

パーシャルスキャンによる  
テスト容易化設計のための  
順序回路のテスト生成アルゴリズム

TEST GENERATION ALGORITHM FOR SEQUENTIAL  
CIRCUITS WITH PARTIAL SCAN STRUCTURE.

細川 利典      本原 章      太田 光保      秋濃 俊郎  
Toshinori Hosokawa   Akira Motohara   Mitsuyasu ohta   Toshiro Akino

松下電器産業株式会社      半導体研究センター  
Matsushita Electric Industrial Co.,Ltd.      Semiconductor Research Center

あらまし    スキャン設計 [1, 2] は、VLSI の設計に不可欠な技術であるが、付加回路の増加等の問題点があり適用できない場合も多い。付加回路を減らすために一部のフリップフロップ (以下 FF) だけをスキャンする「パーシャルスキャン法」を採用すると少ない付加回路で高い検出率が得られるという報告がなされている。[3, 4] パーシャルスキャンを用いるテストの課題は、(1) スキャンする FF の数をできるだけ少なくするために順序回路のテスト生成でできるだけ多くの故障を検出する (2) できるだけ少ないスキャンで検出率を高くするために、スキャンする FF をうまく選択する、という 2 点である。本報告では前者の問題を検討し、パーシャルスキャンを用いる際に効率よくテスト生成を行うためのいくつかの手法を紹介する。

Abstract    In this paper we discuss test generation algorithm for sequential circuit with partial scan. Indeed scan path [1,2] is an essential technique to VLSI design. However, due to the disadvantages such as excessive chip area, in many cases we have to give up to use this approach. Many efforts reduce area overhead are made. Especially, partial scan approach is known as a good compromise achieving high fault coverage with small amount of extra circuitry. [3,4] To fully enjoy these benefits, we have to consider the following subjects: (1) how to detect as many faults as possible by sequential test generation without scan, and (2) how to choose those flip-flops which are going to be replaced with scannable ones. We concentrate on the first subject and introduce several techniques to generate tests for sequential circuits with partial scan structure.

## 1. はじめに

順序回路のテスト生成 (ATG) は、故障伝搬、状態正当化、状態初期化により実現される。従来の順序回路の ATG アルゴリズムは、拡張 D アルゴリズム [5] に代表される反復組合せ回路上で組合せ回路用の ATG アルゴリズムを適用する方法、故障伝搬経路をあらかじめ求めておき、時間軸と逆向きに状態正当化と状態初

期化を行い、テスト系列を生成する方法 [6]、故障伝搬経路選択、状態正当化、状態初期化のすべてを時間軸と逆向きに行う方法 [7] などに分類される。これまでに報告されているパーシャルスキャンは、これらの順序回路の ATG または組合せ回路用の ATG [8] を用いてテスト生成、及びテスト生成上クリティカルなフリップフロップの同定を行い、後処理でできるだけ検出率を高くするように、フリップフロ

プを選択する組合せ問題を解くというものがほとんどであった。

本報告では、テスト生成の途中あるいは終了後に、一部のフリップフロップをスキャンフリップフロップに置き換えるテスト容易化設計のためのテスト生成アルゴリズムを提案する。特徴として、(1) 順序回路のテスト生成で失敗した場合、スキャンにより検出率の向上が図れるよう、各故障について組合せ回路部分のテスト生成を行い、その後故障伝搬経路の決定と状態正当化を行う。(2) 多重後方追跡テーブルを用いる、(3) 疑似外部入力に仮定した故障のテスト系列を利用する、(4) スキャン可能となったフリップフロップの情報を利用する、(5) クロック機構に関する情報を利用する、順序回路のテスト生成アルゴリズムを紹介する。2章では、本報告書で用いる回路モデルと表現についての説明を行う。3章では、テスト生成アルゴリズムを説明する。4章では、今後の課題問題点について述べる。

