

SAT を用いた到達不能状態判定法

日大生産工 〇二関 森人 日大生産工 細川 利典
 京都産大 吉村 正義 日大生産工 新井 雅之
 徳島大 四柳 浩之 徳島大 橋爪 正樹

1. はじめに

近年の半導体微細化技術の進歩に伴って、超大規模集積回路(Very Large Scale Integrated circuits : VLSI)が大規模化・複雑化し、テストパターン生成時間の増加が問題となっており、テストパターン生成の高速化、容易化が求められている。

テストパターン生成を容易化する技術の一つとして、回路内のフリップフロップ(Flip-Flop : FF)を外から制御、観測可能とするスキャン設計[1,2]が提案されている。しかしながら、スキャン設計が施された回路では、MUX などが挿入されることによるハードウェアオーバーヘッドや、スキャン FF に値を設定、観測するためのシフト動作によるテスト実行時間の増加が課題として挙げられる。また、スキャン設計が施された回路は容易に内部状態を外部に出力することが可能なため、機密情報の流出などにつながる可能性が高くなり、セキュリティ上の課題も発生している。さらに、回路内のスキャン FF の値を自由に設定・観測でき、本来ならば機能動作ではあり得ない内部状態に遷移してテストを行う可能性があるため、過剰テストを行っていると考えられる。上述の課題を解決するために、非スキャンベースでのテストパターン生成[3]が求められている。

しかしながら、非スキャンベースでの順序回路のテストパターン生成では、内部状態を外部入力のみから設定し、故障の影響を外部出力のみで観測するため、スキャン設計なしでは高い故障検出率を得ることが困難である。また、非スキャンベースでのテストパターン生成では、テスト不可能故障に対してテストパターン生成を実行すると、テスト不可能故障判定に多大な時間を必要とする。そのため、スキャン設計なしで高い故障検出率を得るためのテスト容易化設計手法や、テストパターン生成時間を削減するために、テスト不可能故障をあらかじめ判定する手法が提案されている[4-9]。

テスト不可能故障判定法の一つとして、文献[9]で、順序回路の通常の機能動作において遷移しえない状態（到達不能状態）を同定し、同定された到達不能

状態を用いてテスト不可能故障判定を行うアルゴリズムが提案されている。

テスト不可能故障判定の他にも、到達不能状態情報を取得し、その情報を用いることによって、回路の単純化を行う手法[9]や、テスト生成の効率化を行う手法[10]が提案されている。

一方、近年では衝突項の学習[11]や非辞書式バックトラック[12]、ブール制約伝搬[12]の技術により、充足可能性問題(satisfiability problem : SAT)を解く処理速度が急速に進歩している。

本論文では、FF ペア、FF トリプルといった FF の組合せを用い、これらの組合せの状態正当化判定することで、少ない情報量で多数の到達不能状態を判定できる可能性があること、また、SAT を利用することで高速な充足可能判定が実現できることに着目し、SAT に基づく到達不能状態判定法を提案する。

本論文の構成は以下の通りである。第 2 章では、到達不能状態について述べる。第 3 章では、充足可能性問題について述べる。第 4 章では、SAT を用いた到達不能状態判定法について述べる。第 5 章では、実験結果を示し、第 6 章では本論文の結論と今後の課題について述べる。

2. 到達不能状態

順序回路において時刻*i*の内部状態は、時刻*i-1*の外部入力の値と時刻*i-1*の内部状態によって決定される。通常の順序回路では、初期状態としてアンノウン（値が定まっていない）状態が設定されており、初期状態から到達不可能な状態のことを到達不能状態とよぶ。到達不能状態の情報をを用いることで、回路の単純化を図る手法[9]や、テスト生成の効率化を図る手法[10]が提案されている。図1に順序回路の内部状態を状態遷移図として表した図を示す。

3. 充足可能性問題

充足可能性問題 (Satisfiability problem : 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

An Unreachable State Identification Method Using SAT Solver

Morito NISEKI, Toshinori HOSOKAWA, Masayoshi YOSHIMURA, Masayuki ARAI, Hiroyuki YOTSUYANAGI and Masaki HASHIZUME

