Applets for Learning Digital Design and Test

R.Ubar[1], A.Jutman[1], M.Kruus[1], H.-D.Wuttke[2]

Tallinn University of Technology_1,
Ilmenau Technical University_2

Key words: *Tools for interactive learning and teaching, Web and Computer-based learning, Reusable Learning Objects*

Abstract:

An environment targeted to e-learning is presented for teaching design and test of electronic systems. The environment consists of a set of Java applets 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, 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. The tasks chosen for hands-on training represent simultaneously research problems, which allow to foster in students critical thinking, problem solving skills and creativity.

1 Environment of Web-based learning tools for Design and Test

Learning in practical situations and learning by doing is an efficient way to learn because a student will form mental pictures about the things to be learned, and they will be remembered better, too.

The e-learning software developed supports the action based training in digital design and test via internet. It offers a set of tools to inspect the different objectives of testing digital logic to be learned, access to multiple learning modules, a big library of examples and the possibility to generate new personal examples. It provides easy action and reaction (click and watch) by using "living pictures", the possibility of distance learning, and learning by doing. The core of that concept are Java-applets (the interactive modules) running over network, using standard browsers like Netscape and Internet Explorer with Java 1.2 runtime plug-in, or with Java 2 applet viewer.

Different types of applets have been created for learning design and test of electronic circuits or systems at the low logic level and at the higher register transfer or behaviour levels.

One class of applets developed can be used for teaching and learning the basics of logic level digital test and testable design as illustrative tool explaining the problems of fault simulation, test generation, design for testability and fault diagnosis. The work window of this program (Fig.1) consists of a test pattern insertion panel, a view panel for design schematics, and a view panel for simulation results like waveforms, fault tables, diagnostic information etc. Test patterns can be inserted manually or generated automatically by different methods. The boxes at the lines on schematics are clickable for inserting or viewing signals during the test generation or fault diagnosis. The described environment is accessible at [4].

Entering the SoC era with its new concepts involves teaching the design of electronic systems on higher levels of abstraction like register transfer level (RTL), instruction set architecture or

behavioral levels. Another class of Java applets has been developed for learning high-level design and test problems in control intensive digital systems. Such topics as design of data-flows and micro-programs of computing algorithms, investigation of tradeoffs between speed and hardware cost in digital design, RTL simulation, design for testability, test generation, built-in self-test (BIST), diagnostic analysis and other related problems are covered by these applets. The work window of this program (Fig.2) consists of three parts – a control panel, a view panel for design schematics, and a panel for micro-program development and viewing simulation results. The described environment is accessible at [5].

