

マルチサイクルキャプチャテスト集合を用いたフルスキャン設計回路の単一故障診断法の評価

日大生産工(院) ○諏訪 弘樹
明大 山崎 浩二

日大生産工 細川 利典

1. はじめに

近年、半導体微細化技術の進歩に伴い、大規模集積回路（Very Large Scale Integrated circuit: VLSI）の故障診断[1]には多大な時間と費用を要するようになっている。故障診断の効率化は、コストの削減のために極めて重要である。

故障診断[1]とは、故障回路の外部出力応答を説明できる故障箇所を指摘することである。故障を原因とし、その故障回路の外部出力応答を結果とするならば、それらの因果関係の分析法は原因・結果分析法[2-5]と結果・原因分析法[6], [7]の2つが考えられる。

従来提案されている原因・結果法[2-5]は入出力応答表を用いることで高い診断分解能をもつ。しかしながら、LSIが大規模化・複雑化することにより、診断に用いる入出力応答表の作成に多くのメモリと時間が必要になるため原因・結果法を用いて現実的な時間で大規模回路の診断を行うことは困難である。

また、結果・原因分析法は、誤り経路追跡法[6]や故障シミュレーションを用いる診断法[7]で被疑故障箇所を比較的拘束に少数個に絞り込めるため、メモリ使用量と診断実行時間を削減することができる。

本論文では、フルスキャン設計の単一縮退故障を対象としてシングルサイクルキャプチャテストの故障診断によって効果的であった誤り経路追跡法と故障シミュレーションを用いた故障診断法を、マルチサイクルキャプチャテストに適用し、その故障診断分解能を評価する。

2. 故障モデルと諸定義

本論文で取り扱う単一縮退故障モデルについて説明する。縮退故障とは、信号線の値が0または1に固定される故障である。0に固定される故障を0縮退故障、1に固定される故障を1縮退故障という。

本論文で故障診断に用いたテスト方式とテスト生成モデルについて、以下に定義を示す。

（定義1：k サイクルキャプチャテスト）

k サイクルキャプチャテストとは、スキャンテストのキャプチャモード時のサイクル数がkであるスキャンテスト方法である。k $\geq$ 2のとき、マルチサイクルキャプチャテストという。

（定義2：k 時間展開モデル）

k 時間展開モデルとは、フルスキャン設計回路に対して1時刻目のスキャンフリップフロップの出力を擬似外部入力とし、k 時刻目のスキャンフリップフロップのデータ入力を擬似外部出力として、k 時間分時間展開した組合せ回路モデルである。

（定義3：フェイルパターン）

欠陥LSIにテストパターンを印加した際に故障を励起し外部出力、もしくは擬似外部出力で誤りが観測されたテストパターンをフェイルパターンという。そのフェイルパターンによって、誤りが観測された（擬似）外部出力をフェイル（擬似）外部出力という。

（定義4：パスパターン）

欠陥LSIにテストパターンを印加した際に外部出力、もしくは擬似外部出力で期待値が観測されたテストパターンをパスパターンという。また、期待値と一致した値が観測された全ての（疑似）外部出力をパス（疑似）外部出力という。

3. 誤り経路追跡法と故障シミュレーションを用いた故障診断法

本論文で用いた誤り経路追跡法と故障シミュレーションを用いた故障診断法について説明する。

誤り経路追跡法[6]は、診断分解能が欠点ではあるがその課題を解決することで実用的な手法となる。また故障シミュレーションを用いた故障診断[7]は、誤り経路追跡法に比べ多大な計算時間がかかってしまうものの、診断分解能は精度が高く、故障辞書法のような大量のメモリは不要であるという利点がある。

A Evaluation of Fault Diagnosis Method for Full Scan Design Circuits Using
Multi Cycle Capture Test Sets

Hiroki SUWA, Toshinori HOSOKAWA
,and Koji YAMAZAKI

3-1. 誤り経路追跡を用いた故障診断

誤り経路追跡は、各フェイル（疑似）外部出力から追跡された信号線の集合の積集合を求めることで最終的な被疑故障集合を得る。誤り経路追跡法(以下、手法1)は診断時間が高速であるが被疑故障数が多く同定される。図2に誤り経路追跡のアルゴリズムを示す。

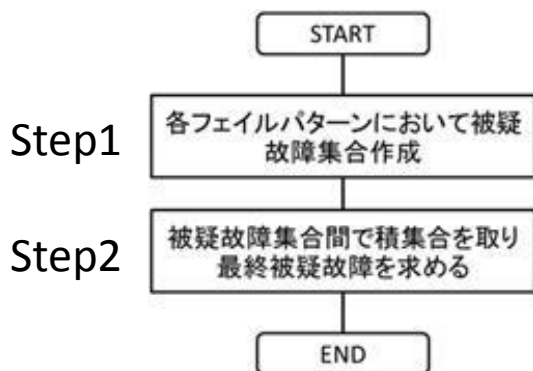


図 1. 誤り経路追跡のアルゴリズム

(step1)

