

不確かなテスト集合によるブリッジ故障診断

西山 隆広[†] 樋上 喜信^{††} 山崎 浩二^{†††} 高橋 寛^{††} 高松 雄三^{††}

[†] 愛媛大学大学院理工学研究科 〒790-8577 愛媛県松山市

^{††} 愛媛大学工学部情報工学科 〒790-8577 愛媛県松山市

^{†††} 明治大学情報コミュニケーション学部 〒214-8571 神奈川県川崎市

あらまし 本稿では、不確かな検出テスト集合と非検出テスト集合を用いて組込み自己テスト (BIST) 環境におけるブリッジ故障診断法について述べる。BIST 環境下においては、検査結果として得られる出力応答が高圧縮であるため、被検査回路の故障を検出するテスト (検出テスト) のみで構成されたテスト集合と検出できないテスト (非検出テスト) のみで構成されるテスト集合に分類することは容易ではなく、検出テストの中に非検出テストが含まれる場合がある。そこで、本研究では、ブリッジ故障を対象に検出テストと非検出テストによって構成された不確かな検出テスト集合と非検出テストのみによって構成される非検出テスト集合で故障候補を推定する診断法を提案する。

キーワード 故障診断, 組込み自己テスト, ブリッジ故障

Bridging Fault Diagnosis Using Ambiguous Test Set

Takahiro NISHIYAMA[†], Yoshinobu HIGAMI^{††}, Kouji YAMASAKI^{†††}, Hiroshi TAKAHASHI^{††},
and Yuzo TAKAMATSU^{††}

[†] Graduate School of Science and Engineering, Ehime University Matsuyama, Ehime, Japan

^{††} Dept. of Computer Science, Faculty of Engineering, Ehime University Matsuyama, Ehime, Japan

^{†††} School of Information and Communication, Meiji University Kawasaki, Kanagawa, Japan

Abstract Fault diagnosis under BIST environment is more difficult because highly compacted signatures make it difficult to obtain the information necessary for diagnosis. Therefore the failing test set that is identified in BIST session includes accidentally non-failing tests. We call the test set that includes failing tests and non-failing tests an "ambiguous detecting test set". In this paper, we propose a method for diagnosing bridging faults assuming that an ambiguous detecting test set and non-failing tests are given.

Key words fault diagnosis, BIST, bridging fault

1. ま え が き

論理回路の大規模化に伴って、テストコストの増大が問題になっている。テストコストにはテスト時間やテストのコストなどが影響している。また、回路の高速化に伴って、実動作速度でテストする at-speed テストの必要性が高まっている。これらの問題を解決する手法として、組込み自己テスト (BIST) 法が提案され、実用化されている。一方、故障と判定された回路に対して故障位置や故障原因を指摘する故障診断の重要性が高まっている。故障診断においてはどのテストが検出テストまたは非検出テストであるかという情報が必要となる。検出テストとは少なくとも一つの外部出力が正常値と異なる値となるテストのことであり、非検出テストとは全ての外部出力の値が正常値と一致するテストのことであり、BIST においては出力応答

は圧縮された署名として得られるが、それを元に、印加したテストを検出テストと非検出テストに分類することは容易ではない。そのため、BIST において故障と判定された回路に対して故障診断を行う場合、外部テストを用いて再びテストを印加し、検出テストと非検出テストに分類した後に故障診断を行う方法がある。しかしながら、この場合には、時間やテストコストに関する BIST の利点を損なうことになる。そこで、BIST によって得られた圧縮された出力応答から診断に必要な情報を取り出し、診断する手法が提案されている [1-6]。特に、文献 [6] では長さの異なる二つのサイクリングレジスタを用いることにより、できるだけ正確に検出テストと非検出テストを分類する方法を提案している。しかしながら、BIST において圧縮された出力応答から全てのテストを正確に検出テストと非検出テストに分類することは困難である。従って、検出テストと非検出

テストが完全に分離されないような診断テスト集合を用いた故障診断法を開発する必要がある。このように検出テストと非検出テストからなるテスト集合を不確かな検出テスト集合と呼ぶ。

検出テスト、非検出テストの例を示す。表1は故障 f に対して、5つのテスト t_1, t_2, t_3, t_4, t_5 を印加したときに検出されるかどうかを表している。○は故障 f を検出することを、×は故障 f を検出しないことを表す。

