

ノンスキャン BIST を実現するテスト容易化設計の線形計画問題

山口 賢一* · 村田 絵理**

Linear Programming Problem to Design for Testability of Non-Scan BIST

Ken'ichi YAMAGUCHI and Eri Murata

テスト用容易化設計において、付加する回路要素の挿入を最小限に抑えるために様々なヒューリスティックアルゴリズムが提案されている。本論文では、筆者らが多く提案してきたノンスキャン BIST におけるアルゴリズムを一般化し、線形計画法を用いて設計するための手法について提案する。

1. 背景

スキャン方式による組込み自己テスト (Built In Self-Test: BIST) の欠点であるテスト時間の増加を抑える方法として、ノンスキャン BIST がある。著者らは、ノンスキャン BIST のためのテスト容易化設計法として様々な手法を提案しているが、設計変更に伴うアルゴリズムが複雑であるため大規模な回路への適用の大きな障壁となっている。そこで、本論文ではそれらのアルゴリズムを線形計画問題によって一般化し、様々な回路に対して適用可能となるようにする。

2. 諸定義

2.1. データパスグラフ

設計された回路に対してテスト容易化設計を行うために、回路をグラフとしてモデル化して取り扱う。RTL 回路におけるデータパスをグラフ化したものをデータパスグラフという。定義 1 でデータパスグラフを定義し、図 1 にデータパスとデータパスグラフを示す。

[定義 1]

データパスグラフ G は、以下の要素から構成される。尚、 G は有向グラフであり、データパスと同一視できる。

$$V = \{V_{PI}, V_{PO}, V_{seqin}, V_{seqout}, V_{comin}, V_{comout}\}$$

V_{PI} : 外部入力 (PI: Primary Input) 集合

V_{PO} : 外部出力 (PO: Primary Output) 集合

V_{seqin} : 順序回路要素の入力端子集合

V_{seqout} : 順序回路要素の出力端子集合

* 奈良工業高等専門学校情報工学科

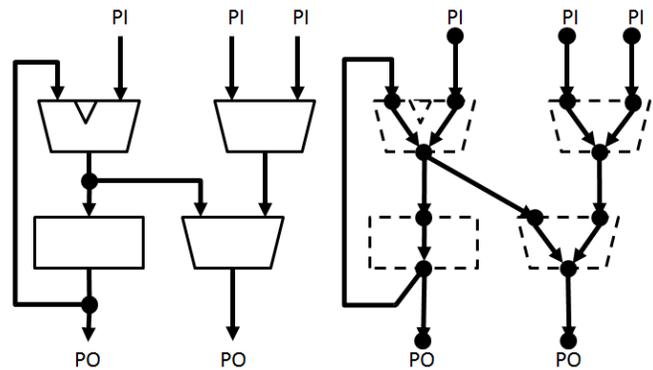
** 奈良先端科学技術大学院大学情報科学研究科

V_{comin} : 組合せ回路要素の入力端子集合

V_{comout} : 組合せ回路要素の出力端子集合

・ E はノード間の接続関係を表す

・ C は各辺を実現するのに付加する回路要素の面積などであり、各辺に対して割り当てる。



(a) データパス (b) データパスグラフ

図 1 データパスとデータパスグラフ

2.2. スルー機能

スルー機能は、演算モジュールのいずれかの入力端子と出力端子の間で任意の値の伝搬を保障する機能であり、ノンスキャン BIST におけるテスト容易化設計のための付加要素である。演算モジュールに対するスルー機能の実現方法には 2 通りの方法がある。

・ 演算モジュールが、入力端子に印加され得る任意の値の集合に対して単位元をもつ演算を実現している場合：マスク阻止を演算モジュールに付加することにより低ハードウェア・オーバーヘッドで実現できる。マスク阻止とは、1つの入力端子と1つの出力

端子, 1つの制御端子を持ち, 演算モジュールの演算に対して単位元を発生することのできる付加回路である.

- ・また, 単位元によってスルー機能を実現する場合, 辺のコストを1とする.
- ・そのような単位元が存在しない場合: マルチプレクサを演算モジュールに付加することにより実現できる. また, マルチプレクサによってする機能を実現する場合は, 辺のコストを18とする.

図2にスルー機能の実現例を示す. また, このときに付加された回路要素(図2の点線で囲まれた回路要素)のことをDFT要素という. また, 残りの辺はコストを0とする.

