

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5311351号  
(P5311351)

(45) 発行日 平成25年10月9日(2013.10.9)

(24) 登録日 平成25年7月12日(2013.7.12)

(51) Int.Cl.	F I
GO 1 R 31/3183 (2006.01)	GO 1 R 31/28 Q
GO 1 R 31/28 (2006.01)	GO 1 R 31/28 G
GO 6 F 11/22 (2006.01)	GO 6 F 11/22 3 1 O B

請求項の数 7 (全 24 頁)

(21) 出願番号 特願2010-525650 (P2010-525650)	(73) 特許権者 504174135 国立大学法人九州工業大学 福岡県北九州市戸畑区仙水町1番1号
(86) (22) 出願日 平成21年7月30日(2009.7.30)	
(86) 国際出願番号 PCT/JP2009/063586	(74) 代理人 100116573 弁理士 羽立 幸司
(87) 国際公開番号 W02010/021233	(72) 発明者 宮瀬 紘平 福岡県飯塚市大字川津680番地の4号国立大学法人九州工業大学内
(87) 国際公開日 平成22年2月25日(2010.2.25)	
審査請求日 平成24年5月16日(2012.5.16)	(72) 発明者 温 暁青 福岡県飯塚市大字川津680番地の4号国立大学法人九州工業大学内
(31) 優先権主張番号 特願2008-211473 (P2008-211473)	(72) 発明者 梶原 誠司 福岡県飯塚市大字川津680番地の4号国立大学法人九州工業大学内
(32) 優先日 平成20年8月20日(2008.8.20)	
(33) 優先権主張国 日本国(JP)	

最終頁に続く

(54) 【発明の名称】 生成装置、生成方法及びプログラム

(57) 【特許請求の範囲】

【請求項1】

与えられる集合であって、故障検出対象の論理回路に入力される入力ビットが、論理値1若しくは論理値0の論理ビットからなるテストベクトルの集合、少なくとも一つのビットが未定値の未定値ビットを含み残余のビットが論理ビットであるテストキューブの集合、又は、テストベクトルとテストキューブとの組み合わせの集合から、前記与えられた集合の故障モデルを維持しつつ又は故障検出率を維持しつつ若しくは向上させつつ新たな集合を生成する生成装置であって、

前記論理回路内の特定の内部信号線を抽出する特定内部信号線抽出手段と、

前記特定の内部信号線を、当該特定の内部信号線が有する論理値に基づいて区別する特定内部信号線区別手段と、

前記入力ビットにおける論理ビット及び未定値ビットとすべきビットを特定する特定手段とを備え、

前記特定の内部信号線には後段回路が接続されており、

前記特定手段が、

前記特定内部信号線抽出手段により抽出された前記特定の内部信号線であって、前記特定内部信号線区別手段により、前記後段回路を活性化させるそれぞれに指定される論理値を有している第一の特定内部信号線、前記後段回路を非活性化させるそれぞれに指定される論理値とは異なる論理値を有している第二の特定内部信号線、及び、未定値を有している第三の特定内部信号線に区別された内部信号線に関し、

前記与えられた集合に基づき、前記故障モデルを維持しながら又は前記故障検出率を維持しながら若しくは向上させながら、

前記第一の特定内部信号線が有する当該各指定される論理値の少なくとも一つを未定値の状態に変更させること、

前記第二の特定内部信号線には当該異なる論理値の少なくとも一つを維持させて当該維持された異なる論理値を有する第二の特定内部信号線を前記指定される論理値以外の状態とさせること、及び、

前記第三の特定内部信号線には当該未定値の少なくとも一つを維持させ又は前記異なる論理値に変更させて当該維持された未定値を有する又は当該変更された異なる論理値を有する第三の特定内部信号線を前記指定される論理値以外の状態とさせることについて、

少なくともいずれかが可能な、入力ビットにおける論理ビット及び未定値ビットとすべきビットを、

特定する、生成装置。

【請求項 2】

前記後段回路はフリップフロップ回路群であり、前記特定の内部信号線は前記フリップフロップ回路群を制御するための信号線である、請求項 1 記載の生成装置。

【請求項 3】

前記特定の内部信号線に論理値 1 又は論理値 0 を割り当てる割当手段をさらに備え、

前記割当手段が、前記特定手段により特定された未定値ビットとすべきビットを含む入力ビットにおける未定値ビットに対し、

前記第一の特定内部信号線が未定値の状態に変更可能なものについては前記各指定される論理値とは異なる論理値を有するように論理値 1 又は論理値 0 を割り当て、

前記第三の特定内部信号線が維持された未定値を有している場合には前記各指定される論理値とは異なる論理値を有するように論理値 1 又は論理値 0 を割り当てる、請求項 2 記載の生成装置。

【請求項 4】

与えられる集合であって、故障検出対象の論理回路に入力される入力ビットが、論理値 1 若しくは論理値 0 の論理ビットからなるテストベクトルの集合、少なくとも一つのビットが未定値の未定値ビットを含み残余のビットが論理ビットであるテストキューブの集合、又は、テストベクトルとテストキューブとの組み合わせの集合から、前記与えられた集合の故障モデルを維持しつつ又は故障検出率を維持しつつ若しくは向上させつつ新たな集合を生成する生成方法であって、

特定内部信号線抽出手段が、前記論理回路内の特定の内部信号線を抽出する特定内部信号線抽出処理と、

特定内部信号線区別手段が、前記特定の内部信号線を、当該特定の内部信号線が有する論理値に基づいて区別する特定内部信号線区別処理と、

特定手段が、前記入力ビットにおける論理ビット及び未定値ビットとすべきビットを特定する特定処理とを含み、

前記特定の内部信号線には後段回路が接続されており、

前記特定処理において、前記特定手段が、

前記特定内部信号線抽出処理において抽出された前記特定の内部信号線であって、前記特定内部信号線区別処理において、前記後段回路を活性化させるそれぞれに指定される論理値を有している第一の特定内部信号線、前記後段回路を非活性化させるそれぞれに指定される論理値とは異なる論理値を有している第二の特定内部信号線、及び、未定値を有している第三の特定内部信号線に区別された内部信号線に関し、

前記与えられた集合に基づき、前記故障モデルを維持しながら又は前記故障検出率を維持しながら若しくは向上させながら、

前記第一の特定内部信号線が有する当該各指定される論理値の少なくとも一つを未定値の状態に変更させること、

前記第二の特定内部信号線には当該異なる論理値の少なくとも一つを維持させて当該維

10

20

30

40

50

持された異なる論理値を有する第二の特定内部信号線を前記指定される論理値以外の状態とさせること、及び、

前記第三の特定内部信号線には当該未定値の少なくとも一つを維持させ又は前記異なる論理値に変更させて当該維持された未定値を有する又は当該変更された異なる論理値を有する第三の特定内部信号線を前記指定される論理値以外の状態とさせることについて、

少なくともいずれかが可能な、入力ビットにおける論理ビット及び未定値ビットとすべきビットを、

特定する、生成方法。

【請求項 5】

前記後段回路はフリップフロップ回路群であり、前記特定の内部信号線は前記フリップフロップ回路群を制御するための信号線である、請求項 4 記載の生成方法。

10

【請求項 6】

割当手段が、前記特定処理により特定された未定値ビットとすべきビットを含む入力ビットにおける未定値ビットに対し、

前記第一の特定内部信号線が未定値の状態に変更可能なものについては前記各指定される論理値とは異なる論理値を有するように論理値 1 又は論理値 0 を割り当て、

前記第三の特定内部信号線が維持された未定値を有している場合には前記各指定される論理値とは異なる論理値を有するように論理値 1 又は論理値 0 を割り当てる、

割当処理を含む、請求項 5 記載の生成方法。

【請求項 7】

20

請求項 4 から 6 のいずれかに記載の生成方法をコンピュータに実行させることが可能なプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、生成装置、生成方法及びプログラムに関し、特に組み合わせ回路又はフルスキャン順序回路の組み合わせ回路部分等の故障検出の対象回路に対するテストについての生成装置、生成方法及びプログラムに関する。

【背景技術】

【0002】

30

図 9 に示すように、半導体論理回路は、設計、製造、テストの三段階を経て出荷される。ここで、テストとは、製造された半導体論理回路に対して 0 又は 1 の論理値が各論理ビットに定められたテストベクトルを印加し、半導体論理回路からテスト応答を観測し、それを期待テスト応答と比較して良品、不良品の判別を行う。その良品率を歩留りと呼び、歩留りは半導体論理回路の品質、信頼性及び製造コストを大きく左右する。

【0003】

図 10 は、一般的な論理回路におけるフルスキャン順序回路の模式図である。

【0004】

一般に、半導体論理回路は主に順序回路である。順序回路は、アンド (AND) ゲート、ナンド (NAND) ゲート、オア (OR) ゲート、ノア (NOR) ゲート等の論理素子からなる組合せ回路部 1201 と、回路の内部状態を記憶するフリップフロップ 1203 とよりなる。この場合、組合せ回路部 1201 は、外部入力線 (PI)、フリップフロップの出力線である擬似外部入力線 (PPI)、外部出力線 (PO)、フリップフロップの入力線である擬似外部出力線 (PPO) を有する。組合せ回路部 1201 への入力、外部入力線より直接与えられるものと、擬似外部入力線を介して与えられるものからなる。また、組合せ回路部 1201 からの出力は、外部出力線に直接現れるものと、擬似外部出力線に現れるものからなる。

40

【0005】

順序回路の組合せ回路部 1201 をテストするために、組合せ回路部 1201 の外部入力線 (PI) と擬似外部入力線 (PPI) から所要のテストベクトル  $v$  を印加し、組合せ

50

回路部 1201 の外部出力線 PO と擬似外部出力線 PPO からテスト応答  $f(v)$  を観測する必要がある。1 つのテストベクトルは、外部入力線と擬似外部入力線に対応する入力ビットからなる。また、1 つのテスト応答は、外部出力線と擬似外部出力線に対応する出力ビットからなる。

【0006】

しかし、順序回路のフリップフロップ 1203 の出力線（擬似外部入力線）と入力線（擬似外部出力線）は一般に外部より直接アクセスできない。従って、組合せ回路部 1201 をテストするためには、擬似外部入力線の可制御性及び擬似外部出力線の可観測性に問題がある。