## 2. 定義

### 2-1. 回路モデル

図1に本報告書で紹介するアルゴリズムが対象とする回路モデルを示す。回路モデルは以下の要素から構成される。

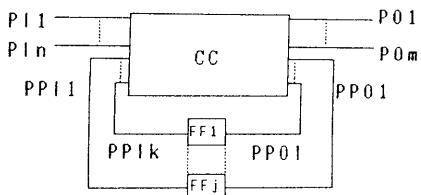


図1. 回路モデル

- (1) CC: 組合せ回路部分
- (2) FF $i$  ( $i = 1, \dots, j$ ): フリップフロップ
- (3) PI $i$  ( $i = 1, \dots, n$ ): 外部入力
- (4) PO $i$  ( $i = 1, \dots, m$ ): 外部出力

- (5) PPI $i$  ( $i = 1, \dots, k$ ): 疑似外部入力 (FFの出力)
- (6) PPO $i$  ( $i = 1, \dots, l$ ): 疑似外部出力 (FFの入力)

なおPPOは、クロック入力、セット・リセット入力も含む。

### 2-2. 信号値セット

信号値として次の5値を用いる。

- (1) 0: 信号値0
- (2) 1: 信号値1
- (3) X: 信号値未定またはドントケア
- (4) P: 正のパルス (立ち上がりエッジ信号として用いる。)
- (5) N: 負のパルス (立ち下がりエッジ信号として用いる。)

### 2-3. 故障信号

故障信号を表すのに正常回路の値と故障回路の値の対を用いる。例えば正常時に0、故障時に1となる故障信号は0/1と表す。

### 2-4. 時刻

ここで用いる時刻の概念は、状態変化の時間的な前後関係 (半順序関係となる) を数値の大小で表したものである。したがってテスト生成での同時刻のイベントはどちらが先に発生してもよいものとなる。時刻を表すのに、整数を用いる。最初に回路中の故障信号線に着目した時刻を0とし、正及び負の整数を用いて時刻を表す。

信号値または故障信号と時刻を対にしてイベントを表す。例えば「時刻4で0」は、0(4)と表す。また繰り返しイベントの時刻を、周期と周期内での相対位置の対で表現する。例えば、「 $\dots, -2, 3, 8, \dots$ の時刻(周期5、相対位置3)に立ち上がる。」イベントはP(3:5)と表す。

### 3. テスト生成アルゴリズム

#### 3-1. 基本的な戦略

テスト生成においてバックトラック数を減らすための基本的な戦略は [6, 7] に述べられているように「一意に決定できることは、できるだけ早く、できるだけ多く決める」ということである。しかしながら順序回路については、故障伝搬経路の選択など一意に決定できないものも多く、またバックトラック以外の処理にも状態初期化のために繰り返し後方追跡を行うなど多くの計算時間が必要となる。本報告では、種々の処理のうち過去の「成功事例」を適用できる場合にこれを適用し、さらに積極的に「成功事例」を適用できるように値の割り当てを行うことにより、高速化を図る順序回路のテスト生成方法を提案する。以下に示す「知識」あるいは「成功事例」を繰り返し適用する。

(1) FFR (樹枝状部分回路) 毎に前処理で多重後方追跡テーブルを作成し、多重後方追跡の時にこれを参照する。

(2) PPI 毎に、その PPI に故障信号を仮定して、その故障信号を外部出力へ伝搬させるためのテスト系列を求めておき、故障伝搬経路を決定する際にこの情報を用いる。

(3) PPI がテスト容易化設計によりスキャン可能となった FF に対応する場合、その情報を故障伝搬経路を決定する際に用いる。

(4) クロック機構に関する情報を前処理で求めておき、この情報を故障伝搬経路決定や状態正当化で用いる。

#### 3-2. 全体の流れ

以下に全体の処理の流れの概要を示す。

(ステップ1)  
可検査性コストの計算、多重後方追跡テーブル作成、PPI テスト生成、クロック動作の認識

(ステップ2)

