

回路比較照合システム — ルールベースを利用した擬似エラーのチェック —

池内敦彦¹ 高島誠² 田中利一³ 小島昌一⁴ 斎藤玉樹⁵ 坂田純一⁶

1-2 (株) 東芝 集積回路事業部

3-4 (株) 東芝 システム・ソフトウェア技術研究所

5-6 (株) 東芝マイコンエンジニアリング

本稿では、LSIのマスクパターンの接続検査を行う回路比較照合システムについて報告する。

本システムでは精度のある回路の比較照合を行うためルールベースを利用し擬似エラーを判定する処理を追加した。本処理により、ルールベースに登録した擬似エラーについては全てその出力を抑止することが可能となる。

"Circuit Comparison System - A Rule Based System for Pseudo Error Detection-" (In Japanese)

by Atsuhiko IKEUCHI, Makoto TAKASHIMA (Semiconductor Division Toshiba Corporation, 580-1, Horikawa-cho, Kawasaki, 210, Japan), Toshikazu TANAKA, Shoichi KOJIMA (Systems & Software Engineering Laboratory, Toshiba Corporation, 70, Yanagi-cho, Kawasaki, 210, Japan), Tamaki SAITO and Jun-ichi SAKATA (Toshiba Microcomputer Engineering Corporation 215-1, Ekimaeon-cho, Kawasaki, 210, Japan)

This paper describes a system which performs circuit connectivity comparison. This system has been developed for verifying circuit connectivity extracted from LSI mask artwork.

A new approach is used in this system, which uses a rule based method to detect pseudo errors.

This approach enables the system to eliminate pseudo errors.

1. はじめに

近年LSIのマスキレイアウトの自動設計のCADの開発が進んでいるが、バイポーラ素子、高集積が要求されるメモリ等、依然としてマニュアル設計に頼っているものは少なくなく、その接続検証のCADは必要不可欠である。マスクパターンの論理接続検証を行うのに、マスクパターンより抽出した回路と論理図や回路図とを比較する回路比較照合法によるCADツールがあり、数多く開発されている。¹⁻²

レイアウト設計においては、レイアウトスペースの制約及び回路特性の向上等の理由で、もとの回路図と異なる回路を作成することがある。この場合、回路の構造を比較照合すると、比較照合以前に両者の回路の対応が取れていないので、不一致として出力されてしまう。

以上のように出力される不一致は、擬似エラーと呼ばれ、これが発生すると接続検証作業に大きな負担がかかってしまうことが問題となっていた。この様な擬似エラーの発生を抑止するために、比較照合以前に回路の縮約処理等を行う対策が用意されているが、あらゆる等価変換に対応するものを実現するのは難しく、完全に抑止するには至っていない。³⁻⁴

今回、従来のシステム(CCOMP)の縮約処理で対処出来なかった擬似エラーに対して、ルールベースを用い判定し、その出力を抑止する処理を追加した回路比較照合システムCCOMP/EXを開発した。

ここでは、CCOMP/EXのシステム構成、擬似エラーの抑止手段としての縮約処理、及びルールベースを利用した判定処理とその実験結果について報告する。

2. システム構成

図1にパターン検証システムの一部であるCCOMP/EXのシステム構成図を示す。

本システムは、MOSトランジスタ、バイポーラ素子BI-MOS素子等を対象に10万素子規模のデータの検証が可能である。以下の3種類を入力データとしている

(a) レイアウト回路記述データ⁵⁻⁶

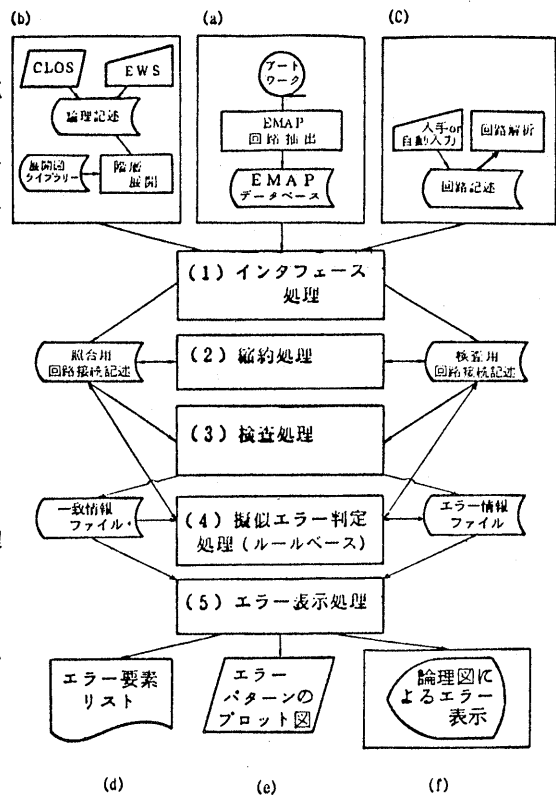
マスクパターンよりEMAPで回路抽出されたフラットなトランジスタレベルの回路接続記述
DRC等のチェックはEMAPで行われる

