A Project-based Computer Engineering Curriculum

Prof. Jeremy N. Thomas, DigiPen Institute of Technology

Jeremy Thomas has been at the DigiPen Institute of Technology since 2010. He has a BA in Physics from Bard College, and a MS in Physics and a Ph.D. in Geophysics both from the University of Washington (UW).

Before joining DigiPen, he was a Postdoc at the U.S. Geological Survey and held faculty positions at the UW and Bard College. Currently, he holds affiliate positions at the UW and NorthWest Research Associates. His areas of specialization are space physics and electrical engineering, including atmospheric electricity, radio wave propagation, and digital signal processing.

He and his students are currently conducting research in both geophysics (e.g., fusion of lightning and satellite data) and computer engineering (e.g., human interface devices and handheld gaming consoles). He has authored more than 20 peer-reviewed publications, including some with DigiPen students.

Christopher Theriault, DigiPen Institute of Technology
Prof. Charles Duba, DigiPen Institute of Technology
Dr. Lukas P van Ginneken, Digipen Institute of Technology

Lukas van Ginneken is a professor of computer engineering at Digipen Institute of Technology. His interests are digital logic, field programmable gate arrays, computer architecture, hardware description languages and microcontrollers. Formerly Mr. Van Ginneken was Chief Scientist and a co-founder of Magma Design Automation Inc. Before joining Magma, he was at Synopsys, where he worked in the Design Compiler group and the Advanced Technology Group. From 1989 to 1995 he was at IBM's T.J. Watson Research Center, where he worked on logic synthesis. Mr. van Ginneken holds a Ph.D. in Electrical Engineering from Eindhoven University.

Nicholas James Rivera
Mr. Brian Michael Tugade, DigiPen Institute of Technology

©American Society for Engineering Education, 2015

A Project-based Computer Engineering Curriculum

Abstract

This paper documents an innovative, project-based approach to teaching computer engineering. A project-based undergraduate computer engineering curriculum, with an embedded systems focus, has been offered since 2004 at a small, private college in the Northwestern US. The main goals of the curriculum are twofold. The first is to engage students in engineering problems starting in the first semester of the Program, thus providing them with a sense of pride and ownership in their work. The second is to prepare students for engineering careers by involving them in complex, team projects, which are typically only conducted outside of required undergraduate coursework, at the graduate level, or in industry.

Most undergraduate computer engineering programs require a 1-year senior capstone design course. In this Program, team projects start in the first semester and are required in each of the following semesters. As they develop through the Program, students have increasingly more creative control over their projects and are responsible for component selection, design, testing, and implementation of their own hardware and/or software systems. Design constraints that are encountered in industry are followed, such as developing use models, cost, power, and portability. Examples include robotic toys, human interface devices, hand-held gaming consoles, and a stratospheric balloon data acquisition / telemetry system. These projects complement rigorous coursework in computer science, engineering, programming, mathematics and physics.

In this paper, we discuss the pedagogy of project-based learning and provide a survey of some existing computer engineering programs. We describe our project-based curriculum in detail, including examples of student projects. Student outcomes related to both technical and soft skills are assessed using student surveys and project evaluation rubrics. We discuss these assessment results and highlight some successes and limitations of the experiential curriculum.

Introduction

The computer engineering discipline is the science and technology of design, prototyping, implementation, testing, and maintenance of computing systems, including software and hardware[1]. How to best educate undergraduate computer engineering students is not at all clear. This is in part due to the nature of the discipline, which has continually changing technology and organization. In this paper, we describe a project-based computer engineering curriculum, which complements more traditional lectures and laboratory courses. We compare this curriculum with curricula from other small universities in the US. We show how our curriculum is in agreement with a mixed-mode approach that combines projects with traditional techniques. An assessment of student outcomes is presented and successes and limitations are discussed.

Critical issues in engineering education

In 2003, Mills and Treagust[2] summarized the critical issues in engineering education as identified by accrediting bodies and industry as:

1. Engineering curricula are too focused on engineering science and technical courses without providing sufficient integration of these topics or relating them to industrial practice. Programs are content driven.
2. Current programs do not provide sufficient design experiences to students.
3. Graduates still lack communication skills and teamwork experience and programs need to incorporate more opportunities for students to develop these.
4. Programs need to develop more awareness amongst students of the social, environmental, economic and legal issues that are part of the reality of modern engineering practice.
5. Existing faculty lack practical experience, hence are not able to adequately relate theory to practice or provide design experiences. Present promotion systems reward research activities and not practical experience or teaching expertise.
6. The existing teaching and learning strategies or culture in engineering programs is outdated and needs to become more student-centered.

Project-based learning

To address some of these issues, some universities have implemented project-based learning into their curricula. As described by Mills and Treagust[2] and Perrenet et al.[3], project-based learning is characterized by the following:

- A large number of phases or stages through which to pass during the project.
- Student-initiated research is relied upon for the student to progress through the project as well as for their own learning.
- Require high levels of student initiative; students need to develop motivation and organization skills.
- Open-ended outcomes: allowing the student the opportunity to choose, after appropriate research, an outcome that interests them.
- Observational skills are identified as having a high priority, especially in the initial stages during identification of the problem.
- Student reflection is important. They are encouraged to evaluate fully the outcomes they have achieved.
- Rely on team-work.
- Are often multiple terms in duration.
- Project work is more directed to the *application* of knowledge, whereas problem-based learning is more directed to the *acquisition* of knowledge.
- Project-based learning is usually accompanied by subject courses (e.g., mathematics, physics, etc. in engineering).
- Management of time and resources by the students as well as task and role differentiation is very important in project-based learning.
- Self-direction is stronger in project work, compared with problem-based learning, since the learning process is less directed by the problem.

