

レジスタ転送レベル回路に対する連続透明化設計法

米田 友和[†] 藤原 秀雄[†]

Design for Consecutive Transparency Method of RTL Circuits

Tomokazu YONEDA[†] and Hideo FUJIWARA[†]

あらまし 本論文では、SoC に組込まれるコアとしてレジスタ転送レベル回路を対象とし、回路が連続透明性を満たすように設計変更する連続透明化設計法を提案する。コアの連続透明性とは、形状（テストモード）を選択することによって、各入力端子に入力される任意の長さの系列を値を変えことなく出力側に連続したクロックサイクルで伝搬可能としたり、各出力端子から出力される任意の長さの系列を入力側から連続したクロックサイクルで伝搬可能とする性質をいう。したがって、連続透明なコアが SoC に組み込まれた場合、そのコアは SoC 内の他のコア及び信号線への連続テストアクセスのための経路として利用可能となる。連続テストアクセス可能な SoC は、コア及び信号線に対して任意の故障モデルを対象とした任意のテスト系列が提供された場合でも、その系列を用いて想定した故障を完全にテストすることが可能である。また実験により、提案する連続透明化設計法による面積オーバーヘッドは、マルチプレクサのみを用いて入力から出力までの迂回路で連続透明性を実現した場合に比べ小さいことを示す。

キーワード テスト容易化設計、システムオンチップ、テストアクセス機構、連続透明性、連続可検査性

1. ま え が き

近年、半導体技術の進歩により、シリコンチップ上に搭載することのできるトランジスタ数は増加の一途をたどっている。これにより従来は複数の LSI で構成していたシステムを、各 LSI をコアと呼ばれる機能ブロックとして再利用し、システム全体を一つの LSI で実現するシステムオンチップ (Systems-on-a-Chip, 以下 SoC) が注目されている。複数の LSI の機能を 1 チップに集積した場合、ボードへ搭載するチップ数の低減による実装面積の縮小化、実装コストの低減、更に高速化といった効果をもつ。また、設計済みの IP (Intellectual Property) コアを利用することで短期間で大規模な回路の設計が可能となる。しかし現在、この SoC のテスト技術の開発が課題 [1] となっている。

SoC では、各コアは SoC の内部に組み込まれ、SoC 外部からコアの入出力を直接制御/観測することは不可能である。そのため、各コアに供給されたテスト系列を SoC 外部からコアへ伝搬し、その応答を SoC 外部へ伝搬するテストアクセス機構が必要不可欠となる。

SoC のテスト容易化設計 (Design for Testability, 以下 DFT) では、このテストアクセス機構の実現が重要な課題であり、これまでにテストバス方式 [4] ~ [6]、境界スキャン方式 [2], [3]、透明経路方式 [7] ~ [10] などのテストアクセス方式が提案されている。

SoC において、コアに供給されるテスト系列は、機能テストベクトル、スキャンベクトル及び非スキャン設計された順序回路に対するテスト系列など様々である。また近年の半導体の微細化、高速化により、縮退故障などのような論理故障のみならず、遅延故障などのようなタイミング故障のテストも重要となっている。したがって、SoC のコアに対してはタイミング故障を対象としたテスト系列が供給される場合も想定される。更にモデル化できない欠陥の検出を目的として実動作速度テストを可能にしているコアも存在する。信号線に関しても同様に、論理故障、タイミング故障及びクロストーク故障などを対象とした様々なテスト系列が想定される。このように、コア及び信号線には様々な故障モデルを対象とした様々なテスト系列が提供され、その系列を用いて想定した故障を完全にテストすることが必要となる。このため、SoC のテストでは内部のコア及び信号線に対し、任意のテスト系列をそのコア及び信号線の実動作速度の連続したクロックサイクル

[†] 奈良先端科学技術大学院大学情報科学研究科, 生駒市
Graduate School of Information Science, Nara Institute of
Science and Technology, Ikoma-shi, 630-0192 Japan

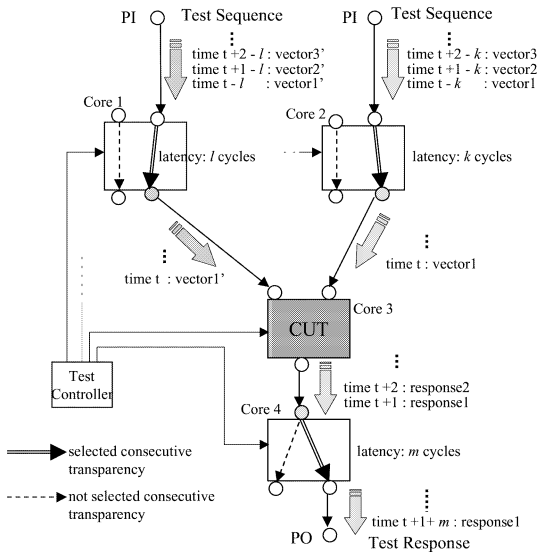


図 1 連続テストアクセス
Fig. 1 Consecutive test access.

