

VLSI 回路の高歩留り化のための  
誤テスト回避と高精度故障診断に関する研究

大和 勇太



# 内容梗概

半導体集積回路（以下 VLSI）は、電子産業の米と言われるほど、情報化社会にとって必要不可欠なものになっている。回路製造技術は絶えず進歩しており、VLSI の微細化・高速化・高機能化をもたらしている。しかし一方で、微細化・高速化により引き起こされる様々な副作用により、VLSI の歩留り（生産された全ての製品に対する良品の割合）が著しく低下している。歩留り低下は、莫大な投資を必要とする半導体産業にとって深刻な問題になっている。

VLSI は製造における欠陥の発生を完全に防ぐことができないため、製造後に良品・不良品を選別する作業としてテストを行う必要がある。テストでは、VLSI の外部入力線からテストパターンを印加し、外部出力線で回路の実応答を観測し、それと期待値とを比較することによって、良品・不良品の判定が行われる。さらに、テストで不良と判定された VLSI については、歩留り、品質、及び信頼性を向上させるために、欠陥の位置特定と原因分析を行って、その結果を設計や製造にフィードバックするという不良解析が必要である。また、物理解析装置を多用する不良解析のコストを削減するため、前もってソフトウェアを用いて欠陥候補箇所を絞り込む作業として故障診断を行う必要がある。

テストでは、正常な VLSI であっても不良と判定されてしまうという誤テストが発生する可能性がある。その要因の 1 つとしては、テスト時には通常動作では起こりえない状態遷移が発生するため、機能動作時の電力よりもテスト時の電力が数倍高くなることが挙げられる。高い平均テスト電力は、発熱による回路破壊をもたらし、また高い最大テスト電力は、瞬間的な電圧降下によって信号遅延を増大させ、タイミング違反を引き起こす可能性がある。誤テスト問題は VLSI の微細化・高速化・低電源電圧化に伴い悪化しており、歩留り低下の要因となっている。特に、最大テスト電力の削減に関しては未だ十分に研究されておらず、効果的な削減手法の確立が求められている。

故障診断では、なるべく精度よく欠陥候補箇所を抽出することが望ましいが、

微細化・高速化により論理的な振舞いが予測し難い欠陥が多発するようになったため、従来の故障診断手法では候補数が増えたり、正しい欠陥候補箇所が得られなかったりすることが多い。そのため、論理的に複雑に振舞う欠陥であっても少ない候補数で正確にその箇所を特定できる故障診断手法が求められている。

本研究は、誤テストを回避するための最大テスト電力低減技術の確立と、診断分解能を向上させるための高精度故障診断技術の確立を目標に実施され、以下の研究成果が得られた。

テスト時の最大消費電力を低減するため、遺伝的アルゴリズム(以下 GA)による状態遷移削減 X 割当手法を提案した。テストベクトルの中には、故障検出とは無関係なビット (X ビット) が高い割合で存在することが分かっている。これらの X ビットに対して、状態遷移が低下するように最適な論理値を割り当てることで、最大消費電力の低減を実現できる。これまでに様々な X 割当手法が提案されているが、削減効果が低いものや実行時間が長いものが多い上、殆どの手法は回路の記憶素子あるいは回路全体の状態遷移の削減を対象としている。しかし実際には、電源線に接続するノードの数の大小などにより、電圧降下の影響を受けやすい箇所とそうでない箇所が存在する。そのため、大域的な状態遷移削減では誤テスト回避には寄与しない可能性がある。そこで本研究では、GA を X 割当問題に適用し、これらの問題の解決を試みた。実験では、提案手法が、高い状態遷移削減効果を高速に実現できることと、局所的な状態遷移削減にも効果があることを確認した。

故障診断手法の診断分解能を向上するため、X 故障モデルによる Per-Test 故障診断手法を拡張した。X 故障モデルによる Per-Test 故障診断手法は、複雑に振舞う欠陥であっても精確に表現できる X 故障モデルと、単一故障の仮定で多重故障を診断可能な Per-Test 故障診断手法を用いることで、高い診断精度を実現している。しかしながら、X 故障モデルは回路内に多数存在するビアの存在を考慮していないため、診断分解能を低下させる恐れがある。そこで本研究では、ビアを考慮した新しい X 故障モデルを提案した。さらに、閾値電圧の順序

関係を考慮した故障値の発生確率を故障候補の順位付けに導入することによって、順位付けの精度向上を試みた。実験では、提案手法により正しい故障箇所  
の平均順位が、従来の 5.8 位から 2.9 位に向上することを確認した。また、特定  
された故障箇所はビアのレベルまで分解されているため、従来手法よりも診断  
分解能が大幅に向上したといえる。

VLSI の歩留り低下は今日では産業界を脅かす大きな問題となっている。この  
問題を解決するには、低消費電力テストによる誤テスト回避と、不良解析によ  
る迅速な原因究明は必要不可欠である。本研究で提案した高速で遷移数削減率  
の高い X 割当によるテストパターン変更手法と、高精度で高分解能な故障診断  
手法は、歩留り向上に大きく貢献することが期待できる。

# 目次

|                                   |           |
|-----------------------------------|-----------|
| <b>第 1 章 序論</b> .....             | <b>1</b>  |
| 1.1 研究の背景 .....                   | 1         |
| 1.2 研究の目的 .....                   | 2         |
| 1.3 論文構成 .....                    | 3         |
| <br>                              |           |
| <b>第 2 章 VLSI のテストと故障診断</b> ..... | <b>5</b>  |
| 2.1 VLSI の設計・製造・テスト .....         | 5         |
| 2.2 VLSI のテストの基本 .....            | 6         |
| 2.2.1 故障モデル .....                 | 7         |
| 2.2.2 故障シミュレーション .....            | 11        |
| 2.2.3 テストパターン生成 .....             | 12        |
| 2.2.4 スキャン設計 .....                | 12        |
| 2.2.5 スキャンテスト方式 .....             | 15        |
| 2.3 誤テストによる歩留り低下 .....            | 17        |
| 2.3.1 CMOS 回路の消費電力 .....          | 17        |
| 2.3.2 誤テスト問題 .....                | 17        |
| 2.4 故障診断の基本 .....                 | 18        |
| 2.4.1 Cause-Effect 法による故障診断.....  | 19        |
| 2.4.2 故障診断の課題 .....               | 20        |
| <br>                              |           |
| <b>第 3 章 低消費電力テスト技術</b> .....     | <b>23</b> |
| 3.1 概説 .....                      | 23        |
| 3.2 低消費電力テスト技術の分類 .....           | 23        |
| 3.3 低消費電力 X 埋込み技術.....            | 24        |
| 3.3.1 正当化操作に基づく X 埋込み手法.....      | 25        |
| 3.3.2 確率計算に基づく X 埋込み手法.....       | 26        |

|              |                                          |           |
|--------------|------------------------------------------|-----------|
| 3.3.3        | ハイブリッド X 埋込み手法.....                      | 27        |
| 3.4          | X 埋込み手法の課題.....                          | 28        |
| <b>第 4 章</b> | <b>遺伝的アルゴリズムによる低消費電力 X 埋込み手法の提案.....</b> | <b>31</b> |
| 4.1          | 概説 .....                                 | 31        |
| 4.2          | 遺伝的アルゴリズムの基本 .....                       | 31        |
| 4.2.1        | 選択 .....                                 | 32        |
| 4.2.2        | 交叉 .....                                 | 34        |
| 4.2.3        | 突然変異 .....                               | 35        |
| 4.3          | 遺伝的アルゴリズムによる低消費電力 X 埋込み手法.....           | 35        |
| 4.4          | 評価実験 .....                               | 38        |
| 4.4.1        | 実験の準備 .....                              | 38        |
| 4.4.2        | 大域的な信号遷移削減効果の評価 .....                    | 38        |
| 4.4.3        | 局所的な信号遷移削減効果の評価 .....                    | 40        |
| 4.5          | まとめ .....                                | 41        |
| <b>第 5 章</b> | <b>X 故障診断技術.....</b>                     | <b>43</b> |
| 5.1          | 概説 .....                                 | 43        |
| 5.2          | Per-Test 故障診断 .....                      | 43        |
| 5.3          | X 故障モデル.....                             | 46        |
| 5.4          | X 故障シミュレーション.....                        | 47        |
| 5.4.1        | 概要 .....                                 | 47        |
| 5.4.2        | X 挿入.....                                | 48        |
| 5.4.3        | X 伝搬.....                                | 48        |
| 5.4.4        | X 分解.....                                | 50        |
| 5.5          | 照合における情報抽出 .....                         | 51        |
| 5.5.1        | 照合の範囲 .....                              | 51        |
| 5.5.2        | 照合の詳細 .....                              | 51        |
| 5.5.3        | 診断値 .....                                | 53        |
| 5.5.4        | 故障診断表 .....                              | 54        |

|              |                           |           |
|--------------|---------------------------|-----------|
| 5.6          | 診断情報を用いたマルチプレットの生成 .....  | 55        |
| 5.6.1        | マルチプレットのスコア付け .....       | 55        |
| 5.7          | X 故障診断のフロー .....          | 56        |
| 5.8          | X 故障診断手法の課題 .....         | 57        |
| <b>第 6 章</b> | <b>拡張 X 診断手法の提案 .....</b> | <b>59</b> |
| 6.1          | 概説 .....                  | 59        |
| 6.2          | 拡張 X 故障モデル .....          | 59        |
| 6.3          | 論理組合せ出現確率 .....           | 61        |
| 6.3.1        | 論理組合せ出現確率の計算 .....        | 61        |
| 6.3.2        | 論理組合せ出現確率の利用 .....        | 64        |
| 6.3.3        | 新しい診断値計算法 .....           | 65        |
| 6.4          | 拡張 X 故障診断手法のフロー .....     | 65        |
| 6.5          | 評価実験 .....                | 66        |
| 6.5.1        | 実験の準備 .....               | 66        |
| 6.5.2        | 拡張 X 故障モデルの評価 .....       | 67        |
| 6.5.3        | 新しい診断値の評価 .....           | 69        |
| 6.6          | まとめ .....                 | 70        |
| <b>第 7 章</b> | <b>結論 .....</b>           | <b>71</b> |
|              | <b>謝辞 .....</b>           | <b>73</b> |
|              | <b>参考文献 .....</b>         | <b>75</b> |
|              | <b>発表論文一覧 .....</b>       | <b>81</b> |



# 目次

|        |                                   |    |
|--------|-----------------------------------|----|
| 図 2-1  | VLSI 設計の主な段階                      | 5  |
| 図 2-2  | VLSI のテスト                         | 6  |
| 図 2-3  | 縮退故障                              | 8  |
| 図 2-4  | ブリッジ故障                            | 9  |
| 図 2-5  | オープン故障                            | 9  |
| 図 2-6  | 遷移遅延故障                            | 10 |
| 図 2-7  | スキャン設計による FF の可制御・可観測化            | 13 |
| 図 2-8  | MUX スキャン方式による FF のスキャン FF 化       | 13 |
| 図 2-9  | シフトモードとキャプチャモード                   | 14 |
| 図 2-10 | スキャンテストの基本動作                      | 15 |
| 図 2-11 | スキャンテストにおける遅延テスト方式                | 16 |
| 図 2-12 | 故障診断の基本的な概念                       | 18 |
| 図 2-13 | Cause-Effect 法による故障診断の概要          | 20 |
| 図 2-14 | 従来 of 故障診断手法による誤診断                | 21 |
| 図 2-15 | トランジスタの短絡により発生する故障                | 22 |
| 図 3-1  | LCP-Fill 法の例                      | 26 |
| 図 3-2  | Preferred-Fill 法の例                | 27 |
| 図 3-3  | JP-Fill 法の例                       | 28 |
| 図 4-1  | GA の基本的なフロー                       | 32 |
| 図 4-2  | 一点交叉の例                            | 34 |
| 図 4-3  | 二点交叉の例                            | 34 |
| 図 4-4  | 一様交叉の例                            | 35 |
| 図 4-5  | GA-Fill 法のフロー                     | 36 |
| 図 4-6  | s38584 での実行時間と最大遷移数削減率にみる各手法の位置付け | 40 |
| 図 5-1  | Per-Test 故障診断手法の概要                | 44 |
| 図 5-2  | SLAT ベクトル                         | 44 |
| 図 5-3  | X 故障モデル                           | 46 |

|       |                 |    |
|-------|-----------------|----|
| 図 5-4 | X 挿入            | 48 |
| 図 5-5 | X 伝搬            | 49 |
| 図 5-6 | X 分解            | 50 |
| 図 5-7 | 照合の範囲           | 51 |
| 図 5-8 | 照合の詳細           | 52 |
| 図 5-9 | X 故障診断手法の流れ     | 56 |
| 図 6-1 | ビアと X 故障        | 60 |
| 図 6-2 | 拡張 X 故障モデル      | 60 |
| 図 6-3 | 中間電圧と起こりうる論理組合せ | 61 |
| 図 6-4 | 新しい X 分解        | 64 |
| 図 6-5 | 新しい X 故障診断の流れ   | 66 |

# 表目次

|       |                       |    |
|-------|-----------------------|----|
| 表 3-1 | PPI と PPO に現れる値 ..... | 25 |
| 表 4-1 | 大域的な信号遷移削減効果の比較 ..... | 39 |
| 表 4-2 | 局所的な信号遷移削減効果の比較 ..... | 40 |
| 表 5-1 | 故障診断表 .....           | 45 |
| 表 5-2 | 診断値を用いた故障診断表 .....    | 55 |
| 表 6-1 | 論理組合せ出現確率 .....       | 63 |
| 表 6-2 | 拡張 X 故障リストの故障数 .....  | 67 |
| 表 6-3 | 故障診断結果の比較 .....       | 68 |
| 表 6-4 | First Hit の評価 .....   | 69 |



# 第1章 序論

## 1.1 研究の背景

半導体集積回路は 1960 年代に発明されて以来, その製造技術の著しい進歩に支えられながら微細化を続け, ムーアの法則に従った高集積化を可能にしてきた. それに伴って大規模化・高速化・高機能化を実現した VLSI は, 現在ではコンピュータを始めとする様々な機器に組み込まれるようになり, 現代の情報化社会の根幹を形成する重要な存在となっている. しかしその一方で, 微細化や高速化により引き起こされる様々な問題が原因となり, VLSI の歩留り低下が顕著になっている. 歩留りとは全ての製品に対する良品率のことであり, 製品の良・不良を選別するテストによって決定する.

テストでは VLSI の外部入力から, テストパターンと呼ばれる, 故障があるときと無いときとで出力応答が異なるような入力パターンを印加する. そのときの外部出力での観測応答と, あらかじめ計算しておいた期待値を比較することによって, 故障の有無を判定する. このときの判定結果は必ずしも正しいものではなく, 場合によっては判定を誤ることがある. その原因の一つにテスト消費電力がある. テストパターンは故障検出を目的とする入力パターンであるため, 機能動作では現れないような多数の状態遷移が生じる. これにより, 動的な消費電力が機能動作時の 2~3 倍増加する. 平均消費電力が高くなると発熱が起こり, 回路を破損してしまう恐れがある. 最大消費電力が高くなると, 瞬間的に電源電圧低下が起こる. 電圧低下によって信号遅延が増大し, タイミング違反を引き起こす. その結果, 正常な回路であってもテスト時に不良品と判定してしまう誤テストが発生する. 平均消費電力削減に関する研究は積極的になされ, これまでに多くの優れた手法が提案されているが, 最大消費電力削減による誤テスト回避に関してはいまだ十分に研究がなされていないのが現状である.

テストで不良と判定された VLSI は, その原因を特定するため不良解析を実施する. 不良解析によって欠陥が特定されると, その対策を設計や製造にフィ

ードバックすることによって再発を防止し、歩留り・品質・信頼性を向上させる。そのため、不良解析による不良原因の特定に長期間を費やしてしまうと、歩留り向上の大きな妨げとなる。不良解析では、回路情報、テストパターン、故障モデル、テスト応答などを用いて、ソフトウェアによって故障箇所を絞り込む故障診断を行ってから、物理解析装置による欠陥箇所の特定を行うのが一般的である。故障診断の精度は、不良解析にかかる期間に大きく影響するため、故障診断ではなるべく精度よく、より少ない候補数で故障候補を抽出することが望ましいが、微細化・高速化により論理的な振舞いが予測し難い欠陥が多発するようになったため、従来の故障診断手法では候補数が増えたり、正しい欠陥候補箇所が得られなかったりすることが多くなってきている。

以上のように、VLSIの歩留りを向上するためには、テスト時の最大消費電力削減による誤テストの回避と、複雑に振舞う欠陥であっても精度よく故障候補を求められる故障診断手法の確立は必要不可欠である。

## 1.2 研究の目的

テスト時の最大消費電力削減は、テストパターンに対する回路の信号値変化を抑制することによって実現される。したがって、信号値変化の少ないテストパターンを生成する必要がある。テスト時の最大消費電力を削減する手法として、X埋込みによる低消費電力テスト技術が提案されている。X埋込みでは、テストパターン中の故障検出とは無関係なビットに、信号値変化が少なくなるように論理値を埋め込む。これにより、テストパターンの故障検出能力を維持しつつ回路の信号値変化の少ないテストパターンを得ることができる。しかし、従来のX埋込み手法は、削減効果が低いものや実行時間が長いものが多い上、殆どの手法は回路の記憶素子あるいは回路全体の状態遷移の削減を対象としている。しかし実際には、電源線に接続するノードの数の大小などにより、電圧降下の影響を受けやすい箇所とそうでない箇所が存在する。そのため、大域的な状態遷移削減では誤テスト回避には繋がらない可能性がある。

X故障診断手法[13]は、複雑な欠陥であっても正確に表現できるX故障モデルと、単一故障の仮定で多重故障を診断可能なPer-Test故障診断方式を組み合

わせた故障診断手法である。また、故障候補の可観測性や照合時の一致不一致情報を診断値として数値化し、回路に存在する可能性の高い順に故障候補を順位付けすることができる。X 故障診断手法は、多重・動的な欠陥に対しても精度よく診断可能であることが示されている。しかし、X 故障診断手法ではファンアウトに存在するビアを考慮しておらず、全てのファンアウト信号線を直接接続された信号線として扱っている。このため、微細化 LSI で頻繁に発生するビアの断線などの欠陥箇所を詳細に特定することができず、診断分解能の低下をもたらす恐れがある。また、欠陥によってファンアウト信号線に発生する故障値の出現確率は実際には異なるが、これを全て等しいものとして扱っているため、診断精度に影響を及ぼす可能性がある。

本研究では誤テストを回避するための最大テスト電力低減技術の確立と、診断分解能を向上させるための高精度故障診断技術の確立によって、VLSI の歩留り向上に貢献することを目的とする。これを達成するため、以下のことを具体的な目標とする。

1. 従来の X 埋込み手法よりも高速かつ効果的な新しい X 埋込み手法を提案する。また、提案する X 埋込み手法によって回路内の任意の領域における信号遷移の抑制も可能にする。
2. X 故障診断手法を拡張し、診断分解能と故障候補の順位付け精度を向上する。
3. ベンチマーク回路に対する実験によって、提案手法の有効性を確かめる。

### 1.3 論文構成

本論文は次のように構成される。第 1 章では、本研究の背景、目的と論文構成について述べる。第 2 章では VLSI のテスト、診断の基本概念について述べる。第 3 章ではこれまでに提案されている低消費電力テスト技術の概要を述べ、それらの課題を明らかにする。第 4 章では遺伝的アルゴリズムによる低消費電力 X 埋込み技術を提案し、ベンチマーク回路に対する実験によって提案手法の有効性を示す。第 5 章では提案する故障診断手法の基

礎となる X 故障診断手法とその問題点について述べる. 第 6 章では X 故障診断手法を拡張した新しい故障診断手法を提案し, ベンチマーク回路に対する実験によって提案手法の有効性を示す. 第 7 章では本論文のまとめと今後の課題について述べる.



## 第2章 VLSI のテストと故障診断

### 2.1 VLSI の設計・製造・テスト

VLSI の設計は通常、複数の工程に分かれている。最上位の設計データである要求仕様にはじまり、設計段階を経るごとに詳細化されていく。図 2-1 に VLSI が製造されるまでの主な設計段階とそのときの回路記述法を示す。

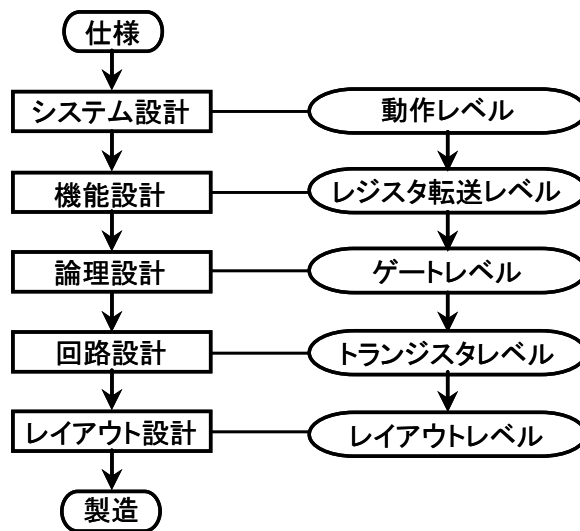


図 2-1 VLSI 設計の主な段階

システム設計での回路記述を、動作レベル (Behavior Level) といい、回路の外部入出力端子と、回路全体で実行する処理で記述する。機能設計での回路記述をレジスタ転送レベル (RTL : Register Transfer Level) といい、ALU、レジスタやメモリを用いて表現する。論理設計での回路記述をゲートレベルといい、論理ゲートやフリップフロップ (FF) を用いて表現する。回路設計での回路記述をトランジスタレベルといい、トランジスタ、抵抗や容量などを用いて表現する。レイアウト設計での回路記述をレイアウトレベルといい、金属、ポリシリコン、コンタクトなどを表す長方形で回路を表現する。レイアウト設計が済むと、設計されたレイアウトデータを基に製造が行われる。

VLSI は、製造時の欠陥の混入を完全に防ぐことはできないため、設計と仕様が正しくても、出来上がった回路が正しく動作しないことがある。欠陥の種類

としては、シリコン基盤の汚れ、マスクのずれや配線の腐食など様々で、これらの欠陥により短絡や断線などが発生し、その影響が出力の誤りとして現れる。そのため、製造された VLSI が正しく動作するか否かを判定する必要がある。この作業をテスト(Testing)という。さらに、テストで不良品と判定された VLSI については、不良解析(Failure Analysis)によって欠陥の位置特定と原因分析が行われ、その結果は設計や製造にフィードバックされる。不良解析は、回路情報、テストベクトル集合、出力応答等を用いてソフトウェアによって欠陥箇所を推定する故障診断(Fault Diagnosis)と、電子ビームテスタやエミッション顕微鏡等の解析装置によって欠陥箇所やその発生原因を特定する物理解析からなる。不良解析からのフィードバックによって、歩留り（生産された全ての製品に対する良品の割合）、品質、信頼性の向上がなされる。

## 2.2 VLSI のテストの基本

VLSI のテストは基本的に、製造された VLSI の外部入力から入力パターンを印加し、外部出力で回路の出力応答を観測することによって行われる。テスト時に用いる入力系列をテストパターン(Test Pattern)あるいはテストベクトル(Test Vector)という。

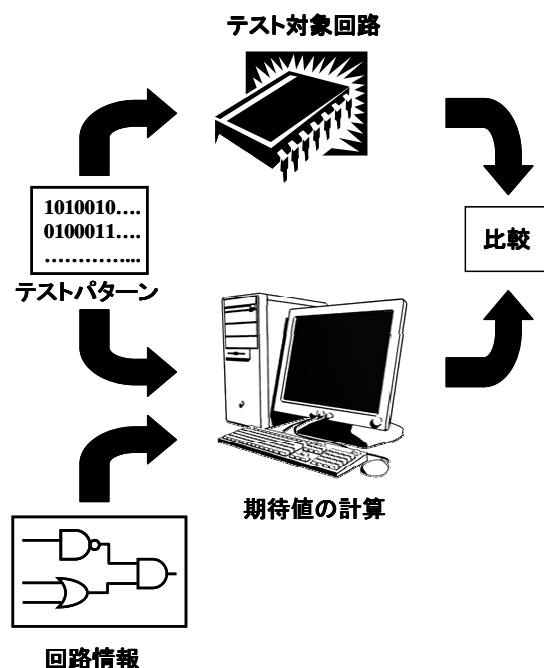


図 2-2 VLSI のテスト

図 2-2 のように、製造した VLSI の外部入力からテストパターンを印加し、その出力応答と前もって計算機で求めておいた期待値とを比較して、それらが一致していなければ不良と判定される。

