

## スキャンテストにおけるキャプチャ時の低消費電力化に 効果的なテスト集合変更について

鈴木 達也<sup>†</sup> 温 暁青<sup>†</sup> 梶原 誠司<sup>†</sup> 宮瀬 紘平<sup>‡</sup> 皆本 義弘<sup>‡</sup>

<sup>†</sup>九州工業大学 〒820-8502 福岡県飯塚市川津 680-4

<sup>‡</sup>科学技術振興機構 〒814-0001 福岡市早良区百道浜 3-8-34

E-mail: <sup>†</sup> suzuki@aries30.cse.kyutech.ac.jp, <sup>†</sup> {kajihara, wen}@cse.kyutech.ac.jp,  
<sup>‡</sup> {miyase, minamoto}@fukuoka.jst-plaza.jp

あらまし スキャンテストのキャプチャ時においてフリップフロップでの論理値の遷移が多く起こると過度のIRドロップを引き起こす可能性があり、それは深刻な歩留まりの低下につながる。本論文では、テスト集合変更に基づいた手法を用いる。提案手法の特徴は(1)与えられたテストパターンに対し、故障検出率を低下させずに、選択したビットをドントケア(X)に変える新しい制約X判定手法と、(2)キャプチャ時の論理値の遷移数を削減するLCP(Low Capture Power)のため、Xへの論理値割り当て手法である。提案手法はチップ面積、タイミング、テスト集合サイズ、および故障検出率に影響なくキャプチャ時の消費電力を削減するため、テスト生成フローへ容易に組み込むことが出来る。ベンチマーク回路での実験結果は提案手法の効果を示している。

キーワード キャプチャ低消費電力、ドントケア判定、ドントケアへの論理値割り当て。

## On Low Capture Power Test Generation for Scan Testing

Tatsuya SUZUKI<sup>†</sup> Xiaoqing WEN<sup>†</sup> Seiji KAJIHARA<sup>†</sup>

Kohei MIYASE<sup>‡</sup> and Yoshihiro MINAMOTO<sup>‡</sup>

<sup>†</sup> Kyushu Institute of Technology 680-4 Kawazu, Iizuka 820-8502 Japan

<sup>‡</sup> Japan Science and Technology Agency 3-8-34 Momochihama Sawara-ku, Fukuoka 814-0001 Japan

E-mail: <sup>†</sup> suzuki@aries30.cse.kyutech.ac.jp, <sup>†</sup> {kajihara, wen}@cse.kyutech.ac.jp,  
<sup>‡</sup> {miyase, minamoto}@fukuoka.jst-plaza.jp

**Abstract:** High switching activity occurs when the response to a test vector is captured by flip-flops during scan testing. This may cause excessive IR drop, resulting in significant test-induced yield loss. The paper addresses this problem with a novel method based on test set modification, featuring (1) a new constrained X-identification procedure that turns a properly selected set of bits in a fully-specified test set into X-bits without fault coverage loss and (2) a new LCP (low capture power) X-filling procedure that optimally assigns 0's and 1's to the X-bits to reduce the switching activity of the resulting test set in capture mode. This method can be readily incorporated in any test generation flow to efficiently reduce capture power dissipation without any impact on area, timing, test set size, and fault coverage. Experimental results on benchmark circuits have shown its effectiveness.

**Keywords:** Low Capture Power, X-Identification, X-filling.

### 1. はじめに

LSI のテストにおいてフルスキャン設計は最も広く採用されているテスト容易化設計である[1]。フルスキャン順序回路はフリップフロップにシフトレジスタ機構を付加し、シフトとキャプチャの2つのモードで動作する。シフトモードでは順序回路の組合せ回路部に対し、シフトインによりテストベクトルをロードし、シフトアウトによりテスト応答を観測するモードである。キャプチャモードではスキャンフリップフロップは通常動作し、シフトモードで前のテストベクトルによる組合せ回路部の出力応答をシフトアウトした後に組合せ回路部の新たな出力応答をスキャンフリップフ

ロップにアップデートする。その結果、フルスキャン順序回路のテスト生成に組合せ回路用のATPGを用いることができる[1]。

スキャンテストには多くの利点があるにも関わらずテストデータ量、テスト実行時間、テスト消費電力が増加するという3つの課題がある。テストデータ量、テスト実行時間増加の問題はディープサブミクロン時代では必然的に起こる問題である。これらの問題はテスト圧縮、多重キャプチャクロックによって解決することが出来る[2]。

