

修士論文

題目

誤り出力数を考慮した組込み自己診断用テスト生成法に関する研究

指導教官

高橋寛 教授

王森レイ 講師

報告者

松田 優大

平成 31 年 2 月 13 日提出

愛媛大学大学院 理工学研究科 電子情報工学専攻 情報工学コース

内容梗概

近年，半導体製造技術の微細化に伴って集積回路の高速化・高集積化が実現されている．集積回路は様々な機器に広く利用されており，我々の生活に欠かせない存在となっている．特に現代社会において発展を続ける製品として自動車が挙げられる．自動車に搭載される車載システムでは，先進自動運転技術などの高度な機能安全性を実現するために集積回路の高信頼化が必要とされている．そのため，あらかじめ集積回路内部に評価，診断用の回路を組み込む手法など，集積回路の評価・診断容易化設計のための要素技術の開発が望まれている．

システムを安全に利用するための技術として，フィールド上で集積回路が正常かどうかを判断する組込み自己テスト (Built-in Self Test: BIST) が存在する．BIST は LSI チップ内に LSI テスターの機能の一部を組み込む手法である．搭載するテスターの機能は，「テストパターン生成回路」，「期待値と回路からの出力値を比較する回路」の 2 つである．BIST では LSI チップの内部でテストパターンを生成するため，テスターへの投資を削減することができコスト面での利点がある．

さらに BIST 機構を拡張して，集積回路の長期機能安全性を保障するための技術として組込み自己診断 (Built-in Self Diagnosis: BISD) の開発が望まれている [2] [3]．BIST が故障の有無を判断するのに対して，BISD は故障の有無に加えて故障箇所の推定を行うことができる．しかしながら，BISD を実現するためには課題がある．BISD 機構では被検査回路に対して，診断用の回路を付与する必要がある．診断用回路とは前述した BIST 機構に加えて，「テストパターン系列に対する被検査回路のパス/フェイル情報を生成するための回路」，「被検査回路のパス/フェイル情報に基づいて診断用署名を生成するための回路」，「テストパターンに対する期待値署名を格納するためのメモリ」が挙げられる．BISD では，あらかじめ被疑故障署名を期待値署名用メモリ内に格納

し、被検査回路から生成された診断用署名を比較することで被疑故障を推定する。診断用署名の生成には、被検査回路の出力側フリップフロップの値を圧縮する必要がある。しかしながら、集積回路の大規模化・複雑化によって診断用回路の規模も増加している中ですべてのフリップフロップを用いて署名の作成を行うことはハードウェア制約として問題がある。

そこで、本研究では、上記の課題を解決するために故障診断用にテストパターンを生成する手法を提案する。まず BISD 機構について説明する。次に、BISD における故障診断能力について説明する。そして、ハードウェア制約を改善するためのテストパターン生成法について説明する。最後に、生成したテストパターンを用いて故障検出フリップフロップ数と、故障診断能力に関する評価を行う。

本論文の構成は以下の通りである。第 1 章では、研究の背景および目的について述べる。第 2 章では、本研究で使用する用語について述べる。第 3 章では、組込み自己診断機構について述べる。第 4 章では、誤り出力数を考慮した組込み自己診断用テスト生成法について述べる。第 5 章では、提案手法によって生成したテストパターンを用いて、誤り出力数と診断能力の評価結果を示す。第 6 章では、本研究のまとめを述べる。

目次

内容梗概	i
第 1 章 序論	1
第 2 章 準備	3
2.1 節 諸定義	3
第 3 章 組込み自己診断	7
3.1 節 BIST 機構	7
3.2 節 BISD 機構	8
3.3 節 署名の生成・比較	10
3.4 節 BISD 機構の診断能力	12
第 4 章 誤り出力数を考慮した組込み自己診断用テスト生成法	13
4.1 節 誤り出力数を考慮した組込み自己診断用テスト生成法	13
4.2 節 シミュレーションモデル	15
4.2.1. 故障辞書	16
4.2.2. 診断用テスト生成法のアルゴリズム	17
第 5 章 評価実験	20
5.1 節 誤り出力数を考慮した組込み自己診断用テスト生成法	20
第 6 章 結論	24

愛媛大学大学院

理工学研究科 電子情報工学専攻 情報工学コース

目次

謝辞	26
参考文献	27
発表論文リスト	28

図 目 次

2.1	遷移故障	4
2.2	2時刻展開回路	6
3.1	BIST 機構	8
3.2	BISD 機構	9
3.3	署名作成・比較	10
3.4	クラスの作成	12
4.1	変更前のテストパターン使用時の回路	14
4.2	変更後のテストパターン使用時の回路	14
4.3	テスト生成シミュレーションモデル	15
4.4	故障辞書	16
4.5	変更前のテストパターンを用いた故障検出	18
4.6	変更後のテストパターンを用いた故障検出	18

表 目 次

5.1	実験に利用した計算機の諸元	20
5.2	提案手法における診断用テストパターンの評価結果 (cs9234)	21
5.3	提案手法における診断用テストパターンの評価結果 (cs13207)	21
5.4	提案手法における診断用テストパターンの評価結果 (cs15850)	22
5.5	故障診断結果 (cs9234)	22
5.6	故障診断結果 (cs13207)	23
5.7	故障診断結果 (cs15850)	23

第 1 章

序論

近年，半導体製造技術の微細化に伴って集積回路の高速化・高集積化が実現されている．集積回路は様々な機器に広く利用されており，我々の生活に欠かせない存在となっている．特に現代社会において発展を続ける製品として自動車が挙げられる．自動車の搭載される車載システムでは，先進自動運転技術などの高度な機能安全性を実現するために集積回路の高信頼化が必要とされている．そのため，あらかじめ集積回路内部に評価，診断用の回路を組み込む手法など，集積回路の評価・診断容易化設計のための要素技術の開発が望まれている．

