

解説

PLA におけるテスト容易化設計法†



山田 輝彦†

1. はじめに

数十万トランジスタ以上の集積度をもつ論理 VLSI を短期間で安価に実現するためには、計算機を用いて論理設計からテスト設計までの処理をできるだけ自動化することが肝要である。規則的な構造をもつ RAM, ROM, PLA (Programmable Logic Array) を多用した設計方法は、主として論理ゲートやフリップフロップを用いる従来の設計方法よりも、設計の自動化が容易であり、仕様の変更にも柔軟に対処できる。また、テスト容易化設計を施すことが簡単になるので、テスト費用を大幅に低減することも可能である。このため、最近の 32 ビットマイクロプロセッサの多くがこの方法で設計されている¹⁾。

PLA は、マイクロプロセッサのマイクロプログラム部や制御部のように論理が複雑で修正・変更の多い部分に用いられている。マイクロプログラム部や制御部はテストのもっとも困難な部分ではあるが、PLA の物理的・論理的な構造の単純さを活用してテスト容易化設計を施せば、高品質のテストが比較的簡単に行える。これが PLA を用いるもう一つの大きな理由である。

本稿では、これまでに提案されている PLA のテスト容易化設計法をテスト入力の性質に着目して分類し、それぞれの代表的な設計法を紹介する。次に、各設計法を比較・評価し、用途に応じた設計法の選択が必要であることを示す。最後に、最適な設計法を選択するためのエキスパートシステムを簡単に紹介する。

2. PLA とその故障モデル

PLA は AND-OR 2 段の論理式を実現するものであり、一般にデコーダ、AND アレイおよび OR アレイで構成される²⁾。

図-1 は、1 ビットデコード方式の PLA を例示したものである。各デコーダでは、入力の肯定信号 x と否定信号 \bar{x} が作られ、それぞれが AND アレイの水平方向の信号線（以下ではビット線という）に出力される。アレイ部分の垂直方向の信号線（以下では積項線という）では、AND デバイス（●印）を介して接続されているビット線の信号の論理積がとられる。すなわち、各積項線では、左から順に積項 $x_1x_2x_3x_4$, $x_1\bar{x}_2x_4$, $x_1\bar{x}_2\bar{x}_3\bar{x}_4$, $\bar{x}_1\bar{x}_2\bar{x}_3$, $\bar{x}_1\bar{x}_3x_4$, $\bar{x}_1x_2x_3$ が実現されている。OR アレイの水平方向の信号線（以下では出力線という）では、OR デバイス（×印）を介して接続されている積項線の信号の論理和がとられる。したがって、各出力線では次の論理式が実現されている。

$$f_1 = x_1x_2x_3x_4 + \bar{x}_1\bar{x}_3x_4 + \bar{x}_1x_2x_3$$

$$f_2 = x_1x_2x_3x_4 + x_1\bar{x}_2x_4 + \bar{x}_1\bar{x}_2\bar{x}_3$$

■ $f_3 = x_1\bar{x}_2\bar{x}_3\bar{x}_4 + \bar{x}_1\bar{x}_3x_4$

PLA のテストを考える場合、次に示すような故障モデルを対象とする^{3), 4)}。

1. 交点故障 (crosspoint fault): AND デバイスおよび OR デバイスが、あるはずの所になかったり、逆にならないはずの所にあったりする故障
2. 縮退故障 (stuck-at fault): 信号線（デコーダ内の信号線、ビット線、積項線、出力線）の値が 0 または 1 に固定される故障
3. 短絡故障 (bridging fault): 隣接あるいは交差す

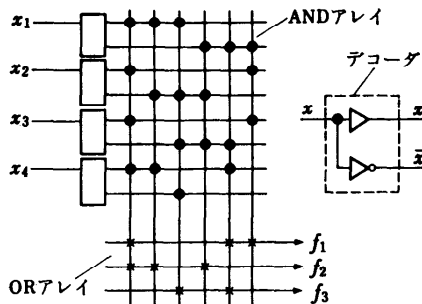


図-1 通常の PLA の構造

† Testable PLA Design Methodology by Teruhiko YAMADA (Department of Computer Science, Meiji University).

