

回路疑似変換による順序回路テスト生成の一手法

大竹 哲史[†] 井上 智生[†] 藤原 秀雄[†]

組合せテスト生成複雑度でテスト生成可能な順序回路では、その順序回路のテスト生成問題は、フリップフロップを信号線に置き換えた（組合せ変換した）組合せ回路におけるテスト生成問題に帰着できる。本論文では、この性質を一般の順序回路に拡張する。具体的には、与えられた順序回路において、平衡構造を有する部分回路を抽出し、その部分回路を組合せ変換した順序回路についてテスト生成を行い、得られたテスト系列を元の順序回路のテスト系列に戻すというテスト生成法を提案する。この手法では、組合せ変換によってフリップフロップ数が減少するので、元の順序回路よりフリップフロップ数が減少している分だけテスト生成時間の短縮が期待できる。提案する手法の有効性を ISCAS '89 ベンチマーク回路による実験によって評価する。

An Approach to Sequential Test Generation by Circuit Pseudo-transformation

SATOSHI OHTAKE,[†] TOMOO INOUE[†] and HIDEO FUJIWARA[†]

The test generation problem for a sequential circuit *capable of generating tests with combinational test generation complexity* can be reduced to that for the combinational circuit formed by replacing each FF in the sequential circuit by a wire. In this paper, we consider an application of this approach to general sequential circuits. We propose a test generation method using *circuit pseudo-transformation* technique: given a sequential circuit, we extract a subcircuit with *balanced structure* which is capable of generating tests with combinational test generation complexity, replace each FF in the subcircuit by wire, generate test sequences for the transformed sequential circuit, and finally obtain test sequences for the original sequential circuit. We also estimate the effectiveness of the proposed method by experiment with ISCAS '89 benchmark circuits.

1. はじめに

近年のデバイス技術の進歩により、デジタル集積回路の集積度が向上し、大規模なシステムをVLSI上に実装することが可能となった。しかし、回路の大規模化にともない、テストはますます困難となり、費用のかかる問題となっている。

増大するテスト費用を抑える1つの方法として、テスト生成時間を短縮する方法が考えられる。順序回路の場合、フリップフロップ（FF）が存在するため、テスト生成アルゴリズムの探索空間はFF数を n とすると 2^n に依存する。このFF数を減らすことができれば、テスト生成アルゴリズムの探索空間が縮小され、FF数が減った分だけテスト生成時間が短縮されることが期待できる。

FF数を減らす方法として、部分スキャン設計法が

ある¹⁾。部分スキャン設計法では、一部のFFをスキャンFFに変換し、テスト生成の対象回路のFF数を減らすことにより、テスト生成時間を短縮することができる。部分スキャン設計には、組合せテスト生成複雑度でテスト生成可能な回路構造である平衡構造²⁾や内部平衡構造³⁾に基づく手法が提案されている。しかし、これらの手法ではテスト生成は容易になるが、付加回路による面積オーバーヘッド、性能劣化などの問題が生じる。これらの問題を避けるために、疑似的に変更した回路を用いてテスト生成を行う手法として、リタイミングを応用したソフトウェア変換法⁴⁾が提案されているが、テストビリティの保存性（4章で述べる）という制約があるため、故障検出率やテスト生成時間の点で、十分な効果が得られていない。

本論文では、テストビリティの保存性の制約を緩和することにより、FF数のより少ない回路に疑似的に変換してテスト生成を行う手法を提案する。具体的には、与えられた順序回路において、平衡構造となる部分回路のFFを疑似的に信号線に置き換えて（回路疑

[†] 奈良先端科学技術大学院大学情報科学研究科
Graduate School of Information Science, Nara Institute
of Science and Technology (NAIST)

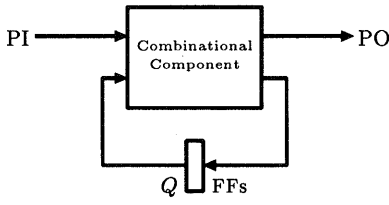


図1 順序回路 S
Fig.1 Sequential circuit S .

似変換して) FF 数を減らしてテスト生成を行い, 得られたテスト系列を元の順序回路のテスト系列に戻すというテスト生成法を示す. これにより, FF 数を減らすことができ, テスト生成時間の短縮と, 故障検出率の向上が可能となる. また, 本手法では, 一部の FF に保持 (ホールド) 機能を持たせる設計変更を必要とするが, それにともなう面積オーバーヘッドは小さい.

本論文では, 2 章で回路疑似変換を定義し, 3 章では, 回路疑似変換した回路でテスト生成を行って得られたテスト系列を用いて, 元の回路をテストするための d -時刻保持変換と d -時刻保持テスト法を定義する. 4 章で, 組合せ回路疑似変換と d -時刻保持変換によるテストバリエーションの保存性を考察し, テスト生成問題の帰着性について述べる. 5 章では, 回路疑似変換を用いたテスト生成法を提案する. 6 章で提案手法の有効性を ISCAS '89 ベンチマーク回路を用いた実験によって評価し, 本手法により, 多くの回路についてテスト生成時間が短縮され, 一部の回路について故障検出率が向上した結果を示す.

2. 回路疑似変換

定義 1 (平衡構造) 順序回路 S に閉路が存在しないとき, S は無閉路構造であるという. 無閉路構造の順序回路 S_A の, 任意の外部入力と任意の外部出力の対について, その入出力を結ぶどの経路の順序深度 (経路上の FF 数) も等しいとき, S_A は平衡構造であるという.

定義 2 (回路疑似変換) 回路 S を疑似的に S^T とは別の回路に変換することを, 回路疑似変換と呼ぶ. ここで, “疑似的” とはテスト生成を行うために, “仮に” 変換するという意味であり, 物理的に回路を変換するのではない. また, S を回路疑似変換 T した回路を S^T と書く.