システムを安全に利用するための技術として，フィールド上で集積回路が正常かどうかを判断する組込み自己テスト (Built-in Self Test: BIST) が存在する．BIST は LSI チップ内に LSI テスターの機能の一部を組み込む手法である．搭載するテスターの機能は，「テストパターン生成回路」，「期待値と回路からの出力値を比較する回路」の 2 つである．BIST では LSI チップの内部でテストパターンを生成するため，テスターへの投資を削減することができコスト面での利点がある．

さらに BIST 機構を拡張して，集積回路の長期機能安全性を保障するための技術として組込み自己診断 (Built-in Self Diagnosis: BISD) の開発が望まれている [2] [3]．BIST が故障の有無を判断するのに対して，BISD は故障の有無に加えて故障箇所の推定を行うことができる．しかしながら，BISD を実現するためには課題がある．BISD 機構では被検査回路に対して，診断用の回路を付与する必要がある．診断用回路とは前述した BIST 機構に加えて，「テストパターン系列に対する被検査回路のパス/フェイル情

報を生成するための回路」,「被検査回路のパス/フェイル情報に基づいて診断用署名を生成するための回路」,「テストパターンに対する期待値署名を格納するためのメモリ」が挙げられる。BISD では,あらかじめ被疑故障署名を期待値署名用メモリ内に格納し,被検査回路から生成された診断用署名を比較することで被疑故障を推定する。診断用署名の生成には,被検査回路の出力側フリップフロップの値を圧縮する必要がある。しかしながら,集積回路の大規模化・複雑化によって診断用回路の規模も増加している中ですべてのフリップフロップを用いて署名の作成を行うことはハードウェア制約として問題がある。

そこで,本研究では,上記の課題を解決するために故障診断用にテストパターンを生成する手法を提案する。まず BISD 機構について説明する。次に, BISD における故障診断能力について説明する。そして,ハードウェア制約を改善するためのテストパターン生成法について説明する。最後に,生成したテストパターンを用いて故障検出フリップフロップ数と,故障診断能力に関する評価を行う。

本論文の構成は以下の通りである。第 1 章では,研究の背景および目的について述べる。第 2 章では,本研究で使用する用語について述べる。第 3 章では,組込み自己診断機構について述べる。第 4 章では,誤り出力数を考慮した組込み自己診断用テスト生成法について述べる。第 5 章では,提案手法によって生成したテストパターンを用いて,誤り出力数と診断能力の評価結果を示す。第 6 章では,本研究のまとめを述べる。

第 2 章

準備

本章では，本論文で使用する各用語について取り扱う [7] [8].

2.1 節 諸定義

本節では，本論文で利用する用語の定義を行う．

定義 1 テストパターン

被検査回路の外部入力線に，ある論理値を印加すると，回路内部の全信号線の論理値が決定する．この時，印加した論理値の組み合わせをテストパターンという．また，複数のテストパターンの集合をテスト集合という．

定義 2 故障リスト

被検査回路内における，故障が発生している信号線番号と故障情報の組み合わせの一覧を故障リストという．

定義 3 故障モデル

あらかじめ起こり得る故障を論理的な振る舞いに注目してモデル化したものを故障モデルという．本論文で取り扱う故障モデルはすべて，回路に一つだけ故障を仮定す

る単一故障であるとする.

定義 4 遅延故障

回路内の信号線の論理値の変化の際に遅延が生じ, その遅延の蓄積がタイミングマージンを超えてしまうと, 遷移故障が生じる. 遷移故障には, 信号線の論理値が 0 から 1 に変化する際の遅延による立ち上り遷移故障と, 信号線の論理値が 1 から 0 に変化する際の遅延による立ち下がり遷移故障が存在する.

図 2.1 に例を示す. 図のように AND ゲート出力で遷移故障が生じているとする. このとき 1 時刻目のテストパターンを t_1 , 2 時刻目のテストパターンを t_2 とし, 図のように論理値の変化が起こるものとする. 正常に論理値が変化すれば, 0 から 1 への立ち上がりの変化が起こるが, この際遅延付加が生じると, 値が立ち上がらずこの AND ゲートの論理値は固定値 0 となる.

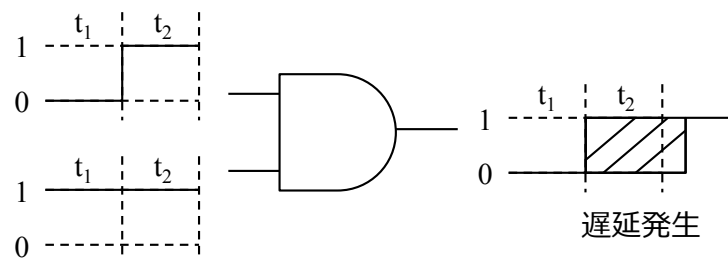


図 2.1. 遷移故障

遷移故障の励起および伝搬条件は以下のようなになる.

励起条件 故障信号線における論理値の変化が立ち上がり (立ち下がり) であるとき.

伝搬条件 故障の影響が外部出力まで伝搬すること.

また, 本研究で扱う遷移故障モデルでは遅延が生じるタイミングまでは考慮しないものとする.

