

ルールベースに基づく論理照合手法

植田雅彦 西山 保

松下電器産業(株) 半導体研究センター

VLSI回路検証手法として、ルールベースに基づく論理認識と論理式の恒等性自動証明から成る論理照合手法を提案する。本手法は、レイアウトパターンから抽出されたトランジスタ回路と参照データとの論理的照合を行うものである。まずトランジスタ回路から論理回路を認識し、次にこの論理回路を表すブール式と参照データを表すブール式の恒等性を証明する。本手法では知識ベースアプローチを採用しているため、極めて柔軟な回路検証を可能とするとともに、従来のトランジスタ接続照合による方法で多く発生した疑似エラーの解消を可能にした。

A Rule-based Approach for Logic Comparison

Masahiko UEDA Tamotsu NISHIYAMA

Matsushita Electric Industrial Co., Ltd.
Semiconductor Research Center

We propose a logic comparison method using rule-based logic recognition and a mechanism which proves automatically that two sets of Booleans are equivalent. In this method, circuit data extracted from artwork is compared with reference data at the logic level. At the first step, circuit data is extracted from artwork data. At the second, equivalency is proved between a set of Booleans, one of which represents circuit data and the other of which represents reference data. With the knowledge based approach, quite flexible circuit verification is possible and pseudo errors typically generated by systems in another method, transistor connection comparison, can be reduced.

1. はじめに

LSI規模の増大に伴い、その設計はますます複雑になるとともに、多品種少量生産化が進み、設計期間短縮に対する要求は強くなる一方である。

このような要求は、当然LSI設計へのCADの導入を促し、最終的には計算機による設計の自動化へとつながっていくと思われる。

しかし、現状では、セルの自動配置配線システム等、レイアウト設計の一部に自動化が実現されているのみであり、論理設計はもとより、レイアウト設計においても、面積効率の点から、人手設計に頼っている部分が少なくない。

しかしながら、人手設計は、しばしば誤りを含んでおり、設計結果が設計者の意図通りに設計されているかどうかを検証するCADシステムが不可欠である。

現在、回路検証手法としては、以下のものがあげられる。

- (1) レイアウトパターンから、トランジスタ等の回路素子を抽出し、参照データとの間で接続照合を行う手法[1]
- (2) レイアウトから抽出した回路をシミュレーションすることによって検証する手法[2]
- (3) レイアウトから抽出した回路から、さらに登録されたセルを認識することにより、論理図を復元し、目視検証する手法

(1)は、参照データがある場合に、広く用いられ、実用化されている。ただし、この手法は、レイアウト結果が参照結果と厳密に一致していることを要求するので、多くの疑似エラーを発生する傾向がある。

(2)、(3)は、参照データがない場合にも有効である。ただし(2)では、回路シミュレーションの計算機負荷が大きく、大規模な回路には適していない。(3)は、最終的には人間による論理図の目視検証に頼っており、誤りが入る可能性を排除できないこと、大規模な回路には有効でない点が問題である。

最近、設計者の知識をルールとして取り込み、人間による設計活動を模倣するようなエキスパートシステム的アプローチをとるCADシステムが、論理合成への応用を中心に検討されている。本手法は、このようなルールベースの手法を回路検証に応用したものであり、人間が回路を見て、その論理を認識することにより回路の正しさを検証する行為を模倣したものである。

本手法では、レイアウトパターンから抽出されたトランジスタ回路を論理回路に変換したものと、参照データとの間で、論理式の恒等性を証明することにより回路検証を行う。

本手法による回路検証は、ルールベースによって論理認識した結果に対し、論理のレベルで照合する。そのため、従来のトランジスタの接続照合手法に比べて、極めて柔軟な回路検証が可能になるとともに、疑似エラーをなくすることができる。このことは、設計の自由度を増すことにつながり、その効率化をもたらす。

以下、第2章では本手法の概要を説明し、第3章では本手法に基づいて試作した認識システムについて、また第4章では論理式証明システムについて述べ、第5章では試作システムによる実験結果を示す。

