AC 2010-541: PROJECT-BASED THEMATIC LEARNING THOUGH A MULTICOURSE MULTIDISCIPLINARY ROBOTICS PROJECT

James Shey, United States Naval Academy

James Shey received the B.S. degrees in electrical engineering and computer science from the United States Naval Academy in 2003, the M.S. in electrical engineering degree from University of Maryland, College Park, in 2008, and the Master of Engineering Management degree from Old Dominion University in 2008. He is currently Active Duty Navy serving as a Senior Instructor in the Electrical and Computer Engineering Department at the United States Naval Academy and is a registered Professional Engineer in the state of Maryland.

Thomas Salem, United States Naval Academy

Thomas Salem received the B.S.E. degree from Grove City College and the M.S. and Ph.D. degrees in electrical engineering from The University of Alabama. He is currently an Associate Professor in the Electrical and Computer Engineering Department at the United States Naval Academy, and is a registered Professional Engineer in the state of Pennsylvania.

Ryan Rakvic, United States Naval Academy

Ryan N. Rakvic received the B.S. degree in computer engineering from University of Michigan and the M.S. and Ph.D. degrees in computer engineering from Carnegie Mellon University. Ryan is currently an Assistant Professor in the Electrical and Computer Engineering Department at the United States Naval Academy in Annapolis, MD. Prior to that, Ryan spent 5 years in the computer research lab at Intel Corporation in Santa Clara, California. During his time at Intel, he invented creative ways to increase the performance of Pentium microprocessors. More recently, he is attempting to utilize the emerging efficiency of Field Programmable Gate Arrays (FPGAs) to outperform a general purpose microprocessor both in terms of performance and power efficiency.

Samara Firebaugh, United States Naval Academy

Samara Firebaugh received the B.S.E. in electrical engineering from Princeton University in 1995 and the M.S. and Ph. D. degrees in electrical engineering from the Massachusetts Institute of Technology in 1997 and 2001, respectively. She is currently an Associate Professor in the Electrical and Computer Engineering Department at the United States Naval Academy. She has conducted research in several areas of Microelectromechanical Systems (MEMS) including microscale chemical reactor systems, integrated photoacoustic spectroscopy, microwave switches, variable thermal radiators and microscale robotics.

Project-Based Thematic Learning though a Multicourse Multidisciplinary Robotics Project

Abstract

The Electrical and Computer Engineering (ECE) Department at the United States Naval Academy has introduced a novel project-based thematic learning approach by incorporating a robotics project into its curriculum. This project first and foremost captures the student interest, while being flexible enough to present ECE topics at all levels of the undergraduate ECE program of study. The robot project spans from Introductory Circuits and Digital Logic Courses through to Capstone Design. In the introductory courses, the student receives a broad overview of ECE with projects designed to capture the student's interest while covering the many facets of the course. Additionally, students in the first year digital logic course are presented with innovative projects that challenge them to program basic autonomous functions into the robot. Using the robot concurrently in both courses emphasizes how the many facets of ECE work together. The robot project continues through the senior year Capstone Project, where it is used to cover such topics as design tradeoffs to advanced navigation algorithms for autonomous robots. Each course highlights different aspects of the robot. As the students understanding of ECE grows, the depth and complexity of the projects increases. By incorporating this robot platform into the curriculum there is a marked improvement in student participation and interest in the major. The robot platform successfully exceeded expectations at all levels. This robotic platform is an ideal multicourse multidiscipline project-based learning tool.

Introduction

The Electrical and Computer Engineering (ECE) Department at the United States Naval Academy has introduced a novel project-based thematic learning approach by incorporating a robotics project into its curriculum. Incorporating robotics into the undergraduate curriculum has been accomplished in many other colleges, but the robot platform is fixed.[1] Additionally, there are many commercial robot kits on the market, but the vast majority of them have predetermined hardware with well defined functions. Our ECE Department wanted something that is not only flexible, but has the minimum number of black boxes; items the students do not fully understand, but often allow plug and play compatibility. This type of project would aid students in understanding each individual component and how they work together to make the whole.