定義 5 論理シミュレーション

計算機上に仮想的な論理回路を再現し，外部入力線にテストパターンを印加することで，外部出力線の論理値を求めることを，**論理シミュレーション**という。

定義 6 故障シミュレーション

故障リストを用いて，被検査回路内の任意の信号線に故障を発生させると，故障時の回路の挙動について調べることが可能である．正常な回路と，故障が発生した回路の 2 つを用いて論理シミュレーションを実行し，得られた論理値を比較することを**故障シミュレーション**という．また，故障リストに含まれるすべての故障に対して故障シミュレーションを実行した結果，検出できた故障の割合を**故障検出率**という。

定義 7 2 パターンテスト

遅延故障は信号線の論理値の変化に起因するため，故障の検出には，変化前の論理値を設定するテストパターンと変化後の値を確かめる二つのテストパターンを連続して印加しなければならない．これを**2 パターンテスト**という。

定義 8 スキャンテスト

被検査回路の遅延故障に対する 2 パターンテストを与える手法としてスキャンテストが広く利用されている．本研究では，図 2.2 のように，順序回路を二つのタイムフレームに時刻展開した 2 時刻展開回路を利用して，代表的なスキャンテスト手法の一つである，LoC 方式 (Launch-off-Capture) を模擬する．LoC 方式のテストでは，2 時刻目の回路に対する入力にあたるフリップフロップの値は，1 時刻目の結果として取り込んだ論理値を利用する．取り込んだ信号値が 0 から 1 へ遷移していた場合は，2 時刻目のフリップフロップの値は立ち上がりとなる．逆に信号値が 1 から 0 へと遷移していた場合は，2 時刻目のフリップフロップの値は立下りとなる．これにより，遷移故障に対する 2 パターンテストを行うことが可能となる。

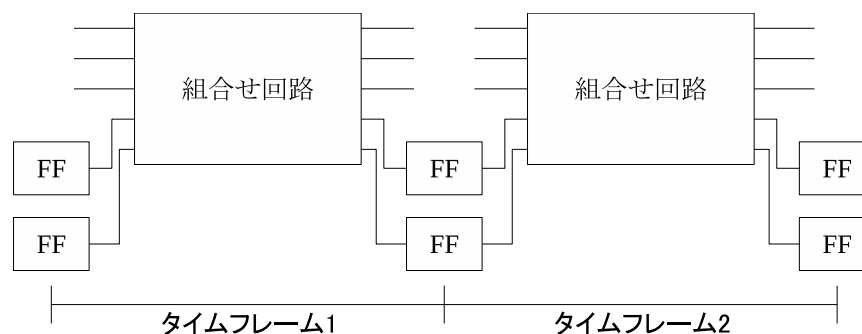


図 2.2. 2 時刻展開回路

定義 9 パス/フェイル情報

対象回路に対してテストを行い，得られた故障の検出 (パス)/非検出 (フェイル) の情報をまとめたものをパス/フェイル情報という。

定義 10 故障診断

テストによって不良と判定された集積回路について，集積回路内部の故障箇所の推定を行うことを故障診断という。故障診断は被検査回路に対してテストを行い，得られた出力値と被疑故障に関するシミュレーション結果とを比較することで行う。シミュレーション結果と被検査回路から得られた出力値が一致する場合，故障箇所の推定を行うことが可能となる。

定義 11 組み込み自己テスト (Built-in Self Test: BIST)

集積回路内部に，フィールドテスト環境を組み込んだ回路を組み込み自己テスト機構という。組み込むフィールドテスト環境は，「テストパターン生成回路」，「期待値と被検査回路からの出力値を比較する回路」の 2 つである。これらのテスト用回路を用いて，集積回路単体で故障の有無の判定を行う。

第 3 章

組込み自己診断

本章では，組込み自己診断 (Built-In Self Diagnosis: BIST) 機構について述べる．まず，BIST 機構について述べる．次に，BIST 機構の概要，基本動作，署名の作成・比較および診断能力についてそれぞれ説明する．

3.1 節 BIST 機構

図 3.1 に疑似ランダムパターンを使用した BIST 機構を示す [1]．BIST 機構の動作については以下の通りである．

入力側 テストパターン生成回路によって疑似ランダムパターンを生成後，被検査回路へと印加

出力側 テストパターンに対応するテスト応答を署名解析器で圧縮

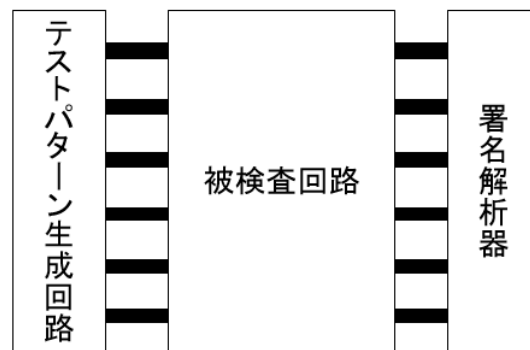


図 3.1. BIST 機構

3.2 節 BISR 機構

ここでは，BIST 機構を拡張した BISR 機構について説明する [4] [6]．まず，図 3.2 に BISR 機構を示す．BISR 機構は，以下の要素から構成される．

