

LSI 出荷時におけるテストの高速化および信頼性向上に関する 研究紹介

土屋秀和

Research introduction on speedup and improving reliability of testing for executing at LSI shipment

by

Hidekazu TSUCHIYA^{*1}

(received & accepted)

Abstract

Recently, LSIs have been used in various products such as smartphones and automobiles etc. are indispensable parts in modern society. Also, the operating speed of LSI is faster and the scale of circuit for LSI is larger. LSIs are always tested on an individual basis at the time of shipment, these induce increasing power dissipation and the amount of test data for testing. For the reduction of these factors, we research the testing method that adjustable for power dissipation and test data size during testing by applying the clock control. The testing method achieve to reduce the test data size by using the method to mix pseudo random vector and ATPG (Automatic Test Pattern Generation) vector. In addition, the testing method achieve to reduce for power dissipation of CUT (Circuit Under Test) during testing by applying the clock control. In this paper, we introduce our research for LSI testing method.

Keywords: LSI Testing, Test pattern generation, Low power

1. はじめに

現在, LSI はあらゆる電子機器に組み込まれ, 我々の生活にとって必要不可欠な電子デバイスとなっている. これまで, LSI は製造技術の進歩により, 回路の大規模化による高機能化や動作速度の高速化などの発展を遂げてきた. しかし, 現在の製造技術において極僅かな異物や環境の変化などの様々な要因によって, 各々の LSI 全てを良品として製造することが困難となっている¹⁾. そのため, LSI 製造時における選別テストが必須となっている. また, LSI の回路の大規模化が進むにつれて, 回路が複雑となり完全テストが困難になっている. そして, この事態は LSI メーカーが抱える問題の一つとなっている. 例えば, 組込み用のマイコンを例に挙げると, 売価は数百円のものも存在する. しかしながら, そのマイコンをテストするためには, 数千円~数億円の高性能 LSI テスタを使用して, 長い時間テストを行わなくてはならない. さらに, LSI の実動作速度でのテストに対応するためには, より高性能な LSI テスタが必要となることから, テストコストも増加する. また, 性能が伴わない LSI テスタを使用してテストを行なった場合は, 実動作速度テストが行なえずにテスト品質の低下を招く.

一方, 近年の LSI の大規模化によって, システムの大部分

を 1 つのチップに搭載する SOC(System On Chip)化が進んでいる. これらの高機能化された LSI に対しては, より多くのテスト項目を必要とし, テスト時間が長大化する²⁾. その結果, LSI テスタの使用時間増加による出荷効率の低下によってテストコストが増加する. また, 不完全にテストが行なわれた場合は, テスト品質の低下のため, 出荷後の不良率が増加する恐れがある. これらの現状を改善する手法の一つとして, BIST(Built In Self Test)³⁾⁴⁾と呼ばれる手法が挙げられる. BIST は, LSI テスタの一部または全てを LSI 内に組込むことで, 自己テストを実現する手法であり, 自己テストによって LSI テスタの使用時間の削減を実現し, テストコストを削減することが可能である. また, 実動作速度テストおよび完全テストの実施によって, テスト品質の向上を実現することが可能である. BIST には, 実動作速度テストや高い故障検出率を実現する高品質テストの実現, テスト時間の短縮による更なるテストコストの削減との両立が要求されている.

これまで, LSI テスタの多くは故障モデルの一つである単一縮退故障を対象として故障検出を行ない, 良品を選別してきた¹⁾²⁾. この単一縮退故障とは, 被テスト回路中の信号線のうち, いずれか 1 つの信号線の論理値が 0 または 1 に固定(縮退)されてしまう故障である²⁾. ここで, NMOS インバータを例に挙げると, Fig.1 のようにドレインとソースが短絡した場合, 入力 X の状態を問わず出力 Z より論理値 0 が出力され, この状態を 0 縮退故障と呼ぶ. 一方で, Fig.2 のようにドレインまたはゲートが断線した場合, 入力 X の状態を問わず出力

^{*1} コンピュータ応用工学科 助教
Department of Applied Computer Engineering,
Assistant Professor

