ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL, 2007, Vol. 36, No. 2

IMPACT OF FUNCTIONAL DELAY TEST COMPACTION ON TRANSITION FAULT COVERAGE

Vacius Jusas, Kęstutis Motiejūnas

Software Engineering Department, Kaunas University of Technology
Studentų St. 50-406, LT-51368 Kaunas, Lithuania

Abstract. Compact test sets are very important for degrading the cost of testing the very large-scale integrated circuits by reducing the test application time. Small test sets also lessen the test storage requirements. The only way to compact functional delay fault tests is to enable multi-input transitions in test pattern pairs. The goal of the paper is to analyze the impact of multi-input transitions in test pattern pairs on the transition fault coverage. The performed research shows that the quality of functional delay fault MIT tests in regard of detection of structural level transition faults is higher than that of functional delay fault SIT tests. However, the application of MIT tests instead of SIT tests for transition fault detection may as well decrease the transition fault coverage. The reasons are in changed conditions of signal propagation from circuit input to circuit output. The possible ways to increase the quality of MIT tests are proposed in this work as well.

Keywords: functional delay faults, test compaction, SIT and MIT tests, transition faults.

1. Introduction

Rapid advances of semiconductor technology lead to higher circuit integration as well as higher operating frequencies. Conventional fault models like the standard single stuck-at model were developed for gate-level logic circuits. Regardless of stuck-at fault model's efficiency for several decades, alternative models need to account for deep sub-micron manufacturing process variations [1]. Increasing performance requirements of circuits make it difficult to design them with large timing margins. Thus imprecise delay modeling, statistical variations of the parameters during the manufacturing process as well as physical defects in integrated circuits can sometimes degrade circuit performance without altering its logic functionality. These faults are called delay faults. However, delay fault testing of deep submicron designs is a complex task [2, 3]. It requires application of two-vector patterns at the circuit's intended operating speed.

Two general types of delay fault models, the gate delay fault model [2] and the path delay fault model [3], have been used for modeling delay defects. Although the path delay fault model is generally considered to be more realistic and effective in modeling physical delay faults, it is often difficult to use in practice due to a huge number of paths in the circuit. Therefore, the gate delay fault model is more feasible for large circuits. The most commonly used gate delay fault model is the transition fault model [2].

Test generation is developed in two directions. The usual trend is when the test is generated for the circuit at the structural level. In this case, the main problem is the test generation time, because it directly influences the time-to-market. The task of test generation is quite complicated, especially for sequential circuits. Therefore the design for testability is applied for such circuits. This helps to reduce the cost of test development. But the scan design allows a synchronous sequential circuit to be brought to states that the circuit cannot reach during functional operation. As a result, it allows the circuit to be tested using test patterns that are not applicable during functional operation. This leads to unnecessary yield loss.

The other important direction of test generation is the functional test development at high level of abstraction. In the initial stages of the design, the structural implementation of the design is not known. Therefore the task of the test generation is more complex, because the test has to be generated for all the possible implementations. But the test development can be accomplished in parallel with other design stages. In this case, the time of test generation is not a critical issue. During design process the software prototype of the circuit is created according to the specification. The software prototype simulates the functions of the circuit, i.e. enables to calculate the output values according to the input values. The functional test can be generated on the base of the software prototype. The test patterns generated in such a way can be used for the verification purposes as well. If the generation of

the functional test encounters some difficulties, in order to facilitate the task of test generation the state variables of the software prototype can be used as the primary inputs and the primary outputs. In such a case, the generated test can be applied only for the scan designed circuit, but the correspondence between the state variables and the flip-flops of the scan register has to be established.

The size of a functional test is usually much larger than that of an implementation-dependent one to assure good fault coverage for many implementations. When the synthesis of a high level description into a particular implementation is completed, the minimization of the functional test according to the particular implementation can be provided in order to exclude the test patterns that do not detect the faults of the particular implementation. Next, the list of undetected faults can be formed, and the deterministic methods can be used to detect the faults from this list. The adaptation of the functional test according to the particular implementation is much simpler than a generation of the test from the scratch. The process of adaptation doesn't require the long hours and it has a weak impact on the overall time of the design. That is a strong advantage of the functional test. If the high level description is resynthesized, the functional test remains the same. It has to be only adapted to the new implementation.

