

6 群(コンピュータ 基礎理論とハードウェア) - 7 編(ディペンダブルコンピューティング)

3 章 設計検証 / テスト

(執筆者: 福本 聡)[2010 年 10 月受領]

概要

現在の半導体に望まれる信頼性は、許容されるコスト的・時間的制約の下で、市場の要求に見合った品質の保証を果たすことで達成される。この品質の保証には、設計品質の保証と製造品質の保証とがある。前者は、半導体の設計が正しく仕様を満たしているかどうかの審査によって、また後者は、作られた半導体が設計どおりに動作するかどうかの審査によって、それぞれに実現される。これらの品質保証は、コストと性能のバランスに優れた多くの実用的システムのディペンダビリティ向上に貢献してきた。また一方、半導体は、発展を続ける情報ネットワーク社会の安全・安心を支える基幹要素であり、今後、要求される信頼性の水準はこれまで以上に高いものになると考えられる。例えば、従来の設計製造段階では検出し得ない故障、あるいはフィールドでの想定外の故障等に耐える特性や、システムへの悪意ある攻撃に対して半導体レベルで対応するなどの特性が必要になりつつある。こうした特性を半導体に備えるために、冗長化に代表されるフォールトトレランス技術や、既にソフトウェアで実績のあるセキュリティ技術などの適用が期待されており、実際に様々なアプローチで多くの研究が行われている。

本章では、これまでに広く普及している技術や現在実用化が進められている技術に焦点を当て、上記の設計品質及び製造品質の保証に関連する代表的な技術について論ずる。半導体の設計において、正しい動作が得られることを検査・保証する具体的な技術としては、各ハードウェアレベルでのシミュレーションが明らかに有効であるが、それによってすべての挙動を網羅的に確認することは困難である。ここでは、システムの仕様を数理論理学手法で形式的に表現するフォーマルアプローチについて解説する。また、製造工程での信頼性を保証する技術として、半導体テスト技術について述べる。具体的には、ロジック LSI を対象とするテスト方式やテスト容易化設計、メモリのテスト手法や耐故障技術、アナログ/ミックスドシグナル系テストなどについて論ずる。なお、広義のテスト技術に含まれる、故障解析/診断技術についても取り扱う。

【本章の構成】

本章では、ハードウェア検証(3-1 節)、故障モデル(3-2 節)、テスト方式(3-3 節)、テスト生成(3-4 節)、テスト容易化設計(3-5 節)、システムオンチップのテスト(3-6 節)、メモリのテスト/耐故障技術(3-7 節)、アナログ/ミックスドシグナル系テスト(3-8 節)、故障解析/故障診断(3-9 節)の各技術について述べる。

6 群 - 7 編 - 3 章

3-1 ハードウェア検証

(執筆者：浜口清治)[2009年4月受領]

本節ではハードウェア検証のなかで、近年注目されているフォーマル検証手法について解説する。ここで取り上げている手法の多くは、原理的にはソフトウェアにも適用可能なものであり、特にモデル検査については、近年実用プログラムへの適用が盛んに試みられている。

3-1-1 等価性検証

ここでの等価性検証は、二つの論理回路が実現している関数が等価であるかどうかを判定することをいう。具体的にはパフォーマンス、消費電力などを改善するために人手であるいはソフトウェアによって改変された回路と、改変前の回路を比較するために用いられ、ハードウェアに対するフォーマル検証でもっともよく利用されている。

組み合わせ回路に対する等価性判定 (equivalence checking) 問題は基本的には論理式に対する充足可能性判定 (satisfiability, SAT) (問題変数への値の割当てのうち、論理式を 1 とする割当てが存在するかどうかを判定する問題) に帰着することができる。例えば、図 3-1 の二つの点線内の回路の等価性判定問題は、次の論理式に対する充足可能性判定問題に等しい。この論理式が充足不能 (unsatisfiable) となる (論理式を 1 とする割当てが存在しない) とき等価となる。

$$\begin{aligned} &(a \equiv \neg x) \wedge (d \equiv a \vee c) \wedge (c \equiv y \wedge b) \wedge (b \equiv \neg z) \wedge \\ &(p \equiv \neg z) \wedge (q \equiv p \vee z) \wedge (r \equiv x \wedge q) \wedge (s \equiv \neg r) \wedge \\ &(w \equiv d \oplus s) \end{aligned}$$

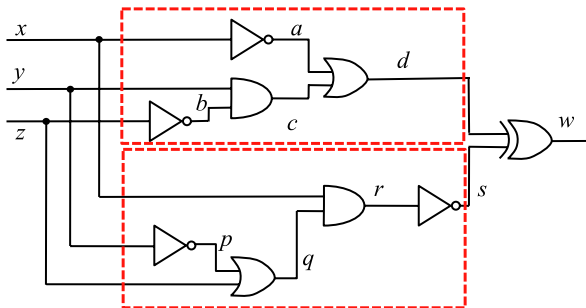


図 3-1 組み合わせ回路の等価性判定

この問題は順序付き二分決定グラフ (Ordered Binary Decision Diagram, OBDD) や充足可能性判定ツール (あるいは SAT ソルバ (SAT solver) と呼ばれる) を用いて解くことができるが、直接的な方法で大規模な回路を直接扱うことは難しい。実際には、回路中の等価点を SAT ソルバなどを用いて入力側から順次発見・併合することによって、回路を縮退させて計算量の軽減化を図っている。詳細及び順序回路に対する等価性判定については、文献 1)などを参照されたい。

3-1-2 モデル検査

モデル検査 (model checking)^{2,3)}は、有限状態機械と検証すべき性質 (プロパティと呼ぶ) を与えて、有限状態機械の動作がそのプロパティを満足するかどうかを調べる技術である。

(1) 時相論理とオートマトン

モデル検査におけるプロパティは時相論理 (temporal logic) または有限オートマトンで与えられる。これらはまた近年ハードウェア検証で用いられるようになってきている標準アサーション言語 Property Specification Language (PSL) や SystemVerilog Assertion (SVA)⁴⁾の基礎ともなっている。

時相論理は通常の論理体系に時間的な関係を表現するための演算子を加えて拡張した体系である。ここでは命題論理を拡張した体系である命題時相論理について説明する。命題時相論理には種々の体系がある。代表的なものとして分岐時間モデルに基づく計算木時相論理 (Computational Tree Logic, CTL) と線形時間モデルに基づく線形時間時相論理 (Linear-time Temporal Logic, LTL) がある。これらの時相論理式の真偽は、クリプキ構造 (Kripke structure) に対して決定される。クリプキ構造は順序機械など有限状態機械のモデルとみなすことができる。順序回路のモデル検査を行う場合は、クリプキ構造への変換または対応づけを行うことになる。

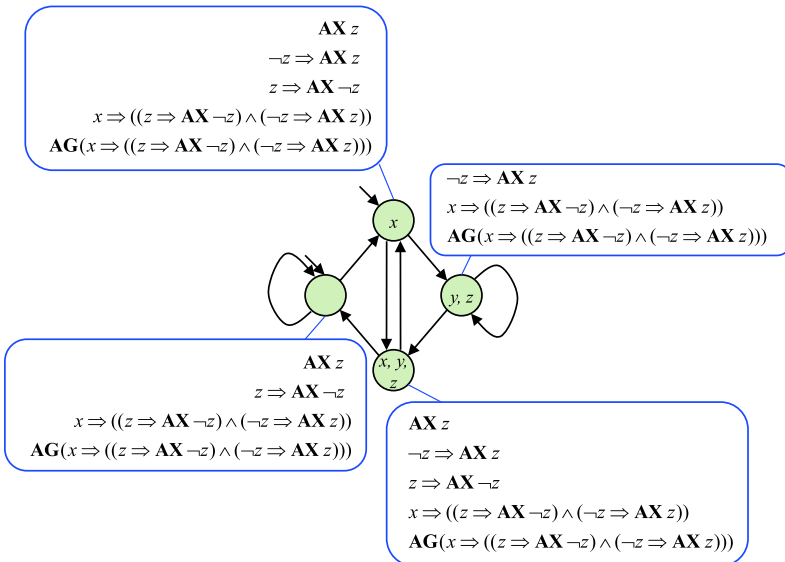


図 3-2 クリプキ構造及びモデル検査アルゴリズムの動作例

以下では、原始命題の集合を P とする。クリプキ構造 $M = (S, S_0, R, L)$ は、状態の有限集合 S 、初期状態の集合 $S_0 \subseteq S$ 、遷移関係 $R \subseteq S \times S$ 、原始命題の集合を各状態にラベルする関数 $L: S \rightarrow 2^P$ からなる。 L は各状態で真となる原始命題を指定する。また、 R は任意の状

態において、必ず一つ以上の遷移先の状態をもたなければならない（遷移先は自分自身でもよい）。クリプキ構造の例を図 3・2 に示す。始点に状態がない矢印が初期状態を示している。各状態に書き込まれている x, y, z は各状態で真となる原始命題を表している。書き込まれていない原始命題は偽である。

CTL 式の典型的な演算子の意味を図 3・3 にまとめる。「すべての系列」を意味する演算子 A のみの場合を示しているが、「ある系列」を意味する演算子 E もあり、 $Ef = \neg A\neg f$ の関係がある。ここで、 p は原始命題、 f, g は CTL 式である。 $M, s \models f$ は「クリプキ構造 M 上の状態 s で CTL 式 f が真である（あるいは、成立する）」ことを意味している。CTL では、 s から始まる M 上の無限の系列について式の真偽が定義される。モデル検査問題は、与えられた M と f について、すべての初期状態 $s_0 \in S_0$ について $M, s_0 \models f$ かどうかを判定する問題である。なお、 $AXg = \neg EX\neg g$, $EFg = E[trueUg]$, $AGg = \neg EF\neg g$, $AFg = \neg EG\neg g$, $A[gUh] = \neg E[\neg hU(\neg g \wedge \neg h)] \wedge \neg EG\neg h$ の関係がそれぞれ成立することから、 EX, EG, EU のみで CTL を定義できることが分かる。

- $M, s \models p \Leftrightarrow p \in L(s)$.
- $M, s \models AXf \Leftrightarrow s$ から始まる M 中の任意の系列を考えたとき、 s の次状態 s' に対して、 $M, s' \models f$.
- $M, s \models AGf \Leftrightarrow s$ から始まる M 中の任意の系列を考えたとき、系列上のすべての状態 s' について、 $M, s' \models f$.
- $M, s \models AFf \Leftrightarrow s$ から始まる M 中の任意の系列を考えたとき、系列上のある状態 s' について、 $M, s' \models f$.
- $M, s \models A[fUg] \Leftrightarrow s$ から始まる M 中の任意の系列 π を考えたとき、最初の状態 s について $M, s \models g$ であるか、 π 上のある状態 s' の手前までの s'' に対しては、 $M, s'' \models f$ かつ $M, s' \models g$.

図 3・3 CTL 式の意味

- $M, \pi \models p \Leftrightarrow \pi$ の最初の状態 s に対して、 $p \in L(s)$.
- $M, \pi \models Xf \Leftrightarrow M, \pi^1 \models f$.
- $M, \pi \models Gf \Leftrightarrow$ すべての $i (i = 0, 1, \dots)$ に対して、 $M, \pi^i \models f$.
- $M, \pi \models Ff \Leftrightarrow$ ある $i (i = 0, 1, \dots)$ に対して、 $M, \pi^i \models f$.
- $M, \pi \models [fUg] \Leftrightarrow \pi$ $M, \pi \models g$ であるか、ある $j (j > 0)$ に対して、 $M, \pi^j \models g$ かつ $i = 0, 1, 2, \dots, j-1$ に対して、 $M, \pi^i \models f$.

