Integrating Embedded Computing Systems into High School and Early Undergraduate Education

Bridget Benson, Arash Arfaee, Choon Kim, Ryan Kastner and Rajesh Gupta

***Abstract*—** Early exposure to embedded computing systems is crucial for students to be prepared for the embedded computing demands of today's world. Unfortunately, this exposure often comes too late in the curricula to make a meaningful impact. This paper describes our experience with integrating embedded computing systems education into high school and early undergraduate curricula to give students that needed early exposure. We believe that our initial foray has been very successful, thus we are currently pursuing further incorporation of embedded systems into our undergraduate curriculum.

***Index Terms*—Embedded Systems Education, High School and Undergraduate Curricula**

I. Introduction

Early exposure to embedded computing systems is a crucial part of high school and undergraduate computer science education. Unfortunately, this exposure often comes too late in the curricula to make a meaningful impact. We are taking a multipronged approach to provide young students opportunities to use and understand embedded computing systems. This includes a four week course to expose high school students to embedded systems as part of the UCSD COSMOS program [1] as well as integration of embedded systems programming into the sophomore-level "Computer Organization and System Programming" course.

The COSMOS program (California State Summer School for Mathematics and Science) is designed specifically for talented and motivated high school students with demonstrated interest and achievement in math and science. We designed the course around six embedded systems labs that made use of the Cypress Semiconductor CY3214 [2] and CY3209 [3] Development kits. These labs were supplemented with lectures on embedded systems concepts and the C programming language to give the students the tools they would need to complete the lab assignments. Once the students completed all six lab assignments, they developed their own embedded systems group projects with the use of one or two development boards.

We are also integrating a similar experience into the UCSD CSE 30 "Computer Organization and Systems Programming" class. This is a required course for all UCSD computer science undergraduates that teaches the ability to program in assembly, reinforces C programming concepts and provides an introduction to computer organization. We have started to augment the class with more experiences related to programming embedded systems boards, which include labs to teach the basic principals for using and programming the development boards as well as an open-ended final project of the student's choosing.

This paper describes our initial experiences in both the COSMOS program and the CSE 30 class. We start by discussing our COSMOS program including content of the lectures (Section II), the lab assignments and the topics of our students' group projects (Section III). We then describe the feedback we received from students, parents, and faculty on the course as well as lessons we learned from teaching the course (Section IV). Finally, we discuss our early experiences with integrating embedded system programming projects into CSE 30 (Section V) and conclude in Section VI.

II. COSMOS Lectures

This section describes the content of the lectures presented to the high school students during the first two weeks of the COSMOS summer course. These concepts were presented early on in the course to give students the tools and understanding they needed to complete their lab assignments and begin their own projects. The lecture period during the last two weeks of the course was divided between giving more lab time to the students to work on their projects and giving students tours of labs at UCSD and an industry tour of Sony. We divided the lectures into two major topics: Embedded Systems Concepts and C Programming.

A. Embedded Systems Concepts

We presented a series of lectures that introduced students to embedded systems and their uses in today's world. We described what embedded systems are and how they are different from other computing systems and presented various examples of some embedded systems we use in everyday life. We then presented a few lectures describing some communication protocols used in many embedded systems, focusing on I2C and USB communication (as the students would use these communication protocols in their lab assignments), and presented a lecture on Cypress' CapSense technology. We concluded our embedded systems concepts lecture series with a guest lecture from Professor Yoav Freund describing the use of embedded systems in his work on real-time face detection.

B. C Programming

The C programming language is used for programming many embedded systems platforms (including the CY3214 and CY3209 used for the COSMOS course) because of its code efficiency and readability, its reduced overhead and development time, and its access to low-level control. Thus it was essential for students to have some C programming knowledge in order to implement their lab assignments and group projects.

Due to a constrained time frame and a diverse range of programming experience (some students had taken an Advanced Placement programming course in high school while other students had never seen a piece of code in their lives), we presented only the programming concepts necessary for completion of the lab assignments and group projects. These programming concepts were presented in the form of six 45 minute lectures each followed by a 45 minute 'workshop.' During the workshop, students were given a worksheet related to the lecture material and worked with one another and the teaching assistants to complete the worksheet to better understand the material. Those students who completed the worksheets early were given more challenging problems and were asked to help other students who were struggling. The topics of the six lectures included, Number Systems, Variables and Functions, Libraries, Control Structures (If, Else, While, For), Bit Manipulations, and CapSense API. We concluded the programming lectures with a guest lecture by one of our undergraduate lecturers, Rick Ord, who gave the students an opportunity to practice some of the programming concepts by writing a simple "Let's Make a Deal" simulator (based on the TV game show).

