

## SAT を用いたマルチサイクルキャプチャテスト生成法

日大生産工 (院) ○川連 裕斗 日大生産工 細川 利典  
日大生産工 (院) 山崎 紘史

## 1. はじめに

近年の半導体微細化技術の進歩に伴って、大規模集積回路 (Large Scale Integrated circuits: LSI) が大規模化・複雑化し、歩留まりの低下やテスト生成時間の増大といった問題が発生している。歩留まりを向上させる策の一つとして、オーバーテストの抑制が挙げられる。一般的にオーバーテストとは過剰なテストのことであり、LSI のすべての通常動作に影響を与えない故障が、LSI のテストでは故障と判断されてしまうことがオーバーテストの原因となっている。LSI のテストとして広く普及しているフルスキャンテスト [1] では、回路の内部状態を自由に設定できるためオーバーテストを行っていると考えられる。一般的なオーバーテストの削減方法として、順序回路の通常動作において遷移しえない状態 (無効状態) をあらかじめ同定し、無効状態を回避したテスト生成を行う方法が提案されている [2]。また、スキャン設計された回路に対して、 $k$  サイクル間順序動作を行い、できるだけ無効状態に遷移させず縮退故障のテスト生成を行うことでオーバーテストの削減を図る方法が提案されている [3]。

一方、近年では衝突項の学習 [4] や非順序式バックトラック [5]、ブール制約伝搬 [5] の技術により、SAT-solver の処理速度が急激に向上している。それに伴い、回路の等価性検証やテスト生成技術など論理関数を処理する分野で、充足可能性問題 (Satisfiability problem: SAT) を利用した手法が提案されている [6]。

本論文では、テスト容易化手法 [1] の一つであるフルスキャン設計 [1] が施された順序回路に対して、無効状態を利用して縮退故障モデル

表 1. 各論理ゲートの CNF 変換規則

ゲートタイプ	入力	出力	CNF
AND	X Y	Z	$(\bar{Z} + X) \cdot (\bar{Z} + Y) \cdot (\bar{X} + \bar{Y} + Z)$
OR	X Y	Z	$(Z + \bar{X}) \cdot (Z + \bar{Y}) \cdot (X + Y + \bar{Z})$
NAND	X Y	Z	$(Z + X) \cdot (Z + Y) \cdot (\bar{X} + \bar{Y} + \bar{Z})$
NOR	X Y	Z	$(\bar{Z} + \bar{X}) \cdot (\bar{Z} + \bar{Y}) \cdot (X + Y + Z)$
EXOR	X Y	Z	$(\bar{X} + Y + Z) \cdot (X + \bar{Y} + Z) \cdot (\bar{X} + \bar{Y} + \bar{Z}) \cdot (X + Y + \bar{Z})$
NOT	X	Y	$(X + Y) \cdot (\bar{X} + \bar{Y})$

を対象とした  $k$  時間展開モデルを用いたマルチサイクルキャプチャテスト生成法 [3, 7, 8] を提案し、判定されるテスト不可能故障数を評価する。また、無効状態に遷移しないような制約をテスト生成の条件に付加することから SAT を用いたテスト生成法が適しているので、SAT を用いたマルチサイクルキャプチャテスト生成を実装した。

## 2. 充足可能性問題

SAT とは、乗法標準形 (Conjunctive Normal Form: CNF) が与えられたときに、全ての変数の値を 1 (真) または 0 (偽) のどちらかに定めることで、式全体の値を 1 (真) にできる割当てが存在するか否かを判定する問題である。CNF を 1 (真) にできる割当てが存在した場合、充足可能 (SAT) といい、与えられた CNF が正当化可能であることを示す。

SAT は CNF を入力として問題を解決するため、SAT を用いてテストパターンの生成を行うためには、論理回路を CNF に変換する必要がある。

## 2.1. 論理ゲートの CNF 変換

表 1 に各論理ゲートの CNF 変換規則を示す。各論理ゲートは表 1 の規則にしたがい CNF に変換される。それぞれの括弧でくく

れた論理和式を項といい, 項を論理積した式が各ゲートの CNF である. 回路全体の CNF は各論理ゲートの CNF を論理積することで表現できる. 回路全体の CNF は各論理ゲートの CNF を論理積することで表現でき, 信号線に値を割当てて処理では, 項の少なくともひとつが偽であった時点で, その割当ての組み合わせでは正当化が不可能であることがわかる. したがって, 早期に矛盾を発見でき, 高速な正当化可能判定処理が期待できる.

