論文95-32A-9-14

연결선의 완벽한 진단을 위한 테스트 패턴의 생성

(A New Complete Diagnosis Patterns for Wiring Interconnects)

朴 成 柱 *
(Sungju Park)

요 약

보드 및 다칩모듈(Multi Chip Module)에서 칩과 칩간의 연결선상에서 발생할 수 있는 다양한 종류의 오점을 점검하는 일은 중요한 문제이다. 바운다리스캔은 연결선 점검 패턴생성과 주입을 간편화 시키기 위하여 개발된 설계기법이다. 연결선 점검 패턴생성을 위한 여러 알고리듬이 개발되었다. 본 논문에서는 현재 개발 사용중인 알고리듬보다 반이 적은 테스트 패턴 생성으로 모든 오점을 점검 및 진단할 수 있는 새로운 알고리듬을 소개한다.

Abstract

It is important to test the various kinds of interconnect faults between chips on a card/module. When boundary scan design techniques are adopted, the chip to chip interconnection test generation and application of test patterns is greatly simplified. Various test generation algorithms have been developed for interconnect faults. A new interconnect test generation algorithm is introduced. It reduces the number of test patterns by half over present techniques. It also guarantees the complete diagnosis of mutiple interconnect faults.

I. Introduction

Scan design techniques have emerged to overcome some of the difficulties of producing test data for different fault models and levels of packaging (e.g. chips, cards, modules). In-circuit tests and functional tests have been the primary method for testing chips on a card and interconnections of chips on a card (from now on the 'card' is meant to be the 'card or module'). In-circuit test based on the bed-of-nails probing technique makes it possible to test each chip and the interconnections among chips. However it requires the automatic test equipment (ATE) to probe each chip pin and the increasing use of surface mounting techniques make it difficult to perform in-circuit test. Although functional test does not need extensive access to chip pins on the card, it can not be guaranteed to derive high coverage test patterns. Also it may require a detailed description of each

* 正會員, 漢陽大學校 電子計算學科
(Dept. of Computer Science and Engineering, Hanyang Univ.)
接受日字:1995年4月29日, 수정완료일:1995年8月30日

chip which often is not available from the vendor. Boundary scan is a new design for testability technique aiming to improve the card level testability by embedding a dedicated boundary scan register or making use of the part of the scan register in each chip. IBM boundary scan design has been developed in support of reduced pin count test and interconnect test where the boundary scan latches belong to the scan register. IEEE 1149.1 boundary scan design which uses an explicit test protocol is becoming a widely adopted industry standard.

It is important to test the various kinds of interconnect faults between chips on a card such as AND, OR and dominating type shorted net faults, open net faults, and stuck at faults. When boundary scan design techniques are adopted, the chip to chip interconnection test generation and application of test patterns is greatly simplified. Various test generation algorithms have been developed for shorted nets faults. The log(n+2) approach (where 'n' is the total number of nets) detects any net fault and can be easily generated using counting technique [3] [4] [7]. In order to diagnose the net faults, the 2*log(n) counting and complement counting algorithm has been suggested [2] [9]. However different types of multiple shorted nets faults can not be diagnosed, since the 2*log(n) approach generates the same output vectors for the different pairs of shorted nets faults. Recently a 2*n marching 0 and 1 patterns approach was developed guaranteeing complete diagnosis [6]. Although sufficient, it is not necessary to apply 2*n patterns. A new n+1 patterns approach is proposed in this paper which has the same level of diagnosis as 2*n patterns while requiring only half the number of patterns. It will be shown that the n+1 patterns is sufficient to diagnose all the possible shorted nets faults.

This paper is organized as follows. We begin with a description of fault models and definitions in section 2. Currently known algorithms are reviewed in section 3. A new interconnect test generation algorithm is described in section 4 followed by a conclusion and future work.

II. Definitions and Fault Models

We consider the following classes of fault models.

