2018 ASEE Annual Conference & Exposition
June 24–27, 2018 Salt Palace Convention Center Salt Lake City, UT

Paper ID #22506

Interdisciplinary Embedded Systems Design: Integrating Hardware-Oriented Embedded Systems Design with Software-Oriented Embedded Systems Development

Ms. Cynthia C. Fry, Baylor University

CYNTHIA C. FRY is currently a Senior Lecturer of Computer Science at Baylor University. She worked at NASA's Marshall Space Flight Center as a Senior Project Engineer, a Crew Training Manager, and the Science Operations Director for STS-46. She was an Engineering Duty Officer in the U.S. Navy (IRR), and worked with the Naval Maritime Intelligence Center as a Scientific/Technical Intelligence Analyst. She was the owner and chief systems engineer for Systems Engineering Services (SES), a computer systems design, development, and consultation firm. She joined the faculty of the School of Engineering and Computer Science at Baylor University in 1997, where she teaches a variety of engineering and computer science classes, she is the Faculty Advisor for the Women in Computer Science (WiCS), the Director of the Computer Science Fellows program, and is a KEEN Fellow. She has authored and co-authored over forty-five peer-reviewed papers.

Mr. Steven Patrick Potter

Interdisciplinary Embedded Systems Design: Integrating Hardware-Oriented Embedded Systems Design with Software-Oriented Embedded Systems Development

Abstract

The rapid proliferation of embedded systems designs has opened opportunities for both computer engineers and computer scientists in recent time, and promises to do so in the foreseeable future. With the advances in this field, however, the need for a more multi-disciplinary approach to embedded systems education has arisen, as modern embedded systems design relies heavily on the integration of both hardware and software design. Due to the diverse backgrounds that are becoming not only recommended, but also required, providing students with an educational experience of the holistic design of a modern embedded system is becoming a bigger challenge. Both the Electrical and Computer Engineering curricula and the Computer Science curricula are filled to overflowing with required courses as well as upper-level technical electives to provide opportunities to explore the application of students' knowledge in depth in particular fields. How then do we develop an Embedded Systems course that provides this holistic design in the limited amount of time afforded by a quarter or semester course? One approach that has been tested in Baylor University's School of Engineering & Computer Science (ECS) is the integration of the Embedded Systems Design course taught in Electrical and Computer Engineering, with the Embedded Systems course taught in Computer Science. [1]

This paper will document the original courses and their prerequisite coursework, the path toward integration of the group design project, the prototype testing of this approach conducted in 2017, an outline of the instruction provided to the students, and the assessment of the interdisciplinary approach to the project phase of both courses.

Introduction

Over the past decade computer technology has become ubiquitous in our everyday lives. From digital camera, to the embedded systems that make up current vehicles, to automated highways, to home security systems, to automated household appliances, to robotic manufacturing, to integrated medical devices, to communications systems, even the term "embedded systems" has transcended previous definitions to now embody any "engineering artifact involving computation that is subject to physical constraints." [2] These are examples of some of the implementations of embedded systems design that are pushing existing technology to its limits, and going past many existing applications, and forcing us to rethink our process of teaching this subject.

In today's automotive industry, as each new vehicle design receives yet another control unit, "software complexity escalates to the point that current development processes and tools can no longer ensure sufficiently reliable systems at affordable cost." [3] In this challenge lies an opportunity for the disciplines of computer science and electrical engineering to recognize the need for professionals who are able to bridge the divide between the disciplines and "integrate computation and physicality for the bottom up," using non-traditional design methods. [4],[5]

In Baylor University's School of Engineering and Computer Science, two existing courses in embedded systems, each taught from differing perspectives, have joined to integrate software and hardware design and implementation in their group projects. This paper will discuss the existing courses, the design and development of multi-disciplinary group projects, implementation of these projects, and lessons learned in the beta test of this design.

Research Goals

The goal of this experiment was to determine whether the students involved in a multi-disciplinary group project had a better, more enhanced experience than those not participating in a multi-disciplinary group, as measured by a pre-project survey and a post-project survey.

ELC 4438, "Embedded Systems Design"

ELC 4438, "Embedded Systems Design," has been taught as a required upper-level course in Baylor University's Department of Electrical Engineering, currently deployed in a 4-hour semester course. Although the focus is mainly on the design and implementation of embedded computer systems using microcontrollers, sensors and data conversion devices, actuators, visual display devices, timers, and applications specific circuits, it also includes some software design using microprocessor cross-development systems and real-time operating system principles.

