AC 2011-376: WIRELESS-INTEGRATED EMBEDDED REAL-TIME CONTROL: A CASE STUDY IN ADOPTING RESOURCES FOR DEVELOPMENT OF A LOW-COST INTERDISCIPLINARY LABORATORY PROJECT

Paul G. Flikkema, Northern Arizona University

Paul G. Flikkema received the PhD in Electrical Engineering from the University of Maryland, College Park. From 1993-1998 he was an Assistant Professor at the University of South Florida, and joined Northern Arizona University as an Associate Professor in January 1999, where he is currently Professor of Electrical Engineering. He has been a JSPS Visiting Researcher at Yokohama National University, a Visiting Research Scientist at Sony Computer Science Laboratories, Tokyo, and a Nokia Fellow at Helsinki University of Technology. In 2007, he co-organized a US-France Workshop on Sensor Networks and the Environment sponsored by the French government. In Spring 2008 he was a Visitor at SAMSI, where was Program Leader of SAMSI's Program on Environmental Sensor Networks.

Kenji Ryan Yamamoto, Northern Arizona University
Carol Haden, Magnolia Consulting, LLC

Carol Haden is a Senior Consultant for Magnolia Consulting, LLC, a small woman-owned research and evaluation company based out of Charlottesville, Virgina. For the past eight years, she has specialized in the evaluation of informal and formal STEM education programs. Dr. Haden has evaluated projects sponsored by the National Science Foundation, NASA, the William and Flora Hewlett Foundation, the Arizona Board of Regents, and the Arizona Department of Education.

Jeff Frolik, University of Vermont
Tom Weller, University of South Florida

Thomas M. Weller received the B.S., M.S. and Ph.D. degrees in Electrical Engineering in 1988, 1991, and 1995, respectively, from the University of Michigan, Ann Arbor. From 1988-1990 he worked at Hughes Aircraft Company in El Segundo, CA. He joined the University of South Florida in 1995 where he is currently a professor in the Electrical Engineering Department and Associate Dean for Research in the College of Engineering.

Wireless-Integrated Embedded Real-Time Control: A Case Study in Adopting Resources for Development of a Low-Cost Interdisciplinary Laboratory Project

Abstract

In the last decade, it has become apparent that the grand challenge problems of this century span disciplines. In spite of this, engineering curricula are still strongly stovepiped, even within each engineering discipline, and both inertia and downward budget pressures encourage curricular conservatism. At the same time, the need is urgent to expose students to the diversity and complexity of real-world problems where there is no "best" solution. How should we help students learn across disciplines and blend disciplinary knowledge to solve problems?

This paper describes a laboratory project suitable for courses in areas of control and embedded systems that weaves critical aspects of control systems design with real-time embedded systems hardware and software, and along the way incorporates additional skills and tools. The project builds on previous efforts that have used the classic "ball-in-tube" experimental platform. We have developed an extremely low-cost experimental platform that student teams assemble from simple parts (e.g., shoeboxes and muffin fans), and that uses wireless communication between the real-time platform and a personal computer that provides a human interface and analytical tools. For real-time data acquisition and control, we adopted the CLIO platform that was designed for the experiential component of MUSE (Multi-University Systems Education, www.uvm.edu/~muse), an NSF-sponsored pedagogical effort to increase the ability of students to become conversant in skills related to systems thinking. In this spirit, the work discussed herein exposes students to experimentation, modeling and design across system layers. While tackling the project, students have also become more adept at (i) architecting distributed applications that integrate embedded and desktop computing systems, (ii) data acquisition, including measurement noise and signal conditioning, (iii) actuation, including motor control, and (iv) wireless communication. We present early assessment results evaluating how effectively the project helps students build critical systems-thinking skills, and the challenges of adopting resources for fast-tracking the development of new laboratory projects.

Introduction

The notion of systems thinking is well-known, but views vary on its specific definition[1,2]. For example, the systems dynamics community emphasizes understanding the temporal dynamics of interconnected parts, including the effects of feedback and emergent behaviors, via conceptual and simulation models[3]. Others have emphasized design in a broader context, including assessment of societal impacts and awareness of economic and societal goals[4].

