Work in Progress – Bring Green Computing to CS Core Curriculum with a Portable Lab

Kai Qian, Chia-Tien Dan Lo
Southern Polytechnic State University, kqian@spsu.edu, clo@spsu.edu
Li Yang
University of Tennessee at Chattanooga, Lyang@utc.edu

Abstract – Underdevelopment is novel reusable labware for computer science (CS) core curriculum. This labware contains a relatively inexpensive portable Wireless Sensor Network kit in a Box (WinBox), along with a variety of predesigned labs ranging from fundamental to advanced concepts in CS core curriculum. Exemplary projects are depicted to show how this WinBox can be reused in several CS core courses. With its portability, the WinBox makes feasible to offer on-line lab intensive courses. Much of our effort is made on inculcating CS students in greening computing due to its increasing importance. By bringing green computing experience to students and instructors through the hands-on projects, we aim to promote students’ innovation in pervasive green computing. This pedagogical model harnesses a wide range of wireless hardware/software co-design skills in CS education to address a need in embedded system workforces. This affordable WinBox renders an educational resource for budget limited institutions.

Index Terms - capstone project; wireless sensor network, product-based; green computing; software engineering process;

INTRODUCTION
Computing technology can help us reduce energy consumption, cut carbon emission of infrastructure and environment, and manage green renewable energy [1-2]. Green computing has become a hot topic nowadays and it has a huge potential and demand in the next decade [4]. However, it has yet become a part of CS curriculum. The rapid growth of green computing has resulted in a shortage of professionals and scientists in the areas such as its vital components: wireless sensor network and pervasive computing. Due to the lack of CS curriculum in this important field [3], and there is a little room for adding additional courses in CS curriculum for wireless sensor network (WSN) and smart pervasive computing, we propose to transform CS core courses by bringing green computing into their lab intensive projects. Students can gain hands-on learning experience in a variety of CS core courses including computer architecture, operating systems (OS), network, embedded systems, and capstone.

LABS IN WINBOX
The proposed new learning model is delivered in a WinBox with a series of semi-conducted projects that bring green computing experience to CS students and instructors through well structured real-world product-based hands-on hardware/software co-design projects to promote students’ innovation in green computing activity. The WinBox is a rapid prototyping platform, which contains a set of MCU based nodes: a USB dongle and two battery powered wireless sensor nodes with ZigBee support. This complete computer system is well qualified for projects in CS curriculum. Yet, its low cost and highly portable properties provide the following advantages. 1) It makes it possible to offer lab intensive course in resource limited institutions either lacking of faculty expertise or lab maintenance/initiation budgets. 2) It promotes lifelong learning at any time any place. 3) On-line lab intensive courses are feasible. Moreover, the integration of latest hardware and software allows students to learn the emerging technology, and to be well-trained for the urgent workforce.

Based on the WinBox, a suite of CS core curriculum modules is developed, including tutorials, and semi-constructed labs or projects with an emphasis on green computing. Therefore, the WinBox can be reused in several CS core courses. Most of the real-world hardware/software co-design projects are modular, which is easy to be tailored for a special need in adopted institutions. Furthermore, a GUI based cross-compilation IDE shortens students’ learning curve and facilitates project development, debugging, and testing.

SAMPLE PROJECTS
Preliminary projects based on this pedagogical model for CS courses such as capstone, computer architecture, operating system, and networking can be found in http://cse.spsu.edu/clo/winbox. The simple 16-bit MSP430 MIPS microcontroller in the WinBox and its supported software allows students to learn fundamental computer systems with projects from basic assembly programming to advanced topics, e.g., power-aware software design. A sample group project based on the WinBox for a CS capstone course is illustrated as an exemplary learning module. This project asks students to design a smart utility meter infrastructure to measure power consumption, and make detailed electricity usage data available online via a distributed wireless network of smart sensors. The data can be used further to monitor and analyze power usage, and possibly detect energy leakage should an abnormal readings were found. Figure 1 shows the configuration of this project.

October 27 - 30, 2010, Washington, DC

978-1-4244-6262-9/10/$26.00 ©2010 IEEE

40th ASEE/IEEE Frontiers in Education Conference
F2F-1
In terms of green computing, for example, an assembly project that exercises the five low power modes provided in the MSP430 MCU, is suitable for a computer architecture (organization) curriculum. In this project, students will learn power saving mechanisms in developing programs that consume less power. Specifically, interrupt service routines will be implemented to revive MCU from low power modes.

A power-aware scheduling project is being designed for the operating system course. This project introduces green computing in the OS level. Students will modify an open source project, B.lu BOS, http://blubos.sourceforge.net/, by integrating low power modes in its process scheduler. The idea is to deactivate MCU when there are no waiting jobs. A simple “sell-like” interface via I/O interrupts will be used to activate a task. This interface serves a testing platform for the power-aware scheduler.

Since the WinBox is based on wireless sensor network, it is indeed a complete learning platform for a computer network curriculum. Instead of a complex protocol stack, the simple Zigbee protocol renders an opportunity for students to learn urgent important topics such as green computing in sensor network. A simple web server is designed as a networking project. This web server is deployed in each sensor node in charge of responding information collected in its ambient environment. This server is designed in power saving mode while waiting for requests. This is crucial to prolong such battery powered operating devices.

EVALUATION

A survey from our capstone students shows that they favor the “learning by doing” approach and are excited after completing their hands-on projects on the portable WinBox. They greatly appreciate the opportunity to apply a wide range of the learnt knowledge, hands-on hardware/software co-design experience from the program in developing their creativity and self-learning potential. They feel confident after developing a variety of projects in the green computing domain, and will be able to participate in national green computing design competitions to demonstrate their creativity and innovation. A majority of students are ready for the career in embedded system development.

CONCLUSION

The new pedagogical model implemented in a highly portable reusable WinBox leverages CS core curriculum with latest technologies (WSN), and emerging computing trend (green computing). This transformation is crucial in several facets: 1) its relative inexpensive characteristic promotes broad adoption especially in resource limited institutions, 2) its highly portable attribute makes on-line lab intensive curriculum possible, 3) it is reusable in several CS core courses, and 4) it serves a platform for an integration course, e.g., operating system and computer architecture. The preliminary assessment shows that this new model benefits students in exercising multi-disciplinary knowledge and skills for the emerging workforce, promotes lifelong learning, and helps in faculty development.

ACKNOWLEDGMENT

This work is partially supported by the National Science Foundation under award: DUE-0942097. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES


AUTHOR INFORMATION

Kai Qian Professor of Computer Science and Software Engineering, Southern Polytechnic State University, Atlanta, Georgia, kqian@spsu.edu.

Chia-Tien Dan Lo Associate Professor of Computer Science and Software Engineering, Southern Polytechnic State University, Atlanta, Georgia, clo@spsu.edu.

Li Yang Professor of Computer Science, University of Tennessee at Chattanooga, Chattanooga, Tennessee, Li-Yang@utc.edu