

シンボリック・レイアウトシステム SYLAによるセル設計

植松 幸雄 川村 弘哉 小池 豊
沖電気工業(株) 超LSI開発センター

CMOS論理LSIのセル設計を支援する、シンボリックレイアウトシステムSYLAについて報告する。SYLAは、デザインルールやプロセス情報を格納したテクノロジーデータベースと、トランジスタのゲート幅や電源線幅等を指定するためのパラメータファイルを持つ。そのため、デザインルールに独立なセル設計を可能とし、かつ同一のシンボル図から、異なる回路性能を持つセルレイアウトの生成を行うことができる。また、セル設計手法としてシンボリックレイアウト手法のみでなく、セルの繰り返し表現に優れた言語記述手法を採用した。このことにより、データパス部等の繰り返し性の高い大規模なセルについての設計を容易にした。

本システムを、ポリセルアレイ型セルの設計に適用したところ、人手設計とほぼ同等のセル面積を有するレイアウト生成結果を得た。

SYLA: Symbolic Layout System for VLSI Cell Design

Yukio UEMATSU, Hiroya KAWAMURA, and Yutaka KOIKE
VLSI Research & Development Center, Oki Electric Industry Co., Ltd.
550-1, Higashiasakawa-cho, Hachioji-shi, Tokyo 193, Japan

This paper describes the symbolic layout system named SYLA that was developed to support CMOS cell design. This system supports the technology independent cell design according to the technology data base. SYLA also enables to generate various cell layout from one symbolic diagram by changing transistor size and/or width of wires according to the parameter file.

In addition to symbolic representation, SYLA may also provide language expression for layout. In hierarchical design, the upper complex cells like data path blocks may be represented easily by the language expression to iterate the lower cells. We describe the goal of SYLA and the method of generating cell layout by using both symbolic and language expression. We also show application results of some circuits.

1. はじめに

近年、ASIC化の傾向はブームとも言える状況を呈しており、そのために各種CADツールの開発・改良が行われている。しかし、それらの大半は開発納期の短縮を最優先としたものであり、出来上がったICの性能やチップ面積の観点からは、必ずしも充分なものとは言えない。そのため、高速・高集積のLSIや量産用のLSIの一部には、依然として従来より行ってきた人手によるレイアウト設計手法を採らざるを得ないものも多い。

しかし、それらの製品群に関してもユーザー側からの開発納期短縮要求や開発コード数の増加により、従来とは異なった観点からのCADツールの開発が必要となっている。そのようなCADツールとは即ち、人手設計に近い性能とチップ面積を実現することを第一優先とし、その範囲内で開発納期の短縮を図ったものである。

当社では、上記の観点から各種CADツールを開発中である。本稿では、CMOS論理LSIのセル設計専用開発した、シンボリックレイアウトシステムSYLAについて報告する。

2. SYLA開発のねらい

2-1. 専用セルの必要性

セルベースのLSI設計手法の中でCAD化が比較的進んだ手法として、スタンダードセル方式と呼ばれる、標準的な機能を持ったセルを多数組み合わせ設計する手法がある。しかしスタンダードセル方式では、セルの機能に汎用性を持たせ、再利用を可能とするため、セルの中には冗長なトランジスタが存在している場合が多い。また、配置・配線の最適化の自由度も低いため、性能やチップ面積は人手設計に比べ、かなり劣る。従って、性能やチップ面積を重視した場合には、使用状況に合った最少のトランジスタ構成で、しかも、数百トランジスタ以上の規模まで配置・配線の最適化が行われた多数の専用セルが必要となる。SYLAの開発に当たっては、このような専用セルを誤りなく短期間で作成可能とする事をめざした。

