

記号シミュレーションに基づくテストビリティ解析を用いたテスト容易化機能的 k 時間展開モデル選択法

日大生産工 (院) ○中村 健太 日大生産工 (院) 石山 悠太
日大生産工 細川 利典 大院大 藤原 秀雄

1. はじめに

近年, 半導体集積技術の発展に伴い, 設計される大規模集積回路(Large Scale Integrated circuits : LSI)の大規模化, 複雑化が急速に進展している[1]. それに伴い LSI のテストの高品質化や低コスト化が重要な課題となっている. 高い故障検出効率を達成するためには, 何らかのテスト容易化設計(Design for Testability : DFT)が必要である.

現在, スキャン設計[1]が LSI の DFT 手法として普及している. 特に回路中のすべてのフリップフロップ(Flip Flop : FF)をスキャン FF で構成するフルスキャン設計では, 組合せ回路に対するテスト生成技術を適用することが可能となるので, 高い故障検出効率を達成することが可能となる. しかしながら, 回路面積の増大や, 消費電力の増加などのハードウェアオーバーヘッドが増大するという問題点がある.

また, フルスキャン設計の問題点を解決するために, 非スキャンテストに基づいたレジスタ転送レベル(Register Transfer Level : RTL)における DFT 手法が提案されている[2][3]. これらの DFT 手法はコントローラとデータバスから構成される回路を対象にし, そのデータバスのみに着目した手法である. そのため, データバスに対する高い故障検出効率を実現するためには, データバスとコントローラがテスト時に分離されていることが前提となる. テスト時に RTL データバスとコントローラを分離するためには付加回路が必要である.

一方, データバスとコントローラを分離しないことを前提とした RTL 回路に対する DFT 手法も提案されている[4]. 文献[4]ではデータバスのテスト容易な構造に基づいて, テスト時にその動作を制御させるようにコントローラを拡大する. しかしながら, これまで提案されてきた順序回路のテスト生成アルゴリズム[5][6]は, 回路構造のみからテスト系列を生成する. それゆえ, テスト容易な構造に基づいて拡大したコントローラの機能通りにテスト生成するとは限らない. 文献[7]ではデータバスのテスト容易な構造に着目したテスト容易化機能的 k 時間展開モデル(Easily Testable Functional k Time Expansion Models: ETF k -TEM)を用いたテスト生成手法が提案されている. 文献[7]の手法では, データバスのテスト容易な構造に基づいてテスト生成を行うための ETF k -TEM を生成し, その生成されたモデルの動作を実行可能にするためにコントローラを拡大する. テスト生成時は, 拡大したコントローラの機能に着目し, 生成した ETF k -TEM の動作を実行するような制御信号・状態信号系列を制約として与える. ETF k -

TEM を用いてテスト生成することにより, 演算器の故障のテスト生成に関しては高速に高い故障検出効率を達成することができている. しかしながら, 文献[7]の手法は, データバス内の演算器のテスト生成のみを対象とした DFT 手法であり, さらに ETF k -TEM を使用するために, 時系列値を制約値として与えることのできる専用の制約付テスト生成を必要とする.

また, 演算器だけでなく回路中のすべてのハードウェア要素(演算器, マルチプレクサ, レジスタ)に対応する ETF k -TEM の生成とその動作を制御可能にするために無効状態[3]の状態遷移を設計するというコントローラ拡大に基づく DFT 手法を提案されている[8]. この状態遷移が設計された状態をテスト活性化用状態[9]と呼ぶ. 文献[8]では, コントローラ中の状態レジスタとデータバスからコントローラへの状態信号に接続している状態信号レジスタをスキャン設計したパシナルスキャン設計を前提としている. コントローラをスキャン設計することで, テスト活性化用状態を含む任意の状態にシフト動作で遷移することが可能で, 任意の状態から k サイクル間状態遷移を実行することにより, 各ハードウェア要素に対する ETF k -TEM が回路構造に基づくテスト生成において実行可能となる.

文献[7][8]で用いている ETF k -TEM は, 同じハードウェア要素に対し, 複数の ETF k -TEM を生成することが可能である. しかしながら, 文献[7][8]では, 複数できる ETF k -TEM の中から, どの ETF k -TEM を使用するかの選択は, ETF k -TEM の時間展開数, 再収斂数, 定数制御演算器数が少ないモデルを選択している. つまり, 時間展開数, 再収斂数, 定数制御演算器数がどの程度 ETF k -TEM のテストビリティの低下に影響を及ぼすかが定量化できていない.

本論文では, 記号シミュレーションに基づくテストビリティ尺度を提案し, その尺度を用いたテストビリティ解析を行い, テスタビリティの高い ETF k -TEM の選択法を提案する.

