ELECTRONICS AND ELECTRICAL ENGINEERING
ISSN 1392 – 1215 *2012. No. 2(118)*
ELEKTRONIKA IR ELEKTROTECHNIKA

ELECTRONICS
T 170
ELEKTRONIKA

Application of Preselection of Test Subsequences in Sequential Test Generation for Functional Delay Faults

E. Bareisa, V. Jusas, K. Motiejunas, R. Seinauskas
Software Engineering Department, Kaunas University of Technology,
Studentu str. 50-406, Kaunas, Lithuania, e-mail: kestutis.motiejunas@ktu.lt
L. Motiejunas
Multimedia Engineering Department, Kaunas University of Technology,
Studentų St. 50-401a, Kaunas, Lithuania

crossref http://dx.doi.org/10.5755/j01.eee.118.2.1179

Introduction

Sequential circuit testing has been recognized as the most difficult problem in the area of fault detection. The difficulty comes from the existence of memory elements. High-performance circuits with aggressive timing constraints are usually very susceptible to delay faults. As the drive towards lower power processors continues, the number of "critical" paths increase, i.e. the delay of such paths is close to the rated speed of the circuit. Small process variations and environmental changes (like temperature increase) may cause such circuits to fail at the rated clock speed. Testing of high-performance circuits for timing failures is becoming very important [1].

A lot of work has been done in the area of delay testing for both combinational and sequential circuits. Most of the proposed delay fault test techniques for sequential circuits involve test methods utilizing scan chains or variable clock speed test application. Inserting scan latches into designs is expensive in terms of chip real estate. On the other hand, testing non-scan circuits using variable clock speeds requires sophisticated testers and clock control circuitry. Due to these drawbacks, delay fault testing in industry has focussed on at-speed test application in non-scan or partial scan circuits [2].

Random test sequences may be used for at-speed testing as well as for simulation-based design verification [3–7]. The presented in [5, 6] research shows that functional tests designed using random test generation exhibit better transition fault coverages than tests produced by deterministic ATPG tools. The main drawback of presented in [4–7] methods is their high computational cost. Therefore, in this paper we are going to propose an approach that allows to speed-up the test generation process.

The rest of the paper is organized as follows. We review the related works in Section 2. In Section 3, we describe the new approach for test generation based on preselection of random generated test subsequences and the experimental results. Section 4 concludes the paper.

Related works

Some relevant papers [3–10], in which various problems of testing of non-scan synchronous sequential circuits are researched, were published in last few years.

In [3] it is shown that a synchronous sequential circuit may have input cubes, or incompletely specified input vectors, that synchronize a subset of its state variables, i.e., it forces them to certain specified values. When an input cube that synchronizes the subset of state variables has a small number of specified inputs, the input vectors covered by it may appear often in a random primary input sequence. As a result, the sequence will force the same values on the state variables repeatedly. This may limit the fault coverage that the sequence can obtain. To address this issue, procedure is described for modifying a random primary input sequence to eliminate the appearance of input vectors that synchronize subsets of state variables. It is demonstrated that this procedure has a significant effect on the fault coverage that can be achieved by random primary input sequences.

The presented in [4] research shows that relatively long random test sequences exhibit better transition fault coverages than tests produced by deterministic ATPG tools. The paper [4] presents an approach for dividing of long test sequences into subsequences. The application of this approach allows increasing the fault coverage of the initial random generated test sequence and minimizing the length of the test by eliminating subsequences that don't

detect new faults.

Two functional fault models for functional delay test generation for non-scan synchronous sequential circuits are proposed in [5]. The obtained results show that the introduced delay test generation method using the presented functional fault models outperforms by the fault coverage the transition test patterns obtained at the gate level by deterministic test pattern generator. Especially, this test generation method obtains good quality results for the circuits, when the long test sequences are needed.

The paper [6] investigates the application of tests that are generated at functional level for detection of gate-level transition faults. Based on experimental results, there is developed a framework of test generation for non-scan sequential circuits. The provided in [5, 6] comparison of experimental results demonstrate the superiority of the delay test patterns constructed at the functional level using the functional fault models against the transition test patterns generated at the gate level by deterministic test pattern generator.

The possibilities of improving random test generation for at-speed testing of non-scan synchronous sequential circuits are explored in [7]. Based on research of distribution of "1" in randomly generated test pattern there is suggested a guidance for management of test generation process. The implementation of semi deterministic algorithms showed that the optimisation of separate steps by construction of test subsequences doesn't improve the final outcome.

