AC 2012-5287: LABS APPROPRIATE FOR LECTURE-BASED INTRODUCTORY SYSTEMS AND CONTROLS CLASSES USING LEGO NXT AND LABVIEW

Greg N Droge, Georgia Institute of Technology
Dr. Bonnie Ferri, Georgia Institute of Technology

Bonnie Ferri received her B.S. degree from the University of Notre Dame in 1981 and her Ph.D. degree from Georgia Tech in 1988. She is currently a professor and Associate Chair for Graduate Studies in the School of Electrical and Computer Engineering. Ferri works in the general area of control theory.

JillL L. Auerbach, Georgia Institute of Technology

Distributed Laboratories: Control System Experiments with LabVIEW and the LEGO NXT Platform

Abstract:

This paper explores the inclusion of control system experiments into lecture-based introductory Systems and Controls courses. The experiments are implemented in two modes: as an in-class experiment and as a take-home project. The LEGO NXT kit with LabVIEW software is the platform. The experiment is supported by a website that includes a tutorial on the fundamental theoretical concepts, a video tutorial on the operation, and an online test representative of questions the students might be asked on an exam in the course. A discussion of the assessment methods for this laboratory module is included.

1. INTRODUCTION

Distributed laboratories contain experiments that can be done in various locations such as homes, classrooms, and dorm rooms. These labs utilize inexpensive equipment and student resources such as laptops and do not require the specialized equipment housed in centralized laboratory locations. As such, these experiments are well-suited for inclusion into lecture-based classes to be done at the desks in the class room or to be taken home as a project. These types of experiments allow for a new pedagogical model that promotes for more complete integration of theory and laboratory experience within the format of a standard lecture-based course[1-2].

To maximize the benefits of incorporating experiments into a lecture course, the laboratory module should have certain features. It should fully support or demonstrate a fundamental principle that is hard to understand from theory alone. The experiments should not necessarily require faculty to change their standard evaluation methods, such as in-class tests. To satisfy these needs, the laboratory modules should contain supplemental material including a tutorial for students on the fundamental concepts being taught and an online quiz for them that gives representative questions on the material that might be found on a standard exam.

To maximize the wide-spread usage of distributed laboratory modules, certain logistical considerations must be met. Essentially, each experimental module should be made as accessible as possible to as wide a range of instructors as possible. These experimental modules should be designed primarily for faculty who do not have resources for high-end experiments nor want to spend a lot of time developing, building or maintaining experiments. Furthermore, the hands-on demos and experiments must be easy for students to use without the need for a lengthy learning period.

A cohesive program to develop distributed laboratories with the above features exists that was funded by an NSF CCLI Phase 2 Grant, which supported the development of the TESSAL Center[3]. TESSAL (Teaching Enhancement via Small-Scale Affordable Labs) includes labs for signal processing[4], digital logic[5], power systems, electromagnetics, and control systems. The control systems modules are discussed in detail in this paper.

Control theory can be a highly abstract subject when taught as part of an introductory Systems and Controls course. For many students, hands-on experience would solidify concepts such as system identification, root locus and Bode plot design, and discretization. Many schools offer a

lab-based senior-level course in control systems while others only offer a control experiment as part of a larger, more broad-based lab. The labs discussed in this paper are not meant to replace these advanced labs but rather to introduce the laboratory experience into earlier, more theoretical courses.

In reference[4], we introduced the usage of the LEGO NXT platform for basic signals and systems experiments. The labs include modules on aliasing, chirp signals, digital filtering, modulation, system identification, motor control demonstration (no lab), and an early version of a motor control experiment. This paper discusses a fully new version of the Control Systems Module introduced in[4] by adding the web support, new projects and labs, and discusses the logistics and experience of offering these labs both in class while students are at their desks and as a take-home project. A website for this module is available.

Section 2 describes the experimental platform that we used to perform the experiment. In Sections 3 and 4, we discuss in detail two projects for this setup: a motor velocity controller and a position controller. We then proceed with the logistics of completing the projects in Section 5, the website support in Section 6, and a discussion of the assessment methods in Section 7.

2. EXPERIMENTAL SETUP

Both LabVIEW and NXC, a C-like language developed for the LEGO NXT kit, have been used for student experiments in our program[4]. It is assumed that students have had some computer programming experience but have not used C or LabVIEW in the past. A main problem with NXC is the integer arithmetic, which corresponds to the lack of a floating point processor on the NXT processor. Students spend a lot of time debugging the system due to integer overflow and syntax errors. LabVIEW, on the other hand, provides many benefits to the text-based languages, especially for students who have little experience with programming. Students can use floating point arithmetic in LabVIEW since the conversion to integer arithmetic is done automatically at a lower level of abstraction. In addition, the graphical user interface allows them to bypass the low level syntax errors common in C. As a result, students enjoy using the LabVIEW more than NXC and spend more of their effort on the control algorithm and less on programming.