Z より論理値 1 が出力され、この状態を 1 縮退故障と呼ぶ。組み合わせ回路に対する単一縮退故障は、ATPG(Automatic Test Pattern Generator)と呼ばれる被テスト対象の回路構成情報に基づくコンピュータシミュレーションより生成されるテストパターンや、LFSR (Linear Feedback Shift Register) と呼ばれるシフトレジスタのビット列の一部を取り出し排他的論理和を取り、入力ビットとしてフィードバックすることで生成される疑似ランダムなテストパターンを用いて検出することができる。しかし、多くの LSI は一般的に順序回路であり、組み合わせ回路の他に記憶回路としてフリップフロップが実装されている。

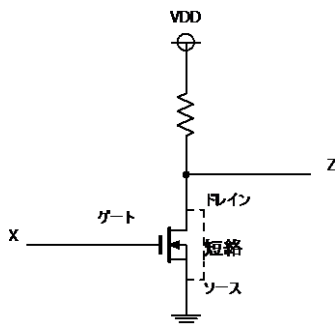


Fig.1 Stuck-at 0 fault

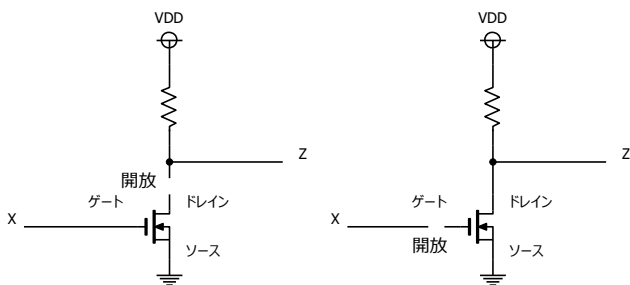


Fig.2 Stuck-at 1 fault

そこで、テストパターンを任意の回路箇所に加えるためには、このフリップフロップの状態を外部より制御できる必要があり、スキャンテストが広く用いられている。これは、Fig.3 に示すように、テスト時には回路内のフリップフロップ FF をシリアル接続に構成し、スキャンチェーンと呼ばれるシフトレジスタとして動作させる。スキャンテストでは、テストパターンをシリアルデータとしてスキャンチェーンに入力（スキャンイン）し、テストパターンに対する被テスト回路の出力応答を LSI の外部へ出力（スキャンアウト）することができる。このように、本来の機能を維持した状態で、テストが容易になるように LSI の構造の一部に変更を加えた設計手法をテスト容易化設計と呼ぶ。テスト容易化設計では、同期して多くの回路を強制的に動作させることもできる。この場合は、テスト時間を短縮することが可能であるが、通常使用時と比べてテスト時の消費電力は増加し、発熱による誤動作や故障、さらに回路の破壊に繋がる恐れがある。また、エレクトロマイグレーションと呼ばれる、イオンが徐々に移動し、半導体内部の構造が欠損していく現象による信頼性の低下も問題となる。

そこで、本稿では LSI 評価の高速化および信頼性向上に関する研究において、「テスト時の消費電力を抑制するテストパ

ターン生成技術」として、テスト時の消費電力を抑制することで歩留まり低下を抑制し、短いテスト時間で高い故障検出率を達成することにより高いテスト品質の実現と共にテストコストを削減し、実動作速度でテストパターンを発生することで高いテスト品質を実現するテストパターン発生器の回路構成およびテストパターン生成法⁵⁾⁶⁾について紹介させて頂く。

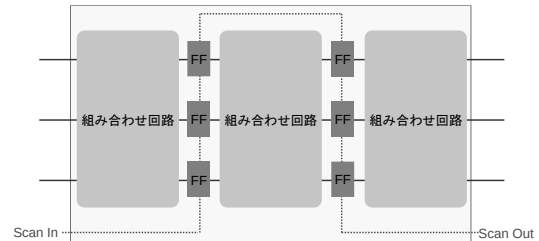


Fig.3 Structure of scan testing

2. テスト時の消費電力を抑制するテストパターン生成技術

2.1 準備

BIST では Fig.4 のように被テスト回路に加えてテストパターン発生器、テストパターンに対する出力応答のデータ圧縮を行なう出力応答圧縮器、テストパターンの出力応答とその期待値を比較するテスト結果比較器によって構成される。