## 2.2.1 故障モデル

生成可能な全ての組合せのテストパターンを印加すれば、より多くの故障を検出できる。これを全数テスト (Exhaustive Testing) といい、 $n$  入力の組合せ回路には  $2^n$  通りのテストパターンの組合せがある。全数テストは、 $n$  が大きくなるとテスト実行時間が長くなりすぎるため、現実的ではない。そこで、起こりうる欠陥の論理的振舞いをモデル化した故障モデル (Fault Model) によって故障を仮定し、その故障を検出するようなテストを考える必要がある。

故障モデルには、起こりやすい欠陥の振舞いをモデル化していることが求められるが、故障検出においては、故障モデルが欠陥の振舞いと完全に一致しなくても、それを対象に生成したテストパターンが結果的に他の故障モデルで表現される故障も検出できることが重要である。

回路内に存在する故障はただ一つであると仮定する故障モデルを単一故障 (Single Fault)、複数箇所での発生も許容する故障モデルを多重故障 (Multiple Fault) といい、他の故障モデルとともに付加的に考慮される。単一故障を考えた場合の故障数が  $N$  のとき、 $k$  重故障では故障箇所の組合せは  $N^k$  となるため、 $k$  が大きくなると故障数は爆発的に増大する。また、互いに関係のない箇所に同時に存在している複数の故障に対しては、いずれか 1 箇所の単一故障を検出できるテストパターンを用いれば故障は検出されるため、多重故障は単一故障に比べて検出が容易である。以上の理由から、単一故障が用いられることが多い。

以下では代表的な故障モデルについて述べる。

### 2.2.1.1 縮退故障モデル

信号線がグラウンドに短絡すると、その信号線の論理値は 0 に固定され、電

源線に短絡すると、その信号線の論理値は1に固定される。このような欠陥の振舞いを表現した故障モデルを縮退故障(Stuck-at Fault)モデルという。特に、論理値が0に固定する故障を0縮退故障といい、1に固定する故障を1縮退故障という。信号線数  $n$  の回路で単一縮退故障モデルを使用する場合、各信号線の0縮退故障と1縮退故障が考えられるので、故障数は  $2n$  となる。縮退故障ではモデル化できない故障も多数存在するが、縮退故障を対象に生成されたテストパターンで他の多くの故障も検出できることが経験的に知られている。また、単一縮退故障モデルは計算機での扱いも非常に容易である。これらの理由から、単一縮退故障モデルは現在のテストにおいて最も広く用いられている。

縮退故障の例を図 2-3 に示す。

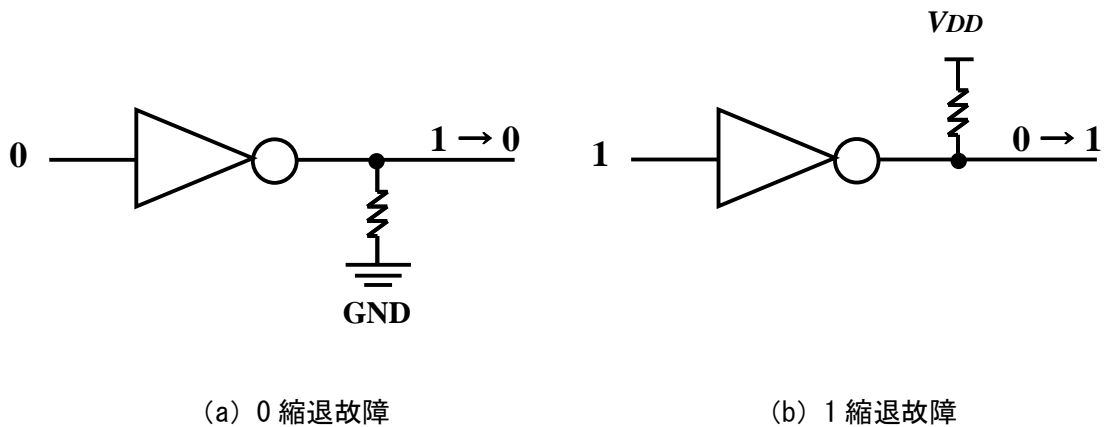
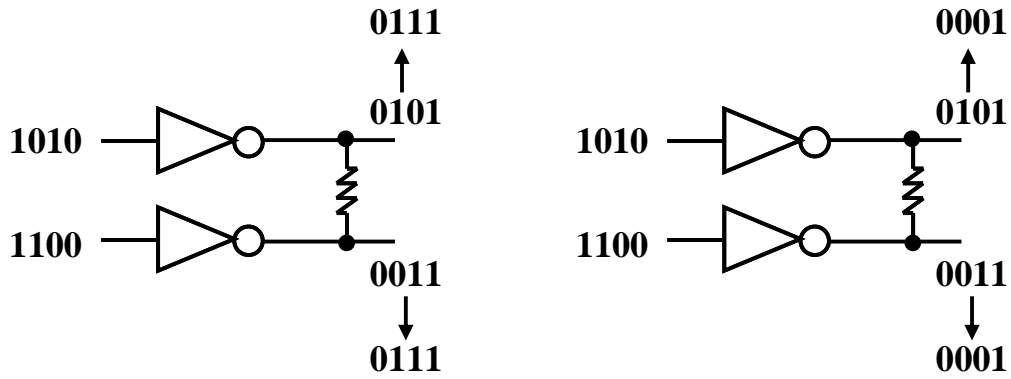


図 2-3 縮退故障

### 2.2.1.2 ブリッジ故障モデル

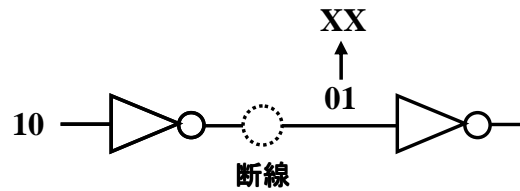
ブリッジ故障(Bridging Fault)は本来接続されていない二つの信号線が短絡する故障モデルである。短絡によって発生する故障値は、短絡抵抗の大きさ等によって決定する。ANDブリッジ故障は短絡による故障値がAND論理をとるような場合の故障モデルで、ORブリッジ故障は故障値がOR論理をとるような場合の故障モデルである。図 2-4 にブリッジ故障の例を示す。



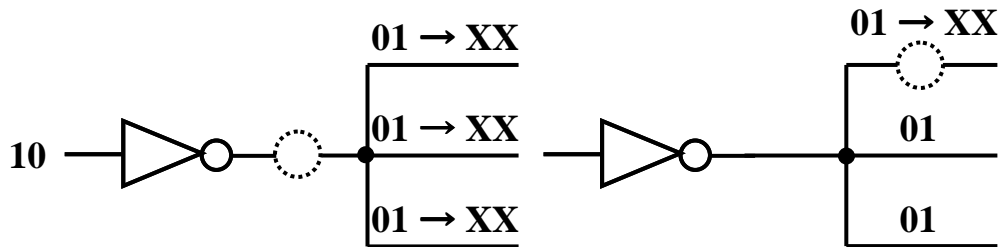
(a) ANDブリッジ故障

(b) ORブリッジ故障

図 2-4 ブリッジ故障



(a) ファンアウトのないゲート出力のオープン故障



(b) ファンアウトシステムのオープン故障 (c) ファンアウトブランチのオープン故障

図 2-5 オープン故障

### 2.2.1.3 オープン故障モデル

オープン故障(Open Fault)は、断線の振舞いをモデル化したものである。図 2-5(a) のようにゲートの出力で断線が起こった場合、出力信号線の電圧を事前に予測することは困難なため、論理値は不定値 X となる。また、ファンアウト

のあるゲートの場合、ファンアウトシステムで断線が起こった場合とファンアウトブランチで断線が起こった場合とで、その振舞いは異なる。図 2-5(b)のようにファンアウトシステムで断線が起こった場合、全てのファンアウトブランチの論理値が不定値となるが、図 2-5(c)のように、ファンアウトブランチで断線が起こった場合は、他のファンアウトブランチは正常に動作する。

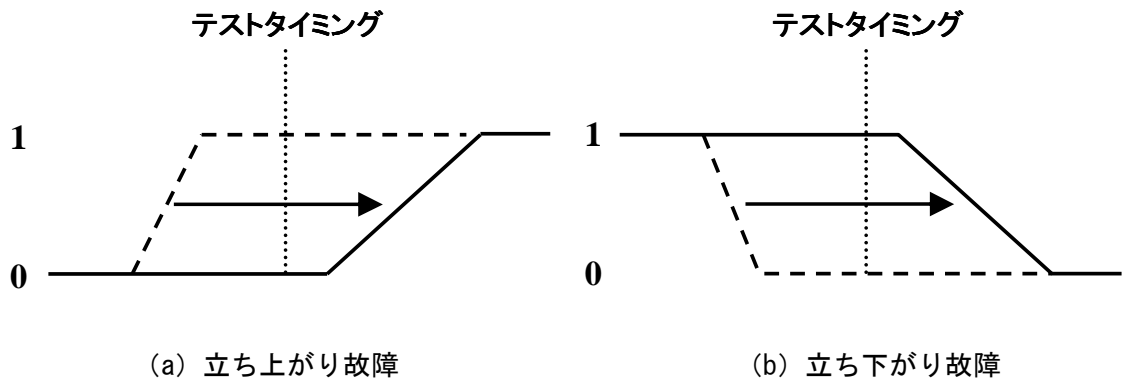


図 2-6 遷移遅延故障

#### 2.2.1.4 遅延故障モデル

遅延故障(Delay Fault)は、半断線などの欠陥による信号線の高抵抗化や、隣接信号線の信号変化の影響、電源ノイズなどによって、許容された遅延時間の内に信号が伝搬しない振舞いをモデル化した故障である。信号値の立ち上がりあるいは立ち下りの時間が増大し、フリップフロップ (以下 FF) の入力値が変化する前に FF に値が格納されることによって、誤動作を生じる。遅延故障は信号の変化を考えた故障モデルであるため、これを検出するには、信号変化前の値と信号変化後の値を設定する連続したテストパターンが必要になる。信号変化前の信号を設定するテストパターンを初期化パターン、信号変化後の信号を設定するテストパターンを検出パターンといい、このようなテストを2パターンテストという。遅延故障を検出するためには、1パターン目と2パターン目の印加をシステムクロックと同等の速度で行わなければならない。このようなテスト方法を実速度テスト (At-speed Testing) という。遅延故障は、遅延が増大する箇所の範囲の考え方によってさらに分類されるが、代表的なものに遷移遅延故障 (Transition Delay Fault) がある。遷移遅延故障は最も単純な遅延故

障で、一つの信号線の遅延の増大によって FF に誤った値を取り込ませる故障である。図 2-6 に遷移遅延故障の例を示す。立ち上がり（立ち下がり）故障の検出には、その箇所には、初期化パターンで 0 (1) を設定し、検出パターンで 1 (0) を設定する必要がある。

## 2.2.2 故障シミュレーション

故障モデルが決まると、テストパターンが検出できる故障を求めることができる。仮定した故障のうち、テストパターン集合によって検出可能な故障の割合を故障検出率 (Fault Coverage) といい、次式で表される。

$$\text{故障検出率(\%)} = \frac{\text{検出可能故障数}}{\text{総仮定故障数}} \times 100$$

故障検出率は、テストパターンの品質を示す指標となり、100%に近いほど高品質であるといえるが、故障を検出するテストパターンは必ず存在するとは限らない。どのようなテストパターンに対しても検出することが出来ない故障をテスト不能故障 (Untestable Fault) といい、テスト不能故障を除いた場合のテスト品質の指標を、故障検出効率 (Fault Efficiency) といい、次式で表される。

$$\text{故障検出効率(\%)} = \frac{\text{検出可能故障数}}{\text{総仮定故障数} - \text{テスト不能故障数}} \times 100$$

テストパターンが故障を検出できるかどうかは、正常回路の応答と故障を仮定した回路の応答を比較することによって求めることができる。テストパターンに対する論理回路の応答を計算する処理を論理シミュレーション (Logic Simulation) という。正常回路と故障回路に対して論理シミュレーションを行い、出力応答が異なっていれば、その故障は検出できると判定される。故障回路に対する論理シミュレーションを特に、故障シミュレーション (Fault Simulation) という。故障シミュレーションは、故障検出率の計算だけでなく、テストパターン生成や故障診断でも用いられる。

### 2.2.3 テストパターン生成

回路内の故障を検出するテストパターンを求める処理を、テストパターン生成 (Test Generation, テスト生成) といい、特に計算機によってアルゴリズム的にテストパターンを求める処理を ATPG (Automatic Test Pattern Generation : 自動テストパターン生成) という。テスト生成では、アルゴリズムによる生成の前にランダムにパターンを発生するのが一般的である。これは、多くの故障はランダムに発生したパターンで検出できるため、設定した故障検出率に達するまでランダムパターンに対する故障シミュレーションを繰り返し行い、残った故障に対してだけ、アルゴリズムでテストパターンを生成する。ATPG には、故障検出率の高いテストパターン集合を、なるべく少ないテストパターン数で高速に求めることが求められる。ATPG アルゴリズムの研究は古くから取り組まれており、これまでに様々な手法が提案されている。その中でも、大規模な回路にも適用可能な経路活性化法 (Path Sensitization Method) に基づく手法が実用的によく用いられている。経路活性化法に基づくテスト生成アルゴリズムは D アルゴリズム [1]をはじめ、PODEM [3], FAN [4], SOCRATES [5]など多くの手法が提案されている。

### 2.2.4 スキャン設計

順序回路のテスト生成は、組合せ回路のそれに比べて非常に困難である。その原因は、順序回路が記憶素子をもち、それを外部から直接制御・観測できないことにある。組合せ回路は、現在の入力によって出力が決定するが、順序回路は現在の入力と過去の入力を合わせて出力が決定する。順序回路では複数の時刻の回路の動作を考慮しなければならないため、テスト生成の際は順序回路の各時刻の動作を空間的に展開して行うが、初期状態の設定と故障伝搬にかかる時刻数が不明なこと、入力数と素子数が時刻の数に比例して増加することなどが問題となってくる。そのため、大規模な順序回路では、テスト容易化設計 (DFT : Design For Testability) によって、設計段階でテストを考慮に入れる。様々なテスト容易化設計があるが、代表的なものにスキャン設計がある。スキャン設計では、フリップフロップ (以下 FF) に付加回路を挿入し、FF の値を



外部から直接制御・観測できるようにする。このような FF をスキャン FF という。FF の値を自由に制御・観測できれば、順序回路は図 2-7 のようにスキャン FF の入出力で切り離された組合せ回路とみなすことができる。スキャン FF の出力を擬似外部入力(PPI : Pseudo Primary Input)といい、スキャン FF の入力を擬似外部出力(PPO : Pseudo Primary Output)という。

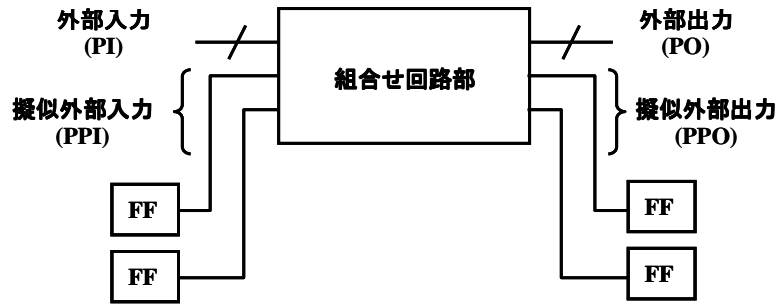


図 2-7 スキャン設計による FF の可制御・可観測化

図 2-8 に、代表的なスキャン設計方式である、マルチプレクスト・スキャン方式の概略を示す。本方式では、FF の入力に 2 入力マルチプレクサ (以下 MUX と表す) を付加する。MUX の制御信号によって通常動作モードとスキャン動作モードの切り替えを行う。

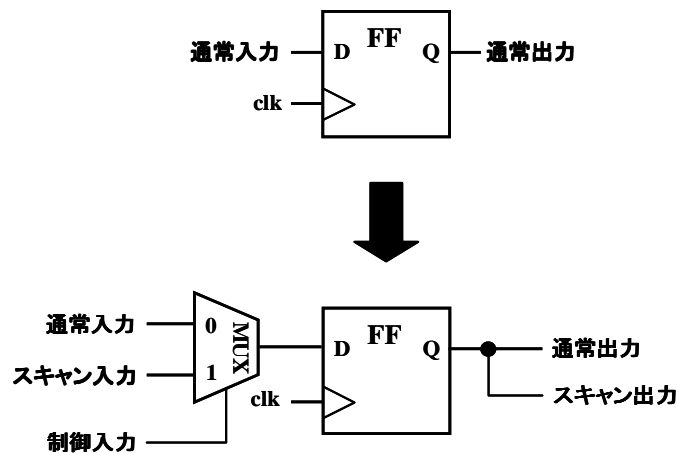
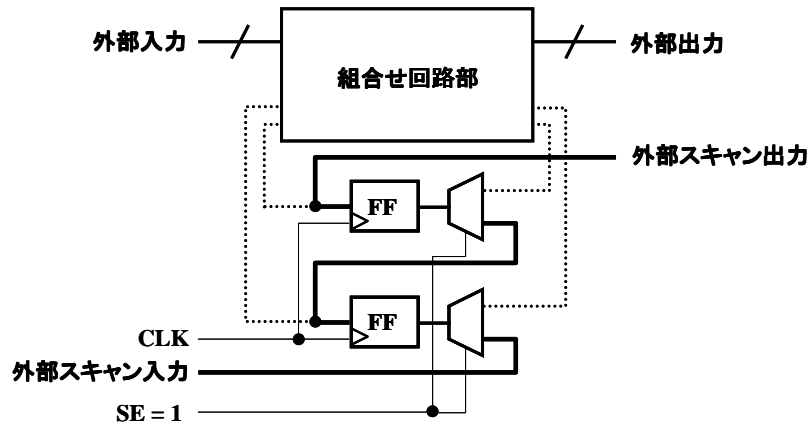
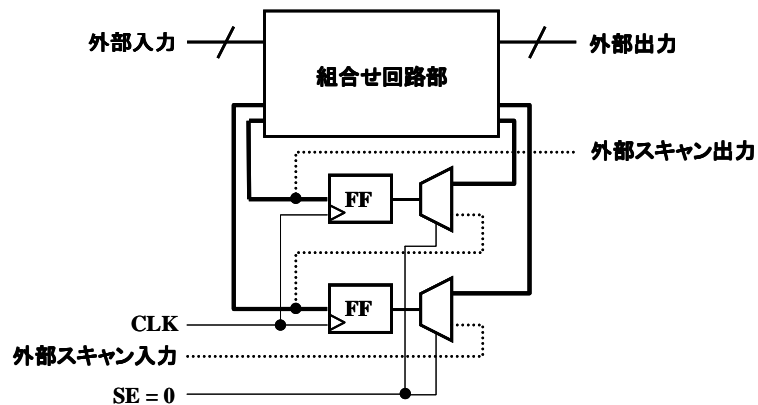


図 2-8 MUX スキャン方式による FF のスキャン FF 化

スキャン動作モードはシフトモードとも呼ばれ、スキャン FF がシフトレジスタとして動作する。通常動作モードはキャプチャモードとも呼ばれ、回路の出力応答をスキャン FF に取り込む。図 2-9 に各動作モードでの回路を示す。



(a) シフトモード



(b) キャプチャモード

図 2-9 シフトモードとキャプチャモード

シフトモードでは、図 2-9 (a) のように、MUX の制御信号に接続する外部入力 SE (Scan Enable) を 1 に設定すると、外部スキャン入力から外部スキャン出力までのパスが形成される。これをスキャンチェーンといい、外部スキャン入力から論理値を順次印加し、シフトレジスタ機能によってスキャンチェーン上のスキャン FF に設定する。また、外部スキャン出力で各スキャン FF の論理値を観測する。キャプチャモードでは、図 2-9 (b) のように、SE を 0 に設定すると、通常動作と同じパスが形成され、回路の応答をスキャン FF に格納することができる。

スキャン設計を用いれば順序回路の FF の値を直接制御・観測できるため、順序回路のテスト生成問題を組合せ回路のテスト生成問題として扱うことがで

きるようになる。しかし、スキャン設計は FF に付加回路を挿入するため、面積増加や性能劣化などの問題点がある。そのため、大規模な回路では、全ての FF をスキャン FF 化するフルスキャン設計ではなく、テスト容易化に特に必要な FF のみをスキャン FF 化する部分スキャン設計を用いることもある。

## 2.2.5 スキャンテスト方式

スキャンテストでは基本的に、まずシフトモードで、外部スキャン入力からテストパターンをスキャン FF に設定する。次にキャプチャモードに切り替えてテスト応答をスキャン FF に格納し、再びシフトモードに切り替えて、格納されたテスト応答を外部スキャンアウトで観測する。図 2-10 にスキャンテストのタイミングチャートを示す。

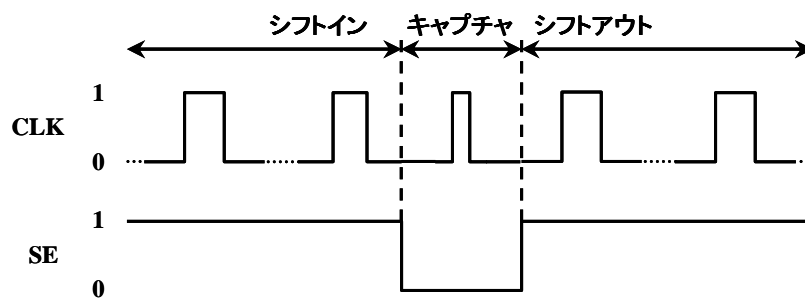
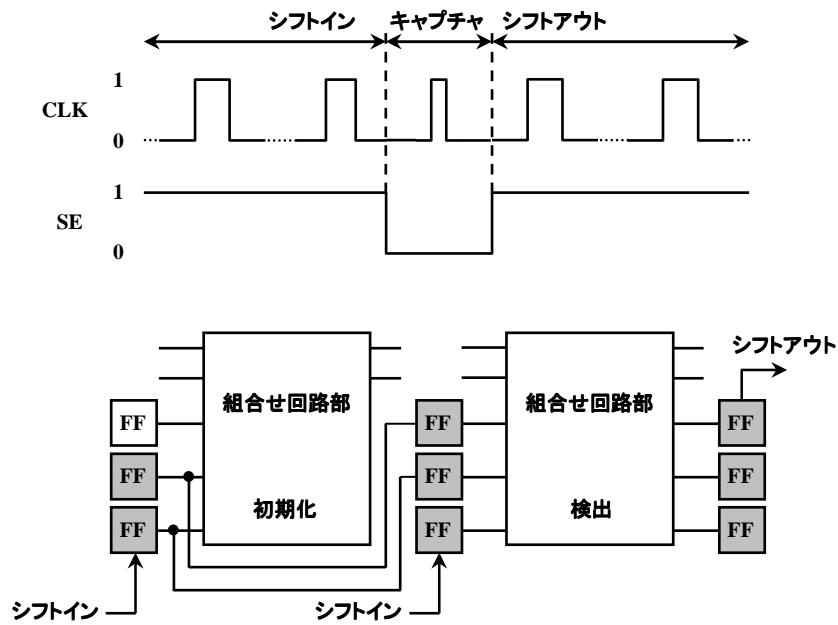
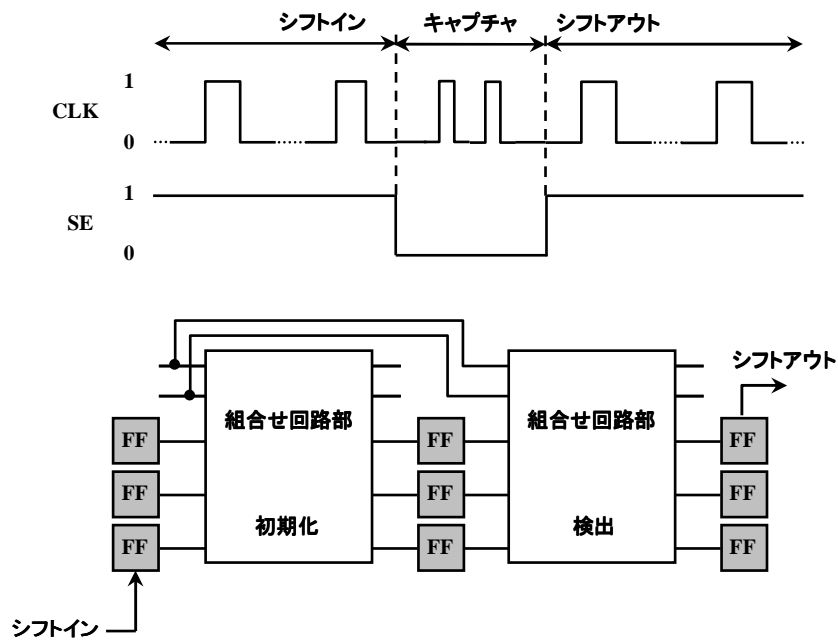