In this paper, we use the LabVIEW NXT Module which is a free toolkit available to any student with access to LabVIEW. It allows for the programming and use of the LEGO NXT in two ways: the program runs independently on the NXT or in tethered operation where the NXT is connected via a USB cable to the laptop while the program executes. The program can run either on the PC or on the NXT. Tethered operation gives quicker feedback to the students since the measured data can be plotted as the experiment is running. This makes it easy to run the program while at the computer and extract the data very quickly for analysis.

There are two options for obtaining the software for running the experiment:
1) Either purchase a student version of LabVIEW or download a free 30 day evaluation version http://www.ni.com/trylabview/. Then download the free LEGO MINDSTORMS NXT Module to add onto an existing version of LabVIEW
https://lumen.ni.com/nicif/us/nilabsnxtmodule/content.xhtml
2) Purchase LabVIEW for LEGO MINDSTORMS which is a version of LabVIEW specifically created, marketed, and sold for use ONLY with LEGO MINDSTORMS. It is actually sold through LEGO Education and not NI, but you can download a free 30 day evaluation from here: https://lumen.ni.com/nicif/us/evallegotoolkit/content.xhtml

Two projects are described in this paper, both based on motor control. The first is a motor velocity controller and the second is a position controller for a LEGO cart. The LEGO NXT motor assembly includes an encoder for measuring its angular position. The velocity is estimated from a filtered rate change of the position. The basic analog control architecture for the velocity controller is shown in Figure 1. The position control experiment has the same architecture with the exception that the derivative filter block used to extract velocity from position is removed.
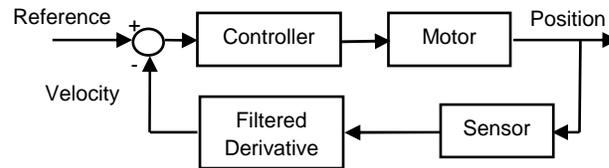

Figure 1: Motor Velocity Control Architecture.

A significant advantage of using the LabVIEW environment is the graphical user interface. The basic programming structure in LabVIEW is called a virtual instrument (VI). We used two VIs in our experiment. The first one, shown in Figure 2, is used to download and run the control program on the NXT. The second one, shown in Figure 3, is used to upload and graph the data that was saved to the NXT during operation. The interface for students is very simple. They have several options in terms of the reference signal and frequency, and to operate the motor open loop or closed loop (by toggling the PID button). Students are able to leave the NXT connected to the computer through the USB cable while they run the control code. Once the NXT program is done, they simply have to run the second VI, which plots the desired velocity and the measured velocity in one plot and the control effort in another. These plots allow the student to zoom in or out at any point very easily to make the necessary observations and calculations. Examples of these plots are shown throughout this paper.

Pedagogically, we feel that it is important for students to understand the program behind the graphical user interface to see how a control system is actually implemented. The LabVIEW programming environment is based on a dataflow structure, so blocks similar to the ones shown in Figure 1 can be wired together to form the block diagram that is behind the graphical user interface shown in Figure 2. An example of the LabVIEW control block is shown in Figure 4. The inputs to the control block in Figure 4 are the reference velocity, the calculated velocity, time, PID gains, and previous values of variables needed for the recursion. The outputs are the motor command and variables to be saved for plotting and the next iteration of the loop. In the full VI for this implementation there are, of course, more wires and blocks that correspond to input and output operations, data storing for the recursion and for data logging, velocity calculation, and timing.
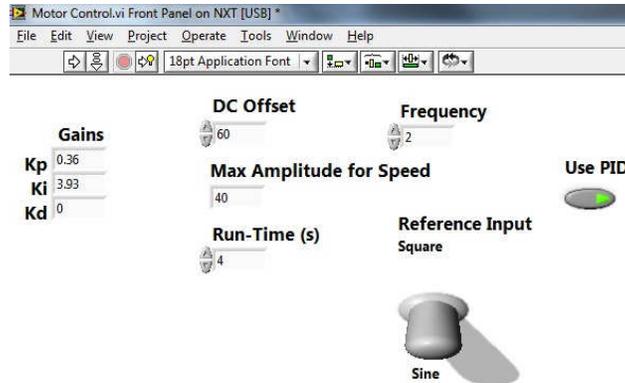
Figure 2: Screen shot of the VI used to control the LEGO NXT

In the motor velocity controller experiment, students must simply use the graphical user interfaces to run the existing VIs. The position control experiment requires students to modify the velocity control program in two ways. First, we have the students walk through a step by step procedure to change the velocity control to a position control, which allows them experience with the environment. Like converting Figure 1 to position control, the only thing that the students have to do is remove the velocity filter and connect the position measurement to the PID controller. We also have the students go into the PID controller block and change it to a lead control. This gives them the opportunity to replace their "black-box" experience with knowledge of what is "under the hood" of a control system.
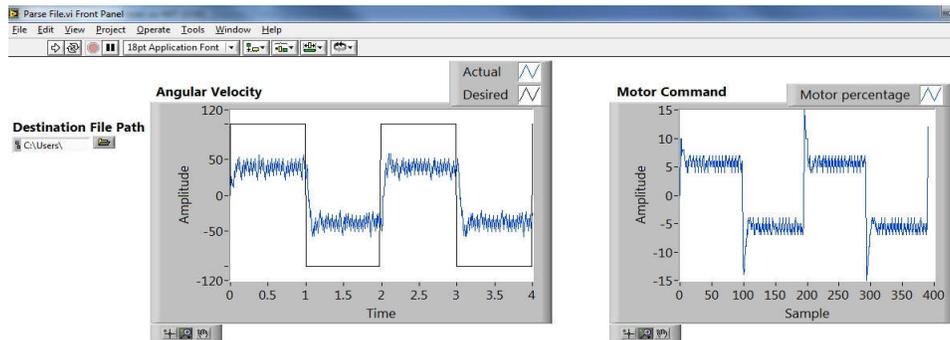