CMOS 回路の消費電力は漏洩電流による静的消費電力と論理値の遷移による動的消費電力からなる。消費

電力全体では動的消費電力が大半を占め、スキャンテストではシフト時とキャプチャ時の両方で電力を消費する。シフト時ではテストベクトルは1ビットごとにスキャン回路のスキャンチェーンにシフトされ、この時シフト消費電力が起きる。キャプチャ時では、組合せ回路部の出力応答はフリップフロップにアップロードされ、この時フリップフロップが保持している論理値とアップロードされる出力応答の論理値が異なる場合に状態遷移が起こり、キャプチャ消費電力が起きる。通常動作時では複数の機能ブロックは相互間で機能的に依存しているため同時に複数のブロックは動作しない。しかし、テスト生成やテストクロックは相互間の機能的な依存性は無視するため、一般にテスト時の消費電力は通常動作時の消費電力よりも大きい[3]。

一般にテスト時の消費電力増大による影響として2つ挙げられる。それは蓄積的な消費電力と瞬間的な消費電力の影響である[4]。蓄積的な影響は過度の熱として現れる。過度の熱によって回路を破損させる事や、電解拡散を加速し信頼性を低下させる恐れがある[5]。瞬間的な影響は1つのクロックパルスにおいて同時に多くのフリップフロップで論理値の遷移が起こればIRドロップを引き起こす可能性がある。そのIRドロップによってフリップフロップの誤動作、回路の遅延の増加などが起こる。その結果、テスト応答が誤ることがあり、歩留まり低下の原因となる[6]。

蓄積的・瞬間的な消費電力の影響はコスト、歩留まりに影響を及ぼし、そのような消費電力の削減は重要な課題になっている。そこでテスト時の消費電力を削減するためにシフト消費電力とキャプチャ消費電力の削減の必要性がある。シフト時の消費電力を削減する手法は多く提案されており、テストスケジューリング、テストベクトル操作、回路操作、スキャンチェーン操作の4つの主なアプローチに基づいている。

シフト時の消費電力削減と比べ、キャプチャ時の消費電力削減は(1)通常、全てのクロックドメインで同時にキャプチャするため論理値の遷移が劇的に増加してしまう。(2)キャプチャのクロックタイミングはシフトのクロックタイミングよりも厳密なので周波数の周期と位相のキャプチャクロックのタイミング操作が難しい。(3)フリップフロップでキャプチャ遷移を減らすために回路操作を行うことは危険である。以上の特性があるため、キャプチャ時の消費電力削減は困難である。よって、キャプチャ消費電力削減にはテストベクトル操作を基にしたアプローチが望ましい。

文献[20],[21]の手法はテスト集合変換に基づいた手法である。これらはまず故障検出率を低下させずに最初に与えられるテスト集合に対してドントケアビット(X)を判定する。次に、キャプチャ時の最大消費電力が減るように $X \rightarrow 0$ または $1$ を割り当てる。

文献[21]で説明されている手法は正当化操作、含意操作に基づくX判定手法を用いる。この手法は、高速な処理が出来るだけでなく、より多くのXビットを判定し、効果的なキャプチャ消費電力削減の可能性がある。しかし、テスト集合中に判定されたXビットがどのように分布するかはXを判定するまでわからないため、

消費電力削減に不必要なXビットが存在する可能性がある。また、X割当てではテストベクトルにおいてビット依存を考慮せずに行われる。そのため電力削減効果に影響を与えてしまう。

本論文では以上の問題点を踏まえ、テスト集合変更手法を用いてキャプチャ時の消費電力削減を行う。テスト集合中に効果的なX分布を得るために、正当化/含意操作ベースのX判定に機能追加したものを用い、判定したXにはビット依存を考慮したXシンボルシミュレーションを用いて論理値を割り当てる。ただしこの論文ではシングルキャプチャクロックを前提としている[22][23]。

論文構成は以下の通りである。2節では準備として研究背景を説明し、3節ではキャプチャ時の消費電力削減のための手法を説明する。4節で実験により提案手法の有効性を示す。最後に5節でまとめを行う。

## 2. 準備

### 2.1. キャプチャ消費電力の削減

図1にフルスキャン回路を示す。 $v$ はテストベクトルである。 $v$ のPIsとPPIsのビットは $\langle v: PI \rangle$ 、 $\langle v: PPI \rangle$ で示す。組合せ回路部分は論理関数 $f$ と仮定し、その出力応答は $f(v)$ とする。 $f(v)$ のPOとPPOは $\langle f(v): PO \rangle$ 、 $\langle f(v): PPO \rangle$ によって示される。