This project was created to help students, not only by interesting them in the subject, but also by providing more project based learning in the curriculum, which results in both practical applications and hands on experience. This approach provides many benefits as noted by Bower, Mays, and Miller.[2] This paper will show that a flexible robot platform for a multicourse multidiscipline project is an ideal project-based learning tool to expose students to ECE fundamentals though advanced topics.

Platform

The key to this robot project is flexibility; enough that it can be used at all levels of under graduate study. To incorporate multiple courses, the platform required the ability to mount various motors, sensor, and breadboards. The platform chosen for this project is the Trekker Chassis from SuperDroid Robots. This aluminum chassis was modified to support multiple motors and has a large enough area to mount PCBs and breadboards. Additionally, it is easy to mount various sensors to the chassis.

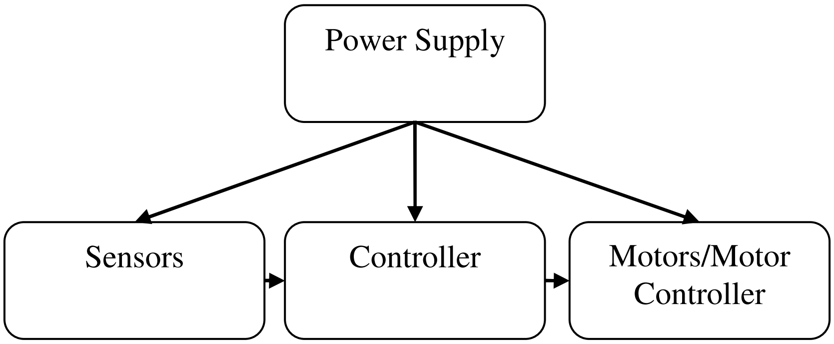


Figure 1: Robot Platform Block Diagram

The students are required to build/develop three blocks as shown in figure 1. These blocks were intentionally left vague to support a wide variety of implementations of the robot. This flexibility allows the project to be implemented at all levels of the ECE curriculum.

Sensors

Currently this chassis uses two types of sensors. The first sensor is the Fairchild Semiconductor QRB1134 Phototransistor Reflective Object Sensor. Using this sensor, allows the students to make a line following robot by properly selecting resistors to properly bias the sensor as shown in Figure 2. This sensor also covers signal conditioning because it outputs a digital signal, but this requires the students to evaluate what voltages are evaluated as a logical “0” and a logical “1” by the controller.

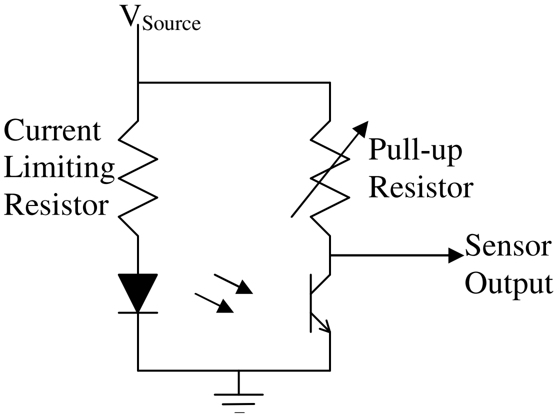


Figure 2: Circuit for biasing QRB1134

The second sensor used is the MaxSonar EZ0 Ultrasonic Range Finder. This sensor offers multiple output formats including analog and RS-232. This allows the robot to navigate to avoid obstacles while it maps out its current operating environment. Beginning students use a simple comparator to identify the presence of a wall and advanced students use an analog-to-digital converter to read the distance.

Controller

The robot platform is flexible enough to have multiple control systems. The types of controllers used will determine what signals the sensors will have to output, whether digital, analog, PWM, etc. Additionally, the chosen controller determines the allowable inputs such as analog, digital, I^2C, RS-232, etc.

