

# E-Learning Environment in the Area of Digital Microelectronics

A. Jutman<sup>1</sup>, A. Sudnitson<sup>1</sup>, R. Ubar<sup>1</sup>, H.-D. Wuttke<sup>2</sup>

<sup>1</sup>Tallinn University of Technology, Raja 15, 12618 Tallinn, Estonia

<sup>2</sup>Technical University of Ilmenau, Helmholtzplatz 1, 98693 Ilmenau, Germany  
email: artur@pld.ttu.ee

**Abstract** — This paper presents a new teaching concept with, subsequently, a method that combines training and research activities in a common environment. It is developed in the frame of a cooperative project supported by the European Union, called “REASON- Research and Training Action for System on Chip Design”. For teaching the advanced topic of design of digital microelectronics we developed a research training environment supporting different research scenarios. These scenarios support analytical and synthetical research. In the paper we present some examples of research directions and the conception of the research training environment.

## I. INTRODUCTION

The rapid developments of technology and design automation tools in the areas of deep-sub-micron electronics are enabling engineers to design larger and more complex hardware systems. At the same time, the common ASIC (Application Specific Integrated Circuit) concept of hardware realization becomes supplemented by newcomers like programmable and reconfigurable logic devices, System-on-Chip (SoC), and Network-on-Chip (NoC). This progress is driving engineers towards new design methodologies. Entering into the SoC/NoC era means that advanced topics of hardware design and test must become an integral part of the VLSI and system design courses.

The task of decomposition has been a classic problem of discrete system theory for many years [1]. Typically, a large hardware behavioral description is decomposed into several smaller ones. One goal is to make the synthesis problem more tractable by providing smaller sub-problems that can be solved efficiently. Another goal is to create descriptions that can be synthesized into a structure that meets the design constraints. In the past, the synthesis focused on quality measures based on area and performance. The continuing decrease in feature size and increase in chip density in recent years have given rise to consider decomposition theory for low power as new dimension of the design process.

The more complex electronics systems are getting, the more important the problems of test and design for testability are becoming. At the same time, the expenses of verification

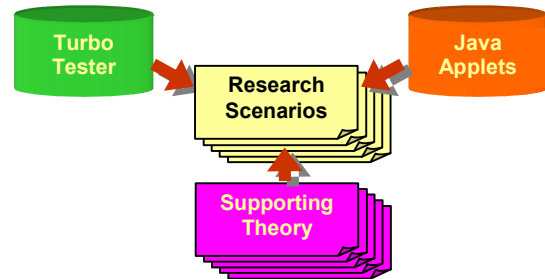


Fig. 1. Overview of the e-learning environment

and testing are getting the major components of the design and manufacturing costs of new electronic products. Thus, the design and test cannot be seen anymore as separate engineering issues. The next generation of engineers involved with VLSI technology should be made aware of the importance of test and trained in test technology to enable them to produce high quality and defect-free products.

This paper describes an e-learning environment developed in Tallinn University of Technology, which makes use of a diagnostic software package called Turbo Tester (TT) and a set of Java applets. It is aimed at teaching basics as well as advanced topics from such areas of digital microelectronics as the hardware design on the register transfer level, automata decomposition, and digital testing and diagnostics of integrated circuits [1,2].

The environment consists of several interrelated modules shown in Figure 1. The PC-based tools installed locally and Java applets invoked remotely via Internet supplement each other and form the engine of the whole concept. The central point of the presented concept is research scenarios – a set of problems and experiments that represent virtual laboratory works, where students learn diagnostic software and get acquainted with common concepts and problems of testing and diagnostics. There are scenarios for beginners and for advanced study. The user-friendly graphical environment created by the applets is the best option for beginners. More advanced study is possible with scenarios, which make use of TT tools. The full text of the scenarios is available in the Web [8].

In the next section of the paper we would like to define the core skills that should be trained in Web-based courses. The

third section explains the methods realized for the research training environment. We will show that they have some features predestinating them for a use in several learning scenarios. Section four gives an overview of the “Living Pictures” concept [3], explains the usage of such modules and gives an example. We will show that our teaching modules are usable as a tool for predefined experiments as well as a generator of new experimental setups.

