

Annals of Computer Science and Information Systems
Volume 4

Proceedings of the E2LP 2014 Workshop

September 7–10, 2014. Warsaw, Poland



Vlado Sruk, Roman Szewczyk, Miodrag Temerinac
(eds.)



Annals of Computer Science and Information Systems, Volume 4

Series editors:

Maria Ganzha,

Systems Research Institute Polish Academy of Sciences and University of Gdańsk, Poland

Leszek Maciaszek,

Wrocław University of Economy, Poland and Macquarie University, Australia

Marcin Paprzycki,

Systems Research Institute Polish Academy of Sciences and Management Academy, Poland

Senior Editorial Board:

Wil van der Aalst,

Technische Universiteit Eindhoven, Netherlands

Marco Aiello,

University of Groningen, Netherlands

Barrett Bryant,

University of North Texas, USA

Ana Fred,

Technical University of Lisbon, Portugal

Janusz Górski,

Gdansk University of Technology, Poland

Mike Hinchey,

University of Limerick, Ireland

Janusz Kacprzyk,

Systems Research Institute Polish Academy of Sciences, Poland

Irwin King,

The Chinese University of Hong Kong, Hong Kong

Juliusz L. Kulikowski,

Należcz Institute of Biocybernetics and Biomedical Engineering Polish Academy of Sciences, Poland

Michael Luck,

King's College London, United Kingdom

Jan Madey,

University of Warsaw, Poland

Andrzej Skowron,

University of Warsaw, Poland

Ivan Stojmenovic,

Deakin University, Australia, and University of Ottawa, Canada

Editorial Associate: Katarzyna Wasielewska,

Systems Research Institute Polish Academy of Sciences, Poland

TeXnical editor: Aleksander Denisiuk,

University of Warmia and Mazury in Olsztyn, Poland

Proceedings of the E2LP 2014 Workshop

Vlado Sruk, Roman Szewczyk, Miodrag Temerinac
(eds.)



2014, Warszawa,
Polskie Towarzystwo
Informatyczne

Annals of Computer Science and Information Systems, Volume 4
Proceedings of the E2LP 2014 Workshop

USB: ISBN 978-83-60810-63-7

WEB: ISBN 978-83-60810-64-4

ISSN 2300-5963

DOI 10.15439/978-83-60810-64-4

© 2014, Polskie Towarzystwo Informatyczne
Al. Solidarności 82A m. 5
01-003 Warsaw
Poland

Contact: secretariat@fedcsis.org
<http://annals-csis.org/>

Cover:

Jana Waleria Denisiuk,
Elbląg, Poland

Also in this series:

Volume 1: Position Papers of the 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), **WEB: ISBN 978-83-60810-55-2, USB: ISBN 978-83-60810-56-9**

Volume 2: Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, **WEB: ISBN 978-83-60810-58-3, USB: ISBN 978-83-60810-57-6, ART: ISBN 978-83-60810-61-3**

Volume 3: Position Papers of the 2014 Federated Conference on Computer Science and Information Systems, **WEB: ISBN 978-83-60810-60-6, USB: ISBN 978-83-60810-59-0**

WELCOME to the E2LP Workshop Application of Innovative Teaching Methods in Embedded Engineering. It is our great pleasure and honour to hold E2LP Workshop collocated with Federated Conference on Computer Science and Information Systems (FedCSIS) as a part of the Education, Curricula & Research Methods (ECRM) thematic area, taking place in Warsaw, Poland, on September 8–11, 2014. On behalf of the E2LP project Consortium and Workshop Organizing Committee we cordially welcome all participants.

Embedded Computer Engineering Learning Platform (E2LP) is a European FP7 project of three years duration, started in September 2012. These annual workshops bring together partners, associated groups and external interested research groups. The aim is to provide a forum for discussing the concepts of Embedded Systems Education, On-Line Learning challenges, Augmented Reality Interfaces, Supported Teaching and Evaluation of Learning Technology and a general theory of software engineering. The present proceedings will be followed by the last proceedings corresponding to next annual workshop to be held in 2015.

The proceedings focus on the outcome of the E2LP project presented through the scientific-technical contributions and aim to be available to a wide scientific community. This workshop provides information about the project structure and different activities around the project with the ongoing documentation of the project.

This project deals with challenges in engineering education for embedded systems at university level which requires a complex and multidisciplinary approach involving understanding of various systems based on different technologies and system solution optimizations. The main idea behind the project is to provide a unified platform which will cover a complete process for embedded systems learning. A modular approach is considered for skills practice through supporting individualization in learning. This platform shall facilitate a

novel development of universal approach in creative learning environment and knowledge management that encourage use of ICT. New learning model is challenging the education of engineers in embedded systems design through real-time experiments that stimulate curiosity with ultimate goal to support students to understand and construct their personal conceptual knowledge based on experiments. In addition to the technological approach, the use of cognitive theories on how people learn will help students to achieve a stronger and smarter adaptation of the subject. Applied methodology requires evaluation from the scientific point of view in parallel with the implementation in order to feedback results to the R&D.

As a result, the proposed Embedded Computer Engineering Learning Platform will help to educate a sufficient number of future engineers in Europe, capable of designing complex systems and maintaining a leadership in the area of embedded systems, thereby ensuring that our strongholds in automotive, avionics, industrial automation, mobile communications, telecoms and medical systems are able to develop. In such manner, the E2LP intends to increase European competitiveness in the learning process of embedded computer engineering, ensuring further technological and methodological development of the educational approach in this field. In-depth information about the project can be found on the project web page <http://www.e2lp.org>.

E2LP Workshop Co-Chairs:

Vlado Sruc, *University of Zagreb, Croatia*

Roman Szewczyk, *Warsaw University of Technology, Poland*

Miodrag Temerinac, *University of Novi Sad, Serbia*

E2LP Workshop

September 7–10, 2014. Warsaw, Poland

TABLE OF CONTENTS

EDUCATION, CURRICULA & RESEARCH METHODS

Call For Papers 1

APPLICATION OF INNOVATIVE TEACHING METHODS IN EMBEDDED ENGINEERING

Call For Papers 3

Computer Engineering Laboratory Course: E2LP Platform Experience 5
Nikolina Frid, Vlado Struk, H. Mlinarić, M. Kovač

FPGA Verification Module 9
Ivan Aleksi, Željko Hocenski

Universities – A View to Pebbles of Mosaic 13
Ivan Kastelan, Gordana Velikic, Milenko Beric, Nikola Teslic, Mark Bocko

An Approach to Improvement of Students Learning of Computer System Engineering 17
Ivan Kastelan, Milos Nikolic, Dusan Majstorovic, Nebojsa Pjevalica

Design of a Platform for Teaching Embedded Systems Engineering 21
Ivan Kastelan, Nikola Teslic, Miodrag Temerinac

Design of Remote Laboratory dedicated to E2LP board for e-learning courses 25
Jan Piwiński, Rafał Kłoda, Roman Szewczyk

Learner-Centered Evaluation of an Augmented Reality System for Embedded Engineering Education 31
Florent Souvestre, Margarita Anastassova, Enara Artetxe Gonzalez, Aratz Setién Gutiérrez, Jorge R. Lopez Benito, Moshe Barak

Small is Beautiful: Embedded Systems Projects in an Undergraduate Software Engineering Program 35
Janusz Zalewski, Fernando Gonzalez, Robert Kenny

Education, Curricula & Research Methods

ECRM is a FedCSIS conference area aiming at interchange of information, ideas, new viewpoints and research undertakings related to university education and curricula as well as recommended methods of doing research in all computing disciplines, i.e. computer science, computer engineering, software engineering, information technology, and information systems. This area spans typical FedCSIS events (conferences, workshops, etc.) with rigorous paper

submissions and review processes as well as panels, PhD and research consortia, summer schools, etc. Events that constitute ECRM are:

- DS-RAIT'14 – 2nd Doctoral Symposium on Recent Advances in Information Technology
- ISEC'14 – 3rd Information Systems Education & Curricula Workshop

E2LP Workshop Application of Innovative Teaching Methods in Embedded Engineering

THE Application of Innovative Teaching Methods in Embedded Engineering workshop is organized by FP7 Project Embedded Computer Engineering Learning Platform (E2LP). It is an international forum devoted specifically to engineering education for embedded systems at university level, establishing of creative learning environment and knowledge management that encourage use of ICT in education. Workshop is hosted by the 2014 Federated Conference on Computer Science and Information Systems (FedCSIS) and is a part of the Education, Curricula & Research Methods (ECRM) thematic area. The E2LP workshop provides the meeting place for the researchers, practitioners and leading professionals from all over the world interested in the field.

TOPICS

Topics of interest include but are not limited to:

- Embedded Systems Education
- On-Line Learning
- Augmented Reality Interfaces Supported Teaching
- Evaluation of Learning Technology

EVENT CHAIRS

Sruk, Vlado, University of Zagreb

Szewczyk, Roman, Warsaw University of Technology

Temerinac, Miodrag, University of Novi Sad

PROGRAM COMMITTEE

Barak, Moshe, Ben-Gurion University of the Negev

Benito, Jorge R. Lopez, CreativiTIC INNOVA S.L Centro Tecnológico de La Rioja

Burkhardt, Jean-Marie, Directeur de recherche Psychologie ergonomie, INRETS, LPC

Gonzalez, Enara Artetxe, CreativiTIC INNOVA S.L Centro Tecnológico de La Rioja

Granic, Andrina, University of Split

Grgic, Dario, University of Freiburg - IMTEK

Kaštelan, Ivan, University of Novi Sad

Kloda, Rafał, Industrial Research Institute for Automation and Measurements

Piwiński, Jan, Przemysłowy Instytut Automatyki i Pomiarów

Skala, Karolj, Rudjer Boskovic Institute

Slubowska, Elzbieta, Vice Head for Education, Institute of Metrology and Biomedical Engineering Faculty of Mechatronics Warsaw University of Technology

Souvestre, Florent, CEA LIST/DIASI, Laboratoire d'Interfaces Sensorielles et Ambiantes

Computer Engineering Laboratory Course: E2LP Platform Experience

Nikolina Frid, Vlado Struk, H. Mlinarić, M. Kovač

Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia
Email: {nikolina.frid, vlado.struk, hrvoje.mlinaric, mario.kovac}@fer.hr

Abstract – In this paper, the preliminary results of E2LP Base Board platform introduction to students of Faculty of EE and Computing - master programme enrolled in Laboratory of Computer Engineering 2 course are presented and discussed. The aim of introduction of new hardware unified platform was to improve practical skills and experience in embedded system design. The students embraced the new platform with enthusiasm and were eager to give the feedback in order to point out the strengths and weaknesses of this platform. The collected information will be a valuable asset for future improvement of the platform.

Keywords – course development; hardware platform, Computer Engineering Laboratory

I. INTRODUCTION

EMBEDDED systems design is a rapidly expanding field growing more diverse as new technologies emerge. While this is an interesting discipline attracting more and more students each year, at the same time rising complexity is rather difficult to master. The working environment is dissimilar from the one peer students enrolled in computer science and engineering programmes encounter (PC and laptop computers). Embedded system design requires comprehensive knowledge and skill for dealing with a wide range of different hardware and software architectures. What is more, because these area is highly related to high-tech industry, the knowledge passed on from teachers to students must be in line with the current standards [2][5].

The goal of Faculty of EE and Computing (University of Zagreb, Croatia) is to provide the students with relevant knowledge and practice in the field of interest. Since the Faculty is one of the E2LP project partners, in academic year 2013/2014 the E2LP platform has been introduced in several courses as a pilot project for testing the new learning platform. In this paper the results of introducing the platform to students enrolled in master programme course Laboratory of Computer Engineering 2 are presented and discussed.

The rest of this paper is organized as follows: Chapter 2 provides general information about the Computer Engineering Laboratory course and the students enrolled in the course; in Chapter 3 the details about laboratory exercises and working environment are given; Chapter 4 discusses the results scored by students together with their feedback and in the final chapter the overall conclusion is given.

II. COMPUTER ENGINEERING LABORATORY COURSE

Computer Engineering Master programme [1] embodies research, design and implementation of computers, computer systems and related software. It includes in-depth knowledge of theory and practical aspects of the design and implementation of computers, computer based systems, mobile and embedded devices/computers, communication systems and other systems that incorporate computers as well as software design with emphasis on applications that require knowledge of the complete system. Computer engineering program provides system based approach to the design of computers, communication system and software as a whole. This program offers core, advanced and forefront knowledge required to creatively envision, conceptualize and design innovative, from simple to complex computers, computer based systems and applications running on those platforms.

The Laboratory of Computer Engineering 2 is a second semester course consisting entirely of laboratory assignments aiming to give the students the chance to put to practice the theoretical knowledge obtained through the study [3][4]. This course is intended to provide practical skills complimentary to theoretical knowledge for students enrolled in two of the following four courses: Computer Systems Reliability, Digital System Design, Real-time Operating Systems and Formal Methods in Computing Systems Design. The exercises try to cover the areas of digital system design, real-time systems, reliability analysis and testing.

A. Students partaking the course

In the summer semester of academic year 2013/2014 twenty five students were enrolled in the Laboratory of Computer Engineering 2 course. The students come with a very diverse background knowledge with respect to embedded system design, which was the core of the Laboratory of Computer Engineering Course 2 in summer semester 2014. The only course which all students have taken is Embedded Systems during third year of Bachelor programme. The majority of students have taken the course Tools for Digital Design and some have taken the course Embedded System Design. The diverse range of prerequisite knowledge about embedded systems the students have, makes designing laboratory task targeting that matter quite a challenge. What is more, the students themselves are not very confident in the

knowledge and skills they poses. In the initial questionnaire, given to students before the course started, on the scale from 1 to 5 they have graded their knowledge of VHDL with 2.4 and skill using Xilinx tools 2.3. However, they have demonstrated the eagerness to learn something new and different in hope it will help them improve their knowledge and skills and possibly prove as an asset in the future development of their careers.

III. EXERCISE OVERVIEW

As mentioned earlier, the Laboratory of Computer Engineering 2 consists only of laboratory exercises. Throughout the semester students are given several problem solving exercises or small projects to gain knowledge and practical skills related to embedded system design.

In summer semester of 2014 the E2LP platform [6][7] was introduced to the course for the first time. The entire set of exercises was written from scratch to focus more on embedded system design and better suit the platform capabilities.

In the rest of this chapter a more detailed insight into the matter covered by the exercises and working environment will be given.

A. Details about exercises

During the semester students did four problem solving exercises on E2LP platform, which were designed to help students gain practical skill and experience in embedded system design.

In the first exercise students were given the task to design their own UART controller for E2LP Base Board platform using VHDL and Xilinx ISE tools [8][9]. This task was intended as an introduction aiming to help students refresh the previously acquired knowledge about basic digital system design.

The following three exercises were more challenging, focusing on designing a more complex system comprising programmable processor, dedicated hardware accelerators and peripheral controllers. Along with the hardware platform, application software was developed too. The idea behind these exercises was to give students a complete overview of embedded system design process – from hardware platform to OS and application software. Higher complexity of these tasks required using more advanced design tools like Xilinx EDK [10][11].

In the second exercise students were required to design a system consisting of programmable MicroBlaze RISC processor [12] connected with basic peripheral controllers (DIP switch, LED and UART). All components of the system were already present as IP cores available as a part of Xilinx EDK Design Suite. After the initial system was implemented and tested, the students had to replace the existing UART core with their own, designed in the previous exercise.

The third exercise introduced multiprocessor system concept to students. They were given the task to design their own matrix multiplier IP core which would serve as a coprocessor to MicroBlaze. While in the second

exercise a simple AXI4-Lite [13] bus interface was appropriate enough to connect all the components in the system, the third exercise required the use of more advanced AXI4 bus interfaces. This provided the students with the opportunity to learn more about the AXI4 bus architecture, one of the most prominent bus architecture standards today.

In the last exercise the emphasis was put on design of software for embedded systems, with operating system kernel at the core. Students were given precise instruction on how to configure the hardware platform (consisting of MicroBlaze processor and basic peripherals) so it would be possible to add a minimal operating system kernel on top of it. The target kernel was *xilkernel* [14][15], specially designed for MicroBlaze processor family.

It must be noted here that a small group of students enrolled in a Digital System Design course did not perform the fourth exercise, but were given a small project as a part of that course. This project also involved designing a SoC with MicroBlaze controller and a set of peripherals, also targeting E2LP Base Board platform.

B. Working Environment

All four exercises targeted E2LP Base Board platform. The components used were the following:

- Spartan 6 FPGA module – for implementing the system designed in VHDL;
- DIP switch, LED and RS-232 interfaces through which the system implemented on FPGA communicated with outer world.

As for the development environment, Xilinx tools ISE and EDK were used. ISE tool was intended for designing a single component (UART controller, matrix multiplier) and EDK for system integration and designing application software. These particular tools were chosen because they are widely used in both education and industry, which implies the students would gain valuable practical knowledge they can put to use in their future careers.

IV. RESULTS AND FEEDBACK

Upon completion of each exercise, the students were graded and given a short questionnaire about their satisfaction with the exercises. In this chapter both the results the students have achieved and the feedback they

Table 1 Lab Scores

	Lab 1	Lab 2	Lab 3	Lab 4*
Total completion	25/25	7/25	7/25	10/15
Minor errors	0	3/25	10/25	5/15
Mostly solved (>60%)	0	15/25	8/25	0
Unsolved or major errors	0	0	0	0

*other 10 students did DSD project

gave through the questionnaires are presented and discussed.

A. Lab Results

The results students achieved in each exercise are given in Table 1. Task completion was expressed using four levels: total completion, minor errors (the system behaved as expected most of the time but some glitches or crashes were possible), mostly solved (some of the required functionalities were missing) and unsolved or with major errors. For each laboratory task the number of students achieving each level of task completion is stated: for example the entry “7/25” under “Total completion” in “Lab3” means that 7 out of 25 students were able to fully complete the exercise.

The statistics show that the second laboratory task was the most difficult for majority of students. This is not surprising since this is the first exercise concerned with SoC design, the subject most students were not very familiar with up to that point. The third lab has also proved to be quite challenging, but the number of students who managed to solve the task entirely or with minor errors has significantly improved.

The significant improvement visible in fourth exercise is not so much the result of a major leap in students’ knowledge level, but more the result of shift of focus from hardware to software design, the topic most students are more familiar with.

B. Student feedback

After each exercise, students were given a short questionnaire about their experience and satisfaction with the laboratory exercise. The questions were the following:

1. On a scale from 1 (lowest) to 10 (highest) how would you grade clarity of theoretical background - documentation, theoretical explanations?
2. On a scale from 1 (lowest) to 10 (highest) how would you grade clarity of technical instructions, exercises and problems?
3. On a scale from 1 (lowest) to 10 (highest) how would you grade total time and efforts required?
4. On a scale from 1 (lowest) to 10 (highest) how would you grade ease of use of the environment - Xilinx software and BIN download software?
5. On a scale from 1 (lowest) to 10 (highest) how would you grade ease of use of the E2LP platform?
6. On a scale from 1 (lowest) to 10 (highest) how would you grade feeling of immersion - being part of the environment, control over the system?
7. On a scale from 1 (lowest) to 10 (highest) how would you grade to what extent do you think you learned something valuable?
8. On a scale from 1 (lowest) to 10 (highest) how would you grade the overall satisfaction with this laboratory exercise?