(b) 論理記述データ⁷

図面自動読取りCLOS、及び論理図入力出来るEWS等で自動的に作成された階層構造をもつ論理接続記述

(c) 回路記述データ

回路解析プログラムの入力データでトランジスタレベルの回路接続記述
サブキットを用いた階層表現が可能である



以上の3種類の入力データはどの組合せでも比較照合可能であり、また同一記述同志の比較照合も可能である。

図1 CCOMP/EXのシステム構成図

CCOMP/EXの処理の概要を順に従って説明する。

(1) インタフェース処理

他システムの回路記述を入力データ(図1 (a)-(c))として、CCOMP/EX固有の記述に変換し、比較照合すべき照合用回路記述と検査用回路記述を作成する。

本システムはフラットなトランジスタレベルで比較を行なうので、階層構造をもつ回路は全て展開される。

トランジスタレベルの比較を採用した理由は、

[1]復元処理よりも展開処理のほうが柔軟性があり、論理図からレイアウトへ変換する際の多様な等価変換に柔軟に対応する

[2]照合用データとして論理図の他に回路図も扱える
である。

(2) 縮約処理

トランジスタレベルの回路が正しく比較照合されるように、前処理として照合側、検査側両回路の縮約を行う。

(3) 検査処理

縮約処理後の両回路の比較照合による接続検査を行う。
回路の比較照合は2つの接続記述を無向グラフに置き換え、その同型判定処理を行なうことによって実現している。

グラフは、素子及びノードを節点とし、素子の端子を枝として構成し、節点には素子及びノードのタイプ、枝には端子のタイプという属性がつけられる。

図2に回路のグラフ表現の例を示す。

本アルゴリズムを利用することによって、次の特徴が得られている。

[1]処理時間の増加は回路規模に対し、ほぼ線形オーダーである

[2]比較実行時に必要な初期対応データは、通常は外部入力端子(パッド部)だけで十分である

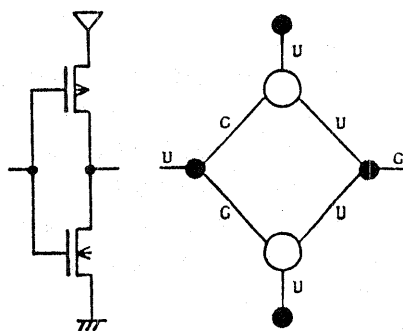
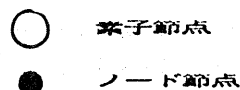


図2 回路のグラフ表現の例

本処理で照合が取れたものは一致情報、不一致となったものはエラー情報としてファイルに格納される。他に回路パラメータ値の比較等の検査も用意されている。

(4) 擬似エラー判定処理

エラー情報及び両者の回路記述を入力データとして、ルールベースシステムを利用して擬似エラーの存在の有無の判定を行い、それにもとずきエラー情報を更新する。

(5) エラー表示処理

エラー情報のファイルをもとに、CCOMP/EXのエラーとして表示する(図1 (d)-(f))ための出力処理を行う。エラーの出力はエラーの種類によってレベル分けされる。

3. 擬似エラーの抑止

レイアウト設計において、もとの回路図とは機能的に等価だが、構造的に異なる回路を作成すると擬似エラー発生の原因となる。それは以下の3つに大別される。

- (i) もとの回路図では1個の素子を複数に分割して直並列に並べた場合
- (ii) 回路図とレイアウトとくらべ、図3に示すような回路の位置的交換があった場合
- (iii) 回路図とはまったく構造の異なる等価変換された場合

