AC 2009-2202: FACILITATING VERTICALLY INTEGRATED DESIGN TEAMS

Gregory Bucks, Purdue University
Greg Bucks is a Ph.D. candidate in Engineering Education at Purdue University with an expected graduation date of May 2010. He received his B.S. from Penn State and M.S. from Purdue University in Electrical and Computer Engineering.

William Oakes, Purdue University
William Oakes is the Director of the EPICS Program, an Associate Professor in the School of Engineering Education with courtesy appointments in curriculum and Instruction and Mechanical Engineering at Purdue University. He is an active member of ASEE having served on the boards of the FPD and CIP as well as co-chairing the 2005 FIE Conference for ERM. He has been recognized by Purdue University, the National Society of Professional Engineers, National Campus Compact and the National Academy of Engineering for his work in engineering education.

Jeffrey Richardson, Purdue University
Jeff Richardson is an Assistant Professor of Electrical and Computer Engineering at Purdue University. He is an EPICS advisor and leading an effort to engage more students from Technology in the EPICS Program.

Facilitating Vertically Integrated Design Teams

Abstract

Engineering educators face daunting challenges in adapting their curricula to prepare students to prosper in the global economy. As technology continues to advance, there is a need to cover the latest technology without compromising the fundamentals that are so important for engineers. Multidisciplinary teams comprised of multiyear engineering students can be utilized to facilitate the broad sets of skills needed for success in engineering. These teams provide an opportunity for the students to learn communication and presentation skills, learn how to work in teams to solve problems, and interact with real customers, to mention a few. While working in multiyear teams, older students, seniors and juniors, mentor younger students to assist in developing leadership skills. The younger students are integrated into real engineering projects earlier in their careers giving them a genuine engineering experience that serves to address retention challenges facing many engineering programs. A challenge, however, is creating projects that both challenge the older students and allow the younger students to participate. At Purdue University, vertically integrated teams are brought together to address the needs identified in the local community. LabVIEW tools are utilized to allow the younger students to get up to speed on the projects and participate in areas such as data acquisition, data analysis, and microcontroller programming. This paper describes how the vertically integrated teams are formed, provides an overview of the processes that are used to integrate the teams, and provides specific examples of projects where these tools are utilized.

Introduction

The importance of significant design experiences to prepare undergraduate engineering students for engineering careers has been well-documented[1-4]. These experiences typically emphasize the application of technical skills as well as professional skills, such as communication in both written and verbal form, working as a team, and customer interaction. The need for such experiences has spawned many innovative approaches to capstone senior design courses. However, capstone senior design courses do not include underclassmen. Earlier design experiences have become more common and have shown to be valuable in motivating students to continue in engineering programs[4]. Most courses of this type are confined to one academic period (e.g. semester or quarter) and are intended to give the students an intense exposure to the design process in a single engineering discipline, since these experiences typically involve students at one academic level from a single major.

An alternative model would involve combining students from varying levels (freshman through seniors) into multidisciplinary teams. Under this model, students would participate for several semesters or years on various projects. As the students progress from first and second-year students to junior and seniors, they take on more leadership roles. In such a model, the more senior students act as mentors for the younger students, helping to motivate them, while teaching them to apply the basic skills they are learning in their introductory and fundamental courses.

This process also serves as a mechanism to retain the students in their various engineering disciplines as it provides an immediate application of the knowledge that they are acquiring.

Mentoring structures have been shown to benefit retention of students, especially with underrepresented populations[5-7]. More often than not, mentoring programs are set up as an add-on to the current curriculum as opposed to a purposeful part of the curriculum. Vertically integrated design teams allow the integration of those functions into the core curriculum, making it more efficient and accessible for students.

The EPICS program at Purdue University uses a service-learning approach to teach design in a way that integrates mentoring activities into the curriculum. The curricular structure is built around the concept of long-term partnerships between the student teams and not-for-profit organizations in the local community. These community service agencies face a future in which they must rely, to a great extent, upon technology for the delivery, coordination, accounting, and improvement of the services they provide. However, they often possess neither the expertise to use, nor the budget to design and acquire a technological solution that is suited to their mission. To this end, they need the help of people with strong technical backgrounds, like these multidisciplinary teams comprised of students. Moreover, the projects developed by the teams will be deployed by the community service agencies.