The results of all four questionnaires are graphically represented using a chart in Figure 1. From the chart it

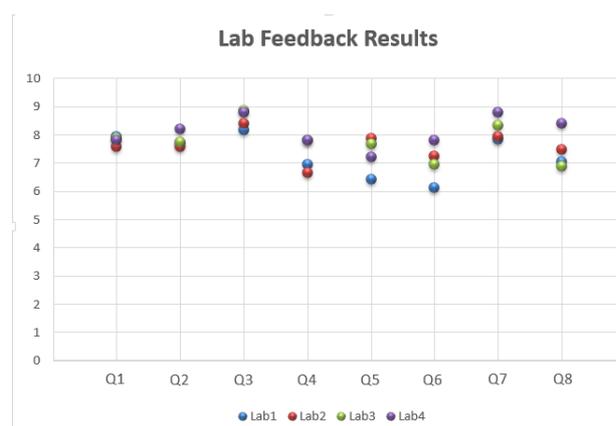


Figure 1 Lab Feedback results

can be seen that the overall impression about the exercises was very good (7 - 9 on the scale from 1 to 10) but the effort invested in solving them was also rather high (8 - 9 on the scale from 1 to 10).

The first two questions were about the satisfaction with the exercise material. The satisfaction was very good (average score ~ 8) and there is no significant difference between the four exercises. As for the effort invested in solving the exercises, the third and fourth exercise proved to be the hardest (average score ~ 8.5). This is not surprising since the matter covered by these exercises was the most complex. Despite the issues with download software and platform drivers, the overall impression with the working environment was rather good (~ 7), but it can be noticed that the fourth exercise, when the JTAG platform cable was used instead of E2LP download software, was rated the best. Finally, very high scores (~8.5) were obtained for the question about learning something valuable, with visible increase in score as the exercises progressed. This is expected since every exercise was an upgrade from the previous one. As mentioned earlier, the overall impression with the exercises was very good (~8), with the fourth exercise leaving the best impression.

The students had also the opportunity to give written comments for each question. The comments can be divided into three categories: positive aspects of the exercise (what students appreciated), negative aspects (critics) and recommendations for future development.

The most common positive comments in the questionnaires were:

- Satisfaction for being able to put theoretical knowledge to practice;
- Learning new skills in embedded system development which can be useful in the future (CV, employment);
- Good support from teaching staff.

The most common negative comments were:

- Platform malfunction (faulty drivers and download application software) happens several times during one exercise which causes frustration and delay.

- Exercises being too difficult/not enough time;
- Lack of prerequisite knowledge;
- Trouble dealing with extensive documentation (other than basic platform manuals);
- Xilinx environment being non-intuitive.

Finally, suggestions for future improvements can be summarized to:

- Fix driver and download application issues;
- Make platform software portable to Linux;
- Learn more about debugging using test benches and simulation.

The feedback results show that the introduction of the E2LP platform was accepted as a welcome change from the regular curriculum even if it required investing more effort in solving the exercises. The students were eager to learn something new and they saw working with this platform as an opportunity to gain valuable new skills in embedded software design, which might help them in their future studies and career development. They were aware that this platform is still in the very early stage of development and production, and thus showed a lot of understanding for the hardware and software issues encountered while working with the platform.

V. CONCLUSION

The results of pilot introduction of E2LP platform to course Laboratory of Computer Engineering 2 show that the students were able to cope with the given platform and a bit more advanced tasks than usual. The students were aware that they were learning something new and gaining valuable practical skills. They especially appreciated the feeling that they were doing something very close to the “real life” problems and thus willing to invest more effort.

The adoption of this new platform required additional effort from both students and teachers. While students were required to learn many embedded systems design concepts they haven’t encountered in their previous study, or have learned in theory but never put to practice, the teachers had to design new exercises to reconcile the students background knowledge, study programme and platform capabilities.

During the use of platform throughout the semester some issue regarding the platform itself, drivers and application software were observed, but both teachers and students understood it as “infant illnesses” related to early production stage of the platform. In the end, the pilot introduction of E2LP platform to this course can be regarded as successful from both subjective impression given by students during the course and the overall results in the form of students achievement and feedback. It must also be noted that the students have been very cooperative in giving their feedback as long as they felt it was going to make a difference.

To make this platform even better several improvements can be suggested. First of all, enabling remote access to this platform and all of its functionalities would eliminate

the need for students to come to the laboratory to do their exercises and thus save a lot of time and effort. Second, additional interfaces for connecting common types of simple peripherals (sensors, step-engines etc.) along with power supply ports of different voltages would be much appreciated. A possible suggestion for peripheral interface expansion would be to provide a proto-board which could use the existing mezzanine interface to connect to the main board. Finally, a public website with all the available materials related to this platform and some sort of means to report bugs or just ask for assistance would significantly improve the platform experience.

REFERENCES

- [1] Faculty of Electrical Engineering and Computing, “ECTS Information Package for Academic Year 2013/2014 – Course Catalogue, Graduate Study”, http://www.fer.unizg.hr/download/repository/2013_FER-EN-ECTS_Information_Package-Course_Catalogue-Graduate%5B4%5D.pdf, 2014.
- [2] G. MacBride, L. Hayward, G. Hayward, E. Spencer, E. Ekevall, J. Magill, A. C. : “Engineering the Future: Embedding Engineering Permanently Across the School-University Interface”, *IEEE Trans. Educ.*, vol. 53, no. 1, pp. 120-127, 2010.
- [3] H. Mahmoodi, A. Goel, C. Chen, H. Jiang, W. Pong, H. Shanasser, A. Montoya, J. Du Franco, C. Rodriguez, J. Carrillo, A. Enriquez, “Hands-on teaching of embedded systems design using FPGA-Based tPad development kit”, *Proceedings of the “2d Interdisciplinary Engineering Design Education Conference – IEDEC 2012”*, Santa Clara, California (USA), March 19, 2012, pp.1-6, ISBN: 978-1-4673-0840-3
- [4] V. Subbian, V., C. Purdy. “Redesigning an Advanced Embedded Systems Course: A Step towards Interdisciplinary Engineering Education.” *IEEE Integrated STEM Education Conference*, Princeton, NJ, March 9, 2013.
- [5] Zilic, Z.; Karajica, B. "Training for inter-disciplinary multi-sensor system design at all levels", *Interdisciplinary Engineering Design Education Conference (IEDEC)*, 2012 2nd, On page(s): 7 - 9
- [6] E2LP Consortium, “E2LP Base Board – Getting Started Manual”, <http://www.fer.unizg.hr/predmet/Iri2>, 2014.
- [7] E2LP Consortium, “E2LP Base Board – Technical Reference Manual”, <http://www.fer.unizg.hr/predmet/Iri2>, 2014.
- [8] Xilinx, “ISE Design Suite”, <http://www.xilinx.com/products/design-tools/ise-design-suite/index.htm>, 2014.
- [9] Xilinx, “ISE In-Depth Tutorial”, http://www.xilinx.com/support/documentation/sw_manuals/xilinx12_4/ise_tutorial_ug695.pdf, 2014.
- [10] Xilinx, “EDK Concepts, Tools and Techniques”, http://www.xilinx.com/support/documentation/sw_manuals/xilinx14_1/edk_ctt.pdf, 2014.
- [11] Xilinx, “Platform Studio and the Embedded Development Kit (EDK)”, <http://www.xilinx.com/tools/platform.htm>, 2014.
- [12] Xilinx, “MicroBlaze Soft Processor Core”, <http://www.xilinx.com/tools/microblaze.htm>, 2014.
- [13] Xilinx, “AXI Interconnect”, http://www.xilinx.com/products/intellectual-property/axi_interconnect.htm, 2014.
- [14] Xilinx, “OS and Libraries Document Collection”, http://www.xilinx.com/support/documentation/sw_manuals/xilinx14_5/oslib_rm.pdf, 2014.
- [15] Xilinx, “Using EDK to Run Xilkernel on a MicroBlaze Processor”, http://www.xilinx.com/support/documentation/sw_manuals/xilinx13_4/ug758.pdf, 2014.

FPGA Verification Module

Željko Hocenski, Ivan Alekski
 University of Osijek, Faculty of Electrical Engineering
 Zeljko.Hocenski@etfos.hr, Ivan.Alekski@etfos.hr

Abstract: This paper addresses verification and debugging tool for development of FPGA modules. Proposed tool is developed for educational purposes in teaching students on Digital Design and VHDL programming language. Main goal of the debugging module is to get/set signal values while the FPGA board is running the module of interest. Two PicoBlaze CPUs are used in order to synchronize the input and output signals between PC and the FPGA. Debugging and verification tool is wrapped around the testing module, and it occupies 14% of the Spartan 3 XC3S200 FPGA device. While using proposed tool, students are getting the knowledge about the PicoBlaze CPU, assembly language, FPGA, VHDL. When using proposed tool, students get deeper understanding of the hardware-software co-design concept. Finally, individual tasks are assigned to student workgroups. Some typical tasks are illustrated in this paper.

Keywords: FPGA, PicoBlaze CPU, Verification, Debugging.

I. INTRODUCTION

FPGA debugging module is proposed in this work. HDL modules debugging is a critical stage in the development of FPGA modules [1]. Verification and validation takes 50% of the product's cost, and it takes 40% or more of the overall design cycle [2]. In addition to shorten design cycle for students on their HDL FPGA project, this paper proposes a debugging platform for FPGA module in the development

phase, as it was done in [3]. Proposed platform extends the number of the input and the output ports by using one or more input or output buffers. The platform consists of the FPGA-PC communication and a PC-based graphical user interface (GUI). In this way, students can have an insight to the signals of FPGA module in the development phase. Signal values are placed in buffers on the FPGA and on the PC's GUI. Each buffer can be set as an input or an output. Buffers values on the FPGA and their corresponding copy on the PC are synchronized by the use of two PicoBlaze CPUs and UART communication module {[4],[5]}, c.f. Fig. 1. Usually, development kits have a limited number of I/O devices: buttons, switches, LEDs, etc. On the other hand, proposed debugging tool enables arbitrary large number of I/O devices.

II. RELATED WORK

Similar work is done in [6] where authors presented their Reconfigurable Virtual Instrumentation (RVI). By the cross-university collaboration virtual devices are developed and shared. In this way the RVI becomes an evolvable low-cost hardware-software co-design (HW/SW) educational platform. This platform saves money since it can replace some of the expensive measurement instrumentation, for example: oscilloscope, logic analyzer, etc. In contrast to their grooving variations of virtual devices, we are proposing simple and

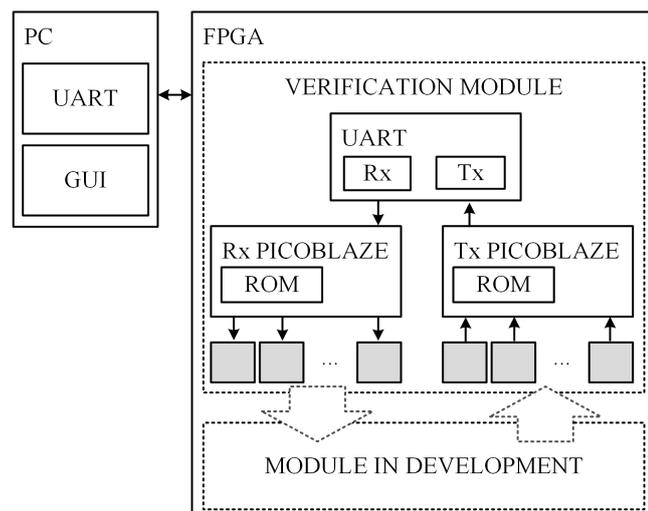


Fig. 1. Virtual devices platform (VDP) with input and output virtual devices (VD)s for functional verification phase of the FPGA modules development.

small VDP that is oriented towards verification and validation of HDL modules via the FPGA-PC interface.

Simple FPGA e-Lab is presented in [7]. By using Windows XP Remote Desktop Connection students can remotely connect to a laboratory PC, which is connected with the Spartan 3-E FPGA device. User interface is done over a GUI that is made with National Instrument's LabView software. Authors made control hardware surrounding the FPGA kit. They also connected a web camera for visual feedback, so one can always see if his experiment is running or not. Students are able to upload final project done at home and implement the bit file into the FPGA device. More sophisticated remote learning FPGA facility is presented in [8]. As an approach it is cost effective when compared with classical non-remote laboratories. Authors considered a cluster of FPGA devices that students can use remotely. Students liked that they can connect later and try again if their project didn't work from the first try. However, {[7],[8]} do not offer system verification and validation, as it is presented in this work. Our device is done in MS Visual Basic and does not require the expensive software tools.

According to [9], FPGAs are the key components for making possible to finish complex projects in a one year period. Students were assigned a task to build mobile robot projects. Before taking the FPGA into education courses, students were focused only on a software design of bought professional robotics equipment. The author concluded that undergraduate students are able to construct competitive robots with the use of the FPGA and a basic robot parts.

The authors in [10] prefer teaching real-time digital signal processing by using FPGAs. They believe that the FPGA technology provides more flexibility when compared with the DSP or the MATLAB. The FPGAs are capable for implementing the microprocessor cores, what exclude the need for a certain DSP that is programmable in an embedded C. With the use of Hardware/Software co-design, the FPGAs allow algorithm improvements by using its hardware and/or even parallel implementation. In reference [11], the same author presented the FPGA-based system on programmable chip (SoPC) development platform with NIOS 32-bit RISC soft processor and the μ Clinux operating system (OS). Students had to implement a VGS and a SRAM interfaces with the SoPC. On the software part of the design, students had to develop routines that control a custom graphical hardware. When combining those two components, the students were exposed to hardware/software co-design, operating systems, reconfigurable hardware and system design concept. The authors concluded that the use of the FPGA-based SoPC in student courses enabled the development of more involved senior design projects.

Authors in [12] presented their work through three consecutive courses in one academic year. Their 1st course deals with logic design using Verilog, followed by 2nd course that deals with logic synthesys and system on chip design, while 3rd course deals with timing and testing of digital design. The topics covered with proposed courses are: digital electronic, Boolean algebra, Karnaugh maps, finite state machine, Verilog HDL, 8-bit RISC SoPC, HW/SW co-design,

testing of digital systems, test economics, fault modeling and simulation, sequential test methods, and design-for-test techniques. Since none of the topics deals with the Digital Design Verification, the authors announced a new course with that topic.

Distance laboratory access to the server with dedicated FPGA hardware is presented in [13]. The mainstream of this concept is that students can register, login and reserve a term in order to access the hardware resources. When their term is active, they can send their CAD design and the server's software can implement it on the FPGA. After the execution, the student receives an e-mail with the results. This design concept saves a lot of time to the professors and it enables an individual self learning process.

A multimedia tool for teaching reconfigurable computing is presented in [14]. In order to get familiar with the VHDL and FPGA, in the first few weeks, students had to implement a few medium complexity VHDL modules: VGA, PS/2, LCD, calculator and a simple processor. Their knowledge about computer architecture is used to speed up the learning process about the reconfigurable computing. Behavioral VHDL model of a MIPS processor is used in order to instantiate a MIPS microprocessor with some parts missing. Students have to recover the missing VHDL parts (ALU, register file, etc.) and test the functionality of their MIPS using the iCmips simulator. Simulator provides a full view into the registers, data memory, while it executes all of the commands on the real hardware.

Remote laboratory for HW/SW co-design that is proposed in [15] accepts a student's HW and SW design files, implements the HDL file on the real FPGA and executes the software test bench file. As the result, the output file is then sent back to the student. The task was to implement a simple 16-bit RISC processor with Harvard architecture, different memories for machine code and storage data and four general purpose register. The CPU design had to execute nine basic instructions: load, store, move, add, sub, compare, halt, conditional and unconditional jumps.

III. FPGA-BASED HARDWARE/SOFTWARE CO-DESIGN

Common system design approach in embedded systems is a Hardware/Software co-design concept [2]. FPGA devices are capable to synthesize such design concept since they possess a large number of programmable logic cells, which can be configured as combinatorial and sequential, respectively. Hardware/Software co-design commonly consists of programmable devices, microcontrollers, microprocessors or signal processors, which have a memory for a software part of the design, c.f. Fig. 1.

Proposed hardware design uses 14% of the Xilinx's Spartan-3 XC3S200 FPGA device, c.f. Table I. It contains of two of the PicoBlaze CPU's, where each occupies 5%. Remaining 4% are used by UART and some signal interchange. Minimum period of implemented logic is 7.922ns, and the maximum operating frequency of VDP is 126.239(MHz). Proposed design consumes 48(mW). Utilization summary for Xilinx's 3s200ft256-5 device is illustrated in Table I.

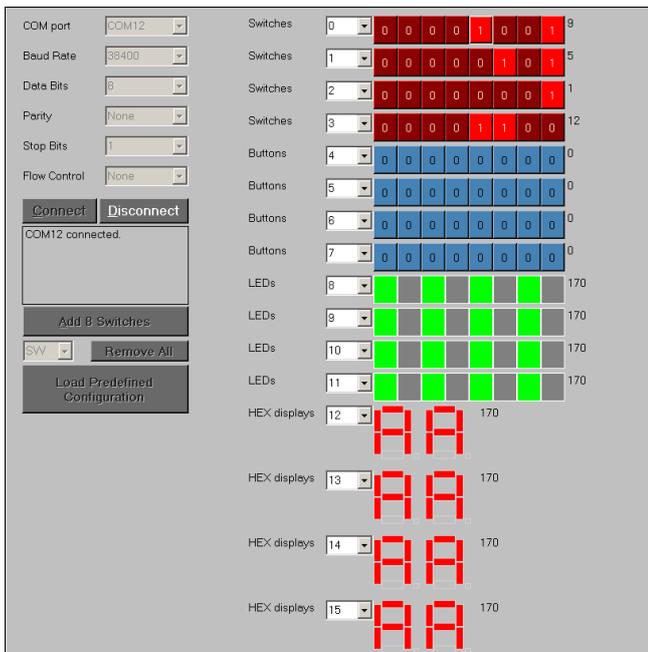


Fig. 2. PC-based graphical user interface for validation and verification phase of FPGA modules development.

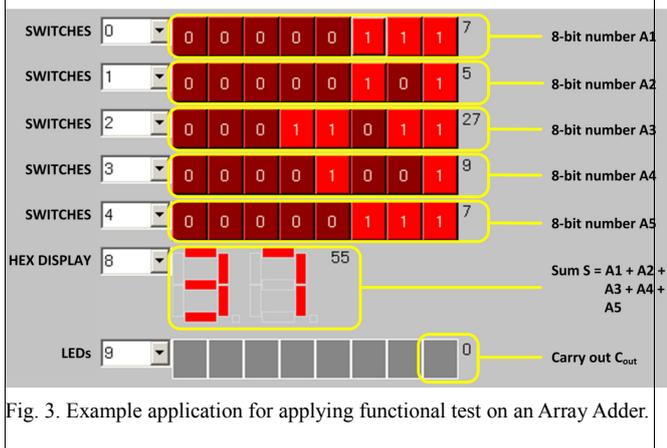


Fig. 3. Example application for applying functional test on an Array Adder.

TABLE I. UTILIZATION SUMMARY FOR XILINX 3S200FT256-5.

