システムＬＳＩ設計技術　107－15
(2002. 11. 28)

符号化技術を用いた多重スキャン設計のテストデータ量削減について

谷口　謙二郎[1], 宮瀬　紘平[1], 梶原　誠司[1,2], イリス　ポメランツ[3], スダカー　M.レディー[4]

1: 九州工業大学情報工学部電子情報工学科　〒820-8502 福岡県飯塚市川津 680－4
2: 九州工業大学マイクロ化システムセンター

3: School of Electrical and Computer Engineering, Purdue University

4: Electrical and Computer Engineering Department, University of Iowa

E-mail:　1 {taniguchi, miyase}@aries30.cse.kyutech.ac.jp, 1,2 kajihara@cse.kyutech.ac.jp

あらまし　本論文では，多重スキャン設計に対するテストデータ量削減手法を提案する．提案手法は２段階でデータ圧縮を実現する．最初に，ATPG で生成したテスト集合を符号化し，多重スキャンに印加するスキャンインのテスト入力数を削減する．次に，符号化したテスト集合のデータ量を，統計符号化技術によりさらに削減し，それぞれのテストピンに印加するスキャンインベクトル長を削減する．統計的符号化ではハフマン符号を用いる．スキャンインベクトル長の削減により，テストロード時間とテストデータ量が削減される．ISCAS-89 ベンチマーク回路に対する実験結果では，提案手法が，圧縮テスト集合のテストデータ量を平均 21.5%に削減できることを示す．

キーワード　テストデータ圧縮，ドントケア判定，多重スキャン設計，ハフマン符号化，ＳｏＣテスト

Test data volume reduction using statistical encoding for multiple scan chain designs

Kenjiro Taniguchi[1], Kohei Miyase[1], Seiji Kajihara[1,2], Irith Pomeranz[3], Sudhakar M. Reddy[4]

1: Department of Computer Sciences and Electronics, Kyushu Institute of Technology

680-4 Kawazu, Iizuka, Fukuoka, 820-8502 Japan

2: Center for Microelectronics Systems, Kyushu Institute of Technology

3: School of Electrical and Computer Engineering, Purdue University

4: Electrical and Computer Engineering Department, University of Iowa

E-mail:　1 {taniguchi, miyase}@aries30.cse.kyutech.ac.jp, 1,2 kajihara@cse.kyutech.ac.jp

Abstract In this paper we propose a new method of test data compression for multiple scan chain designs. The proposed method consists of two phases of data compression. In the first phase, ATPG test vectors applied to multiple scan chains are encoded to reduce the number of test input pins and thus reduce the test data volume. In the second phase, the encoded test vectors are compressed further using statistical encoding to reduce the length of the test sequences applied to each test pin. This reduces test loading time and test data volume. Experimental results for large ISCAS-89 benchmark circuits show that the proposed method reduced the test data volume to 21.5% on average.

Keyword test data compression, don't-care identification, multiple scan chain designs, Huffman's encoding, SoC testing

1. はじめに

SoC デザインの複雑化により，SoC のテストに要するコストは増大している．テストコストを削減するには，テストデータ量を減らし，テスト時間を短縮することが重要になる．

テストデータ量とテスト時間を削減するためのアプローチとして，テストベクトル数が少ないテスト集合を生成する手法が研究されてきた[1,2]．被検査回路に印加するテストベクトル数が少なくなれば，テストデータ量とテスト印加時間を同時に削減できる．しかし，テストベクトル数を少なくするにも故障検出率は維持されなければならないので，テストベクトル数を少なくすることによるテストデータ量とテスト印加時間の削減には限界があり，この手法だけでは不十分である．そのため，テスト容易化設計を利用した研究が行われている．多重スキャンを使うことはテスト印加時間を削減する有効な手法である．しかしながら，多重スキャン設計だけではテストデータ量は削減できない．テストデータ量を削減するために,テストデータを符号化する手法が提案されている[3-6]．符号化技術を用いた手法は，テストデータをテスタからチップに印加する際に必要な入力ピンの数を削減する．もしくは，スキャンインベクトル長を削減する（それにより，テストデータロード時間を削減する）.実際にテストを行う際は，符号化したテスト集合をチップ上のデコンプレッサで復号化し，テスト集合を被検査回路に印加する．多重スキャン設計に対して，[5]と[6]の符号化手法はテストデータ量を削減するが，テストロード時間は削減しない．[3]と[4]等の手法はテストデータ量とテストロード時間を削減するが，多重スキャン設計を考慮していない．

