AC 2011-392: INTRODUCING ADVANCED ENGINEERING TOPICS TO FRESHMEN STUDENTS USING ROOMBA PLATFORM

Farid Farahmand, Sonoma State Univeristy

FARID FARAHMAND is an Assistant Professor in the Department of Engineering Science at Sonoma State University, CA, where he teaches Advanced Networking and Digital Systems. He is also the director of Advanced Internet Technology in the Interests of Society Laboratory. Farid's research interests are optical networks, applications of wireless sensor network technology to medical fields, delay tolerant networks. He is also interested in educational technologies and authored many papers focusing on eLearning and Active Learning models.

Saeid Moslehpour, University of Hartford

Saeid Moslehpour is an Associate Professor in the Electrical and Computer Engineering Department in the College of Engineering, Technology, and Architecture at the University of Hartford. He holds PhD (1993) from Iowa State University and Bachelor of Science, Master of Science (1990), and Education Specialist (1992) degrees from University of Central Missouri. His research interests include logic design, CPLDs, FPGAs, electronic system testing and distance learning. Email: moslehpou@hartford.edu

Introducing Advanced Engineering Topics to Freshmen Students Using Roomba Platform

1. Introduction

This paper describes integration of low-cost Roomba autonomous vacuum cleaner robots into Introduction to Engineering course offered by the Department of Engineering Science at Sonoma State University. The main purpose of introducing Roomba platform is to teach and integrate introductory engineering concepts, such as numbering systems, microcontrollers, serial and parallel interfaces, sensor technologies, wireless technology, and high-level programming language, such as LabVIEW. Through his experiment, as their final design project, teams of freshmen students were required to design their own creative Roomba projects involving sound, light, sensors, and even LEGO for creating arms and other moving parts.

In this paper we also show how interested and more advanced student teams can develop more involved projects. We conclude the paper by showing student responses to the quantitative and free-form questions concerning the integration of Roomba.

2. Background

Almost all engineering programs require students to take at least one course in microcontrollers or microprocessors. Most of the engineering students are also required to take one semester of programming in their first or second year. However, the students often find programming classes uninspiring, criticizing assignments as being too abstract and unrelated.[1] Consequently, by the time they take microcontroller classes, many of the students often have a difficult time to follow algorithms and programming sequences. In fact, some studies show that while students can often learn the syntax of individual commands in any programming language quickly, the skill of combining commands into a program and integrating them with a hardware device is harder to learn.[2]

In order to better prepare freshmen students and introduce them to available lab resources and general engineering and computer science curriculum, the Department of Engineering Science at Sonoma State University has been requiring students to take an introductory course to engineering. Introduction to Engineering (ES110) is a two-credit hour core engineering course with one hour lecture and 2 ½ hours of lab per week. Majority of students enrolled in this course are freshmen and most likely this is their first year experience in college. ES110 is considered to be the first formal introduction to various fields of engineering for many of these recent high school graduates. Furthermore, the course exposes students to basic programming software, such as MATLAB and LabVIEW, as well as engineering topics, such as numbering systems, digital and analog signals, microcontrollers and microprocessors, basic circuit components, etc.[3]

A major shortcoming of ES110 curriculum has always been isolation of topics and lack of a comprehensive and yet, moderately complex project. This paper reports an experimental approach to integrate topics, such as microcontrollers and interfacing, basic programming skills, basic knowledge in electrical circuits, and sensors to the Introduction to Engineering course. In this approach we dedicate the second half of the course to combine the introductory engineering concepts studied in the first half of the class. Thus, through hands-on projects and activities, using Roomba, students can combine their knowledge in basic circuit design, sensors and transducers, programming, and microcontrollers, and design their own functionally working Roomba-based projects.