【0007】

上述の組合せ回路部 1201 のテストにおける可制御性及び可観測性の問題を解決する主な手法として、フルスキャン設計がある。フルスキャン設計とは、フリップフロップをスキャンフリップフロップに置き換えた上で、それらを用いて 1 本または複数本のスキャンチェーンを形成することである。スキャンフリップフロップの動作はスキャンインネーブル（SE）信号線で制御される。例えば、SE = 0 のとき、従来のフリップフロップと同じ動作をし、クロックパルスが与えられると、組合せ回路部からの値でスキャンフリップフロップの出力値が更新され、また、SE = 1 のとき、同じスキャンチェーンにある他のスキャンフリップフロップと 1 つのシフトレジスタを形成し、クロックパルスが与えられると、外部から新しい値がスキャンフリップフロップにシフトインされると同時に、スキャンフリップフロップに現存の値が外部へシフトアウトされる。一般に、同じスキャンチェーンにあるスキャンフリップフロップは同じスキャンインネーブル（SE）信号線を共有するが、異なるスキャンチェーンのスキャンインネーブル（SE）信号線は同一の場合もあれば異なる場合もある。

【0008】

フルスキャン順序回路の組合せ回路部のテストはスキャンシフトとスキャンキャプチャを繰り返すことによって行われる。スキャンシフトは、スキャンインネーブル（SE）信号が論理値 1 にされているシフトモードで行われる。シフトモードにおいては、1 つまたは複数のクロックパルスが与えられ、外部から 1 つまたは複数の新しい値がスキャンチェーン内のスキャンフリップフロップにシフトインされる。また、それと同時に、そのスキャンチェーン内のスキャンフリップフロップに現存の 1 つまたは複数の値が外部へシフトアウトされる。スキャンキャプチャは、スキャンインネーブル（SE）信号が論理値 0 にされているキャプチャモードで行われる。キャプチャモードにおいては、1 つのスキャンチェーンにあるすべてのスキャンフリップフロップに同時に 1 つのクロックパルスが与えられ、組合せ回路部の擬似外部出力線の値がすべてのスキャンフリップフロップに取り込まれる。

【0009】

スキャンシフトは、擬似外部入力線を介して組合せ回路部 1201 へテストベクトルを印加するためと、擬似外部出力線を介して組合せ回路部 1201 からテスト応答を観測するために用いられる。また、スキャンキャプチャは、組合せ回路部 1201 のテスト応答をスキャンフリップフロップ 1203 に取り込むために用いられる。すべてのテストベクトルに対して、スキャンシフトとスキャンキャプチャを繰り返すことによって、組合せ回路部 1201 をテストすることができる。このようなテスト方式はスキャンテスト方式という。

【0010】

スキャンテスト方式では、組合せ回路部 1201 へのテストベクトルの印加は、外部入力から直接行われる部分と、スキャンシフトによって行われる部分とがある。スキャンシフトによって、任意の論理値を任意のスキャンフリップフロップに設定することができるので、擬似外部入力線の可制御性の問題が解決される。組合せ回路部 1201 からのテスト応答の観測は、外部出力から直接行われる部分と、スキャンシフトによって行われる部分とがある。スキャンシフトによって、任意のスキャンフリップフロップの出力値を観測

10

20

30

40

50

することができるため、擬似外部出力線の可観測性の問題が解決される。このように、スキャンテスト方式においては、自動テストパターン生成 ( A T P G ) プログラムを用いてテストベクトル及び期待テスト応答を求めるだけで十分である。

【 0 0 1 1 】

図 1 1 は、テスト入力とテスト応答との関係を示した模式図である。

【 0 0 1 2 】

図 1 1 において、テストベクトルに未定値 X ( 論理値 1 と論理値 0 のいずれでもよいドントケア ( X ) ) が存在する場合、それによってテスト応答にも未定値 X ( ドントケア ( X ) ) が現れる。未定値としてのドントケアが存在する原因は、1 つ又は複数の故障を検出するために、テスト入力の一部のビットのみに論理値が決まれば十分であることによる。このように故障検出などの所定の目的を達成する上で論理値 1 と論理値 0 のいずれでもよいドントケアとできるものはテストキューブと呼ばれ、A T P G 又はドントケア発見手法によって求められる。ドントケアには、自由に論理値 1 又は 0 を割り当てることができる。テスト入力のビットを論理ビットとすべきか或いはドントケアビットとすべきかについては、例えば特許文献 1 に記載のように本願発明者等によって提案されたものがある。

【 0 0 1 3 】

特許文献 1 に記載の技術は、論理回路の入出力関係に基づく条件によりドントケアビットにしてもよい候補ビット及びドントケアビットにはならない固定ビットを設定し、設定された候補ビットのみから特定されたドントケアビットを含むテストキューブに対して入力ビットと出力ビットからなる複数のビットペア間の関係を考慮してドントケアビットに論理値を割り当てる技術である。

【 0 0 1 4 】

なお、ドントケア抽出を可能としている技術としては他にも本願発明者等によって特許文献 2 に記載の技術もある。

【 0 0 1 5 】

特許文献 2 に記載の技術は、それまでの縮退故障検出を対象とするものとは異なって遅延故障検出を対象にでき、テストパターンの圧縮によるサイズの削減を図り、テスト印加時間を短縮できる技術である。

【 0 0 1 6 】

ここで、テストキューブとテストベクトルについて説明しておく。図 1 2 を参照して、X で表されているドントケアビットを含む  $c_1$  ,  $c_2$  ,  $c_3$  からなるものがテストキューブである。論理値 0 と論理値 1 のいずれかの論理ビットのテストベクトルからなる集合がテストベクトル集合である。

【 0 0 1 7 】

次に、テストキューブの求め方も説明しておく。図 1 3 は、テストキューブの求め方の概念を説明する図であってテストデータ操作の例を説明するための図である。図 1 4 は、テストリラクゼーション ( ドントケア判定 ) の例を簡単に説明するための図である。

【 0 0 1 8 】

図 1 3 を参照して、故障検出などの所定の目的を達成する上で論理値 0 と論理値 1 のいずれでもよいドントケアが X で表されている。テストキューブは、A T P G によるテスト生成中にドントケアビットを生じさせる動的技法と、A T P G 後にテストリラクゼーションと言ってドントケアとできるドントケアビットを見つける静的技法とによって求められる。このようなテストキューブは論理値が定められ ( X - Filling )、最終的には論理値 0 と論理値 1 のいずれかによって値が満たされた論理ビットのテストベクトルからなるテストベクトル集合として得られる。以下では、ドントケアビットへの論理値の割当に関して、キャプチャ時の信号値変化が少なくなるように決められる。なお、上記静的技法は圧縮したテストベクトル集合にも適用でき、テストデータ量は少なくなる。

【 0 0 1 9 】

図 1 4 を参照して、テストキューブ内に含まれるドントケアにできるビットを見つけるテストリラクゼーションには、故障シミュレーション、含意操作、正当化操作が用いられ

10

20

30

40

50

、ドントケアは制約のもとで見つけられるが、一般的にいう制約は特定の故障モデルについて故障検出率を調整することをいう。ここで、通常、60%～90%のビットがドントケアとできる。それに対して以下では故障検出率は変化させない制約を可能としている。なお、当初のテストベクトル集合は動的圧縮或いはランダム割り当てによってコンパクトなテストベクトル集合として与えられることも可能である。

#### 【0020】

さらに、実速度スキャンテストという考えがあり、これはタイミング関連のテスト品質を向上させるために必須とされる。これを実現するために、パスの出発点で遷移がラウンチされ、その応答がシステムの速度でパスの終点でキャプチャされる。実際には、ラウンチオンキャプチャ(LOC)クロッキング方式は実速度スキャンテストに広く用いられている。

10

#### 【0021】

図15はLOCクロッキング方式を説明するための図である。

#### 【0022】

図15を参照して、テストベクトルが $S_L$ を最後とする一連のシフトクロックパルス(L:最長スキャンチェーンの長さ)によってロードされた後で、遷移が最初のキャプチャクロックパルス $C_1$ によって対応するスキャンFFのところからラウンチされる。遷移は $S_L$ によって導入された値と $C_1$ によって取り込まれた値との差によって生じる。なお、遷移ラウンチ( $C_1$ )と応答キャプチャ( $C_2$ )の間のテストサイクルは定格のシステムクロック周期である。

20

#### 【0023】

スキャンテストはタイミング関連のテスト品質を向上させるために不可欠とされるが、その適用可能性はテスト誘発性歩留り損失が大きな課題となっている。このテスト誘発性歩留り損失とは機能的に問題ない集積回路が実速度スキャンテストの間だけに機能しない場合に生じるものと言われる。この問題の主な原因は電源ノイズ、すなわち、IRドロップおよびグラウンドバウンスであり、これらは $C_1$ における過度のラウンチ遷移から生じ、遅延増加を引き起こしてしまう。これまでに電源電圧が10%低下するとパス遅延が30%増大することが示されている文献もある。このことは $C_2$ におけるキャプチャ故障を招き、ひいてはテスト誘発性歩留り損失につながってしまう。ディープサブミクロンで低電力のチップではこの問題は急速に悪化している状況にある。したがって過度の電源ノイズによって誘発される歩留り損失リスクを低減させることが強く望まれる。

30

#### 【0024】

ラウンチ遷移を減らすための従来の技術は以下の3つの技術に基づいている。

#### 【0025】

(1)部分的キャプチャ: $C_1$ (図15)においてキャプチャするFFの数は回路の変更、ワンホットクロッキング、キャプチャクロックスタガリングによって減らすことが可能である。しかし、このアプローチは著しいATPGの変化、テストデータの増加、さらには故障検出率の低下まで引き起こしかねない。

#### 【0026】

(2)低キャプチャ電力ATPG:テストベクトル内の1および0を注意深く生成して、ラウンチ遷移をFFにおける入出力等値化、クロックゲーティング等の技術によって減らすことができる。しかし、このアプローチはテストデータの著しい増加とCPU時間の増大に悩まされうる。

