

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6223967号
(P6223967)

(45) 発行日 平成29年11月1日(2017.11.1)

(24) 登録日 平成29年10月13日(2017.10.13)

| | |
|------------------------|----------------|
| (51) Int.Cl. | F I |
| GO 1 R 31/28 (2006.01) | GO 1 R 31/28 V |

請求項の数 4 (全 17 頁)

| | | | |
|---------------|------------------------------|-----------|--|
| (21) 出願番号 | 特願2014-516759 (P2014-516759) | (73) 特許権者 | 503360115 国立研究開発法人科学技術振興機構 埼玉県川口市本町四丁目1番8号 |
| (86) (22) 出願日 | 平成25年5月14日(2013.5.14) | (74) 代理人 | 100116573 弁理士 羽立 幸司 |
| (86) 国際出願番号 | PCT/JP2013/063393 | (72) 発明者 | 佐藤 康夫 福岡県飯塚市川津680-4 国立大学法人九州工業大学内 |
| (87) 国際公開番号 | W02013/175998 | (72) 発明者 | 王 森▲レイ▼ 福岡県飯塚市川津680-4 国立大学法人九州工業大学内 |
| (87) 国際公開日 | 平成25年11月28日(2013.11.28) | (72) 発明者 | 官瀬 紘平 福岡県飯塚市川津680-4 国立大学法人九州工業大学内 |
| 審査請求日 | 平成28年5月16日(2016.5.16) | | |
| (31) 優先権主張番号 | 特願2012-117842 (P2012-117842) | | |
| (32) 優先日 | 平成24年5月23日(2012.5.23) | | |
| (33) 優先権主張国 | 日本国(JP) | | |

最終頁に続く

(54) 【発明の名称】 故障検出システム、生成回路及びプログラム

(57) 【特許請求の範囲】

【請求項1】

スキャンテストにより論理回路からキャプチャされたテスト出力と、前記論理回路が故障していない場合のテスト出力とを比較することにより故障検出を行う故障検出装置を備えた故障検出システムであって、

複数のフリップフロップを使用するスキャンテストで複数回のキャプチャを行うマルチキャプチャにおいて、

前記同一のキャプチャモードにおける最終キャプチャであることを示す最終信号を、前記キャプチャモードにおける前記最終キャプチャの1つ前のキャプチャが終了した後に生成する最終信号生成手段と、

前記最終信号を受信した場合に、最終キャプチャでキャプチャされた論理値と当該最終キャプチャの直前にキャプチャされた論理値とが同一であるフリップフロップからのみ前記論理値を指定する対象となるフリップフロップを選択する選択手段と、

前記選択手段により選択されたフリップフロップの論理値を最終キャプチャのスキャンアウト前に指定する指定手段と、

前記論理回路からキャプチャされたテスト出力であって前記指定手段が指定した論理値を含むものと、前記論理回路が故障していない場合のテスト出力であって前記指定手段が指定した論理値を含むものとを比較することにより故障検出を行う故障検出装置とを備え、

前記指定手段が前記選択されたフリップフロップの論理値を変更することにより、故障

10

20

検出率を維持しつつスキャンアウト時のシフトパワーの削減を図り、

前記選択手段は、同ースキャンチェーンから複数のフリップフロップを選択する場合には隣接しないフリップフロップから選択し、

前記指定手段は、同ースキャンチェーン内で前記選択されたフリップフロップに隣接するフリップフロップの論理値を指定する、故障検出システム。

【請求項 2】

前記選択手段は、スキャンチェーンのスキャンイン側に近いフリップフロップを優先的に選択する請求項 1 記載の故障検出システム。

【請求項 3】

スキャンテストにより論理回路からキャプチャされたテスト出力と、前記論理回路が故障していない場合のテスト出力とを比較することにより故障検出を行う故障検出回路であって、

複数のフリップフロップを使用するスキャンテストで複数回のキャプチャを行うマルチキャプチャにおいて、

前記同一のキャプチャモードにおける最終キャプチャであることを示す最終信号を、前記キャプチャモードにおける前記最終キャプチャの 1 つ前のキャプチャが終了した後に生成する最終信号生成手段と、

前記最終信号を受信した場合に、最終キャプチャでキャプチャされた論理値と当該最終キャプチャの直前にキャプチャされた論理値とが同一であるフリップフロップからのみ前記論理値を指定する対象となるフリップフロップを選択する選択回路と、

前記選択手段により選択されたフリップフロップの論理値を最終キャプチャのスキャンアウト前に指定する指定回路を備え、

前記指定回路が前記選択されたフリップフロップの論理値を変更することにより、故障検出率を維持しつつスキャンアウト時のシフトパワーの削減を図り、

前記選択手段は、同ースキャンチェーンから複数のフリップフロップを選択する場合には隣接しないフリップフロップから選択し、

前記指定手段は、同ースキャンチェーン内で前記選択されたフリップフロップに隣接するフリップフロップの論理値を指定し、

前記論理回路からキャプチャされたテスト出力であって前記指定手段が指定した論理値を含むものと、前記論理回路が故障していない場合のテスト出力であって前記指定手段が指定した論理値を含むものとを比較することにより故障検出を行う、故障検出回路。

【請求項 4】

スキャンテストにより論理回路からキャプチャされたテスト出力と、前記論理回路が故障していない場合のテスト出力とを比較することにより故障検出を行う故障検出システムにおける故障検出方法であって、

複数のフリップフロップを使用するスキャンテストで複数回のキャプチャを行うマルチキャプチャにおいて、

最終信号生成手段が、前記同一のキャプチャモードにおける最終キャプチャであることを示す最終信号を、前記キャプチャモードにおける前記最終キャプチャの 1 つ前のキャプチャが終了した後に生成するステップと、

選択手段が、前記最終信号を受信した場合に、最終キャプチャでキャプチャされた論理値と当該最終キャプチャの直前にキャプチャされた論理値とが同一であるフリップフロップからのみ前記論理値を指定する対象となるフリップフロップを選択する選択ステップと

、
指定手段が、前記選択手段により選択されたフリップフロップの論理値を最終キャプチャのスキャンアウト前に指定する指定ステップと、

故障検出装置が、前記論理回路からキャプチャされたテスト出力であって前記指定手段が指定した論理値を含むものと、前記論理回路が故障していない場合のテスト出力であって前記指定手段が指定した論理値を含むものとを比較することにより故障検出を行うステップとを含み、