図 3・4 LTL 式の意味

次に、LTL 式の意味を図 3・4 に示す。ここで p は原始命題、 f, g は LTL 式である。 π はクリプキ構造上の無限系列であり、 $\pi^i (i = 0, 1, \dots)$ は π の i 番目の状態以降の系列を表す（したがって $\pi = \pi^0$ となる）。 $M, \pi \models f$ は「クリプキ構造 M 上の系列 π で LTL 式 f が真である（あるいは、成立する）」ことを意味している。LTL 式 f の M 上の状態 s における真偽は、次のように定義される。モデル検査問題の定義は CTL の場合と同様となる。

$$M, s \models f \Leftrightarrow s \text{ から始まるすべての系列 } \pi \text{ に対して } M, \pi \models f$$

プロパティの記述には、時相論理式の代わりに有限オートマトンを用いることができる。有限オートマトン $A = \langle \Sigma, Q, \Delta, Q_0, F \rangle$ は、アルファベット（記号の有限集合） Σ 、状態の有限集合 Q 、初期状態の集合 $Q_0 \subseteq Q$ 、遷移関係 $\Delta \subseteq Q \times \Sigma \times Q$ 、受理状態の集合 $F \subseteq Q$ となる。時相論理と対応させるため、以下では、 Σ の要素を無限に並べた系列 σ を入力した場合の受理 / 非受理について定義する。この場合、 $\Sigma = 2^P$ とする。無限長の入力系列を仮定する有限オートマトンは、 ω 有限オートマトンと呼ばれる。

長さ 1 の入力に対する A の動作は通常のオートマトンと同様である。 σ を入力すると A 上で状態遷移が起こり、状態の遷移列が得られる。これを ρ_σ と書く。 ρ_σ は Q の要素の無限の系列であり、一般に、 A が非決定性をもつため複数存在し得る。 σ が受理されるのは、「ある ρ_σ が存在して、 ρ_σ 中に無限回出現する状態の集合 $\text{inf}(\rho_\sigma)$ と F の間に、 $\text{inf}(\rho_\sigma) \cap F \neq \emptyset$ の関係があること」と定義される。これは Büchi 型の受理条件と呼ばれる（ほかに Rabin 型、Streete 型などの受理条件がある）。 ω 有限オートマトン A をプロパティ記述に使った場合、 $M, s \models A$ は s から始まる M 上のすべての無限の系列が、 A に受理されることを意味する。モデル検査問題は CTL や LTL の場合と同様に定義される。

(2) モデル検査アルゴリズム

まず、CTL に対するアルゴリズムを説明する（詳細は文献 2）を参照されたい）。このアルゴリズムは CTL 式の部分式を、原始命題から順に処理して、最終的に式全体に対するモデル検査を行う。各部分式 f の各状態での真偽は、 f に含まれる真部分式が各状態で真または偽であるかどうか判定した結果を利用して決定される。このアルゴリズムの計算時間はクリプキ構造の状態数と CTL 式の長さの積に比例する。図 3-2 の各状態にラベルされた CTL 式は、次の時相論理式に対して、モデル検査アルゴリズムを適用した結果である。

$$\mathbf{AG}(x \rightarrow ((z \rightarrow \mathbf{AX}\neg z) \wedge (\neg z \rightarrow \mathbf{AX}z)))$$

LTL 式の場合は、式の構造からクリプキ構造に類似した有限状態機械（タブローと呼ばれる）を、また有限オートマトンの場合は決定化操作を行ったオートマトンを準備し、クリプキ構造との間で、ある種の直積グラフを構成してモデル検査を行う。いずれの場合も、クリプキ構造の状態数には線形であるが、式の長さまたはオートマトンの状態数に対しては、最悪指数的な時間計算量を要する。

与えられた時相論理式が成り立たない場合は、反例となる系列を提示できる場合がある。例えば、 $\mathbf{AG}(p)$ が成立しない場合は、 p が成立しないような系列一つを見つけて反例として提示することができる。一方、 $\mathbf{EF}(p)$ が成立しない場合は、すべての系列を示す必要があることから現実的には実現できない（ $\mathbf{EF}(p)$ が成立する系列を示すことはできる）。

一般に、 n 個の状態保持要素（フリップフロップなど）を含むシステムでは、その状態数は n に対して指数的な大きさとなる。この現象は状態爆発（state explosion）と呼ばれる。このため、クリプキ構造を明示的にかたちで保持するアルゴリズムでは、取り扱うことができる設計は小規模に限られる。記号モデル検査アルゴリズムは、非明示的な形式で状態集合を保持することにより、状態数爆発を緩和する。具体的には、状態集合 $A \subseteq S$ に対して、 $s \in A$ であるときそのときに限り $F_A(s) = 1$ となる関数（特性関数または特徴関数）によって A を表現する。 s を l ビットの 2 進数で表現すれば、 $F_A : \{0, 1\}^l \rightarrow \{0, 1\}$ は通常の論理関数となる。

具体的には、この特徴関数を使って状態集合を表現して、集合に対する操作は特徴関数に対

する論理演算によって行う．これには，遷移関係 R を表す特性関数，すなわち $F_R(s, s') = 1 \Leftrightarrow (s, s') \in E$ なる F_R を用いる．例えば，CTL 式 g が成立する状態集合を表す特性関数 F_g が計算済みであれば， EFg が成立する状態集合を表す特性関数を計算するには， $N_0 := F_g$ を初期状態として， $N_{i+1}(s) := (\exists s'. F_R(s, s') \wedge N_i(s')) \vee N_i(s)$ を $N_{i+1} = N_i$ となるまで繰り返す．一般に CTL 式 f をモデル検査する場合は， F_f を計算し， $F_{S_0} \Rightarrow F_f$ が恒真であれば， f は初期状態の集合 S_0 において成立することとなる．

特性関数の表現及びその上での論理演算には，二分決定グラフを利用する手法が広く用いられてきた．しかし，近年になって，二分決定グラフを用いずに，充足可能性判定アルゴリズムを利用して，モデル検査を行うアルゴリズムが開発されている．これについては，以下の (3) 及び (4) で述べる．

(3) 有界モデル検査

有界モデル検査 (bounded model checking)⁵⁾ は与えられた正整数 k に対して，初期状態から k 個分の遷移 (k サイクル分) の範囲に限って，モデル検査を行う手法である．状態遷移関係 R を表現する論理式を k サイクル分時間展開した論理式を準備して，この範囲で与えられたプロパティを偽とする系列が含まれていないかどうかを調べる．これには，通常，論理関数に対する充足可能性判定 (SAT) を用いる (図 3-5)．対象としている範囲内に含まれている巡回経路も考慮する場合がある．

充足可能性判定は，和積標準形 (conjunctive normal form, CNF) の論理式に対して行われることが多いが，論理回路を対象とする場合には，回路構造を反映したアンド・インバータ・グラフ (and-inverter graph) など用いられる⁶⁾．

例として，セーフティ・プロパティのモデル検査，つまり，初期状態から到達可能な状態すべてにおいて，性質 P が成立するかどうかを判定する問題を考える．ここで， P は時相演算子を含まない論理式で表現可能な性質である．このとき，充足可能性判定を行う論理式は次のようなかたちになる．ここで， $I(s_0)$ は初期状態集合を， $R(s_i, s_{i+1})$ は第 i サイクル目の状態遷移関係をそれぞれ論理式で表現したものである． $P(s_i)$ は満足するべき性質を第 i サイクルの変数を使って表現したものである．

$$BMC(I, k) = I(s_0) \wedge \bigwedge_{i=0}^{k-1} R(s_i, s_{i+1}) \wedge \bigvee_{i=0}^k \neg P(s_i)$$

状態集合を保持する必要がないことから，完全なモデル検査に比べてより大規模な設計を扱うことができるという特長がある．特に早期の設計誤りの発見に有効である．

(4) SAT ベースのモデル検査

SAT ソルバを利用することで有界でないモデル検査を行うことができる．本節では， k -インダクション法とインタポレーションに基づく方法を説明する．

k -インダクション法 (k -induction method)^{7, 9)} では，具体的には，次の手続きを実行する (k の増加分は 1 でなくてもよい)* ．

* ここでは文献 9) に示されたアルゴリズムを紹介しているが， k -インダクション法にはこれ以外にも，同じアイデアに基づいた異なるアルゴリズムがある．詳細は文献 7) を参照されたい．

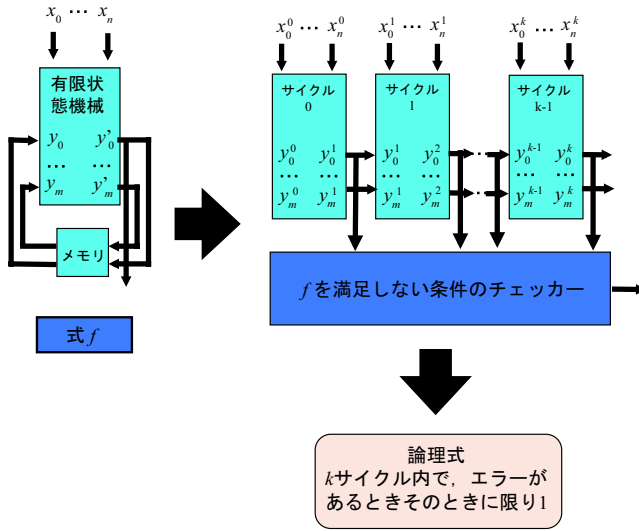


図 3-5 有界モデル検査の概念図

1. 初期状態から始まる、長さ k の系列で、 P を偽とする状態を含むものがあるかどうかを、次の論理式に対する充足可能性判定により調べる。充足可能ならば、 P が成立しない反例が見つかったことになる。充足不能の場合は 2. へ。

$$I(s_0) \wedge \bigwedge_{i=0}^{k-1} R(s_i, s_{i+1}) \wedge \bigvee_{i=0}^k \neg P(s_i)$$

2. 任意の状態から始まる、長さ $k+1$ の「ループを含まない」系列で、 k 番目までの状態では P が成立し、 $k+1$ 番目の状態では P が成立しないものがあるかどうかを充足可能性判定により調べる。充足可能な場合は k を増やして 1. へ。充足不能の場合は終了（プロパティは成立）。

$$\bigwedge_{0 \leq i < j \leq k} s_i \neq s_j \wedge \bigwedge_{i=0}^k R(s_i, s_{i+1}) \wedge \bigwedge_{i=0}^k P(s_i) \wedge \neg P(s_{k+1})$$

2. において、充足不能の場合、 P はすべての状態で成立すると判定することができる。なぜなら、 P がある状態で成立しないとすると、1. により、初期状態から始まる長さが $k+1$ 以上の系列のうち、最後の状態で P が成立しない状態に到達する系列があることになる。しかし、この系列の最後から長さ $k+1$ の部分系列（サフィックス）は、2. の論理式を真にする

系列であり、このような系列が見つからなかったことと矛盾する．ループを含まない系列の長さの上限は状態数であるので、このアルゴリズムはやがて停止する．

k -インダクション法では、調べなければならない k が非現実的に大きくなる可能性がある．次のインタポレーションに基づく方法 (interpolation-based method)⁸⁾ では、到達可能状態集合のオーバ近似集合 (到達可能集合を包含する集合) を利用することにより、これが緩和される．オーバ近似集合は次に定義するインタポーラント (interpolant) として計算される．

充足不能な式 $A \wedge B$ が与えられたとき、そのインタポーラント C は次の三つの性質を満足する論理式をいう．(1) $A \Rightarrow C$ ．(2) $C \wedge B$ は充足不能．(3) C は A と B の共通の変数で表現される． $A \wedge B$ に対する SAT ソルバーの実行結果を利用することにより、インタポーラントを求めるアルゴリズムが知られている．

これを到達可能状態集合のオーバ近似集合を求めるために用いる．具体的には、論理式 $BMC(I, k)$ を次の二つに分割する．

$$Pref(I, k) = I(s_0) \wedge R(s_0, s_1) \wedge \neg P(s_0)$$

$$Suff(I, k) = \bigwedge_{i=1}^{k-1} R(s_i, s_{i+1}) \wedge \bigvee_{i=1}^k \neg P(s_i)$$

このとき、 $Pref(I, k) \wedge Suff(I, k)$ に対してインタポーラントを計算すると、1 サイクル後に到達する状態の集合のオーバ近似集合となることが示せる．更に、 $BMC(I \vee N^i(I, k), k)$ に対して同様の操作を行うと、2 サイクル後に到達する状態の集合のオーバ近似集合を計算することができる．

具体的には、次の手続きを実行する (k の増加分は 1 でなくてもよい)．

1. $BMC(I, k)$ が充足可能ならば反例を出力．そうでなければ、 $N = I$ として、2. ~ 5. を繰り返し実行する．
2. $BMC(N, k)$ が充足可能ならば、 k を増やして、1. からやり直す．充足不能ならば、 $BMC(N, k)$ を $Pref(N, k)$ と $Suff(N, k)$ に分けて、3. へ．
3. $Pref(N, k) \wedge Suff(N, k)$ のインタポーラントを計算して N' とする．
4. $N' \Rightarrow N$ (N' が表している状態集合が、 N が表している状態集合に包含されている) ならば手続き終了 (プロパティは成立)．
5. $N = N \vee N'$ として、2. へ．

2. において、 $BMC(N, k)$ が充足可能なとき、 k を増やしてやり直さなければならないのは、初期状態集合として与えている N がオーバ近似集合となっているため、 P が成立しない状態への系列があっても、実際に反例であるかどうかを保証できないためである．4. において手続きが終了できるのは、(1) もはや N の要素が増えることはないことから N が初期状態から到達可能な状態をすべて含んでおり、かつ、(2) $N' \wedge Suff(N, k)$ が充足不能であることから、 N は P が成立しない状態へは到達しない状態の集合となっていることがいえるからである．

3.~5. を l 回繰り返した状況を考える。 N はオーバ近似集合であることから、初期状態から $l+1$ サイクル以上離れた状態も含んでいる。このため一般に収束 (4. の条件の成立) が早くなることが期待できる。停止性が保証されているが、その詳細及び k の選び方などについての議論は文献 8) を参照されたい。

文献 9) には多数の実例 (85 個の設計, 1182 個のプロパティ) に対して、後述する CEGAR アプローチも含め、種々のモデル検査アルゴリズムを適用した実験結果が報告されている。特にプロパティが成立する場合については、インタポレーションに基づく方法が、取り扱い可能な設計規模及び計算時間の点で多くの例に対して有利であることが示されている。また、プロパティが成立しない場合、特にプロパティの成立しない状態が初期状態から近い場合には、有界モデル検査が有利となることが示されている。