要素 1) ランダムパターン系列を生成するテストパターン生成回路

要素 2) テストパターン毎に被検査回路 (CUD) の故障検出情報であるパス/フェイル情報を生成する回路

要素 3) テストパターン毎のパス/フェイル情報を圧縮してテストパターン系列に対する診断用署名を解析する回路

要素 4) 被疑故障毎の被疑故障署名を格納するメモリ

要素 5) 診断用署名と被疑故障署名を比較し，故障箇所の推定を行う回路

次に，BISR 機構の基本動作について説明する．

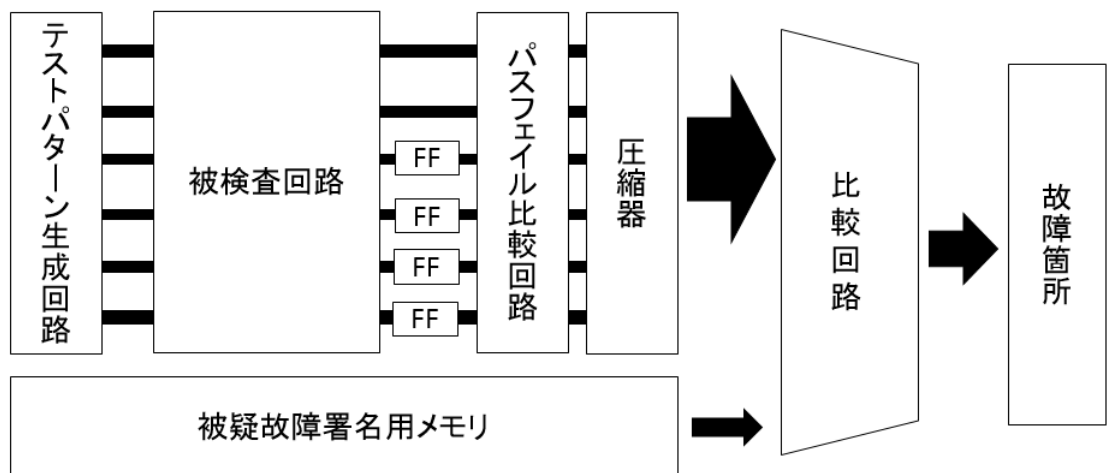


図 3.2. BISD 機構

BISD 機構の基本動作

- (動作 1) テスト生成回路およびリシード機構を利用して遷移故障診断能力の高いテストパターンを生成する。
- (動作 2) テストパターンに対して被診断回路のパス/フェイル情報を生成する。
- (動作 3) 診断用署名生成器を利用して、被診断回路のパス/フェイル情報を圧縮する。
- (動作 4) 所定のテストパターン数に対して、(動作 1) から (動作 3) を繰り返し実行して、被診断回路に対する診断用署名を生成する。
- (動作 5) 故障候補の指摘器を利用して、診断用署名と被疑故障署名を比較することで、情報が一致したアドレス情報から被疑故障を指摘する。

BISD では回路単体でテストパターンを発生させて故障診断を実行するため、外部テストの必要がなくコスト面での利点がある。また、フィールド上で故障箇所の推定が可能のため経年劣化にも対応することが可能である。

3.3 節 署名の生成・比較

ここでは，署名の作成及び比較について説明する [4]．BISD では，被検査回路から得られたパス/フェイル情報から診断用署名を生成するために，診断用署名解析器を利用する．生成された診断用署名と被疑故障署名を署名比較回路を利用して比較し，それらが一致するとき被診断回路の故障箇所を推定することが可能となる．比較に使用する被疑故障署名はあらかじめ故障シミュレーションを実行することで作成し，メモリ内に格納する．次に，被疑故障署名の生成法および診断用署名との比較法について述べる．被疑故障署名の生成および診断用署名との比較について図 3.3 に示す．

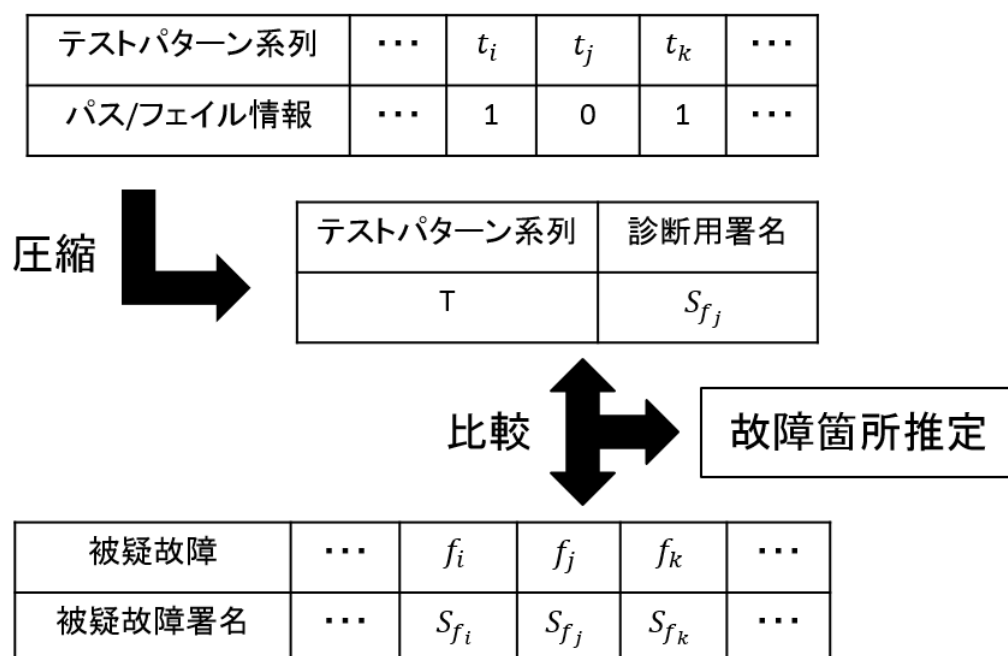


図 3.3. 署名作成・比較

故障診断においては、 I 個のテストパターンからなるテストパターン系列 $t_1, \dots, t_i, \dots, t_I$ が与えられる。また、故障リストは J 個の仮定故障からなるものとする。

まず、仮定故障 f_j に対応する被疑故障署名 S_{f_j} を求めるために、故障シミュレーションを実行する。故障シミュレーションによってテストパターン t_i における故障 f_j をもつ回路のパス/フェイル情報 $pf_{ji} = PF(f_j, t_i)$ を求める。 t_i が遷移故障 f_j を検出するなら $pf_{ji} = 1$ 、検出しないなら $pf_{ji} = 0$ とする。次に、 I 個のテストパターンからなるテストパターン系列に対して求めたパス/フェイル情報から、被疑故障署名 $S_{f_j} = SA(pf_{j1}, \dots, pf_{ji}, \dots, pf_{jI})$ を求める。同様の処理を繰り返すことで、 J 個の仮定故障に対して J 個の被疑故障署名 $S_{f1}, \dots, S_{fj}, \dots, S_{fJ}$ を求める。求められた J 個の被疑故障署名 $S_{f1}, \dots, S_{fj}, \dots, S_{fJ}$ はメモリに格納されているとする。