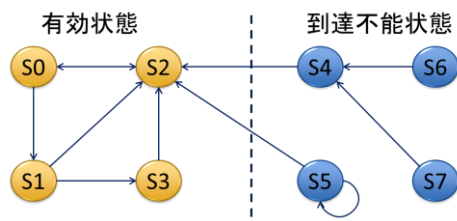


図1. 到達不能状態を含む状態遷移図

表 1. 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	$(\bar{Z} + \bar{X}) \cdot (\bar{Z} + \bar{Y}) \cdot (\bar{X} + \bar{Y} + Z)$
OR	X Y	Z	$(Z + \bar{X}) \cdot (Z + \bar{Y}) \cdot (X + Y + Z)$
NAND	X Y	Z	$(Z + \bar{X}) \cdot (Z + \bar{Y}) \cdot (\bar{X} + \bar{Y} + \bar{Z})$
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	$(X + Y) \cdot (\bar{X} + \bar{Y})$

0 (偽) に定めることで、式全体の値を 1 (真) にできるか否かを判定する問題である。CNF 式を 1 (真) にできる割当てが存在した場合を充足可能といい、存在しない場合を充足不能という。SAT は CNF 式を入力として問題を解くため、提案手法では論理回路を CNF 式に変換する必要がある。論理回路を構成する各ゲートの CNF 変換規則を表 1 に示す。各論理ゲートは表 1 の規則にしたがい、CNF に変換する。それぞれの括弧で括られた論理式を節といい、節を論理積した式が各ゲートの CNF である。回路全体の CNF 式は各論理ゲートの CNF を論理積することで表現できる。信号線に値を割当てた処理では、項の少なくともひとつが偽であった時点で、その割当ての組合せでは CNF 式を 1 にすること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、高速な充足可能判定処理が期待できる。

4. SATを用いた到達不能状態判定法

本章では充足可能性問題を用いた到達不能状態判定法について説明する。4.1で充足可能性問題を用いた到達不能状態判定法の概要について述べ、4.2で充足可能性問題を用いた到達不能状態判定法のアルゴリズムについて述べる。

4.1. SAT を用いた到達不能状態判定法の概要

本論文では論理回路の CNF 式を作成する際に、時間展開モデルを用いる。時間展開モデルとは順序回路の組合せ回路部の接続関係を時間軸に展開した回路である。順序回路は時間展開モデルで表現することによって、組合せ回路として扱うことができる。図 2 に順序回路の例を、図 3 に図 2 の順序回路の時間展開モデル例（時間展開数 2）を示す。

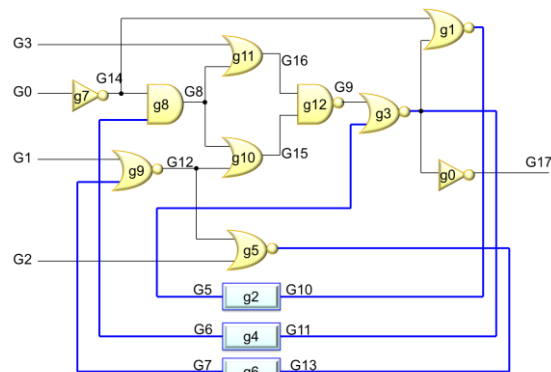


図 2. 順序回路例

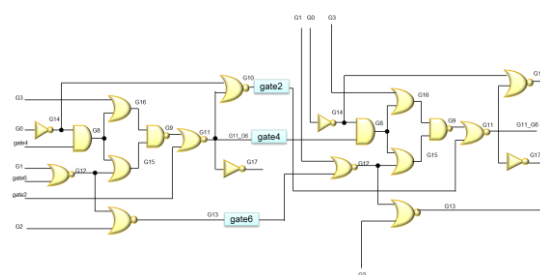


図 3. 時間展開モデル例

本論文で提案する充足可能性問題を用いた到達不能状態判定の概要について説明する。初めに対象となる回路について時間展開モデルを CNF 式に変換する。次に対象モデルにおいて存在する擬似外部出力 (FF) のペア (トリプル) の状態の組合せのうち一つの状態を選択し、時間展開モデルの CNF 式に対し、選択した状態の制約 CNF を付加する。制約 CNF を付与した時間展開モデルの CNF 式に対して充足可能性を判定し、充足可能と判定されれば有効状態である可能性が高いと判断できる。また充足不能と判定された場合は到達不能状態であると判断できる。