表1 ブリッジ故障検出の例

	t_1	t_2	t_3	t_4	t_5
f	○	×	×	○	×

この例では、故障 f に対して、 t_1 および t_4 は、検出テストであり、 t_2 および t_3 は、非検出テストである。BIST ではこのように検出テストと非検出テストを完全に分離することが困難であり、たとえば、検出テストと非検出テストが混ざった集合 $\{t_1, t_2, t_4\}$ と、非検出テストだけの集合 $\{t_3, t_5\}$ が与えられる。このとき、 t_1, t_2, t_4 の中でいずれが検出テストであるか不明である。これまで縮退故障についての不確かなテスト集合を用いた診断法は提案されている [7] が、ブリッジ故障を対象とした診断法はほとんどない。

本研究では、ブリッジ故障を対象に検出テストと非検出テストによって構成される不確かな検出テスト集合、および、非検出テストのみによって構成される非検出テスト集合を用いて故障候補を推定する診断法を提案する。ここでは、検出テストにおいてどの外部出力に故障が伝搬したかという情報は用いない。診断の対象としては、ノンフィードバックブリッジ故障を対象とした、ANDブリッジ故障、ORブリッジ故障、ドライブ故障に対する診断アルゴリズムを提案する。各診断アルゴリズムは、縮退故障シミュレーションに基づいており、不確かな検出テスト集合を考慮するため、故障の検出回数を利用して故障候補の推定を行う。ANDブリッジ故障とORブリッジ故障に対する診断アルゴリズムの有効性はISCASベンチマーク回路に対する実験結果によって示される。実験においては、ノンフィードバックブリッジのANDブリッジ故障やORブリッジ故障を挿入した回路に対して、診断アルゴリズムを適用した。また、フィードバックブリッジ故障を挿入した回路に対しても実験を行った。なお、ドライブ故障に対する診断アルゴリズムの有効性の実験はまだ行っていない。

本稿は以下の構成は次のようである。第2章では、ブリッジ故障モデルについて述べる。第3章では、ANDブリッジ故障またはORブリッジ故障に対する診断アルゴリズムを説明する。第4章では、ドライブ故障に対する診断アルゴリズムを説明する。第5章では、診断アルゴリズムをベンチマーク回路に適用した実験結果を示す。第6章では、本研究のまとめを述べる。

2. ブリッジ故障モデル

ブリッジ故障は、回路内の2本の信号線が短絡した故障であり、その故障動作により、いくつかのモデルに分類される。本稿ではこれらのうち、ブリッジ故障がAND、もしくは、ORと

して機能するANDブリッジ故障、もしくは、ORブリッジ故障、さらに、一方の信号線の値がもう一方の信号線の値となるドライブ故障を対象とする。以下では、信号線 x と y のブリッジ故障を $\langle x, y \rangle$ と表記する。

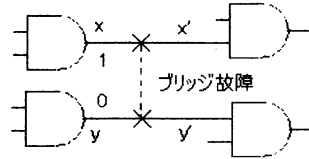


図1 ブリッジ故障の図

例えば、図1に示すように、正常回路における信号線 x, y の値がそれぞれ1, 0となる入力パターンを印加した場合、ANDブリッジ故障では x' と y' は0となり、ORブリッジ故障では x' と y' は1となる。また、ドライブ故障では依存する信号線によって x' と y' は0となったり、1となったりする。このように同じ箇所のブリッジ故障でも、故障のタイプにより誤りの現れ方は異なる。

表2 ノンフィードバックブリッジ故障の振る舞い

aの正常値	bの正常値	AND	OR	aドライブ	bドライブ
0	0	0	0	0	0
0	1	0	1	0	1
1	0	0	1	1	0
1	1	1	1	1	1

短絡した信号線の間に経路がない場合をノンフィードバックブリッジ故障と呼び、経路がある場合をフィードバックブリッジ故障と呼ぶ。表2は、信号線 a と b のノンフィードバックブリッジ故障の信号値を示している。各列は左から、正常回路での a の値、正常回路での b の値、ANDブリッジ故障の場合の信号線 a, b の値、ORブリッジ故障の場合の信号線 a, b の値をそれぞれ表す。 a ドライブおよび b ドライブはドライブ故障で、それぞれ信号線 a の値と一致する場合、信号線 b の値と一致する場合の故障時の値を表している。