2. 概要および特徴

2.1 レイアウト検証における課題

トランジスタレベルの接続照合に基づく従来の回路検証手法では解決が困難な回路検証における課題について以下に説明する。

(1) レイアウト設計時の回路の変更

レイアウト設計は、必ずしも論理設計データに忠実に進むわけではなく、面積の制約等、その場の状況に応じてフレキシブルに回路構成を変形しながら進められる。その結果、レイアウトパターンは、機能は同じでありながら(従って正しく設計されているながら)回路的には参照データと異なったものになったり、参照データ中に含まれる2つの同一セルがレイアウトパターンの中ではそれぞれ異なった回路に対応していたりする。

(2) ゲート入力の変換

ゲートの入力が互いに等価で交換可能であるというようなことは、設計者にとって常識である。レイアウト設計において、その場の状況により自由に、ゲートの入力が交換されている。

(3) 冗長トランジスタの付加

ファンアウトの調整等の理由で、同一の機能を持つ複数のトランジスタがレイアウトされることがよくある。

このようなレイアウトパターンと参照データとの不一致は、従来のレイアウト検証システムにおいて、様々な例外的な工夫により[4]、その一部には対応されているが、基本的には疑似エラー

としてシステムが検出すべき真のエラーと一緒に出力され、その区別は困難である。このような問題を避けるためには、レイアウトパターンに基づいて論理設計結果を修正すればよいが、それでは修正された論理設計の再検証も含めて設計工数の増大を招く。

以上述べた問題点は、従来のトランジスタの接続照合法の持つ根本的な欠点である。

2.2 本手法の概要

以上の問題点を解決するために、本手法では、ルールベースに基づく論理認識と論理式の恒等性の証明による論理レベルでの照合を採用した。

本手法の概要をFig.1に示す。以下、順をおって処理の説明を行う。

(1) 論理認識

レイアウトパターンから抽出されたトランジスタ回路は、ルールベースの論理認識によって論理回路に変換される。ユーザはルールを変更することにより、CMOS、NMOS、ECL等の様々なテクノロジー、また、スタティック、ダイナミック等、様々な回路スタイルでのレイアウト設計に対応することができる。

第3章で詳しく説明するが、ルールベースの手法を用いることによって、従来のセル認識手法に比べて、柔軟な論理認識が可能であり、前記問題点(2)(3)を解決している。

(2) 論理式の恒等性証明

変換された論理回路と参照データである論理設計結果は、回路出力およびレジスタ入力を出力変数とし、回路入力およびレジスタ出力を入力変数とする論理式で表現される。

論理式で表現された2つの回路は、入出力間の対応をとった後、2分決定木への変換に基づいた恒等性の証明により、論理的に等しいことが検証される。従って、レイアウト設計での回路変更があっても、それが論理的に等価なものであれば、本手法ではエラーにはならず、論理が変わってしまうような真のエラーだけを出力することができる。

(3) 論理図作成

論理認識された結果をもとに自動的に論理図を作成することができ、その結果をもとに設計者が目視検証を行うことが可能である。人手によりレイアウトされる回路の中には、メモリーのように論理設計を通さないで直接レイアウトされるも

