PAPER

Fault Detection Capability of an $O(m \cdot n)$ Test Generation Algorithm for PLAs※

Yinghua MIN†, ***Nonmember and*** **Hideo FUJIWARA**††, ***Member***

SUMMARY Programmable Logic Arrays (PLAs) are very suitable to VLSI implementation and very convenient for logic design, because of the structure regularity and programmable flexibility. However, PLAs are random-resistant circuits for which random patterns are not effective for achieving a high fault coverage, because of their high fan-in and fan-out. This paper presents an $O(m \cdot n)$ test generation algorithm to generate ETG (Easy Test Generation) patterns where m is the number of input lines and n is the number of products. The fault coverage of ETG patterns can be calculated based on parameters of a given PLA. Experimental results show that the fault coverage is higher than 90% with the confident 90.3%. An alternative of the $O(m \cdot n)$ test generation algorithm to enhance fault coverage is presented. The given PLA is also possible to be modified to an ETG PLA to reach 100% fault coverage of ETG patterns.

1. Introduction

Programmable Logic Arrays (PLAs) are very suitable to VLSI implementation and very convenient for logic design, because of the structure regularity and programmable flexibility. However, PLAs are random-resistant circuits for which random patterns are not effective for achieving a high fault coverage, because of their high fan-in and fan-out(1),(2). But, on the other hand, random testing is easy to implement in Built-In Self-Test (BIST), and simplifies test generation. Much work has been done in PLA design for random pattern testability(3), (5). As another approach, a special kind of PLA, an ETG PLA, was defined in Ref. (6) for which ETG patterns can be generated by an $O(m \cdot n)$ test generation algorithm, where m is the number of input lines and n is the number of product lines, and the fault coverage is 100%. The design methodology for ETG PLAs is described in Ref. (7), and implemented in a software package, PLAT(8). The hardware overhead is only 5% on an average.

Interestingly, it is found that for general PLAs, not necessarily to be ETG PLAs, ETG patterns cover more than 90% crosspoint defects. This paper demonstrates the fact. Some formulas are given to predict the fault coverage of ETG patterns based on some parameters of the given PLA. Experimental results show the consistency of the calculation and fault simulation. Finally, an alternative of the $O(m \cdot n)$ test generation algorithm to further enhance fault coverage is remarked.

2. ETG PLAs and ETG Patterns

Consider a PLA structure with input lines x_1, x'_1, $\cdots$, x_m, x'_m to the AND array, product lines w_1, $\cdots$, w_n, and output lines z_1, $\cdots$, z_p. The product line w_i is represented by a 0/1/-or x sequence, where "-" represents "don't care", and is considered to be a point set. The crosspoint connecting x_i and w_j in the AND array is represented by (x_i, w_j), and similarly for (w_j, z_k) in the OR array. The fault model considered includes single extra or missing crosspoint faults. It has been shown in Ref. (9) that these will model the effects of all single stuck-at faults as well as most of the bridging faults. Extra/Missing crosspoint defects in the AND or OR array are represented by AE, OE, AM, or OM respectively. Let $W(z_k)$ be the set of product lines connected to output z_k, and let $Z(w_j)$ be the set of output lines connected to product line w_j.

Definition 1: For any product w_i, a point e_i in the Boolean space of dimension m is called the core of w_i if e_i is obtained by replacing all x's ofw_i wth O's.For instance, for $w_1=0111xx$, we have $e_1=011100$.

Definition 2: The neighborhood of e_i, denoted by T^i, is the set of points with at most one bit different from e_i. Points in T^i are called ETG patterns for w_i. The collection of all ETG patterns for every product line is the test set T, i. e., ETG patterns for the PLA.

Definition 3: A PLA is an ETG PLA, if T is capable of detecting all detectable crosspoint faults, that is, the fault coverage is 100%.

From the above definitions, the ETG pattern generation procedure is as follows.

ETG Pattern Generation Procedure for PLAs:

For each product line wi, which is represented by an m-bit 0/1/- sequence ("-" denotes "don't care")

$$w_i = x^*_{i1} x^*_{i2} \cdots x^*_{ic_i} - \cdots -$$

where $x^*_{ij} = x_{ij}$ or $x'_{ij}(1/0)$, $(j=1, 2, \cdots, c_i)$, and c_i is

---

Manuscript received December 25, 1990.