図 2-10 スキャンテストの基本動作

図 2-10 のように、テストパターンの印加とテスト応答の観測を行うには、スキャン FF の数だけクロックを印加する必要がある。一方、遅延故障を検出する 2 パターンテストでは、1 パターン目と 2 パターン目の印加を実速度 (At-speed) で行うため、単純なスキャンテストでは 2 パターンテストを実速度で行うことができない。そのため、2 パターン目の印加を 1 クロックで行うようなスキャンテスト方式が必要である。スキャンテストにおける代表的な実速度テスト手法として、スキュードロード (Skewed-load, LoS : Launch on Shift) 方式[57, 58]と、ブロードサイド (Broad-side, LoC : Launch on Capture) 方式[56]がある。図 2-11 に各方式のタイミングチャートと時間展開モデルによる回路の動作を示す。



(a) スキュードロード方式



(b) ブロードサイド方式

図 2-11 スキャンテストにおける遅延テスト方式

スキュードロード方式では、シフトインの最後の2クロックで初期化パターンと検出パターンを設定した後、システムクロックの速度でキャプチャを行い、

テスト応答を格納する。一方ブロードサイド方式では、シフトインの最後のクロックで初期化パターンを設定した後、システムクロックの速度で2回キャプチャを行う。つまり、1回目のキャプチャ動作で検出パターンを設定し、2回目のキャプチャ動作でテスト応答を格納する。

スキュードロード方式は、構造は単純であるが、キャプチャ動作はシステムクロックの速度で行われるため、高速な回路になると SE 信号を切り替えるタイミングに高い精度が要求される。ブロードサイド方式は、SE 信号を切り替えるタイミング精度はスキュードロードに比べて緩いが、組合せ回路部の論理によっては、検出パターンの設定が困難になってしまうことがある。

## 2.3 誤テストによる歩留り低下

### 2.3.1 CMOS 回路の消費電力

CMOS 回路の消費電力は、負荷容量の充放電による動的消費電力と、トランジスタのリーク電流による静的消費電力に分けることができる。このうち、入力値の変化に伴って生じる動的な消費電力は全消費電力の大部分を占める。そのため、回路全体の消費電力を低くするためには、動的な消費電力を抑えることが重要である。動的な消費電力は、近似的に次式で与えられる。

$$\sum_j \frac{1}{2} \times C_L(j) \times VDD^2 \times f_p(j)$$

ここで、 $C_L(j)$  はゲート  $j$  の負荷容量、 $VDD$  は電源電圧、 $f_p(j)$  はゲート  $j$  の信号値変化回数である。CMOS 回路の消費電力はゲートの信号値変化数に比例して高くなる。

### 2.3.2 誤テスト問題

スキャンテストでは、回路に通常動作とは異なる特殊な動作をさせるため、通常動作では起こりえない状態遷移が発生する。それに伴ってゲートの信号値変化数が多くなり、通常動作時よりもテスト時の消費電力が数倍高くなる。

スキャンテストはその大部分をシフト動作が占めるため、シフト時の信号値変化数が多くなると平均消費電力が増加し、発熱による回路の誤動作をもたらす。一方、最大消費電力が増加すると、瞬間的に電源電圧が低下する IR ドロップが発生する。過度の IR ドロップが発生すると信号遅延が設計上の許容範囲を超過してしまい、タイミング違反を引き起こす。そうするとスキャン FF に誤った値が取り込まれてしまうため、故障のない回路であっても不良品と誤って判定してしまう。これを誤テストという。特に実速度スキャンテストは、キャプチャ動作がシステムクロックと同等の速度で行われるため、遅延増加の影響を受けやすい。

## 2.4 故障診断の基本

テストで不良と判定された回路の故障箇所を特定する処理を故障診断 (fault diagnosis) といい、回路情報、テストパターン、故障モデル、テスト応答を用いてソフトウェアによって故障箇所を推定する。故障診断では図 2-12 のように、出力応答が誤っている (フェイルしている) 外部出力 (PO) あるいは擬似外部出力 (PPO) に伝搬しうる故障を求めるのが基本となる。図ではゲート B の入力二本と、ファンアウトの幹 (ステム) の、合計三箇所が故障候補となる。

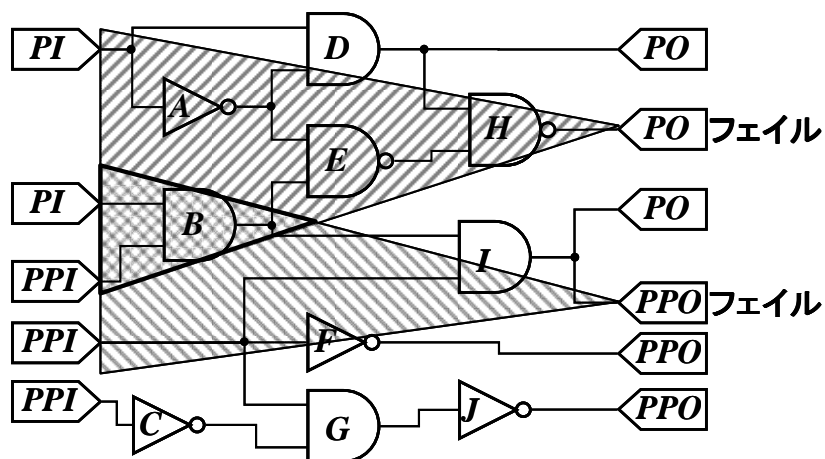


図 2-12 故障診断の基本的な概念

故障診断には Cause-Effect 法[59]と Effect-Cause 法[60]の二つの方式がある。Cause-Effect 法では、故障モデルにより仮定した故障(cause)のシミュレーション

結果と実回路の観測応答(effect)を照合し、一致・不一致の情報を用いて故障候補を絞り込む。以前は、あらかじめ故障シミュレーションを行ってテストパターンとフェイルする出力の対応表を作成しておく故障辞書 (fault dictionary) 法 [59]が用いられていたが、大規模回路になると大量のメモリを必要とするため、故障シミュレーションを実行しながら実応答との照合を行う手法も用いられている。一方、Effect-Cause 法ではフェイルしている外部出力(effect)から故障伝搬経路を追跡し、故障箇所(cause)を求めることにより故障候補を絞り込む。故障候補が絞り込まれた後は、故障モデルとの一致を調べる。Effect-Cause 法は事前に故障モデルを仮定し、故障辞書を作成する必要がないため、メモリ消費は少なく済むが、Cause-Effect 法に比べて故障候補が多い傾向がある。

どちらの手法も故障モデルを用いるため、故障診断の精度は故障モデルの妥当性に大きく依存する。故障モデルは欠陥の振舞いを正確に表現できることが必要であるが、モデル化が詳細すぎると診断の効率が低下する恐れもある。以下では、本論文で対象とする Cause-Effect 法について述べる。

#### 2.4.1 Cause-Effect 法による故障診断

Cause-Effect 法による故障診断では、事前に故障モデルを仮定する。故障モデルによって回路に存在しうる故障をリスト化し、それぞれの故障のシミュレーション応答を実応答と照合するのが一般的である。このときの故障モデルには、単一縮退故障モデルがよく用いられる。照合では、故障シミュレーションの結果が故障診断手法の評価基準下で実応答と一致すればその故障が回路内に存在すると考える。図 2-13 に Cause-Effect 法による故障診断の概要を示す。

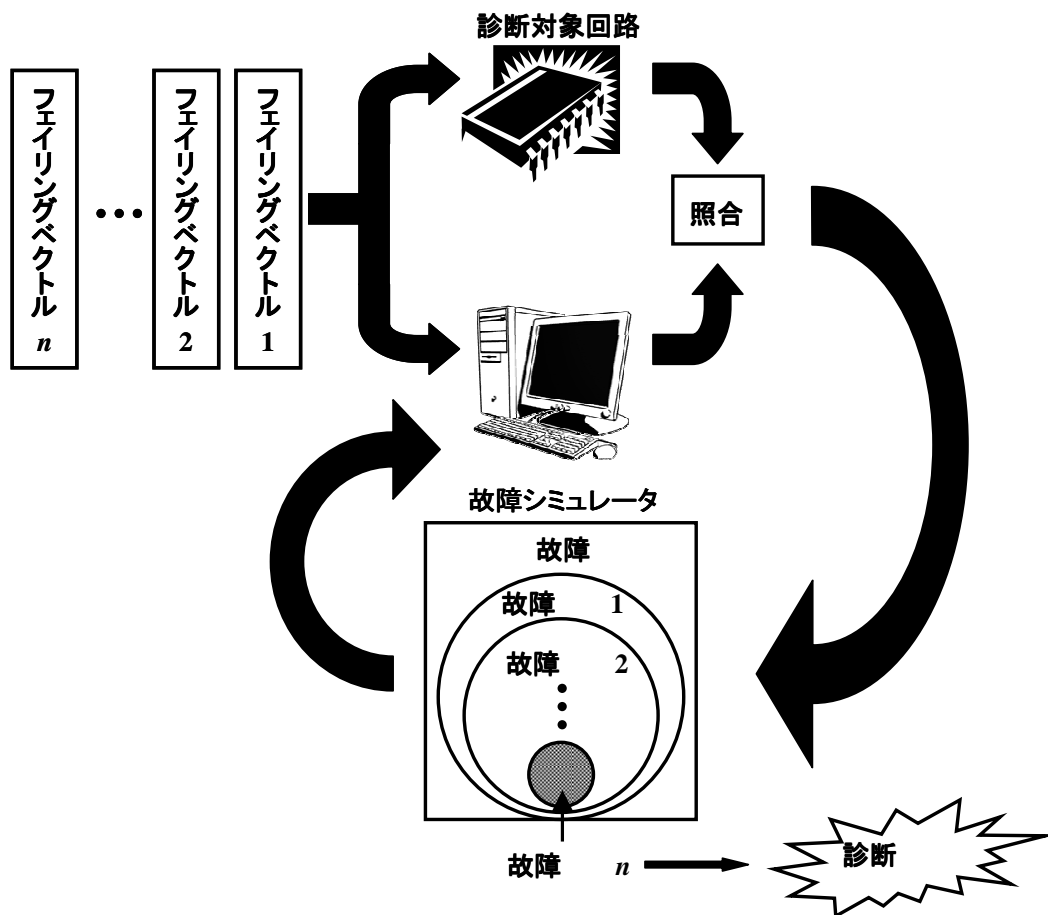


図 2-13 Cause-Effect 法による故障診断の概要

故障診断において使用される入力ベクトルには、実応答に誤りが現れるフェイリングベクトルのみを用いることが多い。従来の故障診断手法では、全てのフェイリングベクトルにおいて応答が一致する故障を故障候補とするため、あるフェイリングベクトルにおいて応答が一致しなかった故障は、故障候補から除外される。これにより、故障シミュレーションにかかる時間を削減することができる。

#### 2.4.2 故障診断の課題

従来の単一縮退故障による故障診断手法によって複数の信号線に影響するような欠陥をもつ回路に対して故障診断を行う場合、誤った診断を行ってしまう可能性がある。図 2-14 にその例を示す。



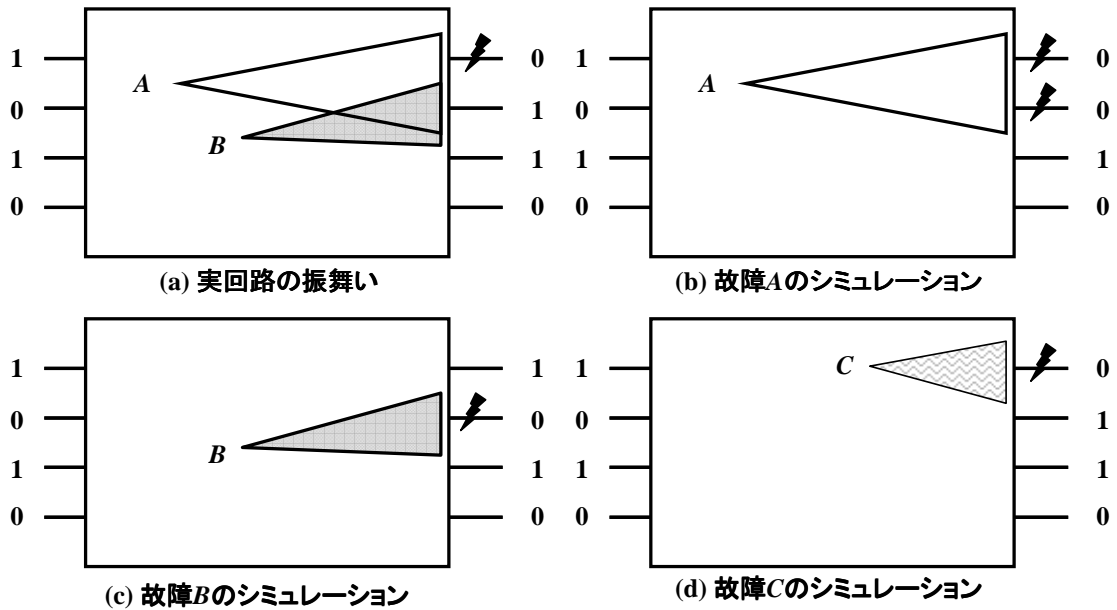


図 2-14 従来の故障診断手法による誤診断

図 2-14(a)のように、実回路上で故障 A と故障 B と同等の振舞いをする欠陥が発生しているとする。このとき外部出力に現れる応答は、一番目の外部出力でのみ誤りが現れ、二番目の外部出力では多重故障の影響により誤りが現れなかったとする。図 2-14(b)では、故障 A の単一故障シミュレーションを行っているが、故障 B の影響がなくなるため二箇所の外部出力に誤りが現れ、実応答とは一致しない。同様に図 2-14(c)では故障 B の単一故障シミュレーションを行っているが、故障 B の影響により、二番目の外部出力に誤りが現れ、実応答とは一致しない。図 2-14(d)のように、一番目の外部出力のみに影響するような故障 C のシミュレーション応答が実応答と一致し、明らかに誤った診断となる。このように、多重欠陥が発生している回路に対して従来の故障診断手法を適用すると、本来の故障箇所とは異なる故障が候補として挙げられてしまう可能性がある。

LSI の微細化・高速化によって、物理欠陥の振舞いは複雑化している。電源やグラウンドへの短絡ならば、縮退故障で表現することができるが、信号線どうしの短絡、断線などの欠陥では、論理値が浮動してしまうため、単純な故障モデルでは欠陥の振舞いを正確に表現することが困難である。図 2-15 に、トランジスタの短絡により発生する故障の振舞いを示す。

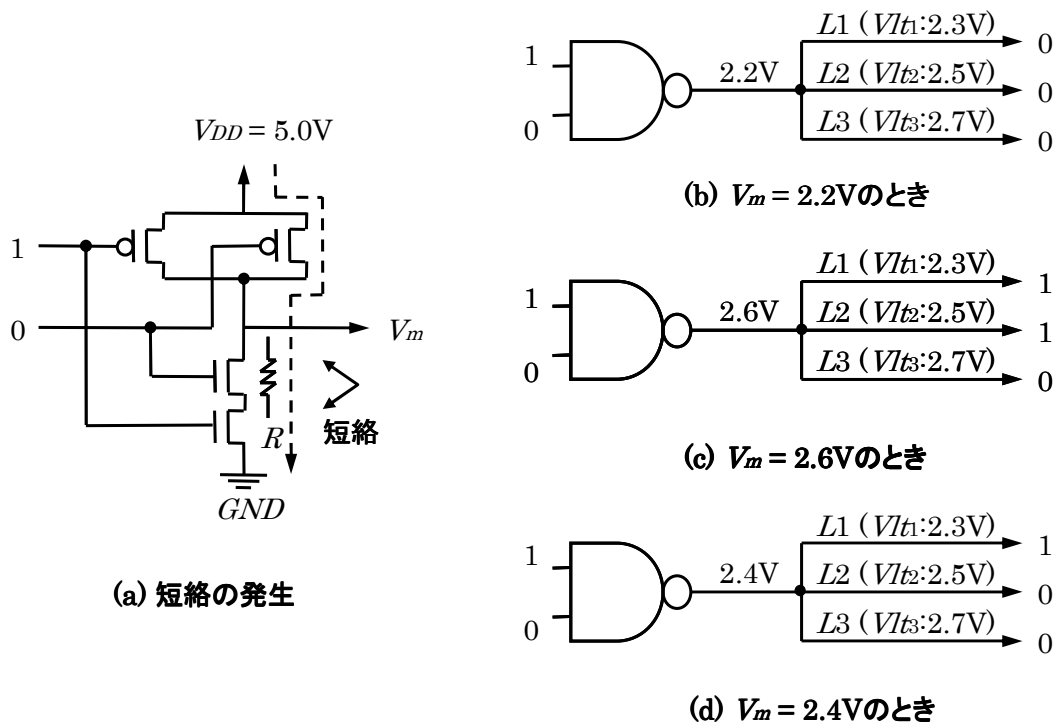


図 2-15 トランジスタの短絡により発生する故障

図 2-15(a)のようにトランジスタに短絡が発生した場合、欠陥電圧  $V_m$  が出力される。ファンアウト信号線の論理値は出力電圧と閾値電圧の大小関係によって決定する。 $V_m = 2.2V$  のとき、ファンアウト信号線  $L1 \sim L3$  の閾値電圧  $V_{LT1} \sim V_{LT3}$  を下回るので、図 2-15(b)のように  $L1 \sim L3$  の論理値は全て 0 となる。この場合はファンアウトシステムの 0 縮退故障として表すことができる。 $V_m = 2.6V$  のときは、図 2-15(c)のように信号線  $L3$  のみが論理値 0 となり、信号線  $L3$  の縮退故障として表すことができる。しかし、 $V_m = 2.4V$  のとき信号線  $L2$  と  $L3$  が論理値 0 となるため、単一縮退故障では表すことができない。欠陥電圧  $V_m$  の値が予測することができない上、微細化 LSI におけるプロセス変動により、信号線の閾値電圧も回路によって異なるため、このような欠陥の振舞いを正確にモデル化するのは非常に困難である。

以上の点から故障診断では、複雑に振舞う欠陥であっても正確に表現できる故障モデルと、多重故障を精度よく診断できる故障診断手法が求められている。

## 第3章 低消費電力テスト技術

### 3.1 概説

スキャンテスト時の消費電力削減は、シフト時とキャプチャ時の両方を考慮しなければならない。これまで、シフト時の消費電力削減に関する研究は積極的に行われてきた。その代表的な手法としては、シフト動作のタイミングをずらす手法や、スキャンチェーン上の連続する FF になるべく同じ論理値を持つようにテストベクトルを生成する、もしくはスキャンチェーン上の FF を並べ替える手法などが挙げられる。これに対し、実速度で行われるキャプチャ動作はタイミング制約があり、シフト時ほど柔軟に消費電力に対応することができないため、消費電力削減は比較的困難である。その一方でキャプチャ時の誤動作に起因する誤テスト問題は深刻化しており、キャプチャ時の消費電力を効果的に削減する技術が求められている。本章では、本論文で対象とする誤テストを回避するための低消費電力テスト技術について述べる。

### 3.2 低消費電力テスト技術の分類

スキャンテストにおけるキャプチャ時の消費電力を削減する技術は、回路変更に基づくものと、テスト生成に基づくものの二種類に大別される。回路変更に基づく手法には、信号値変化を抑えるための付加回路を挿入する手法 [29, 30]、スキャンチェーンを分割する手法 [31]、スキャンチェーン上の FF の順序を変更する手法 [32]、部分キャプチャ法 [33]などがあり、消費電力削減効果は大きいですが、回路面積の増大や、性能劣化などの問題がある。

テスト生成に基づく手法には、消費電力を考慮に入れてテストベクトルを生成する手法 [35, 36] と、故障検出を念頭にテストベクトルを生成した後に、消費電力が低下するようにテストベクトルを変更する手法 [37-42] がある。テストベクトル生成の時点で消費電力を考慮に入れると、所望の故障検出率に達するまでにテストベクトル数が通常のテストベクトル生成時よりも多くなってしまうことがある。テストベクトル変更手法は、消費電力が低下するように生成

されたテストベクトルの順序を変更する手法 [37]と、X 埋込み手法 [38-42] がある。X 埋込み手法では、テストベクトル中の故障検出とは無関係なビット (X ビット) に適切な論理値を埋め込むことによって、キャプチャ前後の信号値変化数を削減する。テストベクトル変更手法は元のテスト集合の故障検出率を保障し、テストベクトル数の増加もないが、1 つのテストベクトルで検出する故障が多い場合、十分な削減効果が得られない可能性がある。

これらの手法のうち、X 埋込み手法は既存 ATPG フローへの組込みが容易で、回路面積やテストデータ量への影響もないため、現在広く用いられている。次節ではこれまでに提案された X 埋込み技術について述べる。

### 3.3 低消費電力 X 埋込み技術

テストベクトルの中には、故障検出とは無関係な入力ビットが多数存在することがわかっている [48]。そのようなビットはドントケア (X) とみなすことができ、任意の論理値を割り当てることができる。X を含むテストベクトルをテストキューブ (Test Cube) といい、テストキューブを得る方法としては、テストパターン生成段階で X を残す方法と、通常のテストパターン生成で得られた初期テスト集合からアルゴリズム的に X を特定する方法が挙げられる [48]。

X は消費電力の削減に用いることができる。X 埋込みによるキャプチャ時低消費電力化技術では、キャプチャ動作前のスキャン FF の論理値とキャプチャ動作後のスキャン FF の論理値がなるべく変化しないように、テストキューブの X に論理値を埋め込む。すなわち、キャプチャ時の PPI と PPO の論理値の変化数を少なくすることによって回路の信号変化数の削減を実現する。X に埋め込む論理値の最適な組合せは、考えられる全ての組合せに対して論理シミュレーションを行い、信号値変化数を計算すれば求めることができる。テストキューブ中の X の数を  $n$  とすると、 $2^n$  回の論理シミュレーションと信号値変化数の計算が必要になる。しかしながら、テストベクトルから特定される X の割合は非常に高く [48]、X の数が増加するとシミュレーション回数は爆発的に増大する為、全数探索は非現実的である。そのため、アルゴリズムによって効率的に信号値変化数の少ない論理値の組合せを求める必要がある。

テストキューブに対して論理シミュレーションを行ったときの PPI と PPO のビット対に現れる値の組合せは、表 3-1 のように A～D の四通りが考えられる。タイプ A は PPI と PPO の両方が X ではないため、X 埋込みに用いることができない。タイプ B の場合は、PPI の X に PPO の論理値と同じ論理値を埋め込めば、容易に信号値変化を抑えることができる。タイプ C とタイプ D の取り扱いについては、様々な手法が提案されており、正当化操作に基づく手法 [38]、確率計算に基づく手法 [39, 40]、これらを組み合わせた手法 [41] が提案されている。

表 3-1 PPI と PPO に現れる値

| タイプ | PPI      | PPO      |
|-----|----------|----------|
| A   | 論理値(0/1) | 論理値(0/1) |
| B   | X        | 論理値(0/1) |
| C   | 論理値(0/1) | X        |
| D   | X        | X        |

### 3.3.1 正当化操作に基づく X 埋込み手法

正当化操作 (Justification) とは、ATPG アルゴリズム PODEM [3] の基本操作のひとつで、値が決定している出力信号線の値と矛盾しないように入力信号線に論理値を割り当てる操作である。LCP (Low-Capture-Power) -Fill 法 [38] はまず、全てのタイプ B ビット対の PPI に対し PPO と同じ論理値を埋め込む。更新されたテストキューブに対して論理シミュレーションを行う。これにより、タイプ C のビット対がタイプ A に、タイプ D のビット対がタイプ B になることがあるため、タイプ B ビット対がなくなるまで埋め込みと論理シミュレーションを繰り返す。次に、タイプ C ビット対に対し、PPO の X に PPI と同じ論理値の正当化を試みる。正当化が成功すると、他の PPI の X の論理値が決定するので、更新されたテストキューブに対して論理シミュレーションを行う。論理シミュレーションによりタイプ D ビット対がタイプ C ビット対になることがあるため、タイプ C ビット対がなくなるまで正当化操作を繰り返す。タイプ D ビット対に対しては、PPI に対する論理値の埋め込みと、PPO に対する正当化操作を選択的に行う。

図 3-1 に LCP-Fill 法の例を示す. 正当化操作に基づく X 埋込み手法は信号値遷移数の削減効果は高いが, 論理シミュレーションと正当化操作を何度も繰り返し行うため, 実行時間が長い.

