AC 2012-3794: A FIRST-YEAR "INTRODUCTION TO ENGINEERING" COURSE AT A COMMUNITY COLLEGE USING HANDS-ON MATLAB EXPERIMENT CONTROL

Prof. Stephen W. McKnight, Northeastern University

Stephen W. McKnight received a Ph.D. in solid state physics from the University of Maryland, College Park, in 1977 and had postdoctoral positions at Emory University and as National Research Council Fellow at the Naval Research Laboratory before joining the faculty in the Physics Department at Northeastern University in 1980. In 1988, he became an Associate Professor in the Department of Electrical and Computer Engineering, and in 2001 was promoted to Professor of electrical and computer engineering. He has published more than 50 refereed journal articles on microwave, far-infrared, and optical materials and devices and on innovative education programs. Since 2000, he has been the Education Thrust Leader for the Center for Subsurface Sensing and Imaging Systems, an NSF Engineering Research Center headquartered at Northeastern, and is the Education Thrust Leader for the DHS ALERT (Awareness and Localization of Explosive Related Threats) Center at Northeastern University. He has served six terms on the Northeastern University Faculty Senate Agenda Committee, including three terms as the elected Secretary of the Faculty Senate, and wrote the ECE Department's self-study report and coordinated the site visit preparations for the Electrical and Computer Engineering ABET accreditation in 2001 and 2007. In 2004-2005, McKnight served as Interim Chair of the Electrical and Computer Engineering Department, and in 2008-2009 served as Acting Vice Provost for Research.

Prof. Michael E. Pelletier, Northern Essex Community College
Dr. Paula G. Leventman, Northeastern University

Paula Goldman Leventman has been Diversity Coordinator and Internal Evaluator for the NSF-funded Center for Subsurface Sensing and Imaging Systems (CenSSIS) from 2000 to 2011. She was Assistant Dean of engineering for women's projects at Northeastern University from 1982-2004. Leventman was Principal Investigator of the NSF-funded Multiple Pathways toward Gender Equity in the U.S. IT Workforce, 2001-2005, and at the same time, she was Principal Investigator of the NSF-funded 4 Schools for Women in Engineering, 2001-2005, which involved faculty and students from NU, BU, RPI, and Tufts with middle school teachers and students. Leventman also has a national reputation in social research and program evaluation. Over the last two decades, she evaluated numerous NSF supported teacher enhancement and engineering center programs. She held instructional academic positions at Boston College and Wellesley College. She is the author of Professionals Out of Work, Macmallan, 1981. She is currently a director and the program chair of the International Network of Women Scientists and Engineers (INWES), Education and Research Institute (ERI).

A First-Year "Introduction to Engineering" Course at a Community College Using Hands-On MATLAB Experiment Control

Introduction

Project-based learning has been recognized as a valuable introduction to engineering education and engineering as a career[1-4]. The engagement of students, often organized into pairs or larger teams, toward the design and achievement of a specific complex project goal provides a better model for what engineering is and what engineers do – and also is often more accessible for women, minority, and other students with unusual learning styles – than the formal science and mathematics courses that are most often the bulk of what first-year engineering students take. This insight has been a contributing factor to the proliferation of first-year "Intro to Engineering" courses taught in the engineering school or department by engineering faculty[5-10]. Simply stated, engineering departments or colleges have not been happy with the output of a typical "weed-out" first-year program, both because good potential engineers are discouraged by the lecture- and theory-intensive math and science courses, and because those students who make it through are not necessarily well-trained in the kind of implementation and innovation skills that characterize excellent engineers.

To address these defects of first-year programs, a feature of many of these new "Intro to Engineering" courses is a team project focused on achieving an engineering goal within design specifications. The engagement of students in designing a system to prevent a dropped egg from breaking, design a canoe made of concrete, or develop a solar car offers an opportunity to learn – by doing – the fundamental engineering art of design within constraints. Design projects that involve systems to help elderly, sick, or handicapped individuals also have the advantage that they put a human face on engineering and allow students to envision the job of an engineer as a way to help people.

Programming is another engineering skill that sometimes is a part of a first-year "Intro to Engineering" course, often with mixed results. Programming includes cryptic commands and procedures which must be reproduced precisely even to successfully navigate the programming interface. The combination of a demand for attention to detail and the often pedestrian outputs create a barrier to the novice.[11,12] Projects or design-based learning in an introductory computer/programming class are often so abstract or artificial that they fail to engage the student. Yet, microprocessor-based control is at the heart of all modern technology. Automobiles have scores of embedded processors that do everything from adjusting the ignition timing to controlling the seat position. Computer/sensor/actuator interfaces are the core of almost all 21st century engineering. It appears to be a lost opportunity if we cannot bring some of the excitement and challenge of computer-controlled systems into the curriculum of our first-year engineering courses. In this paper we focus on project-based, hands-on learning in the context of an engineering problem-solving and computation course. The projects are structured to provide a sequence of activities using MATLAB control of instrumentation that lead to a significant computer/hardware system result. Extensions of the projects for the more advanced students abound.

