

コントローラの遷移故障検出率向上のための追加状態遷移挿入法

日大生産工(院) ○飯塚 恭平

日大生産工 細川 利典

日大生産工 山崎 紘史

京産大 吉村正義

1. はじめに

近年, 大規模集積回路(Large Scale Integrated Circuits: LSI)の微細化や高速化, 電源電圧の低下に伴い, 縮退故障モデルのテスト[1]だけでなく, 遷移故障モデル[2]などのタイミング欠陥に対するテストが必要とされている[3]. 製造テストにおいて, 遷移故障を検出できない場合, 潜在的な故障を見逃す恐れがある.

ブロードサイド方式[4]などの高速スキャンテストにおける遷移故障のテスト生成モデルでは, 遷移故障を検出するために初期化状態と故障励起状態の状態遷移が必要である. 初期化状態は遷移故障の初期値が割当てられた状態であり, 故障励起状態は遷移故障が励起されその影響がFFに伝搬する状態である.VLSIの状態遷移はレジスタ転送レベル(Register Transfer Level: RTL)の設計によって決定される. これらの状態遷移の設計により, テスト不可能な故障に分類される遷移故障が決定する. テスト生成はRTL設計後に論理回路に対して実行されるため, 状態遷移によってテスト不可能な故障として分類された結果を修正することは困難である.

この論文では, 対象とする回路がコントローラとデータパスで構成されていることを前提とする. コントローラは有限状態機械(Finite State Machine: FSM)によって表現され, 特定の時刻に実行される操作を決定するために制御信号をデータパスに出力する. このようにコントローラとデータパスの特性が大きく異なるため, 各特性に適した遷移故障検出率を向上させるテスト容易化設計手法(Design for Testability: DFT)が必要である.

コントローラの遷移故障検出率を高める評価指標としてQDT値[5]が提案されている. QDT値はFFのQ端子の遷移故障検出条件に基づいた, コントローラの状態割当ての評価指標である. QDT値は, 高速スキャンテストのブロードサイド方式を想定して計算される. FFのQ端子の遷移故障検出条件がD端子で満たされるとQDT値が1加算される. QDT値が大きいというとは, 回路にFFのQ端子からD端子への活性化経路数が多いことを意味する. したがって, QDT値が大きいと遷移故障検出率が高いと推測できる.

しかしながら, [5]ではQDT値が最大になるよう状態割り当てが設計されていても, 遷移故障検出率が常に最大化されない回路が存在することが報告されている. それらの回路はFFのQ端子からD端子までの経路が多数存在し, 再収斂構造が原因だと考えられている. したがって本論文では, 活性化可能な経路数を表すRTLコントローラの評価尺度であるPSR値を提案する. PSR値を増加させ, 遷移故障検出率を改善するための追加状態遷移挿入法を提案する.

本論文は次のように構成されている. 2章ではQDT値について説明する. 3章ではテスト不能故障を分析するため, 最大のQDT値のコントローラに対して予備実験を行う. 4章ではPSR値と追加状態遷移挿入法を提案する. 5章で実験結果を示す. 最後に6章で

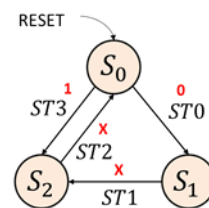


図 1.状態遷移図 例

まとめと今後の課題を示す.

2. QDT 値

QDT 値[5]はコントローラ回路の遷移故障検出率をRTLの状態割当てで評価する値である. QDT 値はQ端子の遷移故障を対象としており, テスト方式は2章2節のブロードサイド方式を想定している. Q端子とは順序回路におけるFFの出力であり, 組合せ回路の入力である. 一方, D端子は順序回路におけるFFの入力, 組合せ回路の出力である.

