

モジュールセット化による低面積組込み自己テストの実現

A BIST Method with Low Hardware-Overhead by Using Sets under Test

岩田 大志[†]青山 瑠美[‡]山口 賢一[§]

Hiroshi Iwata

Rumi Aoyama

Ken'ichi Yamaguchi

1. はじめに

VLSIに対する多様なニーズに伴い、VLSIの複雑さは増大している。そして、VLSIの複雑化に伴い、VLSIのテスト費用の増加が問題視されている。そのため、VLSIに対するテスト容易化設計 (Design For Testability, DFT) は必須となっている。テスト費用を削減するテスト手法として、ノンスキャン組込み自己テスト (Non-Scan Built-In Self Test, Non-Scan BIST) が注目されている。これまでに、Non-Scan BISTを実現するDFTとして階層BIST[1][2]が提案されている。しかし、階層BISTに基づくDFTには、多大な回路面積の増加を伴うという問題点がある。そこで本稿では、回路面積の増加量を抑制可能な、階層BISTに基づくDFTを提案する。

2. 関連手法：階層BIST

本稿で扱うテスト対象は、コントローラとデータパスからなるレジスタ転送レベル回路のデータパス部とする。階層BISTでは、外部入出力にそれぞれ擬似ランダムパターン発生回路 (Pseudo Random Pattern Generator, PRPG) と多入力シグネチャレジスタ (Multiple Input Signature Register, MISR) を配置する。そして、データパスに対して適切な制御信号 (テストプラン) を付加することでテストを行う。

階層BISTに基づくテスト手法[1][2]は、テストプランを付加することにより、PRPGからテスト対象までの経路 (制御経路) とテスト対象からMISRまでの経路 (観測経路) を生成しテストを行う。制御経路と観測経路に共通部分がある場合、テスト対象に対して連続したランダムパターンの付加が行えない。そのため、テスト対象ごとに単一のテストプランを付加することで共通部分を持たない経路を生成する。テストプランの付加だけでは経路生成を行うことができない場合、マルチプレクサ (MUX) などを追加するDFTを行うことで経路生成を行う。[1]や[2]の手法では組合せ回路要素ごとに経路生成を行うため、経路生成に伴う面積増加が多く発生するという問題点がある。そこで、本稿では経路生成を行う対象を限定することで、故障検出率を悪化させることなく面積増加を抑制するデータパスの構造と、その構造を満たすために行うDFT手法について提案する。

3. 回路面積増加を抑制するアイデアの提案

DFTによる回路面積増加を抑制するために、本稿では、以下に示すようなアイデアを提案する。

1. MUXのように、比較的小規模でテストが容易な回路要素に対する経路生成を行わない
2. 演算器のように、比較的大規模でテストが困難な回路要素

のテスト時にテスト容易な回路要素のテストを同時に行う

以上のアイデアは、演算器のテスト中に、制御経路、観測経路に含まれる故障が同時に検出可能であるということに基づいている。まず、アイデア1に従い、テスト対象データパス上の組合せ回路要素に対し、経路生成を行う回路要素集合CTMS (Complex Testable Module Set) と、経路生成を行わない回路要素集合TTMS (Trivial Testable Module Set) の2種類に分類する。CTMSは演算器を要素として含み、TTMSはその他の回路要素を要素として含む。

次に、アイデア2に従い、1個のCTMSに属する回路要素と複数のTTMSに属する回路要素をまとめてセットとして扱う。セットには、複数の回路要素が含まれているが、1つの経路生成対象となる。複数の回路要素をセットとして扱うことにより、生成する経路数を削減できるため、経路生成に伴う回路面積の増加を抑制できる。4章では、アイデア1, 2に基づくセットとその満たすべき性質 (セット可検査性) とセット可検査性に基づくDFTについて示す。5章では、データパス中のセットがテスト可能である性質 (分割集合可検査性) と分割集合可検査性に基づくDFTについて示す。