40

#### 【0027】

(3)テストリラクゼーション&X filling:テストリラクゼーションとは論理値が全て特定されたテストベクトル集合から、故障検出率を低下させることなく、ドントケアビット(Xビット)を特定することである。それから、テストリラクゼーションによって得られた部分的に特定されたテストキューブに対して、できるだけ多くのFFの入力値と出力値を等しくするためにX fillingが実行される。このようにして、ラウンチ遷移が削減される。

50

## 【 0 0 2 8 】

図 1 6 は、テストリラクゼーション & X filling 手法の概念を示した図である。

## 【 0 0 2 9 】

テストリラクゼーションについて説明する。図 1 6 に示すように、テストリラクゼーションは全て特定されたテストベクトル集合  $V$  からドントケアビット ( $X$  ビット) を特定して、部分的に特定されたテストキューブ集合  $C$  を、 $V$  の特性のいくつかは  $C$  によって保持されていることを保証しながら、生成するプロセスである。保持される特性の中には縮退故障検出率、遷移遅延故障検出率、さらに遷移遅延故障検出のための全ての活性化パスまでもが含まれる。

## 【 0 0 3 0 】

$X$  ビットはランダム-fill を無効化することによるテスト生成からも直接に得られるが、ATPG 時間およびテストデータ量が増大する。例えば、ランダム-fill を無効化して低電力  $X$  filling のための  $X$  ビットを放置するとテストベクトル数は 144.8% に増加することが示されている。したがって、ランダム-fill を用いて最大限テストコンパクションを適用して、まずはコンパクトな初期 (全て特定された) テストベクトル集合を生成し、続いてテストリラクゼーションを用いて (部分的に特定された) テストキューブを生成するのが望ましい。こうして得られるコンパクトな最終テストベクトル集合には  $X$  filling を実行した後の付随的な利益もある。

## 【 0 0 3 1 】

次に、 $X$  filling について説明する。図 1 6 に示すように、 $X$  filling はあるテストキューブ内の  $X$  ビットにある目的のために論理値を割り当てるプロセスである。例えば、 $X$  filling はラウンチ遷移 ( $L S A$ ) を削減するために用いられる。ここで、 $L S A$  は  $C_1$  のところで起こる (図 1 5 参照)。多数の低  $L S A$  の  $X$  filling 手法が提案されている。

## 【 0 0 3 2 】

図 1 7 はそれらの 1 つである JP-fill と呼ばれるものを示す図である。

## 【 0 0 3 3 】

図 1 7 では、テストキューブは  $c = \langle 10XX \rangle$  であり、組み合わせ部分の論理関数は  $F$  である。したがって、 $\langle c : PPI \rangle = \langle 0XX \rangle$  であり、 $\langle F(c) : PPO \rangle = \langle XXX \rangle$  である。まず、正当化 (図 1 7 の丸 1) を実行して  $p_1$  が 0 なので  $p_2$  に 0 を設定するよう試みる。続いて、形式  $X X$ 、の  $q_1$   $q_2$  および  $r_1$   $r_2$  のビットペアについては、各  $PP0$  の  $X$  ビットの 0 になる確率および 1 になる確率を算出するには、各入力  $X$  ビットの 0 になる確率および 1 になる確率に 0.50 を設定して確率伝搬を実行する。 $q_2$  が 0 である確率 (0.93) は  $q_2$  が 1 である確率 (0.07) よりも非常に大きいので、 $q_1$  に 0 を割り当てるのは理にかなっている (図 1 7 の丸 2)。しかし、 $r_2$  が 0 である確率 (0.48) は  $r_2$  が 1 である確率 (0.51) に近いので、 $r_1$  については何も決定されない (図 1 7 の丸 3)。この場合、3 値論理シミュレーションが実行され、JP fill のもう一つのパス (図 1 7 の丸 4) が正当化および / あるいは確率伝搬を用いて実行される。要するに、JP fill はスケラビリティを確率伝搬によって向上させると共に、正当化および複数のパスを用いて  $X$  filling の効率を向上させる。このようにして JP fill は効率とスケラビリティをバランスよく実現する。

## 【 0 0 3 4 】

ところで、クロックゲーティング方式の一例を図 1 8 に示すが、クロックゲーティング方式は最も広く実際に用いられている消費電力管理メカニズムである。なお、回路にはクロックゲーティングブロックが複数含まれていてもよい。図 1 8 には  $i$  番目のクロックゲーティングブロックであってスキャンテスト用に強化されているものを示している。シフトモード ( $SE = 1$ ) では、シフト操作が適切に実行されるように全ての FF が常にクロック  $CK$  によって駆動されている。クロック制御信号 ( $EN_i$ ) は  $CK$  の立ち上がりエッジのところで生成され、キャプチャモード ( $SE = 0$ ) の制御を引き継ぐ。 $GEN_i$  は  $CK$  と  $AND$  素子につながれて直接  $FF$  に接続されるゲートクロック  $GCK_i$  を生成する。クロックゲーティングはキャプチャモードにおいてラウンチ遷移を削減するのに用いられる。このためには最終シフトパルス  $S_L$  において  $EN_i$  を 0 に設定すればよい。こうし

10

20

30

40

50

て図19に示すように、ラウンチキャプチャクロックパルス(図15の $C_1$ )は抑えられ、 $GCK_i$ に制御されている全てのFF、すなわち図18に示した $FF_1^i \sim FF_p^i$ はキャプチャしない。すなわち、FFにおけるラウンチ遷移がまとめて削減されたことになる。

【0035】

ここで、以下の定義を行う。あるFFのクロックがゲートクロックである場合、そのFFはクロックゲートFFという。そうでない場合、そのFFは非クロックゲートFFという。同じゲートクロックで制御されているFF群はクロックゲートFF群という。あるクロックツリーを通して接続されている全てのFFは1つのクロックゲートFF群と考えられる。

【0036】

ATPGの観点から、クロックゲーティングは次の2つの基本的なアプローチによって用いられる。

【0037】

アプローチ1(検出指向):テスト生成を実行する際には、より多くのFFを遷移故障検出におけるラウンチおよびキャプチャに使用可能にするために、FFへの信号をできるだけアクティブ化する(すなわちクロックゲーティングを無効化する)ように実行する。多くの商業的ATPGシステムはこのアプローチを陰に陽に用いている。こうすることでラウンチ遷移がより活発になる代わりに、テストベクトル集合が小さくなり、故障検出率が向上し、テスト印加時間が小さくなる。

【0038】

アプローチ2(削減指向):テスト生成においてクロックゲーティングはラウンチ遷移を削減するために積極的に用いられる。しかし、こうすることで遷移故障検出におけるラウンチおよびキャプチャに使用可能なFFの数は減少し、テストベクトル数が増加し、テスト印加時間が増大し、さらにはある条件下では故障検出率の損失まで起こる。

【0039】

このようにFF群が活性化されるか或いは活性化されないかを決定できる信号GCKがあるが、この信号GCKはクロックゲーティング回路を制御する制御信号 $EN_i$ によってその値が決定される。この制御信号 $EN_i$ (或いは信号GCK)の値をコントロールすることができれば、FF群の非活性化による大幅な消費電力削減も可能となる。

【0040】

クロックゲーティング回路の活性化と非活性化をコントロールして消費電力削減を図ろうとしたものには非特許文献1がある。

【先行技術文献】

【特許文献】

【0041】

【特許文献1】特開2007-155339号公報

【特許文献2】国際公開WO2008/001818

【非特許文献】

【0042】

【非特許文献1】R. Illman、外2名著、「ATPG power reduction using clock gate “default” constraints.」First International Workshop on the Impact of Low-Power Design on Test and Reliability(fringe to ETS 2008), Proceedings LPonTR 2008, 29 May 2008, Hotel Majestic Pallanza, Lago Maggiore, Italy

【発明の概要】

【発明が解決しようとする課題】

【0043】

しかしながら、特許文献1に記載のような従来のいずれの技術においても、上記したように擬似外部入力線の可制御性及び擬似外部出力線の可観測性の問題解決の流れもあって、入力ビットと出力ビットとの関係を考慮しているに留まっていた。また、特許文献2に

10

20

30

40

50



記載のような従来のいずれの技術においても、ドントケア抽出を可能としているが、入力ビットと出力ビットを超える着眼、すなわち外部入力線（擬似外部入力線）及び外部出力線（擬似外部出力線）を超える着眼はなされていなかった。

【0044】

さらに、テストリラクゼーション&X filling手法というアプローチは重大な制限に悩まされる。すなわち、テストベクトル集合から特定されたXビットの数が少なかった場合、ラウンチ遷移を減らす効率が不十分になりうる。そのようなXビット不足はテストコンパクションあるいはテストコンプレッションによって引き起こされうる。具体的に説明すると、図20に実用的な回路の一例（ゲート数60万および遷移遅延テストベクトル数600）についてのグラフを示すが、この回路においてテスト生成の初期段階に得られたテストベクトルはXビットが少ない（60%にも満たない）。

10

【0045】

また、非特許文献1の技術は故障検出率と消費電力削減を両立するものではなく、故障検出率を維持して消費電力削減を行うとすれば追加データが必要となり、テストデータ量に関するサイズが大きくなってしまいう問題がある。つまり、FF群を単純に非活性化するとすれば消費電力削減は可能となるが、テストでは故障検出という大前提があり、故障検出率といった最終的に得られるテストベクトルの特性は極めて重要であるため、上記アプローチ1の検出指向により故障検出率をあげようとするればFF群を活性化させる傾向に陥ってしまう。その結果、故障検出率のような特性を考慮しながらも、上記アプローチ2の削減指向によってテストデータ量に関するサイズの増加を起こさず、さらに消費電力削減も可能とするような提案が望まれていた。

20

【0046】

特に、上記したように、実速度スキャンテストでは、過度のラウンチ遷移による電源ノイズに起因する歩留り損失リスクを受けやすく、テストデータ量、故障検出率、パフォーマンス、回路設計には影響を及ぼさず、テストコンプレッションの場合のように入力ビット中のドントケアビットが少ない場合であっても、ラウンチ遷移ひいては歩留り損失リスクを減らすことが期待されている。

【0047】

ゆえに、本願に係る発明は、内部信号線に着眼しつつ、実速度スキャンテストであっても、テストデータ量、故障検出率、パフォーマンス、回路設計には影響を及ぼさず、テストコンプレッションの場合のように入力ビット中の未定値（ドントケア）ビットが少ない場合であっても、ラウンチ遷移ひいては歩留り損失リスクを減らすことが可能で、さらにテストにおける消費電力の削減も可能とすることを目的とする。