10

20

30

40

50

前記指定手段が前記選択されたフリップフロップの論理値を変更することにより、故障検出率を維持しつつスキャンアウト時のシフトパワーの削減を図り、

前記選択手段が、同一スキャンチェーンから複数のフリップフロップを選択する場合には隣接しないフリップフロップから選択し、

前記指定手段が、同一スキャンチェーン内で前記選択されたフリップフロップに隣接するフリップフロップの論理値を指定する、故障検出方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、故障検出システム、生成回路及びプログラムに関し、特に、スキャンテストにより論理回路の故障を検出する故障検出システム等に関する。 10

【背景技術】

【0002】

スキャンテスト時の消費電力が増大すると、電源電圧が一時的に低下するIRドロップが生じる。このとき、論理回路が誤動作し、誤ったテスト応答値がスキャンフリップフロップに取り込まれることがある。すると、通常時には正常に動作する論理回路がテスト時に不良品と判定されてしまうという誤テストが発生する。その結果、歩留りが低下する。特に、半導体論理回路が超大規模化、超微細化、低電源電圧化した場合、誤テストによる歩留り低下は顕著である。そこで、スキャンテスト時の消費電力を低減することが重要となる。 20

【0003】

スキャンテストにおける消費電力は、キャプチャモード時の消費電力（キャプチャパワー）とシフトモード時の消費電力（シフトパワー）に大別される。さらに、シフトパワーは、スキャンインに伴うものとスキャンアウトに伴うものに分けられる。本願の発明者等は、このうち、キャプチャパワーの低減（特許文献1参照）及びスキャンインに伴うシフトパワーの低減（特許文献2参照）を可能とする故障検出システムを開発してきた。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】国際公開第2012/046602号 30

【特許文献2】特願2012-002214号

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかし、スキャンイン時のシフトパワーに比べて、スキャンアウト時のシフトパワーが十分に削減されていない。これは、シフトアウト前のテスト出力パターンを変換することがテスト目的に沿わないという意味で困難であることに起因する。

【0006】

例えば特許文献2の技術を用いて、低シフトパワーのテストパターンを生産したとしても、テスト対象回路から出力されたテストパターンはある程度変化している。とはいえ、故障判定がそもそもの目的であって、シフトパワー削減のために出力テストパターンを安易に変換するわけにもいかなかった。 40

【0007】

そのため、スキャンアウト時のシフトパワーを削減する技術が求められていた。

【0008】

ゆえに、本発明は、故障検出率を維持しつつ、スキャンアウト時のシフトパワーの削減を図ることを可能とする、故障検出システム等を提供することを目的とする。

【課題を解決するための手段】

【0009】

本発明の第1の観点は、スキャンテストにより論理回路の故障を検出する故障検出シス 50

テムであって、複数のフリップフロップと、キャプチャモードにおける最終キャプチャであることを示す最終信号を生成する最終信号生成手段と、前記論理回路とも前記フリップフロップとも異なり、一部の前記フリップフロップに対して、前記最終信号を受信した場合に論理値を指定する指定手段と、前記論理回路からキャプチャされたテスト出力であって前記指定手段が指定した論理値を含むものと、前記論理回路が故障していない場合のテスト出力であって前記指定手段が指定した論理値を含むものとを比較することにより故障検出を行う故障検出装置とを備える、故障検出システムである。

【0010】

本発明の第2の観点は、第1の観点の故障検出システムであって、同一のキャプチャモードにおいて複数回のキャプチャを行うものであり、前記フリップフロップに接続して、他のフリップフロップを経由せずに論理値を取り出す取出回路と、前記指定手段が論理値を指定する対象となるフリップフロップを選択する選択手段とをさらに備え、前記選択手段は、前記取出回路に接続しているフリップフロップから選択する。

10

【0011】

本発明の第3の観点は、第2の観点の故障検出システムであって、前記選択手段は、トグルが集中する箇所のフリップフロップ、スキャンアウト電力に影響の大きいトグルがあるフリップフロップ、又は、前後のフリップフロップのビットの平均値と相違の多い論理値を有するフリップフロップから、優先的に前記論理値を指定する対象となるフリップフロップを選択する。

【0012】

20

本発明の第4の観点は、第1の観点の故障検出システムであって、同一のキャプチャモードにおいて複数回のキャプチャを行うものであり、前記指定手段が論理値を指定する対象となるフリップフロップを選択する選択手段をさらに備え、前記選択手段は、最後にキャプチャされた論理値が直前にキャプチャされた論理値と同一であるフリップフロップから選択する。

【0013】

本発明の第5の観点は、第4の観点の故障検出システムであって、前記選択手段は、同一スキャンチェーンから複数のフリップフロップを選択する場合には隣接しないフリップフロップから選択し、前記指定手段は、選択されたフリップフロップに対して、同一スキャンチェーン内で前記選択されたフリップフロップに隣接するフリップフロップの論理値を指定する。

30

【0014】

本発明の第6の観点は、第2から第5のいずれかの観点の故障検出システムであって、前記指定手段は、前記選択手段に選択されたフリップフロップに対して、0若しくは1の一定の論理値、同一スキャンチェーン内の隣接するフリップフロップが有する論理値、若しくは、同一スキャンチェーン内の2つ前のフリップフロップが有する論理値と同一の論理値を指定し、又は、複数のフリップフロップの論理値を参照して論理値を指定する。

【0015】

本発明の第7の観点は、スキャンテストにより論理回路の故障を検出する故障検出システムにおいて、フリップフロップに対して論理値を生成して与える生成回路であって、前記論理回路ともフリップフロップとも異なるものであり、キャプチャモードにおける最終キャプチャを行ったフリップフロップに対して論理値を与える、生成回路である。

40

【0016】

本発明の第8の観点は、スキャンテストにより論理回路の故障を検出するために、コンピュータを、前記論理回路からキャプチャされたテスト出力の一部に前記論理回路とは異なる指定手段が指定した論理値を与えたものと、前記論理回路が故障していない場合のテスト出力の一部に前記指定手段が指定した論理値を与えたものとを比較することにより故障検出を行う故障検出手段として機能させるためのプログラムである。