未処理故障がなくなるまでステップ3からステップ7を実行

(ステップ3)

テスト生成の対象となる故障をひとつ選出

(ステップ4)

組合せ回路部分のテスト生成を行ない故障信号が外部出力に達している場合ステップ6へ

(ステップ5)

PPI テストへの帰着、または可検査性尺度にしたがってヒューリスティックな故障伝搬経路の選択を行なう。

(ステップ6)

時刻の大きい未正当化信号線から順に正当化

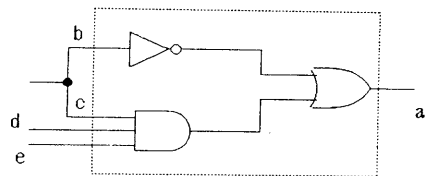
(ステップ7)

故障シミュレーション

#### 3-3. 多重後方追跡テーブル

##### 3-3-1. 多重後方追跡テーブル作成

各 FFR (樹枝状部分回路) について、FFR の出力のファンアウトシステムに目標 0 (1) を設定し、FFR の入力側に向かって多重後方追跡を行なう。その結果得られた FFR の入力の目標群を記憶しておく。図2に、多重後方追跡テーブルの作成例を示す。



(a-0) → (b-1, d-0)

(a-1) → (b-0)

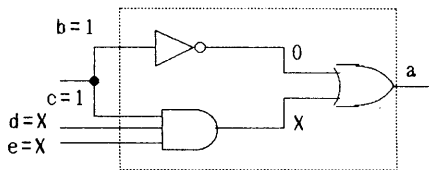
図2. 多重後方追跡テーブル作成

##### 3-3-2. 多重後方追跡

多重後方追跡をファンアウトシステムから行なう際、回路の接続情報をアクセスする前に、多重後方追跡テーブルを参照する。テーブルに書かれている FFR の入力の目標群と、FFR の

入力に現在割り当てられている値が矛盾する場合には、通常の多重後方追跡を行う。値が割り当てていないか、目標群と一致する値が割り当てられている場合には、値が割り当てられていないものをFFRの入力の目標群とする。図3に例を示す。図のようにFFRの入力に  $b = 1$ 、 $c = 1$  が割り当てられているとき、FFRの出力  $a$  を0にする目標から多重後方追跡を行うとき、多重後方追跡テーブルの内容と割り当てられた状態とは矛盾しないので、値が割り当てられていない信号線  $d$  に0という目標が得られる。 $a$  を1にするという目標の時は、多重後方追跡テーブルの内容と割り当てられた状態とが矛盾するので通常の多重後方追跡を行ない、 $(d - 1, e - 1)$  という目標群を得る。

この多重後方追跡テーブルを用いる方法は、値が割り当てられていない各故障についての初期段階で効果的である。



(a-0) → (d-0)

(a-1) → 矛盾

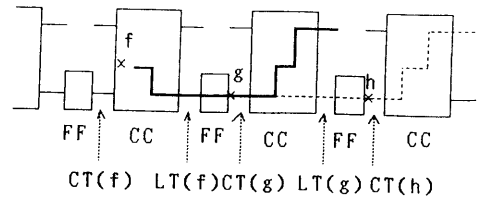
通常の多重後方追跡を行ない

(d-1, e-1)

図3. 多重後方追跡

### 3-4. PPIテストへの帰着

PPI (疑似外部入力) まで伝搬した故障信号は、ある条件が成立するとき前処理で作成したPPIに仮定した故障を検出するテスト系列を用いて外部出力まで伝搬させることができる。図4を用いてその方法の概要を説明する。組合せ回路部分について求めた故障  $f$  のテスト  $CT(f)$  と、 $CT(f)$  によりPPO (疑似外部出力) に達した故障信号をラッチするためのPPOのパターン  $LT(f)$  (一般的には未正当



