120th ASEE Annual Conference & Exposition
FRANKLY, WE DO GIVE A D*MN
June 23-26, 2013

Paper ID #7170

Control Systems Design Course with a Focus for Applications in Mobile Robotics

Dr. Rocio Alba-Flores, Georgia Southern University

Rocio Alba-Flores received her M.S. and Ph.D. in Electrical Engineering from Tulane University. She is an Assistant Professor in the Department of Electrical Engineering at Georgia Southern University. Her main areas of interest include control systems, robotics, digital systems, microprocessors, and signal and image processing.

Dr. Fernando Rios-Gutierrez, Georgia Southern University

Fernando Rios-Gutierrez was born in Mexico City, Mexico. He graduated with a bachelor's degree in Electrical Engineering and Communications from the National Polytechnic Institute, Mexico City, in 1978. He was awarded a full scholarship to pursue graduate studies at the Electrical Engineering and Computer Sciences Department of Tulane University, New Orleans LA, where he obtained a M.S. degree and a Ph.D. in Electrical Engineering in 1998 and 2000 respectively. From 2002 -2007 he was an Assistant Professor at the Electrical and Computer Engineering department of the University of Minnesota Duluth. Since September 2007 he has been an Assistant Professor at the Electrical Engineering department at Georgia Southern University. His main research interests include robotics, remote sensing, learning techniques for robots, digital systems, and microprocessor applications.

Control Systems Design Course with a Focus for Applications in Mobile Robotics

Introduction

Currently, the majority of electrical engineering, mechanical engineering, and mechatronics programs offer in their curriculum a type of control systems course. Traditionally the focus of these control courses is on the theory of classical control. There are some programs that offer the course with a focus in applications and programming of microcontrollers. There are also programs where the emphasis of the course is on the study of instrumentation and programmable logic controllers.

One difficulty in teaching control systems is to provide a good balance between theory and practice. By incorporating a laboratory component, it could help to provide some connection between the abstract control theory and the real world applications.

In the present paper we describe the educational experience gained by including team-based projects into the control systems course. In these projects students design and implement different controllers for autonomous navigation in a mobile robot. In particular, the design and implementation of three main types of controllers are assigned to teams of students, namely: 1) a traditional PID controller, 2) a Fuzzy Logic based controller, and 3) an Artificial Neural Network based controller. The controllers are first simulated in Matlab, and then implemented in a Field Programmable Gate Arrays (FPGA) card. Using the implemented controllers in a mobile robot platform, students can compare and evaluate the performance of the different controllers in actual navigation situations.

Background

At our university, the Control Systems is a 3 credit course (2hr lecture, 3hr lab, 15 weeks) that is required for students in the EE program. Students tanking this course are seniors and graduate students. The course offers an introduction to classical control theory and applications, with emphasis on feedback systems. The prerequisite to this course is Linear Systems. Traditionally in our department, the lab component in this course has been focused only on simulation of control systems using Matlab software. Student's comments in the course evaluation and faculty assessment data indicate that students perceive this course as very theoretical and it is not easy to visualize the practical applications of the material they learn in class. Students' comments include adding hands-on experiments.

Although in the market exists a variety of educational lab equipment that could be used for control experiments (DSpace, Educational Control Products (ECP), Quanser, etc. [1]) and that can enhance the learning experience with hands-on activities, these are very expensive products. Even if one or two of these products could be purchased, each student would have a very limited opportunity to actually use the equipment. Therefore, the instructors decided to seek a different option that allowed the use of equipment and material already available in the EE department to develop some hands-on experience for the control system course. The topic of robotics always have interested and attracted the attention from students, and our department has a mobile robot laboratory with several mobile robot platforms, therefore the authors decided to use the mobile robot platforms to incorporate a 4-week design project where students had to design, simulate, and implement different controllers for autonomous navigation. Teams of three students were formed by the instructor. When possible, a graduate student was included in the team.

In the present paper, we describe the design and implementation of three different controllers that were developed by the students as part of the control systems final project. The project consisted in the design, simulation, and implementation of different controllers for autonomous navigation. Three types of controllers were assigned to the teams: 1) Traditional PID, 2) Fuzzy Logic (FL) based, and 3) Artificial Neural Networks (ANN) based.

The main objective of implementing these projects was to compare the performance of the PID, Fuzzy Logic, and Artificial Neural Network controllers when applied to control the navigation of an autonomous robot. Simulation models in MATLAB and physical hardware implementation in a FPGA card were designed and developed by each team of students. The controllers utilized the same mechanical platform without any modifications for testing the navigation of the mobile robot in an indoor unknown environment.

Course Setting

Students taking the control system course can be either, seniors in the electrical engineering program, or graduate students in the MS in Applied Engineering program. This setting allows the formation of teams with seniors and graduate students, allowing the development of more challenging projects.