III. COSMOS Labs

We designed the COSMOS course around six embedded systems labs that made use of the CY3214 and CY3209 Development kits donated by Cypress Semiconductor. These labs gave the students a good introduction to some capabilities of the Cypress boards to give them ideas for their group projects. This section describes the objectives for each lab as well as a brief description of the students' group projects.

A. Labs

The six lab assignments were adapted from Cypress tutorials. Each lab built on concepts learned in earlier labs and introduced students to different embedded systems concepts.

1) *Pushbutton and Lights:* The objective of this assignment was to introduce students to the CY3209 development board and Cypress' PSOC Designer 5 development environment and PSOC Programmer v. 3.0. This lab made use of PSOC Designer's System Level Design flow, which uses a GUI interface, so no C programming was required. Students learned the design flow of a PSOC board by designing a simple project to use a push button input to control LED outputs.

2) *I2C Slave and Master Communication*: The objective of this assignment was to learn how to design an I2C Master and Slave communication between two PSOC modules. Students made use of the CY3209 board and once again used the GUI interface to program one quadrant as the Master and another quadrant as the slave. They had to adjust the potentiometer on the slave and observe the master's multi-digit LED display and the slave's LED change in response to the changes in the potentiometer.

3) *Pushbutton Counter Implementation:* The objective of this assignment was to learn how to write C code for an embedded environment and get it to run on the CY3214 development board. Thus the students used the PSOC Chip-Level design flow as opposed to the system-level design flow. The students specifically had to write code to count the number of pushbutton inputs and alter the LCD display and LEDs based on this count.

4) *USB Interface Design:* The objective of this assignment was to learn how to design a USB interface between PSOC CY3214 and a Personal Computer (PC). The students had to click buttons on a PC client program and see the corresponding LEDs on the CY3214 light up. Students could also manually turn LEDs on or off on the CY3214 and see the corresponding buttons light up on the PC client program.

5) *CapSense Input Design:* The objective of this assignment was to learn how to incorporate CapSense button input on the CY3214. The students used the CapSense Wizard Utility to program the CapSense slider and buttons and then wrote C code to have the LCD show "Button Pressed" when the CapSense button was pressed and show a sliding bar graph on the LCD as the student moved his finger across the CapSense slider.

6) Wireless USB (WUSB) Master and Slave Design: The objective of this assignment was to learn how to design a wireless USB Master and Slave communication protocol between two CY3209 modules. This lab assignment was very similar to Lab #2, in that the students adjusted the potentiometer on the slave device to observe changes in the master's multi-digit LED display and slave's LED.

These six lab experiments provided the students with a basic understanding of some of the capabilities of their Cypress boards and a hands-on experience with simple embedded systems. The lab assignments allowed the students to become familiar with the PSOC development environment and design flows to enable them to complete their group projects.

B. Group Projects

Starting in the third week of the course, the students were asked to come up with a group project (made up of 2-3 students) that made use of the CY3214 and/or the CY3209 board. We asked that their project would be fun and challenging for them, but at the same time doable in the less than two-week time period they had for completion. We informed the students they could choose projects that ranged in complexity from modifying their lab assignments to fully creating their own design. This section describes the nine projects the students implemented.

1) Tilt Controlled Vehicle: Our most advanced group of students designed a remote controlled vehicle which made use of three CY3209 boards. Two boards made up the 'body' of the car and were used to control the servo motors that powered the vehicle. The third board, connected to the 'body' via WUSB, was used as the vehicle controller. The students used the accelerometer on the controller board to control the direction and speed of the vehicle.

2) *Electronic Keyboard:* The students made a single octave keyboard out of copper tape mounted on Plexiglas. The copper tape 'keys' acted as their own capacitive sensors. They connected the copper keys to the CY3214 board and programmed the board to play different notes when different keys were pressed.