30

【課題を解決するための手段】

【0048】

請求項1に係る発明は、与えられる集合であって、故障検出対象の論理回路に入力される入力ビットが、論理値1若しくは論理値0の論理ビットからなるテストベクトルの集合、少なくとも一つのビットが未定値の未定値ビットを含み残余のビットが論理ビットであるテストキューブの集合、又は、テストベクトルとテストキューブとの組み合わせの集合から、前記与えられた集合の故障モデルを維持しつつ又は故障検出率を維持しつつ若しくは向上させつつ新たな集合を生成する生成装置であって、前記論理回路内の特定の内部信号線を抽出する特定内部信号線抽出手段と、前記特定の内部信号線を、当該特定の内部信号線が有する論理値に基づいて区別する特定内部信号線区別手段と、前記入力ビットにおける論理ビット及び未定値ビットとすべきビットを特定する特定手段とを備え、前記特定の内部信号線には後段回路が接続されており、前記特定手段が、前記特定内部信号線抽出手段により抽出された前記特定の内部信号線であって、前記特定内部信号線区別手段により、前記後段回路を活性化させるそれぞれに指定される論理値を有している第一の特定内部信号線、前記後段回路を非活性化させるそれぞれに指定される論理値とは異なる論理値を有している第二の特定内部信号線、及び、未定値を有している第三の特定内部信号線に区別された内部信号線に関し、前記与えられた集合に基づき、前記故障モデルを維持しな

40

50

がら又は前記故障検出率を維持しながら若しくは向上させながら、前記第一の特定内部信号線が有する当該各指定される論理値の少なくとも一つを未定値の状態に変更させること、前記第二の特定内部信号線には当該異なる論理値の少なくとも一つを維持させて当該維持された異なる論理値を有する第二の特定内部信号線を前記指定される論理値以外の状態とさせること、及び、前記第三の特定内部信号線には当該未定値の少なくとも一つを維持させ又は前記異なる論理値に変更させて当該維持された未定値を有する又は当該変更された異なる論理値を有する第三の特定内部信号線を前記指定される論理値以外の状態とさせることについて、少なくともいずれかが可能な、入力ビットにおける論理ビット及び未定値ビットとすべきビットを、特定する、ものである。

【0049】

請求項2に係る発明は、請求項1において、前記後段回路はフリップフロップ回路群であり、前記特定の内部信号線は前記フリップフロップ回路群を制御するための信号線であるものである。

【0050】

請求項3に係る発明は、請求項2において、前記特定の内部信号線に論理値1又は論理値0を割り当てる割り当て手段をさらに備え、前記割り当て手段が、前記特定手段により特定された未定値ビットとすべきビットを含む入力ビットにおける未定値ビットに対し、前記第一の特定内部信号線が未定値の状態に変更可能なものについては前記各指定される論理値とは異なる論理値を有するように論理値1又は論理値0を割り当て、前記第三の特定内部信号線が維持された未定値を有している場合には前記各指定される論理値とは異なる論理値を有するように論理値1又は論理値0を割り当てる、ものである。

【0051】

請求項4に係る発明は、与えられる集合であって、故障検出対象の論理回路に入力される入力ビットが、論理値1若しくは論理値0の論理ビットからなるテストベクトルの集合、少なくとも一つのビットが未定値の未定値ビットを含み残余のビットが論理ビットであるテストキューブの集合、又は、テストベクトルとテストキューブとの組み合わせの集合から、前記与えられた集合の故障モデルを維持しつつ又は故障検出率を維持しつつ若しくは向上させつつ新たな集合を生成する生成方法であって、特定内部信号線抽出手段が、前記論理回路内の特定の内部信号線を抽出する特定内部信号線抽出処理と、特定内部信号線区別手段が、前記特定の内部信号線を、当該特定の内部信号線が有する論理値に基づいて区別する特定内部信号線区別処理と、特定手段が、前記入力ビットにおける論理ビット及び未定値ビットとすべきビットを特定する特定処理とを含み、前記特定の内部信号線には後段回路が接続されており、前記特定処理において、前記特定手段が、前記特定内部信号線抽出処理において抽出された前記特定の内部信号線であって、前記特定内部信号線区別処理において、前記後段回路を活性化させるそれぞれに指定される論理値を有している第一の特定内部信号線、前記後段回路を非活性化させるそれぞれに指定される論理値とは異なる論理値を有している第二の特定内部信号線、及び、未定値を有している第三の特定内部信号線に区別された内部信号線に関し、前記与えられた集合に基づき、前記故障モデルを維持しながら又は前記故障検出率を維持しながら若しくは向上させながら、前記第一の特定内部信号線が有する当該各指定される論理値の少なくとも一つを未定値の状態に変更させること、前記第二の特定内部信号線には当該異なる論理値の少なくとも一つを維持させて当該維持された異なる論理値を有する第二の特定内部信号線を前記指定される論理値以外の状態とさせること、及び、前記第三の特定内部信号線には当該未定値の少なくとも一つを維持させ又は前記異なる論理値に変更させて当該維持された未定値を有する又は当該変更された異なる論理値を有する第三の特定内部信号線を前記指定される論理値以外の状態とさせることについて、少なくともいずれかが可能な、入力ビットにおける論理ビット及び未定値ビットとすべきビットを、特定する、ものである。

【0052】

なお、上記請求項1及び4における特定処理において、第二の特定内部信号線について「少なくとも一つ」と表現しているが、できるだけ多くについて異なる論理値を維持させ

10

20

30

40

50

ることが好ましく、全てについて異なる論理値を維持させることがより好ましい。同様に、上記特定処理において、第三の特定内部信号線についても「少なくとも一つ」と表現しているが、できるだけ多くについて未定値を維持させ又は異なる論理値に変更させることが好ましく、全てについて未定値を維持させ又は異なる論理値に変更させることがより好ましい。また、上記特定処理について、3つの処理に関して「少なくともいずれかが可能な」としているが、未定値が発生していない場合には第三の特定内部信号線の処理ができない場合もあり得るが、3つの処理が全て行われることが好ましい。

【0053】

請求項5に係る発明は、請求項4において、前記後段回路はフリップフロップ回路群であり、前記特定の内部信号線は前記フリップフロップ回路群を制御するための信号線であるものである。

10

【0054】

請求項6に係る発明は、請求項5において、割当手段が、前記特定処理により特定された未定値ビットとすべきビットを含む入力ビットにおける未定値ビットに対し、前記第一の特定内部信号線が未定値の状態に変更可能なものについては前記各指定される論理値とは異なる論理値を有するように論理値1又は論理値0を割り当て、前記第三の特定内部信号線が維持された未定値を有している場合には前記各指定される論理値とは異なる論理値を有するように論理値1又は論理値0を割り当てる、割当処理を含む、ものである。

【0055】

なお、請求項2、3、5、又は、6における前記フリップフロップ回路群を制御するための信号線としては、クロック制御信号線が挙げられるが、ゲートクロック信号線として捉えてもよい。

20

【0056】

請求項7に係る発明は、請求項4から6のいずれかに記載の生成方法をコンピュータに実行させることが可能なプログラムである。

【発明の効果】

【0057】

本願に係る発明によれば、特定の内部信号線を第一、第二及び第三の特定内部信号線に区別し、第一の特定内部信号線については各指定される論理値を未定値の状態に変更させることが可能で、第二の特定内部信号線及び第三の特定内部信号線については各指定される論理値以外の状態とさせることが可能となるように、入力ビットにおける論理ビット及び未定値ビットを特定するので、与えられた集合のサイズを変更することもなく、すなわち特性の低下を抑止しながら、特に第一の特定内部信号線の状態をコントロールできる新しいテストキューブを含む新しい集合を生成できる。

30

【0058】

その結果、例えば特定の内部信号線がフリップフロップ群の活性化と非活性化をコントロールする信号線であれば、できるだけ多くの特定の内部信号線が非活性化することを選択することで数多くのフリップフロップを非活性にできて、テストにおける消費電力の削減が可能になる。

【0059】

40

また、実速度スキャンテストであっても、テストデータ量、故障検出率、パフォーマンス、回路設計には影響が及ぼされず、テストコンプレッションの場合のように入力ビット中のドントケアビットが少ない場合であっても、ラウンチ遷移ひいては歩留り損失リスクを減らすことが可能となる。

【図面の簡単な説明】

【0060】

【図1】本発明の実施の形態にかかるテストキューブ集合の生成方法を実施可能な変換装置を示したブロック図である。

【図2】図1の変換装置の動作を具体的に説明するための図である。

【図3】CTXの基本的な考え方を説明するための図である。

50

【図4】CTXの基本概念を示したフロー図である。

【図5】CTXのためのテスト生成を目的とする、m個のクロックゲーティングブロックを含む回路のモデルを示す図である。

【図6】制限付きテストリラクゼーションを説明するための図である。

【図7】実験結果を説明するための二つの表を示す図である。

【図8】パーシャルスキャン回路を説明するための図である。

【図9】半導体論理回路が、設計、製造、テストの三段階を経て出荷されることを示した図である。

【図10】一般的な論理回路におけるフルスキャン順序回路の模式図である。

【図11】テスト入力とテスト応答との関係を示した模式図である。

10

【図12】テストキューブとテストベクトルについて説明するための図である。

【図13】テストキューブの求め方の概念を説明する図であってテストデータ操作の例を説明するための図である。

【図14】テストリラクゼーションの例を簡単に説明するための図である。

【図15】LOCクッキング方式を説明するための図である。

【図16】テストリラクゼーション&X filling手法の概念を示した図である。

【図17】多数の低LSAのX filling手法の1つであるJP-fillと呼ばれるものを示す図である。

【図18】クロックゲーティング方式の一例を示した図である。

【図19】図17のクロックゲーティング回路に関する信号のタイミングチャートを示した図である。

20

【図20】実用的な回路の一例（ゲート数60万および遷移遅延テストベクトル数600）についてのテストベクトル数と未定値Xの割合を示したグラフである。

【発明を実施するための形態】

【0061】

図1は、本発明の実施の形態にかかるテストキューブ集合の生成方法を実施可能な変換装置を示したブロック図である。

【0062】

図1を参照して、この変換装置1は、故障検出対象の論理回路内の内部信号線の中から特定の内部信号線を抽出する特定内部信号線抽出部3と、抽出された特定内部信号線を後述するように区別する特定内部信号線区別部5と、入力ビットにおける入力未定値ビット及び入力論理ビットを特定する特定部7と、特定された入力未定値ビットを含む入力ビットにおける未定値ビットに論理値1又は論理値0を割り当てる割当部9とを備える。特定部7は、入力未定値ビット特定部11と、入力論理ビット特定部13とを備える。このような構成を有する変換装置1は、入力ビットに関しての与えられた集合の一例である初期テストベクトル集合15から新たな集合の一例としてのテストキューブ集合17を生成し、さらにテストキューブ集合17から最終テストベクトル集合19に変換してテストとして故障検出対象の論理回路へ印加される集合を生成できる。