表3 フィードバックブリッジ故障の振る舞い

a	b	ANDブリッジ故障		ORブリッジ故障	
		経路あり	経路なし	経路あり	経路なし
1	1	保持	1	1	1
1	0	発振	0	1	1
0	1	0	0	発振	1
0	0	0	0	保持	0

フィードバックブリッジ故障では発振や直前のテストの値を保持する現象が現れることがある [8]。表3は文献 [8] に示されている、信号線 a と b のフィードバックブリッジ故障が生じた場合の故障回路の振る舞いを表している。信号線 a が入力側、 b が出力側であると仮定している。経路ありとは、 a から b への経路上のゲートのサイド入力すべてが非制御値であるような

場合である。また保持とは、直前に印加したテストに依存した値を保持する場合を示している。

3. ANDブリッジ故障またはORブリッジ故障に対する診断アルゴリズム

3.1 概要

本稿では、ノンフィードバック ANDブリッジ故障を対象とした ANDアルゴリズムとノンフィードバック ORブリッジ故障を対象とした ORアルゴリズム、ノンフィードバックドライブ故障を対象としたドライブアルゴリズムを提案する。ここでは、ANDアルゴリズムとORアルゴリズムについて述べる。ANDアルゴリズムとORアルゴリズムに共通する診断手順の概要は以下の通りである。

ANDアルゴリズム (ORアルゴリズム)

入力: 不確かな検出テスト集合 T_u

非検出テスト集合を T_n

対象故障の集合 Set_CFP

故障の検出数に対する閾値 θ

(ただし、不確かな検出テスト集合と非検出テスト集合の間に共通するテストは無いものとする。)

フェーズ 1: テスト $t \in T_u$ に対して縮退故障シミュレーションを行う。その結果、少なくとも一つのテスト $t \in T_u$ によって検出される故障を Set_CFP から削除する。

フェーズ 2: Set_CFP 中にある故障候補について、テスト $t \in T_u$ に対する検出回数を計算し、その値が θ 以下のものを Set_CFP から削除する。

以降説明で用いる α は ANDアルゴリズムの場合は 0 を、ORアルゴリズムの場合は 1 を表すものとする。

3.2 ノンフィードバックブリッジ故障の検出テストの条件

提案法では、縮退故障シミュレーションの結果を基に、ブリッジ故障が検出されるかどうかを調べる。そこで、テスト t が、ノンフィードバックブリッジ故障 $\langle a, b \rangle$ の検出テストとなる条件をいかに示す。ただし、論理値が α となる信号線の集合を N_t 、 α 縮退故障が検出される信号線の集合を F_t 、 α 縮退故障が検出されない信号線の集合を P_t とする。

[補題 1]: テスト t がノンフィードバックブリッジ故障 $\langle a, b \rangle$ の検出テストとなるのは次のどちらかの条件を満たす場合である。

1: $a \in N_t$ かつ $b \in F_t$

2: $a \in F_t$ かつ $b \in N_t$

証明: ノンフィードバックブリッジ故障 $\langle a, b \rangle$ がテスト t によって検出される場合、信号線 a, b の論理値は異なる。また正常回路の論理値が $\bar{\alpha}$ である信号線は α 縮退故障が検出されなければならない。したがって、信号線 a または b の一方が N_t に含まれ、他方が F_t に含まれる場合、故障 $\langle a, b \rangle$ がテスト t によって検出される。

3.3 非検出テスト集合を用いた故障候補の削減

診断アルゴリズムのフェーズ 1 では、補題 1 を用いて少なく

とも一つのテスト $t \in T_u$ で $a \in N_t$ かつ $b \in F_t$ 、または、 $a \in F_t$ かつ $b \in N_t$ となる故障 $\langle a, b \rangle$ を Set_CFP から削除する。もし、そのような $\langle a, b \rangle$ が存在したならば、 t が検出テストとなり、 T_u に含まれるすべての t が非検出テストであるという事実と矛盾する。従って、そのような $\langle a, b \rangle$ は被検査回路に存在しないと考えられる。フェーズ 1 の詳しい手順を以下に示す。