Figure 3: Screen shot of the VI used to read and plot data from the LEGO NXT.

3. VELOCITY CONTROL EXPERIMENTS

The velocity control project is divided into an in-class component and an out-of-class component. The experimental platform consists of the NXT processor connected to the NXT motor assembly as shown in Figure 5.

The features specific to this experiment include

- Hardware: LEGO NXT processor, 1 motor, student laptop
- Software: LabVIEW, NXT Toolkit Add-on, instructor-supplied VIs
- Fundamental Concepts: PID Control, Zeigler-Nichols tuning rules, steady-state error, sinusoidal tracking, system performance with different controllers, frequency response, controller bandwidth, resonance, system identification

In-Class Component: The in-class experiment can be given at a time in the semester when PID control is first introduced. It is based on the Zeigler-Nichols tuning rules[2], so it does not require a model of the system nor does it use more advanced design methods such as root locus or Bode plot design. This experiment provides the students the opportunity to see how well the Zeigler-Nichols tuning rules work on an actual piece of hardware. Having students work the experiment in class gives an opportunity to introduce them to the LabVIEW environment and allows students to get accustomed to the setup.

The response of the system with a square wave velocity reference signal of amplitude 100 is first observed with the proportional gain (K_P) set to 0.1, and the integral and derivative gains (K_I and K_D respectively) both set to zero.

In Figure 6, you can see that the system is stable but has an undesirable steady-state error. The students are asked to measure the steady-state error of the system by estimating the average value of the noisy signal and comparing it to the desired value.
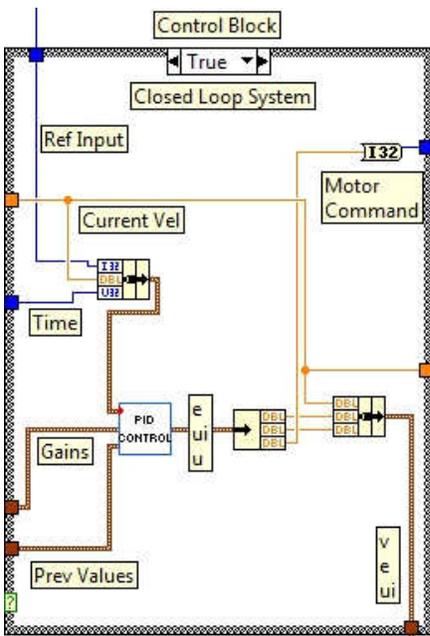

Figure 4. Control block in LabVIEW VI for the velocity.

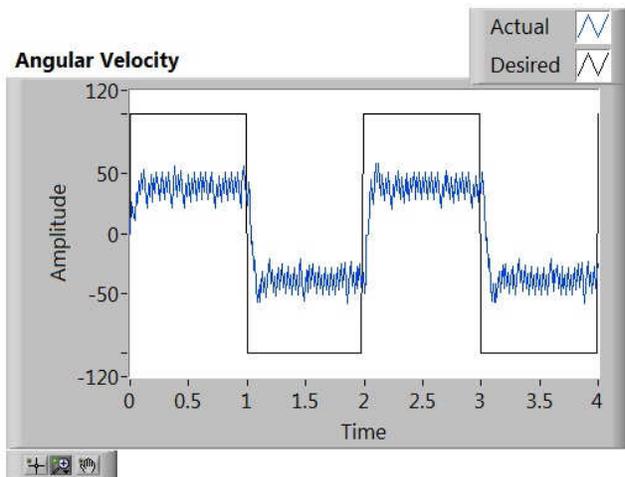Figure 5: Experimental platform consisting of an NXT and a motor.