のも少なくない。目視検証は、このような参照データがない場合に有効である。

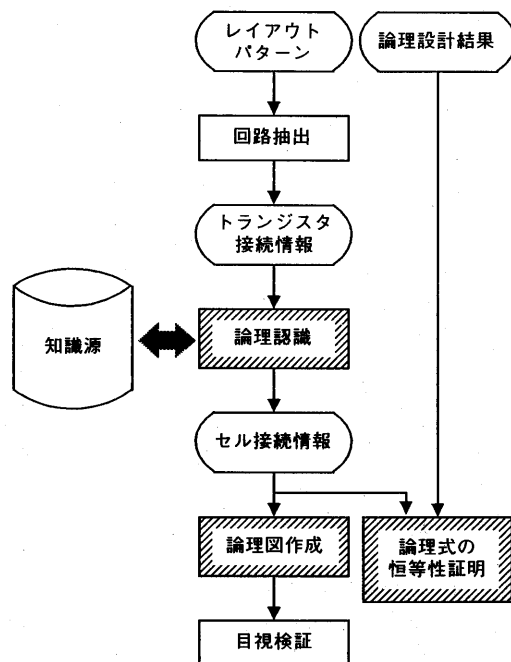


Fig.1 本手法の概要

3. ルールベースの論理認識

論理認識部は、従来のアルゴリズム的なセル認識手法よりも柔軟な認識を行うために、Prologベースのプロダクションシステム上に実現している。

ワーキングメモリに取り込まれたトランジスタ接続情報は、階層的な回路認識を表現するプロダクションルールに従って変換され、最終的にはセルの接続情報に変換される。ここでは論理だけを問題にしているのでトランジスタは接続情報だけを持ち、その物理的情報を失っている。

3.1 変換ルール

ルールの例をFig.2に示す。このルールは、「もしもPMOSの直列接続とNMOSの並列接続が図のように接続されており、しかもそれぞれのゲートの集合であるリストL1とL2が順序に依らず同じ要素を持つならば、それを対応するNORゲートに置き換える」ことを表現している。

推論システムはワーキングメモリ上のデータから、条件にマッチする回路パターンを検索すると、それをワーキングメモリから削除し、実行部に記述された回路パターンをワーキングメモリに書き込む。

ここで注意すべきことは、このルール1つで任意入力数NORゲートの、しかも入力に関するあらゆる置換に対応しているということである。これによって本手法は、ゲートの入力置換に関する問題点を解決している。

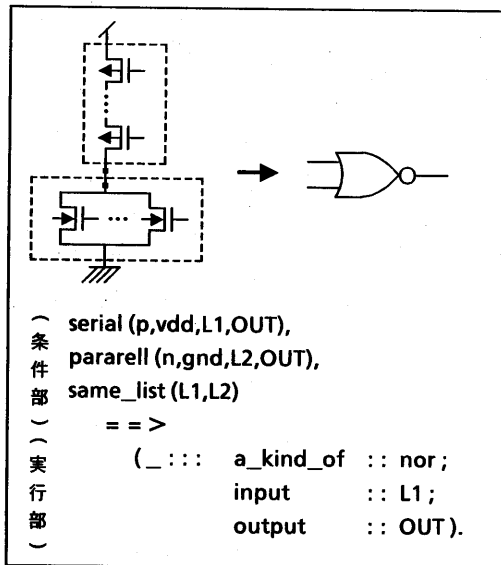


Fig.2 変換ルールの例

3.2 処理フロー

論理認識部の処理フローをFig.3に示す。以下順番に処理の内容を説明する。

(1) 冗長トランジスタの除去

認識に先立って、Fig.4に示すような冗長なトランジスタがあれば、これをワーキングメモリから除去する。

(2) 並列-直列接続の認識

まずトランジスタの並列接続、直列接続を認識する。Fig.5に示すように互いに直列に接続された同極性のトランジスタは、トランジスタの直列接続を意味する仮想的な回路素子に変換される。各トランジスタのゲートは、1つのリストのま