Under the approach presented in [8], the input vectors comprising the test sequence are fixed in advance. The process of generating the test sequence consists of ordering a set of precomputed input vectors such that the resulting test sequence has as high fault coverage as possible. The test generation process thus searches a limited set of input vectors for an appropriate order instead of exploring a search space that is limited only by the number of primary inputs of the circuit. The stuck-at faults are considered.

The paper [9] studies the possibility of reducing the complexity of deterministic sequential test generation by using subsets of primary input vectors of limited sizes during test generation for target faults. It proposes a test generation procedure that uses subsets of primary input vectors of size N, for increasing values of N starting with N = 1. The subsets consist of primary input vectors from the test sequence already generated, and of random primary input vectors. The results indicate that all or most of the detectable single stuck-at faults in benchmark circuits can be detected using small subsets of primary input vectors.

The paper [10] suggests an input/output transition (TRIO) fault model for functional test selection at the register-transfer level (RTL). It is defined with respect to the primary inputs, primary outputs, and state variable of the module. The proposed metric has small computational overhead and it is easy to incorporate into existing RTL simulation flows used in design validation. However, this model is approximate because it does not stipulate toggle propagation all the way to the primary outputs.

Preselection of test subsequences in sequential test generation

The sequential circuit is comprised of two parts: the combinational logic and the flip-flops synchronized by a common clock signal. Only the primary inputs (PIs) of the circuit are controllable and the primary outputs (POs) are observable. For delay fault testing of non-scan sequential circuits, test application consists of the following steps: (a) initialization of the circuit to a known state, (b) fault activation to stimulate the fault being tested and (c) propagation of the fault effect to a primary output (PO). It may require a number of input vectors to initialize the circuit and to propagate the fault effects to a PO [2].

We consider functional level delay faults. The ITC'99 benchmark circuits are used for experiments. The models of the benchmark circuits are written in C programming language. The random search is used for test pattern generation.

Further in the paper we will use terms "subsequence" and "functional delay fault" defined in [4] and [6] respectively.

Definition 1. The subsequence is a sequence of input patterns which starts with a set of initialisation patterns. The subsequence is composed of two parts of input patterns: the first part Sub(In) is a set of initialisation patterns that lead the circuit to the known state and the second part Sub(test) is a set of test patterns. The number of input patterns in Sub(test) defines the length of the subsequence.

Definition 2. A functional delay fault (FD) is a tuple (I, O, tI, tO), where I is a primary input x_i (i=1, …, n) or a bit of previous state q_l (l=1, …, v) of the generic cell, O is a primary output y_j (i=1, …, m) or a bit of next state p_k (l=1, …, v), tI is a rising or falling transition at I, and tO is a rising or falling transition at O.

A test subsequence S detects the functional delay fault fd_i if the subsequence S satisfies all three mentioned at the beginning of this section conditions: 1. the circuit is brought into state, required for activation of fd_i; 2. the functional delay fault fd_i is activated; 3. the effect of fd_i is propagated to a primary output. According described in [6, 7] approaches the test generation process proceeds in following way. The whole particular subsequence S_k is generated randomly, then using fault simulation the set of functional delay faults that S_k detects is defined, and if S_k detects any not yet detected faults, S_k is included into resulting test sequence.

As already mentioned, the main drawback of most reviewed in this paper approaches is their high computational cost by dealing with industrial or big benchmark circuits. For example, according to presented in [9] experimental results the test generation for benchmark circuit b14 took 1945.3 hours (81.05 days) on a Linux machine with 3 GHz processors. Therefore, every concept that allows to speed-up the test generation process for non-scan sequential circuits is very valuable.

We introduce now a new fault model, namely, simplified functional delay fault (SFD). The definition of

simplified functional delay fault is the same as for functional delay fault (see Definition 2). Only the conditions of detection of SFD are changed. We say that a test subsequence S detects the simplified functional delay fault sfd_i if on the subsequence S there are satisfied following two conditions: 1. the circuit is brought into required for activation of sfd_i state; 2. the simplified functional delay fault sfd_i is activated. Thus, we have discarded the phase of propagation of the fault effect to a primary output. During our experiments with ITC'99 benchmark circuits we have observed that many functional delay faults, which are activated on particular test subsequence, are successfully propagated to primary output. This observation was the background for introducing of simplified functional delay fault model.

Next, we propose to use two stages in random test generation process. In the first stage we suggest to employ for selection of test subsequences the simplified functional delay fault model. We call this stage "Test Preselection". Let's say, T is the initial set of randomly generated test subsequences. Hence, we get the set $T_{Pres} \subseteq T$ after the test preselection stage. Then in the second stage, we consider only the set of preselected test subsequences T_{Pres} and use for fault simulation already the usual functional delay fault model. Thus, we get the resulting test set $T_{Res} \subseteq T_{Pres}$.