Our aim is to demonstrate that learners can solve more complex tasks and get a deeper insight of the learning subject by using that method.

## II. THE CORE SKILLS FOR WEB-BASED COURSES

Having in mind the new trends of development of electronic technologies and increasing importance of testing, the core skills, we want to teach in the form of Web-based courses are the following: basic concepts of test generation and fault diagnosis, advanced topics of design, test, and design for test as well as skills in the boundary scan and interconnect diagnosis. In detail these topics can be characterized as follows:

### A. Basic concepts of test generation & fault diagnosis

Main principles of logic-level test generation and fault diagnosis are the fundamentals of all the modern VLSI CAD and diagnostic software. Good understanding of these basics gives students vital background and skills in their future profession. Such basic aspects of testing should undoubtedly be taught to modern students as: manual and pseudo-random test pattern generation (TPG), fault simulation, combinational and sequential fault diagnosis, and investigation of built-in-self-test (BIST) solutions [2]. All these topics should be considered in a simplest possible manner. The target audience will be the very beginners.

### B. Advanced topics of design, test, and design for test

Here, we would like to combine and illustrate many different problems related to register-transfer- (RT-) level control intensive digital design, test, and design for test (DFT). In this way, a unique possibility to teach these topics is going to appear in a consecutive iterative approach. The range of considered problems includes:

- design of the data path and the control part (micro-program) on RT-level (RTL),
- investigation of tradeoffs between speed of the system and hardware cost,
- RT-level simulation and validation,
- deterministic test generation and functional testing,
- fault simulation,
- logic BIST, circular BIST, functional BIST, etc.,
- general concepts of design for testability.

The target audience are the students, who have been already acquainted with the basic concepts of test generation and fault diagnosis, possibly by means considered in the previous subsection.

### C. Decompositional synthesis of the control part

At first, students need to understand the essence and importance of this problem for today’s hardware designer. The fundamentals of decomposition - the algebraic structure theory of sequential machines should be understood by students in the first place.

On the other side, the construction of network of final state machines (FSMs) that realizes the controller is an important practical part of the concept. It is based on decomposition partitions on the set of states of prototype FSM. The designer should be able to choose decomposition partitions to meet a requirement on the distribution of primary inputs and outputs among sub-FSMs. This is NP-hard problem, and amounts to solving a face hypercube-embedding problem [1,4].

Another important problem related to FSM decomposition is the power constrained design. The idea is to introduce additional *idle* state into the FSM. Then the network of FSMs should be designed so that only one component is operating at a time and all the other components are suspended. The students will learn how to decompose an FSM into such a network.

### D. Boundary Scan and interconnect diagnosis

After the students have learned the basic and advanced topics in testing they should finally get some ideas about modern testing techniques and standards used in industry. The main test strategy for SoC is the IEEE P1500 “Standard for Embedded Core Test.” This standard has evolved from the earlier development in this area, namely, the IEEE standard 1149.1 “Test Access Port and Boundary-Scan Architecture”, which was created by Joint Test Action Group (JTAG) and aimed at testing PC-Boards [2]. It is of no doubt that these very important new standards have to be taught to future designers and test engineers. In the first place this holds for the latter and the simpler one, shortly called the Boundary Scan (BS) standard.

## III. METHODS REALIZED FOR WEB-BASED TRAINING

Our methodical approach is aimed at step by step involvement of students into research topics. In every level of knowledge they get the possibility to repeat experiments demonstrated by the lecturer as well as to create own examples and to make own experiments. That is why we create already the applets for teaching basic knowledge in the form of “Living Pictures”. Typically, “Living Pictures” can be used like tools, supporting a design or test step. In that way students become familiar with self creation of examples and exploration strategies. In short, the research and training method can be explained on the following example.