【0017】

本発明の第9の観点は、スキャンテストにより論理回路の故障を検出する故障検出シス

50

テムにおける故障検出方法であって、キャプチャモードにおける最終キャプチャの後に、フリップフロップに対して論理値を指定する指定ステップと、前記論理回路からキャプチャされたテスト出力であって前記指定ステップにおいて指定された論理値を含むものと、前記論理回路が故障していない場合のテスト出力であって前記指定ステップにおいて指定された論理値を含むものとを比較することにより故障検出を行う故障検出ステップとを含む、故障検出方法である。

【0018】

なお、指定手段は、フリップフロップ（FF）が有するものであってもよいし、FFの外部に存在してもよい。

【0019】

また、選択手段は、スキャンチェーンのスキャンイン側のFFを優先的に選択してもよい。これにより、シフトパワーをさらに効率よく削減することが可能となる。

【発明の効果】

【0020】

本発明の各観点によれば、キャプチャモードで最終的にFFにキャプチャされた出力論理値（テスト出力）に対して、新たな論理値を与えた上で故障検出を行うことが可能となる。そのため、新たな論理値を与えるべきFF及び論理値を適宜選択することにより、故障検出率を維持させつつスキャンアウト時のシフトパワーの削減を図ることが可能となる。

【0021】

ここで、通常のスキャンテストであれば、故障判定対象である論理回路からのテスト出力を故障判定前に変更することは考えられない。本願の着眼の独創性は、「一定の条件を満たすFFについては、キャプチャした論理値をシフトアウトする前に論理値を変更したとしても故障検出率を維持できる」との知見に基づいて、故障判定前にFFの論理値を変更し、スキャンアウト時のシフトパワーを削減する点にある。

【0022】

本発明の第2の観点によれば、取出回路に接続されたFFは、同一のキャプチャモードにおいて複数回の故障判定が行われる。そのため、取出回路に接続されたFFがキャプチャモードで最終的にキャプチャする論理値を改めてシフトアウトせずともよい。よって、取出回路に接続されたFFに新たな論理値を与えることにより、故障検出率を維持しつつシフトパワーの削減を図ることが容易となる。

【0023】

本発明の第3の観点によれば、顕著なシフトパワー削減の効果が得られる。

【0024】

本発明の第4の観点によれば、最後の2回で同じ論理値をキャプチャするFFに新たな論理値を与えることによりシフトパワーを削減することが可能となる。発明者らによる実験結果を後述するとおり、最後の2回に同じ論理値をキャプチャするFFは遷移遅延故障の故障検出率に寄与しない確率が高いからである。このため、遷移遅延故障の故障検出率を維持しつつシフトパワーの削減を図ることが容易となる。

【0025】

本発明の第5の観点によれば、シフトパワーを削減する確率の高い論理値を与えることが可能となる。

【0026】

本発明の第6の観点によれば、指定手段が、同一スキャンチェーン内で隣接するFFに同一の論理値を与えることにより、シフトパワーを削減することが可能となる。

【図面の簡単な説明】

【0027】

【図1】本願発明の実施例に係る故障検出システム1の概要を示すブロック図である。

【図2】リセット入力によりスキャンフリップフロップに論理値0を与える回路構成を例示する図である。

10

20

30

40

50

【図3】シミュレーションにより期待値を生成する手順の例を示すフロー図である。

【図4】マルチプレクサによりスキャンフリップフロップに所定の論理値を与える回路構成を例示する図である。

【図5】セット/リセット入力により隣接するFFの論理値を与える回路構成を例示する図である。

【図6】隣接するSFFと論理値が異なる場合に、セット/リセット入力により隣接するSFFの論理値を与える回路構成を例示する図である。

【図7】隣接するSFFと論理値が異なる場合に、マルチプレクサにより隣接するSFFの論理値を与える回路構成を例示する図である。

【図8】オーバーヘッド増加率を示すグラフである。

10

【発明を実施するための形態】

【0028】

以下、図面を参照して、本願発明の実施例について述べる。なお、本願発明の実施の形態は、以下の実施例に限定されるものではない。

【実施例1】

【0029】

図1は、本願発明の実施例に係る故障検出システム1の概要を示すブロック図である。以下、図1の故障検出システム1の概要を説明する。

【0030】

故障検出システム1（本願請求項の「故障検出システム」の一例）は、スキャンテストにより組合せ回路部3（本願請求項の「論理回路」の一例）の故障を検出するものである。故障検出システム1は、組合せ回路部3から出力される論理値をキャプチャする保持回路部5と、保持回路部5に論理値をシフトインするシフトイン部7と、保持回路部5に対して最終キャプチャ後に論理値を指定する論理値指定部9（本願請求項の「指定手段」の一例）と、保持回路部5が保持した論理値を観測する観測部11と、観測部が観測した論理値に基づいて故障検出を行う故障検出装置13（本願請求項の「故障検出装置」の一例）とを備える。保持回路部5は、複数のフリップフロップ（FF）6_n（nは、2以上の自然数）（本願請求項の「フリップフロップ」の一例）を有する。複数のFF6_nは、1つ又は複数のスキャンチェーンを形成する。ここで、一部のFFには、他のFFを経由せずに論理値を取り出す取出回路10（本願請求項の「取出回路」の一例）に接続されている。

20

30

【0031】

論理値指定部9は、論理値を指定する対象となるFF又はスキャンフリップフロップ（SFF）を選択する選択部15（本願請求項の「選択手段」の一例）と、選択されたFF又はSFFに指定する論理値を決定する論理値決定部17と、キャプチャモードにおける最終キャプチャであることを示す最終信号を生成する最終信号生成部19（本願請求項の「最終信号生成手段」の一例）とを有する。

【0032】

観測部11は、キャプチャモードにおいて取出回路10が取り出した論理値を観測するキャプチャモード観測部21と、シフトモードにおいてシフトアウトされた論理値を観測するシフトモード観測部23とを有する。故障検出装置13は、組合せ回路部3が正常である場合に出力されることが期待される論理値である期待値を算出するシミュレーション部25と、シミュレーション部25が算出した期待値と観測部11が観測した値とを比較する比較部27と、比較部27の比較結果に基づいて故障判定を行う故障判定部29とを有する。

40

【0033】

論理値指定部9は、故障検出率を維持しつつシフトパワーの削減を図るために、適切なSFFに対して適切な論理値を指定する必要がある。ここでの基本的な考え方は、故障検出率の向上に寄与しないSFFに対して、スキャンチェーン内で隣接するSFFと同じ論理値を与えるというものである。