The control course is an introduction to classical control theory and applications and is presented with emphasis on feedback and its properties including the concept of stability, stability margins, and the different tools that can be used to analyze control systems properties. The class meets two hours a week for the lecture and 3 hours a week for the lab. The prerequisite for this course is a linear systems course, therefore students already have knowledge about continuous and discrete system modeling, input-output description, block diagrams, stability, Laplace transform and system response. Because of this, faculty teaching this course can require students to develop more advanced projects.

To address the concerns from students and faculty about the lack of hands-on experience in this course, to make the course material more interesting to the students, the instructors teaching this course decided to incorporate team-based projects to provide students hands-on experiences and real-life applications. Because of the type of projects that students have to develop, they need to come up with solutions that are not found in traditional textbooks, therefore the development of the projects allowed the students to expand their self-learning and team-work skills.

Each team-based project consisted in the design, simulation, and implementation of a different type of controller for the control of autonomous navigation. These projects were developed during the last four weeks of the semester. The instructor dedicates a few lectures to introduce students to the material that they will need to develop the assigned projects. The lectures include principles of autonomous navigation, motor drivers, sensors, fuzzy logic, and neural networks. Reading material and homework related to these topics are assigned to the student to ensure they will gain the basic knowledge needed to successfully complete the project.

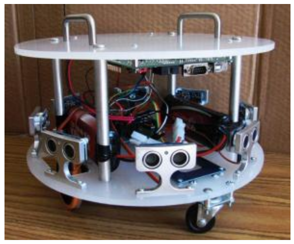
Students taking the control systems class have a solid background in digital electronics, microcontrollers, analog circuits, and Matlab programing. They have completed programming courses where they learned to program in Matlab and C language, and also completed a digital design lab where they learned about the design and implementation of digital systems and CAD tools including Altera Quartus II, VHDL, and FPGAs. Therefore, all the students have the necessary background to develop projects using Matlab and Quartus II. Students were provided with reading material that helped them to get familiar with autonomous navigation concepts [12,13]

Students worked in teams of 2-3 students, when possible one graduate student is included on each team. A typical class size has 10-15 undergraduate students, and 2-5 graduate students. Teams with only undergraduate students are assigned PID and fuzzy logic based controllers. Teams with graduate students are responsible of designing and implementing Neural Network based controllers.

Description of the projects

Robotic platform

The robotic platform used for the projects is shown in fig 1. It consisted of a round platform of 30cm of diameter with eight sonar sensors placed every 45^{o} degrees around the platform. A Field Programmable Gate Array (FPGA) card [2] is installed in the robot platform, as well as, radio frequency identification (RFID) tag reader [8] that is used to identify the goal the robot should find, and a Basic Stamp microcontroller is used for I/O processing.



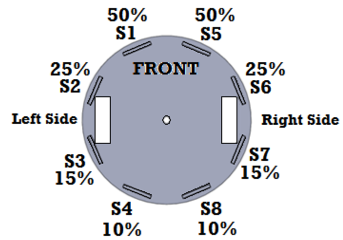


Figure 1. a) Mobile Robot Platform and b) Sonar Sensor Layout and Weights

The Challenge.

Design and implement a controller to be able to drive the robot autonomously in an unknown indoor environment (maze) avoiding static obstacles, and reaching a goal (an RFID tag) placed at an unspecified location.

The class size where this team-based project was included had twelve undergraduate students and two graduate students, and five teams were formed and the assigned projects were:
Team 1 & 2. Design, simulate, and implement a PID controller for the mobile robot.
Team 3 & 4. Design, simulate, and implement an intelligent controller using Fuzzy Logic (FL)Techniques.
Team 5: Design, simulate, and implement an intelligent controller using Artificial Neural Networks (AAN).

Each controller was first designed and simulated in Matlab using Simulink, Fuzzy Logic, and ANN toolboxes. Then the controllers were implemented and simulated in Quartus II software, and finally, the hardware design of each controller was implemented and downloaded to a FPGA card (Altera UP3-1C12) which was mounted onto the mobile robot platform. The response of each controller was tested in the same physical testing environment using a maze that the robot should navigate avoiding obstacles and reaching the desired goal. To evaluate the controllers' behavior each trial run was evaluated with a standardized rubric (Appendix A) based on the controllers' ability to react to situations presented within the trial run.

Sensors and Actuators

The main function of the controller was to collect data from the eight parallax PING sonar sensors [9] and a RFID Tag Reader, fuse these data, and based on results, communicate the required motor speed signals to the Sabertooth Dual 5A motor driver [10]. The motor driver generated the corresponding voltage level to each motor, needed to change the direction the mobile robot platform was traveling.

Each controller: PID, FL, and ANN were individually implemented onto the same mobile robot platform. This allowed for consistency to be able to compare the navigational abilities of each controller. Fig. 2 shows the block diagram of the implemented system.