本論文では，多重スキャン設計に対するテストデータ量削減手法を提案する．提案手法は2段階のデータ圧縮で構成される．最初に,多重スキャンに印加するATPG テスト集合を符号化し，テスト入力ピンの数を削減する．次に，符号化したテスト集合は，統計符号化技術を用いて，データ量を削減し，それぞれのテストピンに印加するスキャンインベクトル長を削減する．符号化には統計的符号化の一種であるハフマン符号を用いる．スキャンインベクトル長を削減することにより，テストロード時間とテストデータ量を削減する．ISCAS-89 ベンチマーク回路に対する実験結果では，提案したテストデータ量削減手法は，テスト集合のテストデータ量を平均 21．5%に削減できた．

本論文は，次のように構成される．2 章では，関連手法を説明する．3 章では，処理手順の概要，4 章では提案手法の処理手順とその例を示す．5 章では，実験結果を示し，最後に 6 章で本論文をまとめる．

2. 関連研究

まず，提案手法で用いるテストデータ量削減手法について，説明する．

2.1. 多重スキャン設計に対するデータ量削減[6]

図 1 のような，4 本の多重スキャンチェーンにスキャンインする 4 つのテストベクトルを考える．図 1 の t_i はテストベクトルを表し，s_j はスキャンインベクトルを表す．スキャンインベクトルの種類（ビットパターンの数）を NDO (number of distinct output pattern) とすると，そのスキャンインベクトルは，$\lceil \log_2 NDO \rceil$ ビットあれば区別できる．つまり，図 1 のようなテスト集合のスキャンインベクトルは，スキャンインベクトルのビットパターンが 7 種類なので,$\lceil \log_2 7 \rceil$ ビット，すなわち 3 ビットの符号で表すことができる．3 ビットの符号を割り当てると,たとえば表 1 のようになり，図 1 のテストベクトルは図 2 のように圧縮できる．

2.2. ハフマン符号化

テスト集合が与えられると，各テストベクトルは n ビットのブロックに分けられ，そしてそれぞれのブロックは可変長のバイナリー符号語に写像される．例えば図 3 のようなテスト集合が与えられたとき，そのテスト集合を 4 ビットのブロックに分けると，図 4 のようなる．その 4 ビットのブロックの出現頻度を求め，出現頻度の多いブロックからハフマンのアルゴリズムに従って短い符号を与えていくと，表 2 のようになる．

t_4 ..... t_1

s_{12} ..... $s_3 s_2 s_1$

0 1 0 0 0 1 0 0 1 0 1 1

1 0 0 0 1 1 1 1 0 1 0 1

0 0 1 1 1 1 1 1 1 1 1 1

0 1 1 1 0 0 0 0 1 0 1 1

図1: 与えられたテスト集合

表1: テスト系列への符号割り当て

テスト系列	符号
1111	000
1011	001
0110	010
1110	011
0011	100
1001	101
0100	110

1 1 1 1 1 0 0 0 0 0 0 0

1 0 0 0 1 1 1 1 0 1 0 0

0 1 0 0 0 1 0 0 1 0 1 0

図2: 圧縮されたテスト集合

110011010011010010011001
111101001010100111110100
010101000101001100111010
111101011101001111010100

図3:与えられたテスト集合

1100	1101	0011	0100	1001	1001
1111	0100	1010	1001	1111	0100
0101	0100	0101	0011	0011	1010
1111	0101	1101	0011	1101	0100

図4:4ビットで分けられたテスト集合

表2:テストデータのハフマン符号化

ブロック	出現頻度	ハフマン符号
0100	5	00
0011	4	111
0101	3	110
1001	3	100
1101	3	101
1111	3	011
1010	2	0101
1100	1	0100

この符号化によって得られたビット数を符号化前のテスト集合のビット数と比べると，符号化前の6×4×4＝96ビットに対し，符号化後は（5×2）＋（4×3）＋（3×3×4）＋（2×4）＋（1×4）＝70ビットになる．この符号化によってテストデータ量は26ビット削減できる．

3. 提案手法の概要

本論文で提案するテストデータ量削減手法は，前節で述べた2つのテスト圧縮法を組み合わせて構成する．以下，提案手法の概要を説明する．なお，提案手法で対象とするのはテストベクトル中のスキャンインデータのみである．与えられたテスト集合を T_0 とする．図

