VLSI設計システム

杉山吉　　唐津修　　須藤常太

（日本電信電話公社　武蔵野電気通信研究所）

1．はじめに

近年、LSIプロセス技術の進歩に伴って、数万～数十万素子／チップのLSIの実現が日程に上って居り、改めてこの設計技術の重要性がクローズアップされている　（1）～（3）。我々は、LSI設計自動化システムの開発を進めて来たが、このほど一通りの完成をみ、運用を開始した。本稿ではこのVLSI設計システムの概要について報告する。

まず最初に、VLSI設計時の問題点をまとめてみよう。
（1）高集積化の進展に伴って、システム的要因からデバイス的要因までが一つのLSIチップの中で複雑に絡みあって来る事である。このため分野の異なる様々な境界条件を考慮にいれながら設計を行なう必要が出てくる。

（2）大規模化に伴って、当然のことながら設計データの絶対量が増加して来る。このため、量の増大に従いデータの作成、検証の困難度が加速度的に増大する。
（3）論理的要因と物理的要因を総合的に捉えて最適化する必要が生じる。これは従来ICをプリント板に並べて論理を設計していたのとは大きな違いである。
（4）設計データの完成度が極めて厳しく要求される事である。プリント板の設計においては、多少の設計ミスも製造段階のボード配線で救済することが出来たが、LSI上でのマスク・パタンのミスは致命的である。
（5）LSI製造技術が日進月歩である現状を考慮にいれる必要がある。使用プロセスや素子の基本性能が改良される度に設計全体をやりなおさなければならないのでは賽の河原の石積みになる。

このような問題点を解決するためにつぎの5点を念頭に置きVLSI設計システムを開発した。
（1）大規模化するデータを効率良く設計処理するために無制限の階層があつかえる構造化設計手法をサポートすること。
（2）システム全体で共通に使用出来る共通設計記述言語を設定し、設計段階間でデータの共用を計ること。
（3）設計データ・ベースを開発し一元的にデータの管理を行なう一方、設計資産として蓄積再利用が計かれるよう工夫すること。

（4）人手がデータに直接関与して、ミスを誘発する事を避ける為、ルーチン化した各種処理のプログラム化を進めること。
（5）設計システムによる設計結果が満足の行くものであるかどうか、設計者が容易に判断出来るよう使いやすいマン・マシン・インタフェースを設定し、人間の経験が十分生かせるインタラクティブな切り口を設定すること。

2．システム構成

図1は、本VLSI設計システムの全体構成図を示す。システムで共通に使用する記述言語は、新たに開発した階層仕様記述言語HSL（H ierarchical Specification L anguage）である。このHSLで記述された設計データはシステムによって様々な処理を経たのち、然るべきマスク・パタンや試験の為のテスト・パタン等の形で出力される。システム中央に設計データ・ベースを配置してあり、HSLで記述された設計データはここに蓄積管理される。データ・ベースの周辺には、HSL記述のコンパイラ、逆コンパイラ、階層展開プログラム、やワイアード論理生成プログラム等のHSL処理ソフトウェア群、論理シミュレータ、タイミング・シミュレータ、回路シミュレータ等の各種シミュレータ群、其の他テスト・パタン発生プログラムやインタフェース・プロ

グラム等が配置されている。各種のシミュレータ等の支援ソフトウェアで検証された設計データは、自動配置配線プログラムによって自動レイアウトされた後マスク・パタン・ジェネレータ等のデータとして出力される。配置配線の結果が妥当かどうかの判断は、インタラクティブ・ターミナルやプロット図での判定のほか、配線結果を電気特性にフィード・バックさせて詳しいシミュレーションを行なう路が用意されている。このように本ＶＬＳＩ設計システムにおいては、一度ＨＳＬによってデータを入力すれば以後の処理は全てシステムによって自動的に行なわれる。人間は要所要所に於ける判定を行なって、ＧＯ／ＮＯＧＯの指示や自動設計手順の指示をだしさえすれば良い。本システムにおいては設計途中のデータを手作業で修正したり、引き残した配線を人手で加えたりする事は一切ない。

