ELECTRONICS AND ELECTRICAL ENGINEERING
ISSN 1392 – 1215 *2009. No. 3(91)*
ELEKTRONIKA IR ELEKTROTECHNIKA

ELECTRONICS
T170
ELEKTRONIKA

On the Enrichment of Static Functional Test

E. Bareiša, V. Jusas, K. Motiejūnas, R. Šeinauskas
Software Engineering Department, Kaunas University of Technology,
Studentų str. 50-406, Kaunas, Lithuania, phone: +370 37 454229, e-mail: kestutis.motiejunas@ktu.lt

Introduction

The complexity of digital devices is growing continuously. The testing problem is becoming the most crucial part of overall design process that delays the time-to-market of the digital device. In order to alleviate the test generation complexity and to reduce the time-to-market, one needs to model the actual defects that may occur in a digital device with fault models at higher levels of abstraction. This process of fault modelling considerably reduces the burden of testing because it obviates the need for deriving tests for each possible defect and is performed in parallel with other design activities. Such a parallelism enables to reduce the time-to-market. The single fault at the higher level of abstraction is mapped to many physical defects. This also makes the fault model more independent of the fabrication technology.

The test designer prepares the functional test according to the specification of the device under test. The functional test is used to verify the next steps of the design and it can be used for the development of the manufacturing test as well. Such a test usually verifies the function of the device and it cannot guarantee the full coverage of the gate level faults of the device. Therefore, when the synthesis of the device is completed, the list of undetected faults is formed, and the deterministic methods are used to detect the faults from this list.

Models of physical and fabrication faults are needed at higher levels of abstraction in order to be able to develop tests from functional or behavioural descriptions. Many efforts have been devoted to the problem of finding behavioural level fault model [1-9]. However, no such fault model has been discovered at behavioural or higher level which is universally accepted.

Behavioural level fault models can be broadly classified into two main categories: 1) fault models related to the description code [1-3]; 2) black box fault models related to input stimuli and output responses [4-9]. Testing at higher level of abstraction has a lot in common with software testing. Therefore the pattern generation methods based on the fault model related to the description code can be further classified, namely, code oriented methods and fault oriented methods. The code oriented methods exploit the most widely used metrics developed for automated software testing: statement coverage, branch coverage and path coverage. Although there are similarities there are also important differences due to different sources of errors/faults and models in these two cases. The purpose of software validation is to detect design errors whereas the purpose of testing is to detect physical defects and fabrication faults.

Black box fault models are more universal as they do not depend on the description code; however, such black box fault models are of little use still. During design process the software prototype of circuit is created according to the specification. The software prototype simulates the functions of the circuit, enables to calculate the output values according to the input values, and can be regarded as the black box model of the circuit. The functional test can be generated on the base of the software prototype. The most prominent features of these models are the following: 1) a circuit is treated as a single module; 2) limitation of the number of circuit‘s inputs was not observed; 3) the tests constructed on the base of these models have the manageable size.

The functional test is based on the function of the circuit, which can be designed in many ways. The possible defects of the circuit depend on the implementation. The test is usually developed according to the specific implementation and it is designed to detect the defects of this particular implementation. The manufacturing test can be developed only on the base of the specific implementation. Meanwhile, the functional test is not related to the particular implementation because it is generated from a circuit’s specification rather than its gate level implementation. The implementation independence of functional test has several advantages over implementation-dependent test. The functional test can be used to correct testability problems early in the design process [10], to identify the design errors [11], to test many potential implementations [12-14], and to detect hard-to-detect faults at the gate level implementation [13, 15].

One of the approaches for deriving tests to achieve high defect coverage is based on the generation of n – detection tests. An n -detection test is one where each fault is detected either by n different tests, or by the maximum number of different tests that can detect the fault if this number is smaller than n. In this work, we describe a

postprocessing procedure for static functional test enrichment. The procedure targets a test pattern set T for pin pair (PP) faults [7]. The proposed procedure modifies the test patterns in T so that the number of detections of PP faults is increased.

The remainder of this paper is organized as follows: in Section 2 we review the related work. Section 3 presents the procedure of functional test enrichment. Section 4 presents experimental results, and Section 5 concludes this paper.

Related work

The possibilities of test quality enhancement using various approaches are analyzed in [16-29].