3. Classroom Experiment

In order to effectively implement the proposed add-on Roomba project, we revised the lectures and laboratory activities in ES110 and focused on the following leaning goals through hands-on activities:

1- gaining basic knowledge in digital/analog signals and circuits;
2- receiving hands-on experience with microprocessors and interfacing technologies;
3- being able to develop simple embedded codes and algorithms to control electromechanical systems;
4- obtaining working knowledge about sensors and feedback systems.

Furthermore, in designing the introductory materials for freshman students we tried to highlight the following four broad educational objectives:

1- designing an inspiring and stimulating learning environment;
2- promoting student creativity by asking students to utilize their knowledge and talents in solving practical problems.
3- encouraging students to interact and collaborate towards an innovative design project;
4- preparing students with necessary knowledge for more advanced engineering and computer science courses in the future.

3.1. Methodology

Table 1 lists the details of each scheduled class lecture. We introduced the Roomba experiment in the middle of the semester, around the 6^{th} week. As noted in the table, the first and second introductory lectures include a brief review of numbering system, analog and digital signals, and microcontrollers. Many of these topics were also discussed in the first half of the semester.

In weeks seven and eight, we introduced the Roomba platform. We demonstrated various features of Roomba, including its actuators, motors, power source, and LEDs, and described how a Roomba is controlled via its microcontroller (Roomba uses a Freescale, formerly Motorola, MC9512E).

Through demonstration and hands-on experience, students become familiar with how they can send basic commands to Roomba directly using a standard terminal emulator such as *Realterm*.[9] They tested different commands and learned about various capabilities of Roomba.

In weeks eight and nine, students were briefly introduced to wireless technologies. In particular, they learned about Bluetooth technology and how it can be utilized to interface a PC to the Roomba. They were also introduced to basic electronic components (e.g., resistors, capacitors, transistors, and relays) and built a few simple circuits.

In weeks nine and ten, the lectures focused on introducing LabVIEW[5] and its applications. We covered basic features of LabVIEW programming and demonstrated a few programming examples. Each lecture was followed by a separate laboratory time for hands-on practice with LabVIEW and completing lab assignments.

In week ten, we introduced the students to our own in-house developed LabVIEW programs, or Virtual Instruments (VIs). Using these simple VIs students could interface the PC to Roomba in order to program the Roomba to perform various functions, such as, playing music, moving in different directions, reading different internal sensors, and controlling external attachments.

Table 1. Lecture and laboratory materials for introducing the add-on Roomba experiment in a 12-week Introduction to Engineering course (**) .

Week	Lectures	Duration	Lecture coverage
6,7	3	3:45	- Introduction to Hexadecimal and Binary numbering systems (*) - Introduction to microcontrollers (*) - Understanding analog and digital signals (*) - Understanding basic interfaces to a microcontroller
7,8	2	2:30	- Introduction to Roomba platform - Examples of commands used in Roomba - Introduction to various sensors used in Roomba
8,9	2	2:30	- Introduction to Wireless and Bluetooth - Introduction to basic electronics circuits (*)
9,10	3	3:45	- Introduction to LabVIEW - Demonstrating LabVIEW examples - Basic LabVIEW experiments
10	-	-	- Deciding on a project idea - Creating groups (two students per group) - Creating a blog page and describing the project objective
11,12	-	-	- Presenting the final working prototype - Submitting the formal project report
(*) The topic was also covered in the first six weeks of the course.			
(**) An example of student blog can be found in reference [3].			

In week ten students created their own groups and proposed a project idea; no more than two students were allowed in each group. Each group was given three weeks to complete their proposed design project and present a formal final report. A final report template was provided to each group. We asked all projects to include controlling an external attachment or interfaceing with at least one of Roomba's internal sensors.

Throughout weeks 10-12, we required all the students to post their progress, issues, and problems, on the class discussion group, powered by groups.yahoo.com. All groups were encouraged to create their own web page or Blog. In order to assist the students, we offered a few informal half-an-hour extra sessions for students to discuss their problems and questions. In each session the instructor also addressed some of the issues posted on the discussion group.