1. *S-at-1 and S-at-0*: The conventional stuck at fault model.
2. *S-open*: The fault model for CMOS implementations which models any open net fault as either a pull-up or pull-down circuit. Initialization and transition patterns, that is, a two pattern test is required to detect a stuck-open fault.
3. *Shorted Nets Faults*: AND, OR, OPEN, DOMINATOR:The fault model for shorted nets faults can be classified into AND, OR, OPEN and DOMINATOR type faults. Suppose two nets: (A, B) are shorted and let the logic values at each net be V(A) and V(B) respectively then:
   (a) An AND type short results in logic 0 if either net is logic 0.
   (b) Conversely an OR type short results in logic 1 if either net is logic 1.
   (c) We call A DOMINATES B if V(A) appears at both nets regardless of V(B). Similarly B DOMINATES A if V(B) always appears at both nets regardless of V(A).

The following definitions are used through the rest of this paper.

Definition 1 A *net* is defined as the interconnect wiring between output pins and input pins of chips or card. The *source* of a net is defined as chip output pins or card input pins which can drive signals on the net. Likewise *sink* of a net is defined as chip input pins or card output pins.

Definition 2 A *pin* is called a *twostate* driver if it is fed by only one data input source which can be an output boundary scan (B-S) cell for chip pin or directly from an ATE driver for card pin. A pin is called a *tristate* driver if its input data is gated by an ENABLE signal. A *bidi* pin of a chip is defined as a pin that can drive signals as well as receive signals through the control of an ENABLE signal. It is called as CIO pin on a card.

Definition 3 A *test pattern* is a set of logic values which need to be assigned to each boundary scan cell and card pin.

Example 1 In Table 1 the lst test pattern is '00001111' which is the first column of the input vector.

Definition 4 An *input vector* is defined to describe the stream of input patterns of which each element belongs to a different test pattern. *output vector* is similarly used to illustrate the stream of output patterns corresponding to an input vector. In general, the fault free output vector is supposed to be the same as the input vector in a boundary scan designed card.

Example 2 For example, '001' in In Table 1 is an input vector for net n2.

Definition 5 The detection is the ability to detect faults. The diagnosis is the ability to determine which nets are shorted together or faulty.

Definition 6 The complete detection and diagnosis is the ability to detect and diagnose any number of multiple shorted nets fault.

Table 1. Log(n) detection patterns

Nets	*Input Vector*		
n1	0	0	0
n2	0	0	1
n3	0	1	0
n4	0	1	1
n5	1	0	0
n6	1	0	1
n7	1	1	0
n8	1	1	1

We have presented the fault models and some definitions. In the following section a new Shorted Nets Test generation algorithm is described along with the currently widely known algorithms.

Ⅲ. Review of the Currently Known Algorithms

It is important to test the various kinds of interconnect faults between chips on a card such as AND, OR and DOMINATOR type shorted nets faults, open net faults, and stuck at faults. Many algorithms have been developed to detect and locate the wiring interconnect net faults [2] [3] [4] [5] [6] [7] [8] [9] [10]. Some approaches made assumptions such that certain physical layout information is available to determine adjacency relationships between nets and hence the maximum distance of nets which can be shorted in the physical implementation. However, in practice test generators typically do not have ready access to physical design information. Our approach to interconnect test generation removes the dependency of having to have physical design information available in order to generate tests for

shorted nets faults. Our design focuses on an one-step detection and diagnosis approach as opposed to other so-called adaptive methods which require a two-step process: i.e. detection and then diagnosis for only the faulty nets. It should be noted however, that our method can be easily extended to the adaptive method which relies on a direct interface to ATE.