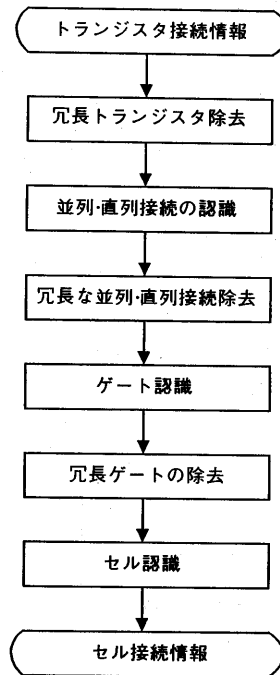


Fig.3 論理認識部の処理フロー

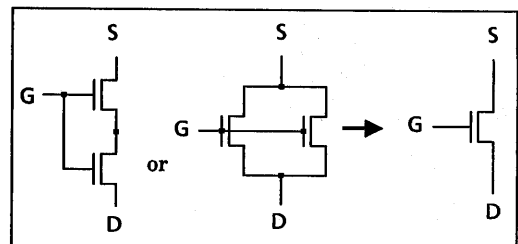


Fig.4 冗長トランジスタの除去

められている。ここで各トランジスタの代わりに並列接続もしくは直列接続であってもよい。

(3) 冗長な並列-直列接続の除去

Fig.6に示すように冗長な並列接続や直列接続があれば、これをワーキングメモリから除去する。ここでは、トランジスタの順序が入れ替わっていても同じものとみなしている。

(4) ゲート認識

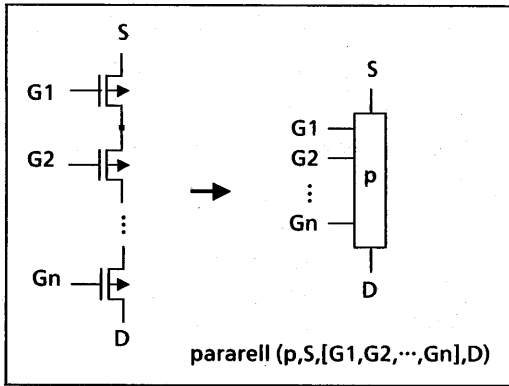


Fig.5 トランジスタ直列接続の認識

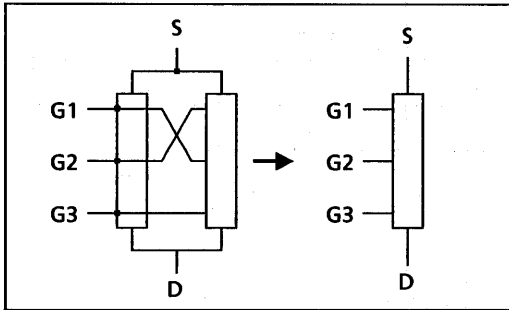


Fig.6 冗長な直列接続の除去

次にトランジスタもしくは直列接続と並列接続が相補的に接続されている場合、それらをインバータもしくはNOR、NANDゲートで置き換える。Fig.2にNORの例を示しているが、この中の任意のトランジスタは直列接続、あるいは並列接続であってもよいから、このルールは単純なNORだけではなく、AND-NOR型の複合ゲートをも認識することが可能である。

(5)冗長ゲートの除去

ゲートレベルでも、直列インバータ等の論理的冗長性の除去を行う。

(6)セル認識

ゲート認識された回路は引き続いて、それらのゲートによって構成されるセルへと変換される。ここでも認識は階層的に進められる。セル認識の最大の目的は、この後に続く論理照合のため

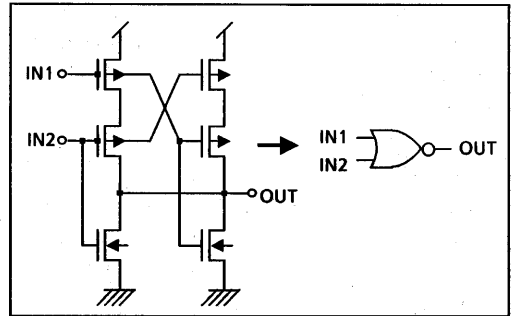
に回路の中からレジスタを認識することである。

3.3特徴

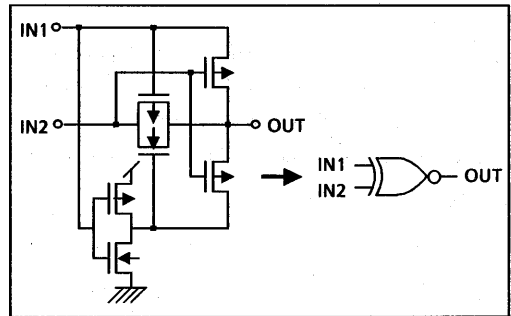
論理認識の特徴を以下に示す。

(1)階層的認識

トランジスタ回路から、直接セルを認識するのではなく、トランジスタの直列・並列接続から始まってゲートの認識等多くの中間的な回路要素の認識を階層的に行うことによって、認識処理を効率化するとともに、エラーがあった場合にその解析を容易にしている。