Compact test sets are very important for degrading the cost of testing the very large-scale integrated (VLSI) circuits by reducing the test application time. This is especially important for the scan-based circuits as the test application time for these circuits is directly proportional to product of the test set size and the number of storage elements used in the scan chain. Small test sets also reduce the test storage requirements [4, 5].

The only way to compact functional delay fault tests is to enable multi-input transitions in test pattern pairs. The goal of this paper is to analyze the impact of multi-input transitions in test pattern pairs on the transition fault coverage and, on the basis of performed analysis, to propose the ways to increase the quality of compacted test sets. We consider combinational and fully enhanced-scanned sequential circuits. At the high level of abstraction every bit of the state variable is regarded as extra input and output. The test patterns at the functional level are generated for the combinational part of the virtual scan register. When synthesis of the circuit is completed, the bits of the state variables are linked to the flip-flops of the particular implementation.

The rest of the paper is organized as follows. Section 2 presents the related work. The comparison of single-input and multi-input transition tests is presented in Section 3. The case study is given in Section 4. Finally, Section 5 presents the conclusions.

2. Related work

Various functional fault models are proposed in [6-12]. Under functional delay fault models employed in [6-8], a fault is a tuple (*I, O, t I, t O*), where *I* is an input of the circuit under test (CUT), *O* is a CUT output, *t I* is a rising or falling transition at *I* , and *t O* is a rising or falling transition at *O*. Thus, four functional delay faults are associated with every input/output (I/O) pair and the total number of faults is $4\times n\times m$, where n is the number of inputs of the CUT and m is the number of outputs of the CUT. Under the model introduced in Underwood et al. [6], only one pair of test patterns must be generated per fault. This model was expanded in Pomeranz and Reddy [8] by considering Δ different test patterns per fault. Δ is a positive integer, usually in the low hundreds, and is given as an input parameter for each CUT. Pomeranz and Reddy [7] proposed to generate all possible patterns for each fault. This model guarantees detection of all robustly testable path delay faults in any gate-level implementation. However, the sizes of the resulting test set as well as the test generation times are very large and make this model impractical, especially for large circuits [7, 8].

A cell fault [9] implicitly models all defects that alter a CUT's specification and so provides a high degree of realization independence. The main drawback of the cell fault model is that it can only be applied to very small circuits, because this model requires an exhaustive test set comprising all possible input vectors. The suggested in paper [10] fault model is called a coupling fault model. This model is devoted to testing stuck-at faults and is applicable to test path delay faults. The corresponding coupling delay tests detect all robust path delay faults in any realization of the function. The size of a coupling delay test set is very large compared to that of a typical path delay test set, however. The paper [11] presents the criteria of the quality assessment of the functional test consisting of the pairs of patterns with multiple signal transition. The introduced criteria use the new functional delay fault model of the programming prototype. The fault model is based on the direct and indirect impact of the inputs to the outputs.

Another fault model for functional ATPG based on input-output stuck-at faults testing, called pin pair (PP) fault model, is presented in [12]. The pin pair 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. There are $2\times n+2\times m$ possible pin faults. Input-output pin stuck-at fault pairs (x_i^t, z_j^k), t=0, 1, k=0, 1 are called pin pair faults (PP). The number of possible pin pair faults of the circuit is at most $4\times n\times m$. Note that, in general, it is not possible to relate the PP fault with the defects of the module unambiguously, because the PP fault doesn't fix exactly the signal propagation path in the circuit. The PP fault test transformation into functional

delay fault test rule was presented in [13]. We recall this rule.

Rule. If the input pattern q detects the PP fault (x_i^t, z_j^k), t=0, 1, k=0, 1, then the pair of input patterns $<p, q>$, where the signal value of input x_i in the pattern q is $\bar{t}$ and the signal value of input x_i in the pattern p is t, detects the functional delay faults: (x_i, z_j, $r\,x_i$, $r\,z_j$), when t=0, k=0; (x_i, z_j, $r\,x_i$, $f\,z_j$), when t=0, k=1; (x_i, z_j, $f\,x_i$, $r\,z_j$), when t=1, k=0; (x_i, z_j, $f\,x_i$, $f\,z_j$), when t=1, k=1.

Note that every of PP tests according to Rule composed test pattern pair is single-input transition test (SIT) [14] and, therefore, every test pair propagates the transition from a primary input to a primary output in a function robust manner [14]. The test pattern pairs constructed according to Rule have the change of signal value only on one input. Suppose we have an input pattern w that detects q PP faults. Thus for detection of the corresponding q functional delay faults, maximum $k<=q$ pairs of input patterns are built on the base of this pattern (signal transition on one input can cause signal transitions on s outputs, consequently, only one pair of input patterns is needed for detection of s functional delay faults). Therefore, the number of obtained test pattern pairs using Rule is equal to the number of inputs associated with PP fault detection on considered test pattern.

