HYBRID CONSTRAINT-BASED TEST PATTERN GENERATION

AYUB CHIN ABDULLAH

Universiti Teknologi Malaysia

HYBRID CONSTRAINT-BASED TEST PATTERN GENERATION

AYUB CHIN ABDULLAH

A thesis submitted in fulfilment of the

requirements for the award of the degree of

Master of Engineering (Electrical)

Faculty of Electrical Engineering

Universiti Teknologi Malaysia

JUNE 2013

ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my supervisor, Dr. Ooi Chia Yee, for encouragement, guidance, critics and friendship. Without her continued support and interest, this thesis would not have been the same as presented here.

I am also indebted to Universiti Teknologi Malaysia (UTM) for funding my Master study. Librarians at UTM also deserve special thanks for their assistance in supplying the relevant literatures.

My fellow postgraduate students should also be recognised for their support. My sincere appreciation also extends to all my colleagues and others who have provided assistance at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family members.

ABSTRACT

The role of testing in Integrated circuit (IC) is to determine the correctness of manufactured circuits. Therefore, testing is important since the fraction of good chips sold in the market yields the quality of the product. Automatic test equipment (ATE) is equipment that used in manufacturing test. The advance in IC technology make the memory needed for ATE is big. However, this could be lessened by reducing the data volume inserted in the ATE. This could be achieved through test compaction like fault collapsing. In this research, hybrid constraint-based test pattern generation proposed that include fault injection, Automatic test pattern generation (ATPG) and fault simulation. The fault injection is creating the faulty circuits while the ATPG searching the test patterns. For the fault simulation, it calculates the fault coverage of the ATPG system. In the fault injection, the functional fault modeling adopted which applied the fault collapsing. This fault collapsing is aiding in test compaction. Five experiments being done on eight Circuit under tests (CUTs) from the International Torino Conference (ITC) '99 benchmark circuits. The high-level and gate-level fault coverage, and the test length for this CUTs have been obtained from this experiments. The average of compaction ratio that obtained by the proposed method is 5.42%.

ABSTRAK

Peranan pengujian dalam Litar bersepadu (IC) adalah untuk menentukan ketepatan litar perkilangan. Oleh itu, pengujian adalah penting memandangkan pecahan cip baik yang dijual di pasaran mempersembahkan kualiti produk. Peralatan ujian automatik (ATE) merupakan peralatan yang digunakan dalam pengujian pembuatan. Kemajuan dalam teknologi IC membuatkan memori yang diperlukan oleh ATE adalah besar. Walau bagaimanapun, ini boleh dikurangkan dengan mengurangkan jumlah data yang dimasukkan ke dalam ATE. Ini boleh dicapai melalui pengujian pemadatan seperti peruntuhan kesalahan. Dalam kajian ini, cadangan penghasilan corak ujian hibrid berasaskan kekangan yang merangkumi suntikan kesalahan, penghasilan corak ujian automatik (ATPG) dan simulasi kerosakan. Suntikan kesalahan mewujudkan litar rosak manakala ATPG mencari corak ujian. Bagi simulasi kesalahan, ia mengira liputan kerosakan sistem ATPG. Dalam suntikan kesalahan, pemodelan kesalahan berfungsi dipakai yang mengaplikasikan peruntuhan kesalahan. Peruntuhan kesalahan ini membantu dalam pengujian pemadatan. Lima eksperimen dilakukan pada lapan litar dalam pengujian (CUT) dari lita penanda aras daripada Persidangan Antarabangsa Torino (ITC) '99. Liputan kesalahan tahap-tinggi dan tahap-get, dan panjang pengujian untuk CUT-CUT ini telah diperolehi daripada eksperimen-eksperimen ini. Purata nisbah pemadatan yang diperolehi oleh kaedah yang dicadangkan adalah 5.42%.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ATPG – Automatic Test Pattern Generation

CLP - Constraint Logic Programming

CUT - Circuit Under Test

HTT - Hard-To-Test

IC – Integrated Circuit

ITC - International Test Conference

RTL - Register Transfer Level

VHDL - Very-High-Speed-Integrated-Circuit Hardware Description Language

VLSI – Very Large Scale Integration

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

This chapter gives the introduction and problem statements of this research. This is followed by research objectives, scope, and contribution. After that, research methodology and thesis organization are described.