In our practice-oriented view, systems thinking is already pervasive across engineering disciplines, highly valued in industry, and normally learned on the job. The best systems thinkers become technical leads and managers in part because they become experts in systems thinking as part of their professional and technical career growth. Systems thinking is in our experience often developed through informal mentoring, and is larger in its aims and scope than systems engineering (as often executed, e.g., using trade studies involving several variables). Our definition includes the systems dynamics viewpoint and encompassing contexts, but also involves thinking about the design process in ways that (i) span traditional disciplinary boundaries, (ii) integrate design layers from high-level abstractions to low-level implementation, and (iii) reveal the relationships that couple seemingly disconnected models. Two aspects are critical: modeling and context; context may be broad, including human, societal, and environmental impacts.

Our interest in systems thinking is driven in part by our motivation to help engineering students develop the skills that will be critical to the pressing technological challenges of this century. While it is widely recognized that the engineering graduates of today and the future should have better systems thinking skills, the challenge of inculcating those skills remains. In this paper, we describe an attempt to do this within the constraints of an embedded systems course that is typically found in an ECE curriculum.

Overview of the Approach

The focus of this effort was the development of a final laboratory project in the context of a senior-level EE course. The course, EE 410 – Embedded Control, focuses on deeply embedded systems, and has been offered by one of the authors four times in the last decade at Northern Arizona University, with a range of topics including robotics[5]. With each iteration, the course content has been updated to keep up with industry trends, including the transition from assembly language dominance to the use of the C language, moving from 8- to 16-bit architectures, the evolution of multiple clock sources/domains and low-power modes, and the increasing importance of networked embedded systems, especially implemented using wireless communication.

The course strongly emphasizes experiential content: the laboratory projects account for 80% of the final grade. Student teams consisted of at most two students. For all projects, a team's project grade is based on both its degree of success in demonstrating achievement of objectives in the laboratory and a comprehensive written report.

To help students equip themselves with the skills to tackle a substantive final project that involves systems thinking, we adopted a progressive learning method, with continual reinforcement and synthesis of skills and tools via the sequence of projects (Table 1).

Table 1: Course projects leading up to Final Project.

Project	Concepts	Skills and Tools
0: CLIO Quick Start	Digital I/O review; looping and software delays; program compilation and loading; port/pin configuration	Integrated development environments; familiarity with hardware platform and MSP430 MCU; hardware interfacing
1: Heartbeat Simulation	Functions; parameterized software design; real time systems	Flowcharts; defined constants
2: Switch-controlled Heartbeat Simulation	Interrupts and program state; externally-triggered interrupts; digital inputs; mechanical switch characteristics	Interrupt service routines (ISRs); switch debouncing
3: Software LED Dimmer with Switch Control	Clock and clock domains; timers; pulse-width modulation (PWM); internally-triggered interrupts	Software integration of multiple functions; timer modes; timer configuration and management

Table 1 (continued): Course projects leading up to Final Project.

Project	Concepts	Skills and Tools
4: Pseudorandom Number Generator and Reporting to PC	Linear feedback shift registers; maximal-length sequences; random and pseudo-random processes; statistics: sample mean and autocorrelation; user interfaces	Advanced bit-oriented processing in C; serial communication; UART programming
5: Temperature Sensing	Sensing; analog-to-digital conversion (ADC); periodic schedulers; MCU low-power modes	ADC peripheral configuration and operation; low-power modes on the MSP430 via ISR's; PC communication (more)
6: Motor Control	Actuation; DC motors; motor control; power electronics; pulse-width modulation	H-bridge interfacing with MCU; hardware-based PWM
7: Transducer Characterization	Data acquisition; transducer signal processing; resolution; noise, bias, and non-linearity	Advanced ADC peripheral configuration and operation; data conversion

The linear project sequence was designed to cover a cloud of concepts that span from the general—traditional ECE subdisciplinary silos (Figure 1, bottom)—to specific conceptual components required for the final project. Thus the concept map of the course topics and how the development of concepts leads to the final project is hierarchical in nature, and the final project integrates knowledge from across subdisciplines as shown in Figure 1.