1: 全てのテスト $t \in T_u$ に対して 2~5 の処理を行う。

2: Set_CFP 中にあるノンフィードバックブリッジ故障を構成する信号線全てを含む集合を Set_L_CFP とする。

3: 論理シミュレーションを行い、 Set_L_CFP 中の信号線で論理値が α となる信号線の集合を Set_Val_α とする。

4: Set_L_CFP 中の信号線で α 縮退故障が検出される信号線の集合を Set_detect_α とする。

5: Set_detect_α に属する信号線と Set_Val_α に属する信号線の組合せからなるノンフィードバックブリッジ故障が Set_CFP にあるなら Set_CFP からその故障を除く。

上のアルゴリズムを以下の例を用いて説明する。まず、 $\text{Set_CFP} = \{ \langle l_1, l_2 \rangle, \langle l_1, l_3 \rangle, \langle l_2, l_4 \rangle, \langle l_3, l_5 \rangle, \langle l_4, l_6 \rangle \}$ 、 $T_u = \{ t_1, t_2 \}$

が与えられたとする。処理 2 では、 Set_CFP 中にあるノンフィードバックブリッジ故障を構成する信号線全てを含む集合を Set_L_CFP とする。ここでは、

$$\text{Set_L_CFP} = \{ l_1, l_2, l_3, l_4, l_5, l_6 \}$$

となる。次に、処理 3 では、論理シミュレーションを行い、 Set_L_CFP 中の信号線で論理値が α となる信号線の集合を Set_Val_α とする。ここでは、

$$\text{Set_Val}_\alpha = \{ l_1, l_4 \}$$

が得られたとする。さらに、処理 4 では、 Set_L_CFP 中の信号線で α 縮退故障を検出する信号線の集合を Set_detect_α とする。ここでは、

$$\text{Set_detect}_\alpha = \{ l_2, l_3 \}$$

が得られたとする。その後、処理 5 では、 Set_detect_α に属する信号線と Set_Val_α に属する信号線の組合せからなるノンフィードバックブリッジ故障が Set_CFP にあるならそれを Set_CFP から除く。まず、 Set_detect_α に属する信号線と Set_Val_α に属する信号線の組合せは

$$\{ \langle l_1, l_2 \rangle, \langle l_1, l_3 \rangle, \langle l_2, l_4 \rangle, \langle l_3, l_4 \rangle \}$$

となり、このうち Set_CFP 中にある

$$\{ \langle l_1, l_2 \rangle, \langle l_1, l_3 \rangle, \langle l_2, l_4 \rangle \}$$

を Set_CFP から削除し、

$$\text{Set_CFP} = \{ \langle l_3, l_5 \rangle, \langle l_4, l_6 \rangle \}$$

となる。

これを、 t_2 に対しても行うことにより故障候補を削減する。

3.4 不確かな検出テスト集合を用いた故障候補の削減

診断アルゴリズムのフェーズ 2 では、フェーズ 1 で得られた故障候補に対して、不確かな検出テスト集合 T_u 中のテストを印加した場合の検出回数を求め、検出回数が θ 以上のノンフィー

ドバックブリッジ故障のみを故障候補とする。これは、 T_u の中には、検出テストと共に非検出テストが含まれている可能性があるからである。フェーズ 2 の詳しい手順を以下に示す。

- 1: Set_CFP に含まれる全てのノンフィードバックブリッジ故障に対するカウンタ変数を作成し、その値をすべて 0 とする。
- 2: Set_CFP の中にあるノンフィードバックブリッジ故障を構成する信号線の全てを含む集合を Set_L_CFP とする。
- 3: 全てのテスト $t \in T_u$ に対して 4~6 の処理を行う。
- 4: 論理シミュレーションを行い、Set_L_CFP の中の信号線で論理値が α となる信号線の集合を Set_Val α とする。
- 5: Set_L_CFP の中の信号線で α 縮退故障が検出される信号線の集合を Set_detect α とする。
- 6: Set_detect α に属する信号線と Set_Val α に属する信号線の組合せからなるノンフィードバックブリッジ故障が Set_CFP にあるなら、そのノンフィードバックブリッジ故障に対するカウンタを 1 増やす。
- 7: Set_CFP の中の各ノンフィードバックブリッジ故障について、カウンタの値が θ 以下ならばその故障を除く。