2-2. デザインルールに独立な手法の導入

高速・高集積のLSIではチップサイズの制限から最先端の製造プロセスを使用する機会が多いが、先端のプロセスでは、その安定化や製

造装置の相違によりパターンデザインルールが変更される場合がある。そのためには、デザインルールから独立なレイアウト設計手法を導入する必要がある。一方、量産用のLSIにおいても、その製造が長期間に渡るため、コスト削減を目的として途中でより微細な製造プロセスに変更する場合があります。デザインルールから独立な設計手法をとるメリットは大きい。このような観点から、SYLAでは2-1節で述べた専用セル設計にたいする要求を満たし、なおかつデザインルールと独立にセル設計を行えるような手法を採用した。

2-3. システムの統合化

セルの設計工程においては、単にレイアウトパターンの作成のみでなく、各種の検証作業も必要である。多数の専用セルを短期間で作成するためには、それらの作業をサポートするCADツール群が効率良く結合し、設計工程におけるフィードバックによる繰り返し、短いTATで実現できる事が必要である。SYLAでは、ワークステーション上に統合化された既存のレイアウト設計・検証ツールにシンボリックレイアウト機能を組み込むことにより、全体機能を実現した。

3. デザインルールに独立なセル生成

本章では、デザインルールに独立なセル生成手法として、従来より提案されている各種の設計手法について、2章で述べた目標機能の実現性をもとに比較を行った後、SYLAで採用した手法について述べる。

デザインルールと独立なセル生成手法として、代表的な4種類の手法を図1に示す。4種類の手法とは、①. 回路接続情報と入出力端子位置、レイヤー情報からセル内のレイアウト構成要素のトポロジーを自動的に決定し、レイアウトデータを生成する自動生成手法^[1]、②. セルのレイアウトを、あらかじめルールに従って図形表現しておくスケーリング手法、③. シンボル化されたレイアウト構成要素でセル内のトポロジーを入力するシンボリックレイアウト手法^[2]、④. 言語を用いてレイアウトと接続情報を記述する言語記述手法^[3]である。

