E-Learning Environment for WEB-Based Study of Testing

R.Ubar, A.Jutman, J.Raik, S.Devadze, M.Jenihhin,
I.Aleksejev, A.Chepurov, A.Chertov, S.Kostin,
E.Orasson
TU Tallinn
Tallinn, ESTONIA
raiub@pld.ttu.ee

H.-D. Wuttke
TU Ilmenau
Ilmenau, GERMANY
dieter.wuttke@tu-ilmenau.de

Abstract— **An environment targeted to e-learning of test issues in microelectronics is presented. The environment consists of a set of Java applets, desktop applications and of a web based access to the HW equipment which can be used in the classroom, for learning at home, in laboratory research and training, or for carrying out testing of students during exams. The tools support university courses on digital electronics, computer hardware, VLSI design and architectures, testing and design for testability to learn by hands-on exercises how to design digital systems, how to make them testable, how to build self-testing systems, how to generate test patterns, how to analyze the quality of tests, and how to localize faults in hardware.**

Keywords - e-learning, defects, fault models, test generation, fault simulation, built-in self test, boundary scan, fault diagnosis.

I. INTRODUCTION

Digital hardware (HW) has exhibited high reliability in the past; however, future nanometer-scale HW devices will become a source of problems. The more complex electronics systems are getting, the more important the problems of test and design for testability are becoming. These topics are often underestimated in educating electronics and system engineers. This is because in the today's university curricula test issues are usually neglected: students learn how to design electronic systems but not how to test them. The next generation of engineers involved with SoC and NoC technologies should be made better aware of the importance of test, and trained much more in test technology to enable them to develop, design and produce high quality and defect-free products.

The main goal of the conducted work was the creation of a homogeneous e-learning environment for studying the test and diagnostics of digital systems. In the paper a conception and means are presented to improve the skills of students educated for HW and SoC design in test related topics. The method presented here deals with the goal to put interactive teaching modules ("living pictures") to the Internet that can be used as tools in a lecture as well as for individual self-studies. The modules can be accessed independent of time and place. On one hand, teachers can demonstrate simulations of different examples and procedures of test related topics using living pictures during the lessons. On the other hand, students can use the same simulation modules on their home computer, if the living pictures are available on the Internet. Finally, the same modules can be used during examination.

In the following we present a set of well linked tools for learning test issues like test generation and fault simulation at different levels of abstraction. Three levels are represented: gate level, macro- or sub-circuit level where macros represent Fanout Free Regions (FFR), and Register Transfer Level (RTL). Dedicated tool sets are developed for learning Built-in Self-Test (BIST) and fault diagnosis problems. For investigating realistic physical defects in microelectronic circuits a novel web based HW/SW environment based on a special education chip ("DefSim") has been developed. It allows carrying out remote experiments via Internet with different realistic defects selected remotely in the DefSim. To learn and investigate interconnect testing issues at the printed circuit board (PCB) level, a multi-functional system, which provides a simulation, learning, research, and CAD environment for IEEE 1149.1 Boundary Scan (BS) standard has been developed.

The presented environment consists of the following tools: (1) logic level applets (LLA), (2) register transfer level applets (RTLA), (3) tool for investigation of physical defects (DefSim), (4) BIST analyzer (BISTA), (5) tools for fault diagnosis (DIAGNOZER), (6) tool for Boundary Scan research (BScan), and (7) tool set for logic level test generation and fault simulation (Turbo Tester - TT). The test related topics covered by the presented tool set are illustrated in Table 1.

Table 1. HW testing topics covered by the e-learning environment

Tools	Fault models	Test generation	Fault simulation	Chip level BIST	Board level test	Fault diagnosis
LLA		+	+	+		+
RTLA		+	+	+		
DefSim	+	+	+			
BISTA			+	+		
DIAGNOZER			+			+
BScan	+				+	
TT		+	+	+		

While the Java applets are platform-independent by their nature, all the desktop applications are available for Windows, Linux and UNIX platforms.

II. DESCRIPTION OF THE ENVIRONMENT

A. *Logic Level Test*

The engine of the concept of teaching logic level test consists of PC-based tools [1] installed locally and Java-applets invoked remotely via Internet [2]. By using the interaction possibilities of the Java-applets the students can have the first acquaintance with how to produce input test stimuli for testing logic level circuits, how to simulate faults, and how to locate the faulty gates or faulty connections (Figure 1). Different learning tasks and exercises are described in Section III, which make use of the applets.