50

【 0 0 3 4 】

選択部 15 は、取出回路 10 に接続されて中間取出が行われる S F F から選択する。中間取出を行った S F F は、すでに故障検出に寄与しており、最終キャプチャ時に論理値を変更しても差し支えないからである。ただし、後に述べるルールに従って他の S F F を選択する方法を適用してもよい。

【 0 0 3 5 】

また、選択部 15 は、最後の 2 回で同じ論理値をキャプチャする S F F に新たな論理値を与えることによりシフトパワーを削減することが可能となる。最後の 2 回に同じ論理値をキャプチャする F F は、多くのパターンに対して値が変化しない F F である可能性があり、確率的に故障検出の可能性は低いと考えられる。その結果、遷移遅延故障の故障検出率に寄与しない確率が高いからである。このような F F が多数あることを以下に示す。

10

【 0 0 3 6 】

表 1 は、テスト対象の論理回路ごとにキャプチャする論理値がほとんど変化しない F F の割合を示したものである。テスト対象回路は、I T C 9 9 ベンチマーク回路である b 1 4 s、b 1 5 s、b 2 0 s、b 2 1 s 及び b 2 2 s とした。実に半数近くの F F は、3 万のテスト入力パターンに対して、最終キャプチャ時に論理値が反転する割合が 1 % 以下であった。

【 0 0 3 7 】

【表 1】

| 回路 | b14s | b15s | b20s | b21s | b22s |
|----------------------|--------------|--------------|--------------|--------------|--------------|
| 全FF数 | 245 | 449 | 490 | 490 | 735 |
| 反転頻度1%以下のFF数 (割合) | 131 (53%) | 271 (60%) | 230 (47%) | 234 (48%) | 440 (60%) |

20

【 0 0 3 8 】

また、本願の発明者らは、このような F F の論理値を変換しても、故障検出率はほとんど低下しないことを確認した。表 2 は、表 1 において縮退故障検出率及び遷移故障検出率についての、故障検出率の変化例を示す表である。なお、全 F F のうち、取出回路 10 に接続された F F 2 0 %、最後の 2 回に同じ論理値をキャプチャする F F 2 0 % の合わせて 4 0 % の F F について論理値を指定した。

30

【 0 0 3 9 】

全ての回路について、故障検出率が維持されているか、0 . 1 % 未満しか低減していない。すなわち、F F を適切に選択することにより、故障検出率の維持とスキャンキャプチャ時のシフトパワーの削減の両立を図ることが可能であることが示されている。

【 0 0 4 0 】

40

【表 2】

| 回路 | 縮退故障検出率(%) | | 遷移故障検出率(%) | |
|------|------------|-------|------------|-------|
| | 元の値 | 論理値指定 | 元の値 | 論理値指定 |
| b14s | 89.00 | 88.90 | 58.71 | 58.59 |
| b15s | 78.69 | 78.17 | 55.00 | 54.16 |
| b20s | 91.19 | 91.19 | 80.55 | 80.47 |
| b21s | 91.91 | 91.91 | 82.49 | 82.47 |
| b22s | 90.78 | 90.78 | 79.68 | 79.61 |
| 平均値 | 88.31 | 88.19 | 71.29 | 71.06 |

10

【 0 0 4 1 】

以下、SFFに対して指定した論理値を与える回路構成例を複数挙げて説明する。まず、選択されたSFFに一定の論理値を与える回路構成例を説明する。なお、本実施例では、設計時に全部又は一部のSFFに付随して論理値指定部9の回路が作製される。その上で、実際のテスト時に、回路が接続されたSFFの中から論理値を指定するSFFを選択部15が選択する。しかし、1つのSFFごとに論理値指定部9の回路を付随して作製する代わりに、複数のSFFに対して1つの論理値指定部9の回路を作製することとしてもよい。

20

【 0 0 4 2 】

図2は、本実施例における論理値指定部9と、選択部15により選択されたSFF31との回路構成を示す図である。ここでは、SFF31のFF33が有するリセット入力(Reset)を利用してSFFに論理値0を与える回路構成(本願請求項の「生成回路」の一例)を例示する。

【 0 0 4 3 】

通常のSFF同様、FF33は、組合せ回路部3からの出力信号線(DI)とスキャンイン信号線(SI)とを入力ストリームとすると共にスキャンイネーブル信号線(SE)を選択制御入力とするマルチプレクサ35の出力ストリームに接続されている。また、SFF31には、最終信号(LC)とクロック信号(CLK)とを入力とするNANDゲート37がリセット端子に接続されている。リセット端子に が付いているのは、Lowアクティブでの接続を意味する。すなわち、0の信号入力時にリセットが有効となり、0が出力される。SFF31は、出力としてデータ出力線(DO)とスキャンアウト信号線(SO)を有する。

30

【 0 0 4 4 】

スキャンキャプチャモードにおいて、NANDゲート37に接続する最終信号線の値が0であれば、リセット入力が1で無効となり、SFFは、DIの値を出力する。一方、最終信号線の値が1(最終キャプチャ時)となれば、リセット入力が0となり、SFF31は0を出力する。これにより、SFF31が最終キャプチャでキャプチャした論理値によらず、最終キャプチャ後のSFF31に論理値0を指定して出力させることが可能となる。シフトパワーは、シフトモードにおいてスキャンチェーン内で隣接するSFF間の論理値が異なる場合に増大する。したがって、隣接するSFFの論理値が0であれば、最終キャプチャ後にSFF31に論理値0を与えることにより、シフトパワーの低減を図ることが可能となる。

40

【 0 0 4 5 】

ここで、論理値指定部9は、FF33のリセット入力をを用いて論理値0を指定する代わりに、セット入力を同様に用いて論理値1を指定することとしてもよい。

50

【 0 0 4 6 】

続いて、図 3 を参照して、最終キャプチャ後に論理値を与える場合における、故障検出のための期待値生成について述べる。図 3 は、シミュレーション部 2 5 がシミュレーションにより期待値を生成する手順の例を示すフロー図である。