†† 明治大学理工学部情報科学科

る信号線が短絡して、ワイヤード AND または OR として機能する故障

交点故障は PLA 特有の故障であるが、交点故障のテストにより縮退故障と短絡故障の多くを検出できることが知られている。そのため、交点故障を主対象としてテストを考えることが多い。また、通常は、単一故障（故障が一つしか存在しないこと）が仮定される。

3. 代表的なテスト容易化設計法

VLSI に埋め込まれた PLA のテストをチップの I/O ピンだけから実行することは必ずしも容易ではない。したがって、テスト入力の発生と出力の判定ができるだけチップ内で行えるテスト方式（組み込み自己テストが可能な方式）を採用することが望ましい。

出力の判定については、出力の応答を逐次的に比較するのではなく数〜数十ビットに圧縮し、その結果を期待値と比較する方法が適していると思われる。これに対して、どのようなテスト入力を用いるかは、テスト入力の生成・印加の容易さに加えて、対象とする故障のモデルとその検出率、許容される付加回路量、テスト時間などを考慮して決定する必要がある。そこで、これまでに提案されている多数のテスト容易化設計法を、テスト入力の性質に着目して分類し、その代表的なものを紹介する。

(1) 万能テスト方式

図-2 は、図-1 の PLA に次の回路を付加したものである。

1. 各ビット線上的 AND デバイスの数に奇数パリティをもたせるための積項線 α

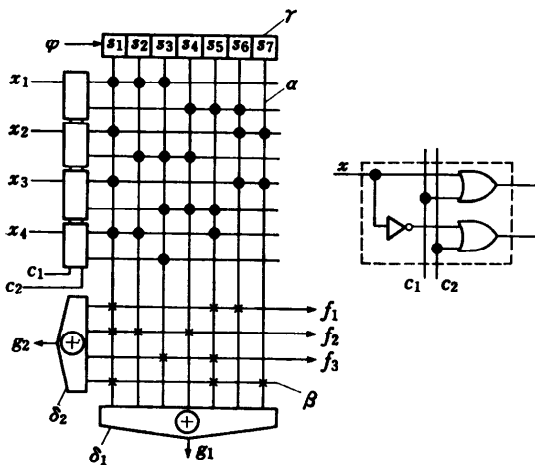


図-2 万能テスト方式の実現例

表-1 $n=4, m=7$ の場合の万能テスト

x_1, x_2, x_3, x_4	c_1, c_2	$s_1, s_2, s_3, s_4, s_5, s_6, s_7$	g_1, g_2
-----	--	0 0 0 0 0 0 0	0 0
0 0 0 0	1 0	1 0 0 0 0 0 0 0 0 0 0 0 0 1	1 1 1 1
1 1 1 1	0 1	1 0 0 0 0 0 0 0 0 0 0 0 0 1	1 1 1 1
-----	1 1	1 0 0 0 0 0 0 1 1 1 1 1 1 0	1 - 0 -
0 1 1 1 1 1 1 0	0 1	1 1 1 1 1 1 1	0 - 0 -
1 0 0 0 0 0 0 1	1 0	1 1 1 1 1 1 1	0 - 0 -

2. 各積項線上的 OR デバイスの数に奇数パリティをもたせるための出力線 β

3. 積項線の任意の一つを活性化するためのシフトレジスタ γ

4. ビット線の任意の一つを活性化するためのデコーダの制御入力 c_1, c_2

5. 各ビット線上的 AND デバイスの数のパリティを検査するパリティチェッカ δ_1

6. 各出力線上的 OR デバイスの数のパリティを検査するパリティチェッカ δ_2

図-2 の PLA に対して表-1 に示すテストを実行すれば、 g_1, g_2 を観測するだけで単一の交点故障と縮退故障のすべてが検出できる。

表-1 のテストは図-2 と同じサイズの任意の PLA に適用可能であり、任意のサイズの PLA に対しても同様にテストを作成することができる。外部入力の本数を n 、積項線の本数を m とすると、テストの本数は $2n + 3m$ となる。

上記の設計法は藤原・樹下⁵⁾ によって提案されたものであり、類似の設計法が Hong・Ostapko⁶⁾ によっても提案されている。これらの研究が先駆けとなり、以後 PLA のテスト容易化設計法の研究が盛んになった。