以上の原因で発生する擬似エラーに対してCCOMP/EXでは、

- 1] 回路の比較照合前に、縮約処理を行い擬似エラーの出力を最小限にする対処をし、
- 2] 比較照合後、1] で対処出来なかった擬似エラーをルールベースシステム上で判別し、その出力を抑止して対処している。

3.1 縮約処理

処理は以下の2種類に分類することが出来る。

- [1] 抵抗、容量、ダイオード、及びゲート信号が等しいトランジスタ等の直並列接続素子に対し、それぞれ1個の等価な素子に変換する
- [2] ゲート信号が異なる同タイプの直並列接続のMOSトランジスタについて、図5に示すように変換して架空のゲート（以下、縮約ゲートと呼ぶ）を作成する

[1],[2] の縮約において、各素子のサイズ等のパラメータ値は、それぞれ合成計算される。

図4に[1]の縮約処理の例を示す。

図6に[2]の縮約処理の例を示す。ここで示す縮約ゲートは、入力信号の位置情報を持たないことを特徴としている。

[1]の縮約によって、前記(i)によって発生する擬似エラーを抑止することが出来る。

また[2]の縮約は、前記(ii)のケースに対応している。図7は、回路の位置的交換がある照合側、検査側両回路を[2]の縮約で、回路構造的に等価にしている例を示している。

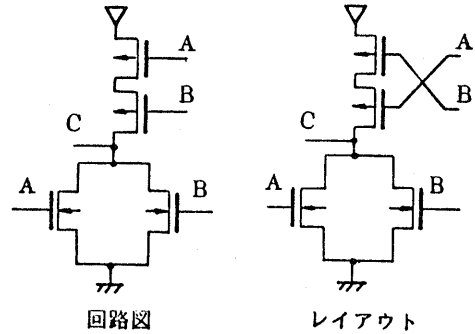


図3 回路の位置的交換の例

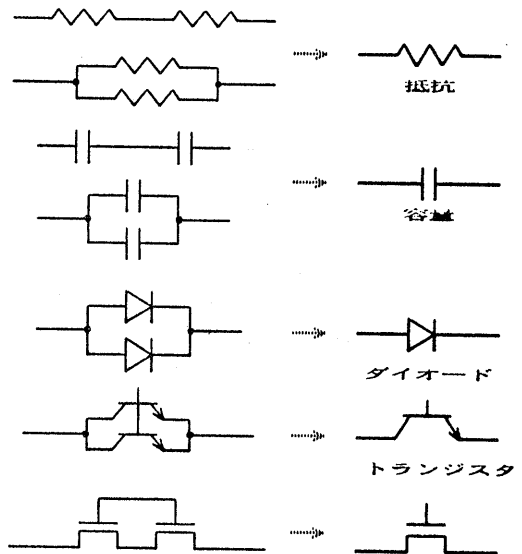


図4 縮約処理 [1]

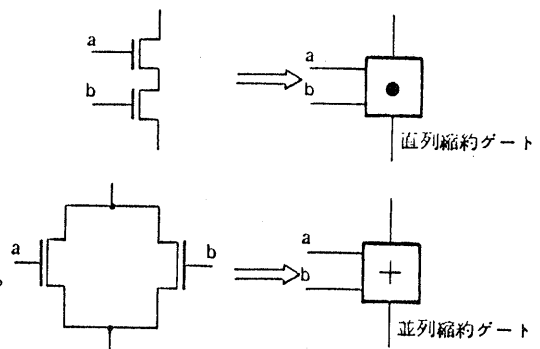


図5 縮約処理 [2]

縮約される回路の中には、図8に示すように回路の位置的交換をエラーとしなければ不都合な回路も存在する。このような回路に対して以下の処理が用意されている。

- [1] インタフェース処理で位置的交換の許されないトランジスタを識別するために、該当するトランジスタに交換禁止フラグという属性を付与する。
- [2] 縮約処理は、交換禁止フラグのセットされているトランジスタも区別なく実行する。この際、交換禁止フラグは縮約ゲートにも伝搬させる。
- [3] 回路比較照合後、照合の取れた縮約ゲートのうち、交換禁止フラグのセットされているものについては、再度トランジスタレベルに展開し回路構造の比較照合を行う。

例えば図8の場合、照合側でクロック入力に交換禁止フラグをセットしておけば、縮約処理後の比較照合では両者は不一致とならないが、縮約ゲートに交換禁止フラグが伝搬されているためトランジスタレベルでの再比較の対象となり、この段階で不一致が検出される。