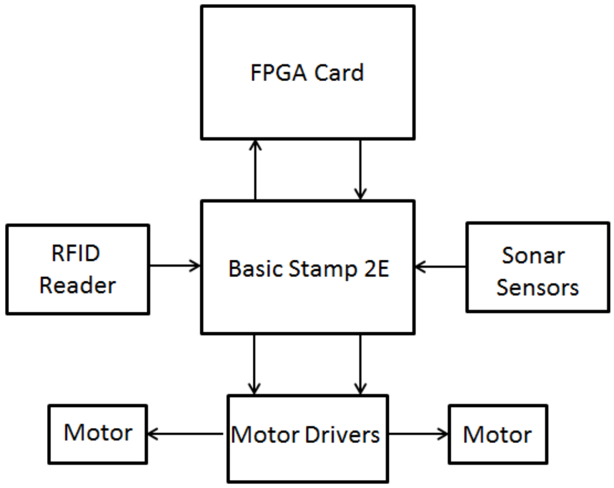


Figure 2. Mobile Robot Platform Hardware Architecture

The four sonar sensors on each half of the mobile robot platform are fused and weighted to minimize the number of inputs to each controller (fig. 1b). This allows one sonar sensor input value from the left side and one sonar sensor input value from the right side. The input value for the left side was calculated using eq. 1 (similar equation was used for the right side).

$$left\ side = 50\% * \left(\frac{Time\ in\ \mu s\ from\ S1}{\frac{29.033\mu s}{cm}} * 0.5 \right) + 25\% * \left(\frac{Time\ in\ \mu s\ from\ S2}{\frac{29.033\mu s}{cm}} * 0.5 \right) + 15\% * \left(\frac{Time\ in\ \mu s\ from\ S3}{\frac{29.033\mu s}{cm}} * 0.5 \right) + 10\% * \left(\frac{Time\ in\ \mu s\ from\ S4}{\frac{29.033\mu s}{cm}} * 0.5 \right) - -(eq. 1)$$

Model for the Mobile Robot Platform

The mobile robot platform was modeled as two identical plants, the two DC motors. The models were derived from the electromechanical representation of the DC motors as shown in fig. 3. The parameters used for the DC motor simulation were: armature resistance of 1Ω, armature inductance of 500mH, a motor inertia of 0.01Kg-m^2, and a gear ratio is 30:1[11]. The voltage source has a maximum voltage of 12VDC, and the output is measured in revolutions per minute (rpm).

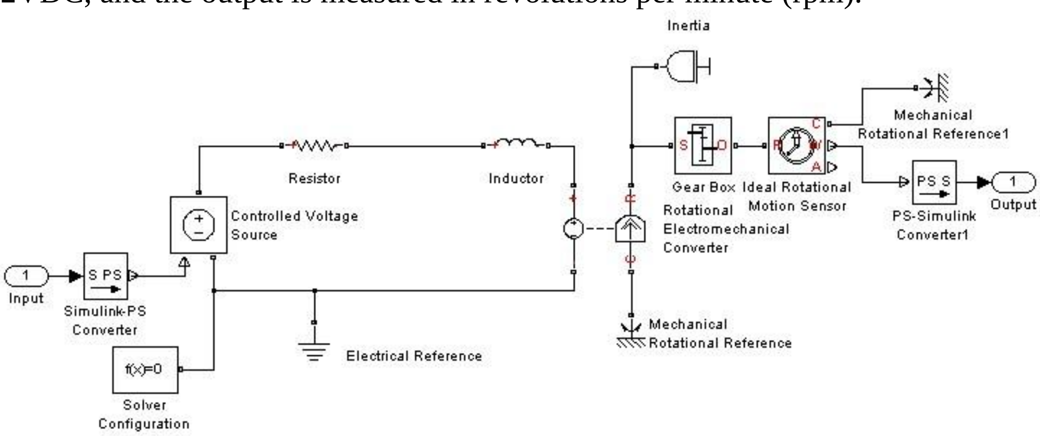


Figure 3. Electromechanical representation of DC motors

PID Controller

Fig. 4 shows the PID Simulink design. The controller design incorporated two plants which are the left DC motor and the right DC motor of the mobile robot platform. The set point of this specific system was 30cm. This set point was chosen as the most favorable distance for the mobile robot platform for navigation and obstacle avoidance. With this set point, the simulation requires the robot platform to react to a sensor value less than or equal to 30cm. A random number generator was used to simulate random sensor values between 0-30cm for both the left and right sides. After the PID controller calculates the corresponding output, one output is sent to the left DC motor, and the second output is sent to the right DC motor. A resultant change in the rpm of each motor corrects course navigation of the mobile robot platform.