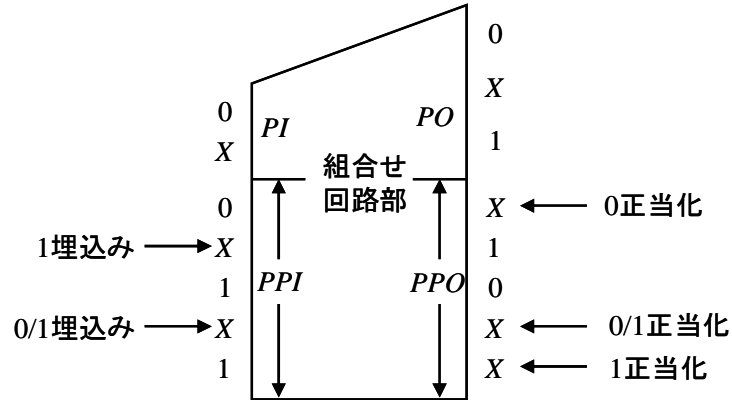


図 3-1 LCP-Fill 法の例

### 3.3.2 確率計算に基づく X 埋込み手法

確率計算に基づく X 埋込み手法は, 回路の PI と PPI に論理値をとる確率を設定してから, ゲートの種類によって定められた計算法[55]に基づいて確率を出力方向に伝搬し, PPO の論理値の確率によって PPI に埋め込む論理値を決定する処理を基本とする. Preferred-Fill 法 [39]では, 全ての PI と PPI に対し, 0 をとる確率 (0-確率) と 1 をとる確率 (1-確率) をそれぞれ 50% に設定し, 確率計算を行う. この処理によって PPO の 0-確率と 1-確率を保持しておき, テストキューブ中の X に対応する PPO の 0-確率が 1-確率よりも高いとき 0 を埋め込み, 1-確率が 0-確率よりも高いとき 1 を埋め込む. 論理値確率が等しいときは, ランダムに論理値を埋め込む. この処理を全てのテストキューブに対して行う. 図 3-2 に Preferred-Fill 法の例を示す.

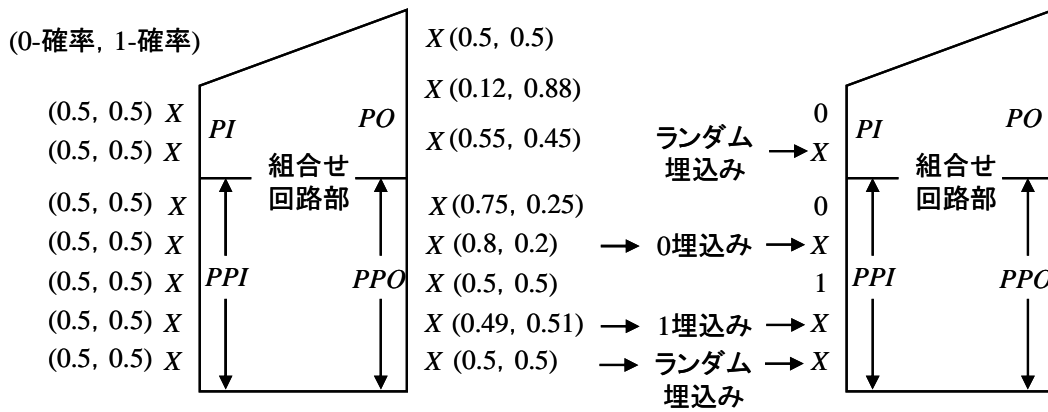


図 3-2 Preferred-Fill 法の例

PWT (probabilistic weighted transition) -Fill 法 [40]では、テストキューブ中の X の 0-確率と 1-確率をそれぞれ 50%，論理値 0 と 1 の入力はそれぞれ 0-確率と 1-確率を 100%とし、論理値確率の計算を行う。その後、優先順位の高い X ビットを一つ選び、PWT 値を計算し、得られた PWT 値に基づいて X に埋め込む論理値を決定する。PWT 値は回路内の全ノードの確率を元に算出される。PWT-Fill 法では PPI の X に論理値が埋め込まれるたびに確率の再計算を実施し、論理値埋め込みの精度向上を図っている。

Preferred-Fill 法は一度の確率計算によって全てのテストキューブの X に対する埋込み論理値を決定するため、非常に高速である。しかし、テストキューブ中の論理値が決定している入力で確率の相違が生じる上、確率計算はその特性上、分岐再収斂のある回路では計算された確率と実際の確率が異なることがある。そのため、効果的な信号遷移削減効果が得られないことがある。PWT-Fill 法はテストキューブ毎に確率を計算し、PPI の X に埋め込む論理値が決定するたびに確率を再計算するため、確率計算の精度は比較的高い。そのため信号遷移削減効果は高いが、テスト集合に含まれる X の割合が多いと処理に膨大な時間を要することがある。

### 3.3.3 ハイブリッド X 埋込み手法

ハイブリッド X 埋込み手法は、正当化操作と確率計算の両方を組み合わせた手法で、JP (Justification-Probability) -Fill 法として知られている [41]。JP-Fill

法ではまず、タイプ B とタイプ C のビット対に対して、正当化操作に基づく X 埋込み手法と同様の処理を行う。その後、タイプ D ビット対に対しては確率計算によって PPI の X に埋め込む論理値を決定する。このとき、一度の確率計算で全ての PPI の論理値を決定せずに、PPO の 0-確率と 1-確率の差が大きい箇所のみに対して PPI の論理値を決定し、確率の再計算を行うことによって、論理値埋め込みの精度を向上している。図 3-3 に JP-Fill 法の例を示す。

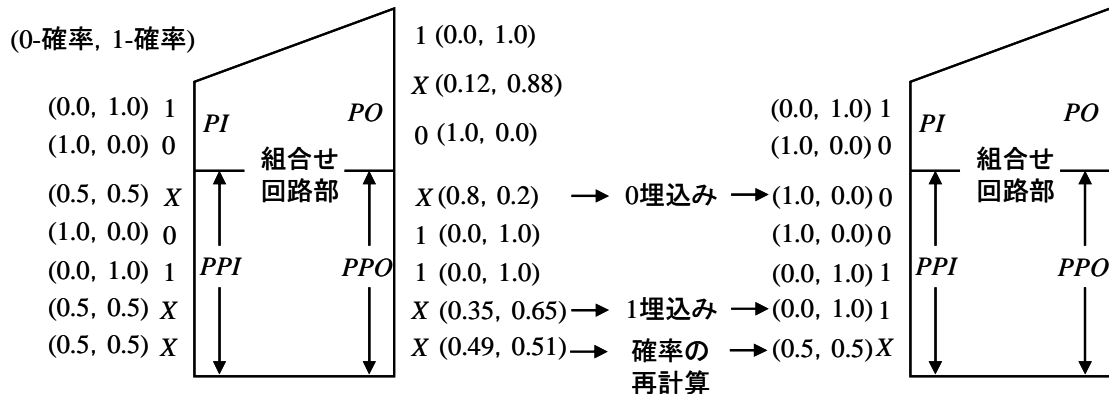


図 3-3 JP-Fill 法の例

JP-Fill 法は正当化操作と確率の再計算を組み合わせることで、短い実行時間と高い信号遷移削減効果を実現している。

### 3.4 X 埋込み手法の課題

従来の X 埋込み手法は、全ての PPI の X の論理値を一度に埋め込むものと、X の論理値埋込みを 1 ビットずつ行うものがあるが、前者は削減効果が低く、後者は実行時間が長い。また、従来の X 埋込み手法は基本的に、キャプチャ前後のスキャン FF、あるいは回路全体の信号値遷移数の削減を目的としている。信号値の遷移は電源電圧降下による遅延増大と密接なつながりがあるが、信号値遷移数の大小と誤テストは必ずしも直接相関があるわけではない。テストベクトルを印加すると、多数のパスが活性化するが、出力に遷移が到達するタイミングはパスが長いほど遅くなるため、長いパスほど電圧低下による遅延増加の影響を受けやすくなる。長いパスの近傍に位置する信号線の論理値の遷移が多いと、回路全体に占める信号値変化の割合が低くてもタイミング違反を引き



起こす可能性がある。したがって、スキャン **FF** あるいは回路全体の信号値遷移数を削減する従来手法では十分に誤テストを回避できない可能性がある。そのため、局所的な信号遷移削減が可能な **X** 埋込み手法が求められている。



# 第4章 遺伝的アルゴリズムによる低消費電力 X 埋込み手法の提案

## 4.1 概説

X 埋め込み問題の解空間は非常に大きく、PPI と PPO に現れる論理値の組合せを基に埋め込む論理値を決定する従来手法では得られない良解が存在する可能性がある。そこで本研究では入力 of X のみに着目し、探索アルゴリズムによるアプローチによって最適な埋め込み論理値を求める方法をとる。信号遷移削減効果の高い埋め込み論理値の組合せをより早く求めるため、多くの探索アルゴリズムの中でも、比較的短時間で優れた解を求めることができる遺伝的アルゴリズム (GA : Genetic Algorithm) による X 埋込み手法を提案する。提案手法によって X 埋込みの高速化と信号遷移削減降下の向上を図る。また、提案手法は適応度評価関数の変更によって、信号遷移を削減したい箇所に狙いを定めた X 埋込みを可能にする。信号遷移を削減したい領域を適応度評価関数に導入し、局所的な信号遷移の削減を目指す。

## 4.2 遺伝的アルゴリズムの基本

GA は、生物の進化の過程に着目して考案されたアルゴリズム [52]であり、最適化手法の一つである。GA の基本的なフローを図 4-1 に示す。

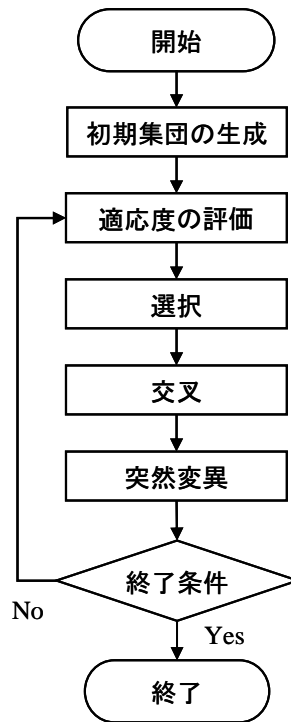


図 4-1 GA の基本的なフロー

GA ではまず解の候補を、遺伝子 (Gene) で表現した個体 (Individual) を複数用意し、集団 (Population) を形成する。続いて、各個体の適応度 (Fitness) が適応度関数 (Fitness Function) によって求められ、適応度の高い個体を優先的に選択して交叉・突然変異などの遺伝的操作によって集団を更新しながら解を探索する。

以下ではそれぞれの遺伝的操作について述べる。

#### 4.2.1 選択

選択 (Selection) では、次世代の個体を生成するため、親となる個体を選択する。適応度の高い個体がより選ばれやすくなるような選択方法をとるのが基本である。代表的な選択アルゴリズムには以下のようなものがある。

- ルーレット選択

ルーレット選択は個体の適応度に比例した確率で選択されるモデルで、適応度が  $f_i$  である個体  $i$  が選択される確率  $p_i$  は次式で表される。

$$p_i = \frac{f_i}{\sum_{k=1}^N f_k}$$

ルーレット選択は適応度に応じて選択確率が決定するため、個体間の適応度の格差が大きいと、適応度の高い個体を選ばれる確率が高くなる。これにより、同じ個体が複数回の交配に参加することになるので、その遺伝子が集団中に爆発的に増えてしまい、探索が早い段階で収束してしまう。

- **ランキング選択**

ランキング選択では各個体を適応度によってランク付けし、ランクに対して定められた確率で個体を選択する。この方法は、ルーレット選択のように適応度の格差が選択確率に直接反映されないため、初期収束は起こりにくい。しかし、逆に適応度の格差が小さい場合でも選択確率に大差が生じる可能性も示唆している。また、世代が変わるたびにランク付けのためのソートを行う必要がある。

- **トーナメント選択**

トーナメント選択は、集団から決められた数（トーナメントサイズ）の個体をランダムに抽出し、その中で最も適応度の高い個体を選択する。トーナメントサイズは2であることが多いが、選択圧を高めるためにより大きなトーナメントサイズを設定することもある。

- **エリート選択**

集団の中で最も適応度の高い個体をそのまま次世代に残す方法をエリート選択といい、上記の選択と組み合わせて用いられることがある。エリート選択によって、その時点で最もよい解が破壊されなくなるが、エリートの遺伝子が集団中に急速に広がる可能性が高くなるため、局所解に陥る危険もある。

## 4.2.2 交叉

交叉（Crossover）では、選択された個体対の遺伝子を組み換えて子の個体を生成する操作である。代表的な交叉アルゴリズムには以下のようなものがある。

- 一点交叉

一点交叉は最も単純な交叉方法で、交叉する位置をランダムに一箇所選び、その点より後ろの遺伝子を入れ換える。

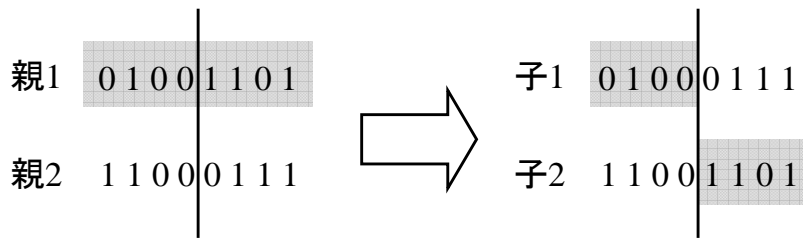


図 4-2 一点交叉の例

図 4-2 のように、4 番目と 5 番目の遺伝子の間に交叉点が設けられると、子 1 は 4 番目までの遺伝子を親 1 から、5 番目以降の遺伝子を親 2 から受け継ぎ、子 2 はその逆となる。

- 複数点交叉

交叉点を複数設ける方法を複数点交叉といい、二点交叉が一般的に広く用いられている。図 4-3 に二点交叉の例を示す。

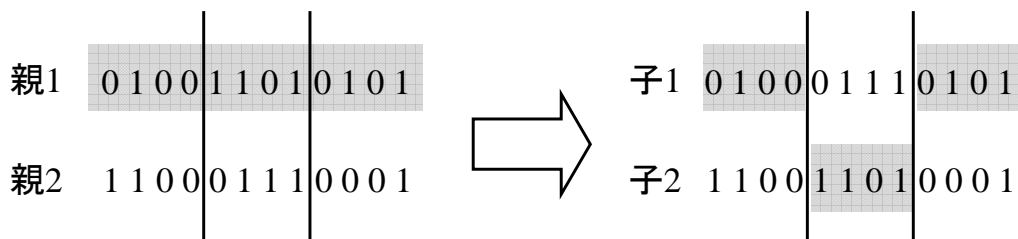


図 4-3 二点交叉の例

子 1 は一つ目の交叉点までの遺伝子と二つ目の交叉点以降の遺伝子を親 1 から、二つの交叉点の間の遺伝子を親 2 から受け継ぎ、子 2 はその逆となる。

- 一様交叉

一様交叉では、交叉時にマスクを用いることにより、どちらの親の遺伝子を受け継ぐかを決定する。マスクの値は任意に決定することができる。

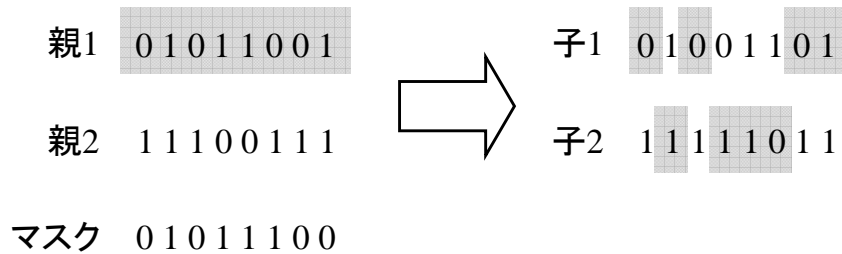


図 4-4 一様交叉の例

図 4-4 のように、マスクのビットが 0 のとき、子 1 は親 1 の遺伝子を受け継ぎ、1 のとき、親 2 の遺伝子を受け継ぐ。子 2 はその逆となる。

### 4.2.3 突然変異

突然変異は、解の多様性を保つために個体の遺伝子の一部を一定の確率で変化させる操作である。一般的に 0.1%~1%程度の確率が設定される。突然変異率を低くしすぎると局所解に陥りやすくなり、高くしすぎると解の特性を失ってしまい、ランダムサーチと化してしまう恐れがある。

## 4.3 遺伝的アルゴリズムによる低消費電力 X 埋込み手法

本研究で提案する、遺伝的アルゴリズムを適用した新しい低消費電力 X 埋込み手法を GA-Fill 法とよぶ。図 4-5 に、GA-Fill 法のフローを示す。

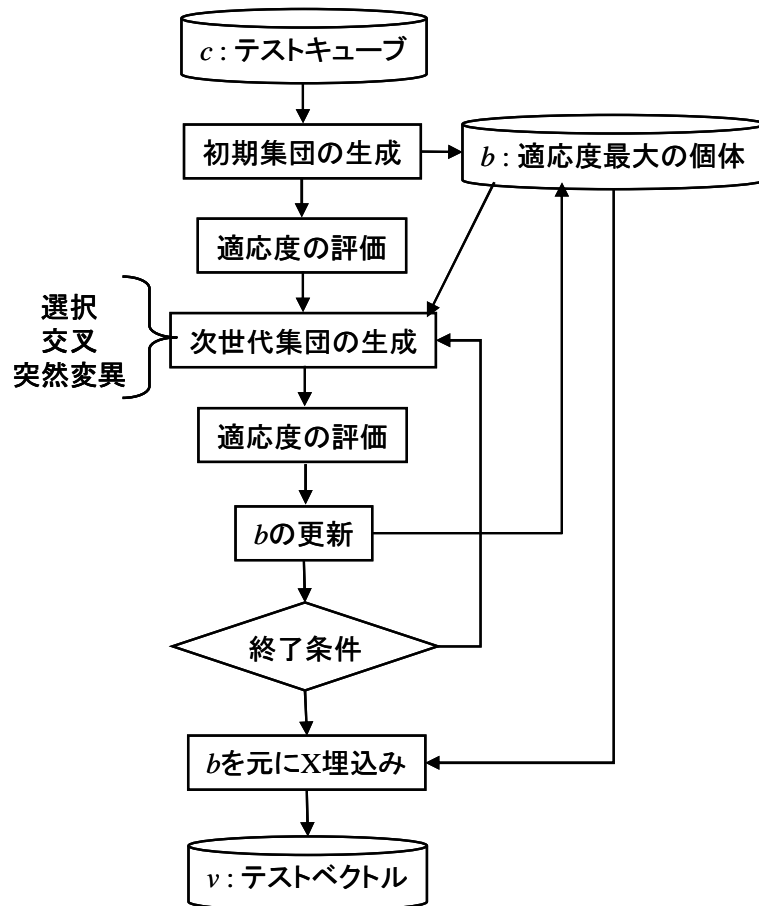


図 4-5 GA-Fill 法のフロー

GA-Fill 法ではテストキューブ中の X ビットを遺伝子とみなし、信号値遷移の削減を目的とした適応度評価関数にしたがって、解を探索する。また、世代交代時の個体の適応度悪化による最良解の見落としを防ぐため、現世代までの最良解  $b$  を常に保持しておく。終了条件に達したとき、最良の個体  $b$  を基にテストキューブ中の X に論理値を埋め込み、テストベクトルを得る。GA-Fill 法は適応度評価関数を変えることによって、信号値遷移削減の対象を自由に変えることができる。基本的には対象とする信号線の重み付き信号遷移数 (WSA : Weighted Switching Activity) の削減を目的とするように適応度評価関数を設定する。WSA の重みは、ノードに複数のファンアウトがあるときはファンアウト数+1, そうでないときは 1 となる。大域的な削減を目的とする場合、その対象は従来の X 埋込み手法と同様に FF あるいは全ノードとなる。GA-Fill 法では効果と速度の両立を図るため、テストキューブ中の X の割合に応じて適応度評価関数を切り替える。個体  $i$  の信号遷移の大域的削減を目的とする場合の評価関



数  $fitness(i)$  は、テストキューブ中の  $X$  の割合が 60% を越えるとき、

$$fitness(i) = 1 - \frac{WSA\_FF(i)}{WSA\_FF\_All} \dots\dots\dots(i)$$

そうでないとき、

$$fitness(i) = 1 - \frac{WSA\_Node(i)}{WSA\_Node\_All} \dots\dots\dots(ii)$$

を用いる。ここで、 $WSA\_FF(i)$  と  $WSA\_FF\_All$  はそれぞれ、個体  $i$  を  $X$  に割り当てたテストベクトルの FF での WSA と、全ての FF が遷移したと仮定したときの WSA を表し、 $WSA\_Node(i)$  と  $WSA\_Node\_All$  はそれぞれ、個体  $i$  を  $X$  に割り当てたテストベクトルのノードでの WSA と、全てのノードが遷移したと仮定したときの WSA を表している。式(i)は FF の WSA のみを計算すればよいので、全ノードの WSA を計算する式(ii)よりも計算時間が短い。遷移数の削減効果は式(ii)を用いたほうが高い。そのため、GA-Fill 法ではテストキューブ中の  $X$  が多いときは遷移数の削減が容易であると考え、式(i)を用いて FF の遷移数削減に集中する。 $X$  が少ないときはより高い削減効果を得るため、式(ii)を用いて、可能な限り全ノードの WSA を削減する。

レイアウト情報や電源情報などによって、回路内の電圧降下の影響を受けやすい領域（クリティカルエリア）が分かっているとき、GA-Fill 法は適応度評価関数を変更しクリティカルエリアに属するノードの信号遷移の削減に集中することができる。たとえば、活性化しているクリティカルパスがあると、その近傍のノードでの信号遷移はクリティカルパスのタイミングに影響する恐れがある。このような場合、クリティカルパス上のノードとその近傍のノードをクリティカルエリアとして抽出し、優先的に信号遷移を抑制する必要がある。GA-Fill 法では局所的な遷移数削減のため、次式のように適応度評価関数を変更する。

$$fitness(i) = 1 - \frac{WSA\_CS\_Node(i)}{WSA\_CS\_Node\_All} \dots\dots\dots(iii)$$

ここで、 $WSA\_CS\_Node(i)$ と  $WSA\_CS\_Node\_i$  はそれぞれ、個体  $i$  をテストキューブ中の  $X$  に割り当てたテストベクトルのクリティカルエリア内のノードでの  $WSA$  と、全てのクリティカルエリア内のノードが遷移したと仮定したときの  $WSA$  を表している。

## 4.4 評価実験

### 4.4.1 実験の準備

GA-Fill 法を Intel Xeon 3.0GHz, メモリ 24GB の計算機上に C 言語を用いて実装し、実験を行った。ISCAS'89 フルスキャンベンチマーク回路に対する、大域的な信号遷移削減効果と局所的な信号遷移削減効果を従来手法と比較した。内製 ATPG ツールによって生成された遷移遅延故障用テストパターンに対し、XID[48]によって  $X$  を抽出したテストキューブを使用した。GA における集団サイズは 64, 選択方法はルーレット選択, 交叉方法は二点交叉とし, 交叉率と突然変異率はそれぞれ 100%, 0.1%とした。世代数が 100 に到達, あるいは最良解の適応度が 0.85 を超過したとき GA を終了するように終了条件を定めた。

### 4.4.2 大域的な信号遷移削減効果の評価

GA-Fill 法の大域的な信号遷移削減効果の評価するため, 式(i)と式(ii)を切り替える適応度評価関数を用いた GA-Fill 法を実装し, 従来手法の LCP-Fill 法[38]と Preferred-Fill 法[39]の  $WSA$  削減率を比較した。表 4-1 に実験結果を示す。表 4-1 の Test Set はテスト集合の情報であり, Test Vec. #, Fault Cov.(%),  $X$  (%) はそれぞれ, テストベクトル数, 故障検出率, テストキューブ中の  $X$  の割合の平均を表している。Reduction Ratio (%)は  $WSA$  の削減率であり, Max.  $WSA$ -FF, Max.  $WSA$ -Node はそれぞれ, FF, ノードの最大  $WSA$  削減率を表している。CPU(s)は実行時間 (秒) である。