† The author is with CAD Laboratory, Institute of Computing Technology, Academia Sinica, Beijing, China 100080.

†† The author is with the School of Science and Technology, Meiji University, Kawasaki-shi, 214 Japan.

※ This paper is partly supported by the National Science Foundation of China under Granto 6883011.

the number of variables appeared in w_i.

(a) replacing-'s with O's results in the core of the product.

$$T_0^i = x_{i1}^* x_{i2}^* \cdots x_{ic_i}^* 0 \cdots 0$$

(b) for each j, $(i \leqq j \leqq m)$, let

T_j^i=the vector complementing the j-th bit of T_0^i

and get the $(m+1)$ patterns for the i-th product. Then, the test set is as follows :

$$T = \bigcup_{i=1}^{n} \{T_0^i, T_1^i, \cdots, T_m^i\}$$

Obviously, the computational complexity of the procedure is $O(m \cdot n)$ where m is the number of input lines and n is the number of product lines. It can be simply executed on a tester during test application.

3. Fault Detection Probability with ETG Patterns

Let's consider the probability of detecting different kinds of crosspoint faults with ETG patterns first.

3. 1 Single AE Fault

Case I (AE fault f_1 in Fig. 1):

Suppose that an arbitrary input pattern is applied at primary inputs with equal probabilities of 0 and1. In order to activate the fault f_1, the pattern should activate w_i without f_1, but deactivate it due to the fault. The probability of propagating the error due to fault f_1 to product line w_i is 0.5^{c_i}, where c_i is the fan-in of w_i. On the other hand, when ETG patterns are applied, there are $(m+1-c_i)$ patterns, which can propagate the error to w_i.

To detect the fault f_1, we have to propagate the error further to one of the output lines connected to w_i. To propagate the error to an output line $z_k \in Z(w_i)$, all other product lines connected to z_k except w_i should be 0. In case of applying random patterns, the probability is

$$\prod_{w_j \in W(z_k), j \neq i} (1-0.5^{c_j})$$

under the assumption of independency of signal propagation. Hence, the probability of propagating the error to at least one of the output lines is

$$1-\prod_{z_k \in Z(w_i)} \left(1-\prod_{w_j \in W(z_k), j \neq i} (1-0.5^{c_j})\right)$$

In cane of applying ETG patterns, let us assume that each product line $w_j \in W(z_k)$ has the value 0 with the same probability as in case of random patterns, i.e., 0.5^{c_j}. Then we can derive the probability of error propagation in the same way

$$1-\prod_{z_k \in Z(w_i)} \left(1-\prod_{w_j \in W(z_k), j \neq i} (1-0.5^{c_j})\right)$$

The probability of detecting the AE fault with a random pattern is thus

$$P_{\mathrm{RAND}}(f_1) = 0.5^{c_i}\left(1-\prod_{z_k \in Z(w_i)} \left(1-\prod_{w_j \in W(z_k), j \neq i} (1-0.5^{c_j})\right)\right)$$

On the other hand, the probability of detecting the AE fault with its ETG pattern is

$$P_{\mathrm{ETG}}(f_1) = 1-\prod_{z_k \in Z(w_i)} \left(1-\prod_{w_j \in W(z_k), j \neq i} (1-0.5^{c_j})\right)$$

It is apparent that the probability of detecting the AE fault with an ETG pattern is much larger than that with a random pattern. In fact, if $c_i=1$, $P_{\mathrm{RAND}}(f_1)$ is only half of $P_{\mathrm{ETG}}(f_1)$, and the larger the c_i, the smaller the probability of random pattern detections. In addition, there are $(m+1-c_i)$ ETG patterns being able to activate w_i, and the fault cannot be detected only if all the ETG patterns cannot propagate the error to any of the output lines. Therefore, the probability of detecting the AE fault with ETG patterns is

$$P_{\mathrm{ETG}}(f_1) = 1-\left[\prod_{z_k \in Z(w_i)} \left(1-\prod_{w_j \in W(z_k), j \neq i} (1-0.5^{c_j})\right)\right]^{(m+1-c_i)} \qquad (1)$$