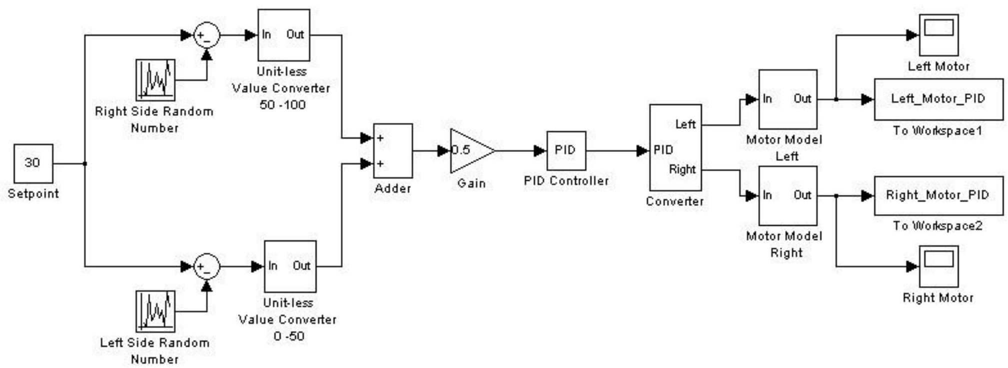


Figure 4. PID Simulink Design

The physical implementation of the PID controller was completed through hardware, designed and simulated in Quartus II software, and then downloaded to an FPGA card (Altera UP3-1C12). The FPGA system clock runs at a speed of 48 MHz.

Fuzzy Logic Controller

Simulink software was used to design and simulate the Fuzzy Logic (FL) controller. Fig. 5 shows the block diagram of this controller. The FL controller has three distinct sections, fuzzification, rule processing, and defuzzification. The fuzzification section handles the Membership Functions (MF) for the two inputs. The rule processing section compares the left and right fuzzy inputs to the rule base to produce fuzzy outputs. The defuzzification section uses the centroid method to produce a clear output.

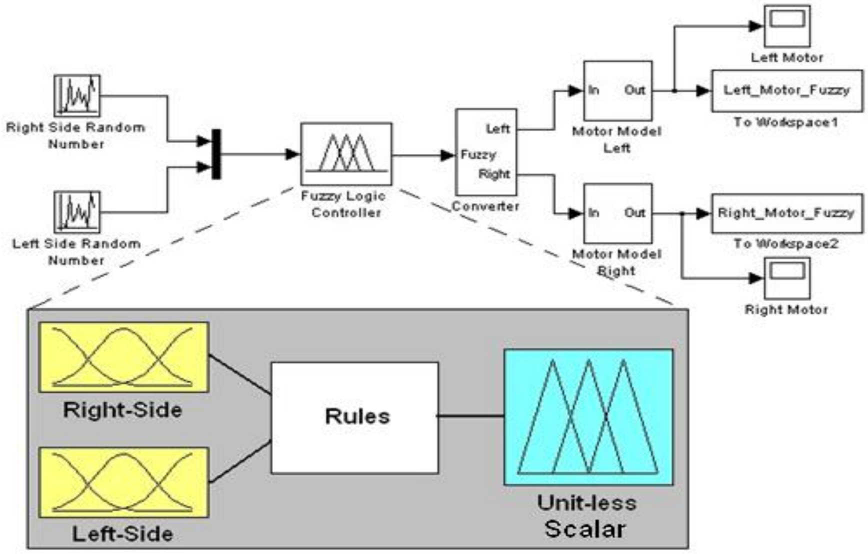


Figure 5. Fuzzy Logic Simulink Design

The simulation design begins with a random number generator to create sensor values in the range of 0-30 cm. This range is based on pre-planning for physical implementation. The simulation produces two of these inputs, one for the left side and one for the right side of the simulated system. Both values are propagated straight into the FL controller. The Fuzzy Logic controller contains two input membership functions (MF). The functions are identical, but one processes the left sensor inputs and the other processes the right sensor inputs. Five linguistic variables were used as: {VS, S, M, W, VW } denoting Very Strong , Strong , Medium, Weak, and Very Weak. An input falling within the VS membership category indicates and object is very close, and conversely an input within the VW membership category means an object is a safe distance away. Each end of the linguistic variable spectrum has a shoulder shaped MF. The middle linguistic variables are triangular shaped as shown in fig. 6.

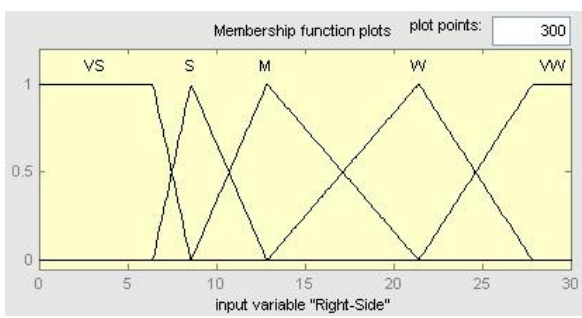


Figure 6. Fuzzy Logic Input Membership Functions

After each input produces five MF values as fuzzy inputs for the rule processing, they are referenced to the rule base. The number of rules is determined by both, the number of input membership functions and the number of linguistic variables per membership function. For this project the FL controller requires 25 rules to define the fuzzy outputs. All rules use an AND logic operation. The fuzzy outputs correspond to the linguistic variables in the output MF. The goal of this controller is to control the motors, so the output linguistic variables reference steering direction with Medium Left (ML), Slight Left (SL), Zero (ZR), Slight Right (SR), and Medium Right (MR). The rules can also be visualized in a rule matrix (fig. 7).