次に、テストパターン毎の被診断回路のパス/フェイル情報 $pf_i = PF(CUD, t_i)$ を求める。 t_i が被診断回路の故障を検出するならば $pf_i = 1$ 、検出しないならば $pf_i = 0$ とする。 I 個のテストパターン系列のパス/フェイル情報から診断用署名 $S_{CUD} = SA(pf_1, \dots, pf_i, \dots, pf_I)$ を生成する。

最後に、故障候補を指摘するために、 S_{CUD} と $S_{f1}, \dots, S_{fj}, \dots, S_{fJ}$ を順次比較する。もし、 $S_{CUD} = S_{fj}$ ならば、被診断回路に遷移故障 f_j が存在すると推定する。

3.4 節 BISD 機構の診断能力

ここでは、BISD 機構の診断能力について説明する [4]。テストパターン系列において J 個の故障を仮定し、それらに対してそれぞれ J 個の被疑故障署名が与えられるとする。このとき、ある故障 f_j に対する被疑故障署名 S_{fj} がその他 $J - 1$ 個の被疑故障署名と相異なるならば、テストパターン系列によって f_j は診断可能である。診断可能とは、与えられたテストパターン系列において、ほかの故障と区別が可能であることを指す。一方で、 S_{fj} と同一の被疑故障署名がその他 $J - 1$ 個の被疑故障署名にあるならば、 f_j は区別できない故障となり診断不可能となる。このとき、区別できない故障は 1 つにまとめることでクラスと表す。クラスの作成についての例を図 3.4 に示す。

	f_1	f_2	f_3	f_4
t_1	0	1	0	0
t_2	1	0	1	1
t_3	1	0	0	0
t_4	1	0	0	0
応答署名	S_1	S_2	S_3	S_3

図 3.4. クラスの作成

BISD 機構の故障診断能力は、「検出可能な故障数」、「診断可能な故障数」、「クラス数」、および「クラスの要素数」によって評価する。検出可能な故障数とは、 J 個の故障に対して出力まで伝搬したものを指す。診断可能な故障数が多いほど、故障診断能力が高いと評価する。クラス数は、区別できない故障をまとめたクラスの総数を指す。また、クラスに含まれる被疑故障数が少ないほど故障箇所の推定が容易となるため、要素数が少ないクラスが多いほど故障診断能力が高いと評価する。クラスの要素数は、クラス内の最大値および平均値によって評価を行う。

第 4 章

誤り出力数を考慮した組込み自己診断用 テスト生成法

本章では，誤り出力数を考慮した組込み自己診断テスト生成法について述べる．まず，提案する組込み自己診断用テスト生成法の概要について説明する．次に，テスト生成法のシミュレーションモデルについて説明する．さらに，故障辞書とテスト生成法のアルゴリズムについてそれぞれ説明する．

4.1 節 誤り出力数を考慮した組込み自己診断用テスト生成法

ここでは，誤り出力数を考慮した組込み自己診断テスト生成法の概要について説明する．誤り出力数を考慮した組込み自己診断テスト生成法とは，故障診断を行った際ランダムパターンと比較して，誤りを検出するフリップフロップ数が減少するテストパターンを生成することである．

図 4.1 にランダムパターンを使用する場合の診断用回路図を示す．BISD 機構に用いられる診断用回路では，故障検出の有無を用いて診断用署名の生成を行う．しかしながら，回路規模が増加することにより署名生成用回路の規模も増加してしまいハードウェアオーバーヘッド増加の問題が挙げられる．この問題に対して先行研究では，診断用署名生成に用いるフリップフロップ数の削減を行う手法により改善を図った [5]．これ

はフリップフロップに信号線の接続数を用いてランク付けを行い，ランクの上位順で指定した割合のフリップフロップのみを診断用署名生成に使用するという手法である．これに対して本研究では，入力側であるテストパターンを変更することにより，出力側で故障を検出するフリップフロップ数の削減を行った．提案手法によって生成したテストパターンを使用する場合の診断用回路を図4.2に示す．