—— 故障伝搬経路

----- 再帰的に帰着する場合

図4. PPIテストに帰着させるテスト生成方法

化信号となる) とで故障信号をPPIまで伝搬させた後、そのPPIに仮定した故障  $g$  のPPIテスト  $CT(g)$  への帰着が可能か否かを判定し、PPIテストに帰着可能な場合は、 $CT(f)$ 、 $LT(f)$ 、 $CT(g)$  を生成する際に、発生した未正当化信号線の正当化の処理に移る。PPIテストに帰着不可能な場合は、ヒューリスティックな伝搬経路選択を行なう。ただし、PPIの故障のテスト生成は1つ前のタイムフレームの故障の影響を出力側へ伝搬するために行っているため、PPIの故障は0縮退故障でも、1縮退故障でも構わない。

### 3-4-1. PPIテストへの帰着条件

以下の条件1から3がすべて成立するとき故障  $f$  はPPIテスト  $CT(g)$  への帰着が可能であるとい

$$f \in PPI T(CT(g))$$

と表す。ただし  $g$  はPPI  $i$  に仮定できる故障とする。

(条件1) 組合せ回路部分で求めたテスト  $CT(g)$  により故障  $g$  の故障信号が外部出力に到達する。条件1の概念図を図5に示す。

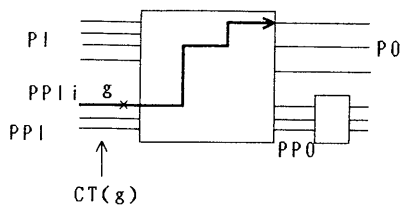


図5. 条件1の概念図

(条件2) 故障  $f$  の故障信号が、組合せ回路部分で求めたテスト  $CT(f)$ 、及び故障信号を  $FF$  の出力側へ伝搬させるために  $PP0$  に割り当てるパターン  $LT(f)$  により  $PP1i$  に到達している。条件2の概念図を図6に示す。

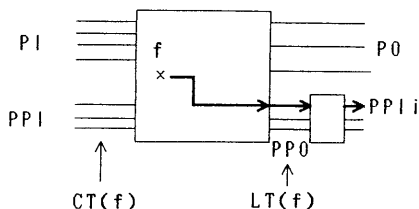


図6. 条件2の概念図

(条件3)  $CT(g)$  で値を割り当てられている  $PP1i$  以外の  $PP1$  の集合を  $SPP1$  とし、 $CT(f)$ 、及び  $LT(f)$  を印加した結果故障信号が存在する  $PP1$  の集合を  $DPP1$  とするとき次式が成り立つ。

$$SPP1 \cap DPP1 = \emptyset$$

条件3の概念図を図7に示す。図7では、故障  $f$  の影響は  $CT(f)$  及び  $LT(f)$  により2本の  $PP1$  まで伝搬している。故障信号が伝搬している2本の  $PP1$  の故障 (図7では  $g$ 、 $h$ ) の中から1つ故障を選び出し (この場合  $g$  を選択)、選んだ  $PP1$  の故障のテスト ( $CT(g)$ ) が故障信号が伝搬している他の  $PP1$  (この場合  $h$  を仮定した  $PP1$  が存在する) に関係なく求められるとき、条件3が成立する。