3-1-3 効率化のための技術

(1) 抽象化の利用

モデル検査における抽象化 (abstraction)^{2, 10)} は、与えられたクリプキ構造 (あるいは有限状態機械) の記述から、複数の状態を併し縮小したクリプキ構造を生成することを意味する。モデル検査において探索しなければならない状態数を削減することが目的である。通常、抽象化した後のクリプキ構造またはその非明示的な表現が、抽象化前のクリプキ構造上の系列すべてに対応する系列を含むように生成する。

LTL 式や ACTL 式 (ACTL は CTL のうち、意味的に時相演算子が A しかないような論理式に制限したクラス) に対する検証では、ある論理条件が「すべての系列について」満足することを要請する。この場合、抽象化されたモデル (クリプキ構造) に対してプロパティが成立すれば、抽象化前のモデルについてもプロパティの成立が保証できる。一方、成立しない場合に得られる反例は、抽象化前のモデルに存在する場合と存在しない場合がある。この場合、抽象化後のモデルに対する検証結果からでは、プロパティの成立・不成立を結論できない (図 3-6 参照)。

抽象化したモデルを自動的に得るための手法が種々提案されている。以下で触れるいずれの手法においても、抽象化前のクリプキ構造を明示的に表現することが一般に非現実的であるため、論理式や二分決定グラフで表現された遷移関係を直接変形して抽象化モデルを得る。

コーン・オブ・インフルエンス (cone of influence) は最も単純な抽象化手法であり、検証しようとするプロパティに含まれている変数に影響を与える部分をすべて取り出して、これを抽象化モデルとする (この場合、抽象化前後のモデル検査結果は変わらない)。データ抽象化は、設計中に現れるデータに注目して抽象化モデルを生成する。整数値 a を法 k のもとの値 $a \pmod{k}$ に写像したり、正負に注目して、 $a > 0, a = 0, a < 0$ の場合に分けて写像する。このとき、もとの遷移関係に存在していた遷移に対応する遷移が、抽象化された遷移関係にも含まれるように写像する。 $a > 0$ といった条件は述語とみなすことができるが、より一般に、述語の集合を与えて、この述語間の関係として遷移関係を抽象化することができ、これは述語抽象化 (predicate abstraction) と呼ばれる。有界モデル検査を利用した抽象化では、 k サイクル分展開した遷移関係の論理式が充足不能となる (反例がみつからない) 場合、論理式のなかで充足不能の証明に使われた部分 (充足不能コアと呼ばれる) のみを取り出して、これを抽象化モデルとして用いる。

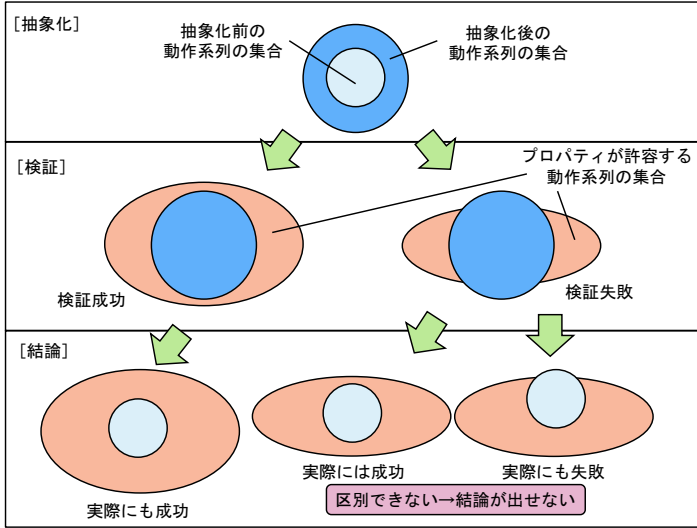


図 3・6 抽象化を利用した検証

抽象化したモデルに対して反例が得られた場合は、抽象化前のモデル上で実際に反例となっていることを確認する必要がある。抽象化前のモデル上で反例となっていないことが判明した場合は、抽象化の程度を弱めたモデルを生成する必要がある。これが詳細化である。得られた反例そのものを利用して詳細化する手法（具体的には、この反例が抽象化モデル上でも反例でないように詳細化する）が考案されており、反例による抽象化 / 詳細化手法（counter-example guided abstraction refinement, CEGAR）と呼ばれる。

(2) コンポジショナル検証

検証すべきプロパティまたは検証対象となっている設計を分解し、それぞれについて、検証を行うことにより、設計全体に対してプロパティが成立していることを保証する方法をコンポジショナル検証（compositional verification）あるいはコンポジショナル推論（compositional reasoning）という⁴⁾。例えば、 $f \wedge g$ をモジュール M に対して検証する際、 f と g を別々に証明するやり方が一例である。

コンポジショナル検証の典型的な例は、アシューム・ギャランティ（assume-guarantee）パラダイムである。この枠組みでは、各モジュール M が満たすべき性質 g に加えて、動作時の仮定となる論理条件 f が与えられることが前提となっている。 f 及び g は、有限オートマトン、LTL 式、ACTL 式などで与えられる。これを $\langle f \rangle M \langle g \rangle$ と表記することにする。ここで、二つのモジュール M_1 と M_2 について、 $\langle True \rangle M_1 \langle g \rangle$ 及び $\langle g \rangle M_2 \langle f \rangle$ であるとする。

このとき、 M_1 について $True$ （動作について特に制約がないことを示す）のもとで、 M_2 について g のもとで、それぞれ g と f が成立することを示せれば、合成したシステム $M_1 \parallel M_2$ について f が成立することが結論できる。ただし、 $\langle f \rangle M_1 \langle g \rangle$ 及び $\langle g \rangle M_2 \langle f \rangle$ のように仮定と

満たすべき論理条件が循環関係にある場合には、それぞれのモジュールの検証が成功しても、 $M_1 \parallel M_2$ に対して f や g が成立することは保証できないので注意が必要である。合成したシステムにおいて、仮定である f や g が満足される状況がない可能性があるためである。

$\langle f \rangle M \langle g \rangle$ を示すには、まず、 f が LTL 式や ACTL 式の場合は、 f を成立させる系列を反映した、タブローと呼ばれるクリブケ構造に類似した構造を生成する。次に M と直積をとる操作を行って、得られたクリブケ構造に対して g のモデル検査を行う。

(3) 定理証明方式の併用

モデル検査など自動的な検証技術で扱うことができる設計規模や複雑度には依然として限界がある。HOL や ACL といった定理証明システム (theorem prover) 上であれば、一般に高階論理なども用いた記述が可能であり、状態数に上限がないような設計も記述できる。また、これらの記述に対して、仕様として与えられた性質の証明が可能となることも多い。定理証明システムを用いる場合は、設計を公理のかたちで与え、このもつで満たすべき性質を証明することになる。

定理証明システムの枠組みの範囲でも、モデル検査などの自動検証を行えないことはないが、速度などの点で効率が良くない。HOL や ACL への組込みは、通常自動検証を行う手続きを呼び出すかたちで実装されている。定理証明システムでは、別途呼び出した手続きにより証明に成功した性質を、定理または補題として用いることができ、最終的に設計全体に対する性質を示していくことになる。定理証明システムを用いれば、数学的帰納法なども使うことができるが、証明指針をインタラクティブに与える必要があり、一般に熟練を必要とする。

より簡便な定理証明システムの利用法として、例えば、文献 11) では、浮動小数点演算器の正当性を示すために、検証問題を種々の条件で場合分けする方法がとられている。場合分けした個別の検証問題は自動検証技術 (記号シミュレーションと呼ばれる技術を用いている) によって示し、最終的にすべての場合を尽くしているかどうかを定理証明によって保証するアプローチがとられている。このほか、定理証明系を用いて述語抽象化のための述語を半自動的に抽出して、得られた抽象化モデルに対してモデル検査を行うというアプローチも提案されている¹²⁾。

参考文献

- 1) P. Molitor and J. Mohnke, "Equivalence Checking of Digital Circuits, Fundamentals, Principles, Methods," Kluwer Academic Publishers, 2004.
- 2) E.M. Clarke, O. Grumberg and D. Peled, "Model Checking," MIT Press, 1999.
- 3) 米田友洋, 梶原誠司, 土屋達弘, "ディペンダブルシステム," 共立出版, 2005.
- 4) H. Foster, A. Krolnik and D Lacey, "Assertion-Based Design," Second Edition, Kluwer, 2004.
- 5) A. Biere, A. Cimatti, E.M. Clarke, M. Fujita, and Y. Zhu, "Symbolic Model Checking Using SAT Procedures instead of BDDs," Proceedings of Conference on Design Automation, Vol.98-2, pp.142-170, 1992.
- 6) A. Kuehlmann, M.K. Ganai, V. Paruthi, "Circuit-based Boolean Reasoning," Design Automation Conference, pp.232-237, 2001.
- 7) M. Sheeran, S. Singh, and G. Stalmarck., "Checking Safety Properties Using Induction and a SAT-solver," Formal Methods in Computer-Aided Design, LNCS 1954, pp.108-125, 2000.
- 8) K. McMillan., "Interpolation and SAT-Based Model Checking," Computer Aided Verification, LNCS 2735, pp.1-13, 2003.

- 9) N. Amla, X. Du, A. Kuehlmann, R.P. Kurshan and K.L. McMillan, "An Analysis of SAT-based Model Checking Techniques in an Industrial Environment," Correct Hardware Design and Verification Methods, LNCS 3725, pp.254-268, 2005.
- 10) C. Wang, G.D. Hachtel and F. Somenzi, "Abstraction Refinement for Large Scale Model Checking," Springer, 2006.
- 11) M.D. Aagaard, R.B. Jones, T.F. Melham, J.W. O'Leary and C-J. H. Segar, "A Methodology for Large-Scale Hardware Verification," Formal Methods in CAD, LNCS 1954, pp. 263-282, 2000.
- 12) S. Ray and R. Sumners, "Combining Theorem Proving with Model Checking through Predicate Abstraction," IEEE Design and Test of Computers, vol.24, no.2, pp.132-139, 2007.

6 群 - 7 編 - 3 章

3-7 メモリのテスト / 耐故障技術

(執筆者: 安藏顕一) [2008 年 10 月受領]

3-7-1 メモリの製造テストとオンラインテスト

メモリデバイスの製造テストは、デバイス製造時の欠陥をスクリーニングするためのものであり、実使用時に想定される温度や電圧、周波数などの仕様範囲で動作することを保障するために行われる。メモリ機能単体を 1 チップとする汎用メモリに対しては、メモリテストを用いて多数個同時テストが可能であり、これによりテストコストを抑制しながら、必要な品質を達成するためのテストを行うことができる。一方システム機能の一部としてチップに搭載されるメモリでは、一つの LSI チップ上でも、多種のメモリデバイスや多数の異なる構成のデバイスがシステム内に埋め込まれており、テストからの同時テストが困難である。このため、後述の組込み自己テスト (BIST: built-in self test) の技術が必須となる。

一方システムテストは、メモリデバイスを実システムに組み込んだ状態で行われる。特にシステムが実際に稼働している状態で行うオンラインテスト (on-line testing) では、製造テストのような網羅的なテストは行えない。このため、稼働時間の経過途上に現れ始める不良や、一過性の不良などをとらえ、可能ならば修復を行う機能が必要となる。オンラインテストにも、組込み自己テストの構造が用いられることが多い。

3-7-2 メモリのテストアルゴリズム

メモリの代表的な機能的故障モデルには、縮退故障、開放故障、遷移故障、状態相関故障、多重選択故障などがあり、これらのモデルを対象としたテストアルゴリズムの開発が行われている¹⁾。一方メモリデバイスのレイアウト上の特徴を考慮して、想定される実際の物理的欠陥を対象としてテストアルゴリズムを開発する、帰納的故障解析 (IFA: inductive fault analysis) 手法も提案され、各アルゴリズムの欠陥検出能力の比較が行われている²⁾。

メモリテスト内容は、マーチング (marching) やウォーキング (walking)、ギャロッピング (galloping) などのテストアルゴリズムと、電圧・温度やアドレス方向、データバックグラウンドなどのテストストレス条件の組み合わせで表現される³⁾。マーチングテストのいくつかは、前述の機能的故障モデルや IFA での欠陥検出能力が高く、テスト時間もメモリのサイズ (ワード数) に対し線形で収まるので、基本的なアルゴリズムとして用いられている。DRAM やフラッシュメモリに対してはほかに、デバイス特性に応じたテストが追加される。

3-7-3 メモリの組込み自己テスト技術

メモリの組込み自己テストは、メモリテストに必要なテスト用信号を生成し、結果を判定する論理回路を、メモリデバイス自身あるいはメモリが埋め込まれた LSI に搭載する技術である。特に多数のメモリを搭載した SoC において重要度が高い。汎用メモリでも、少数ピンテストによる多数個同時テストの要求から、自己テスト技術が用いられる。この場合デバイス自体に BIST 回路が組み込まれるか、テスト治具上に自己テスト用機能をもつデバイスを搭載する BOST (built-out self test) 手法が用いられる。

BIST は例えば RAM に対しては、アドレスやデータ、書込みや読出し動作制御などのメモリ入力信号を自動的に生成して各メモリに与え、一方各メモリの出力に不良があるかを判