で印加可能であり、その応答を実動作速度の連続したクロックサイクルで観測可能なテストアクセス（連続テストアクセス）機構が必要である。しかし、これまでに提案されているテストバス方式、境界スキャン方式及び透明経路方式では、すべてのコアと信号線に対する連続テストアクセスは不可能である。

文献 [11] で筆者らは、すべてのコア及び信号線に対して連続テストアクセスが可能で性質として SoC の連続可検査性を提案した。連続可検査な SoC は連続透明なコアで構成され、コアの連続透明性とコア間の信号線を用いて連続テストアクセスを実現している (図 1)。コアの連続透明性とは、形状 (テストモード) を選択することによって、各入力端子に入力される任意の長さの系列を値を変えことなく出力側に連続したクロックサイクルで伝搬可能としたり、各出力端子から出力される任意の長さの系列を入力側から連続したクロックサイクルで伝搬可能とする性質をいう。したがって、連続透明なコアが SoC に組み込まれた場合、そのコアは SoC 内の他のコアへの連続テストアクセスのための経路として利用可能となる。文献 [12] では、対象とするコアとして連続透明なコアに加え、連続透明性を満たさないコア及び BIST 化されたコアを対象とし SoC の連続可検査性を拡張した。連続透明性を満たさないコアが存在する場合、そのコアは他のコアへの連続テストアクセスのための経路として利用不

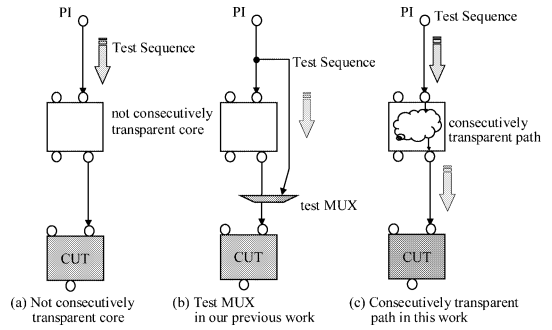


図 2 連続テストアクセスを可能とする DFT
Fig. 2 DFT for consecutive test access.

可能である (図 2 (a))。文献 [12] では、コア内部は設計変更不可能であると仮定し、マルチプレクサを用いてコアを迂回することでコア内部を設計変更することなく、連続テストアクセスを実現している (図 2 (b))。しかし、コアはソフトコア (レジスタ転送レベル回路)、ファームコア (ゲートレベルのネットリスト) 及びハードコア (レイアウトマスク) に分類され、ソフトコアはコア内部の設計変更が容易である。コア内部の設計変更が可能である場合、コア内部の既存の経路を有効利用することで、マルチプレクサを付加しコアを迂回するよりも低い面積オーバーヘッドで連続透明経路を実現できると考えられる。そこで本論文では、コアとしてレジスタ転送レベル (Register Transfer Level、以下 RTL) 回路を対象とし、回路が連続透明性を満たすように設計変更する連続透明化設計法を提案する。

提案手法では、回路内部の既存の経路を有効利用し、各外部入出力に対する連続透明経路を小さい面積オーバーヘッドで実現する (図 2 (c))。ベンチマーク回路を用いた実験により、提案手法による面積オーバーヘッドは、先に述べた連続透明経路をマルチプレクサによる迂回路のみで実現した場合の面積オーバーヘッドよりも小さいことを示す。

以下、2. でコアの連続透明性、対象とする RTL 回路及びその回路を表現する端子グラフを定義する。3. では、連続透明化設計法を提案する。4. で、提案手法をベンチマーク回路に適用し、その有効性を評価し、5. でまとめを行う。

2. 諸定義

2.1 対象とするレジスタ転送レベル回路

レジスタ転送レベルにおいて、回路はコントローラとデータバスから構成され、コントローラとデータバス

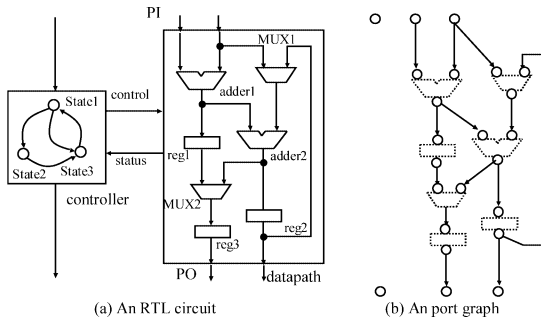


図 3 RTL 回路及び端子グラフ
Fig. 3 An RTL circuit and its port graph.