コントローラ回路は順序回路で, 状態割当てと状態遷移によってFFの値の遷移が決定する. そのためFFのQ端子の遷移故障が初期化及び故障励起は状態割当てと状態遷移で決定する. 言い換えると, FFのQ端子の遷移故障がテスト可能かどうかは状態割当てで決定する. Q端子は

組合せ回路の入力であるため, Q端子は組合せ回路部の入力であるため, Q端子の遷移故障のテストが可能ならば活性化信号線数は多いと考えられ, 内部信号線の遷移故障もテストできる可能性が高いと考えられる.

また, 故障影響の観測点が多くなると, 活性化部分が大きくなる. 観測点はFFで, その入力であるD端子も観測点である. QDT値はQ端子の遷移故障の影響が多くのD端子に伝搬する状態割当てを求める評価指標である.

QDT値の計算方法を説明する. 1つのFFのQ端子の故障が1つのFFのD端子に伝搬するとQDT値に1加算する. FF数を n とすると n^2 通りQ端子・D端子対の組合せがある. 遷移故障モデルは立上り遷移遅延と立下り遷移遅延の2種類があるため, QDTの最大値は $2 \times n^2$ である.

図2にQ端子の立上り遷移故障の場合のQDT値加算条件の例を示し, 説明する. FF数は2ビットでノード内の0,1の数値は状態割当てで, 左からを(FF2, FF1)とする. 状態数は3で S_0, S_1, S_2 の順に遷移するとする. また, 状態変数はそれぞれ $S_0(0,1), S_1(1,1), S_2(1,0)$ と状態割当てされている. 例としてFF2のQ端子に立上り遷移故障を仮定する. この時コントローラの入力に0を設定すると, 正常時は $S_0 \rightarrow S_1((0,1) \rightarrow (1,1))$ の状態遷移を行うが, 故障時は $S_0 \rightarrow S_0((0,1) \rightarrow (0,1))$ の状態遷移を行う. 立上り遷移故障の影響により $0 \rightarrow 1$ の遷移が遅延し, 正常時と故障時に異なる遷移を行うためである. 故障時に, S_0 へ状態遷移をしたことにより, 再び $S_0 \rightarrow S_1((0,1) \rightarrow (1,1))$ の状態遷移を行う. 通常時は上記のタイミングで $S_1 \rightarrow S_2((1,1) \rightarrow (1,0))$ の状態遷移を行う. そのため, 通常時は $S_2(1,0)$, 故障

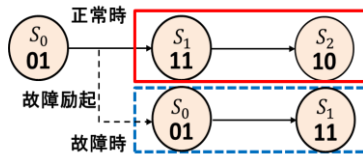


図 2. QDT 値計算例

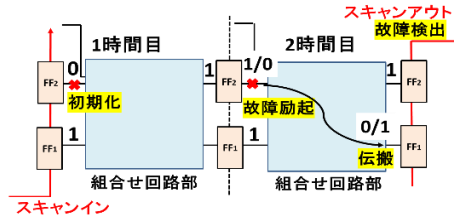


図 3. QDT 値加算条件例

時は $S_1(1,1)$ と到達する状態が異なる．このとき状態変数が FF1 で異なるため，FF1 の D 端子に故障影響が伝搬したとして，QDT 値に 1 を加算する．

図 2 に対応するゲートレベルの回路図を図 3 に示し，説明する．最初の時刻では，シフトイン動作で (FF2,FF1) にそれぞれ (0,1) を割当てる．また，コントローラの入力に 0 を割当てる．2 時刻目は $S_0 \rightarrow S_1$ の通常動作により (1,1) が割当てられる．しかしながら，FF2 の立上り遷移故障により $S_0 \rightarrow S_0$ の状態遷移を行い，(0,1) が割当てられる．この時，故障励起が行われる．2 時刻目の回路応答も通常動作で決定するため，正常時の状態遷移は $S_1 \rightarrow S_2$ となり故障時の状態遷移は $S_0 \rightarrow S_1$ となる．このとき状態変数は正常時 $S_2(1,0)$ と故障時 $S_1(1,1)$ となり FF1 の D 端子に故障影響が伝搬する．その後，シフトアウト動作を行うことで故障を検出する