1.1 Introduction

The role of testing in integrated circuit (IC) is to determine the correctness of manufactured circuits. Therefore, testing is important since the fraction of good chips sold in the market yields the quality of the product. One of the aspects that determine product quality is test quality, which is quality of test vectors developed in test development process. In very large scale integration (VLSI) realization process, test development is done after the IC design and verification process, as can be seen in Figure 1.1. At gate level, test vector generation process has become more time consuming due to increasing complexity of VLSI circuits. According to the Moore's Law, the number of transistors in IC doubles every 18 months [1]. Nowadays, an IC

can easily have millions of gates [2]. In order to cope with this situation, the top-down design methodology starting with high-level description has been introduced.

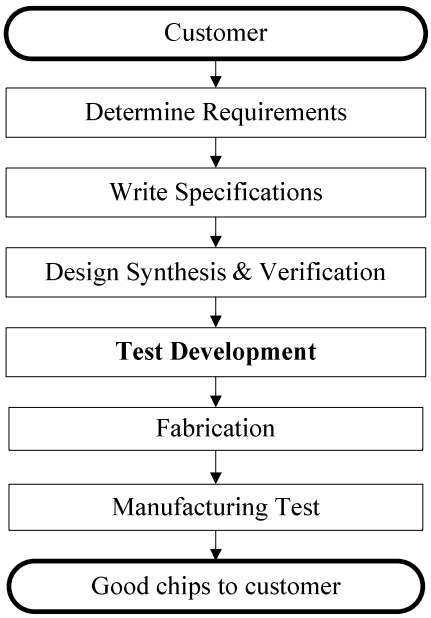

Figure 1.1 Testing in VLSI Realization Process

Test generation algorithms designed for the high-level models are usually direct extensions of those for the gate-level models, in which the functional modules are treated as primitive components so fewer components are evaluated during test generation. Since digital design is first described at high-level, the test generation could be done earlier even before design synthesis into gate-level circuit. The resulting reduced structural complexity makes high-level test generation attractive. Thus, functional test generation and functional fault simulation of a digital system described in high-level models offers an attractive testing alternative.

1.2 Problem Statements

Technology pace is governed by Moore's law. According to Moore's law, the number of transistors in IC is doubled in every 18 months. The factory that produces this IC is driven by cost, quality, productivity, and speed [3]. Figure 1.2 shows the data volume requirements for Automatic Test Equipment (ATE). The test data volume is increasing year after year. As seen in Figure 1.2, the data volume requirements are high for non-hierarchical and non-compressed circuit. The increase in data volume will make the memory size needed for the ATE bigger. At the same time, it makes the test length and computation time longer, which will impact time-to-market. Test compaction is a method that could reduce the test length. Hence, it is useful in combating the increasing value of the data volume [3].