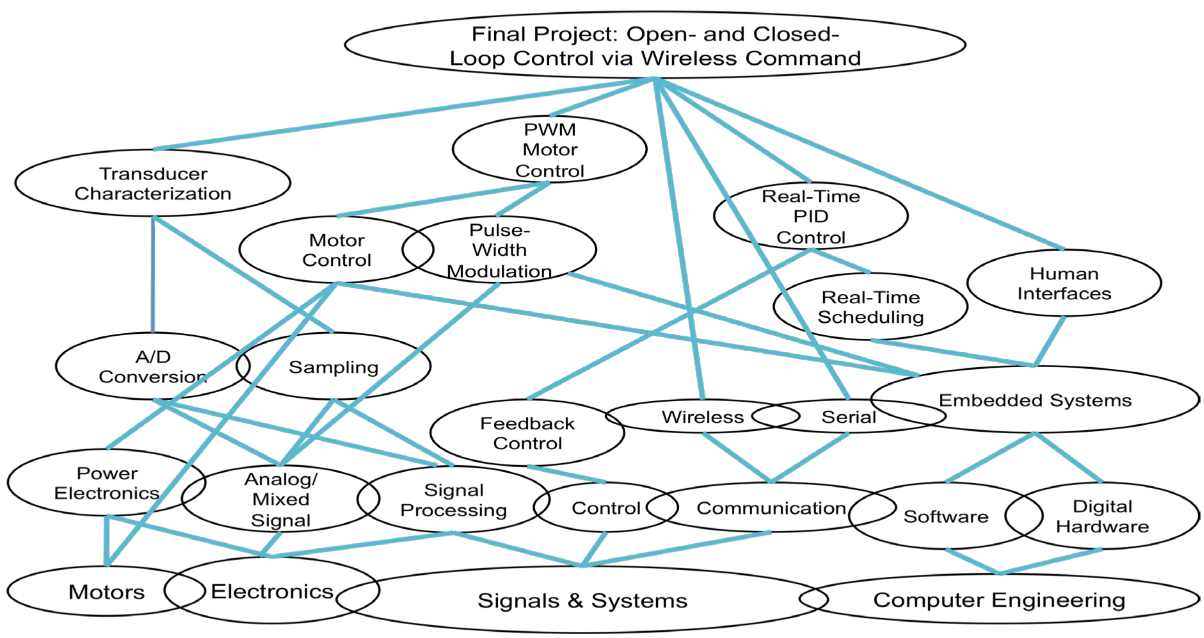


Figure 1: Hierarchical concept map. Connectedness of concepts is shown by overlapping nodes in addition to edges. Note that all possible links and overlaps are not shown.

Final Project

For the final project, our primary goals were (i) to allow students to tackle a hands-on design project that requires systems thinking skills and that spans the EE subdisciplinary silos and (ii) to maximize portability by making the project easy to replicate at extremely low cost. We chose the well-known "ball-in-tube" (BIT) system[6,7,8], where the goal is to control fan thrust to move the ball to a set point corresponding to a user command that can be entered at any time.

We designed our BIT system (shown in block-diagram form in Figure 2) to use easily available hardware (Table 2) and allow rapid prototyping by student teams of the entire experiment, including the mechanical construction.

