Paper ID #25336

Impact of an Embedded Systems Course on Undergraduate Capstone Projects

Maddumage Karunaratne, University of Pittsburgh, Johnstown

Dr. Maddumage Karunaratne is an Associate Professor and the Head of the Electrical Engineering Technology department at the University of Pittsburgh at Johnstown, PA. The department offers two undergraduate degrees in Electrical Engineering Technology and Computer Engineering Technology. Dr. Karunaratne earned a Bachelor of Science degree from the University of Moratuwa (Sri Lanka), a Master of Science from the University of Mississippi (Oxford), and a Ph.D. from the University of Arizona (Tucson).

Before joining academia, he gained fourteen years of extensive industry experience working in the semiconductor industry performing software development, application engineering, design, testing and verification of digital integrated circuits. He has taught electrical and general engineering technology classes at Pitt-Johnstown since 2004.

His research and teaching interests include Semiconductor circuit Testing and Verification, Low Power Design Analysis, Digital and Embedded Systems, Electromagnetic Wave Scattering, and IC Design Automation Software development. He has authored or coauthored 29 publications and 3 US patents.

He can be reached at maddu@pitt.edu 209 Engineering and Science Building University of Pittsburgh at Johnstown Johnstown, PA 15904

Impact of Embedded Systems Course on Undergraduate Capstone Projects

Abstract:

When electrical (EE) and computer engineers (COE) are trained, it is imperative that nearly all acquire some level of exposure to microcontrollers and associated software development skills to effectively function as engineers in their careers. At this university, electrical engineering and computer engineering majors take a one-semester course in microcontrollers to learn embedded system development in their junior year. In prior semesters, they take two programming courses and an electronics course. This embedded course is designed to teach data acquisition techniques, detail hardware operations in data processing, and how to drive peripheral components. While there are many choices of microcontroller development boards such as Raspberry Pi and Arduino, this embedded course and associated labs use hardware development boards based on Freescale (NXP) devices due to their simplicity and legacy.

The same EE and COE students take a mandatory capstone design project course spanning two semesters in their senior year. Students, individually or in groups of two, propose a few engineering problems to the faculty at the start of the first semester. After several iterations involving discussions with the faculty, each group selects one project to build a prototype to demonstrate the solution. In the first semester, they complete the engineering design steps producing documents for the selected project to develop a final design solution. Each group eventually completes the project work with a live demonstration at the end of the second semester.

This paper briefly explains the structure of the capstone project course for EE and COE seniors, and the design milestones. The author offers and discusses results from a survey conducted on senior students in a capstone project course taken by both EE and COE students. The survey and the follow-up discussion attempts to determine whether the embedded systems course influenced or facilitated the selection of microcontroller based projects compared to other types of projects. Another objective is to see if the teaching level on the application of microcontroller boards may be reduced since the public domain offers a vast amount of open source libraries to do almost anything related to them and associated peripheral components.

Key words - capstone design projects, embedded systems, microcontroller

Part I: Introduction

Electronics industry continues to make vast strides in providing functionally rich inexpensive devices for consumers and various industries including agriculture. Smart mobile robots – both personal and industrial - are proliferating fueled by efficient manufacturing combined with high energy battery technology. All these devices contain many sensors driven by embedded controllers and computing processors that are termed as microcontrollers. These microcontrollers typically offer less performance than the latest processors used in laptop and desktop personal computers, but they are sufficiently powerful and frugal in cost and energy use.

Electrical and computer engineers will not be successful without having a good skill set and an understanding of how these proliferating smart devices work and how they are integrated into complex systems. While most electrical and computer baccalaureate graduates may not enter careers to design semiconductor devices, they might still be designing or maintaining systems developed using off the shelf embedded processors. Also, graduates still need to understand the trends, adapt, and adopt new technologies to be successful in their careers.