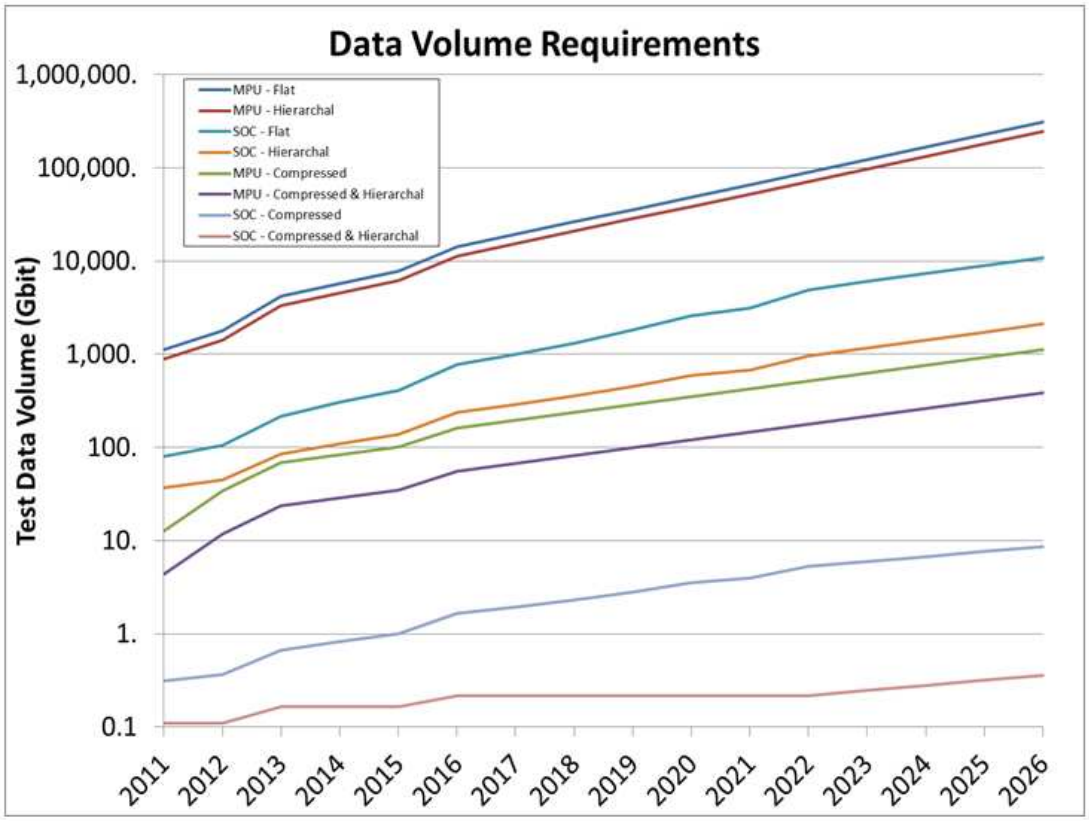

Figure 1.2 Data Volume Requirements [3]

The increasing size of transistors in IC also makes the gate-level test generation more complex. In high-level test generation algorithms, functional modules are treated as primitive components. So, fewer components are evaluated during test generation. It is also in line with top-down design methodology. The reduced structural complexity makes high-level test generation attractive [1].

1.3 Objective

The objective of this research is to develop a hybrid automatic test pattern generation (ATPG) system which, when compared to gate-level ATPG alone, has:-

a) Shorter test length.

Functional fault model is used in the hybrid ATPG system, which is useful in test compression. When generating test patterns using functional fault model, compressed test patterns are generated. Thus, total test length becomes shorter.

b) Comparable fault coverage.

While achieving shorter test length, fault coverage of the hybrid ATPG system is made comparable to the gate-level ATPG by including both functional fault model and gate-level fault model in the proposed ATPG system.

1.4 Scope of Work

In order to achieve the objective of the research, the following research scope has been outlined as in Figure 1.3.

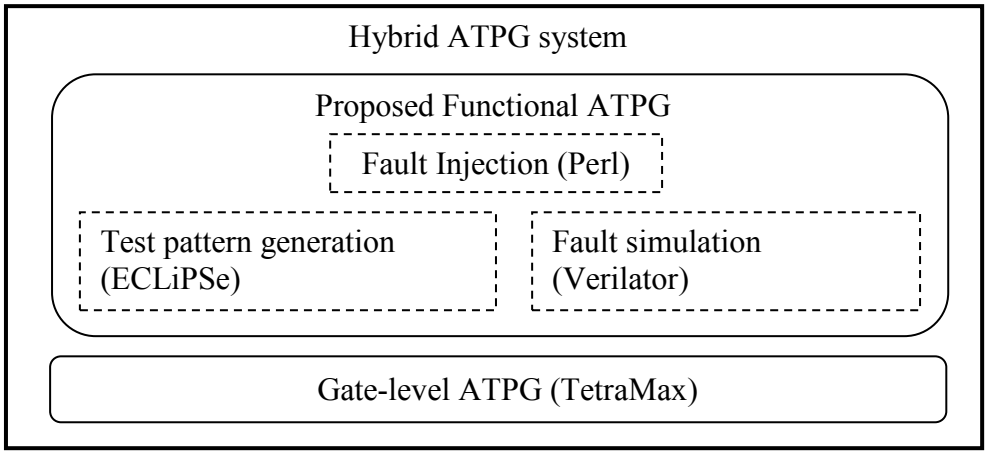

Figure 1.3 Research Scope

The research scope covers:-

a) The proposed hybrid ATPG system which is composed by the newly developed functional ATPG system and existing gate-level ATPG system. Functional test pattern generation tool is developed using a constraint logic programming (CLP) tool called ECLiPSe CLP which searches the test patterns using the increasing order 'search' functions in ECLiPSe [6]. The high-level test generation will employ the functional fault model to generate compressed test patterns.