There is another way described in [15] to obtain functional delay fault tests from PP tests. By applying the approach from [15], every input pattern that detects PP faults is transformed only into one input pattern pair in such a way that the signal value transition occurs on every input that is associated with PP fault detection on the considered test pattern. Consequently, if the test for PP faults consists of p input patterns, the constructed functional delay fault test has p input pattern pairs too. Thus the obtained test is much shorter than by applying Rule. However, the test pattern pairs constructed by applying the approach from [15] possess the change of signal value on more than one input. Therefore, these pattern pairs are multi-input transition (MIT) tests [14] and some of functional delay faults that are functional robustly detectable on SIT test may become functional non-robustly detectable [14] or even worse not detectable on considered test pattern pair, because some activation conditions needed for signal transition propagation from particular input to particular output may be corrupted.

A compromise between these two approaches is proposed in [16]. In the paper [16], the necessary conditions for the transition propagation from the considered input to considered output in the multi-input transition test pattern pair are defined. It is proved that the test pattern pairs composed according these conditions guarantee function-robustly propagation of signal value transition on the input to the output. On basis of presented lemma and its corollaries there is developed a procedure that enables pin pair test transformation into compact test without loss of functional delay fault coverage.

3. Comparison of single-input and multi-input transition tests

In this section we are going to analyze the impact of multi-input transitions in functional delay fault test pattern pairs on the transition fault coverage, i.e. on the fault coverage of structural level faults. We will compare the single-input transition functional delay fault test sets composed according to the approach presented in [13] (SIT test) and multi-input transition functional delay fault test sets built according to the approach presented in [16] (MIT test).

The non-redundant ISCAS'85 benchmark circuits have been selected for experiments. The test sets for PP faults were generated for the black-box model of the circuit [12] using a random search procedure. Recall, the black-box model represents a system by defining the behavior of its outputs according to the values applied to its inputs without the knowledge of its internal organization. The black box models were written in the C programming language.

The test sizes and transition fault coverages of SIT and MIT tests are given in Table 1 and Figure 1. The considered tests for each circuit were obtained from the same pin pair tests. Note that all tests presented in Table 1 have 100% coverage of targeted faults, i.e. functional delay faults, and that one test is generated for each targeted fault in the circuit. The best transition fault coverages are shown in bold.

If we examine the results presented in Table 1 and Figure 1, we can see that MIT tests are on the average 1.83 times shorter than SIT tests. Despite this fact, they expose on the average 0.92% better transition fault coverage. Therefore we can claim that the quality of functional delay fault MIT tests in regard of detection of structural level transition faults is higher than that of functional delay fault SIT tests. However the results are inconsistent. The comparison of transition fault coverages shows that in five of ten cases better results expose SIT tests and as many MIT tests. The MIT test approach produces the worst results for the circuit c1355 where the transition fault coverage is 2.12% lower than of SIT test. On the other hand, for the circuit c3540 the transition fault coverage of MIT test is even 11.13% higher than of SIT test.

Remind that all considered tests have 100% coverage of targeted functional delay faults. Thus, naturally, there raises a question what are the sources of such discrepancy of presented experimental results. We are going to research this problem in the next section in greater detail.

Table 1. Test sizes and transition fault coverages

Circuit	SIT test [13]		MIT test [16]	
	Test size	Transition fault coverage %	Test size	Transition fault coverage %
c432	348	**95.56**	244	94.69
c499	5180	**94.40**	1159	93.00
c880	1001	98.91	743	**100.00**
c1355	5162	**97.13**	1068	95.01
c1908	2359	**95.24**	1814	94.58
c2670	1820	96.51	1153	**98.12**
c3540	1457	83.08	1166	**94.21**
c5315	4950	98.41	3382	**99.91**
c6288	1065	99.75	903	**99.89**
c7552	5801	**99.21**	4331	98.03
Average	2914	95.82	1596	**96.74**