3.2. Project Grading

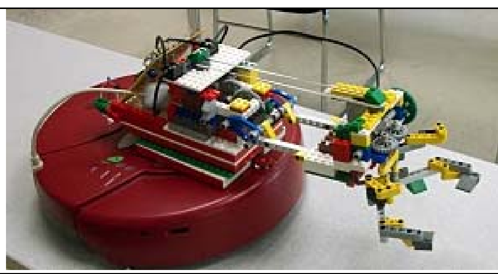
Each group received a project grade according to the following criteria: complexity and innovation (25%); user-friendly interface (25%); external/sensor interface complexity (25%); and design reliability (25%). Incomplete projects did not receive any grades.

3.3. Experiment Details

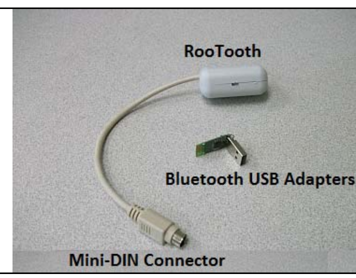
In this section we briefly describe the details of the add-on Roomba experiment. We also elaborate on experiment's hardware and software requirements. Overall, the experiment was divided into four sections:

1- Understanding the Roomba ROI commands through which student could control the Roomba's actuators, such as motors, LEDs, and speakers, and communicating with sensors.
2- Creating a simple circuitry to interface external devices to the Roomba for additional functionality, e.g., adding an external arm to the Roomba.
3- Learning LabVIEW and being able to create simple VIs.
4- Integrating LabVIEW and ROI commands such that students can write their own LabVIEW programs to create their own Roomba behavior with additional functionalities; for example, making Roomba to perform various moving sequences, play songs, or detect objects in its vicinity.

Figure 1(a) shows the complete Roomba with its external LEGO arm connected to it.



(a) Complete Roomba with its LEGO arm



(b) Wireless interface to the Roomba

Figure 1: Modified Roomba and its wireless interface.

Roomba and ROI Command List: The Roomba Open Interface (ROI), originally called the Serial Communications Interface (SCI),[5] allows users to control the Roomba through its external serial port (Mini-DIN connector), as shown in Figure 1(b). As we mentioned above, the ROI includes commands to control Roomba's actuators (motors, LEDs, and speaker). The ROI protocol is very simple and operates at 8 bits. In this section students learn about opcodes and data bytes, and the sequence in which data bytes are transmitted serially from a PC to the Roomba. For instance, using opcode 139 (decimal) Roomba's LEDs can be controlled, their color can be changed, and their light intensity can be modified. Similarly, using opcode 140 students can play various notes and determine the duration of each note; thus, eventually making their own song! All of these steps require complete understanding of bits, bytes, opcodes and commands, and numbering systems, including binary and hexadecimal.

In addition to the wired serial port interface, another possible flexible method of communication with Roomba is Bluetooth, which uses the unlicensed 2.4 GHz ISM (Industrial Scientific Medical) band. In order to implement Bluetooth interface, we used a RooTooth[8] and a Bluetooth USB adaptor. We encouraged interested students to build their own Bluetooth interface for much lower cost.[7]

Designing Simple Circuitries: An interesting, yet time consuming, portion of the experiment was introducing and building simple transistor-based drivers required to interface the Roomba with external devices, such as motors, LED, and relays. In this section, students became familiar with transistors and were asked to design their own simple pnp-transistor switching circuit. We had built several drivers, so students could test and use them as example to design their own. The cost of each driver, including the prototype board, is less than $3 and we asked students to purchase their own parts. Figure 2(a) shows the simple driver circuitry students were given to interface with the Roomba arm. Figure 2(b) shows the LEO-based Roomba arm, which contains two motors for moving the arm in different directions. Figure 2(c) shows the modified Roomba with a LEGO platform so students can design their own LEGO-based electromechanical devices and install them on top of the Roomba platform.