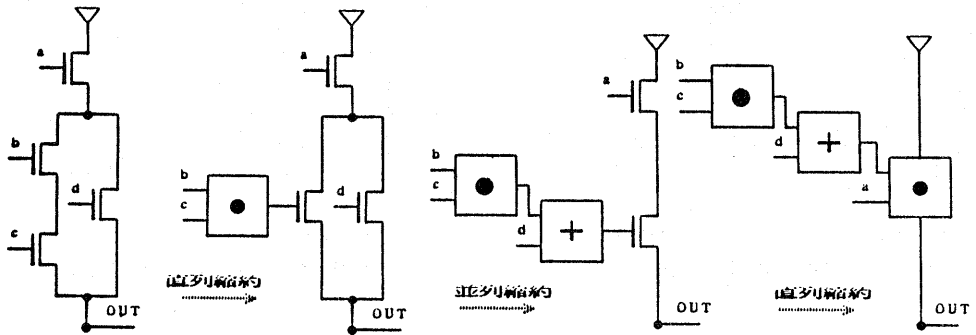


図6 [2]の縮約処理の例

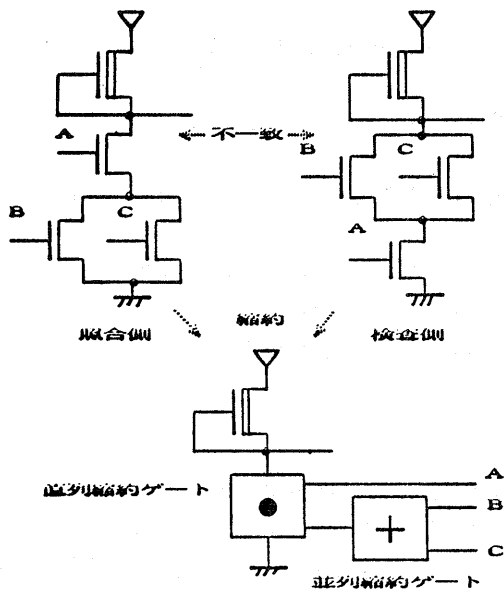


図7 位置的交換がある回路をそれぞれ縮約した例

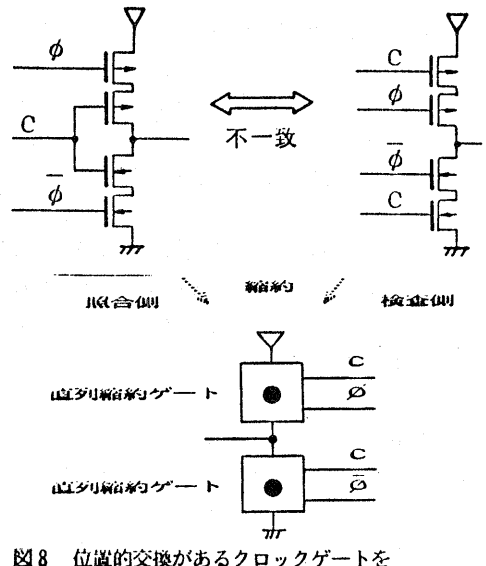


図8 位置的交換があるクロックゲートをそれぞれ縮約した例

3. 2 ルールベースシステムを利用した擬似エラーの判定

縮約処理で対処出来ない回路とは、前記(iii)で示したケースで、両者をプログラムによって回路構造的に等価にすることは難しい。例えば、図9に示す機能的等価な回路をそれぞれ縮約しても図10に示す回路となり、両者は回路構造的に異なり不一致になってしまう。

ここでは、縮約処理で対処出来なかった擬似エラーの回路をエラーとして出力しないために、以下の3つの処理を行い対処している。

- step 1 エラー素子を含む回路ブロック（以下、不一致回路ブロックと呼ぶ）を抽出する
- step 2 抽出されたブロックについて擬似エラーの有無をルールベースを用いて判定する
- step 3 判定内容をもとにエラー回路情報を更新する