2. テスト容易化設計手法

2.1. ETF k -TEM

ETF k -TEM[7]とは, データバスのテスト容易な構造に着目して生成された k サイクルテスト生成モデルである. ETF k -TEM は, データバスの全てのハードウェア要素(演算器, マルチプレクサ, レジスタ)をテストできるように生成する必要がある. 図 1 にデータバス例を示し, 図 2 に図 1 の演算器*1 をテ

An Easily Testable Functional k -Time Expansion Model Selection Method Using Testability Analysis Based on Symbolic Simulation

Kenta NAKAMURA, Yuta ISHIYAMA, Toshinori HOSOKAWA and Hideo FUJIWARA

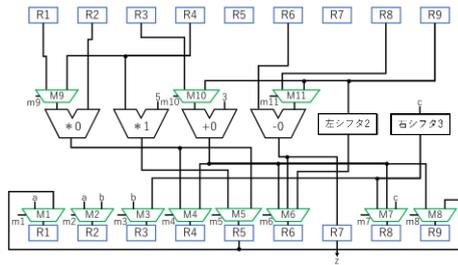


図 1. データパス例

ト可能な ETF5-TEM の例を示す. 表 1 に図 2 の ETF5-TEM の動作を実行するための各時刻の制御信号値 $C \in \{0,1,X\}$ と状態信号値 $S \in \{0,1,X\}$ の時系列であるテスト動作制御・状態信号系列[7]を示す. 図 1 において, a, b, c は外部入力, z は外部出力, R1~R9 はレジスタ, M1~M11 はマルチプレクサ, $m1 \sim m11$, は制御信号, *0, *1 は乗算器, +0 は加算器, -0 は減算器, 左シフト 2 は 2 ビット左にシフトするシフト, 右シフト 3 は 3 ビット右にシフトするシフトである. 図 2 において, a_1, a_2 は同じ外部入力 a であるが, 時刻が異なるため, 異なる値を入力できるため, 別々の変数として表している. 表 1 において, $t0 \sim t4$ は時刻を表し, 図 2 における四角で囲まれた時刻 0~4 にそれぞれ対応している. ETF k -TEM において, 外部入力または時刻 0 のスキャンレジスタからハードウェア要素の入力何らかの値を伝搬でき, ハードウェア要素の出力から何らかの値を外部出力または最終時刻のスキャンレジスタに伝搬できるものを, その ETF k -TEM で動作可能[8]であるという. マルチプレクサの左から n 番目の入力を入力 n と表す(n は非負の整数). 図 2 の ETF5-TEM において, *0, *1, 左シフト 2, -0, R1, R2, R4, R6, R7, R8, R9, M1 の入力 0, M2 の入力 0, M4 の入力 1, M6 の入力 2, M8 の入力 1, M11 の入力 0 が動作可能である.

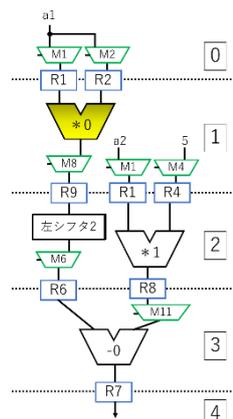


図 2. 表 1 で動作可能な ETF5-TEM 例

表 1. テスト動作制御・状態信号系列の例

	m1	m2	m3	m4	m5	m6	m7	m8	m9	m10	m11
t0	0	0	x	x	x	xx	xx	x	x	x	x
t1	0	x	x	1	x	xx	xx	1	x	x	x
t2	x	x	x	x	x	10	xx	x	x	x	x
t3	x	x	x	x	x	xx	xx	x	x	x	0
t4	x	x	x	x	x	xx	xx	x	x	x	x

2.2. コントローラ拡大

コントローラ拡大[3][7][8]とは, コントローラに状態や状態遷移を追加する設計手法のことである. コントローラ中にはリセット状態から遷移し得ない状態が存在する場合があります, その状態を無効状態[3]という. 文献[8]の手法では, コントローラの状態レジスタをスキャン設計することで, テスト時において, 全状態から無効状態に遷移可能となる. したがって, 本手法では, 文献[8]の手法を適用することで, ETF k -TEM の動作を実行するためのテスト動作制御・状態信号系列を出力する状態遷移を無効状態にのみ設計する. この無効状態のことをテスト活性化用状態[9]と呼ぶ. データパスのテスト容易な構造に着目して生成された ETF k -TEM の動作を実行する機能は, コントローラに備わっていない可能性がある. よって, ETF k -TEM を考慮したテスト動作制御・状態信号系列を新たな状態遷移としてテスト活性化用状態の状態遷移に設計することで, ETF k -TEM の動作を実行可能となる. なお, ETF k -TEM を考慮したコントローラ拡大時に無効状態数が不足する場合は, コントローラの状態レジスタのビット幅を増加させて, コントローラの無効状態数を増加させる.