The main objective of the course is for students to learn to design and implement embedded computer solutions that meet specific system needs and/or requirements. In addition, during the course students are expected to:

1. Demonstrate understanding of embedded system design criteria
2. Design embedded systems and produce design rationale
3. Implement embedded system software solutions
4. Demonstrate understanding of real-time scheduling, priorities, and operating systems
5. Select hardware solutions that meet physical, computational, and interface requirements
6. Demonstrate a basic understanding of the Internet of Things (IoT) and distributed systems
7. Complete a final project that demonstrates the lifecycle of systems development

The means to evaluating these objectives was a combination of homework (10%), a series of labs (30%), two midterm exams (30%), and a final project (30%). The homework and labs, in particular, provided hands-on opportunities to apply what was learned, leading up to the final project.

Homework was assigned to cover topics including:

- SysTick: intended to familiarize students with the SysTick Timer and begin to think about how to use it to control time in the labs
- Digital Thermometer Design: intended to demonstrate knowledge and application of the design process
- TWIM Diagrams: application of Two-Wire Interface Master (TWIM) protocol to communicate with the temperature sensor
- Scheduling: application of various scheduling algorithms, dependencies, and the graphical representation of these schedules
- Project Idea Submittal: research and compilation of various group project ideas

The labs included a series of in-class demonstrations of the understanding and integration of:

- Lab 1: Introduction to the Atmel SAM4L XPlained Pro – GPIO and Polling
- Lab 2: LCD Light Sensor Lab - LCD Control, Light Sensor Input, Serial I/O
- Lab 3: Interrupt Driven Lab – Interrupt Priorities, Nested Interrupts
- Lab 4: Digital Thermometer Lab – TWIM (I^2C) Temperature Sensor Interface/Integration
- Lab 5: Static Scheduler – Real-Time Scheduling, Timer Counters
- Linux Lab – Linux Refresher
- BeagleBone Black Wireless (BBBw) Setup
- BBBw Flashlight Lab - GPIO
- BBBw TMP36 Lab - ADC
- BBBw LED Dimmer Lab – DAC (PWM)
- BBBw Motor Control Lab – Power Circuitry
- BBBw IoT Network Lab – Network Sockets

CSI 4v96, "Embedded Systems"

CSI 4v96, "Embedded Systems," is an upper-level computer science elective that was introduced in the fall of 2016. It is a variable-hour "Special Topics" course in the computer science curricula, currently deployed in a 3-hour semester course. Different than ELC 4438, the course introduces embedded systems from a computer science prospective using the BBBw embedded Linux platform. The course assumes mastery of systems programming, software design, algorithms, and a variety of operating systems. However, the course also expects students to

- Review, understand, and apply basic circuits principles
- Understand and demonstrate the scheduling of hardware resources
- Demonstrate knowledge and application of hardware constraints when designing software for embedded systems

- Demonstrate understanding of discrete components, use of transistors and FETs as switches, interconnection/interface to logic gates, and analog-to-digital conversion
- Demonstrate application of control of BBBw general purpose input/output (GPIO) pins through software
- Demonstrate application of cross-compilation and the Eclipse integrated development environment (IDE)
- Demonstrate understanding of bus communication
- Demonstrate understanding and application of the IoT using a variety of devices communicating through a variety of communication protocols

Evaluation of these objectives was conducted through a variety of Labs and Assignments (30%), two midterm exams (40%), and a final project (30%). The purpose of the labs and assignments, in particular, prepared students to apply what they had learned in the application of their final projects.