は制御信号線及び状態信号線で接続される(図 3(a)). コントローラは外部入力, 外部出力及び状態遷移図で記述されるものとする. データパスは回路要素及び回路要素を接続するデータ信号線で記述される. 回路要素は外部入力, 外部出力, ラッチ, レジスタ, マルチプレクサ, 演算モジュール, 観測モジュールに分類される. 回路要素に対してデータ信号線が接続される端子をデータ端子, 制御信号線が接続される端子を制御端子, 状態信号線が接続される端子を状態端子と呼ぶ. データ端子は回路要素に信号を入力するためのデータ入力端子と回路要素からデータを出力するためのデータ出力端子に分類される. 以下, データ入力端子を入力端子, データ出力端子を出力端子と呼ぶ. また, データパス上のすべての回路要素のデータ端子は等しいビット幅をもつものとする. データパス上の回路要素のデータ端子が異なるビット幅をもつ場合は, 各データ端子をすべてのデータ端子のビット幅の最大公約数 n で分割し, 等しいビット幅 n をもつデータパスとして扱うものとする. 同様に, コントローラのすべての外部入力及び外部出力も等しいビット幅をもつものとする. 以下で, 各回路要素について説明する.

外部入力, 外部出力: 外部入力 (PI) はデータパス外部からデータパスにデータを入力するための端子, 外部出力 (PO) はデータパスから外部にデータを出力するための端子である.

マルチプレクサ: マルチプレクサは入力端子, 出力端子及び制御端子をもつ. 制御端子の値に従って, 対応する入力端子の値をそのまま出力端子に出力する.

演算モジュール: 演算モジュールは入力端子, 出力端子, 制御端子及び状態端子をもつ. 入力端子に与えられた値に対して演算を行い, その結果を出力端子に出力する.

観測モジュール: 観測モジュールは入力端子及び状態端子をもつ. 入力端子に与えられた値に対して演算を行い, その結果を状態端子に出力する.

ラッチ, レジスタ: ラッチ, レジスタはいずれも記憶素子である. ラッチは一つの入力端子と一つの出力端子をもつ. 入力端子に与えられた値を記憶し, その値を次のクロックサイクルで出力端子に出力する. レジスタは一つの入力端子と一つの出力端子, 1 ビットの制御端子をもつ. 制御端子の値によって, 入力端子の値を新たに記憶するか (ロード), 既に記憶している値を保持する (ホールド). 記憶している値は次のクロックサイクルで出力端子に出力する.

データ信号線は相異なる回路要素の出力端子と入力端子を接続する. 複数のデータ信号線は同一の出力端子に接続できる (ファンアウト可能) が, 入力端子に接続する信号線は 1 本のみとする.

2.2 端子グラフ

[定義 1] RTL 回路に対して端子グラフ $G = (V, E)$ を次の有向グラフとして定義する.

$$\bullet V = V_{PI} \cup V_{PO} \cup V_{port}$$

V_{PI} は RTL 回路中のすべての外部入力端子の集合, V_{PO} は RTL 回路中のすべての外部出力端子の集合, V_{port} は RTL 回路中の外部入出力端子を除くすべてのデータ入出力端子の集合とする.

$$\bullet E = \{(x, y) \in V \times V \mid \text{RTL 回路中のすべてのデータ信号線の集合}\} \quad \square$$

図 3(a) の RTL 回路に対する端子グラフを図 3(b) に示す.

2.3 連続透明性

[定義 2] コア (RTL 回路) の連続透明性

コアの外部入力端子 I の i 番目のビットを $I(i)$, コアの外部出力端子 O の j 番目のビットを $O(j)$ とする. テストコントローラからの制御入力信号 T により, $I(i)$ と $O(j)$ 間に経路 P が存在し, $I(i)$ に連続して入力される任意の長さの系列を $O(j)$ で連続して観測できるとき, P は連続透明経路であるといい, $I(i)$ と $O(j)$ は連続透明であるという. T により P を実現している状態をコアの形状という. 外部入力端子 I のビット幅を n とする. I に関して, テストコントローラからの同一の制御入力信号 T_I により, I の全ビットを同時に連続透明とする n 個の 1 ビット連続透明経路 (n 個の 1 ビット連続透明経路は同じ外部出力端子を終点とする必要はない) が存在するとき, I は連続透明であるといい, T_I により I と一つ以上の外部

出力端子間に実現される合計 n ビットの経路を I の連続透明経路という。同様に、外部出力端子 O のビット幅を m とする。 O に関して、テストコントローラからの同一の制御入力信号 T_O により、 O の全ビットを同時に連続透明とする m 個の 1 ビット連続透明経路 (m 個の 1 ビット連続透明経路は同じ外部入力端子を始点とする必要はない) が存在するとき、 O は連続透明であるといい、 T_O により O と一つ以上の外部入力端子間に実現される合計 m ビットの経路を O の連続透明経路という。更に、すべての外部入出力端子が連続透明であるとき、コアは連続透明であるという。 □

3. 連続透明化設計法

本章では、与えられた RTL 回路を連続透明な回路に設計変更するための連続透明化設計法を示す。

3.1 問題の定式化

与えられた RTL 回路において、各外部入力端子 (各外部出力端子) が連続透明性を満たさない、すなわち連続透明経路が存在しない場合、回路を連続透明とするために付加回路 (連続透明化要素) を用いて新しい経路と動作を付加する。以下で、新しい経路と動作の追加が必要となる三つの場合と、各場合に対する連続透明化要素の付加について説明する。