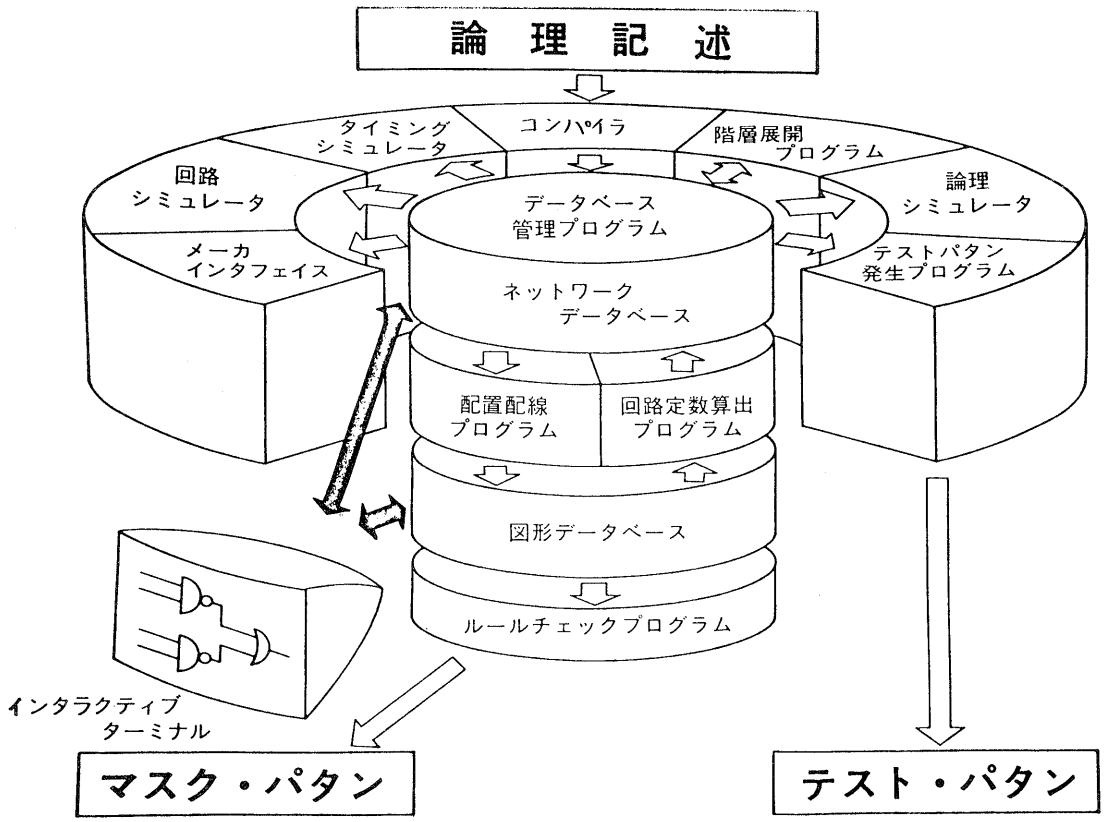

図１　ＶＬＳＩ自動設計システムの構成

３．階層仕様記述言語ＨＳＬ

ＨＳＬの記述は（１）共通管理情報の記述、及び（２）モジュールの記述からなっている。共通管理情報部ではＬＳＩのチップ名、版数、設計年月日等の管理項目を記述する。モジュールはＬＳＩ設計を行なう際の記述の単位となるもので、以下のようなデータを記述する。

（ｉ）　モジュール管理情報：　モジュールの名称、使用目的、階層名。
（ii）　外部ピン属性　　：　モジュールの外部ピン（外枠に並んだ端子）の名称とその属性。
（ iii）座標情報　　：　モジュールの図面や配置の情報。
（iv）　接続情報　　：　モジュール内の接続情報、引用している下位のモジュー