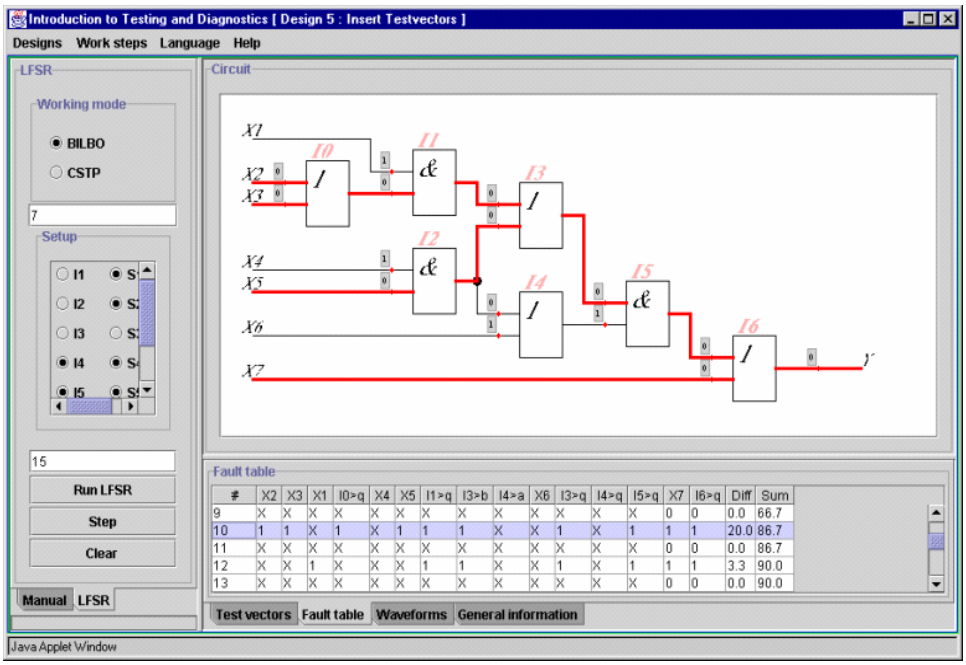

Fig.1. *Living pictures for investigating logic circuits*

2 Learning scenarios on the applets

The applets can be used for carrying out different teaching or learning scenarios. In each scenario they can be reused and embedded into an other context. Thus they are a kind of reusable learning objects (RLOs). For example, logic level research involves the following tasks: Manual test generation for a given gate-level circuit, generating tests with automatic tools and analyse the quality of tests by fault simulation, generating diagnostic tables, creating fault locating procedures, finding a fault in a circuit, creating cost-effective procedures for fault diagnosis.

In Fig.1 a pattern is applied on the inputs of the circuit. On the upper view panel the lines tested by the applied pattern are highlighted. On the lower panel a fault table is shown. The rows correspond to the test patterns and the columns correspond to the lines of the circuit. The entry *x* in the table means that a particular line is not tested by the given pattern. The entry 0 or 1 means that a fault stuck-at-0 or stuck-at-1 of the particular line is tested by the given pattern. In the last column also the percentages of faults detected by the test patterns are shown.

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 a set of test patterns. Another learning scenario is to generate a set of 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 even more exciting.

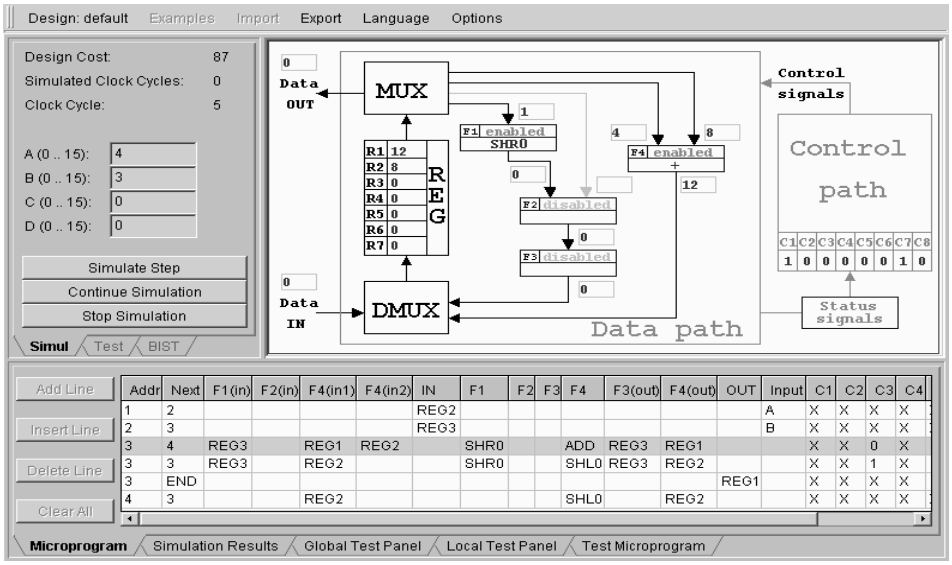

Fig.2. *Applets for investigating digital systems*

Using applets for investigating digital systems a student can exercise register-transfer level (RTL) implementations of more complex functionalities represented by data flow graphs or micro-programs (like multiplication, division, signal processing algorithms etc.). A structure of the data-path and the needed functional blocks for implementing particular micro-operations can be found from the library. Different architectures can be explored and experimented for implementation of a given set of functions or algorithms. The control path is a micro-programmed controller which implements a finite state machine. For the RTL data-path designed by the student, a gate-level implementation is also generated, and the number of gates used in the generated circuit will determine the hardware cost of the implemented design. Students can compare different hardware solutions for the given algorithm and find out the design tradeoffs. For every chosen architecture, the system calculates the cost of the hardware, and the speed of processing (number of clock cycles needed by the developed micro-program) can be measured by simulation.