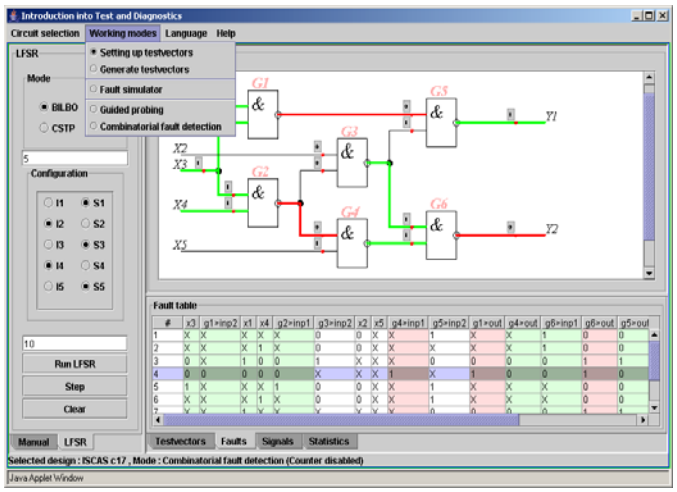

Figure 1. LLA interface

The PC-based tool set, which supports learning test at a more advanced level, is called Turbo Tester (TT). TT includes a rich set of tools for test generation implemented for different algorithms like deterministic, genetic, random, and allows in such a way to compare different approaches and methods of test generation. The circuits can be represented at different levels (gate and FFR-levels) which allow investigating the scalability of different test generation algorithms, and how the efficiency of the algorithms depends on the complexities of circuits.

Figure 2. Login page for WebTT

The set of TT tools provides good possibilities for laboratory training and experimental research. Among other convenient features this toolset has a web access to the tools installed on a server. This interface is called WebTT (Figure 2).

B. *Register Transfer Level Test*

Two types of applets are used in the environment: applets for investigating and learning test problems in simple gate level circuits, and applets for practicing design and test problems in more complex digital systems represented on RTL level, consisting of control and data paths [3] (Figure 3).

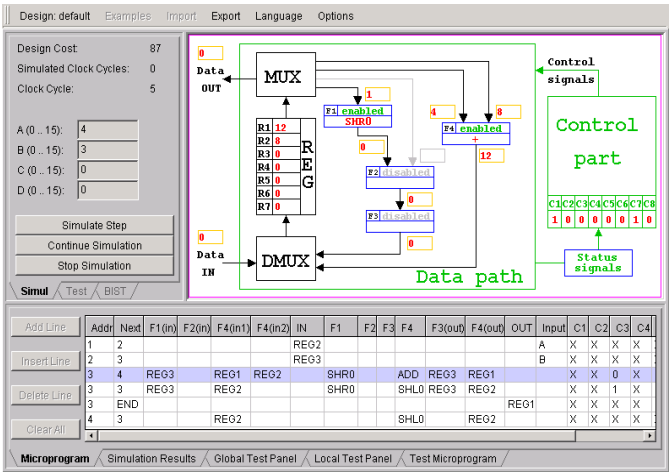

Figure 3. RTLA interface

Students can exercise RTL implementations of processors represented by data flow graphs or micro-programs (like multiplication, division, signal processing algorithms etc.). Such topics as design of data-flows and microprograms of computing algorithms, investigation of tradeoffs between speed and hardware cost in digital design, RTL simulation, design for testability, functional level test generation, built-in self-test (BIST), diagnostic analysis and other related problems are covered by these applets. By gate-level fault simulation it is possible to evaluate the fault coverage of functional tests. The applet fully reflects the "easy action and reaction" concept which was taken as the major target for its creation. Each field in the microprogram, each functional unit in the circuit map, and other modules are clickable. Their functions can be changed or further adjusted. The reaction on each action is instantly reflected by highlighting of selected modules.