Resource	Number of resources	[%]
Number of Slices	282 out of 1920	14 %
Number of Slice Flip Flops	323 out of 3840	8 %
Number of 4 input LUTs	521 out of 3840	13 %

Verification is done with an arbitrary large set of input and output buffers that enables a real-time insight into internal FPGA signals. Each buffer is 8-bit long with assigned value and unique address. Buffers exist on FPGA and have corresponding copy on the PC. FPGA and PC buffer values are synchronized by using two of the PicoBlaze CPU's which communicate with the UART-based full-duplex RS232 communication protocol @ 38400 baud rate. One PicoBlaze CPU is used for data transmission, while the other one is used for data reception. Software algorithms for buffer array synchronization are described with Alg. 1 and Alg. 2.

Algorithm 1. Reception PicoBlaze (PC to FPGA).

1. $byte \leftarrow GetByte()$
2. Set $data \leftarrow byte$
3. $byte \leftarrow GetByte()$
4. Set $address \leftarrow byte$
5. Set $BufferArray[address] \leftarrow data$
6. Go to Line 1

Algorithm 2. Transmission PicoBlaze (FPGA to PC).

1. **For each** $address = 8$ to $address = 16$ with step +1
2. Set $data \leftarrow BufferArray[address]$
3. SendByte($data$)
4. SendByte($address$)
5. Go to Line 1

Buffer values on the PC are displayed with the GUI, c.f. Fig. 2. PC-based GUI is implemented in the Visual Basic programming language on a standard PC. It represents the software part of the design that communicates with the FPGA over the RS2323 serial port. USB to RS232 adapter is used in this work. The GUI is made by using dynamical programming techniques with the following features.

- Management of serial port communication;
- Dynamic buffer creation and deletion;
- Selection of arbitrary address for a certain buffer;
- Data transmission from the PC to FPGA's input buffers;
- Data reception from FPGA's output buffers to the PC;
- Displaying buffer values on the screen.

Proposed debugging and verification platform is adaptive since it can dynamically create/delete buffers with arbitrary address and buffer's type: switches, buttons, LED and HEX displays, c.f. Fig. 2.

IV. EXPERIMENTAL RESULTS

In this section, one Array Adder is considered as an FPGA module in the development phase. In this work we used the array adder with 5 8-bit inputs, one 8-bit output and one bit output, c.f. Fig. 3. Array adder has a task to sum 5 8-bit numbers and provide a result as an 8-bit sum with 1-bit carry out. Firstly, the VHDL code was successfully tested with simulations. Subsequently, proposed debugging module was used to test the design's functionality. After successful synthesis, students are very happy to be able to see their design doing required task on an FPGA board.

V. CONCLUSION

Proposed debugging platform is the simple device that helps students to accomplish their final project successfully. It is easy to use and will have a lot of applications on future testing of student projects. It provides the student with extended I/O devices via FPGA-PC interface and makes easier to check does the student's product meet the requirements of a given project by using proposed GUI. Proposed approach is quite challenging for students since it includes

multidisciplinary tasks: hardware design, embedded assembly software and verification process. Therefore it can be applied in the final years of their study.

REFERENCES

- [1] M. Pezzé and M. Young, "Software Testing and analysis: Process, Principles, and Techniques," Wiley, ISBN 13: 978-0-471-45593-6, USA, 2007.
- [2] R.C. Cofer, B.F. Harding, "Rapid System Prototyping with FPGAs," Newnes, ISBN: 0750678667, September, 2005.
- [3] Ž. Hocenski, I. Aleksi, V. Sruk, "Adaptive Virtual Devices Platform for Verification of FPGA Modules in Student Courses on Digital Design," 7th IEEE International Conference on e-Learning in Industrial Electronics (ICELIE), pp. 22-27, ISBN: 978-1-4799-3180-4, Vienna, Austria, 10-13.11.2013.
- [4] Xilinx Inc., "PicoBlaze 8-bit Embedded Microcontroller User Guide for Spartan-3, Spartan-6, Virtex-5, and Virtex-6 FPGAs," UG129 (v2.0), January 28, 2010.
- [5] Pong P. Chu, "FPGA Prototyping VHDL Examples, Xilinx Spartan-3 Version", Wiley-Interscience, ISBN: 978-0470185315, February, 2008.
- [6] A. Cicuttin, M.L. Crespo, A. Shapiro, N. Abdallah, "Building an Evolvable Low-Cost HW/SW Educational Platform--Application to Virtual Instrumentation," IEEE International Conference on Microelectronic Systems Education (MSE'07), pp.77-78, 2007.
- [7] R. Hashemian, J. Riddley, "FPGA e-Lab, a Technique to Remote Access a Laboratory to Design and Test," IEEE International Conference on Microelectronic Systems Education (MSE'07), pp.139-140, 2007.
- [8] Y. Rajasekhar, W.V. Kritikos, A.G. Schmidt, R. Sass, "Teaching FPGA system design via a remote laboratory facility," International Conference on Field Programmable Logic and Applications, DOI: 10.1109/FPL.2008.4630040, pp.687-690, September 2008.
- [9] M.A. Soderstrand, "Role of FPGAs in undergraduate project courses," MSE, pp.0109, 1997 International Conference on Microelectronics Systems Education (MSE '97), 1997.
- [10] T.S. Hall, D.V. Anderson, "A Framework for Teaching Real-Time Digital Signal Processing With Field-Programmable Gate Arrays," IEEE Transactions on Education, vol. 48, no. 3, pp. 551-558, August 2005.
- [11] T.S. Hall, J.O. Hamblen, "Using an FPGA Processor Core and Embedded Linux for Senior Design Projects," MSE '07 Proceedings of the 2007. IEEE International Conference on Microelectronic Systems Education.
- [12] J.D. Lynch, D. Hammerstrom, R. Kravitz, "A cohesive FPGA-based system-on-chip design curriculum," IEEE International Conference on Microelectronic Systems Education, 2005. (MSE '05), DOI: 10.1109/MSE.2005.5, ISBN: 0-7695-2374-9, pp. 17-18, June 2005.
- [13] R. Seinauskas, "A distance laboratory for computer-aided design", IEEE International Conference on Microelectronic Systems Education, 1997., ISBN: 0-8186-7996-4, pp. 107 - 108, Jul 1997.
- [14] I. Skliarova, "A Multimedia Tool for Teaching Reconfigurable Computing", Computer and Electrical Engineering, 2009., ICCEE '09, vol.1, pp.204-208, 28-30, Dec. 2009.
- [15] J.S. Pastor, I. Gonzalez, J.Lopez; F.Gomez-Arribas, J.Martinez, "A remote laboratory for debugging FPGA-based microprocessor prototypes," IEEE International Conference on Advanced Learning Technologies, pp. 86-90, DOI: 10.1109/ICALT.2004.1357380, September 2004.

Universities: A View to Pebbles of Mosaic

Gordana Velikic, Milenko Beric,
 Nikola Teslic
 RT-RK Institute for Computer Based
 Systems
 Novi Sad, Serbia

Ivan Kastelan
 University of Novi Sad
 Faculty of Technical Sciences
 Novi Sad, Serbia

Mark Bocko
 University of Rochester
 Rochester, NY, USA

□ *Abstract*— Educational systems vary widely between continents, countries, and even towns. In the United States, universities may be either publicly or privately supported (or a combination) and may be categorized as either research oriented or teaching oriented. In this paper we present an informal discussion of the relative advantages and disadvantages of the various university models from the perspective of both students and faculty.

I. INTRODUCTION

TO keep the edge of competitiveness, Universities are forced to go global, become more diverse, and to nurture the spirit of entrepreneurship. In the rise of knowledge economy, where knowledge replaces physical resources as the main driver of economic growth, the traditional roles of Universities are changing [1]-[8], Fig. 1.

New technologies introduce new concepts, and an illusion that a need for a traditional classroom and fulltime faculty will not be necessary. However, since education is not the facts transmission, but rather about learning to argue and reason, and adding dimensions with a “human touch”, common sense suggests that community of scholars will outlast the changes. On the other side – new technologies improve visibility of Universities. For example, offering courses on Coursera [9] enabled Universities to reach individuals across the globe who would not have otherwise had an opportunity to engage with them.

Ranking of Universities has become increasingly important; mainly because it is envisioned to help prospective students simplify complex decisions and to prevent Universities to stagnant. Although ranking is important in pursuit of higher quality, ranking systems are criticized to provide an incomplete picture and that many important aspects are left aside or not covered with appropriate merits. Nevertheless, the best ranked Universities offer highly competitive environment, fast adaptation to changes, and attract individuals with a “drive” for success (Table I and II).

A perfect methodology to rank Universities does not exist, and each ranking system puts weight on a different main

focus. For example, the “Shanghai rankings” mainly focus to pure research [10], QS [11] to academic reputation, and Times Higher Education (THE), powered by Thomson Reuters, to teaching.

A membership to world-class University is not self proclaimed. It requires a long-term vision and dedication, and once there a strategies to keep a place in the club [12]. In the paper we present an informal discussion of fundamental power intrinsic to a nature of higher educational environment.

TABLE I.
 TOP 5 PRIVATE US UNIVERSITIES IN THE QS WORLD UNIVERSITY RANKINGS

University	World Ranking 2013/2014	World Ranking 2008/2009
Massachusetts Institute of Technology (MIT)	1	9
Harvard University	2	1
Stanford University	7	17
Yale University	8	2
University of Chicago	9	8

TABLE II.
 TOP 5 PUBLIC US UNIVERSITIES IN THE QS WORLD UNIVERSITY RANKINGS

University	World Ranking 2013/2014	World Ranking 2008/2009
University of Michigan	22	18
University of California, Berkley	25	36
University of Wisconsin-Madison	37	55
University of California, Los Angeles (UCLA)	40	30
University of North Carolina, Chapel Hill	54	102

The research leading to these results has received funding from the European Union’s Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 317882.

II. FRAGMENTS OF MOSAIC

A. Categorization

One can count more than three thousand Universities in the United States, with 600 major fields of study. Universities offer a wide range of undergraduate and graduate degrees, and serve as centers of research and scholarships. Public Universities are normally supported from state funding where as private Universities have endowment – a savings account built up from private donations and tuitions payments. Private Universities apply for government funding for research, and students at private Universities often receive government scholarships. Generally, the tuition in private Universities is higher than it is at public Universities. However, private universities offer financial aid for those students who excel academically regardless the financial background.

The selectivity of students varies widely among Universities. Some of the finest Universities in the country are public (Table II), such as University of California Berkeley, and there are many fine private Universities such as Harvard, Princeton, or University of Rochester (Figure 1), also Table I. There is spectrum of sizes in both public and private Universities. In general, the very largest Universities in the country are public, with as many as 50.000 students, but there are many public and private Universities with only a few thousand students. Most of the smallest Universities are private. The class size within Universities differs accordingly. In some very large Universities a class can have more than 100 students which forces lectures to be conducted in traditional lecture style. In smaller Universities class size averages around 20 students, which gives an opportunity for students to take an active role in discussions with professors.

Universities may be further categorized as research or teaching oriented. Research-oriented Universities provide the facilities for faculty and students, graduate as well as undergraduate, to conduct basic and applied research. Although teaching remains an important aspect in research Universities, a greater percent of faculty time is devoted to research in such Universities.



Fig. 1. Logo of the University of Rochester, one of the finest private, research oriented Universities in the USA. The University is an outstanding example of the academia-industry collaboration with return in investment ratio of 30:1, i.e. thirty million dollars in investment returns per every million dollars [8].

Teaching oriented Universities emphasize education. The majority of such Universities are relatively small. Such Universities, despite their small size, offer a richness of courses and often have modern facilities to support classes. However, due to limited facilities, research typically is not a priority in such institutions and most do not offer graduate degrees at all, or possibly only master degree.

B. "Inside" the academic year

The academic year may be divided into three semesters: fall, spring and summer, or four quarters. One semester is 14 weeks in length and courses extends throughout the semester. Students normally take 4 or 5 courses each semester.

Most courses meet 2 or 3 times a week for a total of 3 hours. If lab work accompanies the course, it requires additional 2 hours per week. Instructor office hours are fixed during the week, but additional can be arranged by appointment. For most courses, students are given around 10 homework assignments, and two exams (midterm and final). To pass a course a student must meet the course requirements during the semester. There is not examination period after the course. Those students who fail may repeat the course next time it is offered. That way student is forced to leave no gaps in their course progress, or to postpone. Professors have to help students to understand and pass a course and the quality of professors' work is assessed with standardized forms filled in at the end of each course by the students. Comments from such forms provide useful feedback to the professors and university's administration to enable improvements in teaching.

Teaching assistants are graduate students paid from university-based funding (teaching, research or lab assistantships, tuition and fee waivers and stipends), and distributed according to the needs of specific courses to conduct recitations, assist in labs and hold office hours.

Fundamental courses, in the major disciplines, are regularly offered, and curriculum doesn't differ much between universities. Specialized topics courses may not be offered every year, and some courses may be specific for individual universities.

Course work is measured in credit hours. A course typically accounts for 3-4 credit hours, and about 120-130 credit hours are required for graduation. Most Universities require students to take courses in liberal arts (humanities, social and natural sciences), before settling on a specific field of study – a major, usually completed in the final two years.

Graduate students in a PhD program take courses and seminars that are designed to give a comprehensive knowledge of their chosen field. PhD research is the student's original work and advisor (a faculty member) monitors and guides the process, although the student carries out the research independently. Some graduate programs include a research master's degree programs leading to Master of Science and Master of Art for the master degree. Professional master's degree programs provide specific skills for particular profession and lead directly to

employment; examples are Master of Business Administration or Education.

C. What makes a good University good?

University quality is not tied to the public/private or research/teaching categorization. Although many of the highest ranked Universities are private, there are many public Universities that have earned that position among the elite. Most of the universities are distinguished by their specific programs. Factors that determine Universities ranking include: excellence in research, faculty members held in high professional regard, high quality courses, large palette of courses for different programs, well equipped labs and classrooms, extensive libraries, and faculty who encourage undergraduate and graduate students to take an active role in learning and research.

In Universities with highly ranked programs in selected areas, there is often a positive influence on related programs.

Throughout their studies, undergraduate students are encouraged to participate in research, to implement their knowledge, to explore and to write papers, as much as their time permits. This experience will prepare them for academic or industry career after graduation.

A drawback at some schools is the practice of allowing graduate students to conduct complete courses, which may lower the quality of lecturing due to lack of experience and a heavy workload of graduate students. In high quality programs, graduate students may offer a few lectures besides recitation, under supervision of professor as a part of student's preparation for academic career.

In teaching oriented universities, public or private, professors must have an ability to relay knowledge and enthusiasm. In research-oriented Universities, faculty must combine excellence and enthusiasm for teaching with ability to conduct cutting edge research. In either settings faculty must show an initiative and be dedicated to their work.

III. CONCLUSION

Periods of transition are always difficult. Learning by trial and error results when experience of other is ignored. Educational systems are built from people. No matter which model is adopted, people devoted to their work, having expertise, good equipment, well-constructed courses and attractive research programs that "draw" grants will result in success, which will attract high quality students and faculty to such universities.

REFERENCES

- [1] C. Symes: "Selling futures: A new image for Australian universities?", *Studies in Higher Education*, 1996, Vol. 21, Issue 2, pp. 133-147
- [2] K. Chen, M. Kenney: "Universities/Research Institutes and Regional Innovation Systems: The Cases of Beijing and Shenzhen", *World Development*, 2007, Vol. 35, Issue 6, pp. 1056-1074
- [3] J.A. Hansen, M. Lehmann: "Agents of change: universities as development hubs", *Journal of Cleaner Production*, 2006, Vol. 14, Issues 9-11, pp. 820-829
- [4] R.K. Lester: *Universities, Innovation, and the Competitiveness of Local Economies*, MIT Industrial Performance Center, 2005
- [5] U. Teichler: "The changing roles of the university and non-university sectors of higher education in Europe", *European Review*, 1998, Vol. 6, Issue 4, pp. 475-487
- [6] J.A. Mathews, M-C. Hu: "Enhancing the Role of Universities in Building National Innovative Capacity in Asia: The Case of Taiwan", *World Development*, 2007, Vol. 35, Issue 6, pp. 1005-1020
- [7] P. Jarvis: "Global Trends in Lifelong Learning and the Response of the Universities", 1999, Vol. 35, Issue 2, pp. 249-257
- [8] G. Velikic, P. Bjelic, Innovate and prosper: building the prestige of the greater Rochester NY region, *Actual Problems of Economics*, Sep2013, Vol. 147 Issue 9, p264
- [9] Coursera, <http://www.coursera.org>
- [10] K. K. Ho: "Research output among the three faculties of business, education, humanities & social sciences in six Hong Kong universities", *Higher Education*, 1998, Vol. 36, pp. 195-208
- [11] QS University rankings, <http://www.topuniversities.com/university-rankings>
- [12] J. Salmi: *The Challenge of Establishing World-Class Universities*, The World Bank, 2009

An Approach to Improvement of Students Learning of Computer System Engineering

Milos Nikolic
RT-RK Institute for Computer Based Systems
Novi Sad, Serbia
e-mail: milos.nikolic@rt-rk.com

Dusan Majstorovic, Ivan Kastelan, Nebojsa Pjevalica
University of Novi Sad
Faculty of Technical Sciences
Novi Sad, Serbia

□
Abstract—This paper presents an approach to the study programs applied for the subject Logic Design of Computer Systems II. Study program of this subject is based on unified embedded engineering learning platform, E2LP. E2LP intended to cover complete process of embedded systems learning, considering modular approaches for unifying education in relevant embedded system technologies. Study approach in this subject intends to be a step forward in helping students to adopt knowledge better and to have stronger understanding of the complex and multidisciplinary approach in this course. This implies splitting course in two parts, one in which the students are introduced to the environment and taught how to use it, and second in which students are fulfilling the specific task with minor assistant help.

I. INTRODUCTION

THIS paper presents an approach to the study program for the subject Logic Design of Computer Systems II. We discuss the rationale for new approach, first results and effects of it. We are presenting also the problems that we have faced and the student opinions about this approach.

Laboratory work and practical examples are very significant in electrical engineering education. This approach is forcing students to actively participate in learning process instead of being a passive listener [1]. Embedded system engineering is becoming more wanted profession since the computer based embedded systems have more important role in computer based embedded systems [2]. For these and many other reasons many technical faculties have put more emphasis on embedded systems learning by introducing a number of active learning laboratory-based courses [1].

Efficiency of laboratory work in embedded systems learning usually suffers from introduced overhead in both time and effort needed to get students familiar with hardware platforms and software tools for each course. Because of that, it is sometimes quite difficult to steer the student's attention from tools to fundamental principles.

Based on these observations, and building on comprehensive industrial experience in platform-based

engineering, a modular common design platform has been developed. The aim of the platform is to increase efficiency of laboratory-based courses. The platform is designed to cover all aspects of embedded systems learning: (1) Digital system design, (2) Computer system design, (3) Digital signal processing of audio, video and data signal streams in real-time, (4) Computer networks and interfaces and (5) System integration.

Approach based on laboratory exercises is presented during several semesters in courses in a computer engineering undergraduate curriculum in University of Novi Sad and it also applies to Logic Design of Computer Systems II. As one of subjects which are teaching students embedded system design, it was a perfect candidate to join the collaboration between scientific institutions gathered around unified platform on a European level, E2LP [3].