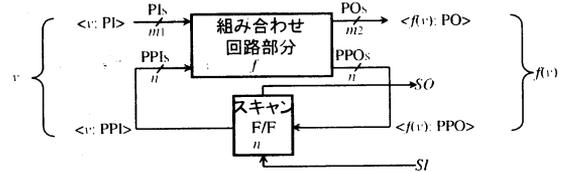


図1 一般的なフルスキャン回路

図2で示すように $\langle v: PPI \rangle$ と $\langle f(v): PPO \rangle$ が異なると、キャプチャモードでスキャンフリップフロップ (FF) は論理値の遷移が起こる。テストベクトルとキャプチャ時の遷移は密接に関連している[20-23]。よってキャプチャ時の遷移数を抑えることはキャプチャ電力削減につながる。

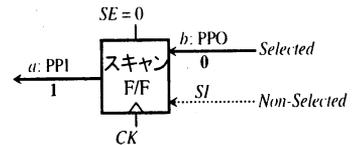


図2 キャプチャ時に起こるスキャンFFでの遷移

キャプチャ時の遷移数を削減するため動的または静的にテスト集合を操作する手法がある。動的な手法はキャプチャ遷移を減らすためにテスト生成時に動的圧縮したテストキューブに0または1を最適に割り当てる処理を行う。このアプローチはより良い削減効果を得ることが出来るが、テスト集合サイズが大きくなるという問題がある。よってテスト集合サイズが変化しない、一度生成されたテスト集合を変更するという静的な手法が有効である。

## 2.2. X判定

X判定にはシミュレーションベースと正当化/含意操作ベースの2つのアプローチがある。文献[21]のX判定はシミュレーションベースの手法である。この手法の処理時間は故障数、テストベクトル数、テストベクトル中のビット数におおよそ比例する。また、Xビットの数はテストベクトル順序、テストベクトル中の全てのビットの処理を行なう順序に依存している。

文献[24]に説明されているX判定はXIDと呼ばれる正当化/含意操作ベースの手法である。XIDの処理時間はおおよそ故障の数と、テストベクトル数に比例し、一般的にシミュレーションベースのものよりも短い[21]。また、この処理はより多くのXビットを特定できる。しかし、Xの判定の際に何も制約が無いのでキャプチャ遷移を効果的に減らすことが出来ないようなXの分布になってしまう場合がある。例えばキャプチャ遷移が多いテストベクトルにXビットが少ない場合などは遷移をあまり減らすことができない。XIDのもう一つの問題は遷移が起こらないビットに対してX判定をしてしまう可能性があり、そのようなビットがXと判定されても消費電力削減の効果は無く、遷移の起こっているビットに対してXビットが少なくなる場合がある。

## 2.3. X割り当て

X割り当ては2つのアプローチがある。論理値を単純にXビットに割り当てるだけのアプローチ[21]とLCP(Low Capture Power)-X割り当て[22]があり、後者はテストベクトル中のXビットに論理値を割り当てるだけでなく、そのテスト応答に現れるXビットにも論理値を割り当て、正当化操作を行う。LCP-X割り当ての処理は、テストベクトル中のXビットにのみ論理値を割り当てる処理(割り当て)、テスト応答に現れるXビットに論理値を割り当て、正当化操作を行う処理(正当化)、そして両方を行う処理(割り当て/正当化)の順となる。これは処理の複雑さが順に増加するためである。

文献[21]、[22]の手法はテストベクトルによるビットの依存を考慮せずにビットペアのみを基準にして処理を行っている。ビット依存を考慮せずにX割り当てを行うとキャプチャ遷移削減に影響を及ぼす恐れがある。図3に示す例では、 $\langle v : PPI \rangle = \langle a, b, c \rangle = \langle 1, 1, 1 \rangle$ 、 $\langle f(v) : PPO \rangle = \langle x, y, z \rangle = \langle 0, 1, 1 \rangle$ でキャプチャ遷移は1つ起こっている。 $a$ がXビットと判定された場合、キャプチャ遷移を減らすために通常はビットペアで同じ論理値をXビットに割り当てる。例では0を通常は割り当てるのだが、0を割り当てることによって、ここに新たなキャプチャ遷移が起こってしまう場合がある。これは $a$ のビットが $x, y, z$ のビットに依存しているためである。この場合、 $a$ のビットは0よりも1のほうが遷移数を削減できる。