30

【0063】

ここで、特定内部信号線抽出部3が抽出する特定の内部信号線としては、例えば、後段回路としてのフリップフロップ回路群の活性化と非活性化を決定できるクロックゲーティング回路へ入力されるクロック制御信号線が挙げられる。そして、特定内部信号線区別部5は、上記のようなクロック制御信号線が論理値1(0)の場合と論理値0(1)の場合とを区別し、例えば、論理値1を持つクロック制御信号線を第一の特定内部信号線とし、論理値0を持つクロック制御信号線を第二の特定内部信号線とする。なお、与えられる集合に未定値(ドントケア)ビットが含まれている場合には、特定内部信号線区別部5は、未定値Xを持つクロック制御信号線を第三の特定内部信号線としてさらに区別できるものとする。

40

【0064】

図2は、図1の変換装置の動作を具体的に説明するための図である。

50

## 【 0 0 6 5 】

図2 (A) に示したものは図1の初期テストベクトル集合15の一例である。初期テストベクトル集合は、例えば、自動テストパターン生成 (ATPG) プログラムにより生成される。これに対して、特定内部信号線抽出部3及び特定内部信号線区別部5の動作により入力ビットに関する制約ビット集合21が決定される。この制約ビット集合21の中で「\*」がついたビットは未定値 (ドントケア) Xとできることが期待されるビットであり、「」がついたビットは論理値1と論理値0のいずれかである論理ビットとされるビットである。特定部7は、フリップフロップ群を活性化させる例えば論理値1 (指定される論理値) を有している上記した第一のクロック制御信号線、フリップフロップ群を非活性化させる例えば論理値0 (指定される論理値とは異なる論理値) を有している第二のクロック制御信号線に関し、故障シミュレーションにより、故障検出率といった特性の低下を抑止しながら第一のクロック制御信号線が有する論理値1を未定値の状態に変更させること、及び、第二のクロック制御信号線には論理値0を維持させて論理値1ではない状態とさせることの両者を可能な限り行うように、入力ビットにおける論理ビット及び未定値ビットを特定する特定処理を行う。その結果、図2 (C) に示すような、で囲まれて特定されている未定値ビットにすべきビット (図1の入力未定値ビット特定部11により得られる。) とで囲まれて特定されている論理ビット (図1の入力論理ビット特定部13により得られる。) が示された集合が得られる。これによって、図2 (D) に示すようなテストキューブ集合17が新たな集合として生成される。新たなテストキューブ集合17は、初期テストベクトル集合との関係ではテストデータ量というサイズの増加はなく、故障検出率というような特性の低下は抑えられつつ第一のクロック制御信号線を、例えば論理値1から論理値0のようにフリップフロップ群を活性化させる論理値から非活性化させる論理値に変更させえる集合である。最後に、割当部9が動作して、特定処理により特定された未定値ビットにすべきビットを含む入力ビットにおける未定値ビットに対し、第一のクロック制御信号線が未定値の状態に変更可能なものについてはフリップフロップ群を非活性化させる例えば論理値0を有するように論理値1又は論理値0を割り当てる割当処理を行い、図2 (E) に示すような最終テストベクトル集合19が得られる。

10

20

## 【 0 0 6 6 】

なお、与えられる集合に未定値 (ドントケア) ビットが含まれている場合には、特定内部信号線区別部5は、未定値Xを持つクロック制御信号線を第三の特定内部信号線としてさらに区別できるため、特定部7は、第三のクロック制御信号線には当該未定値を維持させ又は例えば論理値0 (指定される論理値とは異なる論理値) に変更させて論理値1ではない状態とさせることを可能な限り行って、入力ビットにおける論理ビット及び未定値ビット (未定値ビットにすべきビットを含む) を特定する特定処理を行えばよい。そして、割当部9は、第三のクロック制御信号線が維持された未定値を有している場合にはフリップフロップ群を非活性化させる例えば論理値0を有するように論理値1又は論理値0を割り当てる割当処理を行えばよい。

30

## 【 0 0 6 7 】

また、上記の説明では、特定の内部信号線としてクロック制御信号線を一例として挙げたが、それ以外の内部信号線を特定内部信号線としてもよく、全て同一種類の内部信号線を特定内部信号線とする必要は必ずしもない。

40

## 【 0 0 6 8 】

さらに、上記では特定内部信号線の一例としてクロック制御信号線を挙げ、クロック制御信号線が有する論理値として後段のフリップフロップ群を活性化する場合の論理値を1、非活性化する場合の論理値を0としたが、クロック信号との出力をANDとするかORとするかによっても異なり、それぞれで指定される論理値を持つものを第一の特定内部信号線とし、それぞれで指定される論理値とは異なる論理値を持つものを第二の特定内部信号線とすればよい。

## 【 0 0 6 9 】

さらに、第一の内部信号線と第二の内部信号線と第三の内部信号線の全てに関して上記

50

した特定処理が行われる必要はなく、第一の内部信号線のみ、第二の内部信号線のみ、第三の内部信号線のみ、或いはそれらの間の組み合わせに関して上記した特定処理が行われてもよい。加えて、例えば第一の内部信号線が複数ある場合には少なくともその一つに関して上記の特定処理が行われるものであってもよく、これに関しては第二、第三の内部信号線に関するも同様である。

【 0 0 7 0 】

以下、図 1 及び図 2 を用いて説明した本願発明に関してさらに説明する。

【 0 0 7 1 】

[ C T X (Clock-Gating-Based Test Relaxation) の基本概念 ]

図 3 は、C T X の基本的な考え方を説明するための図である。図 3 に示す基本的な考え方は、クロックゲーティングを巧みに用いたテストリラクゼーション & X filling である。図 3 ( A ) では 4 つのフリップフロップ ( 以下、 F F ) が E N A と E N B の 2 つのクロック制御信号によって制御されており、 E N A と E N B の値は初期テストベクトルに対しては 1 である。このことからすべての F F がアクティブ、すなわち、 C <sub>1</sub> ( 図 1 5 ) においてキャプチャすることになり、結果として 4 つの初期ラウンチ遷移が生じる。C T X は次の 2 つのステージからなる。

10

【 0 0 7 2 】

< ステージ 1 ( クロック無効化 ) >

図 3 ( B ) に示されているように、まずテストリラクゼーションを実行してできるだけ多くの有効化クロック制御信号 ( 値 = 1 ) をニュートラル信号 ( 値 = X ) に置き換え、次に図 3 ( C ) に示されているように、X filling を実行してできるだけ多くのニュートラルクロック制御信号を無効化信号 ( 値 = 0 ) に置き換える。無効化クロック制御信号 ( 図 3 ( C ) の F F <sub>1</sub> および F F <sub>2</sub> ) に制御されたすべての F F が非アクティブなので、ラウンチ遷移は効率よくまとめて減らされる。このステージ 1 の処理が、図 1 の変換装置 1 により行われる。

20

【 0 0 7 3 】

< ステージ 2 ( F F 静止化 ) >

テストリラクゼーションを実行してできるだけ多くのアクティブ遷移 F F ( 入力値 出力値、すなわち、図 3 ( C ) の F F <sub>3</sub> および F F <sub>4</sub> ) をニュートラル F F ( 入力値または出力値 = X、すなわち、図 3 ( D ) の F F <sub>3</sub> および F F <sub>4</sub> ) に置き換え、次に X filling を実行してニュートラル F F の入力値と出力値を等値化することで、できるだけ多くのニュートラル F F を非遷移 F F ( 入力値 = 出力値、すなわち、図 3 ( E ) の F F <sub>4</sub> ) に置き換える。このようにして、個々の F F におけるラウンチ遷移の数が減らされる。このステージ 2 の処理は、本願発明者等によって他の出願にて提案された技術等が用いられる。

30

【 0 0 7 4 】

C T X 方式の主な貢献は下記の通りである。( 1 ) クロック無効化に基づくテストリラクゼーション & X filling : C T X は実速度スキャンテストにおけるクロックゲーティングの省電力可能性を完全に探索する。( 2 ) F F 静止化に基づくテストリラクゼーションおよび X filling : C T X はアクティブのままの F F の入力値と出力値をできるだけ一致させて個々の F F におけるラウンチ遷移をさらに減らす。( 3 ) クロックゲーティングの非干渉的用法 : テストデータ操作を通じてクロックゲーティングを用いることで、C T X は A T P G の変化もテストデータの増加も故障検出率の減少も一切引き起こさない。( 4 ) X ビット効率 : クロック無効化および F F 静止化によって C T X は X ビットがほとんどなくともラウンチ遷移を著しく減らす。

40

【 0 0 7 5 】

[ 問題の形式化 ]

実速度スキャンテストにおけるクロックゲーティングに基づくラウンチ遷移を減少するという課題は以下のようなになる。 V <sub>initial</sub> をあるテストベクトル集合と仮定する。以下の条件下で新しいテストベクトル集合 V <sub>final</sub> を見つけよ。( 1 ) V <sub>final</sub> の故障検出率は V <sub>initial</sub> の故障検出率より小さくはない。( 2 ) V <sub>final</sub> のサイズは V <sub>initial</sub> のサイズと等しい

50

。(3)  $V_{final}$  のピークラウンチ遷移はクロック無効化およびFF静止化を用いて  $V_{initial}$  のピークラウンチ遷移よりもできるだけ小さくする。