The authors of [16] present a method to enrich the transition-fault test with additional test patterns of a certain property. This property is related to the paths sensitized by the test pattern. The idea is to add a number of new test patterns in the set such that they still detect the targeted transition faults but through paths that have not been sensitized by the original test pattern set. The generated test pattern sets have higher quality since events propagate through many critical paths and thus are more likely to detect a delay violation in the circuit. It is expected that, on average, this procedure will allow the number of paths along which the event propagates to increase proportionally to the number of test patterns per transition fault [16].

To improve the quality of tests for path delay faults, an m-tuple test generation procedure for path delay faults is described in [17]. Under an m-tuple test, each m-tuple of target faults is detected by at least one test pattern. An m-tuple test has advantages similar to an n -detection test in that it results in several test patterns for every target path delay fault p, thus increasing the likelihood of testing p under worst-case delay conditions. In addition, it increases the likelihood of accidentally detecting non-target path delay faults [17].

Another possibility of test quality enhancement called sensitivity of adjacent input patterns is proposed in [18]. Sensitive adjacent input vectors can be generated for each test pattern of the test set. Since a change in the value of a single input of sensitive adjacent input vectors changes the output vector, it is likely that the presence of a fault on a path from a sensitive input to a sensitive output will be detected. Generated sensitive adjacent input vectors are likely to be sensitive to the presence of a defect, and are likely to result in higher fault coverage [18].

The authors of [19] suggest complementing the existent test suites of the IP core with all sensitive adjacent patterns or with the subset of sensitive adjacent patterns. Then the suitable test patterns for the synthesized gate level implementation have to be selected on the base of the fault simulation. The presented in [19] experiment proves that such a complement enhances the test quality for any synthesized IP core gate level description. The authors point out that the practice of sensitive adjacent patterns is a cheap way to adopt test patterns for the re-synthesized gate level description of IP core, because the fault simulation is not so critical task as test generation [19].

A deterministic procedure of adjacent stimuli generation was suggested in [20]. It is based on the assumption that input stimuli that are similar to test patterns have good testing features. The search among such input stimuli improves the overall efficiency and the convergence speed of the search. It is evaluated that the adjacent stimuli generation allowed improving the efficiency of random search up to 30%. Consequently, it is recommended the integrated use of random and adjacent stimuli generation during functional test design process [20].

The generation of n-detection tests and their capabilities in detecting untargeted faults and defects were studied in [21-25]. An n -detection test detects each target fault n times, by n different test patterns. By increasing the number of detections of target faults, n -detection test generation for n >1 increases the likelihood of detecting untargeted faults and defects. Generation of an n -detection test requires repeated applications of a test generation process to target faults that are not yet detected n times. Each time a fault is targeted, a different test pattern must be generated for it. This increases the complexity of test generation.

A procedure for forming n-detection tests without applying a test generation procedure to target faults is described in [26]. The proposed procedure accepts a one-detection test. It extracts test cubes for target faults from one-detection test and then merges the cubes in different ways to obtain an n-detection test. Merging of cubes does not require test generation. Fault simulation is required for extracting test cubes for target faults [26].

N-detection may lead to large tests where many test patterns do not help increase the defect coverage [27]. The problem of control of test size addition is considered in [27-29].

The authors of [27] introduced variable n-detection tests where different target faults are targeted different number of times. In a variable n-detection test, only selected faults are targeted n times. Other faults are targeted between 1 and n-1 times. The motivation for introducing variable n-detection tests was to control the size of test pattern set as n was increased. The number of times each fault is targeted is determined by a parameter that measures the usefulness of multiple test patterns for the fault in detecting defects. This parameter is based on the number of paths through the fault site [27]. The use of variable number of fault detections while transforming the pin pair test into functional delay test is suggested in [28]. The performed experiments show the effectiveness of this proposal. The restriction of number of fault detections allowed shortening the test size almost twice [28].

In the paper [29], three parameters of an n-detection test to measure the saturation of the test generation process were defined: 1) the fraction of faults detected n times or less by the test; 2) the fraction of faults detected fewer than n times by the test; and 3) the test set size relative to the size of a one detection test. Based on these parameters and the rationale for computing n-detection tests, the authors defined saturation to occur at the value of n where one of the three parameters reaches a threshold specified for it. The thresholds were selected based on experimental results [29].

The main drawback of all reviewed techniques is the test size addition. Another disadvantage of almost all mentioned approaches lies in use of test generation for enrichment of test pattern sets.