Given a number of research tools dealing with simulation, test generation, fault simulation, and fault diagnosis and given a set of research objectives in the form of benchmark circuits (combinational circuits, sequential circuits and digital systems). Each research tool has some internal algorithms that can be used for special fault types and test tasks. Otherwise each benchmark circuit can be tested at different representation levels. For teaching the use of such tools we

developed a research training environment supporting the following research scenarios:

**Analytical research** (*for getting new knowledge*): Study a tool at fixed parameters (working mode, algorithm, restrictions) for a research objective (circuit) at different representation levels, of different type and at different parameters and investigate a tool at different working modes (or tools with different algorithms) for a research objective (circuit) at a fixed level, type and fixed parameters

**Synthetical research** (*for creating new methods, algorithms*): Combining different tools for reaching synergism of their positive features (to increase performance, accuracy, quality, to optimize the use of resources).

#### A. The basic concepts of test generation & fault diagnosis

A fault in a digital circuit is a permanent and unintended derivation from the specification, which results in a wrong behavior of the circuit. The effect of such misbehavior is usually observed at the outputs of the circuit. There are many different fault models studied in the literature. However, the most common one is still the stuck-at fault model. It assumes that a circuit is represented by a network of interconnected logic gates called net list. Each connection between the gates can be stuck to either 0 or 1.

There are different test generation methods proposed for this fault model. The simplest one suggests tree basic steps of test generation: fault sensitization, fault propagation, and line justification. The students are supposed to use this technique for manual test generation. There are also various automatic test generation techniques available. The students should begin with a simple pseudo-random one, which is commonly used in different BIST structures.

Another important basic concept of testing is the fault diagnosis. Diagnosis consists of locating the physical fault(s) in a structural model of the unit under test (UUT). The degree of accuracy to which faults can be located is called diagnostic resolution. The diagnosis process is often hierarchical, carried out as a top-down process (with a system operating in the field) or bottom-up process (during the fabrication of the system). In general, the process of diagnosis can be classified into two different types: combinational and sequential diagnosis.

This combinational approach does most of the work before the testing experiment. It uses fault simulation to determine the possible responses to a given test in the presence of faults. The database constructed in this step is called a fault table or a fault dictionary. To locate faults, one tries to match the actual results of test experiments with one of the pre-computed expected results stored in the database. The result of the test experiment represents a combination of effects of the fault to each test pattern. That's why we call this approach the combinational fault diagnosis method. If this look-up process is successful, the fault table (dictionary) indicates the corresponding fault(s).

In sequential fault diagnosis the process of fault localization is carried out step by step, where each step

depends on the result of the diagnostic experiment at the previous step. Such a test experiment is called adaptive testing. Sequential experiments can be carried out either by observing only output responses of the UUT or by pinpointing by a special probe also internal control points of the UUT (guided probing). Sequential diagnosis procedure can be graphically represented as diagnostic tree. The applet for teaching these basic concepts was presented in [4].

#### B. Register transfer level design, test, and design for test

The toolkit of the modern design and test engineer contains quite a few methods of testing of a SoC design. All of them are derived from the earlier tools and have been adopted for the new paradigm. With our teaching system we are aimed at showing a variety of different modern testing techniques including functional and deterministic testing and a number of BIST solutions.

**Functional Test:** It is the cheapest test technique to be studied. It does not require designing special test programs and embedding of special test structures into the system. The same unmodified micro-program and data path configuration are used instead. The required level of fault coverage must be achieved then by only a smart selection of input data. The sole checkpoint allowed for catching the fault is the data path primary output. Moreover, this output only can be observed at that time, when the micro-program outputs the final result.

**The BIST:** The deterministic testing via primary inputs/-outputs is one of the most efficient ways of testing. However, it does not provide access to internal signals of the system under test. This problem is addressed by various DFT and BIST solutions. Usually it is a scan-path with a random test pattern generator (TPG) and one or more signature analyzers (SA). In scan-path technology the inputs and the outputs of the combinational blocks in the data path are directly accessible by TPGs, SAs or TPG/SA (combined TPG and SA).

It is common that in the Logic BIST (L-BIST) method the TPG and SA functions must be separated and implemented in different registers. On the contrary, in Circular BIST (C-BIST) both TPG and SA are situated in the same register. The latter approach, being cheaper, usually does not provide a test of the same quality level as the former one.