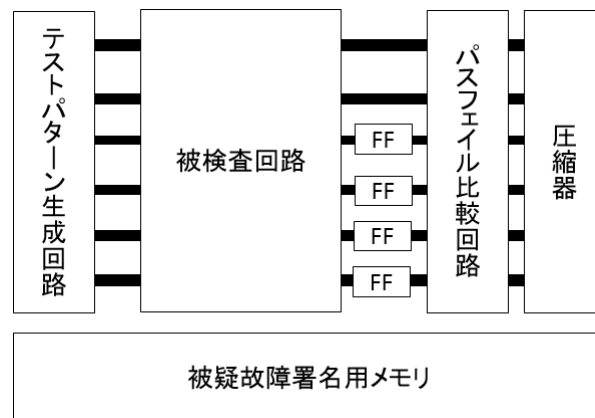


図 4.1. 変更前のテストパターン使用時の回路

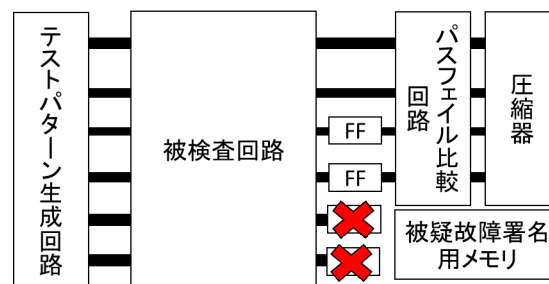


図 4.2. 変更後のテストパターン使用時の回路

4.2 節 シミュレーションモデル

ここでは、提案するテスト生成法におけるシミュレーションモデルについて説明する。まず、提案するテスト生成法におけるシミュレーションモデルを図4.3に示す。

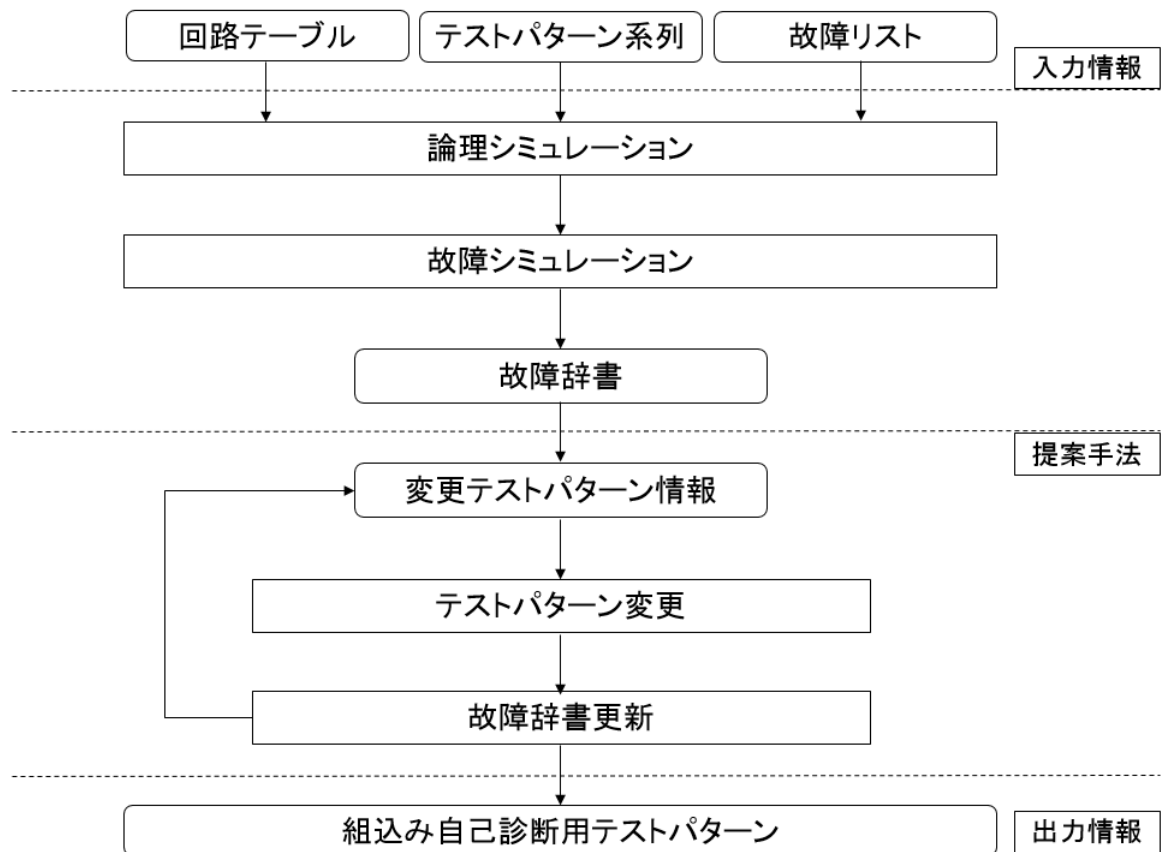


図 4.3. テスト生成シミュレーションモデル

被検査回路に対して故障診断を行うために、次の3つの情報が与えられる。

情報 1) I 個のテストパターンからなるテストパターン系列 $t_1, \dots, t_i, \dots, t_I$

情報 2) 回路テーブル

情報 3) J 個の仮定故障からなる故障リスト

まず、誤りを検出したフリップフロップ数を求めるために与えられた3つの情報を用いて、論理シミュレーションおよび故障シミュレーションを実行する。シミュレーション結果を用いて、誤りを検出したフリップフロップ数をまとめた故障辞書を作成する。次に、作成した故障辞書を用いて変更するテストパターンを決定する。この詳細は4.2.1.で述べる。次に、変更対象となったテストパターンに対して変更操作を実行する。この詳細は4.2.2.で述べる。変更対象となったすべてのテストパターンに対して変更操作を実行した後、誤りを検出したフリップフロップの最大値を更新する。更新した後、再び変更するテストパターンの決定へと戻る。この操作を故障を検出したフリップフロップの最大値の更新が停止するまで実行し、停止した後に求められたテストパターンが、故障診断用テストパターンとなる。

4.2.1. 故障辞書

ここでは提案手法において作成する故障辞書についての説明を行う。まず被検査回路に対して入力された3つの情報を用いて論理シミュレーション、故障シミュレーションを実行する。得られた結果から、それぞれのテストパターン、故障リストで誤りを検出したフリップフロップの数を求める。これらを記録したものを故障辞書と呼ぶ。図4.4に、作成する故障辞書の例を示す。

	f_1	f_2	f_3
t_1	6	0	5
t_2	10	3	4
t_3	2	0	4

図 4.4. 故障辞書

本研究で提案する診断用テストパターン生成法では、誤りを検出したフリップフロップの最大値を減少させることを目的としている。したがって、テストパターン系列の中から最大値を含むものを変更するテストパターンとして決定する。まず、作成した

故障辞書から誤りを検出したフリップフロップ数の最大値を求める．次に，得られた最大値を持つテストパターンをリストにまとめる．これにより，テストパターン系列の中から変更対象となるテストパターンのみを選択することができる．

4.2.2. 診断用テスト生成法のアルゴリズム

ここでは，診断用テスト生成法のアルゴリズムについて説明する．診断用テスト生成を行うために，次の 2 つの情報が与えられる．

情報 1) 変更対象となった I 個のテストパターン $t_1, \dots, t_i, \dots, t_I$

情報 2) テストパターンのビット幅 J

診断用テスト生成法のアルゴリズムは以下の通りである．

診断用テスト生成法のアルゴリズム

(手順 1) 故障辞書より変更対象となったテストパターンをリスト化．

(手順 2) i および j を 1 に決定．

(手順 3) テストパターン t_i を変更対象として決定．

(手順 4) (手順 1) で選択したテストパターンにおけるビット j を変更対象として決定し，ビットを反転させる．

(手順 5) 論理シミュレーションおよび故障シミュレーションの実行

(手順 6) 制約条件を満足しない場合ビット j を変更前に戻す．

(手順 7) j に 1 加算し (手順 3) に戻る． j が J に達するまで繰り返す．

(手順 8) i に 1 加算し (手順 2) に戻る． i が I に達するまで繰り返す．

(手順 9) 誤りを検出したフリップフロップを再計算して故障辞書を更新後，(手順 1) に戻る．最大値が更新されなくなるまで繰り返す．

まず，変更対象となるテストパターンを選択する．変更対象となるテストパターンは，故障辞書によりリスト化されたテストパターンの中から順に選択する．次に選択されたテストパターンの先頭1ビットに対して反転操作を実行する．対象となるビットが1である場合は0に，0である場合は1に変更する．