4.2. SATを用いた到達不能状態判定法のアルゴリズム

本章では充足可能性問題を用いた到達不能状態判定法のアルゴリズムについて説明する。

初めに、FF ペア (FF トリプル) を作成するために、回路内に存在する FF 毎の影響外部入力を調べる。影響外部入力とは FF の値割当てに影響を与えている可能性のある (擬似) 外部入力のことである。図 2 の回路において FF (g2) の入力となっている G10 の影響外部入力について調べると、(G3, G0, G6, G1, G7, G5) という (擬似) 外部入力に影響していることがわかる。次に、図 2 の回路で FF ペアを求めると、(G10, G11), (G10, G13), (G11, G13) という 3 つのペアが存在することがわかる。例として図 4 に、図 2 の 1 時間展開モデルにおいて FF ペアとペア同士で共通している影響外部入力を求めた例を示す。



図4. FFペア例

ここでFFペア（FFトリプル）を求めている理由として、小さな情報を制約として与え、充足可能かを判定することにより、充足不能だった場合、少ない情報量で多数の到達不能状態を表すことができるためである。例として、図2の回路において、(G10,G11)=(1,1)という組合せが充足不能だった場合、全状態のうち(G10,G11,G13)=(1,1,0)、(1,1,1)の2状態を(G10,G11)=(1,1)という情報で表すことが可能である。このことから、更にFF数が多い回路の場合、充足不能なFFペアの情報は多数の到達不能状態の情報を持つことがわかる。また、FFペア（FFトリプル）の影響外部入力を調べる理由として、FF間の論理関数がより多く共通の（擬似）外部入力変数を用いている場合、それらの状態正当化を行うときに、（擬似）外部入力値が衝突する可能性が高くなると考えたためである。全てのFFペアに対する探索を行った後、対象となる時間展開モデルのCNF式を論理ゲートのCNF変換規則に従い生成する。

CNF式生成後、時間展開モデルのCNF式に前段階で求めたFFペア（FFトリプル）の状態を制約として与え、充足可能性判定を行う。図4の例では、(G10,G11)、(G10,G13)、(G11,G13)という3つのペアが存在しているが、制約としてCNFに付与する場合、値割当てについても考慮する必要がある。ペアの値割当ての組合せとしては、(1,1)、(1,0)、(0,1)、(0,0)という4つの状態が存在する。そのため一つのペアに対して、4つの値割当ての組合せを試行することとなる。制約付与例として、(G10,G11)=(1,1)を制約条件としてCNF式に付与する場合、(G10)(G11)のように単位項で制約条件をCNF式に追加する。

制約を付与したCNF式に対して充足可能性判定を行う。判定後、充足不能となったFFペア（FFトリプル）と値割当ての組合せの情報を利用してCNF式に対して禁止状態制約を追加する。例として、図2の1時間展開モデルにおいて、(G10,G11)=(1,1)が充足不能だった場合、FFの入力となっているG10、G11と出力となっているG5、G6に対して禁止状態制約を追加することが可能となる。例の場合、G10、G11とG5、G6の

