

BIST 環境下におけるメルセンヌ・ツイスタアルゴリズムの評価

Adopting Mersenne Twister Algorithm as Test Pattern Generator under BIST

吉田 拓弥[†]
Takuya Yoshida

里中 沙矢香[‡]
Sayaka Satonaka

山口 賢一[§]
Ken'ichi Yamaguchi

岩田 大志[§]
Hiroshi Iwata

1 研究背景

LSI (Large Scale Integrated circuit) の微細化により、回路規模は増大している。そのため、LSI 製造後の出荷テストにおいて高品質かつ低コストなテストを行うことが困難になっている。出荷テストには非常に高価なテストを用いて行われるので、テスト時にテストを専有することはテストコスト上昇の一因となる。

テストコストを削減する手法の一つに、テスト機能をもつ回路を LSI 内部に搭載する組込み自己テスト (Built-In Self Test) がある。BIST には、テストパターン発生器をテスト対象回路に直接接続する方法や、図 1.1 のようにテスト対象回路を完全スキャン設計した上で、スキャンチェーンにテストパターンを印加する STUMPS アーキテクチャなどの構成方法が存在する。BIST では、テストに必要な機能は少なく済むためテストコストを削減できる。

BIST のテストパターンには擬似乱数が用いられる。擬似乱数を用いる場合、テストパターン生成器には擬似乱数生成器の一つである線形フィードバックシフトレジスタ (Linear Feedback Shift Register: LFSR) がよく利用される。 n ビット出力の LFSR は、 $2^n - 1$ 周期の擬似乱数生成が可能である。また D フリップフロップ n 個と数個の XOR のみで構成できるため、回路面積が小さい。これらの理由から、LFSR はテストパターン生成器として広く用いられている。多くの組合せ回路モジュールに対しては少ないパターン数で高い故障検出率を得られることが知られているが、順序回路に対しては十分な故障検出率を得られない場合がある [1]。また n が大きい場合 ($n < 200$)、周期が $2^n - 1$ になる特性多項式が得られていない。

別の擬似乱数生成アルゴリズムに、メルセンヌ・ツイスタ (Mersenne Twister: MT) アルゴリズムがある。メルセンヌ・ツイスタアルゴリズムの周期は $2^{19937} - 1$ と非常に長い。また出力の連続した値間での相関が無視できる程小さく、乱数性が良いという特徴を持つ [2]。

本論文では、LFSR が検出できない故障を検出する

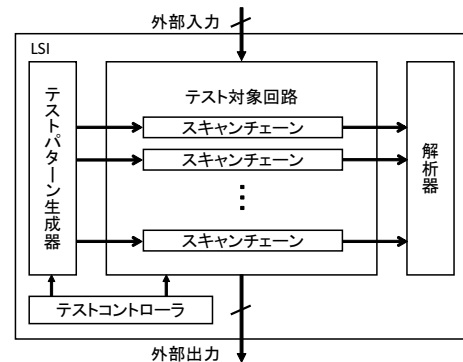


図 1.1: STUMPS アーキテクチャ

ために、擬似乱数生成アルゴリズムの一つであるメルセンヌ・ツイスタアルゴリズムをテストパターン生成器として実装し、LFSR とメルセンヌ・ツイスタアルゴリズムの故障検出率、及び面積増加について評価する。

2 信頼性に関する実験

本章では、ITC'99 ベンチマーク回路に対して、STUMPS アーキテクチャのテストパターン生成器にメルセンヌ・ツイスタを用いた際の故障検出率を LFSR と ATPG を用いて生成したテストパターンと比較することで信頼性の評価を行う。

擬似乱数は 32 本のスキャンチェーンを通して並列に印加し、外部入力には故障シミュレータ (TetraMax) 制約による固定値入力、外部出力は観測しないモデルを利用する。また、テスト対象回路の ITC'99 ベンチマーク回路は Design Compiler を用いて完全スキャン設計を行う。本実験では、遷移故障モデルを実験の対象とする。

実験結果を表 2.1 に示す。故障検出率は 50 万パターン印加した時の数値である。MT19937 及び LFSR は各テストパターン生成器の故障検出率、ATPG は各ベンチマーク回路に対してテスト生成を行った際の故障検出率を示している。実験を行った 8 つの回路全てで、メルセンヌ・ツイスタアルゴリズムは LFSR に比べて故障検出率が向上した。しかし、メルセンヌ・ツイスタアルゴリズムは実験を行った全ての回路に対して、テスト生成で求めた故障検出率を下回る結果となった。