We implemented two test generation procedures: one stage Procedure 1, in which we apply for test subsequences selection only functional delay fault model and two stages Procedure 2, which uses the simplified functional delay fault model for the test subsequences preselection in the first stage. As it is well known, random search requires some termination condition to be defined. The ratio of detected faults may be the best condition. However, the number of detectable functional delay faults is not known. The simplest termination conditions are the number of randomly generated test subsequences or/and test generation time, but these conditions say nothing about the quality of found solution. It is possible to relate the condition of the termination of random search dynamically to the number of the last selected subsequence as well. The generation can be terminated when the total number of generated random subsequences exceeds the number of the last selected subsequence multiplied by a coefficient K [7].

Table 1. Comparison of test generation results

Circuit	Length	Procedure 1			Procedure 2					Imp. in %
		$\|T\|$	$\|T_{Res}\|$	Det. FD	$\|T\|$	$\|T_{Pres}\|$	Det. SFD	$\|T_{Res}\|$	Det. FD	
b12	1600	130998	48	642	876183	54	842	52	734	14.33
b14	200	75860	2531	19255	436101	2966	22286	1956	19880	3.25
b15	60	90120	1069	11381	1344420	3300	39643	1377	15656	37.56
b17	80	5940	406	5651	169350	2012	30588	812	9067	60.45
b20	250	9900	2558	38242	62048	3762	45934	2550	39631	3.63
b22	100	11212	3311	51322	60197	5383	70630	3620	54926	7.02
Average		54005			491383					21.04

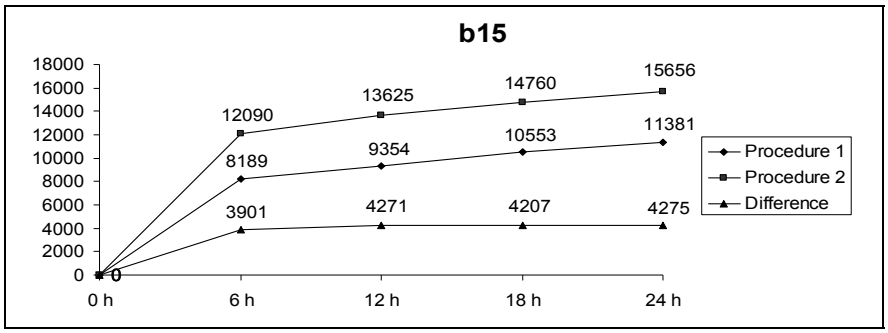

Fig. 1. Distribution of number of detected FD over time

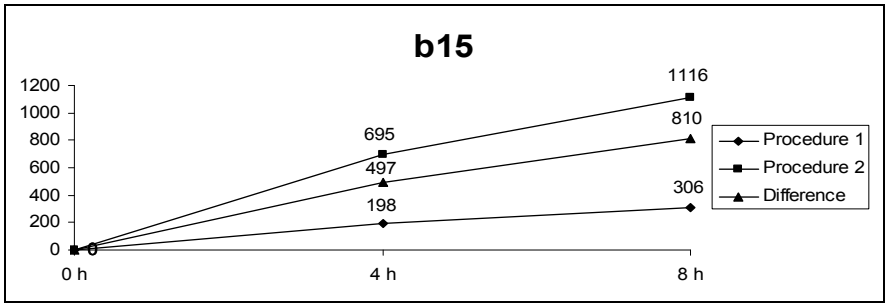

Fig. 2. Numbers of detected FD in test supplement mode

In this paper we are going only to compare the efficiency of Procedures 1 and 2, therefore, the test generation time of 24 hours was chosen as termination condition. We used for experiments a Windows machine with 3.4 GHz processors. 24 hours were spent for every benchmark circuit by applying for test generation Procedure 1 and the next 24 hours by applying Procedure 2. The comparison of test generation results of Procedures 1 and 2 is provided in Table1.

In Table 1 for each circuit, circuit name (Circuit), the applied length of test subsequences (Length), total number of randomly generated subsequences ($|T|$), number of subsequences in resulting test set ($|T_{Res}|$), number of detected functional delay faults (Det. FD), number of preselected test subsequences ($|T_{Pres}|$), number of detected simplified functional delay faults (Det. SFD) and improvement in per cent (Imp. in %) are provided. The improvement is calculated as follows: (Det. FD(Procedure 2)/Det. FD(Procedure 1) - 1)*100.

