

## スキャン構造を意識したテストパターン生成

### A TEST GENERATION METHOD OF DIGITAL CIRCUITS FOR GATE ARRAY WITH AUTOMATIC MODIFICATION TO SCAN DESIGN

野田浩明、松本敏行、藤原紳一、藤本徹哉、奥村憲三、神戸尚志

H.NODA,T.MATSUMOTO,S.FUJIWARA,T.FUJIMOTO,K.OKUMURA,T.KAMBE

シャープ株式会社

SHARP CORPORATION

あらしし ゲートアレイの開発期間短縮を目的として自動化がおこなわれているテスト設計を効率化するシステムを開発した。本システムではスキャン方式に基づいたテストパターンならびに検査回路の自動生成を行う。本稿ではスキャン構造および検査回路の自動生成を考慮したテストパターン生成手法、また検査回路をゲートアレイ上に実現するための構成手法、レイアウト手法をしめす。

**Abstract** Though gate array design is almost automated, test design still depends on designer's skill. In this paper we describe the efficient CAD system for test design of digital circuits for gate array. A test generation method with automatic circuit modification to scan design, and a layout method of additional circuits for test are discussed.

#### 1. はじめに

LSIの製造技術が向上し、今日では最も設計が容易なLSIであるゲートアレイによっても数万ゲートにおよぶ大規模で複雑な回路の実現が可能となった。このためゲートアレイの需要は従来に増して高まり、またその適用分野である電子機器の製品開発サイクルの短期化とあいまって、ゲートアレイの開発期間をさらに短縮することが強く望まれている。

元来ゲートアレイはすでに用意されたの基本ゲートを用いてこれを相互に結線することによって必要な機能を実現するものであるため、その設計は自由度が低いもののフルカスタムLSIに比べて非常に短い期間で行うことができる。論理設計は、用意されたライブラリを用いた設計となり、レイアウト設計はカスタマイズが必要とされる部分は配線層だけであり、またそのレイアウトモデルの簡素さから完全な自動化がなされている。

しかしLSIを製造するにあたっては、製造されたLSIを良品と不良品に識別するためのテストパターン(検査入力)を設計する必要がある。このテストパターンの設計は自動化がおこなわれていまだに人手に負うところが多く、ゲートアレイを用いた場合でもフルカスタムLSIの場合と変わらぬ労力を要している。数万ゲートにまで大規模化した回路では、高い故障検出率を示すテストパターンを設計するために多大な時間を費やしてしまうため、論理設計、レイアウト設計の期間が短いゲートアレイでは開発期間のうち多くをこのテストパターン設計が占めてしまっている。

このようにテストパターン設計が開発期間短縮の最も大きな障害のひとつとなっている今日、これを効率化することによってゲートアレイの開発期間を短縮するシステムを開発した。本稿では、本システムの概要、本システム中で用いられた手法ならびに、適用結果について報告する。

## 2. システム概要

テストパターン設計の自動化に関しては、これまで種々の自動生成アルゴリズムが提案されてきている。しかしこれらは、数多くの内部状態を持つ大規模な順序回路をそのまま取り扱うには、自動生成にかかるコスト、生成されるテストパターン長を考えると実用的なものであるとは言えない。そこで最近では、回路自体に何らかのテストコストを削減する構造を持たせた設計を行う検査容易化設計が行われる。その代表的なものとして、スキャン方式があげられる。このスキャン方式の設計が行われた回路では、内部状態を外部端子から観測および制御することが可能となるため、テストパターンを設計する場合には、順序回路を等価的な組み合わせ回路として取り扱えることになる。したがってこの種の回路に対しては、これまで提案されてきた組み合わせ回路に対するアルゴリズムによって実用的なコストでの自動生成が可能となる。反面、スキャン方式の設計を行った場合、記憶素子の内部状態を外部端子から観測、制御する機能を追加するためのハードウェアの増加、またこの機能を動作させるためのテストパターンの増加、スキャン構造を意識した回路設計等、ハードウェアコスト、テストコスト、設計コストの各々の面での負担が増加する。そこでスキャン方式の設計を行う場合にも、これらのコスト増加を何らかのかたちで抑制する必要がある。