3. テスト容易化機能的 k 時間展開モデル選択法

3.1. 提案手法の概要

従来手法[7][8]では, ETF k -TEM を生成する際にデータパス内の 1 つのハードウェア要素をターゲットとし, ターゲットとなるハードウェア要素の入力から外部入力またはスキャンレジスタまでの制御経路を探索する. また, ターゲットとなるハードウェア要素の出力から外部出力またはスキャンレジスタまでの観測経路を探索し, 観測経路内のハードウェア要素の観測経路ではない入力の制御経路を探索する. 次に, 生成された ETF k -TEM のうち, 最も時間展開数, 再収斂数, 定数制御演算器数が少ない ETF k -TEM を選択している. しかしながら, この方法では時間展開数, 再収斂数, 定数制御演算器数がどの程度 ETF k -TEM のテストバリエーションの低下に影響を及ぼすかが定量化できていない. 本手法では ETF k -TEM に対する記号シミュレーションに基づくテストバリエーション尺度を提案し, その尺度を用いてテストバリエーション解析を行い, テストバリエーションの高い ETF k -TEM の選択法を提案する.

3.2. テスタバリエーション解析

テストバリエーション解析では, まず ETF k -TEM に対し記号シミュレーションを行い, 可制御性を計算, 可観測性の計算を行う. ETF k -TEM の可制御性・可観測性は各ハードウェア要素に接続されている信号線の値がどの程度制御・観測可能なかを求める. 制御信号線の各時刻の論理値は固定されているため, レジスタとマルチプレクサを通過しても可制御性・可観測性は変わらないと考える. 図 3 に図 2 の ETF5-TEM のレジスタとマルチプレクサを省略した図を示す. また, n ビットの回路で, 各信号線は 2^n 通りの値の中で 2^i 通りの値を制御・観測可能である時, そ

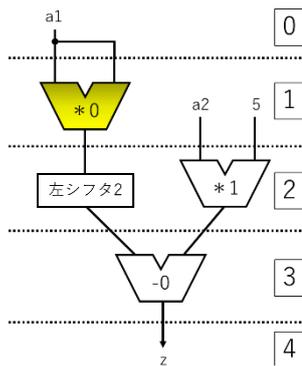


図 3. 省略した ETF5-TEM

の信号線は j/n 制御・観測可能であると表す。

3.2.1. 記号シミュレーション

記号シミュレーションでは、ETF k -TEM から各信号線の数式を求める。図 3 の ETF5-TEM で外部入力から記号シミュレーションを行う。*0 の入力はどちらも「 a_1 」なので出力は「 $a_1 \times a_1$ 」となる。左シフト 2 は 2 ビット左シフトを行い、2 ビット左シフトは「入力の値 $\times 4$ 」と等価である。左シフト 2 の入力は「 $a_1 \times a_1$ 」なので出力は「 $a_1 \times a_1 \times 4$ 」となる。*1 の左入力は「 a_2 」、右入力は「5」なので出力は「 $a_2 \times 5$ 」となる。-0 の左入力は「 $a_1 \times a_1 \times 4$ 」、右入力は「 $a_2 \times 5$ 」なので出力は「 $(a_1 \times a_1 \times 4) - (a_2 \times 5)$ 」となる。つまり図 3 の ETF5-TEM の記号シミュレーションの結果の数式は「 $z = (a_1 \times a_1 \times 4) - (a_2 \times 5)$ 」となる。

3.2.2. 可制御性の計算

可制御性の計算は時刻 0 から順に行う。記号シミュレーションの結果から、各演算器の出力信号線が何通りの値を制御することができるかを計算し、可制御性を求める。図 3 の ETF5-TEM では、32 ビットの回路だとすると、外部入力 a_1 , a_2 は 32 ビット中 32 ビットの値を制御することができるので、外部入力信号線の可制御性は 32/32 となる。*0 の出力信号線は「 $a_1 \times a_1$ 」が取りうる値が 2^{16} 通りであることから 16/32 となる。左シフト 2 の出力信号線は「 $a_1 \times a_1 \times 4$ 」が取りうる値が 2^{14} 通りであることから 14/32 となる。*1 の出力信号線は「 $a_2 \times 5$ 」が取りうる値が $2^{29.68...}$ 通りであることから 29.68.../32 となる。-0 の出力信号線は「 $(a_1 \times a_1 \times 4) - (a_2 \times 5)$ 」が取りう