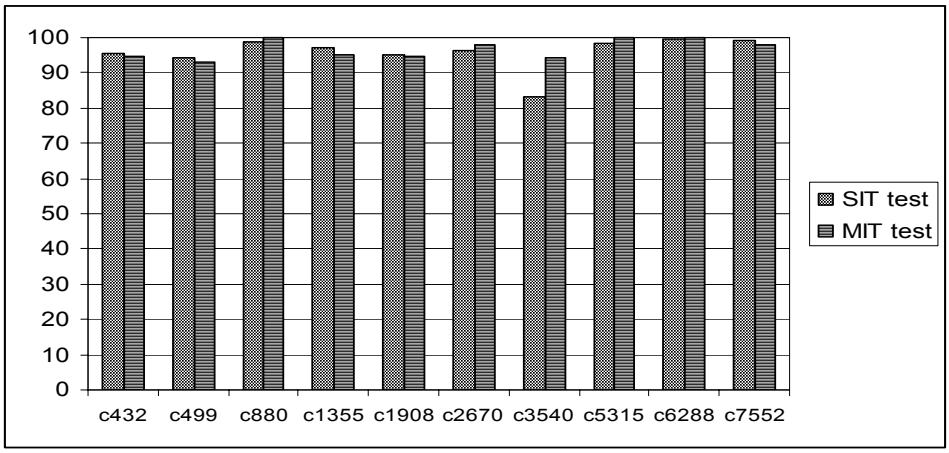

Figure 1. Transition fault coverage comparison

4. Case study

The performed experiments show that the application of MIT tests instead of SIT tests for transition fault detection may as well as increase and decrease the transition fault coverage. The reason is in changed conditions of signal propagation from circuit input to circuit output. In some cases the MIT test sensitizes additional branches in the circuit and hereby detects additional transition faults. In other cases some circuit branches sensitized by SIT test disappear and therefore the number of detected transition faults drops.

Case 1. Sensitization of additional branches. Let's consider the circuit presented in Figure 2. The input pattern ($x1\ x2\ x3$)=<100> detects the following PP faults: ($x1^0$, $y1^1$), ($x2^1$, $y2^0$) and ($x3^1$, $y3^0$). The transformation of this input pattern according to approach [13] results into the SIT test ($x1\ x2\ x3$)=<(000,100); (110, 100); (101,100)> which detects the following transition faults: $r\,x1$, $f\,y1$, $f\,b$, $r\,y2$, $f\,c$ and $r\,y3$. The MIT test constructed according to approach [16] is ($x1\ x2\ x3$)=<011, 100>. This input pattern pair sensitizes extra branches $x2$, $x3$ and a of the example circuit and detects prior listed transition faults as well the faults $f\,x2$, $f\,x3$ and $r\,a$ additionally.

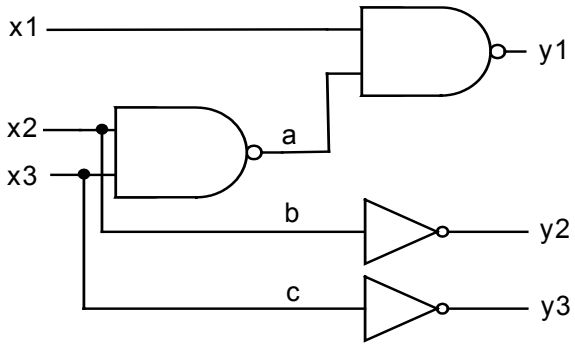

Figure 2. Example circuit for Case 1

Therefore the application of MIT test instead of SIT test may result into sensitization of circuit's branches that the SIT test doesn't sensitize. Then the outcome is a higher coverage of transition faults.

Case 2. Disappearance of signal transition on gate output (single-path). Now let's analyze the

circuit presented in Figure 3. The input pattern ($x1\ x2$)=<01> detects PP faults ($x1^1$, $y1^1$) and ($x2^0$, $y2^1$). The SIT test obtained of this input pattern according to approach [13] is ($x1\ x2$)=<(11,01); (00, 01)>. This test detects transition faults $f\,x1$, $r\,a$, $f\,y1$, $r\,x2$ and $f\,y2$. However the employment of MIT test ($x1\ x2$)=<10, 01> leads to disappearance of signal transition on the output of the NAND gate and, as subsequence, the signal transition disappears on all lines of the single-path that passes from this gate to the circuit output (on lines a and $y1$). Thus the MIT test doesn't detect transition faults $r\,a$ and $f\,y1$ which are detectable by using SIT test. Additionally, the transition fault $f\,x1$ robustly detectable by SIT test is tested as nonrobust.

