キャプチャセーフテストベクトルを用いた 低消費電力指向テスト生成法

日大生産工(院) 〇平井 淳士 日大生産工 細川 利典

日大生産工 山内 ゆかり 日大生産工 新井 雅之

1. はじめに

近年,半導体微細化技術の発達に伴い,大規模集 積回路(Large Scale Integrated Circuits: LSI)テス トにおける実速度スキャンテストは必要不可欠な 技術となっている[1]. 一般に実速度スキャンテスト の消費電力は、LSI が通常動作する際の消費電力と 比較して大きくなることが知られている[2,3].実速 度スキャンテストにおける消費電力は, テストベク トルをスキャンチェインに印加するシフトイン動 作と,組合せ回路部のテスト応答をスキャンアウト から出力するシフトアウト動作により発生するシ フト電力と, テストベクトルに対する組合せ回路部 のテスト応答をスキャン FF に取り込む際,スキャ ン FF の出力の論理値が遷移することで発生するキ ャプチャ電力に分類できる.本論文ではキャプチャ 電力に着目する.キャプチャ電力の増大は,瞬間的 に過度の IR ドロップ[2]を引き起こし, FF の誤動作 や遅延の増大により誤テストの原因となる. キャプ チャ電力削減のための手法は多数提案されており, それらは一般に、回路構造変更による手法[4,5]と、 テストデータ変更[6-9]による手法類することがで きる.回路構造変更による手法では、テスト対象回 路に対してテスト容易化のための回路構造変更を 行う方法[4]や、テスト専用の回路を付加する方法 [5]などがある.一方で,テストデータ変更による手 法はドントケア(X)割当てによる手法[6,7]と再テス ト生成による手法[8,9]に分類される.X割当て手法 としては,正当化や含意操作などの決定論的アルゴ リズムを用いてテストベクトル中のXビットに適切 に 0 または 1 を割当てることでキャプチャ時の FF における論理値遷移数を抑制する LCP (Low Capture Power)-Fill 手法[6]などが挙げられる. ま た,低キャプチャ電力テスト生成手法では,従来の 故障検出重視のテスト生成手法により生成された

テスト集合内の高消費電力テストベクトルに対し て、消費電力制約を満たすようにテストベクトルの 再生成を行う.しかしながら、これらの手法の多く は決定論的アルゴリズムを採用しているため、大規 模な LSI に対するテスト生成に要する時間が課題と なる.本論文では、シンプルな故障シミュレーショ ンを用いたアルゴリズムを採用した低キャプチャ 電力テスト生成手法を提案する.提案手法では,低 キャプチャ電力テストベクトルを利用することで 決定論的な操作を必要とせず、新たに低キャプチャ 電力テストベクトルを生成する.本論文の構成は以 下の通りである. 第2章では, 提案手法の根拠とな る予備実験の結果を示す. 第3章では,提案手法の テスト生成アルゴリズムを示す.第4章に ISCAS'89, ITC'99 ベンチマーク回路を用いた実験結果を示し, 第5章では本論文のまとめと今後の課題について示 す.

2. 予備実験

A. 低キャプチャ電力テスト生成問題

本論文では、キャプチャ電力を見積もる方法として、重み付き信号遷移確率(Weighted Switching Activity: WSA)[7]を採用する.以下に WSA 値を求 める式を示す.

$$WSA(v_j) = \sum_{i=1}^{G} tran(g_i) \times (1 + fanout(g_i))$$

上記式において、 $WSA(v_j)$ はテストベクトル v_j の WSA 値を表し、テストベクトル v_j を回路に印加した 際の内部信号線における論理値の遷移数を見積も る. Gは回路内の総信号線数を表し、 $tran(g_i)$ はテス トベクトル v_j を回路に印加した際に、ゲート g_i の出 力で論理値の遷移が発生した場合には 1、遷移が発 生しない場合には 0 を返す関数である.また、 $fanout(g_i)$ はゲート g_i の出力のファンアウト数を表

A Low-Capture-Power Test Generation Method Using Low Capture-Power Test Vectors

Atsushi HIRAI, Toshinori HOSOKAWA, Yukari YAMAUCHI and Masayuki ARAI

す. このようにして WSA では, 各テストベクトル に対するキャプチャ電力の見積りを算出する.