3. 予備実験結果

本章では，最大の QDT 値をもつ MCNC'91 ベンチマーク回路[7]の遷移故障の原因を解析し，遷移故障のテスト生成によってテスト不能故障として識別された遷移故障に対して故障の分類を行う．予備実験では，以下のように特定されたテスト不能故障に対して，3 種類のテスト生成を連続して実行する．

(解析 1) QDT 値が最大のコントローラである順序回路から組合せ回路部分を抽出する．テスト不能故障と識別された遷移故障に対応する組合せ回路の縮退故障を対象とし，テスト生成を実行する．これらの縮退故障がテスト不可能である場合，冗長故障に分類する．

(解析 2) 解析 1 の回路とテスト可能故障を対象とする．順序回路の外部出力に対応する組合せ回路の外部出力を観測不可とする制約を加えて，テスト生成を行う．これらの縮退故障がテスト不可能である場合，外部出力の観測不能が原因のテスト不能故障に分類する．

(解析 3) 解析 1 の回路と解析 2 のテスト可能故障を対象とする．ブロードサイド方式の 1 時刻目の初期化条件を無視してテスト生成を行う．与えられた縮退故障に対応する遷移故障としてテスト生成し，これらの遷移故障はテスト可能である場合，遷移故障は初期化条件を原因とするテスト不能故障に分類する．

3 つの解析以外のテスト不能故障を 2 時刻目で割当てられた状態を正当化できないと見なし，正当化を原因とするテスト不能故障に分類する．

この予備実験でコントローラの状態割当ては，[5]で

表 1. 対象回路

circuit	#In	#Out	#FFs	#State	#Trans	#QDT	#num	#samp
lion	2	1	2	4	10	8	12	12
mc	3	5	2	4	8	8	10	10
tav	4	4	2	4	4	8	10	10
train4	2	1	2	4	8	8	8	8
dk17	2	3	3	8	23	18	19873	199
ex6	5	8	3	8	31	18	35621	356
shiftreg	1	1	3	8	16	18	8092	81
bbsse	7	7	4	16	56	32	7279	73

表 2. 予備実験結果

circuit	#All	#Redun	#Pounob	#Uninit	#Unjust
lion	7.60	2.10	1.40	1.15	2.95
mc	7.10	2.45	1.65	0.80	2.20
tav	15.60	5.03	3.77	2.11	4.69
train4	8.40	2.60	1.90	1.20	2.70
dk17	18.62	3.43	3.69	2.67	8.83
ex6	68.73	11.21	13.45	10.31	33.76
shiftreg	23.46	8.45	4.41	2.08	8.52
bbsse	180.54	58.92	38.91	18.35	64.36

提案されている最大の QDT 値を持つ回路を使用する．最大 QDT 値の状態割当てが設計されたすべての順序回路は，コントローラごとに合成される．論理合成ツールは Synopsys 社の Design Compiler を使用した．テスト生成ツールは内製の SAT ベースの ATPG を使用し，縮退故障および遷移故障のテスト生成を実行した．

表 1 に評価回路の特性を示す．表 1 の「circuit」は回路名，「#In」は外部入力数を示す．「#Out」は外部出力数，「#FFs」は FF 数である．「#State」は状態数，「#Trans」は状態遷移数，「#QDT」は最大の QDT 値，「#num」は最大 QDT 値をもつ状態割当て数，「#samp」は実験のサンプリング数である．「bbsse」に関しては，10,000 通りの状態割当てがランダム生成され，QDT 値を計算した．8 以上の状態を持つコントローラの場合，最大 QDT 値を持つ状態割当ての 1% が，実験で使用するためにランダムにサンプリングされた．

表 2 に予備実験結果を示す．表 2 の「circuit」は回路名，「#All」はブロードサイド方式で最大の QDT 値を持つすべての回路のテスト不能な遷移故障の平均数である．「#Redun」はテスト不能故障によって発生した冗長故障の平均数，「#Pounob」は外部出力の観測不能を原因とするテスト不能故障の平均数，「#Uninit」は初期化を原因とするテスト不能故障の平均数，「#Unjust」は正当化を原因とするテスト不能故障の平均数である．