Let us assume that the target card consists of 'n' nets. The log(n) patterns algorithm was developed to detect all shorted net faults except stuck-at-faults [7]. It is clear that by applying a unique input for each net, any shorted nets fault will result in the different output vector from the input vector. These patterns can be easily generated using the log(n) bit counter. Since the patterns include vectors with all zeros and all ones, the S-at-0 or S-at-1 on these nets can not be detected. The enhanced version (log(n+2)) which detects all shorted nets faults as well as all stuck-at faults has been suggested in [3]. By excluding the all zeros and all ones, (by generating unique input patterns for 'n+2' nets by the counting method and assigning non zeros and non ones input vectors to 'n' nets), the log(n+2) test patterns detects all shorted nets faults. Test Patterns for n=8 nets for both log(n) and log(n+2) is shown in Table 1 and Table 2 respectively.

The previous algorithms have focused on the detection of shorted nets faults (any multiple fault). Now let us focus on the diagnosis of shorted nets faults. The basic idea to diagnose shorted nets faults, is to compare the faulty output vectors and find the set of nets which result in the same faulty output vectors. In order to obtain the unique faulty output vector i.e., to eliminate any ambiguity to achieve complete diagnosis, first, no fault free output vector should be the same as the faulty output vector, and second, any faulty output vector has to be distinct from all other faulty output vectors.

Table 2. (n+2) detection patterns

Nets	*Input Vector*			
n1	0	0	0	1
n2	0	0	1	0
n3	0	0	1	1
n4	0	1	0	0
n5	0	1	0	1
n6	0	1	1	0
n7	0	1	1	1
n8	1	0	0	0

For example, using the input vectors of Table 2, a two-way OR type short between n3 and n4 is indistinguishable from the three-way OR type short between n3, n4, and n7, since both faults would produce the vector '0111' on all three nets. Similarly, the two-way OR type short between n5 and n6 produces the vector '0111' on both of those nets. It follows that if '0111' is observed on nets n3, n4, n5, and n6 (and all other nets produce their correct responses) then it is impossible to diagnose the short because any of eleven possible shorting configurations could be the culprit:

1. n3 and n4 shorted;n5 and n6 shorted
2. n3 and n4 shorted;n5, n6, and n7 shorted
3. n3 and n5 shorted;n4, n6, and n7 shorted
4. n3 and n6 shorted;n4, n5, and n7 shorted
5. n3 and n7 shorted;n4, n5, and n6 shorted
6. n4 and n7 shorted;n3, n5, and n6 shorted
7. n5 and n6 shorted;n3, n4, and n7 shorted
8. n5 and n7 shorted;n3, n4, and n6 shorted
9. n6 and n7 shorted;n3, n4, and n5 shorted
10. n3, n4, n5, and n6 shorted
11. n3, n4, n5, n6, and n7 shorted

With the above underlying ideas regarding

diagnosis, let us review some algorithms. The Counting and Complementary Counting algorithm was proposed by [9] and [2]. The algorithm applies log(n) counting patterns and then another log(n) complementary of the initial counting patterns, totally 2log(n) patterns. ([9] initally used 2log(n+2) but it can be obviously observed [2] that 2log(n) patterns give the same coverage.) The 2log(n) patterns are shown in Table 3. Although these patterns can diagnose any single shorted nets fault, it can only diagnose part of multiple fault. Suppose n1 and n8 are shorted with an OR type fault, and n2 and n7 are shorted with OR type fault. Both faulty output vectors are '111111', hence there is no way to tell which pairs of nets are shorted together.

Table 3. 2Log(n) diagnosis patterns

Nets	*Count Vector*			*Complement Vector*		
n1	0	0	0	1	1	1
n2	0	0	1	1	1	0
n3	0	1	0	1	0	1
n4	0	1	1	1	0	0
n5	1	0	0	0	1	1
n6	1	0	1	0	1	0
n7	1	1	0	0	0	1
n8	1	1	1	0	0	0