Barriers to a more widespread introduction of such computer control applications into first-year engineering courses include 1) the cost and complexity of equipment, 2) lack of instructor expertise in computer control applications, and 3) lack of appropriate models and teaching materials for such a curriculum, and, more broadly, a lack of time and ability to develop such materials. We report here on a collaboration to introduce such computer-control project-based learning into a new EST104 "Engineering Essentials and Design," first-year, engineering course at Northern Essex Community College (NECC) in Haverhill, MA. The collaboration includes the NECC classroom instructor, the NECC engineering program leader and fellow-instructor, and a faculty member from Northeastern University who directs the Education Thrust of the Department of Homeland Security ALERT Center of Excellence. MATLAB was selected as a computer language with the Data Acquisition and Instrument Control Toolboxes used to control a stepper motor, ultrasound transmitters and receivers, and a Thorlabs, "no-moving-parts" spectrometer. Four relatively low-cost computer control modules were implemented in the first semester of EST104, and several others are on-line to be introduced into future offerings of EST104 beginning in Spring 2012. An extensive evaluation of the experience revealed a dramatic improvement in the students' self-evaluation of their skills and knowledge in problem-solving and programming skills, a majority of the students found the experience "challenging," "enjoyable," experienced "a sense of accomplishment," and felt they "learned a lot." Over 80% of the students supported the use of such modules in future classes. (See Table I and II below.)

EST104 "Engineering Essentials and Design" at NECC

Northern Essex Community College serves students from cities and towns north of Boston. The pre-engineering program at NECC is rapidly expanding, with most students planning on transferring to the engineering program at the University of Massachusetts at Lowell (UMass-Lowell). Other state or private universities also enroll NECC graduates. UMass-Lowell recently introduced a required first-year "Intro to Engineering" course into their engineering curriculum and, as a part of their articulation agreement with NECC, agreed that students who complete the EST104 course at NECC will receive transfer credit for the equivalent course at UMass-Lowell. UMass-Lowell does their first-year programming instruction in MATLAB, so it was appropriate that the NECC course also use MATLAB. Two sections of EST104 were offered in the Fall 2011 semester, each with 20 students enrolled.

High-Tech Tools & Toys Workshops

ALERT (Awareness and Localization of Explosives-Related Threats) is a multi-university Department of Homeland Security Center of Excellence (COE). The ALERT partnership is made up of national and international academic, industrial and government partners. For the last two summers, the ALERT COE has sponsored summer workshops for community college faculty members on the topic of computer instruction using real-time modules that Northeastern University (NU) has introduced in their first-year engineering curriculum using a "High-Tech Tools & Toys Laboratory" (HTT&TL). At NU, the HTT&TL is used to teach MATLAB and C++ to first-year engineering students through a set of structured exercises leading the students to image a shape concealed in opaque gelatin using 1MHz ultrasound (MATLAB) and to use a stepper motor mechanism to color-sort dyed Ping-Pong balls imaged by a video-cam (C++). The community college faculty members were participants in an NSF-supported STEP grant, and were supported through ALERT stipends to attend the workshops.

Four faculty members from NECC, including one of the authors (MP) and a fellow engineering program faculty member who serves as the coordinator of the NECC engineering program, attended the workshop. After participating in the workshop and experiencing the NU HTT&TL experiments, the NECC faculty members were enthusiastic about implementing similar computer control projects in the new EST104 course. One of the authors (MP) attended the summer workshops in both 2010 and 2011. The 2010 Summer Workshop included the projects that were used in the NU classes using both MATLAB and C++. The 2011 Summer Workshop was organized after it was clear that the community colleges would be using only MATLAB as a programming environment. The NECC faculty extended the NU experiments in two new projects that they wanted to integrate into EST104.

Implementation of HTT&TL Modules in EST104

EST104 was scheduled in a single 4-hour class once per week in a lab/classroom with 10 experimental stations located on the outside of the classroom. Each station had a computer and a rack-mounted oscilloscope and signal generator. The first several classes were used for traditional instruction in the Microsoft Office applications PowerPoint and Excel, after which the HTT&TL projects were introduced.

The classroom instructor was assisted in the presentation of the HTT&TL modules in EST104 by the NU faculty member author (SM) who was assisting the dissemination of the HTT&TL modules to the STEP partner community colleges as part of his sabbatical leave. A typical class included a 30-minute presentation on the instrumentation and MATLAB programming features that would be used that class. The rest of the four-hour class was taken up with students working in pairs at lab stations to accomplish control and analysis tasks and answer conceptual and empirical observation questions that were included in a handout prepared for each module. The classroom instructor and the visiting university faculty member would move around between the student pairs working at the 6-8 experimental stations. Usually, at some point in the class, additional material would be presented to the class as a whole, addressing common issues of confusion in the modules. An extended break was usually permitted at the middle of the session, but often it was observed that some teams continued to work on the projects during the breaks.

1) Project 1: Speed of Sound in Air

The first of the modules was based on a project used at the NU HTT&TL to measure the speed of sound of 40 kHz ultrasound in air, model the decrease in sound amplitude with distance, and use what they had found to make an ultrasonic range-finder, as might be found in an auto-focus camera For this module, the students recorded their data, plotted out scatter charts, and found "Trendline" fits with Excel. The equipment for this module included a transmitter/receiver pair of 40 kHz ultrasound transducers, available from Jameco for about $7/pair, and the Tectonics oscilloscopes with time and amplitude cursor controls. Since the NECC signal generators did not have a burst mode capability, a microprocessor-based 40 kHz pulse generator interfaced to the ultrasound transducer was designed and produced by Machine Science, Inc., a non-profit