LabVIEW Software: We spent three lectures introducing basics of LabVIEW. Students completed various class exercises in order to become familiar with the software. Each group was given a CD with the following materials:

a) A complete set of lecture materials, LabVIEW program examples, and final report template.
b) Interfacing programs to communicate with the Roomba wirelessly; using these programs students could perform basic functionalities such as moving Roombas' motors, turning on its LEDs, and programming it to play different sounds.
c) Complete circuit diagrams used to interface the Roomba to external devices.
d) All reference materials related to Roomba, including ROI Spec and a complete list of commands.

During practice labs, we asked each student to execute the provided VIs and become familiar with the way the VIs operate. Then, we asked students to modify the basic VIs and include additional functionalities using new opcodes. For examples, students had to modify one VI to make the Roomba move in a square pattern.

Team Projects: In the last part of the experiment, we asked each group to design its *own* innovative project. In this part, students had to write their own LabVIEW program to add new functionalities to the Roomba's operation. For example, one team created a singing Roomba using its piezo electric beeper. Using appropriate opcodes, each note is specified by a note number (MIDI note definition) and its duration can be modified in fractions of a second. In this simple project, students became familiar with concepts such as frequency and amplitude. They also demonstrated the MIDI notes using an oscilloscope. Figure 3 shows the VIs that some students created to control the movement of the Roomba and play notes. Figure 4 shows the actual code for controlling Roomba's movements and an attached external arm connected to it.

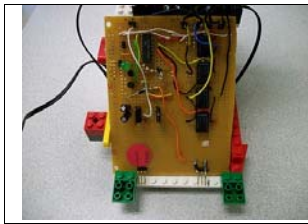
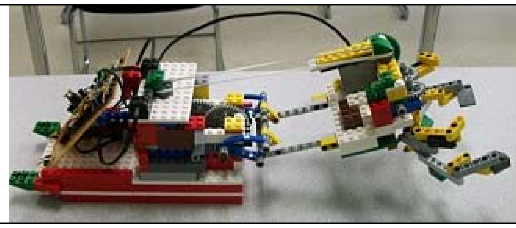
		
(a)The transistor circuitry for interfacing the LEGO arm to the Roomba	(b)The LEGO arm designed by students for Roomba	(c)Roomba with Bluetooth Interface and LEGO platform

Figure 2: Components of the modified Roomba.

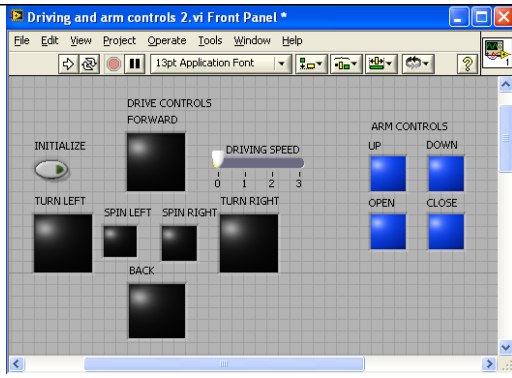
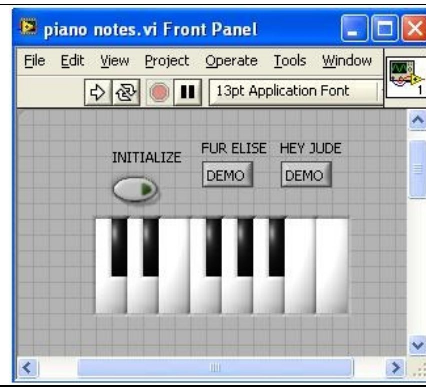
	
(a) Basic VI for Roomba movement and LEGO arm control.	(b) VI for playing notes by Roomba.

Figure 3: LabVIEW front panels used in the Roomba experiment.

3.4. Student Project Examples

Majority of our students did not have any problem finding a project idea and most students were highly motivated. Examples of projects designed by our students were as follow:

- Making Roomba to draw simple shapes;
- Interfacing Roomba to more complex electromechanical devices;
- Adding sensors to the arm;
- Developing a VI to interface with all the internal sensors in the Roomba.