定義 3 (核回路) 順序回路 S (図 1) 中のすべての FF の集合を Q とする. Q の部分集合を Q_E として, S から Q_E を取り除いた回路 S_K を核回路と呼ぶ. このとき, Q_E の FF を外部 FF, $Q_I = Q - Q_E$

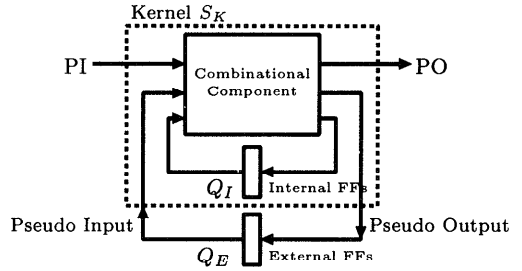


図2 S を核回路 S_K と外部 FF Q_E に分割
Fig.2 Kernel circuit S^K and set of external FFs Q_E .

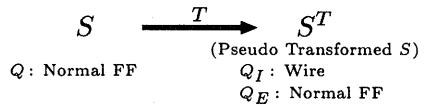


図3 組合せ回路疑似変換
Fig.3 Combinational circuit pseudo-transformation.

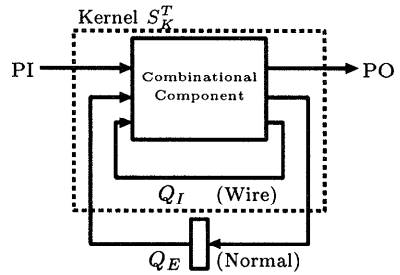


図4 順序回路 S^T
Fig.4 Sequential circuit S^T .

の FF を内部 FF と呼ぶ. ただし, 核回路は無閉路構造とする. 核回路の入力には, S の外部入力と Q_E の FF からの疑似入力があり, 出力には, S の外部出力と Q_E の FF への疑似出力がある (図 2 参照). また, 核回路の入力から出力への経路の最大の順序深度を d として, この d を核回路の順序深度と呼ぶ.

定義 4 (組合せ回路疑似変換) 順序回路 S が無閉路構造の核回路と外部 FF に分割されたとき, 内部 FF を信号線に置き換える変換を組合せ回路疑似変換と呼ぶ (図 3, 図 4 参照).

3. d -時刻保持テスト法

定義 5 (k -クロック) k を 0 以上の整数定数とする. 通常のクロックの 1 クロックサイクルを $k+1$ に伸ばしたクロックを k -クロックと呼ぶ.

定義 6 (k -時刻保持 FF) k を 0 以上の整数定数とする. k 時刻値を保持した後, 次の時刻で値を取り込む FF を k -時刻保持, 1-時刻取り込み FF と呼び, k -時刻保持 FF と略記する.

通常の FF は 0-時刻保持 FF である. 通常のクロック



図5 通常 FF
Fig. 5 Normal FF.

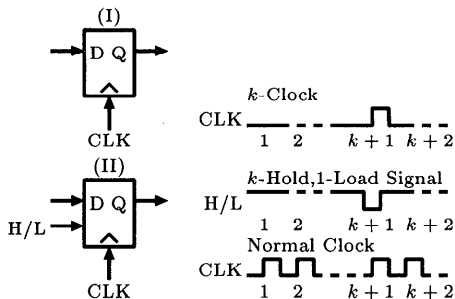


図6 k-時刻保持 FF
Fig. 6 k-clock hold FF.

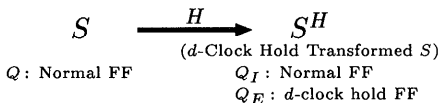


図7 d-時刻保持変換
Fig. 7 d-clock hold transformation.

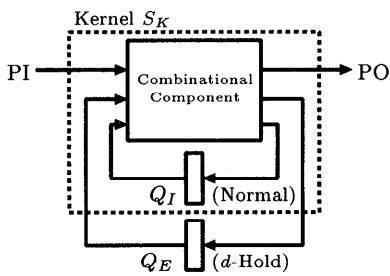


図8 順序回路 S^H
Fig. 8 Sequential circuit S^H .

クで動作する FF を図 5 に示す. k -時刻保持 FF を実現する方法は, 図 6 (I) のように, k -クロックを用いる方法や, 図 6 (II) のように Hold/Load 信号による方法などが考えられる.

定義 7 (d -時刻保持変換) 順序回路 S (図 1) が核回路と外部 FF に分割されたとき (図 2), 核回路の順序深度を d として, 外部 FF を d -時刻保持 FF に置き換える変換を d -時刻保持変換と呼び, S を d -時刻保持変換 H した回路を S^H と書く (図 7, 図 8 参照).

定義 8 (k -時刻保持系列) 系列 t の系列長を n として, t の各ベクトルを $t_i (i = 1, 2, \dots, n)$ とする. た

だし, k を 0 以上の整数定数として, n を $k+1$ の倍数とする. $1 \leq m \leq n/(k+1)$ となる任意の整数 m について, $(m-1)(k+1) < i \leq m(k+1) \wedge (m-1)(k+1) < j \leq m(k+1) \wedge i \neq j$ ならば $t_i = t_j$ となるとき, t を k -時刻保持系列と呼ぶ.

定義 9 (d -時刻保持テスト法) 順序回路 S を核回路と外部 FF に分割したとき, 核回路の順序深度を d として d -時刻保持変換した回路を S^H とする. S^H に対して, テスト系列に d -時刻保持系列を用いて, 外部 FF は d -クロックで動作させるテスト法を d -時刻保持テスト法と呼ぶ.

この d -時刻保持テスト法では, S^H は次のような動作をする.