ここで人手によるテストパターン設計について考察すると、設計者は、回路を機能単位でモジュールに分割し各モジュールの機能をふまえたテストパターンを設計する。このため回路が効率よく活性化されゲートレベルの回路で自動生成されたものに比べてより短いテストパターンで多くの故障を検出することが可能となる。しかし設計者は、各モジュールを機能面からとらえ回路構造的にはブラックボックスとして取り扱っているために、モジュール内部の故障を検出するテストパターンを設計する場合には多くの時間を費やし、自動生成されたものよりも特に効率的であるとは言えない。また回路の大規模化に伴って抽象的なレベルで設計が行われ、詳細な回路構造が設計者からますます見えにくくなり、モジュール内部の故障を検出するテストパターンを設計することは、実質的には不可能となってきている。

そこで本システムでは、人手設計による高効率性および自動生成による完全性の両立をねらった。すなわち、設計者の機能を主眼においたテストパターンを利用し、さらにこのテストパターンでは検出できない故障に対してのみテストパターンを自動生成することでテストコストの増加の抑制を図った。またハードウェアコスト、設計コストの増加を抑えるために、自動生成されたテストパターンを動作させるために必要なスキャン構造はテストパターン生成の過程で得られる情報をもとに自動的に生成される。本システムは、以下の特徴を持つ。

- (1) 設計者が最も時間を費やす回路細部の故障に対してテストパターンを自動生成するため、テストパターン設計期間を大幅に短縮できる。
- (2) スキャン回路を自動的に生成するため設計者は、回路設計に際してスキャン構造を意識する必要がない。
- (3) テストパターン生成の過程で、スキャン構造を決定するため、ハードウェアコストの増加及び、テストパターン長の増長を抑制できる。
- (4) 回路のレイアウト設計においてもスキャン構造が考慮されるため、既存の回路のレイアウトに与える影響が小さい。

本システムのシステムは、

- (1)対象回路の検証および故障の選択
- (2)テストパターンの自動生成
- (3)検査回路の自動生成
- (4)テストパターンの実回路への写像

の処理手順で行われる。各処理の詳細については次章以降で順次述べていく。

## 3. 対象回路の検証および故障の選択

多種多様な回路がゲートアレイによって実現されるが、その回路構造によってはスキャン方式を適用できない場合がある。本システムは現在のところ同期式のフリップフロップ型の記憶素子によって構成される回路を対象としている。また記憶素子にスキャン機能を追加した場合には記憶素子のタイミングに若干のずれが生じるため、テストパターンを自動生成するに先だって対象とする回路を構造、動作面で本手法との整合性を検証しておく必要がある。