表 2 から，テスト不能故障のなかで正当化を原因とするテスト不能故障数の比率は最も高いことが分かる．また，これらの故障は RTL で DFT によって検出できる可能性があるため，正当化によって引き起こされたテスト不能故障を削減する追加状態遷移挿入手法を提案する．

4. 状態遷移挿入手法

本章では，コントローラの遷移故障検出率向上を目的とした状態遷移挿入手法を説明する．言い換えれば，正当化によって発生したテスト不能故障がテスト可能となるように，追加状態遷移を挿入する手法を提案する．4.1 節で，サイド FF を定義し，状態遷移を追加する指標 PSR(Path Sensitizability between Stataas Registers)を提案する．4.2 節に，PSR 値を最大化するための追加状態遷移挿入法のアルゴリズムを提案する．

4.1. PSR 値

文献[6]の QDT 値による評価は FF の Q 端子から D 端子の少なくとも 1 つの経路を活性化できる状態割当ての設計が目的である。そのため、Q 端子から D 端子の経路が複数ある場合、QDT 値に基づいた状態割当てを設計した場合でも遷移故障の検出率は不十分になる可能性がある。このことから、本論文では全経路の活性化を目的としたコントローラの設計を検討する。検討に必要な用語を以下に定義する。

(定義 1: (FF_iQ, FF_jD) のテスト生成) コントローラの FF_j の D 端子で、 FF_i の Q 端子に発生した遷移故障の影響を観測するテスト生成は、 (FF_iQ, FF_jD) 遷移故障のテスト生成と定義する ($1 \leq i, j \leq n$)。ここで、 n は FF 数である。

(定義 2: (FF_iQ, FF_jD) のパス) FF_i の Q 端子から FF_j の D 端子までの経路を (FF_iQ, FF_jD) のパスと定義する。

(定義 3: サイド FF) (FF_iQ, FF_jD) 遷移故障のテスト生成時に、 FF_i 以外の FF をサイド FF と定義する。

遷移故障のテスト生成では、故障の伝搬は 2 時刻目で実行される。言い換えると、故障伝搬経路は 2 時刻目で活性化される。2 時刻目の活性化経路は、サイド FF の値で決定する。

図 4 に 2 時刻目のサイド FF の例を示す。図 4 では、 (FF_2Q, FF_1D) 立上り遷移故障のテスト生成であるため、 FF_1 はサイド FF である。2 時刻目の FF_1 の値が 0 の場合、実線経路が活性化して故障の影響が伝搬し、 FF_1 の値が 1 の場合、破線の経路が活性化して故障の影響が伝搬する。サイド FF の値の組合せ網羅するように状態遷移を追加することにより任意の (FF_iQ, FF_jD) のパスを活性化することができ、遷移故障検出率が向上する可能性があると考えられる。

本論文では、QDT 値[5]が最大になるように状態割当てされたコントローラを対象としたサイド FF の値の組合せ数を増やす手段として、PSR 値を提案する。PSR 値が大きいと、活性化可能な (FF_iQ, FF_jD) のパス数が増加する。以下に、PSR 値を定義する。

(定義 4: PSR 値) 各 (FF_iQ, FF_jD) 立上り(立下り)遷移故障のテスト生成において、2 時刻目で設定可能なサイド FF の

値の組合せ数は PSR として定義される。各 (FF_iQ, FF_jD) 遷移故障のテスト生成における PSR の合計が PSR 値として定義される。最大の PSR 値は $2 \times n^2 \times 2^{n-1}$ である。ここで n は FF 数である。

図 4 では、 (FF_2Q, FF_1D) 遷移故障のテスト生成の数は 8 である。各 (FF_2Q, FF_1D) 遷移故障のテスト生成におけるサイド FF の値の組合せ数は 2 である。したがって、PSR 値の最大は 16 となる。

