Fully Remote Project-Based Learning of Hardware/Software Codesign

Rajesh C Panicker
Department of Electrical
and Computer Engineering
National University of Singapore
Singapore
Email: rajesh@nus.edu.sg

Deepu John
School of Electrical
and Electronic Engineering
University College Dublin
Ireland
Email: deepu.john@ucd.ie

***Abstract*—This Innovative Practice Category Work-In-Progress paper describes the innovative way in which a course on hardware/software codesign was conducted fully online during a near-lockdown necessitated by the COVID-19 pandemic. A novel remote lab setup was established in a very limited time, which was used to ensure that students were able to achieve hands-on experience through a project in spite of the course being taught fully online. Students could run the development environment on their own computers and needed to access the online lab setup only for running their program on the actual hardware, which was connected to a light server with the serial and programming ports forwarded over the internet. Lab exercises were modified appropriately to fit the constraints imposed by the remote setup while not compromising on the rigor and desirable course learning outcomes. A wiki-based platform was used for the dissemination of information, scaffolding, collaboration, as well as booking of slots to access the remote lab setup. Zoom video conferencing tool was used for consultations as well as evaluations. The results are encouraging, with students satisfied with the experience gained, without having to compromise on practical knowledge. This opens up the potential to implement such remote lab-based hands-on projects in MOOCs and continuing education scenarios to enhance student learning.**

***Index Terms*—Project-based learning, remote labs, hardware/software codesign.**

I. Introduction

Project-based learning is effective in serving the purpose of imparting theoretical as well as hands-on knowledge in a simultaneous manner. It motivates students to become active learners, and to take a higher responsibility for their own learning. The project activity provides a better context to the theoretical knowledge, and enables them to better appreciate the related issues better. This improves knowledge retention, and facilitates the integration of knowledge from various sources/courses [1]. If well-designed, project-based learning covers all six levels described in the Bloom's taxonomy of learning [2]. This will also result in engineering graduates who are better prepared to tackle the various challenges encountered in the workplace. It has been applied to a large variety of fields, including electrical/computer engineering [3], [4].

Teaching during the COVID-19 pandemic has been a challenge for educators worldwide, and many institutions had to resort to online teaching [5]. This was especially so in the early days when many were not prepared for it. Conducting project-based learning online was attempted by many educators, before as well as during the pandemic. In this paper, we explore an attempt at fully remote project-based learning in the context of embedded system education, with an emphasis on hardware/software codesign. The project makes use of Zedboard, a board using a Zynq system-on-chip that has dual ARM Cortex A9 processors, as well as programmable logic (FPGA). Students obtain hands-on experience in hardware acceleration and hardware/software codesign through a problem which is a hot topic right now - acceleration of machine learning (ML)/neural networks [6]. Conducting such a project-based course presents many unique challenges. Some of these were overcome by a combination of appropriate tools, scaffolding techniques, and most importantly, the use of a custom remote lab setup. A number of remote lab setups have been reported in the literature for teaching embedded systems, including those which are FPGA-based. Most of them involve simple interactions such as LEDs and buttons [7]. A bitstream for FPGA is uploaded to the cloud, and real or virtual switches and LEDs are remotely operated. This, however, does not give direct control of the FPGA. Debugging using techniques such as single-stepping, an essential part of hardware/software codesign process, is not possible [8]. Hence, in this paper, we propose a novel setup which gives students more or less full control of the FPGA, except those features involving physical access while the development environment is run local on their own computers.

The rest of the paper is organized as follows: Section

II gives details about the remote project-based learning setup and support channels. The results of feedback surveys and student performance are given in Section III. Section IV concludes the paper and describes the future directions.

II. Remote Project-Based Learning

A. Course Details

The course EE4218 is meant to give students the background and experience in designing systems involving custom hardware accelerators. 3rd and 4th year undergraduate students in electrical engineering and computer engineering form the target audience. Students are expected to have a reasonable background in digital systems and microprocessors as a pre-requisite for this course. Students work in pairs implementing a complete hardware/software codesigned system, learning collaboratively in the process. Students were expected to come up with 3 designs - a pure software version, and two hardware/software codesigned versions. For the latter two, one had to be a hardware system implemented from scratch using a hardware description language (HDL), whereas the other had to be generated using a state of the art High-Level Synthesis (HLS) tool which automatically generates hardware from a high-level language. More details of the course were presented in [9]. It is usually offered only in semester 1 (fall semester). However, as a number of students had their internships and exchange programmes cancelled due to the pandemic, it was offered in semester 3 (summer semester) as well to avoid possible delays in graduation. Semester 3 was a short semester, compressing what is usually done over 13 weeks into 5-6 weeks.