		Right Side MF				
		VS	S	M	W	VW
Left Side MF	VS	ZR	SR	SR	MR	MR
	S	SL	ZR	SR	MR	MR
	M	SL	SL	ZR	SR	SR
	W	ML	ML	SL	ZR	SR
	VW	ML	ML	SL	SL	ZR

Figure 7. Fuzzy Logic Controller Rule Matrix

The output MF shown in fig. 8 was used for defuzzification. Here the five fuzzy inputs from the rule processing coordinate to the Y-axis. The final distinct output that is propagated to the plant is determined by the centroid method. The corresponding value on the X-axis is a unit-less output processed by the remainder of the system.

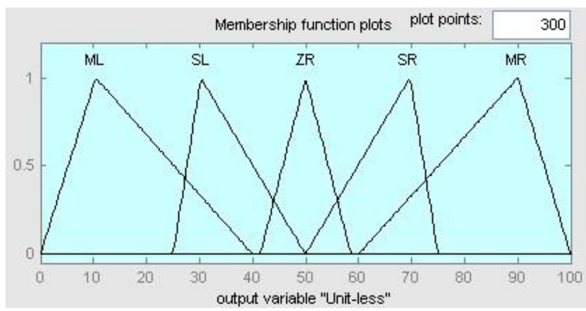


Figure 8. Fuzzy Logic Output Membership Function

The FL controller communicates with the BASIC Stamp 2e (input/output processing device). The BASIC Stamp starts the FL process by reading in the sensor values, weighting individual sensors (eq. 1). The two resultant values are sent to the FPGA card as equivalent millisecond pulse widths. These two resultant values are the inputs to the FPGA card. Each input value propagates trough five ROM blocks, where each ROM block represents an individual linguistic variable of the input MF. Within each block, the input value provides a MF input value.

ANN Controller

The ANN controller was first designed in Matlab using the Simulink and the ANN Toolboxes. This software was also used to simulate and train the controller. Fig. 9 shows the overview of the ANN design**.**

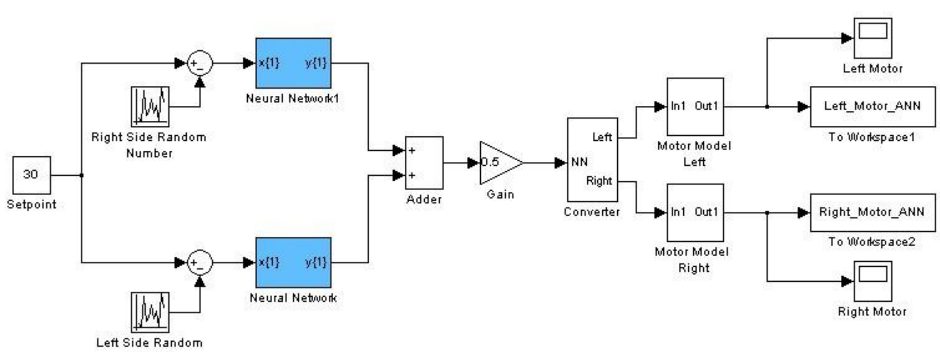


Figure 9. ANN Simulink Design

The set point is set to 30cm to be used during physical implementation and navigation. The system is divided into two sides. A random number generator simulates sensor values within a 0-30cm range for both the left and right sides. The difference between the setpoint and the random number (sensor values) generate the error. The error is the value propagated into the controller. This controller's architecture utilizes two identical ANNs; one for the left side and one for the right side. Both ANN's utilized within this controller had an identical structure, but produced scalar output values in different ranges (Fig.10).

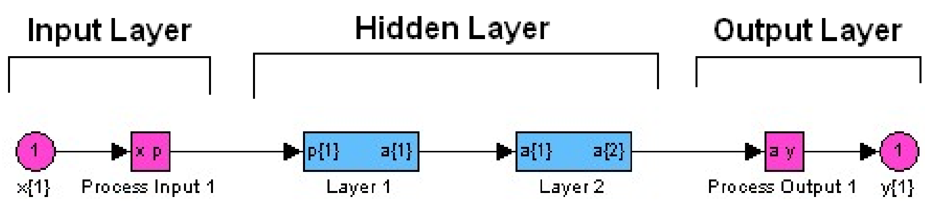


Figure 10. ANN Matlab Block Diagram (Left and Right)

The input layer collects data from a source. For this controller, the input data is the error calculated via the feedback. Each network has only a single neuron in the first layer because there is only one input data value per side. The two hidden layers provide the main computational processing (Fig. 11). The first hidden layer is made up of five neurons. The input to this layer is weighted and a bias is added. The connection weights are determined during training. The activation function used by the first hidden layer is a sigmoid. It determines the output activation to the second hidden layer. The second hidden layer has a single neuron and receives five inputs with weighted connections from the first hidden layer. The inputs are summed together by this single neuron. The activation function for the second hidden layer is linear and it determines the output activation to the output layer. The output layer processes a scalar value to propagate to the remainder of the system. After the controller interprets the data, the two ANN's produce a scalar value each that are averaged together.