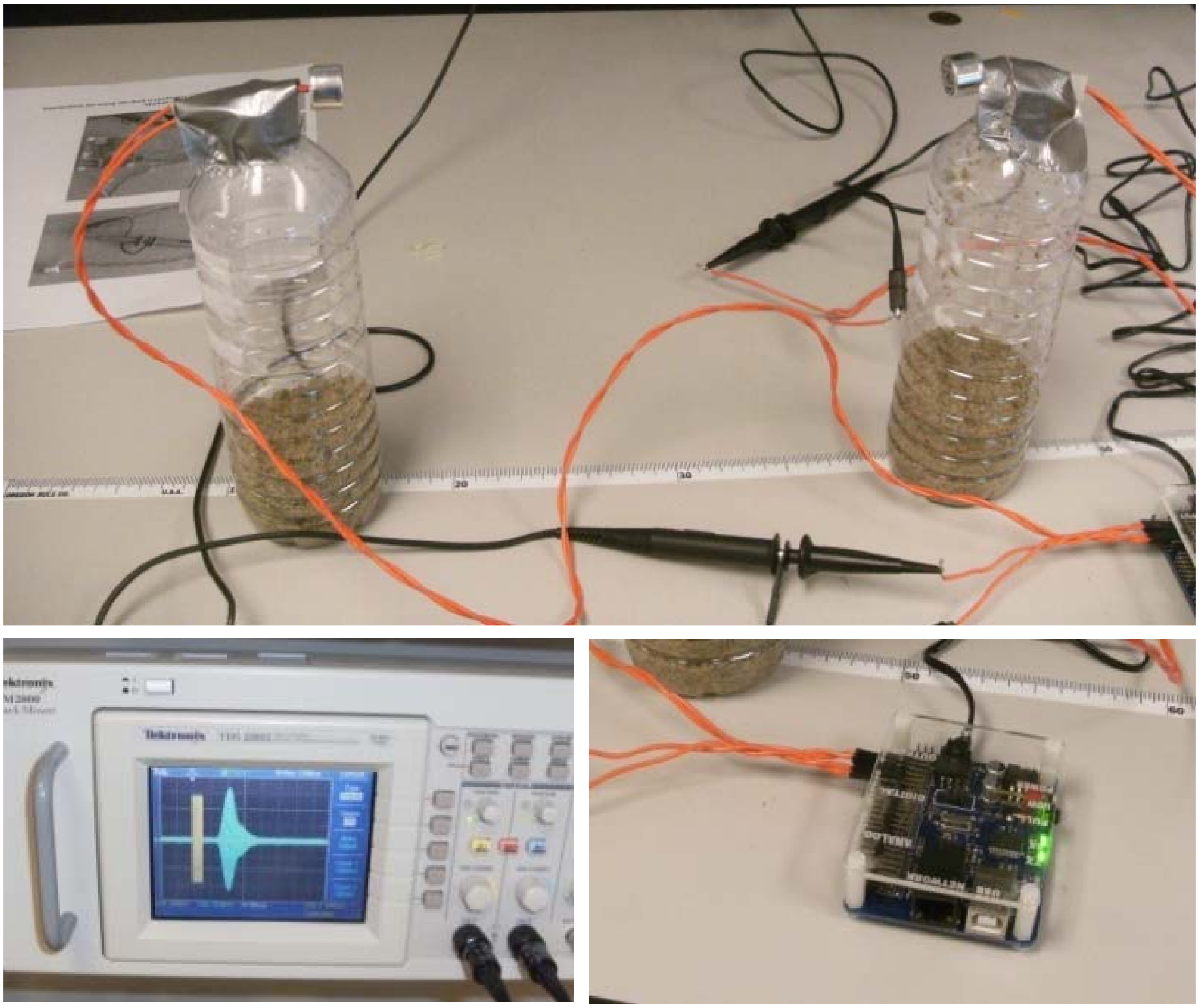


Figure 1: Speed of 40 kHz ultrasound in air experiment. a) (top) Ultrasound transducers attached to sand-weighted plastic beverage containers as bases. Also visible is the tape-on distance scale. b) (bottom right) Machine Science, Inc. microprocessor 40kHz burst-mode pulser with power, transmitter, and oscilloscope probe connections. c) (bottom left) Oscilloscope display of transmit pulse (left – gold trace) and receive signal (right – blue trace).

company based in Cambridge, MA that specializes in educational microprocessor kits. The cost of the Machine Science burst-pulse generator, including the ultrasound transmitter/receivers, connecting wiring, and oscilloscope probe points was $100 per station.

Students were asked to bring in a matched pair of plastic beverage bottles. In class, these were half-filled with sand and the ultrasound transmitter and receiver were duct-taped to the top to provide stable holders for the transducers that provided a clean signal, free from desk-surface bounce effects. These ultrasound transducer holders are illustrated in Figure 1a, and the Machine Science pulse generation module is shown in Figure 1b. The oscilloscope trace of the transmit and receive signal is shown in Figure 1c. The separation between the bases of the ultrasound holders is measured by a tape-on metric scale affixed to the lab bench.

Using the oscilloscope Time Cursor feature, the students recorded the delay between the beginning of the transmit pulse and the beginning of the receive pulse as a function of the separation of the transducer bases. Using the Excel "Trendline" feature on a scatter plot of distance vs. time delay yields a linear fit with a slope equal to the speed of sound in air. (The y-intercept, caused by physical and electronic offsets between the active elements of the transducers and the measured displacement between the bases of the holders, can be ignored.) The speed of sound found for a 10-point fit is typically within 1% of the textbook speed of sound in air of 343 m/s at 20° C.

Because the Machine Science pulsers cannot be adjusted in frequency or amplitude, the extensions that we have used at NU to measure the (lack of) dependence of the speed of sound on frequency or intensity were not possible at NECC. Using the Voltage Cursor feature the NECC students were able to measure the decrease in intensity of the signal as the distance between the transducers in increased. This can be displayed as a scatter plot in Excel, and the various fitting models available in the Trendline feature can be applied. Since the decrease in intensity of 40 kHz ultrasound in air is dominated by geometrical effects rather than by absorption at these distance scales, the best fit can be found to be a power-law $y = Ax^{-n}$ (and *not* an exponential decrease: $y = A\,e^{-\alpha x}$). The best fit value for *n* is usually found within 10% of $n=1$ which is appropriate for the inverse-square power law required by energy conservation, once it is understood that the voltage signal on the ultrasound receivers is proportional to the *pressure* of the ultrasound wave Δp and the power in Watts/m^2 is given by $S = \frac{1}{2}\frac{|\Delta p|^2}{\rho_o\,c}$. This important and subtle result can also be obtained by finding the slope of a log-log plot of intensity *vs.* distance.