【 0 0 4 7 】

まず、選択部 1 5 が論理値を指定すべき S F F を選択し、論理値指定部 9 と接続する (ステップ S T 0 0 1)。続いて、期待値を生成すべき最初のテストパターンを選択する (ステップ S T 0 0 2)。シミュレーション部 2 5 が、選択されたテストパターンに対して組合せ回路部 3 からキャプチャされる論理値を算出する (ステップ S T 0 0 3)。最終キャプチャとなるまで、キャプチャが繰り返される。最終信号生成部 1 9 は、最終キャプチャの 1 つ前のキャプチャが終了後、最終キャプチャ前に、最終信号を生成する (ステップ S T 0 0 4)。

10

【 0 0 4 8 】

最終信号が生成されると、論理値決定部 1 7 が、選択された S F F ごとに指定すべき論理値を決定する。続いて、論理値指定部 9 が、選択された S F F に決定された論理値を指定する (ステップ S T 0 0 5)。全ての S F F について処理が行われると、選択されたテストパターンについての期待値が決定される (ステップ S T 0 0 6)。インクリメントが行われ (ステップ S T 0 0 7)、未処理のテストパターンの有無が判定される (ステップ S T 0 0 8)。未処理のテストパターンがあれば、全てのテストパターンについて処理がされるまで、ステップ S T 0 0 3 から S T 0 0 8 が繰り返される。

20

【 0 0 4 9 】

比較部 2 7 は、図 3 のフローにより生成された期待値と、観測部 1 1 が観測した論理値とを比較する。故障判定部 2 9 は、比較結果に基づいて故障判定を行う。

【 実施例 2 】

【 0 0 5 0 】

以下、S F F に指定した論理値を与える回路構成例を複数説明する。本実施例では、マルチプレクサを用いる回路構成例について説明する。

【 0 0 5 1 】

図 4 を参照して、選択部が選択した S F F 3 9 は、組合せ回路部 3 からの出力信号線及び論理値 0 を与える信号線を入力ストリームとすると共に最終信号を選択制御入力とするマルチプレクサ 4 1 の出力ストリームに接続されている。

30

【 0 0 5 2 】

キャプチャモードにおいて、最終信号線の値が 0 であれば、マルチプレクサ 4 1 は、DI の値を S F F 3 9 に与える。最終信号線の値が 1 (最終キャプチャ時) となれば、最終信号線の値が 1 となり、論理値 0 が S F F 3 9 に与えられる。

【 0 0 5 3 】

なお、マルチプレクサ 4 1 を介して、論理値 0 の代わりに論理値 1 を S F F 3 9 に与えることとしてもよい。

【 実施例 3 】

【 0 0 5 4 】

続いて、スキャンチェーン内で隣接する S F F の論理値を与える回路構成について述べる。図 5 において、選択された S F F 4 3 のセット端子 (SET) には、N A N D ゲート 4 7 の出力線が L o w アクティブで接続されている。N A N D ゲート 4 7 は、隣接する S F F 4 5 のキャプチャ出力信号線と最終信号線とクロック信号線とを入力とする。また、S F F 4 3 のリセット端子にも、N A N D ゲート 4 9 の出力線が L o w アクティブで接続されている。N A N D ゲート 4 9 は、隣接する S F F 4 5 のキャプチャ出力信号線の否定と最終信号線とクロック信号線とを入力とする。

40

【 0 0 5 5 】

キャプチャモードで最終信号が与えられた場合において、隣接する S F F 4 5 がキャプチャする論理値が 1 のとき、S F F 4 3 のセット入力は 0 で、リセット入力は 1 となる。

50

そのため、Lowアクティブのセットが有効化され、SFF43は1を出力する。一方、隣接するSFF45がキャプチャする論理値が0のとき、SFF43のセット入力は1で、リセット入力は0となる。そのため、リセットが有効化され、SFF43は0を出力する。

【0056】

結果として、最終信号が与えられると、SFF43は常に隣接するSFF45と同じ論理値を保持する。そのため、スキャンアウト時のシフトパワーの削減を図ることが可能となる。

【0057】

なお、図4において、マルチプレクサ41の入力ストリームのうち論理値0を与えるとした信号線をスキャンチェーン内で隣接するSFFのキャプチャ入力信号線と接続してもよい。このようにしても、最終信号が与えられると、隣接するSFFと同じ論理値をSFF39に保持させることが可能となる。

10

【実施例4】

【0058】

次に、キャプチャモードにおいてSFFがキャプチャした最後の2回の論理値が同じ場合にのみ、隣接するSFFの論理値を与える回路構成について説明する。

【0059】

図6を参照して、選択されたSFF51のリセット端子には、NANDゲート55の出力が接続されている。NANDゲート55は、スキャンチェーン内で隣接するSFF53のキャプチャ出力信号線の否定と最終信号線とクロック信号線と一致回路57の出力信号線の否定とを入力とする。一致回路(XNORゲート)57は、SFF51のDI及びDOを入力とする。

20

【0060】

最終信号線、クロック信号線の値が1であり、SFF51のDI及びDOの値が等しければ、かつ、隣接するSFF53がキャプチャする論理値DIの値が0であれば、SFF51のリセット入力が0が与えられ、SFF51は0を出力する。

【0061】

同様に、SFF51のセット入力にも、NANDゲート58の出力が接続されている。NANDゲート58は、隣接するSFF53のキャプチャ出力信号線と最終信号線とクロック信号線と一致回路57の出力信号線とを入力とする。隣接するSFF53がキャプチャする論理値DIの値が1であるときに限り、SFF51のセット入力が0を与え、SFF51に1を出力させる。

30

【0062】

既述のとおり、最後の2回のキャプチャにおいて同じ論理値をキャプチャするSFFは、故障検出率の向上に寄与しない確率が高い。本実施例の回路によれば、隣接するSFF53と同じ論理値をSFF51にキャプチャさせることで、スキャンアウト時のシフトパワーを削減する確率の高い論理値を与えることが可能となる。

【実施例5】

【0063】

さらに、図7を参照して、マルチプレクサを用いて、セット/リセット入力を利用せずに実施例4と同じ結果を得る回路を例示する。

40

【0064】

SFF59は、マルチプレクサ63の出力ストリームを入力とする。マルチプレクサ63は、組合せ回路部3からの出力信号線と隣接するSFF61のキャプチャ出力信号線とを入力ストリームとすると共にANDゲート65の出力を選択制御入力とする。ANDゲート65は、最終信号線と、一致回路67の出力線とを入力とする。一致回路67は、SFF59のDI及びDOを入力とする。