The program of the subject is such that it can easily be adapted for the new platform, with minor changes in the program. Transition to new platform has been recognized as a good moment for structural changes of the course, and instead of only laboratory exercises a small tutorial sub-course was introduced.

Course is following the previous semester course Logic Design of Computer Systems I which is also based on embedded engineering learning platform E2LP [4]-[6]. It covers the topics of computer architecture and organization and software hardware interfaces. Main emphasis of the lab part is on methods and tools for hardware/software system-on-chip co-design. Laboratory work is project oriented with a set of introductory lab exercises designed to prepare students for project challenges. Students are organized in small project teams (2-4). Project definitions are given in a way to leave a lot of room for various design decisions and implementation methods motivating the students to take the lead.

II. E2LP UNIFIED BOARD CONCEPT

E2LP is supporting the following learning objectives:

1. embedded microprocessors & computer architectures programming (software aspects),

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 317882.

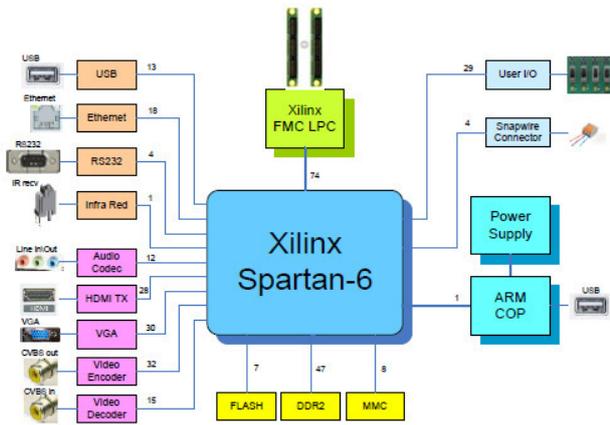


Fig. 1. Architecture of the E2LP base board

2. digital signal processing (audio, video and data) and its real-time implementation,
3. FPGA digital system design and verification,
4. FPGA accelerated computing,
5. networks & interfaces,
6. system integration.

E2LP platform consists of the Base Board which is presented in Fig. 1, and a set of extension boards which are not relevant for this course.

III. ARCHITECTURE OF FPGA DESIGN

Using Xilinx Platform Studio [7] (XPS), the advanced embedded system can be created relatively easy, following the steps from the course documentation. Basic architecture of such system is shown in Fig. 2.

MicroBlaze processor presents a predefined entity in the embedded system. In XPS environment, user can use predefined peripherals available within the XPS, or they can define and implement custom peripheral devices and connect them to the MicroBlaze using XPS tools. Such procedure can be learned very quickly and provides a perfect base for getting introduced into the embedded systems.

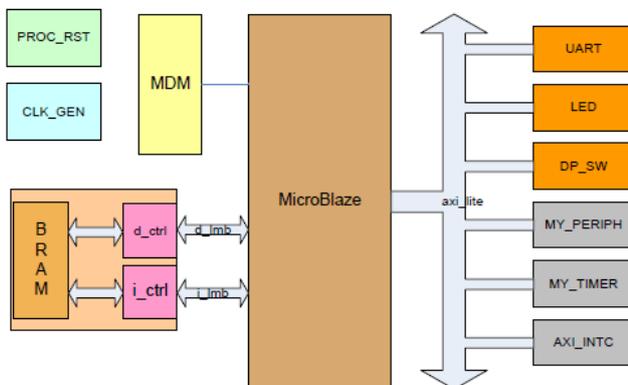


Fig. 2. Simple Xilinx Platform Studio design

IV. LOGIC DESIGN OF COMPUTER SYSTEMS II – COURSE OVERVIEW

Logic design of computer systems II is aiming to teach students fundamental topics in digital systems:

- Hardware design languages (VHDL)
- Properties of real electronic circuits
- Design of multi-clock systems
- Principles of complex digital system design

Prerequisites for Logic design of computer systems II in general should be basic knowledge of digital electronics and design and VHDL.

Introduction to this course includes basic knowledge of digital electronics and digital design, which is covered with subject Logic Design of Computer Systems I. In Logic Design of Computer Systems I course, students were introduced to VHDL, Xilinx ISE [7] tool chain, E2LP Board with its application and have used basic user I/O (LED, switches, buttons, LCD, etc.) [4]. In this course a step forward should be made, towards the complex systems.

A. Exercises

Program of the Logic Design of Computer Systems II is divided into 6 exercises:

1. Clock management techniques, involves command line tool chain tutorial and fundamentals of clock management and distribution, introduction to digital clock manager units and timing constraints. Examples include counters, FSMs and combinational logic
2. VGA interface, includes introduction to VGA interface, text and graphic modes. Students design logic for dynamic display of simple graphical objects and text. Emphasis is on parameterized design.
3. XPS introduction includes introduction to Xilinx XPS (Hardware/Software) tool chain and high level computer system design. Students are instructed to create complex MicroBlaze [8] based computer system with a number of standard XPS peripherals (PS2, UART, memory controllers, etc). Simple C program demonstrates system capabilities.
4. XPS video adapter (hardware part), includes introduction to concepts of memory mapping and bus handling. Students extend existing VGA module functionality and wrap it into an XPS peripheral. Text display is controller from simple MicroBlaze code.
5. XPS video adapter (software part), includes introduction to device driver concept. Students develop a number of low-level driver functions for video adapter peripheral and implement simple console application.
6. XPS advanced exercise, includes introduction to real-time (embedded) programming (interrupts etc) and illustration of system architecture impact on performance (DMA, data cache, bus organization, etc).

First two exercises are introduction and are not evaluated. Other four exercises are rewarded with 5 points, with 1 additional point for special results (quick or elegant solution of the problem). In total, exercises count as 20% of the overall grade in this course.

B. Tutorial sub-course

Tutorial sub-course is organized such that one or a small number of students work on a topic and meet with the instructor weekly for discussion and guidance. For tutorial course we have prepared next tutorials:

1. Factorial coprocessor,
2. Fibonacci coprocessor,
3. VGA stopwatch,
4. Parking machine,
5. Pong game,
6. Snake game,
7. Color VGA,
8. UART VGA terminal,
9. UART LCD terminal.

Depending on the complexity various tutorials were evaluated differently as shown in Table I

Students (23 of them in the Spring 2014 semester), are in position to form teams for each tutorial, with number of team members which is less or equal to number predicted for each tutorial. To have more freedom to be creative and to organize time better, students were allowed to access the laboratory and E2LP platform more often than twice a week, which is defined by the subject program. Weekly meetings with instructor were used for all questions and discussions.

TABLE I.
TUTORIAL COMPLEXITY
MAXIMUM NUMBER OF TEAM MEMBERS AND POINTS

Tutorial name	Team members	Points
Factorial coprocessor	3	40
Fibonacci coprocessor	3	40
VGA stopwatch	3	40
Parking machine	2	40
Pong game	4	45
Snake game	4	40
Color VGA	3	45
UART VGA terminal	3	40
UART LCD terminal	3	45

V. TUTORIAL RESULTS

Here we are presenting some examples of tutorial projects.

A. Stopwatch

For tutorial named VGA stopwatch (Fig. 3), the goal was to create system that was able to measure elapsed time from the moment when start button is pressed, with precision of 100ms. The result of measuring should be displayed on VGA screen.

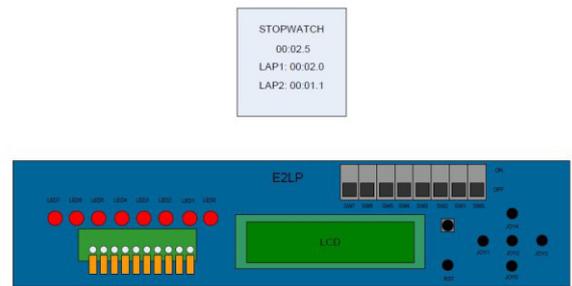


Fig. 3 VGA stopwatch

Besides this, the system should also display overall time and lap time. Lap time is determined with another push button. Third push button is for reset.

Peripheral device, implemented in VHDL, which is connected with MicroBlaze processor with internal AXI bus, is shown in Fig. 4.

This system is measuring time and generating interrupts based on time triggered events and push button states. These interrupts are handled by software part of the system which is generating information and sends it to the VGA screen again using custom made peripheral (Fig. 5).

This solution is successful implementation of all required tasks listed in tutorial documentation. This tutorial was realized by the group of three students, where they did the planning, task scheduling and implementation alone.

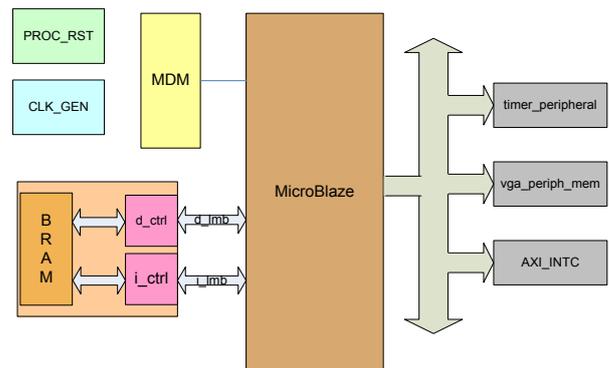


Fig. 4. VGA stopwatch peripheral device

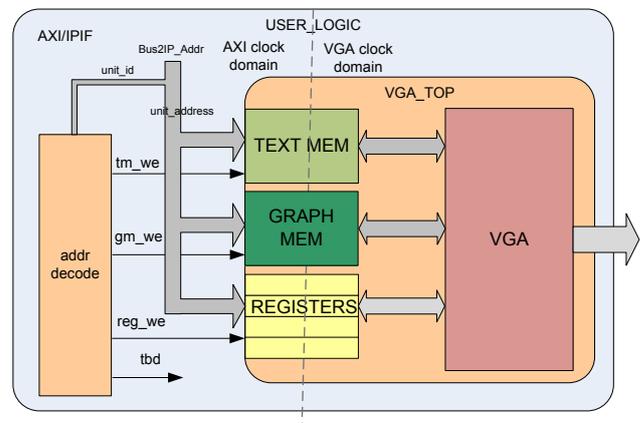


Fig 4 Block diagram of VGA peripheral

B. Pong Game

In this tutorial the goal was to create a system that provides playing Pong game for two players. The game is displayed on VGA screen. It should support reception of commands from UART terminal for player 1, command reception from push buttons for player 2, moving of the rackets and ball on the screen and racket-ball collision detection. Block diagram of the system is shown in Fig. 6 and the VGA screen snapshot of the game in Fig. 7.

This tutorial was realized by the group of three students and realization was successful.

VI. STUDENT COMMENTS

One of the biggest problems was the need to switch from previously used E2LP application for communication with the board. This was necessary because of XPS constant access to the FPGA.

Most of the students commented that introduction to the XPS was too short. They saw this later as a problem during the realization of their tasks. Anyhow, some students expressed that this approach was good enough.

Students found more problems in realization of hardware part of the project and feel more comfortable in software environment. Some tasks were transferred to software part of the system just to avoid problems with hardware part.

Students faced a lot of problems in realization of new peripherals, like UART terminal character reception in software application.

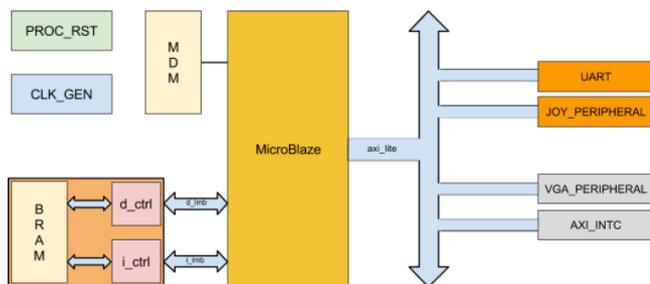


Fig 5 Block diagram of the system architecture

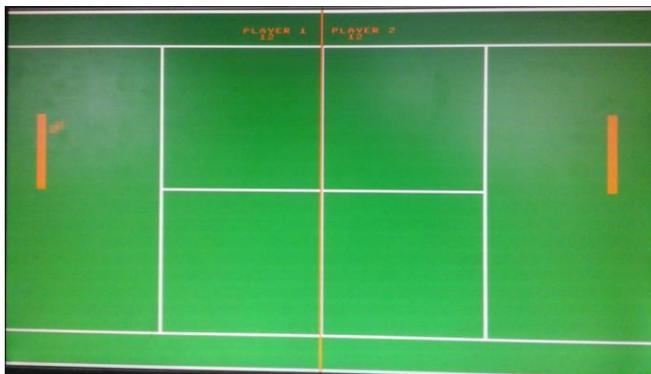


Fig 6 VGA screen snapshot

During the realization of projects which included the use of push buttons, students had problems with bouncing and learned that de-bouncing module is practically an obligatory part of the system.

These are some of many conclusions which are showing what students have learned by only trying to design embedded system by themselves.

VII. CONCLUSION

In this paper we presented the approach to improvement of students learning of computer system design. It is based on the developed unified platform designed to cover a complete process of embedded systems learning. The presented approach addressed efficiency of active-learning laboratory-based courses.

Academic experience justified the approach. As expected, positive effects were more pronounced in second half of the course where presented approach offered significant improvements without requiring any extra effort from teaching staff. Students were facing problems and tried to solve them alone, which forced them to read documentation more carefully and to even go out of the course documentation, which they saw at the end as a very positive experience. Students had an impression that they learned a lot, and they felt more confident after successful realization of the tasks.

As part of future work, we plan to introduce new tutorials in the course that will integrate all student knowledge from embedded system design. Another direction of the future work is to prepare a re-design of the teaching platform in order to respond to students' comments.

REFERENCES

- [1] Holbert, K.E., Karady, G.G., "Strategies, Challenges and Prospects for Active Learning in the Computer-Based Classroom", IEEE Transactions on Education, Volume 52, Issue 1, Feb. 2009 pp. 31 – 38
- [2] Ron Schneiderman, "Economy and Shortages Affect the European Job Outlook", IEEE Spectrum, March 2010, Available from (accessed on 2011-12-10) <http://spectrum.ieee.org/at-work/tech-careers/economy-andshortages-affect-the-european-job-outlook/0>
- [3] Temerinac, M.; Kastelan, I; Skala, K.; Medved Rogina, B.; Reindl, L.; Souvestre, F.; Anastassova, M.; Szewczyk, R.; Piwinski, J.; Benito, J.R.L.; Artetxe Gonzalez, E.; Teslic, N.; Sruk, V.; Barak, M., "E2LP: A Unified Embedded Engineering Learning Platform," *Digital System Design (DSD), 2013 Euromicro Conference on*, vol., no., pp.266,271, 4-6 Sept. 2013
- [4] Kastelan, I; Katona, M.; Majstorovic, D.; Neborovski, E., "Students perspective of the embedded engineering learning platform — A case study in digital design," *MIPRO, 2011 Proceedings of the 34th International Convention*, vol., no., pp.1178,1182, 23-27 May 2011
- [5] Majstorovic, D.; Neborovski, E.; Katona, M., "A cross-curriculum embedded engineering learning platform," *MIPRO, 2010 Proceedings of the 33rd International Convention*, vol., no., pp.1039,1044, 24-28 May 2010
- [6] Kastelan, I; Majstorovic, D.; Nikolic, M.; Eremic, J.; Katona, M., "Laboratory exercises for embedded engineering learning platform," *MIPRO, 2012 Proceedings of the 35th International Convention*, vol., no., pp.1113,1117, 21-25 May 2012
- [7] Xilinx design tools, <http://www.xilinx.com/products/design-tools/>
- [8] Xilinx MicroBlaze, <http://www.xilinx.com/tools/microblaze.htm>

Design of a Platform for Teaching Embedded Systems Engineering

Ivan Kastelan, Miodrag Temerinac
University of Novi Sad
Faculty of Technical Sciences
Novi Sad, Serbia
e-mail: ivan.kastelan@ieee.org

Nikola Teslic
RT-RK Institute for Computer Based Systems
Novi Sad, Serbia
e-mail: nikola.teslic@rt-rk.com

□ *Abstract*—This paper presents the design of a platform for teaching majority of courses in the embedded computer engineering curriculum. The learning platform is a result of the E2LP project whose main idea is to provide a unified platform which will cover a complete process for embedded computer systems learning. The main body of the platform is the base board with FPGA connected to a wide range of interfaces – audio, video, communication, memory and user I/O. Additionally the base board can be connected to one extension board via the standardized Mezzanine interface. Extension boards broaden the range of applications of the platform beyond digital system design and allow the platform to be used in the entire computer engineering curriculum. The first application of the platform in laboratory exercises has shown a positive acceptance of the platform by students.

I. INTRODUCTION

AS embedded systems are becoming more complex, the industry requires embedded system engineers to tackle increasingly complex problems and to be able not only to solve basic everyday issues in system design, but also to envision and suggest new solutions and systems in order to achieve a specific task. This increasing complexity in the industry requires a significant change in the education of future embedded system engineers. E2LP project [1]-[3] aims to bring that change with its usability in the whole curriculum and with its elimination of the overhead in teaching.

Increased role of computer based embedded systems in various industrial applications has produced a growing need for embedded system engineers. As of today, the job market is very competitive for highly qualified electronics systems engineers [4]. Furthermore, the European Centre for the Development of Vocational Training in its publication "Skills supply and demand in Europe, Medium-term forecast up to 2020" [5] predicts that increase of 10.7% in new engineering positions across all disciplines in Europe will be reached up to 2020, compared to the 2010 level. Similar projections are available for USA [6] and Asia [7]. Consequently, many technical faculties have put more

emphasis on embedded systems learning by introducing a number of active learning laboratory-based courses [8].

The approach in E2LP project targets the lab education efficiency with the idea to use a single comprehensive platform for the complete curriculum. The main intention is to make the educational process more efficient and to introduce more interaction between the education and further embedded system research and development, which facilitates an optimal solution for a specific problem. The main E2LP objective is to efficiently educate future engineers capable of coping with current challenges in real-time embedded computer engineering field. It will further provide a learning environment that moves focus from hardware to software and encourages learning of the embedded systems, but without compromising knowledge related to the hardware design.

This paper describes the hardware design of a platform which is the result of E2LP project. The platform in its current form can be used for teaching the majority of embedded engineering curriculum at the university level. Due to its increased modularity, the platform is easily extendable in order to provide support for an even larger set of courses, spanning outside of the range of embedded engineering courses into the fields of electrical engineering and systems control, to name a few.

The paper describes the electrical and mechanical design of E2LP Base Board based on Field Programmable Gate Array (FPGA) and two of its extension boards connected to it through Mezzanine interface. The Base Board and its Mezzanine extension boards are intended to provide the unified platform for the complete embedded system engineering curriculum.

This paper is organized as follows: after the introduction, section 2 gives a brief overview of the E2LP platform. Section 3 describes in more detail the hardware of E2LP system – Base Board and two extension boards. Section 4 gives a very brief summary of the software support for E2LP platform. Finally, section 5 gives some concluding remarks.

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 317882.