b) *Fault injection tool* written in Perl language which creates the corresponding faulty circuits from the circuit-under-test (CUT). Faulty CUTs are needed for functional test generation and functional fault simulation. The CUT used in functional ATPG system is written in functional description. Hence, fault models which are introduced at functional level will be used.

c) Functional fault simulation tool which is developed using Verilator to evaluate the fault coverage of the high-level test patterns generated by functional test generation. Verilator is an open source tool which converts a design described in Verilog to the corresponding object oriented C++

description. The development of fault simulation could be done by creating a C++ top-level file that instantiates the C++ objects of both CUT and faulty CUT, injects the test patterns obtained from the functional test generation and analyzes their output responses to detect a difference between CUT's output and faulty CUT's output that represents a fault detection.

d) The translation tool that translates CUT in Verilog file to a description in ECLiPSe CLP language is written in Perl language. The translation is necessary because the proposed functional test generation can only read a design in ECLiPSe CLP.

e) After collecting a set of test patterns from functional ATPG system, the test patterns will be used to perform gate-level fault simulation by TetraMax to obtain its gate-level fault coverage. The remaining undetected faults will be processed for the test patterns using test generation of TetraMax.

f) Comparison between the proposed hybrid ATPG system and gate-level ATPG system that is TetraMax from Synopsys will be done to evaluate the metrics of fault coverage and test length. In order to accomplish the comparison, high-level description of designs need to be synthesized into gate-level circuit. This is done using Synopsys Design Vision.

1.5 Research Contributions

In this research, a hybrid ATPG system that consists of constraint-based functional ATPG and gate-level ATPG has been introduced. A comparison with an industry ATPG tool called Synopsys TetraMax has been done where an average of 5.42% compaction ratio has been achieved. This contributes to shorter test length than that of the gate-level ATPG. Consequently, test data volume required to be

stored in ATE memory is reduced as well. In today's industry, test development is mostly done at gate-level. The usual flow in the digital circuit design is from high-level towards low-level. This research has shown that top-down methodology can be applied for ATPG as well such that ATPG can be performed earlier. This then leads to the faster time-to-market of product and saves the production cost.

1.6 Research methodology

In order to make this research successful, a proper planning is essential and all working procedures should be identified clearly. This research involves fault injection, translation, test pattern generation, fault simulation, integration of the above- mentioned tools and experiment with Synopsys TetraMax. The research workflow is shown in Figure 1.4.

The work begins with literature review on CLP-based ATPG. Problem formulation and scope identification are done after sufficient knowledge on CLP-based ATPG obtained. Then, the fault injection tool is developed. This tool is created using Perl script so a study on Perl programming is conducted first. After that, the translation tool is developed. This translation tool translates the Verilog file to ECLiPSe CLP.

The main part of this system which is the functional ATPG is then developed. This ATPG finds the test pattern for corresponding functional fault. In order to verify the test pattern, functional fault simulation is invoked. So, a fault simulation is developed. The fault simulation is executed using Perl script which first creates the top level C++ file for Verilator.