Example project-based learning curriculum

As described by Mills and Treagust[2], Aalborg University in Denmark has had a project-based engineering program since 1974 when the university was founded. The curriculum includes 50%

project work, 25% course work in engineering domain-specific topics, and 25% coursework in foundational topics such as mathematics and the sciences. The updated curriculum at Aalborg is found on their website[4]. Aalborg was in the unique position to make a direct comparison of student outcomes with traditional engineering programs at Denmark Technical University. According to Mills and Treagust[2], Aalborg graduates had better team and communication skills and were better at completing larger projects. Thus, they were more directly employable after completing the programs. Denmark Technical University graduates had a stronger grasp of engineering fundamentals, but often needed additional training prior to employment or on-the-job training. The retention rates were noticeably different. Aalborg's overall retention rate was 75-80%, whereas Denmark Technical University was about 60%.

Conclusion from Mills and Treagust[2]

In their study of project-based learning, Mills and Treagust[2] conclude by recommending mixed-mode curricula that include both project and traditional coursework. They suggest that this is the best way to satisfy the needs of industry and also retain the engineering fundamentals. They emphasize that this approach must start in the early years of the program, with the project-based components growing in extent, complexity and autonomy as the students move through the program.

Other computer engineering programs at small US institutions

Here we briefly describe computer engineering curricula at other small, undergraduate focused institutions: the Milwaukee School of Engineering, Rose-Hulman Institute of Technology, and Harvey Mudd College. All of these programs are somewhat mixed-mode in nature and include some project-based courses. However, most project-based courses are reserved for the 3rd or 4th year of study, unlike DigiPen Institute of Technology which includes project-based courses in every semester.

The Milwaukee School of Engineering (MSOE) computer engineering curriculum is somewhat mixed-mode and includes lectures, labs, and hands-on activities[5]. The fully project-based experiences are reserved for the three courses in the senior design sequence. According to Meier et al. 2007[6] and 2008[7], MSOE redesigned their computer engineering curriculum starting in 2006. They had two main goals. First, balance topics such that each academic quarter must include one course in computer software, one in computer hardware, and one in math or science, and second, include computer engineering specific courses in the 1st year. The freshman-first approach was implemented by moving digital logic design and hardware description languages into the 1st year of the program. Meier et al. 2008[7] reported that these curriculum changes increased first-to-second year retention from 60-70% to 78%. Since 2008, MSOE made some additional changes, including moving more CE courses to 1st year and moving physics courses to the 2nd year[5].

Rose-Hulman's computer engineering program includes five project-based courses: a 2-credit 1st year design course and four 3-credit design courses in 3 and 4th year[8]. Like MSOE and DigiPen Institute of Technology they have CE courses in the 1st year: dc circuits, an introduction to signal processing, and software development. Harvey Mudd College has a general engineering degree program where students can focus in specialized areas such as computer engineering. The program has five project-based courses, including a design course in the 1st year, an engineering systems course in the 2nd year, and three clinic courses in the 3rd and 4th years where students work on projects for an outside client[9].

History of DigiPen Institute of Technology computer engineering program

DigiPen Institute of Technology is an educational institution offering eight undergraduate and two graduate degree programs. In addition to the main campus in Redmond, Washington, there are branch campuses in Spain and Singapore. The degree programs range from digital arts and music to computer science and computer engineering. DigiPen Institute of Technology began offering the Bachelor of Science in Computer Engineering (BSCE) degree in 2004. The first cohort of students graduated from the Program in 2007, and the Program has graduated 29 students in total from 2007 to 2014

Overview of curriculum

All the Electrical & Computer Engineering (ECE) Department project courses include significant design experience to prepare students for engineering practice. Students are required to take eight project courses, one each semester starting in the first semester of the Program (CS 100L, ECE 110, ECE 220L, ECE 260, ECE 310L, ECE 360L, ECE 410L, & ECE 460L). The overall course sequence can be seen in Figure 1. The degree consists of 154 credits over 8 semesters with 17 – 20 credits per semester.

In these courses, students apply knowledge and skills acquired in other courses (e.g., mathematics, physics, computer programming, electronics, and communication) to design, build, program, document, and test interactive embedded devices, such as robotic toys or handheld gaming systems. Students, working in small teams, are expected to integrate a microprocessor with various peripheral devices such as storage, input, sensors, and display devices into a portable embedded platform. These projects follow design constraints that are encountered in industry such as use model, cost, power, and portability. Moreover, students are expected to develop team management skills, presentation skills, and critical design processes, as well as study and implement human-machine interaction.

In addition to the project work, these courses have weekly lectures related to engineering practice. Topics include the history of computer engineering, the electronics development cycle, professional ethics, multidisciplinary team environments, common development tools used in industry, communication and professional skills (e.g., interview preparation, resume/CV writing, and presentations), engineering management, testing and quality control, and statistical methods.

The project courses are designed to support student outcomes that are recommended by ABET. A summary of the courses and these outcomes is included in Table 1a and 1b. This table is then followed by an analysis of each individual course.