(場合 1): 外部入力端子 (外部出力端子) から外部出力端子 (外部入力端子) への経路は存在するが、経路中にマルチプレクサまたはレジスタが存在し、通常動作時にコントローラから供給される制御信号では連続に値を伝搬することができない (図 4)。

付加する連続透明化要素: AND ゲート, OR ゲート, マルチプレクサ

以下のように連続透明化要素を制御信号線に付加することでマルチプレクサの入出力端子間の経路の固定及びレジスタの動作を常にロードとする新しい動作を実現する (図 4)。

- 0 可制御性 (その信号線の値を 0 に設定する能

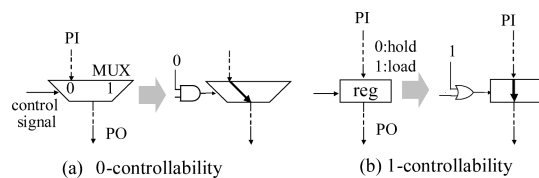


図 4 新しい動作の追加
Fig. 4 Addition of new behavior.

力)が必要な場合, AND ゲートを付加

- 1 可制御性 (その信号線の値を 1 に設定する能力)が必要な場合, OR ゲートを付加

- 0/1 可制御性 (その信号線の値をある形状では 0 に設定し, 他の形状では 1 に設定する能力)が必要な場合, 1 ビットのマルチプレクサを付加

(場合 2): 外部入力端子 (外部出力端子) から外部出力端子 (外部入力端子) への経路は存在するが、経路中に演算モジュールが存在し、任意の値の伝搬ができない (図 5)。

付加する連続透明化要素: スルー機能

演算モジュールにスルー機能を付加することで任意の値の伝搬を可能とする (図 5(a))。スルー機能は、演算モジュールの一つの入力端子に印加した値を変化することなく出力端子へ伝搬させる機能であり、その演算に対する単位元を発生する定数発生器 (図 5(b)) またはマルチプレクサを用いた迂回路 (図 5(c)) として実現可能である。

(場合 3): 外部入力端子 (外部出力端子) から外部出力端子 (外部入力端子) への経路が存在しない (図 6)。

付加する連続透明化要素: マルチプレクサ

マルチプレクサを用いて外部入力端子 (外部出力端子) から外部出力端子 (外部入力端子) への新しい経路を付加する (図 6)。

図 7 に、上記の連続透明化要素を用いた提案手法のテスト機構を示す。与えられた RTL 回路にテストピン (連続透明性を実現するための外部入力) とテストコントローラを付加し、テストコントローラによって新たに付加した経路と動作の制御を行う。図 7 では、加算器 adder 2 のスルー機能、マルチプレクサ MUX

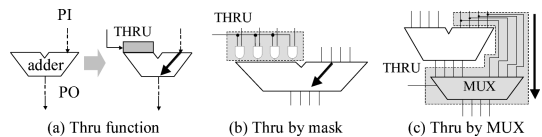


図 5 スルー機能の付加
Fig. 5 Addition of thru function.

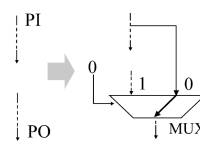


図 6 マルチプレクサによる経路の付加
Fig. 6 Addition of a path with MUX.

1 の 0 可制御性，レジスタ reg 2 の 1 可制御性及びテストコントローラを付加することで，灰色の連続透明経路を実現している．テストピン t_{in} に入力を与えることで，テストコントローラからデータバスに制御信号を供給し，マルチプレクサ MUX 1 の左側の経路，加算器 adder 2 のスルー機能及びレジスタ reg 2 のロードを実現している．

提案手法では，与えられた RTL 回路に対して，図 7 のテスト機構と前述の連続透明化要素を用いて連続透明性を満たすように設計変更を行う．このとき，連続透明化要素の付加による面積オーバーヘッドを最小とすることを提案手法の目的とする．そこで，RTL 回路を連続透明にするための設計変更を次の最適化問題として定義する．

[定義 3] 連続透明化問題

- 入力：RTL 回路（端子グラフ）
- 出力：連続透明な RTL 回路及びその形状
- 最適化目標：付加する連続透明化要素（マルチプレクサ，スルー機能，AND ゲート，OR ゲート）のハードウェア量最小 □