Procedure of enrichment of functional test

In this section we give a detailed description of the proposed procedure. We consider the enrichment of static functional test that is generated for detection of pin pair faults. The pin pair fault model is exhaustively described in [7].

Next we provide a brief presentation of the main concepts of this model. The behavioural view or the "black box" represents the system by defining the behaviour of its outputs according to the values applied on its inputs without the knowledge of its internal organization. In this case, the input-output relationship can be determined only. Let the circuit have a set of inputs $X = \{x_1, x_2, ..., x_i, ... , x_n\}$ and a set of outputs $Z = \{z_1, z_2, ... ,z_j, ... ,z_m\}$. The pin fault model considers the stuck-at-0/1 faults occurring at the module boundary, and has a weak correlation with the circuit's physical faults. We write x_i^1 and x_i^0 for the input stuck-at-1/0 faults, and z_j^1 and z_j^0 for the output stuck-at-1/0 faults. There are 2*n+2*m possible pin stuck-at faults. Input-output pin stuck-at fault pairs (x_i^t, z_j^k), t=0,1, k=0,1 are called pin pair (PP) faults. The number of possible pin pair faults of the circuit is at most 4*n*m.

In the rest of the paper the following notations will be used:

- T – the test pattern set;
- $t_{k,i}$ – the signal value (1 or 0) of test pattern T_k ($T_k \in T$) on circuit input i;
- F – the set of PP faults of particular circuit;
- F_k – the set of PP faults that are detected on test pattern T_k ($T_k \in T$);
- n – the number of circuit inputs.

The pseudocode of procedure for Enrichment of Static Functional Test (procedure EoSFT) is shown in Fig. 1.

The procedure EoSFT modifies each test pattern T_k of the set T in such way that the modified test pattern T_k^* detects all detectable on the initial pattern T_k PP faults and, probably, some additional not detectable on T_k PP faults. Therefore, the enriched test pattern set T^E may detect some PP faults that are not detectable on the test pattern set T or, at least, increases the number of detections of some PP faults. The most prominent features of the proposed static functional test enrichment procedure are: 1) procedure EoSFT does not expand the initial test pattern set T, i.e. there is no test size addition; 2) procedure EoSFT does not require test generation. The described approach enriches the test patterns using PP fault simulation. Thus, the computing time of the procedure EoSFT depends linearly on the test size.

The procedure EoSFT can be incorporated into test generation system and then used as dynamic test enrichment procedure. At that point the proposed procedure processes each test pattern before it is included into test pattern set T. In this case many faults not yet detectable on the generated test pattern set T may be detected, particularly at the beginning of the test generation. The outcome of dynamic test enrichment should be the reduction of test size.

```
procedure EoSFT
    INPUT:   circuit C, corresponding PP fault
             set F, test pattern set T, number of
             iterations It
    OUTPUT:  enriched test pattern set T^E
1.  l=1
2.  repeat
3.      for each T_k ∈ T do
4.          determine F_k
5.          for i=1 to n do
6.              t_k,i=NOT(t_k,i)
7.              determine F_k^*
8.              if F_k ⊆ F_k^* then
9.                  F_k=F_k^*
10.             else
11.                 t_k,i=NOT(t_k,i)
12.         end
13.     end
14.     l=l+1
15. until k > It
end procedure
```

Fig. 1. The pseudocode of procedure EoSFT

Additionally, procedure EoSFT can be easily modified on purpose to use it for the relaxation of test pattern sets. The relaxation of test patterns means changing, where it is possible, of fully specified test pattern bits into unspecified (don't care) bits. The modification affects only one line (Line 9) of the procedure EoSFT. The Line 9 has to be changed from $F_k=F_k^*$ to $t_{k,i}$="unspecified". The motivation behind relaxing of test pattern sets is to make these tests amendable to addressing additional issues beyond detection of the targeted faults [24]. For example, the unspecified bits can be specified in such manner that power dissipation during test application is minimized or the unspecified bits can be specified appropriately to detect additional faults. Such flexible tests are important in various compression schemes for on-chip or off-chip test embedding [24].

Experimental results

In this section we present results of the application of the proposed test enrichment procedure to ISCAS'85 benchmark circuits.

The test pattern sets for PP faults were generated for the black box model of the circuits [7]. Remind the black box model represents a system by defining the behaviour of its outputs according to the values applied to its inputs without the knowledge of its internal organization. The black box models written in the programming language C were used by the test generation for the PP faults.