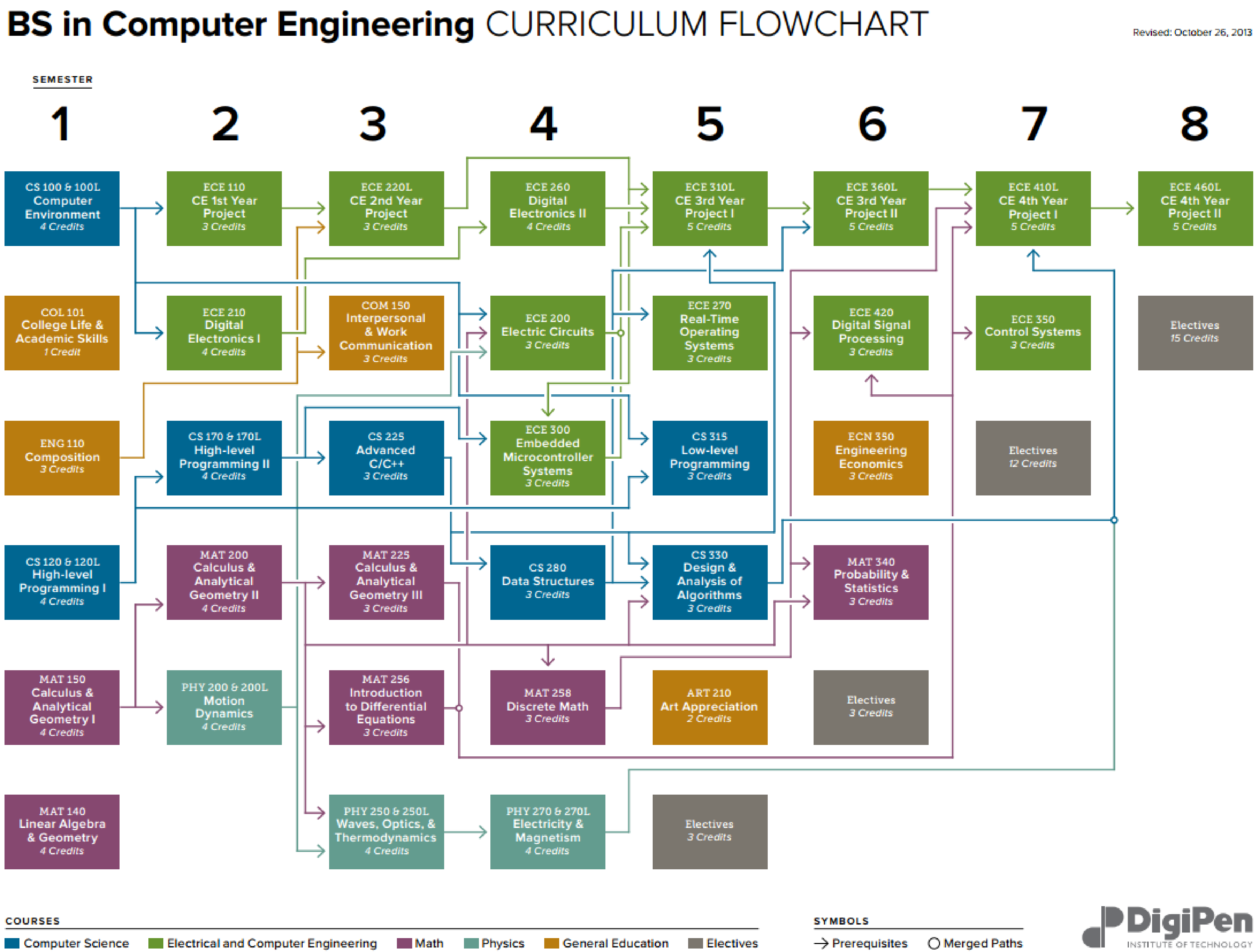

Figure 1: BSCE curriculum flowchart

Table 1a: BSCE project course educational outcomes (a-e).

Course\Outcome	(a) An ability to apply knowledge of mathematics, science, and engineering	(b) An ability to design and conduct experiments, as well as to analyze and interpret data	(c) An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability	(d) An ability to function on multidisciplinary teams	(e) An ability to identify, formulate, and solve engineering problems
CS 100	Students apply this knowledge to successfully program an autonomous car	Students must debug and analyze the code for their autonomous car	The autonomous car design includes certain specified behaviors	Work on the autonomous car is team-based	Students must solve problems related to navigating autonomously
ECE 110	Students apply this knowledge to design their project	Subsystems of the project must be tested for functionality	The students design a project application that is suitable for the hardware being provided to them	Students work in teams and peer review is part of the project evaluation	Problems with the project must be identified and corrected; the device must be designed with certain requirements
ECE 220L	Students apply this knowledge to design their device	Subsystems of the device must be tested for correct behavior	The project must be developed with a budget, be RoHS and ITAR compliant	Students develop writing skills by producing technical and marketing documentation	Problems with the device must be identified and fixed; the device must be designed with certain requirements
ECE 260	Students apply this knowledge to design their Verilog project	Simulation tools are used to validate code prior to synthesis	The students design a project application that is suitable for the hardware being provided to them	Students participate in group work during class sessions	Problems with the project must be identified and corrected; the device must be designed with certain requirements
ECE 310/360/410/460L	Students apply this knowledge to design their device	Subsystems of the device must be tested for functionality	The project must be developed with a budget, be RoHS and ITAR compliant	Students work in teams and peer review is part of the project evaluation	Problems with the device must be identified and corrected; the device must be designed with certain requirements

Table 1b: BSCE project course educational outcomes (f-k).

Course\Outcome	(f) An understanding of professional and ethical responsibility	(g) An ability to communicate effectively	(h) The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context	(i) Recognition of the need for, and an ability to engage in life-long learning	(j) Knowledge of contemporary issues	(k) An ability to use the techniques, skills, and modern engineering tools necessary for engineering practices
CS 100	**(Not emphasized in this course)**	**(Not emphasized in this course)**	**(Not emphasized in this course)**	**(Not emphasized in this course)**	**(Not emphasized in this course)**	Students use lab equipment to implement circuits
ECE 110	Students are expected to act in a professional manner on their team	All students are expected to participate in their team presentations and document their work	Students must write a report outlining the impact of their project in these types of areas	Students are introduced to doing research as part of project design	Lecture and reading material addresses contemporary topics	Students use lab equipment to implement the project
ECE 220L	Students are expected to act in a professional manner on their team	All students are expected to participate in their team presentations	Students must write a report outlining the impact of their device in these types of areas	Students are required to participate in a professional organization relevant to their field	Lecture and reading material addresses contemporary topics	Students use lab equipment to implement the project
ECE 260	Students are expected to act professionally, behave with courtesy and respect, and exhibit academic integrity	Students present written reports on their work	**(Not emphasized in this course)**	Students are introduced to doing research as part of project design	Lecture and reading material addresses contemporary topics	Students use lab equipment to implement the project
ECE 310/360/410/460L	Students are expected to act in a professional manner on their team	All students are expected to participate in their team presentations and document their work	Students must write a report outlining the impact of their device in these types of areas	Students are required to participate in a professional organization relevant to their field	Lecture and reading material addresses contemporary topics	Students use lab equipment to implement the project

First year courses: CS 100 & ECE 110

In CS 100, Computer Environment, students are exposed to number systems, Ohm's Law, logic gates, and how these elements can combine to form basic computer hardware such as memory. Students then spend the final third of the course working on generating code for an autonomous car. The car consists of a standard commercial radio-controlled racing vehicle that has the radio receiver replaced with an 8-bit PIC microcontroller. The microcontroller is then responsible for the motor and steering controls, based on input it receives from IR sensors attached to the vehicle. To help students learn basic computer architecture, they are required to write their programs in assembly. At the end of the semester, students race their vehicles through a course where they earn points for speed and avoiding obstacles. An image of the complete platform is included as Figure 2. This course currently has over 200 students each year. Students participate in a single lecture session and then split into lab sessions that may have 50 – 75 students in each section.