定する回路をもつ必要がある。これらはメモリデバイスに対して一対一に付加されるか、回路サイズを抑えるため複数デバイスで共有される。BIST とメモリ間のデータ転送には、データ幅分を一度にやり取りするパラレル BIST 方式と、シリアルなシフトバスでやり取りするシリアル BIST 方式とがある。テスト発生がハードウェアで固定的に実現され、回路設計後に変更することができない固定パターン BIST が歴史的に適用されてきたが、プロセス技術の微細化により必要となるパターン追加や、市場不良品の解析などの面で限界があるとされている。このため、マイクロコードをテスト時にロードすることによりテストの設定をある程度自由に行える、プログラマブル BIST も実用化されている⁴⁾。

3-7-4 メモリの故障診断技術

故障診断は、製造テストやシステムテストにおいて不良と判定されたデバイスの、物理的な解析や統計データ収集のための情報を与えるものである。不良メモリデバイスの物理解析においては、不良動作が確認されたアドレスとビット位置の情報を、メモリセル全面に対して与えるフェイルビットマップ (FBM: fail bit map) 形式の情報が、一般的に用いられる。BIST を用いて FBM を作成する場合は、通常の良否判定テスト (go/nogo testing) とは異なるモードで実行して、不良情報を逐次外部へ出力することが必要であり、一般的には時間がかかる。これに対し、BIST 実行時に不良の型をオンチップで判定する、オンチップ故障診断技術も開発されている。これはデバイスの良否判定のみでなく、不良の型判定をオンチップで行い結果を出力するもので、この情報を用いて製造ラインの統計解析や、それをもとにした歩留り改善を行うことができる。オンチップ故障診断には、BIST でのテスト結果を圧縮してシグネチャとして出力して、オフラインでの解析と診断を行うもの⁵⁾や、オンチップでロウ不良、カラム不良、ビット不良やそれらの組み合わせなどの、不良型の解析まで行うものなどがある。

メモリの故障モデルは、単独の書込みあるいは読み出し動作により発生する静的故障 (static fault) と、連続した動作により発生する動的故障 (dynamic fault) に分類できる⁶⁾。動的故障は、不良が検知されるヴィクティムセル (victim cell) とは別のアグレッサセル (aggressor cell) が原因で発生している場合がある。このような場合、不良の検知セルを示す前述のフェイルビットマップは、メモリデバイスの物理解析を行う情報としては不適であり、原因セルを特定するための、動的故障診断専用のテストパターンを別に必要とする⁷⁾。

3-7-5 メモリの不良救済と耐故障技術

(1) 不良救済

メモリデバイスの不良救済は、デバイス部品を冗長にもつ手法や、デバイス内に冗長救済機構をもつ手法により行われる。冗長部品をもつ場合、組込みのテスト回路や複数部品間の出力比較などで不良部品を検知し、冗長部品と置換する。一方、メモリデバイスは構造上規則性が高いので、デバイス内に冗長構造と救済機構をもつ場合も多い。一般的には一つ以上の予備カラム (spare column) や予備ロウ (spare row)、あるいはその両方と、これらを用いて不良のカラムやロウを置換する回路で構成される。置換回路は、デバイス外部に論理回路で構成することも可能である。

救済解析 (repair analysis) は、メモリデバイスの不良部分の特定と、救済に用いる予備カラ

ム・ロウの割当てを行う処理である．外部メモリテストを用いてこの処理を行う場合，メモリの出力をテスト上のデータメモリに保存し，故障診断に用いられるフェイルビットマップと同様の情報を構築する．これからテスト上の救済解析ソフトウェアを用いて，冗長救済の解を求める．求めた救済解のデータを基に実際の冗長救済処理を行うが，これは光学的ヒューズや電氣的ヒューズなどのヒューズデバイスや，不揮発性のメモリデバイスを用いたプログラムを行うことで実現される．

上記メモリテストを用いた救済は，大規模集積回路の埋込みメモリのように多種多数個のメモリデバイスが埋め込まれた状態では，テスト用のピン数やテスト時間の制約から，適用するのが難しい．組込み冗長割付（BIRA: built-in redundancy allocation）回路は，BIST を用いたメモリのテストと平行して救済解析を行うもので，上記のような制約をもたずに処理を行える．また，組込み自己救済（BISR: built-in self repair）回路は，求められた冗長救済解からプログラムデバイスへのデータ転送，プログラム，プログラム結果の検証などを自動で行うものである．BIST 及び BIRA 回路と併用することにより，メモリのテスト，救済解析，救済処理と救済後の再テストまでの一連の動作を，オンチップで自動的に行うことができる．製造テスト時の上記処理のように冗長救済を恒久的にプログラムする手法を，ハードリペア（hard repair）と呼ぶ．一方，システムの起動時や稼動中にオンラインテストを行い，不揮発性メモリデバイスなどに救済情報をもつことで，起動中の不良動作をマスクするソフトリペア（soft repair）手法が，BISR 技術を用いることで実現可能である．

（２）耐故障技術

メモリデバイスは製造テストを経て出荷され，システム上に実装される．システム稼動中にメモリデバイスに現れる不良には，製造テストの抜けによるものや，出荷後の経年による不良，過渡的な不良などがある．製造テストの抜けによる不良は，製造テスト時と同等のテストでは検出され得ない．経年不良に関しては，システムの起動時あるいはシステム稼動中に BIST を用いることにより検出し，可能なものは救済処理を行うことができる．

過渡的な不良のうち，放射線などによる影響で回路が誤動作を起こすソフトエラー（soft error）に関しては，メモリデバイスの微細化度・集積度が向上するにつれ，その発現確率であるソフトエラー率（SER: soft error rate）が，急激に増加すると見られている．これらは製造テストでは捕えられないものであり，プロセス・デバイスや回路設計の面からの，あるいはシステムの対策が必要とされている．プロセス・デバイスの対策では，SOI（silicon on insulator）を用いる方法など，また回路設計では SRAM セル内に冗長なラッチをもたせて耐性を上げる手法³⁾などがある．システムの対策では，ECC（error correcting code）が最もよく用いられる．例えば単一ビットエラー訂正コード（single bit error correction code）をメモリデバイス周辺の回路により実現すれば，単一ワード内のソフトエラーは 1 ビットまでは許容される．

参考文献

- 1) C.-W.Wu, "Memory Testing and Built-In Self-Test," in VLSI Test Principles and Architectures, ed. L.-T. Wang et al, Morgan Kaufmann Publishers (an imprint of Elsevier Inc.), pp.462-515, 2006.
- 2) J. P. Shen, W. Mary, and F. J. Ferguson, "Inductive Fault Analysis of CMOS Integrated Circuits," IEEE Design & Test of Computers, vol.2, no.6, pp.13-26, December, 1985.

- 3) A. J. van de Goor, S. Hamdioui and R. Wadsworth, "Detecting Faults in the Peripheral Circuits and an Evaluation of SRAM Tests," IEEE International Test Conference, pp.114-123, 2004.
- 4) J. Dreibelbis, J. Barth, H. Kalter, R. Kho, "Processor-based Built-In Self-Test for Embedded DRAM," IEEE Journal of Solid-state Circuits, vol.33, no.11, pp.1731-1740, November. 1998.
- 5) J. T. Chen, J. Khare, K. Walker, S. Shaikh, J. Rajski and W. Maly, "Test Response Compression and Bitmap Encoding for Embedded Memories in Manufacturing Process Monitoring," IEEE International Test Conference, pp.258-267, 2001.
- 6) S. Hamdioui, Z. Al-ars, A. J. van de Goor, M. Rodgers, "Dynamic Faults in Random Access Memories: Concept, Fault Models and Tests," Journal of Electronic Testing: Theory and Applications, vol.19, pp.195-205, 2003.
- 7) S. K. Thakur, R. A. Parekhji, A. N. Chandorkar, "On-chip Test and Repair of Memories for Static and Dynamic Faults," International Test Conference, vol.30, no.1, 2006.
- 8) K. Chakraborty and P. Mazumder, "Fault-Tolerance and Reliability Techniques for High-Density Random-Access Memories," Prentice Hall, 2002.

6 群 - 7 編 - 3 章

3-8 アナログ/ミックスドシグナル系のテスト

3-8-1 テストの概要

(執筆者：三浦幸也)[2008年10月受領]

多くのアナログ・ミックスドシグナル(AMS)回路では、連続値である電圧や電流(連続信号)が意味のある信号として扱われる。また AMS 回路には多種多様な回路機能や回路構成が存在し、回路機能と回路構成が一意に対応しない¹⁾。このため、AMS 回路のテストにおいては、故障モデルの構築が困難であるので抽象的な故障モデルの概念が存在しない。また AMS 回路では、回路内のパラメータ変動や使用環境の変動などに対して回路特性を安定させるために帰還回路が用いられており、AMS 回路自体が耐故障性を有している。更に、故障箇所における故障情報の現れ方が複雑であることや、故障箇所から特定方向への故障の影響の伝搬が容易でないことなどから、外部出力での故障影響の観測が困難である。

AMS 回路が正常か否かを識別することは、その回路の仕様(スペック)や用途に深く依存している¹⁾。このように AMS 回路のテストは、回路仕様を満たすか否かを調べることに帰着するので、テスト項目が多数あり、また高い計測精度が要求される。このほかに、回路構造や回路機能に依存しないテスト方法として、自己発振回路を構成して動作速度や回路特性を調べるオシレーションテスト、回路内の素子パラメータの変動による回路動作への影響を調べる感度解析に基づいたテスト、信号のタイミングを評価するジッターテスト、などがある。近年は高周波 AMS 回路をテストするために低コスト・広帯域・高精度のテスト方法への要求も高まってきている。

3-8-2 AD コンバータ, DA コンバータのテスト

(執筆者：三浦幸也)[2008年10月受領]

AD コンバータ(ADC)と DA コンバータ(DAC)の特性を表す代表的な指標として、量子化誤差と非直線性誤差がある。図 3-7 に 2 ビット出力の ADC の理想伝達特性を示す。n ビット出力の場合、アナログ入力電圧範囲(フルスケールレンジ:FSR)に対して、FSR/2ⁿ ごとの入力に対して出力コードが変化する。量子化誤差はアナログ値(連続値)とデジタル値(離散値)との変換過程で本質的に発生し、図 3-7 に示す理想コンバータにおいても最大で $\pm(1/2)$ LSB の量子化誤差がある。非直線性誤差は微分非直線性誤差(DNL)と積分非直線性誤差(INL)に分けられる。それぞれ、1LSB に対応する理想電圧と実際のアナログ入力(DAC では出力)電圧との差、及び理想変換直線との最大偏差を表す。このほかに ADC や DAC の特性を表す指標としてゲイン誤差やオフセット誤差などがある。

ADC の一般的な非直線性誤差のテスト法としてヒストグラム法がある。図 3-8 は、図 3-7 の変換特性に対して、ランプ電圧を印加しながら 4 サンプル/出力コードで出力コードの発生回数をカウントしてヒストグラムを作成した例である。この方法では、検出可能な DNL の精度は 1LSB ごとのサンプル数に依存する。ADC と DAC にも各種の回路構成方法があるため、実際には回路構成ごとにテスト項目(評価特性)が異なる。

3-8-3 テスト容易化設計手法, テストインタフェース

(執筆者：亀山修一)[2008年10月受領]

電子回路の高集積化に伴い、従来の実装基板試験手法であるインサーキットテスト(ICT)

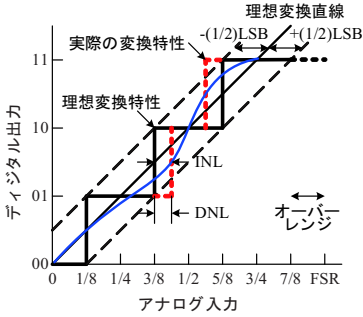


図 3・7 2 ビット ADC の理想伝達特性

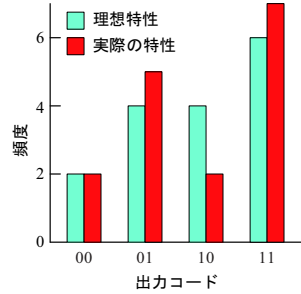


図 3・8 ヒストグラム法

や自動外観検査 (AOI) ではテストカバレッジ低下の問題があり, IEEE1149.1 規約に基づくテスト容易化設計手法 (バウンダリスキャンテスト / JTAG テスト) が広く用いられるようになった。デジタル回路を対象とした 1149.1 規約をアナログまで拡張したものが 1149.4 規約 (通称アナログバウンダリスキャン) である^{2,3)}。

1149.4 規約では物理的なテストインタフェースピンとして従来の 1149.1 規約での 5 ピン (TDI/TDO/TCK/TMS/TRST) に加え, アナログテストポート (AT1/AT2) の 2 ピンが追加されている。LSI デバイスは図 3・9 のようにパッケージピンと内部コア間にテスト用の回路が組み込まれ, デジタル信号ピンには従来のバウンダリスキャンレジスタと同等のデジタルバウンダリモジュール (DBM), アナログ信号ピンにはアナログバウンダリモジュール (ABM) が組み込まれている。各 ABM はアナログテストバス (AB1/AB2) でアナログテストポートと接続される。AT1/AT2 は基板上のテストバスに接続され, 外部のテストに接続される。

テストインタフェースにおいて, 1149.4 規約は 1149.1 規約の上位互換となっており, 各々の規格に基づく LSI を基板上で共通のテストバスに接続可能である。