While there are several aspects to creating and nurturing vertical design teams, this paper focuses on a way to level the technological entry point for younger students while maintaining a technical level appropriate for upper-level students. The EPICS program, described in this case study, has leveraged technology from National Instruments, including the LabVIEW programming environment, to accomplish this task.

Curricular Structure

A unique curricular structure enables each course division or team to maintain a long-term relationship with its community partner and to successfully design and deliver products that have significant technical complexity and community impact. The EPICS program is implemented as a track of courses, where a team corresponds to a division or lab section of the course. Each team is large – 8 to 20 students – and vertically integrated – composed of freshmen, sophomores, juniors and seniors. A student may be a member of a team for up to four years, registering for 1 to 2 credits each semester. When seniors graduate each year, returning students move up a year and new students are added to the team. Many teams have developed formal training processes for new members. The large team size, vertical integration, and credit structure enable each team to continue with a core of returning students each semester and year. In effect, the teams function as a small engineering design firm, with the community partner as its customer. This enables the teams to tackle and complete projects of significant size, complexity, and impact in the community. Some teams have been in operation for twelve years and have delivered a series of projects to their community partner.

From an educational point of view, the long-term continuity enables the students to experience the whole design cycle, from problem definition through support of fielded projects and, when appropriate, the exploration of commercialization of their designs. The long-term continuity also enables each student to experience different roles on the team, from trainee to design engineer to project or team leader.

Complementing the long-term structure of the teams is their multidisciplinary nature. The program started in electrical and computer engineering but has spread rapidly both across engineering and outside engineering to computer science, sociology, and many other disciplines. Each team recruits the students and disciplines it needs each semester for its projects. The multidisciplinary nature of the teams adds an important educational dimension and has proven critical to the quality of the products that the teams develop and deliver.

Each student attends a weekly two-hour meeting of his/her team in the EPICS laboratory. During this laboratory time the team members will take care of administrative matters, do project planning and tracking, and work on their project. Class meeting rooms are located adjacent to design, test and build space for the teams in order to facilitate the project work. Divisions are supervised by faculty, practicing engineers from local companies or university staff such as IT professionals.

All students must also attend a number of one-hour lectures or interactive workshops each semester. The lectures/workshops are designed to supplement the work on the project. Many of the lectures/workshops are by guest experts, and have covered a wide range of topics related to engineering design, communication, professional development or in-depth technical topics. The long-term nature of the program has required some innovation in these experiences, since students may be involved in the program for several semesters. This has been addressed by creating a core series of experiences for students in their first-semester and then rotating topics on a cycle of two to three years for more senior students. We have also create specialized lecture supplements called skill sessions, which provide information and training on a specific technical skill, that students can substitute for lectures they have already seen and are approved by their own instructor.

The design courses are used as substitutes within their home departments. In most engineering majors, the courses can be substituted for a technical elective. Outside of engineering, the credits can be used to meet various requirements such as the core social responsibility requirement in Liberal Arts. In ECE, the courses can count toward their lab credits and with approval, for their capstone design experience.

Supporting First-Year Students

There are many benefits to engaging students in real projects early in their careers but the challenges that first-year students face are unique. The EPICS program enrolls students at all points in their academic careers, but there are specific issues with first-semester students.

Being at college and, for many, away from home for the first time adds a different set of concerns for this group of students. They are adjusting to the level of college work, having to set their own schedules without input from parents, and learning to live on their own. First-year students have a very wide range of skills that can be used on a project, but many have never worked on an unstructured project before and find it intimidating. They also may not have a clear idea of what they want to major in.

To support the first-year students, we have moved to enrolling students in the vertically integrated teams as part of a learning community. The learning community places cohorts of students into three classes together, one of which is the design course. The other two classes are only with first-year students from their learning community. One is either an introductory English or communication course. The other is the introductory engineering course that is required for all first-year engineers. Students are also given the opportunity to live in the same residence hall. These provide support for the students as they adjust to college and to the experience of the design course.