The CS 100 course is common to all BS students regardless of degree. The percentage of BSCE students enrolled is typically 10% or less. In the first two-thirds of the course, the practical work revolves around building simple circuits (such as using relays to make a flip-flop) as would be found in a traditional lab course. Students who wish to do well on the autonomous car portion of the course must spend much more development time on the project than is available during the scheduled lab hours. Since this course appears in the first semester of the Program, students are not expected to work on certain aspects of project development, such as design proposals or project timelines. These aspects of project work are developed later on in separate courses. The focus for this course is to immediately get students working on something "hands-on". This is a contrast to some programs such as that at Aalborg University, where the first semester project courses (`Technological Project Work`) focus on teaching project development principles ahead of project completion. Other institutions, such as Rose-Hulman, have taken a similar approach (students create a robot in their ECE 160 course, Engineering Practice).

Anecdotal evidence suggests that the results of this approach are mixed. The end-of-year competition is generally a great source of excitement for everyone, but the majority of the students are not computer engineering majors. In these cases, this course could be their last experience interacting at a low-level with a microcontroller. For these students, a small percentage will express a new interest in computer engineering, but many more find themselves being turned off to it due to a misperception that computer engineering is all about programming in assembly.

In ECE 110, CE 1st Year Project, students are exposed to common engineering hardware and software tools (MATLAB, SPICE modeling, lab equipment, etc.) and given a project to work on. The project development cycle is further developed, but at this time, it is still incomplete. As in the CS 100 course the students completed in their previous semester, the students are assigned a project that must be completed to meet certain requirements. In this case, the project is to develop a sensor & transmitter system that can be deployed on a weather balloon at the end of the semester. Unlike with the autonomous car project, students have some options on how the project is to be completed: they choose from a variety of sensors to use and follow constraints of cost, size, and weight. Thus, their ability to perform design work is furthered. The schematic of the analog sensor circuit is included as Figure 3. The 555-timer in this circuit would then be replaced by a microcontroller in further iterations of the circuit. This course currently has 6 – 10 students each year.

The lab component of the course requires the students to design and implement a simple analog sensor circuit. The sensor must measure a property of the atmosphere that changes with altitude other than temperature. Examples include pressure, humidity, light & other wavelengths of EM radiation, wind, etc. This continues to build up their practical experience with implementing dc circuits and using test equipment such as multimeters and oscilloscopes. Later in the course students interface their chosen sensors with a microcontroller. However, in this case, unlike in the previous course, the students are allowed to write their code in the C programming language (although assembly is allowed, if desired). Also, all students are expected to use CAD design tools to create a PCB for their sensor system, which is fabricated for the students to populate. There are two main motivations for this. First, having their system on a PCB instead of a regular solderless breadboard helps to meet the weight requirements for the balloon launch. Second, providing students with early exposure to the PCB design process will improve their efficiency in later projects by minimizing the time required to get a custom PCB up and running. Currently, a student's first attempt at completing a PCB may come in their 3rd or 4th year project when a more complex design is required. In such cases, their project development time is greatly increased. This course is being offered for the first time in the Spring 2015 session. We anticipate that the balloon launch will provide a similar motivating experience to the CS 100 car races and will continue to build students' excitement with the Program.

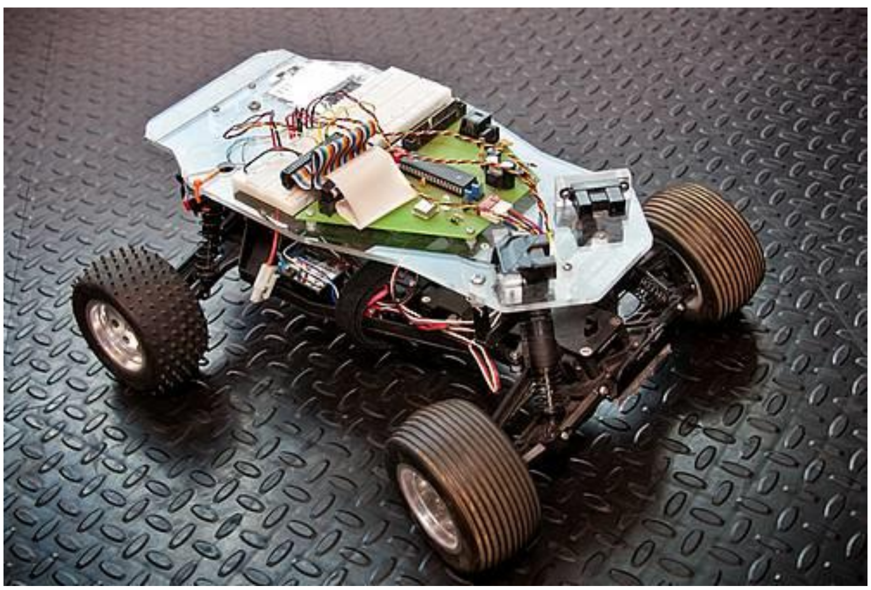

Figure 2: First year project**,** CS 100 car.