The results of test pattern set enrichment are reported in Table 1. The initial test pattern sets T1 and T2 were obtained using two different test pattern generation systems. Both tests expose 100% coverage of targeted PP faults. In Table 1, after the circuit name we show the

number of detectable pin pair faults, and the average number of detections under T1, computed as follows. The sum of detections of all faults is divided by number of detectable PP faults. Next, we show the number of detections under enriched test pattern set $T1^E$, and the improvement of the average number of detections, expressed in percent. Columns 6-8 list the same data for the test pattern set T2.

Table 1. Results of test pattern set enrichment

Circuit	PP faults	Av. T1	Av. $T1^E$	Imp. in %	Av. T2	Av. $T2^E$	Imp. in %
c432	540	4.6	6.8	49.4	4.8	7.5	55.8
c499	5184	10.1	10.2	0.6	9.8	9.9	0.6
c880	1326	18.8	22.3	19.1	20.4	24.2	18.9
c1355	5184	9.6	9.6	0.5	9.9	10.0	0.5
c1908	3004	23.9	24.2	1.2	24.0	24.3	1.2
c2670	3320	19.2	21.8	13.3	36.0	39.8	10.7
c3540	2588	20.3	20.9	2.8	23.1	23.8	2.8
c5315	10540	40.6	44.1	8.6	39.7	43.1	8.6
c6288	3068	27.0	27.0	0.0	25.5	25.5	0.0
c7552	12188	69.6	73.6	5.8	76.7	81.6	6.4
Aver.	4694	24.4	26.1	10.1	27.0	29.0	10.6

The following points can be seen from Table 1. The procedure EoSFT was able to enrich the PP fault test pattern set almost in all cases, there is only one exception, namely, the circuit c6288. Independently from initial test pattern set, the average number of fault detections was increased at 10% on average. The improvement of the average number of detections ranges from 0 to 56 percent. The experiment shows that procedure EoSFT converges after two iterations.

As already mentioned, the proposed approach can serve and for other purposes: dynamic test enrichment and after slight modification for test relaxation. However, these purposes are secondary in this paper, and we performed only mini experiments on circuit c880. We obtained following results.

The PP fault test pattern set was constructed using portions of 30 test patterns. The first portion, generated without and with dynamic test pattern enrichment and included into set T, produced 61.9 % and 67.3% PP fault coverage respectively. The application of dynamic test pattern enrichment brought 5.4 % improvement of fault coverage. Then the 67.3% already detectable on test pattern set T PP faults were excluded from fault list F, and the second portion of 30 test patterns was generated in the same way. The outcome was 81.4% and 83.3% PP fault coverage respectively (improvement 1.9%). The results for third portion were 85.4% and 86.5% (improvement 0.9%) and so on. At the end of this experiment we got test pattern set T of size 187. The size of test pattern set generated without test pattern enrichment was 381. Therefore, the application of our approach for dynamic test pattern enrichment reduced the test size at 51%. The application of modified procedure EoSFT for test relaxation changed 34% fully specified test set bits into unspecified in not enriched test pattern set T.

Many authors state that the n-detection tests are effective in detecting untargeted faults and defects [21-25]. To examine the influence of the improvement of average number of detections in detecting untargeted faults, we simulated stuck-at and transition faults under the test pattern sets T and T^E. Note that the tests for PP faults are generated at functional level and then applied for detection of structural level faults.

Table 2. Results of stuck-at fault simulation

Circuit	T1	$T1^E$	T2	$T2^E$
c432	98.84	99.20	95.71	95.93
c499	100	100	100	100
c880	100	100	99.83	99.83
c1355	100	100	100	100
c1908	96.10	96.10	94.64	94.82
c2670	99.75	99.81	99.43	99.43
c3540	99.00	99.01	98.19	98.19
c5315	100	100	100	100
c6288	100	100	99.97	99.97
c7552	99.80	99.83	99.45	99.61
Average	99.35	99.40	98.72	98.78

Table 3. Results of transition fault simulation

Circuit	$T1_D$	$T1_D{}^E$.	$T2_D$	$T2_D{}^E$
c432	87,28	87,86	84,38	84,52
c499	91,23	91,60	91,46	91,55
c880	90,90	90,98	89,61	90,28
c1355	92,36	92,39	92,42	92,48
c1908	81,56	81,58	80,59	80,67
c2670	90,44	90,79	89,83	90,03
c3540	88,28	88,34	85,70	85,75
c5315	97,94	97,94	97,94	97,94
c6288	98,72	98,72	98,62	98,62
c7552	97,30	97,84	96,91	97,16
Average	91,60	91,80	90,75	90,90