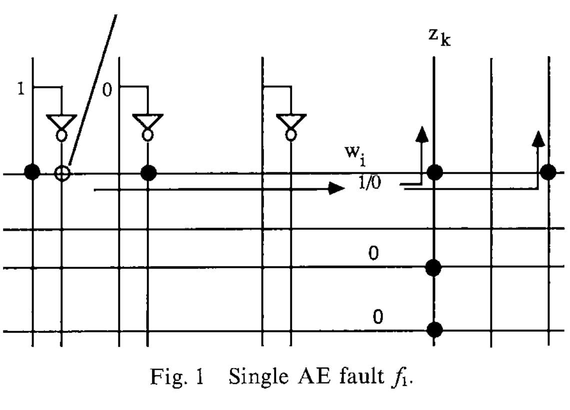

Fig. 1 Single AE fault f_1.

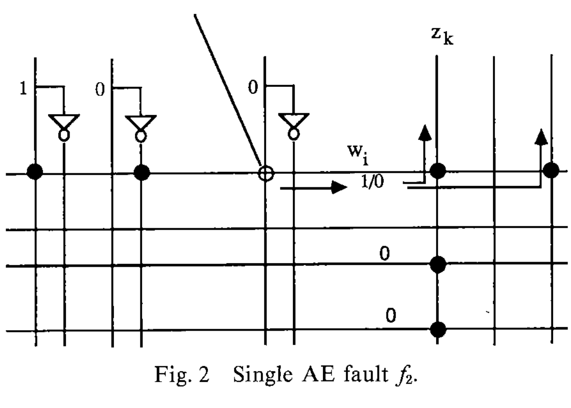

Fig. 2 Single AE fault f_2.

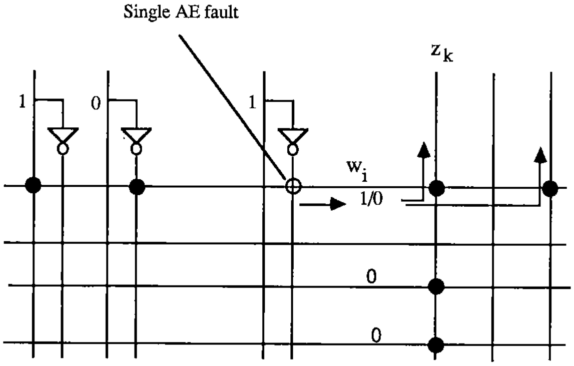

Fig. 3 Single AE fault f_3.

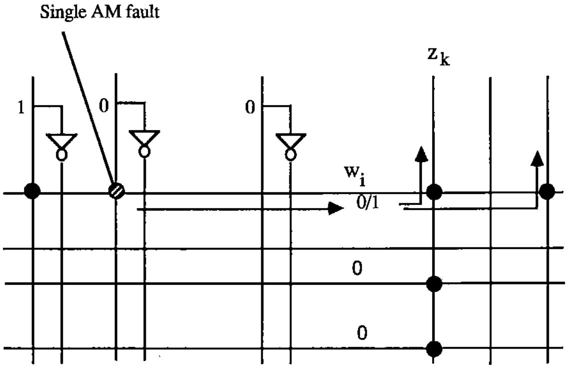

Fig. 4 Single AM fault f_4.

Case II (AE fault f_2 in Fig. 2):

Notice that there are $(m-c_i)$ patterns being able to sensitize f_2, and activate w_i. Similar to Case I,it can be obtained that the probability of detecting AE fault f_2 with ETG patterns is

$$P_{\mathrm{ETG}}(f_2)=1-\left[\prod_{z_k\in Z(w_i)}\left(1-\prod_{w_j\in W(z_k),j\neq i}(1-0.5^{c_j})\right)\right]^{(m-c_i)} \quad (2)$$

Case III (AE fault f_3 in Fig. 3) :

Notice that there is only one pattern being able to sensitize f_3, and activate w_i. Similar to Case I, it can be obtained that the probability of detecting AE fault f_3 with ETG patterns is

$$P_{\mathrm{ETG}}(f_3)=1-\prod_{z_k\in Z(w_i)}\left(1-\prod_{w_j\in W(z_k),j\neq i}(1-0.5^{c_j})\right) \quad (3)$$

3. 2 Single AM Fault (AM Fault f_4 in Fig. 4)