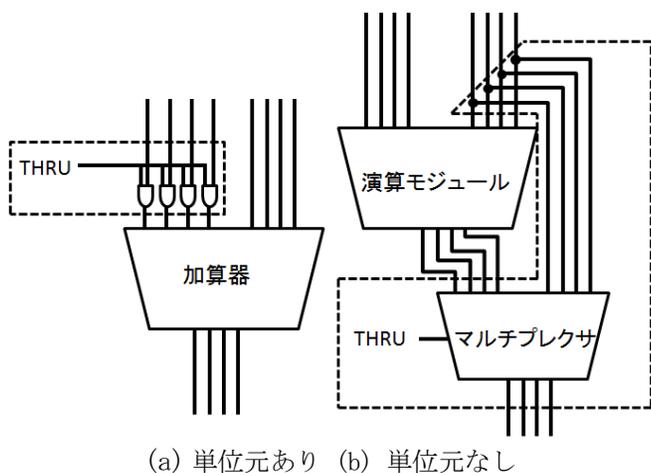


図2 スルーの実現例

2.3. 単一制御可検査性 [1]

井筒らは, データパスがテスト可能である性質として単一制御可検査性を定義し, データパスがその性質を満たすような設計変更を行うように提案している. 次に単一制御可検査性の定義を示す.

[定義2] 単一制御可検査性

データパスグラフにおいて, 各組合せ回路要素 M に対して, 以下の条件を満たす互いに共通部分を持たない単純経路 P_1, P_2, P_3 が存在し, 各経路に対応するデータパス上の経路に沿って任意の値が伝搬可能であるときデータパスは単一制御可検査であるとする.

- ・外部入力を始点とし, M の入力端子を終点とする経路 P_1 及び P_2 .
- ・ M の出力端子を始点とし, 外部出力を終点とする P_3 .

3. 提案手法

提案手法では, データパスの各組合せ回路要素に対して単一制御可検査性を満たすために, 各回路要素に対し

て経路生成を行う. 経路生成とは, 各組合せ回路要素をテストするときに利用する経路であり, 線形計画法の解法に用いる式を生成し, 線形計画法を解くことによって得られる. また, これに基づいてデータパスに回路要素を付加することによりテストを容易にする.

線形計画法の式では, データパスグラフの辺を変数として取り扱う. また, 線形計画法を解いた結果, 各変数 $x(\in E)$ は1又は0の値をとる. 変数が1の場合は, 単一制御可検査性の経路に含まれ, 0の場合は経路に含まれない. また, 線形計画法の解を求めるために用いる式はデータパスグラフに前処理を施したものをを用いる.

3.1 提案手法の全体の処理

図3に提案手法全体の流れを示す. 提案手法は, データパスに含まれる各組合せ回路要素に対して経路生成に必要な式を求め, 経路生成を行う. このとき, 経路生成対象は組合せ回路の出力の出次数が多いものから順に経路生成を行う. これは, 出次数が多ければ多いほど経路生成対象の出力から外部出力まで最悪の場合に選択できる経路は多くなり, DFTの早い段階で必要な箇所にDFT要素が挿入されると考えられるためである.

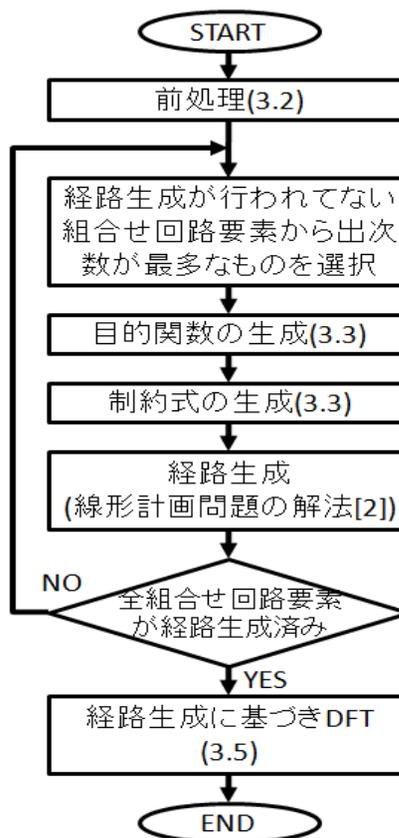


図3 提案手法全体の流れ

1 経路生成対象の組合せ回路の出力がレジスタに繋がっている場合は, 組合せ回路の出力の出次数+レジスタの出力の出次数-接続しているレジスタ数を出力とする.