We have found that the experience of students in their first-years on the vertical teams has a bimodal distribution. It can be the greatest experience they have had, but there are also some with negative experiences. The data from the negative experiences motivated the movement toward the learning community for early students to add support.

Another factor affecting early students is having appropriate mentors. Each student new to the course is assigned a mentor to help integrate them to the projects. Our focus had been on technical mentoring but it appears that a more holistic mentoring approach would add value and is under development.

The last factor for success of younger students is real engagement on the project. With first-year students on teams, they, and the more senior students, may not see ways for them to actively participate on the projects. This has been the power of LabVIEW and other technology from National Instruments, which is discussed below. It has been a way to bring younger students into meaningful technical components of the projects quickly and early in their careers.

Senior Design Students

While not a focus of this paper, it is important to address the senior end of the spectrum of the vertical design experience. The vertical design course is used as an option for ECE students to do their senior design experience. Clearly, it would not be appropriate for a senior to use the same technology as a first-year student in their capstone. The academic rigor for senior students must be maintained.

To facilitate this, an assessment and scaffolding system has been put in place. First, seniors write a proposal on their project and the role they will play in its development at the start of the semesters they will use for senior design. The proposal is approved by a designate from ECE. There are specific outcomes for capstone students in ECE and these are tracked with the use of an outcomes matrix. We actively and intentionally engage the students in discussions on how the outcomes will be met and how they are documented. They know that if all of the outcomes are not met, they would not receive credit for their capstone experience. We work with each student to identify specific experiences, such as lectures, workshops or reflections, which are needed to document outcomes that are not coming naturally out of the project work.

Finally, each student meets individually with a faculty review team to discuss their individual progress and role on the team. This has been important with the projects being team-based and the work distributed across the teams. These individual meetings have been very valuable to identify their contribution and guide them to the appropriate level of work.

Design Tools

When working with vertically integrated and multidisciplinary design teams, the need arises for tools that can help facilitate the work that these students with varying levels of expertise need to perform on their projects. These tools must be easily accessible to students with limited technical backgrounds and powerful enough to enable upper level students to implement advanced and complex solutions. One set of tools that we have found to help support teams requiring a programming or embedded element are the hardware and software options produced by National Instruments Corporation, specifically the LabVIEW graphical programming environment.

To illustrate, one problem that many teams face is a lack of knowledge and expertise in the use of microcontrollers. Many of the projects that teams work on need to perform some form of monitoring and control. While some project are simple enough that they can be implemented with discrete components, most require some type of microcontroller in order to handle the complex logic needed. Unfortunately, the course in the ECE department which teaches microcontroller development is generally not taken until the senior year, if at all. This means that there are few students within the program who have the necessary expertise to work on these aspects of the projects. LabVIEW can be targeted to a number of different embedded controllers, allowing students with little knowledge of microcontroller development or assembly languages to continue work on the project.

Introduction to LabVIEW

LabVIEW is a programming environment developed by the National Instruments (NI) Corporation in the mid 1980's as a tool to allow engineers and scientists with little to no programming background to create programs to automate experiments and collect and analyze data. What makes LabVIEW unique from other, more traditional, programming languages is that it is a graphical programming language. In a graphical language, programming does not occur by writing lines of text, but instead is developed by placing graphical blocks representing different functions on the screen and connecting them together to create the program, similar to a flowchart or block diagram. These types of representations are very familiar to both engineering and non-engineering disciplines, making LabVIEW an easy entry point into the world of programming.

There are several advantages LabVIEW has over other programming environments for working on project-based, vertically-integrated teams. The first is that the ease with which novice programmers can get into the environment and make contributions toward the completion of the project, since the programming paradigm draws on concepts for which many first-year students have some experience with (flowcharts and block diagrams). LabVIEW is also a fully functional programming language, providing upper level students with the power needed to create sophisticated solutions.

