



## Designing a Sustainable Large-scale Project-based Learning (PBL) Experience for Juniors in Electrical and Computer Engineering

**Prof. Stephen Schultz, Brigham Young University**

Stephen M. Schultz has received B.S. and M.S. degrees in electrical engineering from Brigham Young University, Provo, UT, in 1992 and 1994, respectively. He received a Ph.D. in electrical engineering from the Georgia Institute of Technology, Atlanta, GA, in 1999. He worked at Raytheon Missile Systems from 1999-2001. He has taught at Brigham Young University since 2002 and is currently a Full Professor. He has authored or coauthored over 100 publications and holds 10 patents. His research interests are in the area of optical fiber devices with an emphasis on optical fiber based sensors.

**Prof. Brad L. Hutchings, Brigham Young University**

Brad L. Hutchings received the PhD degree in Computer Science from the University of Utah in 1992. He is currently an associate professor in the Department of Electrical and Computer Engineering at Brigham Young University. In 1993, Dr. Hutchings established the Laboratory for Reconfigurable Logic at BYU and currently serves as its head. His research interests are custom computing, embedded systems, FPGA architectures, CAD, and VLSI. He has published numerous papers on FPGA-related topics and is an inventor/co-inventor for 60+ patents.

# Designing a Sustainable Large Scale Project-Based Learning (PBL) Experience for Juniors in Electrical and Computer Engineering

## Abstract

This paper presents a large-scale Project-Based Learning (PBL) curriculum that can handle 200 students per year without requiring an undue commitment of faculty or teaching-assistant time. The following strategies were used to attain the student benefits of the PBL curriculum while accommodating a large number of students and while keeping the faculty and teaching-assistant commitments to reasonable levels. (1) A top-level hardware/software specification of the system (laser-tag) is provided to the students. (2) Students must test their software and hardware using both their own methods and with provided test software and hardware fixtures. (3) How-to and demonstration videos are provided via a dedicated YouTube channel. (4) Students implement the system by completing a series of scheduled milestones. (5) The same PBL project is completed every year. This large scale PBL curriculum is conducted during the junior year and has been in place for several years in the Electrical and Computer Engineering Department.

## 1 Introduction

There are two main objectives for developing a project-based learning (PBL) curriculum. (1) It provides better student motivation. There has been a significant amount of work showing a link between student motivation and engagement in schoolwork<sup>1,2,3,4</sup>. A PBL curriculum provides this motivation because the students can see what the end goal of the project is and are actively engaged in the development<sup>5</sup>. (2) It provides deeper learning. PBL provides for deep learning because it requires the student to integrate knowledge gained in several classes into the creation of a product<sup>6,7</sup>.

When students work on a laboratory exercise that is simply following a set of detailed steps, the learning is similar to what can be attained by listening in a lecture<sup>8</sup>. In order to gain the benefits of PBL, there are several requirements<sup>6</sup>.

- It needs to be a complex and challenging problem.
- It needs to be an authentic or real-world problem.
- It requires working relatively autonomously over an extended period of time.

- It needs to culminate in a realistic product (often called an artifact).

In addition to these requirements for a PBL environment, the project also needs to have a high success rate.

This paper presents the main concepts that enable a PBL curriculum to be established that can handle 200 students a year without requiring an undue commitment of faculty or teaching-assistant time. This work is based on the implementation of a complex laser-tag system. Laser tag is used because it meets the basic requirements listed above and because students find it fun and engaging.

- Implementing the laser tag system requires students to design and build analog amplifiers, implement digital signal processing (DSP) theory, implement embedded programming with C, and coordinate the operation of all of these pieces to achieve a working laser-tag system.
- The basic operation of a laser tag system is understood by the students. The implementation of a fully working laser tag system is a very authentic problem.
- The implementing of the laser tag system spans the entire junior year. During the fall semester, students implement laser-tag specific analog circuitry in a 4 credit hour analog electronics course; they study Digital Signal Processing (DSP) theory and simulate DSP filters in Matlab in a 4 credit hour DSP course and, students study and practice principles of embedded programming with C in a 4 credit hour embedded programming course. During winter semester students combine the knowledge and skills acquired during these fall-semester courses to create an advanced laser-tag game as part of a 3 credit hour junior project course.
- The end result is a working artifact that consists of a laser tag gun and the control pack that provides a laser-tag experience comparable to that found at commercial laser-tag venues. In experiments, even groups of students who had nothing to do with the construction of the system had a lot of fun playing laser tag.