- 外部入力では, 核回路全体に信号が伝搬する $d+1$ 時刻の間, d -時刻保持系列により入力値が保持される.
- 疑似入力では, 核回路全体に信号が伝搬する $d+1$ 時刻の間, d -時刻保持 FF により値が保持される.
- 外部出力では, 通常のテスト法と同様に出力値が観測される.
- 疑似出力では, d -時刻保持 FF により, 核回路全体に信号が伝搬した $d+1$ 時刻目の値が FF に取り込まれる. 核回路全体に信号が伝搬していない d 時刻の間の値は無視される.

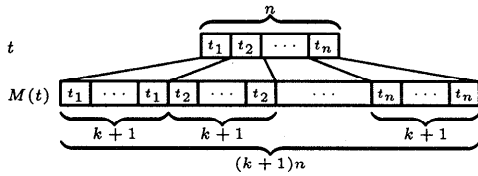
4. テスト生成問題帰着性

以下では, 回路が与えられると, その回路に対するテスト法も与えられるものとする.

定義 10 (テスト可能) 回路 C の故障を f とし, C_f を f によって故障した回路とする. また, C の入力系列 t に対して, t を C に印加して得られた出力系列と, t を C_f に印加して得られた出力系列が異なる場合, t を f のテスト系列と呼び, f は t でテスト可能であるという.

定義 11 (テストバリティ保存性) 回路 C を変換 τ によって変換して得られた回路を C^τ とする. C, C^τ の全故障集合をそれぞれ F, F^τ とする. このとき, 次のすべての条件を満たす変換 τ はテストバリティを保存するという.

- (i) F から F^τ への写像 φ_τ が存在する.
- (ii) F の任意の故障 f に対して, f が C のテスト法でテスト可能ならば, $\varphi_\tau(f)$ は C^τ のテスト法でテスト可能である.
- (iii) F の任意の故障 f に対して, f が C のテスト法でテスト不可能ならば, $\varphi_\tau(f)$ は C^τ のテスト法でテスト不可能である.

図9 k -時刻保持系列変換Fig. 9 k -clock hold sequence transformation.

定義 12 (テスト生成問題帰着性) 回路 C を変換 τ によって変換して得られた回路を C^τ とする。回路 C , C^τ の全故障集合をそれぞれ F , F^τ とする。このとき、 C と C^τ について次の 2 つの条件を満たすならば、 C のテスト生成問題は C^τ のテスト生成問題に帰着できるという。

- (i) 変換 τ はテストバリエーションを保存する。
- (ii) 変換 τ における F から F^τ への写像を φ_τ とする。 $\varphi_\tau(F)$ のうち C^τ のテスト法でテスト可能な故障の集合を F' とする。 F' の任意の故障 f について、 $\bigcup_{t \in T(f)} \{\sigma(t)\} \subseteq \bigcap_{g \in \varphi_\tau^{-1}(f)} T(g)$ となる写像 σ が存在する。ここで、 $T(f)$ は故障 f に対応する回路のテスト法で f をテスト可能なすべてのテスト系列の集合とする。

定義 13 (k -時刻保持系列変換) 系列 t の系列長を n として、 t の各ベクトルを t_i ($i = 1, 2, \dots, n$) とする。一方、 k を 0 以上の整数定数として、系列長が $n(k+1)$ である k -時刻保持系列 t^k を考える。この t^k の各ベクトルを t_i^k ($i = 1, 2, \dots, n(k+1)$) とする。 $(i-1)(k+1) < j \leq i(k+1)$ となる任意の整数 i, j に対して $t_j^k = t_i$ となるような t から t^k への変換を k -時刻保持系列変換と呼び、この変換を M と書く。

例 1 (k -時刻保持系列変換) 系列長が n の系列 t が与えられたとき、 t を k -時刻保持系列変換 M すると、図 9 のように k -時刻保持系列 $M(t)$ が得られる。

順序回路 S (図 1) を核回路 S_K と外部 FF Q_E に分割したとする (図 2)。内部 FF を Q_I 、核回路の順序深度を d する。 S を d -時刻保持変換した回路を S^H (図 8)、 S を組合せ回路疑似変換した回路を S^T (図 4) とする。 S^H のテスト法は d -時刻保持テスト法、 S^T のテスト法は通常のテスト法である。ここで、 S^H から S^T への変換 HT を考える。変換 HT は、 S^H の通常 FF を信号線に、 d -時刻保持 FF を通常 FF に置き換える変換である。

これらの変換では、FF が信号線や d -時刻保持 FF に置き換えられるか、または d -時刻保持 FF が FF に置き換えられるだけである。したがって、信号線に変換される FF 以外の故障については、変換後も同じ故

障を対応させることができる。一方、信号線に変換される FF については、FF の入出力の故障は信号線の故障に代表させることができる。以下では、変換にともなう故障の対応としては、このような対応を考える。

S^H と S^T の間のテスト生成問題帰着性に関して、定理 1, 2 が成立する。

定理 1 S_K が平衡構造のとき、 S^H のテスト生成問題は S^T のテスト生成問題に帰着できる。

(証明) テスト生成問題帰着性の定義 12 の条件 (i), (ii) を満たすことを示す。

(定義 12-(i)) 変換 HT がテストバリエーションを保存することを示すために、定義 11 の条件 (i), (ii), (iii) が成り立つことを以下順に示す。

(i) d -時刻保持変換と組合せ回路疑似変換の定義より、 S^H と S^T は Q_I を除いて同形である (図 8, 図 4 参照)。したがって、その部分回路中の故障は、 S^H と S^T で 1 対 1 に対応する。一方、 S^H における Q_I の FF の入力故障と出力故障は出力故障で代表でき、それは、 S^T における対応する信号線の故障に 1 対 1 に対応させることができる。