構造的には、

(1) Dラッチ等のスキャン機能を追加できない記憶素子の使用。

(2) NAND,NOT等の基本論理ゲートの非同期帰還閉路による記憶回路の構成。

また動作的には、

(1) 各記憶素子のセットアップ、ホールドタイムのマージン不足。

(2) クリティカルなレーシング状態。

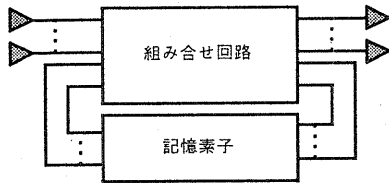
(3) クロックのスキューマージン不足。

が検証され、これらの違反が検出された部分回路へのスキャン方式の適用を禁止する[1]。

つづいて設計者のテストパターンを用いて故障シミュレーションを行い、この時未検出となった故障のみをテストパターン自動生成の対象とする[2]。

#### 4. テストパターンの自動生成

対象とする回路は【図-1】のように記憶素子部分とこれに挟まれた組み合わせ回路部分にモデル化される。

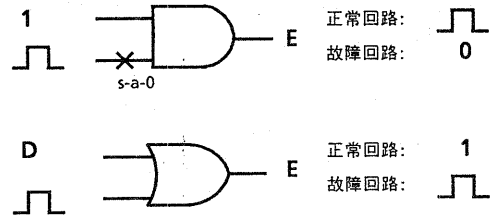


【図-1】順序回路モデル

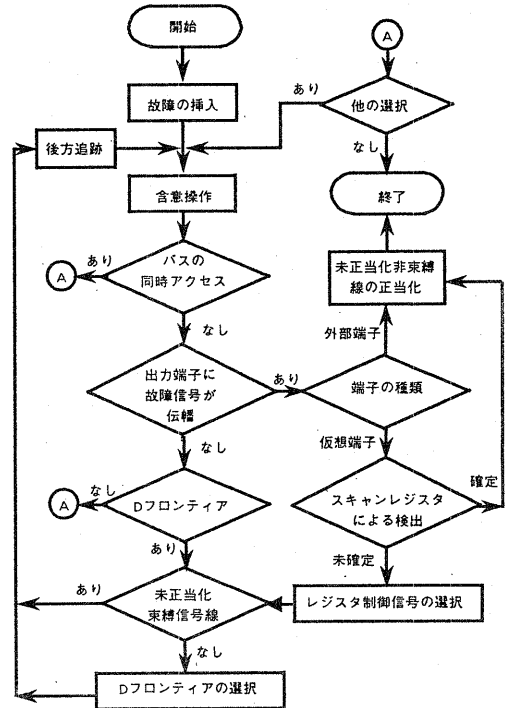
原則として組み合わせ回路部分で取り扱える素子は、

NAND NOR AND OR NOT XOR

の基本論理ゲートであるが、これにCMOS回路で一般的に用いられるトライステート素子、バス素子を加え実回路へ適用性を高めている。組み合わせ回路部分と記憶素子部分との境界部分については、回路検証時にスキャン機能の追加が可能と判断された記憶素子に関しては、スキャンレジスタとして取り扱い、仮想入力端子もしくは仮想出力端子を設定する。それ以外の記憶素子に関しては、常に不定値を出力する開放入力端子および故障検出が不可能な開放出力端子を設定する。このようにして得られたトライステート素子、バ



【図-2】クロック故障信号



【図-3】テストパターン生成フロー

ス素子を含んだ組み合わせ回路に対してテストパターンの生成を行う。生成アルゴリズムは、組み合わせ回路のテストパターン生成アルゴリズムであるFANアルゴリズム[3]にもとづき、回路の後方追跡とこれに伴う含意操作によってテストパターンを生成していく。組み合わせ回路から拡張されたスキャンレジスタ、トライステート素子、バス素子に関してはその動作に対応した処理が必要となる。例えば仮想出力端子に故障が伝搬しただけではその故障は検出できない。外部端子へスキャンアウトできるようにレジスタ内に取り込む処理が必要となる。

そこで論理演算モデルとして通常信号値0、1、故障信号値D、 $\bar{D}$ にトライステート、バス素子の論理を取り扱うためのハイインピーダンス値Z、スキャンレジスタに内部信号をラッチさせるクロック信号値C、 $\bar{C}$ 、クロック故障信号値E(正常回路でクロック信号、故障回路でレベル信号、もしくはその逆)、Eの反転値 $\bar{E}$ 【図-2】を加え、これらの組み合わせによる多値論理モデルを用いることで、効率的なテストパターン生成を行っている【図-3】。次に組合せ回路から拡張された処理について説明する。

(1) スキャンレジスタのデータ入力端子に故障信号が伝搬した場合は同レジスタのクロック端子にクロック要求を発生させクロック経路を活性化する入力の組合せを一致操作により求める。この操作により伝搬した故障信号を同レジスタ内にラッチすることが可能となり、ラッチされた故障信号をスキャンアウトすることによって故障検出が可能となる【図-4-a】。