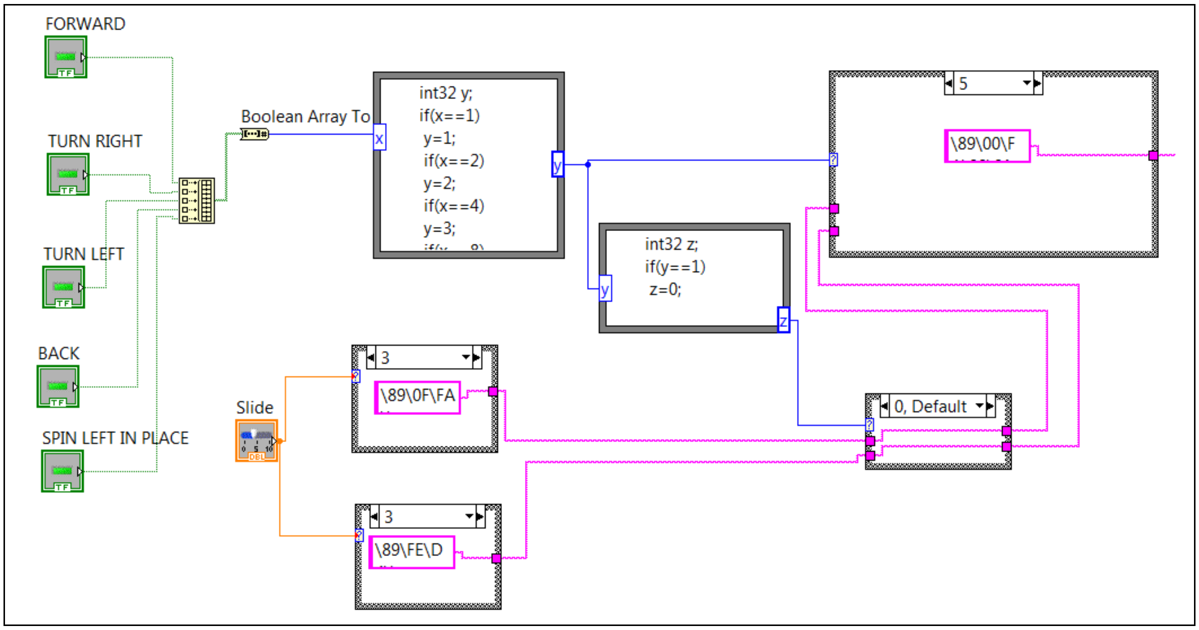


Figure 4: LabVIEW code for activating the motors in Roomba by sending the appropriate opcodes.

4. Experiment Outcomes

Our main motivation in developing this experiment was introducing Robotics, microcontrollers, sensor technology, and programming to the Introduction to Engineering course. We also wanted our students to become involved in a project, which integrate all the above topics.

4.1. Assessment Results

The Roomba experiment was integrated in both sections of first year introduction to engineering (ES110). In the second half of the semester, we included the related lectures, as shown in Table 1. At the end of the semester, the students were asked to complete a mandatory short survey for the course and give their feedback on the Roomba experiment. The survey was primarily based on quantitative questions, with the last question asking for student comments and suggestions. The combined demographics of the two sections were 25 percent females and 75 percent freshmen. The responses to some of the quantitative questions concerning the sensor laboratory experiment are shown in Table 2.

In their comments, many students indicated that, in spite of the fact that they felt the project was very educational, more time must be given to complete the assignment. Some students believed that instructions listed in the lab handouts were not very clear. In fact, throughout the experiment, the students were very active in providing corrections. Their feedback greatly assisted us in updating lab handouts.