Figure 6: Screen shot of the closed-loop system response to a square wave reference with amplitude 100, KP = 0.1 and KI = KD = 0.
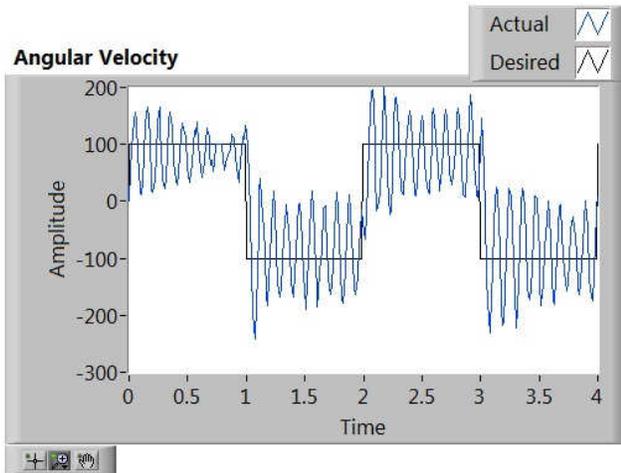
Figure 7: System response at KP = K_u = 0.8

The students are then prompted to follow the Zeigler-Nichols process for tuning the gains. The tuning rules assume a time delay in the system, which is present due to computational delays and the effect of the zero-order-hold. The process requires increasing the proportional gain until the system undergoes sustained oscillations, as shown in Figure 7 with the corresponding gain of K_u = 0.8, and to determine the period of the oscillations. The desired P, PI, and PID gains are then determined with the formulas given in[8]. Students are asked to plot the results of the three controllers, as shown in Figure 8, and to compare the performance of the controllers in terms of steady-state error, rise time, and noise as well as the maximum control effort required. It can be seen that the steady-state error is zero when there is an integral term included. The noise between the system responses seems to be comparable, but plots of the control effort (not shown in this paper) demonstrate to the student that the average control effort is the largest for the PID control and that the noise is amplified in that control signal due to the derivative term.
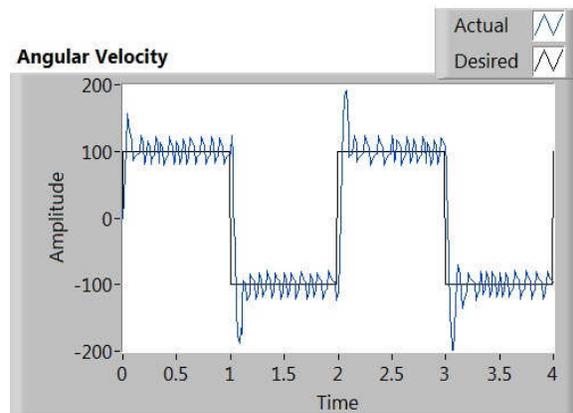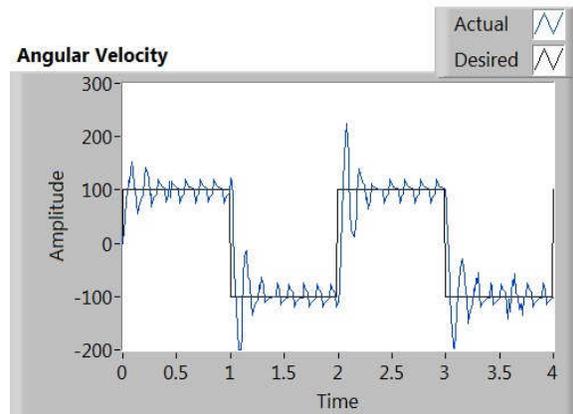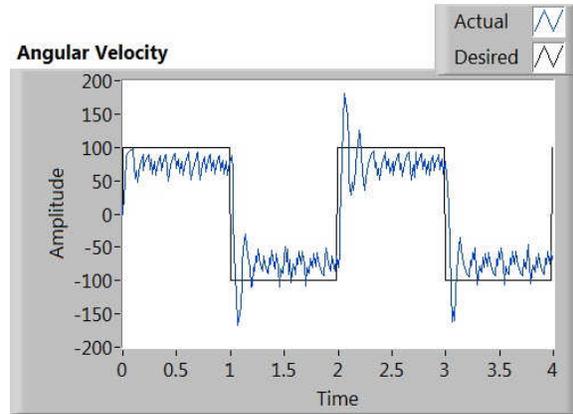

Figure 8: Screen shots for the closed –loop system response. Rows correspond to P, PI, and PID controllers from top to bottom.

The in-class portion of the experiment was quite easy and could be performed in a 50-minute class period. The students adjusted to the user-interface very quickly. There were only two problems encountered worthy of note. The first was that students had a difficult time interpreting the idea of sustained oscillations. Many of them initially ramped up the proportional gain past the marginal stability point and hit a limit cycle due to the saturation. The other difficulty that we experienced was when we modified the LabVIEW VI to make it more versatile, we added additional local variables. The memory management of additional variables increased the computation time, thereby changing the values of the Ziegler-Nichols gains. Purposeful changes in the delay can be used to modify the experimental setup from semester to semester so that students cannot use previous student results.

Take-Home Project: The second part of the project, done at home, requires students to measure sinusoidal responses of both the open loop and the closed-loop motor at different frequencies and then to plot the frequency responses. This part of the project could be done in class, except that it does take time to gather all of the data.

For identification purposes, the input signal to the closed-loop system is defined as the reference velocity while the input signal to the open loop system is the motor command. These signals are automatically routed in the LabVIEW VI when the students select the PID on/off toggle button on the front panel shown in Figure 2. The students obtain data for the closed-loop systems using the final PI and PID controllers designed in the in-class portion of the experiment. They obtain the results by using the LabVIEW interface to input sinusoidal inputs with different frequencies into the LEGO NXT and then measure the corresponding output amplitudes and phases using the plotting tools of the VI shown in Figure 3. In order to avoid the deadzone around zero velocity, students are asked to include a DC offset so that the velocity always remains positive. Students can calculate both the magnitude and phase of the system from the plots at each of the frequencies.