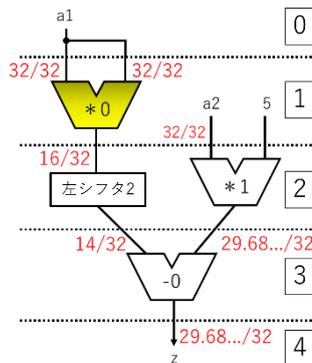


図 4. ETF5-TEM の可制御性

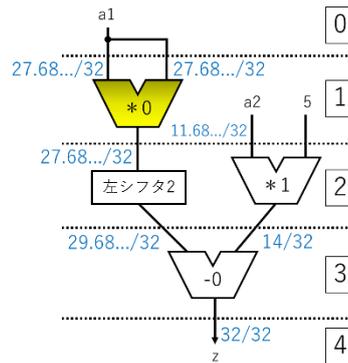


図 5. ETF5-TEM の可観測性

る値が $2^{29.68...}$ 通りであることから 29.68.../32 となる。図 4 に図 3 の ETF5-TEM の可制御性の計算結果を示す。

3.2.3. 可観測性の計算

可観測性の計算は外部出力から順に行う。可観測性の計算は、演算器の他方の入力信号線の可制御性と演算器の出力信号線の可観測性を乗算することで求める。しかしながら、定数制御演算器、シフトの場合は例外として、出力信号線の可観測性から \log_2 定数値/ビット幅で求める。図 2 の ETF5-TEM では、外部出力 z は 32 ビット中 32 ビットの値を観測することができるので、外部出力信号線の可観測性は 32/32 となる。-0 の右入力信号線の可観測性は、他方の入力信号線の可制御性が 29.68.../32、出力信号線の可観測性が 32/32 なので、それらを乗算し、29.68.../32 となる。-0 の左入力信号線の可観測性は、他方の入力信号線の可制御性が 14/32、出力信号線の可観測性が 32/32 なので、それらを乗算し、14/32 となる。*1 の左入力信号線の可観測性は、定数制御演算器で定数値が「5」であるので出力信号線の可観測性の 32/32 から $\log_2 5$ を引き、11.68.../32 となる。左シフト 2 の入力信号線の可観測性は、シフトで 2 ビットシフトなので定数値の「4」の定数制御乗算器として考えるので、出力信号線の可観測性の 29.68.../32 から $\log_2 4$ を引き、27.68.../32 となる。*0 の右入力信号線の可観測性は、他方の入力信号線の可制御性が 32/32、出力信号線の可観測性が 27.68.../32 なので、それらを乗算し、27.68.../32 となる。*0 の左入力信号線の可観測性は、他方の入力信号線の可制御性が 32/32、出力信号線の可観測性が 27.68.../32 なので、それらを乗算し、27.68.../32 となる。図 5 に図 3 の ETF5-TEM の可観測性の計算結果を示す。

3.2.4. テスタビリティ尺度

ETF k -TEM に対し記号シミュレーションを行い、可制御性を計算、可観測性の計算を行った結果からテスタビリティ尺度として用いるのはターゲットとしているハードウェア要素の入力信号線の可制御性と出力信号線の可観測性である。つまり、図 2 の ETF5-TEM の例では、ターゲットとしているハードウェア要素は *0 なので、*0 の左入力信号線の可制御性 32/32、*0 の右入力信号線の可制御性 32/32、*0 の出力信号線の可観測性 27.68.../32 がテスタビリティ尺度となる。

表 2. 故障検出率とテストバリエーション尺度の比較

	時間展開数	再収数	定数制御演算器数	テストバリエーション尺度	故障検出効率 (%)
モデル1	7	0	1	0.77	99.92
モデル2	7	0	2	0.93	100.00
モデル3	7	0	1	1.00	100.00
モデル4	7	1	2	0.79	100.00
モデル5	7	1	1	0.77	99.53
モデル6	7	1	2	1.00	100.00
モデル7	7	1	0	0.84	99.84
モデル8	7	2	1	0.37	99.56
モデル9	7	2	0	0.91	99.94
モデル10	7	2	1	1.00	100.00
モデル11	7	1	1	1.00	100.00
モデル12	7	0	1	1.00	100.00
モデル13	7	0	1	0.84	99.91
モデル14	7	0	2	0.91	100.00