これらの内、自動生成手法は、任意の端子位置が設定可能であり、セル内部のレイアウト構

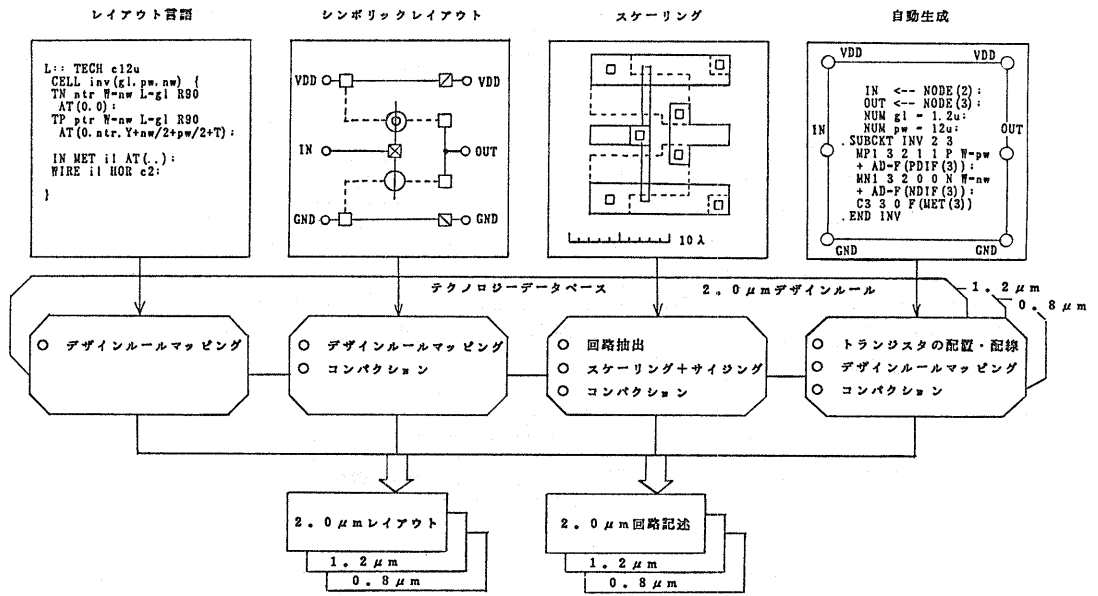


図1 デザインルールと独立なセル生成のための各種設計手法

造を記述する必要が全く無い、等の利点を持っている。反面、他手法に比べ技術的に最も実現困難であり、現状では人手設計と比べてセル面積の大幅な増大は避けられないと考えられる。また、スケーリング手法は、各プロセス毎のデザインルールが比較的同一規則にのっとっている場合は、セル面積は最も小さくなる。しかし、レイアウトのコンパクションが困難であり、ゲート幅等の再指定ができないという欠点をもつ。シンボリックレイアウト手法は、セルの構造をある程度固定化してしまうという欠点を持っているが、レイアウト構造の入力の際、デザインルールを意識しなくても良いため、入力の工数が少なく済む。また、デザインルール変更時にゲート幅等の再指定が可能である。言語記述手法は、下位のセルを繰り返しによって配置する場合や、配置するセルを特定の条件に従って選択する場合に対して最も適しており、デザインルールやトランジスタ間の接続も記述可能である。しかし、デザインルールと独立にレイアウトを言語で記述するには多大な工数を必要とする。

以上のような手法の内、SYLAでは、シンボリックレイアウト手法と言語記述手法を併用する事にした。即ち、レイアウト構造がある程

度固定化されたセルに対してはシンボリックレイアウト手法を適用し、デザインルールと独立なセルを妥当な面積、少ない工数で生成する。また、それらのセルを上位セルで繰り返し配置する場合には言語記述手法を適用し、繰り返し数や配置条件の変更等に対して柔軟に対処する。

4. SYLAによるセル設計

本章では、シンボリックレイアウト手法と言語記述手法を併用したセル設計について、具体例を挙げながら説明する。

4-1. システム構成

図2にSYLAのシステム構成、及び既存ツールとの関係を示す。以下、システム構成を、シンボリックレイアウト設計のフローに従って説明する。

①. グラフィックエディタ上でシンボル図を対話的に作成する。その際、トランジスタのゲート長、ゲート幅、及び配線幅等にパラメータ名を付与し、シンボル図上の所定の位置にテキストとして挿入する。作成したシンボル図は、接続情報とともにシンボルファイルに出力される。一方シンボル図上に指定した各パラメータに対応する実寸法は、テキストエディタを用いてパラメータファイルに記述する。