【0065】

最終信号線の値が0(最終キャプチャ時以外)であれば、SFF59は、組合せ回路部

50

3からの出力をそのままキャプチャする。最終信号線の値が1（最終キャプチャ時）であっても、S F F 5 9のDI及びDOの論理値が異なれば、S F F 5 9は、組合せ回路部3からの出力をそのままキャプチャする。一方、最終キャプチャ時であって、S F F 5 9のDI及びDOの値が等しい場合、S F F 5 9は、隣接するS F F 6 1のキャプチャする値をキャプチャする。

【実施例6】

【0066】

続いて、選択部がS F Fを選択して論理値決定部が論理値を決定する際のルールについて述べる。すでに、中間取出をおこなうS F Fから選択することが好ましいことは述べた。例えば、他にも以下のルールを適用することでスキャンアウト時のシフトパワーを効果的に削減することが可能となる。

10

【0067】

<ルール1>

スキャンイン側に近いS F Fを優先的に選択する。

【0068】

スキャンイン側に近いS F F間で隣接する論理値を一致させる方が、スキャンアウト側に近いS F F間で論理値を一致させるよりも多いスキャン数後にもシフトパワー削減の効果が得られるからである。

【0069】

<ルール2>

1つ前のS F Fの値を指定する場合は、隣接するS F F同士を選択しない。

20

【0070】

例えば、スキャンチェーン内で隣接する4つのS F F（S F F 1～S F F 4）が最終キャプチャにおいてキャプチャした論理値が、1 0 1 0であったとする。仮に選択部がS F F 2～4を選択して、最終キャプチャ後に隣接するS F Fの論理値を与えるとすると、S F F 2～4が最終的に保持する論理値は、S F F 1、2及び3がキャプチャした論理値をそれぞれ反映して1 0 1となる。反転が残り、シフトパワーを削減することが出来ない。

【0071】

そこで、隣接するS F Fの全てを選択するのではなく、例えば1つおきに選択する。上記の例では、S F F 2及びS F F 4のみを選択する。すると、S F F 2～4が最終的に保持する論理値は、S F F 1、3、3がキャプチャした論理値をそれぞれ反映して1 1 1となる。こうして、反転を解消してシフトパワーを削減可能となる。

30

【0072】

<ルール3>

1つ前のS F Fの値を指定する場合において、1つ前に隣接するS F Fも選択されているときは、2つ前のS F Fの論理値を指定する。

【0073】

本来、ルール2にあるように、隣接するS F F同士を選択しない方がシフトパワー削減上は好ましい。しかし、設計上、論理値が変わりやすい領域があり、そのような領域のS F Fについて集中的に中間取出を行う場合がある。すると、隣接するS F Fが選択部によって選択される場合があり、ルール3が有用となる。

40

【0074】

例えば、隣接する3つのS F F（S F F 5、S F F 6、S F F 7）が最終キャプチャにおいてキャプチャした論理値が、0 1 0であったとする。このうち、S F F 6及びS F F 7を選択部が選択しているとする。ルール3によれば、S F F 6が最終的に保持する論理値として、1つ前に隣接するS F F 5の論理値0が指定される。また、1つ前に隣接するS F F 6も選択されていることから、S F F 7が最終的に保持する論理値は2つ前のS F F 5の論理値0が指定される。最終的に、S F F 5～7が保持する論理値は、0 0 0となり、シフトパワーを削減することが可能となる。

【0075】

50

なお、上記では、論理値決定部は、1つのSFFが保持する論理値を、指定する論理値として決定した。他の手法として、選択されたSFFの前に存在する複数のSFFが保持する論理値を参照して、指定する論理値を決定してもよい。

【0076】

また、取出回路10に接続されたSFFを選択することと、ルール1～3は、同時に適用してもよいし、同時に適用しなくともよい。

【実施例7】

【0077】

以下、本発明に係る故障検出システムを用いた場合のスキャンアウト時のシフトパワーの削減効果を示す実験結果について述べる。テスト対象回路は、ISCAS89ベンチマーク回路であるs38417、s38584及びs35932と、ITC99ベンチマーク回路であるb14s、b15s、b20s、b21s及びb22sとした。

【0078】

表3に、スキャンイン時及びスキャンアウト時のSFFのトグル率を示す。トグル率は、スキャンシフトモードにおけるフリップフロップの反転割合を示す。FFの反転が多いほどシフトパワーが増大するため、トグル率を低下させることは、シフトパワーの低減に直結する。テスト対象回路は、s38417、s38584、s35932、b14s、b15s、b17s、b20s、b21s及びb22sとした。また、比較対照手法として、特許文献1及び特許文献2のみのテスト手法を用いた。表2内の「固定値」及び「隣接値」の欄の数値は、特許文献1及び特許文献2のテスト手法を用いた上で、それぞれ固定値又は隣接するSFFの論理値を与えた場合のトグル率を示す。マルチキャプチャ（同一のキャプチャモードにおける複数回のキャプチャ）の方式として、1つのキャプチャモードにおける最初の10回はクロック間隔を緩くして観測せず、その後の10回のクロック間隔を狭くして観測を行った。中間観測を行うSFFは、全体の20%とした。なお、SFFの選択は、ランダムとした。

【0079】

【表3】

| 回路 | スキャンイン&アウトトグル率(%) | | | | | |
|--------|-------------------|--------------|------|--------------|------|--------------|
| | 比較手法 | | 固定値 | | 隣接値 | |
| | SI | SO | SI | SO | SI | SO |
| s38417 | 7.56 | 19.01 | 7.56 | 18.07 | 7.56 | 15.02 |
| s38584 | 7.89 | 18.68 | 7.89 | 16.98 | 7.89 | 15.71 |
| s35932 | 7.62 | 15.09 | 7.62 | 16.07 | 7.62 | 14.07 |
| b14s | 8.24 | 17.37 | 8.24 | 18.43 | 8.24 | 14.08 |
| b15s | 7.74 | 14.29 | 7.75 | 11.76 | 7.75 | 11.66 |
| b17s | 7.90 | 10.88 | 7.90 | 8.84 | 7.90 | 8.68 |
| b20s | 7.54 | 17.86 | 7.69 | 19.41 | 7.69 | 15.56 |
| b21s | 7.54 | 17.82 | 7.69 | 16.71 | 7.69 | 13.96 |
| b22s | 7.69 | 18.98 | 7.69 | 17.51 | 7.69 | 14.59 |
| AVE | 7.75 | 16.67 | 7.78 | 15.97 | 7.78 | 13.70 |