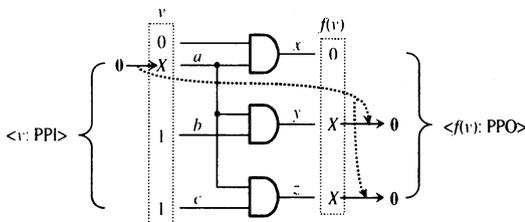


図3 ビット依存を考慮したX割り当て

## 2.4. 従来手法の課題

本論文では、2.2節、2.3節で述べたX判定とX割り当ての問題点を新しいアプローチにより改善する。まずキャプチャ遷移削減に必要なXビットのみを判定していくための制約情報の生成手法を提案する。そして、従来のLCP-X割り当てに加え、Xシンボルシミュレーションを用いてXビットに適切な論理値を割り当てるための手法を提案する。

## 3. 新しいキャプチャ消費電力削減手法

### 3.1. 処理手順の概要

与えられたテスト集合を変換する新しいキャプチャ消費電力削減手法の処理の流れを図4に示す。

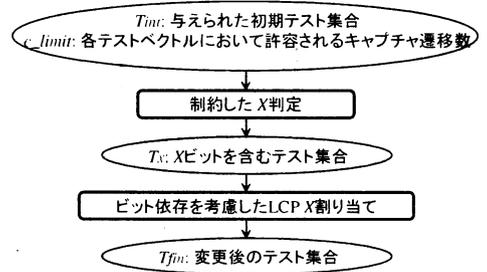


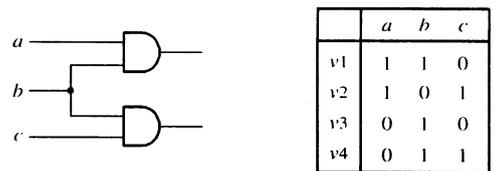
図4 処理の流れ

処理はまず与えられたテスト集合  $T_{ini}$  に対して判定するビットの制約を付加したX判定を行い、 $T_n$ を求める。次にビット依存を考慮したLCP-X割り当ては  $T_n$  に対して行われる。最後にキャプチャ時の消費電力が削減されたテスト集合  $T_{fin}$  が求まる。

### 3.2. 制約つきX判定

#### 3.2.1. 基本的な処理手順

制約つきX判定は文献[25]で説明されている。制約つきX判定の特徴は可変ビットと呼ばれる指定したビットに対しX判定を行うことである。その情報は行列によって表される。図5に例を示す。



(a) 回路と初期テスト集合

	a	b	c
v1	*	-	-
v2	-	-	-
v3	-	-	*
v4	*	*	-

	a	b	c
v'1	1	1	0
v'2	1	0	1
v'3	0	1	X
v'4	X	1	1

(b) 制約行列と制約つきX判定後の結果

図5 制約つきX判定の例

図 5(a)は回路と与えられた初期テスト集合を示す。(b)は制約情報を示す行列とその情報に基づいて X 判定をしたテスト集合である。制約行列の“\*”は可変ビットを示す。制約つき X 判定は X と判定したいビットに X ビットが多く分布するテスト集合を求めることができる。X の分布は制約行列によって変化するので、最適な制約行列を生成することが重要である。

### 3.2.2. 制約行列の生成

制約行列の生成手順は図 6 を用いて説明する。

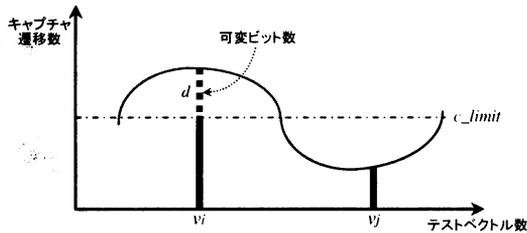


図 6  $c\_limit$  の意味

図 6 に示すように、全てのテストベクトルに対し、それぞれキャプチャ遷移数を計算する。 $v_1$  のような  $c\_limit$  を超えるようなテストベクトルを対象に行う。この時キャプチャ遷移数と  $c\_limit$  の差が  $d$  となり、 $d$  に対応するビットを可変ビットとして制約行列を生成していく。可変ビットの選択方法はまず、キャプチャ遷移が起こっているビットを X に変え、3 値シミュレーションを行う。3 値シミュレーションを行ったときの PPI と PPO のビットペアの組み合わせは表 1 に示すように 4 つに分類される。可変ビットにするビットは入力に X がある Type-B と Type-D である。2.3 節で説明したように Type-B は割り当てベースの X 割り当ての処理になるので、一番単純な処理となる。Type-D は割り当て/正当化ベースの X 割り当て処理になるので処理が複雑になる。よって Type-B のような場合を優先的に可変ビットにする。Type-B で選択した可変ビット数が  $d$  の数を満たさない場合は Type-D も選択しなければならないが、このとき Type-D では  $\langle f(v) : PPO \rangle$  から PPI まで構造的に到達可能な PPI に X ビットを多く含むものから選んでいく。これは正当化操作を容易にするためである。