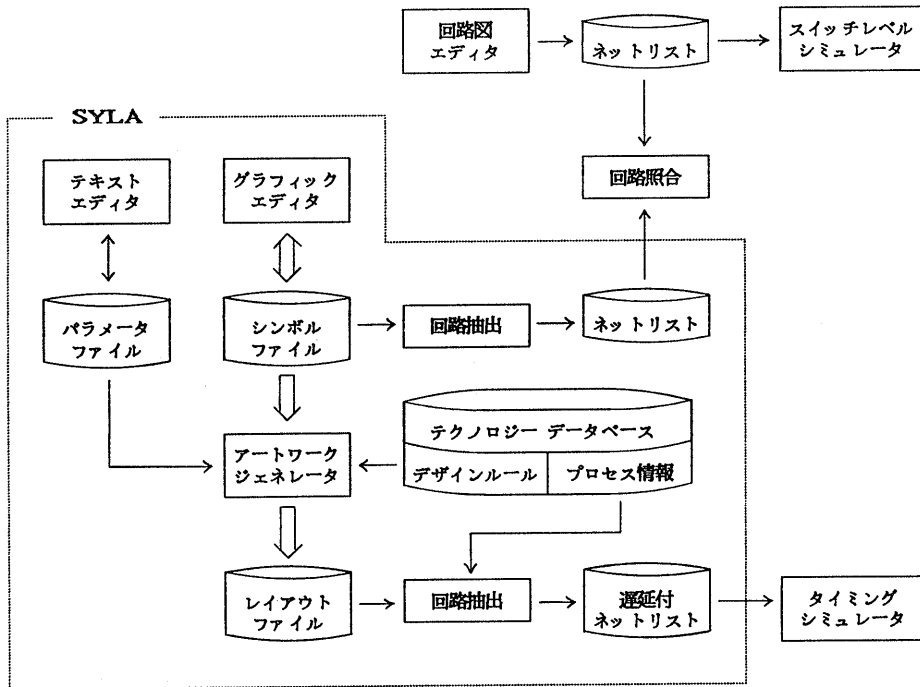


図2 システム構成

②. 回路抽出プログラムは、シンボルファイルに含まれるシンボル間の接続情報からトランジスタレベルのネットリストを抽出する。このネットリストは、あらかじめ回路図入力された検証済のネットリストと比較照合される。

③. アートワークジェネレータは、シンボルファイルとパラメータファイルからレイアウトデータを生成する。その際、テクノロジーデータベースに定義されたデザインルールに従って仮想グリッド方式によるコンパクション処理を行う。階層化設計されたシンボル図は、コンパクション処理を行う前に展開される。

④. テクノロジーデータベースには、デザインルールの他にプロセス情報が含まれており、これを参照することにより、レイアウトデータから配線容量を含んだネットリストを抽出する。このネットリストはタイミングシミュレータに入力され、生成レイアウトのタイミング検証が行われる。

4-2. 使用シンボル

図3にSYLAで使用するシンボルを示す。トランジスタ、コンタクト、及び外部端子の各

シンボルは、特定層の配線が接続できるターミナルを1個ないしは複数個持っている。配線シンボルはレイアウト上における配線図形に変換されると同時に、シンボル間の接続情報を表現している。レクタングルは、ウェル等の連続領域を生成するために使用するが、後述するようにゲート長・ゲート幅、配線幅等の指定パラメータの有効範囲を指示する役割も果たす。

シンボリック設計は、まずトランジスタ、コンタクト、外部端子の各シンボルを、相対位置

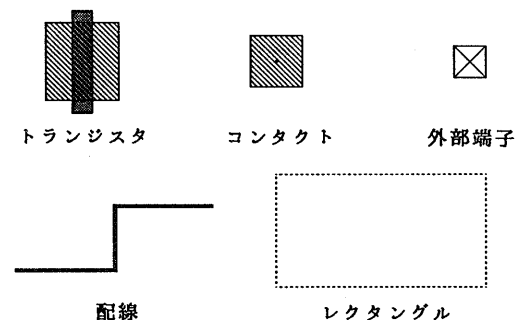


図3 使用シンボル

関係を考慮しながら仮想的な格子点上に配置した後、シンボルのターミナル間を、経路を指定しながら配線シンボルで接続する。シンボル図を階層化設計する場合は作成済の下位セルをマクロシンボルとみなし、他のシンボルと同様に配置、配線する。

4-3. セルのパラメータ表現

シンボル図から生成するセルレイアウトは、性能と面積の両立を図りながら最適化されていなければならない。そのためには、セルを構成するトランジスタのディメンジョンや配線幅を、所望の回路性能を満たした上で個別に最適化する必要がある。また、外部端子の位置は、隣接セルの端子位置に合わせて変更可能である事が望ましい。これらの条件は、セル設計の初期段階では一意に決定できないため、SYLAではトランジスタのゲート長・ゲート幅、配線幅、及び外部端子の位置をパラメータとして指定する方法を採用した。シンボル図上にはパラメータ名のみをテキストとして挿入し、個々のパラメータに対応する実寸法は、シンボル図とは別に用意したパラメータファイルに記述する。この方法により、同一のシンボル図から異なる性能・面積のセルレイアウトを、パラメータファイルの内容を変更するだけで容易に生成できる。

