

平成 21 年 6 月 12 日現在

研究種目：基盤研究 (C)	
研究期間：2006～2008	
課題番号：18500038	
研究課題名 (和文)	構造テストに高故障検出効率を保証するプロセッサの命令レベル自己テスト法
研究課題名 (英文)	Software-Based Self-Test for Processors to guarantee high fault efficiency for structured faults
研究代表者	
	井上 美智子 (INOUE MICHIKO)
	奈良先端科学技術大学院大学・情報科学研究科・准教授
	研究者番号：30273840

## 研究成果の概要：

本研究では、機能テストと構造テストの特長を活かしたテスト手法である、プロセッサの命令レベル自己テスト法の研究を行った。パイプラインプロセッサに対し、モジュール単体でのテスト生成と命令列探索を組み合わせて効率のよいテスト生成手法を提案し、パス遅延故障に対し高い故障検出効率を得られることを示した。さらに、自己テストプログラムを効率よく生成する手法であるテンプレートを用いて生成された自己テストプログラムのためのテスト容易化設計手法を提案した提案法は、テンプレートレベル故障検出効率 100%、すなわち、誤りマスクを完全に回避できることを特長とする。

## 交付額

(金額単位：円)

	直接経費	間接経費	合計
18年度	2,100,000	0	2,100,000
19年度	800,000	240,000	1,040,000
20年度	800,000	240,000	1,040,000
年度			
年度			
総計	3,700,000	480,000	4,180,000

研究分野：総合領域

科研費の分科・細目：情報学・計算機システム・ネットワーク

キーワード：設計自動化、VLSI のテスト、テスト容易化設計

## 1. 研究開始当初の背景

VLSIの微細化、大規模化を推進するために、半導体製造技術、設計技術、テスト技術など、様々な技術の進歩が必要とされている。特に、製造後のVLSIに対して行うテスト技術は、微

細化による半導体回路の欠陥の多様化や複雑化、大規模化によるテストパターン生成、テストスケジューリングの複雑化など多くの問題を抱えている。マイクロプロセッサなど高速動作を必要とする回路には、プロセッサの性能を損ねることなく高品質なテストを

実現する技術が求められている。プロセッサのテストは、大きく機能テストと構造テストからなる。機能テストはプロセッサの機能を確認するためのテストで、テスト品質を保証するための重要視される要件の一つである実動作速度（回路の通常動作時の速度）でのテストが可能であるが、回路の物理的欠陥をすべて検出できるとは限らない。これに対し、構造テストは回路の構造に基づいた故障（構造故障）を検出するテストで、機能テストと比較して高いテスト品質を保証する。しかし、スキャン設計、組み込み自己テスト法といったテストのための設計変更であるテスト容易化設計を必要とし、動作速度、回路面積に関して回路の性能低下を招く、テスト実行時に通常動作時に比べ高い消費電力を必要とする、実動作速度テストが困難であるなどの問題点が指摘されていた。

そこで、本研究では、機能テストと構造テストの特長を活かしたテスト手法である、プロセッサの命令レベル自己テスト法の研究を行うこととした。プロセッサの命令レベル自己テストとは、プロセッサがプログラムを実行することにより、プロセッサ自身をテストする自己テスト手法である。プロセッサをテストするためのプログラムを、以降、テストプログラムと呼ぶ。この手法では、プログラムの実行というプロセッサの通常動作時の機能を利用してテストを実行するため、実動作速度テストが可能であり、また、テスト実行時に過剰に電力を消費するという問題も回避される。しかし、命令レベル自己テスト法では、命令によって実現可能なデータフローのみを用いてテストを行うため、一般に構造故障に対して高い故障検出効率を得ることが困難となっている。本研究では、命令レベル自己テスト法のためのテストプログラム生成法、テスト容易化設計法を提案し、

構造故障に対し高い故障検出効率を保証することを目標とした。

## 2. 研究の目的

本研究は、プロセッサに対し、高い故障検出効率（テストの対象となる全故障数に対する検出故障を保証する構造テストを命令列を用いて実現することを特色とする。ここで、故障検出効率とは以下の式で表されるテストプログラム生成の評価尺度である。

故障検出効率 (%) =

$$\frac{\text{検出故障数} + \text{冗長識故障数}}{\text{対象となる全故障数}}$$

本研究では、故障を検出するテストプログラムを生成するだけでなく、テストプログラムでは検出することのできない故障を冗長故障として識別することを目標とする。実際、プロセッサへの入力には命令セットアーキテクチャにより限定されるので、命令列だけでは検出できない故障が存在する。故障検出効率を評価することにより、テストプログラムの質をより正確に表現することができる。

命令列を実行することによる自己テスト法では、テスト実行をプロセッサの機能を用いて実現するため、通常動作時の動作速度で、かつ、通常動作時に許容される消費電力の範囲でテスト実行が可能といった利点がある。通常動作時の速度でのテスト実行は縮退故障といった静的な動作に関する故障モデルには必要な条件でなく、広く普及しているスキャン設計などでは、テスト時の動作速度は通常動作時に比べて遅いことが多い。本研究では、通常動作時の動作速度でテストを実行するため、対象とする故障が縮退故障であっても、遅延故障など他の故障モデルを検出す

ることができるという意味で、高いテスト品質を実現することを特長とする手法の提案を目標とした。また、テスト実行時の消費電力は、通常動作時に許容される消費電力の範囲となるので、テスト実行が回路を破壊するといった危険性がなく、また、消費電力を抑えるための設計変更も必要としない。

テストプログラムの実行によつてテストでは、テスト時に通常動作時のデータフローを利用するため、テスト容易化設計（故障検出効率を向上させるための設計変更）を行った場合でも、その遅延オーバーヘッド、面積オーバーヘッドをスキャン設計などの従来法に比べ格段に小さくできると予想された。