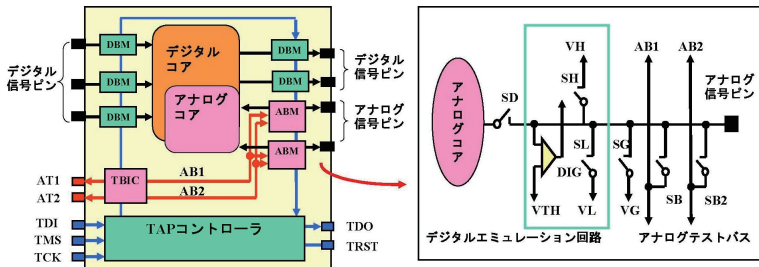


図 3・9 IEEE1149.4 対応デバイスの構造と ABM 回路

(1) インタコネクションテスト (EXTEST)

ABM のデジタルエミュレーション回路により、アナログネットを 1149.1 でのデジタルネット等価にし、LSI 間のインタコネクション (オープン / ショート) 試験を行う機能 (図 3・10)。

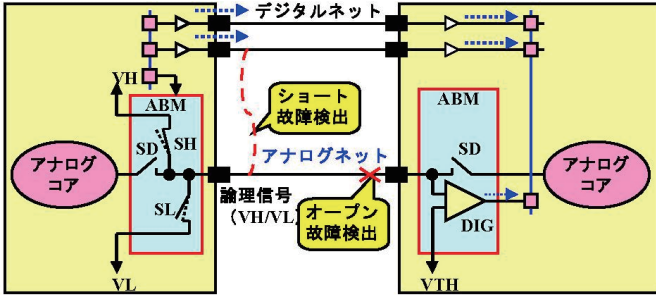


図 3・10 インタコネクションテスト

(2) パラメトリックテスト (インピーダンス測定)

アナログ信号ピンに接続されている基板上の部品 (抵抗, コンデンサなど) の値を ABM 経由で外部計測器に接続して測定し,正しい部品が実装されているかテストする機能 (図 3・11)。

(3) アナログブローブ

システム動作中の任意のアナログピンの信号波形を ABM 経由で外部計測器に接続して観測する機能 (図 3・12)。

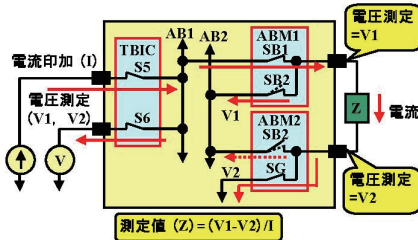


図 3・11 パラメトリックテスト

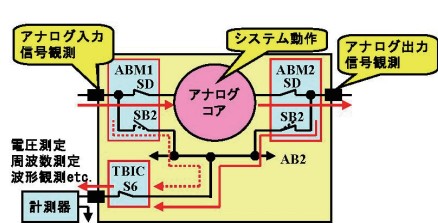


図 3・12 アナログブローブ機能

(4) インテスト (INTEST)

LSI 内部のアナログ / デジタル混在回路 (DAC, ADC など) をテストする機能。

参考文献

- 1) P. Kabisatpathy, A. Barua, and S. Sinha, "Fault diagnosis of analog integrated circuits," Springer, The Netherlands, 2005.
- 2) "IEEE Standard for a Mixed-Signal Test Bus (IEEE Std 1149.4-1999)," IEEE, New York, 2000.
- 3) Kenneth P. Parker, "THE BOUNDARY-SCAN HANDBOOK Third Edition," Kluwer Academic Publications, 2003.

6 群 - 7 編 - 3 章

3-9 故障解析 / 故障診断

(執筆者：佐藤康夫，益子洋治)[2009年7月受領]

故障解析では、LSIの信頼性及び歩留品質の向上のために、故障サンプルの原因究明を行い、設計、製造プロセス、あるいは製造テストへフィードバックを行う。故障の原因としては、静電破壊（ESD: Electrostatic Discharge）のような外的要因によるもの、配線のショートや断線、あるいはゲート酸化膜のリークなどの物理欠陥などのような内的要因によるものがある。後者に関しては、特に最近の微細加工の製造プロセスで、銅（Cu）配線の断線、貫通孔（ビア）のポイド、低誘電率（Low-K）層間膜や薄膜化した酸化膜の欠陥などが問題となっている。また、ホットキャリア注入（HCI: Hot Carrier Injection）、負バイアス温度不安定性（NBTI: Negative Bias Temperature Instability）、経時絶縁破壊（TDDB: Time Dependent Dielectric Breakdown）、エレクトロマイグレーション（EM: Electro Migration）、あるいはストレスマイグレーション（SM: Stress Migration）などのフィールドでの劣化現象は、製造テストでの検出が困難な面がある。しかし、LSIの信頼性に影響を及ぼす重要な課題であるとされている。製造バラツキによる特性変動などによる故障は、物理欠陥による故障とは少し性質を異にするが、設計段階で見積もった特性変動量とで上がり特性変動量の差異の問題とかかわって深刻な問題となりつつある¹⁾。

故障解析の方法としては、製造モニタチップ（テストチップまたはTEG: Test Element Group）などのように、あらかじめ解析に特化した設計構造を入れ込み、測定を行う方法もあるが、多くはでき上がった製品チップを、テスト測定結果、発光解析装置やEB（Electron Beam）テストなどの装置を用いた物理観測結果を利用する手法、あるいはそれらを組み合わせた方法であり、本節ではそれらの詳細技術を紹介する。

3-9-1 故障解析の分類

故障解析はその目的によって、(1) 開発デバッグ時の不具合あるいは出荷後の不具合チップの解析、及び、(2) 生産時の不良の原因解析、に分けられる。(1)は、出荷後にボード組立て工程、実機試験、あるいはフィールド使用状態で不都合が見つかったチップの解析である。製品チップの性能や品質に直接影響する可能性があるため、迅速で、かつ慎重な解析が求められる。一方、(2)は生産の歩留まりを向上する目的で行われる。一つひとつの不良の原因究明よりは、歩留まりに大きな影響を与えている不良の原因究明が主目的である。これは従来TEGを用いたり、製品のメモリ部を解析したりすることが主流であったが、近年、配線系の複雑な故障や、特性上のマージナルな故障を解析するために製品チップも用いるようになった。この目的で使用される故障診断手法は、多数のチップを対象にすることから、大量故障診断（Volume Diagnosis）と呼ばれる場合もある。各々の目的により故障解析の手法や手順も多少異なるが、共通な技術も多い。本節ではまず従来から行われてきた(1)を中心に、現状の技術を紹介する。

3-9-2 故障解析手順

図 3-13 に故障解析の代表的な手順を示す。

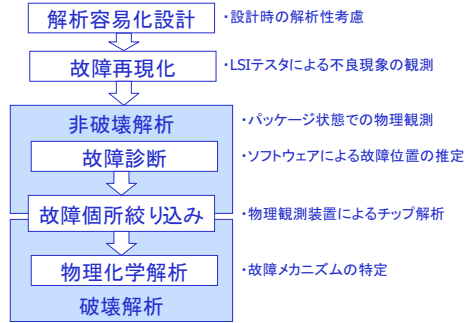


図 3・13 故障解析の手順

(1) 解析容易化設計

LSI の解析を効率的に行うには、設計段階から解析容易化設計を行うことが重要である。解析容易化設計としてはアドホックな手法が多く用いられてきたが、近年、テスト容易化設計が普及するに従い、解析でもそれが利用されるようになってきた。オンチップの SRAM などでは、メモリ BIST (Built-In Self-Test) が広く用いられているが、解析時には通常、フェイル*したメモリビットの位置情報をダンプできる。論理部にはスキャン設計が用いられるので、本節で紹介する故障診断 (Diagnosis) を適用するようになってきた²⁾。

(2) 故障再現化 (解析用テストパターン作成)

生産テスト以降で発見された故障を解析するには、まず故障現象を LSI テスタなどの測定環境で観測可能にする必要がある。これを故障の再現化と呼ぶ。生産テストで良品とされて出荷されたチップが対象のときは、この再現化は困難を伴うことが多い。テスト時のタイミング条件や電圧条件を変えて不良の動作条件を見つける手法 (シューム・プロット) も故障再現化の一種といえよう。故障の箇所がある程度推定されている場合には、故障診断での故障位置推定の分解能を高める解析用のテストパターンを作成する場合もある。

(3) 非破壊解析

パッケージ状態のチップをいったん開封すると、不良現象の再現が困難になったり、不良箇所が物理ダメージで破壊されたりする可能性があるため、通常、まずパッケージ状態で可能なテスト測定、外観解析、X 線透視観測、超音波探傷解析 (SAT: Scanning Acoustic Tomography) などの物理観測を行う。

LSI テストによる故障現象の再現が成功した場合は、テスト結果からのソフトウェアによる故障位置の推定を行う。これを故障診断 (Diagnosis) と呼ぶ。故障診断が対象とするテスト手法はスキャンテストが主流であるが、論理シミュレーションパターンを流用したファンクションテスト (機能テスト) でも適用可能な場合もある。I_{DDQ} (Quiescent Power Supply Current) テストにおいて、測定ポイントに依存して電流値のパス / フェイルが異なる場合は、

* LSI テスタでの観測結果があらかじめ期待した値と一致することをテスト合格の意味で“パス”、一致しないことをテスト不合格の意味で“フェイル”と呼ぶ。

I_{DDQ} テスト用の故障診断が適用可能である．測定ポイントによらず電流が流れっぱなしのサンプルは論理的な情報が得られないため，発光解析などの物理解析手段による必要がある．

物理的手法による故障箇所絞り込み技術は，不良の種類にかかわらず広く適用可能である．故障 LSI のなかで生ずる様々な物理現象は何らかの特異性を示す．物理的な観測装置を用いて，その特異情報をもつ情報担体を検出することで故障箇所を絞り込むことになる．故障診断で候補位置がある程度推定できた場合は，この物理的手法による故障箇所絞り込みを適用することで故障位置の確度と精度を高めることができる．故障箇所絞り込み技術は，基本的には動作状態での解析を前提とするので，パッケージは開封されてもチップ自体の破壊は行わない．しかし，故障位置が回路ブロックや特定回路などにある程度絞り込まれれば，破壊解析による故障箇所絞り込みで更に精度の向上を図ることもできる．最終的な故障位置の確定は，直接物理化学解析による形態的異常などの観察による場合と，故障箇所絞り込み技術を破壊解析に適用して更に精度の向上を図る場合とがある．後者は，チップ全体が動作しない場合でも，適用技術の性格上，電圧や電流の供給による部分的な故障の活性化が必要である．

(4) 物理化学解析

故障のメカニズム（故障原因）を明らかにするためには，まず故障箇所が発生している物理現象の把握が不可欠となる．このためには様々な物理化学解析の手法が適用される．故障位置が特定できた後の解析となるため，この解析では解析対象領域のみが保全されれば，あとは完全な破壊解析になる．この物理化学解析には，ナノメートルから原子サイズレベルまでの空間分解能が要求され，故障の内容によって，単純な形状情報の取得から，材料の組成分析，極微量不純物やドーパント元素分布，機械的な応力のナノ領域での分布，結晶欠陥・結晶の方位分布を含む結晶性の評価などが実施される．用途ごとに解析装置も異なり，多くの最先端の装置が必要とされる．

3-9-3 故障診断による故障位置推定技術

(1) 故障診断の種類

故障診断は LSI テスタで観測した情報を基に LSI 内の故障位置を推定する技術である．表 3-1 に各故障診断手法とその故障情報の観測情報を示す．

表 3-1 故障解析の観測情報

テスト手法	観測情報
スキャンテスト	全FFの論理値
圧縮スキャンテスト	圧縮値
論理BIST	シグナチャ値(符号)
部分スキャンテスト	スキャンFFの論理値
機能テスト	LSI出力ピン値
IDDQテスト	電源電流値

・スキャンテストによる故障診断

(注：スキャンテストについては本章 3-5-2 節を参照のこと)

スキャンテストでは、論理回路内の記憶素子 (FF) 及び出力ピン (スキャンテストで観測する、これらをまとめて疑似プライマリ出力という、本節では簡単のため単に FF と記すことがある) の値が LSI テスタに読み出せるので、テストパターンごとに各 FF の値が期待値と一致するか否かを判定することができる。近年、大規模な LSI に対してはテストデータ量を削減する目的で、スキャンテストを改良した圧縮スキャンテストが用いられている。圧縮スキャンテストでは出力の FF 値は圧縮されるため、各 FF の値は直接には分からない。そこで故障診断用に出力値の圧縮をバイパスさせて各 FF の値を非圧縮で読み出す方法や、圧縮された出力値から各 FF の値を推定する手法などが提案されている。スキャンテストベースの論理 BIST (Built-In Self-Test) は、テスト結果全体を圧縮したシグナチャ値を LSI テスタで観測し合否判定を行う。故障解析時には、図 3-14 のように、テスト全体をいくつかに分割し、誤り結果が含まれる部分のみを再テストする例が提案されている。そのときにその部分の結果は、故障診断用に出力値の圧縮回路をバイパスさせる³⁾。