表 1 3 値シミュレーションによって現れるビットペアの組み合わせ

	$\langle v : PPI \rangle$	$\langle f(v) : PPO \rangle$
Type-A	0 or 1	0 or 1
Type-B	X	0 or 1
Type-C	0 or 1	X
Type-D	X	X

以上のことより制約行列の生成は以下の 3 つの特徴を持っている。(1) キャプチャ遷移の起こるビットペアは可変ビットとして X 判定を制約する情報となる。(2) テストベクトルに対し許容キャプチャ遷移数を決めることで各テストベクトルにバランスのよい分布の X ビットの判定をすることが出来る。(3) 可変ビットは X 割り当ての処理の単純化を考慮して選択される。

### 3.3. ビット依存を考慮した LCP-X 割り当て

X 割り当てを行う際、テストベクトル中に少なくとも 1 つの X ビットがあると仮定し、そのベクトルを X テストベクトル ( $v'$ ) と呼ぶ。ビットペアの組み合わせは表 1 のように 4 つのタイプがある。従来の X 割り当てでは各ビットペアに対してのみに着目して論理値の割り当てを行っていた。しかし、より良いキャプチャ遷移削減効果を得るためにはビット依存も考慮しなければならない。以下ではビット依存を考慮した LCP-X 割り当て手法を提案する。

割り当て・正当化を行う処理というのは従来の手法と同じである。問題はどの X ビットに最適な論理値を割り当てれば従来手法に比べてキャプチャ時の消費電力を削減できるかである。ビット依存を考慮した X 割り当ては X シンボルシミュレーションを用いてビット依存をあらかじめ調べることによって X ビットに最適な論理値を割り当てる。

#### 3.3.1. X シンボルシミュレーション

X シンボルシミュレーションの基本概念は近年発表された故障診断手法で提案されている [26]。本論文では X ビットの伝搬先を特定することにより、ビット依存を考慮したキャプチャ遷移数の削減を実現する。

X シンボルシミュレーションは以下のように行われる。まず X テストベクトル ( $v'$ ) は  $m$  個の X ビットを持っているとする。これらの X ビットは  $X_1, X_2, \dots, X_m$  のように  $m$  個の異なる X 記号を持つように置き換える。X 記号を持ったテストベクトル ( $v''$ ) をシンボルテストベクトルと呼ぶ。そのシンボルテストベクトルを用いて X シンボルを伝搬させるシミュレーションを行う。

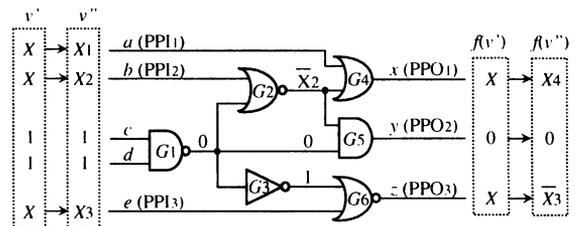


図 7 X シンボルシミュレーション

図 7 に例を示す。3 つの X ビットを持つ X テストベクトル ( $v'$ ) は 3 つのシンボルに置き換えたシンボルテストベクトル ( $v''$ ) になる。そして X シンボルを伝搬させる。G2 の出力は  $X_2$  の逆数をとる。G4 の出力は  $X_1$  と  $X_2$  の逆数の OR となり新しいシンボル  $X_4$  として表す。この X シンボルを持った  $\langle v'' : PPI \rangle, \langle f(v'') : PPO \rangle$  のビットペアの情報は X ビットに最適な論理値を割り当てるために用いる。

#### 3.3.2. 論理値の割り当て

論理値を割り当てるビットペアは  $\langle X, 0 \text{ または } 1 \rangle, \langle 0 \text{ または } 1, X \rangle, \langle X, X \rangle$  の 3 つの種類がある。これらの論理値を決定する方法は以下のように行われる。まず、各 X シンボルに 0, 1 どちらを割り当てた方が良いか決定する。論理値を決定する際の指標の計算方法を示す。