フェーズ 1 とフェーズ 2 の違いは、主に二つある。一つは、用いられるテストであり、フェーズ 1 では非検出テスト集合を用い、フェーズ 2 では不確かな検出テスト集合を用いる。もう一つの違いは、候補でないノンフィードバックブリッジ故障を削除するタイミングである。フェーズ 1 では各テストに対して検出されるノンフィードバックブリッジ故障を求め、それを Set_CFP から削減する。すなわち、故障候補の削減を各テストに対して逐次的に行う。フェーズ 2 では全てのテストに対して検出されるノンフィードバックブリッジ故障を求めた後、故障候補を一度に削減する。フェーズ 2 では、検出回数を調べるため、全てのテストについて検出されるかどうか調べる必要があり、フェーズ 1 と同様の手順を適用することはできない。フェーズ 1 では、非検出テストのみを用いるため、少なくとも一つの非検出テストで検出されたノンフィードバックブリッジ故障は故障候補とならないため、各テストに対して故障候補の削減を行う。一方フェーズ 2 では、不確かな検出テスト集合のすべてのテストに対して検出回数を求めるため、すべてのテストに対して故障シミュレーションした後、故障候補の削減を行う。

4. ドライブ故障に対する診断アルゴリズム

4.1 概要

ここでは、ノンフィードバックドライブ故障に対する診断アルゴリズム(ドライブアルゴリズム)を述べる。ドライブアルゴリズムも、AND アルゴリズムや OR アルゴリズムと同様、非検出テスト集合に対する故障シミュレーション、不確かな検出テスト集合に対する故障シミュレーションの二つのフェーズからなっている。ドライブアルゴリズムでは、二つの信号線 a,b に対して、a ドライブと b ドライブの二つの可能性を考慮し、両方の可能性がなくなった場合のみ、a,b からなるドライブ故障を故障候補集合から削除する。ドライブ故障に対する診断アルゴリズムの概要は次のようである。

ドライブアルゴリズム

入力: 不確かな検出テスト集合 T_u

非検出テスト集合を T_u

対象故障の集合 Set_CFP

故障の検出数に対する閾値 θ

(ただし、不確かな検出テスト集合と非検出テスト集合の間に共通するテストは無いものとする。)

フェーズ 1: すべてのテスト $t \in T_u$ に対して、縮退故障シミュレーションを行う。Set_CFP の中にある故障候補について、もし a ドライブと b ドライブの各故障が、少なくとも一つのテストで検出されるならば、その故障を Set_CFP から削除する。

フェーズ 2: Set_CFP の中にある故障候補について、テスト $t \in T_u$ に対する検出回数を計算し、a ドライブまたは b ドライブのいずれかの検出回数が θ 以上となった故障を候補とし、それ以外の故障を Set_CFP から削除する。

4.2 非検出テスト集合を用いた故障候補の削減

ここではフェーズ 1 の詳しい手順を示す。ドライブ故障では、a ドライブと b ドライブの二つのタイプを考慮する。各故障に対して flag_a, flag_b という二つの変数を用意し、それらは、与えられた非検出テスト集合に対して a ドライブや b ドライブが検出されるかどうかを示している。いずれもが検出された場合のみ、対応するドライブ故障を候補集合から削除する。また、AND アルゴリズムや OR アルゴリズムでは 0 縮退故障または 1 縮退故障のみを調べるだけで十分であったが、ドライブアルゴリズムでは、0 縮退故障と 1 縮退故障の両方の検出について計算を行っている。

1: Set_CFP に含まれる全てのドライブ故障に対する flag_a, flag_b を作成し、その値を全て 0 とする。

2: 全てのテスト $t \in T_u$ に対して 3~7 の処理を行う。

3: Set_CFP の中にあるドライブ故障を構成する信号線全てを含む集合を Set_L_CFP とする。

4: 論理シミュレーションを行い、Set_L_CFP の中の信号線で論理値が 0 となる信号線の集合を Set_Val_0 とし、論理値が 1 となる信号線の集合を Set_Val_1 とする。

5: Set_L_CFP の中の信号線で 0 縮退故障が検出される信号線を Set_detect_0 とし、1 縮退故障が検出される信号線を Set_detect_1 とする。