(ii) S^H の任意の故障を f^H として、 f^H に対応する S^T の故障を f^T とする。 f^H が S^H のテスト法でテスト可能ならば、 f^T は S^T のテスト法でテスト可能であることを示す。 f^H がテスト可能ならば、テスト系列 t^H が存在する。 S^H のテスト法は d -時刻保持テスト法なので、 t^H は d -時刻保持系列である。ここで、 t^H の系列長を $(d+1)n$ とする。 S^H に入力系列 t^H を印加して得られる出力系列を $O^H = o_1^H, o_2^H, \dots, o_{(d+1)n}^H$ とする。また、 O^H から $d+1$ 周期でベクトルを抽出した系列を $O^{H'} = o_{d+1}^H, o_{(d+1)2}^H, \dots, o_{(d+1)n}^H$ とする。 d -時刻保持系列変換 M の逆変換を M^{-1} として、 S^T に $M^{-1}(t^H)$ を印加して得られる出力系列を $O^T = o_1^T, o_2^T, \dots, o_n^T$ とする。このとき、核回路が平衡構造なので $O^T = O^{H'}$ であり、 f^H の影響は $O^{H'}$ に現れることから、 f^T の影響は O^T に現れる。さらに、(i) で示した f^H と f^T の対応関係より、 f^H が S^H のテスト法でテスト可能ならば、 S^T において f^T は $M^{-1}(t^H)$ でテスト可能である。

(iii) S^T の任意の故障を f^T として、 f^T に対応する S^H の故障を f^H とする。 f^T が S^T のテスト法でテスト可能ならば、 f^H は S^H のテスト法でテスト可能であることを示す。 f^T がテスト可能ならば、 f^T のテスト系列 t^T が存在する。ここで、 t^T の系列長を n とする。 S^T に入力系列 t^T を印加して得られる出力系列を $O^T = o_1^T, o_2^T, \dots, o_n^T$ とする。 S^H のテスト法は d -時刻保持テスト法なので、 f^H のテスト系列は

d -時刻保持系列でなければならない。 d -時刻保持系列変換を M として、 S^H に $M(t^T)$ を印加して得られる出力系列を $O^H = o_1^H, o_2^H, \dots, o_{(d+1)n}^H$ とする。また、 O^H から $d+1$ 周期でベクトルを抽出した系列を $O^{H'} = o_{d+1}^H, o_{(d+1)2}^H, \dots, o_{(d+1)n}^H$ とする。このとき、核回路が平衡構造なので $O^{H'} = O^T$ であり、 f^T の影響は O^T に現れることから、 f^H の影響は $O^{H'}$ に現れる。さらに、(i) で示した f^H と f^T の対応関係より、 f^T が S^T のテスト法でテスト可能ならば、 S^H において f^H は $M(t^T)$ でテスト可能である。

(定義 12-(ii)) 上記 (定義 12-(i)) の (i) より故障が 1 対 1 に対応し、(定義 12-(i)) の (iii) より S^T において S^T のテスト法でテスト可能な任意の故障 f^T のテスト系列の集合 $\mathcal{T}(f^T)$ に対して、 $\bigcup_{t \in \mathcal{T}(f^T)} \{M(t)\}$ は f^T に対応する S^H の故障 f^H の S^H のテスト法におけるテスト系列となっている。すなわち、 $\bigcup_{t \in \mathcal{T}(f^T)} \{M(t)\} = \mathcal{T}(f^H)$ となる変換 M が存在する。ここで、 $\mathcal{T}(f)$ は故障 f に対応する回路のテスト法で f をテスト可能なすべてのテスト系列の集合とする。

以上より、定理 1 は証明された。 □

定理 2 S_K が無閉路構造のとき、 S^H のテスト生成問題は、必ずしも S^T のテスト生成問題に帰着できない。

(証明) 変換 HT がテスト安定性を保存しない例を示す。

図 10 の順序回路 S_1 を考える。核回路が無閉路構造になるように FF2 を外部 FF に選び、 S_1 から FF2 を取り除いた部分回路を核回路とする。核回路の順序深

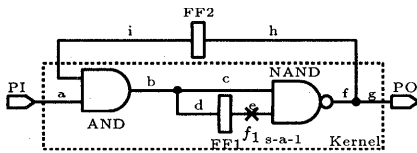


図 10 順序回路 S_1
Fig. 10 Sequential circuit S_1 .

度は 1 である。 S_1 を 1-時刻保持変換した回路を S_1^H (図 11), S_1 を 組合せ回路疑似変換した回路を S_1^T (図 12) とする。このとき、 S_1 の信号線 e の 1-縮退故障 f_1 を考える。図 11 と図 12 から分かるように、 S_1^H と S_1^T にも f_1 に対応する故障 f_1^H と f_1^T がそれぞれ存在する。 S_1^H におけるテスト法は 1-時刻保持テスト法で、 f_1^H のテスト系列 t_1^H は、1-時刻保持系列 $t_1^H = [0011]$ となり、3 時刻目で外部出力に誤りが観

測できる (図 13 参照)。これに対して、通常のテスト法の S_1^T において、信号線 c に 1, e に 0 を同時に設定できないので f_1^T はテスト不可能である (図 12 参照)。

以上より、定理 2 は証明された。 □

S と S^T の間のテスト生成問題帰着性に関して、定理 3, 系 1 が成立する。

定理 3 S_K が平衡構造のとき、 S のテスト生成問題は、必ずしも S^T のテスト生成問題に帰着できない。(証明) 組合せ回路疑似変換 T がテスト安定性を保存しない例を示す。

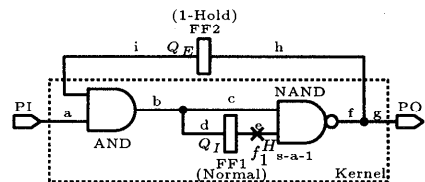


図 11 順序回路 S_1^H
Fig. 11 Sequential circuit S_1^H .