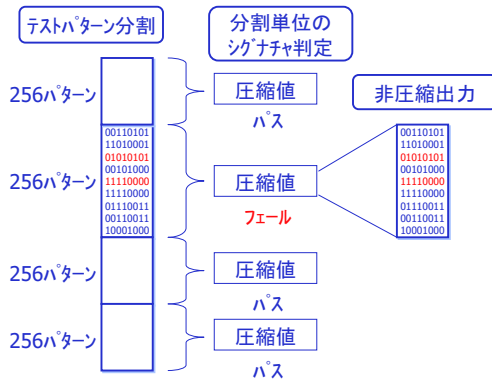


図 3-14 論理 BIST の観測方法例 (文献 3) を元に著者が作成)

・部分スキャンテストによる故障診断

プロセッサなどのテストでは LSI の性能を確保するために FF のスキャン化を部分的に行っている⁴⁾。非スキャン FF の値は、値が分かっているスキャン FF から複数個のクロックサイクルを経て値が確定する。そのため故障診断時には、故障が観測された FF から複数クロック分を逆トレース (本章 3-9-3 節 (2) に詳細を示す) して故障発生箇所を求める必要がある。

・機能テストによる故障診断

機能テストの故障診断は順序回路の問題となるため一般には難しくなる。部分スキャンテストと同様の方法で複数のクロックサイクルを逆トレースする方法が考えられるが、トレース範囲が、クロック段数が増えるに従い広がるため技術的には難しくなる。文献 5) では図 3-15 に示すように異なる方法でマイコンの故障診断を実用化している。処理 A でテストパターンから命令コードを抽出・逆アセンブルし、故障のモジュールを推定する。処理 B では

故障モジュールに対して最初の数十パターンに対応する故障辞書（本章 3-9-3 節 (3) に詳細を示す）を作成する。この故障辞書から故障候補のリストを作成し、処理 B に戻って次のパターンに対応する故障シミュレーションを行う。これを繰り返すことで故障候補を絞り込んでいく。

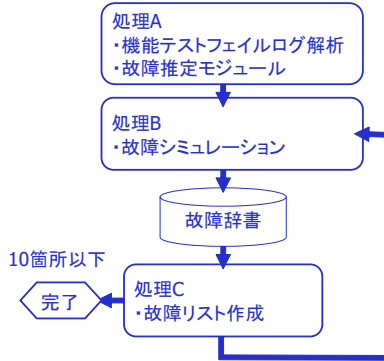


図 3-15 機能テストの故障診断例（文献 5）を元に著者が作成）

・ I_{DDQ} テストによる故障診断

I_{DDQ} テストでは、テストパターンごとに電源電流値を測定し、バス/フェイルを判定する。生産テストに通常用いられるテストパターン数は、数パターンから数十パターンであり、スキャンテストに比べてはるかに少ない。例えば、5 パターンのテストを考えると、テスト結果から推定できる情報量は $2^5 (= 32)$ 通りしかないので、故障診断を行うにはテストパターン数を増加して、図 3-16 に示すように分解能を高めた再測定を行うことが望ましい。

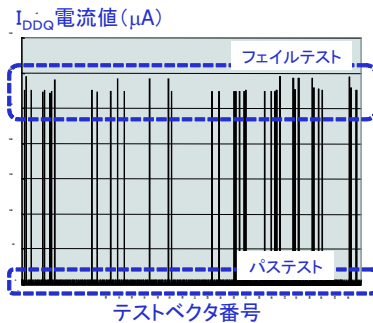


図 3-16 多数の I_{DDQ} テストパターン測定例

(2) 故障位置の推定方法

故障位置の推定方法として原因結果法と結果原因法がよく知られる。原因結果法は、故障辞書と呼ぶ故障箇所と出力値の対応表と、テストでの観測結果を照合して、故障候補箇所を

絞り込む方法である。故障辞書概念を図 3・17 に示す。正常回路は故障のない状態の回路を示し、 f_1, f_2, f_3 は故障のある回路を示す。回路の出力 O は回路の出力値が観測される箇所、スキャン FF または出力ピンに相当する。一般には複数箇所存在するが、各箇所に対してここで示した故障辞書を作成する。 V_1, V_2, V_3, V_4 は入力パターンを示し、それぞれに対応する出力値が表に書かれている。斜体の論理値は正常回路と比べて誤り値になることを示す。この対応表を用いて、実際に故障の影響が観測されたスキャン FF に共通な故障を故障位置の推定箇所とする。大規模 LSI では故障辞書の大きさが膨大になる問題と、故障辞書の作成に故障モデルが必要なので多様な故障モデルへの展開が難しい問題がある。そこで、故障シミュレーションを実行時に動的に故障辞書作成と絞り込みを行う辞書レス手法も用いられる⁶⁾。

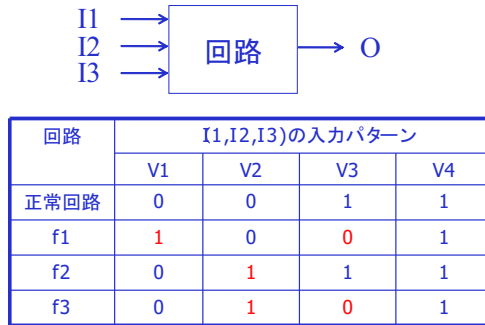


図 3・17 故障辞書概念

結果原因法は、観測結果から論理を逆トレースすることにより故障位置の推定箇所を求める。図 3・18 に逆トレース手法の例を示す。図の論理値は正常回路の値を示し、太い線は FF からの逆トレース結果を示す。例えば 2NAND 回路の片側の入力値が 0 だと、仮に他方の入力値に誤った値が来てもそこから前方に故障値が伝搬することはない。したがって、0 入力値がある場合はそちら側をトレースしていく。両方とも 0 入力がない場合は、両方をトレースする。このようにトレースすると結果は図 3・19 のようなコーン状になる（コーン内の回路は省略）。フェイルした各 FF から逆トレースされたコーン内に共通に含まれる回路があれば、それらが故障位置として候補にあげられる。実際にはいろいろ工夫された手法が提案されている⁶⁾。

(3) 故障モデル

欠陥の動作を計算機でシミュレーションするため数々の故障モデルが提案されている。故障診断で用いる故障モデルは、故障を起こしている個々の LSI で、欠陥の動作をできるだけ忠実に模擬すること、及び計算機で効率的に処理できることなどが求められる。テストパターン生成でも故障モデルを用いたが、故障診断ではテストパターン生成と同じ故障モデルが用いられ、更に精度を高めるために複雑な故障モデルが提案されたりしている。

LSI テスタでの実際の各 FF 値の観測結果と、故障モデルを想定してシミュレーション（故障シミュレーション）により故障が伝搬すると推定された FF の論理値との関係は、

- ・「一致」: 両者の一致

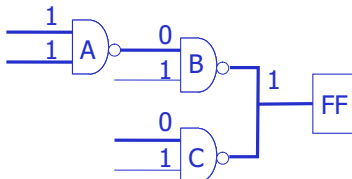


図 3-18 逆トレース手法の例

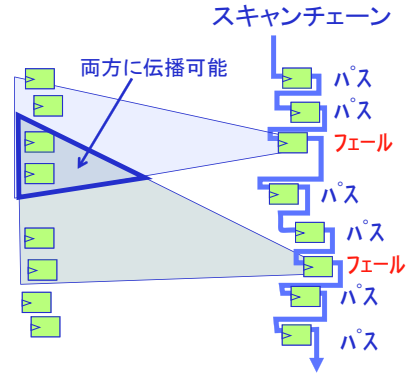


図 3-19 故障箇所の推定

・「誤指摘」：故障モデルでは故障観測（誤り値），LSI テスタでは故障が非観測（正常値）
 ・「非予測」：LSI テスタでは故障観測（誤り値），故障モデルでは故障が非観測（正常値）
 の 3 ケースに分けられる。「一致」の数が多いほど，確からしい故障候補位置及び故障モデルと考えることができる．特に「非予測」は，想定した故障以外が発生していることを示し，指摘自体が誤っている可能性も高いと思われる^{4,7)}．故障診断で指摘された故障候補位置のリストと，真の故障位置が完全に一致する場合を「完全一致」と呼ぶ．故障候補位置のリストの集合のなかに真の故障位置を含むが，それ以外も含む場合を「部分一致」と呼ぶ．故障診断の評価としては，故障候補位置のリストに真の故障位置を含むことと，また「部分一致」の場合は，リストに含まれる候補箇所数が少ないことが重要である．代表的な故障モデルを以下に示す．

・縮退故障モデル

故障時の論理値が 0 または 1 に固定されるモデル（本章 3-2 節を参照）．

・ブリッジ故障モデル⁸⁾

二つの信号間がショートしていることにより発生する論理値誤りを表す．ワイアド AND 及びワイアド OR モデルは，二つの配線ノードの値がそれぞれ正常時の値の AND（論理積）または OR（論理和）で決まる場合を指す．A ドミナント及び B ドミナントモデルは，二つの信号ノードの論理値が片方の信号線の値だけで決まる状態を表す．これは片方の信号線の回路駆動能力が他方よりも常に大きい場合に適用される．コンポジット・シグナチャ⁹⁾はこれらのモデルよりも汎用的なモデルで，二つの信号ノード A, B の値が，{0 縮退故障 (A), 0 縮退故障 (B), 1 縮退故障 (A), 1 縮退故障 (B)} の四つの故障の可能性を，まとめて扱って故障辞書を作成する故障診断法である．これは 2 信号のブリッジ故障のあらゆる可能性を包含している．

・オープン故障モデル

信号線が電氣的に切断された状態の欠陥を表し完全オープン故障とも呼ぶ．これに対して，信号線が高抵抗状態で繋がっているオープン故障を高抵抗オープン故障と呼ぶ．完全オープン故障の場合，シンク側の論理値は電氣的接続が切れるので不安定な状態となり，一般に予

測が困難となる．そこで故障時の論理値を X 値として解析する故障診断手法などが提案されている^{10, 11)}．またコンポジット・シグナチャの拡張としてネットモデルが提案されている．ネットの幹と枝にそれぞれ 0 縮退故障と 1 縮退故障を考え、ネット全体をまとめて故障辞書を考える¹²⁾．

・ディレイ故障モデル

ディレイ（遅延）故障は何らかの原因による高抵抗状態で引き起こされる．テストパターン生成と同様に、2 パターンテストで故障箇所論理的な信号遷移が伝わらないとする遷移遅延故障モデルや、ロバスト条件などで活性化された論理パスの遅延値が遅れるとするパス遅延故障モデルなどが用いられる（本章 3-2-2 節を参照）．

(4) 精度向上のための診断手法

論理的な故障モデルを用いた故障診断手法について紹介したが、近年の微細加工の製造プロセスの故障に対応するため、更に詳細な故障診断手法が提案されている．

・多重故障の故障診断

チップ内に複数の故障発生箇所が存在する多重故障は、これまで紹介した手法ではうまく故障候補位置を絞り込むことができない．しかし一つのテストパターンに着目すると、ただ 1 箇所のみ故障が観測されることが多いことが報告されている．ただ 1 箇所のみが活性化され故障伝搬されるテストパターンを SLAT (single location at-a-time) パターンと呼ぶ¹³⁾．一つの SLAT パターンに単一故障の故障診断手法を適用すると、真の故障は多くとも 1 個しか候補として含まれない．推定候補数は非常に多く出てくるが、多数の SLAT パターンに対して故障診断を行い、それらすべてを矛盾なく説明でき、また最も少ない故障の組を多重故障の候補位置とする．

・レイアウト情報を用いた故障診断

ブリッジ故障については、配線パターンの隣接箇所と故障モデルを用いた候補箇所を突き合わせる故障診断手法が広く用いられている．またオープン故障についてもレイアウト情報を用いた手法が提案されている¹⁴⁾．オープン状態になった信号線は、隣接配線からの容量結合によりその電位が変動する．そこで、配線パターン上のピア位置などに故障を仮定し、隣接配線からの容量結合による電位変動を計算する．故障モデルにより求めた複数の故障候補位置に対してこうした検証を行い、LSI テスタの観測結果と合致する故障候補位置を求める．

・回路ベースの故障診断

回路レイアウトに故障を注入し回路シミュレーションなどにより予測した動作が、LSI テスタで観測された動作と一致するかを調べる．SPICE による回路シミュレーションは計算機時間が膨大になるため、スイッチレベル回路シミュレーション（トランジスタのオン・オフのみを計算する）で高速化を図った例も報告されている^{15, 16)}．

3-9-4 物理的手法による故障箇所絞り込み技術

実際の TEG や製品チップの上で故障箇所を最終的に絞り込むためには、物理的手法による故障箇所絞り込み技術が用いられる．ここでは実用的に用いられている代表的な物理的手法・技術を取り上げて解説する．

故障に対して直接、物理的手法による故障箇所絞り込みが適用される場合もあるが、近年、多くは故障診断と併用されるようになってきた．故障診断によって得られた故障箇所候補を