2) Project 2: Control of Stepper Motor

The second module introduced in EST104 was also based on an NU HTT&TL module where MATLAB digital output commands from the Data Acquisition Toolbox are used with to control the direction and step inputs of the stepper motor control circuit. Application of direction and step commands within "for" loops provide a concrete and dramatic first introduction to programming concepts. The stepper motor face, shown in Figure 2, has two photo resistive sensors near the 0° and 90° points which are occluded by the pointer when it rotates. Sampling the resistance of these sensors with the analog input features of the Data Acquisition Toolbox can be used to create "while" loops and conditional branching statements. In the implementation at NECC, the 5V stepper motor (about $30) was driven by a National Instruments USB6008 A/D module (about $125). The aluminum stepper motor control boxes with IC controller chip, as well as the Plexiglas stepper motor holder with the photo sensors, angle dial, and pointer were made by students from the nearby Whittier Vocational/Technical High School based on schematics provided by the Department of Electrical and Computer Engineering technicians at NU. The total component cost of the stepper motor and control box (including the motor) was less than $100 each.

This project goal is to write programs to determine the degrees/step (3.75° for the half-stepped motors that we used) that the stepper motor rotates each time it receives a 5V pulse from the A/D modules. Since this is the first programming experience that many of the students have had, primitive MATLAB programs **onerot.m**, **cc.m**, **cw.m**, and **readcell.m** are provided to the students along with a startup program **setup_rot.m**. Except for **readcell (**cell#**),** these programs are script m-files and can be used in the MATLAB workspace or in student-written script files to

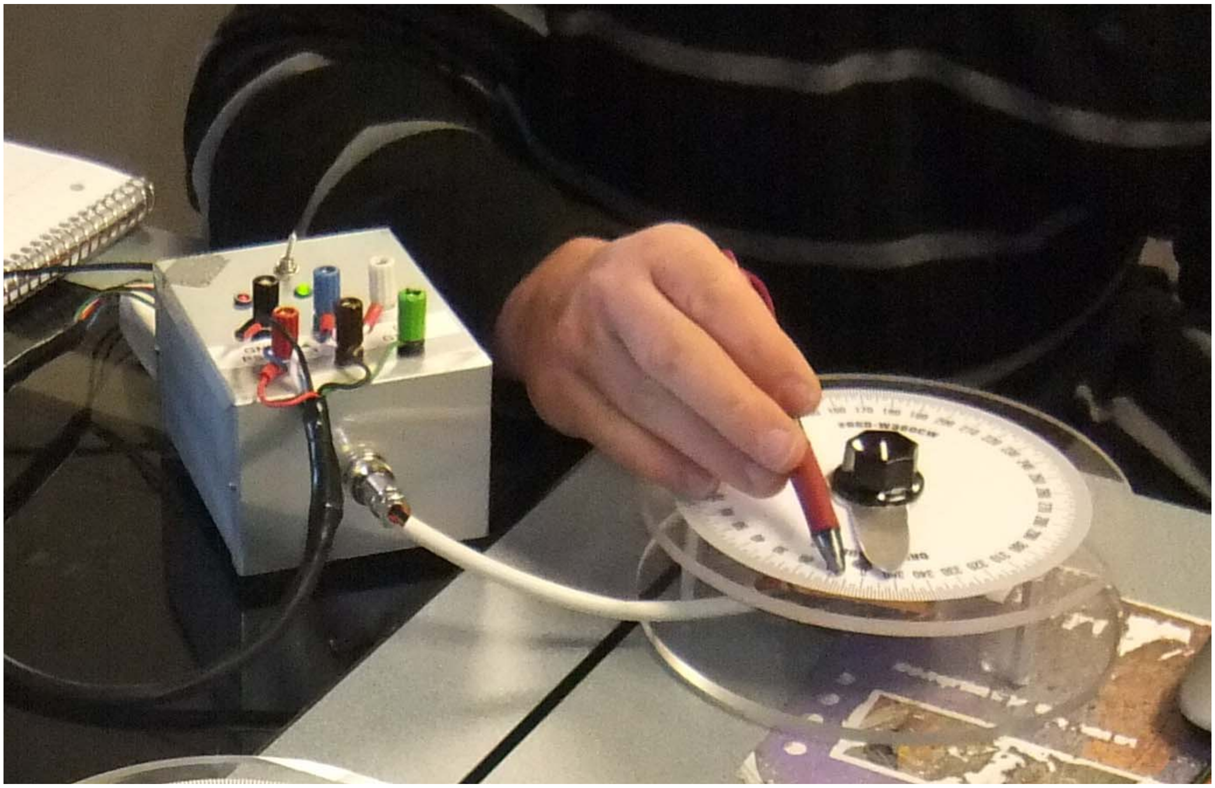


Figure 2: Stepper-motor and control box. Connecting wires from the NI USB6008 A/D module are attached to the binding posts of the control box. One of the photoresistive sensors can be seen at 0° on the angle dial just to the right of the pen point.

create simple programs that move the rotor dial clockwise and counterclockwise, ask for input, and monitor the photocell outputs as they are covered and uncovered by the pointer.