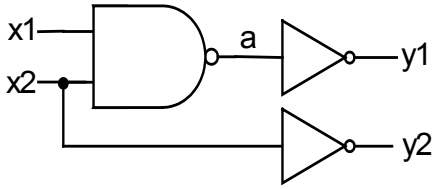

Figure 3. Example circuit for Case 2

Case 3. Disappearance of signal transition on gate output (multi-path). The next example is presented in Figure 4. The input pattern ($x1\ x2\ x3$)= <101> detects the following PP faults: ($x2^1$, $y1^1$) and ($x3^0$, $y3^1$). On the basis of this input pattern, we get SIT test ($x1\ x2\ x3$)=<(111,101); (100, 101)> that detects transition faults $f\,x2$, $r\,a$, $r\,b$, $f\,y1$, $r\,c$ and $f\,y2$. The MIT test constructed according to approach [16] is ($x1\ x2\ x3$)=<110, 101>. This input pattern pair determines stable signal value for line b on both input patterns. Consequently, the transition fault $r\,b$ detectable by SIT test becomes undetectable.

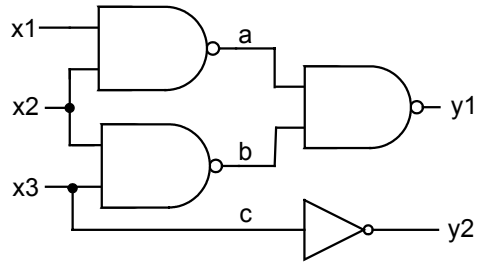

Figure 4. Example circuit for Case 3

The analysis of cases 2 and 3 demonstrates that the application of MIT test instead of SIT test may result into disappearance of sensitization of some circuit's branches that the SIT test sensitizes. Then the outcome is a lower coverage of transition faults.

Case 4. Change of detection type. Let's consider the circuit presented in Figure 5. The input pattern ($x1\ x2\ x3$)=<010> detects PP faults ($x11$, $y10$), ($x2^0$, $y2^1$) and ($x3^1$, $y3^0$). The SIT test obtained of this input pattern according to approach [13] is ($x1\ x2\ x3$)= <(110,010); (000, 010); (011,010)>. This test robustly detects transition faults $f\,x1$, $r\,y1$, $r\,x2$, $f\,y2$, $f\,x3$ and $r\,y3$. However the employment of MIT test ($x1\ x2\ x3$) =<101, 010> may lead to signal hazard (1→0→1) on line a that contradicts the conditions of robust path sensitization. Thus the transition faults $f\,x3$ and $r\,y3$ robustly detectable by SIT test become nonrobustly detectable using MIT test.

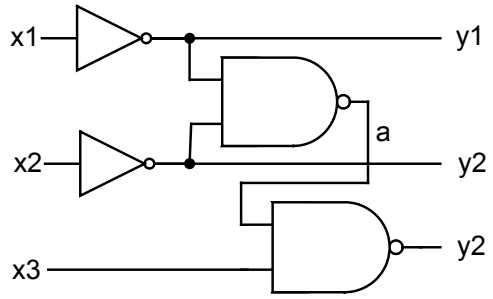

Figure 5. Example circuit for Case 3

Of course, the main attention in developing of functional delay fault test compaction methods has to be paid to cases 2 and 3. The main problem is that at functional circuit description level we don't know the circuit structure and, therefore, we can analyze the circuit behavior only on the circuit boundary. We have solved the question of disappearance of signal transition on gate output by single-path sensitization (case 2) and the solution is already implemented in a procedure whose pseudocode is presented in [16]. The main idea is in comparison of circuit responses of SIT and MIT tests. If the circuit responses don't coincide, the circuit inputs (input) that cause this discrepancy are identified and the signal transitions on these inputs are disabled. The problem of disappearance of signal transition on gate output by multi-path sensitization (case 3) is more complex and will be investigated in the future. One possible solution is in identification of circuit inputs that have influence to separate circuit outputs and then in allowing of multiple signal transitions only on these inputs.

5. Conclusions

In this paper we have explored the properties of MIT functional delay fault tests. In [17] it is stated that SIT test sequences are more effective than MIT sequences to obtain high robust delay fault coverage. That is probably true for path delay faults, however this statement misfits for transition fault detection using functional delay tests. Our experiments show that MIT tests are on average 1.83 times shorter than SIT tests. Despite this fact, they expose on the average 0.92% better transition fault coverage. Therefore we can claim that the quality of functional delay fault MIT tests in regard of detection of structural level transition faults is higher than that of functional delay fault SIT tests. However the results are not unambiguous. The performed analysis shows that the application of MIT tests instead of SIT tests for transition

fault detection may as well as increase and decrease the transition fault coverage. The reasons are in changed conditions of signal propagation from circuit input to circuit output. The possible ways to increase the quality of MIT tests are proposed in this work as well.