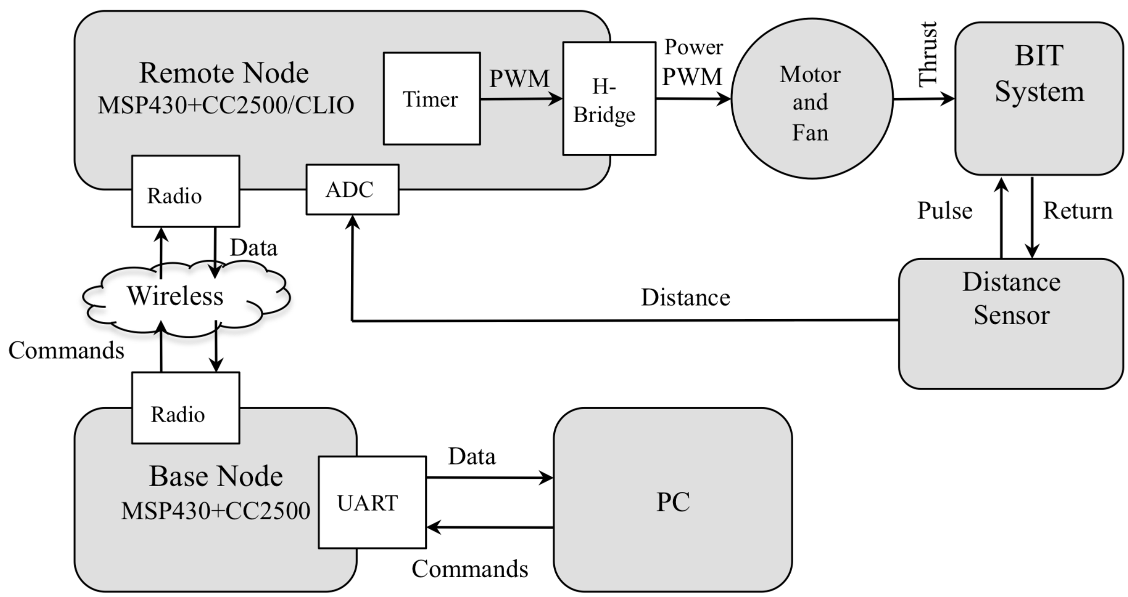


Figure 2: Block diagram of complete Ball-In-Tube (BIT) control system. Distance information can be used for feedback control of ball height.

The students were shown a prototype mechanical design and encouraged to improve upon it (Figure 3). Designs for the electronic hardware are available on the web (www.cefns.nau.edu/~pgf/ETM/ETM_index.html). For development of the embedded software on the base and remote nodes, we used Code Composer Studio CCS, an integrated development environment (IDE) from Texas Instruments. In keeping with the objective of low cost, a free version is available; this version is program memory-limited, but the limit is well above that needed for any of the eight projects. The base node communicates with the PC via the eZ430-RF2500's ability to tunnel asynchronous serial communication through the USB connection. The required driver is provided as part of CCS. For the user interface, we used a simple terminal emulator application (e.g., Putty). Wireless communication between the base and remote nodes was via the Texas Instruments CC2500 single-chip 2.4 GHz transceiver on the eZ430-RF2500.

Drivers and hardware-abstraction layer code were provided for the students to interface their code with, allowing them to learn the important concepts of structures and unions in the C language.

Table 2: Parts list for Ball-In-Tube project.

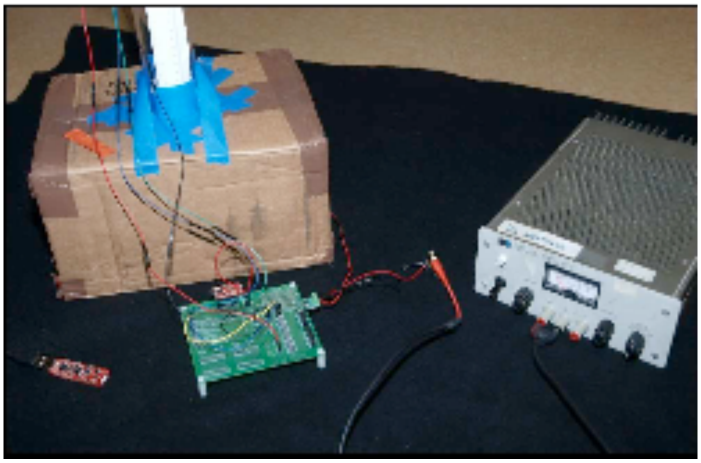
Subsystem	Model Number/Description	Manufacturer/Source
Remote Node	EZ430-RF2500 (MSP430+CC2500)	Texas Instruments
	CLIO wireless sensor node development tool	For connecting EZ430-RF2500 with H-Bridge motor controller; www.cefns.nau.edu/~pgf/ETM/ETM_index.html
	CLIO-AMC H-bridge peripheral motor control board (based on Si9986 buffered H-bridge motor control IC)	Vishay Siliconix; similar H-Bridge IC's could be substituted
Ball-In-Tube (BIT) System	Styrofoam ball	Hobby/craft store
	Extruded acrylic tubing, 2.25" OD, 2.00" ID	One source is US Plastic (usplastic.com)
	Shoebox	Supplied by student team
Motor & Fan	Generic muffin computer cooling fan (uses brushless DC motor)	e.g., NewEgg.com
Ultrasonic Distance Sensor	LV-MaxSonar®-EZ4	Maxbotix (one vendor is sparkfun.com)
Base Node	EZ430-RF2500	Texas Instruments (note: 1 kit contains two EZ430-RF2500 nodes)
PC	Windows PC (or Mac with bootcamp or virtualization software and Windows) and USB port	Generic
Miscellaneous parts	Connectorized jumper wires (e.g. from sparkfun.com), or plain wire and wire-wrap tool; 12 V power supply; it is also helpful to have a basic oscilloscope	