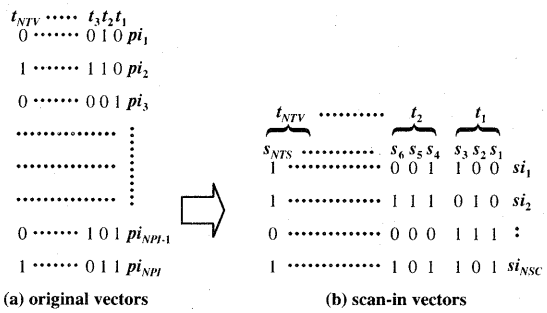

図5: 与えられたテスト集合 T_0

5中の *NPI* を擬似外部入力の数（すなわち，回路のスキャンフリップフロップ数）とし，*NTV* をテストベクトルの数とする．提案手法は，多重スキャン設計に対するデータ圧縮であるので，実際に被検査回路に印加されるテストデータは，図5(b)のようになる．回路のスキャンチェーンの数を *NSC* とし，スキャンイン入力に印加されるスキャンインベクトル数を *NTS* とする．*NTS* は *NTV***NPI*/*NSC* の式により得られる．

図6に提案手法によるテスト圧縮の工程を示す．全ての入力値が0か1であるテスト集合 T_0 が与えられると，まず単一縮退故障に対する故障検出率を下げることなく，T_0 中の論理値をできるだけ多く不定値(X)に変える[7]．それぞれの X に対して[6]の手順を基に適切な論理値を割り当てる．その結果得られたテスト集合を T_1 とする．次に[6]の手順により T_1 をテスト集合 T_2 に符号化する．図6(d)では，符号化により削減されたスキャンインベクトルのビット幅を *NSI*(number of scan-in inputs)としている．次に，ハフマン符号化を用いて T_2 の各入力ピンに印加するスキャンインベクトルを符号化して，最終的なテスト集合 T_3 を得る．ハフマン符号化は，ベクトルでなく系列を圧縮する（空間方向の圧縮でなく，時間方向の圧縮となる）ので，

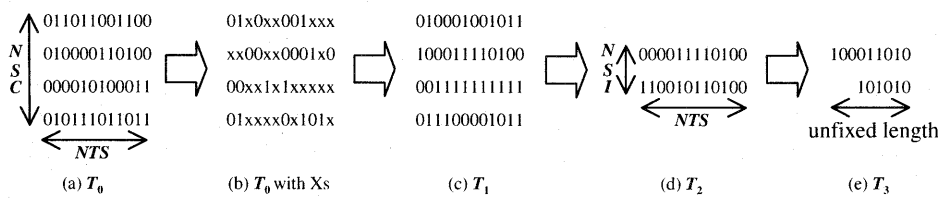

図6: テストデータ圧縮工程

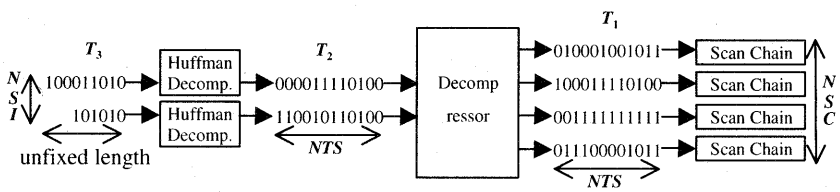

図7: 多重スキャン設計に対するテスト集合の印加法

最終的に得られる T_3 の各入力に対する系列長は，同じにならない．

圧縮したテスト集合 T_3 を印加して，実際にテストを行う場合は，図7に示すように，チップ上の2つのデコンプレッサを使って圧縮したテスト集合を復号する必要がある．

4. 提案手法の詳細

ここでは，上記2つのデータ量削減手法の組み合わせ効率を上げるための手法を述べる．

第1段階の圧縮（図6の T_1 から T_2 への圧縮）では，スキャンチェーン本数に対して，スキャン入力数を減らすことを目的とした．復号のためのデコンプレッサのことを考えなければ，スキャンインベクトルに対する符号割当は自由に行える．そこで，第1段階で符号化する際の符号割当を，第2段階の統計符号化（図6の T_2 から T_3 への圧縮）の効率が高まるように行うことで，提案手法全体として高い圧縮効果を得ることができる．