【0080】

スキャンイン時のシフトパワーは、いずれの手法を用いても同じ値で低く抑えられてい

る。一方、スキャンアウト時のシフトパワーは、選択されたSFFに固定値を与える手法を用いることで、平均して0.7%程度削減された。また、隣接するSFFの論理値を与える手法を用いると、1~4%程度削減され、平均して約3%削減された。

【0081】

なお、上記の実験では、中間取出を行うFFのみを選択対象とした。そのため、最後の2回のキャプチャで論理値が反転しないFFをも選択対象とすることにより、さらなる削減効果が期待される。

【0082】

また、半導体論理回路が超大規模化、超微細化、低電源電圧化しており、SFFのトグル率の1%低減だけでも効果が大きい点に留意すべきである。従来の手法では、スキャン
10
イン段階でのシフトパワー削減策で間接的にスキャンアウト時のシフトパワーを削減する以上に削減することが困難であった。したがって、本発明に係る故障検出システム等を用いることにより、スキャンアウト時のシフトパワーを効果的に削減する意義は大きいといえる。

【実施例8】

【0083】

さらに、中間取出を行うFFを選択する際に故障検出率に加えてシフトパワー削減をも考慮して選択した場合の、本発明に係る故障検出システムを用いた実験結果について述べる。具体的には、3つの選択基準を用いてシフトパワー削減を考慮した。

【0084】

基準1：<トグル密度>

中間取出を行うFFとして、トグルが集中する箇所のFFを優先的に選択する。

【0085】

基準2：<トグル密度分解>

中間取出を行うFFとして、スキャンアウト電力に影響の大きいトグルがあるFFを優先的に選択する。

【0086】

基準3：<移動平均>

前後のFFのビットの平均値(移動平均値)とFFの値を比較し、中間取出を行うFFとして、移動平均値と相違の多い論理値を有するFFを優先的に選択する。

【0087】

表4に、スキャンイン時及びスキャンアウト時のSFFのトグル率を示す。ここで、「固定」は固定値を指定した場合のトグル率、「隣接」は1つ又は2つ隣の値を指定した場合のトグル率、「Hybrid」は選択されたFFの並びに応じて固定と隣接を使い分けるハイブリッド制御を行った場合のトグル率をそれぞれ示す。

【0088】

基準1から3のいずれの場合においてもスキャンアウト時のシフトパワーが表3に比べて、確実に削減されており、回路によっては10%以上もシフトパワーが削減された。また、いずれの基準でも、ハイブリッド制御の場合、固定値を指定した場合、隣接値の場合の順に削減効果が高かった。特に、ハイブリッド制御を行う場合、シフトイン時のシフト
40
パワーとシフトアウト時のシフトパワーが2%も変わらないほどまでシフトパワーの削減が可能となった。なお、値を指定するFFを適切に選択することにより、表4の場合の故障検出率は、表3の場合の故障検出率をほぼ維持することも可能であった。

【0089】

【表 4】

| 回路 | 入力 | 出力 制御無 | 基準1(トルゲル密度) | | | 基準2(トルゲル率分解) | | | 基準3(移動平均) | | |
|-----|------|-----------|-------------|-------|--------|--------------|-------|--------|-----------|-------|--------|
| | | | 固定 | 隣接 | Hybrid | 固定 | 隣接 | Hybrid | 固定 | 隣接 | Hybrid |
| b14 | 8.24 | 22.82 | 7.36 | 12.24 | 7.36 | 7.47 | 12.18 | 7.47 | 7.88 | 12.15 | 7.52 |
| b15 | 7.74 | 13.06 | 8.76 | 9.54 | 8.46 | 11.30 | 9.35 | 8.49 | 8.78 | 9.06 | 7.99 |
| b20 | 7.54 | 16.40 | 9.12 | 10.93 | 8.93 | 10.52 | 10.19 | 8.82 | 10.25 | 10.34 | 8.80 |
| b21 | 7.54 | 16.40 | 9.09 | 10.77 | 8.89 | 10.28 | 10.25 | 8.71 | 10.21 | 10.35 | 8.79 |
| b22 | 7.69 | 17.37 | 8.70 | 11.06 | 8.49 | 9.73 | 10.51 | 8.53 | 9.96 | 10.60 | 8.80 |
| AVE | 7.75 | 17.21 | 8.61 | 10.91 | 8.42 | 9.86 | 10.50 | 8.40 | 9.42 | 10.50 | 8.38 |

10

【0090】

続いて、上記の顕著な効果が見られた本発明に係る故障検出システムのオーバーヘッド増加率について述べる。図8は、オーバーヘッド増加率を示すグラフである。固定値を指定する場合は、ほぼ全てのオーバーヘッドがマルチキャプチャに由来する。一方、隣接値を指定する場合は、さらに約1.4%だけオーバーヘッドが増加した。ハイブリッド制御の場合、固定値を指定するFFの割合を75%以上とすることで、オーバーヘッド率を固定値指定の場合とほぼ変わらないレベルに抑えることが可能であった。特に、基準1において、固定値を指定するFFの割合を約95%以上とすることで、オーバーヘッド率は、

20

【0091】

なお、選択部15は、最後の2回のキャプチャでキャプチャする論理値が反転しないFFから選択するにあたり、シミュレーション部が行うシミュレーションにより故障検出率に及ぶ影響が小さいことが判明したFFを選択してもよい。また、反転しない論理値が0か1かをシミュレーションで確認し、FFを選択してもよい。