4. 複数の回路要素からなるセットに対する可検査性

本章では複数の回路要素から成るセットごとにテストを行うことによって、高故障検出率を実現するDFTアルゴリズムを示す。また、本稿で提案するDFTアルゴリズムは、以下に定義するRTLデータパスと同一視可能な有向グラフ、データパスグラフDPGを対象とする。

定義1 データパスグラフ

$$\begin{aligned}
 DPG &= (V, E) \\
 V &= V_{op} \cup V_r \cup V_m \cup V_{pio} \quad (V_{pio} = V_{PI} \cup V_{PO}) \\
 E &= E_{in}(v_o) \cup E_{out}(v_o) \cup E_{ot} \\
 E_{in}(v_o) &= (V_r \cup V_m \cup V_{pio}) \times (v_o) \\
 E_{out}(v_o) &= (v_o) \times (V_r \cup V_m \cup V_{pio}) \\
 E_{ot} &= (V_r \cup V_m \cup V_{pio}) \times (V_r \cup V_m \cup V_{pio}) \quad \blacksquare
 \end{aligned}
 \tag{1}$$

$v_o \in V_{op}$ は演算器、 V_r はレジスタ、 V_m はMUX、 V_{PI} は外部入力、 V_{PO} は外部出力の集合である。

4.1. 高故障検出率を実現するセット可検査性の定義

3章で示したセットは、データパスグラフ上で、以下のよう

に定義する。

$$\begin{aligned}
 \text{定義2 セット (Set)} \\
 Set &= (V_s, E_s) \\
 V_s &= V_{seti} \cup V_{el} \cup V_{seto} \\
 V_{el} &= V_{mux} \cup V_{reg} \cup v_o \quad (v_o \in V_{op}) \\
 E_s &= (V_{seti} \times V_{el}) \cup (V_{el} \times V_{el}) \cup (V_{el} \times V_{seto}) \quad \blacksquare
 \end{aligned}
 \tag{2}$$

[†]奈良工業高等専門学校 電子情報工学専攻

[‡]奈良先端科学技術大学院大学 情報科学研究科 情報処理学専攻

[§]奈良工業高等専門学校 情報工学科

V_{seti} はセット入力端子であり, V_{seto} はセット出力端子である. セットはテスト対象であるため, セット入出力端子に対して経路生成を行う必要がある. この経路生成を行う端子を経路生成端子として定義する.

定義 3 経路生成端子

経路生成入力端子 全ての V_{seti} に対し辺を持つ v_{so} を考え, 全ての辺の容量を 1 としたとき, v_{so} から vo へ, vo の入力辺数を流量とするフロー問題を解いたときに通過する V_{seti}
 経路生成出力端子 全ての V_{seto} からの辺を持つ V_{si} を考え, 全ての辺の容量を 1 としたとき, vo から V_{si} へ, vo の出力辺数を流量とするフロー問題を解いたときに通過する V_{seto}

経路生成端子の決定例を図 1 に示す.

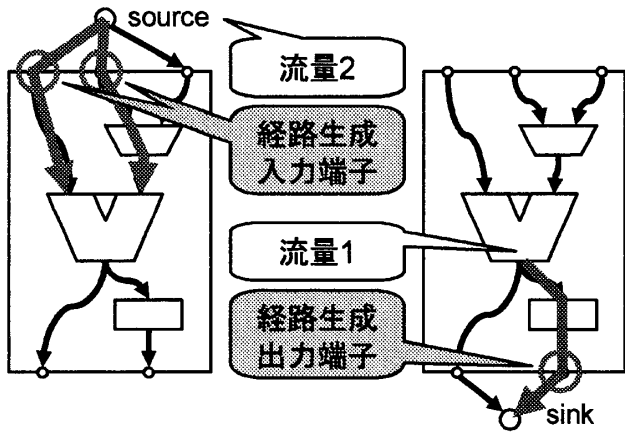


図 1: 経路生成入力端子と経路生成出力端子の決定例

この経路生成端子の定義を利用して, セット可検査性を定義する. セット可検査性は, 各セットに対して故障検出率を保証するものであり, データパス上ですべてのセットがセット可検査性を満足すればそのデータパスのテストにおいて, 十分な故障検出率を保証できる.