B. Remote Lab Setup

Running a course involving project-based learning during the uncertain times of an almost-lockdown posed a significant and unexpected challenge. When the decision to offer this course was made, it was hoped that the hands-on activity could still be conducted face to face, with perhaps a reduced number of students per session. When this proved to be infeasible, cloud-based solutions were considered as an alternative. However, they were costly or did not allow low-level programming as was required for the course. Another option was to ship boards to students – but there were not enough boards for each student apart from logistical and time constraints. Making the labs simulation-only would have affected the students' experience substantially, especially given the essence of the course which is to have a custom hardware interacting with a custom software. The existing tools do not simulate such a scenario to any reasonable degree of realism. Many remote access facilities (e.g. Nimbix cloud) operate by having a virtual computer connected to the equipment (FPGA in our case), which can be accessed remotely through a remote desktop. This, however, lacks a native feel for the user, and is possibly demanding on the host processor, especially if multiple sessions are involved.

Hence, a setup which involves remotely accessing the board without full remote access to the computer itself by forwarding just the programming and serial ports was implemented. This allowed for the server to be very lightweight - an old PC running Linux with 5 FPGAs connected to it was used as the server, and there was practically little difference for students between having the board physically connected or remotely connected, giving them a native feel. The setup is shown in Fig. 1. The programming interface of each board is forwarded to a separate TCP/IP port, by running multiple instance of the Xilinx hardware server program [10] with the cable ID used as a filter. Similarly, the USB-UART interface of each board was also forwarded to separate TCP/IP port using the ser2net program [11], with each port identified using the USB ID.

Students could connect to a board by setting the programming and console connections (via a raw TCP/IP socket) to respective ports after booking a particular board on the wiki. They were required to release the programming and serial connections once their slot was over. During lab sessions, only those who had signed up for the session could use it in 20-minute slots. During other times, booking 1-hour slots was permitted. Students were not allowed to place advance bookings for more than 3 slots, for no more than two slots in a day, and no more than 1 booking per team for a slot.

C. Support Channels

Some of the labs required modifications to fit the constraints imposed by the remote labs. Lab 1 used to be an introduction to software design. Lab 2 used to be about hardware design. In Lab 3, the two were combined into a hardware/software codesigned system. Lab 2 involved a hardware design which required physical interaction with the board using switches and LEDs. This was not feasible with remote labs and was thoroughly modified. Hence, Lab 2 was done first, purely in simulation. Self-checking testbench templates were provided so that students could test the hardware design thoroughly in simulation. Usually, the integration between the hardware and software is among the most difficult aspects of the labs. The project was also streamlined with more support provided to work around the limitation of time in the compressed semester where students practically get two weeks for the project, as opposed to five weeks in a normal semester.