4-3-1. トランジスタディメンジョンの指定

シンボル図上に配置したトランジスタシンボルは、特に指定の無い場合、テクノロジーデータベースにデフォルト値として定義された最小ゲート長・ゲート幅のトランジスタに変換される。トランジスタのディメンジョンを変更する場合は、図4に示すようにゲート長・ゲート幅の値をパラメータ化して指定する。指定方法は、まず対象となるトランジスタシンボルと、それに接続するシンボル（コンタクト、拡散配線）を囲むように拡散層のレクタングルを配置する。次にレクタングルの左下格子点に、指定したいゲート長・ゲート幅を示すパラメータテキストを挿入する。オートワークジェネレータは、コンパクション処理を行う前に、レクタングルに囲まれたトランジスタのディメンジョンを、パラメータファイルに記述された値に置き換える。また同時に、レクタングルに囲まれた各シンボルの、拡散層の外形が内接するような拡散層の

レクタングル図形を生成し、レイアウト上に生ずる拡散層のノッチを埋める。このように、トランジスタのディメンジョンをパラメータ化してシンボル図と別管理することにより、シンボル図を修正することなく、生成セルの性能を変更することができる。

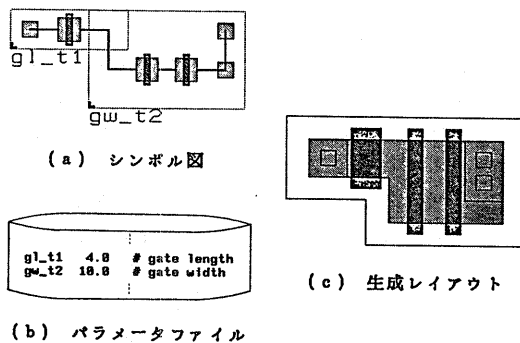


図4 トランジスタディメンジョンの指定

4-3-2. 配線幅の指定

シンボルの端子間を結ぶ配線シンボルは、テクノロジーデータベースにデフォルト値として定義された、最小の幅を持つ配線図形に変換される。電源配線のように幅の広い配線を生成したい場合は、トランジスタディメンジョンの場合と同様に、配線幅の値をパラメータ化してシンボル図上にテキスト指定する。図5に配線幅の指定例を示す。指定方法は、まず対象となる配線シンボルと、その配線が接続されているシンボルを囲むように、対象配線層のレクタングルを配置する。次にレクタングルの左下格子点に、指定したい配線幅を示すパラメータテキストを挿入する。オートワークジェネレータは、レクタングルに囲まれた配線シンボルに対応する配線図形の幅を、パラメータファイルに記述

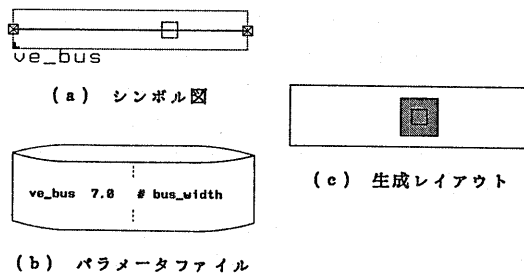


図5 配線幅の指定

された値に置き換える。その際、配線シンボルで接続される両端のシンボルが、一方でもレクタングルに含まれていない場合は、パラメータの指定対象から除外される。なお、パラメータを指定せずにレクタングルのみを配置した場合、配線図形の幅は両端シンボルのレイアウトを構成する配線層図形の幅に変更される。

4-3-3. 外部端子位置の指定

シンボル図をレイアウトデータに変換する際、アートワークジェネレータは生成セルの面積ができるだけ小さくなるようにコンパクション処理を行う。一方、複数のセルを隣接配置（バッティング）して上位セルを組み立てる場合、セルを部分的にストレッチして外部端子の位置を合わせる必要がある。このような要求に対処するため、SYLAでは外部端子の置かれた仮想格子の間隔をパラメータ化し、シンボル図上に指定可能とした。図6に指定例を示す。指定方法は、まず間隔を指定したい仮想格子の区間を示すためのシンボルを配置する。次に区間の上側を示すシンボルと同一の格子点に、指定したい格子間隔に対応するパラメータテキストを挿入する。アートワークジェネレータは、コンパクション処理の際、指定された格子の間隔をパラメータファイルに記述された値に設定する。格子間隔を指定値に設定した結果デザインルール違反が生ずる場合は、メッセージを出力し、デザインルールを満たすようにレイアウトを生成する。