6: ドライブ故障の一方の信号線を a, もう一方を b とする。もし、信号線 a が Set_detect_0(1) に含まれ、信号線 b が Set_Val_0(1) に含まれるならば、flag_a を 1 とする。同様に、もし信号線 b が Set_detect_0(1) に含まれ、信号線 a が Set_Val_0(1) に含まれるならば、flag_b を 1 とする。

7: flag_a=1 かつ flag_b=1 のドライブ故障を Set_CFP から除く。

4.3 不確かな検出テスト集合を用いた故障候補の削減

ここではフェーズ 2 の詳しい手順を示す。ドライブアルゴリズムでも、AND アルゴリズムや OR アルゴリズムと同様、不確かな検出テスト集合に対する各故障候補の検出回数を調べる。

各故障候補に対して変数 count_a, count_b を用意し、それぞれ a ドライブ, b ドライブに対する検出回数を表す。検出回数を調べた結果、a ドライブまたは b ドライブの検出回数が閾値 θ より大きい場合、対応する故障を故障候補とする。

- 1: Set_CFP に含まれる全てのドライブ故障に対する count_a, count_b を作成し、その値を全て 0 とする。
- 2: Set_CFP の中にあるドライブ故障を構成する信号線の全てを含む集合を Set_L_CFP とする。
- 3: 全てのテスト $t \in T_a$ に対して 4~6 の処理を行う。
- 4: 論理シミュレーションを行い、Set_L_CFP 中の信号線で論理値が 0 となる信号線の集合を Set_Val_0 とし、論理値が 1 となる信号線の集合を Set_Val_1 とする。
- 5: Set_L_CFP 中の信号線で 0 縮退故障が検出される信号線の集合を Set_detect_0 とし、1 縮退故障が検出される信号線の集合を Set_detect_1 とする。
- 6: ドライブ故障の一方の信号線を a, もう一方を b とする。もし、信号線 a が Set_detect_0(1) に含まれ、信号線 b が Set_Val_0(1) に含まれるならば、count_a を 1 増やす。同様に、もし信号線 b が Set_detect_0(1) に含まれ、信号線 a が Set_Val_0(1) に含まれるならば、count_b を 1 増やす。
- 7: Set_CFP 中の各ドライブ故障について、「flag_a=0 かつ count_a > θ 」または「flag_b=0 かつ count_b > θ 」ならばそのドライブ故障を Set_CFP に残り、そうでなければ除く。

5. 実験結果

5.1 閾値の決定

閾値の決定のためにフェーズ 2 における故障候補の検出回数を調べた。表 4 は c7552 の回路に AND ブリッジ故障を挿入し、AND アルゴリズムでのフェーズ 2 を行ったときの各故障候補に対する検出回数を表したものである。

表 4 故障候補の検出テスト集合での検出回数 (c7552 の回路で AND ブリッジ故障を挿入し、AND アルゴリズムで診断)

検出回数	故障候補数
0	516
1	4
2	2
47	1

この表 4 より、フェーズ 2 の Set_CFP に含まれる故障候補の検出回数は 0 付近に集中し、検出回数が多い故障候補は 1 個しかないことがわかった。また、多くの回路で同様に故障候補の検出回数は 0 付近に集中し、検出回数が多い故障候補は数個しかないという結果が見られた。このことから、閾値は不確かな検出テスト集合内の検出テスト数を越えない範囲で検出テスト数に近づければ近い程良いと考えられる。結論として、実験においては、検出回数に対する閾値 θ を不確かな検出テスト集合のテスト数の半分とした。

5.2 ノンフィードバックブリッジ故障

診断アルゴリズムを C 言語で実装し、ノンフィードバック

AND ブリッジ故障とノンフィードバック OR ブリッジ故障に対する実験を行った。実験には、ISCAS'85 ベンチマーク回路および ISCAS'89 ベンチマーク回路の組合せ回路部とパターン長 1024 のランダムテストを用いた。不確かな検出テスト集合の中に含まれる検出テスト数を表す確からしさを、以下の式で定義する。

$$\text{確からしさ} = \frac{\text{不確かな検出テスト集合中の検出テスト数}}{\text{不確かな検出テスト集合中のテスト数}}$$

実験においては、非検出テストをランダムに選択し、検出テストと合わせて不確かな検出テスト集合を作成した。ブリッジ故障は、ISCAS'85 ベンチマーク回路に対しては故障解析ツール Carafe[9] によって得られた故障を用いた。ISCAS'89 ベンチマーク回路に対してはランダムに選択した故障を用いた。各ベンチマーク回路に対して、対象故障からランダムにサンプリングした 1 個の故障を AND ブリッジ故障または OR ブリッジ故障として挿入したそれぞれ 10 個の故障回路に対して診断を行った。