ハフマン符号化による圧縮率を高める条件として，n ビットのブロックに分割したときに現れるブロックパターンの種類が少ないこと，および，最大の出現頻度を持つ n ビットブロックパターンの出現頻度がより大きいことである．まず，T_0 から T_1 へ変換するとき，特定のスキャンインベクトルが多く現れるように，Xに論理値を割り当てる．そのため，いくつかの出現頻度の高いスキャンインベクトルと，その他の出現頻度の低いスキャンインベクトルで，テスト集合 T_1 を構成する．よって，最も出現頻度の高いスキャンインベクトルに全て1か0のビット符号を割り当て，残りのスキャンインベクトルについては出現頻度の高いものから順に，最も出現頻度の高いものに割り当てた符号からハミング距離が近い符号を割り当てる．そうすると，ハフマン符号化する際のビットブロックにも全て1か0，またそのハミング距離が近いビットブロックの出現頻度が高くなり，圧縮の効果が高まる．

提案手法を用いて符号化する例を，図を用いて示す．図8のテスト集合は，図6(c)に対応するテスト集合である．各パターン下部に表記している英文字は，スキャンインベクトルの種類を表している．図8のテスト集合は6種類（NDO=6）のスキャンインベクトルで構成されているので，それらは，3ビット（$\lceil \log_2 6 \rceil$ ビット）の符号で識別できる．次に各スキャンインベクトルの出現頻度を求める．この例では，ハフマン符号化する際のブロックのビット数は3である．最も出現頻度が大きいスキャンインベクトルには，000 が割り当てられる．次に，残りのスキャンインベクトルに出現頻度の高いものから順に，最も出現頻度の高いスキャンイ

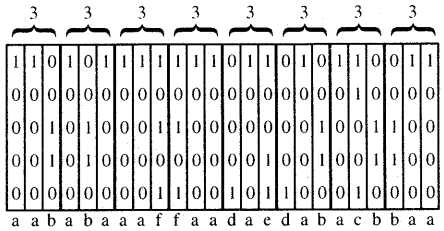

図8：テスト系列のパターン数を最小化したテスト集合

表3：テスト系列のパターン

		ビット数		
パターン	頻度	1	2	3
a	13	3	6	4
b	5	3	1	1
d	2	0	0	2
f	2	1	0	1
e	1	1	0	0
c	1	0	1	0

表4：3ビットの符号の割り当て

パターン	3ビットパターン
a	000
b	001
d	010
f	100
e	101
c	110

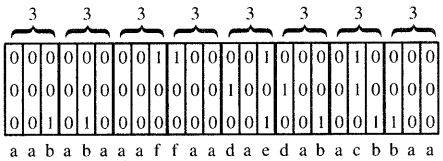

図9：3ビットの符号を割り当てたテスト集合

000	000	001	100	001	000	010	000
000	000	000	000	100	100	010	000
001	010	000	000	001	001	001	100

図10：3ビットブロックに分けたテスト集合

表5：ブロックパターンのハフマン符号化

パターン	出現頻度	ハフマン符号
000	11	0
001	6	10
100	4	110
010	3	111

ンベクトルのビット符号からハミング距離が近い符号を割り当てる．表4に，どのスキャンインベクトルにどの符号を割り当てるかを示す．第1段階の符号化の結果，図9のテスト集合を得る．次に図9のテスト集合を3ビットのブロックで区切ると，図10のようになる．そのブロックの種類と出現頻度を求め，出現頻度の大きいブロックから順に短い符号を割り当てると，表5のようになる．この符号化によって得られたビット数を符号化前のテスト集合のビット数を比較すると，符号化前の120ビットに対し，符号化後は，(11×1)+(6×2)+(4×3)+(3×3)=44 ビットになる．この符号化によってテストデータ量は，76ビット削減できる．

5. 実験結果

提案手法を AthlonXP1800+MHZ，メモリ 256MB，OS FREEBSD4.5 の計算機上で C 言語により実装し，ISCAS'89 ベンチマーク回路に対して実験を行った．テスト集合は[1]のテスト生成の手順で生成される．ハフマン符号化で用いるブロックのサイズは，8 ビットである．表6に実験結果に示す．表のパラメータの *NTV*, *NPI*,*NSC*,*NSI* は，すでに定義したものと同一である．"bits"の下の2つの欄は，テスト集合 T_2 と T_3 それぞれのビットの総数を表す．"%"の下の2つの欄は，*NTV***NPI* で求まる元のテスト集合のビット数に対する，圧縮したテスト集合 T_2 と T_3 の比率を示す．最後の欄は，[6]の手順を用いてテスト集合 T_1 を導いた後から，T_3 を導くまでの実行時間である．統計符号を用いなかった（テスト集合 T_2 に対応する）場合は，元のテストデータ量の 42.6%までしか削減できなかったのに対して，提案手法は平均で元のデータ量の 21.5%までテストデータ量を削減できた．また例として，s35932 (*NSC*=24)のテストデータ量は，統計符号を用いない場合が，元のテストデータ量の 29.2%にしか削減できないのに対して，統計符号を用いた場合は，元のテストデータ量 7.8%に削減できた．

