手で行うが，TTLからECLへの変換はマクロ変換により自動的に行う．変換後の論理接続情報が元のTTL論理と合致していることを検証してLSIデータベースとして登録する．

ゲートアレーの配置・配線に際しては，速度上最適化するようにまたセラミックモジュールの入出力信号数の制約を考慮し，予め同一セラミックモジュール上に搭載するゲートアレーの組合わせを選択し，ゲートアレーの信号ピンの割付けも行った．つまりゲートアレーの大部分の信号ピンアサインを予め指定して，配置・配線設計を行った．このような方式では，論理分割が完了した段階で，LSIの実装設計（例えばセラミックモジュールのパターン設計）が開始できること，またカード上のセラミックモジュールの配置およびゲートアレーの配置まで事前に（ゲートアレーの設計が完了する以前に）把握できるというメリットがある．

その他設計段階で考慮した要因は，速度上のクリティカルパスの遅延時間の管理，クロックのタイミング管理，テスト性の向上などである．特に速度上のクリティカルパスは，MSI／SSIの論理とゲートアレーの論理とではかなり違ってくるので注意を要した．

図3．ECLゲートアレーの内部ゲートおよび入出力信号ピンの使用割合いについて，約80品種の結果を示す．

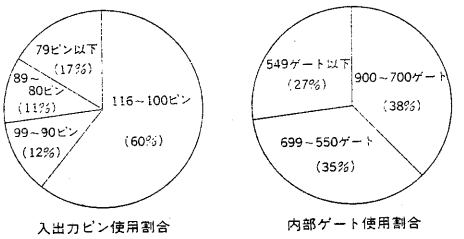

図3．ECLゲートアレーの内部ゲートおよび入出力信号ピンの使用割合い

4. 設計支援ツール

図4. に100品種のLSI製造で使用したCADシステム構成を示す。
CADシステムは論理再構成，論理図自動作成，論理シミュレーション／テストジェネレーション，LSIチップ自動レイアウトの4つのサブシステムで構成されている。

論理再構成

このCADシステムの中心となるサブシステムでNAND系のMSI，SSIで構成されている既存論理をOR，NOR系のLSIに自動的に再構成することが可能である。論理再構成の概略手順と機能は次のようである。

(1) 論理のLSIへの分割

元の論理回路から1つのLSIに再構成するバウンダリを指定し，LSIの切り出しを行う。

(2) ファンナウト調整

元の論理回路上で，信号のファンナウト数がLSIで許される制限を越えるものに対し，1入力ORゲートを挿入しファンナウト数を制限内におさめる。

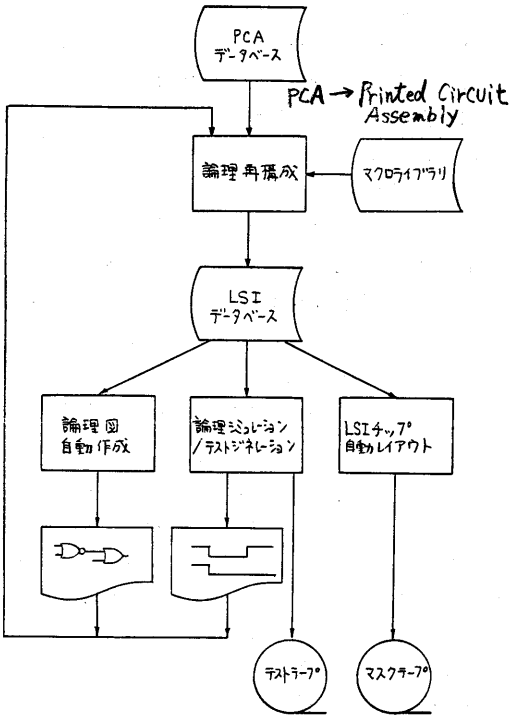

図4. CADシステム構成

(3) ライブラリによる置き換えとLSIマクロ展開

既存論理回路のマクロをLSIのマクロに置き換え，さらにマクロ展開を行う。

(4) LSIマクロ内不要素子・リダクション

一般に，LSIマクロ素子は汎用的な素子として用意されているため，実際の論理回路中で使われる時，そのすべての機能を使用するとはかぎらない。このため，これらの機能の一部を使用する場合，その不要部分の削除を行う。

(5) インバータ・リダクション

NAND系の論理をOR，NOR系の論理に置き換えることにより多くのインバータが発生するが，ファンナウトの制限を守って論理を変えることなくインバータを必要最小限にする。

(6) ゲートの圧縮

LSIマクロ内不要素子・リダクションによりゲートの入力端子が減り，その素子がインバータとなる場合や，素子が次段のゲートに吸収可能となる場合がある。また，LSIマクロ展開によりインバータが挿入されNOR-INV，OR-INVの構成になることがあり，これらはOR，NORと等価である。これらのゲート圧縮処理を行うことにより，素子数を必要最小限にする。