There is another BIST approach, which we are going to implement. It is called the Functional BIST (F-BIST). This idea has very much in common to functional testing. The only difference between the two concepts is that in the former one there is possibility to insert SAs at any arbitrary point within the data path. In this way, the observability of the system is increased, since each such SA is capable of collecting data at each clock by compressing it into an observable signature.

#### C. Decompositional synthesis of the control part

The task of the FSM decomposition is essential to sequential circuits design optimization in implementation-independent manner. During the last ten years, researches on decomposition techniques for low power have intensified; a range of

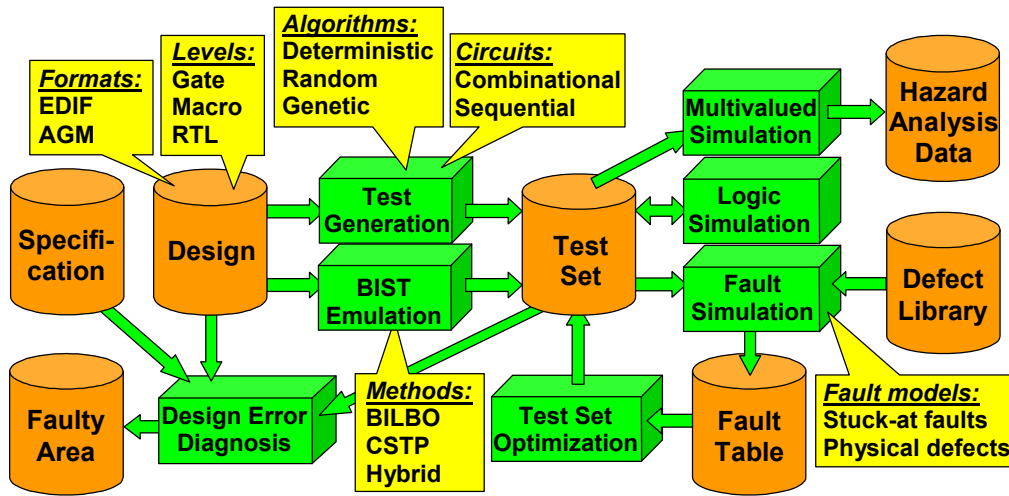


Fig. 2 Overview of Turbo Tester environment

techniques has been proposed for the optimization of circuits for low power [5]. Theoretical background of our system is the partition pair algebra proposed in [6]. The importance of this theory lies in the fact that it provides a direct link between algebraic relationships and physical realizations of FSMs. The mathematical foundation of this theory rests on algebraization of the concept of “information” in a machine. It provides the algebraic formalism necessary to study problems about the flow of this information in machines as they operate.

Our main goal has been the simplification of teaching methods of decomposition synthesis of high complexity FSMs and implementation of them as a web-based computer design system. The system is targeted at the algebraic structure theory of FSMs and its development in accordance with the needs of digital systems design practice to handle the task of partition of hardware description into a network of interconnected FSMs targeting various optimization criteria. Consideration of decomposition synthesis leads to investigation of hard NP-complete combinatorial problems. The synthesis tool, which we are developing, should not be only an educational system but it should be also a research instrument.

#### D. The Boundary Scan and interconnect diagnosis

Before the BS standard was introduced, in 70s and 80s the main board test technique was the “bed of nails”, which ensured the physical contact to any desired point of the circuitry. This method, however, was limited to boards with two conductive layers only. Moreover, the circuit packaging was restricted to the “dual in line”. Modern PC-Boards have a number of inaccessible internal layers as a rule, and the modern circuit packaging includes types like various “grid arrays” (PGA, BGA), which have a lot of inaccessible pins. These difficulties have led to introduction of the Boundary Scan standard.

The BS architecture implies the insertion of scan chains in such a way, that each pin of each chip receives an internal control point. During the normal operation mode the scan cells are transparent, while in the test mode they provide the possi-

bility of driving any of the control points to a desired value as well as reading the results of circuit operation. The standard defines also the Test Access Port (TAP) and the TAP Controller. All these structures are quite complicated and hard to be taught in a traditional manner during a lecture. Therefore, new effective dynamic teaching concepts must be introduced.