Some of the specifics of the laser tag system will be covered in this paper. However, the paper focuses primarily upon the lessons learned in creating a PBL curriculum that can handle a large number of students while keeping the faculty and teaching-assistant commitments to reasonable levels and while achieving a high rate of student success that results in increased student confidence. Implementing a complex, authentic system typically requires a significant amount of faculty-student interaction in order to ensure that most students successfully implement and debug the system without becoming inordinately frustrated or completely discouraged. This PBL achieves high student success with reasonable faculty loading by (1) providing a detailed, block-level specification with dated milestones for the laser-tag system, (2) thorough testing for each milestone via provided test software and hardware fixtures, and (3) instructional and demonstration videos accessed on a dedicated YouTube channel.

## 2 Laser-Tag System Specifications

The laser tag system constructed by the students runs on battery power, is portable, can detect “hits” from opponents that are up to 120 feet away (in daylight) and supports game-play between up to 10 players. The laser tag system is built using a high-performance embedded system that consists of an ARM processor, an Analog to Digital Converter (ADC), touch-screen TFT and other support components. Students build analog circuits that interface between the embedded system and a Nerf gun that has been retrofitted with an LED and a photodetector. Students implement real-time algorithms on the embedded system that transmits a modulated beam of light from an LED when the gun-trigger is pressed, and that can detect “hits” from the modulated beam from an opponent’s gun. The entire system consists of about 4,200 lines of C code; about 1/2 of that is written by the student, the other 1/2 is instructor-provided test and verification code.

The entire system, including battery, student-constructed analog boards, is housed in a transparent plastic box that is worn like a backpack during actual game-play. These boxes are provided for the students to use. Sensors are distributed across the box and shoulder straps. A small speaker is provided for sound; various sounds play during the game to make it more engaging and to cue the player when the gun fires, when they are hit, etc. The box is shown in Figure 1. The empty area is where the student’s analog electronics will be mounted; the speaker is blue. Various USB connections are used for battery connections and to program the board. The white connectors located at the top of the box are used to connect the sensors that are contained in the shoulder straps. A photo of the finished system, worn by a student, is shown in Figure 2.

The detailed specification of the laser-tag system is presented to students as a series of milestones. Figure 3 shows a small screen shot of a class web-page that contains the links to the individual web pages that describe these milestones. Milestones break the system into digestible concepts and major subsystems. Milestones, in turn, are typically broken down into a series of tasks to help students manage workload and project timeline. For example, Figure 4 is a screen-shot of the web-page containing Milestone 3 with its associated set of tasks. The concept of breaking down a system into smaller pieces is a very common approach used in industry to build a complex system. The approach of simplifying a complex learning environment into a simpler learning environment is often called scaffolding<sup>9,10,11</sup>.

Milestones and their constituent tasks are designed to have clear, unambiguous requirements and pass-off criteria. In cases where system behavior may be complex, videos and color plots are often used to show exactly how a properly working system should appear so that students know if they have completely met the requirements for pass-off. Comprehensive testing and strict pass-off requirements are essential; approximate or ambiguous pass-off requirements that allow interpretations by the TAs or students will ultimately lead to non-functional systems and considerable frustration by students. Student success is defined, in part, as a fully-working system that meets the original specifications and the path to success is to guarantee that each module meets an unambiguous set of requirements so that it will work correctly when connected to other similarly-tested modules.