(7) 入出力バッファの挿入

最後にLSIの外部入出力信号に対してバッファ回路を挿入し，論理再構成操作を終了する。

論理図自動作成

論理再構成プログラムによりLSI化された論理回路を論理図自動作成プログラムにより作画する。作画されたLSIの一部を図5.に示す。

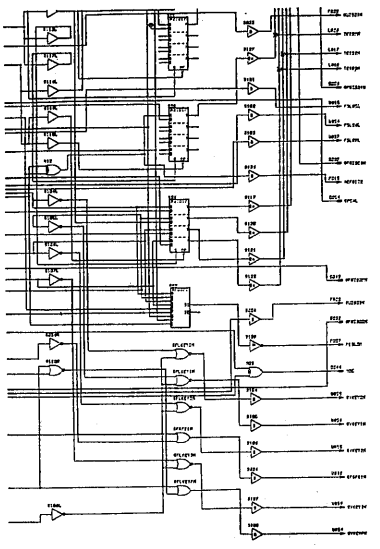

図5. 論理図自動作成結果

論理シミュレーション／テストジネレーション

LSI化された論理のシミュレーションとしてファンナウトを考慮した仮想遅延，レイアウト結果の実装配線長を考慮した実装遅延シミュレーション，タイミング検証を行うことによりクリティカルパス等のチェックを行う。さらに，テストジネレーションを行いLSIのテストテープを作成する。

LSIチップ自動レイアウト

図6.にLSIのチップ構造を示す。自動レイアウトプログラムは配置処理，配線処理とから成っており各処理の概略機能は次のようである。

(1) 配置処理

総配線長の最小化を目的として，ブロック配置，ゲート配置の階層的手法を用いて配置を行う。

(2) 配線処理

まずチャネル割当て処理で各信号を各チャネルに割り当て，次にチャネル内の各信号を各トラックに割当てる階層的手法を用いて配線を行う。

図7.にLSIの自動レイアウトを行った結果のプロット図を示す。

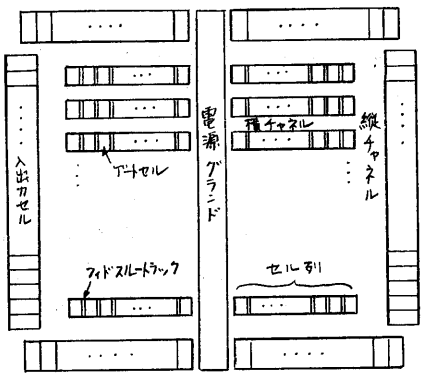

図6. LSIのチップ構造

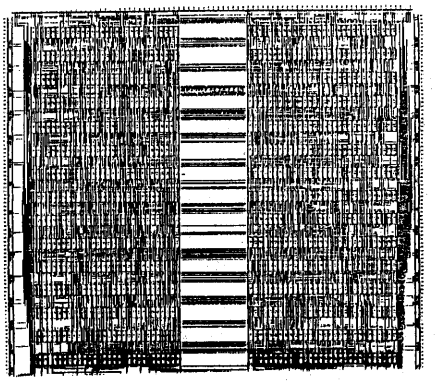

図7 LSIのレイアウト結果

5. LSIのデバッグ

デバッグは下記の3段階で行った。

（1）ゲートアレー単体でのデバッグ

（2）MSI／SSIマシンにゲートアレーを実装してデバッグ

（3）LSI化マシンでのデバッグ

（1）はLSIテストプログラムによるテストが主体であり，（2）および（3）はマシンでのデバッグであり，マイクロ診断プログラムやハードウェア機能診断プログラムによりデバッグを行った。多品種のLSIが全品種揃うまでにはかなりの期間がかかる。実際には数品種ずつが順次出来上ってくるので，シリアルにデバッグする方法として（2）の段階のデバッグ方法を採用した。MSI／SSIのカードにほぼ対応してLSI化したので，この方法が採用できた。

約100品種のデバッグの結果，約20％に不具合いがあったが，この殆んどは（2）の段階で発見できた。またこの不具合いの原因の大部分は，人手介入によるミスである。基本的にはマクロ自動変換によりLSI化設計を行ったものの，性能向上や機能追加そらにはテスト性向上のための論理追加などで人手介入を行ったが，この時の論理シミュレーションに不備があったことによる。逆に自動変換を行った部分での不具合いは皆無であった。

6. おわりに

汎用計算機の基本処理装置，主記憶制御装置，チャネル制御装置の大部分の論理をLSI化した結果，装置の大きさは約4分の1に，消費電力は約2分の1に小さくできた。今回のLSI化では，既に検証された論理をもとにLSI化したが，やはり設計段階のシミュレーションの徹底とLSIのデバッグが重要である。特に同一チップ内に2ヶ所以上の論理ミスがある場合，1ヶ所の不具合いのため他の論理ミスが発見できないことがある。このような場合は同じLSIを2度以上作り直すことになる。つまり不具合い点を切り離して他の論理のデバッグができるように設計しておくことが非常に重要であると思われる。