#### IV. STRUCTURE AND CONCEPTS OF REALIZATION

Teaching the basics of digital test and testable design means teaching a lot of complex interrelations that have to be explained, at first, one by one and then in their dynamic interactions. Traditional teaching methods using desk, overhead or “PowerPoint™” presentations can explain those connections only partially. After the lecture the dynamic part of the lecture, the connections created by the teacher between different subjects, gets lost. The static part of the whole scenario that students find in their notes, lecture handouts or books does not help well when they try to solve some given problems by using the learned methods. They can do it just as good as they can recall the lecture.

An enhancement of this situation can be reached, if the whole material, containing all scenes of the dynamic process would have been well illustrative and easily accessed by students. The traditional VLSI test generation and fault simulation software does not help to improve the situation. It is usually expensive (cannot be installed on any computer) and unable to handle large numbers of students simultaneously. Moreover, professional CAD systems are not originally designed for teaching purposes in educational courses. They are rather too bulky and hard to comprehend for a beginner.

Therefore, the core of the teaching conception presented here is a combination of educational PC-based CAD software and a set of Java-applets of a special type, which we developed accordingly to the idea of “Living Pictures” [3].

The PC-based tool set is called Turbo Tester (TT) [7,9] and consists of the following main tools (Fig. 2): test generators based on various algorithms, logic and fault simulators, a test

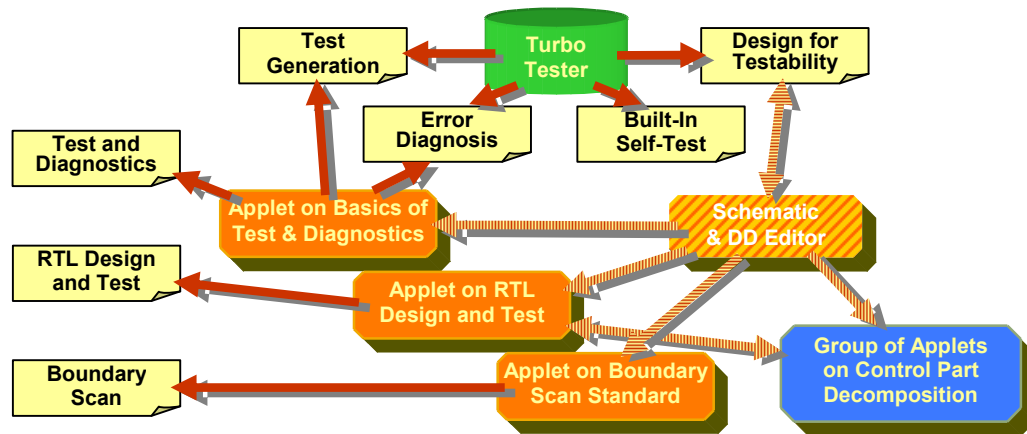


Fig. 3 Relationship between the applets on hardware design and test, the TT, and research scenarios

optimizer, a module for hazard analysis, a simulator and test generator for defects, built-in self-test simulators, design verification and design error diagnosis tools. This range of compatible diagnostic tools forms, via their interaction and complementary operation, a homogeneous research environment, which provides good possibilities for laboratory training and experimental research.

The general idea behind the Java applets is a bit different. They mainly aimed at supporting the concept of game-like style of learning via easy action and reaction, learning by doing, and concentration on most important topics in the simplest possible way. The main features of “Living Pictures” conception incorporate: graphical representation of the learning subject, dynamic content, user-friendly interface, availability of various examples. Most of the applets are completed and available by now on-line [10], while some are still under development and testing phase.

The first large group of Java applets covers the microelectronics hardware design and test area. The second group of applets targets the problem of decomposition of final state machines (FSM). Figure 3 represents an overall structure of relations between these applets, the Turbo Tester, and the research scenarios. As it is seen from the figure, the scenarios form two distinct groups: Java applet-based (and therefore available on the Web) ones and TT-based (advanced) ones. Several advanced scenarios also make use of some functionality of the applets. Another group of scenarios fully exploit the functionality of the corresponding applet and therefore each such applet-scenario pair represents a self-contained system aimed at teaching target area of knowledge and engineering.