表 4-1 大域的な信号遷移削減効果の比較

| Circuit | Test Set    |                |       | Reducion Ratio (%) |       |             |               |       |             | CPU (s) |       |            |
|---------|-------------|----------------|-------|--------------------|-------|-------------|---------------|-------|-------------|---------|-------|------------|
|         | Test Vec. # | Fault Cov. (%) | X (%) | Max. WSA-FF        |       |             | Max. WSA-Node |       |             |         |       |            |
|         |             |                |       | LCP                | pref. | GA          | LCP           | pref. | GA          | LCP     | pref. | GA         |
| s13207  | 323         | 79.5           | 67.4  | 23.5               | 19.2  | <b>23.2</b> | 11.7          | 7.9   | <b>13.2</b> | 40      | 9     | <b>28</b>  |
| s15850  | 221         | 70.2           | 54.3  | 35.2               | 31.7  | <b>36.7</b> | 26.2          | 23.3  | <b>27.6</b> | 33      | 8     | <b>29</b>  |
| s35932  | 337         | 82.5           | 92.0  | 44.5               | 29.3  | <b>41.4</b> | 7.6           | 10.0  | <b>10.4</b> | 230     | 26    | <b>152</b> |
| s38417  | 270         | 98.0           | 47.1  | 36.2               | 23.0  | <b>35.9</b> | 14.2          | 14.2  | <b>20.1</b> | 654     | 27    | <b>296</b> |
| s38584  | 412         | 83.9           | 73.9  | 17.8               | 11.2  | <b>17.7</b> | 14.9          | 8.7   | <b>15.0</b> | 2086    | 42    | <b>194</b> |
| Ave.    | —           | —              | 54.7  | 31.4               | 22.9  | <b>31.0</b> | 14.9          | 12.8  | <b>17.3</b> | —       | —     | —          |

表 4-1 より、GA-Fill は最大 WSA 削減率の点では、FF の WSA 削減効果は LCP-Fill 法とほぼ同等で、ノードの WSA 削減効果において従来手法を上回っていることが分かる。これは、従来手法がどちらも FF の信号遷移削減を目的としているためである。このことから、回路内部の信号遷移を削減するために FF の WSA を削減するだけでは不十分であることが分かる。また、CPU 時間の点では、一度の確率計算によって全ての X の埋込み論理値を決定する Preferred-Fill 法には及ばないものの、LCP-Fill 法を大きく上回っていることが分かる。

図 4-6 は横軸に実行時間 (秒)、縦軸にノードの最大 WSA 削減率をとり、ISCAS'89 ベンチマーク回路の中で最大規模の回路である s38584 に対する各手法の実験結果を位置付けたグラフである。グラフは、GA-Fill 法は高い最大 WSA 削減率を高速に得られていることを示している。

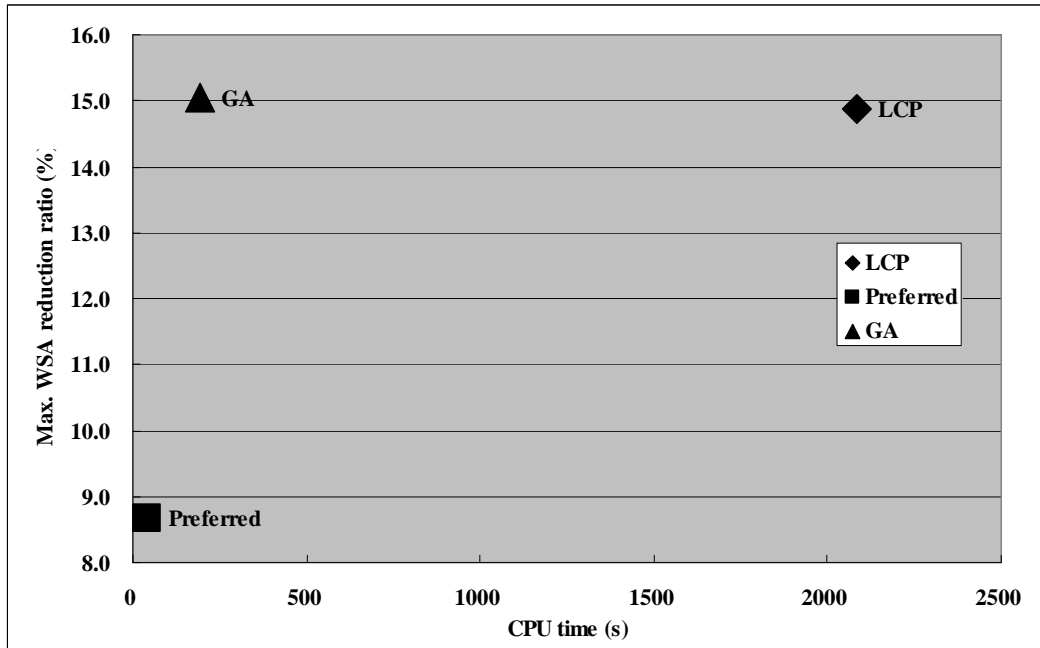


図 4-6 s38584 での実行時間と最大遷移数削減率にみる各手法の位置付け

#### 4.4.3 局所的な信号遷移削減効果の評価

GA-Fill 法の局所的な信号遷移削減効果の評価するため、式(iii)の適応度評価関数を用いた GA-Fill 法 (GA-2) を実装し、局所的な信号遷移削減効果を、4.4.2 節の実験で用いた大域的な信号遷移削減を目的とした GA-Fill 法 (GA-1) と比較した。なお、本実験ではクリティカルノードとして回路の全ノード数の 10% の信号線をランダムに選択し、クリティカルエリアを構成した。また、GA-2 の GA におけるパラメータは GA-1 と同様である。

表 4-2 に実験結果を示す。

表 4-2 局所的な信号遷移削減効果の比較

| Circuits | # Critical Nodes | MAX. WSA Reduction Ratio (%) |      |                |      | CPU (s) |      |
|----------|------------------|------------------------------|------|----------------|------|---------|------|
|          |                  | All Nodes                    |      | Critical Nodes |      | GA-1    | GA-2 |
|          |                  | GA-1                         | GA-2 | GA-1           | GA-2 |         |      |
| s13207   | 1401             | 13.1                         | 11.7 | 11.5           | 12.7 | 30      | 20   |
| s15850   | 1655             | 29.2                         | 28.2 | 31.0           | 31.6 | 31      | 22   |
| s35932   | 3771             | 10.4                         | 10.3 | 10.4           | 10.5 | 149     | 99   |
| s38417   | 4012             | 20.6                         | 20.6 | 18.0           | 18.0 | 312     | 98   |
| s38584   | 4018             | 15.0                         | 15.0 | 16.9           | 16.9 | 205     | 132  |
| Ave.     | —                | 17.7                         | 17.2 | 17.6           | 18.0 | —       | —    |

表 4-2 の, # Critical Nodes はクリティカルノード数を表す. MAX. WSA Reduction Ratio (%) は最大 WSA 削減率であり, All Nodes と Critical Nodes はそれぞれ全ノードの最大 WSA 削減率, クリティカルノードの最大 WSA 削減率を表す. CPU (s) は実行時間 (秒) を表す.

表 4-2 より, 最大遷移数削減率は全ノード, クリティカルノードともに大きな差は見られなかった. その原因としては, 回路規模が小さいこと, レイアウト情報を用いず, クリティカルノードをランダムに抽出していることなどが考えられる. 実行時間においては, クリティカルノードのみを考慮する GA-2 の方が高速であった. 大域的な信号遷移削減を目的とする GA-Fill においても, 回路全体ではなく一部のノードを抽出して適応度評価の対象とすれば, 削減率に影響を与えずに高速化できる可能性がある.

## 4.5 まとめ

本章では GA による新しい低消費電力 X 埋込み手法を提案した. GA の最大の利点は対象とする問題が変わっても柔軟に対応できることである. そのため, 従来の X 埋込み手法では不可能であった局所的な信号遷移の削減にも対応できる可能性がある. 実験では, GA-Fill 法は従来最も高速であった Preferred-Fill 法に近い実行時間で, WSA 削減率の高い LCP-Fill 法を上回る削減率を得ることができることを確認した. また, ランダムなノードの WSA の削減に関しては高い削減効果は得られなかったが, これは全ノードの WSA 削減の高速化につながる可能性がある.

今回の実験ではランダムにノードを選択して WSA 削減率を評価したが, 提案手法の有効性を確かめるには, レイアウト情報や電源情報を用いて実際のクリティカルノードを抽出し, WSA 削減によって得られた消費電力の削減効果を, 消費電力解析ツールなどを用いて検証する必要がある. また, 実験に用いた回路はベンチマーク回路であり, 実用されている回路に比べると回路規模は非常に小さいため, より大規模な回路で実験を行うことも必要になる. そして, 提案手法で用いられている GA は選択, 交叉, 突然変異率など様々なパラメータを持っており, 用いるパラメータによって探索結果も大きく変わる可能性があ

るため、提案手法のパラメータを色々な組合せに変更して、最適なパラメータを見つけ出すことが今後の課題として挙げられる。

## 第5章 X 故障診断技術

### 5.1 概説

X 故障診断手法 [13]は、複雑な欠陥であっても正確に表現できる X 故障モデルと、単一故障の仮定で多重故障を診断可能な Per-Test 故障診断方式を組み合わせた故障診断手法である。また、故障候補の可観測性や照合時の一致不一致情報を診断値として数値化し、回路に存在する可能性の高い順に故障候補を順位付けすることができる。X 故障診断手法は、多重・動的な欠陥に対しても精度よく診断可能であることが示されている。

本章では提案手法の基礎となる X 故障診断手法について述べる。

### 5.2 Per-Test 故障診断

多重・動的な欠陥に対しても正確に故障候補が得られる故障診断手法として、Per-Test 故障診断手法が注目されている [16]。Per-Test 故障診断手法の最大の特徴は、テストベクトルごとの診断結果を独立に取り扱う点である。従来の故障診断手法は、あるテストベクトルに対する故障診断で得られる故障候補のみを、次のテストベクトルに対する故障診断で用いる。しかし、多重故障により故障の影響が出力まで伝搬しない場合、従来手法は正しい故障箇所を見逃してしまう恐れがある。Per-Test 故障診断手法は、全てのテストベクトルに対して全ての故障候補を診断の対象とするため、間欠的な故障や複数の信号線に影響するような故障に対しても有効であるが、従来手法よりも時間的コストが大きいという欠点がある。また、Per-Test 手法では各診断結果の組合せにより最終的な故障候補を得るので、得られる故障候補の数が多くなってしまう可能性がある。図 5-1 に Per-Test 故障診断手法の概要を示す。

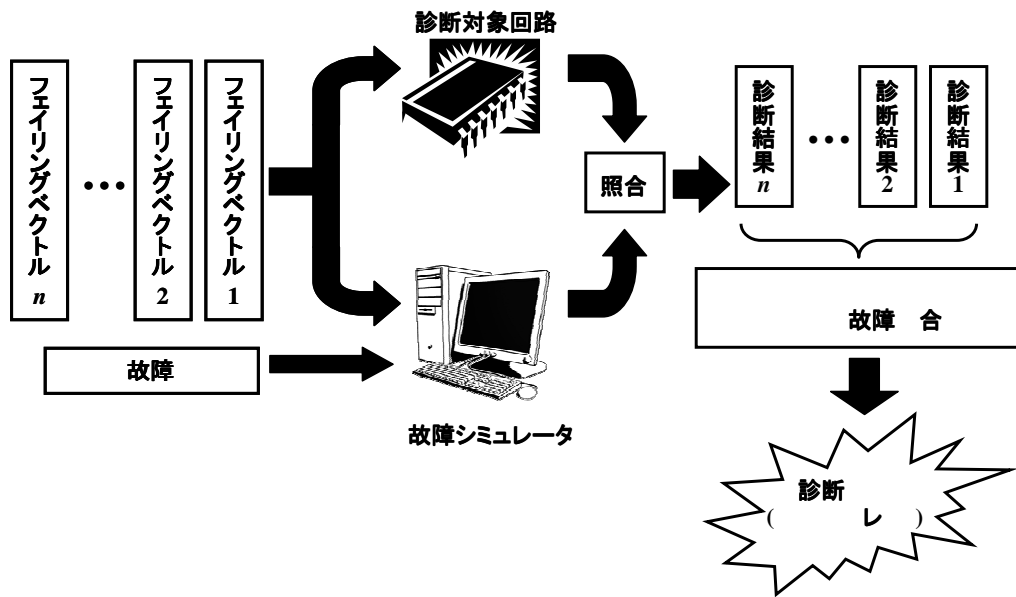


図 5-1 Per-Test 故障診断手法の概要

Per-Test 故障診断手法の、各テストベクトルに対する診断結果を独立に扱うという考え方は、以下に示す二つの仮定に基づいている。

**仮定 1:** 回路に複数の故障が発生しているとしても、そのうちただ一つの故障の影響のみを伝搬するようなテストベクトルが存在する。

**仮定 2:** 仮定 1 を満たすようなテストベクトルによる単一故障シミュレーションの応答が実応答と一致する場合、その故障は実回路に存在する可能性が高い。

仮定 1 で示したようなテストベクトルを SLAT ベクトル(Single Location At a Time)という [16]。SLAT ベクトルの例を図 5-2 に示す。Per-Test 故障診断手法では、SLAT ベクトルのみを用いる。

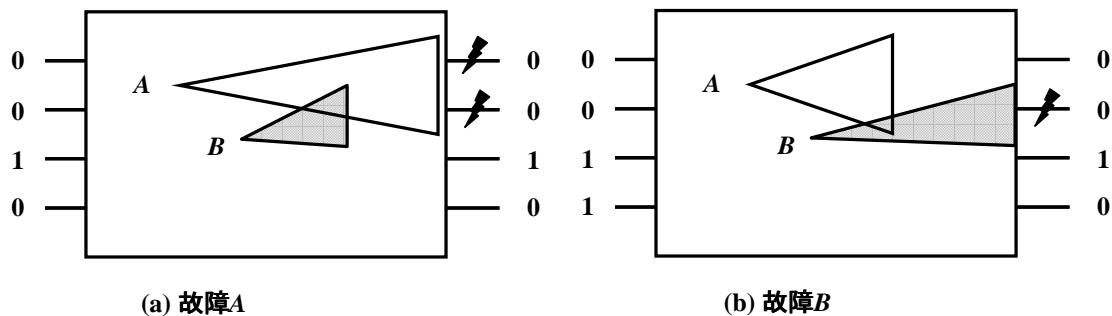


図 5-2 SLAT ベクトル



図 2-14(a)の実回路と同じ故障が発生しているとする、図 5-2(a)のテストベクトルは、故障 A の影響のみを伝搬させ、図 5-2(a)のテストベクトルは、故障 B の影響のみを伝搬させるため、これらのテストベクトルは SLAT ベクトルであるといえる。しかし、図 2-14 のようなテストベクトルは故障 A と B 両方の影響を伝搬させるため、SLAT ベクトルではない。Per-Test 故障診断手法では、このようなテストベクトルは用いない。以下では、Per-Test 故障診断手法の具体的な手順について説明する。

Per-Test 故障診断手法は大きく 2 つの処理で構成される。

**処理 1 (情報収集):** フェイルしたベクトルに対する故障シミュレーションを行って SLAT ベクトルを抽出し、一致あるいは不一致の情報を故障診断表に格納する。

**処理 2 (情報処理):** 処理 1 によって生成した故障診断表から故障箇所候補の組合せを診断結果として抽出する。

処理 1 では、フェイリングベクトルの誤りを説明できる故障を求めるため、フェイリングベクトルに対する故障シミュレーションを行う。一つでも単一故障の故障シミュレーション結果と一致すれば、その時のフェイリングベクトルは SLAT ベクトルと判定され、各単一故障の一致・不一致の情報をまとめる。この一致・不一致の情報をまとめたものを故障診断表と呼ぶ。表 5-1 に故障診断表の例を示す。この故障診断表では、各 SLAT ベクトルにおける各故障の一致・不一致情報を 0 と 1 で表しており、1 のとき一致、0 のとき不一致を意味している。

表 5-1 故障診断表

|     | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ |
|-----|-------|-------|-------|-------|-------|
| sv1 | 1     | 1     | 0     | 0     | 0     |
| sv2 | 0     | 0     | 1     | 1     | 0     |
| sv3 | 1     | 0     | 1     | 0     | 0     |
| sv4 | 0     | 0     | 0     | 0     | 1     |

処理 2 では、故障診断表から SLAT ベクトルをすべて説明できる最小サイズの故障集合を探索することで診断結果を抽出する。つまり、実回路の観測応答に現れた誤りの説明がつく単一故障ではなく、故障の集合を求めることで多重故障の診断を行うことが出来ることになる。そのような最小サイズの故障集合をマルチプレット(Multiplet) [16]と呼ぶ。

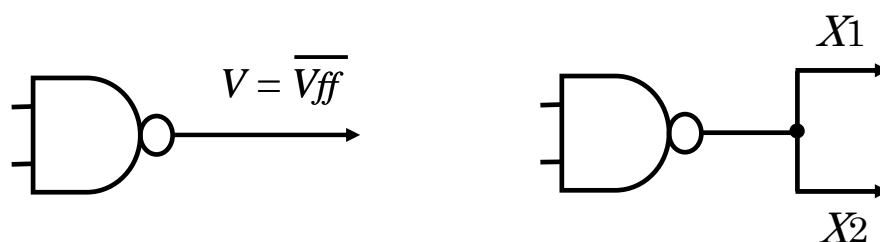
表 5-1 の故障診断表からマルチプレットを求めると、 $\{f_1, f_3, f_5\}$ ,  $\{f_1, f_4, f_5\}$ ,  $\{f_2, f_3, f_5\}$  の三つのマルチプレットが求められる。この場合、この三つのマルチプレットが診断の結果として出力される。つまり実回路には三箇所故障があり、その故障箇所は  $\{f_1, f_3, f_5\}$ ,  $\{f_1, f_4, f_5\}$ ,  $\{f_2, f_3, f_5\}$  のうちどれかであると考えられる。

### 5.3 X 故障モデル

X 故障モデルは以下のように定義される。

**定義 5-1 :** 各ゲートは、ゲート内部もしくはファンアウト上の任意の欠陥に対応する X 故障を 1 つもつ。ファンアウトのないゲートには、X 故障の故障値として正常回路の論理値を反転した値を割り当てる、ファンアウトのあるゲートには、ファンアウト信号線に起こりうる故障値全てを表すため、X 故障は各ファンアウト信号線に異なる X 記号を割り当てる。

図 5-3 に X 故障モデルの例を示す。



(a) ファンアウトのないゲートの X 故障 (b) ファンアウトのあるゲートの X 故障

図 5-3 X 故障モデル

図 5-3 (a) は、ファンアウトのないゲートの X 故障を表している。ファンアウトがない場合、場合故障値として正常論理値  $V_{ff}$  の反転値  $\overline{V_{ff}}$  を割り当てる。図 5-3 (b) のようにファンアウトのあるゲートの場合、ファンアウト信号線に起こ

りうる全ての故障値を考慮するため、各ファンアウト信号線に異なる X 記号  $X_1$  と  $X_2$  を割り当てている。これにより、出力電圧が動的に変化するような欠陥の振舞いも正確に表現することができる。

X 故障モデルは、以下のような特徴を持っている。

(1) **一般性**：一つの欠陥が原因で多重故障が現れる場合に、単一 X 故障はその欠陥によって起こりうる論理値の組合せをすべて表すことができる。

(2) **サイズ**：回路に対する X 故障の総数は、回路のゲート数に等しく、一般的に用いられる単一縮退故障よりも少ない。

(3) **正確性**：X 記号を用いることで信号電位が変化する故障の特定できない振舞いを扱うことができるので、診断情報に欠損が生じない。

(4) **柔軟性**：X 故障モデルは、1つの故障に対して同じテストベクトルでも異なる振舞いをするのを許容するので、動的欠陥に対応している Per-Test 故障診断方式に適している。

## 5.4 X 故障シミュレーション

### 5.4.1 概要

仮定した X 故障を外部出力まで伝搬させ、出力応答を求めることを **X 故障シミュレーション**という。X 故障  $f$  の、テストベクトル  $v$  に対する X 故障シミュレーションの応答を  $SimRes(f,v)$  とする。X 故障  $f$  によって起こりうる故障値の組み合わせの数を  $k$  ( $k \geq 1$ ) とし、各故障値に対する外部出力の応答を  $\{R_1, R_2, \dots, R_k\}$  とすると、 $SimRes(f,v) = \{R_1, R_2, \dots, R_k\}$  となる。ファンアウトのないゲートに仮定された X 故障のシミュレーションは、単一縮退故障のシミュレーションと同様に扱うことができるが、ファンアウトのあるゲートに仮定された X 故障は記号を用いるため、論理シミュレーションを用いる一般的な故障シミュレーションとは異なった方法で故障シミュレーションを行う。ファンアウトのあるゲートの X 故障シミュレーションは、次の三つのステップで構成される。

- (1) **X 挿入** : ゲートのファンアウト信号線に異なる X 記号を割り当てる.
- (2) **X 伝搬** : X 記号を外部出力まで伝搬させる.
- (3) **X 分解** : 具体的なシミュレーション結果を得るために, 外部出力に伝搬した X 記号が対応する論理値を求める.

以下では, 各ステップの処理の詳細を説明する.

### 5.4.2 X 挿入

X 挿入では, ゲートのファンアウト信号線  $b_1, b_2, \dots, b_p$  に初期 X 記号  $X_1(b_1), X_1(b_2), \dots, X_p(b_p)$  を割り当てる. X 記号は, 文字 X, 異なる X 記号を区別するための添え字, 括弧内のファンアウト信号線名からなる. X 挿入で割り当てた初期 X 記号は, 対応するファンアウト信号線名を一つ持つ.

X 挿入の例を図 5-4 に示す.

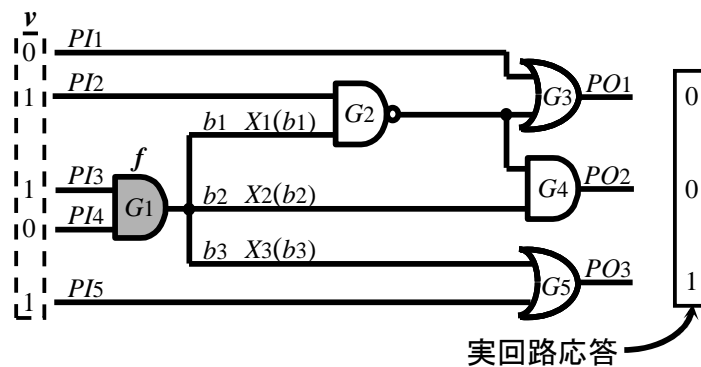


図 5-4 X 挿入

図 5-4 はゲート  $G_1$  の X 故障  $f$  における X 挿入を表している. ゲート  $G_1$  のファンアウト信号線  $b_1, b_2, b_3$  に異なる X 記号  $X_1(b_1), X_1(b_2), X_3(b_3)$  を割り当てる. 各 X 記号は, 対応するファンアウト信号線名を保持している.

### 5.4.3 X 伝搬

X 伝搬では, X 挿入によって割り当てた X 記号を外部出力まで伝搬させる. このとき, 二つの規則を繰り返し適用する.

**規則 5-1 (反転情報の保持) :**  $X_i(B)$  の反転を  $\overline{X_i(B)}$  で表す.  $\overline{X_i(B)}$  の反転は  $X_i(B)$  である. また,  $X_i(B)$  と  $\overline{X_i(B)}$  の論理積(和)を論理値 0 (1) で表す. ここでの  $B$  はファンアウト信号線名の集合である.

規則 1 ではメモリ消費を減らすために反転情報のみを持たせている. 規則 1 の適用例を以下に示す.

$$\begin{aligned} \text{NAND}(\overline{X_1}(b_1, b_2), 1) &= X_1(b_1, b_2) \\ \text{OR}(X_1(b_1), \overline{X_1}(b_1), 0, X_2(b_2)) &= 1 \end{aligned}$$

**規則 5-2 (ファンアウト情報の保持) :**  $X_{i1}(B_1), X_{i2}(B_2), \dots, X_{ih}(B_h)$  に対する論理演算の結果が 0 でも 1 でもない場合, 新しい X 記号  $X_j(B_1, B_2, \dots, B_h)$  で表す. ここでの  $j$  はまだ用いられていない新しい添え字である.

規則 2 により, その信号値が影響を受けるファンアウト情報を保持しているので, X 記号が外部出力で現れた際に, どのファンアウト信号線の影響がそこに伝播しているのかは分かる. 規則 2 の適用例を以下に示す.

$$\begin{aligned} \text{NOR}(X_3(b_1, b_2), X_2(b_2), 0, X_4(b_3)) &= X_5(b_1, b_2, b_3) \\ \text{AND}(\overline{X_1}(b_1), X_3(b_3)) &= X_6(b_1, b_3) \end{aligned}$$

X 伝搬の例を図 5-5 に示す. X 伝搬で得られたシミュレーション結果を初期シミュレーション応答と呼ぶ. 図 5-5 のように, 初期シミュレーション応答は X 記号を含んでいる場合がある.