実速度スキャンテストにおいて,キャプチャ電力 が極めて高いテストベクトルは,不必要に歩留り損 失を引き起こす恐れがあるためテストに使用する ことができない. そのため、キャプチャ電力が閾値 を超えるようなテストベクトルは低キャプチャ電 カテスト生成の技術を用いて再テスト生成を行う 必要がある.本論文では、キャプチャ電力が閾値以 下のテストベクトルをキャプチャセーフテストベ クトル[9]、キャプチャ電力が閾値を超えるようなテ ストベクトルをキャプチャアンセーフテストベク トル[9],初期テスト集合中において、キャプチャア ンセーフテストベクトルでのみ検出可能な故障を アンセーフ故障,アンセーフ故障以外の検出故障を セーフ故障と定義する. そのため,本論文では低キ ャプチャ電力テスト生成問題を以下のように定式 化する.

(低キャプチャ電力テスト生成問題)

入力:初期テスト集合 T
出力:WSA 値がP_{th}以下の最終テスト集合 T'
制約:WSA 値閾値P_{th}
最適化:アンセーフ故障数の最小化

B. テストベクトル操作による消費電力の変化

図1に予備実験の手順を示す.図1に示すように、 ランダムに生成された入力ベクトル(Step1)に対し、 指定された割合のビットを反転(ベクトル操作)する ことにより、ベクトル操作前とベクトル操作後の WSA 値の変化について解析を行った(Step2-6).本 論文ではベクトル操作前とベクトル操作後のそれ ぞれのベクトルに対する WSA 値の相関係数を算出 し、評価した.相関係数の算出式は以下のとおりで ある.

 $CC = \frac{\sum_{i=1}^{n} (before_i - \overline{before})(after_i - \overline{after})}{\sqrt{\sum_{i=1}^{n} (before_i - \overline{before})^2} \sqrt{\sum_{i=1}^{n} (after_i - \overline{after})^2}}$

上記式において, n はランダムベクトル数を表し, before_iとafter_iはベクトル操作前とベクトル操作後 のベクトル i の WSA 値を表し, beforeとafterはベ クトル操作前とベクトル操作後の WSA 値の平均を 表す.本論文では, ランダムベクトル数を 1024, ベ クトル操作の割合を 5, 10, 20, 30%として予備実 験を行った.予備実験結果を表 1 に示す.表 1 に示 すように, ベクトル操作するビットの割合が小さい 場合(5%や 10%), ベクトル操作前とベクトル操作後 の WSA 値の相関係数は 1.0 に近い値を示し, 非常 に強い正の相関があることがわかる.

	C : Circuit
4	analysis(C){
	1. V = generate_random_vectors();
	2. for each vector v_i in V_i
1	B. $before_i = WSA(C, v_i);$
4	4. $v'_i = invert_fixed_bit(v_i);$
4	5. $after_i = WSA(C, v_i');$
	5. }
	7. return correlation_cofficient(BEFORE, AFTER);
	1

図 1. 予備実験 実験手順

表 1. 予備実験結果

	# of	correlation coefficient(CC)									
Circuit		5%	10%	20%	30%						
	F1+FF1	inverted	inverted	inverted	inverted						
s5378	214	0.82	0.64	0.39	0.25						
s9234	247	0.85	0.71	0.43	0.26						
s13207	700	0.85	0.74	0.48	0.33						
s15850	611	0.83	0.69	0.45	0.25						
s35932	1763	0.86	0.75	0.54	0.34						
s38417	1664	0.81	0.62	0.39	0.21						
s38584	1464	0.88	0.79	0.57	0.35						
b14	277	0.91	0.84	0.75	0.73						
b15	485	0.77	0.66	0.38	0.22						
b17	1452	0.78	0.60	0.37	0.20						
b18	3357	0.81	0.64	0.45	0.26						
b19	6666	0.80	0.67	0.46	0.29						
b20	522	0.80	0.69	0.53	0.42						
b21	522	0.80	0.68	0.53	0.41						
b2.2	767	0.80	0.70	0.50	0.38						

予備実験結果より、本論文で提案する低キャプチャ電力テスト生成手法では、キャプチャセーフテストベクトルを利用し、その一部のビット(5%~10% 程度)を操作することにより、他の故障を検出するキャプチャセーフテストベクトルを新たに生成する アルゴリズムを採用する.