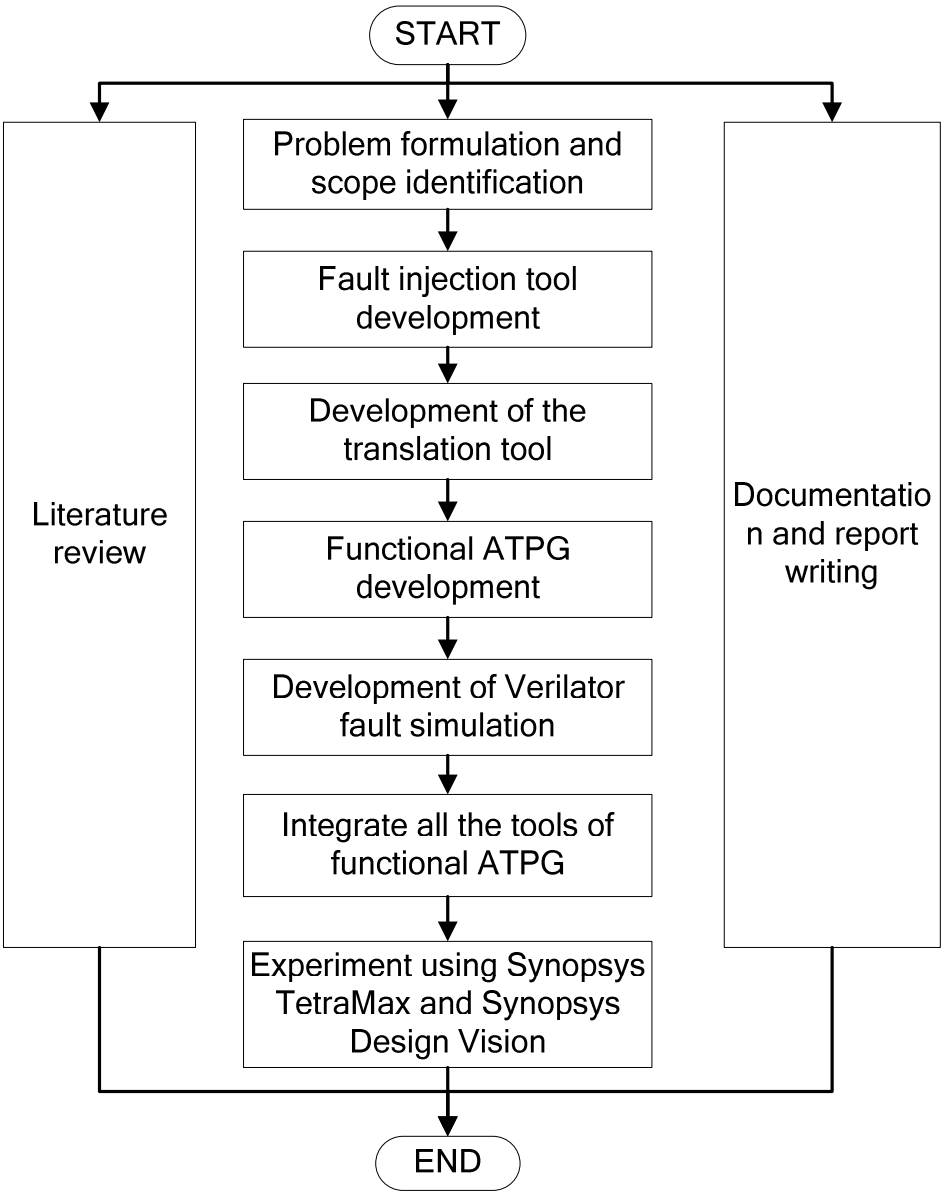

Figure 1.4 Research methodology

After that, another Perl script is developed to integrate all the tools of functional ATPG system which are the fault injection tool, translation tool, functional test generation and Verilator-based fault simulation tool. This integration script also plays the role in saving the report of this functional ATPG system which includes the fault coverage, test length and the Verilog testbench. Then, Synopsys

Design Vision and TetraMax are included as part of the proposed hybrid ATPG system.

To show the advantages of the proposed hybrid ATPG system, a comparison between the gate-level ATPG and the hybrid ATPG is done using the International Test Conference (ITC) '99 benchmark circuits. The existing gate-level ATPG used in the comparison is Synopsys TetraMax.

1.7 Organization of the Thesis

The thesis is organized into five chapters. The first chapter presents the motivation and research objectives. This is followed by research scope and research contribution. After that, research methodology and thesis organization are described.

Chapter 2 provides literature review prior to engaging the mentioned scope of work. Several topics related to this research are reviewed to give an overall picture of the background knowledge involved. Summary of the literature review is given to clarify the research rationale.

Chapter 3 presents functional ATPG. It starts with the explanation of the functional fault model, and the Perl programming used to develop the fault injection. Then, it presents the functional test pattern generation. After that, functional fault simulation is elaborated.

Chapter 4 presents the experimental result and analysis. It starts with the experiment procedure that shows the effectiveness of the proposed hybrid ATPG system in terms of test length and fault coverage.

In the final chapter of the thesis, the research work is summarized and deliverables of the research are stated. Suggestion for potential extensions and improvements to the design is also given.

REFERENCES

1) Michael L. B. and Vishwani D. A. (2002). *Electronic Testing for Digital, Memory and Mixed-signal VLSI Circuits*. New York: Kluwer Academic Publishers.