The simplest control can be obtained by using digital logic gates. Though this is possible, it is not practical because the robot evolves becoming increasingly more complex as it is required to perform varied and multiple tasks. The controllers chosen for this project are the Altera Cyclone II field-programmable gate array (FPGA) on DallasLogic's Niomite board and the PIC 16F884 Microcontroller.

The Niomite board was chosen for several reasons, first the physical form factor, 2.1 by 1.6 inches, allowed it to be easily mounted on the robot. Secondly, it contains Altera's Cyclone II FPGA which is familiar to the students and has more than enough room to accommodate students' advanced projects.

The PIC 16F884 was chosen because it is common microcontroller that the students are taught in their second year in the ECE curriculum. The PIC does have several advantages over the Niomite board. Its cost is considerably less, while it has additional features including onboard A to D converters and PWM output schemes. The PIC does have the disadvantage of running only a serial program at the same speed as the FPGA.

Motors Controller

The standard motor controller for the robot uses a transistor switch as shown in figure 3. This scheme is simple to implement especially for the students exposed to ECE concepts for the first time and does allow for easy digital control from either the FPGA or microcontroller. Furthermore, this scheme can be easily used for PWM control for advanced students. Some thought was given to using commercial motor controllers, but it was decided that having a simpler controller that the students could understand would be more beneficial than introducing a black box.

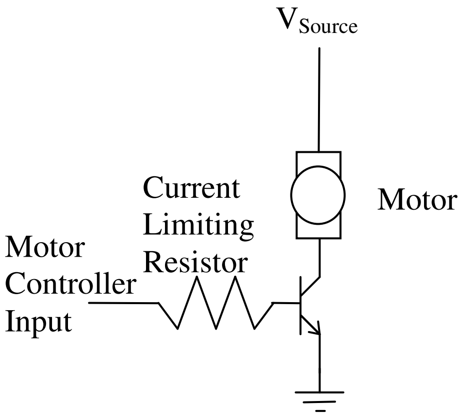


Figure 3: Motor Controller

Power Supply

To have an autonomous platform, various rechargeable batteries are supplied to the students. The students must determine the minimum and maximum voltages for the FPGA or microcontroller, the motors and the sensors. Additionally, voltage regulators are used to provide a fixed voltage for the FPGA and microcontroller. The students are required to compare the characteristic of each battery and pick one that will best suit their robot's individual needs.

Custom Printed Circuit Boards (PCBs)