Table 4. 2*n walking patterns Nets

Nets	*Walking 1*								*Walking 0*							
n1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0
n2	0	0	0	0	0	0	1	0	1	1	1	1	1	1	0	1
n3	0	0	0	0	0	1	0	0	1	1	1	1	1	0	1	1
n4	0	0	0	0	1	0	0	0	1	1	1	1	0	1	1	1
n5	0	0	0	1	0	0	0	0	1	1	1	0	1	1	1	1
n6	0	0	1	0	0	0	0	0	1	1	0	1	1	1	1	1
n7	0	1	0	0	0	0	0	0	1	0	1	1	1	1	1	1
n8	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1

To achieve complete diagnosis, the walking ZEROs and ONEs algorithm has been developed [6]. Diagonal independence of the test vector set guarantees complete detection and diagnosis(refer to [6] for more details). These '2n' walking patterns can be systematically generated and are illustrated in Table 4.. Although '2n' patterns are sufficient, they are not necessary for complete diagnosis.

Ⅳ. A New N+1 Algorithm

We are proposing a new 'n+1' algorithm (based on marching patterns) which can completely diagnose any multiple shorted nets fault. A sample set of test patterns is shown in Table 5.

Theorem 1 The *n+1* marching patterns can completely detect and diagnose multiple shorted nets faults.

Proof: First we will show that the patterns can detect any multiple fault. Since all input vectors for each net are different and contain at least one *zero* and *one*, they can detect all Shorted Nets faults and all S-at faults. Now let us prove complete diagnosis. It can be easily seen that any S-at fault results in an all *zeros* or all *ones* output vector which is distinct from any shorted nets faulty output vector(AND, OR or DOMINATING type faults includes at least one *one* and *zero* element in the output vector). Hence we need to consider only AND, OR and DOMINATING type faults from now on. Now we ask the question:Can any faulty output vector be the same as the fault free output vector? Every input vector contains different number of *ones* with a *covering* relation (we say that net n8 *covers* n7 if the input vector for n8 contains *ones* wherever the input vector for n7 has *ones*).

Table 5. n+1 complete diagnosis patterns

Nets	*Input Vector*								
n1	0	0	0	0	0	0	0	0	1
n2	0	0	0	0	0	0	0	1	1
n3	0	0	0	0	0	0	1	1	1
n4	0	0	0	0	0	1	1	1	1
n5	0	0	0	0	1	1	1	1	1
n6	0	0	0	1	1	1	1	1	1
n7	0	0	1	1	1	1	1	1	1
n8	0	1	1	1	1	1	1	1	1

For example, in Table 5 n8 covers all other nets and n2 covers only n1. The key characteristic of these new patterns is that any faulty output vector will be the same as one of the input vectors of the faulty nets due to the covering relation. For example if n1 and n2 are shorted together with an AND type fault, then the resulting faulty output vector will be the same as the fault free vector of n1 in Table 6. Hence no faulty output vector can be the same as a fault free output vector. Finally we show that any faulty output vector has to be different from other faulty output vectors regardless of the type of shorted nets fault. It is not difficult to see that no AND type faulty output vector can be the same as an OR or DOMINATING type faulty output vector, since each faulty output vector is obtained from one of the input vectors of shorted nets. Therefore, any type of multiple shorted nets fault will be diagnosed with these patterns. In some cases the patterns may not distinguish an AND(or OR) type fault from the DOMINATING type fault, however the faulty net pairs are completely classified. It may be noted that the 'n+1' patterns are sufficient but they may not be necessary.

We have proved the sufficiency of our new 'n+1' algorithm. As an example, suppose n1 and n8 are shorted with OR type fault, n2 and n7 shorted with OR type, n5 dominates n6 in the nets shown in Table 5. The faulty output vectors are shown in Table 6, where it is easy to see pairs of shorted nets. As with the other approaches test patterns for the n+1 algorithm are easily generated.