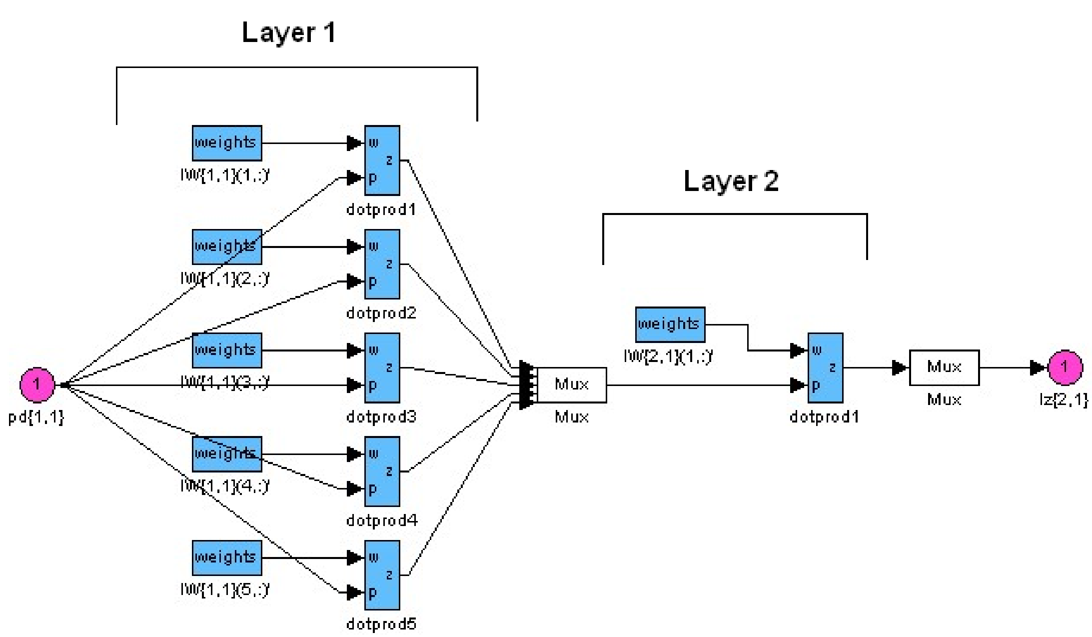


Figure 11. ANN Hidden Layers in Matlab Design

The BASIC Stamp code starts with the sensor values read in, individual sensors weighted, the left side sensor values combined into a single value representing the left side, and the right side sensor values combined into a single value representing the right side. The error is calculated for each the left and right side using the established set point of 30 cm and the sensor feedback values. The left side error and right side error are sent as two equivalent millisecond pulses to the FPGA card. The overview of the ANN controller implemented in hardware on the FPGA card is shown in fig. 12. It contains two input blocks to process the two input pulses (ms) from the BASIC Stamp, the controller block, and an output block. The left and right side error values are used to locate an address of the correct output response. The left and right side values are added together to produce an address location. This location value is converted from a 21 bit value to a 10 bit value before entering the ROM memory. Within the ROM memory a look up table is stored holding scalar value outputs. The ANN look-up table was created using Microsoft Excel to produce all possible combinations of inputs to the network versus all possible outputs.