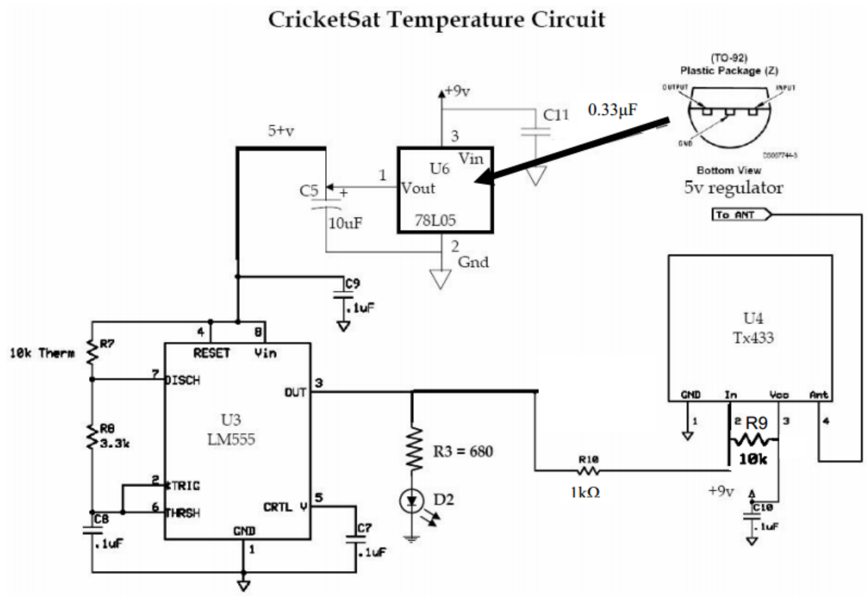

Figure 3: ECE 110 analog sensor circuit (courtesy University of Washington, ESS 205 course).

Second year courses: ECE 220L & ECE 260

Sophomore students typically begin with ECE 220L, CE 2nd Year Project, in which students learn more about and are expected to follow the complete project development cycle for the first time: design, implementation, testing, and validation. The students' ability to write technical documentation is also further developed as they must create project timelines, test plans, user manuals, etc., for their project. The project is more open-ended than the first year work. Students are given a set of constraints for the project (such as 'must interact with the environment' or 'must be self-powered') and a budget. The constraints are such that a typical project team will develop a robot, but non-robotic projects are encouraged if they still meet the required number of project criteria. This course currently has 4 – 6 students each year.

Students receive lectures on project development and other topics of interest such as basic sensors, soldering, and simple communication protocols such as UART. Students are also exposed to how the social sciences interact with engineering. For example, in previous offerings of the course, students have watched films that depict how technological changes affect society and how women are represented in the workplace. As the DigiPen Institute of Technology curriculum currently does not have humanities courses that explicitly relate social issues to engineering, it was decided to use some of the upper-level project courses as a venue for this material. Other peer institutions, such as the Milwaukee School of Engineering, do in fact have courses in the curriculum to discuss issues such as these. However, in cases such as this, the institution typically has fewer major project courses compared to DigiPen Institute of Technology.

Over the course of the semester students typically learn how easy it is to over-scope a project that has a hard deadline for completion. Insufficient time is spent on the integration of the subcomponents worked on by the various team members, and as a consequence there is a minimal amount of time available for debugging and improvements (typically, this is done the last few days before the project is due). An example project from 2014 is included as a photograph in Figure 4.

The student derived use-case for the project is as follows:

Project: Four-Legged Friend
Team: Jimi Huard and Jason Dempsey

"The FLF (Four-Legged Friend) is a first aid supply platform and general first responder tool. Utilizing Quadrupedal movement, the FLF provides stable, autonomous support to hazardous situations. Utilizing two metal feelers, it can avoid obstacles, navigate to a person or persons in distress, and provide immediate relief in times of crisis.

The basic operation of the FLF is split into 2 main components: a rescue body, and a command station remote. The body sports the sensing equipment, a wireless communications module, an MBED micro-controller (to make decisions on the fly), regulates power consumption, and allows for the housing of aid provisions. It can also operate autonomously from its counterpart, the controller.

The controlling remote – aptly dubbed the FLF Controller – features a pair of joysticks, a Texas Instruments Tiva C Launchpad microcontroller (for communications and analog-to-digital conversion), as well as a wireless communications module. While not necessary for the FLF's body component to function, it offers a way for outside influence over its operation. The remote does this by allowing autonomous operation to be enabled or disabled by the simple press of a button, handing full control of the FLF to the remote operator."

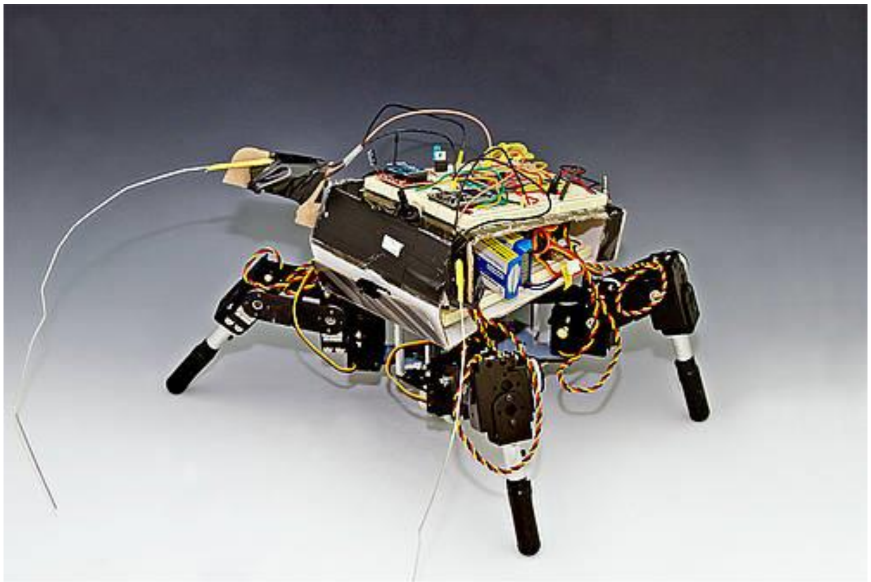

Figure 4: Second year project, the Four-Legged Friend.

ECE 260, Digital Electronics II, takes students into the realm of Hardware Description Languages (HDLs), typically, Verilog is used. In addition to learning the Verilog language during the lectures, students spend the lab hours running their Verilog code under simulation and compiled on a FPGA. The course concludes with a FPGA project. Development of the project includes writing a proposal and a final project paper in addition to the project itself. As the ECE 260 course is not a major project course, requirements for the project papers are less stringent than for the project courses. For example, the bill of materials and test plan might be omitted, and students might skip researching what existing projects have already been implemented using similar designs.