各フェイルパターンについて、誤り経路追跡を行い被疑故障集合を生成する。誤り経路追跡はフェイル外部出力から以下の2つの条件にしたがって行われる。

- ゲートの入力信号線に制御値が含まれている場合、制御値が設定されている信号線を任意に選択し後方追跡を行い、非制御値の入力信号線に対しては後方追跡を行わない。
- ゲートに非制御値しか入力されていない場合、その入力信号線全てに対して後方追跡を行う。

(step2)

(step1)で作成された各フェイルパターンの被疑故障集合間で積集合を取ることで、最終被疑故障集合を求める。

3-2. 故障シミュレーションを用いた故障診断

故障シミュレーションを用いた故障診断はテストパターンをフェイルパターンとパスパターンの二つに分け、フェイルパターンで検出可能な故障の積集合からパスパターンで検出可能な故障の和集合を削除することで最終被疑故障集合が求められる。故障シミュレーションを用いた故障診断は誤り経路追跡よりも診断時間がかかるが、被疑故障をより精度高く求められる診断法である。図3に故障シミュレーションを用いた故障診断のアルゴリズムを示す。

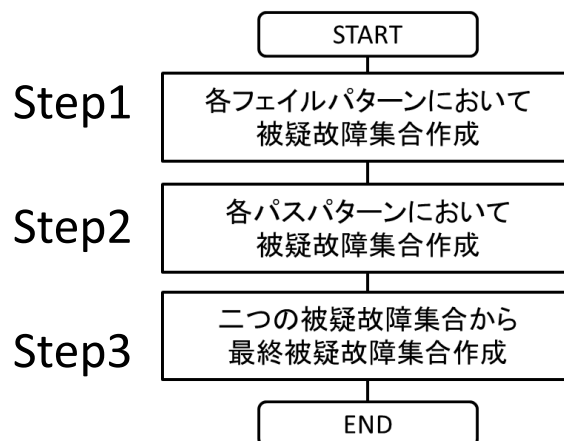


図 2. 故障シミュレーションを用いた故障診断のアルゴリズム

(step1)

各フェイルパターンで検出可能な故障を集合とし、被疑故障集合を作成する。その被疑故障集合間で積集合を取り、フェイルパターンにおける被疑故障集合とする。

(step2)

各パスパターンで検出可能な故障を集合とし、非故障集合を作成する。その非故障集合間で和集合を取り、パスパターンにおける非故障集合とする。

(step3)

被疑故障集合の中で、非故障集合に含まれていない故障を最終被疑故障集合とする。

3-3. 誤り経路追跡と故障シミュレーションを用いた故障診断

誤り経路追跡と故障シミュレーションを用いた故障診断(以下、手法3)は、初めに手法1を用いて被疑故障数を総故障数からある程度絞り込むことで、被疑故障数を減らし、故障シミュレーションにかかる時間を減らす。誤り経路追跡法の後で故障シミュレーションを用いた故障診断を行うことで、故障診断全体の時間を削減しつつ、分解能の精度の高い手法2と等しい分解能を得ることができる。図4に手法3のアルゴリズムを示す。

(step1)

手法1を用いて、初期被疑故障集合を作成する。

(step2)

(step1)で作成した初期被疑故障集合と各フェイルパターンを用いて故障シミュレーションを実行し、被疑故障集合を作成する。

(step3)

(step2)で作成した被疑故障集合と、パスパターンを用いて作成した非故障集合を用いて被疑故障数を削減し、最終被疑故障集合とする。

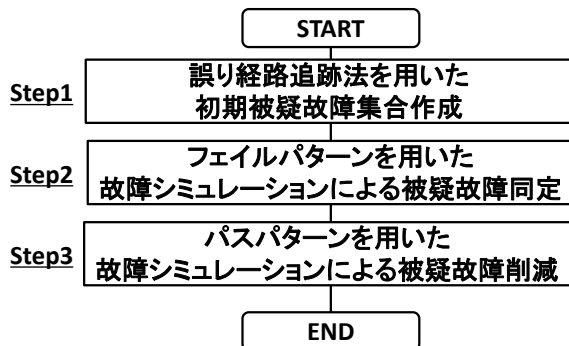


図 3. 誤り経路追跡と故障シミュレーションを用いたアルゴリズム

4. 実験結果

本論文は、誤り経路追跡を用いた故障診断と故障シミュレーションを用いた故障診断をそれぞれ行った。各回路における時間展開数、平均被疑故障数、実行時間を表 1 に示す。