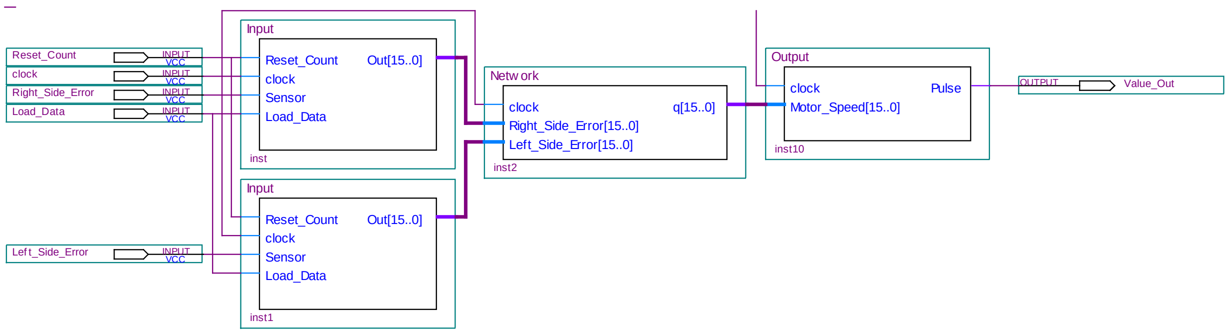
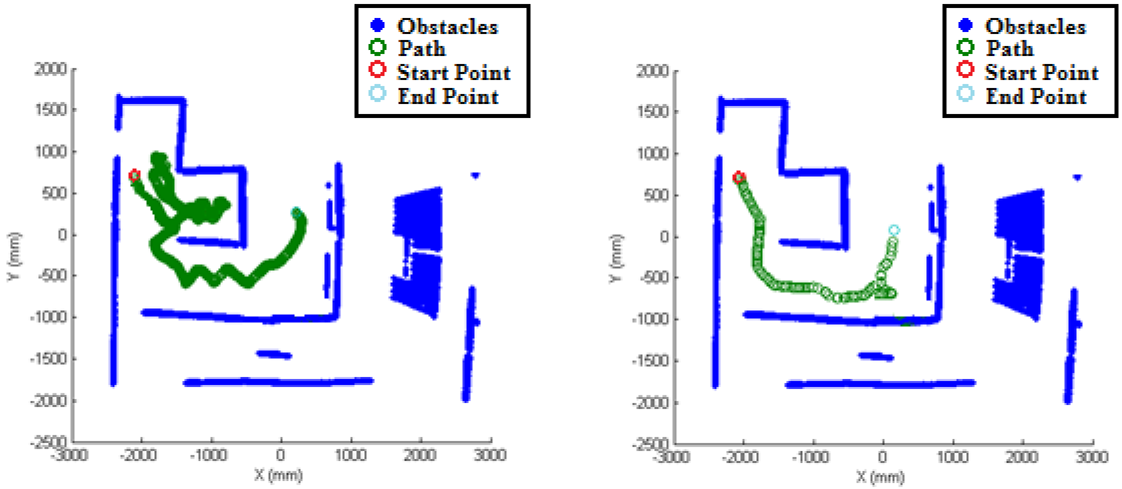


Figure 12. ANN Controller on FPGA

Controller Testing and Performance Evaluation

Each controller was independently implemented onto the FPGA card and mounted on the mobile robot platform. They were tested on their ability to deal with different situations in real time, in an unknown indoor environment. Appendix A shows the rubric used to evaluate the controller's ability to react to situations presented within the trial run. Each controller (PID, FL, and ANN) was tested in the same environmental layout and independently evaluated on how well they navigate on the environment. A "pass" indicates the controller navigated through the testing environment and found the RFID Tag goal in a timely manner. A "fail" indicates the controller either did not find the RFID Tag goal in a timely manner, or it was unable to successfully navigate the environment. Each controller was given a time limit of 5 minutes to complete the course and find the goal. If the controller is experiencing extreme difficulty within a situation, it is given a 60 second time limit before the run is terminated.

Fig. 13 shows the trajectory of the mobile robot traversed through the maze for two controllers, a PID and a ANN. The map was generated from a camera mounted on the ceiling. The results of both, the Matlab simulation and the actual implementation showed that the two intelligent controllers, ANN and FL, outperformed the PID controller. The ANN controller was marginally superior to the FL controller in overall navigation and intelligence.



a) PID controller b) ANN controller

Figure 13. Maze (goal seeking) scenario. a) PID controller, b) ANN controller

Course Assessment

The assessment for the team-based project component in the control system course was performed using direct and indirect assessment tools. Direct assessment included an oral presentation, a formal technical report, and the actual demonstration of the working system. Indirect assessment included student surveys at the end of the semester. Each team gave a 15 min oral presentation explaining their design and results to peers and faculty. All team members had to participate in the oral presentation, each team member expending about the same amount of time. The student survey was divided in three main parts: questions addressing team work experience, questions addressing level of difficulty of the assigned project, overall experience developing the project. Table 1 provides sample questions of student survey. The Likert Scale was used for the survey (5-strongly agree, 4-agree, 3-neutral, 2-disagree, and 1-strongly disagree).

Table 1. Sample questions of student survey

Question	**Avg. score**
The project was very interesting and increased my motivation in learning control systems	4.5
I felt a positive team work environment that helped me to learn	3.2
The amount of time I worked in the project was overwhelmed	4.2

Comments provided by the students in the survey included:

- Too much time expended in the project, but at the end it was worth
- Starting the project was very tedious and time consuming. The professor should provide better guidance at the beginning.
- Interesting and very challenging project. It would help if more classes can cover material related to the project.