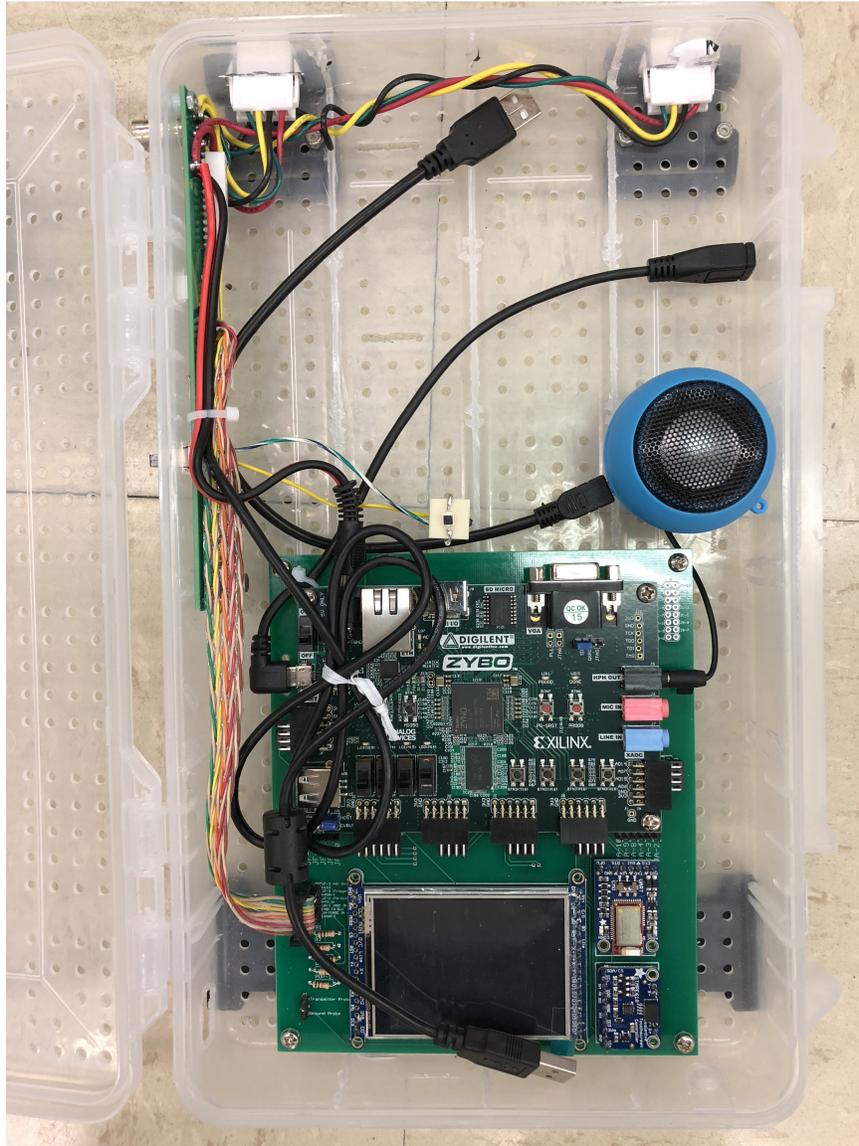


Figure 1: Laser-Tag System



Figure 2: Laser-Tag System Worn By Student

# Junior Core Laser Tag Project

This is the wiki page for the Laser-Tag Project that is part of the Junior Core experience.

## Quick Links to Milestones and Tasks

- [Milestone 1: \(Queues and Analog Board Verification\)](#)
- [Milestone 2: \(Implementing Filters in Matlab\)](#)
- [Milestone 3: \(Implementing Filtering Code, State Machines, in C\)](#) [🌐 Task 1](#), [🌐 Task 2](#), [🌐 Task 3](#).
- [Milestone 4: \(Adding the Guns and Analog Boards with Testing\)](#)
- [Milestone 5: \(Implement a Complete Laser Tag Game with Sound\)](#)

Figure 3: Laser-Tag Milestones

### Goal

Demonstrate the system working with a feedback connection (built into the ECEn Development Board) connecting the output of the transmitter to the input of the ADC. You will demonstrate the ability of your system to:

- detect hits on each of the 10 frequencies,
- no false detects,
- no bouncing of the trigger switch, i.e., register one hit for each push/release of the trigger,
- display the computed power of each of the channels on the TFT with the provided histogram package.

### Organization

This milestone will be divided into three tasks that will be due and passed off as recorded in Learning Suite.

1. **Task 1.** Implementation and verification of the FIR and IIR filters (filter.c). Demonstrate all sets of of IIR coefficients.
2. **Task 2.** Implementation of the transmitter state-machine (transmitter.c), the trigger state-machine (trigger.c), the hitLedTimer state-machine (hitLedTimer.c), the lockoutTimer state-machine (lockoutTimer.c), and the code to compute the power in the output from each IIR band-pass filter.
3. **Task 3.** Implementation of the detector (detector.c) and integration with all other parts of the project in order to perform the final pass-off for the overall milestone using the feedback provided by the development board.