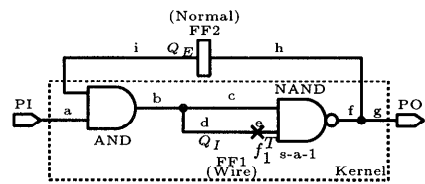


図 12 順序回路 S_1^T
Fig. 12 Sequential circuit S_1^T .

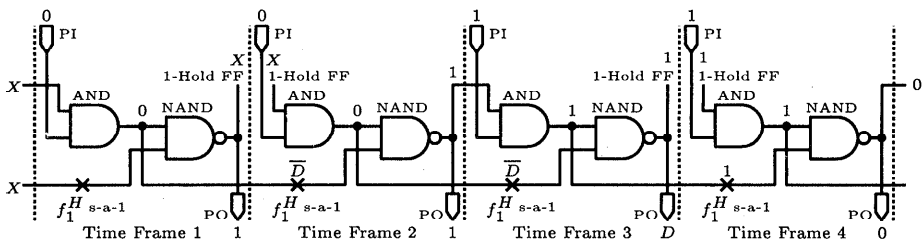


図 13 S_1^H のタイムフレーム展開
Fig. 13 Time frame expansion of S_1^H .

図 14 の順序回路 S_2 を考える．核回路が平衡構造になるように FF3 と FF4 を外部 FF に選び， S_2 から FF3 と FF4 を取り除いた部分回路を核回路とする． S_2 を組合せ回路疑似変換した回路を S_2^T (図 15) とする．このとき， S_2 の信号線 c の 1-縮退故障 f_2 を考える．図 15 から分かるように， S_2^T にも f_2 に対応する故障 f_2^T が存在する． S_2 における， f_2 のテスト

ト系列 t_2 は， $t_2 = \begin{bmatrix} 0XX1XX \\ 01X01X \end{bmatrix}$ となる (図 16 参

照)．これに対して， S_2^T において FF3 に 1，FF4 に 1 を同時に設定できないので， f_2^T はテスト不可能である．(図 15 参照)．

以上より，定理 3 は証明された． □

系 1 S_K が無閉路構造のとき， S のテスト生成間

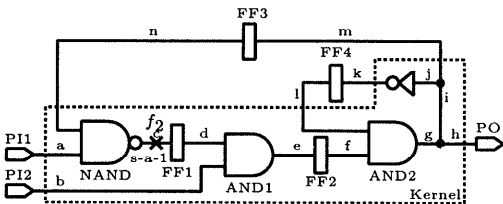


図 14 順序回路 S_2
Fig. 14 Sequential circuit S_2 .

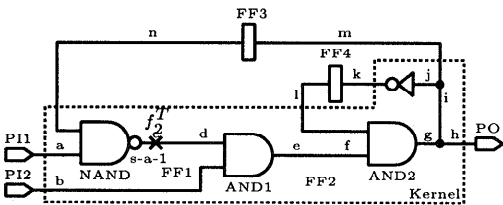


図 15 順序回路 S_2^T
Fig. 15 Sequential circuit S_2^T .

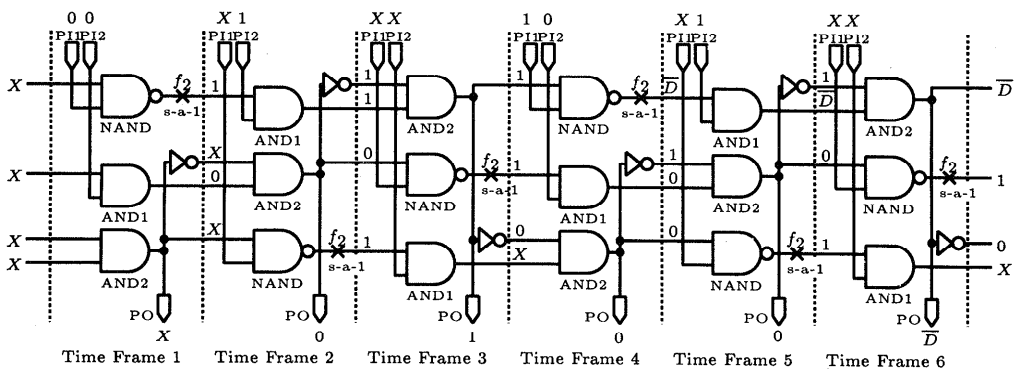


図 16 S_2 のタイムフレーム展開
Fig. 16 Time frame expansion of S_2 .

題は，必ずしも S^T のテスト生成問題に帰着できない．(証明) { 無閉路構造の順序回路 } \supset { 平衡構造の順序回路 } なので，定理 3 より明らか． □

5. 回路疑似変換によるテスト生成法

この章では，核回路に平衡構造を採用した，組合せ回路疑似変換によるテスト生成法を提案する．以下では，与えられた順序回路を S ，平衡構造となる核回路を S_K ，外部 FF を Q_E とする．また，内部 FF を Q_I ，核回路の順序深度を d とする．

組合せ回路疑似変換によるテスト生成は次のように行う．

- (Step 1) S (図 1) を S_K と Q_E に分割 (図 2) する．このとき，核回路の順序深度を d とする．
- (Step 2) S を組合せ回路疑似変換して S^T (図 4) を求める．
- (Step 3) S^T に対して順序回路テスト生成アルゴリズムを適用してテスト生成し，テスト系列 t^T を求める．
- (Step 4) テスト系列 t^T を d -時刻保持系列変換して t^H を求める．