4.2. 状態遷移挿入アルゴリズム

PSR 値を最大化するために、コントローラのテスト時のみ動作する追加状態遷移を挿入する手法を提案する。図 5 に提案手法のアルゴリズムを示す。アルゴリ

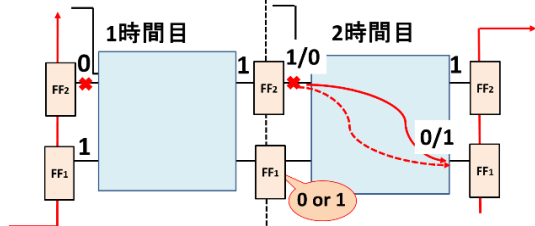


図 4. サイド FF による活性化経路の変化

```

1. Input : RTL controller  $C$ 
2. Output : Augmented RTL controller  $AC$ 
3. Augment_Controller( $C$ ) {
4.    $C = \text{QDT\_Based\_State\_Assignment}(C)$ ;
5.    $AST = \text{PSR\_Based\_Additional\_State\_Transition\_Search}(C)$ ;
6.    $AC = \text{Controller\_Augmentation}(AC, AST)$ ;
7.   return( $AC$ );
8. }

```

図 5. 追加状態遷移挿入法アルゴリズム

ズムの入力は RTL コントローラ C であり、出力は状態遷移が追加された拡張コントローラ AC である。

はじめに、 C の QDT 値が最大となる状態割当てを行い、 C を更新する(行 4)。次に PSR 値を最大化するために必要な追加の状態遷移集合 AST を探索する(行 5)。 AST の状態遷移が C に挿入され、 AC が生成される(行 6)。コントローラの拡張により、機能動作時とテスト時の状態遷移を識別する必要があるため、外部入力を追加する。追加の状態遷移数に応じて、追加の外部入力数は決定する。最後に AC がアルゴリズムの出力として返される。(行 7)。

MCNC'91 ベンチマーク回路[7]の mc での提案手法の適用例を説明する。図 6 に mc 回路の RTL コントローラを示す。mc は状態数 4、状態遷移数 8 の回路である。状態遷移の X(ドントケア)を含む論理値は mc の外部入力値を示す。また、mc の外部出力値は省略する。mc のゲートレベルの回路では FF 数は 2 である。図 6 は QDT 値が最大になる状態割当てが設計されている。状態割当てはそれぞれ $S_0(00)$, $S_1(01)$, $S_2(11)$, $S_3(10)$ である。表 3 に (FF_2Q, FF_1D) 遷移故障のテスト生成の情報示す。表 3 の「テスト生成」は対象の (FF_2Q, FF_1D) 遷移故障である。「2 連続 ST」はブロードサイド方式のテスト生成使用される 2 つの連続した状態遷移を示し、「故障時 ST」は故障時の 2 時刻目の状態遷移を示す。「サイド FF 値」は 2 時刻目のサイド FF の値を示す。表 3 に示す通り、状態遷移追加前の mc 回路の PSR 値は実現可能な 2 連続 ST 数の 8 である。図 6 の有限状態機械に対応する回路の (FF_2Q, FF_1D) 立上り遷移故障のテスト生成では、2 時刻目のサイド FF (FF_2) の値は 0 である。2 連続 ST の 1 時刻目である 00 は、シフトイン動作により、スキャン FF (FF_2 および FF_1) に設定される。そして、外部入力に 111 が適用されると 01 が状態遷移によってスキャン FF に設定される。故障時はシフトイン動作と外部入力の適用後、 FF_1 の Q 端子の立上り遷移故障によりスキャン FF に 00 が設定される。続いて、外部入力に 000 が適用され、状態遷移によって機能動作時は 00、故障時は 01 がスキャン FF に設定される。機能動作時と故障時で FF_1 の D 端子の値が異なるため、故障影響がシフトアウト動作により検出される。