(a)



(b)

Fig.7 変則的ゲートの認識

(2)冗長回路要素の除去

冗長なトランジスタやその直列・並列接続等をデータの中から取り除くことにより、柔軟な論理認識が可能である。例えば本手法ではFig.7(a)に示すような回路も特別な例外処理を加えることなく、正しく認識することができる。

(3)論理の認識

本手法では、従来のアルゴリズム的なセル認識手法とは異なり、認識されるのは個々のトラン

ジスタ接続パターンと1対1に対応するセルではなく、入力数や回路構成等、多様な自由度を持った抽象的な論理である。従って、ゲート入力 of 置換や論理回路の同一セルが複数の回路構成で実現される問題は、本手法では例外的なテクニックを用いてではなく本質的に解決されている。本手法のこの特徴は主に、リストによる柔軟なデータ構造によってもたらされる。

(4) ルールベース手法

本手法では、回路変換はプロダクションルールで記述される。推論システムは、Prologのパターンマッチングによってワーキングメモリからルールの条件にマッチする回路パターンを検索する。従って、Fig.7(b)に示すような変則的なゲートも例外ルールを設けることにより、容易に対応できる。

4. 論理式の恒等性証明

論理認識した結果と参照論理回路の照合は、両回路の論理式表現を作成し、論理式の恒等性を証明することにより行う。そのようにした理由は

(1) Fig.8に示したように論理ゲートレベルでは、表現の多様性のために、接続照合による手法が有効ではないこと、

(2) またレイアウト上の制約や、配線容量の増大を保証するバッファの挿入等、レイアウト時に様々な事情で論理の等価変換が行われるが、そのような場合にも柔軟に対応できるようにするためである。

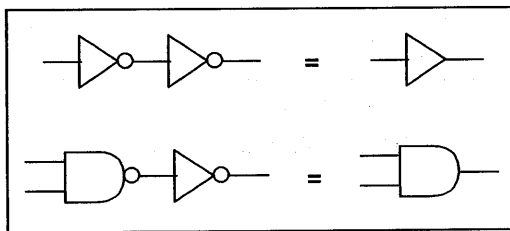


Fig.8 論理回路の表現の多様性

4.1 アルゴリズムの検討

論理式の恒等性を証明するアルゴリズムとして、次に示す2つの方法を検討した。[5]

(1) 2つの論理式の排他的論理和を作り、それをDeMorganの法則等に基づいて積和標準形に展開して、それが恒等的に0であることを確認する。

(2) 2つの論理式の排他的論理和を作り、それを論理関数のShannon展開に基づいて2分決定木に変換してそれが0であることを確認する。

それぞれについてProlog上で記号処理に基づくプログラムを作成し、いくつかの論理式を使って処理時間を比較した結果をTable1に示す。この結果から本手法ではShannon展開に基づく方法を採用した。

Table.1 論理式恒等性証明アルゴリズムの比較

回路	積和展開	Shannon展開
#1	3.42	0.17
#2	1.32	0.32
#3	1.19	0.31
#4	7.31	0.55

(単位sec)

4.2 Shannon展開に基づく論理照合手法

Shannon展開に基づく論理関数の照合手法として、次のような手法が提案されている。[5]

(1) 入力変数値で場合分けする方法[6,7]

(2) 2分決定木グラフへ変換する方法[8]

[6]は2つの論理の排他的論理和を作り、入力変数を順番に0もしくは1に固定して出力を評価し、出力が決定するまでそれを繰り返す。このようにして全ての場合について出力値が0であれば2つの論理は恒等的に等しいということが出来る。

[7]は同じ手法をキューブ表現された論理関数に適用している。

[8]は2つの論理関数を、入力変数の順序を固定した特別な2分決定木グラフに変換する。このようにして作られたグラフは論理関数の正準表現になっていることが知られ、グラフの同形判定によって2つの論理の恒等性を判定できる。