The programming language in the embedded systems course is C due to its efficiency in memory allocation, run time, and its ability to directly manipulate hardware components, although higher level functional languages such as Python are becoming popular for microcontrollers. Over a third of the embedded course is allotted to learn the processor specific assembly language and processor architecture to make students understand the intricacies of processors. After they have become proficient in assembly coding, the course starts using C programming to teach a wider verity of microcontroller applications.

Both EE and COE students take a mandatory capstone design project course that spans two complete semesters in their senior year. Very early on the first semester, students, individually or in groups of two, propose a few engineering problems they dream of to the electrical and computer engineering faculty. After several iterations involving discussions with the faculty, each group selects one project to work on to build a prototype to demonstrate their problem solving abilities. It is mandatory that students built a working prototype to fulfill the course requirements. In the first semester, they complete several design steps for the selected project while deriving the final design solution. The design steps are the typical engineering project milestones backed by written documentation and oral presentations at the department seminar. Students complete the project work with a live demonstration at the end of the second semester in front of an audience consisting of a group of general public and experienced EE/COE engineers.

Part II of this paper emphasizes important parts of the embedded systems course leading to a discussion of the capstone project course and its requirements. Part III provides a detail discussion of the results of a survey completed by students after they have received the grades for the proposal and design phases of the project, but before starting the second semester of their projects. Finally, part IV summarizes the findings with future plans to explore more on this topic.

Part II: course flow and course contents

By the time EE and COE students take the capstone project, both majors must have taken the core technology related courses as depicted in Table 1 below. They take two physics courses among sciences followed by a rigorous C programming course along with a basic electric circuits class. After they have taken two more courses in analog and digital electronics, students are eligible to take embedded systems course which is a precursor to their most important capstone project course. With the heavy emphasis on hands-on experience at electrical and computer engineering programs at this university, each of those courses has a 3-hour lab in addition to having a 3-hour lecture class. By the time students get to the embedded systems course, they have done many lab experiments and developed good programming skills. COE students take different courses among computer science subjects giving them software engineering skills while EE majors take more electricity and signals courses such as machines, power, communications, and electromagnetics.

Teaching assembly coding is the best way for students to understand and appreciate the details of the actions carried out by the processor. It is challenging to learn and to apply assembly language compared to using C or higher level languages since each data manipulation has to be broken down into atomic actions taken by hardware – that is the basis of assembly code. Without learning assembly instructions, students would not realize the need, or whether it is even possible, to write a really compact code desired by miniature embedded controllers. Additionally, students see real-world usage of the topics they have learned in digital electronics classes, particularly logic circuit blocks such as ALUs (Arithmetic and Logic Units), registers, and data direction control blocks, etc. High-level languages typically do not expose students to the architecture of processors and their atomic actions. There are many concepts and techniques in high-level programming that can only be analyzed and scrutinized by assembly level actions. Assembly coding teaches them details about the processor and microcontroller functions, and builds their confidence to use embedded systems later in their courses and careers. In addition, there is a vast amount of open resources in public domain on how to use microcontrollers if students want to pursue a hobby or develop an entrepreneurial venture into products containing embedded systems such as drones and robots for agriculture, fishing, etc.

Science	Mathematics
Programming	Electric circuits
Analog electronics	Digital electronics
Signal processing	Embedded systems
Capstone project	

Table 1: Course Progression of EE and COE Majors

When students become proficient in assembly level coding, ironically the course switches to C-coding to program the microcontroller board. While assembly coding is important to learn the hardware intimately, the higher level C language is more productive in terms of development costs, and also C is less error-prone. Students find that the embedded course is rigorous and challenging throughout the semester, but builds their confidence in the subject and usage of the microcontroller development board – Dragon 12B plus. Laboratory exercisers use several different sensors and motor driver modules as peripherals in addition to the built-in components on the hardware board.

The embedded systems lab experiments build confidence in students so that when their embedded class final (exam) project [1] is assigned, they are not fazed by it. That project is to develop a vehicular system based on toy cars by students, in groups of two, to identify crossroads, stationary and moving obstacles, parking areas while automatically navigating to reach a destination. Every year it is a different road track and objectives. Each of their product is a unique system since each toy car is different in weights, handling, steering, tire grip, etc. They had to build, tryout, and fine-tune the algorithms, timing, battery selections, etc., which might be similar to what they would do in their future capstone projects.