[†]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

[§]奈良工業高等専門学校 情報工学科

表 2.1: ITC'99 ベンチマーク回路に対するメルセンヌ・ツイスタ MT19937 と 32bitLFSR の故障検出率 [%]

	b10	b11	b12	b13	b14	b15	b17	b19
MT19937	96.45	91.47	94.89	96.52	90.33	79.76	66.40	84.62
LFSR	94.05	88.07	87.62	89.94	89.83	75.59	63.05	83.07
ATPG	97.51	92.21	98.46	96.75	97.31	94.70	88.49	97.09

ATPG によるテスト生成は、順序回路に対しては故障検出率が低くなる傾向があるが、組合せ回路に対しては高い故障検出率を得ることができる。本実験ではテスト対象回路を完全スキャン設計しているため、テスト対象回路は組合せ回路とみなしてテスト生成が行えるため、擬似乱数によるテストパターンより高い故障検出率が得られたと考えられる。

3 実装コストに関する評価

本論文では、テストパターン生成器を LSI 内部に組込むテスト手法を対象としているため、本章では面積増加によるコストについて評価を行う。

STUMPS アーキテクチャによるテストを実現するには、完全スキャン設計を行うことに伴う面積増加とテストパターン生成器の面積増加が必要となる。本実験で用いた ITC'99 ベンチマーク回路を STUMPS アーキテクチャで実現する際の面積増加率を表 3.1 に示す。

表中の元回路は各ベンチマーク回路を Design Compiler で制約を指定しない論理合成を行った際の、2 入力 NAND ゲートを 1 とした時の回路面積である。scan は各ベンチマーク回路を完全スキャン設計した際の回路面積の増加率であり、LFSR、及び MT は完全スキャン設計後回路面積に各テストパターン生成器の回路面積を足したものの面積増加率を示している。ここで、表中の LFSR には 32bitLFSR、MT には TT800 の回路面積を用いている。

表 3.1 より、b10、b11、b12、b13 のような回路面積が小さな回路に対しては面積増加が支配的になり適さないが、b19 のような、実用的な回路に比べて小さな

回路でも、LFSR を用いた既存の BIST アーキテクチャに比べて 1.9 倍程度の面積増加を許容することで、メルセンヌ・ツイスタを用いた STUMPS を実装できる。

4 まとめと今後の課題

LSI の出荷テストは高い品質と信頼性を確保するための重要な工程である。本論文では、メルセンヌ・ツイスタアルゴリズムによるテストパターン生成器を用いた出荷テストの評価のために、LFSR と比較し信頼性(故障検出率)と実装コスト(面積増加)の点から比較を行った。

信頼性に関する実験では、ITC'99 ベンチマーク回路 8 つに対し、全ての回路においてメルセンヌ・ツイスタアルゴリズムは LFSR より高い故障検出率が得られた。

実装コストに関する実験では、メルセンヌ・ツイスタアルゴリズムを STUMPS アーキテクチャにより実装する場合でも、既存の BIST アーキテクチャに比べて 1.9 倍程度の面積増加を許容すれば、TT800 を用いた STUMPS を実装できることを示した。

今後の課題として、本実験で行っていない TT800 の信頼性の評価が挙げられる。また、既存手法である LFSR の故障検出率を評価するにあたり、乱数間の相関を小さくするフェーズシフタを用いて故障検出率を比較する必要がある。

謝辞

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

参考文献

- [1] Hiroshi Iwata, Sayaka Satonaka and Ken'ichi Yamaguchi, "An Efficient Test Pattern Generator - Mersenne Twister-," The 18th Workshop on Synthesis And System Integration of Mixed Information Technologies, R1-12, pp.62-67, Oct. 2013.
- [2] Makoto Matsumoto and Takuji Nishimura, "Mersenne Twister: A 623-dimensionally equidistributed uniform pseudo random number generator," ACM Transactions on Modeling and Computer Simulation, Vol.8, No.1, pp.3-30, Jan., 1998.

表 3.1: STUMP の実装に必要な面積増加率

	元回路	scan[%]	LFSR[%]	MT[%]
b10	334	5.09	106.59	1,827.25
b11	770	3.90	47.92	794.29
b12	2,023	5.98	22.74	306.82
b13	679	7.07	57.00	903.39
b14	11,891	1.81	4.66	52.99
b15	10,109	4.12	7.47	64.32
b17	29,834	4.40	5.54	24.80
b19	170,955	3.53	3.73	7.09