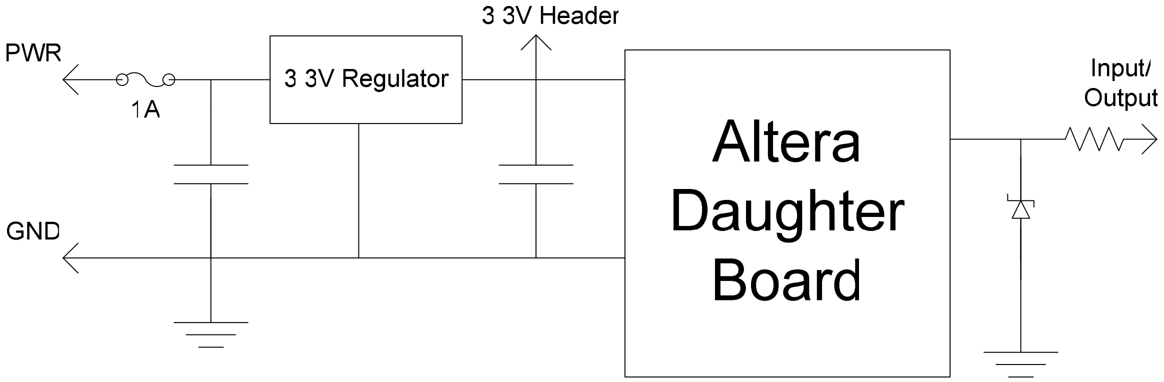


Figure 4: Robot Protection Board Schematic, Note: Only One Input/Output Shown

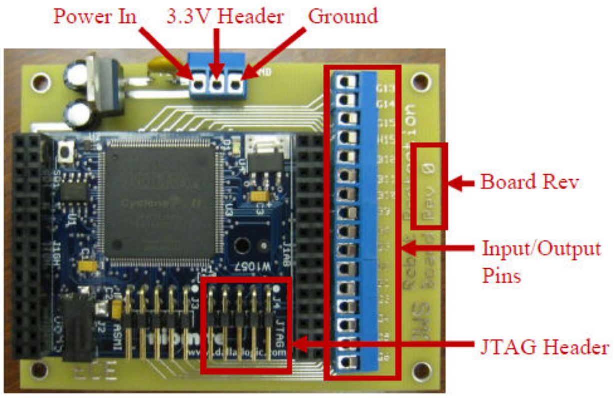


Figure 5: Robot Protection Board

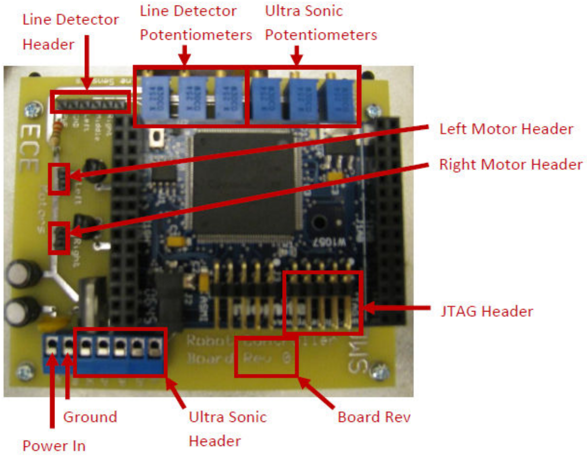


Figure 6: Robot Controller Board

There were two custom PCBs developed for this project. The first is the “Robot Protection Board.” This board was developed to protect the FPGA. The Niomite board uses 3.3V logic and will be damaged by higher voltages, so a zener diode was used to limit the voltage seen by the Niomite board and a resistor was used to limit the current flow as shown in figure 4. Additionally, a 3.3V Regulator was placed onboard to provide power to the Niomite Board. The completed board is shown in figure 5.

The second custom PCB is called the “Robot Controller Board.” This board was developed to be a test bed for student’s very-high-speed integrated circuit hardware description language (VHDL) programs. This board (see figure 6) contains all the sensor biasing resistors and motor controllers, and provides a platform free of potential errors in the student’s hardware.

Courses Utilizing the Robot Platform

The benefit of the robot platform was the flexibility. This allows the ECE Department to currently use the robot as a project-based thematic learning tool in four courses: two introductory courses, a third year course, and the fourth year capstone course. Most of the improvement to the curriculum made by this project is seen in the introductory courses. This project brings design to the introductory level and sets the foundation that students will build on in later courses. The change in the curriculum is similar to changes implemented at Duke University, but differentiates itself by tying the two introductory courses together.[3] The introductory circuits course develops the hardware; the "brawn", while the introductory digital logic course programs the controller; the "brain".

EE221: Introduction to Electrical Engineering I

The first semester ECE course on circuit analysis fabricated the robot platform and introduced the students to the basic concepts of engineering design. In total, six hours of class and laboratory time were specifically dedicated to the project. Three additional laboratory periods were spent on experiments that were helpful for the project; biasing a light emitting diode (LED), signal processing of sensor data, and a sensor controlled motor operating circuit. For the robot project, students were permitted to work in teams of three and each team was given the opportunity to select a design objective, speed or endurance, for their robot similar to competitions outlined by Jackson and Ricks.[4] Documentation was provided that detailed the necessary assembly instructions as well as individual component datasheets for the sensors, batteries, transistors, and motors. A preprogrammed FPGA containing a line following state machine was supplied to the students, although several students opted to use their own state machine design from EE242. Students were expected to make design decisions on selecting the battery, wheel size, and motors to be used in their project to best meet their design objective (see Appendix A for parts list). Project deliverables were a functioning robot and complete design project report documenting the team's experience. The culmination of the project was a course-wide competition in which the students competed across all class sections for a grade bonus based on their design objectives. Testing of the speed objectives was accomplished on a course consisting of several curves, and the endurance objective was tested on course designed as a large oval.