Notice that there is only one pattern being able to sensitize f_4, and deactivate w_i. Therefore, the probability of detecting AM fault f_4 with ETG patterns is

$$P_{\mathrm{ETG}}(f_4)=1-\prod_{z_k\in Z(w_i)}\left(1-\prod_{w_j\in W(z_k),j\neq i}(1-0.5^{c_j})\right) \quad (4)$$

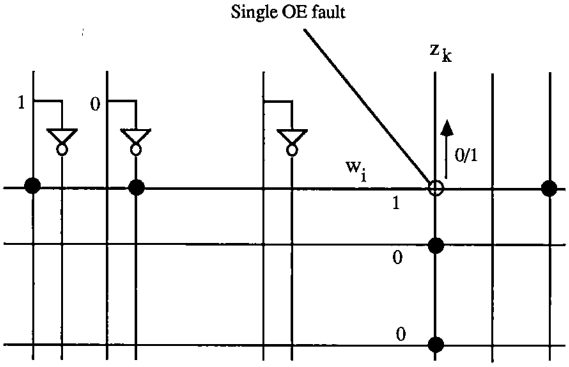

Fig. 5 Single OE fault f_5.

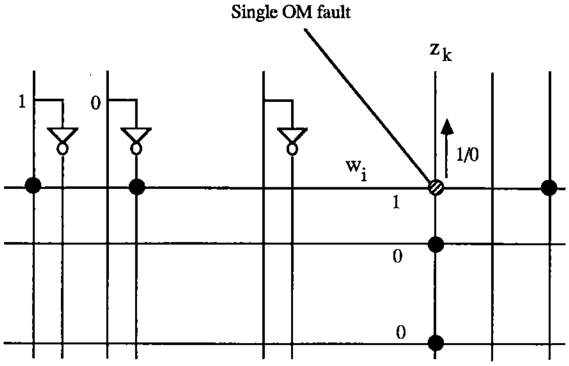

Fig. 6 Single OM fault f_6.

3. 3 Single OE Fault (OE Fault f_5 in Fig. 5)

To detect the fault f_5, it is necessary that the product line w_i takes the value 1. To propagate the error further to the output line z_k, all product lines that are connected to z_k should be 0. Notice that there are $(m+1-c_i)$ ETG patterns activating w_i. Therefore, under the assumption of independency of signal propagation, the probability of detecting OE fault f_5 with ETG patterns is

$$P_{\mathrm{ETG}}(f_5)=1-\left(1-\prod_{w_j\in W(z_k)}(1-0.5^{c_j})\right)^{(m+1-c_i)} \quad (5)$$

3. 4 Single OM Fault (OM Fault f_6 in Fig. 6)

Notice that there are $(m+1-c_i)$ patterns activating w_i. Therefore, the probability of detecting OM fault f_6 with ETG patterns is

$$P_{\mathrm{ETG}}(f_6)=1-\left(1-\prod_{w_j\in W(z_k),j\neq i}(1-0.5^{c_j})\right)^{(m+1-c_i)} \quad (6)$$

To evaluate the above detection probabilities, let

Table 1 Number of faults of each category.

Category	f_1	f_2	f_3	f_4	f_5	f_6
#	nc	n(m–c)	n(m–c)	nc	n(p–a)	na

us consider the case when the fan-out of product lines, the fan-in of output lines, and the fan-in of product lines are all equal, respectively, i. e., $|Z(w_i)| \equiv a_i \equiv a$, $|W(z_k)| \equiv b_k = b$, and $c_i = c$ for all i and k. If they are unequal, take the means as the parameters of the PLA, i.e.,

$$\begin{aligned} a &= (a_1 + a_2 + \cdots + a_n)/n \\ b &= (b_1 + b_2 + \cdots + b_p)/p \qquad (7) \\ c &= (c_1 + c_2 + \cdots + c_n)/n \end{aligned}$$

Then Eqs. (1)-(6) can be simplified as follows.