C. *Defect Level Test*

The central element of the DefSim environment (Figure 4) is the IC with a large variety of shorts and opens physically inserted into a set of digital standard cells and small circuits [4,5]. The IC is attached to a dedicated measurement box serving as an interface to the computer. It is possible to select any defect of interest by addressing it in the circuit. Then the user can apply an arbitrary input test sequence and measure the circuit's response to it in terms of both the binary logic values and current levels (IDDQ). It is also possible to compare its behavior over the correct copy of the same circuit. From the didactical point of view, the DefSim environment targets (but it is not limited to) two main areas of expertise: defect modeling and defect observability. The students will learn in practice that

some simple defects represent a real challenge especially from the diagnostics (defect localization) point of view. The environment allows carrying out remote experiments via Internet.

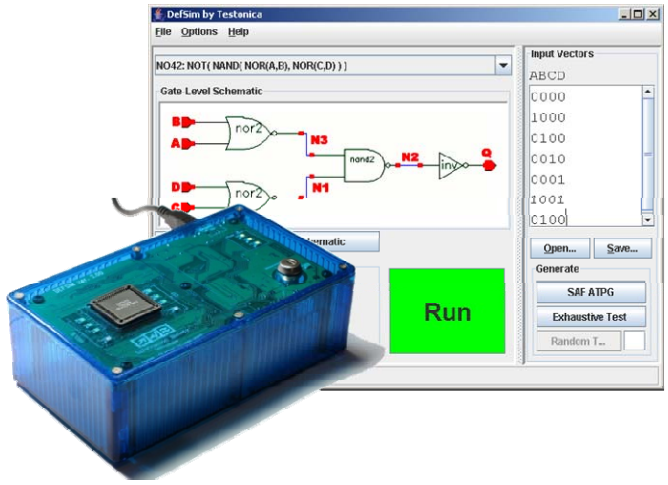

Figure 4. DefSim environment and its IC

D. Boundary Scan

The BS standard is a very important state-of-the-art testing technique of modern complex integrated systems. The main goal of the BScan tool (Figure 5) is to introduce the basic concepts of the standard [6,7].

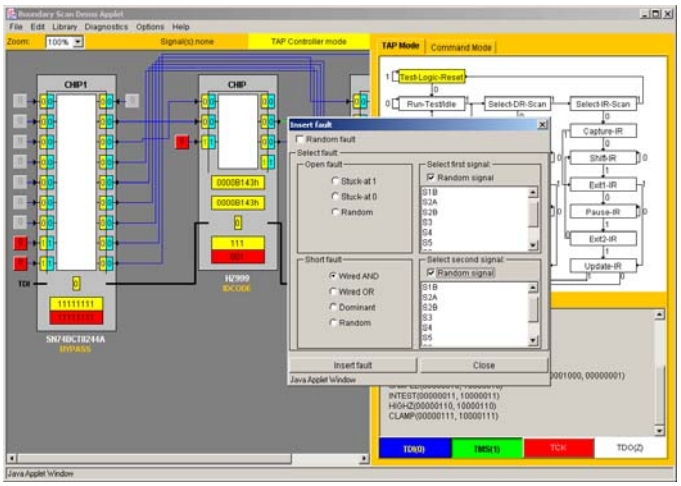

Figure 5. BScan tool

The students should learn the BS instructions and working modes and see from inside how the BS structures are operating. According to the standard, all the chips on the board are connected into a scan chain via TDI (Test Data In) and TDO (Test Data Out) pins. Hence, all the instructions and test data can be inserted via the single TDI input only. Therefore, the task of controlling the system of several BS chips is not a trivial one. First of all, students should study the Test Access Port (TAP) Controller, which is the key device in the whole BS conception. They learn to move from state to state on the state diagram and insert different BS instructions via TDI. The next step is the study of data registers and their proper usage. When the main principles of BS operation are understood, the students face the task of interconnect diagnosis. They should learn how to properly select test vectors in order to find interconnect defects of a given type. The final and the most advanced task is to write a description of an own chip according to given parameters using BSDL format.

E. Built-in Self Test

Linear Feedback Shift Registers (LFSR) and other Pseudo-Random Pattern Generators (PRPG) have become one of the central elements used in test and self test of contemporary complex electronic systems like processors, controllers, and high-performance integrated circuits. We have developed a training and research tool BIST Analyzer (BISTA) (Figure 6) for learning basic and advanced issues related to PRPG-based test pattern generation [8,9].