The first program the students write asks the operator to enter the initial angle and number of steps *n* to be taken. The motor then moves for *n* steps in a clockwise direction, prompts the operator to enter the final angle and the number of times that the pointer has crossed zero degrees, and prints out the calculated degrees/step. This simple program can provide a dramatic demonstration of the improvement in precision that can result from taking more steps. Since the program steps lead to easily observed effects (*e.g.,* the movement of the pointer), this is an excellent way to see how programs work. The physical movement of the pointer in response to program commands is typically found by students to be quite engaging.

As extension projects, students were asked to write programs a) to automatically determine the number of times the pointer crossed 0° by monitoring the photocell at 0°, b) to move the pointer to 0°, make *N* complete revolutions, counting the number of steps taken *n_steps*, and then find the degrees per step from *N*360/n_steps*, c) to move the pointer to 270° and make one complete revolution, plotting the output of each photocell as a function of angle, and d) make up their own program that causes the stepper motor to perform some task.

3) Project 3: Spectroscopic Identification of Colored Filters and Oils

More advanced MATLAB programming concepts were introduced with the use of a Thorlabs "no-moving-parts" CCS100 minispectrometer illustrated in Figure 3 below. The CCS1000 is a research-grade spectrometer with a fiber-optic input and a 1200 lines/mm diffraction grating that disperses input wavelengths from 350-750 nm across a detector array with 3648 pixels giving a spectral resolution of less than 0.5 nm FWHM (full width at half maximum) at 435nm. It takes a complete spectrum in 10 ms or less and can be continuously downloaded. The CCS100 spectrometer costs about $2000 – with the tungsten light source and sample chamber shown in the figure, the cost of a station is about $3000. Four spectrometers were acquired for NECC from the ALERT grant and another four were borrowed from the NU HTT&TL.

The CCS100 spectrometers are being developed to be part of the NU freshman HTT&TL experience, but the EST104 course was their first use in an "Intro to Engineering" class. Handouts describing the use of the instruments and projects to be accomplished using the spectral information were created for the EST104 class. The spectrometer provided an excellent vehicle to introduce MATLAB array operations. The spectrometer comes with a Thorlabs proprietary software package SPLICCO, but we developed MATLAB ".mex32" files to operate the instruments with MATLAB. These .mex32 files are used to set the detector array integration

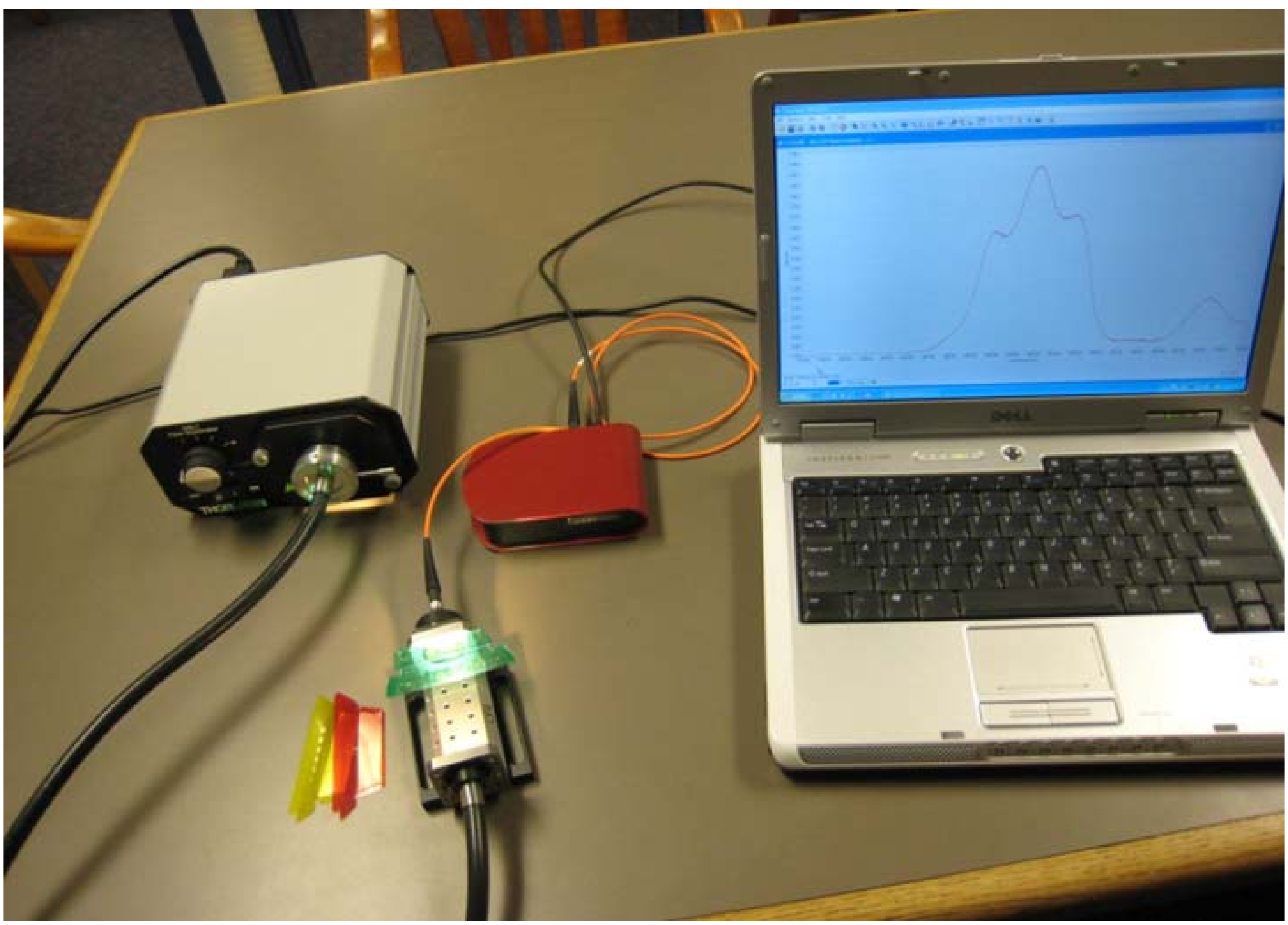
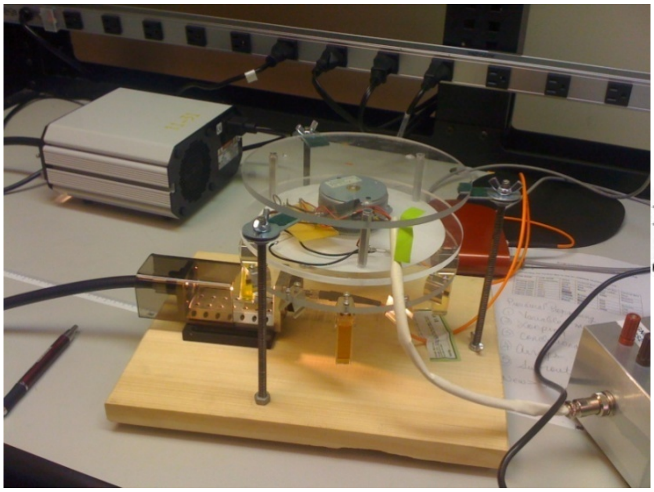