Figure 4: Tasks for Milestone Three

## 3 Testing

In addition to scaffolding, the students also need formative feedback throughout the project in order for the students to know if they are on schedule to create a fully functioning laser tag system. The formative feedback needs to be automated; otherwise, the load on faculty and teaching assistants will be too high for the PBL curriculum to be sustainable. The formative feedback is built into the milestones by providing students with test software and hardware fixtures with each task in each milestone. Students must test their software and hardware using both their own methods and with provided test software and hardware fixtures. All tests are strictly go/no-go tests, meaning that students may not proceed until the software/hardware passes the required tests.

Comprehensive testing and verification for each milestone task are essential to the success of this project. The combination of analog circuitry that produces millivolt signals and complex DSP algorithms running continuously at 100 kHz results in a system that exhibits very counter-intuitive behavior when bugs arise. Each module of the overall system must be carefully tested to ensure that the overall system will work properly when all modules are brought together (analog hardware and software). It is relatively straightforward for students to test and debug the laser-tag modules in isolation; it would be very difficult for even a seasoned expert to debug the

complete laser-tag system without these module-level tests.

Though some may question whether the verification code should be instructor-provided or written by students, the instructors decided to provide comprehensive test software for the following reasons.

- The majority of the students are novice programmers. Only those students with significant out-of-class experience are capable of writing comprehensive module tests. The same is true for the analog electronics.
- Comprehensive test software often dwarfs the software-under-test in both size and complexity. For example, for any given module in the laser-tag system, the provided module-testing code typically contains 5-10 times as many lines of C code and is much more complex. Graphical code is sometimes provided to plot the output from digital filters, etc., on the TFT display on the embedded system hardware (see Figure 5). Expecting the typical inexperienced EE junior to write this kind of code is unreasonable.
- No matter how hard you try, English descriptions of software always contain ambiguity. Though the behavior of all of the modules is thoroughly described using English text in the milestone and task descriptions, the provided test-code augments the English description by demonstrating exactly how the code should behave in various corner cases, etc.

It is worth noting that students do write some of their own test code, specifically for simpler parts of the system. In addition, when their code fails the instructor-provided tests, students usually write their own simpler test code that focuses on specific areas of the suspected software.

For example, to help students verify filter correctness, additional provided test code plots the actual frequency response of each of the student's filter code on the TFT display (mounted on the main system board) so that students can compare the response of each of their filters against correctly-functioning filters. Figure 5 depicts the image of the plotted frequency response of a correct FIR filter as it appears on the TFT display. Note that graphical comparison works well here; the precise shape of the filter isn't that important as the filter response will vary somewhat from student to student. As long as the plotted filter meets the specifications, the filter is considered to be correct.

#### **4 YouTube Channel**

Figure 6 shows a screen-shot for the class's dedicated YouTube channel. A YouTube channel is used so that the students know where to go to find helpful videos. The YouTube channel also allows new videos to be broadcast to all of the students that subscribed to the channel.

Two distinct categories of videos are provided: 1) "how-to" videos that primarily instruct students how to use commercial software and instruments such as oscilloscopes, and 2) "demonstration" videos that graphically depict the correct graphical output of their system for a specific milestone.