例 2 (組合せ回路疑似変換によるテスト生成)

図 17 の順序回路 S_3 について考える．ここで， C_1, C_2, C_3, C_4 は組合せ回路部分を表す．(Step 1) で図 17 のように， S_3 を核回路 S_{3K} と外部 FF $Q_E = \{FF5, FF6\}$ に分割する．核回路の順序深度は 2 である．(Step 2) で内部 FF $Q_I = \{FF1, FF2, FF3, FF4\}$ を信号線に置き換えた回路を S_3^T とする (図 18)．(Step 3) でテスト生成したテスト系列 t^T を (Step 4) で 2-時刻保持系列変換して，図 19 のようにテスト系列 t^H を得る．

次に， S のテスト実行は次のようなテスト法で行う．

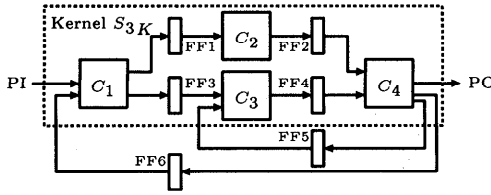


図 17 順序回路 S_3
Fig. 17 Sequential circuit S_3 .

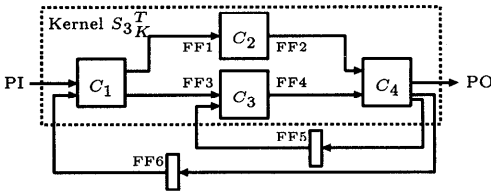


図 18 順序回路 S_3^T
Fig. 18 Sequential circuit S_3^T .

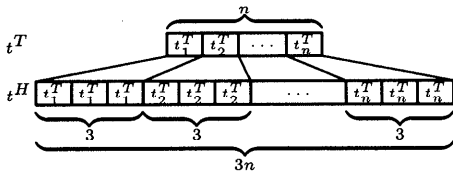


図 19 t^T の 2-時刻保持系列 t^H
Fig. 19 2-clock hold sequence of t^T .

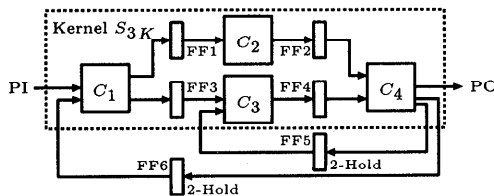


図 20 順序回路 S_3^H
Fig. 20 Sequential circuit S_3^H .

(Step 1) S を d -時刻保持変換して S^H (図 8) を求める。

(Step 2) 組合せ回路疑似変換によるテスト生成で得られたテスト系列 t^H を用いて、 d -時刻保持テスト法により S^H に対してテスト実行する。

例 3 (テスト実行) 例 2 の回路 S_3 と得られたテスト系列 t^H について考える。(Step 1) では、 S_3 を 2-時刻保持変換して図 20 の S_3^H を得る。(Step 2) でこのテスト系列 t^H で S_3^H をテスト実行する。

組合せ回路疑似変換によるテスト生成とテストの流れを図 21 に示す。

ここで、組合せ回路疑似変換のテスト安定性について考える。4 章で述べたように、核回路が平衡構造のとき、一般に S^H は S^T のテスト生成問題

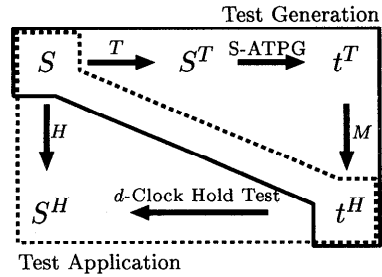


図 21 組合せ回路疑似変換によるテスト生成とテスト実行
Fig. 21 Test generation/application by combinational circuit pseudo-transformation.

に帰着できる (定理 1) が、 S は S^T のテスト生成問題に帰着できない (定理 3)。すなわち、元の回路に対してテスト生成して得られるテスト系列と、本手法によって得られるテスト系列では、同じ故障を検出できるとは限らない。そこで、検出できる故障の違いについて、次章の実験によって評価する。

6. 実験結果

この章では、与えられた順序回路 S に対してテスト生成を行った結果と、 S を組合せ回路疑似変換した順序回路 S^T でテスト生成を行った結果を示す。テスト生成ツールには、FASTEST⁵⁾を用いた。使用した計算機は S-4/20 model 712 (Fujitsu) である。 S には、ISCAS '89 ベンチマーク回路を用いた。ISCAS '89 ベンチマーク回路の回路特性を表 1 に示す。circuit の欄は回路名を示し、#gate, #PI, #PO, #FF の欄はそれぞれゲート数、外部入力数、外部出力数、FF 数を表す。

組合せ回路疑似変換は、C 言語で作成したプログラムによって行った。核回路の回路構造には平衡構造を用いている。組合せ回路疑似変換を行うプログラムは、次のような構成である。

(Step 1) 与えられた回路 S を、無閉路構造を有する部分回路 S_A と外部 FF Q_A に分割する。

(Step 2) S_A を平衡構造を有する部分回路 S_B と外部 FF Q_B に分割する。

(Step 3) S_B を核回路として、 $Q_A \cup Q_B$ を外部 FF とし、内部 FF をバッファに置き換える。

(Step 1) では、MFVFS (Minimum Feedback Vertex Set)⁶⁾を求めることによって無閉路構造を有する部分回路を求めている。(Step 2) の平衡構造を求めている方法は文献 2) で述べられているが、実験では、次のような方法で簡単に求めた。

無閉路構造に変換した回路に対して、各入力点 (外

表1 ISCAS '89 ベンチマーク回路の回路特性

Table 1 Circuit characteristics of ISCAS '89 benchmark circuits.