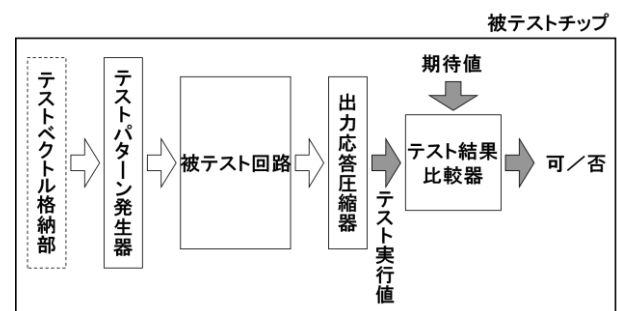


Fig.4 BIST principle diagram

更に、ATPG によって生成したテストパターンの部分集合など、あるテストベクトルを元にテストパターンを生成する場合は、ROM などのテストベクトル格納部の併設や外部の LSI テスタのメモリを使用する。また、BIST にはテストベクトル格納部などから元となるテストベクトルを利用してテストパターンを生成する方式や、元となるテストベクトルを利用せずに、LFSR (Linear Feedback Shift Register: 線形フィードバックレジスタ) やカウンタなどのハードウェアによってテストパターンを生成する方式がある。生成されたテストパターンは被テスト回路に印加され、そのテストパターンに対する出力応答が MISR (Multiple Input Signature Register: 多入力シグネチャシフトレジスタ) などの出力応答圧縮器によって圧縮される。全てのテストベクトルを ROM などに格納すると大容量のメモリが必要になってしまうため、BIST では出力応答を LFSR やカウンタなどの出力応答圧縮器で圧縮し、圧縮した出力応答（シグネチャ）と正しい期待値を圧縮した値とを比較してテストを行なう。このように出力応答圧縮器を用いて圧縮した出力応答と正しい期待値を圧縮した値を比

較するテスト方式をシグネチャ解析と言い、圧縮器をシグネチャレジスタまたはシグネチャアナライザと言う。この中でも、BIST 用テストパターン発生器 (TPG : Test Pattern Generator) は、BIST の構成要素の中でもテストの品質を左右する部分であり、より少ないハードウェア量で効率的なテストパターンを生成し、高い故障検出率を達成することが課題とされている。LFSR のみによって擬似ランダムベクトルを生成する場合は、被テスト回路の構成を考慮しないため、故障検出に貢献しないテストベクトルが多く生成されてしまう。従って、高い故障検出率を達成するための膨大なテストパターン数を縮小することが課題となっている。加えて、高い故障検出率を達成できない場合がある。また、ATPG ベクトルと擬似ランダムベクトルを混在させたテストパターンを生成する場合は、ATPG ベクトルの出現頻度を高めることによって、短いテスト長で高い故障検出効率を達成する可能性がある。しかし、ATPG ベクトルを格納するためには ROM などのメモリが必要となるため、短いテスト長で高い故障検出効率を達成しながらも、ATPG ベクトルを格納するメモリ量を削減することが課題である。

TPG に関しては、LFSR による擬似ランダムベクトル系列による故障検出効率を改善するための手法として、重み付き擬似ランダムパターン手法⁷⁾、ビットフィックス手法⁸⁾、ビットフリップング手法⁹⁾などが提案されている。また、擬似ランダムベクトルと ATPG ベクトルを混在させる手法として、LFSR の初期値を可変するリシード手法¹⁰⁾、ATPG ベクトルの部分集合を元にし、分割シフトパターンを生成する分割シフト法¹¹⁾、分割スキャンチェーンをローテッドスキャンするパーシャルスキャン手法¹²⁾などが提案されている。一方、近年における動作速度の高速化や回路の大規模化に伴って、テスト時の消費電力が問題となっている。これらの改善については、いくつかの Gated Clock 手法が研究され報告されている^{13)~16)}。Gated Clock 手法では、分割した回路ごとにクロック供給を制御することによって、消費電力を削減することができる。本稿では、擬似ランダムベクトルと ATPG ベクトルを混在させる手法として、Gated Clock 手法を適用した分割シフト法 TPG の回路構成とテストパターン生成法を提案し、高い故障検出率の達成に必要なテスト長とメモリ量、平均電流、総消費電力に関して、SR の分割数及び Gated Clock による動作率と特性との関係性を評価した。CUT には ISCAS' 85 ベンチマーク回路を用い、比較対象は、ATPG 及び LFSR とした。