定義 4 セット可検査性 (Set Testability)

すべての経路生成端子に対し, 共通部分を持たない制御経路と観測経路が生成可能である回路構造を持つ

また, セット入力端子を始点, セット出力端子を終点とするセット内の経路には, 演算器を含まない経路が存在することがある. このような経路をセットスルーと定義する.

定義 5 セットスルー (Set Thru)

セット V_s において, 以下の経路 P_{th} を持つ場合, V_s はセットスルー可能であると定義する.

$$P_{th} = (v_0, v_1, \dots, v_\alpha, \dots, v_{i-1}, v_i) \quad (3)$$

$$\begin{cases} v_0 \in V_{seti} \\ v_\alpha \in V_{el} - v_0 \quad (1 \leq \alpha \leq i-1) \\ v_i \in V_{seto} \end{cases}$$

セットがセットスルー可能であるとき, セット入力からセット出力への伝搬が可能な経路が存在する.

4.2. セット可検査性を満たす DFT アルゴリズム

経路生成端子に対して行う経路生成には, 階層 BIST と同様にテストプランを与えることによって既存の経路を利用する. しかし, テストプランの付加による経路生成ができない場合には, 経路生成を行う DFT が必要となる. 本稿では, テストプランの付加による経路生成ができない経路生成端子に対して, PRPG や MISR の機能を持つテストレジスタを利用することで経路生成を行うことを考える. いくつかのレジスタをテストレジスタへ設計変更すると, すべての経路生成端子に対して経路生成が可能となる場合, それらのレジスタをテストレジスタ候補 (Test Register Candidacy, TRC) として扱う¹. TRC は MISR や PRPG の機能を持つレジスタとして扱い, TRC とテストプランの付加による経路生成によってすべての経路生成端子に対して経路生成を行う.

セット可検査性を満足するデータパスを得るために, 始点 (Source) を v_{so} , 終点 (Sink) を v_{si} とする最小費用流問題を適用する DFT アルゴリズムを提案する. そのため, 最小費用流問題を適用するグラフ, 演算器遮断データパスグラフ (Operator Cut DPG, OCG) を定義する. OCG は, DPG に対して次の変更を行ったグラフである.

- CTMS に属する回路要素である $vo \in V_{op}$ を除く V_{op} と, V_{op} に接続する辺を取り除く
- v_{so} と V_{PI} , V_{PO} と v_{si} を辺 E_{sio} によって接続する
- レジスタは費用を伴う PRPG, MISR としても扱うため, v_{so} と $vr_\beta \in V_r$ を辺 $e_{ri\beta}$ によって接続し, vr_β と v_{so} を辺 $e_{ro\beta}$ によって接続する
- PRPG, MISR として扱うための費用は c_β とする
- Source からの流出量と Sink への流入量を等しくするため, v_{so} と vo を辺 e_{soi} によって接続し, vo と v_{so} を辺 e_{soo} によって接続する

DPG での頂点集合が, $V_{op} = \{vo_0, vo_1, \dots, vo_\alpha, \dots, vo_n\}$, $V_r = \{vr_0, vr_1, \dots, vr_\beta, \dots, vr_m\}$ のとき, OCG は各 vo_α について存在する. そこで vo_α を含む g_α を式 (4) に定義する.