$$P_{\mathrm{ETG}}(f_1) = 1 - (1 - (1 - 0.5^c)^{b-1})^{a(m+1-c)} \qquad (8)$$

$$P_{\mathrm{ETG}}(f_2) = 1 - (1 - (1 - 0.5^c)^{b-1})^{a(m-c)} \qquad (9)$$

$$P_{\mathrm{ETG}}(f_3) = 1 - (1 - (1 - 0.5^c)^{b-1})^{a} \qquad (10)$$

$$P_{\mathrm{ETG}}(f_4) = 1 - (1 - (1 - 0.5^c)^{b-1})^{a} \qquad (11)$$

$$P_{\mathrm{ETG}}(f_5) = 1 - (1 - (1 - 0.5^c)^{b})^{m+1-c} \qquad (12)$$

$$P_{\mathrm{ETG}}(f_6) = 1 - (1 - (1 - 0.5^c)^{b-1})^{m+1-c} \qquad (13)$$

All single crosspoint faults fall exclusively into six categories. The number of faults of each category is listed in Table 1. Assuming all faults are equally likely to occur, the probability of occurrence of fault of each kind is equal to the number of faults of the category divided by the total number of faults. Known the probability of detection of each category under the condition of occurrence of fault of the category, as indicated in Eqs. (8)-(13), by the complete probability formula, we obtain the probability of detecting single crosspoint fault with ETG patterns, i.e.,

$$\begin{aligned} \mathrm{Pr} = \{ & cP_{\mathrm{ETG}}(f_1) + (m-c)P_{\mathrm{ETG}}(f_2) + (m-c)P_{\mathrm{ETG}}(f_3) \\ & + cP_{\mathrm{ETG}}(f_4) + (p-a)P_{\mathrm{ETG}}(f_5) \\ & + aP_{\mathrm{ETG}}(f_6) / (2m+p) \end{aligned} \qquad (14)$$

For a randomly given PLA with a large number of crosspoints, the probability of detecting single faults with ETG patterns is equal to the fault coverage of ETG patterns, that is, the ratio of the number of detected faults by ETG patterns to the total number of faults.

4. Experimental Results

The Eqs.(7)-(14) are applied to 62 practical PLAs to compare the calculated fault coverage of ETG patterns to the simulated one. The experimental results are listed in Table 2, where

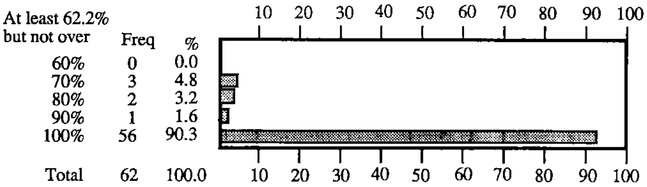

Fig. 7 The bar graph for simulated fault coverage.

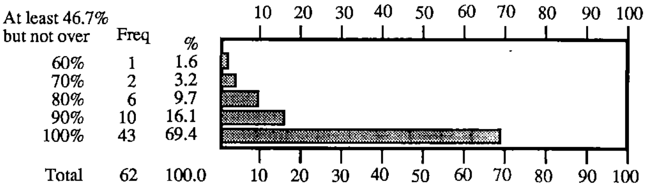

Fig. 8 The bar graph for calculated fault coverage.

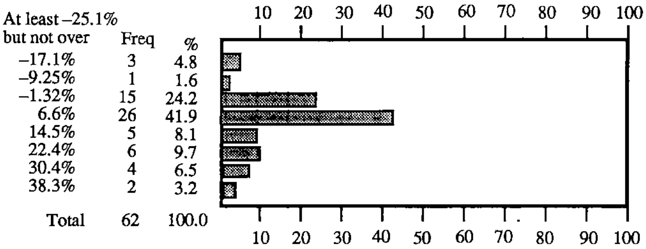

Fig. 9 The bar graph for errors between simulated and calculated fault coverage.

NI: Number of Inputs,
NP: Number of Products,
NO: Number of Outputs,
#_TEST: Number of ETG patterns,
SF-COV: Simulated Fault Coverage,
CF-COV: Calculated Fault Coverage,
DIFF: Difference between SF-COV and CF-COV.

For the simulated fault coverage, SF-COV,
Mean=96.0%
Variance=0.72%
Minimum=62.2%
Maximum=100%

The bar graph is shown in Fig. 7. The statistical analysis demonstrates that ETG patterns cover more than 90% single crosspoint faults for general PLAs of 90.3%.