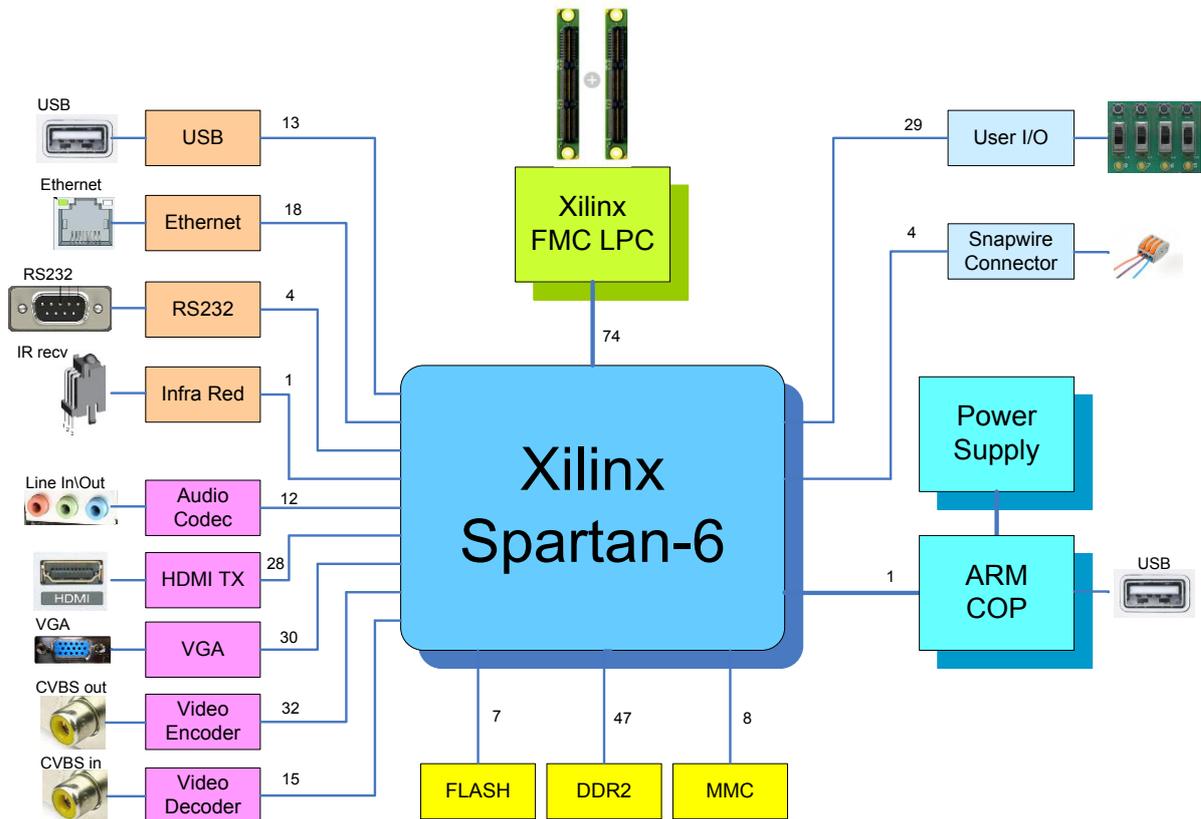


Figure 1. E2LP Base Board

II. PLATFORM OVERVIEW

The embedded engineering learning platform [1] is based on FPGA which is connected to various interfaces and peripherals on the platform. It is intended to be used in multiple courses in the curriculum of computer engineering. The base board of the platform supports teaching courses in digital and computer system design, using FPGA for implementation of the designed digital systems.

Through its mezzanine connectors, the base board communicates with extension boards which support other courses in the curriculum, e.g. an extension board with a digital signal processor (DSP) supports courses in signal processing. With this approach the platform aims to cover the most of the computer engineering curriculum reducing the time students spend adjusting to different platforms and tools. Instead, the students have more time to learn important topics in relevant subjects in computer engineering.

III. E2LP PLATFORM HARDWARE

This section explains in more detail the hardware of the components of E2LP platform:

1. Base Board;
2. Extension board based on ARMADA processor;
3. Extension board based on a simpler ARM processor.

A. Base Board

The E2LP Base Board performs the following functions:

- based on FPGA, provides the central point of the E2LP platform on which all other parts are connected;
- supplies power for the whole E2LP platform;
- controls programming the FPGA and Central Processing Units (CPUs) on extension boards;
- provides a basic user interface;
- provides storage, multimedia and communication interfaces for the platform;
- provides the platform for digital system design;
- provides test points for debugging.

The block diagram (Fig. 1) gives a high level overview of the E2LP Base Board.

The key building modules of the E2LP Base Board are:

- Xilinx Spartan-6 FPGA;
- ARM-based control processor;
- Mezzanine connector to extension board (Xilinx FMC LPC standard);
- DDR2, flash and multimedia card memory;
- user interface (8 switches, 6 buttons, 8 LEDs, alphanumeric LCD screen);
- snapwire connector (8 pins);
- CVBS video encoder and decoder;
- video output (VGA, HDMI);

- audio sub-system;
- communication interfaces (USB, Ethernet, RS232 and Infra-red).

FPGA is directly connected to all interfaces on the Base Board, as well as the Mezzanine connector which allows its connection to external extension boards. It is the central point of the platform and supports teaching digital system design, computer system design, FPGA accelerated computing, reconfigurable systems, System-on-Chip design, hardware/software co-design and design of multimedia and communication systems.

The choice of interfaces for the Base Board was decided based on the extensive study of teachers' and students' requirements which was conducted at the start of E2LP project. Investigation of learning models showed that E2LP platform should aim at satisfying learning requirements of the four major types of learners: convergers, divergers, accommodators and assimilators. It should also aim at satisfying needs of visual, auditory and tactile learners.

Teachers' and students' requirements were collected by conducting a study which involved surveys and data analysis. Teachers' and students' questionnaires revealed the needs for both groups from the functional and technical point of view. After careful analysis of the results of this study, the final list of interfaces was compiled taking care of the limitations of the chosen FPGA.

Fig. 2 presents the 3D overview of the base board. The board is enclosed in the box which has a transparent cover. The cover is removable and consists of two parts: a smaller and a larger cover. The following are the three use cases of the board, with respect to the state of the covers:



Figure 2. 3D view of the base board

- both covers closed – suitable when the user wants to protect the board from external influences and still be able to see the contents on the board through a transparent cover; this case can be used with the Base Board only, or with the extension board which does not require cables;
- small cover open, large cover closed – suitable when the user wants to protect the board from external influences, but the extension board requires external connection with cables, which go through the open small cover;
- both covers open – suitable when the user wants to have physical access to all components of the board.

JTAG chain is used to configure FPGA, Complex Programmable Logic Device (CPLD) and Platform Flash on the E2LP base board. CPLD is used as a central router of the JTAG chain on the board. The board supports three configuration cases:

- Platform Cable – Platform Flash – CPLD – FPGA: In this configuration, the Xilinx Platform Cable is used to configure the three JTAG components on the board; this case is a required starting point for board configuration during the board bring-up, which configures the CPLD; this case can also be used whenever the configuration of Platform Flash and FPGA is performed using the Platform Cable;
- COP – Platform Flash – FPGA: This configuration is triggered by the control processor (COP) on the board. The COP sends the signal to the CPLD to re-route the JTAG chain such that the COP becomes the JTAG master. COP can then configure both Platform Flash and FPGA on the board. This is the suggested default configuration case since the E2LP software can be used to configure the FPGA on the board;
- COP – Platform Flash – FPGA – Mezzanine: This configuration is triggered by the Mezzanine board. When attached, the signal bit informs the CPLD to re-route JTAG chain such that the COP becomes the JTAG master and can configure Platform Flash, FPGA and Mezzanine extension board, if the JTAG-configurable component exists on the extension board. The extension board must physically close the JTAG chain even if there are no JTAG-configurable components, in order for the complete chain to be closed.

B. Extension Boards

The initial set of extension boards (Fig. 3) built for E2LP platform contains:

- A board based on Marvell ARMADA 1500 [9] processor for high-end signal processing;
- A board based on NXP LPC processor [10] for simpler programming and control tasks [2].

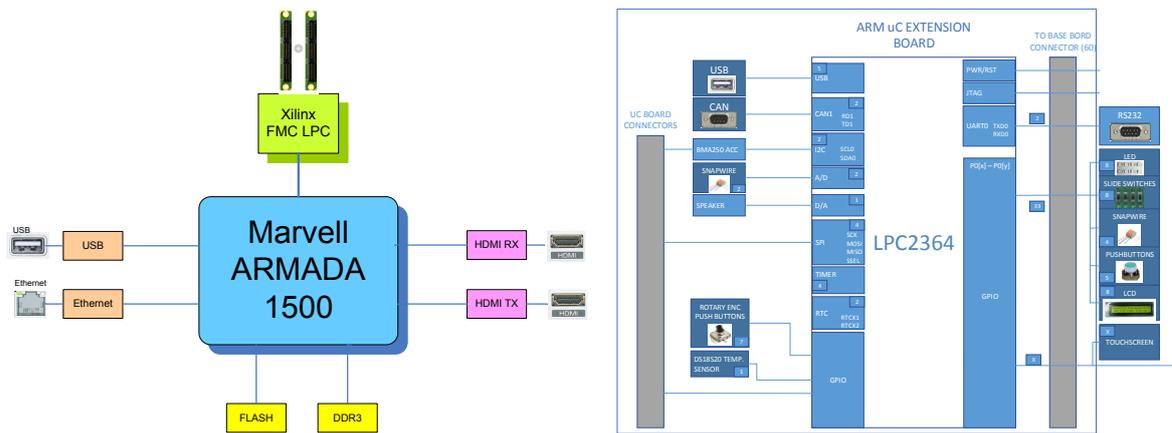


Figure 3. Block diagrams of two extension boards for E2LP platform – ARMADA-based (left) and LPC-based (right)

The extension board based on Marvell ARMADA 1500 has the following functions:

- based on ARM processor, provides the extension to the E2LP platform suitable for highly sophisticated signal processing and execution of real-time software;
- connects to the exterior with USB, LAN and HDMI interfaces and to the Base Board via Mezzanine;
- provides the extension to the E2LP platform suitable for implementing laboratory exercises in the field of digital signal processing, real-time system software, computer networks and system integration.

This extension board is useful for developing laboratory exercises in the field of signal processing, networking and embedded systems design, due to its powerful processor.

For simpler programming assignments, in the first years of computer engineering education, the extension board with a simpler ARM processor is sufficient. It contains:

- LPC2364 microcontroller;
- DS18S20 High-Precision 1-Wire Digital Thermometer;
- LM386 Low Voltage Audio Power Amplifier;
- BMA250 digital accelerometer and I2C;
- Snapwire connector with 8-pins;
- Push-button switches, rotary encoder and LEDs;
- TJA1040 High speed CAN transceiver.

IV. E2LP PLATFORM SOFTWARE

E2LP platform software allows the student and teacher to use the platform in a convenient way, through an application running on the personal computer (PC). The graphical user interface of the PC application allows the student to download the bit file to FPGA without the need for the platform cable. After the bit file is selected, the software calls Xilinx software to convert bit file to xsvf file ready for download. Afterwards, the transfer to the ARM on board is initiated and performed via USB connection to the board. Finally, FPGA is configured with the transferred xsvf file via JTAG between the ARM and FPGA on the board.

V. CONCLUSIONS

The E2LP Base Board together with its extension boards, is working in fully satisfying the main requirement of the E2LP platform – to be used in the complete embedded engineering curriculum and significantly reduce the overhead in engineering education. Extension boards presented in this paper allow achievement of education goals in areas in which the Base Board alone could not be used – signal processing, computer networks, system software and application software.

REFERENCES

- [1] I. Kastelan, M. Temerinac, N. Teslic: "E2LP: A Unified Embedded Engineering Learning Platform", IT System Conference, 2013, pp. 1-4
- [2] M. Zagar, N. Frid, J. Knezovic, D. Hofman, M. Kovac, V. Sruk, H. Mlinaric: "Unified, Multiple Target, Computer Engineering Learning Platform", IEEE Global Engineering Education Conference EDUCON, 2014, pp. 926-929
- [3] E. Artetxe Gonzalez, A. Setien Gutierrez, J.R. Lopez Benito: "Augmented Reality System for Training, Assistance and Decision Making in Real Time situations in the Embedded Electronic field", 5th Joint Virtual Reality Conference JVRC, 2013, pp. 113-115
- [4] R. Schneiderman: "Economy and Shortages Affect the European Job Outlook", IEEE Spectrum, March 2010, Available from (accessed on 2011-12-10) <http://spectrum.ieee.org/at-work/tech-careers/economy-and-shortages-affect-the-european-job-outlook/0>
- [5] The European Centre for the Development of Vocational Training (Cedefop): "Skills supply and demand in Europe, Medium-term forecast up to 2020", Publications Office of the European Union, 2010, ISBN 978-92-896-0536-6, Available from (accessed on 2011-12-10) http://www.cedefop.europa.eu/en/Files/3052_en.pdf
- [6] Bureau of Labor Statistics, U.S. Department of Labor, Occupational Outlook Handbook, 2010-11 Edition, Available from (accessed on 2011-12-10) <http://www.bls.gov/oco/>
- [7] M.A. Thompson: "Employment Outlook Asia: Focus on China and India", 2011-04-20, Available from (accessed on 2011-12-10) <http://www.goinglobal.com/articles/864/>
- [8] S. Nooshabadi, J. Garside: "Modernization of teaching in embedded systems design-an international collaborative project", IEEE Transactions on Education, Volume 49, Issue 2, May 2006 pp. 254-262
- [9] Marvell ARMADA 1500, <http://www.marvell.com/digital-entertainment/armada-1500/>
- [10] NXP LPC2364, http://www.nxp.com/documents/data_sheet/LPC2364_65_66_67_68.pdf

Design of Remote Laboratory dedicated to E2LP board for e-learning courses.

Jan Piwiński
Przemysłowy Instytut Automatyki
i Pomiarów PIAP, Aleje
Jerozolimskie 202, 02-486
Warszawa, Poland
Email: jpiwinski@piap.pl

Rafał Kłoda
Przemysłowy Instytut Automatyki
i Pomiarów PIAP, Aleje
Jerozolimskie 202, 02-486
Warszawa, Poland
Email: rkloda@piap.pl

Roman Szewczyk
Przemysłowy Instytut Automatyki
i Pomiarów PIAP, Aleje
Jerozolimskie 202, 02-486
Warszawa, Poland
Email: rszewczyk@piap.pl

Abstract—Recently, with advent and exploitation of computer and communication technologies, remote laboratories have been widely popular among many universities. They are built in order to enhance learning and minimize the gap between theory and practice. Remote laboratories provide on-line pervasive workbenches, which allow an interactive learning environment that maintains student attention. This paper reports on development of remote laboratory, which is currently performing under E2LP FP7 project. The paper addresses many solutions in the development stages along with powerful technologies involved. E-learning portal will create and provide new teaching methods in embedded systems design through the real and simulated experiments.

I. INTRODUCTION

Laboratories, which are found in all engineering and science programs, are an essential part of the education experience. Not only do laboratories demonstrate course concepts and ideas, but they also bring the course theory into alive. In a traditional laboratory, the user interacts directly with the equipment by performing physical actions (e.g. manipulating with the hands, pressing buttons, turning knobs) and receiving sensory feedback (visual and audio). However, equipping a laboratory is a major expense and its maintenance can be difficult. [1]

Since the experiments are performed in a laboratory that contains expensive equipment, the students must be supervised which limits the time they have. This also requires a class with many groups performing the experiment at the same time, and thus many instruments are required to support each group. Laboratory experiments are also a serious problem for distance learning students who may not have an access to the laboratory at all. [2]

As an alternative, virtual online laboratories can expose students to hands-on learning without incurring the high costs of instructional facilities. [3]

Remote laboratories are those laboratories that can be controlled and administrated online. They differ from the virtual simulated laboratories as they are interacting with physical instruments. [4]

II. REMOTE LABORATORY

In E2LP (Embedded Computer Engineering Learning Platform) project [5] a Remote Laboratory (RL) is an experiment, demonstration and a process running locally to design and control an experiment board based on a FPGA device, but with the ability to be monitored and controlled over the Internet (future E-learning portal).

In the base case, the RL can be an experiment board connected to a computer through a standard interface and with the host computer connected to the Internet, which provide remote access. The client can be any computer connected to the Internet with an ability to see the same interface as the local host and also have the same programs, interfaces, modules etc.

The concept of E2LP RL should allow the user to do several actions over an Internet connection, which are the list of E2LP Remote Laboratory functionalities:

1. Dedicated software and hardware solutions will provide an access to laboratory equipment and enable students to set them up and operate them at the required level to carry out exercises.

2. Users could access the essential data sheets, tutorials and software tools, which are available on the E-learning portal as an introduction to the course, which is a RL's content management system (CMS). Each laboratory exercise is presented to the user through tabs and such division will be implemented into Moodle based platform for e-learning course (Basic information, Theoretical explanations Instructions, Feedback questionnaire for lab evaluation).

3. After booking in a given time space Users could remotely program given set of exercises over the Internet and simultaneously, in real time could monitor the evolution of the experiment on implemented dedicated Graphical User interface (GUI) of the Front Panel of the board. Fig. 1

4. Automatic verification of course assignments, supported by Moodle plug-in, will allow an advanced management of assignments and submissions together with feedback information mechanisms for both teachers and stu-

dents, which will verify whether the students designs work or not according to the specifications.

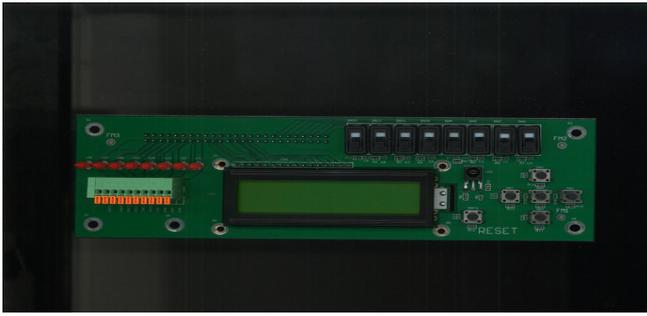


Fig. 1 E2LP board Front Panel

The main advantage of E2LP Remote Laboratory is when students interact with the dedicated boards UI, implemented as a web service and work with software applications, they are actually operating the same tools and instruments like they would be in classic lesson in laboratory.

RL is a gate which provide an access to continuously refreshed interfaces and signals from the real board and enable user remotely control and program the board directly from their computer at home, having instant visual feedback.

To achieve this, it is necessary to forward data directly to the server over common interfaces or over local network by using dedicated hardware solutions and specified proper router configuration.