The application of Procedure 2 has allowed in all cases to improve the numbers of detected functional delay faults. This improvement ranges from 3.25% (circuit b14) to 60.45 (circuit b17) and is 21.04% high on average. Another important sighting is that Procedure 2 is 9 times on average faster than Procedure 1: the total number (491383 on average) of randomly generated and considered test subsequences of Procedure 2 is 9 times higher than number (54005 on average) of Procedure 1. Therefore, we can conclude that the test preselection is an effective technique for speeding-up the random test generation process for functional delay faults.

In Figure 1, the distribution of number of detected functional delay faults over time is presented. We can see that the biggest difference of number of detected FD occurs in the first hours of test generation. Later the growth of number of detected FD stabilises for both procedures. We observed such distribution for all considered benchmark circuits as well. Hence, it is possible to make an assumption that Procedure 2 looses its efficiency over time or there is another reason for this appearance. We think that we have an explanation of this fact. There are many functional delay faults that are easy detectable; and they are detected at the beginning of test generation process. Simply Procedure 2 begins earlier to deal with faults that are more difficult to detect. The presented in Figure 2 diagram supports this assumption.

The achieved best number of detected FD after 24 hours of test generation for circuit b15 was 15656. We applied both procedures for 8 hours every in test supplement mode. Figure 2 shows the numbers of additionally detected FD for both procedures. We can see that Procedure 2 exhibits its unambiguous superiority over Procedure 1 again.

At the beginning of this section we have mentioned that many functional delay faults, which are activated on particular test subsequence (in our case it conforms to detection of simplified functional delay faults), are successfully propagated to primary output (it conforms to detection of functional delay faults). In Table 2, we present corresponding data that are obtained after application Procedure 2 (calculation time - 24 hours).

Table 2. The propagation of SFD

Circuit	Det. SFD	Det. FD	Propagated in %
b12	827	734	88.75
b14	22286	19880	89.20
b15	39643	15656	39.49
b17	30588	9067	29.64
b20	45934	39631	86.28
b22	70630	54936	77.78
Average			68.53

In Table 2 for each circuit, circuit name (Circuit), number of detected simplified functional delay faults (Det. SFD), number of detected functional delay faults (Det. FD), and per cent of successfully propagated to primary output SFD (Propagated in %) are provided. The data from Table 2 show that over 30% on average of activated functional delay faults are not propagated to primary output. Thus, there emerges a question that probably by using preselection on set T of test subsequences we miss to many functional delay faults that the whole set T detects.

Therefore, we made an additional experiment. For circuit b15 we generated 10000 random test subsequences of length 60. Then we applied Procedures 1 and 2 for this test sequence. Procedure 2 was employed using various numbers of detection of SFD. The results of this experiment are reported in Table 3.

Table 3. Using n-detection for test preselection

Procedure 1			
-	6668	206	-
Procedure 2			
N	Det. FD	Time	Loss in %
50	6668	127	0.00
20	6661	94	0.10
10	6619	77	0.73
3	6498	35	2.55
1	6390	24	4.17

In Table 3, number of detections of SFD (N), number of detected functional delay faults (Det. FD), calculation time (Time) and per cent of loss of detectable on considered test sequence FD (Loss in %) are provided. At least, this experiment shows that loss of detectable on considered test sequence FD is not high and that the use of big numbers of detection of SFD allows us to reduce this loss to zero. However, increasing of N leads to higher computational times. Therefore, there is needed to find a reasonable compromise between calculation time and increasing of number of detections of SFD. This problem requires a deeper research and will be investigated in the future.

Conclusions

We introduced a new fault model, simplified functional delay fault model, and proposed to use two stages in random test generation process. In the first stage

called test subsequences preselection we employed for selection of test subsequences the simplified functional delay fault model. Then in the second stage, we consider only the set of preselected test subsequences and use for fault simulation already the usual functional delay fault model. Experimental results were presented to demonstrate the effectiveness of proposed approach. The application of two stages in random test generation process allowed to speed-up test generation 9 times on average and to improve the detection of functional delay faults at 21% on average. It was shown that application of n-detection of simplified functional delay faults in the test subsequences preselection stage may be a perspective way for further test quality enhancement. This problem will be investigated more detailed in the future.