Both of the courses were established in the hallway. This is noteworthy in that it generated significant interest among students who were passing through the building. These passersby would typically stop and engage the EE students in conversation about the details of their robots. In fact, this phenomenon was also observed when the students were testing their projects prior to the competition.

This project also highlights several problems and successes the students had with the project. Properly biasing the line sensors proved to be the most difficult part of the project. Reading datasheets provided an additional challenge. There were several positive outcomes, including the use of lab notebooks to review topics from previous labs as well as trouble shooting experiences.

Every group successfully completed the "brawn" part the robot project and competed in a challenge (See Appendix B). The students expressed a positive experience on this project even though it required numerous hours of work outside of the allotted class time.

EE242: Digital Systems

The first semester ECE course on digital logic lays down the foundation for FPGAs, microcontrollers, and programming thus allowing the students to program the "brains" of the robots. This course introduces students to topics including: Boolean algebra, Karnaugh mapping, binary arithmetic, decoders, encoders, multiplexers, latches, flip-flops, which all lead to state machine design. In the laboratory, the students learn to program VHDL, and test their designs on FPGAs. In particular, by the end of the course, the students have the ability to program both combinational and state machine designs with VHDL.

Students are given three specific projects with regard to the robot, termed Sheybot. The first project covers a combinational logic line follower. After the robot demonstrates its ability to navigate a serpentine course, it is presented with a course requiring 90° turns, and due to the build of the robot, the sensors will leave the line and the robot will become "lost." This brings the students to the next project: creating a state machine line follower. With this project students will create a state machine that will remember the robot is turning even when the robot looses contact with the line. The final part of the project is creating a state machine that will allow the robot to avoid objects without the assistance of the line. Once an object was detected in front of the robot, the robot would turn 90°. In the case where the robot is in a corner, it will turn out of the corner.

The results of Sheybot in EE242 this semester were extremely positive. All students completed the first two parts of the laboratory, and approximately 85% completed the third task. Sample Student Code can be found in Appendix C. Additionally, the feedback from the students was also very positive.

From the instructors' point of views, we noticed a more proficient understanding of VHDL due to the Sheybot project. Part one helped the students get acclimated to the Robot, dealing with inputs and outputs of the design. In this part, the students also learned that not all designs are required to have memory. Parts two and three of the Sheybot tested their ability to use VHDL to design a state machine. Part two required them to design their own state machine where each state represents the direction that the robot is going. Part three required the use of the "distance" sensors in order to perform object avoidance. In addition to using a state machine in this part, the students also needed to keep track of the time for a turn. Although the students had just performed well on an exam which tested their understanding of state machine design with VHDL, it was apparent from this part of the project that they did not truly understand the concurrency of VHDL. One 'A' level student questioned why his *for* loop inside of a process was not counting. He soon became aware that the process was controlled by a clock which is the hardware way to count. Unlike a software *for* loop, a hardware loop cannot keep track of time for you alone.

Additionally, it was obvious to the instructors that the students were extremely motivated by this real-life example. Overall, students voluntarily had a marked increase in their lab time. They could now visually see the impact of their logic design on a real moving robot that they are physically designing and building in a separate course.

EE361: Microprocessor-Based Digital Design

In EE361 students were given the opportunity to use the robot as a final project. For this project the design was not focused on design tradeoffs, but on using the microcontroller as the controller for the robot. The students were not given a choice of motors, and were tasked with developing a line following robot, with the condition that should the robot get off of the line it would remember to keep turning to reestablish contact with the line.