Figure 3 ThorLabs CCS100 mini-spectrometer (center) with light source (left), sample chamber (bottom), and controlling computer. The spectrum displayed is the transmission of a green plastic hang-folder tab.



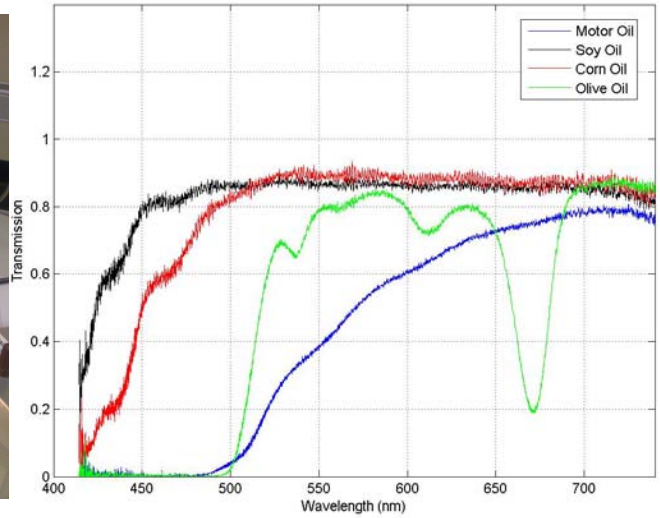


Figure 4: (Left) Project to automatically rotate vials with different oil samples into the spectrometer and automatically identify which vial contains which oil. (Right) Spectra of four oils.

time and to download (1x3648) arrays containing the wavelengths corresponding to the array pixels and the integrated optical intensity obtained at each wavelength.

The EST104 students wrote preliminary programs to download and average several spectra, and to take a ratio of the spectra called the normalized transmission, denoted by $T(\lambda)$ and calculated by $T(\lambda) = \text{Sample}(\lambda) / \text{Background}(\lambda)$. Three different spectral tasks were accomplished by the students using their MATLAB programs to control the spectrometer. First, raw spectra from several different light sources – sunlight, incandescent light bulbs, compact fluorescent bulbs, room fluorescent light, high-efficiency LED bulbs, and the Thorlabs tungsten source – were recorded and plotted. The dramatic differences in the spectra from these different light sources are striking. Fluorescent and compact fluorescent sources display a number of relatively narrow spectral lines characteristic of the phosphors coated on the inside of the tubes, whereas incandescent lights and sunlight have a continuous spectra of a blackbody radiator at 5800K (sunlight) or below 3600K (tungsten filament).

Second, sets of colored hanging-folder tabs were provided for each two-student team and they were asked to measure the normalized transmission of each color tab and then write a program to prompt the user to take the sample out (to get a background spectra), then to put the sample in, automatically calculate the normalized transmission, and print out the color of the filter. To accomplish the last task, the students needed to identify the pixel numbers corresponding to five or six spectral bands and create a set of conditions (if-else statements) on the average transmission in each of the bands to unambiguously identify which of the seven colored samples (red, yellow, green, blue, purple, or clear) was measured. Since the color intensity of the tabs varied slightly between different tabs and the intensity of the light source could not always be reproduced exactly, finding a robust solution for this task was a challenging exercise for the students. Nevertheless, all the student teams in EST104 were able to demonstrate a successful program to identify the color of the tabs.

Finally, the students were given vials of four different oils – olive oil, soy oil, corn oil, and 10W-40 motor oil – mounted on a rotating holder and asked to complete a program to use the stepper motor to rotate each sample in turn into the spectrometer beam and take a spectrum, rotate the sample out to take a background spectrum, divide the two to find a normalized transmission ratio, and identify which oil was in each vial. As can be seen in the spectra in Figure 4, the olive oil spectrum is uniquely distinguishable from the other oils.