表 2 最適な論理値を割り当てるための指標計算

$\langle v'' \rangle$ : PPI	$\langle f(v'') \rangle$ : PPO	$X_i$	
		0-preference	1-preference
$X_i$	$b_2$	+1 if $b_2 = 0$	+1 if $b_2 = 1$
$b_1$	$X_i$	+1 if $b_1 = 0$	+1 if $b_1 = 1$
$b_1$	$\bar{X}_i$	+1 if $b_1 = 1$	+1 if $b_1 = 0$

( $b_1, b_2$ : 0 or 1)

シンボルテストベクトル( $v''$ )における $\langle v'' \rangle$ : PPIと $\langle f(v'') \rangle$ : PPOのビットペアをそれぞれチェックする。もし $\langle X_i, 0 \rangle$ または $\langle 0, X_i \rangle$ のビットペアならば表 2 を用いて、その  $X$  シンボルは 0,1 どちらの論理値を割り当てるべきか計算する。ビットペアが $\langle X_i, X_i \rangle$ のときは最適な論理値は分からないので計算しない。この計算は1つのテストベクトルに対し全てのビットペアをチェックするまで更新される。そして論理値を決定するとき“0-preference”と“1-preference”の値を比較し値の大きいほうの論理値に決定する。例を図 12 に示す。

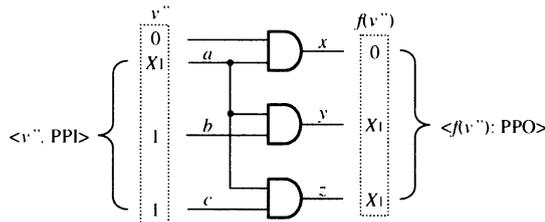


図 12 論理値の決定

ここでは各ビットペアをチェックする時、表 2 を用いて計算すると  $X_i$  の“0-preference”は 1, “1-preference”は 2 となり  $X_i$  には 1 の論理値を割り当てたほうがキャプチャ遷移数を効果的に削減できる。実際、 $X_i$  に 0 を割り当てたとき、キャプチャ遷移数は 2 で、1 を割り当てたときは 1 となっている。

#### 4. 実験結果

キャプチャ時の消費電力を削減するため提案手法を Pentium III 1.3GHz, 512MB メモリの計算機上で C 言語により実装し、ISCAS'89 のベンチマーク回路に対して実験を行った。

表 3 は各回路の基本情報を示す。表中の欄の“# of PIs”、“# of Pos”、“# of F/Fs”、“# of Vectors”は順に外部入力数、外部出力数、スキャンフリップフロップ数、テストベクトル数を示し、ATPG で縮退故障用のテスト集合を生成した時の故障検出率を“Fault Coverage”に示す。また、“Original Max Trans”は、はじめに与えられるテストベクトル集合においてフリップフロップで起こる最大キャプチャ遷移数を示す。

表 3 各回路の基本情報

Circuit	# of PIs	# of Pos	# of F/Fs	# of Vectors	Fault Coverage	Original Max. Trans.
s1238	14	14	18	125	94.9	18
s1423	17	5	74	24	99.1	49
s5378	35	49	179	100	99.1	102
s13207	31	121	669	235	98.5	380
s15850	14	87	597	97	96.7	282
s35932	35	320	1728	12	89.8	1548
s38417	28	106	1636	87	99.5	590
s38584	12	278	1452	114	95.9	925

表 4 は従来の手法を用いた時の実験結果で、制約の無い  $X$  判定[24]をした  $X$  テストベクトルに対し、ビット依存を考慮していない LCP-X 割り当て手法[21]を用いたものである。“ $X$ ”、“CPU”は制約無し of  $X$  判定を行なった時に判定される  $X$  ビット数の割合と CPU 時間である。最初に与えられる初期テスト集合の平均約 65.3% が故障検出率を低下させずに  $X$  ビットになっている。“Random X-Filling”の“Max trans”は  $X$  ビットにランダムに論理値を割り当てた時の最大キャプチャ遷移数を示し、“Old LCP X-Filling”の“Max Trans”、“Red Rate”、“CPU”は従来の LCP X 割り当て手法で論理値を割り当てた時の最大キャプチャ遷移数、表 3 の“Original Max Trans”からの削減率、CPU 時間を示す。従来手法でも“Original Max Trans”と比較すると削減できている。またランダムに割り当てるよりも“Old LCP X-Filling”が良い結果を示している。