circuit	#gate	#PI	#PO	#FF
s382	99	3	6	21
s400	106	3	6	21
s444	119	3	6	21
s641	107	35	24	19
s713	139	35	23	19
s953	311	16	23	29
s1196	388	14	14	18
s1238	428	14	14	18
s1423	490	17	5	74
s5378	1004	35	49	179
s9234.1	2027	36	39	211
s9234	2027	19	22	228
s13207.1	2573	62	152	638
s13207	2573	31	121	669

部入力と疑似入力) ごとに平衡構造に変換していく。具体的には、1つの入力点について、その入力点から外部出力と疑似出力に至る経路のみを対象として、以下の処理を行う。

(Step 1) 入力点の順序深度を0とし、始点とする。

(Step 2) 各始点(始点が入力点の場合は1つ)について、その始点から直接到達可能な収斂点(入力点からの複数の経路を入力とするゲート)を求める。ここで、直接到達可能とは、始点から他の収斂点を通らずに到達できるという意味である。収斂点が存在しないならば、処理を終了する。

(Step 3) (Step 2) で得られた各収斂点について、(Step 3.1)、(Step 3.2) の処理を行う。

(Step 3.1) 収斂点の各入力について、直前の収斂点または入力点を求め、その点の順序深度とその点から収斂点までの経路上のFF数の和を、その収斂点の入力の順序深度とする。

(Step 3.2) (Step 3.1) で求めた収斂点のすべての入力の順序深度を比較して外部FFを選択する。具体的には、順序深度0の収斂点の入力が存在すれば、順序深度0以外の収斂点の入力となる経路について、収斂点の直前のFFすべてを外部FFとして、その収斂点の順序深度を0とする。順序深度0の収斂点の入力が存在しない場合は、最も多い順序深度をその収斂点の順序深度として、それ以外の順序深度を持つ収斂点の入力となる経路について、収斂点の直前のFFを外部FFとする。また、収斂点のすべての入力が同じ順序深度ならば、その順序深度をその収斂点の順序深度とし、外部FFは選択しない。

(Step 4) (Step 2) で得られたすべての収斂点

表2 組合せ回路疑似変換後の回路特性

Table 2 Circuit characteristics after combinational circuit pseudo-transformation.

circuit	#I-FF	#E-FF	d	cpu(sec)
s382	6	15	1	0.45
s400	6	15	1	0.44
s444	6	15	1	0.54
s641	4	15	1	1.71
s713	4	15	1	2.21
s953	23	6	1	2.08
s1196	2	16	1	2.07
s1238	2	16	1	2.01
s1423	2	72	1	4.70
s5378	55	124	2	44.65
s9234.1	18	193	4	162.00
s9234	18	210	4	161.50
s13207.1	197	441	8	429.57
s13207	198	471	8	418.67

について、それらを始点とし、(Step 2) に戻る。

S を組合せ回路疑似変換した回路 S^T の回路特性を表2に示す。#I-FF、#E-FFの欄はそれぞれ内部FF数、外部FF数を表す。この#I-FF数が信号線(バッファ)に置き換わるFF数で、減少するFF数を表している。また、 d の欄は核回路の順序深度を表し、cpu(sec)の欄は組合せ回路疑似変換にかかった時間(秒)を表している。

はじめにテスト生成した結果を示す。表3がFAST-ESTの結果である。表中の#fault、#vec.、%f.cov.、cpu(sec)の欄はそれぞれ全故障数、テストベクトル数、故障検出率([検出された故障数]/[全故障数])、実行時間(秒)を示している。また、 S 、 S^T はそれぞれ S に対してテスト生成を行った結果、 S^T に対してテスト生成を行った結果であることを示している。ここで、 S^T におけるテストベクトル数は、 d -時刻保持系列変換後のベクトル数([S^T に対してテスト生成して得られたテストベクトル数] × [核回路の順序深度(d) + 1])である。s382, s400, s444, s713, s1423で故障検出率、テスト生成時間ともに S よりも S^T の方が良い結果が得られている。また、 S と S^T で故障検出率が同じだったs1196, s1238, s9234.1, s9234では S^T でテスト生成時間が短縮されていることが分かる。s13207.1, s13207では、 S^T で故障検出率は落ちているが、テスト生成時間が短縮されている。その他の回路では、大きな差は見られない。

次に、この実験結果について S と S^T で検出された故障の違いを解析する。検出された故障の違いを表4に示す。表中の# S_D 、# S_D^T の欄はそれぞれ S 、 S^T で検出された故障数を示す。# $S_D S_{UD}^T$ 、# $S_{UD} S_D^T$ の欄はそれぞれ S で検出され、かつ、 S^T で検出されなかつ

表3 FASTESTの結果

Table 3 Experimental results of FASTEST.

circuit	#fault	S			S ^T		
		#vec.	%f.cov.	cpu(sec)	#vec.	%f.cov.	cpu(sec)
s382	764	51	55.63	630	172	82.98	399
s400	800	51	54.63	680	140	81.00	388
s444	888	47	18.92	1680	174	79.73	410
s641	1278	146	87.32	48	318	87.25	53
s713	1426	137	83.10	234	282	83.24	233
s953	1906	12	7.92	20	14	7.82	14
s1196	2392	347	99.87	210	768	99.87	204
s1238	2476	368	96.65	555	668	99.65	515
s1423	2846	550	87.16	7298	954	88.97	5627
s5378	10590	856	78.57	154397	1605	77.75	168018
s9234.1	18468	51	10.02	236206	245	10.02	212121
s9234	18468	4	0.38	3170	20	0.38	3059
s13207.1	26358	117	11.64	226916	513	11.53	188256
s13207	26358	161	7.23	250668	702	6.84	153687

表4 FASTESTで検出された故障の違い

Table 4 The difference between faults detected in S and faults detected in S^T.