定義 6 演算器遮断データパスグラフ g_α

$$g_\alpha = (V_\alpha, E_\alpha)$$

$$V_\alpha = (\alpha v_0, \alpha v_1, \dots, \alpha v_\gamma, \dots, \alpha v_o)$$

$$\alpha v_\gamma \in \{ \alpha v_r, \alpha v_m, \alpha v_p, \alpha v_{so}, \alpha v_{si}, v_{o\alpha} \}$$

$$E_\alpha = (\alpha e_0, \alpha e_1, \dots, \alpha e_\delta, \dots, \alpha e_p) \quad (4)$$

$$\alpha e_\delta \in \left\{ \begin{array}{l} \alpha e_{ot}, \alpha e_{oi}, \alpha e_{oo}, \alpha e_{sio}, \\ \alpha e_{ri\beta}, \alpha e_{ro\beta}, \alpha e_{soi}, \alpha e_{soo} \end{array} \right\}$$

$$e_{sio} \in E_{sio} = (v_{so} \times V_{PI}) \cup (V_{PO} \times v_{si})$$

$$e_{ri\beta} = v_{so} \times vr_\beta, \quad e_{ro\beta} = vr_\beta \times v_{si}$$

$$e_{soi} = v_{so} \times v_o, \quad e_{soo} = v_o \times v_{si}$$

ただし, $v_r \in V_r$, $v_m \in V_m$, $v_p \in V_{pio}$ であり, $e_{ot} \in E_{ot}$, $e_{oi} \in E_{in}(vo_\alpha)$, $e_{oo} \in E_{out}(vo_\alpha)$ である.

この OCG を対象とする最小費用流問題を線形計画問題として定式化を行う. vo_α の制御経路と観測経路は同時に考える必要があるが, vo_α を通過するという最小費用流問題の定

¹経路生成が不可能な経路生成端子に対しては, バイパスレジスタを追加する前処理を行い経路生成を可能とする.

式化はできない。そこで、制御経路用の $OCG, g_{\alpha.c}$ と、観測経路用の $OCG, g_{\alpha.o}$ を作成し、制御経路と観測経路が共通部分を持たない制限式を加える。 $g_{\alpha.c}$ の流量は $|Ein(vo_{\alpha})|$ であり、 $\alpha.c.e_{soi} = 0, \alpha.c.e_{soo} = |Ein(vo_{\alpha})|$ とすることで vo_{α} を通過させる。同様に、 $g_{\alpha.o}$ の流量は $|Eout(vo_{\alpha})|$ であり、 $\alpha.o.e_{soi} = |Eout(vo_{\alpha})|, \alpha.o.e_{soo} = 0$ とする。また、他の辺 e の容量は $1 (e \leq 1)$ とする。これらの制限式を式 (5) に、共通部分を持たない制限式を式 (6) に、TRC として扱う費用は式 (7) で定式化する。

Subject to:

$$G = \{g_{0.c}, g_{0.o}, g_{1.c}, g_{1.o}, \dots, g_{\alpha.c}, g_{\alpha.o}, \dots, g_{n.c}, g_{n.o}\} \quad (5)$$

$$\begin{cases} \alpha.c.e_{soi} = 0, & \alpha.c.e_{soo} = |Ein(vo_{\alpha})| \\ \alpha.o.e_{soi} = |Eout(vo_{\alpha})|, & \alpha.o.e_{soo} = 0 \end{cases}$$

$$\alpha.c.e_{\delta} + \alpha.o.e_{\delta} \leq 1 \quad (if \ \alpha.e_{\delta} \neq \alpha.e_{soi} \wedge \alpha.e_{\delta} \neq \alpha.e_{soo}) \quad (6)$$

$$C = \{c_0, c_1 \dots c_{\beta}, \dots, c_m\} \quad (7)$$

$$c_{\beta} \geq B \cdot (\alpha.c.e_{ri\beta} \vee \alpha.o.e_{ro\beta})$$

また、TRC を使用する費用 B は $B \gg 1$ を満たす適当な整数値であり、本稿ではレジスタをテストレジスタに変更する際に必要なゲート数とした。これらの制限式 (5), (6), (7) を満たす最適化目標を式 (8) に定義する。

minimize:

$$\sum_{\beta=0}^m \left(c_{\beta} + \sum_{\delta=0}^n (\alpha.c.e_{ri\beta} + \alpha.o.e_{ro\beta}) \right) \quad (8)$$

c_{β} は既存のレジスタを TRC として扱う費用の総和、 $\alpha.c.e_{ri\beta}, \alpha.o.e_{ro\beta}$ は TRC を利用するコストである。 $c_{\beta} \gg e$ であるため、 e を使用可能であるにもかかわらず使用せず、新しく c_{β} を使用する解を除外する。