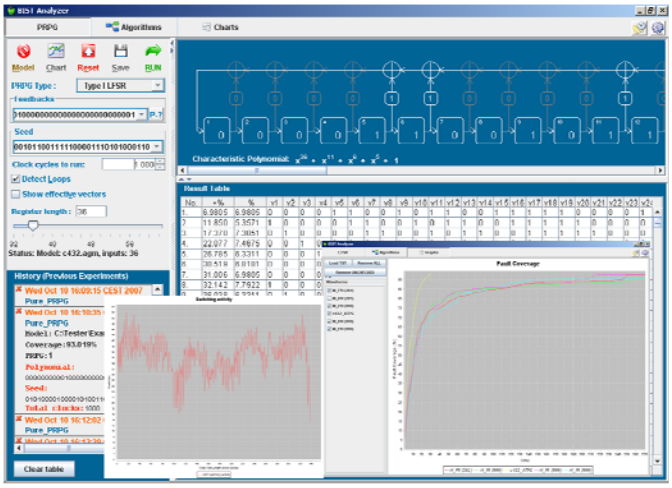

Figure 6. BISTA tool

Unlike other similar systems, this tool facilitates study of various test optimization problems, allows fault coverage analysis for different circuits and with different LFSR parameters. The main didactic aim of the tool is presenting complicated concepts in a comprehensive graphical and analytical way. The multi-platform JAVA runtime environment allows for easy access and usage of the tool both in a classroom and at home. The BISTA represents an integrated simulation, training, and research environment that supports both analytic and synthetic way of learning.

F. Fault Diagnosis

The tool set DIAGNOZER (Figure 7) represents a multifunctional remote e-learning environment for teaching research by learning and investigating the problems of fault diagnosis in electronic systems [9,10]. It is a collection of software tools which allow simulating a system under diagnosis, emulating a pool of different methods and algorithms of fault location and analyzing the efficiency of different embedded self-diagnosing architectures, and investigating the effect of real physical defects in electronic circuits. Both, fault model based and fault model free approaches to fault diagnosis as well as cause-effect and effect-cause techniques of fault location are supported in the presented environment. Also different embedded BIST and self diagnosis architectures are emulated to evaluate the efficiency of diagnosis. The emerging novel fault model free approach is

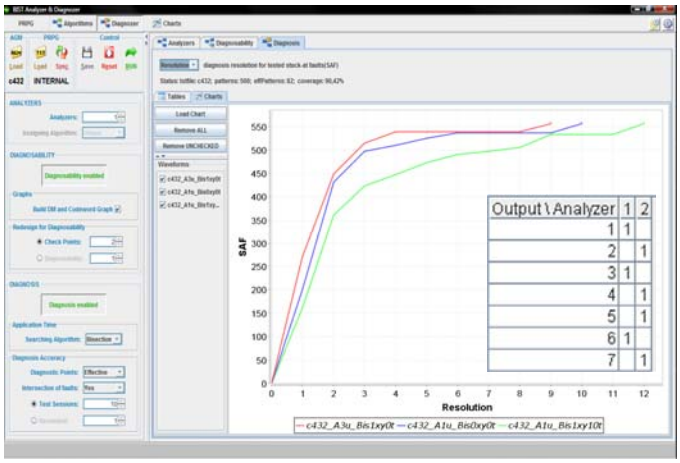

Figure 7. DIAGNOZER tool

supported by the tool DefSim which allows to investigate real physical transistor-level defects in electronic circuits and to map them on the higher logic level for interpreting the fault model free diagnostic results.

III. TEACHING SCENARIOS

Based on the presented teaching tools the following hands-on laboratory scenarios have been developed and are currently used in teaching and self-learning of test related topics.

A. Introductory Topics of Digital Test

Scenario 1: Logic level testing of digital circuits

Learning logic level test involves the following exercises with using the applets LLA:

- manual test generation for a given gate-level circuit,
- analyzing the quality of tests by fault simulation,
- generating fault tables and fault locating procedures,
- creating procedures for fault diagnosis and locating faults in a circuit.

The task of test generation consists of finding a set of test patterns which is able to detect all the possible faults in the circuit. The students can try to minimize such sets of test patterns. Another exercise is to generate a set of diagnostic test patterns which can be used to locate any possible fault.