After completing the embedded systems course, students start to brainstorm and discuss among themselves to look for potential capstone project ideas to propose to the faculty by the beginning of the following semester. The course requires students to work in small design teams of two to solve a significant engineering problem at the undergraduate level. The group size is kept small to make each student effectively contribute to the project and gain experience in different aspects of engineering projects. A typical group initially proposes two to three projects by describing the initial thinking in one or two paragraphs along with drawing sketches and their sources cited. Students are free to select project partners among the project's class. Groups also have the freedom to propose projects as long as they are related to their learned knowledge from the curriculum. However, they need to be appropriately challenging for the senior level graduating class. Their projects do not have to involve microcontrollers, although almost any modern digital controller utilizes a microcontroller. Instead, students may propose projects based on LabVIEW; Programmable Logic Controllers (PLC); power grid or micro grids related issues such as power quality estimation and measurements, power flow control, redundancy systems, or many topics related to Supervisory Control and Data Acquisition (SCADA) Systems; Programmable Hardware Boards (FPGA); and mobile device-based software apps.

Student groups need to analyze their selected project on several aspects to show that it is an engineering project with practical value to the society. For the program accreditation requirements, the capstone projects should consider ethical and social implications of the project in addition to its business implications. Each team presents information, in both written and oral formats, to peers and faculty at various developmental milestones of their project. The milestones include a brief initial project proposal followed by (if accepted by faculty) an elaborated description of what is being proposed as the product and at least a quick sketch of the prototype envisioned although it may not be the final design. Without such a forward-thinking, a project might be selected, just to find out that it is beyond the capabilities of undergraduates. After the project has been accepted with or without additional features suggested by faculty, students develop functional and engineering specifications aided by a house of quality matrix.

When the specifications have been finalized, several different implementation methods are to be considered by the design group before selecting the best one. Each project group presents the findings in a final design document which would include an estimate of man-hours, materials, and expenses required to complete the project. Each milestone of those documents involves an oral presentation at the department seminar to all the EE/COE students and to the faculty by each project group. In the follow-up semester, the finalized design is to be implemented under the guidance of a faculty member who holds weekly meetings. The project groups produce evidence of progress on a weekly and monthly basis using memos and status reports. Students consider the capstone project course as their most important and significant course work in the entire program although the total credit count is less than that of two regular lecture courses. The effort and skills student gain is valuable in obtaining employment in industry, and the practice of managing their project over a long period surely benefit their career prospects.

part III: student survey of capstone project designs

The author conducted a survey of the senior EE and COE students just after they have completed the 1st semester of the capstone project and received the grades. By then they have designed the product, but have not started to build the product and they have yet to use the microcontrollers if the project incorporated them. Some projects such as making a 3-phase power system fault monitoring panel would not require students to acquire and use an embedded controller since the panel would already have it built in. The very same applies to LabVIEW or PLC based projects to control mechanical devices such as motors, elevators, pumps, etc. In software app development or data flow study projects, microcontrollers are not involved explicitly.

Type of the embedded controller being used also depends on the specifics of the project. Some projects perform image recognition, such as identification of a face and eyes or a distant object with a target pattern to land an unmanned aerial drone for recharging. Another example project may be to build a remote control robot to drive and drill to test the depth of an ice layer on a lake. Image processing has attracted the use of Raspberry Pi along with Python programming language, while some use Arduino processor boards for simplicity. Some project groups with COE majors may develop software apps on a smartphone to control their embedded processor-based module, for instance, an outdoor barbecue grill controlled and monitored by an app developed for a smartphone.

The survey results from two years (29 students) are summarized in Table 2. The responses are based on the typical rubrics of 1 to 5 (1-strongly disagree, 2-disagree, 3-neutral, 4-agree, 5-strongly agree). Column 2 lists what each question was trying to assess based on students' perception and knowledge. Final three columns show minimum, average, and the maximum rubric values of their answers.