遷移故障のテスト生成において、2 時刻目のみのテスト生成モデルは縮退故障モデルと同じである。例えば、 (FF_1Q, FF_1D) 立上り遷移故障のテスト生成では、表 3 に示す通り 2 時刻目でサイド FF (FF_2) に 1 を設定することは不可能である。立上り遷移故障の 2 時刻目の故障モデルは 0 縮退故障に対応する。よって、 (FF_1Q, FF_1D) 0 縮退故障のテスト生成を検討する。11 がシフトイン動作によりスキャン FF (FF_2, FF_1) に設定される。外部入力に 100 が適用され、11 が状態遷移によってスキャン FF に設定される。故障時では、シフトイン動作後 FF_1 の Q 端子の 0 縮退故障のため、10

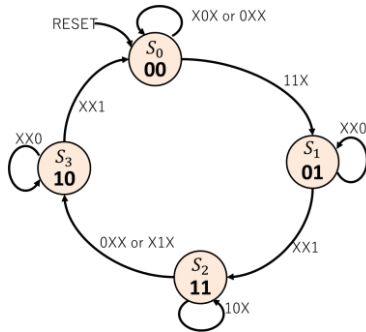


図 6. mc の状態遷移図

表 3. mc の PSR 値

テスト生成	(FF ₁ Q, FF ₁ D) 立上り	(FF ₁ Q, FF ₁ D) 立下り	(FF ₂ Q, FF ₂ D) 立上り	(FF ₂ Q, FF ₂ D) 立下り
2連続 ST	00→01→01	11→10→10	00→01→11	11→10→00
故障時 ST	00→00	11→11	00→01	11→10
サイドFF値	(FF ₁)=(0)	(FF ₁)=(1)	(FF ₂)=(0)	(FF ₂)=(1)
テスト生成	(FF ₂ Q, FF ₂ D) 立上り	(FF ₂ Q, FF ₂ D) 立下り	(FF ₁ Q, FF ₁ D) 立上り	(FF ₁ Q, FF ₁ D) 立下り
2連続 ST	01→11→10	10→00→01	01→11→11	10→00→00
故障時 ST	01→11	10→00	01→01	10→10
サイドFF値	(FF ₂)=(1)	(FF ₂)=(0)	(FF ₁)=(1)	(FF ₁)=(0)

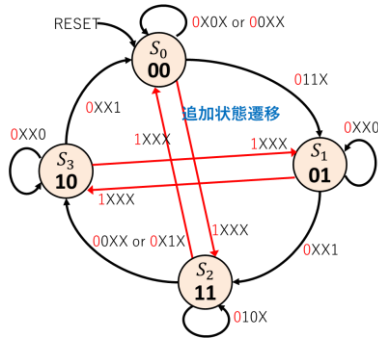


図 7. mc の状態遷移図 (拡張後)

表 4. 拡張後の mc の PSR 値

テスト生成	(FF ₁ Q, FF ₁ D) 立上り	(FF ₁ Q, FF ₁ D) 立下り	(FF ₂ Q, FF ₂ D) 立上り	(FF ₂ Q, FF ₂ D) 立下り
2連続 ST	00→11→11	11→00→00	00→11→10	11→00→01
故障時 ST	10→10	01→01	10→00	01→11
サイドFF値	(FF ₂)=(1)	(FF ₂)=(0)	(FF ₂)=(1)	(FF ₂)=(0)
テスト生成	(FF ₂ Q, FF ₂ D) 立上り	(FF ₂ Q, FF ₂ D) 立下り	(FF ₁ Q, FF ₁ D) 立上り	(FF ₁ Q, FF ₁ D) 立下り
2連続 ST	01→10→00	10→01→11	01→10→10	10→01→01
故障時 ST	00→01	11→10	00→00	11→11
サイドFF値	(FF ₁)=(0)	(FF ₁)=(1)	(FF ₂)=(0)	(FF ₂)=(1)