2) Chien-In Henry Chen (2003). Behavioral test generation/fault simulation. *Potentials, IEEE*. 22(1), 27 – 32. IEEE.

3) Roger B. (2011). Test & Test Equipment. *ITRS Winter Public Conference Presentations*. 14 December. Songdo ConvensiA, Incheon, Korea, 1- 62.

4) Fujiwara, H. and Shimono, T. (1983). On the Acceleration of Test Generation Algorithms. *Computers, IEEE Transactions on*. 32(12), 1137 – 1144. IEEE.

5) Corno, F., G. Cumani, et al. (2001). Effective techniques for high-level ATPG. *Proceedings of the Asian Test Symposium*. 19 – 21 November. Kyoto, 225 – 230.

6) Corno, F., Cumani, G., Sonza Reorda, M. and Squillero, G. (2000). An RT-level fault model with high gate level correlation. *High-Level Design Validation and Test Workshop, 2000. Proceedings. IEEE International*. 8 – 10 November. Berkeley, CA, 3 – 8.

7) Corno, F., M. S. Reorda, et al. (2000). RT-level ITC '99 benchmarks and first ATPG results. *Design & Test of Computers, IEEE*. 17(3), 44-53. IEEE.

8) Corno, F., M. S. Reorda, et al. (2000). High-level observability for effective high-level ATPG. *VLSI Test Symposium, 2000. Proceedings. 18th IEEE*. 30 April – 4 May. Montreal, Que, 411 – 416.

9) Sinanoglu, O. and Orailoglu, A. (2003). Hierarchical constraint conscious RT-level test generation. *Digital System Design, 2003. Proceedings. Euromicro Symposium on*. 1 - 6 September. Belek-Antalya, Turkey, 312 – 318.

10) Giuseppe Di, G., Cristina, M., Graziano, P. and Franco, F. (2009). Test Generation Based on CLP. Kenichi Takahata. *Micro Electronic and Mechanical Systems*. (pp. 431 - 454). USA: In-Tech.

11) Antoni Niederlinski (2011). *A Quick and Gentle Guide to Constraint Logic Programming via ECLiPSe*. (9th ed.). Poland: Jacek Skalmierski Computer Studio.

12) Giuseppe Di, G., F. Franco, et al. (2009). Test Generation Based on CLP. *8th International Workshop on Microprocessor Test and Verification, Common Challenges and Solutions*. 5 – 6 December. Austin, 98 – 105.

13) Guglielmo, G. D., Fummi, F., Marconcini, C. and Pravadelli, G. (2007). Improving high-level and gate-level testing with FATE: A functional automatic test pattern generator traversing unstabilised extended FSM. *Computers & Digital Techniques, IET*. 1(3), 187 – 196. IEEE.

14) Kwang-Ting C. and Krishnakumar, A.S. (1993). Automatic Functional Test Generation Using The Extended Finite State Machine Model. *Design Automation, 1993. 30th Conference on*. 14 - 18 June. Dallas, TX, USA, 86 – 91.

15) Brand, S. (2001). Sequential Automatic Test Pattern Generation by Constraint Programming. *In: Proc. CP 2001 Workshop on Modelling and Problem Formulation*. 1 December. Paphos, Cyprus, 1 – 8.

16) Hochbaum, D.S. (1996). An optimal test compression procedure for combinational circuits. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*. 15(10), 1294 – 1299. IEEE.

17) Guo, R., S. M. Reddy, et al. (2003). Reverse-order-restoration-based static test compaction for synchronous sequential circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 22(3), 293-304. IEEE.

18) Hsiao, M. S., Rudnick, E. M. and Patel, J. H. (1997). Sequential circuit test generation using dynamic state traversal. *European Design and Test Conference, 1997. ED&TC 97. Proceedings*. 17 - 20 March. Paris, 22 - 28.

19) Zainalabedin Navabi (2011). *Digital System Test and Testable Design Using HDL Models and Architectures*. (1st ed.). New York: Springer.

20) Corno, F., Reorda, M.S. and Squillero, G. (2000). RT-level ITC'99 benchmarks and first ATPG results. *Design & Test of Computers, IEEE*. 17(3), 44 – 53. IEEE.

21) http://rpmfind.net/linux/RPM/epel/beta/6/i386/vhd2vl-2.3-1.el6.i686.html