## 2.2. 充足可能性問題を用いたテスト生成

本論文における SAT を用いたテスト生成では, SAT ベース ATPG である TEGUS[9]で提案された CNF 生成規則を用いる. TEGUS で用いられる CNF は, 論理回路を表 1 の CNF 変換規則に基づいて, 正常値を持つ CNF と, 故障影響による故障値を持つ CNF の二つについて変換した後, 故障影響を外部出力まで伝搬させるという制約式を変換した CNF に追加することで生成される. 生成された CNF について, SAT-solver を用いて充足可能性問題を解くことで, 式全体を 1 (真) にできる変数の割当てが存在するとき充足可能 (SAT) であるといい, そのときの外部入力にあたる変数の値がテストパターンとなる. 式全体を 1 (真) にできる変数の割当てが存在しない場合には充足不可能 (UNSAT) であるといい, その故障はテスト不可能故障である.

## 2.3. 無効状態の CNF 変換

無効状態へ遷移しないような制約を CNF に追加するために, 無効状態の CNF 変換を行う [10]. 初めにテスト対象となる回路に対して無効状態の同定を行う. 次に同定された無効状態の情報から, 擬似外部入力に無効状態が割当てられると充足不可能となる制約 CNF を生成する. 2.2 節で説明したテスト対象回路の CNF に無効状態の制約 CNF を追加することで, SAT-solver を用いて充足可能性問題を解くときに無効状態を回避したテスト生成が行われる.

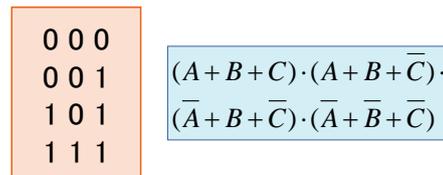


図 1(a)無効状態例 図 1(b)制約 CNF  
図 1. 無効状態例と制約 CNF

図 1 を用いて制約 CNF について説明する. FF を 3 個持つ回路に対して無効状態の同定を行ったとき, 1 行を一組の内部状態とした図 1(a)の無効状態を得られたと仮定する. 図 1(a)の左から 1 列目の FF に A, 2 列目の FF に B, 3 列目の FF に C のリテラルが割り当てられたとすると, 擬似外部入力に無効状態が割当てられると充足不可能となる制約 CNF は図 1(b)のようになる.

## 3. k サイクルキャプチャテスト生成

### 3.1. k サイクルキャプチャテスト

k サイクルキャプチャテスト[3,11]とは, スキャンテストのキャプチャモード時のサイクル数が k であるスキャンテスト方法である. k  $\geq 2$  のとき, マルチサイクルキャプチャテストという.

図 2 に従来のスキャンテストのスキャンイネーブルとクロックの波形を示す. 図 3 に k=4 における k サイクルキャプチャテストのスキャンイネーブルとクロックの波形を示す. 図 2 で示すように従来のスキャンテストのキャプチャモード時のサイクル数は 1 サイクルである. 従来のスキャンテストのキャプチャモード時のサイクル数が 1 サイクルであるのに対し

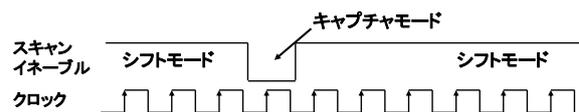


図 2. 従来のスキャンテスト

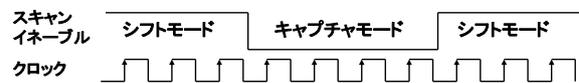


図 3. k サイクルキャプチャテスト (k=4)

て図 3 で示すように  $k$  サイクルキャプチャテスト ( $k=4$ ) では、キャプチャモード時に  $k$  サイクル間 ( $k=4$ ) 順序動作を行う。そのため、 $k$  サイクルキャプチャテストを行うことで、シフト動作によって回路を無効状態に遷移させてテストする回数が減り、オーバーテストを抑制できると考えられる。