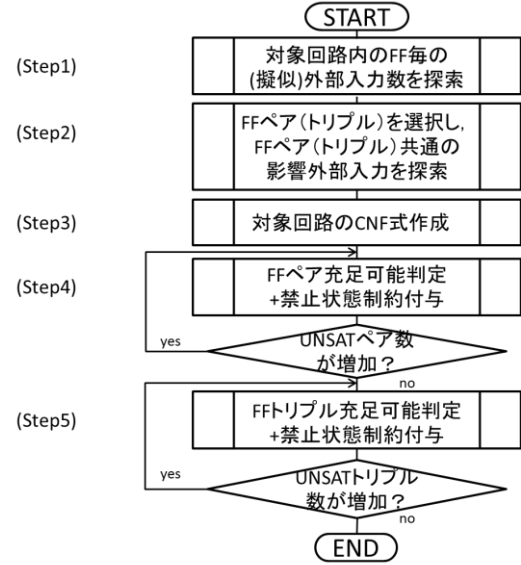


図5. 全体アルゴリズム

2つのペアの変数に対して、変数のどちらかが0に割当てられるように $(\neg G10 + \neg G11)(\neg G5 + \neg G6)$ というCNFを禁止状態制約として追加する。このような禁止状態制約を追加することで、到達不能状態から遷移することができる状態に対する制限がなされるため、図1に表されているように、到達不能状態のみから到達可能な状態を判定することが可能となる。

全てのFFペア（FFトリプル）の組合せに対して状態正当化を実行した後、充足不能となったFFペア（FFトリプル）数の比較を行い、充足不能となったFFペア（FFトリプル）数が増加していれば、再度全てのFFペア（FFトリプル）の組合せに対して充足可能性判定を行い、状態正当化を行う。これはFFペア（FFトリプル）が充足不能と判定された際に追加する禁止状態制約によって新たな到達不能状態が判定できる可能性があるためである。図5に全体アルゴリズムについて示す。

5. 実験結果

本論文ではSATを用いた到達不能状態判定法をC言語で実装した。実験に使用したPCはWindows 8.1, メモリ 8GB, CPUはIntel Core i7-4790 CPUで3.6GHzである。対象回路はISCAS'89ベンチマーク回路で、FF数が21以下の回路に対しては2時間展開モデルでFFペアとFFトリプルを探索し、実験を行った。FF数が21を超える回路に対しては1時間展開モデルでFFペアを探索し、実験を行った。

表2. FF数21以下の回路での実験結果

回路名	FF数	全内部状態数	到達不能状態数		実行時間(sec)	ペア ループ 数	トリプル ループ数
			文献[9]	本手法			
s208	8	256	225	239	1.48	2	2
s298	14	16,384	7,680	15,416	36.75	2	5
s386	6	64	51	51	0.72	2	2
s420	16	65,536	64,800	65,519	160.47	2	2
s444	21	2,097,152	1,789,952	2,045,992	411.03	2	2
s526	21	2,097,152	983,040	1,503,232	299.52	2	4
s641	19	524,288	512,512	521,048	224.80	2	2
s713	19	524,288	512,512	521,048	213.00	2	2
s1196	18	262,144	222,592	256,056	107.10	2	2
s1238	18	262,144	220,672	256,056	106.20	2	2

表3. FF数が21を超える回路での実験結果

回路名	FF数	FFペア×状態数	UNSATFFペア 状態数	実行時間 (sec)	ペア ループ 数
s838	32	1,984	435	3.68	3
s953	29	1,624	158	5.22	2
s1423	74	10,804	4	39.81	2
s5378	179	63,724	661	198.91	2
s9234	228	103,512	98	1129.22	3
s13207	669	893,784	1,562	2675.84	3
s15850	597	711,624	1,800	33652.35	3
s38417	1636	5,349,720	782	160602.13	2
s38584	1452	4,213,704	2,403	823515.58	3

表2にFF数が21以下の回路に対する実験結果を示す。左から回路名, FF数, 全内部状態数, 文献[9]による到達不能状態数, 本手法による到達不能状態数, 実行時間, ペアループ数, トリプルループ数である。表2から, 多くの回路で文献[9]の実験結果よりも多くの到達不能状態を判定することができていることがわかる。