	The survey questions	Min	Ave	Max
1.	In the embedded systems course, I learned how to programs a microcontroller.	4	4.4	5
2.	I think I did ok in the embedded systems course & lab.	3	4.4	5
3.	In that course, I had to learn lots of details about microcontroller and its inside.	4	4.2	5
4.	I learned a more by doing the car project than by studying for a written exam	3	4.7	5
5.	My senior project is designed to use a microcontroller that needs to be programmed by the project members.	2	4.2	5
6.	I did not hesitate to consider and select my Sr. project idea even though it would need a microcontroller.	3	4.4	5
7.	The car project in the embedded systems class built up my confidence to work with microcontrollers.	2	4.0	5
8.	I could have started and completed my current Sr. project by the middle of junior year.	2	3.0	4

Table 2: Student Survey Results

For question 1, the minimum rubric value of 4 (agree) implies that most of them must be thinking about the embedded class they took about half a year ago and what they learned in that class. Additionally, they made a great effort to complete the final (car automation) project in that class which would really make them learn how to program microcontroller boards. These survey data validate the effort the instructor puts into the embedded course to make it a hands-on learning experience.

Students responding to question 2 must all have passed the embedded course, few individuals may not have done great on course assessments. The embedded course had about 10 short assessments, and two exams prior to the final hands-on exam (car project). Some students are very practical, but not savvy on theoretical assessments. The course grades showed several D grades. Therefore, it is not a surprise that there would be some who feel unsure about things even after they had done the tasks assigned. The final (car) project is a 2-person group effort, and hence some may have relied on the partner to do the complex programming tasks.

The responses to question 3 several months after the course indicate the impact the embedded systems course made on students. Details of an embedded controller are overwhelming, and the students put a fair amount of extra effort in that class compared to other technical courses. The average answers to question 4 points to a strong conviction among students that they learned embedded systems course well, and most likely do well in projects with a similar scope. This hands-on car project molded students to be comfortable taking up projects that rely on embedded systems programming. The low minimum value indicates that a few are still not sure, even after passing that project and the embedded systems course.

Question 5 tries to make students be comfortable in thinking about microcontrollers and their capstone project along with the fact that they have taken embedded systems course with its challenging final (car) project. A low value of 2 indicates that the student group thinks their project either does not include an embedded controller or does not require programming them explicitly (such as using a PLC or a similar device). However, all the projects done by the surveyed students had a microcontroller either integrated into (such as the SCADA safety panel) or used as a discrete component requiring it be programmed explicitly. The next three questions really inquire their comfort level in using a microcontroller without help from the faculty.

The high average value the question 6 received shows that students are confident in taking up the challenge to adopt a microcontroller to their project if it warrants a one. They would not be given the Dragon 12 boards they are familiar with to be used in senior projects. Students need to purchase the microcontroller boards they need for the project. Most of the time they purchase an Arduino or a Raspberry Pi and learn how to program it or adapt it to their project. While a vast amount of support software and libraries exist as open source, students have to take the initiative to determine the processor they need, purchase it, and make it work after programming on their own. This process was really aided by the experience they gained in the embedded class.

Students' response to question 7 varies from *disagreeing* to *strongly agree* with the average being *agree*. The low values might be from a few students who passed the final embedded systems class project because they were teamed up with high academic achievers. This class (car) project is complex and filled with so many unknowns that need to be sorted out by trial and error. Some of the unknown factors include the selection of the battery type and quantity, determining the parameters for pulse width modulation of motor drive signals, timing and delays for the sensors, mounting methods of the hardware board to the toy car, and determining whether the H-bridge driver circuit built-in to the board sufficient to drive the specific toy car used in the individual project. A successful project would have to determine many such features uniquely to the system each group would build.

The question 8 reassess students' self-confidence and perceived ability to handle microcontrollers. The low response value (2) means *disagree*, and high rubric implies an *agreement* (to the question) while the overwhelming majority said do not know (neutral). Capstone projects that require a microcontroller be explicitly incorporated and controlled (programmed) as a component would surely be very challenging to students if they have not taken a rigorous embedded course. Only a small fraction of students might have a prior microcontroller experience at high school or at student clubs/competitions sufficient enough to utilize them as discrete components in capstone projects, unless they take the embedded systems course. Those students who otherwise *agreed* (that they could have completed the capstone project in junior year) may have said so due to two reasons. One is that their current capstone project does not require direct programming of such a device. Their project may have originated from a summer internship they held and even funded by the employer. The other reason probably – the ideal outcome of a course - they feel that they have known the embedded systems material all along (which is not true). In summary, the responses to this question validate that the embedded systems class was a major factor in aiding students' choice of microcontrollers in their capstone project.

part IV: conclusion

At this university, the embedded systems course has been around for a few decades. It has evolved into a highly practical, but rigorous course throughout the semester. However, the course teaches technical skills in the subject matter and instill confidence in the use of microcontrollers, as evident from both the student surveys and the class performance of the self-navigating toy car project.

Most of the capstone projects designed and built by EE/COE undergraduates at this university tend to use a microcontroller of individual choice as the main data processor and the controller of their prototype device. The survey conducted in this research to gauge the confidence level of senior students in selecting projects relying on embedded controllers revealed that they have acquired a high level of sophistication and confidence from the embedded systems course taken prior to the capstone project. In the foreseeable future, the embedded systems course will be a part of the EE/COE curriculum until STEM related robotic programs in K-12 grades teach the majority of students the concepts and skills required to take on embedded controllers with little help in engineering colleges. The author will maintain the rigor with hands-on projects in the embedded course regardless of what type of development board or a processor the course would be based on.

The author plans to collect more specific data of student grades, major (EE or COE), etc., along with surveys for several academic years to see if correlations exist among those factors. Any useful findings will be published in a relevant venue.

References:

[1] M. Karunaratne, "A Self-Driving Car project in Embedded Systems Class," *the ASEE Annual Conference & Exposition, Columbus, OH, USA, June 25-28, 2017.*

[2] C. LeBlanc, C. Dundorf, M. Sabin, "An Engineering Technology Capstone Project: The Snow Load Network", *the ASEE Annual Conference & Exposition, Columbus, OH, USA, June 25-28, 2017.*

[3] H. Zhuang, Z. Roth, O. Masory, N. Erdol, "Assistive Technology in Capstone Engineering Design," *the International Symposium on Computer, Consumer and Control (IS3C), Taichung, Taiwan, July 4-6, 2016.*

[4] M. L. Nelson, D. Rice, "Introduction to Algorithms and Problem Solving," *the Frontiers in Education Conference, Kansas City, MO, USA, October 18-21, 2000.*

[5] M. J. Paulik, M. Krishnan, “A Competition-motivated Capstone Design Course: the Result of a Fifteen-year Evolution,” *IEEE Transactions on Education*, vol. 44, issue 1, Feb. 2001.

[6] D. R. Raymond, D. J. Welch, D.J., "Integrating Information Technology and Programming in A Freshmen Computer Science Course," *the Frontiers in Education Conference, Kansas City, MO, USA, October 18-21, 2000.*

[7] S. Ludi, J. Collofello, "An Analysis of the Gap Between the Knowledge and Skills Learned in Academic Software Engineering Course Projects and those Required in Real Projects," *the Frontiers in Education Conference, Reno, NV, USA, October 10-13, 2001*.

[8] S. Fincher, "What are We Doing When We Teach Programming?," *the Frontiers in Education Conference, San Juan, PR, USA, November 10-13, 1999*.

[9] E. Frontoni, A. Mancini, F. Caponetti, P. Zingaretti, "A Framework for Simulations and Tests of Mobile Robotics Tasks," *MED '06, The 14th Mediterranean Conference on Control and Automation, Ancona, Italy, June 28-30, 2006.*