ル名など。

(v) 遅延情報 : ネットワークのもつ配線遅延や素子の遅延情報

(vi) 機能情報 : ネットワークを構成する基本素子の論理機能や回路素子の種類。

以上の様な記述を必要なだけ繰り返し行なう事により階層的にＬＳＩ全体の設計を進める。

図2、3に、ＨＳＬのコーディング例を示す。記述はキーワード方式の８０カラム・カード・イメージでフリー・フォーマット形式を採用している。この外に、論理図によるグラフィック入力の方法も設定されている。

共通管理情報部	IDENT :CPU ; VERSION :VR01.00 ; DATE :80/08/03; AUTHOR :DENDEN TARO ; PROJECT :VLSI03 ; COLLECTION :VLSI; COMMENT : MICRO RPOCESSOR ;
モジュール管理情報部	NAME :BUSSW ; PURPOSE :LOGSIM,ROUTER,CKTANAL ; PROCESS :EDMOS2U; LEVEL :BLOCK ;
外部ピン属性	EXT :A,B,E,O,DATABUS<0:31>,VDD,VSS; DEFAULT :1,1,1,A ; INPUTS :.A,.B,.E; OUTPUTS :.O; POWERS :.VDD,.VSS ; BUS :.DATABUS<0:31> ;
座標情報	TYPES :INV,TRINAND ; INV :G1,I2,I3 ; TRINAND :G2,G3,SW1<0:31>,SW2<0:31> ; MPOSITION :.A(1,1),.B(1,30),.E(1,50), .O(100,50),G1(30,10),G2(30,20) ;
接続情報	NETA =FROM(.A) TO(G2.2) ; NETO =FROM(G2.3,G3.3) TO(.O) ;
遅延情報	MDELAYS :DEL035 :NETA ,NETO ; DELAYS :DEL035,35,32,37,35,32,37 ;
機能情報	FUNCTION:TRINAND,2,1,DTRINAND; DELAYS :DTRINAND, 6, 5, 7, 4, 3, 6 ; ELEMENTS :PMS3 ,15,2 ; PARAMETER:BULK=15,VT=.7,BETA=2.7E-5, MOB=99,CLM=1,GZI=5.1E-5, GAMMA=.5,ALS=.436,GMD=.01,PHI=.32 ;