2.2 テストパターン発生器の回路構成

Fig.5 に Gated Clock によりテスト時の消費電力を抑制するテストパターン技術を適用したテストパターン発生器 TPG の回路構成を示す。ここで、シフトレジスタ SR の分割数を k 、同時に動作する SR 数を g 、被テスト回路 CUT の入力数を m とする。TPG はメモリ MR とクロック制御部 CP と k 個に分割されクロック制御が適用されたシフトレジスタ SR により構成される。MR は k ビットのバス幅を持ち、ATPG ベクトルの部分集合が格納され、格納されたデータがクロック制御に従って各 SR に分配される。CP の入力であるイネーブル信号 (enable_1~enable_k) の組み合わせにより、SR 毎にクロック制御が行なわれ、CUT において分割した回路毎にクロック供給

のタイミングを制御することで、消費電力を削減可能である。TPG において、ランダムパターンで検出困難な故障である rpr(random pattern resistant)故障の検出を擬似ランダムなテストパターンのみで試みた場合、膨大なテストパターンが必要となることや、検出できずに故障検出率の低下を招き出荷後の動作不良の発生率が高くなる可能性がある。そこで、本技術を適用した TPG ではクロック制御が適用された SR により、周期的に出現する被テスト回路の回路構成を考慮し生成した ATPG ベクトルの部分集合を用いることで、rpr 故障を検出する。また、rpr 故障以外の故障は SR のシフト動作による擬似ランダムベクトルを使用して検出する。このように、rpr 故障以外の故障検出には擬似ランダムベクトルを使用し、rpr 故障検出には ATPG ベクトルを使用することで、ATPG ベクトルの全てをメモリに格納せず、その部分集合のみをメモリに格納するため、メモリに格納する ATPG ベクトル数の削減が可能である。これにより、LSI テスタのメモリ増設が必要となる頻度を低減可能であり、コスト削減に寄与する。また、rpr 故障を ATPG ベクトルで検出することで、故障検出に冗長なテストベクトルを削減可能となり、短いテスト長で高い故障検出率の達成が可能である。

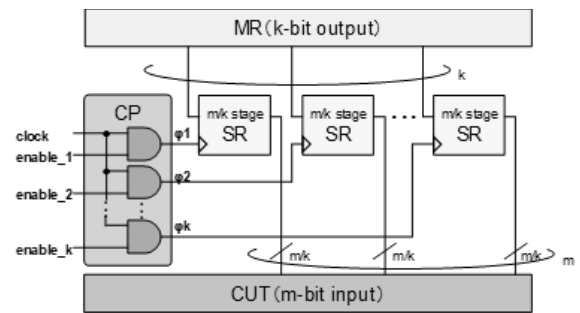


Fig.5 Circuit structure for test pattern generator

2.3 テストパターンの生成

本技術では、スキャンテストが適用された回路の単一縮退故障を対象とし、故障検出効率を次式で定義する。

$$\text{故障検出効率} = \frac{\text{検出故障数}}{\text{全故障数} - \text{冗長故障数}} \times 100\% \quad (2.1)$$

本技術によるテストパターン生成について述べる。テストパターンの生成は、大きく分けて 2 つの工程で構成される。ATPG ベクトルの部分集合の選択では、実テスト時間および必要なメモリ量を削減するために、ATPG ベクトルの部分集合を求める操作と共に圧縮操作を行なう。この操作は、大きく分けて 4 つのステップで構成される。また、TPG によるテストパターン生成では、実際に被テスト回路にテストパターンを印加するために、ATPG ベクトルの部分集合を元に TPG によりテストパターンを生成する。この操作は、大きく分けて 2 つのステップで構成される。以下に各操作の詳細を示す。

①ATPG ベクトルの部分集合の選択

Step 1 : ATPG ツールを使用して、被テスト回路に対する高い故障検出効率を達成する ATPG ベクトル集合を生成す

る。

Step 2: 分割数 k および同時に動作する SR 数 g の構成に基づいて、分割シフト動作によって分割シフトされたテストパターンを求める。

Step 3: 分割シフトされたテストパターンを使用して、故障シミュレーションを行ない、高い故障検出効率を達成するまでに使用された部分集合から、元となる ATPG ベクトルの部分集合を求める。選択した部分集合が前回求めた ATPG ベクトルの部分集合と比較し、同量または縮小された場合は Step 4 に移り、それ以外は前回求めた ATPG ベクトルの部分集合を最小とみなし確定する。

Step 4: 求めた ATPG ベクトルの部分集合の最後に位置するベクトルを rpr 故障検出に必要なベクトルと期待し、Fig.6 のように先頭に並び替えた後 Step 2 に戻り、この手順を繰り返す。

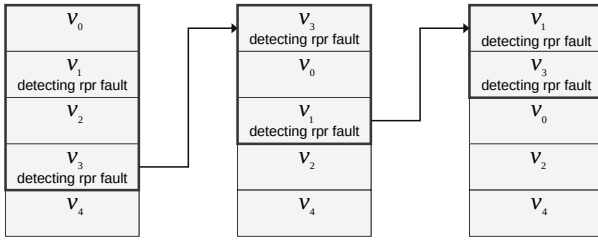


Fig.6 Reordering vectors

②TPG によるテストパターン生成

Step 1: “①ATPG ベクトルの部分集合の選択” で求めた部分集合をメモリに格納する。

Step 2: ATPG ベクトルの部分集合を元にクロック制御が適用された k 個の SR により、分割シフトされたビット幅が m ビットのテストパターンが生成される。テストパターンには、シフト動作によって生成される擬似ランダムベクトルと ATPG ベクトルが混在する。

ここで、同時に動作する SR 数を g とした時の SR の動 A.R を次式で定義する。

$$\text{動作率 A.R.} = \frac{g}{k} \times 100\%, \quad 1 \leq g \leq k \quad (2.2)$$

元になる ATPG ベクトルは、分割された k 個の SR によってシフトされる。すなわち、それぞれの SR において ATPG ベクトルは m/k シフトごとに出現する。また、それぞれの SR は Gated Clock によって k/g ごとにシフトする。すなわち、分割シフト法 TPG において ATPG ベクトルは $(m/k) \times (k/g) = m/g$ 回ごとに出現する。元となるベクトル数を q としたときに生成可能なテストベクトル数及びその中に含まれる ATPG ベクトル数、擬似ランダムベクトル数は以下に示される。

$$\text{生成可能なベクトル数} = (q-1) \times (m/g) + 1 \quad (2.3)$$

$$\text{ATPG ベクトル数} = q \quad (2.4)$$

$$\text{擬似ランダムベクトル数} = (q-1) \times (m/g) + 1 - q \quad (2.5)$$

テストパターンの生成例を CUT の入力数 $m=4$, 分割数 $k=2$,

動作率 A.R.=50%として、Fig.6 に示す。 $v_0(v_{00}, v_{01}, v_{02}, v_{03})$ と $v_1(v_{10}, v_{11}, v_{12}, v_{13})$ を元にして ATPG ベクトル 2 個 $(v_{00}, v_{01}, v_{02}, v_{03})$, $(v_{10}, v_{11}, v_{12}, v_{13})$, 擬似ランダムベクトル 3 個 $(v_{11}, v_{00}, v_{02}, v_{03})$, $(v_{11}, v_{00}, v_{13}, v_{02})$, $(v_{10}, v_{11}, v_{13}, v_{02})$ を生成し、 $v_1(v_{10}, v_{11}, v_{12}, v_{13})$ と $v_2(v_{20}, v_{21}, v_{22}, v_{23})$ を元にして ATPG ベクトル 2 個 $(v_{10}, v_{11}, v_{12}, v_{13})$, $(v_{20}, v_{21}, v_{22}, v_{23})$, 擬似ランダムベクトル 3 個 $(v_{21}, v_{10}, v_{12}, v_{13})$, $(v_{21}, v_{10}, v_{23}, v_{12})$, $(v_{20}, v_{21}, v_{23}, v_{12})$ を生成している。すなわち、ATPG ベクトル 1 個あたり ATPG ベクトル 1 個、擬似ランダムベクトル 3 個を生成している。このように、MR に格納された ATPG ベクトルの部分集合を、Gated Clock が適用された SR でシフトしながらテストパターンの生成を行うため、実テストの面で以下の効果が期待される。

- (1) 高い故障検出効率の達成が可能である。
- (2) 実動作速度でのテストパターン生成が可能である。
- (3) Gated Clock による SR の動作率の変更によって、テスト長と元となるベクトル数、テスト長と平均電流の各トレードオフを可能とする。

$$\begin{array}{l} v_0 = (v_{00} \ v_{01} \ | \ v_{02} \ v_{03}) \\ v_1 = (v_{10} \ v_{11} \ | \ v_{12} \ v_{13}) \\ v_2 = (v_{20} \ v_{21} \ | \ v_{22} \ v_{23}) \end{array} \quad \Rightarrow \quad \begin{array}{l} (v_{00} \ v_{01} \ | \ v_{02} \ v_{03}) \\ (v_{11} \ v_{00} \ | \ v_{02} \ v_{03}) \\ (v_{11} \ v_{00} \ | \ v_{13} \ v_{02}) \\ (v_{10} \ v_{11} \ | \ v_{13} \ v_{02}) \\ (v_{10} \ v_{11} \ | \ v_{12} \ v_{13}) \\ (v_{21} \ v_{10} \ | \ v_{12} \ v_{13}) \\ (v_{21} \ v_{10} \ | \ v_{23} \ v_{12}) \\ (v_{20} \ v_{21} \ | \ v_{23} \ v_{12}) \\ (v_{20} \ v_{21} \ | \ v_{22} \ v_{23}) \end{array}$$

Fig.7 Example of the pattern generation

3. 評価方法

前述のテスト時の消費電力を抑制するテストパターン生成技術について、分割数 $k=1, 2, 4, 8$, 動作率 A.R.[%]=100, 50, 25, 12.5 とし、評価尺度を高い故障検出効率の達成に必要なテスト長、元となるベクトル数、平均電流として SR の分割数およびクロック制御による動作率と特性との関係性を評価した。元となるベクトル集合は ATPG ツールで生成し、2.3 項の手順で圧縮し選択を行なった。ターゲットデバイスとして、プログラムにより自在にデジタル回路を構成可能なデバイスである FPGA (Xilinx 製 Virtex-E:xcv405e-8fg676) に Xilinx 社製デザインツール ISE9.1i を用いて、評価対象の回路をインプリメントした。また、論理合成した CUT 及び TPG に対して、ISE9.1i に搭載されている消費電力シミュレータ (XPower) を用いて平均電流の評価を行った (ロジック部の電圧 = 1.8V)。CUT には ISCAS' 85 ベンチマーク回路を用いた。Table 1 に評価対象を示す。論理合成結果は、FPGA の論理ブロックである 4 入力 LUT (Look Up Table) 数で示した。比較対象は、ATPG および原始 LFSR とした。また、本技術を適用した TPG, ATPG, 原始 LFSR によって生成されるテストパターンは、それぞれ高い故障検出効率を達成するものとした。

Table 1 Evaluation targets

CUT	機能	入力数	出力数	ゲート数	4入力LUT数
C432	27チャンネル割り込みコントローラ	36	7	160	91
C499	32ビットシングルエラー訂正回路	41	32	202	78
C880	8ビットALU	60	26	383	101
C1355	32ビットシングルエラー訂正回路 (C499のXORを4入力NANDに置換)	41	32	546	78
C1908	16ビットエラー検出／訂正回路	33	25	880	130
C2670	12ビットALU／コントローラ	233	140	1193	158
C5315	9ビットALU	178	123	2307	389
C6288	16 × 16 乗算器	32	32	2406	703
C7552	34ビット加算器／入力パリティチェック付き大小比較器	207	108	3512	442

4. 評価結果

評価結果としてベンチマーク回路 C499 に着目し、Table 2 に示す。左の列から、CUT：ISCAS' 85 ベンチマーク回路の回路名、#k：SR の分割数、%A.R.：全体の SR に対して同時にシフト動作している SR 数の割合、%F.E.：故障検出効率、#L：高い故障検出効率を達成するために必要なテスト長、#MR：MR に格納される ATPG ベクトルの部分集合、#power：テスト実行時の平均電流を示す。この結果をもとに、ATPG および LFSR を対象として本技術について比較検討を行った。また、SR の分割数 k およびクロック制御による動作率 A.R. と特性との関係を検討した。

Table 2 Evaluation result (C499)

CUT	#k	% A. R.	% F. E.	#L	#MR	#power [mA]
C499	1	100	100	334	9	1.77
	2	50	100	578	15	0.89
		100	100	258	13	1.61
	4	25	100	535	14	0.54
		50	100	373	19	1.02
		100	100	231	24	2.03
	8	12.5	100	801	21	0.34
		25	100	521	27	0.61
		100	100	176	36	2.12
	41 (ATPG)	100	100	52	52	3.31
	LFSR	100	100	1167	0	2.15

4.1 ATPG との比較

ATPG に比べてテスト長は長くなるものの、元となるベクトル数および平均電流を大幅に低減可能である。先ず、元となるベクトルの縮小率すなわち ATPG ベクトルをすべて印加した場合に対する ATPG ベクトルの縮小の度合いに関しては、ATPG パターンをすべて印加した場合に比べて、少ない ATPG ベクトル数（平均 60.8%）で高い故障検出効率（99.5～100%）を達成することが可能である。また、平均電流の縮小率すな

わち ATPG ベクトルをすべて印加した場合に対する平均電流の縮小の度合いに関しては、ATPG パターンをすべて印加した場合に比べて減少する（平均 42.4%）。

例えば C499 において、ATPG（分割数 k = 41）に対して k = 8 の場合を考える。動作率 A.R.=25%では、元となるベクトル数が 52 から 27 に減少し約 0.52 倍、平均電流は 3.31mA から 0.74mA に減少し約 0.22 倍に減少する。また、動作率 A.R.=12.5%では、元となるベクトル数が 52 から 21 に減少し約 0.40 倍、平均電流は 3.31mA から 0.47mA に減少し約 0.14 倍に減少する。

4.2 LFSR との比較

LFSR に比べて、元となるベクトルが必要となるが、テスト長を大幅に削減することが可能であり、平均電流も動作率 A.R. に応じて大幅に削減することができる。テスト長の縮小率すなわち LFSR のテスト長に対する縮小の度合いに関しては、LFSR で生成したパターンを印加した場合に比べて、短いテスト長（平均 35.9%）で高い故障検出効率（99.5～100%）を達成する。

例えば C499 において、LFSR に対して分割数 k=8 を比べると、動作率 A.R.=25%では、テスト長が 1167 から 521 に減少し約 0.45 倍、平均電流が 2.15mA から 0.74mA に減少し 0.34 倍に減少する。また、動作率 A.R.=12.5%の場合は、テスト長が 1167 から 801 に減少し約 0.69 倍、平均電流が 2.15mA から 0.47mA に減少し 0.22 倍に減少する。

4.3 分割数 k および動作率 A.R.

出力部の SR の分割数に応じて、ATPG ベクトルの出現頻度を変えることができる。すなわち、分割数 k の減少に従って、擬似ランダムベクトルの出現頻度が高まり、テスト長は長くなるものの、元となるベクトル数を削減することが可能である。

評価した範囲では、分割数が最大の k = m (ATPG) に対して分割数が最小の k = 1 の場合は、動作率 A.R. = 100%では、元となるベクトル数は平均 45.5%に減少する。また、動作率 A.R. に関しては、その率を低下させるに従って、テスト長は長くなるが、平均電流を削減することが可能である。評価した範囲では、分割数 k = 8 の場合、動作率が最大の A.R. = 100%に対して動作率が最小の A.R. = 12.5%の場合、平均電流は平均 18.5%に減少する。