この線形計画問題を解くことにより、各 OCG においてその演算器 vo_{α} のための経路 ($\alpha.e_{\delta} = 1$ となる辺) と、TRC となるレジスタ ($e_{ri\beta} \vee e_{ro\beta} = 1$ となる $e_{r\beta}$) が得られる。また、図 2.a のように、OCG 集合 G を重ね合わせることで OCG 固有の経路は部分、または全体を観察することができる (図 2.b)。観察可能な経路に含まれる頂点 $V_r \cup V_m \cup vo_{\alpha}$ をその vo_{α} のセットとする。どの経路にも含まれない頂点は出力先の頂点が含まれるセットに含め、含めることができない場合には入力元のセットに含める。このとき、セットの境界と経路の交点が経路生成端子となる。

これらにより、セット化を行ったセット可検査性を満足するセットグラフ (Set Graph, SG) を式 (9) に定義する。

定義 7 セットグラフ SG

$$SG = \{V_{sg}, E_{sg}\}$$

$$V_{sg} = (V_s, V_t, V_{pi}, V_{po}, v_{so}, v_{si})$$

$$s_{\omega} \in V_s = V_{seti} \cup V_{set} \cup V_{seto} \quad (v_{set} = \{V_{el}\})$$

$$PI(s_{\omega}) = v_{pi} \in V_{pi}, \quad PO(s_{\omega}) = v_{po} \in V_{po}$$

$$TRI(s_{\omega}) = v_t \in V_t, \quad TRO(s_{\omega}) = v_t \quad (9)$$

$$E_{sg} = (E_{data}, E_{st})$$

$$E_{data} = V \times V$$

$$E_{st} = V_{seti} \times V_{seto} \quad (if \ exists \ P_{th})$$

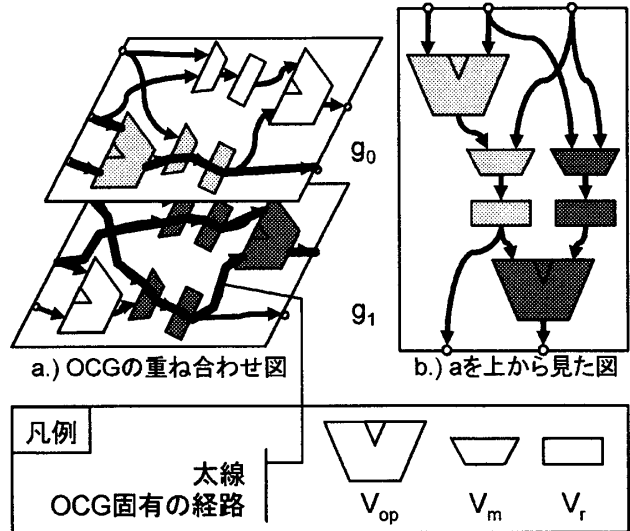


図 2: OCG の重ね合わせとセット化

ただし、 V_s はセット集合であり、 V_t は TRC 集合である。 V_s の各要素 s_{ω} は、セット入力端子、セットの構成要素集合、セット出力端子を要素として持つ。また、 s_{ω} は使用する外部入出力、テストレジスタの情報を PI, PO, TRI, TRO の各写像により保持する。 E_{data} は各頂点を結ぶ辺であり、 E_{st} はセットスルーである。