図2　ＨＳＬの記述例

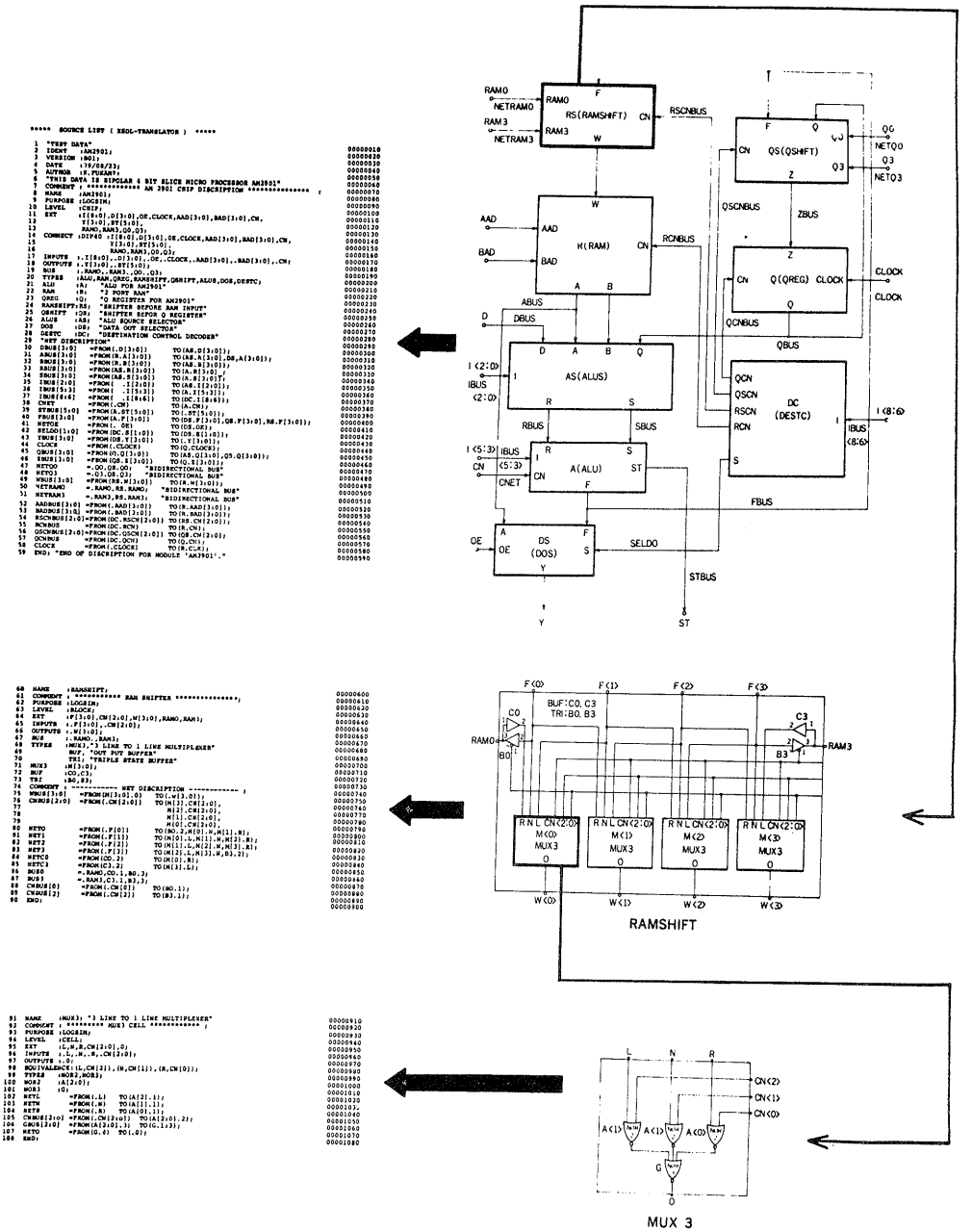

図3　HSLによる階層記述の例

４．自動配置配線

ＬＳＩにおいて配置配線の設計品質が、出来上がりレイアウトのパッキング密度にあることは論を待たない。本システムでは次の４つの特徴をもつ自動配置配線ソフトウェア群により高密度化の課題に挑戦した。