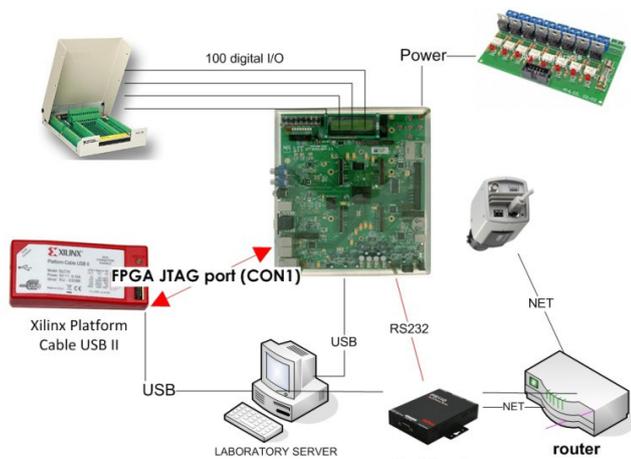


Fig. 2 Remote Laboratory concept of solution

The Fig. 2 above presents remotely controlled environment concept of solution. The whole environment is managed by powerful E2LP Server, which is equipped with all common interfaces, which are essential for internal hardware and software compatibility. E2LP Server is connected via Ethernet interface to the local network, which is responsible for seamless data communication between environment's components. The crucial component of the remotely

controlled environment is an experiment base board, which is controlled by programming device (Xilinx Platform HW-USB-II-G). This programming device provides integrated firmware to deliver high-performance, reliable and user-friendly configuration of the base board and enables user to program other Xilinx CPLD devices. This programming device is fully integrated and optimized for use with specialized Xilinx iMPACT software, which enable users to perform remote operations such as programming and configuring FPGA via JTAG interface.

The NI PCI-6509 digital card with 96 bidirectional I/O lines enable user by dedicated GUI interface (Error: Reference source not found) to control each pin in the boards front panel interface and consequently enable him to control each led, switch and button. Furthermore specific module communicates with LCD pins on boards front panel interface and translate them into RS232 ASCII chars.

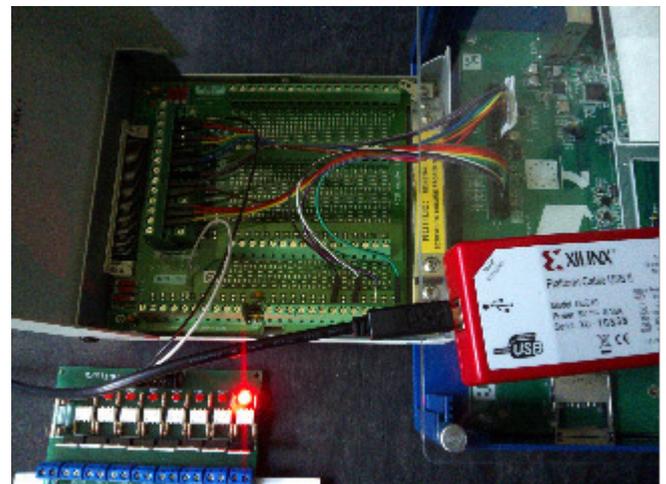


Fig. 3 NI PCI-6509 digital card integration with E2LP board front Panel

In remote operations user firstly power the board from the website (this is done by power controller module and digital card) and run iMPACT on one computer but the operations are performed on a device attached to another computer through a Xilinx Cable Server. Xilinx ISE includes such program as well as provide a set of programming tools, which allow user to perform operations remotely. To use this functionality user only needs to specify a remote server address in proper configuration in iMPACT software. This is the most important feature of programming device, from the RL point of view.

It should be pointed that all exercise could be done remotely, but feedback from some interfaces is not available. Error: Reference source not found below represents implemented necessary interfaces according to requirements.

Connection with the Remote Laboratory is provided via e-learning portal and is based on Apache server, PHP and SQL server. It provides an access to knowledge (exercises, data sheets) through a web user interface and has an ability

TABLE I
IMPLEMENTED INTERFACES IN RL

Component		Implementation in RL	Access to the interface
Type	Direction (In/Out)		
LCD 16x2 Character Display	f/☒	(Not-tested) Full ¹	GUI
Dip Switches	☒/f	Full	GUI
Push Buttons	☒/f	Full	GUI
LED	f/☒	Full	GUI
RS-232	☒☒	Full	Standard tool ²
Power Supply ON/OFF	☒/f	Full	GUI

¹ – now under development
² – e.g. HyperTerminal, Putty, etc.
 GUI – web interface

to exchange information between laboratory hardware and software applications. The second role of e-learning portal is management of users, which enables them access to the laboratory hardware and software (booking functionality and authorization). In E2LP project the e-learning platform is based on Moodle Platform, which is one of the most popular open source learning management systems.

III. SCENARIO OF USAGE RL

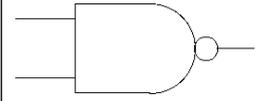
For remote operations user is provided user manual guideline, dedicated software and an access to laboratory equipment, which enable students to set them up and carry out exercises. Our aim was to provide instant feedback to the board in a way that user would work with the real board as if it was connected locally. This functionality was a purpose to developed GUI web interface of front panel that exactly looks like the real board, which has connections to real signals from the real board. States of the physical signals are continuously refreshed at defined times (250 ms) and tests confirmed that it is sufficient to inform the user (by observing GUI) that his program – made during the lab exercise – works correctly. Below we described current scenario of RL usage during the exercise of design the logic circuits:

- User write code of his program according to exercise on local Xilinx ISE environment
- Generate bit file
- Remotely configure FPGA (setup the board, power up, connect the programming modules,)

- Test solution using GUI and standard tool

In the design of digital systems logic circuits are used as fundamental components. TABLE II presents the NAND (Negated AND or NOT AND) logic table its truth table inherited from Boolean algebra. After setting the input voltages to the desired values (low or high), these circuits are capable of calculating values of Boolean functions they represent and present the results of Boolean functions on their outputs. and this exercise will not go into detail on how these circuits are implemented.

TABLE II
NAND LOGIC TABLE

$Y = X_1'X_2$ 	X_1	X_2	Y
	0	0	1
	0	1	1
	1	0	1
	1	1	0

System entity represents the system as seen from outside. Imagine observing the system on Fig. 4 from outside, without the possibility to see what is inside the system. The only things you will see are its input and output ports. Input ports are like input variables to a function, while output ports are like results of the functions. Each output port in a combinational digital system represents the result of one Boolean function of input ports. If the system has N output ports and M input ports, it computes N Boolean functions of M variables, i.e. functions of the same input ports.

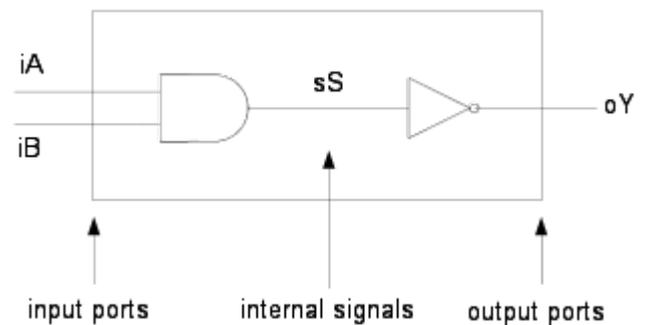


Fig. 4 An example of the digital system

In order for the tool to know to which components on the board we want to connect inputs and outputs of our system, we need to specify which pin from FPGA we want to associate with which port of our system. Let us connect inputs iA and iB to two switches and output oY to a LED. You can always refer to the complete list of FPGA pins on E2LP platform and to which components they are connected. Pin assignment is done in a special tool for that, the Xilinx PlanAhead. It can be run from Xilinx ISE.

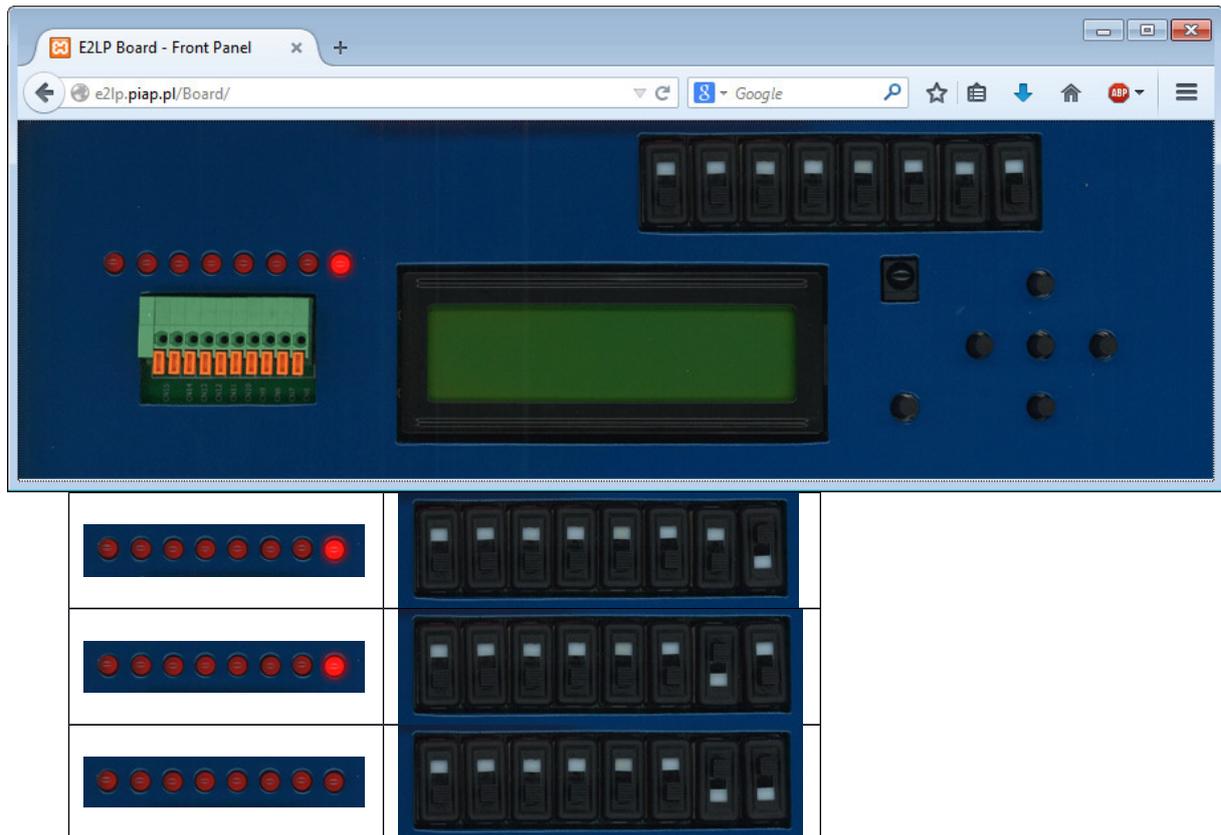


Fig. 5 Evolution of the exercise on GUI web interface

If configuration completes without errors, the FPGA_DONE diode on the board should be turned on, meaning that the FPGA is configured and working. If you change the state of the switches connected to inputs *iA* and *iB*, you should observe the corresponding change to the LED connected to output *oY* based on a Boolean function which the circuit implements (NAND). Fig. 5

IV. CONCLUSIONS

Proposed solutions based on integrated together Remote Laboratory components and e-learning Moodle Platform enable student to acquire desired knowledge about digital systems and significantly support learning process.

Remote laboratory and e-learning portal enable user to access E2LP base board over the Internet, configure it compiling VHDL code and having feedback immediately on his own computer. In the same time he can monitor E2LP base board and operate on remote laboratories equipment.

LISTING: UCF CONSTRAIN FILE DEFINITION OF THE DIGITAL SYSTEM FROM FIGURE 4

```
NET "iB" LOC = Y24;
NET "oY" LOC = N24;
NET "iA" LOC = W19;
```

LISTING: VHDL DESCRIPTION OF THE NAND FUNCTION IMPLEMENTATION DIGITAL SYSTEM FROM FIGURE 4

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity MyFirstDigitalSystem is
    Port ( iA : in  STD_LOGIC;
          iB : in  STD_LOGIC;
          oY : out STD_LOGIC);
end MyFirstDigitalSystem;

architecture Behavioral of
MyFirstDigitalSystem is

    signal sS : std_logic

begin

    sS <= iA and iB;
    oY <= not(sS);