4-3-4. パラメータの相互関係の記述

4-3-1項から4-3-3項で説明した各種のパラメータは、個々のパラメータの値が相互に依存し合っている場合が多く、それぞれ単独で値が決定できるわけではない。従って、これまで説明したように、すべてのパラメータの値を直接パラメータファイルに記述する方法では、あるパラメータの値を変更する際、影響を及ぼす他のパラメータの値もすべて変更しなければならない。このような問題を解決するため、パラメータファイルには、パラメータ間の相互関係の記述も可能としている。図7に記述例を示す。相互関係を記述するためには、まず基準となるパラメータを宣言し、次に他のパラメータと基準パラメータとの関係を定義する。定義済みのパラメータは、基準パラメータと同様に、他のパラメータを定義する際に参照できる。

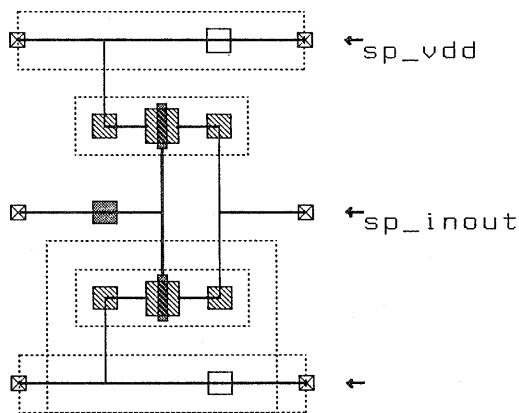


図6 外部端子位置の指定

```
#####
# Example of Parameter file
#####

BASE      gw_tn1    15.0
BASE      gl_tn1    2.0
BASE      ve_bus    5.0

DEFINE    gw_tp1    gw_tn1 * 1.2
DEFINE    gl_tp1    gl_tn1
DEFINE    gw_tn2    gw_tn1 / 2
DEFINE    gl_tn2    gl_tn1 * 2
DEFINE    gw_tp2    gw_tp1 / 2
DEFINE    gl_tp2    gl_tp1 * 2
```

図7 パラメータの相互関係の記述

4-4. 言語による上位セルの記述

複数のセルを組み合わせて高密度な上位セルを生成するためには、セル間に配線領域を設けずに、セルの外部端子位置を合せて、できるかぎり隣接配置する事が望ましい。その際、単一のセルが規則的な繰り返しで配置される場合には、上位セルを言語記述する手法が便利であり、繰り返し数も容易に変更できる。また、ある条件に従って配置するセルを選択する場合にも、判断命令等によって対処可能である。そこでSYLAでは、シンボリックレイアウト手法で生成した下位セルのレイアウトを、上位セルに隣接配置する部分に、言語記述手法を適用した。図8に記述例を示す。記述手順としては、まず作成するセルの名前を定義し、下位セルの繰り返し数等、可変としたい数値を定義したセルの引数として宣言する。次に、アートワークジェネレータを起動する関数によって、必要な下位セ

```

CELL adder (nbit)
{
  NUM    i = 0;
  CALL_SYMBOL "adder_one" CELL adder_one_bit
    (gw_tn=7, ve_bus=5, sp_term_a=26);
  WHILE(++i <= nbit) {
    IF(ODD(i)) {
      INST adder_one_bit bit[i];
      IF(i>1)
        bit[i].gnd1 AT bit[i-1].gnd1;
    }
    IF(EVEN(i)) {
      INST adder_one_bit bit[i] RX;
      IF(i>1)
        bit[i].vdd1 AT bit[i-1].vdd1;
    }
  }
}