The other advantage of LabVIEW is the ease with which the software environment can interface, through NI hardware, with the real world. Many of the projects these teams work on require some form of control or monitoring. Teams working with LabVIEW can develop and test their software in the virtual environment and then, by making a few minor changes to the code and connecting a simple data acquisition (DAQ) device, they can start testing their code with real world signals.

In addition, LabVIEW can also be compiled to a number of different targets, from FPGA's, 32-bit microprocessors, and other embedded targets to PC software programs, and even web-based applications. This allows for quick development and testing of algorithms in a familiar environment before taking the same code and pushing it to a target for use in the final project.

Examples of LabVIEW Scaffolding:

There are many scaffolding supports in place for the students and teams that choose to use NI hardware and software. These supports are discussed next.

Teaching Assistants:

Each team is assigned a teaching assistant (TA) to both help with the grading of assignments and to provide technical assistance. At the beginning of each semester, all TAs are introduced to NI LabVIEW and NI hardware. This ensures that each team has access to someone who can provide some direction and assistance with the project. In addition, there is a TA with extensive LabVIEW and NI hardware experience who is available to help any team that needs it.

Skill Sessions:

As part of the semester requirement for participation in EPICS, a student must satisfy a certain number of activity credits. These can be fulfilled in several different ways. First, there is a lecture during the week, common to all teams, that covers topics centered around engineering design and analysis techniques, communication, leadership, and ethics. Attending a lecture counts for one of the required activity credits.

The other way for a student to fulfill the required number of activity credits is to attend what are referred to as skill sessions in EPICS. These are short, one to two hour sessions generally held by the TAs which allows students to gain information or skills on topics that are not traditionally covered in their normal curricula.

Every semester, a series of three skill sessions is offered to introduce students to the basics of programming using LabVIEW and to introduce students to the hardware options available. These sessions provide students with guided, hands-on experience with LabVIEW to help prepare them for working on their projects.

The first two sessions in the series focus on programming in LabVIEW. Students are first introduced to the programming environment and the features available. Following this introduction, the rest of these two sessions work through the main programming concepts (mathematics, arrays, logic statements, looping structures, and subroutines) and how they are manifested in LabVIEW.

After each set of concepts is introduced, the students are guided through an activity, during which they create a program utilizing those concepts. Each activity builds upon the first, until they have developed a fairly sophisticated program capable of gathering, analyzing, displaying, and recording data.

The final session is devoted to the hardware aspect. During this session, students are shown the different types of hardware available from NI as well as shown several demonstrations of what it can do. One of the demonstrations utilizes the program created during the first two sessions to show how making a few minor changes can change the program from a simulation to a full-fledged data analysis tool.

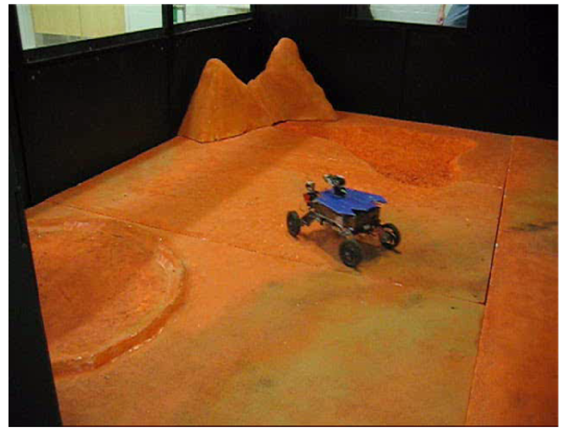
Informal surveys of students given at the end of the skill session series show that the students see the benefits of using LabVIEW and other NI technology, and that they feel comfortable using it. This is also reflected in the feedback given by students through the skill session system. There is an online system students use to sign up for skill sessions, which allows them to rate the skill session on a scale of one to ten and also provide comments. The LabVIEW skill session series is traditionally one of the high rated skill sessions given during a semester.

Examples of LabVIEW Projects:

There have been a number of EPICS teams that have used NI hardware and software for their projects. These teams have been made up of students with a variety of different backgrounds and from first-year through senior design students. This section will discuss two of these projects.