本研究では誤りを検出するフリップフロップの数を削減することを目的としている．したがって変更前後におけるテストパターンで，故障が伝搬しているフリップフロップが減少している必要がある．変更前のテストパターンを用いた故障の伝搬の様子を図4.5に示す．また，変更後のテストパターンを用いた故障の伝搬の様子を図4.6に示す．

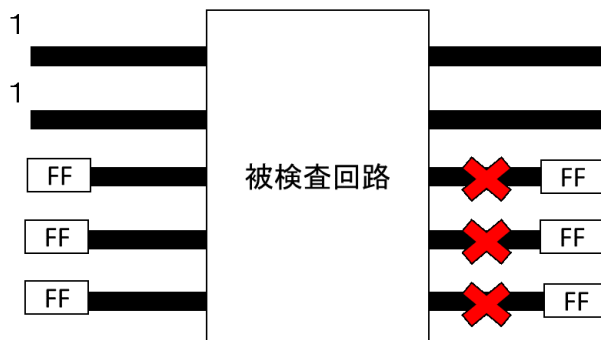


図 4.5. 変更前のテストパターンを用いた故障検出

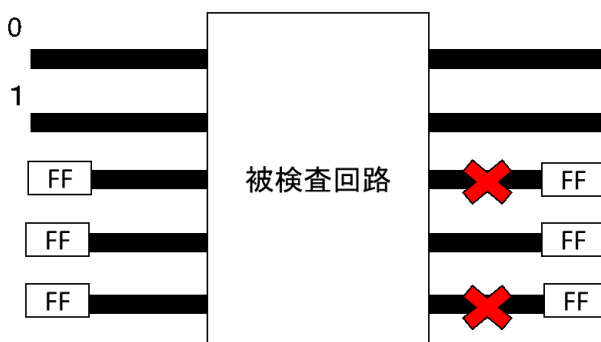


図 4.6. 変更後のテストパターンを用いた故障検出

本研究ではテスト生成の際，故障検出率を維持しなければならない．したがってビット反転において，以下の制約条件を設定した．

制約条件

条件 1) 故障検出率を維持または向上

条件 2) 被検査回路において誤りを検出したフリップフロップ数の低下

制約条件を満たさない場合，反転したビットを元の状態に戻す．この反転操作を，先頭のビットから順に 1 ビットずつ実行する．また，リスト化されたすべてのテストパターンに対して同様の操作を実行する．リスト化されたすべてのテストパターンに対して全ビットの反転操作が終了すると，次に故障辞書の更新を行う．誤りを検出したフリップフロップ数の最大値を求め，再度最大値を持つテストパターンをリスト化しビット反転を実行する．なお，最大値がテスト生成法を適用する前後で変化が無くなった場合，これまでに変更したテストパターンを出力する．出力されたテストパターンが診断用テストパターンとなる．

第 5 章

評価実験

本章では，本研究で提案したテスト生成法によって生成されたテストパターンを用いて，故障検出したフリップフロップ数および故障診断能力の精度を評価する．また，本実験で使用した計算機の諸元を表 5.1 に示す．

表 5.1. 実験に利用した計算機の諸元

CPU	Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz
Memory	16.0GB
OS	Ubuntu 18.04.1 LTS

提案手法を評価するために，提案手法を C 言語を用いて実装し評価実験を行った．評価実験の対象とする回路は，ISCAS'89 ベンチマーク回路を使用した．テストパターンはリシード法により生成された 2,048 パターンを使用した．

5.1 節 誤り出力数を考慮した組込み自己診断用テスト生成法

本節では，第 4 章で述べた誤り出力数を考慮した組込み自己診断用テスト生成法における評価実験結果を示す．提案したテスト生成法により生成された診断用テストパターンにおける誤りを出力するフリップフロップ数を評価した．

表 5.2 から表 5.4 に，実験結果を示す．診断用テストパターンの有用性は以下の指標によって評価する．

- 検出可能な故障数
- 誤りを検出したフリップフロップの総数
- 誤りを検出したフリップフロップの最大数

また，提案手法により生成されたテストパターンとの比較対象として，リシード法により生成されたテストパターンでの実験結果を示す．

表 5.2. 提案手法における診断用テストパターンの評価結果 (cs9234)

使用した テストパターン	検出可能な 故障数	誤りを検出した フリップフロップの総数	誤りを検出した フリップフロップの最大値
変更前	7501	5225310	19
変更後	7501	5192515	16

表 5.3. 提案手法における診断用テストパターンの評価結果 (cs13207)

使用した テストパターン	検出可能な 故障数	誤りを検出した フリップフロップの総数	誤りを検出した フリップフロップの最大値
変更前	10638	11316003	19
変更後	10665	11298108	15

表 5.4. 提案手法における診断用テストパターンの評価結果 (cs15850)

使用した テストパターン	検出可能な 故障数	誤りを検出した フリップフロップの総数	誤りを検出した フリップフロップの最大値
変更前	12569	10326089	43
変更後	12646	10100606	29

さらに，提案手法により生成されたテストパターンを用いて故障診断を実行し，評価を行った．

表 5.5 から表 5.7 に，実験結果を示す．診断用テストパターンの有用性は以下の指標によって評価する．

- 対象故障数
- 診断可能な故障数
- クラス数
- クラス内の最大故障数
- 各クラスの平均故障数

表 5.5. 故障診断結果 (cs9234)