確定、または、更に詳細に絞り込むために物理的手法が適用される。特に大規模な回路では解析の大幅な効率化を図る目的で、まず故障診断が適用される。

故障診断による故障箇所絞り込みでは故障を、主に“0”か“1”の論理の状態を扱うのに対して、物理的手法を用いる故障箇所絞り込みでは、“ショート”や“オープン”あるいは“高抵抗”などの電気的異常状態から生ずる、発熱や発光などの物理的現象として扱うことになる。そのため、単に故障回路のノード部分の検出のみでなく、コンタクト部分とかトランジスタのゲート部分といった、より詳細な部品にまで立ち入った故障箇所位置でとらえることもできる¹⁷⁾。これは、故障の物理的現象解明のために引き続いて実施される物理化学解析にとっても極めて有効な情報となる。特に微小薄片の試料を切り取って取り扱う透過電子顕微鏡 (TEM) などでの解析には重要な情報となる。

物理的手法を用いた故障箇所絞り込みでは、物理的な情報が得られるため、簡単な故障モードでは、物理化学解析にまで運ばずに、そこでの現象を把握、原因を推測できる場合もある。また、いったん経験した(解析したことのある)故障については、物理的手法を用いた故障箇所絞り込みのみで、その特徴的な物理情報から、故障原因との照合や故障モードの分類も可能となる。

(1) 故障箇所絞り込みに用いられる主な技術

現在よく用いられている故障箇所絞り込み技術について、静的な状態で行うスタティック解析と動作状態で行うダイナミック動作解析に分類し、更に、ショート箇所の検出やタイミング解析などの目的によって分類すると、おおむね以下の表のように表される。

表 3-2 物理的手法による故障箇所絞り込み技術

ースタティック解析ー		ーダイナミック動作解析ー	
電流リーク箇所 (絶縁破壊, pn 接合リークなど)	発光解析 (PEM / EMS) 発熱顕微鏡	電流リーク箇所	IDDQ+PEM (EMS) 発熱顕微鏡
電流リーク経路 (回路ショート)	OBIRCH (IR-OBIRCH) RCI (EBAC)	タイミング解析	TREM, EBT
回路オープン (配線断線)	EBT, VC-SEM, VC-SIM	信号波形観測	EBT, LVP
		マージナル故障解析	SDL / LADA

(2) 代表的な絞り込み技術の原理と用途

(a) 発光解析 PEM (Photo Emission Microscopy) / EMS (Emission Microscopy)

LSI チップから発せられる微弱な光をとらえて発光位置を画像化する解析手法である^{18, 19, 20)}。発光現象には、電子 - 正孔対の再結合、加速されたキャリアのフォノンとの相互作用によるエネルギー放出、制動放射、そして温度による熱放射 (黒体放射) がある。少数キャリアの再結合による発光では、pn 接合の順方向電流飽和モードのバイポーラトランジスタや CMOS のラッチアップがあり、高電界で加速されたキャリアの発光には、飽和領域での MOS トランジスタ、ホットキャリア (パンチスルー、アバランシェ) によるもの、pn 接合の逆バイアス電流 (接合リーク) など空間電荷領域での現象のほかに絶縁膜のリーク箇所での発光がある。熱放射では回路のショートなどによる発熱箇所での発光がある。発光解析ではこのような直接的な観察のほかに、例えば回路のどこかにオープン箇所が存在した場合に、そのネットに接続されたゲート電圧が (浮いて) 中間電位になることで、予期せぬトランジスタが動

作・発光し、間接的にショート箇所にとどり着くこともできる。

当然この手法は、対象となる箇所にはバイアスが掛かっていることが前提であり、例えばリーク箇所があっても、スタンバイ状態でその両端に電位差がなければ発光は確認されず、リーク箇所も見逃されることになる。このためより有効な故障箇所検出を実現する目的で、論理の静止状態での電源電流 I_{DDQ} テストと組み合わせた発光解析が用いられる²¹⁾。すなわち、テストパターン依存性をもつ I_{DDQ} 電流異常値を示す場合には、その状態において発光解析を行うことで、リーク箇所などを検出することができる。

(b) 時間分解発光解析 TREM (Time Resolved Emission Microscopy)

発光解析 (PEM) は一般に、静的 (スタテック) な状態での解析を意味する。これに対して回路を動作させた状態で発光を時間的にとらえて、その回路の動作タイミングを解析する方法は、静的なものとは区別して TREM²²⁾ と呼ばれる。ピコ秒でのタイミング解析が可能であり、タイミング故障解析には有効手法である。この技術で最初に報告された PICA (Picosecond Imaging Circuits Analysis)²³⁾ は、位置情報をもった二次元像を得るタイプであるが、現在は、二次元画像ではなく対象とする部分 (トランジスタ) をシャッターウィンドウで選択して、制限された領域からの発光を時間計測する TREM が一般的に用いられており、より簡便で短時間の計測を実現している²⁴⁾。

(c) OBIRCH (Optical Beam Induced Resistance Change)

レーザービームを照射すると、その部分は加熱され、電気抵抗の変動が生ずる。この現象を利用し、温度係数の特異な箇所の検出や電流経路の可視化を行う技術である^{25)–29)}。抵抗の変化は回路に定電圧または定電流を印加した状態で、電流の変化または電圧の変化によって検出される。レーザービームを走査しながらこれに同期をとって抵抗の変動を画像化している。このため、加熱温度や抵抗の温度係数の差によって、配線中に含まれるポイドや析出物を可視化することができる。

この技術が故障箇所絞り込みに最も効力を発揮するのは、電流の流れる経路を可視化できることにある。検出器に接続された配線経路のみが加熱による抵抗変動を検出するが、ほかの配線は加熱されても抵抗変動は検出されない。逆にその注目の回路に何らかのショートが存在してそこから系への電流の流入があれば、ショートしているほかの配線も可視化されることになる。

異常な電流経路やショート箇所の同定には大きな威力を発揮し、最近のデバイスの故障箇所絞り込みにおいては欠かせない技術の一つとなっている。最も一般的に活用されているのは、シリコンを透過する $1.3\mu\text{m}$ 波長の近赤外レーザーを用いた IR-OBIRCH (Infrared OBIRCH) で、配線の多層化の進行やフリップチップ化などによりチップ上面からの照射が困難になった最近の LSI に対しても、裏面からの解析でその有効性を発揮している。

(d) 電子ビームテスト (EBT: Electron Beam Testing)

電子ビームを試料表面に照射したときに、表面の電位分布に依存して二次電子の検出捕集効率が増加する原理 (電位コントラスト) を利用し、二次元的な回路配線の電位分布や任意のノードでの信号波形を観察する技術である³⁾。

装置 (電子ビームテスタ / EB テスタ) としては走査電子顕微鏡 (SEM) と同じ構成をもつが、電子ビームにパルスビームを用い、被測定 LSI デバイスは LSI テスタによって駆動される。LSI デバイスの駆動周期と同期させて電子ビームを発生し、ストロボスコーピックな計

測を実現している．任意の位相での電位コントラスト像や，回路上の任意のポイントでの信号波形をとることができる．

実際の故障箇所を絞り込みへの応用としては，故障デバイスと正常動作品との電位分布像の比較（差画像法）による故障位置の特定や，各ポイントでの信号波形の計測，CAD データからの期待値との比較を順に行いながら故障箇所を追跡するガイドドプローブ法，マージナルな故障に対して電源電圧を変化させながら電位分布像をとることで，変化する部分と変化しない部分の情報から故障位置を割り出すダイナミック故障画像法などがある．

電子ビームを用いるために空間分解能は高いものが得られるが，電子ビームがターゲット位置に到達しかつ電位情報をもった二次電子を検出するため，配線層数の多い多層配線デバイスでは適用が困難になってきている．

電子ビームテストを用いずに，通常の SEM や集束イオンビーム装置（FIB: Focused Ion Beam）のなかで，簡単に静的状態の電位コントラスト像を観察する手法として，VC-SEM（Voltage Contrast - SEM；電位コントラスト SEM）³⁰と VC-SIM（Voltage Contrast - Scanning Ion Microscopy；電位コントラスト走査イオン顕微鏡法）³²がある．配線の断線 オープン箇所を簡便に検出する手法として有効である．

(e) LVP（Laser Voltage Probing）レーザボルテージプロービング

プローブとして，シリコンのバンドギャップに相当するエネルギーのパルス IR レーザ（波長 1064 nm）を用いたもので，LSI チップの裏側からシリコン基板を透過させて，表面に形成された素子からの反射光を検出する技術である^{33)–35)}．この技術は，チップ裏面から内部の信号波形が観測でき，電子ビームテスト（EBT）では対応が困難になった幾層にもなる多層配線やフリップチップデバイスに対しても有効である．

この技術は，半導体や絶縁物に高い電界がかかるとバンドギャップナローイングにより光吸収端が長波長側に移る現象（フランツ・ケルデッシュ効果：Franz-Keldysh effect）を利用している．pn 接合にバンドギャップ近傍のエネルギーをもつ光を照射し，その pn 接合に電圧を印加すると，この効果によりその反射率は大きく変化する．この変化を検出することによって，pn 接合に印加された電圧の変化，すなわち印加されている信号波形を知ることができる．反射率の計測のほかに反射光の位相を検出することで同じ情報を得ることができる．信号波形は，電子ビームテスト（EBT）と同じストロボスコーピックな方法で検出している．

(f) RCI/EBAC（RCI: Resistive Contrast Imaging, EBAC: Electron Beam Absorbed Current）

電子ビームを走査・照射し，注入された電子によって形成される吸収電流をとらえて，画像化する技術である．RCI³⁶⁾がオリジナルな名称であるが，国内では EBAC³⁷⁾の呼称も使用されている．

電子は，照射されても電流検出器に到達しなければ像にはならないので，検出器につながった回路，基本的には配線が画像化される．経路の抵抗の大きさによって画像のコントラストが形成され，断線箇所が存在すれば，それより先（検出器より遠い方向）の画像は消えることになる．注目のノードにプロービングをして電流を検出することで，配線の断線や高抵抗箇所の検出のほか，ショートしているネットの特定が可能である．電子ビームの浸入深さを決める加速電圧を調整することで，多層配線にも対応可能である．

(g) SDL (Soft Defect Localization) /LADA (Laser Assisted Device Alteration)

レーザ照射されたときに、デバイスの動作に影響の現れる箇所を検出することによって、故障要因を含む箇所を絞り込む手法である。マージナルな動作不良（マージン不良）の故障箇所の絞り込みに用いられる。

通常は、シミュレーション（本章 3-9-2 節 (2) を参照のこと）でテスト条件を決めて行われる。レーザ照射による熱的影響を見る場合と、レーザ光励起によるキャリアの発生（OBIC: Optical Beam Induced Current）を利用する場合がある。

熱的影響としては、温度上昇に伴う異常部の抵抗の変化が信号の伝播遅延につながる。適当な照射条件を設定し、レーザを走査照射しながら、レーザ照射ポイントごとにデバイスの動作テストを行っていく。故障要因を含む箇所に照射された場合には、良否判定がほかのポイントと異なってくることで、その位置の特定が可能となる³⁸⁾。光励起によるキャリアの発生を利用する場合については、発生したキャリアの流入によりトランジスタの動作タイミングの変動が起きる³⁹⁾。

(h) SIL (Solid Immersion Lens: 固浸レンズ)

最近のデバイスの故障箇所絞り込みにはシリコン基板裏面からの解析が不可欠であり、そのためシリコンを透過する光が利用されている。しかし、光を使う限り空間分解能には限界がある。シリコンの光の吸収端波長は $1\mu\text{m}$ 付近であり、使用する光の短波長化による空間分解能の向上は望めない。この空間分解能の向上を実現したのが固浸レンズ (SIL)⁴⁰⁾⁻⁴³⁾ の利用である。

光の解像度は開口数 NA の逆数に比例し、 NA は、媒質の屈折率 n 、集光半角 θ とすると、 $\langle NA = n \cdot \sin \theta \rangle$ で決まる。このため屈折率 n の大きな固浸レンズを用いることで、短波長化することなく空間分解能の向上が図れる。屈折率が空気中の約 3.5 倍のシリコンで形成した半球型固浸レンズを用いた場合、 $1.3\mu\text{m}$ 波長の近赤外光でも、理論的には分解能 $0.19\mu\text{m}$ にも達する⁴²⁾。

光を用いた現在のデバイスの解析に SIL は不可欠技術であり、LVP、LSM（レーザ走査顕微鏡）、発光解析の高空間分解能化を実現している。

(i) 故障解析のための CAD ナビゲーション

LSI の微細化、回路規模の増大並びに多層配線層数の増加などによって、物理的手法による故障箇所絞り込みは、ますます、限られた層の極めて狭い局所的な視野から得られる情報に頼らざるを得なくなっている。更に、寸法の微細化は、故障箇所絞り込み装置の空間分解能を相対的に低下させている。このため、回路の接続関係も把握しにくく、故障位置絞り込み精度の低下につながっている。特にシリコン基板裏面からの解析においては、この問題は更に深刻さを増す。このため、CAD (Computer Aided Design) によるナビゲーションは不可欠のものとなっており⁴⁴⁾、特に故障箇所絞り込みを目的とした解析装置や、それを支援する FIB (集束イオンビーム装置) などの前処理加工装置には、所望の位置・ノードを示してくれる CAD ナビゲーションシステムを搭載しているものが多い。CAD ナビゲーションの入力には、LSI のレイアウトデータ、論理接続データ、GDS (Global Data Segment) データなどが使われる。

また、故障箇所絞り込みに用いる解析装置の出すデータと、解析での反応箇所との位置関係のあいまいさに加え、反応箇所が判明しても複数個の候補がある場合や、原因は別の箇所