上記の設計法に対して多重故障の検出やパリティチェッカ δ_1 の除去などの工夫がなされ⁷⁾⁻⁹⁾、さらにテスト出力の累積パリティに規則性をもたせて組み込み自己テスト (built-in self-test) を可能にした設計法¹⁰⁾が提案された。組み込み自己テスト可能なその他の設計法として、ウォーキング (walking) 0 および 1

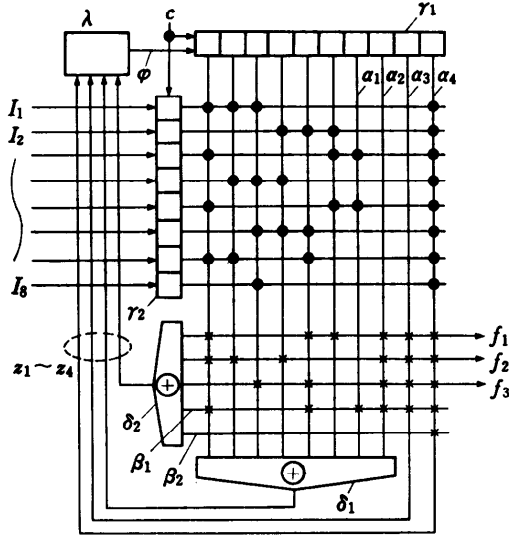


図-3 自律テスト方式の実現例

パターンをテスト入力とする設計法^{11),12)}も提案されている。

(2) 自律テスト方式

出力信号を入力側に帰還させる手段を有する回路を自律回路 (autonomous circuit) という。図-3 に例示する PLA の設計法は矢島・荒巻¹³⁾によって提案されたものであり、外部からテストパターンを加えることなくクロック信号 c のもとで自律テストを実行することができる。この設計法は組込み自己テスト可能な PLA の研究の先駆けとなった。

同図において、シフトレジスタ γ_1 と γ_2 、パリティチェッカ δ_1 と δ_2 、積項線 α_1 および出力線 β_1 は、図-2 の PLA と同様に本来の論理を実現する部分を万能テスト可能にするためのものである。また、自律テストの本質的要素である帰還信号 ϕ を生成するために、次の回路が付加されている。

1. β_2 を除くすべての出力線に接続される積項線 α_2 と α_3
2. すべてのビット線および出力線に接続される積項線 α_4
3. 積項線 α_4 に接続される出力線 β_2
4. $\phi = z_1z_4 + z_1z_2z_3 + z_1z_2z_3 + z_2z_3z_4$ を実現する帰還信号生成回路 λ

この PLA では、時刻 $(t-1)$ におけるシフトレジスタ γ_1 と γ_2 の状態と帰還信号 ϕ から、クロック信号 c に同期して、時刻 t で論理アレイ部に加えられるテストパターンが生成される。外部入力の数を n 、積

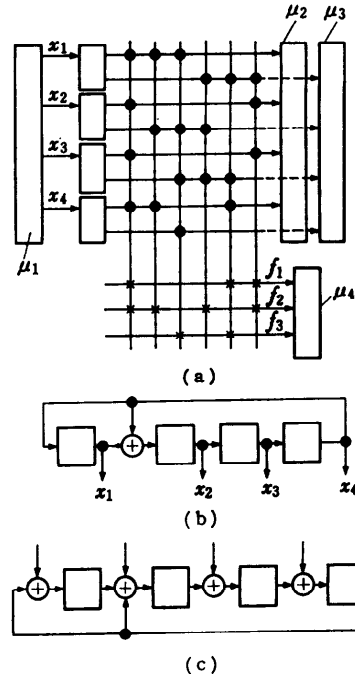


図-4 全数テスト方式の実現例

項線の数を m とすると、 $n+2m$ クロック後のシフトレジスタ γ_1 の内容により、単一の縮退故障と交点故障のすべてが検出できる。

(3) 全数テスト方式

全数テストとは、 n 入力の回路に 2^n 個の入力パターンのすべてを加えるテストをいう。 n が大きくなるとテスト時間が膨大となるので、適当な入力数をもつ部分回路に分割して、各部分回路に対する全数テスト (擬似全数テスト) を行うことが必要である。

図-4 (a) は、LFSR (linear feedback shift register) を用いた全数テストの例を示している¹⁴⁾。この PLA では、LFSR μ_1 で生成されるパターンをテスト入力とし、ビット線の信号系列を LFSR μ_2 と LFSR μ_3 で、また出力線の信号系列を LFSR μ_4 で圧縮する。