[8]にはグラフを取り扱うための各種アルゴリズムが与えられており、論理関数からグラフへの

変換はサブグラフ間の論理演算によって行われる。

4.3 記号処理に基づくグラフへの変換

ルールベースの論理認識との整合性から、本手法ではPrologの記号処理を使って論理式から2分決定木グラフへ変換することにより論理式の恒等性を証明している。

すなわち、[8]のようなサブグラフ間の論理演算ではなく、直接Shannon展開(論理式の1つの入力変数を1または0に固定した時の論理式)を記号処理によって効率的に求めることによって高速のグラフ変換を実現している。

Table.2にサブグラフ間の論理演算による方法と直接Shannon展開による方法を、共にProlog上にインプリメントして処理時間を比較した結果を示す。記号処理を使った直接Shannon展開による方法がかなり効率的であることが分かる。

Table.2 2分決定木グラフ変換手法の比較

回路	サブグラフ間演算	直接Shannon展開
#1	1.35	0.17
#2	1.52	0.32
#3	2.38	0.31
#4	6.10	0.55

(単位sec)

Fig.9に論理式とそれに対応する2分決定木グラフの例を示す。

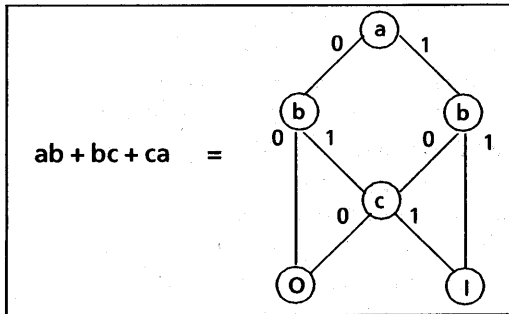


Fig.9 2分決定木グラフ

5.実験結果

本論理照合手法に基づいた試作システムをVAX8650上のQuintus Prologでインプリメントし、実際の設計データを使ってテストし、その有効性を確認した。

処理時間も論理認識については約2000トランジスタのデータに対して約30分で処理しており、十分実用的である。Fig.10に実験から得られた回路規模と処理速度の関係を示す。

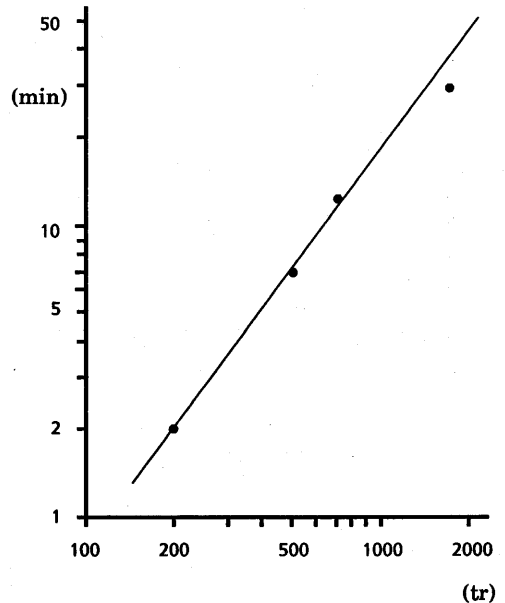


Fig.10 論理認識の処理時間

また論理式の恒等性証明についても、Fig.11に実験から得られた論理式に含まれる入力数と2分決定木への変換時間(処理時間の主要部分)の関係を示す。

論理式の恒等性証明の問題は、NPハードな問題として知られており、実際ワーストケースでは、処理時間は入力数に対してexponentialに増大しているが、白丸で示した実際的な問題に対しては処理時間は実用的なレベルに抑えられている。

また、Fig.12に本試作システムによって論理認識し、その結果を基に自動生成した論理図を示す。信号の流れを反映した高品質な論理図を生成することにより、設計者の目視検証を容易にしている。