The results of fault simulation are reported in Table 2 and 3. The static tests don't suit directly for detecting of delay faults. Therefore, they were transformed into functional delay tests according Rule 3 presented in [30]. Such transformation increases the test size twice. The test pattern sets for detecting of functional delay faults $T1_D$, $T1_D{}^E$, $T2_D$ and $T2_D{}^E$ were obtained from T1, $T1^E$, T2 and $T2^E$ respectively. In Table 2, after the circuit name we show the stuck-at fault coverage of test pattern sets T1, $T1^E$, T2 and $T2^E$. Table 3 lists the transition fault coverage of test pattern sets $T1_D$, $T1_D{}^E$, $T2_D$ and $T2_D{}^E$. The fault coverage is expressed in percent.

From Tables 2 and 3 it can be seen that the PP fault tests are very effective in detecting of untargeted stuck-at faults, whereas, the functional delay tests obtained from them expose moderate quality in regard of detecting of untargeted transition faults. The test enrichment

accomplished using proposed procedure EoSFT contributed in test quality improvement in all cases if we take in account average fault coverage. If we examine tests for separate circuits and exclude from examination such tests that expose 100% fault coverage or no improvement of the average number of detections (circuit c6288), we can see that the test enrichment contributed to augmentation of fault coverage of untargeted faults in 23 cases of 29. Thus, the proposed postprocessing procedure for static functional test enrichment is a cheap way to enhance the quality of initial test pattern set.

Concluding remarks

We described an approach for static functional test enrichment. The proposed postprocessing procedure modifies each test pattern of the test in such way that the modified test pattern detects all detectable on the initial test pattern pin pair faults and some additional pin pair faults. The enriched test pattern set may detect some pin pair faults that are not detectable on the initial test pattern set T or, at least, increases the number of detections of pin pair faults. The test enrichment procedure does not increase the test size and it is fast because the procedure does not require test generation. The described approach enriches the test patterns using pin pair fault simulation. The performed experiments demonstrated effectiveness of the proposed approach. We showed that our test enrichment procedure can be incorporated into test generation system and then used as dynamic test enrichment procedure or after slight modification; it can be applied for relaxation of test pattern sets.