3. 提案手法

本論文で提案する低キャプチャ電力テスト生成 手法では、テストベクトルを故障励起のための割当 てと、故障伝搬のための割当てに分けて考える.図 2に提案手法の概念図を示す.図2の例では、対象 故障 fを励起するためのテストキューブ(X,1,X,0,X) と故障伝搬のためのテストキューブ(1,1,X,X,0)を組 み合せることで対象故障 fを検出するテストキュー ブ(1,1,X,0,0)を生成している.提案手法では、故 障励起のための(疑似)外部入力割当てをキャプチャ アンセーフテストベクトルから経路追跡法により 抽出[10]し、故障伝搬のための(疑似)外部入力割当 としてキャプチャセーフテストベクトルを利用す る.これは、故障励起のための割当て(疑似)外部入 力割当て数は多くても全外部入力数の 10%程度で あり、予備実験の結果からもわかるとおり、利用す る低キャプチャ電力テストベクトルの消費電力特 性を崩すことなく,新しいテストベクトルを生成で きると考えるためである.



図 2. 提案手法 概念図

C. 提案手法アルゴリズム

提案手法では、キャプチャアンセーフテストベク トルの故障励起割当てと,キャプチャセーフテスト ベクトルを利用し、テストベクトル操作を実行する ことで、対象故障に対するテスト生成を実現する. 図 3 は提案テスト生成の全体アルゴリズムである. 提案手法では,初期テスト集合 Tから同定されたキ ャプチャセーフテスト集合Tsafeとキャプチャアンセ ーフテスト集合Tunsafeを入力としてテスト生成を実 行する.はじめに、故障シミュレーションを用いて テスト生成対象(アンセーフ)故障の同定(Step1)を 行い, すべてのアンセーフ故障に対して再テスト生 成(Step2-6)を実行する. 再テスト生成では, キャプ チャアンセーフテストベクトルの故障励起割当て と、キャプチャセーフテストベクトルのテストベク トル合成にもとづくテスト生成と,動的テスト圧縮 を実行する. 全アンセーフ故障に対する再テスト生 成が終了した後,低消費電力を指向した静的テスト 圧縮(Step7)を実行し,再生成後のテストベクトルと キャプチャセーフテスト集合の和集合を求め,最終 テスト集合 T'を取得する.

<i>C</i> :	Circuit
T_{sc}	ife : Capture Safe Test Set
T_u	nsafe : Capture Unsafe Test Set
tesi	generation(C, T _{safe} , T _{unsafe}){
1.	F _{target} = target_fault_selection(C, T _{safe} , T _{unsafe});
2.	for each fault f _i in F _{target} {
3.	$v' = synthesis_based_test_generation(C, T_{safe}, T_{unsafe}, f_i);$
4.	$v' = dynamic_test_compaction(C, v', T_{unsafe});$
5.	$T_{gen} = T_{gen} U v';$
6.)
7.	T _{comp} = static_test_compaction(C, T _{gen});
8.	$T' = T_{comp} \cup T_{safe};$
}	

図 3. 提案手法全体アルゴリズム

D. 対象故障選択

テスト生成対象故障の選択方法について説明する. 表 2 に対象故障選択の例を示す.表 2 は,初期テス ト集合 $T=\{tp1, tp2, tp3, tp4, tp5\}$ に対して故障シミ ュレーションを実行した結果である. f1~f10 は Tの検出故障である. tp1 と tp2 はキャプチャアンセ ーフテストベクトルである.提案手法では,キャプ チャアンセーフテストベクトルでのみ検出可能な 故障は, $F_{target}=\{f1, f2\}$ となる.