3.2 連続透明化設計法

本節では，連続透明化問題を解くためのアルゴリズムを示す．本アルゴリズムは以下の 2 ステップからなる．

ステップ 1： 端子グラフの拡張

ステップ 2： 連続透明経路の決定と連続透明化要素の付加

ステップ 1 では，3.1 で述べた連続透明化要素を用いて実現可能なすべての経路を端子グラフに辺として

追加する．更にすべての辺に対して，容量とコストを定義することで端子グラフを拡張する．ステップ 2 では，拡張された端子グラフ上ですべての外部入出力端子に対する連続透明経路を決定する．各外部入出力端子に対する連続透明経路は拡張された端子グラフ上で最小費用流問題を解くことで求めることができるが，提案手法では，すべての入出力端子の連続透明経路を最小のコストで実現するために連続透明経路決定問題を整数計画問題として定式化する．以下で各ステップについて詳しく説明する．

ステップ 1： 端子グラフの拡張

端子グラフに対して，以下のように頂点と辺を追加し，その追加した各辺に対してコストと容量を定義する．コストはデータ信号線のビット幅を n としたときのコストである．既存のデータ信号線のコストは 0 とし，追加した辺を含むすべての辺の容量を 1 とする．図 3 (b) の端子グラフを拡張したグラフを図 8 (a) に示す．図 8 (a) において，数字はその辺のコストを表す．

• ダミー入力頂点，及びダミー入力頂点から外部入力への辺

ダミー入力頂点は最大費用流問題を解く際に始点となる頂点である．また，ダミー入力頂点から外部入力への辺のコストは 0 とする．

• ダミー出力頂点，及び外部出力からダミー出力頂点への辺

ダミー出力頂点は最大費用流問題を解く際に終点となる頂点である．また，外部入力からダミー出力頂点への辺のコストは 0 とする．

• マルチプレクサの入力端子から出力端子への辺
マルチプレクサに対して通常動作時には実現できない経路を付加するためには，マルチプレクサの制御入力

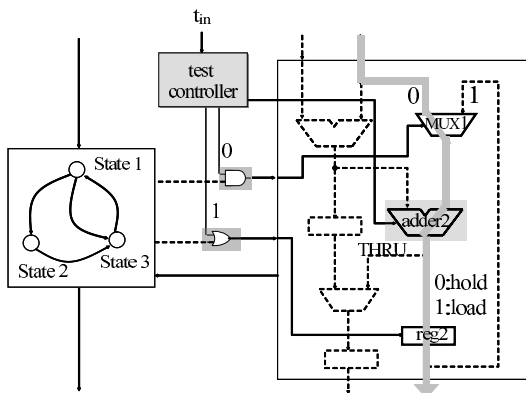


図 7 提案手法のテスト機構

Fig. 7 Test architecture of the proposed method.

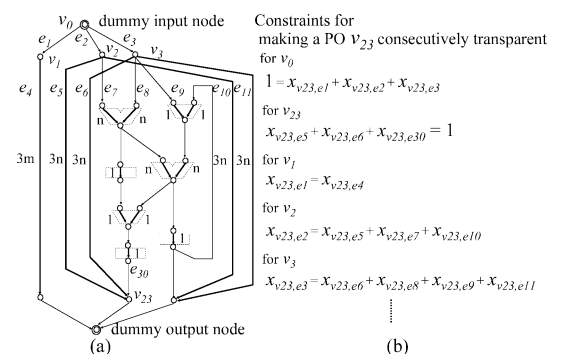


図 8 拡張された端子グラフ及び制約式の例

Fig. 8 An extended port graph and constraint examples.

端子のビット幅を l とすると l 個の AND ゲートまたは OR ゲートが必要であるので、この追加した辺のコストを l とする。

- 演算モジュールの入力端子から出力端子への辺演算モジュールのスルーをマスク素子によって実現する場合、単位元を発生させる必要のある端子数を k とすると、 $k \times n$ 個の AND ゲートまたは OR ゲートが必要であるので、この追加した辺のコストを $k \times n$ とする。一方、演算モジュールのスルーをマルチプレクサによって実現する場合、 n ビットのマルチプレクサが必要である。1 ビットのマルチプレクサは三つのゲートで設計されるものとし、この追加した辺のコストを $3 \times n$ とする。

- ラッチ及びレジスタの入力端子から出力端子への辺

ラッチは通常動作時に常に経路を実現しているので、ラッチの入力端子から出力端子への辺のコストは 0 とする。一方、レジスタの入力端子から出力端子への経路の実現には一つの AND ゲートまたは OR ゲートが必要であるので、この追加した辺のコストを 1 とする。

- 観測モジュールの入力端子に接続する辺の始点となる出力端子からダミー出力頂点への辺

観測モジュールにはデータ出力端子が存在しないため、端子グラフにおいては観測モジュールの入力端子は常に経路の終点となる。新しい経路の付加はマルチプレクサで実現されるので、この追加した辺のコストを $3 \times n$ とする。

- データパスの各外部入力からデータパスのすべての外部出力への辺

各外部入出力の連続透明経路は、端子グラフ中の経路を利用せずに連続透明化要素のマルチプレクサのみを用いた迂回路として実現することも可能である。外部入力から外部出力への辺はこのマルチプレクサによる迂回路として実現されるので、この追加した辺のコストを $3 \times n$ とする。この辺は、各外部入出力の連続透明経路はただか $3 \times n$ のコストで実現できることを保証する。