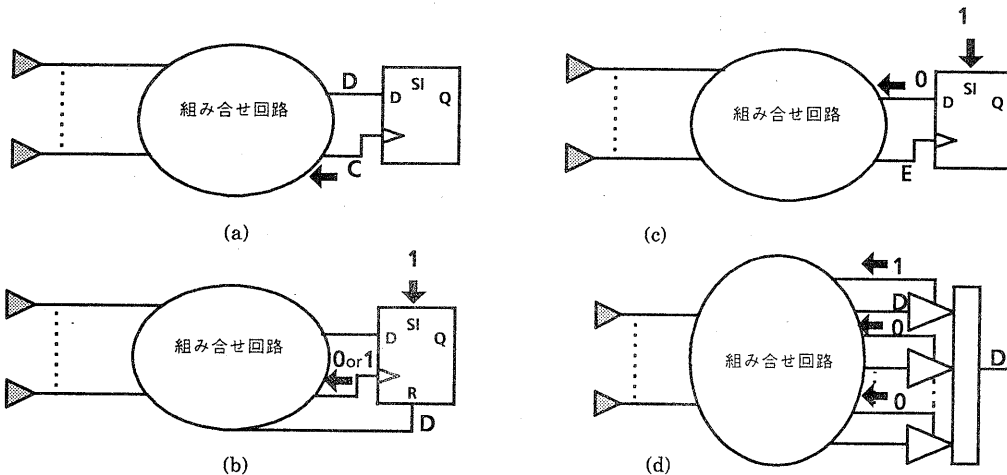
(2) スキャンレジスタのセット/リセット端子に故障信号が伝搬した場合は同レジスタのセット/リセット動作に正常回路と故障回路との間で差異が生じるため、同レジスタに、リセット端子に故障信号が伝搬した場合は1を、セット端子に故障信号が伝搬した場合は0をスキャンインしておけば同レジスタの内部状態をスキャンアウトすることによって故障検出が可能となる【図-4-b】。

(3) スキャンレジスタのクロック端子にクロック

故障信号が伝搬した場合は同レジスタのラッチ動作に正常回路と故障回路との間で差異が生じるため、同レジスタのデータ入力端子に0もしくは1のいずれかの値を設定しその反転値をスキャンインしておけば同レジスタの内部状態をスキャンアウトすることによって故障検出が可能となる【図-4-c】。

(4) トライステート、バス素子のデータ端子に故障信号が伝搬した場合はこれに対応したコントロール端子に1を、その他のコントロール端子に0を設定し故障を伝搬させる【図-4-d】。これらの素子に関してはバスの同時アクセスが生じた場合には回路破壊につながる可能性があるため、すべての故障に対してバスの同時アクセスを回避する値を各々のコントロール端子に設定する。またテストパターン生成中にバスの同時アクセスが生じた場合はその入力の組合せをとり止め他の入力の組合せを検索する。

本手法では、生成時に起きる種々の選択の際にその選択の尺度となる各信号線の可観測性、可制御性にその信号線を観測、制御した場合に増加するハードウェアコストの要素を加えた。生成時にこの可観測性、可制御性を各記憶素子へのスキャン機能の追加状況にあわせて更新することによって検査のために付加される回路の規模を制御している。このため可観測性、可制御性はテストパターン生成過程で頻繁に更新されるためより簡便なGOLDSTEINの組合せ回路に関する尺度[4]をもとに次のように定義した。