表 2. 対象故障選択

WSA	Т	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10
unsafe	tp1	•	•	0							
unsafe	tp2		•	0	0						
safe	tp3			0	0	0	0				
safe	tp4						0	0	0		
safe	tp5								0	0	0

E. テストベクトル合成にもとづくテスト生成

図4にテスト生成アルゴリズムを示す.テストベ クトル合成では、対象故障 ftarget の故障影響を励起 し,出力側の最も近いファンアウトまで故障伝搬す るためのテストキューブ texをキャプチャアンセー フテストベクトルから同定(Step1)し、ftargetに対し て利用可能な全キャプチャセーフテストベクトル に対し Step2 から Step9 の処理を繰り返す.本論文 において利用可能なテストベクトルとは,対象故障 ftaraetに対して出力側の最も近いファンアウトの故 障影響を(疑似)外部出力に伝搬可能なキャプチャセ ーフテストベクトルとする. テストベクトルの選択 に成功した場合(Step2), Step1 で同定した故障励起 のためのテストキューブ t_{ex} と, Step2 で選択された テストベクトルtbaseに対し、表3に示す合成規則を 用いてテストベクトル合成を実行し、得られたテス トベクトルをt_{gen}とする(Step3). 図 5 にテストベク トル合成の実行例を示す.図5では、対象故障ftaraet を励起するテストキューブtex = (X,1,0,X,X,X)と利用 するテストベクトルtbase = (1,0,1,1,0,0)を表 3 の合 成規則にもとづき合成することにより、新たなテス トベクトル $t_{gen} = (1, 1, 0, 1, 0, 0)$ を生成している. その 後, t_{aen}に対し WSA 値の算出(Step4)と故障シミュ レーション(Step5)を実行し、 t_{gen} がキャプチャセー フテストベクトルであり、かつ対象故障 f_{target} を検 出可能なとき、対象故障 f_{target} を検出するテストベ クトルとしてtgenを返す. Step2 から Step9 までの 処理を終了後,対象故障f_{target}を検出可能なキャプ チャセーフテストベクトルが生成されない場合, テ スト生成に失敗したとしてφを返す(Step10).

C : (Circuit
Tsa	fe : Capture Safe Test Set
T _{un}	safe : Capture Unsafe Test Set
f_{tar}	aet : Target Fault
synt	hesis_based_test_generation(C, T _{safe} , T _{unsafe} , f _{target}){
1.	t _{ex} = get_fault_excitation_cube(T _{unsafe} , f _{target});
2.	while((t _{base} = select_reuse_test_vector(T _{safe} , f _{target})) exists){
3.	$t_{gen} = test_vector_synthesis(t_{base}, t_{ex});$
4.	WSA(C, t_{gen});
5.	fault_simulation(C, t _{gen} , f _{target});
6.	if(t_{gen} is capture-safe && t_{gen} detects f_{target}){
7.	return t _{gen} ;
8.	}
9.	}
10.	return φ;
}	

図 4. テストベクトル合成 アルゴリズム

4. 実験結果

提案手法を C 言語で実装し ISCAS'89 ベンチマー ク回路を対象として Synopsis 社のテスト生成ツー ル TetraMAX の低消費電力モードを用いたテスト 生成結果との比較を行った.実験環境には 3.4GHz CPU, 8GBメモリを用いた.本論文において,キャ プチャセーフ判定のための閾値は,初期テスト集合 における最大 WSA 値の 70%と 90%に設定した.表 4に実験結果を示す.表4において,"Initial Test Set" は初期テスト集合の最大 WSA 値とアンセーフ故障 数を示し, "Number of Unsafe Faults"は再テスト 生成後のアンセーフ故障数を, "Test Generation Time"は再テスト生成の実行時間を示している.表 4の結果より、提案手法の実験結果ではアンセーフ 故障数を平均 99%削減できたのに対し、TetraMAX では平均19%削減できている.また,再テスト生成 の時間では,提案手法の方が平均3倍高速に再テス ト生成を実行していることがわかる.

5. まとめ

本論文では、シミュレーションベースの低キャプ チャ電力テスト生成手法を提案した.提案手法では、 初期テスト集合内のキャプチャセーフテストベク トルを利用することにより、従来の低キャプチャ電 カテスト生成手法と比較して74%多く、アンセーフ 故障数を削減することに成功した.また、提案手法 では初期テスト集合内の利用可能な情報を利用し てテスト生成を実行するため、複雑な処理を必要と せず、シンプルな実装と高速なテスト生成を実現し た.