References

1. **Corno F., Prinetto P., Sonza Reorda M.** Testability analysis and ATPG on behavioral RT-level VHDL // Proceedings of IEEE International Test Conference. – October 1997. – P. 753–759.
2. **Chiusano S., Corno F., Prinetto P.** A Test Pattern Generation Algorithm Exploiting Behavioral Information // Proceedings of Seventh Asian Test Symposium (ATS'98). – Singapore. – December 1998. – P. 480–485.
3. **Rudnick E. M., Vietti R., Ellis A., Corno F., Prinetto P., Sonza Reorda M.** Fast Sequential Circuit Test Generation Using High-Level and Gate-Level Techniques // Proceedings of IEEE Design, Automation and Test in Europe. – Feb. 1998. – P. 570–576.
4. **Psarakis M., Gizopoulos D., Paschalis A.** Test generation and fault simulation for cell fault model using stuck-at fault model based test tools // Journal of Electronic Testing. – 1998. – Vol. 13. – P. 315–319.
5. **Yi J. and Hayes J. P.** A Fault Model for Function and Delay Testing // Proc. of the IEEE European Test Workshop, ETW'01. – 2001. – P. 27–34.
6. **Yi J. and Hayes J. P.** The Coupling Model for Function and Delay Faults // Journal of Electronic Testing: Theory and Applications. – 2005. – Vol. 21, No. 6. – P.631–649.
7. **Bareiša E., Jusas V., Motiejūnas K., Šeinauskas R.** The Realization-Independent Testing Based on the Black Box Models // Informatica. – Vilnius, Institute of Mathematics and Informatics. – 2005. – Vol. 16, No. 1. – P. 19–36.
8. **Bareiša E., Jusas V., Motiejūnas K., Šeinauskas R.** Functional Delay Clock Fault Models // Information Technology and Control. – Kaunas: Technologija, 2008. – Vol. 37, No. 1. – P. 12–18.
9. **Bareiša E., Jusas V., Motiejūnas K., Šeinauskas R.** Functional Delay Test Quality Assessment at High Level of Abstraction // Information Technology and Control. – Kaunas: Technologija, 2007. – Vol. 36, No. 1. – P. 7–15.
10. **Pomeranz I. and Reddy S. M.** On Testing Delay Faults in Macro-based Combinational Circuits // Proceedings of Int. Conf. Computer-Aided Dsign. – San Jose, CA. – 1994. – P. 332–339.
11. **Ferrandi F., Fummi F., Pravadelli G., Sciuto D.** Identification of Design Errors Through Functional Testing // IEEE Transactions On Reliability. – December 2003. – Vol. 52, No. 4. – P. 400–412.
12. **Kim H. and Hayes J. P.** Realization-Independent ATPG for Designs with Unimplemented Blocks // IEEE Trans. on CAD. – 2001. – Vol. 20, No. 2. – P. 290–306.
13. **Ferrandi F., Fummi F., Sciuto D.** Implicit Test Generation for Behavioral VHDL Models // Proceedings of International Test Conference. – 18-23 October 1998. – P. 587–596.
14. **Bareiša E., Jusas V., Motiejūnas K., Šeinauskas R.** The Use of Software Prototype for Verification Test Generation. Information Technology and Control. – Kaunas: Technologija, 2008. – Vol. 37, No. 4. – P. 265–274.
15. **Jusas V., Motiejūnas K.** Generation of Functional Delay test with Multiple Input Transitions. Information Technology and Control. – Kaunas: Technologija, 2007. – Vol. 36, No. 3. – P. 259–267.
16. **Neophytou N., Michael M. K. and Tragoudas S.** Functions for Quality Transition-Fault Tests and Their Applications in Test-Set Enhancement // IEEE Transactions on Computer Aided Design of Integrated Cicuits and Systems. – 2006. – Vol. 25, No. 12. – P. 3026–3035.
17. **Pomeranz I. and Reddy S. M.** Tuple Detection for Path Delay faults: A Method for Improving Test Set Quality // Proceedings of the 18th International Conference on VLSI Design held jointly with 4th International Conference on Embedded Systems Design. – 2005. – P. 41–46.
18. **Pomeranz I. and Reddy S. M.** Pattern Sensitivity: A Property to Guide Test Generation for Combinational circuits // Proceedings of 8th Asian Test Symposium. – 1999. – P. 75–80.
19. **Bareiša E., Jusas V., Motiejūnas K., Packevičius Š., Šeinauskas R.** The Improvement of Test Independence from Circuit Realization // Information technology and control. – Kaunas: Technologija, 2004. – No. 4(33). – P. 45–52.
20. **Bareiša E., Jusas V., Motiejūnas K., Šeinauskas R.** Functional Test Generation Based on Combined Random and Deterministic Search Methods // Informatica. – 2007. – Vol. 18, No. 1. – P. 3–26.
21. **Benware B., Schuermyer C., Tamarapalli N., Tsai K.-H., Ranganathan S., Madge R., Rajski J., and Krishnamurthy P. I**mpact of multiple-detect test patterns on product quality // Proceedings of Int. Test Conf. – Sep. 2003. – P. 1031–1040.
22. **Venkataraman S., Sivaraj S., Amyeen E., Lee S., Ojha A., and Guo R.** An experimental study of n-detect scan ATPG patterns on a processor // Proceedings of VLSI Test Symp. – Apr. 2004. – P. 23–28.
23. **Pomeranz I. and Reddy S. M.** Worst-case and average-case analysis of n-detection test sets // Proceedings of Des. Autom. Test Eur. Conf. – Mar. 2005. – P. 444–449.
24. **Neophytou S., Michael M. K.** On the Relaxation of n-detect Test Sets // Proceedings of the 26th IEEE VLSI Test Symposium. – 2008. – P. 187–192.
25. **Geuzebroek J., Marinissen E. J., Majhi A., Glowatz A. and Hapke F.** Embedded Multi-Detect ATPG and Its Effect