表3にFF数が21を超える回路に対する実験結果を示す。左から回路名, FF数, FFペア, UNSAT (充足不能) となったFFペア, 充足可能性判定の実行時間である。表3から, UNSATとなったFFペアがどの程度存在したかがわかる。例として, s1423においてはUNSATとなったFFペアの数が4となっているが, モンテカルロ法を用いて到達不能状態数を概算した結果, ²⁷²よりも多くの到達不能状態が判定できていることがわかった。これは, 文献[9]で判定されている到達不能状態数と同じ値を示しており, UNSATとなったFFペアが少数でも, 多くの到達不能状態が内包されていることが分かる。

6. おわりに

本論文ではSATを用いた到達不能状態判定法を提案した。また, 提案した到達不能状態判定法を用いて実験を行い, 文献[7]との到達不能状態判定数についての評価を行った。実験結果から, FF数が21以下の回路に対しては, 文献[7]よりも多くの到達不能状態数を判定できていることがわかる。また, FF数が21を超えている回路に対しては, 多くの到達不能

状態情報を内包している可能性のあるUNSATFFペアを判定できていると考えられる。

今後の課題として, より多くの到達不能状態数を求めるために制約条件を追加すること, テスト不可能故障判定への応用, 上流情報を用いた到達不能状態判定法の構築などが挙げられる。

「参考文献」

- [1] H. Fujiwara., “Logic Testing and Design for Testability”, The MIT Press, 1985.
- [2] M. Abramovici, M. A. Breuer and A. D. Friedman., “Digital systems testing and testable design”, IEEE Press, 1995.
- [3] Dong Xiang, Yi Xu and H. Fujiwara., “Non-Scan Design for Testability for Synchronous Sequential Circuits Based on Conflict Analysis”, Test Conference, 2000. Proceedings. International, Oct 2000, pp. 520-529.
- [4] Kazuya Sugiki, Toshinori Hosokawa, and Masayoshi Yoshimura, “A Test Generation Method for Datapath Circuits Using Functional Time Expansion Models”, The Ninth Workshop on RTL and High Level Testing (WRTL’08), pp.39-44, Nov. 2008.
- [5] Tetsuya Masuda, Jun Nishimaki, Toshinori Hosokawa, Hideo Fujiwara., “A Test Generation Method for Data Paths Using Easily Testable Functional Time Expansion Models and Controller Augmentation”, Test Symposium (ATS), 2015 IEEE 24th Asian, Nov 2015, pp.37-42.
- [6] M. A. Iyer, D. Long, and M. Abramovici., “FIRE:a fault independent combinational redundancy identification algorithm”, IEEE Trans, VLSI systems, June 1996, pp. 295-301.
- [7] M. S. Hsiao., “Maximizing Impossibilities for Untestable Fault Identification”, Design, Automation and Test in Europe Conference and Exhibition, 2002. Proceedings, Mar 2002, pp. 949-953.
- [8] V. D. Agrawal and S. T. Chakradhar., “Combinational ATPG theorems for identifying untestable faults in sequential circuits”, IEEE Trans, Vol.14, Nov 2006, pp. 1155-1160.
- [9] 四柳 浩之, 梶原 誠司, 樹下 行三., “到達不能状態に基づく順序回路の冗長除去手法”, 電子情報通信学会論文誌 D-I Vol. J81-D-I No.2, Feb 1998, pp.204-212.
- [10] Yung-Chieh Lin, Feng Lu, Kai Yang, Kwang-Ting Cheng., “Constraint Extraction for Pseudo-Functional Scan-based Delay Testing”, Design Automation Conference, 2005. Proceedings of the ASP-DAC 2005. Asia and South Pacific, Jan 2005, pp166-171.
- [11] J. P. Marques-Silva, K. A. Sakallah., “GRASP: A Search Algorithm for Propositional Satisfiability”, IEEE Transactions on Computers, Vol.48, May 1999, pp506-521.
- [12] M. W. Moskewicz, C. F. Madigan, Ying Zhao, Lintao Zhang, Sharad Malik., “Chaff: Engineering an Efficient SAT Solver”, DAC '01 Proceedings of the 38th annual Design Automation Conference, 2001, pp530-535.