3.2. 前処理

前処理では、線形計画法を解くことによって必ず対象の組合せ回路要素に対する単一制御可検査性の経路を発見できるようにノード及び辺の付加を行う。

(i) PI を 1 つにまとめる V_{dPI} , PO を 1 つにまとめる V_{dPO} を付加する。

(ii) 外部入力から各回路要素の入力まで直接値を伝播できる辺 $e_1 = v_1 (\in V_{PI}) \times v_2 (\in \{V_{comin}, V_{sequin}\})$, 各回路要素の出力から外部出力まで直接値を伝播できる辺 $e_2 = v_1 (\in \{V_{comout}, V_{regout}\}) \times v_2 (\in V_{PO})$ を付加する。

(iii) 付加した辺 e_2 に対しては v_1 から V_{dPO} までの最小距離をコストとし、辺 e_1 に対しては V_{dPI} から v_2 までの最小距離に辺 e_2 の最大コストを加算したものとす。

図 3 に図 2(b) に対して前処理を施した結果を示す。

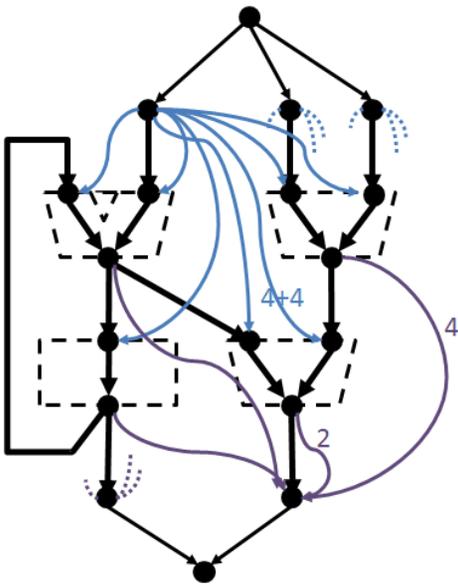


図 4 図 2(b) に前処理を施したグラフ

3.3. 目的関数

テスト容易化設計における最適化目標は、付加する回路面積を小さくすることである。そのため目的関数を $\sum_{i=1}^n x_i \times C_i$ 最小とし、回路実現に必要なコストを最小とする。ただし、既に経路生成によって求められた経路にコスト 0 以上の辺が経路に含まれていた場合は、その辺のコストを 0 とする。また、経路生成対象の入出力間の辺のコストは 0 とする。

3.4. 制約式

制約式は、回路の接続関係および単一制御可検査性を満たすための条件を 4 つの式で表す。また、解法で正しく行うために、さらにもう 1 つ式を付加する。以降、3.4.1 から 3.4.5 でそれぞれについて説明する。

3.4.1. 回路要素の接続関係の制約式

回路要素の接続関係を表す。各ノード $v (\in \{V - V_{dPI} - V_{dPO}\})$ に対して、 $\sum_{i=1}^n a_i \times x_i = 0$ (ただし x_i が $v (\in V_{comout})$ であり、経路生成対象の出力端子の入力辺は $a_i = 0.5$, x_i が v の入力の場合 $a_i = 1$, x_i が v の出力の場合 $a_i = -1$, その他の場合 $a_i = 0$) とする。更に、 x_i の係数 $a_i = 0.5$ の場合は、 $x_i = 1$ とする。なぜなら、 x_i が経路生成対象であり、必ず経路として含まれるからである。

3.4.2. 2 入力回路要素に対する制約式

2 入力回路要素に対する条件を制約式に表す。2 入力回路要素は、両方の入力を同時に出力することはできない。そのため、2 入力回路要素 $v (\in V_{comout})$ に対して $\sum_{i=1}^n a_i \times x_i \leq 1$ (ただし x_i が v の入力の場合 $a_i = 1$, その他の場合 $a_i = 0$) とする。また、2 入力回路要素 $v (\in V_{comout})$ が経路生成対象の場合、両方の入力に PI からの経路が必要のため、 $\sum_{i=1}^n a_i \times x_i \leq 2$ とする。