Project Objectives. The project was separated into two parts defined by the following functional objectives in the project assignment:

> Part I: *D Mode* (duty cycle mode). You will be able to enter a PWM duty cycle at the PC and it will be successfully sent to the Remote Node.
>
> Part II: *D Mode* and *S Mode* (setpoint mode). Your system will respond to *D Mode* or *S Mode* commands, where an S mode command is simply the PC user's desired ball height in inches, also known as the setpoint.

The students were not initially informed that *D Mode* was a form of open loop control, while *S Mode* involved unity-feedback closed-loop control.

Student Preparation. At the outset of the 2.5 week project, students were given a written project assignment and two mini-lectures: one on structures, unions, and radio-chip interfacing, and one on how this control problem shared fundamental concepts with the problem of energy storage and propulsion control in flywheel-based hybrid race cars. Midway through the project, a 20-minute lecture reviewing the physics and ordinary differential equation model of a BIT system and closed-loop PID control was given. Almost all students had already taken the standard junior-level course in signals and systems, but none had taken a course in control systems.



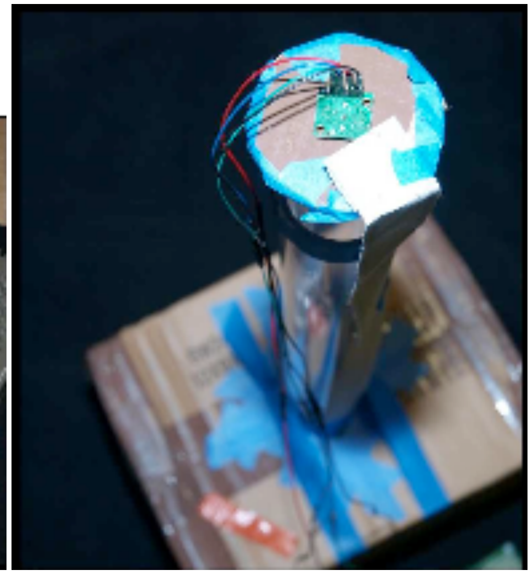


Figure 3: Example experimental setup. Left: Acrylic tube mounted with masking tape to shoebox (with motor/fan mounted inside), remote node with eZ430-RF2500 and H-bridge peripheral board, and 12V power supply. Base node eZ430-RF2500 is at bottom left, not plugged into a PC. Right: The ultrasonic distance sensor is placed on a platform above the tube to allow air flow and a smaller measurement deadzone.

Assessing the Effectiveness of the Course in Enhancing Student Learning

Student Learning. In previous work[9,10], we presented systems thinking explicitly and persistently throughout the semester as context for technical topics. In this effort, we took a different approach in an attempt to see how well students' grasp of systems-thinking concepts emerged as a result of the progressive nature of the laboratory projects. For this reason, the instructor only discussed systems-thinking concepts as they arose either naturally or from student questions.

Overall, the student teams did remarkably well. Two of thirteen teams were able to achieve reasonable performance in Part II (D mode and S mode), a very surprising result since stable and responsive constants for the three feedback paths are normally computed using nontrivial algorithms[11,12].