表 5 は、確からしさを 66% とし、ノンフィードバック AND ブリッジ故障を挿入し、AND アルゴリズムを適用した結果を示している。表中の第 2 列から第 5 列には、故障候補数が 1 個、2~5 個、6~10 個、11 個以上となった回路数を示している。実験の結果、ほとんどの回路で故障候補を 1 個とすることができた。

表 5 確からしさ 66% で、AND ブリッジ故障を挿入し、AND アルゴリズムで診断した診断結果

回路名	故障候補数			
	1	2-5	6-10	11-
c432	10	0	0	0
c499	9	1	0	0
c880	10	0	0	0
c1355	9	1	0	0
c1908	8	1	0	1
c2670	9	1	0	0
c3540	9	1	0	0
c5315	9	1	0	0
c6288	9	1	0	0
c7552	10	0	0	0
cs9234	10	0	0	0
cs13207	9	1	0	0
cs15850	10	0	0	0
cs35932	10	0	0	0
cs38417	10	0	0	0
cs38584	10	0	0	0

さらに、ノンフィードバック AND ブリッジ故障を挿入し OR アルゴリズムを適用した場合、ノンフィードバック OR ブリッジ故障を挿入し OR アルゴリズムまたは AND アルゴリズムを適用した場合についても実験を行った。それらの結果を表 6 に示す。表 6 では、実験を行った 16 種類のベンチマーク回路それぞれに対して 10 個の故障回路、すなわち 160 個の故障回路に対する実験結果を示している。第 1 列には、挿入した故障の種

類(上側)と適用したアルゴリズム(下側)を示す。第2列は確からしさを示し、100%または66%とした場合の結果を示す。“故障候補数”の列には、故障候補数が、0個、1個、2~5個、(または、1~5個)、6~10個、11個以上となった故障回路数を示す。また、“有”の列には、得られた故障候補中に挿入した故障が含まれていた場合を、“無”の列には、得られた故障候補中に挿入した故障が含まれていなかった場合を表す。AND/ANDで66%の行は、表5の結果を足し合わせたものと同一である。

表6 診断結果

故障/ アルゴ リズム	確か らしさ	故障候補数								
		有					無			
		0	1	2-5	6-10	11-	1-5	6-10	11-	
AND/ AND	100%	0	147	11	1	1	0	0	0	0
	66%	0	150	9	0	1	0	0	0	0
AND/ OR	100%	134	13	7	0	0	5	0	1	
	66%	137	16	2	0	0	4	1	0	
OR/ AND	100%	132	17	4	1	0	5	1	0	
	66%	133	14	7	0	0	6	0	0	
OR/ OR	100%	0	145	13	0	2	0	0	0	
	66%	0	146	11	1	2	0	0	0	

表6より、ほとんどの故障回路において、挿入したブリッジ故障と同じタイプの診断アルゴリズムを適用した場合には、故障候補を1個に絞り込み、異なったタイプの診断アルゴリズムを適用した場合には、故障候補を0個とすることができた。さらに、挿入したブリッジ故障と異なったタイプの故障アルゴリズムを適用した場合でも、高い確率で故障候補の中に挿入したブリッジ故障が含まれていた。

また、確からしさが66%の結果と100%の結果を比較した場合、確からしさが66%の方が良い結果となる場合がある。これは、66%の場合の方が不確かな検出テスト集合が大きくなり、その結果、検出回数に対する閾値 θ も大きくなったため、検出回数の少ない故障が候補から削除されたためと考えられる。

5.3 フィードバックブリッジ故障

ここではフィードバックブリッジ故障を挿入し、提案アルゴリズムを適用した実験の結果を示す。ISCAS'89ベンチマーク回路5種類についてそれぞれ10個の故障回路を作成し実験した。確からしさは100%および70%とした。故障回路に対して印加したテストを分類する際、発振となるテストは検出テストと非検出テストのいずれにも分類せず、診断には用いなかった。