### 3.2. $k$ 時間展開モデル

$k$  時間展開モデルとは、1 時刻目のスキャンフリップフロップの出力を擬似外部入力とし、 $k$  時刻目のスキャンフリップフロップのデータ入力を擬似外部出力として、 $k$  時間分順序回路を時間展開した組合せ回路モデルである。このときの展開数が多いほど回路の実動作に近づき、無効状態に遷移する回数が減少する。

### 3.3. SAT ベース $k$ サイクルキャプチャテスト生成

本論文における、SAT を用いた  $k$  サイクルキャプチャテスト生成では、フルスキャン設計された回路を対象とする。それらの回路から  $k$  時間展開モデルを作成し、その  $k$  時間展開モデルに対してテスト生成を行う。

図 4 にフルスキャン設計された回路例を示す。図 4 において、 $X$  は外部入力を表し、 $Z$  は外部出力を表している。 $Y1_D$ 、 $Y2_D$  はフリップフロップの入力を表し、 $Y1_Q$ 、 $Y2_Q$  はフリップフロップの出力を表している SAT を用いた  $k$  サイクルキャプチャテスト生成では、図 4 のようにスキャン設計された回路から図 5 のように  $k$  時間展開 ( $k=3$ ) した時間展開モデルを作成し、これに対して CNF を生成することでテスト生成を行う。図 5 において、 $x(i)$  ( $1 \leq i \leq 3$ ) における外部入力を表し、 $z(i)$  は時刻  $i$  における外部出力を表している。 $y1(1)$ 、 $y2(1)$  は擬似外部入力を表し、 $Y1(3)$ 、 $Y2(3)$  は擬似外部出力を表している。故障モデルは単一縮退故障を仮定する。ただし、 $k$  時間展開モデル上で同一の故障信号線が異なるタイムフレームに何度も出現するので、故障は多重縮退故障として取り扱う必要がある。生成されるテスト

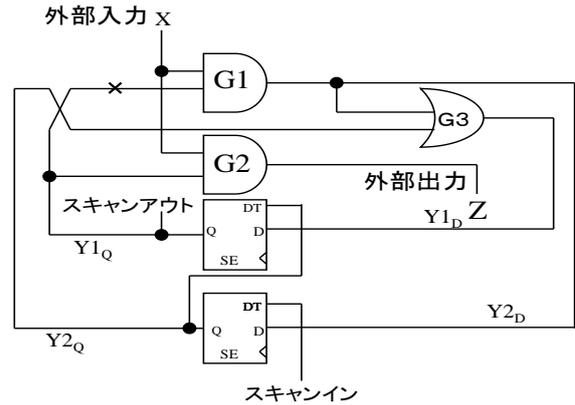


図 4. フルスキャン回路

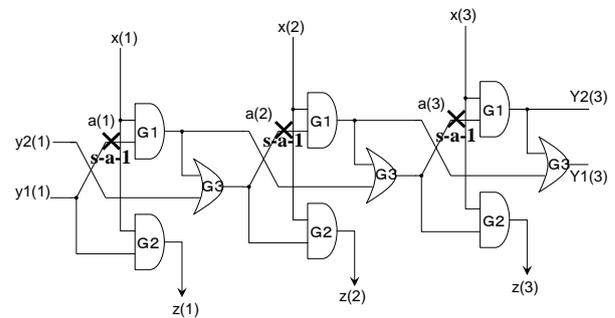


図 5.  $k$  時間展開モデル ( $k=3$ )

系列は、シフトインするパターンと時間展開数分の外部入力系列とシフトアウトするパターンである。

また、図 5 の  $k$  時間展開モデルに対して無効状態を回避する制約 CNF を追加した場合について説明する。図 4 の回路では  $Y1$  と  $Y2$  の 2 つの FF が存在する。無効状態が  $(Y1, Y2) = (0, 0)$  だと仮定すると、図 5 において、疑似外部入力である  $y1(1)$ 、 $y2(1)$  に  $(0, 0)$  と割当てられないような制約 CNF を追加する。つまり、 $(y1(1) + y2(1))$  という制約 CNF を追加することで、無効状態への遷移を回避することが可能となる。このとき、CNF を充足させるためには  $y1(1)$ 、 $y2(1)$  が  $(0, 1)$ 、 $(1, 0)$ 、 $(1, 1)$  のいずれかに割当てられなければならない。