Imagination Station Team – Mars Rover:

One of the most advanced projects an EPICS team has undertaken, the Mars Rover project was designed for a local children's museum, the Imagination Station. The purpose of the project was to develop a hands-on, interactive exhibit to teach young children about NASA's missions to Mars with the rovers Spirit and Opportunity. These goals were accomplished through a large exhibit, complete with mission control station, representative Martian landscape, and miniature rover, modeled after the actual Spirit and Opportunity rovers and controlled from the mission control station. A picture of the exhibit is shown below:

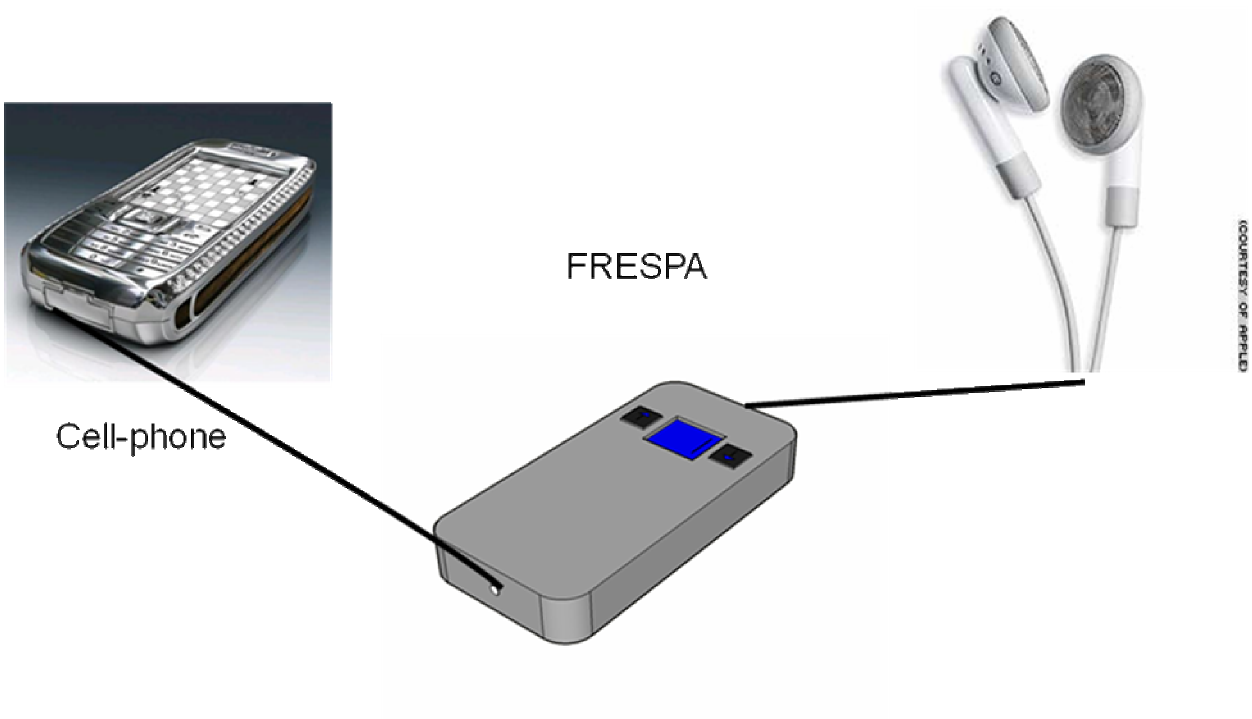
To use the exhibit, a student sits at the mission control station. From here, the student has two options: to drive to rover around the landscape in a free drive mode or select from one of four missions. The missions are tailored after the real missions the rovers Spirit and Opportunity performed. Each mission leads the student through a series of tasks and requires them to answer questions in order to advance to the next step, helping them to learn about the actual missions. The entire software user interface and control system was developed in LabVIEW. This system performs a number of different functions. First of all, it monitors all of the joysticks and buttons used to control the rover and interact with the program. This is done through the use of a USB data acquisition (DAQ) card sold by NI. The system also communicates, via a Bluetooth connection, with the rover, sending serial commands to the microprocessor on the rover. The software system also controls all of the mission logic necessary to advance through the missions. It feeds video feeds from two Bluetooth cameras mounted on the rover itself as well as from a USB webcam mounted above the terrain for use while driving. Two PCI video processing cards produced by NI are used to acquire and process the video. The webcam is used for position tracking, allowing the program to locate the rover on the landscape to determine whether it has arrived at the correct place to proceed with a mission.