【0076】

[基本概念]

上記の課題を解決するために、CTX (Clock - Gating - Based Test Relaxation and X - Filling) を提案する。以下はCTXの手順を述べる上で有用な用語であり、以下のように定義する。

【0077】

定義A：入力ベクトルにおけるクロック制御信号ENの値が論理値1(0)であるとき、ENは有効である(無効である)という。値がXであるとき、ENはニュートラルであるという。例えば、図3(A)(B)(C)における  $EN_A$  はそれぞれ、有効な、ニュートラルな、無効なクロック制御信号である。

10

【0078】

定義B：FFあるいはクロックゲートFF群へのクロックが有効(無効)であるとき、そのFFあるいはそのクロックゲートFF群はアクティブ(非アクティブ)であるという。例えば、図3(C)における  $FF_1$  および  $FF_2$  ( $FF_3$  および  $FF_4$ ) は非アクティブ(アクティブ)FFである。クロックゲートFF群  $\{FF_1, FF_2\}$  ( $\{FF_3, FF_4\}$ ) は非アクティブ(アクティブ)である、ともいう。

【0079】

定義C：FFの入力値と出力値が同じ(異なる)論理値であるとき、そのFFは非遷移FF(遷移FF)であるという。FFの入力値あるいは出力値がXであるとき、そのFFはニュートラルFFという。例えば、図3(D)の  $FF_3$  および  $FF_4$  は両方ともニュートラルFFである。図3(E)において、 $FF_3$  は遷移FFであり、 $FF_4$  は非遷移FFであり、両方ともアクティブである。

20

【0080】

CTXは以下の2つの見解に基づいている。

【0081】

見解1：全ての有効クロック制御信号が実際に有効である必要はない。このことが起こるのは、あるテストベクトル内の故障を検出するためにクロック制御信号を有効にしたものの、後にその故障がそのクロック制御信号が有効である必要のない他のテストベクトルによって検出されるかもしれない場合である。

30

【0082】

見解2：故障検出のためにはあるFFのみがアクティブでさえあればよい場合であっても、同じクロックゲートFF群にあるFFは同じクロック制御信号を共有しているので、他の全てのFFもアクティブでなければならない。この結果、クロックゲーティングが粗い場合には特に、不必要な遷移が起こる。

【0083】

図4は、CTXの基本概念を示したフロー図である。

【0084】

1番目の見解は図4に示したCTXのステージ1(Clock - Disabling)に通じている。テストリラクゼーションを実行してできるだけ多くの有効クロック制御信号をニュートラル信号に置き換え、次にX fillingを実行してできるだけ多くのニュートラルクロック制御信号を無効信号に置き換える。このようにして、クロックゲーティングをまとめて削減する能力のおかげで、ラウンチ遷移はクロックゲートFF群レベルで効率よく削減される。図1の変換装置1の動作が実行される内容である。

40

【0085】

2番目の見解は図14に示したCTXのステージ2(FF静止化)に通じている。テストリラクゼーションを実行してできるだけ多くのアクティブ遷移FFをニュートラルFFに置き換え、次にX fillingを実行してできるだけ多くのニュートラルFFを非遷移FFに置き換える。このようにして、ラウンチ遷移は個々のFFレベルでさらに削減される。

50

## 【 0 0 8 6 】

## [ 回路モデル ]

図5は、CTXのためのテスト生成を目的とする、m個のクロックゲーティングブロックを含む回路のモデルを示す図である。

まず、図15に示すように全て特定されたテストベクトルは最終シフトパルス $S_L$ の立ち上がりエッジの時にロードされる。 $S_L$ はFFの出力値に対応するPPI部< :PPI >および初期入力値に対応するPI部< :PI >からなる。クロック制御論理回路の組み合わせは、図5に示されているもののよう、m個のクロックゲーティングブロックに対応するm個のクロック制御信号 $EN_1$ 、 $EN_2$ 、 $\dots$ 、 $EN_m$ を生じさせる。クロック制御信号はそれぞれ無効かもしれず有効かもしれず、対応するクロックゲートFF群がラウンチキャプチャパルス $C_i$ 時にアクティブであるか否かを決定する。

10

## 【 0 0 8 7 】

## [ CTXの手順 ]

上記した図4に示したCTXの基本概念に基づいてCTXの手続きは次のように記述できる。

## 【 0 0 8 8 】

入力： $V^0 = \{ v_i^0 \mid i = 1, 2, \dots, n \}$  // 初期テストベクトル集合

出力： $V^2 = \{ v_i^2 \mid i = 1, 2, \dots, n \}$  // 最終テストベクトル集合

## 【 0 0 8 9 】

<ステージ1 (クロック無効化)>

20

(1-1)  $i = 1, 2, \dots, n$ に対して、 $RS(v_i^0) = \{ v_i^0 \text{下で少なくとも1つの有効クロック制御信号から到達可能な } v_i^0 \text{内のビット} \}$ とする。

(1-2)  $T^1 = RS(v_1^0) \cap RS(v_2^0) \cap \dots \cap RS(v_n^0)$ とする。

(1-3)  $V^0$ に制限付きテストリラクゼーションを実行して、 $V^0$ の故障検出率を保持しつつ、 $T^1$ 内のできるだけ多くのビットをXビットに置き換える。得られた部分的に特定されたテストキューブを $C^1 = \{ c_i^1 \mid i = 1, 2, \dots, n \}$ とする。

(1-4)  $i = 1, 2, \dots, n$ に対して、 $c_i^1$ 内の各いくつかのXビットに対して適切な論理値を割り当てることで $c_i^1$ 下の各ニュートラルクロック制御信号に対して0を正当化しよう試みる。

(1-5)  $i = 1, 2, \dots, n$ に対して、 $c_i^1$ 内に残っているXビットに対して低LSAのX fillingを実行する。得られた全て特定されたテストベクトルの集合を $V^1 = \{ v_i^1 \mid i = 1, 2, \dots, n \}$ とする。

30

## 【 0 0 9 0 】

<ステージ2 (FF静止化)>

(2-1)  $i = 1, 2, \dots, n$ に対して、 $RF(v_i^1) = \{ v_i^1 \text{下でアクティブ遷移FFに対応する } v_i^1 \text{内のビット} \}$ とする。

(2-2)  $T^2 = RF(v_1^1) \cap RF(v_2^1) \cap \dots \cap RF(v_n^1)$ とする。

(2-3)  $V^1$ に制限付きテストリラクゼーションを実行して、(1)  $V^1$ の故障検出率および(2)全てのクロック制御信号の論理値を保持しつつ、 $T^2$ 内のできるだけ多くのビットをXビットに置き換える。得られた部分的に特定されたテストキューブを $C^2 = \{ c_i^2 \mid i = 1, 2, \dots, n \}$ とする。

40

(2-4)  $i = 1, 2, \dots, n$ に対して、 $c_i^2$ 内のXビットに対して低LSAのX fillingを実行する。得られた全て特定されたテストベクトルの集合を $V^2 = \{ v_i^2 \mid i = 1, 2, \dots, n \}$ とする。

## 【 0 0 9 1 】

## [ 制限付きテストリラクゼーション ]

CTXのキーとなる操作は、全て特定されたテストベクトル集合Vに実行される制限付きリラクゼーションであり、(1) Vの故障検出率および(2) S内のターゲット列の論理値を保持したまま、T内のできるだけ多くのターゲットビットをXビットに置き換えるために実行する。結果として部分的に特定されたテストキューブCが得られる。以上のこと

50



は図6に示されており、\*印はターゲットビットの位置を表している。ステップ(1-3)では $S =$  を仮定することが可能であり、ここでの制限付きテストリラクゼーションでは故障検出率のみが保持される必要がある。

【0092】

[制限付きテストリラクゼーションの手順]

図6は、制限付きテストリラクゼーションの手順を説明するための図である。

【0093】

まず、図6において、入力 $V$ は全て特定されたテストベクトル集合である。 $T$ はターゲットビット集合である。 $S$ はターゲット列集合( $CTX$ のステップ(1-3)では、 $S =$  )である。出力 $C$ : 結果として得られる部分的に特定されたテストキューブ集合である。

10

【0094】

ステップ $S-1$ では、 $V$ の全ての必須故障を特定する。それらの故障はそれぞれ $V$ 内のあるテストベクトルによってのみ検出され、2回検出故障シミュレーションによって特定される。

【0095】

ステップ $S-2$ では、 $V$ 内のビットであって、(1)全ての必須故障を検出するために、および、(2) $S$ 内の全てのターゲット列の論理値を保持するために、論理値が必要とされる全てのビットを、 $T$ 内のビットはできるだけ避けるようなやり方で、特定する。そのようなビットは $ATPG$ でも広く応用されている正当化操作を用いて容易に見つけ出される。続いて $V$ 内の特定されたビットを $X$ ビットに置き換え、中間テストキューブ集合 $C'$ が得られる。

20

【0096】

ステップ $S-3$ では、3値故障シミュレーションを $C'$ に実行して $V$ の全ての非必須故障を求める。検出されなかった全ての非必須故障については、それらを検出するために論理値が必要とされる $V$ 内の $X$ ビットを、 $T$ 内のビットはできるだけ避けるようなやり方で、特定する。続いて特定された $X$ ビットを $V$ 内のそのビットの元の論理値を用いて回復する。こうして最終テストキューブ集合 $C$ が得られる。

【0097】

なお、制限付きテストリラクゼーションの手順の時間的コストは $O(M \times N)$ である。ここで、 $M$ および $N$ はそれぞれ故障の数およびテストベクトルの数である。

30

【0098】

[実験結果]