After obtaining the data, students are instructed to plot the frequency responses in linear scales for all three systems. A typical response for each of the systems is shown in Figures 9 through 11. The linear scales are easier to use for distinguishing the damping and bandwidth differences between the plots. As can be seen, the PID has the highest bandwidth, which is achievable with slightly better damping than that of the PI controller. It should be noted, again, that Bode plot and root locus methods are not used for this design, only the Ziegler-Nichols tuning rules, which yields low damping.

Students must determine the bandwidth of the PI and PID controllers. They are then asked to record the response for both controllers to a sinusoidal reference signal at a frequency of 1/4 the bandwidth and repeat for twice the bandwidth frequency. This enables them to verify that the controller tracks well at ¼ of the bandwidth frequency and to see the phase lag and reduced amplitude at twice the bandwidth.

The students are also required to determine the transfer function of the open-loop system from their data by using Bode scales as shown in Figure 12. The filtered derivative used to compute the velocity from the position was set at
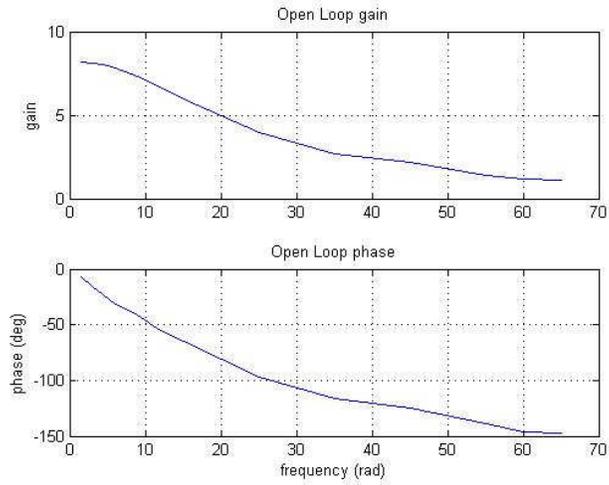
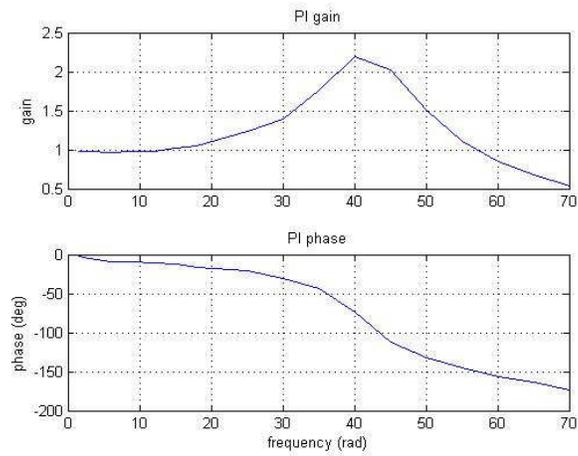Figure 9: Open Loop magnitude and phase plots in linear scales

Figure 10: Closed-Loop magnitude and phase plots in linear scales using a PI control
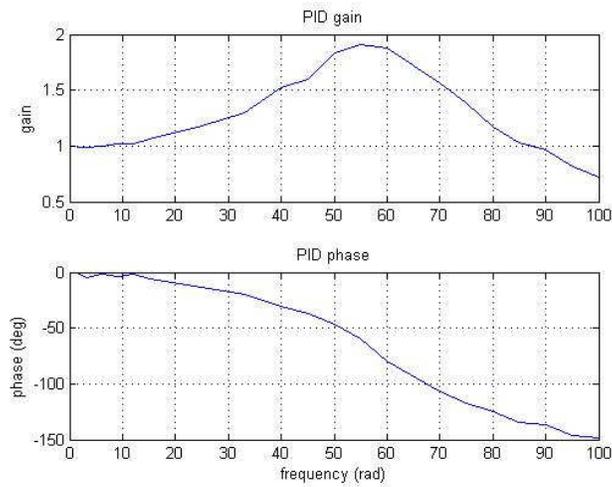

Figure 11: Closed Loop magnitude and phase plots in linear scales using a PID control

30 rad/sec, so the combined open-loop transfer function has the form:

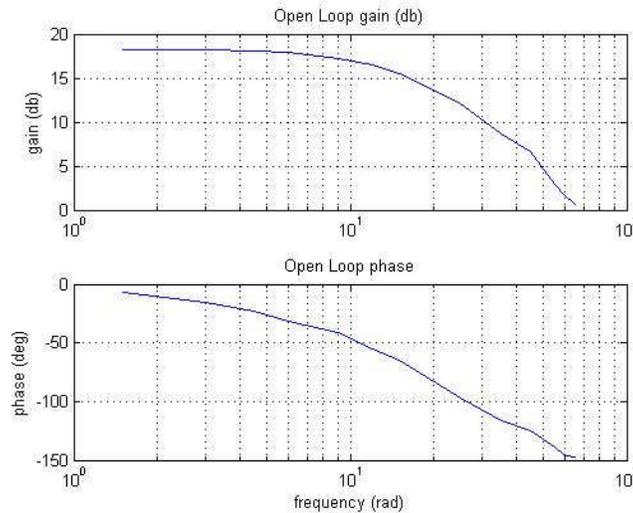$$\frac{V(s)}{U(s)} = \frac{30A}{(s+b)(s+30)}$$