For the calculated fault coverage, CF-COV,
Mean=92.0%
Variance=1.2%
Minimum=46.7%
Maximum=100%

The bar graph is shown in Fig. 8. The theoretical estimate demonstrates that ETG patterns cover 80% single crosspoint faults for general PLAs of 85.5%, and cover more than 90% for general PLAs of 69.4%.

Table 2 Experimental results.

PLA name	NI	NP	NO	#_TEST	SF-COV	CF-COV	DIFF
ALUTEST	14	36	21	337	1.000	1.000	0
BARNEW	8	33	27	136	0.982	0.999	–0.017
BAR	8	29	25	121	0.991	0.999	–0.008
CERBERUS	18	50	37	653	0.949	1.000	–0.051
COND	10	24	2	156	0.930	0.901	0.029
MASTER	15	27	13	262	0.951	0.999	–0.048
NEWALU	15	26	28	298	0.985	1.000	–0.015
RECUR	7	9	9	47	1.000	0.999	0.001
RIMP	12	39	21	262	0.971	1.000	–0.029
TRAFFIC	5	8	7	24	1.000	0.976	0.024
XCC2	4	10	7	15	1.000	0.879	0.121
XCC3	9	21	9	72	1.000	0.996	0.004
XCC4	5	31	3	32	1.000	0.723	0.277
XCC5	14	14	1	106	1.000	1.000	0.000
XCC6	10	38	11	148	1.000	0.998	0.002
XCC7	12	19	8	51	0.735	0.889	–0.154
XCC8	7	49	3	110	0.998	0.835	0.163
XCC9a	10	25	12	64	0.995	1.000	–0.005
XCC9b	10	47	12	183	1.000	0.981	0.019
XCC10	8	39	7	118	0.979	0.970	0.009
XCC11	15	29	17	212	0.999	1.000	–0.001
XCC12	7	60	4	127	1.000	0.729	0.271
XCC13	8	58	5	202	0.973	0.918	0.055
XCC14	6	50	12	64	1.000	0.943	0.057
XCC15	8	28	31	98	0.999	1.000	–0.001
XCC17	8	75	5	200	1.000	0.797	0.203
XCC19	9	88	1	246	0.699	0.467	0.232
XCC20	8	77	8	178	0.972	0.861	0.111
XCC21	10	70	8	272	0.675	0.864	–0.189
XCC22	10	72	8	216	0.622	0.820	–0.198
XCC23	7	127	3	128	1.000	0.635	0.365
XCC24	32	32	7	529	1.000	1.000	0.000
XCC25	8	122	5	252	1.000	0.843	0.157
XCC29	15	107	11	952	0.973	0.999	–0.026
XCC30	15	107	11	958	0.973	0.999	–0.026
XCC31	33	54	23	1564	0.984	1.000	–0.016
XCC33	25	110	8	1842	0.937	0.967	–0.030
XCC34	19	135	10	1660	0.966	0.999	–0.033
XCC35	27	101	6	2015	0.886	0.958	–0.072
XCC36	39	82	5	2305	0.742	0.993	–0.251
XCC37	27	120	7	2012	0.915	0.990	–0.075
XCC38	35	73	29	1946	0.933	1.000	–0.067
XCC46	30	117	63	3195	0.952	1.000	–0.040
XCD11a	4	4	2	12	1.000	0.859	0.141
XCD11b	5	6	4	24	0.990	0.939	0.051
XCD12	4	4	3	14	1.000	0.969	0.031
XCD13	4	4	3	12	1.000	0.970	0.030
XCD14	4	4	3	11	1.000	0.968	0.032
XCD15	4	4	3	12	1.000	0.966	0.034
XCD16	4	5	2	14	1.000	0.850	0.150
XCD17	4	6	3	15	1.000	0.860	0.140
XCD18	6	6	3	24	0.905	0.953	–0.048
XCD21	5	8	4	24	0.989	0.939	0.050
XCD31	4	11	3	16	1.000	0.796	0.204
XCD32	5	9	7	18	1.000	0.989	0.011
XCD41	4	9	3	13	1.000	0.617	0.383
XCD42	5	14	13	30	1.000	0.978	0.022
XCD50	8	14	9	58	1.000	0.996	0.004
XCD51	8	75	5	200	1.000	0.797	0.203
XCD71	8	125	8	246	0.995	0.921	0.074
XCD91	9	135	5	378	1.000	0.748	0.252
XZZZ	5	14	13	30	1.000	0.971	0.029