## 4. 実験結果

本論文では、充足可能性問題を用いたマルチサイクルキャプチャテスト生成を実装した。予備実験として ISCAS'89 ベンチマーク回路に

対して、 $k=1$  の時間展開モデルを生成し、無効状態を回避する制約 CNF を追加した場合としない場合とで比較し、テスト生成時間とテスト不可能故障数の評価を行った。表 2 に無効状態制約なしの実験結果、表 3 に無効状態を回避するための制約 CNF を追加した場合の実験結果を示す。結果から、無効状態を回避することでテスト不可能故障を発見できていることが分かる。

表 2. 無効状態制約なし

回路名	FF数	FF状態数	無効状態数	対象故障数	テスト不可能故障数	故障検出率(%)	テスト生成時間(s)
s27	3	8	2	32	0	100.00	0.01
s208	8	256	239	215	0	100.00	0.09
s298	14	16384	16166	308	0	100.00	0.12
s344	15	32768	27426	342	0	100.00	0.21
s386	6	64	51	384	0	100.00	0.24
s420	16	65536	65519	430	0	100.00	0.27

表 3. 無効状態制約あり

回路名	FF数	FF状態数	無効状態数	対象故障数	テスト不可能故障数	故障検出率(%)	テスト生成時間(s)
s27	3	8	2	32	2	93.75	0.01
s208	8	256	239	215	60	72.09	0.72
s298	14	16384	16166	308	35	88.64	76.61
s344	15	32768	27426	342	25	92.69	249.84
s386	6	64	51	384	70	81.77	0.62
s420	16	65536	65519	430	221	48.60	1316.11

## 5. おわりに

実験結果から、無効状態を回避することによりテスト不可能故障の検出を避けるテストパターンが生成できた。ただし、テスト生成時間の増大が問題である。今後の予定として、時間展開数を増やしたモデルでマルチサイクルキャプチャテスト生成を行い、各時間展開数毎と無効状態回避制約 CNF を追加した場合とで比較し、テスト不可能故障数とテスト生成時間の評価を行う。

### 「参考文献」

- 1) H. Fujiwara, "Logic Testing and Design for Testability," The MIT Press, (1985) pp298.
- 2) Yung-Chieh Lin. Feng Lu. Kai Yang. Kwang-Ting Cheng., "Constraint Extraction for Pseudo-Functional Scan-based Delay Testing",

Dept. of ECE, University of California, Santa Barbara Santa Barbara, CA93106, USA. ASPDAC, (2005) pp166-171.

- 3) 大森 悠翔, 細川 利典, 吉村 正義, 山崎 浩二, 小河 宏志, "マルチサイクルキャプチャテストを用いたフルスキャン設計回路のテスト生成", 信学技報, vol.107, No.482, (2008) pp19-24.
- 4) Joao P. Marques-Silva., "GRASP: A Search Algorithm for Propositional Satisfiability", IEEE TRANSACTIONS ON COMPUTERS Volume 48 (1999), pp506-521.
- 5) Matthew W. Moskewicz., "Chaff: Engineering an Efficient SAT Solver", 38th annual Design Automation Conference (2001), pp530-535.
- 6) Tracy Larrabee., "Test Pattern Generation Using Boolean Satisfiability", IEEE TRANSACTION ON COMPUTER-AIDED DESIGN, VOL.11, No.1 (1992).
- 7) I. Pomeranz and S. M. Reddy., "Static Test Compaction for Scan-Based Designs to Reduce Test Application Time", IEEE Asian Test Symposium, (1998), pp541-552.
- 8) J. Abraham, U. Goel and A. Kumar., "Multi-cycle sensitizable transition delay faults", VLSI Test Symposium, (2006), pp306-313.
- 9) Paul Stephan, Robert K. Brayton, and Alberto L. Sangiovanni-Vincentelli., "Combinational Test Generation Using Satisfiability", IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 15, NO. 9, (1996) pp1167-1176.
- 10) 佐藤 駿, "充足可能性問題の解法を用いた無効状態を考慮したテスト生成に関する研究", 平成 22 年度日本大学生産工学部卒業論文 (2010).
- 11) 小河 宏志, 細川 利典, 吉村 正義, 山崎 浩二, 小河 宏志, "マルチサイクルキャプチャ遷移故障テスト生成を用いた テスト不可能故障の原因解析", 信学技報, vol.109, No.416, (2010) pp13-18.