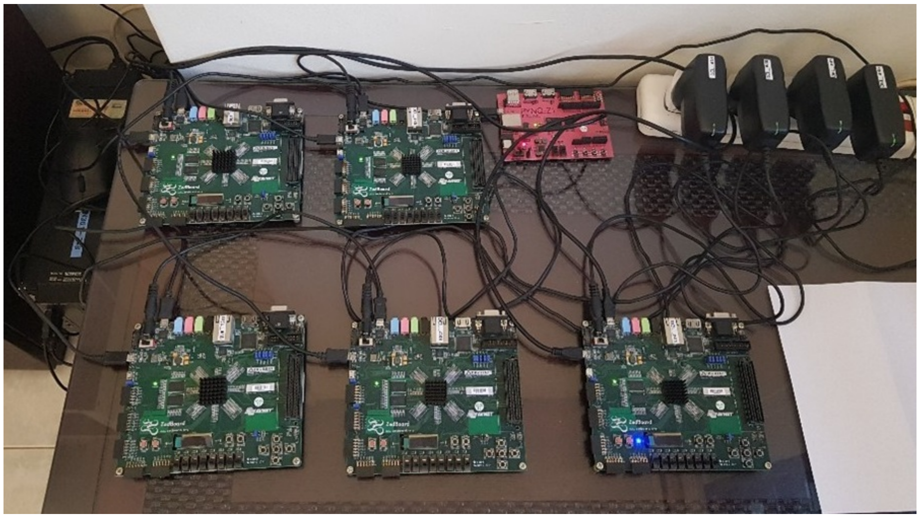

Fig. 1: Photograph of the remote lab setup

In a normal semester, providing support for the labs is easier, through face to face sessions. This is certainly more challenging in an online teaching scenario. Hence, support channels such as Wiki and Zoom were used, the former for easy dissemination of information and collaboration [12], and the latter for virtual consultations. The wiki is accessible at https://wiki.nus.edu.sg/display/ee4218. While a wiki has been used for this course before, it was more extensively used as a support channel in the compressed semester. A total of about 200 comments were posted on the wiki during the semester. The use of a wiki has the advantage of facilitating discussions which are visible to and benefit everyone, and corrections could be made instantaneously. Wiki was also used as the FPGA booking mechanism, where bookings could be made in a transparent manner, and was used for creating the project evaluation schedule. In addition to two lab slots per week conducted over Zoom, there were regular Zoom-based consultation sessions - a total of eight sessions over two weeks. A flipped classroom approach [13] was used, where lecture videos from the previous semester were released to students before the course commenced, so that students could make effective use of their time. The lecture hours were used as interactive sessions to discuss problems, lab briefings, as well as current technological trends. A guest speaker from the industry was also invited, he spoke about the state of the art in Network-on-Chip technology. Three summative quizzes ensured that students kept up with the lectures. The final exam format was open-book to eliminate the need for rote-learning.

III. Results and Discussions

A. Student Performance

The remote lab setup effectively addressed the sudden constraints and was well-utilized – a total of 810 slots were booked across 18 teams, averaging 50 slots/team. Only 11 slots over the entire 5-6 weeks had 100% utilization, which means that much more often than not, students were able to access the boards at will. Additionally, in spite of the project being done over 2 weeks in Semester 3 (remote labs) as opposed to 5 weeks in Semester 1 (normal face to face labs), all teams managed to meet the basic requirements of the project (in Semester 1, two teams had not managed to meet the basic requirements).

The university conducts formal surveys at the end of each semester to evaluate learning outcome accomplishment as well as teaching effectiveness for every course. There were 18 respondents out of a class of 43 students in Semester 1, and 13 out of 36 students in Semester 3. The course has 3 learning outcomes [9], and students were required to rate their own achievement level, the results of which are summarized in Fig.2. The proportion of students who felt that course learning outcomes were achieved all the time increased by 25% from Semester 1 to Semester 3. The individual teaching feedback survey results for the instructor also improved from Semester 1 to Semester 3, with a perfect score of 5/5 in Semester 3, as shown in Fig. 3. The percentage of students who met each of the three requirements of the project to a reasonable degree is shown in Fig. 4. From all these results, it is clear that the various performance metrics improved from the previous semester, in spite of the lack of face to face interaction.

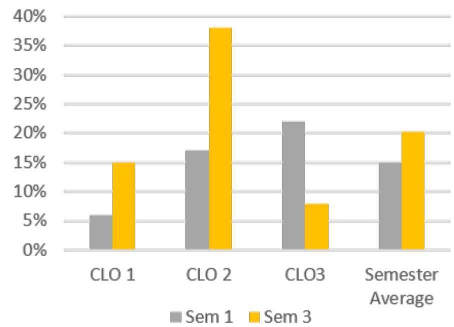

Fig. 2: Improvements in % achieved Course Learning Outcome (achieved "All the time")

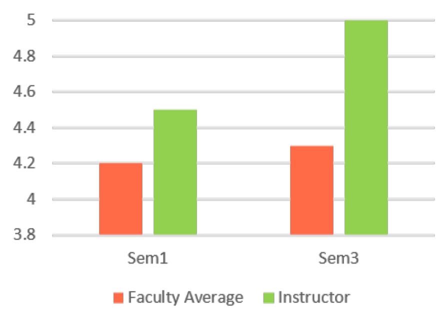

Fig. 3: Improvement in Teacher Effectiveness

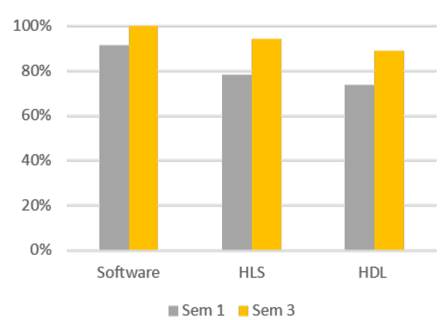

Fig. 4: % of teams that met the three requirements

The project, which is on a cutting-edge topic, was motivating to many students. The students found the labs challenging, requiring a good amount of debugging. Yet, they found the teaching-learning process in the course enjoyable and helped immensely in their learning, in spite of it all being done remotely. Students' comments from the qualitative feedback surveys in Semester 3 indicate that "the labs are [sic] very fun and good", and that "give [sic] the current situation we are unable to have physical Lab but, the prof did exceptionally well with the remote FPGA and all the consultation sessions". Students also felt "they (labs) were good and aided a lot in the learning process" and that the lecturer succeeded in "giving meaning to the concepts taught by bringing in industrial relevance".