In regards to microcontrollers and understanding binary and hexadecimal numbering systems, some students expressed that they needed more time and practice. They also mentioned that more assistance should be provided during the semester, in particular with regards to LabVIEW programming. In general, most students expressed satisfactions with their lab partner and found her/him helpful in completing the project. Almost 95 percent of the students completed a working prototype by the end of the semester.

Table 2. Student responses to the Roomba experiment.

1- The Roomba experiment was			
☐Easy	☐Ok	☐Complex	
25%	**65%**	**10%**	
2- The Roomba project made me enhance my understanding of microcontroller/programming technologies and electronics			
☐Strongly Agree	☐Agree	☐Disagree	☐Strongly disagree
30%	**60%**	**10%**	**0%**
3- The time dedicated to complete the Roomba project was			
☐Too short	☐About right	☐Long	☐Too Long
0%	**50%**	**45%**	**5%**
4- The Roomba project should be continued in the future			
☐Strongly Agree	☐Agree	☐Disagree	☐Strongly disagree
19%	**75%**	**9%**	**7%**
5- Students should be given more time to complete the Roomba			
☐Strongly Agree	☐Agree	☐Disagree	☐Strongly disagree
40%	**40%**	**20%**	**0%**
6- I will probably take other classes covering Robotics			
☐Strongly Agree	☐Agree	☐Disagree	☐Strongly disagree
10%	**40%**	**30%**	**20%**
7- I feel this class helped me to better understand some laboratory aspects of my major classes			
☐Strongly Agree	☐Agree	☐Disagree	☐Strongly disagree
30%	**40%**	**20%**	**10%**
8- It was fun to collaborate with other students			
☐Strongly Agree	☐Agree	☐Disagree	☐Strongly disagree
30%	**50%**	**15%**	**5%**
9- Students were give sufficient instructions on using the hardware and software tools			
☐Strongly Agree	☐Agree	☐Disagree	☐Strongly disagree
5%	**25%**	**50%**	**20%**

We believe that early rigorous introduction of engineering concepts and projects to Engineering and Computer Science students can excel student performance in more advanced engineering courses, transforming them to more knowledgeable and experienced professionals. It is our intension to assess the effectiveness of this an similar experiments by gathering more data and observing overall student performance in more advanced courses.

5. Conclusion

In this paper we presented a practical Roomba experiment, which can be added to the first-year Introduction to Engineering course. Student responses to the quantitative and free-form questions concerning the implementation of the Roomba experiment indicated that majority of students found the experiment interesting and helpful in enhancing their learning. The complexity of the experiment appeared to be satisfactory to most students and, overall, their feedback was positive. Based on student feedback, we learned that the lecture time to complete some topics can be extended and students must be given more time to complete the assignments. This experiment also proved to be a good starting point for future development of courses in Robotics technologies.

References

1. Lahtinen, E., Ala-Mutka, K., Jarvinen, H-M. A Stduy of the Difficulties of Novice Programmers. ACM SIGCSE Bulletin, Vol. 37, Iss. 3, 2005.
2. Christopher R. Carroll, "Innovative 'HCS12 Microcontroller Lab Station Using Limited Lab Resources." Proceedings of the 2008 *ASEE* North Midwest Sectional Conference.
3. Lab manuals and complete course information can be obtained from www.sonoma.edu/users/f/farahman .
4. National Instruments. (2009) Data Logging and Supervisory Control with LabVIEW. Retrieved March 5, 2009, from http://www.ni.com/pdf/products/us/4msw60-61.pdf
5. iRobot® Roomba® Serial Command Interface (SCI) Specification The full ROI spec released by iRobot is available at: http://irobot.com/images/consumer/hacker/Roomba_ROI_Spec_Manual.pdf
6. Hacking Roomba: http://hackingroomba.com/?s=avr
7. Build your own Rootooth: http://hackingroomba.com/projects/build-a-roomba-bluetooth-adapter/comment-page-1/#comment-66964
8. Rootooth can be purchased from http://www.sparkfun.com/products/684
9. *RealTerm* Terminal Software: www.realterm.sourceforge.net/