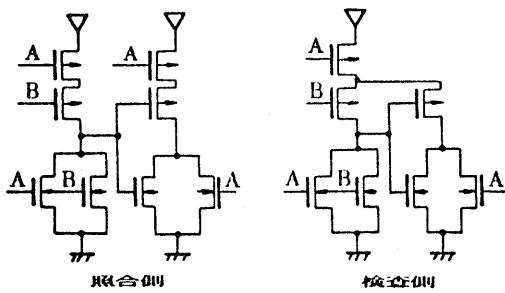


図9 機能的に等価な回路の例

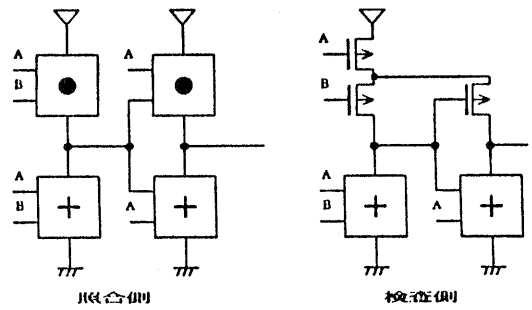


図10 図9の回路をそれぞれ縮約した回路

不一致回路ブロックの抽出

不一致回路ブロックは照合側と検査側の回路ブロックのペアで構成される。1個の不一致回路ブロックを抽出するには以下の処理を行う。

- [1] 回路比較照合結果、不一致となった素子をもとに、検査側の回路をたどり回路ブロックを抽出する。

ブロックの抽出は電源、GND、出力ノード以外のゲートに入力するノードを終点として回路をたどる。

- [2] [1]で抽出された回路中の素子と一致する照合側の素子が存在すれば、それをもとに照合側の回路ブロックを[1]と同様に抽出する。
- [3] [2]で抽出された回路中の素子と一致する検査側の素子でブロック外に存在するもの（未抽出素子）があれば、その素子をもとに検査側の回路ブロックを再び抽出する。
- [4] 以下、抽出された回路中の素子と一致する素子が完全にブロック内に含まれるまで抽出処理を繰り返す。

擬似エラーの判定処理

ここでは、不一致回路ブロック中にあらかじめ知識ベースに登録しておいた擬似エラーのパターンが存在するルールベースを用いて判定している。抽出された全ての不一致回路ブロックが判定対象となる。

擬似エラーのパターンは、図10に示すような縮約処理後の照合側、検査側ペアの部分回路で構成される。これをIF-THEN形式のプロダクションルールとして知識表現した。

この知識表現は、個々のパターンを表現した個別のルールであり、CAD開発者のみ把握している比較照合アルゴリズムの知識とは違い、LSI設計者でも容易に表現出来ることを特徴としている。

擬似エラーの判定処理は、以下に示すように大きく3つに分けて考えることが出来る。

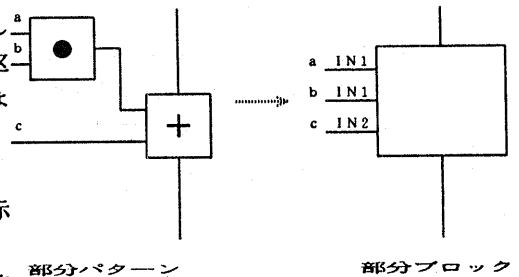
- 1) 照合側と検査側回路ブロックが擬似エラーのパターンを含むかチェックする。
- 2) 1)の擬似エラーのパターンを考慮し、照合側と検査側で対応が取れるかチェックする。
- 3) 2)で対応が取れた両回路ブロックの電源、GND、及び入出力線が正しく対応しているかチェックする。

推論の大部分を占める1)のチェックにおいて、1つの擬似エラーのパターンをまとめてルールの条件部に登録するのは実用上問題が生じるので、推論方法は擬似エラーのパターンを分割し、それぞれの部分パターンについてマッチングを取り、それらをまとめあげていくようにしている。

このような推論を効果的に行うために、ここでの回路接続表現の知識は以下の特徴を持つ。

[1] 交換可能な接続線の処理

部分パターンの素子は入力信号の位置的交換可能な表現をされているので、その組合せ数分のマッチングが発生してしまい処理時間がかかる。これに対して、入力信号を区別する述語表現をし、マッチングの回数が最小限になるようにしている。



[2] 部分パターンの抽象化

部分パターンとマッチングが取れたら、それを図11に示すように部分ブロックとして抽象化して階層表現する。

部分ブロックは階層が上がっても従来と同じ素子表現が出来る、下位レベルの位置的交換の不可の属性も保存される
 例えば図11の部分ブロックの場合、同じ属性IN1を持つ信号aとbとは交換可能であり、属性IN2を持つ信号cはa、bとは交換不可であることを示す。以上のように階層的表現をすることよりデータ量が減り、大規模回路の知識表現を可能としている。