今後の課題として,提案手法の効果は大規模回路 において大きい傾向が見られたため,さらに大規模 な回路に対する実験を行うとともに,テストベクト ル数の増加を抑制するテスト圧縮手法の考案が必 要であると考えられる.

文 献

- [1] Y. Sato, S. Hamada, T. Maeda, A. Takatori, Y. Nozuyama and S. Kajihara, "Invisible Delay Quality SDQM Model Lights Up What Could Not Be Seen," Proc. ITC, Paper 47.1, 2005.
- [2] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreeprakash and M. Hachinger, "A case study of IR-drop in structured at-speed testing," *Proc. ITC*, pp. 1098-1104, 2003.
- [3] Y. Zorian, "A Distributed BIST Control Scheme for Complex VLSI Devices," Proc. VTS, pp. 4-9, 1993.
- [4] N. Ahmed, M. Tehranipoor and V. Jayaram, "Transition delay fault test pattern generation considering supply voltage noise in a SOC design," *Proc. DAC*, pp. 533-538, 2007.
- [5] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault and S. Pravos-soudovitch, "A gated clock scheme for low power scan testing of logic ICs or embedded cores," *Proc. ATS*, pp. 253-258, 2001.
- [6] X. Wen, Y. Yamashita, S. Kajihara, L. 'T. Wang, K. K. Saluja and K. Kinoshita, "Low-Capture-Power Test Generation for Scan Testing," Proc. VTS, pp. 265-270, 2005.
- [7] Y. Yamato, X. Wen, K. Miyase, H. Furukawa and S. Kajihara, "A GA-Based Method for High-Quality X-Filling to Reduce Launch Switching Activity in At-Speed Scan Testing," *Proc. IEEE PRDC*, pp. 81-86, 2009.
- [8] V. R. Devanathan, C. P. Ravikumar, and V. Kamakoti, "A stochastic pattern generation and optimization framework for variation-tolerant, power-safe scan test," *Proc. ITC*
- [9] X. Wen, K. Miyase, S. Kajihara, H. Furukawa, Y. Yamato, A. Takashima, K. Noda, H. Ito, K. Hatayama, T. Aikyo and K. K. Saluja, "A Capture-Safe Test Generation Scheme for At-Speed Scan Testing," Proc. ETS, pp. 55-60, 2008.
- [10] K. Miyase and S. Kajihara, "XID: Don't Care Identification of Test Patterns for Combinational Circuits," *IEEE Trans. Comput. Aided Design Int. Circuits & Syst.*, vol. 23, no. 2, pp. 321-326, 2004.

表 4. 実験結果

	Initial 7	Fest Set		Number of Unsafe Faults								Test Generation Time(sec.)				
Circuit	Max. WSA	# of USF		70%					90%				70%		90%	
		70%	90%	Pro	oposed	oosed TetraMAX		Proposed		TetraMAX						
				# of USF	Reduction Ratio (%)	# of USF	Reduction Ratio (%)	# of USF	Reduction Ratio (%)	# of USF	Reduction Ratio (%)	Proposed	TetraMAX	Proposed	TetraMAX	
s5378	1588	156	71	0	100	156	0	0	100	39	29	0.23	0.12	0.06	0.09	
s9234	2513	519	210	49	91	493	5	0	100	25	34	4.73	50.39	0.09	0.18	
s13207	2742	372	85	2	99	358	4	0	100	18	5	1.81	25.96	0.05	0.20	
s15850	3218	120	51	0	100	120	0	0	100	4	89	0.46	2.05	0.05	0.21	
s35932	12486	4922	2326	0	100	4922	0	0	100	812	0	34.34	0.65	8.69	0.56	
s38417	11746	2688	1448	10	100	2673	1	5	99	85	87	202.20	39.00	58.25	25.53	
s38584	6428	1448	1095	0	100	1448	0	0	100	551	11	32.03	0.84	6.80	0.55	
Avg.					99		1		100		37					