B. Challenges and Limitations

Since there was no access control mechanism in place for the remote lab, it was reliant on students following the rules. In some cases, students used the boards without having made a booking, so those with a legitimate booking could not access it. This required the instructor to force a disconnection manually. In some instances, simultaneous multiple access caused the boards to hang, requiring a manual power cycling. This requires physical access to FPGAs and could not have been done if the system was set up in a place which is not easily accessible to the instructor. There was also the risk of hacking from outside as there were no real security mechanism in place, though it did not happen for these five weeks.

The biggest limitation of this study lies in the fact that the way the course was conducted in Semesters 1 and 3 changed, but the external circumstances also changed. As a result, the improvement in the course feedback cannot be attributed solely to the change in the way the course was conducted; the lockdown could have affected the perception of remote learning in students.

IV. Conclusions and Future Work

A fully remote project-based learning for hardware/software codesign was accomplished, with students expressing high degree of satisfaction and perception of having gained knowledge. The success of the remote teaching opens up the possibility of scaling the system to support MOOCs, where hands-on aspects are often not incorporated, especially if it involves hardware or hardware/software codesign.

A video feed from the FPGA boards can perhaps be implemented as well, to give students a better feel of the system they are designing/interacting with. A better security, booking, access control and enforcement mechanism is imperative for a larger-scale implementation. More work needs to be done from a technological and pedagogical perspective to measure the effectiveness and improve student learning in online labs.

Acknowledgment

This work was supported in part by the Irish Research Council under the New Foundations Scheme.

REFERENCES

[1] H. A. Hadim and S. K. Esche, "Enhancing the engineering curriculum through project-based learning," in *32nd Annual Frontiers in Education*, vol. 2, 2002, pp. F3F–F3F.

[2] B. S. Bloom, M. Englehart, E. J. Furst, W. H. Hill, and D. R. Krathwohl, "Taxonomy of educational objectives: Handbook i," *Cognitive domain. New York: David McKay*, 1956.

[3] N. Hosseinzadeh and M. R. Hesamzadeh, "Application of project-based learning (pbl) to the teaching of electrical power systems engineering," *IEEE Transactions on Education*, vol. 55, no. 4, pp. 495–501, 2012.

[4] J. C. Nwokeji and P. S. T. Frezza, "Cross-course project-based learning in requirements engineering: An eight-year retrospective," in *2017 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2017, pp. 1–9.

[5] J. Daniel, "Education and the covid-19 pandemic," *Prospects*, vol. 49, no. 1, pp. 91–96, 2020.

[6] L. K. John, "Machine learning accelerators and more," *IEEE Computer Architecture Letters*, vol. 39, no. 05, pp. 4–5, 2019.

[7] F. Morgan, S. Cawley, and D. Newell, "Remote fpga lab for enhancing learning of digital systems," *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, vol. 5, no. 3, pp. 1–13, 2012.

[8] P. Orduna, L. Rodriguez-Gil, J. Garcia-Zubia, I. Angulo, U. Hernandez, and E. Azcuenaga, "Labsland: A sharing economy platform to promote educational remote laboratories maintainability, sustainability and adoption," in *2016 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2016, pp. 1–6.

[9] R. C. Panicker, A. Kumar, and D. John, "Introducing fpga-based machine learning on the edge to undergraduate students," in *2020 IEEE Frontiers in Education Conference (FIE)*, 2020, pp. 1–5.

[10] "Xilinx Hardware Server", Accessed May. 17, 2021. [Online]. Available: https://www.xilinx.com

[11] "Ser2net project page", Accessed May. 17, 2021. [Online]. Available: http://ser2net.sourceforge.net/

[12] P. D. Duffy and A. Bruns, "The use of blogs, wikis and RSS in education: A conversation of possibilities," in *Online Learning and Teaching Conference, Brisbane*, 2006.

[13] B. Tucker, "The flipped classroom," *Education next*, vol. 12, no. 1, pp. 82–83, 2012.