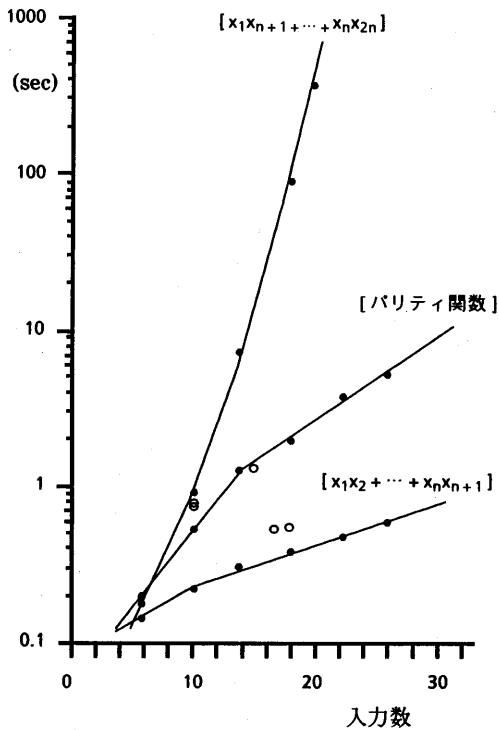


Fig.11 論理式証明の処理時間

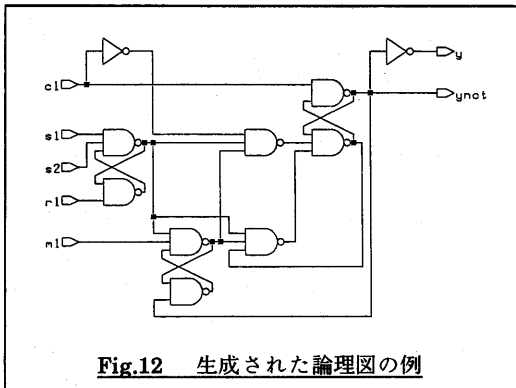


Fig.12 生成された論理図の例

6.まとめ

ルールベースの論理認識と論理式の恒等性自動証明を用いたVLSI回路検証手法を提案し、その試作システムを開発して評価を行った。

本手法は従来のトランジスタ接続照合による検証手法では困難であったような問題点を本質的に解決している。

その結果、レイアウト設計における自由度が増し、またレイアウト検証のための参照データの修正のような余計な作業が不要になる。

また、参照データのない場合、もしくは比較的小規模な回路の場合には、論理認識した結果から論理図を自動作成してそれを目視検証することも有効である。

試作システムの実際の設計データを使ったテストによって、処理時間も十分実用レベルであることが分かった。

謝辞

最後に本研究に際し、日頃御指導や貴重な御助言を頂きました当社デバイス開発研究所所長 堀内司朗博士、ならびに本研究開発に惜しみない御協力を頂いた松本典子、泉野祥吾の各氏に深く感謝致します。

参考文献

- [1]R.L.Spickelmier and A.R.Newton : WOMBAT : A New Connectivity Verification Program, Proc. of ICCAD,1983.
- [2]大江, 後藤 : LSIレイアウト自動検証システム, 信学技報 CAS87-116,1987.
- [3]植田, 他 : 論理合成エキスパートシステム LODES - 概要・評価 -, 情報処理学会第36回全国大会,1988.
- [4]K. Sato, et al. : A Method of Connectivity Checking for MOS Circuit, Proc. of ISCAS85, pp1285-1288,1985.
- [5]藤沢, 藤田 : 各種論理照合アルゴリズムの比較と統合, 第18回FTC研究会資料,1987.
- [6]G. L. Smith, et al. : Boolean Comparison of Hardware and Flowcharts, IBM J. RES. DEVELOP., VOL. 26, NO. 1, pp106-116, 1982.
- [7]G. D. Hachtel, et al. : Algorithms for Multi-level Tautology and Equivalence, Proc. of ISCAS85, pp1277-1280,1985.
- [8]R. E. Bryant : Graph-Based Algorithms for Boolean Function Manipulation, IEEE Transaction on Computers vol C35 NO.8 1986