（１）　設計階層に合せて各段階でレイアウト品質を判定出来る様にしたこと。

（２）　プログラムの構成を機能モジュール単位で分割し、色々な配置配線アルゴリズムを自由に組み合わせたり、実行順序を入れかえるなどの自由度を持たせたこと。

（３）　設計者は知恵を用いるのみで直接には手を下さないシステムとすること。

（４）　グラフィック・インタフェースを駆使してシステムの出力の良否を設計者が容易に判定出来るようにすること。

次ぎに具体的な手順を説明する。ＨＳＬで記述されたデータは論理設計上の階層構造にもとずいて準備されている。レイアウト時にはこの階層構造をなるべく利用する所から出発する。例えば十ゲート前後の基本論理を集めたセル、セルを数十集めたブロック、ブロックを数十集めたチップと言った具合である。まず詳細なレイアウトに先だって大まかなブロック配置を行なう。ここではブロックの大きさ、形状、ブロック間配線の通しかた、ブロック端子の凡その位置などを決める。この作業によってＬＳＩチップの外観が決まる。論理設計時のブロック構成がレイアウト用には不適当であった場合は、ブロックの分割併合などのユーティリティを使ってブロックの再構成を行なう。次ぎに、セルの配置配線を自動で行なう。このセルを用いてブロック内の配置、配線、ブロック間の配線処理を進め、チップ周辺部を自動配線して完成する。前に述べたように、各レイアウトのステップがすすむたびに設計者による可否の判定を行なう。このとき設計品質が十分高くないと判定されれば、場合によってはブロックの再分割にまで溯ってやりなおす。しかし全ての処理が自動で行われるので一回の試行に要するターン・アラウンド時間は極めて短い。レイアウトの品質がＯＫとなれば設計データはマスク・パタンの形に変換されて出力される。

セル自動配置配線の結果の例を図４に示す。論理式またはＨＳＬ入力して、セル・パタンを自動設計する。ここでは、極力拡散配線で済むよう配置を最適化し、ＶＤＤとＶＳＳ線間にセル内配線を収めるようにしている。１品種の処理時間は１秒前後である。

約２０００ゲートのブロック自動配置配線結果を図５に示す。このブロックは、８つのサブ・ブロックから構成されている。各サブ・ブロックの形状はゲート密度が最大になるようにアスペクト比を最適化してある。端子位置はサブ・ブロック間の配線が最短になるよう設定されている。配線はアルミ２層にポリシリコン１層の３層配線である。

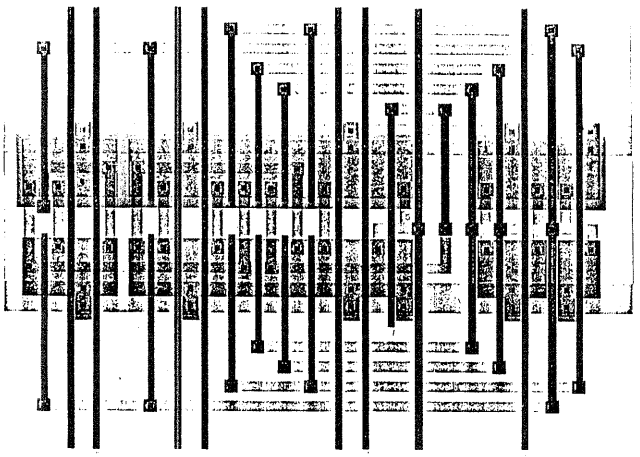

図４　　自動設計されたＣＭＯＳセル

図5　自動設計されたCMOSブロック

5．入力出機能

ＨＳＬの入力は先述の通り、カード・イメージ又は論理図によるグラフィック入力の２通りを用意してある。
設計者はＨＳＬでコーディングするか又はグラフィック・ターミナルを見ながら図面を入力するかによって、データの登録を行う。既にデータベースに格納したデータもグラフィック・ターミナルに呼び出して確認、修正などが可能である。自動配置配線した結果のレイアウト品質を判定するためには、グラフィック・ディスプレイ、プロッタ等に随時出力して見ることが出来る。

6．本ＤＡシステムの効果

表１はＨＳＬの使用による初期データ作成工数の削減の様子を示す。本表は図面を基に入力カードを作成してデータベースに登録する場合の例である。比較のために同じ図面から各シミュレータ用の記述を行った工数を記載してある。ＨＳＬ記述では記述能力の高さの為に、初期データ入力の場合でも、大幅に工数が削減されている。一度データをデータベースに登録すれば以後は各種ＤＡプログラムに直接入力されるので、例えば論理シミュレータ用、回路シミュレータ用、配置配線プログラム用と一々データを作りなおす必要が無く、データの手作業変換といった工数が全て省略出来る事は今までの説明の中で繰り返し指摘した事である。