The students succeeded at programming the PIC microcontroller to read from the QRB1134 sensors and determine which motors should operate however, they failed in their implementation of the motor controller. The group did succeed in programming the PIC using a C-compiler as well as programming in assembly language.

EE411: Electrical Engineering Design I

The robot platform was used in the fourth year design course as a model design project similar to the capstone course project outlined by Jackson and Ricks.[4] The students were divided into teams of two and presented with the task of creating an autonomous line-following robot. They were given the same set of parts as was used in the introductory courses as well as relevant data sheets. Unlike the introductory-level course, this project was not partitioned into tasks for the students. The fourth year students had to break the system down into its components and determine how to implement each component. This process reinforced the course emphasis on the design process and program management. The project also provided a useful review of circuits, electronics and digital design, which laid the foundation for development of their own capstone projects later in the semester.

This project exposed many weaknesses in the fourth year students. Being able to read and effectively use datasheets and breaking down the project into smaller parts were both issues that had to be addressed in class. These issues resulted in 80% of the groups completing the project, within the allotted two-week time period. But even for the groups that did not finish, there were benefits. This project provided more experience in trouble shooting, as well as design tradeoffs which will be used on their senior capstone project.

In the future we will expand the project for the fourth year students to differentiate it from what the students would have then seen in their earlier courses. For example, the addition of a radio-frequency control module would facilitate a review of the communications systems concepts that are studied in the third year. Also, more design alternatives like the use of the PIC microcontroller as an alternate controller could be presented to reinforce the evaluation of engineering trade-offs.

Problems Encountered By Students

There were several issues encountered by a majority of the students, however, the most common encountered was properly biasing the line following sensor. Numerous line sensor LEDs burned out due to not using a current limiting resistor, even with full access to the datasheets and, in some cases, previous labs on biasing a LED. To address this issue more time will be spent covering datasheets.

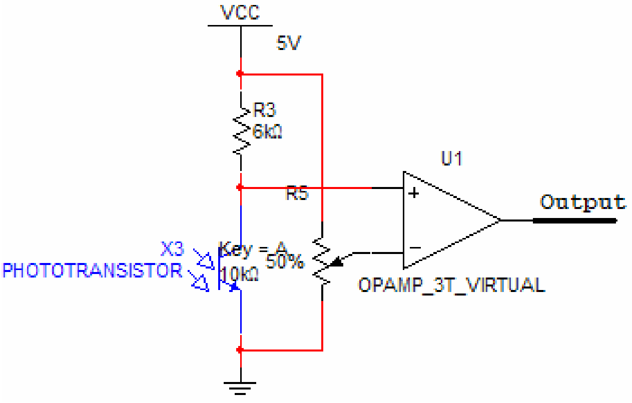


Figure 7: Improved Line Sensor

Additionally, the students had problems setting the potentiometer to give a high enough voltage for a logical one, while having a low enough voltage for a logical zero. Next year, the ECE Department will be implementing a slightly different circuit for the line follower. The Comparator is covered in EE221 so all students working on this project will have had exposure to this device and it will not add excessive complexity as shown in figure 7.