Summary and Conclusions

Adding educational hardware equipment to include hands-on experiments in our current Control System course requires considerable financial resources. Currently our Control Systems course have an enrolment of approximately 20-15 students per semester, therefore a minimum of ten hardware stations would be required to support hand-on experiments. Since robot platforms and FPGA cards are already available in our department for other EE courses, instructors teaching the control system course, investigated a lower-cost alternative by incorporating a 4-week team-based project where students designed, simulated, and implemented a control system for autonomous navigation. Implementing three different mobile robot controllers, namely, PID, FL, and ANN, in the same platform provided students with a more clear understanding on how a control system operates on a particular plant. Also, by implementing different types of controllers they learned to evaluate the performance of each controller, and obtained a better sense of the characteristic of the different controllers. Overall, qualitative comments of students indicate that the addition of the team based-project have been a positive addition to the control systems course, that have allowed them to have a better understanding of what a control system is, and how it can be applied to the real world applications. Also the students found an extra motivation by trying to do better than their peers.

Throughout the process of designing and implementing the controllers, the teams encountered many problems and made some mistakes of their own and they had to be identified and acted on accordingly. The top challenge for students and faculty members was to manage the schedule of each team, so that they were all ready to be put together by the end of the course. Based on the time constraints that we had, along with a limited budget, the final result that we obtained were better than expected.

References

[1] Educational Control Products (ECP), http://www.ecpsystems.com

[2] Hamblen, J.O., Hall, T.S., and Furman, M.D., "Rapid Prototyping of Digital Systems, Quartus II Edition", Springer, 2006

[3] Parallax Inc., (2010). RFID card reader serial & usb(#28140/28340) (V.2.2), doi: http://www.parallax.com/Portals/0/Downloads/docs/prod/audiovis/28140-28340-RFIDreader-v2.2.pdf

[4] Antonini, P., Ippoliti, G., and Longhi, S., "Learning control of mobile robots using a multiprocessor system",. *Control Engineering Practice*, *14*, 2006, pp. 1279-1295.

[5] Gupta, V., Khare, K., and Singh, R.P., " Efficient FPGA design and implementation of digital PID controllers in Simulink", *International Journal of Recent Trends in Engineering*, 2(6), 2009, pp.147-150.

[6] Normey-Rico, J.E., Alcala, I., Gomez-Ortega, J., and Camacho, E., "Mobile robot path tracking using a robust PID controller", *Control Engineering Practice*, no. 9, 2001, pp. 1209-1214.

[7] Peri, V.M., and Simon, D. (2005), "Fuzzy logic control for an autonomous robot", *Proceedings of the Annual meeting of the North American fuzzy information processing society*, (pp. 337-342, 10.1109/NAFIPS.2005.1548558

[8] Parallax Inc., (2010). RFID card reader serial & usb(#28140/28340) (V.2.2), doi: http://www.parallax.com/Portals/0/Downloads/docs/prod/audiovis/28140-28340-RFIDreader-v2.2.pdf

[9] Parallax Inc. (2009). Ping))) ultrasonic distance sensor (#28015). Retrieved from http://www.parallax.com/Portals/0/Downloads/docs/prod/acc/28015-PING-v1.6.pdf

[10] Dimension Engineering, (2007). Sabertooth 2x5 user manual. Retrieved from http://www.dimensionengineering.com/datasheets/Sabertooth2x5.pdf.

[11] Lynxmotion Inc. (2010). Gear head motor. Retrieved from http://www.lynxmotion.com/images/data/ghm01.pdf

[12] Ronald C. Arkin, *Behavior-Based Robotics*, The MIT press, Cambridge, MA, 2000.

[13] D. Pack and S. Barrett, *Microcontroller Theory and Applications*, 2nd ed. Prentice Hall, 2008.

Appendix A
Situational Rubric for PID, ANN, and FL controllers navigating a real unknown indoor environment

Controller to be evaluated: **PID** **ANN** **FL**

Evaluator:

Situation	**4**	**3**	**2**	**1**	**Score**
Room	Enters situation fluidly, successfully navigates through without collisions or hesitations in a timely manner	Enters situation fluidly but may show hesitation and/or minimal collisions with delayed completion time	Enters situation with non-fluid movement, encounters multiple collision, completes situation in excessive time	Avoids situation or unable to successfully navigate through	
Corridor					
Small Object	Anticipates object and shows course redirection while fluidly moving around object collision free with no hesitation	Anticipates object and shows attempt at course redirection with non-fluid movement while remaining collision free	Collides with object while attempting to navigate around	Collides with object with no attempt to navigate around	
Large Object					
Overall Navigation	Traverse through testing environment fluidly, with no difficulty, and collision free	Increased effort in fluid movement with minimal collisions	Moves through environment with inconsistent movements (fluid and non-fluid), inconsistently collides with objects	Shows difficulty moving through the environment, and unable to avoid collision	
Overall Intelligence	Shows ability to adapt and learn, shows improved performance throughout run	Shows ability to anticipate and react to situations well before approach	Reacts to situation, but has inconsistent reaction upon approach	Shows no ability to react to situations	
				Total:	

Comments:

Time: ______________________ Pass: __________ Fail: ______________