Figure 12: Open-loop Bode plot of the LEGO NXT motor

The take-home section of the experiment allows students to see the behavior of the PI and PID controls that they created in the previous part of the experiment. They are able to apply concepts learned in class to see the frequency response of different systems and use the frequency response to create a Bode plot from which they are able to derive a transfer function for the system. The main difficulty for the students in this section is caused by noise from both the sensor and the velocity calculation. Although the noise makes it more difficult to determine the exact amplitude and phase shift of the output, it does give the students experience with the real physical data.

The velocity control project listed above can be extended or a separate project given for control design using root locus or Bode plot methods. The students would start with the model that they derived from the frequency response.

4. POSITION CONTROL EXPERIMENTS

The position control experiment is a modified version of an experiment first reported in[6] and later in[5]. The version discussed in this paper uses the LabVIEW VI's shown in Figures 2 and 3 rather than NXC as was used in the previous versions of the experiment. As in the case of the velocity control experiments, the advantage is that LabVIEW allows students the ability to avoid low level syntax errors and signal overflow due to poor integer arithmetic scaling. This experiment shares the same basic platform as the velocity control experiment with the exception of the motor being mounted on a cart, which provides some model uncertainty due to the resistance of the wheels on different surfaces. Thus the features are

- Hardware: LEGO cart[4].
- Software: LabVIEW with instructor-supplied VIs

- Fundamental Concepts: Time domain specifications, system identification, lead control, root locus, discretization, difference equation implementation, robustness

Students are first instructed how to modify the motor velocity controller LabVIEW VI to replace the velocity feedback with the measured angle. Step-by-step instructions allow them to become more familiar with the dataflow connections used in LabVIEW.

To obtain the model from the step response, it is necessary to close the loop first with a proportional controller. The students can use the closed-loop step response to determine the open-loop system parameters. Students can compare the poles found from this procedure to the poles found from the velocity control experiment model ID.

Students must design a lead compensator to achieve a settling time of about 0.3 seconds with negligible overshoot. They must discretize their controller using whatever method they choose (most use the bilinear transformation), and implement it in LabVIEW. LabVIEW does have control blocks as options, but this project purposely does not use them. The desire is for students to understand that a digital controller is implemented as a difference equation. This concept is obvious to instructors, but even graduate controls students do not fully understand this concept if the only control experience that they have had is with simulation software. Our goal is to replace the "black-box" experience with a deeper understanding. The project encourages students to look at the PID code block and/or the velocity filter code block and to use this as a sample for building the lead compensator.

Modifying the code was not very difficult for most students due to the graphical nature. The difference equation that implements a lead controller has the following form:

$$u[n]=0.7699u[n-1]-1.658e[n-1]+1.8841e[n]$$

The graphical representation of this equation is shown in Figure 13. The output cluster is used to store u[n] and e[n] for the next iteration and to output the command to the motor. Students are instructed to show all design steps, iterations, simulated step response, and actual step response. They must show the response for the cart running with the wheels in the air and on the ground in order to explore the robustness of their controller.

The step response to our controller is shown in Figure 14 for a reference position of 30°. Note that the controller meets the desired specifications on settling time and overshoot and also that, as expected, the position does not quite reach the reference value due to the dead zone.
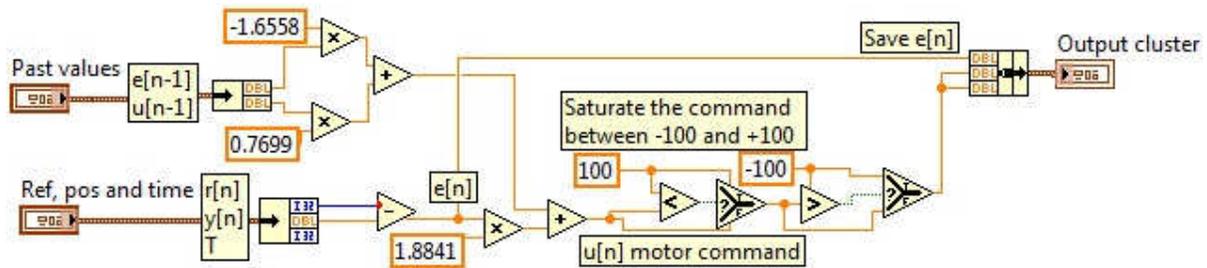
Figure 13: LabVIEW implementation of a difference equation that corresponds to a lead controller. Included in the lower right is a saturation operation.
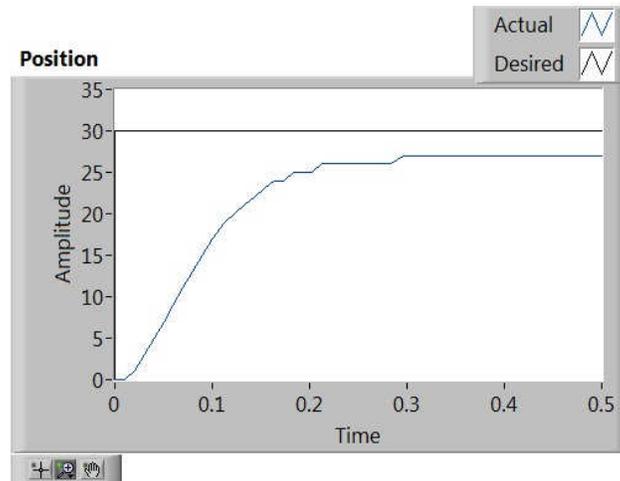