4. 実験結果

本論文では、本手法の有効性を示すため、図 1 のデータパスの*0をターゲットとした E_{TFk}-TEM を 14 通り生成し、14 通りのモデルを用いた実験結果を示す。本実験の故障モデルは単一縮退故障であり、E_{TFk}-TEM のターゲットとしたハードウェア要素である*0のみの故障を対象としている。実験回路の論理合成には Synopsys 社の DesignCompiler を用いたテスト生成は内製のテスト生成ツールである FTEM-ATPG を用いた。テストバリエーション尺度は*0の入力信号線の可制御性と出力信号線の可観測性を総積した値を用いた。本実験では、実験対象の 14 通りの E_{TFk}-TEM に対し、それぞれテスト生成を行い、テストバリエーション尺度と故障検出効率との相関を解析した。次に、従来手法の E_{TFk}-TEM の選択法と提案手法の E_{TFk}-TEM の選択法を比較した。

表 2 に故障検出効率とテストバリエーション尺度の比較を示す。表 2 において、1 列目に実験対象モデルを示し、2 列目に時間展開数、3 列目に再収数、4 列目に定数制御演算器数、5 列目にテストバリエーション尺度を示し、6 列目に故障検出効率 (%) を示す。従来手法では、時間展開数、再収数、定数制御演算器数を基に E_{TFk}-TEM を選択していたので、モデル 1、モデル 3、モデル 7、モデル 12、モデル 13 がテストバリエーションの良いモデルとして選択されていた。しかしながら、モデル 1、モデル 7、モデル 12 の 3 つのモデルは故障検出効率が低い結果となった。また、提案手法であるテストバリエーション尺度を基に選択すると、モデル 3、モデル 6、モデル 10、モデル 11、モデル 12 がテストバリエーションの良いモデルとして選択する。テストバリエーション尺度を基に選択したモデルの故障検出効率は全て 100.00%となり、テストバリエーションの良いモデルの選択に成功している。

しかしながら、故障検出効率が高いが、テストバリエーション尺度が低い E_{TFk}-TEM も存在する結果となった。

5. おわりに

本論文では、記号シミュレーションに基づくテストバリエーション解析を用いたテスト容易化機能的 k 時間展開モデル選択法を提案した。14 種類の E_{TFk}-TEM を用いて従来手法の選択法と、提案手法の選択法を比較すると、提案手法の選択法では、故障検出効率が高い E_{TFk}-TEM の選択に成功した。しかしながら、故障検出効率が高いが、テストバリエーション尺度が低い E_{TFk}-TEM も存在する結果となった。今後の課題と

しては、テストバリエーション尺度の精度を向上することや、より大規模な回路での実験が挙げられる。

参考文献

- [1] 藤原 秀雄, デジタルシステムの設計とテスト, 工学図書株式会社, 2004.
- [2] H. Wada, T. Masuzawa, K.K. Saluja, and H. Fujiwara, "Design for strong testability of RTL data paths to provide complete fault efficiency," Proc. of 13th Int. Conf. on VLSI Design, pp. 300-305, 2000.
- [3] S. Ohtake, T. Masuzawa, and H. Fujiwara, "A non-scan approach to DFT for Controllers Achieving 100% Fault Efficiency, " Journal of Electronic Testing: Theory and Applications (JETTA), Vol. 16, No. 5, pp. 553-566, Oct. 2000.
- [4] L.M. Flottes, B. Rouzeyre, L. Volpe, "A Controller Resynthesis Based Methods for Improving Datapath Testability," IEEE International Symposium on Circuits and Systems, pp. 347 -350, May 2000.
- [5] W.T. Cheng, "The back algorithm for sequential test generation," Proc. 1988 IEEE Int. Conf. on Computer Design, pp. 66-69, Oct. 1988.
- [6] T.M. Niermann and J.H. Patel, "HITEC : A Test Generation Package for Sequential Circuit, " Proc. of the European Design Automation Conf. pp. 214-218, Feb. 1991.
- [7] T. Masuda, J. Nishimaki, T. Hosokawa and H. Fujiwara, "A Test Generation Method for Datapaths Using Easily Testable Functional Time Expansion Models and Controller Augmentation, " IEEE the 24th Asian Test Symposium (ATS'15), pp. 37-42, Nov. 2015.
- [8] Y. Ishiyama, T. Hosokawa and H. Yamazaki, "A Design for Testability Method for k-Cycle Capture Test Generation, " Proc 25th International Symposium on On-Line Testing and Robust System Design (IOLTS 2019), pp.40-43, 2019.
- [9] Y. Takeuchi, T. Hosokawa, H. Yamazaki, M. Yoshimura, "A Controller Augmentation Method to Improve Transition Fault Coverage for RTL Data-Paths," IEEE 25th International Symposium on On-Line Testing and Robust System Design, pp.293-298, Rhodes Island, Greece, July, 2019.