本研究の成果により、従来法に比べて質の高いテストを低いオーバーヘッドで実現できることが予想される。これにより、プロセッサのテストコストを大幅に削減することが期待された。

### 3. 研究の方法

機能テストと構造テストの特長を活かしたテスト手法である、プロセッサの命令レベル自己テスト法の研究を行う。本研究では、命令レベル自己テスト法のためのテストプログラム生成法、テスト容易化設計法を提案し、構造故障に対し高い故障検出効率を保証する。

テストプログラム生成法に関しては、プロセッサの構成要素であるモジュールごとにテスト生成ツールを用いてテストパターン生成を行い、得られたパターンをプロセッサの外部からモジュールに伝搬し、モジュールが出力するテスト応答をプロセッサの外部まで伝搬する命令列を求めテストプログラムを生成する。

提案するテストプログラム生成法は、計算時間の大きいテスト生成を規模の小さいモ

ジュール単位で行い、効率のよいテストプログラム生成を可能にしている。しかし、モジュール単体に対するテスト生成では故障の影響を考慮するが、命令列の生成では故障の影響を考慮しておらず、モジュール単体では検出されるが、テストプログラムでは検出できない故障が存在する故障マスクが生じることがある。そこで、故障マスクを回避するためのテスト容易化設計法を提案する。

提案するテストプログラム生成法とテスト容易化設計法を用いることにより、高い故障検出効率を保証する命令レベル自己テストを実現する。

### 4. 研究成果

まず、パイプラインプロセッサに対し、モジュール単体でのテスト生成と命令列探索を組み合わせることで効率のよいテスト生成手法を提案した。パイプラインプロセッサの動作をパイプライン命令実行グラフで表現し、そのグラフからプロセッサを構成する各モジュールをテストする命令列を生成する手法を提案した。提案法では、パス遅延故障に対し、モジュール単体テストにおける 100%の故障検出効率を達成した。

さらに、故障検出効率を向上させるために、テスト容易化設計の研究を行った。自己テストプログラムを効率よく生成する手法として、テンプレートをを用いた自己テストプログラム生成法が知られている。テンプレートをを用いる自己テストプログラム生成法では、モジュール単体でのテスト生成では検出される故障が、合成されたテストプログラムでは検出されないという誤りマスクの問題がある。そこで、本研究ではテンプレートをを用いて生成された自己テストプログラムのためのテスト容易化設計手法を提案した。提案法

では、テンプレートレベル故障検出効率100%、すなわち、誤りマスクを完全に回避できることを特長とする。さらに、提案法は、テストプログラム生成に用いられるテンプレートに依存せず、すなわち任意のテンプレートを用いて生成されたテストプログラムでの誤りマスクを回避するという特長をもつ。そのため、提案法は、テストプログラム生成を行う前に適用可能であり、通常の回路設計のサイクルに容易に取り込むことが可能である。提案法では、回路の数か所に観測点を挿入するが、観測点の挿入位置の最適化を行うことにより、面積オーバーヘッドを小さくすることに成功している。また、観測点の挿入は、対象回路の信号線にファンアウトを追加するのみなので、遅延オーバーヘッドがない、または非常に小さいことを保証する。

#### 5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文] (計 2件)

① Virendra Singh, Michiko Inoue, Kewal K. Saluja and Hideo Fujiwara, "Instruction-based self-testing of delay faults in pipelined processors," IEEE Trans. on Very Large Scale Integration (VLSI) Systems, Vol. 14, No. 11, pp.1203-1215, Nov. 2006, 査読有

② Masato Nakazato, Michiko Inoue, Satoshi Ohtake and Hideo Fujiwara, "Design for testability method to avoid error masking of software-based self-test for processors," IEICE Trans. on Information and Systems, Vol. E91-D, No. 3, pp.763-770, Mar. 2008, 査読有

[学会発表] (計 5件)

① Masato Nakazato, Satoshi Ohtake, Michiko Inoue and Hideo Fujiwara, "Design for testability of software-based self-test for processors," 15th IEEE Asian Test Symposium (ATS'06), pp.375-380, Nov. 2006, 査読有

② Elena Hammari, Michiko Inoue, Einar J. Aas and Hideo Fujiwara, "Delay test of FPGA routing networks by branched test paths," Informal Digest of Papers, 13th IEEE European Test Symposium (ETS'08), May 2008, 査読有

③ Satoshi Ohtake, Naotsugu Ikeda, Michiko Inoue, Hideo Fujiwara, "Unsensitizable Path Identification at RTL Using High-Level Synthesis Information," Digest of papers of 16th IEEE International Test Synthesis Workshop, 2009, 査読有

④ Michiko Inoue, Tomokazu Yoneda, Muneo Hasegawa and Hideo Fujiwara, "Partial scan approach for secret information protection," Proceedings of the 14th IEEE European Test Symposium (ETS'09), pp.143-148, May 2009, 査読有

⑤ 中里 昌人, 大竹 哲史, 井上 美智子, 藤原 秀雄, "プロセッサの命令レベル自己テストのためのテスト容易化設計," 信学技報 (ICD2006-40~59), Vol. 106, No. 92, pp.49-54, June 2006, 査読無

#### 6. 研究組織

##### (1) 研究代表者

井上 美智子 (INOUE MICHIKO)

奈良先端科学技術大学院大学・情報科学研究科・准教授

研究者番号：30273840

##### (2) 研究分担者

大竹 哲史 (OHTAKE SATOSHI)

奈良先端科学技術大学院大学・情報科学研究科・准教授

研究者番号：20314528

米田 友和 (YONEDA TOMOKAZU)

奈良先端科学技術大学院大学・情報科学研究科・准教授

研究者番号：20358971