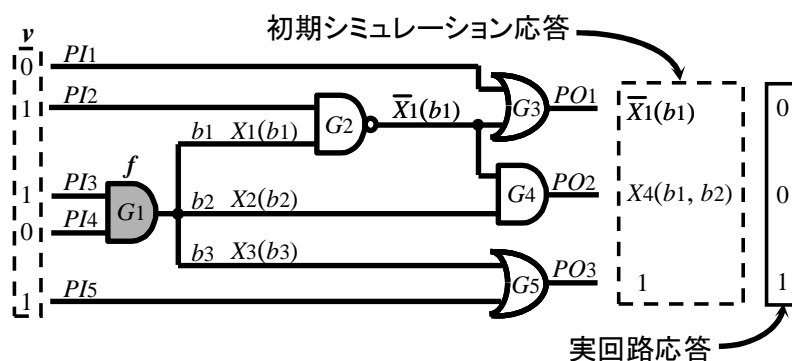


図 5-5 X 伝搬

### 5.4.4 X 分解

X 伝搬により X 記号が外部出力に現れた場合、X 分解を行う。X 分解では、曖昧さを取り除くために初期 X 記号に具体的な論理値を割り当て、二値論理シミュレーションを行う。初期シミュレーション応答に含まれる X 記号が、 $X_{i1}(B1), X_{i2}(B2), \dots, X_{ih}(Bh)$  であるとする。この場合、外部出力まで伝搬するファンアウト信号線の集合は  $B1, B2, \dots, Bh$  であり、 $2^{|B|}$  回の二値シミュレーションを行うことで、ファンアウト信号線の故障値の組合せを全て満たすことができる。

X 分解の例を図 5-6 に示す。図 5-6 は図 5-5 の初期シミュレーション応答に対応している。図 5-5 では  $b1$  と  $b2$  だけが外部出力まで伝搬しているため、故障箇所において 3 通りの故障値の組合せを考慮する必要がある。また、初期シミュレーション応答の中に、図 5-4 で挿入された初期 X 記号がある場合、一意に値を決定することができる。なぜならば、故障診断における故障シミュレーションの目標は実回路の応答と一致するかという情報を得ることであり、図 5-5 における  $PI1$  の初期シミュレーション応答である  $\overline{X1}(b1)=1$  となる場合、明らかに実回路の観測応答と一致することはない。そこで  $\overline{X1}(b1)=0$ 、すなわち、 $X1(b1)=1$  と含意操作を行う。そして、 $X1(b1)=1$  の時の二値シミュレーションを行えばよい。結果として、最終シミュレーション結果は 1 つの論理値の組合せで構成される。一般に、ファンアウトのあるゲートの X 故障  $f$  とテストベクトル  $v$  が与えられた時のシミュレーション結果は、 $R1, R2, \dots, Rk$  ( $k \geq 1$ ) という論理値の組合せの集合となる。つまり、 $SimRes(f, v) = \{R1, R2, \dots, Rk\}$  となる。

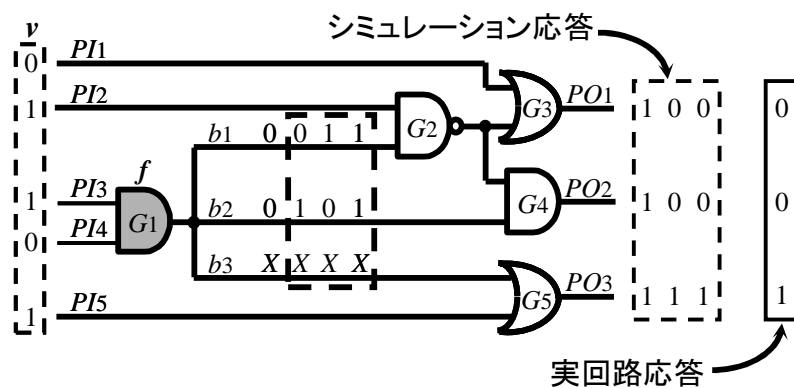


図 5-6 X 分解

## 5.5 照合における情報抽出

### 5.5.1 照合の範囲

X 故障診断手法では、X 故障シミュレーションの結果と実回路応答を照合する際、シミュレートした X 故障が構造的に到達可能な外部出力のみを照合の対象としている。これは、全ての外部出力を照合の対象とすると図 5-7(a)のように故障影響が到達しえない外部出力における不一致による診断情報の欠損を防ぐため、図 5-7(b)のように照合の範囲を限定することで一致という結果が得られる。

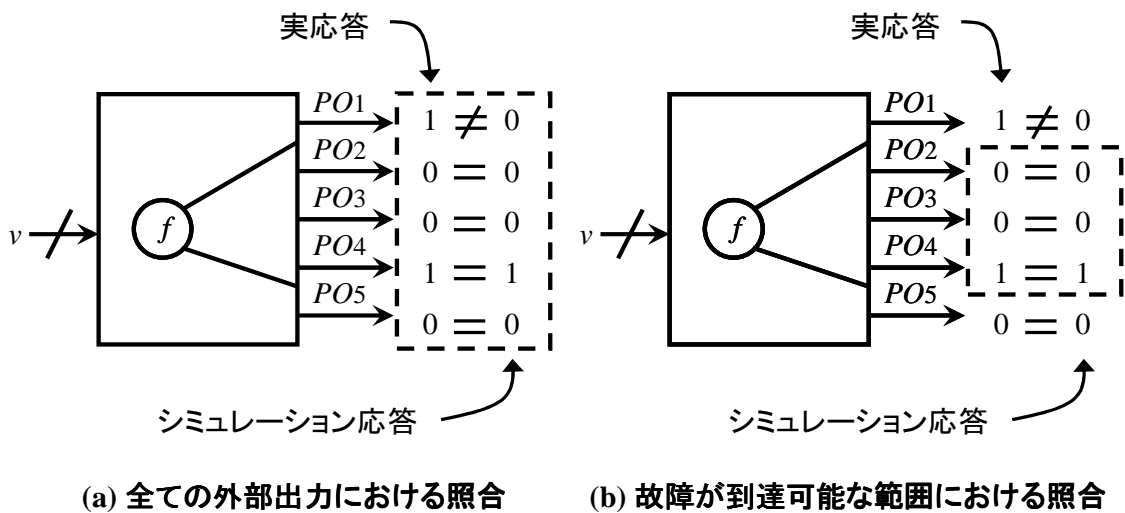


図 5-7 照合の範囲

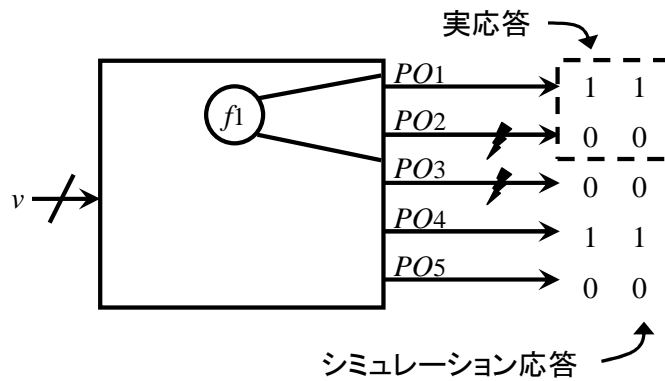
### 5.5.2 照合の詳細

一般的な Per-Test 故障診断手法では、照合の結果は一致あるいは不一致の二値のみであるが、一致の中にも様々な状況があり、図 5-8 は照合の結果は全て一致であるが、詳細は異なっている。X 故障診断手法はこの一致情報を診断結果の生成に役立てている。

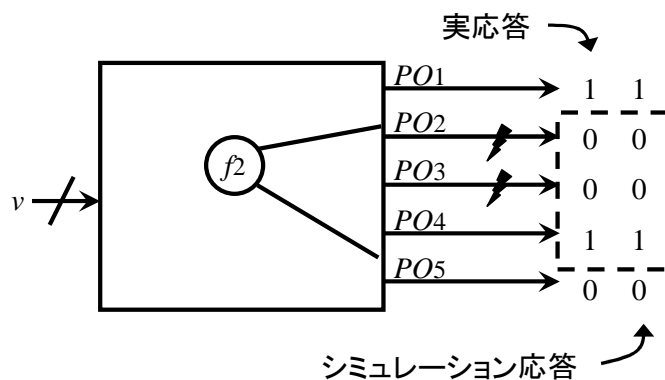
故障が到達可能な外部出力のうち誤りが現れている外部出力のことを一致対象出力と呼び、この一致対象出力数の割合が多い一致情報がより欠陥との関連性が高いと考える。また、故障がどれだけ外部出力からはなれているかの度合いを故障の観測難易度と呼び、この故障の観測難易度が高い一致情報がより欠

陥との関連性が高いと考える。観測難易度は外部出力から故障までのゲート段数によって決定する。

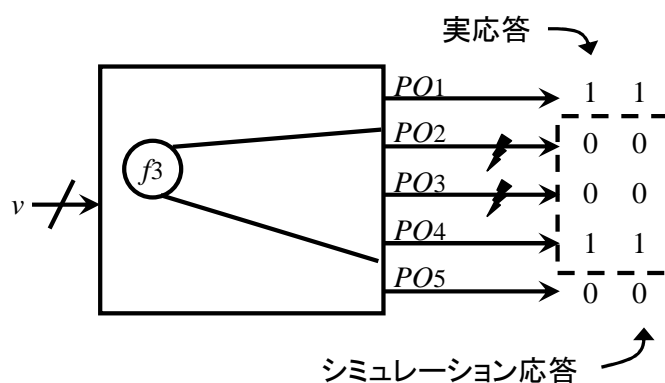
図 5-8 を例にとると、故障  $f_1$  の一致対象出力数は 1、故障  $f_2$  は 2、故障  $f_3$  は 2 である。また、外部出力からのゲート段数は  $f_1 = f_2 < f_3$  である。この場合、 $f_3$ 、 $f_2$ 、 $f_1$  の順に欠陥との関連性が高いとされる。



(a) 故障  $f_1$  の照合結果



(b) 故障  $f_2$  の照合結果



(c) 故障  $f_3$  の照合結果

図 5-8 照合の詳細



### 5.5.3 診断値

一致対象出力数や故障の観測難易度を診断結果の生成に用いるためにはその値を明示的に扱う必要がある。そこで一致対象出力数や故障の観測難易度を数値化した診断値を定義する。

**定義 5-2** :  $v$  をフェイリングベクトルとする。実回路の観測応答に誤りが現れる外部出力の集合を、 $v$  における一致対象出力と呼び、 $Error\_PO(v)$  で表す。

**定義 5-3** : ゲート  $G$  に故障  $f$  があるとする。  $f$  の出力から到達可能な外部出力の集合を、故障  $f$  の到達可能な外部出力と呼び、 $Reach\_PO(f)$  で表す。

図 5-8 では、 $Error\_PO(v) = \{PO2, PO3\}$  であり、 $Reach\_PO(f1) = \{PO1, PO2\}$ 、 $Reach\_PO(f2) = Reach\_PO(f3) = \{PO1, PO2, PO3\}$  である。

**定義 5-4** : 回路における信号線のゲート段数を、それぞれ以下のように定義する。

(1) 外部出力のゲート段数は 1 とする。

(2) ゲート出力のゲート段数が  $l$  ならば、その全ての入力のゲート段数は  $l+1$  とする。

(3) ゲートの全ファンアウト信号線のゲート段数が  $l_1, l_2, \dots, l_h$  ならば、ゲート出力 (ファンアウトシステム) のゲート段数を  $maximum\{l_1, l_2, \dots, l_h\}$  とする。

**定義 5-5** : ゲート  $G$  に故障  $f$  があるとする。  $G$  の出力のゲート段数を  $f$  の故障レベルと呼び、 $Level(f)$  で表す。回路において最も高い故障レベルを  $Lmax$  で表す。

図 5-8 で、 $Level(f1) = Level(f2) = 2$ 、 $Level(f3) = 3$  とすると、 $Lmax = 3$  である。

**定義 5-6** :  $v$  はフェイリングベクトル、 $f$  は X 故障であるとする。実回路の観測応答を  $ObvRes(v)$  とする。また、X 故障シミュレーションを行った結果は

$SimRes(f,v) = \{R_1, R_2, \dots, R_k\}$  ( $k \geq 1$ )であるとする。  $SimRes(f,v)$ と  $ObvRes(v)$ を比較した結果  $d(f,v)$ を診断値と呼び、次式のように求める。

$$d(f, v, R_i) = \frac{Level(f)}{Lmax} \times \frac{|Error\_PO(v) \cap Reach\_PO(f)|}{|Reach\_PO(f)|}$$

$$d(f, v) = \sum_{i=1}^k (d(f, v, R_i) / k)$$

ファンアウトのあるゲートのX故障シミュレーションを行った場合、外部出力において故障値の組合せが  $k$  個存在した場合、診断値の合計を  $k$  で割ったものを診断値とする。不一致の場合には、 $d(f,v,R_i) = 0$  となる。

ここで、 $\frac{|Error\_PO(v) \cap Reach\_PO(f)|}{|Reach\_PO(f)|}$  は一致対象出力の割合、 $\frac{Level(f)}{Lmax}$  は、

故障の観測難易度を表している。

図 5-6 の X 故障シミュレーションの結果は、 $SimRes(f,v) = \{R_1, R_2, R_3\} = \{<111>, <001>, <001>\}$ 、実回路応答  $ObvRes(v) = <001>$ 、正常回路応答は  $<101>$  である。したがって、 $d(f,v,R_1) = 0$ 、 $d(f,v,R_2) = d(f,v,R_3) = (3/3) \times (1/3) = 0.33$  となり、 $d(f,v) = (0 + 0.33 + 0.33) / 3 = 0.22$  が得られる。

#### 5.5.4 故障診断表

診断値は全てのフェイリングベクトルと X 故障に対して計算され、故障診断表に格納される。0 もしくは 1 が格納された通常の故障辞書と比べて、故障診断表は明らかに多くの診断情報を保持している。

表 5-2 に故障診断表の例を示す。また、表中の *ave* は、各故障に対する診断値の平均値であり、故障に対する各 SLAT ベクトルにおける診断値の合計を一致した SLAT ベクトル数で割ったものである。この *ave* の値を平均診断値と呼び、診断結果を生成する時にこの平均診断値を用いて、故障候補に順位を付ける。

表 5-2 診断値を用いた故障診断表

|     | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ |
|-----|-------|-------|-------|-------|-------|
| sv1 | 0.81  | 0.65  | 0     | 0     | 0     |
| sv2 | 0     | 0     | 0.61  | 0.17  | 0     |
| sv3 | 0.26  | 0     | 0.83  | 0     | 0     |
| sv4 | 0     | 0     | 0     | 0     | 0.55  |
| ave | 0.27  | 0.16  | 0.36  | 0.04  | 0.14  |

## 5.6 診断情報を用いたマルチプレットの生成

### 5.6.1 マルチプレットのスコア付け

診断結果は、従来と同じように故障診断表から SLAT ベクトルをすべて説明することができる最小サイズの故障集合であるマルチプレットを求めることで生成する。ただし、故障診断表には診断値を格納しているため、SLAT ベクトルを診断値に 0 が与えられていない故障で被覆することによって得ることが出来る。そして、生成したすべてのマルチプレットに平均診断値に基づいてスコアを付け、このスコアが低いマルチプレットを切り捨てることで故障候補の絞り込みを行う。このマルチプレットに与えられた値は故障シミュレーションの結果と実回路の観測応答の照合から算出しているため客観性の高い結果を得ることができる。

**定義 5-7**：マルチプレットのスコアは、マルチプレットを構成している故障の平均診断値の合計とする。

表 5-2 では、表 5-1 で示した故障診断表の一致情報を示していた 1 の値が格納されていた所に診断値が格納されている。つまり、診断値を用いた故障診断表では値が 0 は不一致情報を表し、0 でない値が格納されている所が一致情報を表している。この例でマルチプレットを生成した場合、表 5-1 のときと同様に  $\{f_1, f_3, f_5\}$ ,  $\{f_1, f_4, f_5\}$ ,  $\{f_2, f_3, f_5\}$  の三つのマルチプレットが得られるが、そのスコアを以下のように求めることができる。

$$\{f1, f3, f5\} = 0.27 + 0.36 + 0.14 = 0.77$$

$$\{f1, f4, f5\} = 0.27 + 0.04 + 0.14 = 0.45$$

$$\{f2, f3, f5\} = 0.16 + 0.36 + 0.14 = 0.66$$

各マルチプレットのスコアによる順位付けに基づき,  $\{f1, f3, f5\}$  が最も高い順位の診断結果となる.

## 5.7 X 故障診断のフロー

X 故障診断の流れを図 5-9 に示す.

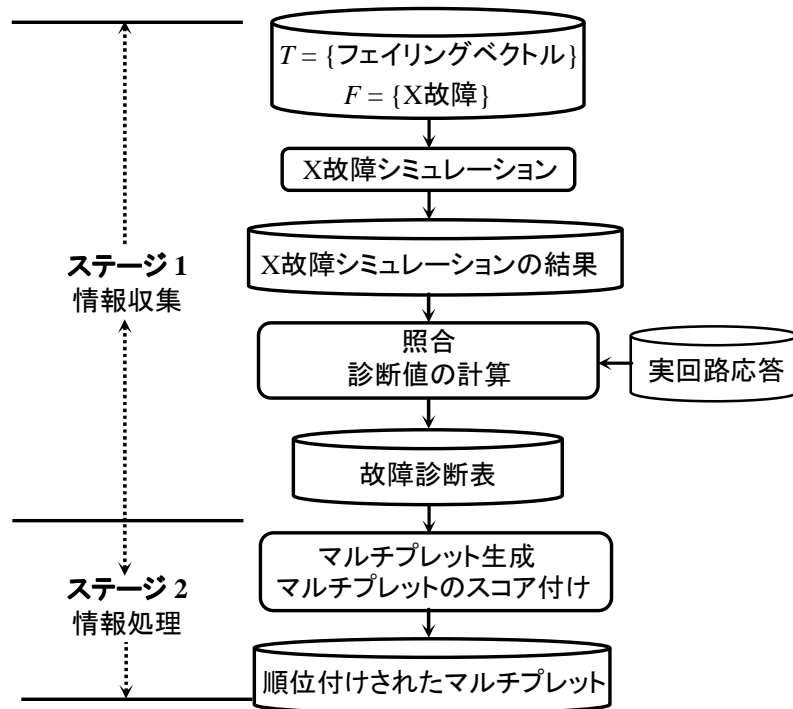


図 5-9 X 故障診断手法の流れ

X 故障診断手法は大きく分けて二つのステージからなり, 基本的には従来の Per-Test 故障診断手法の手順を踏襲するが, X 故障モデル, 診断値を用いる点で従来の Per-Test 故障診断手法と異なる.

**ステージ 1 (情報収集)**: 全ての X 故障とフェイリングベクトルに対して X 故障シミュレーションを行い, シミュレーション応答を実回路応答と比較

した結果を故障診断表に格納する。

**ステージ 2 (情報処理)** : 故障診断表から診断結果を求める。基本的には、全てのフェイリングベクトルを説明できる最小サイズの故障集合を求めることで得られる。そして、マルチプレットに含まれる X 故障が持つスコアの総和をマルチプレットごとのスコアとし、スコアが上位のマルチプレットに含まれる X 故障を診断結果とする。

## 5.8 X 故障診断手法の課題

X 故障診断手法では、多層化された VLSI に多数存在するビアを考慮しておらず、全てのファンアウト信号線を直接接続された信号線として扱っている。このため、微細化 LSI で頻繁に発生するビアの断線などの欠陥箇所を詳細に特定することができず、診断分解能の低下をもたらす恐れがある。また、欠陥によってファンアウト信号線に発生する故障値の出現確率は実際には異なるが、これを全て等しいものとして扱っているため、診断精度に影響を及ぼす可能性がある。



## 第6章 拡張 X 診断手法の提案

### 6.1 概説

従来の X 故障診断手法には、以下の 2 つの問題点がある。

**問題点 1:** ファンアウトに存在するビアを考慮しておらず、全てのファンアウト信号線を直接接続された信号線として扱っている。

**問題点 2:** ファンアウト信号線に起こりうる論理組合せの出現確率を全て等しいものとして扱っている。

本研究では問題点 1 を解決するため、拡張 X 故障モデルを提案し、問題点 2 を解決するため、論理組合せの出現確率の計算法を提案する。また、提案手法を導入した、新しい X 故障診断手法について述べる。実験によって提案手法の有効性を明らかにする。

### 6.2 拡張 X 故障モデル

従来の X 故障モデルでは、ビアの存在は無視されている。図 6-1 のように、ファンアウト数 6 のゲート  $G$  のファンアウト信号線  $L1 \sim L6$  には、ビア  $V1 \sim V3$  の存在に関わらず 6 つの X 記号  $X1 \sim X6$  が割り当てられている。仮にビア  $V2$  に欠陥が発生しているとする。この場合ファンアウト信号線  $L2$  と  $L3$  に故障値が起こりうる。従来の X 故障モデルによる故障診断を行い、最適な診断結果が得られた場合、故障候補として  $f$  が得られる。つまり、欠陥が発生しているビアに接続するゲート出力までは求められても、どのビアで欠陥が発生しているのかまでは特定できない。このように、多くの信号線を同じゲートのファンアウトとして取り扱うことは、診断分解能の低下につながる。実際の微細化 LSI の多くは多層配線であり、広範囲にわたってビアが存在する。本研究ではこのビアの情報をを用いるため、X 故障モデルを以下のように拡張した拡張 X 故障モデルを提案する。

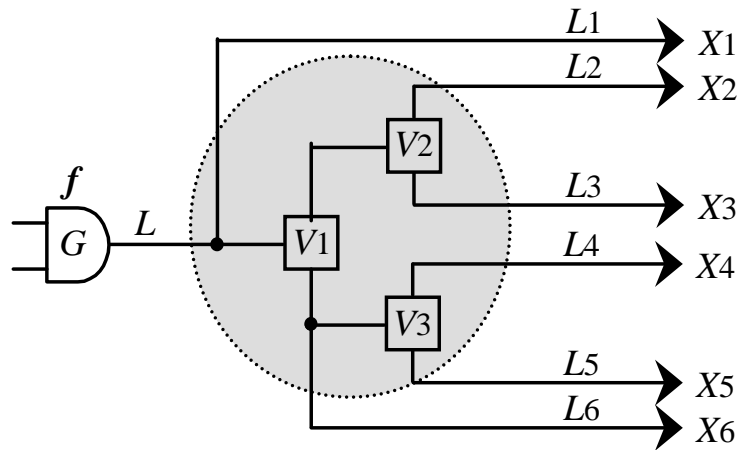


図 6-1 ビアと X 故障

**定義 6-1:** ゲート・ビアは自身もしくはそのファンアウトの欠陥に対応する X 故障を 1 つもつ。ゲート・ビアのファンアウト信号線に起こりうる全ての故障値を表現するため、X 故障はゲート・ビアに接続する全てのファンアウト信号線に異なる X 記号を割り当てる。

拡張 X 故障モデルの例を図 6-2 に示す。拡張 X 故障モデルでは、図 6-1 のようなゲート  $G$  の X 故障だけではなく、ビア  $V_1 \sim V_3$  にも X 故障をそれぞれ仮定する。

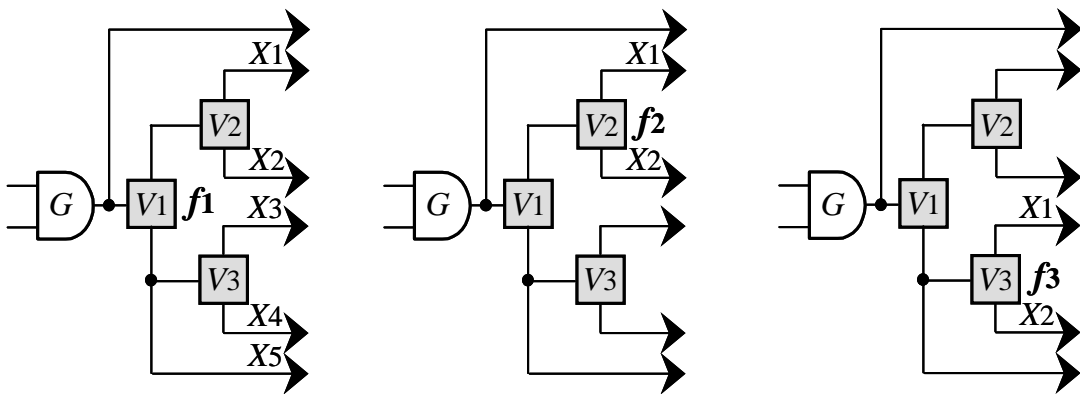


図 6-2 拡張 X 故障モデル

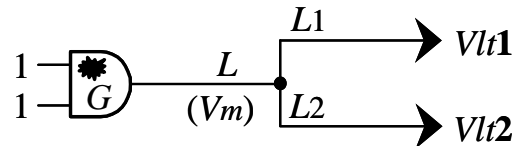
拡張 X 故障モデルの最大の利点は、故障箇所をビアのレベルまで特定できることで、これにより診断分解能を大幅に向上できる。図 6 の場合、診断結果は従来の  $G$  のみから、 $V_1 \sim V_3$  まで分解できる可能性がある。