表6:提案手法の実験結果

circuit	*NTV,NPI*	*NSC*	*NSI*	#bits		compression ratio(%)		time(sec)
				T_2	T_3	T_2	T_3	
s5378	*NTV*=100	8	5	11500	6988	64.2	39.0	0.02
	NPI=179	16	6	7200	5340	40.2	29.8	0.02
		24	7	5600	4607	31.3	25.7	0.01
		32	8	4200	3692	23.5	20.6	0.01
s9234	*NTV*=111	8	6	19314	11703	76.3	46.2	0.06
	NPI=228	16	7	11655	8494	46.1	33.6	0.03
		24	7	7770	6604	30.7	26.1	0.02
		32	8	7104	6169	28.1	24.4	0.02
s13207	*NTV*=235	8	4	78960	13696	50.2	8.7	3.35
	NPI=669	16	5	49350	14505	31.4	9.2	0.76
		24	5	32900	18116	20.9	11.5	0.28
		32	6	29610	17611	18.8	11.2	0.14
s15850	*NTV*=97	8	5	36375	15606	62.8	26.9	0.35
	NPI=597	16	6	22116	15000	38.2	25.9	0.07
		24	7	16975	11966	29.3	20.7	0.04
		32	8	14744	10511	25.5	18.2	0.03
s35932	*NTV*=12	8	7	18144	3906	87.5	18.8	0.03
	NPI=1728	16	8	10368	4017	50.0	19.4	0.01
		24	7	6048	1621	29.2	7.8	0.01
		32	8	5184	3160	25.0	15.2	0.01
s38417	*NTV*=87	8	6	107010	21788	75.2	15.3	2.92
	NPI=1636	16	8	71688	31322	50.4	22.0	0.69
		24	9	54027	39021	38.0	27.4	0.26
		32	9	40716	36413	28.6	25.6	0.16
s38584	*NTV*=114	8	5	103740	18050	62.7	10.9	5.14
	NPI=1452	16	6	62244	21794	37.6	13.2	0.89
		24	7	48678	29962	29.4	18.1	0.35
		32	8	41952	27758	25.3	16.8	0.17
						42.6	21.5	

6. 結論

本論文では，テスト入力ピン数を削減とそれぞれのテスト入力ピンのスキャンインベクトル長を短くする統計符号を用いたテストデータ量削減の手法を提案した．ISCAS-89 ベンチマーク回路に対する実験結果は，提案手法がテストデータ量を効果的に削減していることを示している．

文　献

[1] S.Kajihara, I. Pomeranz, K. Kinoshita and S. M. Reddy, "Cost-Effective Generation of Minimal Test Sets for Stuck-at Faults in Combinational Logic Circuits," IEEE Trans. CAD, pp.1496-1504, Dec. 1995.

[2] I. Hamzoaglu, and J. H. Patel, "Test Set Compaction Algorithms for Combinational Circuits," Proc. Int'l Test Conf, pp.283-289, Oct. 1998.

[3] A.Jas, J.Ghosh-Dastidar and N. A. Touba, "Scan Vector Compression/Decompression Using Statistical Coding," Proc. VLSI Test Symp, pp.114-120, 1999.

[4] A. Chandra and K. Chakrabarty, "Frequency-directed Run-length (FDR) Codes with Application to System-on-a-chip Test Data Compression," Proc. VLSI Test Symp, pp.42-47, April 2001.

[5] I. Bayraktaroglu and A. Orailoglu, "Test Volume and Application Time Reduction Through Scan Chain Concealment," Proc. Design Automation Conference, pp.151-155, June 2001.

[6] S. M. Reddy, K. Miyase, S. Kajihara and I. Pomeranz, "On Test Data Volume Reduction for Multiple Scan Chain Designs," Proc. 20th IEEE VLSI Test Symposium, pp.103-108, April 2002.

[7] S. Kajihara, K. Miyase, "On Identifying Don't Care Inputs of Test Patterns for Combinational Circuits," Proc. ICCAD-2001,pp.364-369, Nov.2001.