LFSR μ_1 は、D-フリップフロップと排他的論理和ゲートを用いて、(b) に例示するように構成される。この LFSR により周期 2^n-1 の最大長系列 (0000 を除くすべてのパターン) が発生できる。一般に n 次の原始多項式を実現する LFSR を用いれば、周期 2^n-1 の最大長系列が発生できる。ただし、パターン 00...0 を加えていないので、厳密に言えば全数テストではないことに注意されたい。

(c) に例示するように LFSR μ_1 と同一の原始多項式を実現する LFSR を $\mu_2 \sim \mu_4$ として、信号系列を圧

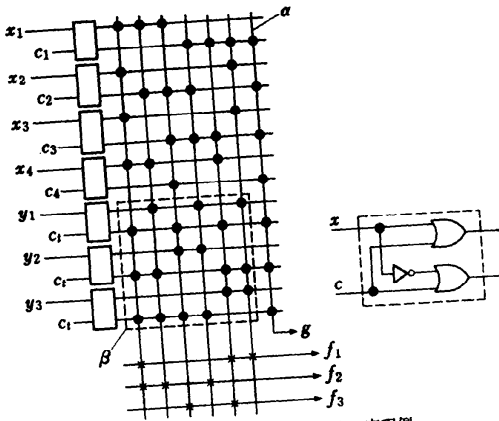


図-5 擬似全数テスト方式の実現例

縮する。このとき、 $\mu_2 \sim \mu_4$ の最終値 (シグネチャ) により、ビット線および出力線の多重縮退故障のすべてと、多重の交点故障および積項線の縮退故障のほとんどが検出できる。

LFSR は通常のデータレジスタに少量の回路を付加することにより容易に実現できる。そこで、上記のテスト方式が、ビット線の信号系列を圧縮する μ_2 と μ_3 を除去した形で、32 ビットマイクロプロセッサ INTEL 80386 内の PLA に適用されている¹⁶⁾。

図-5 は、図-1 の PLA に次の回路を付加して擬似全数テストを可能にした PLA を例示している¹⁶⁾。

1. AND アレイをいくつかのセグメントに分割するための制御入力 c ($c=1$ のデコーダに対応する部分が非活性となり論理的に除去される)

2. AND アレイの分割により検出不能な故障が発生することを防止するために、各積項線に互いに排他的な論理を付加する AND アレイ β

3. デコーダの故障を検出可能とするために、 \bar{x} が出力されるビット線のすべてと接続をもつ積項線 α

この PLA では、通常の論理を実現する AND アレイを適当な大きさのセグメントに分割し、各セグメントとテスト用 AND アレイ β および OR アレイからなる部分回路ごとにシンドロームテスト (すべての入力ベクトルを一度ずつ加えて出力 f および g が 1 となる回数をかぞえて良否を判定するテスト) を実行することにより、単一の交点故障、縮退故障および短絡故障のすべてが検出できる。

(4) ランダムテスト方式

n 段の LFSR で発生される周期 $2^n - 1$ の最大長系

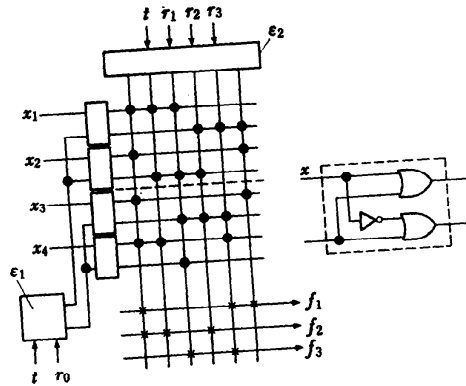


図-6 ランダムテスト方式の実現例

列は、擬似ランダム系列の一種である。したがって、ランダムテストは LFSR を用いることにより簡単に実行できる。しかし、積項線および出力線へのファンイン (fanin) が多い PLA では、非常に長いランダムパターンを加えても高い故障検出率を達成することは困難である。