図7は実験結果を説明するための二つの表を示す図である。

【0099】

$CTX$ テスト生成方式は評価実験のために $C$ 言語を用いて実装されている。ゲートクロックを備えたベンチマーク回路が入手できないため、オープンソースのマイクロプロセッサデザインであるpicoJava(登録商標)を合成し図7(A)の表に示す2つのテスト回路 $TC-1$ および $TC-2$ を作成した。1つのクロックゲーティングブロックは1つのクロックゲート $FF$ 群に対応している。評価実験は $TC-1$ および $TC-2$ に対して行われ、結果は図7(B)の表にまとめた。遷移 $LOC$ 遅延テストベクトルはTetraMAX(登録商標)によって生成される。テストベクトル数および故障検出率はそれぞれ“ベクトル数”および“故障検出率(%)”の下に表示されている。(1)Preferred Fillを用いたXID、(2)JP Fillを用いたXID、(3)提案した $CTX$ 方式を用いた3つの実験を行った。ここで、XIDとはゲートクロックを用いないテストリラクゼーションシステムである。WSA基準による最初のテストベクトル集合に対するラウンチ遷移の削減率はそれぞれ“XID+Preferred”、“XID+JP”、“CTX”の下に示されている。図7(B)の表は $CTX$ 方式が従来のテストリラクゼーションと $X$  fillingに基づいたこれまでの方式よりも効率的であることを示している。また、クロックゲートメカニズムが大量に用いられている低電力デバイスに $CTX$ が特に効率的であることも示されている。

40

【0100】

50

実験結果も踏まえ、以下のことが言える。

(1) CTXは実速度スキャンテストにおける歩留まり損失リスクを削減する上で、CTXはテストデータの増加も故障検出率の低下も回路/クロッキングの変更も回路のパフォーマンスの低下も一切引き起こさないという意味で、非干渉的である。このことがCTXをどんなパワーセーフテスト生成フローにおいても利用価値の高いものにしていく。(2) CTXはテストキューブ内のXビットの数が限られているテストコンパクションおよびテストコンプレッションに適用可能である。CTXは最初のテストベクトル量が不変であるような巧みなやり方でクロックゲートを十分に活用することでそのようなXビット不足を補っている。

【0101】

【まとめ】

以上のように、実速度スキャンテストにおいて歩留り損失を生じさせうるラウンチ遷移を効率的に削減するために、CTX (Clock-Gating-Based Test Relaxation and X-Filling) という方式を提案した。基本的な考え方はクロックゲーティングを用いて故障検出に貢献しないFFをできるだけ多く無効化することである。CTXはテストリラクゼーションおよびX fillingにクロックゲーティングを十分に活用する種類のものとしては初めてのものである。CTXはテストサイズ、故障検出率、ATPG・回路・クロックの設計、および機能上のパフォーマンスに全く影響を及ぼさずに歩留り損失を削減できる。その上、CTXはXビットが限られているようないかなるテストコンプレッション方式にも適用可能である。

【0102】

なお、上記まででは、図10に示すようなフルスキャン順序回路を用いて説明したが、図8に示すようなパーシャルスキャン回路にも適用可能である。ここで、パーシャルスキャン回路とは、一部のフリップフロップをスキャンフリップフロップに置き換えた上で、それらを用いて1本若しくは複数本のスキャンチェーンを形成するものである。また、パイプライン順序回路に対しても適用可能である。

【0103】

具体的には、組合せ回路部分3000と順序回路のフリップフロップ3001とパーシャルスキャン順序回路のスキャンフリップフロップ3002とから構成される。組合せ回路部分3000は図10の組合せ回路部1201と同様ではあるが、その入力は $\langle v : PPI \rangle$ がスキャンシフトによってスキャンフリップフロップ3002に設定される部分 $\langle v : PPI_{n2} \rangle$ とフリップフロップ3001に設定される部分 $\langle v : PPI_{n1} \rangle$ からなる。また、組合せ回路部分3000からの出力は同じく図10の組合せ回路部1201と同様であるが、テストベクトル $v$ に対するテスト応答 $f(v)$ としての外部出力線に直接現れる部分 $\langle f(v) : PO \rangle$ と擬似外部出力線に現れる部分 $\langle f(v) : PPO \rangle$ のうち、 $\langle f(v) : PPO \rangle$ はスキャンキャプチャによってスキャンフリップフロップ3002に取り込まれる部分 $\langle f(v) : PPO_{n2} \rangle$ とフリップフロップ3001に取り込まれる部分 $\langle f(v) : PPO_{n1} \rangle$ からなる。

【0104】

図8(b)は図8(a)におけるスキャンフリップフロップ3002におけるスキャンキャプチャの前と後において論理値に相違が発生する場合の一例を示す。

【0105】

図10(b)とも同様であるが、図8(b)において、テストベクトル $\langle v : PPI_{n2} \rangle$ の要素である一つのビット $a$ と、それに対応するテスト応答 $\langle f(v) : PPO_{n2} \rangle$ が、スキャンフリップフロップ3002で異なる論理値を取ると、キャプチャモードの際に論理値の相違(以下、遷移とする)が発生する。ある一つのテストベクトルに対する遷移の数は、そのテストベクトルを原因とした組合せ回路部分3000を含めた回路全般で発生する消費電力と深く関係しているため、テストベクトルに対するキャプチャ時の遷移の数を削減することで、キャプチャ時の消費電力を低減することができる。このような処理が図4のステージ2(FF静止化)の処理として行われる。

10

20

30

40

50

【 0 1 0 6 】

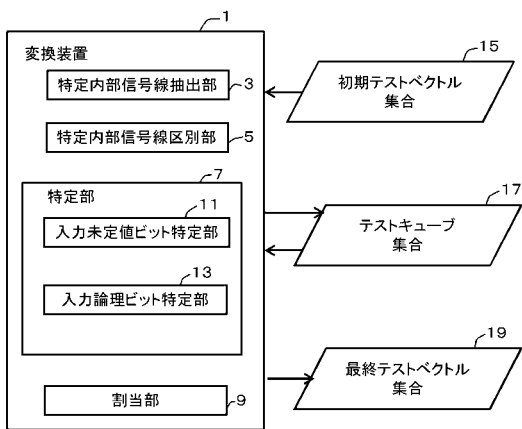
なお、上記では故障検出対象の論理回路の例として、フルスキャン順序回路の組み合わせ回路部分の場合のほか、パーシャルスキャン回路の組み合わせ回路部分の場合も示したが、パイプライン方式の論理回路などの他の論理回路であってもよい。