3) *Relaxation Goggles*: The students made a pair of relaxation goggles out of safety goggles, six LEDs, duct tape and wire. They connected the LEDs on the goggles to the CY3214 board and allowed the user to control the rate at which the LEDS blinked by selecting different buttons on the board. These different rates corresponded to the dominant frequency of brain waves during states of deep sleep, rest, relaxation, and alertness.

4) *Text Messaging System:* The students built a simple text messaging system between two CY3214 boards connected via I2C. The students used the CapSense buttons to scroll through and select letters on the LCD screen and then used a pushbutton to send the message to the other board.

5) Growling Bear: The students put a capacitive sensor inside a small stuffed bear and connected the sensor to a CY3214 board. When a student stroked the bear, the bear would make a happy growling sound. If the bear was left alone for too long, it would make an angry growling sound.

6) *Traffic Light Controller*: The students connected a green, yellow, and red LED to two CY3214 boards to simulate a traffic intersection. Initially the traffic light was green on one board and red on another board. When a 'car' approached the red light (a finger moved along the CapSense slider) the green light would cycle to yellow to red and the red light would turn green. After the car went through the intersection, the lights would cycle back to their original state.

7) Light Dimmer: The students used the CapSense slider on the CY3214 board to act as a light dimmer as they slid their finger across the slider to the right, a row of LEDs would become brighter and as they slid their finger across the slider to the left, the row of LEDs would become dimmer.

8) Song Selector: The students used the CapSense slider to simulate selecting a song on an iPod. As they slid their finger across the CapSense slider, different song title names would appear on the LCD.

9) *Stop Watch:* The students created a stop watch on the CY3214 board. One push button acted as the start/stop button and another as reset. The hour, minute, second count appeared on the LCD.

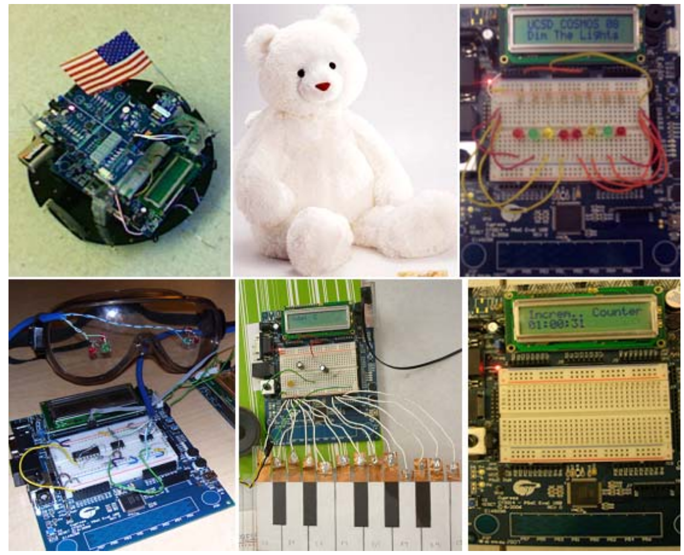

Figure 1. Student Group Projects. From upper left to lower right, Tilt Controlled Vehicle, Growling Bear, Light Dimmer, Relaxation Goggles, Electronic Keyboard, Stop Watch

Each group was required to demo their project and create a power point presentation and poster of their project. They presented their projects to the entire class on the last day of the course and displayed their posters for the COSMOS Open House for their parents and community members. The tilt controlled vehicle group also presented their project to the entire UCSD COSMOS program at COSMOS graduation.

IV. Feedback and Lessons Learned

The feedback we received from the students, parents, faculty, and staff and participated in the COSMOS program was uniformly positive. The students were very proud of the work they did and the faculty, staff, and students were all very impressed with the innovation and quality of work the students produced in just a 4-week period! Many parents expressed sincere thanks for offering the COSMOS program because they themselves knew nothing about embedded systems and could not find any other avenue to foster their child's interest in the field. Parents were also very pleased to give their child a setting to find like-minded motivated friends that shared a common interest in academic achievement and hands-on engineering.

Though we were very pleased with the outcome of the course, we did learn some valuable lessons along the way about how to better teach the course in the future.