An image of a sample ECE 260 project is included as Figure 5. This project includes building a simple 2d game that outputs to a VGA port and takes joysticks as input. The project is written in Verilog and implemented on the Altera DE-2 development board.

The abstract from the student paper is as follows:

Project: FPGA Controller and Platformer Implementation Project (F.C.P.I.P.)
Team: Tyler McGrew

"In honor of the significance of Atari in both the gaming and engineering circles this project combines Atari's 2600 controller with a game on the Altera DE2 Education board. To keep with the feel of the Atari controller my game is very minimalistic similar to how games used to be. The game is a platformer where the player is a small square that is trying to make it to the top of the screen by jumping between different platforms. It is set up to encourage multi-player competition throughout an infinite swarm of random levels. The game itself was programmed using Verilog. The controller is connected using a custom made adapter that plugs into the controller port and the GPIO pins on the DE2 board."

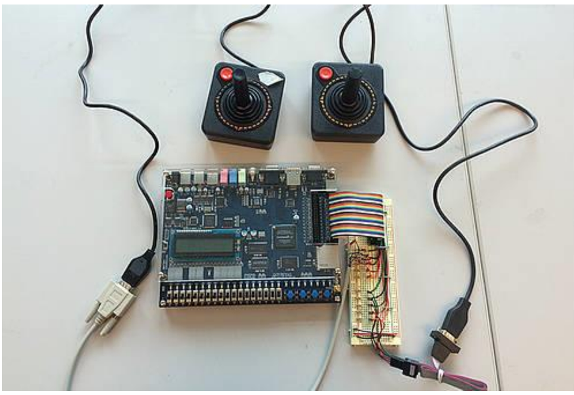

Figure 5: Second year project**,** F.C.P.I.P.

Third and fourth year courses

ECE 310L, 360L, 410L, and 460L make up the two year-long CE 3rd Year and CE 4th Year project courses. It is generally expected that a student will work on two separate year-long projects, but circumstances may allow for other combinations. For example, a really dedicated team might spend two years working on a single project, or a student working without a team may end up implementing two semester-length projects in one year. This course currently has 4 – 6 students each year. The content between the 300-level and 400-level projects is similar enough to be described together. For the final two projects students have numerous responsibilities:

- Prepare a project proposal that is delivered to their peers and faculty members.
- Perform a literature search to compare and contrast their project from existing work.
- Develop a viable use-case for their project.
- Develop the set of system specifications for their project.
- Design and implement the system.
- Submit the project development timeline.
- Complete the project on-time and within budget.
- Validate and test the design and demonstrate that it meets the design specifications.
- Give milestone presentations reporting to their peers and faculty their current progress.
- Produce a final project paper detailing the above items.

The project content is very open-ended. Traditionally, the 3rd year project focused on producing a working game console system, in keeping with DigiPen Institute of Technology's history with the computer game development industry. This option remains popular with students although other projects can be chosen. It is up to the student's team to convince the faculty to support the idea via their project proposal. Some of the projects outside of gaming systems that have been developed include: a portable real-time radar system building on work done at MIT, a custom programming language designed to be used with embedded systems, and an improved routing algorithm for mesh networks.

Lecture content continues to focus on project development. Additionally, "soft skill" topics are presented, such as resume writing and interviewing, to prepare students for a transition into the workforce (DigiPen Institute of Technology also offers an elective course, COL 499, which deals exclusively with such topics). Some lectures may also be geared to topics of particular relevance to the students' projects. Due to the open-ended nature of the allowed project specifications, it is almost inevitable that students will be working with components or techniques that were not covered in their regular coursework. In such cases, students must work with faculty outside of regular class hours to learn about these topics or do extensive research on their own. The upper-level projects will not be successful unless the students learn to engage in a large degree of self-directed learning, which we feel is vital for all engineers. This philosophy is also found in all of DigiPen Institute of Technology's programs, not just the BSCE.

Figure 6 shows the early prototype of the radar project. The core circuit was later migrated from the solderless breadboard to a PCB designed and populated by the student. This setup uses coffee cans for the antenna, and the student discovered that these cans worked just as well, if not better than, other commercial antennas that were available.

The abstract for this project is as follows:

Project: Microwave Radar with Real-Time Data Display
Team: Kevin Secretan

"Microwave frequency and radio frequency radiation is all around us with many wireless devices, such as wi-fi routers or Bluetooth headphones, constantly transmitting and receiving digital data with other devices. But at the analog level, basic physical principles about the transmitted signal itself allow for a low-cost device to be built that can transmit and, from information gathered by the signal reflection from the environment, infer knowledge about the environment. While radar systems have seen increasingly sophisticated applications since they were first developed in the 1930s, perhaps the most common public association is with police “radar guns” which catch speeders. My project, which is heavily based off of MIT's OpenCourseWare radar design, is a portable radar device that when connected to any computer or laptop is capable of displaying the reflected signal information in real-time and thus allowing for a qualitative assessment of a moving target's speed or distance relative to the device. This is the first step toward my original goal of eventually displaying position information of interesting objects in the device's surrounding environment, such as people and walls. The current version of my project represents my improved understanding of the underlying electromagnetic radiation principles and of signal processing in Python, both of which are necessary to move on to my other project goals.”