使用した テストパターン	対象 故障数	診断可能な 故障数	クラス数	クラス内の 最大故障数	各クラスの 平均故障数
変更前	7501	678	1390	52	4.91
変更後	7501	673	1379	52	4.89

表 5.6. 故障診断結果 (cs13207)

使用した テストパターン	対象 故障数	診断可能な 故障数	クラス数	クラス内の 最大故障数	各クラスの 平均故障数
変更前	10638	935	1583	86	6.13
変更後	10665	948	1599	86	6.08

表 5.7. 故障診断結果 (cs15850)

使用した テストパターン	対象 故障数	診断可能な 故障数	クラス数	クラス内の 最大故障数	各クラスの 平均故障数
変更前	12569	1207	2204	69	5.16
変更後	12646	1220	2222	69	5.14

評価実験結果より以下のことが明らかになった。

- 提案手法によって生成されたテストパターンを使用すると，誤りを検出したフリップフロップの総数および最大値が減少した。
- 提案手法によって生成されたテストパターンで故障診断を実行した際，リシード法で生成されたテストパターンと比較して各クラスにおける平均故障数が低下した。これにより，各クラスに対して候補となる故障数が減少するため故障診断能力が向上した。
- 回路規模が増加した場合は診断可能な故障数，検出可能な故障数ともに上昇することが確認できた。

第 6 章

結論

近年、半導体製造技術の微細化に伴って集積回路の高速化・高集積化が実現されている。集積回路は様々な機器に広く利用されており、我々の生活に欠かせない存在となっている。特に、自動車に搭載される車載システムなどでは、高度な機能安全性を実現するために集積回路の高信頼化が必要とされている。システムを安全に利用するための技術として、フィールド上で集積回路が正常かどうかを判断する組込み自己テスト (Built-in Self Test: BIST) が存在するが、BIST では故障の有無しか判断できない。そのため、筆者らは故障の箇所を推定が可能な組込み自己診断 (Built-in Self Diagnosis: BISD) を提案している。しかしながら、集積回路の規模が増加している中ですべてのフリップフロップからの情報を用いて故障診断を行うことは、ハードウェア制約として問題となる。

そこで、本研究では、組込み自己診断用のテストパターンを生成する手法を提案した。提案手法では、誤りを検出するフリップフロップ数を減少させることによって、故障診断に用いるフリップフロップ数を削減することとした。そのためのテストパターン生成手法として、テストパターン生成回路により生成したパターンのビット反転を行った。また、生成したテストパターンの評価および、生成したテストパターンを使用した故障診断の評価をそれぞれ行った。

実験結果から、すべての回路で誤りを検出するフリップフロップ総数および最大値を減少させることに成功したため、有効性を示すことができた。また、生成したテストパターンを使用した場合の故障診断の評価として、診断用に変換する前のテストパ

ターンと比較すると同等の性能を持っていることが確認できた．さらに，回路規模が増加した場合や誤りを検出したフリップフロップ数が多い場合は，診断可能な故障の数が増加することや最大値の減少幅が増加することからより大きな効果を得られることが確認できた．

しかしながら，今後の課題として，誤りを検出するフリップフロップ数をさらに減少するためのビットの選択法を考察する必要がある．また，実際のハードウェアにおける面積の削減率等を求めるため，ハードウェア設計についても考察する必要がある．

謝辞

本研究を進めるにあたり，懇篤な御指導，御鞭撻を賜りました本学高橋寛教授に深く御礼申し上げます。

本論文の作成に関し，詳細なるご検討，貴重な御教示を頂き，また審査頂いた本学王森レイ講師に深く御礼申し上げます。

本論文の作成に関し，詳細なるご検討，貴重な御教示を頂きました本学樋上喜信教授に深く御礼申し上げます。

また，審査頂いた本学二宮崇教授に深く御礼申し上げます。

最後に，多大な御協力と貴重な御助言を頂いた本学情報システム工学講座計算機システム研究室の諸氏に厚く感謝致します。

参考文献

- [1] M. Elm, H. Wunderlich, "BISD:Scan-Based Built-In Self-Diagnosis, " Proc. DATE, pp.1243-1248, 2010.
- [2] A. Cook et.al., "Advanced diagnosis: SBST and BIST integration in automotive E/E architectures, " Proc. DAC, pp.1-6, DOI:10.1145/2593069.2602971, 2014.
- [3] A. Cook, M. Elm, H. Wunderlich, and U. Abelein, "Structural in-field diagnosis for random logic circuits, " Proc. ETS, pp.111-116, DOI:10.1109/ETS.2011.25, 2011.
- [4] 宮本 夏規, 村上 陽紀, 王 シンレイ, 樋上 喜信, 高橋 寛, 大竹 哲史, "組込み自己診断におけるテストパターン系列の診断能力に関して," 情報科学技術フォーラム, pp.273-274, 2015.
- [5] 矢野 郁也, 王 森レイ, 樋上 喜信, 高橋 寛, "組込み自己診断におけるハードウェア制約の改善法," 平成 28 年度電気関係学会四国支部連合大会, pp.101, 2016.
- [6] 高橋 寛, 王 森レイ, 樋上 喜信, 松田 優大, "組込み自己診断向けのテストパターン生成法," 平成 29 年度 電気関係学会四国支部連合大会, 2017.
- [7] 藤田昌宏, "システム LSI 設計工学," 株式会社オーム社, 2006.
- [8] 藤原秀雄, "コンピュータの設計とテスト," 工学図書, 1990.

発表論文リスト

- (1) 高橋 寛, 王 森レイ, 樋上 喜信, 松田 優大, ”組込み自己診断向けのテストパターン生成法,” 平成 29 年度 電気関係学会四国支部連合大会, 2017.