図11 部分パターンの抽象化

エラー回路情報の更新

ルールベースシステムの判定内容をもとに以下の処理を行う。

不一致回路ブロック中の擬似エラーのパターンを構成する回路から回路比較でエラーとなっているものをまとめ、これらをエラー情報ファイルより削除する。

この処理によって更新されたエラー情報ファイルを入力とし、エラー表示処理を行えば、登録されている擬似エラーの出力は抑止することが出来る。

step 1.3は比較照合処理と同様にホストコンピュータ上で稼動する処理であるのに対し、step 2の処理は当社ワークステーションAS3000上で稼動するルールベースシステム上で実行される。

ルールベース構築ツールとして、当社開発のARCH-1 (Advanced Reasoning Core High-Level Language)言語を利用した。図12に本ルールベースを利用した処理環境を示す。

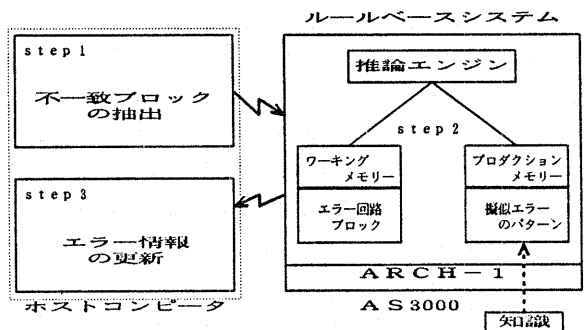


図12 ルールベースを利用した処理の環境

4. 実験結果

表1に、ルールベースを利用した擬似エラーの判定処理の実製品を用いての実験結果を示す。

製品	エラーブロック数	最大ブロックのトランジスタ数	ブロック抽出時間 *	擬似エラーの検出されたブロック数	擬似エラー判定処理時間 **
1	14	(S1) 14 (S2) 16	3秒	4	156秒
2	9	(S1) 55 (S2) 53	2秒	0	751秒
3	32	(S1) 10 (S2) 16	8秒	1	192秒

* ホストコンピュータ
** AS3000

表1 擬似エラー判定処理の実験結果

擬似エラーの判定用のルールはcmosトランジスタで作成した。

製品2において、擬似エラーの判定処理時間が他と比べ多いのは、1ブロックで50素子を超えるブロックの判定中に擬似エラーのパターンに類似する部分パターンが多数存在し、この結果、部分マッチングが沢山行われたからである。

5. おわりに

回路比較照合システム(CCOMP/EX)について、擬似エラーの対策処理を中心として報告した。

現在本システムは擬似エラーの判定処理部を除いては既に実用化され広く使用され、接続ミスの早期発見に役立っている。今後は擬似エラーの判定用のルール数を増し、擬似エラーの判定処理部を含んだシステムの実用化を推進していく予定である。

<参考文献>

- (1)F.Luellau.et al., 'A Technology Independent Block Extraction Algorithm', 21th DAC, pp.610-615, 1983.
- (2)羽山繁他, 'MOS マスク解析における論理検証', 情報処理学会設計自動化研究会, 19-6, 1983.
- (3)加藤雄他, 'バイポーラIC用階層セル接続検証プログラム', 情報処理学会設計自動化研究会, 27-5, 1985.
- (4)Yehuda Shiran, 'YNCC: A NEW ALGORITHM FOR DEVICE-LEVEL COMPARISON BETWEEN TOW FUNCTIONALLY ISOMORPHIC VLSI CIRCUIT', IEEE pp298-302, 1986.
- (5)T.Mitsuhashi.et al., 'An Integrated Mask Artwork Analysis System', 17th DAC, pp.277-284, 1980.
- (6)T.Chiba.et al., 'A Mask Artwork Analysis System For Bipolar Integrated Circuit', Twenty Third IEEE Computer Society International Conference, pp.175-183, 1981.
- (7)太田博, 川本栄二, 情報処理学会第29回全国大会予稿集 2K-3.
- (8)M.Takashima.et al., 'Programs For Verifying Circuit Connectivity Of MOS/LSI Mask Artwork', 19th DAC, pp.544-550, 1982.
- (9) 中村明他, 情報処理学会第34回全国大会予稿集 4L-1.4L-2.