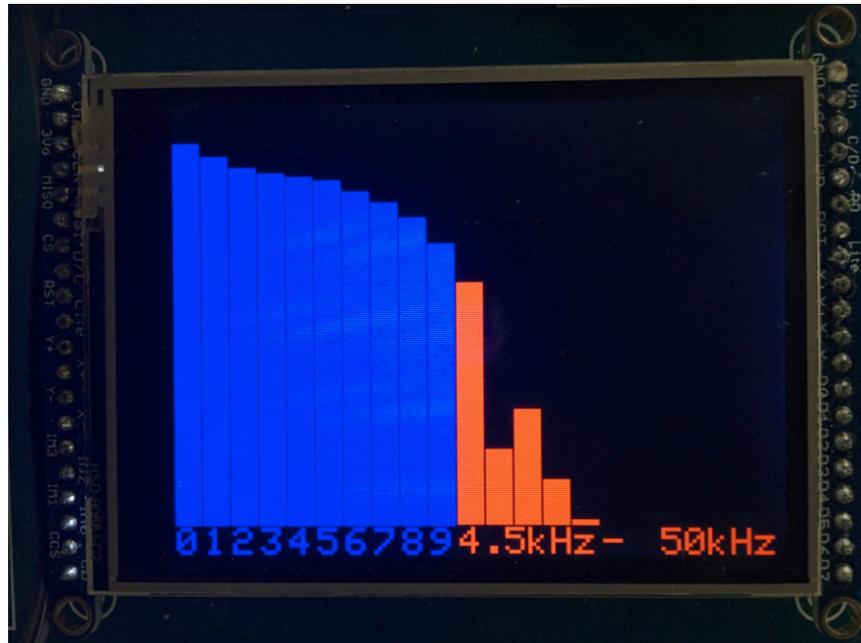


Figure 5: FIR Filter Response Plotted on the TFT Display

## 4.1 How To Videos

The “how-to” videos demonstrate, in step-by-step, form how to perform complex, lab-related tasks consistently and effectively. For example, students develop software using Xilinx’s Vivado Software Development Kit (SDK). The Xilinx SDK is the same software used by industry designers and as such, it is large, and is completely baffling to the typical undergraduate student. Each time a laboratory exercise requires students to master a new aspect of the Xilinx SDK, a corresponding video is provided that carefully leads students through the task in step-by-step fashion, as it directly relates to the lab milestone. Complex procedures to verify the functionality of analog electronics, are also carefully explained using videos.

Currently there are 15 instructional videos. Examples of topics covered by these videos include:

- Initial setup and import of provided system files,
- how to configure compiler settings,
- how to copy boot files to an SD card, setting heap and stack size,
- oscilloscope procedures to verify analog circuit functionality, etc.

The videos are intentionally short and are generally more effective than personal instruction by TAs or faculty. Students work through the video at their own speed, rewinding and then reviewing the videos as necessary. Because the videos are accessible via YouTube, they are accessible anywhere the students have an internet connection.

Breaking down a long technical process into several short (1-2 minute) videos seems to be the



Search



BYU ECEN 390

43 subscribers

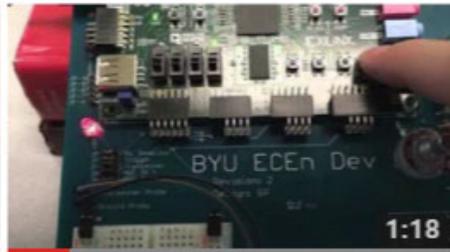
E

PLAY ALL



330 Development  
t (blue board)

1 year ago



BYU ECEn 390: What Shooter  
Mode Should Look Like.

196 views • 1 year ago



BYU ECEn 390: This  
Continuous Mode Lc

233 views • 1 year ago

Figure 6: YouTube channel is used for a common location for videos.

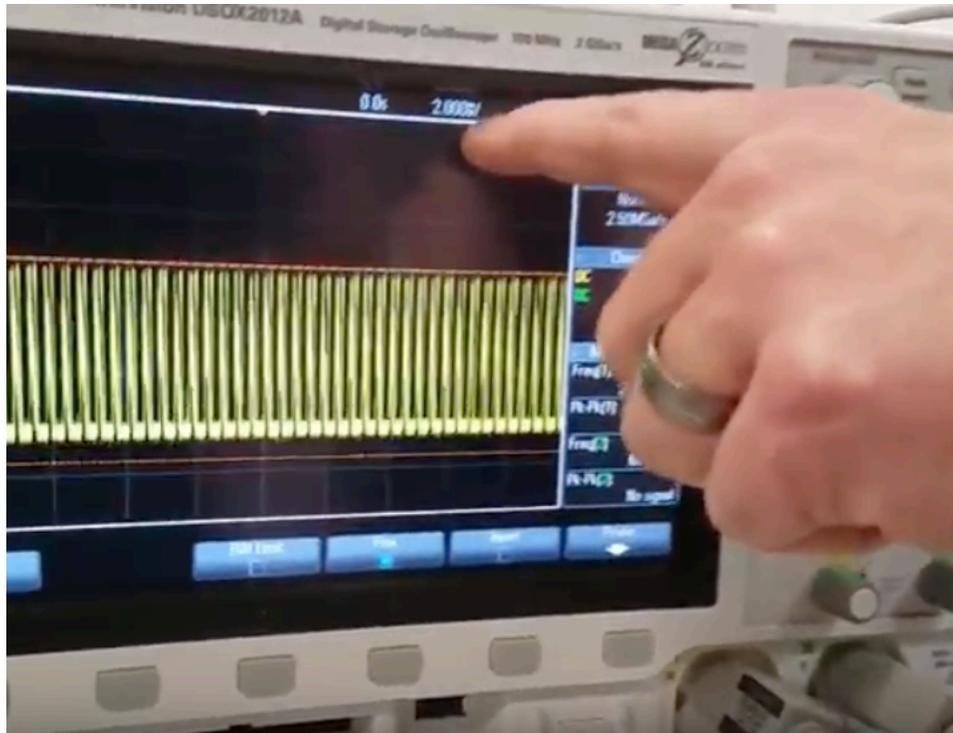


Figure 7: Example video explaining how to use the oscilloscope to measure the frequency response of the system.