References

[1] **S. Ohtake, K. Ohtani, H. Fujiwara.** A Method of Test Generation for Path Delay Faults Using Stuck-at Fault Test Generation Algorithms. *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition* (DATE'03), 2003, 310-315.

[2] **J.A. Waicukauski, E. Lindbloom, B.K. Rosen, V.S. Iyengar.** Transition fault simulation. *IEEE Design and Test, April* 1987, 32-38.

[3] **C.J. Lin, S.M. Reddy.** On Delay Fault Testing in Logic Circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol*.6, *September* 1987, 694-703.

[4] **I. Hamzaoglu, J.H. Patel.** Test Set Compaction Algorithms for Combinational Circuits. *Proceedings of the* 1998 *IEEE/ACM international conference on Computer-aided design*, 1998, 283-289.

[5] **S. Ravi, G. Lakshminarayana, N.K. Jha.** High-Level Test Compaction Techniques. *IEEE Transactions on Computer Aided Design of Integrated Circuit and Systems. Vol*.21, *No*.7. 2002, 827–841.

[6] **B. Underwood, W.O. Law, S. Kang, H. Konuk.** Fastpath: A path-delay test generator for standard scan designs. *Proceedings of* 1994 *International Test Conference*, 1994, 154–163.

[7] **I. Pomeranz, S.M. Reddy.** On testing delay faults in macro-based combinational circuits. *Proceedings of* 1994 *International Conference on Computer-Aided-Design*, 1994), 332–339.

[8] **I. Pomeranz, S.M. Reddy.** Functional test generation for delay faults in combinational circuits. *Proceedings of* 1995 *International Conference on Computer-Aided-Design*, 1995, 687–694.

[9] **M. Psarakis, D. Gizopoulos, A. Paschalis.** Test Generation and Fault simulation for Cell Fault Model Using Stuck-at Fault Model Based Test Tools. *Journal of Electronic Testing*, *Vol*.13, 1998, 315-319.

[10] **Y. Joonhwan, J. Hayes.** The Coupling Model for Function and Delay Faults. *Journal of Electronic Testing*, *Vol*. 21. *No*.6, 2005, 631-649.

[11] **E.Bareiša, V.Jusas, K.Motiejūnas, R.Šeinauskas.** Functional Delay Test Quality Assessment at High Level of Abstraction. *Information Technology And Control, Kaunas, Technologija*, *Vol*.36, *No*.1, 2007, 7-15.

[12] **E. Bareiša. V. Jusas. K. Motiejūnas. R. Šeinauskas.** The Realization-Independent Testing Based on the Black Box Fault Models. *Informatica, Vol*.16, *No*.1, 2005, 19-36.

[13] **E.Bareiša, V.Jusas, K.Motiejūnas, R.Šeinauskas.** Application of Functional Delay Tests for Testing of Transition Faults and Vice Versa. *Information Technology And Control, Kaunas, Technologija, Vol*.34, *No*.2, 2005, 95 - 101.

[14] **M. Michael, S. Tragoudas.** ATPG Tools for Delay Faults at the Functional Level. *ACM Transactions on Design Automation of Electronic Systems. Vol*.7, *No*. 1, *January* 2002, 33–57.

[15] **E.Bareiša, V.Jusas, K.Motiejūnas, R.Šeinauskas.** Functional Delay Test Construction Approaches. *Electronics and electrical engineering, Kaunas, Technologija, No*. 2(74). 2007, 49 - 54.

[16] **E.Bareiša, V.Jusas, K.Motiejūnas, R.Šeinauskas.** Functional Test Transformation to improve Compaction. *Electronics and electrical engineering, Kaunas, Technologija*, 2007.

[17] **A. Virazel, R. David, P. Girard, C. Landrault, S. Pravoussoudovitch.** Delay Fault Testing: effectiveness of Random SIC and Random MIC Test Sequence. *Journal of Electronic Testing Theory and Applications, Vol*.17, *No*. 3/4, 2001, 233-241.

Received May 2007.