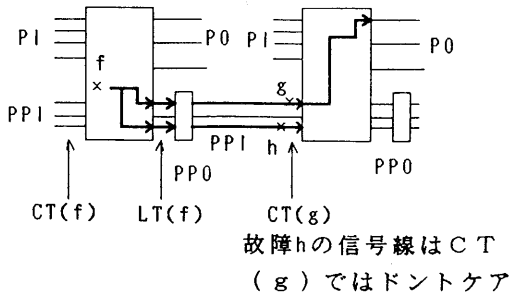


図7. 条件3の概念図

### 3-4-2. $PP1$ テストへの再帰的帰着条件

以下の条件4から6がすべて成立するとき故障  $f$  は  $PP1$  テスト ( $CT(g)$ 、 $LT(g)$ ) への再帰的帰着が可能であるとい

$$f \in PP1TR(CT(g), LT(g))$$

と表す。

(条件4) 組合せ回路部分で求めたテスト  $CT(g)$  により故障  $g$  の故障信号が疑似外部出力に到達する。条件4の概念図を図8に示す。

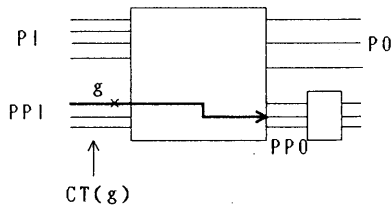


図8. 条件4の概念図

(条件5) 前節の条件2、及び条件3が成立する。

(条件6)  $g \in PP1T(CT(h))$  となる  $PP1$  の故障  $h$  が仮定でき、組合せ回路部分でこの故障について求めたテスト  $CT(h)$  が存在する。または  $g \in PP1TR(CT(h), LT(h))$  となる  $PP1$  の故障  $h$  が仮定でき、

組合せ回路部分でこの故障について求めたテストCT(h)、CT(h)で発生した故障信号をFFの出力側へラッチするためにPPOに割り当てるパターンLT(h)が存在する。条件6の概念図を図9に示す。

このときも、ヒューリスティックな故障伝搬の選択の処理は不要であり(図4の例では波線で表された経路が選択される)、直ちに状態正常化の処理に移ることができる。

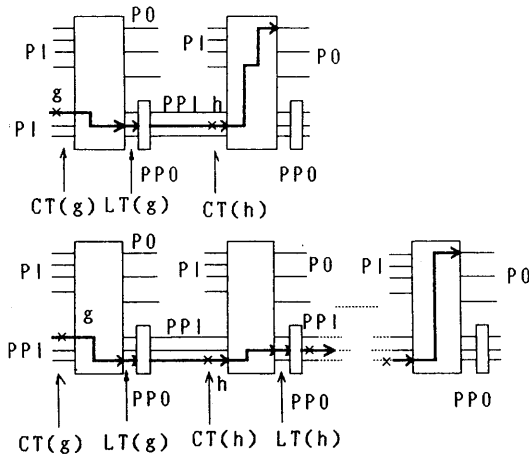


図9. 条件6の概念図

### 3-4-3. PPIテストの情報

3-4-1. 及び3-4-2. で述べた、条件1・条件4・条件6が成立するか否かで、PPIを3種類に分類する。

- (1) クラス0: 条件1、条件4のいずれも成立しない
- (2) クラス1: 条件1は成立しないが、条件4及び条件6が成立する。
- (3) クラス2: 条件1が成立する。

各PPIには、PPIテスト生成の結果により次のような情報が付加される。ただしgは、PPIテスト生成で扱った故障であり、hは

が帰着または再帰的に帰着されるPPIの故障である。

- (1) PPIのクラス
- (2) クラス2のとき、テストCT(g)
- (3) クラス1のとき、テストCT(g)及びLT(g)
- (4) クラス1またはクラス2のとき、テストCT(g)で値が割り当てられているPPI
- (5) クラス1のとき、hに対応するPPI

### 3-5. スキャンフリップフロップの情報

テスト生成を行なう過程で、テスト生成上クリティカルであると判定されたフリップフロップをスキャンフリップフロップに置き換えるというテスト容易化設計を考える。このとき、1度スキャンされたフリップフロップを最大限有効に使うために、スキャンフリップフロップに関して以下の処理を行なう。

(1) スキャンフリップフロップに対応するPPI、PPOの可検査性コストを変更し(それぞれPI、P0として扱う)、これにともなって変化する回路内部の可検査性コストを再計算する。