【 符号の説明 】

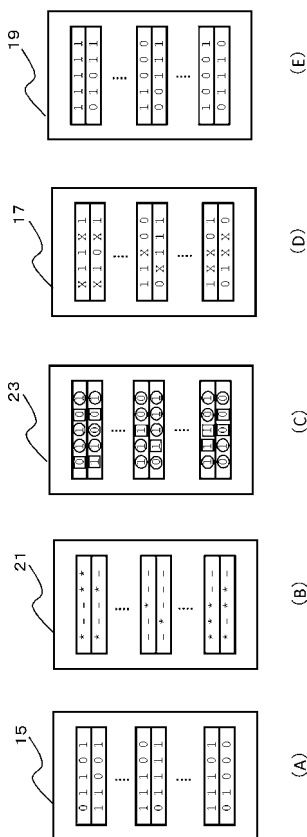
【 0 1 0 7 】

- 7 特定部
- 9 割当部
- 1 1 入力未定値ビット特定部
- 1 3 入力論理ビット特定部

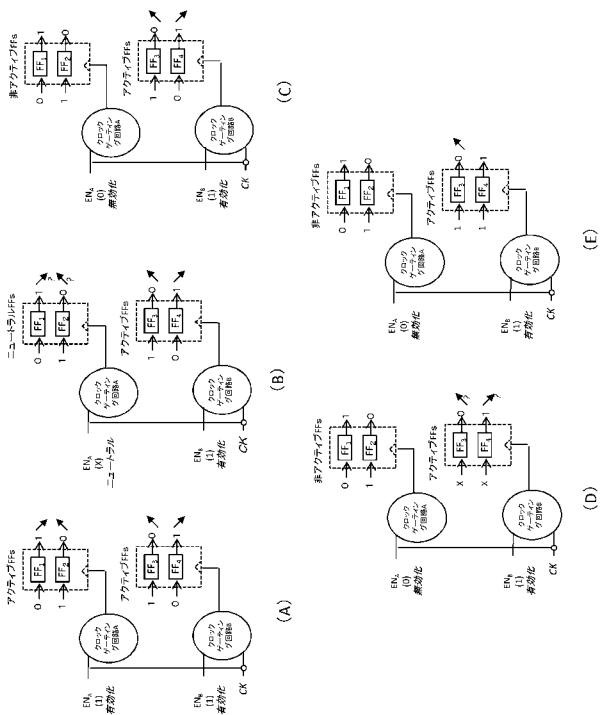
【 図 1 】



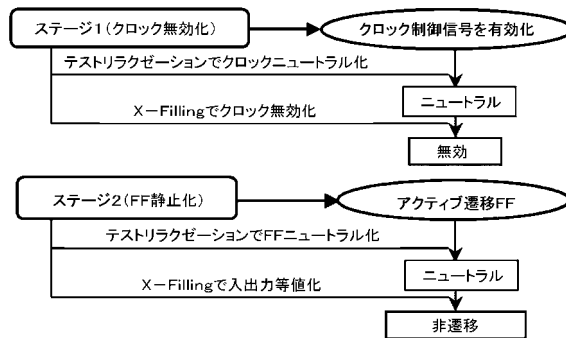
【 図 2 】



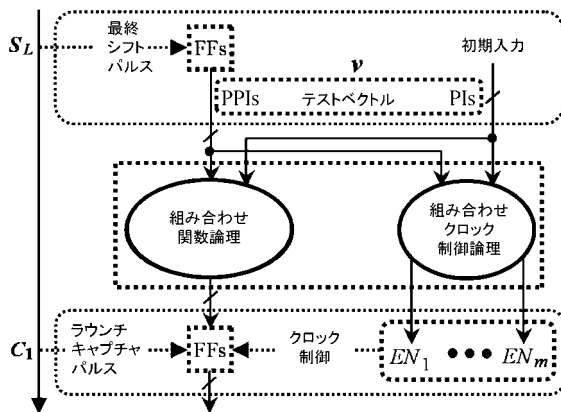
【図3】



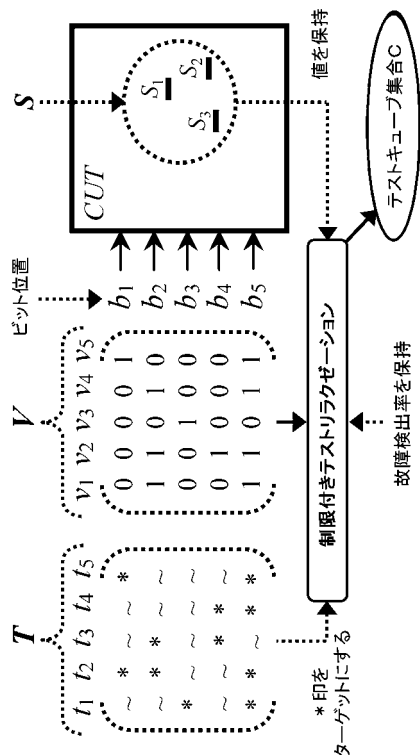
【図4】



【図5】



【図6】



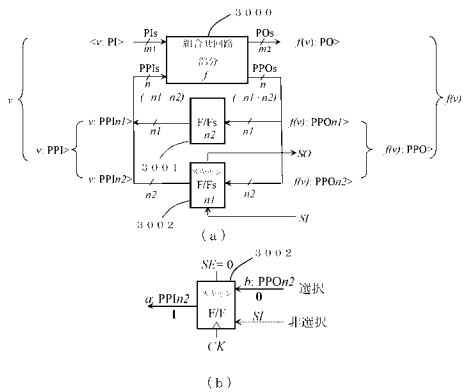
【図7】

回路	ゲート数	FF数	クロックゲートFFの割合	クロックゲートインゲートの数	クロックゲートインゲートの平均値	ラウンチ遷移の削減率	
						XID Preferred	XID JP CUTX
TC-1	207.8K	6,992	79.4%	3,441	1.6	15.2	23.9
TC-2	191.5K	6,992	31.1%	65	33.5	15.6	17.4

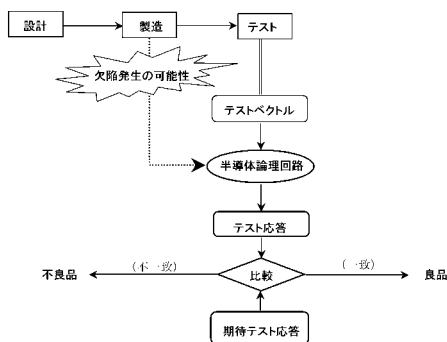
(A)

(B)

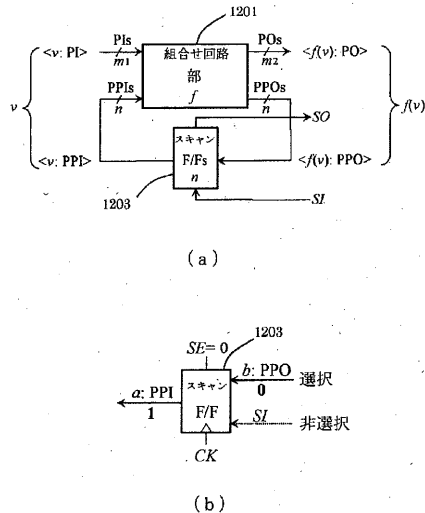
【図8】



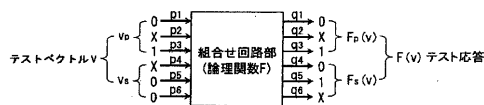
【図9】



【図10】



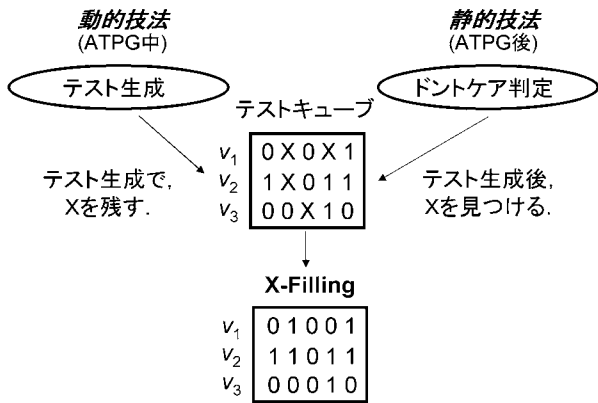
【図11】



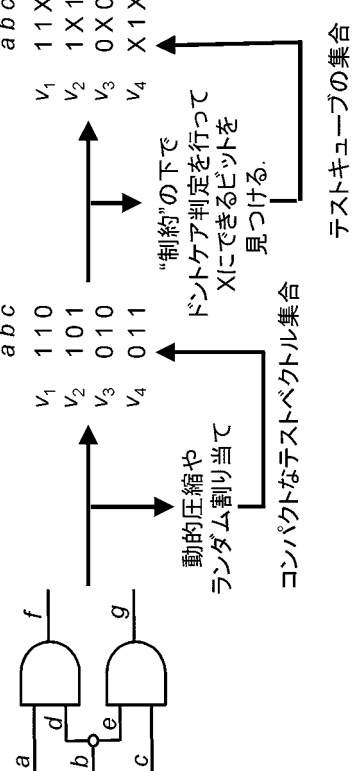
【図12】

テストキューブ	テストベクトル
$c_1$	$v_1$
$c_2$	$v_2$
$c_3$	$v_3$

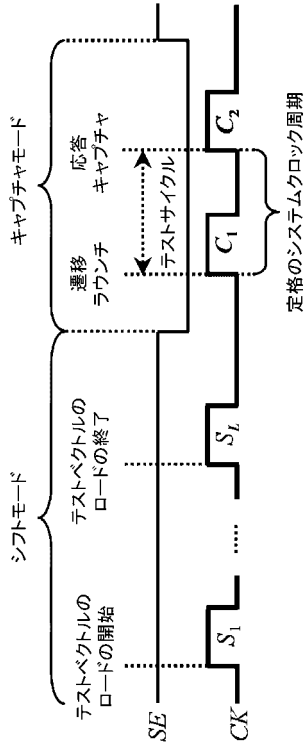
【図13】



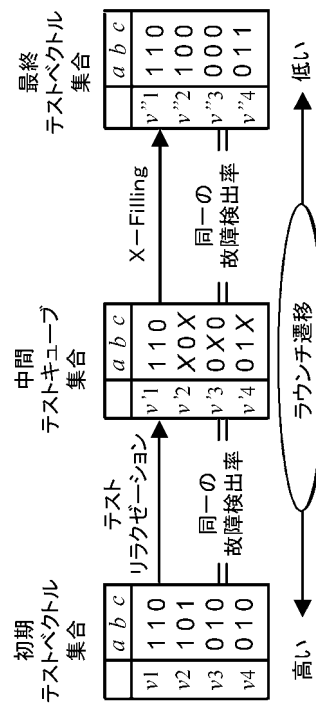
【図14】



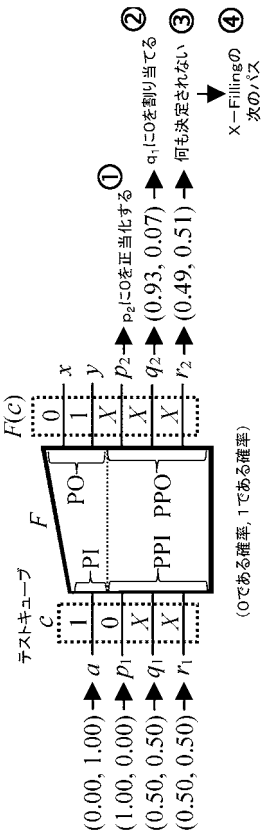
【図 15】



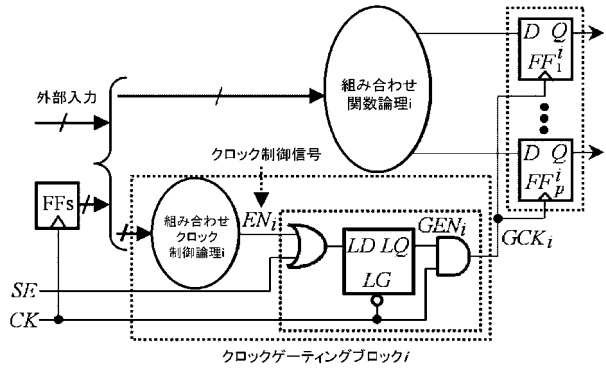
【図 16】



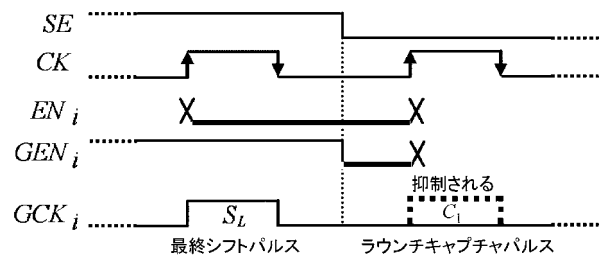
【図 17】



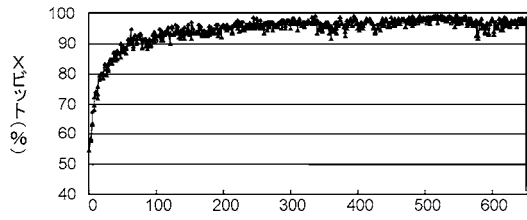
【図 18】



【図 19】



【 20】



---

フロントページの続き

(72)発明者 大和 勇太

福岡県飯塚市大字川津680番地の4号国立大学法人九州工業大学内

審査官 神谷 健一

(56)参考文献 国際公開第2008/001818(WO, A1)

特開2007-155339(JP, A)

特開2006-47013(JP, A)

SEIJI KAJIHARA ET AL, Test Data Compression Using Don't-Care Identification and Statistical Encoding, PROCEEDINGS OF THE 11TH ASIAN TEST SYMPOSIUM (ATS'02), 米国, IEEE, 2002年, pp. 67 - 72

XIAOQING WEN ET AL, A Novel Scheme to Reduce Power Supply Noise for High-Quality At-Speed Scan Testing, IEEE International Test Conference ITC 2007, 米国, IEEE, 2007年10月, pp. 1-10

H. FURUKAWA ET AL, CTX: A Clock-Gating-Based Test Relaxation and X-Filling Scheme for Reducing Yield Loss Risk in At-Sp, 17th Asian Test Symposium ATS'08., 米国, IEEE, 2008年11月, pp. 397 - 402

(58)調査した分野(Int.Cl., DB名)

G01R 31/28 - 31/3193

G06F 11/22 - 11/26