がスキャン FF に設定される。続いて、外部入力に 100 が適用され、状態遷移によって機能動作時は 11、故障時は 10 がスキャン FF に設定される。機能動作時と故障時で FF1 の D 端子の値が異なるため、故障影響がシフトアウト動作により検出される。このとき、サイド FF (FF2) の値は 1 である。状態 $S_0(00)$ または $S_3(10)$ から $S_2(11)$ への状態遷移が挿入される場合、 (FF_1Q, FF_1D) 立上り遷移故障のテスト生成において、2 時刻目のサイド FF (FF2) に 1 を設定可能である。その結果、PSR 値が増加する。図 7 の通り、 S_0 から S_2 、 S_2 から S_0 、 S_1 から S_3 、 S_3 から S_1 の 4 つの状態遷移が、PSR 値が最大値(16)になるようにコントローラに挿入される。表 4 に、拡張コントローラによって追加された (FF_1Q, FF_1D) 遷移故障のテスト生成情報を示す。

5. 実験結果

表 5 に提案手法の ATPG の結果を示す。なお、対象

表 5. 実験結果

circuit	DFT'19[8]		proposed				
	FC(%)	#PSR	FC(%)	#PSR	#Add_In	#Add_Trans	Overhead(%)
lion	88.73	14	89.25	16	1	1.60	15.32
mc	86.34	12	94.82	16	1	2.00	21.58
tav	69.53	12	95.16	16	1	2.00	21.46
train4	79.81	12	92.31	16	1	2.00	22.78
dk17	81.65	54	90.34	72	1	8.31	99.76
ex6	77.24	58	81.48	72	1	6.57	87.68
shiftreg	80.92	56	86.33	72	2	7.88	91.21
bbsse	63.17	248	78.24	256	1	16.37	113.26

回路、実験環境は 3 章と同様である。表 5 において、「circuit」は回路名、「DFT'19」は[5]の状態割当て手法が適用された回路の実験結果を示し、「proposed」は状態割当て[5]と提手法が適用した実験結果である。

「FC」は平均の遷移故障検出率を示し、「#PSR」は平均 PSR 値、「#Add_in」は追加した外部入力数の平均、「Add Trans」は追加した状態遷移数の平均を示す。

「Overhead」は元のコントローラの面積に対する DFT 適用後の追加面積の平均比率である。

表 5 に示す通り、PSR 値を最大化するために状態遷移を追加することにより、すべての回路で遷移故障検出率が平均 10%向上できた。

6. 結論

本論文では、コントローラの遷移故障検出率のテスト可能性尺度として PSR 値を提案し、PSR 値を最大化するための追加状態遷移挿入手法を提案した。MCNC'91 ベンチマーク回路について、遷移故障検出率が 10%改善された。また、最大の面積オーバーヘッドは bbsse 回路で 113%となった。

今後の課題として、無効状態を持つコントローラの PSR 値を最大化するための状態遷移設計手法の提案があげられる。

参考文献

- [1] H. Fujiwara, Logic Testing and Design for Testability, The MIT Press, 1985.
- [2] A. Krstic, and K.-T. Cheng, Delay Fault Testing for VLSI Circuit, Kluwer Academic Publishers, Boston, 1998.
- [3] Y. Sato, S. Hamada, T. Maeda, A. Takatori, Y. Nozu-yama and S. Kajihara, "Invisible Delay Quality - SDQM Model Lights Up What Could Not Be Seen," Proc. Int. Test Conf., Paper 47.1, 2005.
- [4] J. Savir and S. Patil, "On Broad-side Delay Test," Proc. VLSI TestSymp., pp. 284-290, 1994.
- [5] M. Yoshimura, Y. Takeuchi, H. Yamazaki, and T. Hosokawa "AState Assignment Method to Improve Transition Fault Coverage for Controllers," The 32nd IEEE Int. Symp. on Defect and FaultTolerance in VLSI and Nanotechnology Systems, 2019.
- [6] K.-T. Cheng, S. Devadas, and K. Keutzer, "A Partial Enhanced-Scan Approach to Robust Delay-Fault Test Generation for Sequential Circuits," Proc. Test Conf., pp.403-410, 1991.
- [7] S. Yang, Logic Synthesis and Optimization Bench- marks, Version 3.0, Tech. Report, Microelectronics Center of North Carolina, 1991.