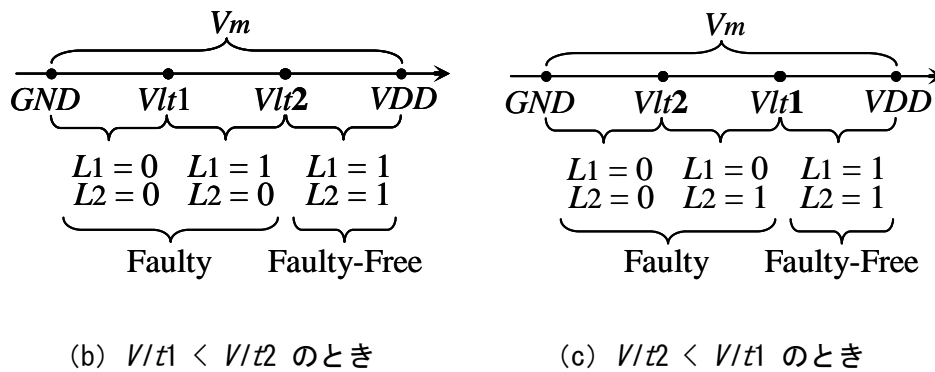
## 6.3 論理組合せ出現確率

### 6.3.1 論理組合せ出現確率の計算

従来の X 故障モデルによる故障診断手法では、ファンアウト信号線での全ての論理組合せは全て同様に起こりうるものとして扱われているが、実際には正確ではない。



(a) 欠陥による中間電圧の発生



(b)  $V_{t1} < V_{t2}$  のとき

(c)  $V_{t2} < V_{t1}$  のとき

図 6-3 中間電圧と起こりうる論理組合せ

図 6-3 のような、2 本のファンアウト信号線  $L1$ ,  $L2$  を持つゲート  $G$  の出力電圧が欠陥により中間電圧  $V_m$  となった場合を考える。  $L1$ ,  $L2$  の閾値電圧がそれぞれ  $V_{t1}$ ,  $V_{t2}$  であるとする。図 6-3(a) のように  $V_{t1} < V_{t2}$  のとき、  $L1$  と  $L2$  に起こりうる論理値の組合せは  $\langle 00 \rangle$ ,  $\langle 10 \rangle$ ,  $\langle 11 \rangle$  で、図 6-3(b) のように  $V_{t1} > V_{t2}$  のとき、  $\langle 00 \rangle$ ,  $\langle 01 \rangle$ ,  $\langle 11 \rangle$  である。低電圧設計の回路では  $V_{t1}$  と  $V_{t2}$  の順序関係は、プロセス変動により予測できないため、どちらの場合も考える必要がある。図 6-3 の場合、  $\langle 00 \rangle$ ,  $\langle 10 \rangle$ ,  $\langle 01 \rangle$ ,  $\langle 11 \rangle$  が発生する確率はそれぞれ  $2/6$ ,  $1/6$ ,  $1/6$ ,  $2/6$  となり、出現確率は異なるものとなる。したがって、全ての論理組合せの出現確率が等しいものとして取り扱っていると、診断分解能の低

下や誤診断を引き起こす可能性がある。

以下では論理組合せ出現確率の計算法について説明する。

ゲートあるいはビアの  $n$  本のファンアウト信号線  $L_1, L_2, \dots, L_n$  の閾値電圧がそれぞれ,  $V_{lt1}, V_{lt2}, \dots, V_{ltn}$  であるとする.  $V_{lt1}, V_{lt2}, \dots, V_{ltn}$  の順序関係が分かっているとき, ファンアウト信号線に起こりうる論理値の組合せは,  $n+1$  通りである[19].

しかしながら, 微細化 LSI のプロセス変動, 低電圧設計による閾値電圧差の微小化によって, ファンアウト信号線の閾値電圧の順序関係を知ることは困難になっているため, 全ての順序関係を考慮する必要がある. これを, 次の定理 1 によって定量的に表現する.

**定理 6-1:** ある素子のファンアウト数が  $n$  のとき, 考えられるファンアウト信号線の閾値電圧の順序関係は  $n!$ 通りである. また, ファンアウト信号線に現れる 0 の個数が  $p(0 \leq p \leq n)$ の論理組合せが発生する確率は

$$\frac{p \times (n-p)!}{(n+1)!}$$

である.

**(証明)**

ファンアウト信号線  $L_1 \sim L_n$  の閾値電圧がそれぞれ  $V_{lt1} \sim V_{ltn}$  であるとし, ファンアウトシステム  $L$  に発生した中間電圧を  $V_m$  とすると,  $V_m$  と  $V_{lt1} \sim V_{ltn}$  によって,  $L_1 \sim L_n$  には異なる論理値が現れる[19].

$n$  本のファンアウト信号線の閾値電圧に順序をつけるとき, 高いものから順に  $n$  通り,  $n-1$  通り, ..., 1 通りの選び方があるので,  $n!$ 通りの順序関係の組合せが考えられる.  $n!$ 通りの閾値電圧の順序関係のそれぞれに対し,  $n+1$  通りの論理組合せが考えられる[19]. したがって, 考えられる全ての論理組合せは,  $(n+1) \times n! = (n+1)!$  通りとなる.

ここで, 論理組合せに  $p(0 \leq p \leq n)$ 個の 0 が現れるということは,  $p$  個の閾値電圧が  $V_m$  より高く,  $(n-p)$ 個の閾値電圧が  $V_m$  より低いことを意味して

いる。つまり、 $p$  個の 0 を含む論理組合せの総数は、 $p! \times (n-p)!$  であることが分かる。ゆえに、このような場合の論理組合せ出現確率は、

$$\frac{p! \times (n-p)!}{(n+1)!}$$

となる。

(証明終)

ファンアウト数  $n$  の場合の論理組合せ出現確率を表 6-1(a)に示す。また、表 6-1(b)に  $n=3$  のときの論理組合せ出現確率を示す。0 の個数によって、論理組合せが発生する確率は明らかに異なっていることが分かる。

表 6-1 論理組合せ出現確率

(a) ファンアウト数  $n$  のとき

| 論理組合せに含まれる0の個数 | 組合せの数              | 発生確率                                |
|----------------|--------------------|-------------------------------------|
| 0              | $n!$               | $1 / (n+1)$                         |
| 1              | $1! \times (n-1)!$ | $1 / (n \times (n+1))$              |
| 2              | $2! \times (n-2)!$ | $2 / ((n-1) \times n \times (n+1))$ |
| ⋮              | ⋮                  | ⋮                                   |
| $p$            | $p! \times (n-p)!$ | $(p! \times (n-p)! / (n+1)!$        |
| ⋮              | ⋮                  | ⋮                                   |
| $n$            | $n!$               | $1 / (n+1)$                         |

(b) ファンアウト数3のとき

| 論理組合せに含まれる0の個数 | 組合せの数 | 発生確率 |
|----------------|-------|------|
| 0              | 6     | 25%  |
| 1              | 2     | ≈ 8% |
| 2              | 2     | ≈ 8% |
| 3              | 6     | 25%  |

### 6.3.2 論理組合せ出現確率の利用

定理 1 に基づいてファンアウト信号線の論理組合せ出現確率を求め、これを X 分解に導入する。

図 6-4 のゲート  $G_1$  の X 故障について考える。  $G_1$  の 3 つのファンアウト信号線  $b_1, b_2, b_3$  で起こりうる故障値の組合せが、それぞれ  $C_1 = \langle 01X \rangle$ ,  $C_2 = \langle 10X \rangle$ ,  $C_3 = \langle 11X \rangle$  であるとする。  $C_1$  は  $\langle 010 \rangle$ ,  $\langle 011 \rangle$  をとり、それぞれの出現確率は 8% であるから  $C_1$  の出現確率は 16% となる。 同様に、  $C_2$  と  $C_3$  の出現確率はそれぞれ 16%, 32% となる。 したがって、どの故障値も等しく 25% の確率で発生すると仮定していた従来の手法とは異なっている。

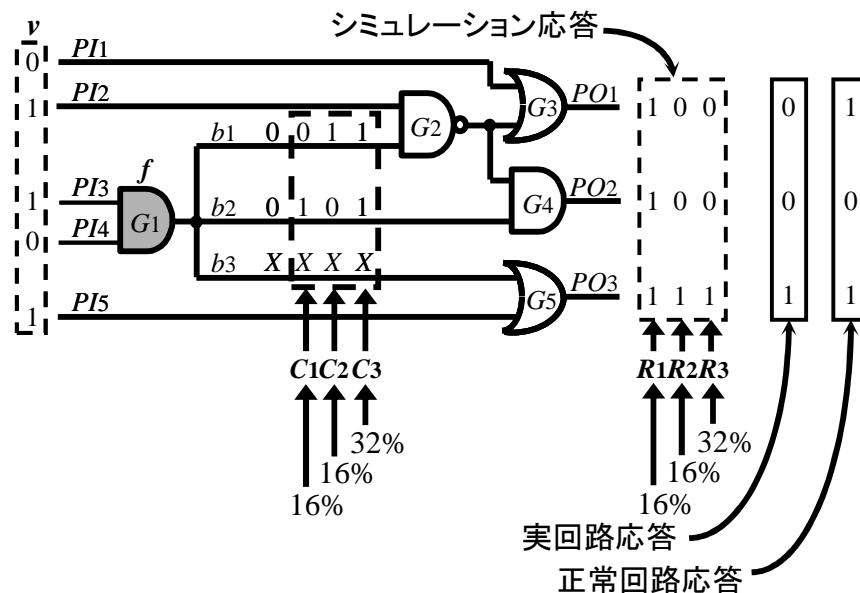


図 6-4 新しい X 分解

X 故障  $f$  のファンアウト信号線に起こりうる論理組合せ  $C_1 \sim C_n$  の出現確率をそれぞれ  $p(C_1), p(C_2), \dots, p(C_n)$ ,  $C_1 \sim C_n$  に対するシミュレーション応答を  $R_1 \sim R_n$  とすると、  $R_1 \sim R_n$  の出現確率  $p(R_1), p(R_2), \dots, p(R_n)$  は、明らかに  $p(C_1), p(C_2), \dots, p(C_n)$  に等しい。

図 6-4 では、ファンアウト信号線に起こりうる論理組合せの出現確率  $p(C_1), p(C_2), p(C_3)$  はそれぞれ 16%, 16%, 32% であるから、外部出力における論理

組合せの出現確率  $p(R_1)$ ,  $p(R_2)$ ,  $p(R_3)$  も同様に、16%, 16%, 32%となる。

### 6.3.3 新しい診断値計算法

5.5.3 節で述べた従来の診断値[9]では、ゲートのファンアウト信号線における故障値の組合せは全て同様に起こりうるという、実際には誤っている仮定の下に定義されていた。以下では、論理組合せ出現確率を考慮した新しい診断値の定義について述べる。

一般に、シミュレーション応答  $SimRes(f,v) = \{R_1, R_2, \dots, R_k\}$  は、実回路応答  $ObvRes(v)$  と比較され、その結果は診断値  $d(f,v)$  として抽出される。新しい診断値の計算法を以下に示す。

$$d(f, v, R_i) = \frac{Level(f)}{Lmax} \times \frac{|Error\_PO(v) \cap Reach\_PO(f)|}{|Reach\_PO(f)|}$$

$$d(f, v) = \sum_{i=1}^k (d(f, v, R_i) \times p(R_i))$$

ここで、 $Error\_PO(v)$ ,  $Reach\_PO(v)$ ,  $Level(f)$ ,  $Lmax$  は 5.5.3 節で定義したものと同一である。

図 6-4 では、 $SimRes(f,v) = \{R_1, R_2, R_3\} = \{<111>, <001>, <001>\}$ ,  $ObvRes(v) = <001>$ , 正常応答は<101>であるから、 $d(f,v,R_1) = 0$ ,  $d(f,v,R_2) = d(f,v,R_3) = (3/3) \times (1/3) = 0.33$  となる。したがって、 $d(f, v) = 0 \times 16\% + 0.33 \times 16\% + 0.33 \times 25\% = 0.14$  となる。この診断値は、5.5.3 節で計算した診断値とは明らかに異なっている。

## 6.4 拡張 X 故障診断手法のフロー

図 6-5 に示すように、拡張 X 故障診断の流れは、基本的には図 5-9 に則っているが、以下の点で従来の手法と異なっている。

- (a) 6.2 節で提案した拡張 X 故障モデルが用いられている。ビアの情報を活用することにより、欠陥箇所をビアのレベルまで特定することができ、診断分解能を向上することができる。

(b) 6.3.3 節で提案した新しい診断値が用いられている。論理組合せ出現確率を考慮することにより、微細化 LSI の実態をよく反映することができ、診断精度の向上に貢献している。

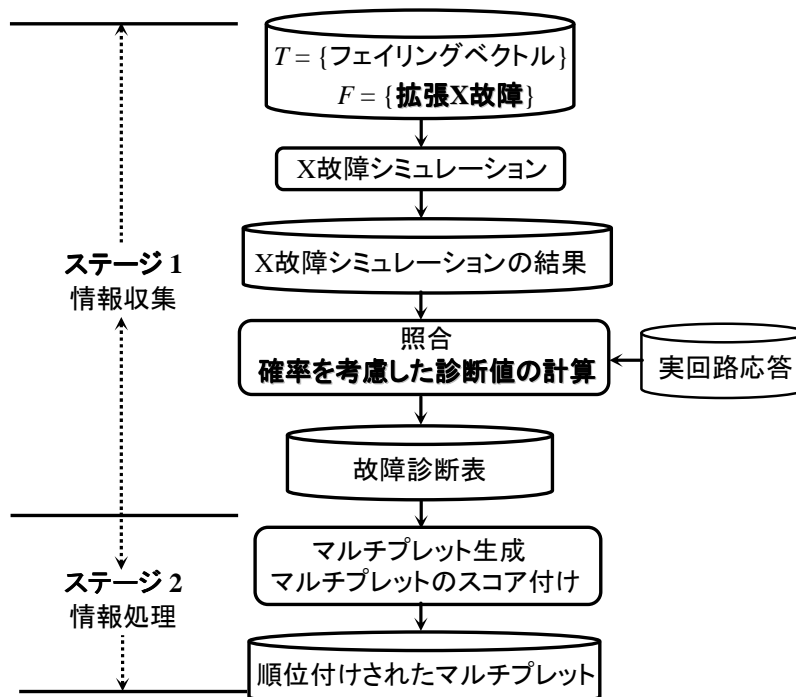


図 6-5 新しい X 故障診断の流れ

## 6.5 評価実験

### 6.5.1 実験の準備

提案した新しい X 故障診断手法を用いて、実験を行った。比較対象は、従来の X 故障診断手法とした。実験は、従来手法と提案手法を Athron 3.0GHz、メモリ 1GB の計算機上に C 言語を用いて実装し、ISCAS'85 と ISCAS'89 のベンチマーク回路に欠陥を仮定したものを診断対象回路として行った。ISCAS'85 ベンチマーク回路は組合せ回路、ISCAS'89 ベンチマーク回路は順序回路の擬似外部出力を観測可能としたフルスキャンバージョンである。

回路に仮定した欠陥はビアの断線としたが、論理回路に対して断線の振舞いを完全に再現するのは困難であるため、本実験ではビアに接続するファンアウト信号線のうちの複数本にランダムに 0 縮退故障もしくは 1 縮退故障を仮定し

た.

## 6.5.2 拡張 X 故障モデルの評価

拡張 X 故障リストは，従来のゲートの X 故障と，ビアの X 故障からなる．まず，拡張 X 故障モデルを用いることによって故障数がどれだけ増加するかを調査した．表 6-2 に調査結果を示す．

表 6-2 拡張 X 故障リストの故障数

| Circuit | #Fault  |             |         |
|---------|---------|-------------|---------|
|         | X-Fault | Ex. X-Fault | Inc.(%) |
| c432    | 160     | 209         | 30.6    |
| c499    | 202     | 302         | 49.5    |
| c880    | 383     | 454         | 18.5    |
| c1355   | 546     | 662         | 21.2    |
| c1908   | 876     | 1093        | 24.8    |
| c2670   | 1193    | 1515        | 27.0    |
| c3540   | 1669    | 2122        | 27.1    |
| c5315   | 2307    | 3318        | 43.8    |
| c6288   | 2416    | 2896        | 19.9    |
| c7552   | 3512    | 4720        | 34.4    |
| s1238   | 508     | 706         | 39.0    |
| s1423   | 657     | 870         | 32.4    |
| s5378   | 2779    | 3371        | 21.3    |
| s9234   | 5597    | 6721        | 20.1    |
| Ave.    |         |             | 29.3    |

表 6-2 の “Circuit” は回路名，“#Fault” は故障数を表す．また，“X-Fault” は従来の X 故障モデルを用いた場合の故障数，“Ex. X-Fault” は拡張 X 故障モデルを用いた場合の故障数を表し，“Inc.(%)” は故障数の増加率を表している．表 6-2 より，拡張 X 故障モデルを用いることによって故障数は平均約 3 割増加していることがわかる．

次に，従来手法(Old)と提案手法(New)の故障診断結果を比較した．実験では各回路に対し 10 種類の故障回路を準備し，故障診断を行った．テストパターンには単一縮退故障用のテストパターンを用いた．表 6-3 に実験結果を示す．表 6-3 の，“#Vector”，“#SLAT”，“#mplt”，“Exact”，“Time” はそれぞれ，テスト

ベクトル数, SLAT ベクトル数, マルチプレット数, 正しい故障箇所を含むマルチプレットの割合, 実行時間を表している. また, “Inc.”は増加率を表している. なお, 実験は各回路につき 10 種の故障回路に対して行ったため, その平均値を記載している.

表 6-3 故障診断結果の比較

| Circuit | #Vector | #SLAT | #mplt |      |         | Exact(%) |      | Time(sec) |       |         |
|---------|---------|-------|-------|------|---------|----------|------|-----------|-------|---------|
|         |         |       | Old   | New  | Inc.(%) | Old      | New  | Old       | New   | Inc.(%) |
| c432    | 28      | 2.6   | 4.7   | 9    | 91.5    | 21.3     | 11.1 | 0.02      | 0.03  | 50.0    |
| c499    | 52      | 4.1   | 28.2  | 35.4 | 25.5    | 3.5      | 2.8  | 0.04      | 0.05  | 25.0    |
| c880    | 21      | 4.7   | 2.3   | 4.3  | 87.0    | 43.5     | 23.3 | 0.04      | 0.06  | 50.0    |
| c1355   | 84      | 8     | 31.1  | 40.1 | 28.9    | 3.2      | 2.5  | 0.38      | 0.32  | -15.8   |
| c1908   | 106     | 2.3   | 35.5  | 43.8 | 23.4    | 2.8      | 2.3  | 0.12      | 0.15  | 25.0    |
| c2670   | 45      | 21.4  | 2     | 5.4  | 170.0   | 50       | 18.5 | 0.43      | 0.73  | 69.8    |
| c3540   | 93      | 5.6   | 9.2   | 14   | 52.2    | 10.9     | 7.1  | 0.54      | 0.75  | 38.9    |
| c5315   | 46      | 9.9   | 13.3  | 19.2 | 44.4    | 6.8      | 4.7  | 1.07      | 1.64  | 53.3    |
| c6288   | 14      | 7.2   | 9.1   | 11.3 | 24.2    | 11       | 8.8  | 5.8       | 5.03  | -13.3   |
| c7552   | 75      | 20.2  | 31.3  | 48.2 | 54.0    | 3.2      | 2.1  | 7.02      | 11    | 56.7    |
| s1238   | 125     | 11.5  | 7.1   | 10.9 | 53.5    | 14.1     | 9.2  | 0.06      | 0.11  | 83.3    |
| s1423   | 24      | 8.9   | 3.1   | 7.2  | 132.3   | 32.3     | 13.9 | 0.43      | 0.47  | 9.3     |
| s5378   | 99      | 24.5  | 10.3  | 15.6 | 51.5    | 9.7      | 6.4  | 1.03      | 1.09  | 5.8     |
| s9234   | 111     | 23.5  | 5.3   | 9.1  | 71.7    | 18.9     | 11   | 22.35     | 26.99 | 20.8    |
| Ave.    |         |       | 13.8  | 19.5 | 65.0    | 16.5     | 8.8  |           |       | 32.8    |

表 6-3 より, SLAT ベクトル数は両手法ともに等しい結果が得られた. 従来の X 故障はファンアウト信号線に現れる全ての振舞いを表すことができる. 拡張 X 故障は従来の X 故障をさらに分解したものであるから, 拡張 X 故障により表される全ての振舞いは従来の X 故障においても表現することが可能である. つまり, 従来の X 故障は拡張 X 故障の振舞い全てを包含する. このため, 従来の X 故障のシミュレーション応答が実応答と一致しなければ, その X 故障に接続する全ての拡張 X 故障は応答が一致しない. このことが原因で SLAT ベクトル数には変化がなかったと考えられる.

故障数の増加によって, マルチプレット数は平均約 6 割増加し, 正しい故障箇所を含むマルチプレットの割合が減少している. また, 故障数の増加は実行時間の増加をもたらし, 本実験では約 3 割の増加がみられた.