Figure 14: Position Step Response of the LEGO NXT

Students are also required to plot the frequency response of the ideal closed-loop system, that is, the closed-loop system based on their model and the analog controller. They are asked to find the bandwidth of the system and to run the system with a reference signal of a sine wave at 1/4 the bandwidth and at twice the bandwidth and compare the results to the expected results obtained from the Bode plot of the system. They are asked to evaluate both the phase delay and the amplitude to verify if the system is performing as expected.

5. LOGISTICS

There is a student version of LabVIEW that students can purchase if they would be using LabVIEW in other classes. For the recent offering of the controls course at Georgia Tech, students downloaded the 30-day evaluation copy of LabVIEW for free. The projects were scheduled to be completed during that period. Many universities have site licenses for LabVIEW, making this experiment easier to perform under licensed conditions. To accommodate the entire class with the available NXTs, we required students to work in groups, and we had to implement a check-out procedure. We allowed the students to check-out the NXT for half-day periods to work on the project (either 9am-4pm or 4pm until 9am the next day, with a reservation system in place). The group size for the in-class portion was 4-5 while the group size for the take-home experiments was 2. This allows us to use 8-10 NXT kits for a class size of 40-50.

6. WEB SUPPORT

The website for the Control System Module, on the TESSAL site, includes downloadable LabVIEW VIs, the project write-ups in MSWord documents (for instructors to modify), a brief tutorial on the fundamental concepts used in the experiment, a video tutorial on the LabVIEW VIs, LabVIEW help for the project, and a set of online test problems (written in FLASH) that reinforce student understanding of fundamental concepts. These test problems are representative of test problems that would be given in a lecture-based course.

7. ASSESSMENT

The objective of the experiments is to enhance student learning of theoretical material through experimentation. In a junior level Systems and Controls course, data has been collected from six classes. Among the six is one pair of a control and an experimental class, both taught by the same instructor. The portable lab experiments that were used were the light sensor project and the motor velocity position and control project. Several observations can be made. Anecdotally, it was found that requiring students to use Bode plots and root locus in the experiment forced them to understand the concepts on a much deeper level. Without the experiments, mediocre or poorer students tend to make a weak attempt at homework problems that use these concepts. With the experiment, especially the demonstration aspect, students have to make the motor work properly in closed-loop response. It forces them to learn the concepts.

To assess the learning objective on a more rigorous level, two assessment tools were used. First, a Concepts Inventory style of test was used, which provides an objective testing method for quantifying students' knowledge of specific topics. The Systems and Control Concepts Inventory Test used here to measure the impact of the experiments includes twenty-five multiple choice questions that address fundamental concepts covered in the course and includes several questions from the Signals and Systems Concepts inventory[8], both continuous-time and discrete-time. A portion of the questions were based on Kent Lundberg's work on a controls concept inventory. Additional questions to test basic control design concepts were drafted and reviewed by faculty teaching in that field. These questions were written in the style of the Signals and Systems Concepts inventories to be conceptual not computational in nature and were on the topics of root locus, Bode plot design, implicit digital control design, Nyquist plots, and PID control.

The Systems and Controls Concepts Inventory was administered in the two classes taught by the same instructor during two consecutive semesters taught by the same instructor. In each class, the test was given at the beginning of the course and again at the end of the course. The Control System Module was used in only one course (the experimental group) and the other course served as the control group. Some of the questions are on material related to the concepts supported by the experiment and some questions are on theoretical topics not related to the experiment. One measure of success is if higher percentages of students perform correctly on the questions related to the experiment as compared to the performance of students in the control class. The preliminary findings from this analysis are presented in Table 1 below.