The students' learning in the final project was assessed by two judges who were experienced engineers working in local industry. The judges were given a rubric (reproduced in Table 3) to guide their evaluations of short oral presentations and demos given by the students at their laboratory benches. Though the judges scored the projects independently, there was considerable agreement. Two teams fell behind in the course and did not complete the final project. With those (zero) scores removed, and the 1.0-4.0 scores normalized to a typical grading scale of 0-100, the mean and median project scores were 82.6 and 81 respectively, with a standard deviation of 8.6.

Table 3: Assessment rubric for the final project.

Criteria	**Outstanding (4)**	**Good (3)**	**Fair (2)**	**Poor (1)**
Project Objectives	Clear and concise statement of the project objectives.	A few deficiencies in clarity and conciseness, but close to the mark.	The statement of what is expected is unclear.	Weak/poor articulation the objectives.
Demonstration – Part I (D mode)	Deft and smooth demo of the performance of the system.	Demo meets expectations overall, but glitches occur.	Demo meets minimum expectations, but is choppy, error-prone, or accompanied by excuses.	Team is unable to demonstrate their system.
Demonstration – Part II (S mode and D mode)	Deft and smooth demo of the performance of the system.	Demo meets expectations overall, but glitches occur.	Demo meets minimum expectations, but is choppy, error-prone, or accompanied by excuses.	Team is unable to demonstrate their system.
Explanation of results (in context of objective; how well it works); interpretation of results – why it works as well as it does (or doesn't)	The team demonstrated mastery in describing and explaining the obtained results. Cogent understanding of constraints and error sources, and their effect on performance.	Team is able to describe and explain the obtained results. A few gaps, but overall understanding of constraints and error sources, and their effect on performance, is solid.	The explanation was incomplete, but shows some understanding of the results. Some understanding of constraints and error sources is stated, but underlying logic appears somewhat flawed or shaky.	Little or no understanding of the results and their meaning. Poor understanding of constraints and error sources.
Key engineering concepts learned (assess from presentation and follow-up questions/discussion); includes knowledge of: key components and interconnections; modeling; performance criteria	Team has an accurate and thorough understanding of key engineering concepts underlying the project.	Team has a solid understanding of key engineering concepts underlying the project.	Team has a limited understanding of key engineering concepts underlying the project.	Team has an inaccurate or severely limited understanding of key engineering concepts underlying the project.

We formally explored students' grasp of systems thinking via a set of Exploratory Questions (Box 1) that student teams were required to answer twice: before work on the final project, and in conjunction with their final project report. The students' work on the project appeared to help improve the teams' answers. A number of teams' new answers indicated increased awareness of the potential for measurement errors that they developed while writing their reports for the Project 7 (Transducer Characterization). From a systems-thinking perspective, the greatest changes were in answers to *Question 4*. A common initial answer was a collection of clip-art images of components with no attempt to interconnect them, or, even less informative, a photo of the ball in the tube. In the teams' second responses, these were often replaced by re-creations of either of two diagrams presented in the brief lecture midway through the project: a re-creation the force diagram and ODE model, or a re-creation of the PID control system block diagram.

Box 1. Exploratory Questions.

You have a ball and tube, along with a fan and a power supply to provide power to the fan. The ball is in the tube, and the fan blows air into the tube. In crude terms, a greater fan speed means the ball has a tendency to move upward in the tube.

Your goal is to develop a ***system*** that you can give commands of the form: "Ball height = *x* cm" and the system will automatically, in real time, control the fan to move the ball to height *x*. You'll also have available an embedded computer (eZ430-RF2500 + CLIO + CLIO-AMC) to control the fan.

Answer the following questions:

1. What hardware components do you need in addition to the fan, tube, ball, power supply, and embedded computer?

2. Draw a diagram of the system, showing interconnections between components and what flows along those interconnections. Label both the components and the flows as precisely as you can.

3. What are the most important components of this system?

4. This system can be represented in other ways beside a block diagram. Develop and show another type of model for this system.

5. Develop a set of criteria to measure the success of this system. Are there any additional tools you will need to do the testing?