最後に本ＶＬＳＩ設計システムに依って設計されたＬＳＩの品質について例を表２に示す。図５にしめした２０００ゲートの論理について、比較の為実際に人手と自動でレイアウトしたものの比較である。設計工数は２０００ゲートの配置配線を完成するのに要する工数である。人手、自動とも結果が気に入らなければ何回でも再試行出来る。しかし、現実の問題として人手レイアウトではその工数とターンアラウンドから考えて何回も繰り返し試行することは不可能に近い。一方自動の場合約一日強で一試行出来るので気に入るまでやりなおすことが可能である。この例では７回の試行を行っている。トータルの設計工数は自動が１０人日、人手が２２５人日で約１：２２である。ここで注目したいのは人手より自動のほうが設計品質が高い、即ち高密度である点である。従来自動レイアウトは人手に比べて工数は減っても品質は常に劣るものとされてきたが、ここにしめすデータは明らかに違っている。これは、大規模ＬＳＩの場合、今述べたようにターンアラウンド時間に決定的な差があるため人手では再試行による最適化が事実上不可能だということの為である。本システムによる経験では自動レイアウトに軍配の上る回路規模は約１０００ゲートをこえる領域である。

表１　　入力データ登録工数

	HSL記述	カードパンチ	ファイル登録・修正	合　計	備　考
500ゲート	16人時	8人時	4人時	28人時	論理図
4000ゲート	140人時	70人時	10人時	220人時	↓ ファイル登録

	回路シミュレータ用記述	カードパンチ	ファイル登録・修正	合　計	備　考
500ゲート	24人時	12人時	8人時	44人時	回路図→ファイル

	論理シミュレータ用記述	カードパンチ	ファイル登録・修正	合　計	備　考
4000ゲート	280人時	140人時	40人時	460人時	論理図→ファイル

表2　　自動設計と人手設計の比較

	集積度	設計工数	配線エリア／チップ面積	設計に必要なDAリソース
自動設計	400 ゲート/mm²	10人日	62％	COSMIC　4分 PLASMA　15分
人手設計	320 ゲート/mm²	225人日	74％	デザインルールチェック　300分 接続チェック（現状は人手）

人日……2,000ゲートのCMOSブロック1回の試行に要する工数
時間……CPUタイム

7．まとめ

武蔵野通研に於いて稼動を始めたＶＬＳＩ自動設計システムの概要と、効果を述べた。現状では、入力データ登録工数が従来の１／４～１／１０、レイアウト設計工数が１／１０～１／１００であるが、データベースへのデータの蓄積が進み、再利用が活発化する時点では、一層の工数削減が期待される。各プログラムの詳細に就いては、別の機会に報告したい。

8．謝辞

日頃ご指導いただく武蔵野通研、集積回路研究部渡辺部長並びに、関係各位に、感謝する。

9．参考文献

設計法

（1）須藤：ＶＬＳＩ設計法、情報処理学会電子装置設計技術研究会予稿１－１

（2）VanCleemput，W.，M.，et al.”Initial design considerations for a hierachical IC Design System”，Proc．11th Asilomar Ccof．Circuits，Systems and Computers （1977）

（3）Lattin，B.”VLSI Design Methodology－The Problem of the 80´s formicroprocessor Design”，Proc．16th DA CON F.，P548

設計言語

(4) VanCleemput , W. M. " A Hierarchical Language for the Structural Description of Digital Systems, " Proc . 14th Design Automation Conf . , 378−385.

(5) VanCleemput , W. M. " Computer Hardware Description Languag es and their Applications " Proc . 16th DA Conf . , P554

レイアウト設計

(6) Uehara , T. , et al. " Optimal Layout of CMOS Functional Arrays ", 16th DA Conf . , P287

(7) Preas, B. , et al. " Methods for hierachical automatic Layout of LSI circuit masks", P roc. 15th DA Conf . , P206

(8) Sato , K. , et al. " MIRAGE−A Simple −Model Routing Program for the Hierarchical Layout Design of IC Masks" 16th DA Conf . P297