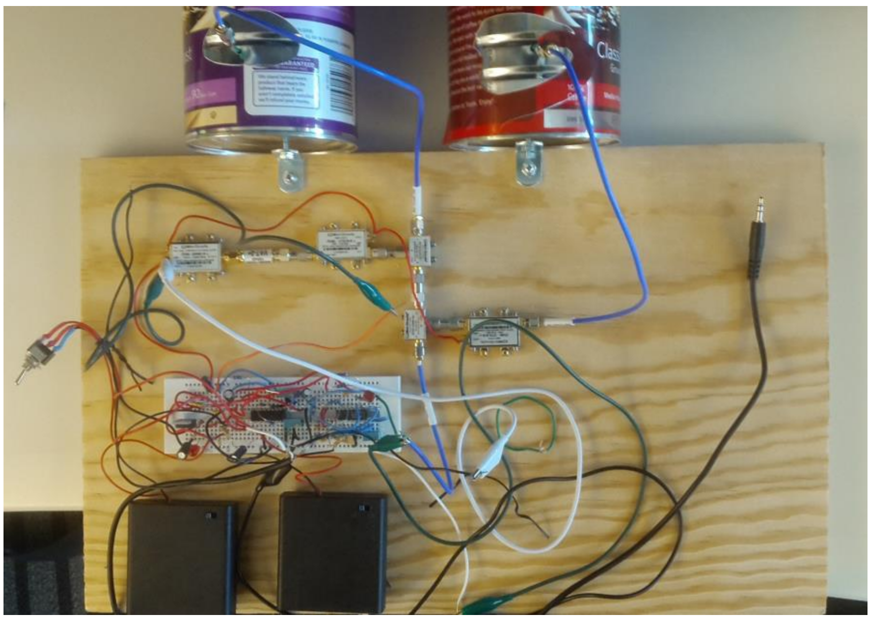

Figure 6: Third and fourth year project, Microwave Radar.

Assessment data

Table 2 shows the student survey results regarding the desired student outcomes of the Program. In this table, each row represents responses from a single student. Table 3 shows the results of the faculty survey regarding the same outcomes. In this table, each row represents the faculty opinion of all of the students who took the listed course in that semester. For some of the courses, a particular outcome may not have been assessed for that course, in which case it is marked `NE`.

Table 2: Student evaluation of student outcomes.

	apply knowledge	design experiments	follow constraints	teams	identify problem domain	ethics	communication	broad education	life-long learning	contemporary issues	modern skills & tools
Date of Survey: Fall 2013											
Year of Enrollment:	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**
2007	4.00	4.00	4.00	4.00	5.00	5.00	4.00	4.00	4.00	4.00	5.00
2007	4.00	4.00	5.00	5.00	5.00	4.00	4.00	4.00	4.00	4.00	4.00
2009	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
2009	4.00	2.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
2010	4.00	4.00	2.00	2.00	5.00	1.00	3.00	1.00	5.00	3.00	5.00
2010	5.00	4.00	3.00	3.00	4.00	3.00	4.00	3.00	5.00	4.00	3.00
2010	5.00	5.00	5.00	2.00	5.00	2.00	4.00	4.00	5.00	3.00	5.00
2011	4.00	4.00	1.00	1.00	3.00	3.00	2.00	4.00	4.00	3.00	4.00
2011	5.00	5.00	5.00	5.00	5.00	4.00	4.00	4.00	5.00	4.00	5.00
2012	4.00	5.00	4.00	3.00	5.00	4.00	3.00	2.00	4.00	4.00	5.00
2012	4.00	4.00	4.00	4.00	3.00	5.00	4.00	4.00	3.00	4.00	3.00
2012	4.00	4.00	3.00	3.00	4.00	4.00	4.00	4.00	3.00	3.00	4.00
2012	4.00	5.00	4.00	5.00	5.00	5.00	5.00	5.00	5.00	4.00	5.00
2013	4.00	3.00	2.00	2.00	2.00	2.00	2.00	3.00	4.00	4.00	2.00
Date of Survey: Spring 2014											
2009	5.00	4.00	4.00	4.00	5.00	4.00	4.00	3.00	5.00	4.00	5.00
2009	5.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
2011	4.00	3.00	4.00	5.00	5.00	4.00	3.00	3.00	4.00	4.00	4.00
2012	4.00	4.00	3.00	3.00	5.00	3.00	3.00	3.00	4.00	3.00	5.00
Average	4.28	4.00	3.61	3.44	4.33	3.61	3.61	3.50	4.22	3.72	4.22
Median	4.00	4.00	4.00	3.50	5.00	4.00	4.00	4.00	4.00	4.00	4.00

Key: 1.00 Strongly Disagree; 2.00 Disagree; 3.00 Neutral; 4.00 Agree; 5.00 Strongly Agree

Each row is a response from a single student

Table 3: Faculty evaluation of student outcomes.

Date of Survey: 2013-4 Year of Enrollment:	A apply knowledge	B design experiments	C follow constraints	D teams	E identify problem domain	F ethics	G communication	H broad education	I life-long learning	J contemporary issues	K modern skills & tools
ECE 210	3.75	3.00	3.50	NE	3.40	NE	3.50	NE	3.25	2.60	4.00
ECE 220L	3.40	3.33	3.00	4.00	3.20	3.25	3.50	2.00	3.17	3.20	3.20
ECE 260	4.00	3.00	3.67	3.75	4.25	3.50	4.75	NE	3.67	4.50	4.40
ECE 300	3.33	2.00	2.00	NE	4.00	3.00	NE	NE	NE	3.00	3.00
ECE 310L	NE	NE	3.33	3.75	3.80	NE	4.50	NE	4.50	3.40	3.00
ECE 350	3.33	NE	NE	NE	3.40	NE	NE	NE	4.00	4.20	3.25
ECE 360L	NE	NE	4.00	4.25	3.80	NE	4.75	NE	4.50	3.80	4.00
ECE 410L	4.29	3.60	3.67	4.50	4.60	3.00	4.33	2.33	4.33	4.60	4.33
ECE 420	3.33	NE	NE	NE	3.60	NE	NE	NE	3.50	4.00	3.50
ECE 460L	4.29	3.60	3.67	4.50	4.60	3.50	4.33	2.33	4.33	4.60	4.33
Average	3.72	3.09	3.35	4.13	3.87	3.25	4.24	2.22	3.92	3.79	3.70
Median	3.58	3.17	3.58	4.13	3.80	3.25	4.33	2.33	4.00	3.90	3.75

Key	1.00	2.00	3.00	4.00	5.00
	Failure	**Unsatisfactory**	**Developing**	**Satisfactory**	**Exemplary**