Assessment

This was the first time that EST104 was offered, so the evaluation task posed a problem: there was no background for student performance or engagement that could be used for comparison. We decided to take a keyed student self-assessment of critical skills and abilities both before and

Table I: Pre- and Post-Module Self-Assessment on Skills in EST104 Real-Time Modules*

		Very poor & Poor	Fair	Good & Excellent
Programming	**Pre**	58	25	17
	Post	10	31	59
Problem solving	**Pre**	6	24	70
	Post	3	29	69
Debugging	**Pre**	53	23	24
	Post	17	38	45
Experimental application	**Pre**	37	50	13
	Post	10	24	66
Writing software	**Pre**	72	25	3
	Post	17	33	51
Electronic components & circuits	**Pre**	46	38	16
	Post	17	31	52
Reading schematics	**Pre**	31	52	17
	Post	10	31	59
Thinking logically	**Pre**	6	18	76
	Post	0	7	93

*Numbers represent percentage of class with given response. The number of respondents was 30 for the "Pre" survey and 29 for the "Post" survey.

after the computer-control projects to measure the self-perceived change in learning that had taken place as a result of the real-time, computer control modules. After the introductory classes in Excel and PowerPoint, and before beginning the HTT&TL modules, a survey was administered to the students to measure their self-evaluation of their "skills or knowledge" in HTT&T areas on a five-point scale (from "Very Poor" to "Excellent"). The percentages are presented in Table I under the "Pre" lines. High percentages of all students rated their skills as good-excellent in the areas of "Problem solving" and "Thinking logically", but other areas were very mixed, mirroring the varied background of the students in programming. Some students had studied programming before, but for others this was their first introduction to programming concepts.

Table II: Post Assessment of EST104 Real-Time Modules*

1. Rate your understanding of how programs control and operate electronic and mechanical components after doing all of the projects.				
Complete understanding	*Good knowledge*	*Some familiarity*	*A small amount*	*I have a lot to learn*
20%	**70%**	**7%**	**3%**	**0%**
2. The pace of the project work in class and out of class was:				
Too slow	*Easy to keep up*	*Moderate pace*	*Fast but I managed*	*Much too fast*
7%	**23%**	**53%**	**17%**	**0%**
3. Rate the materials provided in terms of learning about electronic instrumentation and control:				
Invaluable	*Very helpful*	*Somewhat helpful*	*Had to figure out things*	*Confusing*
20%	**47%**	**27%**	**6%**	**0%**
4. Rate the materials provided in terms of learning about programming.				
Invaluable	*Very helpful*	*Somewhat helpful*	*Had to figure out things*	*Confusing*
33%	**43%**	**23%**	**0%**	**0%**
5. Select the potential usefulness of these projects that you might see in your future:				
I see myself using this	*Good practice*	*Some good theory*	*Nice to know*	*Useless*
30%	**43%**	**17%**	**7%**	**3 %**
6. Overall, how much do you feel you have learned from the HTT&T projects?				
A great deal	*A lot*	*Some*	*Very little*	*Nothing*
33%	**40%**	**23%**	**0 %**	**3 %**
7. How enjoyable did you find the HTT&T projects?				
Very enjoyable	*Enjoyable*	*OK*	*Not enjoyable*	*Awful*
48%	**23%**	**17%**	**13%**	**0%**
8. I think that future EST104 classes should include HTT&T projects:				
Strongly agree	*Agree*	*Neither agree nor disagree*	*Disagree*	*Strongly disagree*
63%	**20%**	**13%**	**3%**	**0%**

*Numbers represent percentage of class with given response. Number of respondents was 29.

The survey at the beginning of the course showed a low self-assessment of "skills or knowledge" in programming, debugging, and writing software. The students considered their problem-solving and logical-thinking skills to be good or excellent before the real-time, computer control modules. Results for their skills or knowledge of "electronic components and circuits" and "reading schematics" were mixed and generally poor.

All the NECC students who completed EST104 were able to complete the three computer control projects and make a coherent PowerPoint presentation about their accomplishments. One group was able to demonstrate the automatic distinguishing of the four oil samples using the stepper motor to rotate the samples into the beam. The "Post" response to the self-evaluation of skills in Table I indicates that students emerged from EST104 with stronger feelings of self-confidence and assessment of their own abilities than when they began the HTT&TL modules. This is true even in the area of "Reading schematics" which was touched on only in a very minor way in the projects. We attribute the improvement in self-assessment here to an overall improved feeling of accomplishment and ability. This is a result which, in the experience of the authors, is often not achieved in introductory engineering courses, contributing to a high attrition rate in engineering programs. Although we have no data at this point to confirm this yet, the improved positive feelings from the HTT&TL projects is consistent with project-based learning being conducive to improved retention.

At the end of the semester, students were also asked to respond to specific questions about the experience. The results are shown in Table II. In addition to the robust self-rating of their understanding that we observed in the "Pre" and "Post" assessment data, we note that most of the students felt the material was useful, over 70% felt they had learned "a great deal" or "a lot" from the projects, over 70% found the projects "enjoyable" or "very enjoyable", and they overwhelmingly recommended that the HTT&TL projects be part of the EST104 course in the future.

Post-program responders were presented with a list of descriptors and asked to check all that apply to them. Table III contains the list of descriptors with the numbers of students that checked each of them. A majority of the students described the learning experience as "enjoyable," claimed they "learned a lot," found it "challenging," and felt a "sense of accomplishment". Many students felt "frustration." A few checked "a lot of work" and "boring." No student felt the learning experience was "too hard" and none checked "not worth it." Two open-ended comment questions at the end of the Post Assessment ("What have you enjoyed about the HTT&T projects?" and "What suggestions do you have to improve the experience for the future?") elicited comments about how satisfying it was to see changes in code reflected in the response of the instruments and a desire to see more hands-on projects.

Table III: Post Assessment of EST104 Real-Time Modules*

1. How would you describe the learning experience?	
Enjoyable	22
Frustrating	12
A lot of work	6
Too hard	0
Learned a lot	18
Not worth it	0
Challenging	16
Boring	4
Sense of accomplishment	18

*Number of students – out of a total of 29 – who checked given response. (Students could check more than one).