end Behavioral;
```

Future development of RL based on integration of external equipment and services and development of Moodle platform will be done in accordance with the exercises necessities and end-users needs. .

V. REFERENCES

- [1] Distance-Learning Remote Laboratories using LabVIEW; Publish Date: Sep 06, 2006 <http://www.ni.com/white-paper/3301/en/>
- [2] A. Nafalski, J. Machotka, Z. Nedic “Collaborative Remote Laboratory NetLab for Experiments in Electrical Engineering”, in *Using Remote Labs in Education. Two Little Ducks in Remote Experimentation*, 2011, University of Deusto, pp. 177-199
- [3] S. Dutta, S. Prakash, D. Estrada, E. Pop “A Web Service and Interface for Remote Electronic Device Characterization” in *IEEE Trans. Education* Vol. 54 No. ,4 pp. 646-651
- [4] M. Tawfik, E. Sancristobal, S. Martin, G. Diaz, M. Castro: “State-of-the-Art Remote Laboratories for Industrial Electronics Applications” in *Technologies Applied to Electronics Teaching (TAE)*, 2012, pp. 359 – 364
- [5] <http://www.e2lp.org/>

Learner-Centered Evaluation of an Augmented Reality System for Embedded Engineering Education

Margarita Anastassova, Florent Souvestre
CEA, LIST, Sensory and Ambient Interfaces Laboratory, Gif-sur-Yvette Cedex, France
Email: {margarita.anastassova, florent.souvestre}@cea.fr

Enara Artetxe González, Aratz Setién Gutiérrez, Jorge R. López Benito
CreativiTIC Innova S.L., La Rioja, Spain
Email: {artetxe, asetien, jrlopez}@creativitic.es

Moshe Barak
Department of Science and Technology Education, Ben-Gurion University of the Negev, Beer Sheva, Israel
Email: mbarak@bgu.ac.il

Abstract—This paper presents an ongoing work on a multisensory Augmented Reality system for engineering education. The system, composed of both software and hardware elements, will be used in embedded electronics courses. We first present similar systems and their potential advantages for education. Then, we discuss the originality of our system. Finally, we present the general approach as well as a number of methods that we will use for the future learner-centered evaluation of the proposed AR system in realistic settings.

I. INTRODUCTION

THE term "Augmented Reality" (AR) was introduced in the early 1990s [1] to designate a specific form of Human-Computer Interaction (HCI), in which views of the real world are enhanced by computer-generated content [2]. The real and virtual elements in an AR system are semantically and spatially related. Compared to Virtual Reality (VR), AR does not aim at representing the real world by a realistic virtual analogy. It aims at promoting "intuitive" and natural multimodal interaction [3]. In addition to 2D and 3D computer-generated visual content, spatial audio, tactile and even olfactory stimulations can be incorporated to enhance the user's perception of the real world.

As underlined by [4], AR offers new possibilities in education. These authors, together with others, cite the following major advantages of using AR in education:

- a possibility of presenting information "just-in-time" and "just-in-place", which will reduce information search, error-likelihood and will enhance memorization and recall (e.g. [5], [6], [7]);
- a possibility of visualizing complex relationships and abstract concepts ([8], [9]);
- a possibility of experiencing phenomena which are unlikely to be experienced in the real world ([10], [7]);

- a possibility of "learning-by-doing" (i.e. of constructing knowledge actively and autonomously, [11], [12]);
- a possibility of improving learners' motivation because of the enthusiasm when interacting with new technologies ([13]).

All this benefits are transposable to engineering education, in general, and to embedded electronics courses, in particular.

There are only few AR reality prototypes for engineering education. Some of them will be presented below.

II. AR IN ENGINEERING EDUCATION

Kaufmann and Schmalstieg [14] developed an AR system for mathematics and geometry education (Fig. 1).

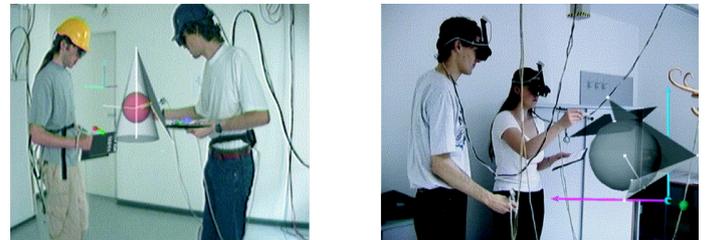


Fig. 1 AR system for geometry education (adapted from [14])

The system is a 3D geometric construction tool for the improvement of spatial abilities and for the maximization of transfer in real settings. This system has not been formally evaluated in a real course. However, an informal evaluation showed that students were motivated to use it and did not need a long familiarization before using it in practice. Several problems such as eye-hand coordination without haptic feedback and fatigue were also pointed out. As for the possible applications of the system, students mentioned interactive conic sections, vector analysis, intersection problems, and elementary geometry.

Another example is the use of tangible interfaces (i.e. physical objects coupled to digital information) and AR

This work was partially supported by EU FP7 E2LP project (<http://www.e2lp.org/>).

models in engineering graphics courses to help students better understand the relationship between 3D objects and their projections [15]. This system was tested with 35 engineering-major students. The study showed that the tangible interfaces significantly enhanced students' learning performance and their abilities to transfer 3D objects onto 2D projections. There was also high engagement with the AR models during the learning process.

AR was also used for teaching embedded electronics courses. When learning electronics, especially embedded systems, students have to face the challenge of understanding the mechanisms of several devices without actually seeing those interactions and functions. Even in laboratory practices with electronic boards, they can only manipulate them through the available inputs and outputs, whilst the operations happening inside the components remain invisible. Consequently, students do not always get to fully understand the studied concepts.

The augmented reality aims to overcome those obstacles in the learning process, especially in the early stages of Computer Engineering studies. Thus, Müller et al. [16] and Andujar et al. [17] proposed an AR system for the improvement of students' interactions with remote laboratories.

In [17], the use case is the design of a digital control system based on an FPGA development board. In this case, AR is used in order to give the user the sensation that certain lab functions can be handled just as they would be in the real laboratory itself. The authors designed the system with the aim of limiting students' possible discouragement due to the lack of physical contact. The system was evaluated with 36 students and 10 teachers. The results, both for students and teachers, showed improved learnability of the theoretical concepts taught in the different courses, high engagement and higher motivation to learn than with traditional methods.

The existing AR prototypes for embedded electronics education are mainly based on visual interaction. In the E2LP project [19], we tried to go beyond visual aspects only. We designed a multisensory AR system for electronics education. This system is briefly presented below.

III. THE E2LP AUGMENTED REALITY SYSTEM

The E2LP system consists of a camera which captures a video of an electronics board. It displays this video on a touchscreen positioned on a support inspired by the structure of an electronic lamp (Fig. 2). The system also comprises a tactile pointer, which indicates its position on the board and with that information and the information from the camera, the AR software displays the corresponding visual enhancement on the real view presented on the tactile screen.

With the help of the camera and the pointer, students can access the specifications of the board components they want to work on.

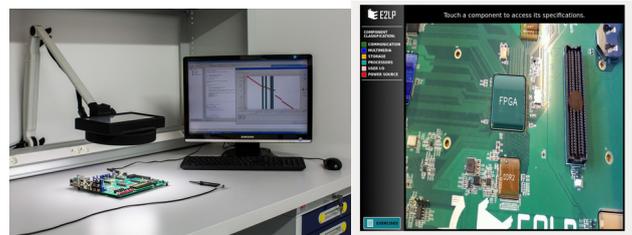


Fig. 2. E2LP AR system: general view and view of the student's screen

From a pedagogical point of view, the AR software is structured in following three levels:

- **Exercises:** These are basic tasks and, in the same time, perfect use cases for AR. They have a well-determined path to resolution and a solution which can directly be superposed on the hardware elements in a visual and animated manner. Exercises are also the first tasks students have to resolve when starting a new course. Therefore, AR can be a valuable tool for facilitation understanding and engagement.

- **Problems:** These are open-ended tasks which can have different solutions or solving methods. Compared to exercises, they are more challenging from the point of view of the information to be displayed by the AR system, as it can be different in the different solutions. However, the AR system can display "clues" to facilitate the resolution of the task.

- **Projects:** These challenging tasks require that students define by themselves both the objectives and the resources to be used for the development of the project. Thus, there is no pre-established path or solution to follow. In this case, the AR software could provide information about the general resources students can make use of.

The E2LP AR system will be evaluated with students and teachers, in real courses. The dimensions to be evaluated and the general approach which we will adopt are presented below.

IV. LEARNER-CENTRED EVALUATION OF THE E2LP AUGMENTED REALITY SYSTEM

We advocate an iterative and learner-centered design approach where learners and teachers are involved all through the design process and where designs and prototypes of the future pedagogical tools are tested iteratively. This learner-centered approach is based on:

- paying particular attention to diverse learners' needs and characteristics,
- providing a setting for fostering authenticity and inclusion,
- encouraging openness to experience and personal growth, and
- encouraging co-creation of knowledge.

The main goal of this approach is to facilitate learners in becoming active, self-directed and self-responsible participants in the learning process, in which peers and the instructor serve as facilitators, motivators and personal resources. From a teaching and design perspective, this is a

very complex, demanding endeavor. As an entry aid, the following four guiding principles of the Alliance for Excellent Education could be used [17]:

- Learning is rigorous and based on career-ready expectations.
- Learning is personalized.
- Learning is collaborative, relevant and applied.
- Learning is flexible, with learning taking place anytime, anywhere.

Based on this general approach, we can use the following more concrete measure of the effectiveness of AR for students and teachers.

A. Measures of technological effectiveness and reliability

These measures will be based on the performance of the AR software and hardware, as well as on the performance of the AR system as whole. We will use a number of well-established indicators of technological effectiveness and reliability such as:

- the number and variety of functions and exercises supported by the AR system,
- its power consumption,
- the real time behavior of events and expected actions,
- its reliability,
- its interoperability.

The cost of each AR component and the cost of the integrated system will also be an important measure of the effectiveness of the final E2LP AR platform. Even though all these measures do not directly stem from the learner-centered approach, they may have an effect on teachers' and learners' first impressions of the system and, consequently, on its further adoption.

Another group of measures will concern the ease-of-use and the user experience with the AR system.

B. Measures of ease-of-use and user experience

These characteristics will be measured using both quantitative and qualitative measures, which will be applied first to the different components of the system (i.e. software, tracking, tactile pointer) and then, to the system as a whole. The quantitative measures will include but will not be limited to:

- the success rate on different learning tasks relevant for the use of the system;
- the time that a given task requires for its execution;
- the error rate;
- the time spent for recovering errors;
- the rate to learners' and teachers' satisfaction with the AR system;
- the number and the reasons for eventual rejections of the system during use, if any.

These measures will be complemented by qualitative measures of user experience. These qualitative measures will include, but will not be limited to:

- the nature of the learning tasks supported by the AR system. This metric will be based on the taxonomy of exercises, problems and projects presented earlier in this document;
- the nature and types of the errors typically made when using the system;
- the strategies of error recovery;
- the typical and personal strategies of using the AR systems in different educational contexts;
- the learners' and teachers values associated with the use of the system;
- the learners' and teachers most important emotions when using the system (e.g. enjoyment, interest, frustration, curiosity, immersion, etc.);
- their motivation before, during and after use;
- the confidence in the system;
- the learners' and teachers' comments on its efficiency, utility, acceptability and further adoption.

The quantitative and qualitative measures of the ease of use and user experience with the AR system will be first evaluated within the framework of formative usability evaluations. Formative evaluation is a type of usability evaluation that helps to "form" the design for a product or service. Formative evaluations involve evaluating a product or service, usually with small user samples, during development, often iteratively, with the goal of detecting and eliminating usability problems. One important aspect of formative evaluation is that the audience for the observations and recommendations is the project team itself, used to immediately improve the design of the product or service and refine the development specifications. Results are usually less formal than in final or "summative" evaluation, as suits the needs of designers, developers, project managers, and other project participants.

The above-mentioned system effectiveness metrics will be then used in the final summative evaluation of the AR system. Summative evaluation will be done on the complete or near-complete platform under realistic conditions. The objective will be to determine if AR improves learning, collaboration and creativity.

This summative evaluation of the usability of the AR system is closely related to the evaluation of the AR system in real educational settings.

C. Measures of adoption of AR in real educational settings

Specific measures for the adoption of the AR system in real education settings will be used. The quantitative measures of adoption will include:

- the number of laboratory exercises, problems, interdisciplinary projects created by AR;
- the number of AR systems installed in universities;
- the number of exercises, problems and projects created by instructors and students.

As for the qualitative measures of teachers' and learners' adoption, they will be based on the Levels Of Technology Implementation (LoTi) framework proposed by [18]. The LoTi framework is based on seven discrete implementation levels ranging from Nonuse (Level 0) to Refinement (Level 6). A synthesis of the LoTi framework is presented below:

- **Nonuse:** a perceived lack of access to technology-based tools or a lack of time to pursue electronic technology implementation.
- **Awareness:** the use of the proposed technological system is generally one step further compared to level 0. However, they still have little or no relevance to the individual teacher's instructional program.
- **Exploration:** technology-based tools serve as a supplement to existing instructional program. The new technology is employed either as extension activities or as enrichment exercises to the instructional program.
- **Infusion:** technology-based tools augment isolated instructional events.
- **Integration:** technology-based tools are integrated in a manner to provide a rich context for students' understanding of the pertinent concepts, themes, and processes. Technology is perceived as a tool to identify and solve authentic problems relating to an overall theme/concept.
- **Expansion:** technology access is extended beyond the classroom.
- **Refinement:** technology is perceived as a tool to help students solve authentic problems related to an identified real-world problem or issue.

Even though we do not expect to see high levels of AR adoption because of the prototypical nature of the current system, we would like to apply this evaluation framework because of its prospective orientation.

V. CONCLUSION

We presented an AR system for embedded electronics education developed within the framework of the E2LP project. A number of measures for evaluating its effectiveness were also presented in this paper. The use of these measures will be based on a learner-centered approach. The results of the evaluation will be presented in the near future.

REFERENCES

- [1] T. Caudell, and D. Mizell. Augmented reality: an application of heads-up display technology to manual manufacturing processes. In *1992 Proc. Hawaii International Conference on System Sciences*, 2, 659-669.
- [2] S. Ch.-Y. Yuen, G. Yaoyuneyong, and E. Johnson. Augmented reality: An overview and five directions for AR in education. *Journal of Educational Technology Development and Exchange*, 4, pp. 119-140, 2011.
- [3] M. Billingham, H. Kato, and I. Poupyrev. The magic book: moving seamlessly between reality and virtuality. *Computer Graphics and Applications*, 21, 2-4, 2001.
- [4] H.-K. Wu, S. W.-Y. Lee, H.-Y. Chang, and J.-C. Liang. Current status, opportunities and challenges of augmented reality in education. *Computers & Education*, 62, pp. 41-49, 2013.
- [5] M. Anastassova, J.-M. Burkhardt, C. Mégard., and P. Ehanno, 2007. L'ergonomie de la réalité augmentée: une revue. *Le Travail Humain*, 70, 97-126. 2007.
- [6] J. R. Cooperstock. Classroom of the future: enhancing education through augmented reality. In M. J. Smith, G. Salvendy, D. Harris, & R. J. Koubek (Eds.), *Usability evaluation and interface design: cognitive engineering, intelligent agents and virtual reality* (pp. 688-692). Mahwah, NJ: Lawrence Erlbaum Associates, 2001.
- [7] U. Neumann, and A. Majoros. Cognitive, performance, and systems issues for augmented reality applications in manufacturing and maintenance. In *1998 VRAIS Proceedings of the IEEE Virtual Reality Annual International Symposium*, 4-1, 1998.
- [8] T. N. Arvantis, A. Petrou, J.F. Knight, S. Savas, S. Sotiriou, M. Gargalagos et al. Human factors and qualitative pedagogical evaluation of a mobile augmented reality system for science education used by learners with physical disabilities. *Personal and Ubiquitous Computing*, 13, pp. 243-250, 2007.
- [9] S. Shelton, and N. Hedley. Using augmented reality for teaching earth-sun relationships to undergraduate geography students. In *Proc. of 2002 1st IEEE International Augmented Reality Toolkit Workshop*. Darmstadt, Germany, Sept, 2002.
- [10] E. Klopfler, and K. Squire. Environmental detectives: the development of an augmented reality platform for environmental simulations. *Educational Technology Research and Development*, 56, pp. 203-228, 2008.
- [11] J. Doswell, B. Blake, J. Green, O. Mallory, and C. Griffin. Augmented Reality learning games: 3D virtual instructors in augmented reality environments. Paper presented at the *2006 Symposium on Interactive Games*, 2006.
- [12] M. Fjeld & B. Voegtli. Augmented chemistry: an interactive educational workbench. In *Proceedings of the 2002 IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR 2002)*, 259-260.
- [13] X. Zhong, P. Liu, N. Georganas, and P. Boulanger. Designing a vision-based collaborative augmented reality application for industrial training. *IT-Information Technology*, 45, 7-18, 2003.
- [14] H. Kaufmann, and D. Schmalstieg. Mathematics and geometry education with collaborative augmented reality. *Computers & Graphics*, 27, 339-345, 2003.
- [15] Y.-C. Chen, H.-L. Chi, W.-H. Hung, and S.-C. Kang. Use of tangible and augmented reality models in engineering graphics courses. *Journal of Professional Issues in Engineering Education Practice*, 137, 267-276, 2011.
- [16] D. Müller, F. W. Bruns, H.-H. Erbe, B. Robben, and Y.-Ho Yoo. Mixed reality learning spaces for collaborative experimentation: A challenge for engineering education and training. *iJOE International Journal of Online Engineering*, Vol. 3, No. 4, 2007. Available at: http://www.informatik.uni-bremen.de/~mueller/en/publ_assets/2007-ijoe-mueller.pdf
- [17] J. M. Andújar, A. Mejías, and M. A. Márquez. Augmented Reality for the improvement of remote laboratories: An augmented remote laboratory. *IEEE Transactions on Education*, 54, pp. 492-500, 2011.
- [18] C. Moersch. Levels of technology implementation (LoTi): A framework for measuring classroom technology use. *Learning and leading with technology*, 23, 1995.
- [19] <http://www.e2lp.org/project-results/6-outputs>
- [20] <http://all4ed.org/wp-content/uploads/2013/10/CultureShift.pdf>

Small is Beautiful: Embedded Systems Projects in an Undergraduate Software Engineering Program

Janusz Zalewski
Dept. of Software Engineering
Florida Gulf Coast University
Ft. Myers, FL 33965, USA
zalewski@fgcu.edu

Fernando Gonzalez
Dept. of Software Engineering
Florida Gulf Coast University
Ft. Myers, FL 33965, USA
fgonzalez@fgcu.edu

Robert Kenny
College of Education
Florida Gulf Coast University
Ft. Myers, FL 33965, USA
rkenny@fgcu.edu

□ **Abstract**—This paper addresses the issue of educating software engineers in embedded systems development. With the rapidly growing markets of embedded devices and their interconnections due to the ubiquitous presence of the Internet, leading to the emergence of cyberphysical systems, educating software engineers and computer scientists on these subjects at the college level is becoming essential. The paper presents an approach to teaching software development for small embedded devices with lab projects at the undergraduate level, to match the fast pace of technological progress and challenges of real-world applications.

I. INTRODUCTION

SOFTWARE engineering is normally associated with substantial size projects, where critical or, at least, important decisions on requirements solicitation, software design, development tools, project management, etc., have to be made. This point of view is usually followed in education of software engineers, since it is expected that they would comply with the mainstream expectations and be adequately prepared to join the workforce.

However, over the recent years, with unprecedented development of computing technologies and systems, the market has evolved to the point that what once has been a niche, encountered mostly in military and scientific applications, has now become the mainstream: a rather chaotic conglomerate of devices, more and more often called the Internet of Things [1-2]. Embedded devices and systems dominate the market in quantities as well as in sales and investments. As stated by the Chief Scientist of the U.S. Air Force, by 2025 there will be 7 trillion IP enabled devices in existence [3], all forming a humongous ecosystem that would need a well educated workforce.

□ This work was supported in part by grants from the National Science Foundation (Award No. DUE-1129437 and Award No. DUE-0632729), and NASA through University of Central Florida's NASA-Florida Space Grant Consortium (UCF-FSGC 66016015). Views expressed herein are not necessarily those of the funding agencies. Additional support has been provided by a grant SBAHQ-10-I-0250 from the U.S. Small Business Administration (SBA). SBA's funding should not be construed as an endorsement of any products, opinions, or services.

Can we, as educators, honestly say that we are adequately preparing the future workforce to meet respective challenges of these new markets? In our opinion, the answer is not necessarily affirmative. Among multiple challenges software engineering educators are facing, such as keeping up with rapid technological pace, following the seemingly evolution of tools, increasing pressure on teaching computer security and safety required for infrastructure protection, etc., there is one particular issue not adequately addressed yet: software development for embedded systems.

The objective of this work is to address the problem of enhancing education of software engineers in embedded systems development. While there are multiple facets of this issue, the paper focuses on one particular aspect: development of cheap lab stations that can be used in mid to senior undergraduate software engineering projects.

The rest of the paper is structured as follows. Section II outlines the pedagogy applied in approaching the subject matter. Section III presents the devices and their selection process and Section IV discusses the actual labs. Section V ends the paper with Conclusion.

II. PEDAGOGY

While there is a clear need to improve and enhance education of software engineers in embedded systems from the engineering perspective, there are probably multiple ways to address it. The authors of this paper believe that one of the most effective but rarely pursued ways of dealing with undergraduate software engineering education is to start, before addressing any technical subjects, with pedagogy. Pedagogy is a crucial factor in offering and use of all engineering labs.

First, what must be made clear is that including the labs in a course actually enhances the learning process. This has to be considered in two aspects: (1) labs illustrate and speed up the process of acquiring knowledge of concepts and techniques, due to the interaction with the lab equipment, and (2) labs broaden the horizons of knowledge in software development, because the students are forced to include into

the picture elements of interactions with multiple additional components, such as networks and people; this prepares them to face heterogeneity of actual implementations and to identify the terms of system complexity, thus, enhancing problem solving skills and application of critical thinking.

Second, what is specific to this particular project is that putting emphasis on the two later phases of the waterfall model of the software development cycle, implementation and testing, as opposed to studying requirements specification and design methodologies, has a very desirable effect on the acquisition of knowledge and skills. This is due to the fact that because of the ease of prototyping the learning process becomes much more attractive, since the student has the opportunity to make actual observations in real time how the developed software behaves.

Third, it is important to balance the theory with practice, where theory is lectured and labs convey the importance and viability of theoretical concepts by conducting practical work. In case of embedded systems courses, the theory in a mathematical and algorithmic sense is replaced by engineering principles. The traditional waterfall model of software development, with requirements, design, implementation and testing, is further shortened and reduced to the prototyping cycle that involves problem description, solution, coding and debugging.