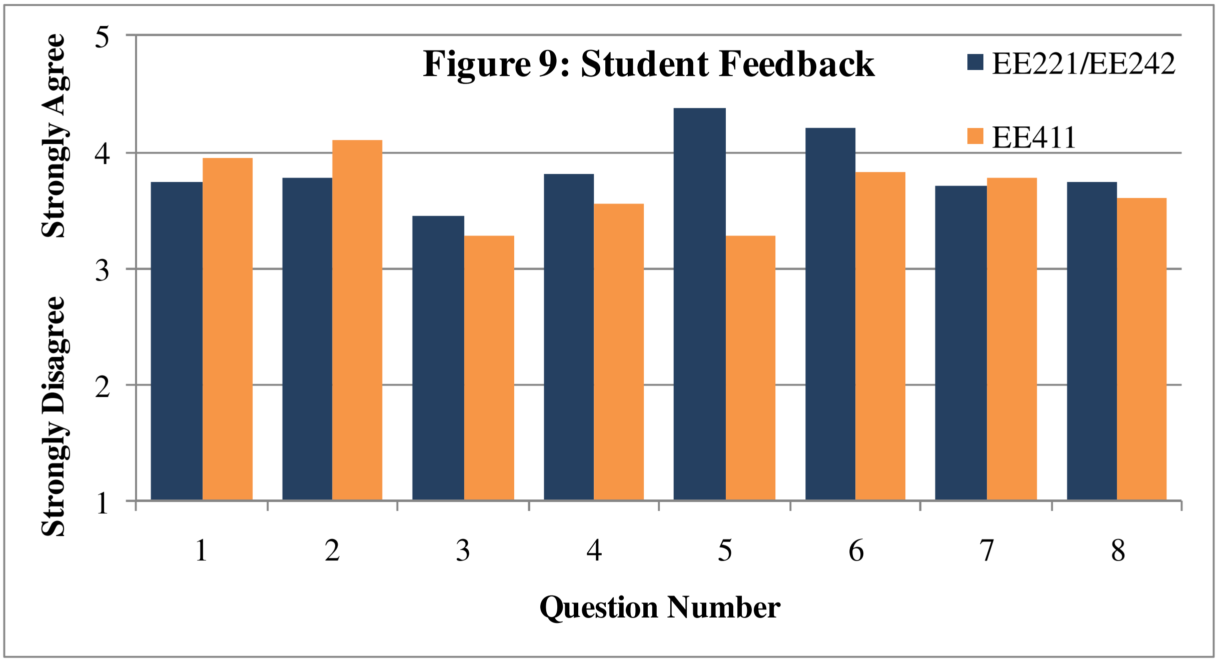
Student Feedback

Student quantitative and qualitative data on all levels of the project has been positive. To evaluate student data an eight-question questionnaire was given to all students. The questions can be found in figure 8. Students rated each question on a scale of 1 (Strongly disagree) to 5 (Strongly agree). The averages can be found in figure 9. EE221 and EE242 are grouped together because all students are required to take both courses at the same time and only one set of surveys was given to these students. Additionally, the students in EE221 and EE242 were asked if the project should be continued in the class next year, and there was a 100% agreement that the project should be continued.

Question:	The project increased my understanding of:
1	Basic Electrical and Computer Engineering Fundamentals.
2	Biasing components for proper operation.
3	Datasheets.
4	Applications of FPGAs to real world problems.
5	Programming VHDL.
6	Concepts I learned in this project I will apply to future projects.
7	This project highlighted design tradeoffs.
8	I enjoyed this project.

Figure 8: Student Feedback Questionnaire

These results, along with the instructors' quantitative and qualitative data show increased participation and point to students being more involved with their learning. The students are taking a more proactive role in their total education.



Future Endeavors

Currently the ECE Department is in the process of expanding the inclusion of the robot platform in additional courses. The benefit of this platform is that it is flexible enough to accommodate a wide variety of sensors, motors, and various actuators along with more complex tasks for the robot to perform. Additional sensors, such as laser range finders, metal detectors, accelerometers, and GPS can easily be integrated. Motor controllers using H-bridges or PWM control schemes have been discussed.

These future endeavors would not be included in the introductory courses, but rather, would be for the third and fourth year students who have already been introduced to the robot and therefore, have more background in ECE allowing them to tackle more advanced topics.

Conclusion

The robot platform successfully exceeded expectations at all levels. This project set out to show that a flexible robot platform for a multicourse multidiscipline project is an ideal project-based learning tool. The exposure to ECE carried the students from fundamentals through advanced topics. The project proved flexible enough to span several different courses, as well as several different topics within the ECE curriculum. It covered introductory topics, to more advanced algorithms, through to design tradeoffs. Future work will only expand the breadth of topics. Additionally, student feedback was positive; the project got students interested in the topics and encouraged learning. Additional research could carry students into multidiscipline areas such as mechanical engineering or systems engineering.