Some of the tasks can be organized in a gaming style or as a competition between students. For example, a fault can be inserted into a circuit by the teacher, and a competition between students will be thereafter carried out in a manner who is the first who can localize the fault i.e. who will be able to use the minimum search steps. This way of working with applets makes learning very exciting.

B. High-Level and Hierarchical Test

Scenario 2: RTL level testing of digital systems

Learning higher RTL level test involves the following scenarios with using the applets RTLA:

- design of a data and control paths (microprograms) on RT level with investigation of the tradeoffs between speed and HW cost,
- RT-level simulation and validation of the created microprograms,
- functional high-level test generation and low level test quality (stuck-at fault coverage) measuring.

In the functional test mode first, the cheapest test technique is investigated, which does not require designing special test programs and embedding of special test structures into the system. The required level of fault coverage must be achieved by a smart selection of input data. The sole checkpoint allowed for catching the fault is the data path primary output.

Scenario 3: Hierarchical testing of digital systems

Hierarchical test mode for digital systems is investigated according to the following scenario:

- for each selected functional unit (FU) in the system, gate-level local tests are generated, and the fault coverage of these tests is calculated,
- for each FU, a dedicated high-level (RTL) symbolic test microprogram is generated,
- for each FU, a hierarchical test by embedding its local tests into the RTL symbolic test is generated,
- the hierarchical fault simulation in order to evaluate the global fault coverage of the whole test program is executed.

C. Defect Level Test

Scenario 4: Introduction to physical defects

Learning defect level testing is supported by the DefSim environment. All the exercises can be divided into two groups: less advanced and more advanced ones. The first group of tasks is targeted on students whose main specialization is general microelectronics:

- getting a truth table of good (without defects) CMOS simple and complex standard gates,
- getting a truth table of good (without defects) small combinational circuits,
- repeating the steps above but with a given defect of a certain type in order to observe how the circuit's function is modified by the defect,
- getting basic knowledge of voltage and current testing principles.

The second group is for students in more specialized courses and described in scenario 5.

Scenario 5: Test generation for defects

In the DefSim environment, the user gets a chance to compare the efficiency of different logic level fault models in the way they are capable of covering all shorts and opens in a CMOS circuit. The students will learn in practice that some simple defects represent a real challenge especially from the diagnostic point of view. Since DefSim supports voltage and IDDQ testing, the user can compare the efficiency of both

methods in terms of fault detection. In most cases their performance is noticeably different. The following DefSim-based exercises are targeting defect level test generation:

- test generation for opens in a small circuit using a transistor-level schematic and in a bigger circuit using a logic-level schematic,
- finding all possible test vectors for a given short and calculation of its truth table,
- checking the efficiency of stuck-at fault (SAF) test in detection of shorts and opens,
- detection and localization of an unknown defect,
- study of shorts that form memory elements inside combinational circuits.

D. Built-In Self-Test

Exercises on the BIST are based on the tool BISTA, and the following problems are targeted: PRPG solutions and their mathematical models, test pattern generation (TPG) configurations, test quality issues, fault coverage improvement, and TPG optimization.

Scenario 6: Introduction to PRPG

The scenario is devoted to investigation of the basic concepts of PRPG and ensures that the students understand the subject and that they are ready to proceed with the rest of the work. The scenario consists of:

- checking of the primitivity of polynomials for the following PRPG types: LFSR, modular LFSR and cellular automaton,
- generation of pseudorandom test patterns with 100% fault coverage for a given small circuit with as short test length as possible.

Scenario 7: PRPG Optimization

In this scenario the students study and compare efficiency of pure PRPG with reseeding technique and hybrid BIST methods. The scenario consists of the following tasks:

- generation of pseudorandom test patterns for a given large circuit with as high fault coverage as possible by using all available PRPG types,
- comparison of different PRPG types,
- investigation of the BIST with reseeding,
- investigation of the hybrid BIST.

E. Fault Diagnosis

Exercises are based on the tool DIAGNOZER, and target the following topics: methods and strategies of diagnosis, improving diagnostic resolution, optimization of procedures, and investigation of the diagnosibility of systems.