Different training and research scenarios of testing the design can be exercised by the applet. A *functional testing scenario* means that instead of dedicated test programs the normal working micro-programs are used for testing purposes. This is the usual strategy how the designers verify the created design. The applet allows to measure the real quality of such type of testing. Usually this type of testing gives a very low fault coverage. Then, other test strategies can be exercised, which are based on creating dedicated test programs for the created structure. Generating such test programs and analyzing their quality develops real skills in the student, which are needed in testing and diagnosing electronic products in the industry.

Today, built-in self-test (BIST) is a new paradigm that has already found its place in the production of complex systems like systems-on-chip (SoC) and networks-on-chip (NOC) [6]. Different BIST architectures can be exercised by this applet with measuring their quality (fault coverage) and with investigating the tradeoffs between quality and cost.

3 Conclusions

An environment consisting of web-based applets for hands-on training is presented for improving the skills of students to be educated for hardware and SOC design in test related topics. Based on the described environment, an e-learning conception using simple applets in a form of “living pictures” can be introduced into study programmes at technical universities for teaching courses on digital electronics, design of digital systems, testing and design for testability. The applets may be used for solving real research and engineering problems in the field of electronics design and test which allow to foster in students critical thinking and creativity in an exciting working environment and stimulating atmosphere.

Acknowledgements:

This work has been supported by the Thuringien Ministry of Science, Research and Art (Germany), and by the Estonian Science Foundation grants G5649 and G5910. We also thank E.Orasson and S.Devadze for programming the applets.

References:

[1] Ubar, R, Wuttke H.-D, “The DILDIS-Project – Using Applets for More Demonstrative Lectures in Digital Systems Design and Test”, Proceedings of the 31st ASEE/IEEE Frontiers in Education Conference, FIE’2001, Oct. 10-13, 2001, Reno, NV, USA, pp.SIE-2-7.
[2] S.Devadze, A.Jutman, A.Sudnitson, R.Ubar, H.-D.Wuttke. Teaching Digital RT-Level Self-Test Using a Java Applet. 20th IEEE Conference NORCHIP’2002, Copenhagen, Denmark, November 11-12, 2002, pp.322-328.
[3] R.Ubar, E.Orasson, J.Raik, H.-D.Wuttke. Teaching Advanced Test Issues in Digital Electronics. 6th IEEE International Conference on Information Technology Based Higher Education and Training. July 7-9, 2005, Santo Domingo, pp. S2B-5 – S2B-10.
[4] http://www.pld.ttu.ee/applets/td/
[5] http://www.pld.ttu.ee/applets/rtl/
[6] R.Ubar, J.Raik. Testing Strategies for Networks on Chip. In “Networks on Chip” by A.Jantsch, H.Tenhunen. Kluwer Academic Publishers, 2003, pp. 131-152.

Authors:

Raimund Ubar, DSc, professor
Tallinn Technical University, Computer Engineering Department
Raja 15, 12618 Tallinn, ESTONIA
Email: raiub@pld.ttu.ee

Artur Jutman, PhD, senior researcher
Tallinn Technical University, Computer Engineering Department
Raja 15, 12618 Tallinn, ESTONIA
Email: artur@pld.ttu.ee

Margus Kruus, PhD, associate professor
Tallinn Technical University, Computer Engineering Department
Raja 15, 12618 Tallinn, ESTONIA
Email: kruus@cc.ttu.ee

Heinz-Dietrich Wuttke, PhD, associate professor
Ilmenau Technical University, Faculty of Informatics and Automation
Post box 10 0565, D-98684 Ilmenau, GERMANY
Email: dieter.wuttke@tu-ilmenau.de