5. セット可検査性を満たすグラフに対する可検査性

セット可検査性を満たす SG において、TRC をテストレジスタに変更することにより、データパス全体に対して高故障検出率を実現することができる。本章では、故障検出率の保証ができるグラフの構造を分割集合可検査性と定義し、さらにテストレジスタへの設計変更に伴う面積増加率を抑制する DFT アルゴリズムを提案する。

定義 8 分割集合可検査性

SG の TRC に経路生成が存在し、TRC 無しにセット可検査性を満足する回路構造

分割集合可検査性を満たす DFT アルゴリズムは以下の 2 ステップからなる。

1. 外部入出力の資源割り当て
2. 割り当てた資源を利用するための設計変更

テストレジスタへの設計変更に伴う面積増加率は多大であるため、既存の PRPG や MISR を最大限利用する。そのため外部入出力を資源として扱い、TRC に割り当てる。テストレジスタは費用を伴う資源として扱う。そして、割り当てた資源を利用するために、割り当てた資源と TRC を接続する (TRC に設計変更する) DFT を行う。

5.1. TRC への外部入出力、テストレジスタへの割り当て

セットが利用する外部入出力を決定し、利用できない場合にはテストレジスタを利用する。そこで、TRC を利用するセットと外部入出力の資源割り当て問題を考える。

$$V_s = \{s_0, s_1, \dots, s_{\alpha}, \dots, s_n\},$$

$$V_{pi} = \{pi_0, pi_1, \dots, pi_{\beta}, \dots, pi_m\},$$

$$V_t = \{tr_0, tr_1, \dots, tr_{\gamma}, \dots, tr_l\} \quad (10)$$

式 (10) の条件下において、入力割り当て問題を線形計画問題として定式化する。

$$\sum_{\beta=0}^m p_{i\beta} + tr_{\gamma} \geq 1$$

$$\gamma p_{i\beta} = 0 \begin{cases} PI(s_{\alpha}) = p_{i\beta} & \text{for All } \beta \\ TRI(s_{\alpha}) = tr_{\gamma} & \text{for All } \gamma \end{cases} \quad (11)$$

制限式 (11) は、TRC に外部入力やテストレジスタを割り当て、同一セットの他の経路生成入力端子に対して同一の外部入力を利用することを防ぐ。しかし、1つのセットで複数の TRC を使用した際に外部入力を同時に利用する解を算出する可能性があるため、共有テストレジスタ候補対をすべて列挙し新しい制約を加える。

定義 9 共有テストレジスタ候補対 (*Shared TRC Pair, STP*) セットが使用する TRC が複数ある場合、それらの組合せにより得られる対をすべて列挙したもの

例えば、 s_0 が tr_0 と tr_1 を使用し、 s_1 が tr_1 と tr_2 を使用する場合、 $STP = \{\{tr_0, tr_1\}, \{tr_1, tr_2\}, \{tr_0, tr_2\}\}$ となる。STP で使用される外部入力は共有され、一方のセットで使用されている外部入力は他方のセットで使用できない。この制約を満たす制限式を式 (12) に示す。

$$STP = \{st_0, st_1, \dots, st_s, \dots, st_o\}$$

$$st_s = \{tr_p, tr_q\}$$

$$\left. \begin{matrix} p p_{i\beta} + tr_p \geq q p_{i\beta} \\ q p_{i\beta} + tr_q \geq p p_{i\beta} \end{matrix} \right\} \text{for All } \beta \quad (12)$$

制限式 (11), (12) を満たす最適化目標を式 (13) に示す。

minimize:

$$\sum_{\gamma=0}^l (B \cdot tr_{\gamma}) \quad (13)$$

出力も同様に行い、TRC への外部入力、外部出力、テストレジスタの割り当てを行う。

5.2. TRC への経路生成とテストレジスタへの設計変更

TRC に対して外部入出力を割り当てた場合、図3に示すように設計変更を行う。テストレジスタに割り当てた場合、同一セットで PRPG と MISR を同時に利用する TRC は CBILBO、どちらか一方のみを利用する TRC は BILBO に設計変更する。