表7 診断結果

故障/ アルゴ リズム	確か らしさ	故障候補数								
		有					無			
		0	1	2-5	6-10	11-	1-5	6-10	11-	
AND/ AND	100%	2	17	10	5	7	2	0	0	
	70%	4	23	6	2	4	3	0	1	
AND/ OR	100%	28	2	0	0	1	7	0	5	
	70%	33	2	0	0	1	2	1	4	

表7のAND/AND、AND/ORの行には、フィードバック

ANDブリッジ故障を挿入し、ANDアルゴリズム、ORアルゴリズムを適用した結果を示す。“確からしさ”、“故障候補数”の各列の意味は表6と同一である。フィードバックブリッジ故障では、印加した1024のランダムテストのすべてが非検出テストであった場合が存在し、そのような場合は表7に含めていない。表7の結果を表6のノンフィードバックの結果と比較した場合、フィードバック故障では、故障候補がやや多くなった。これは、信号値の保持という現象が起こった場合、提案アルゴリズムでは考慮されていないため、検出回数に違いが生じたためと考えられる。また不確かな検出テスト集合のテスト数が少ない場合や、非検出テスト数が少ない場合にも、故障候補数が多くなる傾向があった。

6. むすび

不確かな検出テスト集合を用いたBIST環境下のブリッジ故障診断法を提案し、その評価を行った。実験結果から提案法では、検出テスト集合に非検出テストが含まれるような状況であっても、故障候補を1個~数個程度にすることができた。また、故障のモデルを異なって診断しても、多くの回路で故障候補が0個になるか挿入した故障と同じ箇所が故障候補となる診断結果を得られた。提案アルゴリズムはノンフィードバック故障を対象としているが、フィードバックブリッジ故障が存在する回路であっても、多くの場合、故障候補を数個以下とすることができた。しかしながら、さらに故障候補を少なくするために、フィードバックブリッジ故障を対象とした診断アルゴリズムを今後開発する予定である。また、ドライブ故障に対しては今回提案した診断アルゴリズムの評価実験を行う予定である。

謝辞 本研究は一部、(株)半導体理工学研究センター(STARC)との共同研究、および、文部科学省科学研究費補助金・基盤研究(C)(2)(課題番号15500043)の研究助成による。

文 献

- [1] I. Bayraktaroglu and A. Orailoglu, "Cost-Effective Deterministic Partitioning for Rapid Diagnosis in Scan-Based BIST," IEEE Design & Tests Comp. Jan-Feb 2002, pp.42-53, 2002.
- [2] I. Bayraktaroglu and A. Orailoglu, "Deterministic Partitioning Technique for Fault Diagnosis in Scan-Based BIST," Proc. ITC, pp.273-282, 2000.
- [3] I. Bayraktaroglu and A. Orailoglu, "Improved Fault Diagnosis in Scan-Based BIST via Superposition," Proc. DAC, pp.55-58, 2000.
- [4] Y. Wu and S. M. I. Adham, "Scan-Based BIST Fault Diagnosis," IEEE Trans. CAD, vol. 18, no.2, pp.203-211, 1999.
- [5] J. Rajski and J. Tyszer, "Diagnosis of Scan Cells in BIST Environment," IEEE Trans. CAD, vol. 48, no. 7, pp.724-731, 1999.
- [6] J. G-Dastidar, D. Das and N. A. Toubia, "Fault Diagnosis in Scan-Based BIST Using Both Time and Space Information," Proc. ITC, pp.95-102, 1999.
- [7] H. Takahashi, Y. Tsugaoka, H. Ayano and Y. Takamatsu, "BIST Based Diagnosis Using Ambiguous Test Set," Proc. DFT, pp.89-96, 2003.
- [8] 山田輝彦, 山崎浩二, "組合せ回路における単一短絡故障の診断法," 信学論D-1, vol. J74-D-1, no.1, pp58-64, 1991.
- [9] A. Jee and F. J. Ferguson, "Carafe: An inductive fault analysis tool for CMOS VLSI circuits," Proc. VTS, pp.92-98, 1993.