1) *Programming Experience*: Students that had no programming experience prior to the course struggled a lot more with the lab assignments and the projects. Although the lectures and workshops on programming were useful, it was impossible to expect a student to be able to write a full C program by the end of a two-week period. Though the

students still enjoyed the course, trying to learn programming and design small embedded systems at the same time was a bit overwhelming. Thus, some basic programming knowledge should be a pre-requisite for the course so students can focus more on embedded system design rather than on semantics of programming.

2) *Lab Assignments:* The first few labs contained step by step instructions on how to implement the desired embedded design. We then asked the students to try reading through online tutorials on the third assignment to give them practice on learning how to find references and help for projects on their own. This turned into a bit of a disaster because the manuals and application notes the students were able to find often contained too much jargon making the document inaccessible to neophytes. We quickly determined that we had to keep all lab assignments at a level students could understand to keep them interested and not scared of the assignments.

3) *Lectures:* We determined that limiting the lecture content and allowing for more hands-on work with the development boards proved more interesting and useful to the high school students. Young students have a very short attention span and trying to get material across to them in a conventional college lecture setting was difficult. Students were much more excited about going to lab and 'playing' with the boards and were very quick to pick up concepts they experienced first-hand through experimentation rather than through lecture content. Perhaps the next COSMOS course should have more lab assignments and fewer lectures.

V. Initial Undergraduate Experiences

We are carefully integrating the embedded systems experience into our Computer Science and Computer Engineering undergraduate curriculum. We are doing this by following our tremendously successful COSMOS program. We started this process in Winter Quarter 2009 by offering a special section of CSE 30 "Computer Organization and Systems Programming" class.

This class was a first step into introducing embedded systems material into an early level undergraduate class at UCSD. It included several lectures related to the Cypress boards including a demo by Patrick Kane, the Cypress Educational Liaison. This demo was held in lieu of a class and gave the students a hands-on lab experience using and programming the CY3214 boards. More specifically the students learned how to program the CY3214 board as a USB controller, allowing the on-board joystick to control the cursor on a computer connected to the CY3214 board. The second portion of the lab demonstrated the push button and CapSense functionality. The final programming project of the course largely mirrored the final project for the COSMOS program; the students were tasked with developing a prototype embedded systems of their choosing using the CY3214 board.

Our plans for future versions of this class involve further integration of the embedded systems boards into the course material. This will include weekly labs using this CY3214 boards, held during the initial weeks of the class. This will provide the students an early introduction to embedded systems boards, which we hope will both provide excitement and inspiration for their final project.

We eventually hope to teach both C programming and assembly programming using the Cypress boards. The microcontrollers on the CY3214 and CY3209 use a simple 8-bit assembly language. Since this class feeds into a Computer Architecture class, we feel that teaching in a more familiar 32-bit assembly language (e.g. MIPS) is more useful. We note that Cypress is developing more advanced development boards using an ARM processor. We are excited about the opportunity of using this board to teach both C and assembly, which we view as the ideal scenario.

VI. Conclusion

We described our early experiences of integrating embedded computing systems into both high school and early undergraduate curriculum. We learned that having some programming experience is necessary to better appreciate and understand an embedded systems course and that students are more likely to get enthused about embedded systems through hands-on labs and projects rather than through lecture material. We believe that our initial foray has been very successful and we are currently pursuing further incorporation of embedded systems into undergraduate curriculum. This includes the introduction of an advanced undergraduate (junior/senior level) capstone class to build upon the experiences the students received in CSE 30. This new class has been approved by the university and will be offered for the first time Winter quarter 2010. Furthermore, the embedded systems section CSE 30 will be made a permanent part of the curriculum for all Computer Science and Computer Engineering.

Acknowledgments

We would like to thank Cypress Semiconductor and especially Patrick Kane, the Cypress Education Liason, for their support of COSMOS and CS30. We thank guest speakers, Rick Ord and Dr. Yoav Freund, for taking time out of their schedules to give presentations to the students. We would also like to thank all the COSMOS students and staff, especially Shirley Miranda, Teacher Fellow, and Joe Lunsford, Lab Manager, for helping to make the COSMOS program a great success.

References

[1] UCSD Cosmos Program
http://www.jacobsschool.ucsd.edu/cosmos/
[2] Cypress Semiconductor PSoC CapSensePLUS with USB Evaluation Kit (CY3214 Development Kit)
http://www.cypress.com/
[3] Cypress Semiconductor CY3209 Express Evaluation Kit
http://www.cypress.com/