- コントローラの各外部入力からコントローラのすべての外部出力への辺

各外部入出力の連続透明経路は、コントローラの機能を利用せずに連続透明化要素のマルチプレクサを用いた迂回路として実現する。コントローラの各外部入出力のビット幅を m とすると、1 ビットのマルチプレクサは 3 ゲートで設計されるものとし、この追加した

辺のコストを $3 \times m$ とする。

ステップ 2: 連続透明経路の決定と DFT 要素の付加
ステップ 1 で拡張した端子グラフ G に対して、連続透明化問題を以下の変数を用いた式 (1) から式 (8) で表される整数計画問題として定式化する。

$$x_{v,e} = \begin{cases} 1 & \text{辺 } e \text{ が外部入力または外部出力 } v \text{ の} \\ & \text{連続透明経路に含まれる} \\ 0 & \text{otherwise} \end{cases}$$

$$x_e = \begin{cases} 1 & \text{辺 } e \text{ が一つ以上の外部入出力の} \\ & \text{連続透明経路に含まれる} \\ 0 & \text{otherwise} \end{cases}$$

Minimize:

$$\sum_{e \in E} x_e \cdot \text{cost}(e) \quad (1)$$

$\text{cost}(e)$ は辺 e のコストを表す。

Subject to:

(1) 各外部入力 $v_{pi} \in V_{PI}$ に対して、

$$1 = \sum_{e \in E_{v_{pi}}^{\text{out}}} x_{v_{pi},e} \quad (2)$$

$$\sum_{e \in E_{v_{do}}^{\text{in}}} x_{v_{pi},e} = 1 \quad (3)$$

$$\sum_{e \in E_v^{\text{in}}} x_{v_{pi},e} = \sum_{e \in E_v^{\text{out}}} x_{v_{pi},e} \quad (4)$$

for all $v \in V - \{v_{pi}\} - \{v_{do}\}$

(2) 各外部出力 $v_{po} \in V_{PO}$ に対して、

$$1 = \sum_{e \in E_{v_{di}}^{\text{out}}} x_{v_{po},e} \quad (5)$$

$$\sum_{e \in E_{v_{po}}^{\text{in}}} x_{v_{po},e} = 1 \quad (6)$$

$$\sum_{e \in E_v^{\text{in}}} x_{v_{po},e} = \sum_{e \in E_v^{\text{out}}} x_{v_{po},e} \quad (7)$$

for all $v \in V - \{v_{di}\} - \{v_{po}\}$

(3) 各辺 $e \in E$ に対して

$$x_e \geq x_{v,e} \quad \text{for all } v \in V_{PI} \cup V_{PO} \quad (8)$$

ただし, v_{di}, v_{do} はそれぞれダミー入力頂点, ダミー出力頂点を表すものとする. また, E_v^{in} は頂点 v を終点とする辺の集合とし, E_v^{out} は頂点 v を始点とする辺の集合とする.

式 (2), (3) 及び (4) は各外部入力 v_{pi} に対し, 始点を v_{pi} , 終点をダミー出力頂点 v_{do} とする流量 1 のフローを実現するための条件式である. それぞれの条件式によって端子グラフの各頂点では, 入力辺の流量の和と出力辺の流量の和が保存されている. 同様に式 (5), (6) 及び (7) は各外部出力 v_{po} に対し, 始点をダミー入力頂点 v_{di} , 終点を v_{po} とする流量 1 のフローを実現するための条件式である. 図 8 (b) は, 外部出力端子 v_{23} に対する制約式の例を示す. 更に, 式 (8) によって, 一つ以上の外部入出力端子の連続透明経路に含まれる辺を求め, 式 (1) を目的関数とすることで, 最小のコストで回路を連続透明とする経路を求めることが可能となる. 得られた各入出力端子の連続透明経路, すなわち形状は, 3.1 で述べた連続透明化要素を付加し, その付加した連続透明化要素をテストコントローラで制御することにより実現する. テストコントローラは, テストピンから入力により各連続透明化要素に対する制御信号を出力する組合せ回路である.

表 1 ベンチマーク回路の特性

Table 1 Characteristics of RTL benchmark circuits.

Circuits	Controller		Datapath			Area
	#PI	#PO	[bit]	#PI	#PO	
GCD	1	1	16	2	1	2384
JWF	1	0	16	5	5	10169
LWF	1	0	16	2	2	3377
Paulin	1	0	32	2	2	36203

表 2 連続透明化設計によるオーバーヘッド

Table 2 Hardware and pin overhead of proposed method.