3.4.3. ダミー入出力端子に関する制約式

経路生成対象に入力端子数と出力端子数に合わせて制約式を与える。ダミー入力端子 V_{dPI} において、 $\sum_{i=1}^n a_i \times x_i =$ 経路生成対象入力数 (1 又は 2) (ただし、 x_i が V_{dPI} の出力辺の場合 $a_i = 1$, その他の辺の場合 $a_i = 0$) とする。一方、ダミー出力端子 V_{dPO} においては、 $\sum_{i=1}^n a_i \times x_i = 1$ (ただし x_i が V_{dPO} の入力辺の場合 $a_i = 1$, その他の辺の場合 $a_i = 0$) とする。

3.4.4. 単一制御可検査性の制約式

外部入力からテスト対象モジュールまでの経路、そしてテスト対象モジュールから外部出力までの経路が存在することが単一制御可検査性の条件である。そのため、単一制御可検査の条件を満たした経路には閉路が含まれない。そのため、各閉路に対して $\sum_{i=1}^n a_i \times x_i < \sum_{i=1}^n a_i$ (ただし、 a_i が閉路に含まれている場合は、 $a_i = 1$, その他の場合 $a_i = 0$) の条件を付加する。

3.4.5. 0-1 線形計画問題

各辺はテスト時に利用する又は利用しないかのどちらかである。本稿での目的関数及び制約式から導いた解において、テスト時に利用しない辺に対しては 0 の割り当

て、利用する辺に対しては1の割り当てを行う。そのため、 $\forall e \in E, \forall e \in E \geq 0$ とする。

3.5. 経路生成の結果に基づく DFT

コストが0以上の結果が出た回路に対して、テスト時の経路を生成するには回路要素を付加する必要がある。そのため、そのような回路要素の経路に利用されている辺に対応する回路要素の挿入を行う。

4. 実験手順

図5に実験手順を示す。
実験では、線形計画問題によって単一制御可検査性に回路を設計変更し、得られた回路を論理合成することによって設計変更に伴う面積増加の算出を行った。

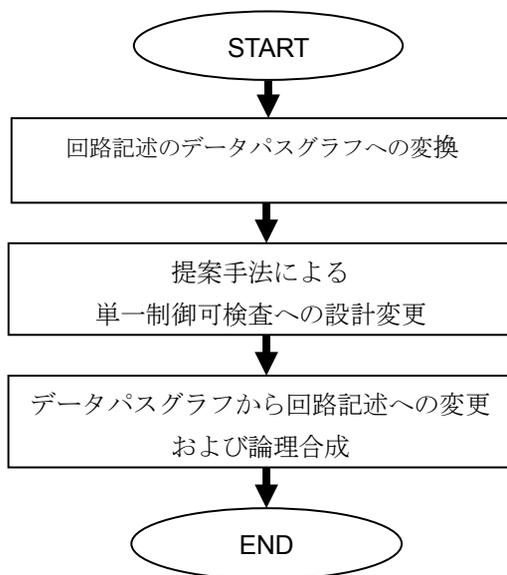


図5 実験手順

5. 実験結果

実験結果より、回路を同じ性質に変更しているのにも関わらず、提案手法と既存手法 [1] の間で面積増加率に違いが生じた。これは、既存手法では回路を単一制御可検査にするための問題を大きく分割して解いているためであり、問題をまとめて扱う提案手法のほうが、有効な位置に回路の付加を行ったため面積の増加を抑えることができたといえる。

表1 面積増加率

回路名	LWF	Paulin
提案手法	20.7%	9.2%
既存手法[1]	24.9%	10.5%

6. まとめと今後の課題

本稿では、問題をまとめて扱う手法として線形計画法を適用し、設計変更アルゴリズムの一般化を行った。具体的には、制御経路と観測経路を同時に求める方法について提案した。今後は、設計変更に伴う面積増加に対するパラメータチューニングを行い、既存手法との厳密な計算量などの違いを比較する必要がある。

参考文献

- [1] 井筒, et.al : "Single-control testability of RTL data paths for BIST," Ninth Asian Test Symposium (ATS'00)
- [2] http://www.mahoroba.ne.jp/~felix/Toolbox/Softwares/Lp_solve/install.html.
- [3] 伊理正夫, 白川功, 梶谷洋司, 篠田庄司 : "演習グラフ理論 = 基礎と応用 =", コロナ社, 初版代10刷発行, (1995).
- [4] K. Yamaguchi, et.al : Hierarchical BIST: Test-Per-Clock BIST with Low Overhead," Proceedings of 3rd Workshop on RTL and High Level Testing, (2002), pp. 42 ~ 47"

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