(2) スキャンフリップフロップに対応するPPIテストに帰着しているクラス1のPPIをクラス2とする。

(3) 可検査性コストが変更した信号線を含むFFRについて多重後方追跡テーブルを再度作成する。

### 3-6. クロック機構の情報

PPIテストへの帰着などを行っても、複雑なクロック機構を持つ回路のフリップフロップのラッチ動作のためのテスト生成は困難な問題として残る。この節では、前処理でクロック動作を調べ、その情報をテスト生成で用いる手法について述べる。

### 3-6-1. クロックの独立性

ある外部入力  $C_i$  に対して以下のすべての条件を満たすフリップフロップ  $FF_j$  ( $j=1, \dots, n_i$ ) を要素とする集合を外部入力  $C_i$  に属するフリップフロップの集合と定義し、 $SFF(C_i)$  と書くことにする。

(1)  $FF_j$  のクロック入力から外部入力側をたどるとき、 $TFF$ 、インバータ以外の素子は現れない

(2) ある時刻に、関係するすべての  $TFF$  の値を  $X$  以外 ( $0$  または  $1$ ) に制御できる

(3)  $FF_j$  のクロック入力に到達可能な外部入力は  $C_i$  のみである。

一般にはどの外部入力にも属さないフリップフロップが存在するが、上記の条件を満たすフリップフロップに対しては、テスト生成上非常に有用な情報を得ることができる。この技法はデータバスと制御回路が分離している回路で、外部入力から直接クロックを共給している場合や、 $TFF$  を用いて周波数変換を行っているような回路に適している。

異なる外部入力に属するフリップフロップ同士は独立にラッチすることが可能であるが、同じ外部入力に属するフリップフロップの間には、動作上の制約が存在する。

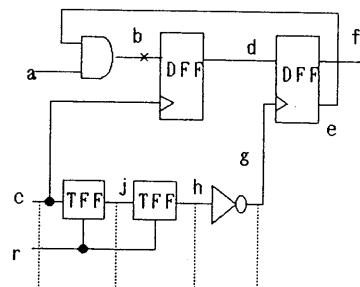
### 3-6-2. クロックの従属性

同じ外部入力に属するフリップフロップの従属性を前処理で調べる方法について述べる。外部入力  $C_i$  に属するフリップフロップ  $FF_j$  がデータをラッチするタイミングは、 $C_i$  から  $FF_j$  のクロック入力へいたる経路を調べて、出会う素子ごとに、表1のクロック変換表により変換を繰り返すことにより求めることができる。 $FF_j$  が立ち上がり (立ち下がり) エッジまたはレベル  $1$  ( $0$ ) でラッチする素子の場合、 $P(0:2)$  ( $N(1:2)$ ) から始める。なお、表1の  $TFF(P, 0)$  は立ち上がりで出力反転、初期化は  $0$  リセットを表す。%はモジュロ演算を表す。

素子	変換前	変換後
インバータ	$P(T:L)$ $N(T:L)$	$P((T+L/2)\%L:L)$ $N((T+L/2)\%L:L)$
$TFF(P, 0)$	$P(T:L)$ $N(T:L)$	$P(T:2L)$ $N(T+L:2L)$
$TFF(P, 1)$	$P(T:L)$ $N(T:L)$	$P(T+L:2L)$ $N((T+3L/2)\%2L:2L)$
$TFF(n, 0)$	$P(T:L)$ $N(T:L)$	$P(T+L/2:2L)$ $N(T+L:2L)$
$TFF(n, 1)$	$P(T:L)$ $N(T:L)$	$P((T+3L/2)\%2L:2L)$ $N(T:2L)$

表1. ラッチタイミング変換表

このようにして求めたクロックの従属性を利用して、ラッチ動作のテスト生成やヒューリスティックな故障伝搬経路の選択を行なう。図10の回路を例にテスト生成の流れを示す。この例では、故障伝搬、状態正当化において、クロックに関する情報が効果的に用いられていることがわかる。図10中の  $TFF$  は  $TFF(P, 0)$  とする。