Throughout the three years that this project was being developed, the team consisted of members from a variety of different engineering disciplines (ECE, ME, AAE) and from first-year students through senior design students. The team was able to make progress with such a diverse group of students because of the ease with which students could work within the LabVIEW environment.

Speech, Language, and Audiology Clinic Team – FRESPA:

Another example of a project using LabVIEW and NI hardware is the Speech, Language, and Audiology Clinic team's FRESPA project. This team partners with the M. D. Steer Speech-language and Audiology Clinic on Purdue's campus, and provides software and devices to help with both the instruction of students in the audiology program as well as the enhance the capabilities of the clinicians.

The FRESPA project (Frequency Specific Personal Amplifier) was started as an attempt to help individuals with mild to moderate hearing loss communicate better over a cellular telephone. The device would connect to the headphone jack, or other audio output port, of a cellular phone and, by choosing from a set of predetermined profiles, would amplify the signal from the cellular phone at those frequency ranges where the individual's hearing impairment takes effect. The concept is depicted below:



The team first developed a software application in LabVIEW to test ideas on how to process the audio signal. Once they completed and tested their software version, they were able to take the same software and compile it down onto a SPEEDY-33 DSP device. This is a simple board produced by National Instruments with a TI DSP chip. With a physical device, the team was able to test their ideas in real time with real phones. This project was so successful that the team has filed a provisional patent and is in the process of working out an agreement with Motorola to fully develop and sell the device.

This team also had members from a variety of disciplines. Students on the team during the year and a half it was in development ranged from ECE, Acoustical Engineering, Interdisciplinary Engineering, and Audiology as well as from sophomores through graduate students. Because of the graphical nature of LabVIEW, the audiology students were able to work alongside the engineering students in developing the software.

Conclusion

Vertically integrated teams have a great potential to engage students early in their careers into highly technical projects. Barriers do exist including the learning curve for younger students to learn enough technology to contribute. There is a danger that by simply putting students together, the younger students, if not properly engaged, can have a negative experience. When proper scaffolding and support are employed, the experiences can be very powerful. The experience described in this paper has shown that the LabVIEW software and associated technology can play a powerful role in reducing the challenges. The technology can also be used to engage undergraduates in technologies that would otherwise be reserved for graduate students.

Bibliography

1. "Improving Engineering Design: Designing for Competitive Advantage", National Research Council, Washington, D.C., 1991.
2. Dekker, Don L, " Design/Build/Test Projects Are Not All Created Equal", Proceedings, ASEE Annual Conference, Session 2225, Charlotte, NC, June 1999.
3. Dahir, M. (1993). Educating engineers for the real world, in Technology Review, Aug./Sept. 1993, pp. 14-16.
4. Dym, C.L., Agogino, A.M., Eris, O., Frey, D.D., Leifer, L. J., (2005) "Design Thinking, Teaching, and Learning" *Journal of Engineering Education,* Vol. 94, No. 1, pp. 103-120
5. Felder, R. M., Felder , G. N., Mauney, M., Hamrin, C. E., & Dietz, E. J. (1995). A Longitudinal Study of Engineering Student Performance and Retention. III. Gender Differences in Student Performance and Attitudes. *Journal of Engineering Education, 84*(2), 151-163.
6. Newton, E., & Linda, W.-G. (1999). Mentors for Undergraduates in Technical Disciplines: A Collaborative Effort by Faculty, Student Development Professionals, and Alumni to Improve Undergraduate Retention and Success in Technical Majors. *Journal of College Student Retention, 1*(4), 311.
7. Seymour, E., & Hewitt, N. M. (1997). *Talking about leaving: Why undergraduates leave the sciences*. Boulder, CO: Westview Press.