にあり単なる結果としての反応を検出しているなど直接故障箇所を示すとは限らない場合もある。このような測定データを解析して、本当の故障箇所にとどり着くためには、CAD とリンクした解析（解析ナビゲーション）が極めて有効となる⁴⁵⁾。

3-9-5 故障解析 / 故障診断技術の動向

故障解析技術は LSI の信頼性及び歩留品質の向上のために重要度が増している。ソフトウェアを使った故障診断技術と物理的手法による故障箇所の絞り込み技術は、車の両輪として発展してきた。

故障診断技術は、スキャン設計を代表とする解析容易化設計の採用に伴い、LSI テスト結果から故障位置を推定する技術として用いられてきた。従来の故障診断技術は、論理情報のみから故障位置を推定するものであったが、複雑な物理欠陥に対応するため、レイアウト情報や回路情報も利用する技術が提案されてきた。こうした参照情報を解析の現場サイドで入手するには、設計の上流工程から故障解析を意識した情報整備が必要である。また歩留品質の向上のために大量故障診断技術も近年発展してきた。大量故障診断には本節で紹介した技術のほかに、プロセスの製造情報や設計情報なども用いて分析や解析を行うため、こうした情報を有機的に結びつけたシステム化が実用化の鍵とされている。

物理的手法による故障箇所絞り込み技術は、これまでも LSI デバイスの進化に応じてかなり活発な技術的ブレークスルーを行ってきている分野でもある。故障箇所を可視化した発光解析、OBIRCH、そして近赤外光によるシリコン裏面からの解析、裏面からの信号観察を実現した LVP、そしてそれらの光利用技術に高空間分解能を与えた SIL の応用技術など、これまで取り上げてきた技術のそのほとんどは、ここ四半世紀以内に新しいコンセプトを掲げて出現してきた技術でもある。ここでは取り上げなかったが、テラヘルツ電磁波放射の利用⁴⁶⁾など、現在も様々な技術開発がこの分野で活発に行われている。それらが実用化された段階では、故障箇所の絞り込みに更に新たな有効情報を付け加えていくと期待されている。

しかし、今後の故障箇所絞り込みにおいても引き続き課題となるものの一つに空間分解能の問題がある。いったんは SIL 技術の応用によりその問題の解決は図れたものの、更なる微細化に対する解決策は見つかっていない。現時点の技術の延長でこれらの問題の解決を考えた場合には、CAD 技術とリンクしたソフト的な支援によりその精度を高めていくことが重要となる。しかし、この分野のこれまでの発展の経緯をみているならば、この問題を抜本的に解決する画期的な技術ブレークスルーへの期待を捨てる必要もない。

また、故障診断技術と物理的手法の故障箇所絞り込み技術は、これまで作業者の手作業により情報を交換し活用してきたが十分とは言いがたい。今後、お互いが更に密結合した総合的な故障箇所の推定と絞り込みの技術開発が望まれる。

参考文献

- 1) LSI テスティング学会編, “LSI テスティングハンドブック,” オーム社, pp.262-272, 2008.
- 2) 半導体技術ロードマップ専門委員会, “2004 年度 STRJ WS 「半導体技術ロードマップ専門委員会」 第二部,” <http://strj-jeita.elisasp.net/strj/>, 2005.
- 3) P. Wohl, J.A. Waicukauski, S. Patel, and G. Maston, “Effective Diagnostics through Interval Unloads in a BIST Environment,” Proc. Design Automation Conference, pp.249-254, 2002.

- 4) S. Venkataraman, and S.B. Drummonds, "POIROT: a logic diagnosis tool and its applications," Proc. IEEE International Test Conference, pp.253-262, 2000.
- 5) 内角哲人・尾崎浩・清水良浩・松本賢和・山中宏樹・藤倉護, "機能テストフェイル情報を用いた故障診断手法," LSI テスティングシンポジウム会議録, pp.259-264, 2004.
- 6) L.-T. Wang, C.-W. Wu, and X. Wen, "VLSI test principles and architectures," Elsevier, 2006.
- 7) D.B. Lavo, and T. Larrabee, "Beyond the Byzantine generals: unexpected behavior and bridging fault diagnosis," Proc. IEEE International Test Conference, pp.611-619, 1996.
- 8) T.J. Vogels, W. Maly, and R.D. Blanton, "Progressive bridge identification," Proc. IEEE International Test Conference, pp.309-318, 2003.
- 9) Steven D. Millman, et al., "Diagnosing CMOS Bridging Faults with Stuck-at Fault Dictionaries," Proc. IEEE International Test Conference, pp.860-870, 1990.
- 10) J.B. Liu, A. Veneris, and H. Takahashi, "Incremental diagnosis of multiple open-interconnects," Proc. IEEE International Test Conference, pp.1085-1092, 2002.
- 11) X. Wen, T. Miyoshi, S. Kajihara, L.-T. Wang, K.K. Saluja, and K. Kinoshita, "On per-test fault diagnosis using the X-fault model," Proc. IEEE International Conference on Computer-Aided Design, pp.633-640, 2004.
- 12) S. Venkataraman, and S.B. Drummonds, "A technique for logic fault diagnosis of interconnect open defects," Proc. IEEE VLSI Test Symposium, pp.313-318, 2000.
- 13) T. Bartenstein, D. Heaberlin, L. Huisman, and D. Sliwinski, "Diagnosing combinational logic designs using the single location at-a-time (SLAT) paradigm," Proc. IEEE International Test Conference, pp.287-296, 2001.
- 14) Y. Sato, I. Yamazaki, H. Yamanaka, T. Ikeda, M. Takakura, "A Persistent Diagnostic Technique for Unstable Defects," Proc. Int. Test Conf., pp.242-249, 2002.
- 15) 吉澤豊・則松研二・佐藤康夫・二階堂正人・真田克, "スイッチング・レベル・シミュレーションを用いたセル内故障診断技術 - 故障動作と診断," LSI テスティングシンポジウム会議録, pp.231-236, 2005.
- 16) M.E. Amyeen, D. Nayak, and S. Venkataraman, "Improving precision using mixed-level fault diagnosis," Proc. IEEE International Test Conference, Paper 22.3.1-10, 2006.
- 17) 益子洋治, "最近のLSI故障解析技術動向," LSI テスティングシンポジウム会議録, pp.261-266, 2005.
- 18) N. Khurana and C-L. Chiang, "Analysis of Product Hot Electron Problems by Gated Emission Microscopy," IEEE Proc. of Int. Reliability Phys. Symp., pp.189-194, 1986.
- 19) N. Khurana and C-L. Chiang, "Dynamic Imaging of Current Conduction in Dielectric Films by Emission Microscopy," IEEE Proc. of Int. Reliability Phys. Symp., pp.72-75, 1987.
- 20) H. Ishizuka, M. Tanaka, H. Konishi, and H. Ishida, "Advanced Method of Failure Analysis using Photon Spectrum of Emission Microscopy," Proc. Int. Symp. for Testing and Failure Analysis, pp.13-19, 1990.
- 21) 吉田岳司・小野山歩・小山徹・小守純子・益子洋治, "高感度裏面エミッション検出によるウエハレベル故障分布解析," LSI テスティングシンポジウム会議録, pp.137-142, 2002.
- 22) J.A. Kash and J.C. Tsang, "Dynamic internal testing of CMOS circuits using hot luminescence," IEEE Electron Dev. Lett., vol.18, Issue7, pp.330-332, 1997.
- 23) M.K. Mac. Manus, J.A. Kash, S.E. Steen, S. Polonsky, J.C. Tsang, D.R. Knebel, and W. Huott, "PICA: Backside failure analysis of CMOS circuits using Picosecond Imaging Circuits analysis," Microelectronics Reliability 40, pp.1353-1358, 2000.
- 24) 日鍋伸二・松本賢和・小松宗雄・藤倉護・花崎裕一・嶋瀬朗・内角哲人・大久保忠之・相馬稔, "時間分解発光解析の評価と解析事例," LSI テスティングシンポジウム会議録, pp.315-320, 2004.
- 25) K. Nikawa and S. Tozaki, "Novel OBIC observation method for detecting defects in Al stripes under current stressing," Proc. Int. Symp. for Testing and Failure Analysis, pp.303-310, 1993.
- 26) K. Nikawa and S. Inoue, "New laser beam heating methods applicable to fault localization and defect

- detection in VLSI devices,” IEEE Proc. of Int. Reliability Phys. Symp., pp.364-354, 1996.
- 27) 名村高・藤井眞治・山下洋・大橋誠治, “OBIRCH 法による大規模配線パターン欠陥検出,” LSI テスティングシンポジウム会議録, pp.193-197, 2001.
 - 28) 二川清, “LSI 故障解析技術のすべて,” 工業調査会, pp.75-103, 149-157, 2007.
 - 29) 二川清・井上彰二・森本和幸・曽根伸哉, “IR-OBIRCH 手法を用いた半導体デバイス故障解析事例,” LSI テスティングシンポジウム会議録, pp.181-186, 1998.
 - 30) 裏克己・藤岡弘, “電子ビームテスティングハンドブック(電子ビーム研究 第7巻: 日本学術振興会荷電粒子ビームの工業への応用第132委員会第98回研究会資料),” 大阪大学電子ビーム研究施設, 1987.
 - 31) 中前幸治, “5.6 電子ビームテスティング (EBT) 法とその故障解析への応用”, LSI テスティングハンドブック, LSI テスティング学会編, オーム社, pp.318-325, 2008.
 - 32) G.B. Ang, Y.N. Hua, S.K. Loh, Yogaspari, S. Redkar, “Application of passive voltage contrast and focused ion beam on failure analysis of metal via defect in wafer fabrication,” Physical and Failure Analysis of Integrated Circuits, 2001. IPFA 2001. Proceedings of the 2001 8th International Symposium, pp.107-111, 2001.
 - 33) M. Paniccia, T.M. Eiles, V.R. M. Rao, and W.M. Yee, “Novel optical probing technique for flip chip packaged microprocessors,” Int. Test Conf.1998, pp.740-747, 1998.
 - 34) M. Paniccia, R.M. Rao, and W.M. Yee, “Optical Probing of Flip Chip Packaged Microprocessors,” J.Vacuum Science and Technologies, vol.B16, pp.3625-3630, 1998.
 - 35) 吉田映二・小山徹・小守純子・益子洋治, “裏面プロービング技術の検討,” LSI テスティングシンポジウム会議録, pp.94-99, 2000.
 - 36) C.A. Smith, C.R. Bagnell, E.I. Cole, F.A. DiBianca, D.G. Johnson, W.V. Oxford, and R.H. Propst, “Resistive contrast imaging: A new SEM mode for failure analysis,” IEEE Transaction Electron Devices, vol.ED-33, no.2, pp.282-285, 1986.
 - 37) 水越克郎・小山田太郎・松本賢和・寄崎眞吾・嶋瀬朗・真島敏幸・小柳肇・野副真理, “電子ビーム吸収電流解析による LSI 配線の不良位置特定,” LSI テスティングシンポジウム会議録, pp.213-218, 2003.
 - 38) M. Bruce, V. Bruce, D. Eppes, J. Wilcox, and E. Cole, P. Tangyunyong, and C. Hawkins, “Soft Defect Localization (SDL) on ICs,” Proc.28 Int. Symp. for Testing Failure Analysis, pp.21-27, 2007.
 - 39) L.A. Rowlette and T.M. Eiles, “Critical Timing analysis in microprocessors using near-IR laser assisted device alteration (LADA),” Proc. IEEE Int. Test Conference, pp.264-273, 2003.
 - 40) S.M. Mansfield and G.S. Kino, “Solid Immersion microscope,” Appl. Phys. Lett., vol.57, pp.2615-2616, 1990.
 - 41) S.B. Ippolito, B.B. Goldberg, and M.S. Unlu, “High spatial resolution subsurface microscopy,” Appl. Phys. Lett., vol.78, p.4071, 2001.
 - 42) T. Koyama, E. Yoshida, J. Komori, Y. Mashiko, T. Nakasuji, and H. Katoh, “High resolution backside fault isolation technique using directly forming silicon substrate into solid immersion lens,” International Reliability Physics Symposium, pp.529-535, 2003.
 - 43) 吉田岳司・小山徹・小守純子・益子洋治, “SIL プレートによる高分解能故障解析,” LSI テスティングシンポジウム会議録, pp.321-325, 2004.
 - 44) 嶋瀬朗・真島敏幸・渡曾慎一・川鍋伸二・松本賢和・内角哲人, “発光 / OBIRCH 解析と CAD ツールとの連動による不良位置特定の効率向上,” LSI テスティングシンポジウム会議録, pp.249-254, 2005.
 - 45) 嶋瀬朗・内角哲人・佐伯光章・渡曾慎一・鈴木猛司・真島敏幸・堀田和宏・寺田浩敏, “解析ナビゲーションシステムの機能向上,” LSI テスティングシンポジウム会議録, pp.247-252, 2006.
 - 46) 山下将嗣・大谷知行・斗内政吉・三浦克彦・中前幸治・二川清, “レーザーテラヘルツエミッション顕微鏡における LS-TEG 観察,” LSI テスティングシンポジウム会議録, pp.347-350, 2007.