A series of homework assignments were assessed, including:

- Assignment (A)1: Compare and contrast several current microcontrollers/small board computers against a wide variety of performance characteristics
- A2: Implementation of a "die" object in Java Script, rolling the die 60,000 times, verifying the frequency distribution
- A2Extra Credit: Creation of a binary clock in Java Script
- A3: Circuits Exercises
- A4: IoT Literature Search
- A5: Use and application of the ThingSpeak API

The labs provided an evaluation of student's understanding of what they had learned through a variety of software and hardware interfaces:

- Lab 1: Debian flashing and setup
- Lab 2: Wifi setup, update packages, install ntpdate, add user account (primary, instead of "root")
- Lab 3: Updating packages
- Lab 4: JavaScript lab, access to BBBw as a local server
- Lab 5: TMP36 Sensor Lab
- Lab 6: TMP36 Sensor Lab, Part II
- Lab 7: LED Lab
- Lab 8: Digital I/O
- Lab 9: Analog I/O
- Lab 10: Apache Web Server
- Lab 11: TMP36 Web Pages

- Lab 12: Java Lab
- Lab 13: Virtualbox Eclipse Lab
- Lab 14: Java VM
- Lab 15: Switches and RBG LED
- Lab 16: Light Sensor
- Lab 17: ThingSpeak
- Lab 18: Bluetooth
- Lab 19: ADXL345 Accelerometer
- Lab 20: RFID Lab

Joint Group Project Design & Development

The multidisciplinary group project in CSI 4v96 and ELC 4438 began right after midterm. Students from CSI 4v96 met with ELC 4438 during the simultaneously-scheduled regular lecture period for both courses. Both classes were introduced to the project phase, where they would decide on projects, with an overview of the evaluation artifacts:

1. Project Documents: Statement of Work (SoW) iterations, hardware design, software design, progress reports, etc.
2. Project Presentation and Results
3. Project Report and Submittal
4. Final Project Code Submit to Git repository

These artifacts were weighted somewhat differently for the CS students and the ELC students, based on the varying objectives of the two courses.

The milestones for the project were presented:

- Beginning of project phase - October 17
- Team formation complete - October 19
- Initial SoW Submittal - October 23
- Preliminary Hardware and Software Design – November 7
- Formative Assessment of teams - November 7
- Project Team Final Presentations - November 28
- Final Report Due/Summative Assessment - December 8

Individual students were invited to present their project ideas in order to develop interest among the students. The final projects selected included:

- Group 1: Music Frequency Display using Bluetooth
- Group 2: Automated Clay Pigeon Shooter
- Group 3: Bear Copter
- Group 4: Musical Multi-Effects Pedal with Analog Control and LED display

- Group 5: Smart Room
- Group 6: Car LED Notification
- Group 7: Mouse Droid

CATME, the Comprehensive Assessment for Team-Member Effectiveness [6],[7], was used to build project teams and to conduct both the formative and summative peer assessments. In building the teams, the various criteria were weighted (including a specific project preference for team members), with the highest weights being time availability, project motivation, project perspective, and project selection.

All teams submitted an initial SoW, and an iterative process was begun where project teams received feedback on the scope of their respective SoWs until a final SoW was determined.

For the remaining weeks in the semester, project teams worked on their projects, reporting status weekly to both instructors. Adjustments were made based on hardware availability and team evaluation of feasibility of project scope. Teams met twice a week in class, as well as outside of class as determined by each team. A formative assessment of team effectiveness was conducted using CATME Peer Assessment, and the results of this assessment were delivered to each student. Each student, using their team's assessment of their work, could make adjustments to their team roles and responsibilities, to afford a better summative assessment.

Final project presentations were conducted over two days, allowing each team roughly fifteen minutes to present their projects and submit their final documents. In many cases, a video-taped and/or live demonstration was conducted and reported during the final project presentation. Each team, based on their final presentation of results, could recover from minor hardware/software malfunctions by making an additional presentation before the last day of classes.

Team members were required to submit a summative peer assessment via CATME. The raw scores of each assessment were evaluated by instructors, with adjustments made based on the thoughtfulness of the assessment, and individual multipliers were applied to the group project grade for each team member.

Summary

Development of this multi-disciplinary group project methodology was developed through a series of design and development meetings. The methodology was tested in the fall of 2017 with the two classes' students participating. Effectiveness of these teams was assessed by the team members themselves and by the instructors, based on the quality of cross training performed within each team, quality of hardware design, quality of software design, effective communication of the project, and professionalism of the final project report.

The students involved in this experiment were invited to report on the effectiveness of the project, with respect to the course content (labs, assignments, exams, projects), the current course format (dates/times of class meetings), the selection of the small board computer selected for the deployment of the course, how final grades were determined, and the final project design. Many lessons were learned, and a collection of best practices were determined based on this student evaluation as well as the instructors' assessment on the respective classes and the group projects. The design of this experiment was marred by the fact that the computer science version of the class was taught as a "Special Topics Course", so the population we had hoped to sample was very small, in fact n = 2. Because of this, no student evaluations were available except those distributed informally toward the end of the class. In our next iteration of this experiment, we will deploy our experiment to test whether the students involved in a multi-disciplinary group project had a better, more enhanced experience than those not participating in a multi-disciplinary group.