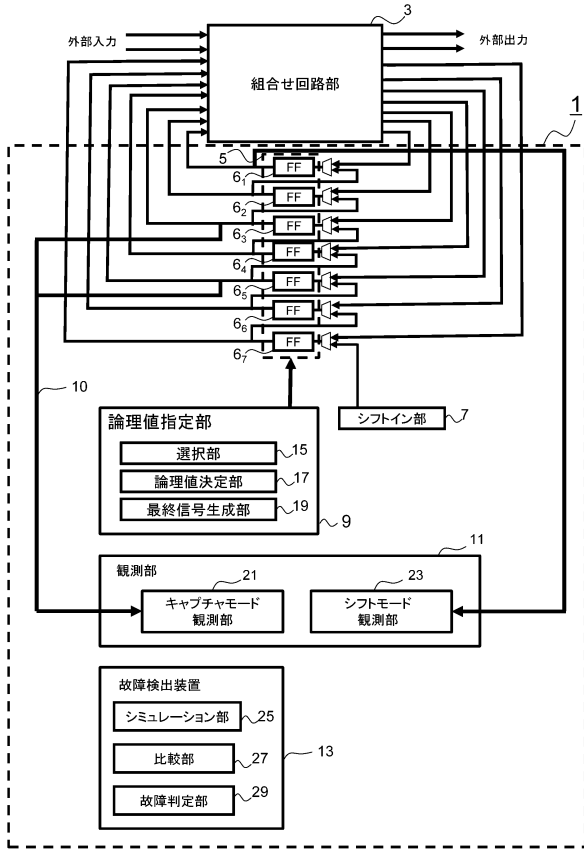
【符号の説明】

【0092】

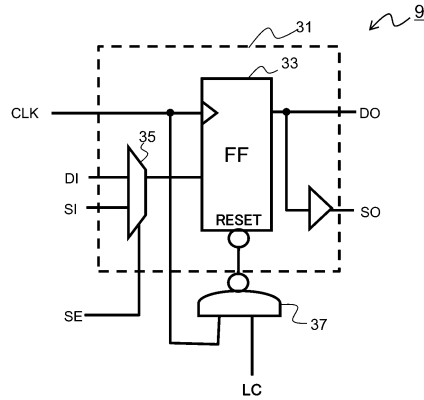
1・・・故障検出システム、3・・・組合せ回路部、5・・・保持回路部、6・・・フリップフロップ、7・・・シフトイン部、9・・・論理値指定部、11・・・観測部、13・・・故障検出装置、15・・・選択部、17・・・論理値決定部、19・・・最終信号生成部、21・・・キャプチャモード観測部、23・・・シフトモード観測部、25・・・シミュレーション部、27・・・比較部、29・・・故障判定部

30

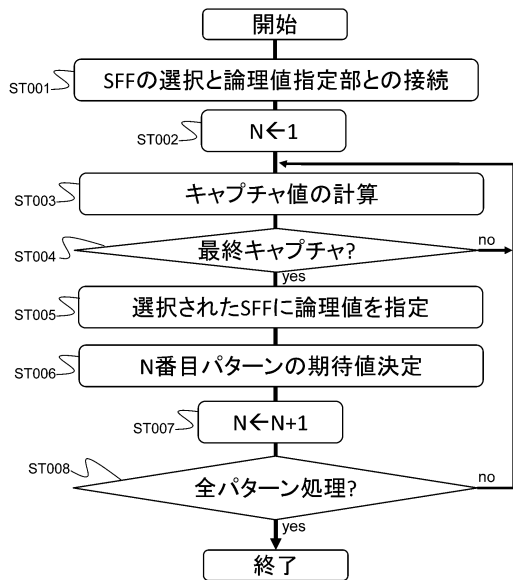
【図1】



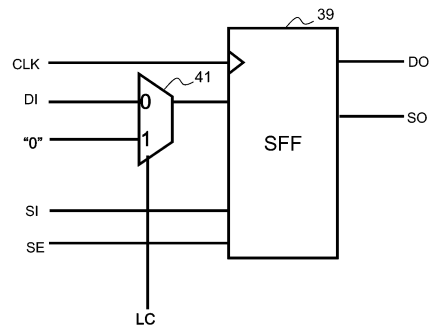
【図2】



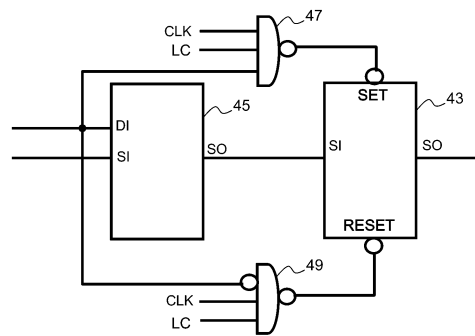
【図3】



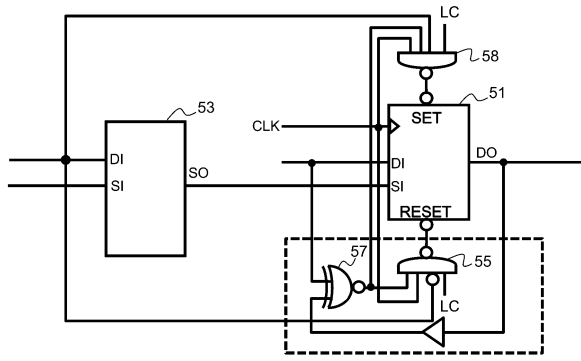
【図4】



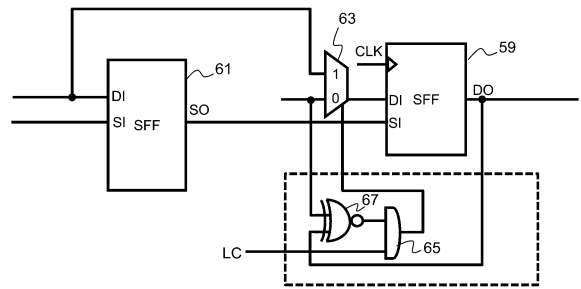
【図5】



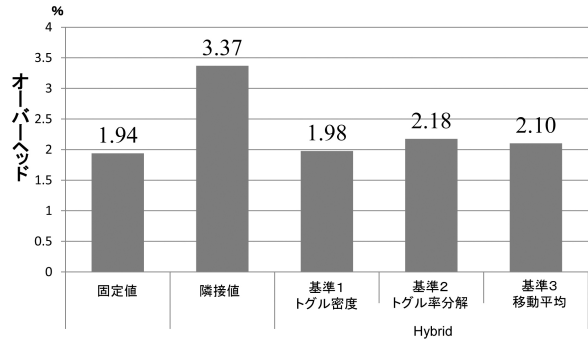
【図6】



【図7】



【図8】



フロントページの続き

(72)発明者 梶原 誠司
福岡県飯塚市川津680-4 国立大学法人九州工業大学内

審査官 越川 康弘

(56)参考文献 特開2009-150823(JP,A)
特表2010-518405(JP,A)
国際公開第2012/046602(WO,A1)

(58)調査した分野(Int.Cl., DB名)
G01R 31/28