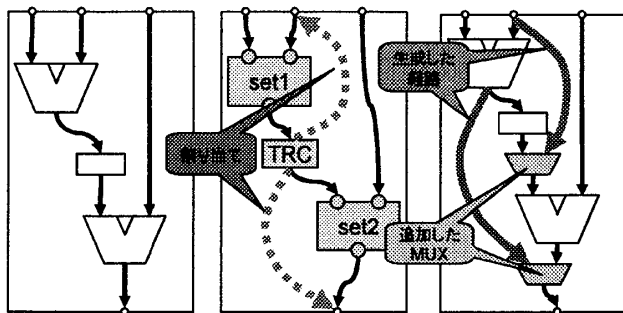


図 3: MUX の挿入による TRC の外部入出力への割り当て

以上により、5章、4章で定義した DFT アルゴリズムを DPG に対して適用し、低面積増加率および高故障検出率を達成する回路構造への変更を行う。

6. ベンチマーク回路に対する分割集合可検査性を満たす DFT アルゴリズムの適用結果

LWF, Paulin, Tseng の 32bit 均一ビット幅ベンチマーク回路に対して、セット可検査性、分割集合可検査性を満たす DFT を行った。その実験結果として DFT に伴う面積増加率と、故障検出率 100% を達成したテスト実行時間を表 1 に示す²。論理合成と故障シミュレーションの実行には DesingAnalyzer と TetraMax を利用した。また、生成した線形計画問題を解くために、lp_solve[3] パッケージを利用した。

表 1: 設計変更に伴う面積増加率とテスト実行時間比較

評価基準	回路名	提案手法	[1] の手法	[2] の手法
面積増加率	LWF	6.30%	32.47%	7.41%
	Paulin	3.14%	7.27%	3.14%
	Tseng	6.63%	17.58%	11.73%
テスト実行時間	LWF	173clock	947clock	616clock
	Paulin	1095clock	3575clock	3121clock
	Tseng	468clock	2234clock	3386clock

7. 関連手法との比較

提案手法は、既存の階層 BIST[1][2] よりも面積増加率の削減に成功した。これは故障検出率の保証が可能なセットに対して、経路生成を行う端子を制限し経路生成を行った結果、新たな経路生成を行った回数を削減したためと考えられる。[2] と比べ、Paulin の面積増加率が同一であった理由については、演算器が共有するレジスタが少なかったためと考えられる。共有するレジスタが少ない場合、TRC の個数が増加する。そのため、TRC を外部入出力へ割り当てる設計変更が多く発生し、面積増加率の削減ができなかったと考えられる。一方、Tseng は多くの共有するレジスタが存在する回路構造を持つため、5% 以上の面積増加率の削減に成功している。

また、提案した DFT を適用したときの故障検出率は、すべての組合せ回路要素に対して 100% を達成した。

テスト実行時間は、[1] や [2] に比べ 35% 以上の削減に成功した。提案手法では、演算器のテスト中に MUX のテストを同時に行っているため、MUX の故障検出に必要とするテスト実行時間が削減されたと考えられる。

8. おわりに：今後の課題

本稿では MUX を TTMS 要素としたが、AND ゲートなどはテストが容易であることが分かっている。この TTMS 要素と CTMS 要素の分類を定義することで、さらなる面積増加率の削減が可能になると予測される。

参考文献

- [1] Masuzawa et al, "Single-control testability of RTL data paths for BIST", Proc of ATS2000.
- [2] 山口, 井上, 藤原: "階層 BIST のためのテスト容易化設計に関する研究", 電子情報通信学会論文誌 (DI), 2003.
- [3] M.Berkelaar, lp_solve, Eindhoven University of Technology, ftp://ftp.ics.ele.tue.nl/pub/lp_solve.

²本研究は東京大学大規模集積システム設計教育研究センターを通し、シノプシス株式会社の協力で行われたものである