Scenario 8: Diagnostic response analysis

In this scenario, students investigate the properties of the response analyzer based on designing different architectures of the analyzer block. The scenario involves the following tasks:

- investigation of the dependency of the diagnostic resolution on the numbers and lengths of signature analyzers in the response analysis block,
- investigation of the dependency of the diagnostic resolution on the length of the diagnostic procedure.

Scenario 9: Diagnostic algorithms

Students study the properties of different methods and strategies of fault diagnosis: binary bisection of patterns, binary bisection of faults, jumping and doubling. They study also the cause-effect and effect-cause diagnostic approaches and analyze the efficiency of the fault model free diagnosis by comparing it with classical SAF model diagnosis. The scenario involves the following research exercises:

- evaluation of the diagnosibility of circuits for different approaches like cause-effect, effect-cause and fault model free diagnosis,
- comparison of the diagnostic resolution and test lengths for different fault diagnosis methods,
- redesign circuits for better diagnosibility.

The presented scenarios are adapted for both analytic and synthetic study, where the students first learn the subject by observation (using prepared examples) and then generate and/or solve their own specific exercises. The scenarios cover various strategies and methods of organizing and optimizing test generation and fault diagnosis.

IV. EVALUATION

The described teaching scenarios had been successfully practiced at many European universities such as Linkoping in Sweden, Darmstadt and Ilmenau in Germany and of course at the Tallinn University of Technology.

At the Ilmenau University of Technology the course was evaluated by the students with very positive results: On a scale from 1(best) to 5 (worst) the overall rating of the lecture was 1,3. The use of the interactive tools was rated with 1,1. The students stated out that they have learned much by own experiments with the interactive tools. It was also positive mentioned that the relation between learning outcome and necessary learning time was very efficient.

V. CONCLUSIONS

The presented environment for carrying out different laboratory research scenarios to get hands-on experience in the field of testing and fault diagnosis of complex digital systems allows students to inspect the taught subjects by individual research experiments. The proposed environment supports distance learning as well as a web-based computer-aided teaching. The interactive modules are focused on easy action and reaction, are attractive and encourage students for critical thinking. The tasks chosen for hands-on training represent simultaneously research problems, which allow fostering in students problem solving skills and creativity.

The described environment is introduced into teaching process at the Tallinn University of Technology in Estonia and is suited for using at all levels – bachelor, master and doctoral study. It has been used also in teaching at several universities in Sweden, Germany and Portugal, and has got a positive feedback from students.

Acknowledgements: The work has been supported by Estonian SF grants 7068, 7483, EC FP7 IST project DIAMOND, ELIKO Development Centre, and Research Centre CEBE funded by EU Structural Funds.

REFERENCES

[1] http://www.pld.ttu.ee/tt/

[2] http://www.pld.ttu.ee/applets/td/td_exercises.html

[3] http://www.pld.ttu.ee/applets/rtl/rtl_exercises.html

[4] W. Pleskacz, A. Jutman, R. Ubar, and S. Devadze, "DefSim ? the defective IC", Design Automation and Test in Europe (DATE 2007), University Booth, Nice, France, April 16-20, 2007, article S-96.

[5] http://defsim.pld.ttu.ee:8080/defsim/

[6] J.M.Ferreira, E.Sousa, R.Ubar, A.Jutman, S.Devadze, "Remote boundary-scan testing through micro-webservers" in Proc. of 3rd IEEE International Conference on e-Learning in Industrial Electronics - ICELIE'2009, Porto, Portugal, Nov 3-5, 2009.

[7] http://www.pld.ttu.ee/applets/bs/bs_exercises.htm

[8] A.Jutman, A.Tsertov, A.Tsepurov, I.Aleksejev, R.Ubar, H.-D.Wuttke, "BIST Analyzer: a Training Platform for SoC Testing" in Proc. of Frontiers in Education Conference (FIE'07), Milwaukee, Wisconsin, USA, October 10-13, 2007.

[9] http://www.pld.ttu.ee/applets/bista/

[10] R.Ubar, A.Jutman, J.Raik, S.Kostin, H.-D.Wuttke. DIAGNOZER: A Laboratory Tool for Teaching Research in Diagnosis of Electronic Systems. Proc. of 2009 International Conf. on Microelectronic Systems Education (MSE'09), San Francisco, July 25.-27, 2009, pp.12-15.