Lessons learned include:

- Current labs provide an ample background to prepare students for the project
- Labs and assignments should continue at the current pace
- CSI 4v96 should be made a 4-hour course, concurrently scheduled with the ELC 4438 course, to afford more time for project work
- CSI 4v96 was perceived to be one of the top candidates for upper-level computer science elective courses (students recommend it be made a permanent course)
- The BBBw has limited online resources, when compared to other small board computers, but the availability of GPIO on the BBBw makes it a good hardware platform for future courses
- There was some confusion on team roles and responsibilities in the integration of the two courses for the group projects

Some of the adjustments that will be made to the two courses, and the multi-disciplinary group project teams, will include:

- Until the two courses' schedules align, the instructors will ensure that students from both courses have at least 150 minutes per week of class time co-located throughout the project phase of the courses.
- Additional assessment of team members' roles and responsibilities will be formalized in individually evaluated assessments, for example, use of the jigsaw technique will be used to divide teams into common roles and responsibilities. [8] Each team member will cross train their team mates on their area of expertise.
- The possibility of CSI 4v96 becoming a permanent 4-hour course will be pursued and evaluated based on department priorities and resources.
- The new Peer-to-Peer reporting function will be incorporated in future offerings.

- The CATME student training resources will be made mandatory modules for students, and will be assessed.
- The importance of team assessment will be formally included in the course, focusing on the importance of team building and team assessment.

The integration of these two courses during the project phase of the course provides great benefit to both groups of students by utilizing their unique but overlapping skillsets. The project possibilities are increased by this multi-disciplinary approach. The electrical and computer engineering students are very capable with electrical and electronic hardware and low-level software (assembly and C), but they often do not have the high-level software background required for more complex software projects, web interfaces, and advanced operating system interactions.

The computer science students have very strong high-level software skills that provide another dimension to the projects. These skills open the door to more advanced IoT networks and controls as well as advanced software algorithms. However, the computer science students do not have as much background in circuits, electronics, board design, and power. The electrical and computer engineering students provide these skills, which opens the doors for more advanced hardware design.

The combination of these skillsets provides increased project possibilities, yielding more exciting and more educational projects, as group members teach each other these new skills as they develop their project. We will continue to integrate these recommendations and best practices into the two courses this spring and in the future.

References

[1] C. Fry and S. Potter, "The Design and Development of a Multi-Disciplinary Project in Embedded Systems Design," in *Proceedings of the ASEE Gulf-Southwestern Section Annual Conference 2018, Austin, TX, April 4-6, 2018.*

[2] T. Henzinger and J. Sifakis, "The Embedded Systems Design Challenge," in *Misra J., Nipkow T., Sekerinski E. (eds) FM 2006: Formal Methods. FM 2006. Lecture Notes in Computer Science, vol 4085. Springer, Berlin, Heidelberg*.

[3] T. Henzinger and J. Sifakis, "The Discipline of Embedded Systems Design," in *Computer, volume: 40, Issue: 10, October 2001.*

[4] D. Gajski, S. Abdi, A. Gerstlauer, and G. Schimer, "Embedded System Design: Modeling, Synthesis and Verification," in *Springer Science & Business Media, August 2009.*

[5] T. Henzinger, "Mission Statement: Center for Hybrid and Embedded Software Systems," Unviersity of California, Berkeley, http://chess.eecs.berkeley.edu, 2002.

[6] CATME SMARTER Teamwork, www.catme.org

[7] R. Layton, M. Loughry, M. Ohland and G. Ricco, "Design and Validation of a Web-Based System for Assigning Members to Teams Using Instructor-Specified Criteria," in *Advances in Engneering Education, v2 n1 Spr 2010*.

[8] K. Van Treuren and C. Fry, "Teams and Team Building at Baylor University: Why Should We Do This and Where Should This Occur in the Curriculum?", in *Proceedings of the 2017 ASEE Annual Conference & Exposition, Columbus, OH, June 25-28, 2017.*