Table 6. Example:n+1 complete dignosis patterns

Nets	*Input Vector*									*Faulty Output Vector*							
n1	0	0	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1
n2	0	0	0	0	0	0	0	1	1	0	0	1	1	1	1	1	1
n3	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	1	1
n4	0	0	0	0	0	1	1	1	1	0	0	0	0	0	1	1	1	1
n5	0	0	0	0	1	1	1	1	1	0	0	0	0	1	1	1	1	1
n6	0	0	0	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1
n7	0	0	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1	1
n8	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1

We have shown various shorted nets test algorithms and introduced a new algorithm. Conclusion and future work is illustrated in the following section.

V. Conclusions and Future work

In this paper, we reviewed the various Shorted Nets Test algorithms which can be used in the boundary scan environment. A new *n+1* Shorted Nets Test algorithm which reduces the total number of test patterns by half of the best known algorithm while guaranteeing the complete diagnosis was introduced. Although both marching *2n* and *n+1* patterns are of linear size, the reduction of test application time will be great on the card with a large number of nets. For the *n+1* patterns are still sufficient, the less number of patterns may be necessary and sufficient for complete diagnosis.

References

[1] R. W. Bassett, M. E. Turner, J. H.

Panner, P. S. Gills, S. F. Oakland and D. W. Stout, *Boundary-scan design principles for efficient LSSD ASIC testing*, IBM Journal of Research and Development, Vol. 34, No. 2/3 March/May 1990. pp. 339-354.

[2] W. T. Cheng, J. L. Lewandowski and E. Wu, *Optimal Diagnostic Methods for Wiring Interconnects*, IEEE Transactions on Computer-Aided Design, Vol. 11, No. 9, Sept. 1992, pp. 1161-1166.

[3] P. Goel and M. T. McMahon, *Electronic Chip-In-Place Test*, Proceedings International Test Conference, 1982, pp. 83-90.

[4] A. Hassan, J. Rajski, and V. K. Agrawal, *Testing and Diagnosis of Interconnects using Boundary Scan Architecture*, Proceedings International Test Conference, 1988, pp. 126-137.

[5] IEEE Standard 1149.1-1990, *IEEE Standard Test Access Port and Boundary-Scan Architecture*, IEEE, June 1989.

[6] N. Jarwala and C. W. Yau, *A new Framework for Analyzing Test Generation and Diagnosis Algorithms for Wiring Interconnects*, Proceedings International Test Conference, 1989, pp. 63-70.

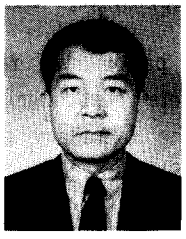
[7] W. K. Kautz, *Testing of Faults in Wiring Interconnects*, IEEE Transactions on Computers, Vol C-23, No.4, April 1974, pp. 358-363.

[8] J. C. Lien and M. A. Breuer, *Maximal Diagnosis For Wiring Networks*, Proceedings International Test Conference, 1991, pp. 96-105.

[9] P. T. Wagner, *Interconnect Testing with Boundary Scan*, Proceedings International Test Conference, 1987, pp. 52-57.

[10] C. W. Yau and N. Jarwala, *A Unified Theory for Designing Optimal Test Generation and Diagnosis Algorithms for Board Interconnects*, Proceedings International Test Confearence, 1989, pp. 71-77.

저 자 소 개



Sungju Park (正會員)

received the B.S degree in electronics from Hanyang University in 1983 and the M.S. and Ph.D. degrees in electrical and computer engineering from University of Massachusetts at Amherst in 1988 and 1992 respectively. He was with the Gold Star Company from 1983 to 1986. He worked for the IBM Microelectronics, Endicott, NY as a development staff in charge of Boundary Scan and LSSD Scan Design from 1992 to 1995. Since March 1995, he has been with the department of computer science and engineering in Hanyang University as an assistant Professor. His reserch interests lie in the area of VLSI testing including scan design(full, partial and boundary), built-in self test, test pattern generation, fault simulation, synthesis for test. Additional interests include design verification and graph theory.