We have selected the Java technology for our interactive teaching system since it is well supported by main operating systems like Windows, Linux, and Solaris. Java allows for graphical content to be easily created. It also has well developed means for creating the user interface. Furthermore, Java applets are running on any standard browser like Netscape and Internet Explorer connected to the Internet. The latter makes it easy for students, from universities all over the

world, to use this system at any time and in any place.

We tried to design our applets in a uniform way; so that the user once got acquainted with the overall style does not have to spend his time learning the new style once again from the beginning. The java applets will be supplemented by a schematic editor that allows to edit circuits in different notation forms as for instance decision diagrams. It will close the methodical gap between the applets for teaching and the tools for research such as the Turbo Tester. Small designs can be explored first by the easy-to-use applets and after that exported to the tools.

In the following we describe the capabilities of the realized applets in different scenarios using the example shown in Fig. 3. This is the applet on RTL design, test and design for test.

The main part of the applet is the schematic view panel that provides the schematic representation of the target system. It reflects the internal structure of the data path, which is well re-configurable (different functions can be selected in each functional unit). The control part of the design should be specified in the micro-program table. This can be done in a relaxed way giving the possibility of designing many different devices using the same basic data path. Even the same device can be designed in many different ways using less or more of the hardware resources. This may result in a lower or higher speed of the system, which can also be measured.

When the device is ready it can be simulated and validated. The results of simulation are stored in a special table as well as shown graphically in the schematic view panel. There is also a special table for RT-level fault simulation. This table provides fault coverage for each functional unit separately as well as for the data path as a whole. There are several modes for test generation. The functional testing mode uses the “normal” operands and simulates the device during normal working mode.

Another mode, the deterministic test mode, allows manual test generation on the gate level for each of the functional units used in the device separately. A gate-level schematic of those units is provided. This mode needs a set of special test

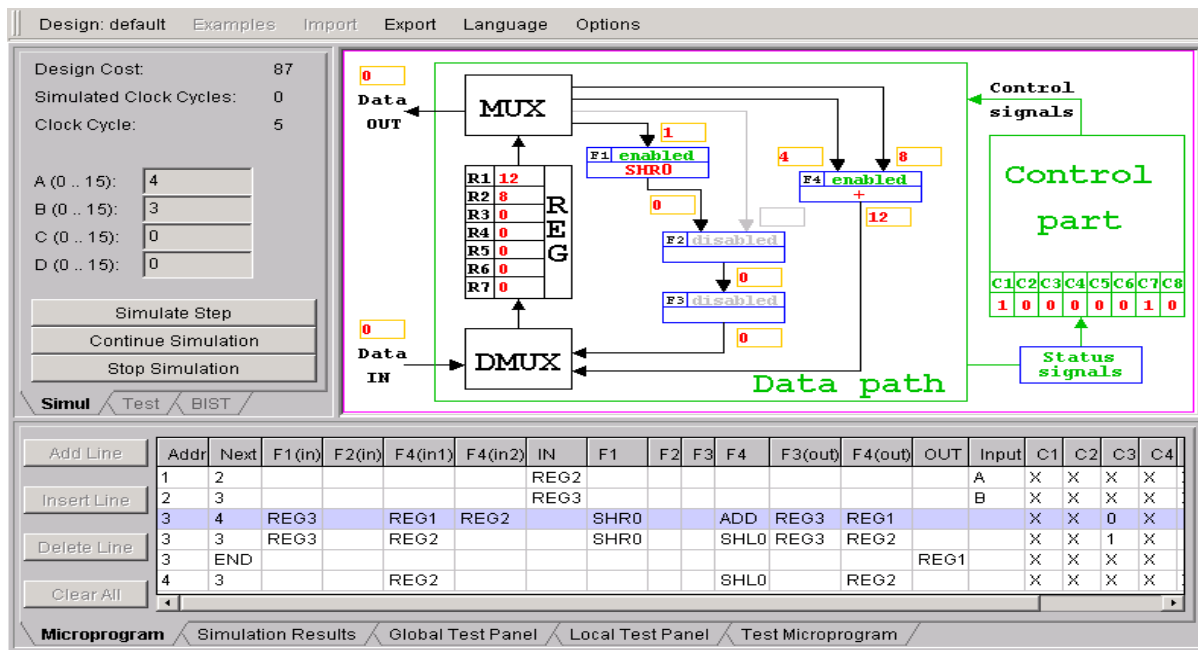


Fig. 3 Java applet on RT-level design, test and design for test

micro-programs, which are generated on the fly and can be modified by the user.