most effective to present this material. Students can quickly select the video by topic and then can quickly jump to the desired section of the video. If the videos are too long, students will often forego viewing the video in favor of asking their classmate or TA/faculty, or they may get lost while viewing the video. The goal is to make the videos the “go-to” instructions so that labs are completed consistently, year to year, and to help the students learn to be independent learners and problem-solvers.

## 5 Conclusions

Maintaining a PBL curriculum of this size and complexity remains an ongoing yet reasonable task, even in its fourth year. Regular assessment is performed “in place” as faculty play an active role in the labs and take note of overall student competency, software and hardware issues, clarity of task and milestone descriptions, etc. Faculty and TAs keep notes regarding lab issues as they come up; issues may relate to effort level to achieve some task, areas where student frustration is excessive, typos or mis-organization of web material, the need for additional videos, etc. This feedback is used to improve the curriculum in succeeding years. Instructors have directly witnessed how the PBL experience has improved year-to-year as it has been refined.

This PBL has impacted the *entire* curriculum of the junior year in unanticipated, yet positive ways. When students demonstrate specific deficiencies while working to implement various tasks in the PBL course, the specific deficiency is “fed-back” to the instructor of the course so that the

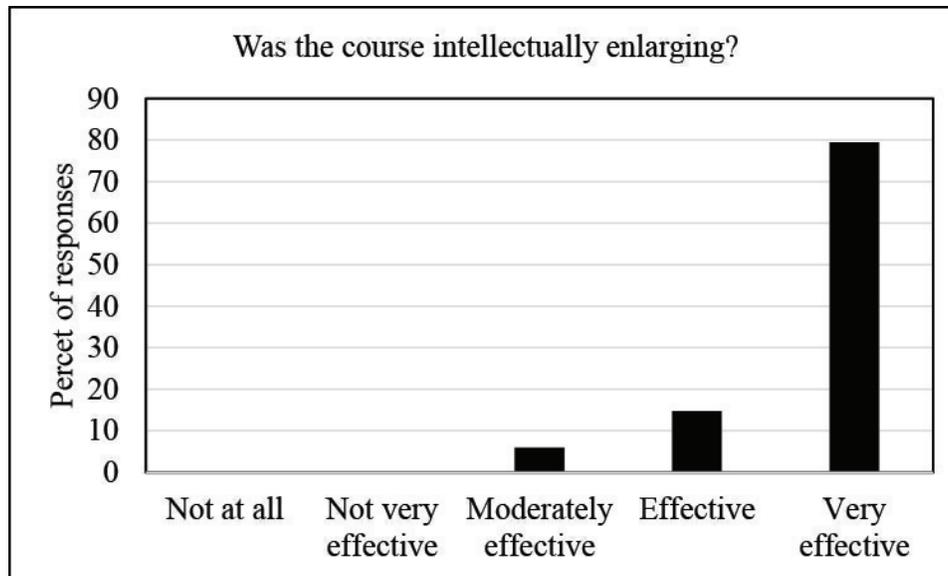


Figure 8: Anonymous survey given to students after the class in 2016 and 2017.

previous pre-requisite course can be improved to overcome this deficiency. This feedback process has led to substantial modification of all of the pre-requisite courses and has had a much bigger and more positive impact than the processes that were previously developed to meet ABET requirements. For example, it was discovered that the students didn't understand digital filtering of sampled data. So more instruction was provided in the fall semester signal course and an additional laboratory exercise was added. Similarly, the students struggled with the real-time aspects of the laser tag implementation so a simple real-time sampling and computation exercise was added to the embedded systems course.

The instructors find it interesting that a course developed specifically to provide students additional experience and confidence turned out to be an excellent tool for continuous improvement of the department curriculum. Moreover, the PBL has provided opportunities for computer-engineering and electrical-engineering faculty to collaborate at the undergraduate level. Both of these outcomes were unanticipated.