References

1. **Krstic A., Cheng K. T.** Delay Fault Testing for VLSI Circuits. – Kluwer Academic Publishers, 1998. – 191 p. DOI: 10.1007/978-1-4615-5597-1.
2. **Pant P., Chatterjee A.** Path–delay fault diagnosis in non–scan sequential circuits with at–speed test application // Proceedings of the International Test Conference (ITC'2000), 2000. – P. 245–252.
3. **Pomeranz I., Reddy S. M.** Primary input vectors to avoid in random test sequences for synchronous sequential circuits // IEEE transactions on computer–aided design of integrated circuits and systems, 2008. – No. 27(1). – P. 193–197.
4. **Bareisa E., Jusas V., Motiejunas K., Seinauskas R.** The non–scan delay test enrichment based on random generated long test sequences // Information Technology and Control, 2010. – No. 39(4). – P. 251–256.
5. **Bareisa E., Jusas V., Motiejunas K., Seinauskas R.** Generating Functional Delay Fault Tests for Non–Scan Sequential Circuits // Information Technology and Control, 2010. – No. 39(2). – P. 100–107.
6. **Bareisa E., Jusas V., Motiejunas K., Seinauskas R.** On delay test generation for non–scan sequential circuits at functional level // Electronics and Electrical Engineering. – Kaunas: Technologija, 2011. – No. 3(109). – P. 67–70.
7. **Bareisa E., Jusas V., Motiejunas K., Seinauskas R.** An Investigation of Possibilities of Improving Random Test Generation for Non–scan Sequential Circuits // Electronics and Electrical Engineering. – Kaunas: Technologija, 2011. – No. 8(114), – P. 11–15.
8. **Pomeranz I., Reddy S. M.** TOV: sequential test generation by ordering of test vectors // IEEE transactions on computer–aided design of integrated circuits and systems, 2010. – No. 29(3). – P. 454–465.
9. **I. Pomeranz.** Subsets of Primary Input Vectors in Sequential Test Generation for Single Stuck–at Faults. // IEEE transactions on computer–aided design of integrated circuits and systems, 2011. – No. 30(10). – P. 1579–1583.
10. **Kang J., Seth S. C., Gangaram V.** Efficient RTL Coverage Metric for Functional Test Selection // Proceedings of the 25th IEEE VLSI Test Symposium, 2007. – P. 318–324.

Received 2011 09 24
Accepted after revision 2012 01 09

E. Bareisa, V. Jusas, K. Motiejunas, R. Seinauskas, L. Motiejunas. Application of Preselection of Test Subsequences in Sequential Test Generation for Functional Delay Faults // Electronics and Electrical Engineering. – Kaunas: Technologija, 2012. – No. 2(118). – P. 33–37.

Testing of high-performance circuits for timing failures is becoming very important. Testing of non-scan circuits using variable clock speeds requires sophisticated testers and clock control circuitry. Due to these drawbacks, delay fault testing in industry has focussed on at-speed test application in non-scan or partial scan circuits. In the paper, we introduced a new fault model, simplified functional delay fault model, and proposed to use two stages in random test generation process. In the first stage called test subsequences preselection we employed for selection of test subsequences the simplified functional delay fault model. Then in the second stage, we consider only the set of preselected test subsequences and use for fault simulation already the usual functional delay fault model. Experimental results were presented to demonstrate the effectiveness of proposed approach. Ill. 2, bibl. 10, tabl. 3 (in English; abstracts in English and Lithuanian).

E. Bareiša, V. Jusas, K. Motiejūnas, R. Šeinauskas, L. Motiejūnas. Testinių segmentų išankstinis atrinkimas funkciniams vėlinimo gedimų testams generuoti nuosekliose schemose // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2012. – Nr. 2(118). – P. 33–37.

Šiuolaikinės schemos yra labai jautrios vėlinimo gedimams. Nuoseklioms schemoms testuoti, naudojant kintamus sinchronizacijos dažnius, reikia sudėtingos testavimo įrangos. Todėl pramonėje vėlinimų gedimų tikrinimas dažniausiai remiasi testavimu naudojant schemos darbinį dažnį. Šiame straipsnyje pateiktas naujas gedimų modelis, supaprastintas funkcinis vėlinimo gedimas, ir pasiūlyta testų nuoseklioms schemoms projektavimo procese naudoti du atsitiktinio testų generavimo etapus. Pirmame etape, vadinamame išankstiniu testinių segmentų atrinkimu, testams generuoti naudojamas supaprastintas funkcinis vėlinimo gedimo modelis, o po to antrame etape yra nagrinėjami tiktai atrinkti testiniai segmentai ir jau naudojamas įprastinis funkcinis vėlinimo gedimo modelis. Straipsnyje pateikti eksperimentiniai rezultatai patvirtina pasiūlytojo metodo efektyvumą. Il. 2, bibl. 10, lent. 3 (anglų kalba; santraukos anglų ir lietuvių k.).