6. What problems do you anticipate in trying to achieve the objective? Be specific.

7. Describe, in as much detail as you can, how you might test your system to see how well it measures up with respect to the criteria you've identified in question 5.

One team developed a flow diagram of the complete system that included all critical components, but omitted the connection between the motor and the ball height sensor. Another team reported that

> *For the last assignment we could not come up with #4, another way to model this problem. For this assignment we now know that this problem can be modeled by complex differential/integral equations.*

A third team initially wrote an algebraic equation relating the ball height to the sensor ADC count, but in their second response noted that

> *…the differential equation for the ball height relative to the fan duty cycle allows for discrete control of a dynamic system.*

Student Evaluation of the Project. The second author, an external evaluator, conducted focus group interviews with students in the course to gather feedback on the experiment and its impacts on their learning. For the interviews, students in the class were divided into three groups and each focus group interview lasted 30 minutes. Students responded to questions about what they most liked and did not like about the project, what they had learned as a result of the project, and how the project affected their awareness and understanding of a systems approach to solving engineering problems.

Across focus groups students commented on how well the final project built upon their learning throughout the semester. Projects earlier in the semester laid the foundation for understanding the components necessary to make the final project work. Students commented that the final experiment was "an actual engineering project that built on skills they had learned", and allowed them to "pull the pieces together—hardware, input and outputs" to learn how they work together as a system. The final project also served to help students understand the learning from previous projects in more depth due as they applied their understanding of the components to constructing the system in the final experiment.

Students agreed that the "real-world" aspect of the project was appealing to them. One student commented, "This is a real-world problem. We'll see something similar to this control system after graduation. The thought process learned from this project will apply." Another stated, "This is a fairly realistic controls problem with the same types of issues hitting on interactions. There are multiple noise sources and we have to consider everything and then decide what we can deal with and what we can't." Related to this, students appreciated having a "physical system" to work with instead of "only numbers or data on a screen" and the opportunity to apply what they had learned from other PWM systems.

Additionally, they felt that the approach of building and testing parts of a system first before bringing them together in the final project was applicable to how they would work in the real world. One stated,

> *We were learning debugging. You start small and build from there. We were building a system from a dozen or so components we built throughout the year. When building that system, you want to build as little as possible at a time and then test it. Testing components along the way allows for the final project to involve looking at how they work together.*

Students appreciated the introduction to wireless systems given through the final project although several commented that they would have liked an introduction to wireless systems earlier in the semester. This aspect of the project was new for them and there was a learning curve to understand the system for application to the final project. Several students commented that they would have liked to have more of an understanding of the code that was developed and given to them for the final project. Others countered that it was realistic in the work world that others would be writing code that they would then apply to their work.

Students commented on how the project affected their awareness and understanding of a systems approach to solving engineering problems. Across all groups, students felt that the project gave them an understanding of how components come together to create a system designed in response to a specific problem or design request. They viewed systems thinking as "putting small pieces together to solve a bigger problem" and as the "integration of all components into one system." One noted, *You can't get the whole to work without the parts. There's not a single part of the project you could eliminate to be able to complete the project. For a system to work it has to have all of its parts functioning properly.*

Students commented that having a "holistic view" was key to understanding the project and to utilizing systems thinking. One student commented,

Systems thinking is instead of just looking at input to output – it's breaking down everything that happens between them that affects the outcome. Over the semester, we have looked at the pieces individually, and now we're looking at how they interact with each other.

Students viewed systems thinking in light of the project as allowing for the integration of different aspects of engineering such as hardware and software, and to incorporate different fields of engineering including electrical engineering and programming.

Conclusion

Results of the student assessments and student interviews indicate that the progressive learning method was successful in laying the foundation for the final, more complex, BIT control problem. Developing an understanding of the components of the final system through the prior course projects allowed for the synthesis and application of earlier learning to the culminating project. With respect to systems thinking, the final project allowed students to step back and see the dynamic interconnectedness of parts and how they contribute to the larger dynamic system. Furthermore, the project provided a "real-world" example of how systems are designed in the field by crossing subdisciplinary boundaries, and designing and testing components before applying them to the whole.

A possible limitation of this course/project combination was the lack of time for students to construct and test simulation models of the BIT system using, e.g., Simulink or Modelica. Given the course's emphasis on embedded systems, and that none of the students had taken a course in control systems, we satisfied with the course's alternative approach of discovering systems-thinking principles in the context of control systems design. Though we have no objective data, we believe that this project motivated a significant number of students to take a senior technical elective course in control systems in the following semester. Given the project's interdisciplinary nature, we believe it could be successfully integrated into a control systems course for students with previous exposure to embedded systems programming.

We note that the hierarchical concept map of the project (Figure 1) was not presented to students. Rather, the map is presented in this paper as a means of providing a summary of the course projects and their relationship to the final project. Students did not use maps such as this one to answer the Exploratory Questions for the final project. We believe that in replicating this progressive learning methodology in future courses, it may be beneficial to use the hierarchical concept map as a tool for instruction. This could be done either through explicitly teaching the hierarchical concept map as a tool for synthesizing concepts and tools, or it could be used as a method of assessment for examining student understanding of the interrelatedness of the concepts and how they build from the general to the specific. Use of the concept map could serve to reinforce the systems thinking approach and development of higher order thinking skills targeted by the final project.

We feel that this project provided a low-cost, content-rich learning opportunity for students. It supported learning of key systems thinking skills in a real-world context that utilized experiential learning to increase student engagement and learning. We encourage instructors to adopt the project for use in their courses. Information on both the final and preceding projects is available to interested instructors.

Acknowledgments

This work was supported by NSF grants DUE-0717326, DUE-0716812, and DUE-0716317.

---

[1] M. Frank. A 3-D approach for curriculum content. IEEE Trans. Systems, Man, and Cybernetics—Part C: Applications and Reviews, Vol. 32, No. 3, pp. 203-214, August 2002.

[2] P. M. Senge, *The Fifth Discipline: The Art and Practice of the Learning Organization*. New York: Doubleday, 1994.

[3] M. Hopper and K. A. Stave. Assessing the effectiveness of systems thinking interventions in the classroom. Proceedings of the 26th International Conference of the System Dynamics Society. Athens, Greece, July 20-24, 2008.

[4] R. Hadgraft, A. Carew, S. Therese, and D. Blundell. Teaching and assessing systems thinking in engineering. Proceedings of the Research in Engineering Education Symposium. Davos, Switzerland, July 7 – 10, 2008.

[5] P. Flikkema. Learning embedded and real-time systems via low-cost mobile robots. Proceedings. 2001 ASEE Annual Conference and Exhibition, 2001.

[6] J.S. Pereira and J.B. Bowles. Comparing controllers with the ball in a tube experiment. IEEE Trans. Fuzzy Syst., Vol. 1, 8-11 pp. 504-510, Sept.1996.

[7] P. Wild, B. Surgenor, and G. Zak. The mechatronics laboratory experience. Mechatronics, Vol. 12, No. 2, pp. 207-215, March 2002.

[8] O. Ziwei , S. Michael and W. Kexin. The experiment "ball-in-tube" with fuzzy-PID controller based on Dspace. IEEE Int'l Conf. Systems, Man, and Cybernetics, pp. 877-881, Oct. 7-10, 2007.

[9] C. Haden, P. Flikkema, T. Weller, J. Frolik, W. Verrei-Berenback, and W. Shiroma. Assessment of a hybrid, online/in-class course developed at multiple universities. Proceedings, 2009 ASEE Annual Conference and Exhibition, 2009.

[10] P. Flikkema, T. Weller, J. Frolik and C. Haden. Experiential learning of complex engineered systems in the context of wireless sensor networks. Proceedings, 2010 ASEE Annual Conference and Exhibition, 2010.

[11] J.G. Ziegler and N.B. Nichols. Optimum settings for automatic controllers. Trans. ASME, vol. 64, pp. 759-768, 1942.

[12] K.J. Åström and T. Hägglund. *PID Controllers: Theory, Design and Tuning*. Research Triangle Park, NC: Instrument Society of America, 1995.