### 6.5.3 新しい診断値の評価

論理組合せ出現確率を考慮した新しい診断値を評価するため、正しい故障箇所を含むマルチプレットの最高順位(First Hit)を比較した。実験は、従来の X 故障診断手法(Old)、拡張 X 故障のみを導入した故障診断手法(New-A)、拡張 X 故障と新しい診断値計算法を導入した故障診断手法(New-B)に対して行った。表 6-4 にその結果を示す。

表 6-4 First Hit の評価

| Circuit | First Hit |       |       |
|---------|-----------|-------|-------|
|         | Old       | New-A | New-B |
| c432    | 2.8       | 2.5   | 2.7   |
| c499    | 27.3      | 2.2   | 1     |
| c880    | 1.2       | 1.4   | 1     |
| c1355   | 9         | 1.2   | 1     |
| c1908   | 16.7      | 6.5   | 7     |
| c2670   | 1.3       | 3.6   | 1.9   |
| c3540   | 1.1       | 5.6   | 6.1   |
| c5315   | 1.5       | 4.9   | 5.4   |
| c6288   | 2.5       | 5.6   | 4     |
| c7552   | 4.8       | 5.4   | 3.9   |
| s1238   | 5.1       | 1     | 1     |
| s1423   | 1.2       | 1.5   | 1.1   |
| s5378   | 5.2       | 1.6   | 1.5   |
| s9234   | 1.1       | 3.1   | 3.4   |
| Ave.    | 5.8       | 3.3   | 2.9   |

表 6-4 より、拡張 X 故障モデルを用いることにより First Hit を大きく向上できていることがわかる。表 6-3 のようにマルチプレット数は増加するが、診断値の計算により正しい故障箇所を上位に順位付けることでこの問題は解決されている。また、拡張 X 故障モデルによって得られる故障候補はビアのレベルまで分解されているので、故障数の増加に影響されることなく診断分解能を向上できていることがわかる。

加えて、論理組合せ出現確率を考慮した診断値を導入することにより、First Hit のさらなる向上を可能にしていることがわかる。

## 6.6 まとめ

本章ではビアの情報を考慮した拡張 X 故障モデルと、ゲートあるいはビアのファンアウト信号線に起こりうる故障値の出現確率を導入した、新しい Per-Test 故障診断手法を提案した。実験結果では、拡張 X 故障モデルの使用により診断分解能を向上できることを示した。また、論理組合せ出現確率を導入した新しい診断値の計算が、故障候補の順位付けの高精度化に貢献できることを確認した。

本章で行った実験で用いた回路は、現実に用いられている回路の規模に比べて非常に小さい。したがって、提案手法の現実性を確認するためには、今後はより大規模な回路に対して実験を行う必要がある。また、X 故障モデルはファンアウト信号線に起こりうるすべての振舞いを考慮するため、場合によっては X 故障シミュレーションで行う二値シミュレーションの回数が指数爆発してしまう可能性がある。そのため、X 故障診断手法の実行時間を短縮するためには、X 故障シミュレーションをより高速化する必要がある。

## 第7章 結論

本論文は VLSI の高歩留り化のための低消費電力 X 埋込み技術と、高精度故障診断技術に関する研究成果について述べたものである。

低消費電力 X 埋込み技術はテストパターン中の故障検出とは無関係なビットに、信号値変化が少なくなるように論理値を埋め込む。これにより、テストパターンの故障検出能力を維持しつつ回路の信号値変化の少ないテストパターンを得ることができる。しかし、従来の X 埋込み手法は、削減効果と実行時間の両立ができていない。また、どの手法も FF あるいは全ノードの状態遷移の削減を対象としているが、実際の回路には、電圧降下の影響を受けやすい箇所とそうでない箇所が存在する。そのため、大域的な状態遷移削減では誤テスト回避に繋がらない可能性がある。

第4章では GA による新しい低消費電力 X 埋込み手法を提案した。GA の最大の利点は対象とする問題が変わっても柔軟に対応できることである。そのため、従来の X 埋込み手法では不可能であった局所的な信号遷移の削減にも対応できる。実験では GA-Fill 法は高速かつ効果的に全ノードの WSA を削減できることを確認した。また、ランダムなノードの WSA の削減に関しては高い削減効果は得られなかったが、これは全ノードの WSA 削減の高速化につながる可能性がある。局所的な削減効果を確認するためには、レイアウト情報や電源情報を用いてクリティカルノードを抽出してから実験を行う必要がある。また、GA は選択、交叉などの遺伝的操作には様々な方法があり、その組合せにより、同じ問題に対しても結果が大きく変わる特徴がある。そのため、GA のパラメータを様々に変更して実験を行い、最適なパラメータを求めることによって、GA-Fill はさらに高速化・あるいは信号遷移削減率の向上ができる可能性がある。

製造技術の進歩に伴う微細化、高速化によって、LSI に発生する欠陥の振舞いは複雑化し、故障診断をより困難なものにしているが、この問題に対応すべく提案された、X 故障診断手法は、論理的振舞いが不確定な欠陥に対しても有効にモデル化できる X 故障モデルと、多重・動的な欠陥をも特定できる Per-Test

故障診断手法を用いることで、診断精度の大幅な向上を可能にしている。しかし、ビアに発生している欠陥に対して故障診断を行う場合、従来の X 故障モデルでは欠陥が発生しているビアに接続するゲートの出力までしか求められないため、診断分解能の低下をもたらす恐れがある。また、診断値の計算によって故障候補を順位付けする際に、全ての欠陥の振舞いが同様に起こりうるという誤った仮定の下に行っていたため、正しい故障箇所を上位に順位付けできない可能性がある。

第 6 章では、ビアを考慮した拡張 X 故障モデル、閾値電圧の順序関係を考慮した論理組合せ出現確率を導入した、新しい X 故障診断手法を提案した。拡張 X 故障モデルはビアに発生する欠陥についても考慮してモデル化しているため、従来の X 故障モデルよりも欠陥との関連性が高い故障モデルである。実験では拡張 X 故障モデルの正確なモデル化により診断分解能を向上できることを示した。また、論理組合せ出現確率を考慮した新しい診断値の計算により、存在する可能性が高い故障を正確に上位に順位付けができ、拡張 X 故障モデルの問題点である故障数の増加に対しても有効であることを示した。

LSI の高速化によって、タイミングに影響する欠陥の問題が深刻化しており、このような欠陥を正確に診断できる故障診断手法が求められている。X 故障モデルは複雑に振舞う欠陥に対しても、欠陥情報を損なうことなくモデル化することができるが、タイミングに影響する欠陥に対しては考慮されていない。今後は、タイミングに影響する欠陥に対するモデル化と、正確な故障診断手法の確立について考えていく必要がある。

# 謝辞

本研究は九州工業大学大学院情報工学府情報創成工学専攻において，温暁青教授の御指導の下で行ったものである．本研究を遂行するにあたり，終始懇切なる御指導と御鞭撻を賜り，また有益な議論および御助言を頂きました温暁青教授に心より感謝いたします．

本研究の遂行にあたり，御助言，御討論を頂いた九州工業大学情報工学研究院電子情報工学研究系梶原誠司教授に深く感謝いたします．同じく本論文の作成に関し，御検討，貴重な御教示を頂いた同大学同研究院同研究系笹尾勤教授，および同大学同研究院情報創成工学研究系吉田隆一教授に深く感謝いたします．また，詳細なコメントと洞察に富む御助言を与えて頂いた同大学同研究院電子情報工学研究系宮瀬紘平助教に深く感謝いたします．

温研究室，梶原研究室の諸氏には一方ならぬ御支援を頂きました．ここに記して感謝いたします．



## 参考文献

- [1] L.-T. Wang, C.-W. Wu, and X. Wen, (Editors), VLSI Test Principles and Architectures: Design for Testability, Elsevier, 2006.
- [2] J. P. Roth. Diagnosis of automata failures: A calculus and a method. IBMJ. Res. Develop., Vol. 10, pp. 278-291, July 1966.
- [3] P. Goel. An implicit enumeration algorithm to generate tests for combinational logic circuits. IEEE Trans. Computers, Vol. C-30, pp. 215-222, Mar. 1981.
- [4] H. Fujiwara, et al.. On the Acceleration of Test Generation Algorithms. IEEE Trans. Computers, C-32, pp.1137-1144, Dec. 1983.
- [5] M. Schulz, et al.. SOCRATES: A Highly Efficient Automatic Test Generation System. IEEE Trans. on CAD., pp. 126-137, Jan. 1988.
- [6] H. Takahashi, K. O. Boateng, K. K. Saluja, and Y. Takamatsu, "On Diagnosing Multiple Stuck-At Faults Using Multiple and Single Fault Simulation", IEEE Trans. on Computer-Aided Design, vol. 21, no. 5, pp. 362-368, 2002.
- [7] S. Venkataraman and S. Drummonds, "POIROT: A Logic Fault Diagnosis Tool and Its Applications", Proc. Intl. Test Conf., pp. 253-262, 2000.
- [8] S. D. Millman, E. J. McCluskey, and J. M. Acken, "Diagnosing CMOS Bridging Faults with Stuck-At Fault Dictionaries", Proc. Intl. Test Conf., pp. 860-870, 1990.
- [9] P. Maxwell and R. Aitken, "Biased Voting: A Method for Simulating CMOS Bridging Faults in the Presence of Variable Gate Logic Thresholds", Proc. Intl. Test Conf., pp. 63-72, 1993.
- [10] W. Mao and R. K. Gulati, "QUIETEST: A Quiescent Current Testing Methodology for Detecting Short Faults", Proc. ICCAD'90, pp. 280-283, 1990.
- [11] D. Lavo, T. Larrabee, and B. Chess, "Beyond the Byzantine Generals: Unexpected Behavior and Bridging Fault Diagnosis", Proc. Intl. Test Conf., pp. 611-619, 1996.
- [12] S. Huang, "Speeding Up the Byzantine Fault Diagnosis Using Symbolic Simulation", Proc. VLSI Test Symp., pp. 193-198, 2002.
- [13] X. Wen, T. Miyoshi, S. Kajihara, L.-T. Wang, K. K. Saluja, and K. Kinoshita, "On Per-Test Fault Diagnosis Using the X-Fault Model", Proc. Int'l Conf. on Computer-Aided Design, pp. 633-640, 2004.

- [14] X. Wen, H. Tamamoto, K. K. Saluja, and K. Kinoshita, "Fault Diagnosis for Physical Defects of Unknown Behaviors", Proc. Asian Test Symp., pp. 236-241, 2003.
- [15] ITRS Roadmap: Test & Test Equipment, 2008 Edition, <http://public.itrs.net/>, 2008.
- [16] T. Bartenstein, D. Heaberlin, L. Huisman, and D. Sliwinski, "Diagnosing Combinational Logic Designs Using the Single Location At-a-Time (SLAT) Paradigm", Proc. Intl. Test Conf., pp. 287-296, 2001.
- [17] J. Waicukauski and E. Lindbloom, "Failure Diagnosis of Structured Circuits", IEEE Design and Test of Comp., vol. 6, no. 4, pp. 49-60, 1989.
- [18] D. Lavo, I. Hartanto, and T. Larrabee, "Multiplets, Models and the Search for Meaning", Proc. Intl. Test Conf., pp. 250-259, 2002.
- [19] X. Wen, H. Tamamoto, K. K. Saluja, and K. Kinoshita, "Fault Diagnosis for Static CMOS Circuits", Proc. Asian Test Symp., pp. 282-287, 1997.
- [20] F. Brglez and H. Fujiwara, "A Neutral Netlist of Combinational Benchmark Circuits and a Target Translator in FORTRAN", Proc. Intl. Symp. on Circuits and Systems, pp. 663-698, 1985.
- [21] J. B. Liu, A. Veneris and H. Takahashi, "Incremental Diagnosis of Multiple Open-Interconnects", Proc. Intl. Test Conf., pp.1085-1092, 2002.
- [22] Y. Sato, I. Yamazaki, H. Yamanaka, T. Ikeda, M. Takakura, "A Persistent Diagnostic Technique for Unstable Defects," Proc. ITC, pp.241-249, 2002.
- [23] Z. Wang, K-H. Tsai, M. Marek-Sadowska, J. Rajski, "An Efficient and Effective Methodology on the Multiple Fault Diagnosis" , Proc. Int'l Test Conf., pp. 329-334, 2003.
- [24] Chunsheng Liu, "An Efficient Method for Improving the Quality of Per-Test Fault Diagnosis", Proc. Int'l Conf. on Computer-Aided Design, pp. 648-651, 2004.
- [25] Y. Sato, S. Hamada, T. Maeda, A. Takatori, Y. Nozuyama, and S. Kajihara, "Invisible Delay Quality - SDQM Model Lights Up What Could Not Be Seen," Proc. of ITC, Paper 47.1, 2005.
- [26] L.-T. Wang, C.-W. Wu, and X. Wen, (Editors), VLSI Test Principles and Architectures: Design for Testability, San Francisco: Morgan Kaufmann, first edition, 2006.
- [27] S. Sde-Paz and E. Salomon, "Frequency and Power Correlation between At-Speed Scan and Functional Tests," Proc. of ITC, Paper 13.3, 2008.



- [28] J. Saxena, K. Butler, V. Jayaram, and S. Hundu, "A Case Study of IR-Drop in Structured At-Speed Testing," Proc. of ITC, pp. 1098-1104, 2003.
- [29] R. Sankaralingam and N. A. Touba, "Inserting Test Points to Control Peak Power During Scan Testing", Proc. of Intl. Symp. on DFT, pp. 138-146, 2002.
- [30] S. Gerstendoerfer and H. Wunderlich, "Minimized Power Consumption for Scan-Based BIST," Proc. of ITC, pp. 77-84, 1999.
- [31] K. Lee, T. Huang, and J. Chen, "Peak-Power Reduction for Multiple-Scan Circuits during Test Application," Proc. of ATS, pp. 453-458, 2000.
- [32] P. Girard, "Survey of Low-Power Testing of VLSI Circuits," IEEE Design & Test of Computers, Vol. 19, No. 3, pp. 82-92, Feb. 2002.
- [33] S. Wang and W. Wei, "A Technique to Reduce Peak Current and Average Power Dissipation in Scan Designs by Limited Capture," Proc. of ASP-DAC, pp. 810-816, 2007.
- [34] V. Dabholkar, S. Chakravarty, I. Pomeranz, and S. Reddy, "Techniques for Minimizing Power Dissipation in Scan and Combinational Circuits during Test Application," IEEE Trans. on CAD, Vol. 17, No. 12, pp. 1325-1333, 1998.
- [35] F. Corno, P. Prinetto, M. Redaudo, and M. Reorda, "A Test Pattern Generation Methodology for Low Power Consumption," Proc. of VTS, pp. 35-40, 1998.
- [36] N. Ahmed, M. Tehranipoor, and V. Jayaram, "Supply Voltage Noise Aware ATPG for Transition Delay Faults," Proc. of VTS, pp. 179-184, 2007.
- [37] W. Li, S. M. Reddy, I. Pomeranz, "On Reducing Peak Current and Power during Test," Proc. of ISVLSI, pp. 156-161, 2005.
- [38] X. Wen, Y. Yamashita, K. Kajihara, L.-T Wang, K. K. Saluja, and K. Kinoshita, "Low-Capture-Power Test Generation for Scan Testing," in Proc. of VTS, pp. 265-270, 2005.
- [39] S. Remersaro, X. Lin, Z. Zhang, S. M. Reddy, I. Pomeranz, and J. Rajski, "Preferred Fill: A Scalable Method to Reduce Capture Power for Scan Based Designs," Proc. of ITC, Paper 32.2, 2006.
- [40] X. Wen, K. Miyase, T. Suzuki, Y. Yamato, S. Kajihara, L.-T. Wang, and K. K. Saluja, "A Highly-Guided X-Filling Method for Effective Low-Capture-Power Scan Test Generation," Proc. of ICCD, pp. 251-258, 2006.
- [41] X. Wen, K. Miyase, S. Kajihara, T. Suzuki, Y. Yamato, P. Girard, Y. Ohsumi, and L.-T. Wang, "A Novel Scheme to Reduce Power Supply Noise for High-Quality At-Speed Scan Testing," Proc. of ITC, Paper 25.1, 2007.

- [42] J.-L. Yang and Q. Xu, "State-Sensitive X-Filling Scheme for Scan Capture Power Reduction," *IEEE Trans. on Computer-Aided Design of Integrated Circuits & Systems*, Vol. 27, No. 7, pp. 1338-1343, July 2008.
- [43] P. Girard, X. Wen, and N. A. Touba, Low-Power Testing (Chapter 7) in *Advanced SOC Test Architectures - Towards Nanometer Designs*. San Francisco: Morgan Kaufmann, first edition, 2007.
- [44] P. Wohl, J. A. Waicukauski, S. Patel, and M. B. Amin, "Efficient Compression and Application of Deterministic Patterns in a Logic BIST Architecture," in *Proc. of the Design Automation Conf.*, Jun. 2003, pp. 566-569.
- [45] T. Hiraide, K. O. Boateng, H. Konishi, K. Itaya, M. Emori, H. Yamanaka, and T. Mochiyama, "BIST-Aided Scan Test - A New Method for Test Cost Reduction," in *Proc. of VLSI Test Symp.*, May 2003, pp. 359-364.
- [46] J. Li, Q. Xu, Y. Hu and X. Li, "iFill: An Impact-Oriented X-Filling Method for Shift- and Capture-Power Reduction in At-Speed Scan-Based Testing," *Proc. of DATE*, pp. 1184-1189, 2008.
- [47] S. Remersaro, X. Lin, S. M. Reddy, I. Pomeranz, J. Rajski, "Low Shift and Capture Power Scan Tests," *Proc. of VLSI Design*, pp. 793-798, 2007.
- [48] K. Miyase and S. Kajihara, "XID: Don't Care Identification of Test Patterns for Combinational Circuits," *IEEE Trans. on Computer-Aided Design*, Vol. 23, No. 2, pp. 321-326, Feb. 2004.
- [49] X. Wen, K. Miyase, T. Suzuki, S. Kajihara, Y. Ohsumi and K. K. Saluja, "Critical-Path-Aware X-Filling for Effective IR-Drop Reduction in At-Speed Scan Testing," *Proc. of DAC*, pp. 527-532, 2007.
- [50] K. Kokrady and C. P. Ravikumar, "Static Verification of Test Vectors for IR Drop Failure," *Proc. of ICCAD*, pp. 760-764, 2003.
- [51] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.
- [52] J. Holland, *Adaptation in Natural and Artificial Systems*, The University of Michigan, 1975., and MIT Press, 1992.
- [53] N. Ahmed, M. Tehranipoor, and V. Jayaram, "Transition Delay Fault Test Pattern Generation Considering Supply Voltage Noise in a SOC Design," *Proc. of DAC*, pp. 533-538, 2007.
- [54] M.-F. Wu, J.-L. Huang, X. Wen, and K. Miyase, "Reducing Power Supply Noise in Linear-Decompressor- Based Test Data Compression Environment for At-Speed Scan Testing," *Proc. of ITC*, Paper 13.1, 2008.

- [55] K. P. Parker and E. J. McCluskey, "Probability Treatment of General Combinational Networks," *IEEE Trans. on Computers*, vol. C-24, no. 6, pp. 668-670, Jun. 1975.
- [56] J. Savir. "On Broad-Side Delay Testing," Proc. of VLSI Test Symposium, pp. 284-290, April 1994.
- [57] J. Savir. "Skewed-Load Transition Test: Part I, Calculus," Proc. of International Test Conference, pp. 705-713, 1992.
- [58] J. Savir. "Skewed-Load Transition Test: Part II, Coverage," Proc. of International Test Conference, pp. 714-722, 1992.
- [59] M. Abramovici, M.A. Breuer, and A.D. Friedman, *Digital Systems Testing and Testable Design*. Computer Science Press, 1990.
- [60] M. Abramovici and M.A. Breuer, "Fault Diagnosis Based on Effect-Cause Analysis: An Introduction," Proc. 17th Design Automation Conf., pp. 69-76, June 1980.



# 発表論文一覧

## 論文誌（査読付）

1. Xiaoqing Wen, Seiji Kajihara, Kohei Miyase, Yuta Yamato, Kewal K. Saluja, Laung-Terng Wang, Kozo Kinoshita "A Per-Test Fault Diagnosis Method Based on the X-Fault Model," IEICE Transactions on Information and Systems, Vol. E89-D, No. 11, pp. 2756-2765, Nov. 2006.
2. Yuta Yamato, Yusuke Nakamura, Kohei Miyase, Xiaoqing Wen, and Seiji Kajihara "A Novel Per-Test Fault Diagnosis Method Based On the Extended X-Fault Model for Deep-Submicron LSI Circuits," IEICE Transactions on Information and Systems, Vol. E91-D, No. 3, pp. 667-674, March 2008.
3. Kohei Miyase, Xiaoqing Wen, Hiroshi Furukawa, Yuta Yamato, Seiji Kajihara, Patrick Girard, Laung-Terng Wang, Mohammad Tehranipoor "High Launch Switching Activity Reduction in At-Speed Scan Testing using CTX: A Clock-Gating-Based Test Relaxation and X-Filling Scheme," IEICE Transactions on Information and Systems, Vol. E93-D, No.1, pp.-, Jan. 2010
4. Kohei Miyase, Xiaoqing Wen, Seiji Kajihara, Yuta Yamato, Atsushi Takashima, Hiroshi Furukawa, Kenji Noda, Hideaki Ito, Kazumi Hatayama, Takashi Aikyo, Kewal K. Saluja, "A Study of Capture-Safe Test Generation Flow for At-Speed Testing," IEICE Transactions on Fundamentals (採録済).

## 国際会議（査読付）

1. Xiaoqing Wen, Kohei Miyase, Tatsuya Suzuki, Yuta Yamato, Seiji Kajihara, Laung-Terng Wang, and Kewal K. Saluja, "A Highly-Guided X-Filling Method for Effective Low-Capture-Power Scan Test Generation," 2006 International Conference on Computer Design, Oct. 2006.
2. Xiaoqing Wen, Kohei Miyase, Tatsuya Suzuki, Yuta Yamato, Patrick Girard, Yuji Ohsumi, and Laung -Terng Wang, "A Novel Scheme to Reduce Power Supply Noise for High-Quality At-Speed Scan Testing," Proc. International Test Conference, Paper 25.1, 2007.
3. Xiaoqing Wen, Kohei Miyase, Seiji Kajihara, Hiroshi Furukawa, Yuta Yamato, Atsushi Takashima, Kenji Noda, Hideaki Ito, Kazumi Hatayama, Takashi Aikyo, Kewal K. Saluja "A Capture-Safe Test Generation Scheme for At-Speed Scan Testing," IEEE European Test Symposium pp. 55-60, May. 2008.
4. Kohei Miyase, Kenji Noda, Hideaki Ito, Kazumi Hatayama, Takashi Aikyo, Yuta Yamato, Hiroshi Furukawa, Xiaoqing Wen, Seiji Kajihara, "Effective IR-drop Reduction in At-speed Scan Testing using Distribution-Controlling X-Identification," International Conference on Computer-Aided Design, pp. 52-58, Nov. 2008.
5. Hiroshi Furukawa, Xiaoqing Wen, Kohei Miyase, Yuta Yamato, Seiji Kajihara, Patrick Girard, Laung -Terng Wang, and Mohammad Tehranipoor, "CTX: A Clock-Gating-Based Test Relaxation and X-Filling Scheme for Reducing Yield Loss Risk in At-Speed Scan Testing," Proc. Asian Test Symposium, pp. 397-402, Nov. 2008.
6. Yuta Yamato, Xiaoqing Wen, Kohei Miyase, Hiroshi Furukawa, Seiji Kajihara, "A GA-Based Method for High-Quality X-Filling to Reduce Launch Switching Activity in At-Speed Scan Testing," IEEE 15th Pacific Rim International Symposium on Dependable Computing, pp. 81-86, Nov. 2009.
7. Kazunari Enokimoto, Xiaoqing Wen, Yuta Yamato, Kohei Miyase, Hiroaki Sone, Seiji Kajihara, Masao Aso, Hiroshi Furukawa, "CAT: A Critical-Area-Targeted Test Set Modification Scheme for Reducing Launch Switching Activity in At-Speed Scan Testing," Asian Test Symposium, pp. 99-104, Nov. 2009.
8. Kohei Miyase, Yuta Yamato, Kenji Noda, Hideaki Ito, Kazumi Hatayama, Takashi Aikyo, Xiaoqing Wen, Seiji Kajihara, "High Launch Switching Activity Reduction in At-Speed Scan Testing Using CTX: A Clock-Gating-Based Test Relaxation and X-Filling Scheme," Proc. IEEE/ACM International Conference on Computer-Aided Design, pp. 97-104, Nov 2009.



## 国内研究会

1. 大和勇太, 温暁青, 梶原誠司, 宮瀬紘平, “X 故障モデルを用いた Per-Test 故障診断手法について”, 第 53 回 F T C 研究会資料, July 2005.
2. 大和勇太, 温暁青, 宮瀬紘平, 梶原誠司, K.K. サルージャ, 樹下行三, “LSI 回路の新しい X 故障による Per-Test 故障診断手法について”, LSI テスティングシンポジウム 2006, pp. 193-198, Nov. 2006.
3. 塔ノ上義章, 温暁青, 梶原誠司, 宮瀬紘平, 鈴木達也, 大和勇太, “低消費電力テストのための制約付テスト生成手法について”, 信学技報, vol. 106, no. 468, ICD2006-190, pp. 109-114, Jan. 2007.
4. 中村優介, 大和勇太, 温暁青, 宮瀬紘平, 梶原誠司 “LSI 回路の X 故障による Per-Test 故障診断手法の拡張について”, 信学技報, vol. 107, no. 213, R2007-33, pp. 23-28, Sep. 2007.
5. 大谷雅志, 温暁青, 大和勇太, 宮瀬紘平, 梶原誠司 “Per-Test X 故障診断手法の診断分解能向上について”, LSI テスティングシンポジウム 2007 会議録, pp. 239-244, Nov. 2007.
6. 福澤友晶, 宮瀬紘平, 大和勇太, 古川寛, 温暁青, 梶原誠司 “実速度スキャンテストにおけるキャプチャ時の低消費電力テスト生成手法について”, 信学技報, VLD2007-71, DC2007-26, pp. 7-12, Nov. 2007.
7. 高嶋敦之, 大和勇太, 古川寛, 宮瀬紘平, 温暁青, 梶原誠司 “実速度スキャンテストにおけるキャプチャセーフテスト生成手法について”, 信学技報, DC2008-30(VLD2008-62), pp. 13-18, Nov. 2008.



## ワークショップ (査読付)

1. Xiaoqing Wen, Yuta Yamato, Kohei Miyase, Seiji Kajihara, Hiroshi Furukawa, Laung. -Terng. Wang, Kewal K. Saluja, Kozo Kinoshita, "An Improved Method of Per-Test X-Fault Diagnosis for Deep-Submicron LSI Circuits," 7<sup>th</sup> Workshop on RTL and High Level Testing, pp. 55-60, Nov. 2006
2. Yuta Yamato, Xiaoqing Wen, Kohei Miyase, Hiroshi Furukawa, Seiji Kajihara "GA-Based X-Filling for Reducing Launch Switching Activity in At-Speed Scan Testing," IEEE Int. Workshop on Defect and Data Driven Testing, Oct. 2008.