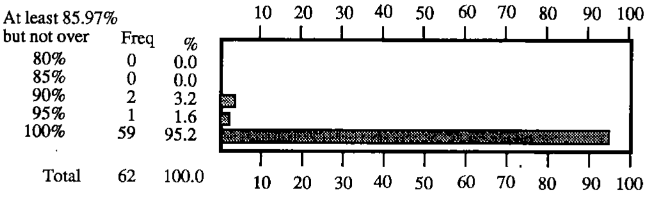

Fig. 10 The bar graph for simulated fault coverage of E01 ETG patterns.

Comparing to Fig. 7, we know it is a conservative estimate.

It is interesting to note the consistency of simulated fault coverage and calculated one. Let

error=simulated_fault_coverage_calculated_fault_coverage

then,

Mean of errors=4.0%
Variance of errors=1.5%
Minimum of errors=−25.1%
Maximum of errors=38.3%

The bar graph for errors is shown in Fig. 9, which shows 41.9% of estimates have the errors in the interval (−1.3%, 6.6%), and 93.6% of calculated fault coverage is either with very small error or with the practical coverage larger than the calculated one, and then is secure.

It is worth noting that most large errors are due to seriously discrete distribution of the parameters a, b and c. For instance, for XCC 36. PLA,

SF-COV=74.2%, CF-COV=99.3%,
error=−25.1%

where a=2.21 with some a_i=minimum 1, and some others=maximum 5. In addition, it has 39 primary inputs, but $c=7.8$, which means there are a lot of "don't care"s, and the fan-outs of the 39 primary inputs are seriously discrete, some are 1, some are 82. As another example, consider XCD 51. PLA,

SF-COV=100%, CF-COV=79.7%, error=20.3%

for which $b=15.0$, but actually $b_0=2$, and $b_3=36$. In these situations the assumption that all fan-out of products, fan-in of output lines, fan-in of products are equal would not be valid, so that the error becomes large.

5. Remarks

This section is to show that if we double the number of ETG patterns, the fault coverage can be further enhanced. We define two cores for each product w_i, the 0-core is obtained by replacing all-'s with O's, while 1-core is obtained by replacing all-'s with 1's, i.e.,

$$w_i = x_{i1}^* x_{i2}^* \cdots x_{ic_i}^* - \cdots -$$

$$T_0^i = x_{i1}^* x_{i2}^* \cdots x_{ic_i}^* 0 \cdots 0$$

$$T_1^i = x_{i1}^* x_{i2}^* \cdots x_{ic_i}^* 1 \cdots 1$$

E01 ETG patterns are input patterns complementing each bit of T_0^i or T_1^i. The ETG patterns mentioned in the previous sections are referred to E0 ETG patterns.

For the same 62 PLAs as indicated in Table 2, the simulated fault coverage of E01 ETG patterns is with the following.

Mean=99.08%
Variance=0.07%
Minimum=85.97%
Maximum=100%

The bar graph, as shown in Fig. 10, shows that E01 ETG patterns cover more than 90% single crosspoint faults for general PLAs of 96.8%, especially the fault coverage is higher than 95% for 95.2% PLAs. The result is comprehensive, because E01 ETG patterns include all E0 ETG patterns, the fault coverage is always higher than that of E0 ETG patterns.

The penalty of the enhancement of fault coverage is the increase in number of tests. Talbe 3 compares the means of number of tests and fault coverage for an optimized test set[10], E0 ETG patterns and E01 ETG patterns.

Table 3 Comparison of test strategies.

Means	Optimized test set	E0 ETG patterns	E01 ETG patterns
# of tests	194	414	750
Fault Coverage	100%	96.05%	99.08%

6. Conclusions

This paper presents an $O(m \cdot n)$ test generation algorithm to generate ETG (Easy Test Generation) patterns which cover more than 90% of single crosspoint faults for 90.3% of general PLAs, which are not necessarily to be ETG PLAs. If E01 ETG patterns are applied, the fault coverage can be higher than 95% for 95.2% of the PLAs.

This can be at least an initial step for PLA test generation. If higher fault coverage is required, a deterministic test generation algorithm can be taken for a very small fraction of crosspoint faults.