図-6 は、図-1 の PLA に次の回路を付加し、ランダムテストを容易にしたものである¹⁷⁾。

1. AND アレイをいくつかのセグメントに分割し、ランダムにその一つを選択する回路 ϵ_1
2. 積項線の一つ (それぞれが他と異なる少なくとも一つの出力線に接続される複数の積項線を一括してもよい) をランダムに選択する回路 ϵ_2

この PLA では、テスト動作時 ($t=1$) に、 $x_1 \sim x_4$ と $r_0 \sim r_3$ にランダムパターンが加えられる。 r_0 により破線で 2 分割された AND アレイの一方が、また $r_1 \sim r_3$ により積項線の一つがランダムに選択され、上記のファンイン数を実効的に減少する。

図-6 の PLA よりも付加回路を削減し故障検出率を向上させるために、重み付ランダムパターンを入力とし積項線の選択にシフトレジスタを用いる方法¹⁸⁾、積項線のファンイン数を実効的に減少させるためにデコーダと AND アレイの間にマスクアレイを挿入する方法¹⁹⁾などが提案されている。

上記のようなテスト容易化をはかっても、高い故障検出率を得るためには、ある程度長いランダムパターンを加えなければならない。したがって、出力系列を LFSR などで圧縮する方法をとることが肝要である。

4. 最適な設計法の選択

これまでに 30 以上のテスト容易化設計法が提案されているが、それぞれに長所・短所があり、任意の要求に応えられる方法はない。したがって、個々の要求に適合する設計法を選択することが必要となる。

テスト容易化設計法は、通常次のような項目を評価尺度として比較される²⁰⁾。

1. テストの品質 (故障モデルとその検出率)
2. オーバヘッド (素子, 面積, I/O ピン, 遅延の増加)
3. テストパターン数とテスト時間
4. テスト生成の容易さ
5. 組込み自己テスト性

項目 4 と 5 を満たす Treuer¹⁰⁾, Daehn¹¹⁾, Liu¹²⁾, Yajima¹³⁾ および Hassan¹⁴⁾ の設計法が, 文献 12) で比較されている。表-2 は, 31 入力, 125 積項, 39 出力の PLA に対する項目 1~3 の評価例を示している。

表-2 より, CMOS 技術を用いてこの PLA を実現する場合は, Liu の設計法が優れている。一方, NMOS・PLA が対象で, テスト時間を最重視する場合は Daehn の設計法が, それ以外なら Treuer の設計法が選択されるであろう。

設計目標と制約が与えられたときに多数の候補のなかから最適な設計法を決定することは,それほど簡単ではない。特に, 目標のすべてを満足する設計法がない場合には, 妥協点を見出すという困難な作業をとまなう。したがって, テスト技術の専門家ではないチップ設計者がこのような作業をする場合には, 設計者を補助し結論に導くツールが必要不可欠であると思われる。

PLA-ESS²¹⁾ は, この目的のために試作されたエ

表-2 テスト容易化設計法の比較例

	Treuer	Daehn	Liu	Yajima	Hassan
テスト数	8001	227	8001	289	2×10 ⁶
テスト時間	664000	1135	40000	74000	10 ^{**}
付加回路量 (ゲート換算)	424	1359	360	977	466
検出可能故障	s-sxb m-sxb	s-sxb	s-sxbc m-sxb	s-sx	s-sx m-s
デバイス技術	NMOS	NMOS	CMOS	—	—

s: 単一, m: 多重, s: 縮退, x: 交点, b: 短絡,
c: CMOS 特有

キスパートシステムであり, 次の要素から構成されている。

1. 知識ベース: 設計法や評価法に関する知識
2. 推論部: システムとユーザの会話を制御し, 必要に応じて他を起動するプログラム
3. 解の探索部: 与えられた目標と制約の下で解を探索し, 解がなければその原因を解析して解決策を提示するプログラム
4. 説明部: ユーザに種々の判断をさせるためにシステムの処理状態を知らせるプログラム
5. トレース部: 別解の探索や再設計のために設計の履歴を追跡するプログラム

ユーザは, メニュー形式でシステムと会話をしながら作業を進めることにより, 設計過程の見直し, 要求項目の一部変更や決定事項の取り消しによる再設計, 別解の探索などを容易に行うことができる。

5. むすび

PLA のテスト容易化設計に関する基礎的な研究は一段落の感があり, 今後は VLSI に適用するための実用化研究が主となる。これにとまない, チップ設計者を補助し最適な設計法を決定する優れたツールの開発が必要となろう。本稿では, テスト入力の性質に着目して設計法の分類を試みたが, 文献 20) には別の観点からの分類が示してあるので一読されたい。