表 4 従来手法を用いた実験結果

Circuit	Unconstrained X-Identification		Random X-Filling	Old LCP X-Filling		
	$X$ (%)	CPU (Sec.)	Max. Trans.	Max. Trans.	Red. Rate (%)	CPU (Sec.)
s1238	55.0	0.1	14	9	50.0	0.0
s1423	41.1	0.1	40	34	30.6	0.0
s5378	71.0	1.3	109	91	10.8	0.2
s13207	91.6	8.6	305	244	35.8	5.9
s15850	76.1	5.3	262	173	38.7	1.8
s35932	34.4	4.0	1532	1517	2.0	0.3
s38417	73.4	13.8	600	323	45.3	24.7
s38584	79.7	17.5	802	437	52.8	22.3
Ave.	65.3	—	—	—	33.2	—

表 5 は提案手法を用いて実験を行なった結果である。本論文では制約付きの  $X$  判定[25]を用いる際、キャプチャ遷移削減のための制約行列生成手法とビット依存を考慮した LCP-X 割り当て手法を提案している。表の欄は表 4 と同様である。表 4 の従来手法ではキャプチャ時の消費電力削減のために最大キャプチャ遷移数を平均約 33.2%削減している。一方、表 5 の今回の提案手法では 44.4%削減している。これは、従来の手法から比べて改善率が 33.7%となっている。キャプチャ遷移数は回路全体のノード遷移数と強い相互関係があるので[10]、ノード遷移数は直接キャプチャ時の消費電力に反映する。実験結果では提案手法がキャプチャ時の消費電力削減により効果的であることを示している。実験結果で注意すべき点は制約付き  $X$  判定のため制約行列を生成するとき、3.3.2 で説明したように  $c\_limit$  を設定しなければならず、今回  $c\_limit$  を Original Max Trans の 5% と設定したときの実験結果になっている。

また、表 4 と表 5 の  $X$  の欄を見ると表 4 は平均 65.3% に対し表 5 は平均 23.7%となっている。明らかに表 5 は表 4 に比べ  $X$  ビットの割合が少ない。それに関わらず、より良いキャプチャ時の消費電力削減効果を得ている。これは、 $X$  ビットの数だけではなく、 $X$  ビットの分布がキャプチャ時の消費電力削減効果を上げる際に重要となっていることを示している。

表 5 提案手法を用いた実験結果

Circuit	Constrained X-Identification		Random X-Filling	New LCP X-Filling		
	X (%)	CPU (Sec.)	Max. Trans.	Max. Trans.	Red. Rate (%)	CPU (Sec.)
s1238	35.2	0.1	14	4	72.2	0.0
s1423	11.5	0.0	43	34	30.6	0.0
s5378	27.1	0.4	108	67	31.4	0.4
s13207	35.7	4.6	333	194	47.9	7.8
s15850	21.9	2.7	252	137	53.2	2.4
s35932	13.7	2.9	1533	1486	4.3	1.2
s38417	21.7	13.9	592	286	50.0	10.3
s38584	22.9	18.7	785	402	57.2	12.5
Ave.	23.7	—	—	—	44.4	—

5. まとめ

本論文では、キャプチャ時の消費電力削減を目的とした2つの新しい手法を提案した。一つは制約つきX判定を行なうための制約行列を生成する手法である。これはキャプチャ時の消費電力削減のためにXビットを効果的に分布させることが出来る。もう一つはXビットに最適な論理値を割り当てるためビット依存を考慮したX割り当て手法である。実験結果では提案手法がキャプチャ時の消費電力削減に効果的であることを示した。

謝辞 本研究の一部は、日本学術振興会科学研究費補助金(c)(課題番号 17500039)の助成により行われた。

文 献

[1] M. Abramovici, M. Breuer, and A. Friedman. Digital systems testing and testable design. Computer Science Press (1990).

[2] L.-T. Wang, X. Wen, H. Furukawa, F. Hsu, S. Lin, S. Tsai, K. S. Abdel-Hafez, and S. Wu. VirtualScan. A new compressed scan technology for test cost reduction. *Proceedings of IEEE International Test Conference (2004)*, pp. 916-925.

[3] Y. Zorian. A distributed BIST control scheme for complex VLSI devices. *Proceedings of IEEE VLSI Test Symposium (1993)*, pp. 4-9.

[4] P. Girard. Survey of low-power testing of VLSI circuits. *IEEE Design and Test of Computers (2002)*, vol. 19, pp. 82-92.

[5] J. Altet and A. Rubio. Thermal testing of ICs. Kluwer Academic Publishers (2002).