Circuits	Area overhead(%)						Pin overhead
	Proposed					Bypass MUX only	
	Thru	Bypass MUX	Controllability for control signal	Test controller	Total		
GCD	0 (0)	0.38 (1bit: 1)	1.38 (and/or: 6, mux:1)	1.22	2.98	10.11 (1bit: 1, 16bit: 2)	3
JWF	1.26 (2)	0.09 (1bit: 1, 32bit: 1)	0.74 (and/or: 12, mux:3)	1.09	3.18	6.62 (1bit: 1, 16bit: 5)	4
LWF	3.79 (2)	0.27 (1bit: 1)	0.47 (and/or: 4, mux:0)	0.56	5.09	7.88 (1bit: 1, 16bit: 2)	3
Paulin	0.35 (1)	0.67 (1bit: 1, 32bit: 1)	0.04 (and/or: 4, mux:0)	0.09	1.15	1.38 (1bit: 1, 32bit: 2)	3

4. 実験結果

本章では, RTL ベンチマーク回路である GCD, JWF, LWF 及び Paulin に対して提案手法を適用した結果を示す. 四つのベンチマーク回路の特性を表 1 に示す. 表 1 において, “circuits” は回路名を表す. “controller” 欄中の “#PI” 及び “#PO” はそれぞれコントローラの外部入力端子数及び外部出力端子数を表す. これらの四つのベンチマーク回路のコントローラ部の外部入出力端子のビット幅はすべて 1 である. “datapath” 欄中の “|bit|”, “#PI” 及び “#PO” はそれぞれデータバスのデータ信号線のビット幅, 外部入力端子数及び外部出力端子数を表す. “Area” は論理合成後の回路面積 (用いたライブラリのセル面積) を表す. 回路面積は, 論理合成ツール Design Compiler (Synopsys) 及び p2lib ライブラリを用いて求めた.

表 2 に, 連続透明化設計に伴う付加回路の論理合成後の面積オーバーヘッド及び外部ピンオーバーヘッドを示す. 表 2 において, “circuits” は回路名を示す. “Area Overhead” は連続透明化設計前の回路全体の面積に対する連続透明化設計後の面積オーバーヘッドを示す. “proposed” 欄中の “thru”, “bypass MUX”, “controllability for control signal” 及び “test controller” はそれぞれ, 提案手法におけるスルー機能, 迂回路を付加するためのマルチプレクサ, 制御信号線上に付加した新しい動作を実現するための連続透明化要素 (AND ゲート, OR ゲート及びマルチプレクサ), テストコントローラによる面積オーバーヘッドを示す. “total” は提案手法による面積オーバーヘッドの合計を示す. “bypass MUX only” は, 外部入力端子から直接外部出力端子へ接続する迂回経路をマルチプレクサを用いて実現した場合の面積オーバーヘッド (テストコントローラも含

む)を示す。“Pin Overhead”は外部ピンオーバーヘッドを示す。“thru”, “bypass MUX”, “controllability for control signal”及び“bypass MUX only”欄中の括弧内に、それぞれの項目に対して付加された連続透明化要素の数を示す。各回路に対して、提案手法により生成した整数計画問題を解くために、*lp_solve* パッケージ [13] を利用した。その結果、四つの回路に対して生成された整数計画問題を解くために必要な時間は、それぞれ 0.01 秒未満であった。

表 2 より、提案手法による面積オーバーヘッドは平均 3.1%であり、すべてのベンチマーク回路に対し、連続透明経路をマルチプレクサによる迂回路のみで実現した場合の面積オーバーヘッドよりも小さい。これは、外部入出力端子間の連続透明経路を実現する際に、既存の経路を有効利用する方法に加えマルチプレクサによる迂回路で実現する方法も選択肢の一つとすることによって、連続透明経路をたかだかマルチプレクサー一つの面積オーバーヘッドで実現できること保証しているからであると考えられる。また、連続透明経路を実現するための制御信号の値は、各外部入出力端子に対して単一であるため、テストコントローラが組合せ回路で実現可能であり、面積オーバーヘッドが小さくなったと考えられる。提案手法により連続透明性を満たす回路は、各外部入出力端子に対してその端子を連続透明とする形状(テストモード)が存在する。このため、回路の外部入出力端子数の和を n とすると、 n 個の形状と回路の通常動作モードを切り換えるためのテスト用の外部ピンが必要となる。この外部ピンオーバーヘッドは、回路の外部入出力端子数の和を n とすると、 $\lceil \log_2 n + 1 \rceil$ 本必要である。本論文では、回路はコアとして SoC に組み込まれ、各回路に付加されるすべての外部ピンは、SoC 内部に組み込まれたテストコントローラで制御することを想定しているため、SoC の外部ピンとはならない。

5. む す び

本論文では、レジスタ転送レベル回路に対する連続透明化設計法を提案した。提案手法では、データパス中の既存の経路を有効利用することで小さい面積オーバーヘッドですべての外部入出力に対する連続透明経路を実現している。ベンチマーク回路を用いた実験により、提案手法による面積オーバーヘッドは、連続透明経路をマルチプレクサによる迂回路のみで実現した場合の面積オーバーヘッドよりも小さいことを示した。連続