The fault coverage of ETG patterns can be calculated based on the parameters of the given PLA. Experimental results show that the calculated fault coverage is very close to the simulated result of the fault coverage, and the theoretical analysis is correct. In a design environment, if the calculated fault coverage satisfies the designer, he or she can use the $O(m \cdot n)$ test generation algorithm to generate ETG patterns without fault simulation. Otherwise, the designer can transform his PLA design to an ETG PLA by using the

ETG_PLA_Designer(5) to reach 100% fault coverage of ETG patterns.

Acknowledgements

This conjunct research is partially supported by the National Natural Science Foundation of China under the grant 6883011 and Wang Kuancheng Education Foundation. The authors would like to thank Dr. Jintao Li for his contributions of the experimental data.

References

(1) Wager K., Chin C. and McCluskey E. J.: "Pseudo-random testing", IEEE Trans. Comput., **C-36**, 3 (March 1987).

(2) Savir J. and Bardell P.: "On random pattern test length", IEEE Trans. Comput., **C-33**, pp. 467-474 (June 1984).

(3) Eichelberger E. M. and Lindbloom E.: "Random pattern coverage and diagnosis for LSSD logic self-test", IBM J. Res. Develop., **27**, pp. 265-272 (May 1983).

(4) Ha D. S. and Reddy S. M.: "On the design of random pattern testable PLAs", Proc. 1986 Int'l Test Conf., pp. 688-695 (1986).

(5) Fujiwara H.: "A design of programmable logic arrays with random pattern testability", IEEE Trans. CAD, **7**, 1 (Jan. 1988).

(6) Min Y.: "A PLA design for ease of test generation", Proc. 14th Fault-Tolerant Computing Symp., Kissimmee, USA (1984).

(7) Min Y.: "An ETG PLA designer", Proc. '89 Joint Symp. on Fault-Tolerant Computing, Chongqing, China, pp. 166-171 (July 1989).

(8) Min Y., Li J. and Li Z.: "PLAT-A software package for PLA testing", Proc. Int'l Conf. on CAD & CG, Beijing, China (July 1989).

(9) Min Y.: "A unified fault model for programmable logic arrays", Technical Report 83-5, Center for Reliable Computing, Stanford University (May 1983).

(10) Li J. and Min Y.: "A product-oriented strategy of test pattern generation for PLAs", to appear in J. of Computer Science and Technology, China.

Yinghua Min was born in Hunan, China, on December 18, 1935. He graduated from the Mathematics Department of Jilin University in 1962, and completed his post-graduate study in 1962-1966. He had visited Stanford University, State University of New York, and Colorado State University for years. He has thirteen year experience on computer control systems working with China Academy of Railway Sciences, and currently is Professor, and Director of The Center for Fault-Tolerant Computing of CAD Laboratory, Institute of Computing Technology, Academia Sinica, Beijing, China. He is the author of two books, and published 50 technical papers. His research interests include Fault-Tolerant Computing, Testing, Design Automation, and Real-Time Control Systems. He is a senior member of IEEE, and is on the editorial board of the Journal of Electronic Testing: Theory and Applications (JETTA) published by KLUWER ACADEMIC PUBLISHERS, Boston, USA.

Hideo Fujiwara was born in Nara, Japan, on February 9, 1946. He received the B.E., M.E. and Ph.D. degrees in electronic engineering from Osaka University, Osaka, Japan, in 1969, 1971, and 1974, respectively. He is currently a Professor at the Department of Computer Science, Meiji University, Tokyo, Japan. Dr. Fujiwara was a Visiting Research Assistant Professor at the University of Waterloo in 1981 and a Visiting Associate Professor at McGill University in 1984. His research interests are design and test of computers, including design for testability, built-in self-test, test pattern generation, fault simulation, computational complexity, parallel processing, neural networks and expert systems for design and test. He is the author of Logic Testing and Design for Testability (MIT Press, 1985). Dr. Fujiwara is a fellow of the IEEE as well as a member of the Information Processing Society of Japan. Currently he is on the editorial boards of IEEE Design and Test of Computers, the Journal of Electronic Testing: Theory and Application (JETTA) and the Journal of Circuits, Systems, and Computers (JCSC).