Table 1			
Systems and Controls Concepts Inventory Test			
% correct comparison / experimental & control classes			
Concept Inventory Questions <u>Directly</u> Related to Lab			
Brief Description of Question	% correct experimental N=30	% correct control N=28	difference
Q A: identify a difference equation corresponding to a transfer function	63.0%	57.1%	+5.9
Q B: select the z-domain pole-zero plots corresponding to a discretized system	11.1%	10.7%	+.4
Q C: determine the transfer function of a digital filter corresponding to a discrete time system	63.0%	46.4%	+16.6
Q D: identify the purpose of a PD controller	74.1%	57.1%	+17.0
Q E: identify the purpose of a PI controller	81.5%	53.6%	+27.9

Another way to look at this data is to consider performance on both the pre-test and post-test and the percent gains for each student cohort. This data is depicted in Table 2, which reveals that on every question related to the Systems and Control module, a higher percentage of students in the experimental class performed better on the post test as compared to the control class.

Table 2		
Systems and Controls Concepts Inventory Test		
% change pre- post- tests / experimental & control classes		
Concept Inventory Questions <u>Directly</u> Related to Lab		
Questions	% change experimental	% change control
QA: identify a difference equation corresponding to a transfer function	+28.6%	+17.1%
Q B: select the z-domain pole-zero plots corresponding to a discretized system	+11.1%	-2.6%
Q C: determine the transfer function of a digital filter corresponding to a discrete time system	+41.1%	+16.4%
Q D: identify the purpose of a PD controller	+58.5%	+43.8%
Q E: identify the purpose of a PI controller	+69.0%	+36.9%

The other assessment tool is a pre-experiment survey and a post-experiment survey. Details of the assessment tools being used for the Control Systems Module are given in reference [9].

While the students in the experimental section reported higher levels of understanding the material covered by the experiments compared to the control class, follow-up surveys administered one semester later support the lasting impact of using the experiments. From the

control class, 21% of students took other Systems and Controls class and 33% did so from the experimental class. More compelling however is that only 29% of students from the control class said that their interest in applications of control engineering had increased since taking the course, yet 62% from the experimental class reported this gain.

On this same survey administered one semester after the Systems and Controls class was completed, students were given a list of nine In the experimental class, portable labs were integrated into the course that covered 9 of the topics listed in Table 3. For several of the topics, the percentages of "solid understanding" students are relatively close for each of the two classes. On other topics, such as root locus methods, discretization of continuous-time systems, discrete time control systems and to a lesser extent implementation of digital filters, the percentage of students reporting "solid understanding" is considerably higher. This pattern is reversed for one of the two topics, Nyquist stability criterion, that was not addressed by the portable labs used in the experimental class. This is noteworthy for two reasons. First, one argument against the integration of experiments in lecture classes is that it can be too time-consuming and can dilute coverage of other topics. Second, since the differences for several of the topics are so large in terms of the percentage of students indicating "solid understanding" from the experimental class, this data may suggest that active learning may increase the likelihood that students will retain the information.

Table 3
Follow-Up Survey for System and Controls Students from Previous Semester
Percentage of Students That Rated "Solid Understanding" for 11 Topics Covered in the Course

	Control	Experimental
Implementation of digital filters	29%	48%
Transient response of 1st and 2nd order systems	71	67
Steady state response of systems	71	71
Root locus methods	36	52
Frequency response methods	43	48
Routh-Hurwitz stability criterion*	36	38
Nyquist stability criterion*	36	10
PID controllers	29	43
Lead and lag controllers	36	38
Discretization of continuous-time systems	14	62
Discrete time control systems	14	43

*Topics not covered by portable lab experiments

8. CONCLUSION

Distributed experiments that are integrated into lecture-based courses, such as those in the TESSAL Center, have a large potential for improving student learning of theoretical material. The Control System Module allows students to gain experience with the theory that they have learned in class while avoiding the common difficulties associated with embedded programming. The LabVIEW interface allows the experiment to be abstracted to the point of letting the students see the structure of the system through the graphical programming inherent in LabVIEW. The position control experiment requires students to modify the code in such a way

that they gain practical experience in the implementation of a control system. The sample test questions on the website relate the experiment back to standard test questions that might be asked in a lecture-based course. Finally, the web support was designed with the goal of lowering the threshold for instructors to adopt these experiments into their courses, especially instructors who do not typically use experiments.

REFERENCES

[1] B. Ferri, J. Auerbach, J. Jackson, J. Michaels, D. Williams, "A Program For Distributed Laboratories In The ECE Curriculum," the 2008 *ASEE Frontiers in Education Conference*.

[2] B. Ferri and J. Auerbach, "Work in Progress - A Program to Incorporate Portable Labs Into Lecture-Based Electrical and Computer Engineering Courses," *Proceedings of the 2010 Frontiers in Education Conference, Oct. 2010, Washington DC.*

[3] TESSAL Center (Teaching Enhancement through Small Scale Affordable Labs) www.ece.gatech.edu/research/tessal/index.html

[4] B.Ferri, S. Ahmed, J. Michaels, E. Dean, C. Garyet, S. Shearman, "Signal processing experiments with the LEGO MINDSTORMS NXT kit for use in signals and systems courses," *American Control Conference,* pp.3787-3792, St. Louis, June 2009.

[5] B. Ferri, J. Auerbach, Hongyi Qu, "Distributed Laboratories: A State Machine Experiment," *proceedings 2010 International Conference on Frontiers inEducation: Computer Science and Computer Engineering (FECS'10).* