設 計 自 動 化 91－1
(1999. 2. 4)

BIST用TPGにおけるATPGテストベクトルの利用

浅川 毅, 金本 直之, 出崎 善久, 岩崎 一彦

東京都立大学大学院工学研究科電気工学専攻
〒192-0397 東京都八王子市南大沢1-2

E-mail: asakawa@ic.eei.metro-u.ac.jp

あらまし

ATPG ツールにより得られたテストベクトルの一部をシフトして得られるベクトルの集合を被テスト回路への入力とするテストパターン生成器(TPG)の提案を行う．提案する TPG はシフトレジスタと少量の ROM で構成される．ISCAS ベンチマーク回路に対して提案した TPG を適用した場合の 100%の故障検出率を得るためのテスト長，及びハードウェア量の評価を行った．また，提案した TPG を使用した場合の 100%の故障検出率を得るためのテスト長の式を導出し，故障シミュレータにより得られたテスト長との比較を行った．

キーワード

BIST, TPG, ATPG, 検出困難故障

A Test Pattern Generation Method for BIST Based on Shifted ATPG Test Patterns

Takeshi Asakawa, Naoyuki Kanamoto, Yoshihisa Desaki, Kazuhiko Iwasaki

Graduate School of Engineering, Tokyo Metropolitan University
1-2 Minami-ohsawa, Hachioji, Tokyo 192-0397, Japan

E-mail: asakawa@ic.eei.metro-u.ac.jp

Abstract

We present a test pattern generation method for BIST. The proposed test pattern generator (TPG) is based on shifted test patterns generated by an ATPG-tool. The TPG consists of shift registers and small amount of ROMs. We evaluated the test length for covering all faults and the amount of its hardware when the TPG is equipped with ISCAS benchmark circuits. We also derived a formula of the test length for covering all faults and compared it with the real length that is given by using a fault simulator.

key words BIST, TPG, ATPG, random pattern resistant fault

1. はじめに

近年，VLSI のテスト手法として，BIST (Built-In Self-Test) が注目されている[1]-[3]．BIST を行う被テスト回路 CUT (Circuit Under Test) には TPG (Test Pattern Generator) と MISR (Multiple Input Signature Register) が付加される．TPG で発生したテストパターンを CUT に与え，出力結果を MISR で圧縮，判定する．一般に TPG の評価は以下の 2 点を基準にして行うことが多い．

・付加回路のハードウェア量．
・要求される故障カバレージを達成するためのテスト長．

一般に，ハードウェア量とテスト長はトレードオフの関係にある．原始多項式に基づく LFSR(原始 LFSR) を TPG としたランダムテストでは，ハードウェアオーバヘッドは少ないが，高い故障カバレージを得るためのテスト長は一般に非常に大きくなる．その原因として，LFSR が発生する擬似ランダムパターンで検出困難な(rpr: random pattern resistant)故障 が存在することが知られている [4]- [8]．また，ATPG (Automatic Test Pattern Generation) ツールで生成したテストパターンの全てを使用する決定論的テストでは，短いテスト長で 100%の故障カバレージを達成することは可能であるが，著しいハードウェアオーバヘッドの増加を招く．

ランダムテストにおけるテスト長を短縮するために，LFSR の初期値を入れ替える方法[9]，LFSR で生成されるテストベクトルに重みを付ける方法[8] ,[10]-[14]など rpr 故障を検出する方法，或いは CUT に回路を追加し rpr 故障を減らす方法 [6],[15],[16]などが提案されている．

本稿では，ランダムテストと決定論的テストの折衷案として，ハードウェアオーバーヘッドが少なくテスト長を短縮するテストパターン生成器 (TPG) 方式の提案を行う．ATPG ツールにより得られたテストベクトルの一部をシフトして得られるベクトルの集合を被テスト回路への入力とする．提案する TPG 方式はシフトレジスタと少量の ROM で構成する．すなわち ATPG ツールで得られたテストベクトルの一部を ROM に保存し，TPG のパターン生成に利用することにより，rpr 故障の検出を早めテスト長を短縮する．ISCAS ベンチマーク回路を使用して，テスト長，及びハードウェア量の評価を行った．その結果，従来の LFSR を用いた TPG 方式に比べて同じ故障カバレージを得るために若干のハードウェアオーバヘッドが増えるものの，テスト長は大幅に短縮されることが確認された．また，提案 TPG 方式のテスト長の近似値を理論的に解析し，故障シミュレーション値との比較を行った．

本稿の構成を以下に示す．次の章では提案方式の TPG 構成を示す．第 3 章では ATPG ツールで生成されたベクトルからどのテストベクトルを選択するかという手順を示す．第 4 章では ISCAS '85，ISCAS '89 ベンチマーク回路に対して提案方式の TPG を BIST に使用した場合のハードウェアオーバヘッドとテスト長の評価を行い，原始 LFSR 方式の TPG との比較結果を示す．第 5 章では提案方式の TPG のテストパターン長に関して理論的な解析を行い，故障シミュレーション値との比較を示す．第 6 章では本研究のまとめを示す．

2. ROM を用いた TPG の構成

mビット入力の CUT において，ある対象故障を検出できるテストベクトルが i 個存在する場合，ランダムテストパターンでこの故障を検出できる確率は，$i/2^m$ であり，m が多く i が少ない対象故障ほど，原始 LFSR 方式の TPG で生成されるテストパターン (擬似ランダムパターン) で検出することは困難となる．提案する方式では ATPG ツールで生成したテストベクトル(ATPG ベクトル)を利用し，このようなランダムパターンで検出が困難な故障に対して検出を行う．

図 1 に提案方式の TPG 構成を示す．m ビット入力の CUT に対して，1 ビット入力，m/k ビット出力のシフトレジスタ k 個と ROM により構成される．ROM には，ATPG ベクトルの一部が保存される．テスト中において，ATPG ベクトルは m/k 回に 1 回の割合で出現し，その中間で生成されるテストパターン

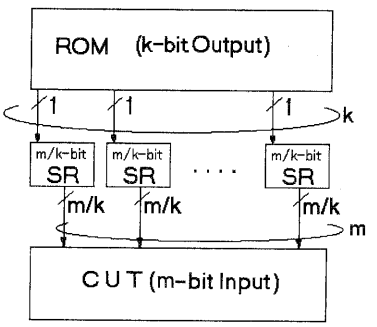

図 1 提案する TPG 構成
Fig.1 Block diagram of proposed TPG.

によっても故障が検出される．従って ROM に保存すべきテストベクトル数は ATPG ベクトル数よりも少なくなることが期待される．

分割数 k=1 とした場合の構成例を図2に示す． 1 ビット入力，m ビット(CUT の入力幅)出力のシフトレジスタは，テストクロックごとに ROM より 1 ビットのデータを読み込み，m ビット幅のテストベクトルを生成し CUT に与える．例えば m=4 の場合を考える．ROM に保存されているベクトル $V_1(v_{13},v_{12},v_{11},v_{10})$とベクトル $V_2(v_{23},v_{22},v_{21},v_{20})$を使用して生成されるベクトルは，図 3 に示す様に $(v_{13},v_{12},v_{11},v_{10})$, $(v_{20},v_{13},v_{12},v_{11})$, $(v_{21},v_{20},v_{13},v_{12})$, $(v_{22},v_{21},v_{20},v_{13})$, $(v_{23},v_{22},v_{21},v_{20})$ の順に生成される．

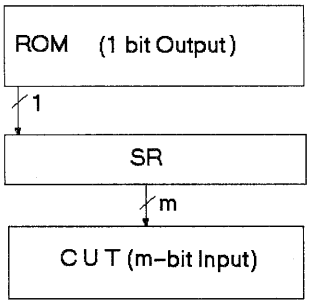

図2 提案する TPG 構成(k=1)
Fig.2 Block diagram of proposed TPG (k=1).

ATPG ベクトルの出現頻度を上げるためには，シフトレジスタの分割数を多くする．k 分割した場合,ATPG ベクトルは m/k 回に 1 回の頻度で出現する．

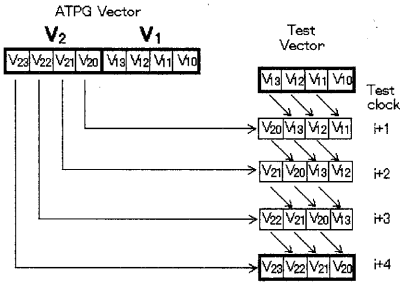

図3 テストベクトル生成 (k=1)
Fig. 3 TestVector generation (k=1).

3. ROM に保存する ATPG ベクトルの選択手順

提案する方式では，テスト長とハードウェアオーバヘッドとのトレードオフが課題となる．シフトレジスタの分割数 k，どの ATPG ベクトルを ROM に保存するかという選択，ATPG ベクトルの保存順序を検討する必要がある．

図 4 に ROM へ保存するベクトルを決定するための手順を示す．

（1）ATPG パターン生成

ATPG ツールを使用し，故障カバレージが 100% となるテストパターン(ATPG ベクトル)を生成する．

（2）故障シミュレーション

TPG によって生成されたテストパターンにより故障シミュレーションを行い,故障カバレージが 100% になった時点でのテスト長を求める．ATPG ベクトル入れ替え操作（（3）で説明）前のテスト長と比較し，改善されたかどうか判定する．改善されなければ終了する．改善されればさらに入れ替えを行う．

（3） ATPG ベクトルの出現順序の入れ替え

前項故障シミュレーションにおいて最後に出現するテストベクトルが rpr 故障の検出に必要なベクトルと期待し，先頭に移動する．

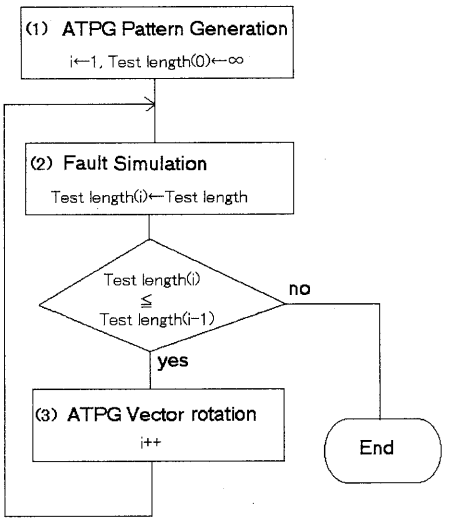

図4 ATPG ベクトルの選択手順
Fig.4 The ATPG vector selection algorithm.

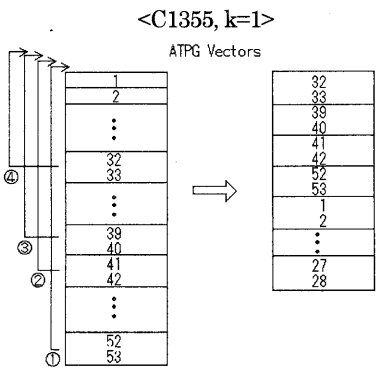

図5　ATPGベクトルの選択例
Fig. 5 Selection of the ATPG vectors.

テスト長が図4の手順により縮小される過程を図5に示す．C1355(入力数 m=41)に対して TPG のシフトレジスタの分割数 k を 1 とした場合の例である．Step1~Step4 の順で ATPG パターンの先頭に ATPG ベクトルの巡回置換を行う．Test length の列は，検出可能故障(Detectable faults: 1566 個)を検出するのに必要なテスト長を示す．ATPG ベクトルを Step1~Step4 の 4 回に渡って並べ替えることにより，テスト長を 2106 から 1415 へ縮小することができる．評価した範囲では，ATPG ベクトルを全て使用した場合に比べてテスト長を平均 k=1:45.5%, k=2:61.5%, k=4:77.5%, k=8:86.5%に縮小することができた．

C1355 ベンチマーク回路に対し TPG のシフトレジスタの分割数 k を 1,2,4,8,m(m=41 : CUT の入力数)

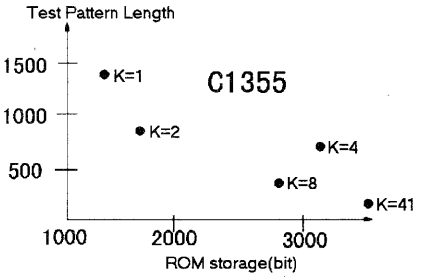

図6 ROM 容量とテスト長(C1355)
Fig.6 ROM storage VS Test pattern length (C1355).

とした場合の ROM 容量とテスト長の関係を図 6 に示す．図中の点(x, y)は，100%の故障カバレージを達成するために x ビットの ROM 容量と y個のテスト長が必要であることを示す．x値の限界値は ATPG ベクトルの総ビット数，y値の限界値は全ての ATPG ベクトルを k=1 で生成する場合のテスト長となる．

ROM 容量はハードウェアオーバヘッドを決定し，テスト長はテスト時間を決定する．従って，これらのトレードオフによって分割数 k の決定を行う．多くの場合，分割数 k が増えるにつれて，ROM 容量は増加し，テスト長は短くなる傾向を示す．

4.　評価結果

ISCAS '85 と ISCAS '89 に含まれる組み合せ回路に対して故障シミュレーションを行い，提案方式の TPG のテスト長とハードウェアオーバヘッドを評価した．冗長故障を除く検出可能な故障数をすべて検出するとき 100%故障カバレージとした．

提案方式の TPG について，分割数k を 1,2,4,8,m とし，100%の故障カバレージ（C1908，S953 についてはそれぞれ 99.5%，99.9%）を達成するのに必要なテスト長と ROM 容量を求めた．また，テストクロック毎に ATPG ベクトルを発生する論理合成方式の TPG(VHDL で設計し，論理合成で作ったシーケンス回路)に対しても必要なテスト長とハードウェア量を求めた．得られた結果を原始 LFSR による TPG の結果に対して，ハードウェアオーバヘッドとテスト長の面で比較し，比率を求めた．TPG のロジック部のトランジスタを基本単位として，ROM 記憶部を 1/10，ROM 周辺部(アドレスデコーダ等)を 2/5 として，ハードウェアオーバヘッドの算出を行った．

表1，表2にこれらの評価結果を示す．k の列において，数値は提案方式の TPG における分割数k を示し，VHDL，LFSR は比較のために使用した方式を示す．ROM の列は，提案方式の TPG に必要な ROM 領域をビット数で示す．Reduction の列は，各方式の TPG ハードウェアオーバヘッドとテスト長に関して，原始 LFSR 方式の TPG の場合を 100%としたときの比率を示す．

評価した提案方式の TPG は原始 LFSR 方式の TPG と比べて，若干のハードウェアオーバーヘッド(7.2%～59.5%)の増加があるものの，テスト長は大幅(0.1%～94.2%)に短縮されることが確認された．論理合成方式の TPG では，テスト長は 0.1%～4.5%の範囲で縮小されるがハードウェアオーバヘッドの面では，15.0%～185.1%の範囲で増加した．C6288 は入力数

表1　ISCAS '85 ベンチマーク回路による評価結果

Table1 Experimental results for ISCAS '85 Benchmark circuits.

CUT	k	Test length	ROM (bit)	Reduction(%) TPG area	Reduction(%) Test length	Fault Coverage (%)
C432	1	154	144	111.3	11.6	100.0
	2	297	576	118.5	22.3	100.0
	4	170	648	118.1	12.8	100.0
	8	84	756	118.4	6.3	100.0
	36	29	1008	119.8	2.2	100.0
	VHDL	29		150.4	2.2	100.0
	LFSR	1331		100.0	100.0	100.0
C499	1	334	328	113.9	28.6	100.0
	2	258	492	115.0	22.1	100.0
	4	231	943	119.3	19.8	100.0
	8	186	1517	124.1	15.9	100.0
	41	52	2091	127.5	4.5	100.0
	VHDL	52		180.7	4.5	100.0
	LFSR	1167		100.0	100.0	100.0
C880	1	727	720	113.8	4.8	100.0
	2	269	480	110.2	1.8	100.0
	4	166	660	110.9	1.1	100.0
	8	124	1020	113.0	0.8	100.0
	60	20	1140	112.5	0.1	100.0
	VHDL	20		133.3	0.1	100.0
	LFSR	15024		100.0	100.0	100.0
C1355	1	1415	1394	129.9	76.1	100.0
	2	848	1640	129.1	45.6	100.0
	4	771	3157	142.0	41.5	100.0
	8	361	2952	138.0	19.4	100.0
	41	84	3403	139.2	4.5	100.0
	VHDL	84		256.7	4.5	100.0
	LFSR	1860		100.0	100.0	100.0
C1908	1	2306	2277	149.3	46.9	99.5
	2	1569	3432	159.5	31.9	99.5
	4	815	3333	154.3	16.6	99.5
	8	408	3366	152.0	8.3	99.5
	33	106	3465	149.7	2.2	99.5
	VHDL	106		285.1	2.2	99.5
	LFSR	4914		100.0	100.0	99.5
C2670	1	10253	10252	121.1	15.6	99.9
	2	5105	10252	121.1	7.8	99.9
	4	2553	10252	121.1	3.9	99.9
	8	1277	10252	121.1	1.9	99.9
	233	45	19252	121.1	0.1	99.9
	LFSR	65536		100.0	100.0	88.8
C5315	1	3594	3560	112.2	94.2	100.0
	2	1412	2670	109.0	37.0	100.0
	4	1286	5162	113.9	33.7	100.0
	8	859	6942	116.9	22.5	100.0
	178	50	8722	118.9	1.3	100.0
	VHDL	50		144.7	1.3	100.0
	LFSR	3814		100.0	100.0	100.0
C6288	1	55	32	107.2	91.7	100.0
	2	33	64	108.5	55.0	100.0
	4	34	128	110.2	56.7	100.0
	8	29	224	111.9	48.3	100.0
	32	14	416	114.3	23.3	100.0
	VHDL	14		115.0	23.3	100.0
	LFSR	60		100.0	100.0	100.0

(32 ビット)と必要とする ATPG ベクトル数（14 個）が少ないため，ハードウェアオーバヘッドは論理合成方式の TPG：15.0%,提案方式の TPG(k=32)：14.3%となった．

提案方式の TPG のテスト長を，他の文献で示されている原始 LFSR 方式 TPG のテスト長と比較した結果を表 3 に示す．LFSR1 の列は，今回の故障シミュレーションで得られた原始LFSR方式での値を示す．LFSR の初期値は ATPG テストパターンの先頭ベクトルとし，原始多項式は文献[17]Appendix C Table C.2 より引用した．LFSR2[6]，LFSR3[18]の列は，文献[6],[18]での値を示す．( )内の値は，故障カバレージ(%)を示す．

いずれの場合も提案方式の TPG のテスト長は，原始 LFSR 方式の TPG に比べ縮小される．

5.　テストパターン長の近似解析

提案方式の TPG について，故障を 100%検出するのに必要なテスト長の近似値を理論的に解析し，故障シミュレーション結果との比較を行った．以下に示す仮定および定義に従って理論式を導いた．

[仮定 1]100%の故障を検出するのに必要なテスト長は，rpr 故障の数とその検出確率によって決定される．

[仮定 2]個々のテストベクトルが検出する rpr 故障数の期待値は同一である．

[仮定 3]ATPG ベクトルは異なる 1 個の rpr 故障を検出する．

以下に記号の意味を示す．

- l　：故障を 100%検出するのに必要なテスト長
- N　：rpr 故障の数
- P_i　：1 個のランダムベクトルが i 番目の rpr 故障を検出する確率
- P　：Pi(i=1～N)の平均
- D_i　：i 番目の rpr 故障を検出できるテストベクトルの総数
- m　：CUT の入力数
- k　：TPG の分割数

$P_i=D_i/2^m$ であり，[仮定 2]より，N 個中の i 番目の rpr 故障を検出できるテストベクトル数 D_i の幾何平均をベクトル総数 2^m で割ることにより，1 個のランダムベクトルが任意の rpr 故障を検出する平均確率 P を式(1)で近似することができる．

$$P = \sqrt[N]{\prod_{i=1}^{N} D_i} \,/\, 2^m \qquad (1)$$

長さ l のランダムパターンで 1 個の rpr 故障を検

表2　ISCAS '89 ベンチマーク回路による評価結果

Table2 Experimental results for ISCAS '89 Benchmark circuits.

CUT	k	Test length	ROM (bit)	Reduction(%) TPG area	Reduction(%) Test length	Fault Coverage (%)
S208	1	210	209	124.7	5.1	100.0
	2	163	342	124.5	3.9	100.0
	4	97	456	128.9	2.3	100.0
	8	55	513	128.6	1.3	100.0
	19	28	513	127.0	0.7	100.0
	LFSR	4140		100.0	100.0	100.0
S298	1	143	136	128.1	33.5	100.0
	2	121	255	132.0	28.3	100.0
	4	73	306	132.0	17.1	100.0
	8	44	374	132.6	10.3	100.0
	17	24	391	131.8	5.6	100.0
	LFSR	427		100.0	100.0	100.0
S344	1	193	192	118.9	91.5	100.0
	2	85	168	116.3	40.3	100.0
	4	62	240	117.4	29.4	100.0
	8	31	240	116.3	14.7	100.0
	24	13	288	116.4	6.2	100.0
	LFSR	211		100.0	100.0	100.0
S349	1	168	168	117.9	127.3	100.0
	2	97	192	117.1	73.5	100.0
	4	61	240	117.4	46.2	100.0
	8	31	240	116.3	23.5	100.0
	24	13	288	116.4	9.8	100.0
	LFSR	132		100.0	100.0	100.0
S382	1	217	216	119.9	18.3	100.0
	2	135	264	119.4	11.4	100.0
	4	91	360	120.6	7.7	100.0
	8	58	456	121.5	4.9	100.0
	24	25	576	122.4	2.1	100.0
	LFSR	1187		100.0	100.0	100.0
S386	1	391	390	147.4	9.5	100.0
	2	289	624	153.3	7.0	100.0
	4	169	728	153.1	4.1	100.0
	8	64	832	153.6	1.6	100.0
	13	64	819	151.3	1.6	100.0
	LFSR	4109		100.0	100.0	100.0
S400	1	150	144	116.9	28.6	100.0
	2	112	288	120.3	21.3	100.0
	4	91	360	120.1	17.3	100.0
	8	52	408	120.4	9.9	100.0
	24	24	552	121.9	4.6	100.0
	LFSR	525		100.0	100.0	100.0
S420	1	634	630	122.2	1.0	100.0
	2	528	1085	126.7	0.8	100.0
	4	313	1365	128.2	0.5	100.0
	8	169	1470	127.7	0.3	100.0
	35	44	1505	126.0	0.1	100.0
	LFSR	65536		100.0	100.0	97.2
S444	1	158	144	116.9	66.4	100.0
	2	137	264	119.4	57.6	100.0
	4	71	264	118.1	29.8	100.0
	8	49	384	119.9	20.6	100.0
	24	25	576	122.4	10.5	100.0
	LFSR	238		100.0	100.0	100.0

CUT	k	Test length	ROM (bit)	Reduction(%) TPG area	Reduction(%) Test length	Fault Coverage (%)
S510	1	576	575	129.6	65.0	100.0
	2	457	950	134.7	51.6	100.0
	4	175	725	127.6	19.8	100.0
	8	127	1008	131.1	14.3	100.0
	25	55	1296	133.6	6.2	100.0
	LFSR	886		100.0	100.0	
S526	1	635	624	132.1	6.4	100.0
	2	397	792	132.6	4.0	100.0
	4	247	1920	151.0	2.5	100.0
	8	136	1080	133.6	1.4	100.0
	24	49	1152	132.6	0.5	100.0
	LFSR	9970		100.0	100.0	100.0
S641	1	756	756	115.7	1.2	100.0
	2	433	864	115.2	0.7	100.0
	4	222	918	114.5	0.3	100.0
	8	115	1026	114.5	0.2	100.0
	54	23	1188	114.3	0.1	100.0
	LFSR	65536		100.0	100.0	98.9
S713	1	811	810	116.3	1.2	100.0
	2	460	918	115.7	0.7	100.0
	4	222	918	114.5	0.3	100.0
	8	113	972	114.0	0.2	100.0
	54	23	1188	114.3	0.1	100.0
	LFSR	65536		100.0	100.0	98.7
S820	1	1082	1081	144.9	2.8	100.0
	2	881	1840	155.8	2.3	100.0
	4	446	2047	155.5	1.2	100.0
	8	189	2162	154.3	0.5	100.0
	23	95	2162	151.3	0.2	100.0
	LFSR	38759		100.0	100.0	100.0
S832	1	987	966	142.2	4.8	100.0
	2	782	1633	151.7	3.8	100.0
	4	441	2024	155.1	2.1	100.0
	8	187	2139	153.9	0.9	100.0
	23	96	2185	151.6	0.5	100.0
	LFSR	20534		100.0	100.0	100.0
S838	1	1811	1809	120.9	2.8	100.0
	2	1332	2680	124.6	2.0	100.0
	4	1185	4958	136.1	1.8	100.0
	8	601	5052	134.9	0.9	100.0
	67	76	5052	132.2	0.1	100.0
	LFSR	65536		100.0	100.0	87.2
S953	1	1216	1215	124.5	2.8	99.9
	2	1189	2430	134.3	2.7	99.9
	4	628	2565	133.0	1.4	99.9
	8	316	2835	133.6	0.7	99.9
	45	77	3420	135.8	0.2	99.9
	LFSR	43434		100.0	100.0	99.9

表3　原始LFSR方式との比較
Table 3 Comparison of proposed method with other methods.

CUT	Test length							
	Proposed					Other Method		
	k=1	k=2	k=4	k=8	F.C.	LFSR1	LFSR2[6]	LFSR3[18]
C432	154	297	170	84	(100)	1331 (100)		
C499	334	258	231	186	(100)	1167 (100)		
C880	727	269	166	124	(100)	15024(100)		16384(99.9)
C1355	1415	848	771	361	(100)	1860 (100)		16384(99.7)
C1908	2306	1569	815	408	(99.5)	4917 (99.5)	6816(100)	16384(99.5)
C2670	10253	5105	2553	1277	(99.9)	65536(88.8)	2142(99.9)	65536(89.2)
C5315	3594	1412	1286	859	(100)	3814 (100)	1792(100)	32768(99.4)
C6288	55	33	34	29	(100)	60(100)	64 (100)	32768(99.5)
S208	210	163	97	55	(100)	4140(100)		2048(100)
S344	193	85	62	31	(100)	211(100)		2048(100)
S349	168	97	61	31	(100)	132(100)		4096(99.4)
S386	391	289	169	64	(100)	4109(100)		2048(100)
S420	634	528	313	169	(100)	65536(97.2)		65536(92.3)
S444	158	137	71	49	(100)	238(100)		16384(97.5)
S510	576	457	175	127	(100)	886(100)		2048(100)
S526	635	397	247	136	(100)	9970(100)		8192(99.1)
S641	756	433	222	115	(100)	65536(98.9)		16384(98.5)
S713	811	460	222	113	(100)	65536(98.7)		65536(93.8)
S820	1082	881	446	189	(100)	38759(100)		65536(100)
S832	987	782	441	187	(100)	20534(100)		32768(98.9)
S838	1811	1332	1185	601	(100)	65536(87.2)		131072(99.6)
S953	1216	1189	628	316	(99.9)	43434(99.9)		8192(99.6)

出できない確率は $(1-P)^l$ であり，最低 1 回は検出できる確率は $1-(1-P)^l$ となる．この場合 N 個の故障が最低1回は検出される確率は $\{1-(1-P)^l\}^N$ である．すなわちテスト長 l のランダムパターンで検出できる rpr 故障数の期待値は，式(2)で示される．

$$\{1-(1-P)^l\}^N \times N \qquad (2)$$

また，テスト中に m/k 回に 1 回 の頻度で出現する ATPG ベクトルによって検出される rpr 故障数の期待値は，[仮定 3]より式(3)で示される．式(3)中の「+1」は，初期値によって 1 個の rpr 故障を検出するものとして加えた．

$$(k \times l)/m + 1 \qquad (3)$$

式(2)，式(3)により，N 個の rpr 故障数とこれらを検出するテスト長 l の関係は，式(4)で示される．

$$\{1-(1-P)^l\}^N \times N + (k \times l)/m + 1 = N \qquad (4)$$

提案 TPG 方式(分割数 k=1)について，式(4)で示されるの理論式を用いてテスト長を求めた．式(1)～式(4)において，N を 5～50 の範囲と仮定して原始 LFSR 方式 TPG で生成した 2^{16} 個の擬似ランダムベクトルによる故障シミュレーション値から D_i を算出し理論値を求めた．

表 4 に理論値と故障シミュレーション値との比較を示す．N=10 のときが最も誤差は少なく，平均誤差率は 46.4%であった．

このように提案方式の TPG において，擬似ランダムベクトルによる故障シミュレーションを実行し，あらかじめ式(4)によってテスト長を概算することにより，分割数を決定する一つのガイドラインとなる．

6.　まとめ

本研究では，ＢＩＳＴ中の 100%の故障カバレージを達成するために，テスト長が短く，かつハードウェアオーバヘッドが少ないＴＰＧ方式を提案した．提案したＴＰＧ方式は，シフトレジスタに少量のＲＯＭを追加して構成する．ＲＯＭには，ＡＴＰＧツールで生成されたテストベクトルの一部を保存し，テスト生成に利用する．提案したＴＰＧ方式の各パラメータを決定するアルゴリズムも提案した．ベンチマーク回路を使用してテスト長の解析を行った．その結果，従来の原始 LFSR 方式のＴＰＧに比べて若干のハードウェアオーバヘッドが増えるものの，テスト長が大幅に縮小することが確認された．また，提案方式の TPG のパラメータを効率よく求めるための，テスト長の近似値に関する理論式を導いた．ROM に保存する ATPG ベクトルに関して，巡回置換以外の並べ替えも可能であるが，試みた範囲では著しい効果はなかった．効率のよい並べ替え法については今後の検討課題の一つで

表4　シミュレーションと理論値によるテスト長の比較
Table 4 Comparison of Theoretical test length with real test length

	(T−S)/S×100%　T :Test length of Theory　S: Test length of Simulation									
CUT ＼ N	5	10	15	20	25	30	35	40	45	50
C432	−6.5	110.4	222.7	313.0	370.8	406.5	426.6	439.0	447.4	451.3
C499	−51.8	5.4	54.5	93.4	124.3	148.2	142.5	127.2	118.0	111.7
C880	−67.0	−25.7	15.5	56.8	98.1	139.3	180.6	221.9	263.2	303.1
C1355	−88.4	−73.9	−59.4	−45.0	−30.8	−17.1	−4.3	7.1	17.2	26.1
C1908	−94.3	−87.1	−80.0	−72.8	−65.7	−58.5	−51.3	−44.2	−37.0	−30.0
C5315	−80.5	−59.6	−46.6	−38.5	−33.1	−29.1	−26.0	−23.5	−21.5	−19.8
S208	−63.8	−18.6	26.7	71.9	117.1	162.4	207.6	251.0	279.5	286.7
S298	−53.1	4.2	56.6	86.0	103.5	115.4	125.2	132.9	132.9	127.3
S344	−55.2	−15.5	4.6	14.9	20.1	21.1	21.1	21.1	21.1	21.1
S349	−48.2	−3.0	20.2	31.5	37.5	38.7	39.3	39.3	39.9	39.3
S382	−56.2	−5.1	35.5	64.5	84.8	99.1	107.4	108.8	110.1	111.1
S386	−86.7	−70.0	−53.5	−36.8	−20.2	−3.6	13.0	29.7	46.3	62.9
S400	−36.7	37.3	96.0	138.0	165.3	186.0	196.0	198.7	200.0	200.7
S420	−77.9	−50.3	−22.7	4.9	32.5	60.1	87.7	115.3	142.9	170.5
S444	−39.9	30.4	86.1	125.9	152.5	172.8	187.3	188.6	189.2	189.9
S510	−82.6	−61.1	−39.8	−19.4	−2.6	7.5	14.1	18.6	22.0	23.6
S526	−84.9	−66.0	−47.1	−28.2	−9.3	9.6	28.5	47.4	66.3	85.2
S641	−71.4	−35.7	0.0	35.7	71.4	107.1	142.9	178.4	212.2	230.6
S820	−91.5	−80.9	−70.2	−59.6	−49.0	−38.4	−27.7	−17.1	−6.5	4.2
S838	−85.2	−66.7	−48.2	−29.7	−11.2	7.3	25.8	44.3	62.8	81.3
S953	−85.2	−66.7	−48.2	−29.7	−11.2	7.3	25.8	44.3	62.8	81.3
Average	67.0	46.4	54.0	66.5	76.7	87.4	99.1	109.4	119.0	126.6

ある。

謝辞

本研究を通じて有益な御討論を頂いた三浦幸也助教授，また，ATPG ツールおよび故障シミュレータを提供頂いた九州工業大学梶原誠司助教授に深く感謝致します。

参考文献

[1] Konemann, B. Bennetts, N. Jarwala and B. N. Dostie, "Built-In Self-Test: Assuring System Integrity", IEEE Desighn & Test of Comput., pp. 39-45, Nov. 1996.
[2] B. T. Murray and J. P. Hayes, "Testing ICs: Getting to the Core of the Problem", IEEE Comput., pp. 32-38, Nov. 1996.
[3] J. M. Miranda, "A BIST and Bounary-Scan Economics Framework", IEEE Design & Test of comput., pp. 17-23, July/Sep. 1997.
[4] K. Iwasaki and H. Goto, "Exact expected test length generated by LFSRs for circuits containing hard random-pattern-resistant faults," IEICE Trans. Fundamentals, Vol. E81-A, No. 5, pp. 885-888, May 1998.
[5] M. Lempel, S. K. Gupta and M. A. Breuer, "Test Embedding with Discrete Logarithms", Proc. of VTS'94, pp. 74-80, 1994.
[6] H. Yokoyama, X. Wen and H. Tamamoto, "Random Pattern Testable Design with Partial Circuit Duplication", Proc. of ATS'97, pp. 353-358, 1997.
[7] C. Fagot, P. Girard and C. Landrault, "On Using Machine Learning for Logic BIST", Proc. of ITC'97, pp. 338-346, 1997.
[8] M. F. AlShaibi and C. R. Kime, "MFBIST: A BIST Method for Random Pattern Resistant Circuits", Proc. of ITC'96, pp. 176-185, 1996.
[9] S. Hellebrand, J. Rajski, S. Tarnick, S. Venkataraman and B. Courtois, "Built-In Test for Circuits with Scan Based on Reseeding of Multiple-Polynomial Linear Feedback Shift Resisters", IEEE Trans. on Comput., Vol. 44, No. 2, pp. 223-233, Feb. 1995.
[10] M.Bershteyn, "Caluculation of Multiple Set of Weights for Weighted RandomTesting", Proc. of ITC'93, pp. 1031-1040, 1993.
[11] J. Savir, "On Chip Weighted Random Patterns", Proc. of ATS'97, pp. 344-352, 1997.
[12] G. Kiefe and H. J. Wunderlich, "Using BIST Control for Pattern Genearation", Proc. of ITC'97, pp. 347-355, 1997.
[13] Michael Bershteyn, "Calculation of Multiple sets of Weights for Weighted random testing", Proc. of ITC'93, pp. 1031-1040.
[14] "A Ring Architecture Strategy for BIST Test Pattern Generation", Proc. of ATS'98, pp. 418-423, 1998.
[15] H. H. S. Gundlach and K. D. Muller Glaser, "On Automatic Testpoint Insertion in Sequential Circuits", Proc. of ITC'90, pp. 1072-1079, 1990.
[16] M. Nakao, K. Hatayama and I. Higashi, "Accelerated Test Point Selection method for Scan-Based BIST", Proc. of ATS'97, pp. 359-364, 1997.
[17] W. W. Peterson and E. J.Weldon Jr, "Error-Correcting Codes", The MIT Press, pp. 476-492, 1972.
[18] S. Wang and S. Gupta, "DS-LFSR: A New BIST TPG for Low Heat Dissipation", Proc. of ITC'97, pp. 848-857, 1997.

付録 A　ISCAS'85, ISCAS'89 ベンチマーク回路

ISCAS'85 および組み合わせ回路化した ISCAS'89 ベンチマークに関しての入出力数と検出可能故障数を示す。

表 A1　ISCAS'85, ISCAS'89 ベンチマーク回路

Table A1 ISCAS ' 85 and ISCAS ' 89 benchmark circuits.

CUT	Inputs	Outputs	Detectable Faults
C432	36	7	520
C499	41	32	750
C880	60	26	942
C1355	41	32	1566
C1908	33	25	1870
C2670	233	140	2630
C3540	50	22	3291
C5315	178	123	5291
C6288	32	9624	34
C7552	207	108	7419

CUT	Inputs	Outputs	Detectable Faults
S208	19	10	215
S298	17	20	308
S344	24	26	342
S349	24	26	348
S382	24	27	399
S386	13	13	384
S400	24	27	418
S420	35	18	430
S444	24	27	460
S510	25	13	564
S526	24	27	554
S641	54	43	467
S713	54	42	543
S820	23	24	850
S832	23	24	856
S838	67	34	857
S953	45	52	1079