Conclusions and Future Plans

We found that the HTT&T elements were very effective as presented in the NECC Engineering Essentials and Design course in Fall 2011. There was marked increase in the self-assessment in every skill and knowledge area, as the data clearly indicates. Students brought varied backgrounds in mathematics and highly varied programming skills to this class. Nonetheless, the feeling of success and accomplishment as expressed by 21 of 29 respondents and the endorsement of the projects for future classes by 24 of 29 students are impressive. Two more sections of EST104 are scheduled at NECC for the Spring semester and an additional instructor is being trained to teach the class using the hands-on HTT&TL projects. Additional projects will be available next semester based on modest hardware upgrades at NECC.

While the existing oscilloscopes in the lab lack a computer interface, new oscilloscopes have been purchased with a USB IEEE488 interface that is compatible with the MATLAB Instrument Control Toolbox. With the possibility of downloading the oscilloscope traces into MATLAB arrays, a number of extensions of the "Speed of Sound in Air" module become possible. The students could program an automated range-finder/intrusion-detection system that would continuously scan for ultrasound reflections and respond based on the location of the reflecting object. Another project, conceived and tested by NECC faculty members at the 2011 HTT&TL Summer Workshop, is to locate an object (a pencil, for example) that is attached to a cylinder rotated by a stepper motor behind a curtain. The smaller radius of the pencil will generally cause a decrease in reflected intensity when the pencil is facing the transmitter/receiver, but the decreased intensity is part of a complex pattern of interference of the ultrasound reflection from the pencil and from the cylinder. This complexity can be a complication to be engineered around and also a dramatic learning experience of the effects in a real system of the phenomena of physical diffraction and interference that the students have studied in physics class.

To add additional activities to the existing activities built around the Thorlabs CCS100 spectrometers, NECC has acquired Logitech webcams for each station (about $50 each) which can be downloaded with the MATLAB Image Acquisition Toolbox to extract the RGB values for every pixel of the 240x320 display of the Logitech webcams. The red (R), green (G), and blue (B) integer values can be programmed to yield Hue/Saturation/Value parameters that can be used to identify the color of objects. A project using the stepper motor and two linear actuators to color-sort a column of painted ping-pong balls into separate tubes has been used for several years at NU, and the electromechanical parts for this project (totaling less than $80 per station) have been built for NECC by students at Whittier Vocational-Technical High School. This is a project with a large student-engagement factor which can be done as an alternative to or in addition to the spectrometer project.

The number of low-cost, high-engagement computer control projects that can be envisioned is essentially limitless, and our experience this semester is that these types of projects are successful with students across a wide range of background and abilities.

This material is based upon work supported by the U.S. Department of Homeland Security under Award Number 2008-ST-061-ED0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

Bibliography

1. Phase 1 Report, Creating a Culture for Scholarly and Systemic Innovation in Engineering Education. Washington, D.C.: American Society for Engineering Education, June 2009.
< http://www.asee.org/about-us/the-organization/advisory-committees/CCSSIE/CCSSIEE_Phase1Report_June2009.pdf >.

2. Piechota, Thomas C. et al, Project-Based Learning in a Freshman Engineering Course: University – High School Partnership. Las Vegas, Nevada, 2003.
< http://faculty.unlv.edu/piechota/proceedings/piechota-asee-psw-2003.pdf >

3. Bransford, John D., Ann L.Brown, and Rodney R.Cocking, *editors. How People Learn: Brain, Mind, Experience, and School: Expanded Edition***.** Committee on Developments in the Science of Learning, National Academy Press, Washington, D.C. 2000.

4. National Research Council, *National Science Education Standards*, National Academy Press, Washington, DC (1996).

5. Musto, J., Lumkes, J., and W. Carnell, "A Freshmen Programming Course for Mechanical Engineers Using Mechatronics Applications," Proceedings of the 2004 ASEE Annual Conference and Exposition , Salt Lake City, Utah (2004).

6. Hall, D., Barker, M., and J. Nelson, "Living with the Lab: Update on the Second Year of Full Implementation for Over 400 First-Year Engineering Students." American Society for Engineering Education Annual Conference and Exposition, June 2009.

7. Perova, N., Hart, R., Johnson, W., and C. Rogers "LEGO and ROBOLAB activities for high school and first year college classes." Paper presented at the Joint North American-European and South American Symposium: Science and Technology Literacy in the 21st Century, Richmond, VA, (2006).

8. Avanzato, R. "Collaborative Mobile Robot Design in an Introductory programming course for Engineers" " Proceedings of the 1998 American Society for Engineering Education Annual Conference & Exposition, Seattle, WA(1998).

9. Rosen W. and E. Carr, "An Application-Based Approach to Introducing Microcontrollers to First-Year Engineering Students," Proceedings of the 2010 American Society for Engineering Education Annual Conference & Exposition, Louisville, KY, 2010.

10. Jaeger, B., Whalen, R., and S. Freeman, "Get With the Program: Integrated Project Instead of a Comprehensive Final Exam in a First Programming Course," Proceedings of the 2005 American Society for Engineering Education Annual Conference & Exposition, Portland, Oregon, 2005.

11. Jenkins, T., "On the Difficulty of Learning to Program," Proceedings of the 3rd Annual Conference of LTSN-ICS, Loughborough, UK 2002. (http://www.ics.heacademy.ac.uk/Events/conf2002/jenkins.html)

12. Fisher, A., Margolis, J., and Miller, F., "Undergraduate women in computer science: experience, motivation and culture," Proc. 28th SIGCSE Technical Symposium on Computer Science Education, ACM New York, NY, USA 1997. The lack of engaging motivation for computer programming was noted as a barrier to learning. Women, in particular, are motivated by computing as "a tool to use within a broader context."