5. おわりに

本稿では、LSI の評価技術の現状および我々が取り組んでいる研究の一部を紹介した。研究として、Gated Clock 手法が適用可能な TPG について紹介し、高い故障検出効率を達成するテスト長、元となるベクトル数、平均電流について評価を行った。比較対象は、ATPG 及び原始 LFSR を、CUT には ISCAS' 85 ベンチマーク回路を用いた。また、SR の分割数及び Gated Clock による動作率と特性との関係の評価した。評価した範囲では分割シフト法 TPG は ATPG と比較して、元となるベクトル数と平均電流を削減することが可能である。また、LFSR と比較して、テスト長と平均電流を削減することが可能である。今後は、引き続き「テスト時の消費電力を

抑制するテストパターン生成技術」に関して、単一縮退故障以外の故障への拡張対応などを行なっていきたい。

参考文献

- 1) LSI テスティング学会, LSI テスティングハンドブック, オーム社(2008).
- 2) 藤原秀雄, デジタルシステムの設計とテスト, 工学図書(2004).
- 3) S. Wang, “A BIST TPG for low power dissipation and high fault coverage,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.15, no.7, pp.777–789 (2007).
- 4) P. Wohl, J.A. Waicukauski, S. Patel, and M.B. Amin, “X-tolerant compression and application of scan-atpg patterns in a bist architecture,” Proc. IEEE International Test Conference, vol.1, pp.727–736, Charlotte, USA (2003).
- 5) 土屋秀和, 阿部高也, 増田良介, 浅川毅, “擬似ランダムベクトルと ATPG ベクトルを利用した低電力指向 TPG,” 電子情報通信学会論文誌 D, vol.J92-D, No.6, pp.777-783(2009).
- 6) Hidekazu Tsuchiya, “Testing method that adjustable for power dissipation and test data size during testing,” IEEE 19th Workshop on RTL and High Level Testing, Hefei, China (2018).
- 7) J. Dworak, “An analysis of defect detection for weighted random patterns generated with observation/excitation-aware partial fault targeting,” Proc.IEEE 25th VLSI Test Symposium, pp.205–210, Berkeley, USA (2007).
- 8) N.A. Touba and E.J. McCluskey, “Bit-fixing in pseudorandom sequences for scan BIST,” IEEE Trans. Comput. Aided Des. Integr. Circuits Syst., vol.20, no.4, pp.545–555 (2001).
- 9) Y. Chaowen, S.M. Reddy, and I. Pomeranz, “Circuit independent weighted pseudo-random BIST pattern generator,” Proc. IEEE 14th Asian Test Symposium, pp.132–137, Calcutta, India (2005).
- 10) H.-S. Kim, Y. Kim, and S. Kang, “Testdecompression mechanism using a variable-length multiple-polynomial LFSR,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.11, no.4, pp.687–690 (2003).
- 11) T. Asakawa and K. Iwasaki, “On using ATPG vectors for BIST TPG,” Proc. IEEE First Asia Pacific ASICs, pp.359–362, Seoul, Korea (1999).
- 12) S.M. Reddy and I. Pomeranz, “On undetectable faults in partial scan circuits using transparent-scan,” Proc. IEEE 22nd International Conference on Computer Design, pp.o82–o84, San Jose, USA (2004).
- 13) T.E. Yu, T. Yoneda, D. Zhao, and H. Fujiwara, “Using domain partitioning in wrapper design for IP cores under power constraints,” Proc. IEEE 25th VLSI Test Symposium, pp.369–374, Berkeley, USA (2007).
- 14) M. Pedram and J. Oh, “Gated clock routing for low-power microprocessor design,” IEEE Trans. Comput. Aided Des. Integr. Circuits Syst., vol.20, no.6, pp.715–722 (2001).
- 15) Y. Bonhomme, P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, “A Gated clock scheme for low power scan testing of logic ICs orembedded cores,” Proc. IEEE 10th Asian Test Symposium, pp.253–258 (2001).
- 16) 土屋秀和, 阿部高也, 浅川毅, “ATPG ベクトルを利用した TPG の電流評価,” 信学技報, DC2007-80 (2008).