$P(0:2)$   $P(0:4)$   $P(0:8)$   $P(4:8)$

図10. クロック機構の情報の利用

(前処理:  $PP$  I テスト生成)

$d$ : クラス1,  $CT(d^{\wedge}) = \emptyset$ ,  $LT(d^{\wedge}) = \{g=P\}$ ,  $f$  に帰着可能  
ただし、 $d^{\wedge}$  は信号線  $d$  の故障を表し、その故

障は0縮退故障でも1縮退故障でもよい。

$f$  : クラス2,  $CT(f^{\wedge}) = \phi$

(前処理: クロック動作認識)

$c = P(0:2)$ ,  $g = P(4:8)$  (初期化は  $8n-1$ :  $n$  は整数)

(組合せ回路のテスト生成)

$CT(b-0) = \{a=1(0), *e=1(0)\}$  (\*は未正当化信号線を表す。)

ただし、 $b-0$  は信号線  $b$  の0縮退故障を表す。

(PPIテストへの帰着)

$LT(b-0) = \{c=p(2)\}$ ,  $CT(d^{\wedge}) = \phi$ ,  $LT(d^{\wedge}) = \{g=P(4)\}$ ,  
 $CT(f^{\wedge}) = \phi$

(状態正当化)

$g = p(-4)$ ,  $*d = 0(-5)$ ,  $c = P(-6)$ ,  
 $a = 0(-7)$ ,  $r = p(-9)$

#### 4. おわりに

順序回路のテスト生成に関するいくつかの手法を提案した。提案した手法のうち、多重後方追跡テーブルは各故障ごとの状態の正当化や状態の初期化において樹枝状回路部分で用いる。またPPIテストは故障伝搬経路の選択のために用いる。クロック機構の情報は複雑なクロック機構を持つ回路のFFのラッチ動作のためのテスト生成に用いる。

現在提案したアルゴリズムを採用したテスト生成システムのプロトタイプを開発中であり、今後実データによる各手法の有効性の評価と問題点の検討を行なう予定である。また、パーシャルスキャン自動挿入のアルゴリズムについても現在開発中であり、テスト容易化設計を含めてテスト自動化の問題点を検討する予定である。

最後に日頃御指導、御助言を頂く明治大学藤

原秀雄助教に感謝の意を表す。

#### 参考文献

[1] S. Funatsu, N. Wakatsuki, and T. Arima, "Test Generation Systems in Japan," Proc. 12th Design Automation Conf., pp. 114-122, June 1975.

[2] E. B. Eichelberger and T. W. Williams, "A Logic Design Structure for LSI Testability," Proc. 14th Design Automation Conf., pp. 462-468, June 1977.

[3] V. D. Agrawal, K. Cheng, D. D. Johnson, and T. Lin, "A Complete Solution to the Partial Scan Problem," Proc. Int. Test Conf., pp. 44-51, Sep. 1987.

[4] H. T. Ma, S. Devadas, A. R. Newton, and A. Sangiovanni-Vincentelli, "An Incomplete Scan Design Approach to Test Generation for Sequential Machines," Proc. Int. Test Conf., pp. 730-734, Oct. 1988.

[5] G. R. Putzolu and J. P. Roth, "A Heuristic Algorithm for the Testing of Asynchronous Circuits," IEEE Trans. on Computers, Vol. C-20, No. 6, pp. 639-646, June 1971.

[6] D. R. Marlett, "An Effective Test Generation System for Sequential Circuits," Proc. 23rd Design Automation Conf., pp. 250-256, June 1986.

[7] M. H. Schulz and E. Auth, "ESSENTIAL: An Effective Self-Learning Test Pattern Generation Algorithm for Sequential Circuits," Proc. Int. Test Conf., pp. 28-37, Aug. 1989.

[8] 奥村憲三、野田浩明、松本比呂志、「部分スキャン方式の自動テストパターン生成」第21回FTC研究会資料、1987年7月