Fourth, the element of pedagogy, which worked for one of the authors over the years in teaching real-time and embedded systems [4], is the structuring of knowledge and skills acquisition by dissecting the lab work into a sequence: (a) demo, (b) exercise, and (c) assignment, and later into (d) experiment and (e) project, possibly leading to (f) supervised research. Associated with this structured approach is an important pedagogical concept of thinking about embedded systems development in terms of hierarchical layers, from hardware architecture to real-time kernel (RTOS) to a programming language to a design methodology, whether applied top-down or bottom-up.

These four pedagogical concepts form the assumptions set forth at the beginning of the course, and are critically assessed after course completion, based on the documentation developed by students in their respective projects. It must be noted that, unlike typical projects in software engineering courses, which focus on team work, these specific projects are meant to be individual, assigned to a specific student, with no shared responsibilities. It is also important to note that contents and structure of the project documentation is clearly defined and follows the project workflow, with sections on (a) Problem Description, (b) Solution, (c) Coding, (d) Experimentation, embraced by Introduction and Conclusion, with References.

III. DEVICE SELECTION AND COURSE CONSIDERATIONS

The Software Engineering and Robotics Lab at FGCU has been in operation for a number of years and has supported multiple embedded devices forming a comprehensive

educational network used in upper level project courses and respective electives. Its design and use have been described in several previous publications [5]. Its most recent emphasis is on web-based access to all devices and lab stations [6-7], which bore it a name lab-by-wire.

What has been noticed in the process of using the lab is that the complexity of devices and programming techniques not necessarily facilitates knowledge acquisition processes in lower level courses, and may even obstruct reaching educational objectives by forcing students to focus more on mastering the technology rather than on learning the concepts. To alleviate these problems, an attempt was made to depart in certain courses from the “heavy-weight” devices existing in the lab, such as Time-Triggered Architecture, SCADA, Coroware or NAO robots, etc., and let the students choose the technologies, which they feel being more familiar with, but still qualify as full-scale embedded systems. The net result of this decision was the initial selection of Arduino-based projects, in the first stage, and expanding this later to move to more diversified but technically equivalent platforms, at the second stage.

Related developments are outlined in the next two subsections, and first experiences, benefits and pitfalls are discussed in Section IV.

A. Arduino-based Projects

Technical Part: This part of the project had two phases. In the first phase, the entire class taking a course on Embedded Systems Programming was trained in using Arduino boards with XBee wireless modules as an application. The learning process essentially followed the Lab Manual [8], with wireless communication application as a learning vehicle. Multiple experiments were developed, loosely correlated, from plain XBee communication to remote temperature monitoring and control system, to remote humidity and dew point measurement.

Objectives three and four, as outlined in Section 2, were met in a sense that the required development sequence (as per objective three), from problem description, through the solution, to coding and debugging, has been followed and mastered up to the experiment’s level (as per objective four), with no real attempt to develop full scale projects, yet. On real Arduino hardware, only elementary RTOS and programming language concepts (learned earlier) were applied without delving into engineering requirements or designs.

Successful reaching these minimal objectives encouraged the instructors to proceed with the second phase, in which a more structured approach to developing Arduino based projects was adhered to. This relied on adding an extra essential component, such as additional computing equipment, to play a more application oriented role, similar to using XBee modules in the first phase. Four such projects are briefly mentioned in this paper:

- Arduino controlling a car with remote commands (drive-by-wire);
- Arduino enhanced with Ethernet communication;
- Arduino controlled from an iPad/iOS application;
- Arduino equipped with Kinect sensing to control a drawing robotic arm.

Details of these projects are discussed in Section IV, and are documented in separate reports.

Teachers Workshop: As a side effect, after the course, a teacher workshop on Arduino was offered for high-school teachers, where students comfortably played a role of lab assistants, which has additionally proven that they mastered the essential concepts. This turned out to be especially meaningful to the learning process, since once someone is able to teach others, even only as a lab assistant, they gain confidence that they have learned the material.

This activity, although unplanned for this course, turned out to be important to the community of stakeholders, since it connected high school teachers with the software engineering program, so they could play a role of emissaries in recruiting potential students to enter the program. On the other hand, current students had a proof that what they learned can be used by others, which has a very positive psychological and motivational effect. Last but not least, teachers themselves also enjoyed this workshop, since they were offered tools they could use towards professional teacher certification.

B. Diversified Platforms

All Arduino projects were highly praised by participating students, as relatively simple, but still hands-on and allowing to have fun. From the Instructor's perspective, they also met the higher-level learning objectives one and two, as listed in Section 2, which is discussed fully in Section 4. However, one issue discovered when projects were coming to an end was that, although diversified regarding applications, they were relatively monothematic and not necessarily ground-breaking with respect to the use of Arduino technology. In this view a number of suggestions have been made to broaden the spectrum of devices used and, thus, make the platforms employed more diversified, which would additionally benefit the participants.

Several additional boards were suggested for use, with functionality slightly or significantly higher than Arduino's, but still within an affordable price range. One immediate suggestion was to add the Raspberry Pi board [9] as it is based on industry standard ARM processor and is running GNU Linux, with Internet connectivity.

A follow-up suggestion included BeagleBone [10], also based on ARM processor, supported by Texas Instruments. It can run multiple versions of Linux. The third board included PandaBoard [11], with OMAP4430 system on a chip (SoC) with ARM Cortex-A9 dual-core processor, allowing the use of Linux Debian-based operating system.

For comparison with technologies previously available in the lab, an Atmel Flash microcontroller board, AVR STK 500 [12], was also chosen, with its own vendor-specific development environment.

The projects assigned for development with these four technologies were selected by students, with Instructor's approval, and consisted of the following tasks;

- Raspberry Pi task – remote control of a rover;
- BeagleBone– minimal HTTP server application;
- PandaBoard – extended HTTP server application;
- Atmel microcontroller task – remote vehicle control.

In all applications, achieving remote connectivity was a key, whether it's been Internet or wireless based, or with both features combined, which is more completely discussed in the next section.

IV. DETAILS OF PROJECT WORK

In this section, all projects mentioned above are discussed, with a goal in mind how they have contributed to reaching the first two learning objectives with respect to pedagogy as outlined in Section II.

- objective two, regarding the emphasis on implementation and testing phases to increase attractiveness of the course by allowing the immediate observation how the device controlled by software behaves;
- objective one, regarding how well the labs illustrate the development concepts and speed up the learning process to facilitate acquisition of problem solving skills and critical thinking skills.

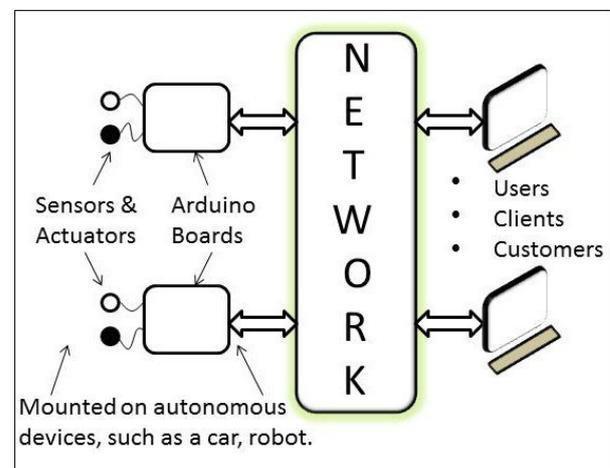


Fig. 1 Outline of the template architecture for Arduino based projects.

From the pedagogical perspective, meeting these specific objectives is meant to positively affect two essential components of knowledge acquisition, its depth and breadth, correspondingly.

A. Arduino-based Projects

All Arduino based projects have a common structure, as illustrated in Figure 1. There are three general components of each project: a device equipped with an Arduino board equipped with sensors and actuators (shown on the left-hand side of the diagram), a user computer (a client, shown on the right side) making requests to control the device, and some sort of a network connecting the two.

Students are given only this general schematic, as an outline of system architecture, and are asked to fill it in with their creativity and ingenuity. It is understood that the Network is just a generic communication facility, so students are free to choose the one best suited for their specific projects.

Arduino Controlled Car with Remote Commands: The essential objective of this project is to verify the functionality of Arduino's wireless connectivity with XBee, and enable it to function as a remotely run controller extending driver's functions (drive-by-wire). The secondary objective is to design an application for an off-the-shelf toy car, just like the ones that can be purchased at the supermarket. The development involved the following activities:

- reverse engineering and rewiring the hardware provided with the car to make it work with Arduino and XBee wireless network;
- producing software responding to sensors, as well as remote user/driver commands, and controlling the DC motors, brakes, and lights;
- designing the human interface to control the car, and enabling wireless connectivity;
- extensive testing of software operation, if it is properly activating various car functions upon remote driver's requests.

Enhancing Arduino with Ethernet Communication: The primary objective of this project is to enhance the functionality of Arduino by adding the Internet connectivity to it and enabling it to function as a web server. The secondary objective is to have the Arduino board respond to sensor information; in this particular case, it is the Passive InfraRed (PIR) motion sensor, making the whole arrangement work as a remote security device.

The development involves the following elements:

- setting up and wiring the hardware (Arduino, PIR sensor and Ethernet shield);
- producing the code to program the communication with the sensor and Ethernet;
- designing the minimal HTTP web server functionality;
- testing the Internet accessibility of all server functions.

Controlling Arduino from an iPad/iOS application: This project's major objective is to investigate what is involved

in building an iOS sensor application for Arduino, which is not a very usual combination. The secondary objective is to additionally check the working of connectivity between two Arduino boards, comparing to previous phase where only a single Arduino board was used. The development involves the following elements:

- setting up, wiring and assembling the hardware elements at both ends;
- developing an iOS application interfacing the iPad with Arduino;
- developing the code for both Arduino components;
- thorough testing of the operation of both devices and the integrated system.

Connecting Arduino with Kinect Sensing to Control a Drawing Robot: In contrast to the previous projects, the main objective of this one is not to focus on connectivity or communication of Arduino, but on enabling the board to receive commands from the user via Kinect sensing device. The development involves the following elements:

- actually making the robotic arm and assembling it with the Arduino board;
- setting up the Kinect graphical software at the server side;
- producing the server code to work for the communication with Arduino;
- extensive testing of the assembly by issuing finger movement commands sensed by Kinect and passed to Arduino to operate the drawing arm.

Problem Solving and Critical Thinking: How the problem solving skills and critical thinking skills are being developed in these types of projects is not a matter of general theory, but more a matter of inspiration and providing to the student an open-ended working environment. Students given only a conceptual framework illustrated in Figure 1 were free to choose their own project topics, devices used, preferred tools, method of connectivity with a sensor, and project's scope, all with instructor's approval. Then, several design decisions had to be made on the project, in each individual case, which forcibly made the students think in terms of solving problems. Sample issues they needed to resolve included:

- Drive-by-Wire (off-the-shelf toy car): How to replace and expand a remotely controlled car's functionality, keeping its design simple and the least expensive?
- Ethernet: How to resolve concurrent access to a board from multiple clients requesting over the Internet to turn the sensor off?
- iPad: How to comply with iOS restrictions and with requirements on remote device to make the solution the simplest possible but still practical?
- Kinect: Why Arduino would more efficiently control the robot via firmware than by software? What graphics libraries would work most effectively in

capturing the dynamic images to control the drawing arm?

These problem solving questions naturally overlap with questions addressing the development of critical thinking skills, which can be summarized as follows: What decision is better? What criteria to use for deciding “what is better”? How to develop these criteria, etc? One particular project-wide problem may shed a light on addressing this issue from the instructor’s perspective: selection of a Network element from Figure 1, to meet project requirements.

It is interesting to note that students thought about the network as a connectivity element, and selected the following options: web connectivity for the Ethernet project, XBee wireless network for both remote car control and remote iPad/iOS communication, and – most interestingly – USB connectivity for the Kinect project (between Kinect server and Arduino).

B. Diversified Platforms Projects

Projects described in this section are meant to use more powerful technologies to expand those Arduino based, by considering the addition of two new enhancement features: (a) remote software development and uploading to the target device; (b) possible extension of the target’s functionality by using on-site network (locally, in addition to the use of the Internet). A general scheme to address the extensions is shown in Figure 2. Consequently, students are required to focus primarily on the server part of the project, whether it is a physically separate unit (lower part of the figure) or an on-board software solution (upper part of the figure).

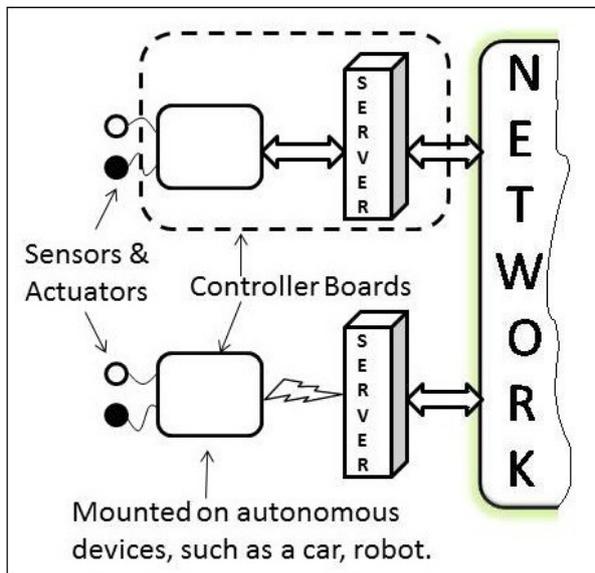


Fig. 2 Outline of the template architecture for diversified platforms.

Raspberry Pi Task: The essential objective of this project is to expand the Arduino project IV-A on remote vehicle

control, by a possibility of remote software development and upload. The development involves the following elements:

- acquiring and applying knowledge of software design issues for cyberphysical systems, including selection of an appropriate design methodology and design notation;
- studying respective networking protocols for accomplishing the task (SSH and WebSockets);
- producing code for remote execution of an application to control operation of a remote device;
- applying principles of remote implementation and remote debugging and testing of an application.

BeagleBone Task: The main objective of this project is to explore the possibility of setting up a web server on an embedded device equivalent to or more powerful than that of Raspberry Pi, with the purpose of handling remote software development, upload and execution. The additional goal is to study and summarize issues with respective networking protocols. The development involves the following crucial elements:

- expanding the assumptions of previous projects for web connectivity with an embedded computer;
- investigating web technologies suitable for this task;
- designing the exact minimal but still useful functionality of the server;
- producing code for file transfer and execution on a remote host;
- configuring the server and testing its operation for the required technologies: CGI, SSH, and HTTP.

PandaBoard Task: This project’s main objective is expanding that of Section IV-A: use a more powerful hardware to investigate the possibility of setting up a web server on an embedded system board, with the purpose of handling remote software development and execution. The secondary objective is to expand the paths of remote programming for server access and communication. The development to meet the primary objective involves the following elements:

- setting up the hardware, and installing and configuring the Linux operating system;
- investigating the suitability of the networking protocols to meet the objective;
- producing the code for network connectivity and file transfer and execution;
- testing the operation of the entire system in the Internet environment.

Atmel Microcontroller Task: The objective of this project is to investigate adding an additional networking component to a remote car control application. The path chosen for this project, in contrast to all previous ones, is to host the server program on a separate machine and make it communicate with a car via a WiFi technology, as opposed to Zigbee used in other projects. The development involves:

- designing and engineering the basic car electronics;
- choosing the right connection media between the Atmel board and the server;
- designing a handshake method for communication between the GUI and the board;
- producing code for the GUI component of the server communication;
- extensive testing of all individual components and the integrity of the entire system.

Problem Solving and Critical Thinking: Projects in this setting were more involved than the Arduino group projects and required thinking more in terms of software engineering than just programming or simple coding. While the Arduino based projects could be qualified, to a large extent, as closer to turnkey systems development than full-scale designs, the diversified projects require from the students significantly more systematic design skills. As a result, problem solving at this level more resembles a real life experience, where interaction with multiple stakeholders reveals questions that need to be addressed. This is evident from the following sample issues that emerged during the projects:

- Raspberry Pi: The minimal life cycle of application development for a remote target device, with software design, cross-compilation and remote debugging, blended into a indistinguishable sequence and required paying close attention to the tool selection and detailed mapping of development activities to the tool's features.
- BeagleBone: Unpredictably, the reliability of a server built on an embedded target board had to be addressed, in particular, to prevent server crashing in case non-compliant code has been uploaded for execution. This situation required relating the testing activities to previous phases of software development, in subsequent iterations.
- PandaBoard: Unexpected difficulties in meeting full-scale requirements caused the need for downsizing the project and providing limited functionality with open ended features, which had a retrofitting effect on phases preceding implementation.
- Atmel microcontroller: Resolving major networking issues with wireless protocol selection, UDP protocol limitations, and firewall settings adjustment, consumed most of the project's resources, leaving less than desired amount of time for true development activities and planned comparison with newer technologies;

Developing critical thinking skills by asking respective decision related questions evolved around specific development phases for each project. Corresponding examples include:

- Requirements Specification phase: Is the suggested technology right to address anticipated user needs?

Does the technology provide sufficient security during device operation?

- Design phase: Is the design tools selection adequate from the perspective of the project requirements and individual tasks? Will the tools facilitate development without a steep learning curve?
- Coding phase: What is the efficiency of the code, in terms of size and execution speed? What are the remote debugging capabilities versus local development and upload? Why are these questions important for a particular project?
- Testing phase: Involved a plethora of questions related to critical thinking, since all projects were subject to an independent verification by Instructor. Most importantly, as most of the students were considering testing to be just showing a demo, the fundamental question to generate critical thoughts turned out to be: "How the software features meet the user requirements (if there were any)?"

Overall, asking these questions revealed a number of issues in the learning process and taught some major lessons on the mismatch between technologies selected and tasks assigned (in a broader sense, on the requirements). In several cases, inadequate prior preparation regarding software engineering principles was revealed, but it must be noticed that taking a course on Software Engineering Fundamentals was not a prerequisite, although several students were taking it concurrently with the projects.

V. CONCLUSION

This paper described the approach to and specific activities in teaching small, but appealing to students, embedded systems projects in undergraduate software engineering courses. The claim that "small is beautiful" has been verified in a number of individual projects that focused on implementation and testing phases of the waterfall model for small devices with increasing complexity of requirements. Meeting four pedagogical objectives were analyzed, of which the most important one, developing concepts leading to the acquisition of problem solving and critical thinking skills, was verified in more detail.

In this view, it is worth noting that the Embedded Systems Programming course, where the devices are used, is just a part of the full Software Engineering degree program, and precedes courses on Requirements Specification and Software Design. Even a course on Data Structures and Algorithms is offered in a later year. Even though the main goal of the simple lab projects, getting the students sufficient hands-on experience to attract their interest in the program, has been achieved, it must be honestly stated that from the perspective of pedagogy the approach used is still experimental and its effectiveness has been only partially validated.

The major conclusion is that developing a lab on this scale poses a tremendous number of challenges. Among the

most critical ones are: Instructor's preparation to face the diversity of projects, the need to have a full time technician to respond timely to technical problems that look minor but are critical for project continuation, cooperation with network administrators for port access, time consuming development of readable documentation, and others. Among the positive aspects were the following: use of diverse technologies (iPad, Kinect, drive-by-wire, Ethernet) drives student innovativeness; networking increased awareness of security protocols (SSH, SSL, IPsec); forcing the interaction with multiple components of the development process helps in overall broadening the professional horizons.

Probably the most important observation is that this type of projects and a lab unquestionably help in the acquisition of specific problem solving skills for embedded software development, as well as in the application of critical thinking. Nevertheless, a more targeted educational, or even psychological, research would be needed to lead to more specific conclusions. This, however, was outside the scope of this work but is a valuable goal to be addressed in the future. So is tracking student performance in upper level project based courses.

ACKNOWLEDGMENT

The authors would like to thank the following students of the FGCU's Computer Science and Software Engineering programs, who were instrumental in developing the lab: J. Carroll, C. Farrell, J. Ferreiro, M. Grojean, N. Hart, R. Ho, N. Nguyen, J. Royal, and J. Serrano.

Anonymous reviewers are gratefully acknowledged for their insightful comments to help improve paper quality.

REFERENCES

- [1] D. Uckelmann, *Architecting the Internet of Things*, Berlin: Springer-Verlag, 2011.
- [2] C. Doukas, *Building Internet of Things with the Arduino*. CreateSpace, 2012.
- [3] M.Y. Maybury, "Air Force Cyber Vision 2025," Invited Talk, *CSIRW-8, 8th Cyber Security and Information Intelligence Research Workshop*, Oak Ridge, Tenn., January 8-10, 2013.
- [4] J. Zalewski, "A Real-Time Systems Course Based on Ada." *Proc. ASEET 7th Annual Ada Software Engineering Education and Training Symposium*, Monterey, Calif., January 12-14, 1993, pp. 25-49.
- [5] J. Zalewski, "A Comprehensive Embedded Systems Lab for Teaching Remote Software Development," *Proc. CSEET 2010, 23rd Annual IEEE-CS Conference on Software Engineering Education and Training*, March 9-12, 2010, Pittsburgh, Penn., pp. 113-120.
- [6] J. Zalewski, "Lab-by-Wire: Fully Web-based Hands-on Embedded Systems Laboratory," *Proc. EDUCON2013, IEEE Global Engineering Education Conference*, Berlin, Germany, March 13-15, 2013, pp. 928-933.
- [7] J. Zalewski, "Web-based Labs for Cyberphysical Systems: A Disruptive Technology," *Proc. WCCE2013, 10th IFIP World Conference on Computers in Education*, Torun, Poland, July 2-5, 2013, pp. 89-97.
- [8] J. Titus, *The Hands-on XBee Lab Manual: Experiments that Teach you XBee Wireless Communications*. Oxford, UK: Elsevier/Newnes, 2012.
- [9] *Raspberry Pi Starter Kit*. Raspberry Pi Foundation, Caldecote, Cambridgeshire, UK. URL: <http://www.raspberrypi.org>
- [10] *What Is Beaglebone?* The BeagleBoard.org Foundation, Richardson, TX. <http://beagleboard.org/Products/BeagleBone>
- [11] *PandaBoard Getting Started*. PandaBoards Volunteers Website. URL: <http://pandaboard.org/>
- [12] *AVR STK500 User Guide*. Atmel, San Jose, Calif., URL: <http://www.atmel.com/Images/doc1925.pdf>

Author Index

A lekski, Ivan	9	M ajstorovic, Dusan	17
Anastassova, Margarita	31	Mlinarić, H.	5
B arak, Moshe	31	N ikolic, Milos	17
Benito, Jorge R. Lopez	31	P iwiński, Jan	25
Beric, Milenko	13	Pjevalica, Nebojsa	17
Bocko, Mark	13	S ouvestre, Florent	31
F rid, Nikolina	5	Sruk, Vlado	5
G onzalez, Enara Artetxe	31	Szewczyk, Roman	25
Gonzalez, Fernando	35	T emerinac, Miodrag	21
Gutiérrez, Aratz Setién	31	Teslic, Nikola	13, 21
H ocenski, Željko	9	V elikic, Gordana	13
K astelan, Ivan	13, 17, 21	Z alewski, Janusz	35
Kenny, Robert	35		
Kłoda, Rafał	25		
Kovač, M.	5		