参考文献

- 1) 笹尾 勲: VLSI における回路設計方式—PLA を中心に, 情報処理, Vol. 28, No. 5, pp. 613-622 (1987).
- 2) Fleisher, H. and Maissel, L. I.: An Introduction to Array Logic, IBM J. R & D, Vol. 19, No. 2, pp. 98-109 (1975).
- 3) Ostapko, D. L. and Hong, S. J.: Fault Analysis and Test Generation for Programmable Logic Arrays, IEEE Trans. Comput., Vol. C-28, No. 9, pp. 617-626 (1979).
- 4) Smith, J.: Detection of Faults in Programmable Logic Arrays, IEEE Trans. Comput., Vol. C-28, No. 11, pp. 845-853 (1979).
- 5) Fujiwara, H. and Kinoshita, K.: A Design of Programmable Logic Arrays with Universal Tests, IEEE Trans. Comput., Vol. C-30, No. 11, pp. 823-828 (1981).
- 6) Hong, S. J. and Ostapko, D. L.: FITPLA: A Programmable Logic Array for Function Independent Testing, Proc. FTCS-10, pp. 131-136 (1980).

- 7) 佐藤, 当麻: 関数に依存しないテスト集合を持つ PLA の新構成法, 信学論 (D), Vol. J65-D, No. 8, pp. 1073-1079 (1982).
- 8) Saluja, K., Kinoshita, K. and Fujiwara, H.: An Easily Testable Design of Programmable Logic Arrays for Multiple Faults, IEEE Trans. Comput., Vol. C-32, No. 11, pp. 1038-1046 (1983).
- 9) Fujiwara, H.: A New PLA Design for Universal Testability, IEEE Trans. Comput., Vol. C-33, No. 8, pp. 745-750 (1984).
- 10) Treuer, R., Fujiwara, H. and Agarwal, V.: Implementing a Built-In Self-Test PLA Design, IEEE Design & Test of Comput., Vol. 2, No. 2, pp. 37-48 (1985).
- 11) Daehn, W. and Mucha, J.: A Hardware Approach to Self-Testing of Large Programmable Logic Arrays, IEEE Trans. Comput., Vol. C-30, No. 11, pp. 829-833 (1981).
- 12) Liu, D.L. and McCluskey, E. J.: Design of Large Embedded CMOS PLA's for Built-In Self-Test, IEEE Trans. CAD, Vol. 7, No. 1, pp. 50-59 (1988).
- 13) Yajima, S. and Aramaki, T.: Autonomously Testable Programmable Logic Arrays, Proc. FTCS-11, pp. 41-43 (1981).
- 14) Hassan, S. Z. and McCluskey, E. J.: Testing PLAs Using Multiple Parallel Signature Analyzers, Proc. FTCS-13, pp. 422-425 (1983).
- 15) Gelsinger, P. P.: Built In Self Test of the 80386, ICCD-86, pp. 169-173 (1986).
- 16) 山田輝彦: シンドロームテスト容易な PLA の一構成, 信学論 (D), Vol. J 66-D, No. 8, pp. 932-938 (1983).
- 17) Eichelberger, E. M. and Lindbloom, E.: Random-Pattern Coverage Enhancement and Diagnosis for LSSD Logic Self-Test, IBM J. R & D, Vol. 27, No. 3, pp. 265-272 (1983).
- 18) Ha, D. S. and Reddy, S. M.: On the Design of Random Pattern Testable PLAs, Proc. ITC '86, pp. 688-695 (1986).
- 19) Fujiwara, H.: A Design of Programmable Logic Arrays with Random Pattern-Testability, IEEE Trans. CAD, Vol. 7, No. 1, pp. 5-10 (1988).
- 20) Zhu, X. and Breuer, M. A.: Analysis of Testable PLA Designs, IEEE Design & Test of Comput., Vol. 5, No. 4, pp. 14-28 (1988).
- 21) Breuer, M. A. and Zhu, X.: A Knowledge Based System for Selecting a Test Methodology for a PLA, Proc. 22nd DAC, pp. 259-265 (1985).

(平成元年7月28日受付)