【図-4】スキャンレジスタ、バス素子における故障の取扱い

任意の信号Xについて1-可制御性、0-可制御性、可観測性を各々 $C^1(X)$ 、 $C^0(X)$ 、 $CO(X)$ と表す。外部もしくは仮想入力端子に接続した信号Iについて $C^1$ 、 $C^0$ は、

$$C^1(I) = 1 + a(I)$$

$$C^0(I) = 1 + a(I)$$

外部もしくは仮想出力端子に接続した出力信号OについてCOは、

$$CO(O) = a(O)$$

とする。この時、他の内部信号については、例えば入力に $X_1$ 、 $X_2$ 、 $X_3$ を持つ3入力ANDの出力信号Yの $C^1$ 、 $C^0$ 、 $X_1$ のCOは、

$$C^1(Y) = C^1(X_1) + C^1(X_2) + C^1(X_3) + 1$$

$$C^0(Y) = \min[C^0(X_1), C^0(X_2), C^0(X_3)] + 1$$

$$CO(X_1) = CO(Y) + C^1(X_2) + C^1(X_3) + 1$$

で表される。ここで $a(X)$ はその時点で入力、出力信号Xを観測、制御した場合のハードウェアコスト関数で、外部入力端子、外部出力端子もしくはすでに対応した記憶素子にスキャン機能が追加されている仮想入力端子、仮想出力端子に接続した信号Xに関しては、

$$a(X) = 0$$

であり、これ以外の仮想入力端子、仮想出力端子に接続した信号Xに関しては、

$$a(X) = K$$

で表される。Kは、記憶素子にスキャン機能を追加した場合に増加するハードウェアコストである。

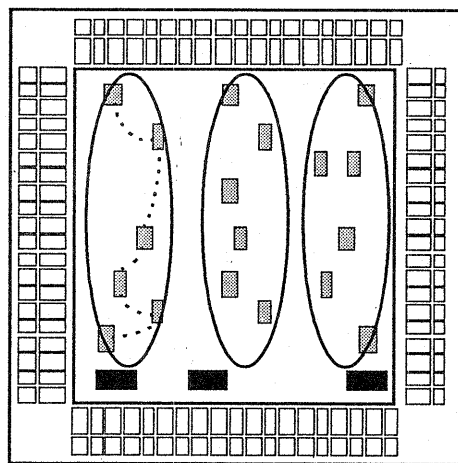
さらに本システムではテストパターン生成後には、故障シミュレーションを行い、同時に検出可能な故障を検索しテストパターン長および生成コストを抑制している。

## 5. 検査回路の生成

スキャン方式を利用したテストを行う場合には、記憶素子は、通常の動作以外に、内部状態のスキャンイン、スキャンアウトを行わなければならない。このためスキャン機能を持つ記憶素子をシフトレジスタとして構成し、これに対応したクロック、セット/リセット信号を供給する必要がある。そこでクロック、セット/リセット信号を供給している回路を解析することによって各記憶素子をクロック系統、セット/リセット系統ごとに分類し、クロックスキューを回避す

るように、これらの信号を制御する回路を生成する。一方、スキャン機能を持つ記憶素子はテストパターン生成過程で自動的に決定されているためシフトレジスタとして構成するにあたってその順序付けが成されていない。さらにシフトレジスタを構成するための配線は回路全体におよび、その結線要求はゲートアレイ等の自動レイアウトシステムを利用する場合、そのレイアウトに少なからぬ影響を与える。そこで本システムでは、シフトレジスタとして構成するにあたって次の手法をとった。まず、付加される回路を見積り、一旦自動配置を行う。次に自動配置によって得られる配置情報等を用いてスキャンレジスタの順序付けを行いシフトレジスタの構成を決定する。最後に付加される回路を再配置し配線処理を施す。このように論理、レイアウトの両面にわたって考慮し検査回路を生成する。以下でその処理の詳細について説明する。

- (1) テストパターン生成の過程で内部状態の観測もしくは制御が必要となった記憶素子は、セルライブラリ中の対応したスキャンレジスタと置換することによってスキャン機能が付加される。また、クロック、セット/リセット信号供給回路を解析した時に生成された制御回路を付加する。