## 6 Summary

The PBL curriculum described in this paper is in its fourth year and has achieved the desired goals of high success rate, and reasonable faculty and TA loading. The course has 120 students and is supported by 5 TAs. Surveys of students indicate that the laser-tag project is largely achieving its goals. Success rates are high, generally in excess of 90%. Students state that the class has increased their confidence and also clarified their knowledge of concepts learned in prior courses. Most students feel that the class is also quite fun and they enjoy actually playing with the laser-tag systems they have built. In addition to the interest in the class, an anonymous survey was conducted to determine if the students learned in the class. Figure 8 shows that the majority of the students felt that the course was intellectually enlarging.

There are four keys to the success of the large scale PBL curriculum. (1) The PBL artifact (the completely functioning laser tag system) needs to be structured into high level hardware and software milestones and each milestone needs to have a specific go/no-go pass-off. (2) Hardware fixtures are created that enable the software and hardware to be tested independently. Software test code is used as part of the milestone pass-off. The Test code enables the students to receive formative evaluation of the status of their project throughout the project. (3) YouTube videos are used to show the students the required state of the project for milestone pass-offs. The videos also provide demonstrations to teach the students how to use the various tools like the software development kit (SDK), oscilloscopes, etc. (4) Because the same PBL is taught over multiple years, trained TAs are readily available as just about any senior can serve as a TA.

## 7 Resources

All class content is freely available online. Links to the class materials, youTube channels, student-created videos, are provided below.

- Junior-Core Class Web Pages:  
<http://ece390web.groups.et.byu.net/dokuwiki/doku.php?id=start>
- Embedded Programming Course Web Pages:  
<http://ecen330wiki.groups.et.byu.net/wiki/doku.php>
- Junior-Core-Project youTube Channel:  
<https://www.youtube.com/channel/UCoDVeJy1UjZCQW9TZ3dNS8w>
- Embedded Programming Class youTube Channel:  
<https://www.youtube.com/channel/UC2DAqectVGE6DIT-b8N2dJg/playlists>
- Laser-Grenade Student Video:  
<https://youtu.be/pvLbELZw1Hc>
- Web-Page Containing All Student-Created Videos from 2016:  
[http://ece390web.groups.et.byu.net/dokuwiki/doku.php?id=groups\\_2016](http://ece390web.groups.et.byu.net/dokuwiki/doku.php?id=groups_2016)

## References

- [1] Carole Ames and Jennifer Archer. Achievement goals in the classroom: Students' learning strategies and motivation processes. *Journal of educational psychology*, 80(3):260, 1988.
- [2] Elaine S Elliott and Carol S Dweck. Goals: An approach to motivation and achievement. *Journal of personality and social psychology*, 54(1):5, 1988.
- [3] Philip H Winne and Ronald W Marx. A cognitive-processing analysis of motivation within classroom tasks. *Research on motivation in education*, 3:223–257, 1989.

- [4] Patricia Pokay and Phyllis C Blumenfeld. Predicting achievement early and late in the semester: The role of motivation and use of learning strategies. *Journal of educational psychology*, 82(1):41, 1990.
- [5] Phyllis C Blumenfeld, Elliot Soloway, Ronald W Marx, Joseph S Krajcik, Mark Guzdial, and Annemarie Palincsar. Motivating project-based learning: Sustaining the doing, supporting the learning. *Educational psychologist*, 26(3-4):369–398, 1991.
- [6] John W Thomas. A review of research on project-based learning. 2000.
- [7] Brigid JS Barron, Daniel L Schwartz, Nancy J Vye, Allison Moore, Anthony Petrosino, Linda Zech, and John D Bransford. Doing with understanding: Lessons from research on problem-and project-based learning. *Journal of the Learning Sciences*, 7(3-4):271–311, 1998.
- [8] National Research Council et al. *How people learn: Brain, mind, experience, and school: Expanded edition*. National Academies Press, 2000.
- [9] Brian J Reiser. Scaffolding complex learning: The mechanisms of structuring and problematizing student work. *The Journal of the Learning sciences*, 13(3):273–304, 2004.
- [10] Sadhana Puntambekar and Roland Hubscher. Tools for scaffolding students in a complex learning environment: What have we gained and what have we missed? *Educational psychologist*, 40(1):1–12, 2005.
- [11] Elizabeth A Davis and Naomi Miyake. Explorations of scaffolding in complex classroom systems. *The journal of the learning sciences*, 13(3):265–272, 2004.