There are different BIST modes selected for this applet. Logic BIST and Circular BIST are aimed at illustration of BILBO and CSTP approaches on the RT level. Functional BIST is a special mode similar to Functional test where it is possible to use a number of signature analyzers to improve the design testability. The main advantage of the applet is the fact that it gives a unique possibility to teach all the mentioned problems in a consecutive iterative way using the same teaching system.

## V. CONCLUSIONS

The core of this environment is a set of research scenarios (virtual laboratory works) based on two types of software: *a)* diagnostic package Turbo Tester and *b)* range of Java applets. While the TT should be installed on a local computer, the applets are invoked remotely via Internet. The Turbo Tester provides a homogeneous research environment, which allows for interesting experimental research to be conducted. The Java applets, in their turn, provide an attractive game-like milieu, which is important especially for beginners.

By the use of web-based media we achieve: presentation of course material independent of place and time, individual learning according to the students' own needs, quick cross-referencing by hyper-linked texts, new forms of communication between teachers and students.

In addition to these benefits, the presented conception allows inspecting the taught subjects by individual experiments and basic research actions. Thus, we are able improving the skills of students to be educated for digital hardware design so that they can get deeper knowledge about inter-

dependencies between several design steps. Therefore, the conception presented here is suitable for a broad audience of learners who are interested in studying different concepts of testing and diagnostics of integrated digital circuits.

## ACKNOWLEDGMENTS

This work was supported partly by the EU Framework V project REASON, by the Thuringian Ministry of Science, Research and Art (Germany), and by the Estonian Science Foundation Grants No. 5643, 5649, and 5910.

## REFERENCES

- [1] P. Ashar, S.Devadas, and A.R. Newton, *Sequential Logic Synthesis*, Kluwer Academic Publishers, Boston, 1992.
- [2] M.L. Bushnell, V.D. Agrawal, *Essentials of Electronic Testing for Digital Memory and Mixed-Signal Circuits*, Kluwer Academic Publishers, Dordrecht: 2000, p. 690.
- [3] H.-D.Wuttke, K.Henke, R.Peukert, "Internet Based Education - an Experimental Environment for Various Educational Purposes". *Proc. of Int. Conf. on Computers and Advanced Technology in Education*, Philadelphia, PA USA. IASTED/ Acta Press No. 292, 1999, pp. 50-54.
- [4] R. Ubar, H.-D. Wuttke, "The DILDIS Project – Using Applets for More Demonstrative Lectures in Digital Systems Design and Test", *31st ASEE/IEEE Frontiers in Education Conference*, October 10-13, 2001 Reno, USA.
- [5] S. Devadze, E. Fomina, M. Kruus, and A. Sudnitson, "Web-Based System for Sequential Machines Decomposition", in *Proc. IEEE EUROCON 2003 International Conference on Computer as a Tool*, Ljubljana, Slovenia, 2003, V.1, pp. 57-61.
- [6] J. Hartmanis, R. E. Stearns, *Algebraic Structure Theory of Sequential Machines*, Englewood Cliffs, N.J.: Prentice-Hall, 1966.
- [7] M.Aarna, E.Ivask, A.Jutman, E.Orasson, J.Raik, R.Ubar, V.Vislogubov, H.-D.Wuttke. "Turbo Tester - Diagnostic Package for Research and Training," in *Scientific-Technical Journal "Radioelectronics & Informatics"*. KNURE. Vol. 3(24), 2003, pp.69-73.
- [8] Laboratory training URL: <http://www.pld.ttu.ee/testing/labs>
- [9] Turbo Tester home page URL: <http://www.pld.ttu.ee/tt>
- [10] Java applets home page URL: <http://www.pld.ttu.ee/applets>