実験環境は、CPU は Core2Duo、メモリは 2GB、OS は Windows7、対象回路は ISCAS'89 ベンチマーク回路である。

表 1 は左から順に対象回路、対象回路における時間展開数、誤り経路追跡を用いた故障診断における平均被疑故障数と診断実行時間、故障シミュレーションを用いた平均被疑故障数と診断実行時間、

表 1 から、 $k=1$ の時から $k=3$ に変化すると実行時間が大きく伸びているのが分かる。それぞれを比較すると手法 1 では s35932 において被疑故障数が約 125 倍以上になっている。これは、手法 1 が単一縮退故障向けの手法であり、時間展開数を増やすことで、多重故障として扱わなければならない、規則に変更を加えなければならない。そのため、被疑故障数が大幅に上昇してしまったと考えられる。

また、手法 2、3 ではそれぞれ被疑故障数の変化は 5 倍程度であるが、実行時間においては 10 倍以上のものがほとんどである。被疑故障数が手法 2、3 で同値であるのは、手法 2 である故障シミュレーションを用いた故障診断法が診断に用いたテストパターンにおける診断分解能の下界であるためで、同じ対象故障に対して平均を求めた場合同値になる。

$k=3$ のときの手法 2 と手法 3 の実行時間を比較すると半分程度となっており、高速化に成功したといえる。しかし、誤り経路追跡による被疑故障集合の削減の影響が少なく、まだ現実的な実行時間ではない。そのため、誤り経路追跡を用いた故障診断を多重故障に向けたアルゴリズムに改善する必要がある。

5. おわりに

本稿では、組合せ回路において有効な故障診断法を順序回路において実装し、実験結果を出して評価した。いくつかの回路で被疑故障を同定し、平均被疑故障数を算出することで、故障シミュレーションを用いた故障診断では、時間がかかるが被疑故障を精度良く絞り込むことができたことが確認できた。しかし、誤り経路追跡を用いた故障診断では、被疑故障数を大幅には削減できないことが確認できた。

ISCAS'89ベンチマーク回路							
対象回路	展開時間数	手法1		手法2		手法3	
		被疑故障数	CPUtime(sec)	被疑故障数	CPUtime(sec)	被疑故障数	CPUtime(sec)
s9234	k=1	25.418	0.183	2.562	0.88	2.562	0.387
s13207	k=1	21.377	0.316	11.7	6.19	11.7	1.078
s15850	k=1	33.75	0.891	2.33	6.71	2.33	1.83
s35932	k=1	8.41	1.361	2.63	10.17	2.63	3.05
s9234	k=3	350.1	1.331	10.4	11.45	10.4	5.6
s13207	k=3	400.2	1.822	22.9	21.78	22.9	13.41
s15850	k=3	721.1	1.701	8.86	29.65	8.86	14.22
s35932	k=3	1078.4	8.565	12.4	50.44	12.4	20.13

表 1.実験結果

今後の課題として、今回用いた 3 つの手法において

被疑故障数が全体的に多すぎるため、故障診断が機能しているとは言い難い。そのため、故障診断のためのテストパターン生成法の考案や、診断対象回路に対してテスト容易化設計を行うなど改善が必要である。そのために、フルスキャン回路での疑似外部出力における値を観測することによってどの程度被疑故障が削減可能であるかを実装、評価したい。また、その疑似外部出力における値をどの程度の割合、または何らかの規則を用いて観測可能にすることで、面積オーバーヘッドをどの程度まで考慮すれば、被疑故障数を組合せ回路と同程度まで削減できるかを実装、評価したい。

参考文献

- 1) H.Y.Chang, E.Manning and G.Metze: "Fault Diagnosis of Digital Systems", John Wiley & Sons, Inc.1970
- 2) M.Abramovici, M.A.Breuer, and A.D.Friedman. "Digital Systems Testing and Testable Design", Computer Science Press., 1990
- 3) V.Boppana and W.K.Fuchs. "Fault dictionary compaction by output sequence removal".In Dig.Int. Conf. on Computer-Aided Design, pp 57-579, 1994
- 4) B.Chess and T.Larrabee,"Creating small fault dictionary". IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, pp18:346-356, 1999
- 5) I.Pomeranz and S.M.Reddy. "On the generation of small dictionaries for fault location". In Dig. Int. Conf. on Computer-Aided Design, pp 272-279,1992
- 6) 山田輝彦 中村芳行. 組合せ回路における単一縮退故障の一診断法.電子情報通信学会論文誌 D-I, Vol.J74-D-I,No.11, pp.774-780,1991
- 7) H.Takahashi, Kwame O. Boateng, "On Diagnosing Multiple Stuck-at Faults Using Multiple and Single Fault Simulation in Combinational Circuit". IEEE,VOL.21,NO.3. pp362-368 ,2002