[6] T. Yoshida and M. Watari. A new approach for low power scan testing. *Proceedings of IEEE International Test Conference (2003)*, pp. 480-487.

[7] R. Chou, K. Saluja, and V. Agrawal. Scheduling tests for VLSI systems under power constraints. *IEEE Transactions on VLSI Systems (1997)*, vol. 5, pp. 175-185.

[8] S. Wang and S. Gupta. ATPG for heat dissipation minimization during test application. *IEEE Transactions on Computers (1998)*, vol. 47, pp. 256-262.

[9] F. Corno, P. Prinetto, M. Redaudo, and M. Reorda. A test pattern generation methodology for low power consumption. *Proceedings of IEEE VLSI Test Symposium (1998)*, pp. 453-459.

[10] R. Sankaralingam, R. Oruganti, and N. Touba. Static compaction techniques to control scan vector power dissipation. *Proceedings of IEEE VLSI Test Symposium (2000)*, pp. 35-40.

[11] S. Kajihara, K. Ishida, and K. Miyase. Test vector modification for power reduction during scan testing. *IEEE VLSI Test Symposium (2002)*, pp. 160-165.

[12] V. Dabholkar, S. Chakravarty, I. Pomeranz, and S. Reddy. Techniques for minimizing power dissipation in scan and combinational circuits during test application. *IEEE Transactions on Computer-Aided Design (1998)*, vol. 17, pp. 1325-1333.

[13] A. Chandra and K. Chakrabarty. Combining low power scan testing and test data compression for System-on-a-Chip. *Proceedings of IEEE Design Automation Conference (2001)*, pp. 166-169.

[14] A. Chandra and K. Chakrabarty. Reduction of SoC test data volume, scan power, and testing time using alternating run-length codes. *Proceedings of IEEE International Conference of Computer-Aided Design (2002)*, pp. 673-678.

[15] A. Hertwig and H. Wunderlich. Low power serial built-in self-test. *Proceedings of IEEE European Test Workshop (1998)*, pp. 49-53.

[16] R. Sankaralingam, R. Oruganti, and N. Touba. Reducing power dissipation during test using scan chain disable. *Proceedings of IEEE VLSI Test Symposium (2001)*, pp. 319-324.

[17] Y. Bonhomme, P. Girard, C. Landrault, and S. Pravossoudovitch. Power driven chaining of flip-flops in scan architectures. *Proceedings of IEEE International Test Conference (2002)*, pp. 796-803.

[18] J. Saxena, K. Butler, and L. Whetsel. A scheme to reduce power consumption during scan testing. *Proceedings of IEEE International Test Conference (2001)*, pp. 670-677.

[19] O. Sinanoglu and A. Orailoglu. Scan power minimization through stimulus and response transformations. *Proceedings of IEEE International Design, Automation and Test in Europe Conference (2004)*, pp. 404-409.

[20] R. Sankaralingam and N. A. Touba. Controlling peak power during scan testing. *Proceedings of IEEE VLSI Test Symposium (2002)*, pp. 153-159.

[21] X. Wen, H. Yamashita, S. Kajihara, L.-T. Wang, K. Saluja, and K. Kinoshita. On low-capture-power test generation for scan testing. *Proceedings of IEEE VLSI Test Symposium (2005)*, pp. 265-270.

[22] X. Wen, Y. Yamashita, S. Morishima, S. Kajihara, L. Wang, K. Saluja, and K. Kinoshita. Low-capture-power test generation for scan-based at-speed testing. To appear in *Proceedings of IEEE International Test Conference (2005)*.

[23] W. Li, S. M. Reddy, and I. Pomeranz. On reducing peak current and power during test. *Proceedings of IEEE Computer Society Annual Symposium on VLSI (2005)*, pp. 156-161.

[24] K. Miyase and S. Kajihara. XID: Don't care identification of test patterns for combinational circuits. *IEEE Transactions on Computer-Aided Design (2004)*, Vol. 23, pp. 321-326.

[25] K. Miyase, S. Kajihara, I. Pomeranz, and S. Reddy. Don't care identification on specific bits of test patterns. *Proceedings of IEEE/ACM International Conference on Computer Design (2002)*, pp. 194-199.

[26] X. Wen, T. Miyoshi, S. Kajihara, L. Wang, K. Saluja, and K. Kinoshita. On per-test fault diagnosis using the X-fault model. *Proceedings of IEEE/ACM International Conference on Computer Design (2004)*, pp. 633-640.