Combining the two introductory courses has given a broader picture to students showing that the ECE curriculum is interdependent; one cannot just understand one topic, when they are all interdependent. Although there is only one semester of feedback from the students, and yet, no long term quantitative data, both the students and faculty agree that this project has stimulated student interest and has facilitated a more project orientated group of young engineers ready to continue their education.

Bibliography

1. M. Lowe, H. Moore, E. Langrall, and C. Gehrman, “Robots in the introductory physics laboratory,” American Journal of Physics, vol. 76, issue 10, pp. 895-902, Oct. 2008.

2. K. C. Bower, T. E. Mays, and C. M. Miller, “Small Group, Self-Directed Problem Based Learning Development In A Traditional Engineering Program,” 34th ASEE/IEEE Frontiers in Education Conference, Session S1B, pp. 16-21, Oct. 2004.

3. L. G. Huettel, A. S. Brown, K. D. Coonley, M. R. Gustafson, J. Kim, G. A. Ybarra, and L. M. Collins, “Fundamentals of ECE: A Rigorous, Integrated Introduction to Electrical and Computer Engineering,” IEEE Transactions on Education, vol. 50, issue 3, pp. 174-181, Aug. 2007.

4. D. J. Jackson, and K. G. Ricks, “FPGA-Based Autonomous Vehicle Competitions in a Capstone Design Course,” Proc.2005 IEEE Int. Conf. on Microelectronic Systems Education, pp. 9-10, June 2005.

Appendix A: Parts List

1-Chassis and hardware
1-Robot Protection Board
1-DallasLogic Niomite Board
3-QRB1134 Phototransistor Reflective Object Sensor
3-LV-MaxSonar-EZ1 Ultrasonic Sensor
2-2N3391 Transistors
3-200KΩ Potentiometer
1-Breadboard
Various Resistors

Motor choices:
HS-322 Servo Motor
HS-422 Servo Motor
Jameco Reliapro 38-006 (Jameco Part Number 253497)

Wheels choices:
Small Wheels (3 ½” diameter)
Large Wheels (4 ½” diameter)

Battery choices:
9V
D-Cell
C-Cell
AA-Cell
AAA-Cell

Appendix B: Student Photographs

Appendix C: Sample Student Code

Part 2 Sample Code

```
library ieee;
use ieee.std_logic_1164.all;
--
entity sheybot2 is
        port (clk, L, R, M : IN std_logic;
                leftm, rightm :out std_logic);
end sheybot2;
--
architecture dataflow of sheybot2 is
        TYPE state IS (turnL, turnR);
        SIGNAL pr_state, nx_state: state;
BEGIN
        PROCESS (clk)
        BEGIN
                IF (clk'EVENT AND clk='0') THEN
                        pr_state<=nx_state;
                END IF;
        END PROCESS;
        PROCESS (pr_state)
        BEGIN
                CASE pr_state IS
                        WHEN turnL=>
                                rightm<='1';
                                leftm<='0';
                                IF (R='1') THEN
                                        nx_state<=turnR;
                                ELSE
                                        nx_state<=turnL;
                                END IF;
                        WHEN turnR=>
                                rightm<='0';
                                leftm<='1';
                                IF (L='1') THEN
                                        nx_state<=turnL;
                                ELSE
                                        nx_state<=turnR;
                                END IF;
                END CASE;
        END PROCESS;
END dataflow;
```

Part 1 Sample Code

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY part1 IS
        PORT(l, m, r : IN
STD_LOGIC;
                lmotor, rmotor :OUT
STD_LOGIC);
END part1;
ARCHITECTURE robot of part1 IS
BEGIN
        lmotor<= NOT r;
        rmotor<= NOT l;
END robot;
```