```

図 8 言語による上位セルの記述

ルのシンボル図からレイアウトを生成する。その際、パラメータファイル中に宣言した基準パラメータの値を関数の引数として指定し、上位セルに要求されるセル面積や性能に合わせて、最適な下位セルのレイアウトを選択、生成する。また、隣接セルの端子位置合せに必要なパラメータの値も、ここで指定する。生成した下位セルのレイアウトは、判断、繰り返し等の命令を使って、配置する。

5. 適用例

本章では、SYLAを使用したセル設計の例を示す。

5-1. パラメータを変更する例

図 9 に、トランジスタのゲート長・ゲート幅をパラメータ表現した 2 ポート出力レジスタの設計例を示す。本回路は、入力部のクロックディンバータと記憶回路中のフィードバックインバータのストレングスを制御することにより、データの書込み・保持・読み取りを行う。ストレングスの制御は、ゲート長・ゲート幅の寸法を変更することにより行う。

5-2. デザインルールの変更例

図 10 に、FIFO 回路 (1bit×4words) を P ウェル構造の 2.0 μm デザインルール、P ウェル構造の 1.2 μm デザインルール、及び N ウェル構造の 0.8 μm デザインルールで生成したレイアウト例を示す。図 10 の FIFO 回路は、繰り返し性の高い 6 つの下位セルで構成されるため、個々の下位セルをシンボリックレイアウト手法で設計した後、言語により上位セルを記述した。

6. 評価

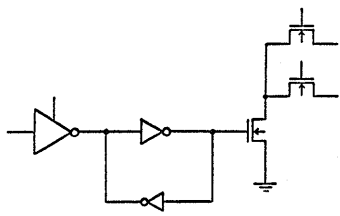
本章では、SYLA を使用して生成したセルの面積評価を行う。面積の評価方法は、人手設計のレイアウトと SYLA で生成したレイアウトとの比較で行う。人手設計のレイアウトとしては、ポリセルアレイ型のスタンダードセルと、フルカスタム LSI 用セルの、2 種類のタイプを取りあげた。比較に当っては、人手設計のレイアウトと SYLA で生成したレイアウトの構造が、できるだけ同じになるようにした。

6-1. ポリセルアレイ型セルでの比較

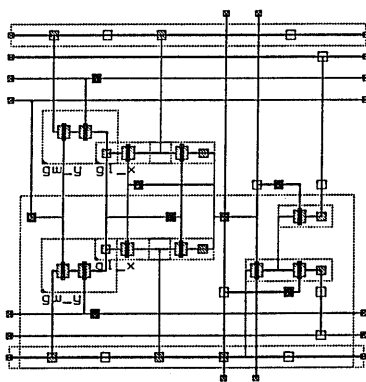
トランジスタ数が数個から数百個の人手設計ポリセルアレイ型セル 51 種類について、SYLA を使用してセル設計を行い、セル面積を比較した。その結果、SYLA で生成したセルの面積は、人手設計のセルに比べ、0 ~ 10 % 増となった。面積が増大しているセルは、人手設計ではトランジスタの形状を折り曲げ、面積減少を図っているセルであった。現状の SYLA では、トランジスタの形状を折り曲げることができないため、面積の増大を招いたと考えられる。