Each row represents the faculty evaluation of all the students who completed the course that semester

The full descriptions of the student outcomes are as follows:
Students should have…

a) an ability to apply knowledge of mathematics, science, and engineering;
b) an ability to design and conduct experiments, as well as to analyze and interpret data;
c) an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability;
d) an ability to function on multidisciplinary teams;
e) an ability to identify, formulate, and solve engineering problems;
f) an understanding of professional and ethical responsibility;
g) an ability to communicate effectively;
h) the broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context;
i) a recognition of the need for, and an ability to engage in life-long learning
j) a knowledge of contemporary issues;
k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

As can be seen by contrasting the two sets of results, the students are generally satisfied that they are meeting these outcomes, while the faculty are more neutral on the subject. In both sets of data, the lowest marks come in areas related to the humanities, understanding of ethics, communication skills, knowledge of contemporary issues, etc. This is likely connected to the fact that the BSCE curriculum at DigiPen Institute of Technology does not require any courses on these topics except for COM 150, Interpersonal and Work Communication. The Program attempts to compensate for this by including such material in the lecture portion of the various project courses. The demanding workload of the project courses also leaves students with little time to devote to these areas on their own.

DigiPen Institute of Technology has begun the formalized assessment process recently. It will continue to assess the students at regular intervals and work to identify trends in the data.

Discussion of successes and limitations

The CE curriculum at DigiPen Institute of Technology follows the suggestions of previous studies and reports, such as Mills and Treagust[2], and includes both project courses and traditional lecture & labs in each semester of the program. Like most ABET accredited CE Programs, the curriculum has electrical & computer engineering, computer science, and math, science, and elective topics. Project courses emphasize technical skills, industry applications, and soft skills such as oral and written communication.

Assessment data for 2013-14 gives a snapshot of how well the program is achieving student outcomes. Students are doing well in the practical engineering skills they require to develop projects. However, they may be falling behind in skillsets derived from studies in the humanities. It is recognized that an engineer's understanding of the impact of their work is as important as the work itself, but what the proper balance between the two is, is not clear.

We also note that the overall job placement for the Program is approximately 94 percent within one year of graduation, which is similar to CE programs at MSOE, Rose-Hulman, and Harvey Mudd (as reported on their websites). This suggests that graduates of the Program have the job skills in demand by employers. Moreover, graduates are employed in many different fields including embedded systems, software and firmware engineering, hardware engineering, and digital signal processing.

There are limitations with a mixed-mode curriculum. It requires adding project courses in addition to traditional courses. The DigiPen Institute of Technology CE Program requires 154 semester credits, more than the typical 128 semester credits or equivalent at other US institutions. Thus, the workload is heavy, and it is difficult for most students to complete the program in eight semesters. Indeed, we have found that most students need nine or 10 semesters to complete the Program. The heavy workload also affects the retention rate for the Program, which is historically about 50%.

To help reduce the workload each term, without removing courses from the curriculum, more courses have been offered during the summer semesters. Moreover, a new scholarship program that provides a 25-75% tuition reduction for up to 10 semesters of study has been implemented starting in 2014. To assist with student planning, we provide incoming students with both 8-semester and 10-semester plans for completing the program.

There are currently two full-time faculty, one part-time faculty, and three full-time staff members available to support the computer engineering students. As the total enrollment of the computer engineering Program varies between 14 – 25 students each year, the student-to-faculty ratio is extremely low. Due to the high expectations and demanding workload of the project courses, students rely heavily on the availability of faculty and staff for support. While this does allow the students to be successful, it is a limitation in that is does not easily scale for increased Program enrollment.

Project costs are another limitation. While students are expected to follow budgetary restraints, there is some flexibility in this area. Students may work on project concepts that necessitate more expensive components provided they can convince the faculty during the design phase that they are capable of completing the project. Students are not expected to pay for their project supplies out-of-pocket. Indeed, DigiPen Institute of Technology currently does not even require students to pay extra lab fees to participate in the BSCE Program. If the Program were to grow in enrollment, lab fees might need to be implemented.

Conclusion

This paper describes an innovative, project-based computer engineering curriculum. The conference presentation will include updated assessment data and additional details of student projects.

Bibliography

1. Joint Task Force on Computer Engineering Curricula: IEEE Computer Society/Association for Computing Machinery, *"Computer Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering"*, 2004, (retrieved on January 26, 2015), http://www.eng.auburn.edu/ece/CCCE/

2. Mills, J. E., Treagust, D. F., Engineering education—Is problem-based or project-based learning the answer?, *Australasian Journal of Engineering Education*, 2003.

3. Perrenet, J. C., Bouhuijs, P. A. J. & Smits, J. G. M. M., The suitability of problem-based learning for engineering education: theory and practice, *Teaching in higher education*, 5, 3, 345-358, 2000.

4. Aalborg University ECE Program website (retrieved on January 26, 2015), http://www.sict.aau.dk/digitalAssets/36/36159_bsc-elektronik-og-datateknik---studieordning.pdf

5. Milwaukee School of Engineering CE Program website (retrieved on January 26, 2015), http://catalog.msoe.edu/preview_program.php?catoid=8&poid=410&returnto=235

6. Meier, R., Barnicki, S. L., Barnekow, W., Durant, E., Work in Progress – A Balanced, Freshman-first Computer Engineering Curriculum, *Proceedings of the 37th ASEE/IEEE Frontiers in Education Conference*, pp. F3H-17 - F3H-18, 2007.

7. Meier, R., Barnicki, S. L., Barnekow, W., Durant, E., Work in Progress – Year 2 Results from a Balanced, Freshman-first Computer Engineering Curriculum, *Proceedings of the 38th ASEE/IEEE Frontiers in Education Conference*, pp. S1F-17 – S1F-18, 2008

8. Rose-Hulman Institute of Technology, CE Program website (retrieved on January 26, 2015), http://www.rosehulman.edu/media/513364/ee_sampleprogramv14r1_1.pdf

9. Harvey Mudd College, Engineering Program website (retrieved on January 26, 2015), https://www.hmc.edu/engineering/curriculum/