透明なコアは、形状を選択することで、その入力端子に入力される任意の長さの系列を値を変えることなく出力側に連続したクロックサイクルで伝搬可能、または出力端子から出力される任意の長さの系列を入力側から連続したクロックサイクルで伝搬可能である。したがって連続透明な回路は、コアとして SoC に組み込まれた場合、他のコアへの連続テストアクセスのための経路としての利用が可能となる。

謝辞 本研究に際し、多くの貴重な意見を頂いた本学の井上美智子助教授、大竹哲史助手、並びに広島市立大学の井上智生助教授、市原英行助手及び本学のコンピュータ設計学(旧情報論理学)講座の諸氏に深く感謝します。本研究は一部、日本学術振興会科学技術研究費補助金・基盤研究 B(2)(課題番号 15300018)の研究助成、及び、新エネルギー・産業技術総合開発機構(NEDO)から半導体理工学研究センター(STARC)に委託された「SoC 先端設計技術の研究開発」の一部として奈良先端科学技術大学院大学に再委託され実施されています。

文 献

- [1] Y. Zorian, E.J. Marinissen, and S. Dey, “Testing embedded-core based system chips,” Proc. 1998 Int. Test Conf., pp.130–143, Oct. 1998.
- [2] N.A. Toubia and B. Pouya, “Testing embedded cores using partial isolation rings,” Proc. 15th VLSI Test Symp., pp.10–16, May 1997.
- [3] L. Whetsel, “An IEEE 1149.1 based test access architecture for ICs with embedded cores,” Proc. 1997 Int. Test Conf., pp.69–78, Nov. 1997.
- [4] S. Bhatia, T. Gheewala, and P. Varma, “A unifying methodology for intellectual property and custom logic testing,” Proc. 1996 Int. Test Conf., pp.639–648, Oct. 1996.
- [5] T. Ono, K. Wakui, H. Hikima, Y. Nakamura, and M. Yoshida, “Integrated and automated design-for-testability implementation for cell-based ICs,” Proc. 6th Asian Test Symp., pp.122–125, Nov. 1997.
- [6] P. Varma and S. Bhatia, “A structured test reuse methodology for core-based system chips,” Proc. 1996 Int. Test Conf., pp.294–302, Oct. 1998.
- [7] M. Nourani and C.A. Papachristou, “Structural fault testing of embedded cores using pipelining,” J. Electron. Test., Theory Appl., vol.15, pp.129–144 1999.
- [8] I. Ghosh, N.K. Jha, and S. Dey, “A low overhead design for testability and test generation technique for core-based systems-on-a-chip,” IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.18, no.11, pp.1661–1676, Nov. 1999.
- [9] I. Ghosh, S. Dey, and N.K. Jha, “A fast and low cost testing technique for core-based system-chips,” IEEE

- Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.19, no.8, pp.863-877, Aug. 2000.
- [10] S. Ravi, G. Lakshminarayana, and N.K. Jha, "Testing of core-based systems-on-a-chip," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.20, no.3, pp.426-439, March 2001.
- [11] 米田友和, 藤原秀雄, "連続可検査性に基づくコアベースシステムオンチップのテスト容易化設計法," 信学論 (D-I), vol.J85-D-I, no.2, pp.173-183, Feb. 2002.
- [12] T. Yoneda and H. Fujiwara, "Design for consecutive testability of system-on-a-chip with built-in self testable cores," J. Electron. Test., Theory Appl. (JETTA), Special Issue on Plug-and-Play Test Automation for System-on-a-Chip, vol.18, no.4/5, pp.487-501, Aug. 2002.
- [13] M. Berkelaar, *lp_solve*, version 3.2, Eindhoven University of Technology, The Netherlands, ftp://ftp.ics.ele.tue.nl/pub/lp_solve

(平成 16 年 3 月 17 日受付, 7 月 9 日再受付)



米田 友和 (正員)

平 10 阪大・工・情報システム卒。平 13 奈良先端科学技術大学院大学博士前期課程了。平 14 同大博士後期課程了。現在奈良先端大・情報科学研究科助手。VLSI CAD, テスト容易化設計, システムオンチップのテストアーキテクチャ及びスケジューリングに関する研究に従事。



藤原 秀雄 (正員:フェロー)

昭 44 阪大・工・電子卒。昭 49 同大学院博士課程了。同大・工・電子助手, 明治大・工・電子通信助教授, 情報科学教授を経て, 現在奈良先端大・情報科学教授。昭 56 ウォータールー大客員助教授。昭 59 マッギル大客員準教授。論理設計論, フォールトトレランス, 設計自動化, テスト容易化設計, テスト生成, 並列処理, 計算複雑度に関する研究に従事。著書「Logic Testing and Design for Testability」(MIT Press)など。大川出版賞, IEEE Computer Society Outstanding Contribution Award, IEEE Computer Society Meritorious Service Award など受賞。情報処理学会会員, IEEE Computer Society Golden Core Member, IEEE Fellow。