on the Detection of Unmodeled Defects // Proceedings of the IEEE International Test Conference ITC 2007. – 2007. – P. 1–10.
26. **Pomeranz I. and Reddy S. M.** Forming n-detection test sets without test generation // ACM Transactions on Design Automation of Electronic Systems. – 2007. – Vol. 12, No. 2.
27. **Pomeranz I. and Reddy S. M.** On n-Detection Test Sets and Variable n-Detection Test Sets for Transition Faults // IEEE Transactions on Computer Aided Design of Integrated Cicuits and Systems. – 2000. – Vol. 19, No. 3. – P. 372–383.
28. **Bareiša E., Jusas V., Motiejūnas K., Šeinauskas R.** Properties of Variable n-Detection Functional Delay Fault Tests // Information Technology and Control. – Kaunas: Technologija, 2008. – Vol. 37, No. 2. – P. 95–100.
29. **Pomeranz I. and Reddy S. M.** On the Saturation of n-Detection Test Generation by Different Definitions with Increased n // IEEE Transactions on Computer Aided Design of Integrated Cicuits and Systems. – 2008. – Vol. 27, No. 5. – P. 946–957.
30. **Bareiša E., Jusas V., Motiejūnas K., Šeinauskas R.** Development of functional delay tests // Proceedings of the 11th EUROMICRO Conference on Digital System Design: Architectures, Methods and Tools DSD 2008. – 2008. – P. 626–632.

Received 2008 11 10

E. Bareiša, V. Jusas, K. Motiejūnas, R. Šeinauskas. On the Enrichment of Static Functional Test // Electronics and Electrical Engineering. – Kaunas: Technologija, 2009. – No. 3(91). – P. 9–14.

The testing problem is becoming the most crucial part of overall design process that delays the time-to-market of the digital devices. In order to alleviate the test generation complexity and to reduce the time-to-market, one needs to begin the test design at higher levels of abstraction. In this paper a new approach for static functional test enrichment is described. The enriched test pattern set may detect some pin pair faults that are not detectable on the initial test pattern set or, at least, increases the number of detections of pin pair faults. The test enrichment procedure does not increase the test size and it is fast because the procedure does not require test generation. The described approach enriches the test patterns using pin pair fault simulation. The performed experiments demonstrated effectiveness of the proposed approach. We showed that our test enrichment procedure can be incorporated into test generation system and then used as dynamic test enrichment procedure or after slight modification; it can be applied for relaxation of test pattern sets. Ill. 1, bibl. 30 (in English; summaries in English, Russian and Lithuanian).

Э. Барейша, В. Юсас, К. Мотеюнас, Р. Шейнаускас. Трансформирование функциональных тестов с целью уменьшения их объема // Электроника и электротехника. – Каунас: Технология, 2009. – № 3(91). – С. 9–14.

Проблема тестирования является критической проблемой всего процесса проектирования цифровых устройств, которая увеличивает время попадания устройств на рынок. С целью уменшения сложности задачи генерирования тестов надо начать их проектирование на функциональном уровне. Предложен новый метод повышения качества статического функционального теста. Предложенный метод не увеличивает длину исходного теста и является быстрым, так как не требует генерации дополнительных тестов. Выполненные эксперименты показали эффективность предложенного подхода. В статье также показано, что предложенную процедуру можно использовать для релаксации тестовых наборов или применить как динамическую процедуру в процессе генерации функциональных тестов. Ил. 1, библ. 30 (на английском языке; рефераты на английском, русском и литовском яз.)

E. Bareiša, V. Jusas, K. Motiejūnas, R. Šeinauskas. Funkcinių testų transformavimas siekiant sumažinti jų ilgį // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2009. – Nr. 3(91). – P. 9–14.

Testavimas yra kritinė yra kritinė viso skaitmeninių įrenginių projektavimo proceso problema, nes dėl to įrenginiai vėliau patenka į rinką. Norint supaprastinti testų generavimo uždavinį, projektuoti testus reikia pradėti jau funkciniame lygmenyje. Straipsnyje pasiūlytas naujas statinio funkcinio testo kokybės pagerinimo būdas. Jis nepadidina pradinio testo ilgio, be to, yra spartus, nes remiasi gedimų modeliavimu, o ne testinių rinkinių generavimu. Atlikti eksperimentai patvirtino siūlomo metodo efektyvumą. Darbe taip pat parodyta, kad sudarytą testų kokybės pagerinimo procedūrą galima pritaikyti ir testinių rinkinių relaksacijai arba naudoti kaip dinaminę procedūrą funkcinių testų generavimo procese. Il. 1, bibl. 30 (anglų kalba; santraukos anglų, rusų ir lietuvių k.).

DOI: 10.5755/j02.eie.10270