■ スキャンレジスタ      ○ スキャンバス  
 ■ 領域保存疑似セル      ..... スキャンバス

【図-5】 スキャンバスレイアウト

(2) パッドの配置、制限等をもとに、挿入可能なスキャンパスの本数、ズキャンアウトに用いるパッドを決定する。次に、各スキャンパスを動作させるのに必要な付加回路を見積り、これに相当した疑似セルを発生させ、既存の回路と合わせて自動配置を行う。この時、スキャンレジスタは検査回路としての結線要求を持たないため、これに影響されない配置が行われる。また見積られた付加回路のための疑似セルは、対応したスキャンアウトパッドに対して疑似的に結線要求を発生させるため、その近辺に配置される【図-5】。

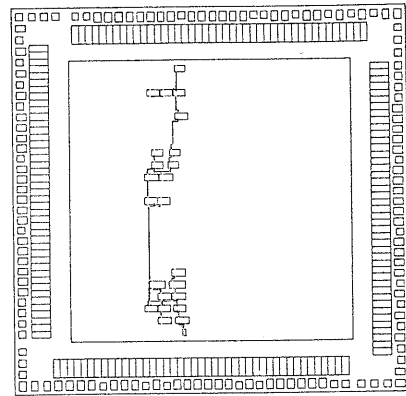
(3) スキャンレジスタをその配置分布から、スキャンパスを構成する配線がセル行に対して直交するように、スキャンパスへの割り付け、順序付けを行い、スキャンパスの経路を決定する【図-5】。これにより、検査回路を付加したことによる配線チャンネルの混雑度増加を緩和している。

(4) 生成されたスキャンパスの動作を制御する回路を生成する。各スキャンパスが要求するクロック信号、制御信号の種類、駆動能力を求め、これらを満たすように各々の信号を分配供給する回路を生成する。

(5) 生成された検査回路のうち実際にセルが配置されていないものを、自動配置の際に疑似セルを配置し保存しておいた領域に割り付け、この部分だけの再配置を行う。この時点で全てのセルの配置及び、結線要求が確定されるので、従来通りに自動配線を行う。

## 6. テストパターンの実回路への写像

自動生成された組み合わせ回路部分のテストパターンを、それぞれの故障検出法、各スキャンレジスタのスキャンパスへの割り付け順序付けに従って、スキャンイン、ラッチ、スキャンアウトの3つの動作を制御する信号を加えスキャン構造を持つ実回路で動作するテストパターンに合成する。またスキャン回路検査用のテストパターンも合わせて発生させる。設計者のテストパターンにこれで得られたテストパターンを加え、故障シミュレーションを行い、最終的タイミング及び故障検出率の評価を行う。



【図-6】 スキャンパスレイアウト例

## 7. 適用結果

本システムを3000ゲートクラスの回路に適用した結果、20%のゲート使用率の増加で、故障検出率は、80%から98%に向上した。【図-6】に生成したスキャンパスの構成を示す。

## 8. むすび

本稿では、ゲートアレイにおけるスキャン方式の導入、およびこれを利用したテストパターン自動生成の一手法を示した。本稿で報告したシステムを適用することで、ゲートアレイの設計者は、スキャン構造を意識することなく論理設計が行え、テストパターンの自動生成を利用することで短期間で高品質なテストパターンの設計が可能となった。このようにテスト設計の効率化、自動化を図ることは、ゲートアレイのみならず、数多くのLSIに開発期間を短縮、高信頼性をもたらすものと考えられる。

### 参考文献

- [1]藤本他、「制御信号系にもとづくタイミング検証」、情報処理学会、第40回設計自動化研究会、1987
- [2]松本他、「未検出故障解析を容易にした故障シミュレータ」、電子情報通信学会、総合全国大会論文集、1987
- [3]H.Fujiwara, "On the acceleration of test generation algorithm", IEEE Trans. Comput., Vol.C-32, pp.1137-1144, Dec.1983.
- [4]L.H.Goldstein, "Controllability/Obseavability analysis of digital circuit", IEEE Trans. CAS, Vol.CAS-26, pp.685-693, Sep.1979