6-2. フルカスタム LSI 用セルでの比較

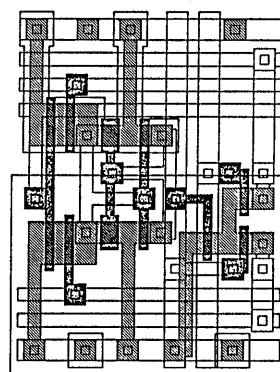
トランジスタ数が数十個のフルカスタム LSI 用セルについても、前節と同様な面積比較を行った。その結果、SYLA の生成したセルの面積は、人手設計セルに比べ、5 ~ 27 % 増となった。この原因としては、前節で述べたトランジスタの形状のほかに、SYLA のコンパクションアルゴリズムである仮想グリッド方式の問題が挙げられる。SYLA では、セルに対する性能や面積の要求に変更が生ずるたびに、パラメータの値を変えて最適なレイアウトを生成し、高密度なセルを実現する。従って、要求を満たさないセルについては、その都度シンボル図からレイアウトを生成し直す必要上、高速なコンパクション処理が可能な仮想グリッド方式を採用した。しかし、仮想グリッド方式では、格子以上に置かれたシンボルの位置関係を変更するようなコンパクション処理はできないため、シンボル図を作成するレイアウト設計者の熟練度によって、生成セルの面積が人手設計の場合と比べ、かなり増大するケースがでてくる。



(a) CMOS回路図

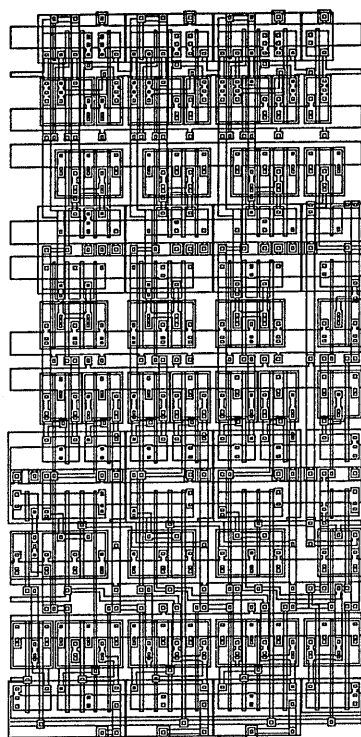


(b) シンボリック設計結果



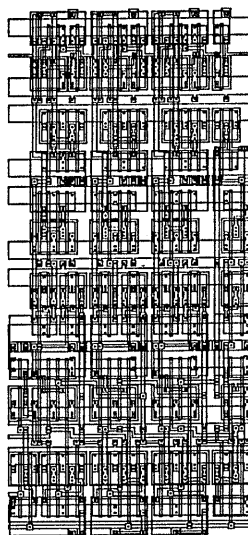
(c) 生成レイアウト

図9 2ポート出力レジスタの設計例



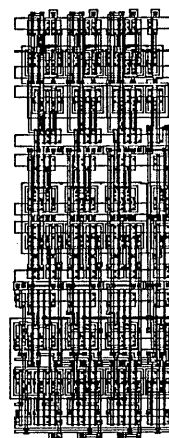
(a) Pウェル構造

2.0 μmデザインルール



(b) Pウェル構造

1.2 μmデザインルール



(c) Nウェル構造

0.8 μmデザインルール

図10 1bit × 4words FIFO回路の設計例

7. おわりに

テクノロジーに独立なシンボリックレイアウトシステム SYLA を使用したセル設計手法について述べた。本システムは現在、プロセッサ系 LSI のデータバスや、ポリセルアレイ型セル等の設計に使用中であり、人手設計に比べてはるかに少ない工数で、妥当なセル面積を実現している。今後は、①. シンボリックレイアウト手法適用規模の拡大による、より高密度なレイアウトの実現、②. 言語記述でセルを隣接配置する際の、端子位置合せの自動化、③. チップレベルの配置・配線システムとの、より有機的な統合、等を課題として改良を続ける予定である。

参考文献

- [1] P. W. Kollaritsch, N. H. E. Weste, "TOPOLOGIZER: An Expert System Transistor Connectivity to Symbolic Cell Layout" *Journal of Solid-State Circuits* pp. 799~804 (1985)
- [2] N. Weste, "Virtual Grid Symbolic Layout" *Design Automation Conf.*, pp. 225~233 (1981)
- [3] J. Rosenberg and N. Weste, "The ABCD Language" *Microelectronic Center of North Carolina Tech Report* 82-01, 1982.