circuit	#S _D	#S _D ^T	#S _D S _{UD} ^T	#S _{UD} S _D ^T
s382	425	634	3	212
s400	437	648	3	214
s444	168	708	0	540
s641	1116	1115	1	0
s713	1185	1187	0	2
s953	151	149	3	1
s1196	2389	2389	0	0
s1238	2393	2393	0	0
s1423	2481	2532	14	65
s5378	8821	8234	182	95
s9234.1	1851	1851	0	0
s9234	70	70	0	0
s13207.1	3069	3041	40	12
s13207	1906	1803	103	0

た故障数, Sで検出されず, かつ, S^Tで検出された故障数を示している。実験結果で, S^Tで故障検出率, テスト生成時間ともに良くなっていたs382, s400, s444, s713, s1423では, #S_{UD}S_D^Tが#S_DS_{UD}^Tを上回っている。特に, s444, s713では, Sで検出された故障はすべてS^Tで検出され, さらにSで検出されなかった故障がS^Tで検出されている。また, SとS^Tで故障検出率が同じだったs1196, s1238, s9234.1, s9234では, #S_DS_{UD}^Tと#S_{UD}S_D^Tともに0であることから, 完全に同じ故障が検出されていることが分かる。その他は#S_{UD}S_D^Tが#S_DS_{UD}^Tを下回っている。

検出された故障の違いを見ることによって, テスタビリティの保存性を評価することができる。SとS^Tで完全に同じ故障を検出している回路に対しては, 組合せ回路疑似変換はテスタビリティを保存するといえる。その他の回路ではテスタビリティは保存していな

いが, S^Tにおいて, Sよりも多くの故障を検出している回路については, 組合せ回路疑似変換によって大幅に故障検出率が向上した。一方, S^Tにおいて, Sよりも検出した故障数が減少した回路もあるが, それらの故障検出率の低下はわずかである。

本手法では, 核回路の順序深度に比例してテストベクトル数が増加し, 回路によっては故障検出率がわずかに低下する場合がある。しかし, 大幅に故障検出率が向上する回路もあり, 多くの回路でテスト生成時間を短縮することができる。

7. あとがき

本論文では, 回路疑似変換によるテスト生成法を提案し, 組合せ回路疑似変換によってテスト生成されたテスト系列でテストを行うd-時刻保持テスト法を提案した。組合せ回路疑似変換とd-時刻保持変換によるテスト生成問題帰着性を明らかにした。さらに, ベンチマーク回路を用いた実験により組合せ回路疑似変換によるテスト生成法の有効性を評価した。本手法により, 多くの回路でテスト生成時間を短縮した。また, 一部の回路では大幅に故障検出率を向上した。今後の課題としては, 与えられた回路がどのような回路構造であれば, 本手法が有効であるかを明らかにすることなどがあげられる。

謝辞 本研究に関し, 多くの貴重な意見をいただいた本学の増澤利光助教授, 井上美智子助手はじめ情報論理学講座の諸氏に感謝します。

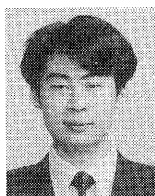
参考文献

- 1) Fujiwara, H.: *Logic Testing and Design for Testability*, The MIT Press (1985).

- 2) Gupta, R., Gupta, R. and Breuer, M.A.: The BALLAST Methodology for Structured Partial Scan Design, *IEEE Trans. Comput.*, Vol.39, No.4, pp.538-544 (1990).
- 3) 藤原秀雄: 組合せテスト生成複雑度でテスト生成可能な順序回路とその応用, 信学技法, FTS95-65, pp.17-24 (1995).
- 4) Balakrishnan, A. and Chakradhar, S.T.: Software Transformations for Sequential Test Generation, *IEEE 4th Asian Test Symposium*, pp.266-272 (1995).
- 5) Kelsey, T.P., Saluja, K.K. and Lee, S.Y.: An Efficient Algorithm for Sequential Circuit Test Generation, *IEEE Trans. Comput.*, Vol.42, No.11, pp.1361-1371 (1993).
- 6) Chakradhar, S.T., Balakrishnan, A. and Agrawal, V.D.: An Exact Algorithm for Selecting Partial Scan Flip-Flops, *31th ACM/IEEE Design Automation Conference*, pp.81-86 (1994).

(平成8年10月18日受付)

(平成9年3月7日採録)



大竹 哲史

昭和46年生。平成7年電気通信大学電気通信学部卒業。平成9年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了。現在、同大学博士後期課程に在学中。テスト生成, テスト容易化高位合成に関する研究に従事。電子情報通信学会会員。



井上 智生 (正会員)

昭和40年生。昭和63年明治大学工学部卒業。平成2年同大学院工学研究科博士前期課程修了。同年松下電器産業(株)に入社。明治大学大学院博士後期課程を経て、現在奈良先端科学技術大学院大学情報科学研究科助手。松下電器産業(株)においてマイクロプロセッサの研究開発に従事。明治大学、奈良先端科学技術大学院大学において、テスト生成、並列処理、テスト容易化設計に関する研究に従事。博士(工学)。IEEE、電子情報通信学会各会員。



藤原 秀雄 (正会員)

昭和21年生。昭和44年大阪大学工学部卒業。昭和49年同大学院工学研究科博士後期課程修了。同大工学部助手, 明治大学工学部助教授, 理工学部教授を経て、現在、奈良先端科学技術大学院大学情報科学研究科教授。昭和56年ウォータールー大学客員助教授。昭和59年マッギル大学客員準教授。論理設計, 高信頼設計, 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。工学博士。著書“Logic Testing and Design for Testability”(The MIT Press)など。大川出版賞。電子情報通信学会会員。IEEE Fellow。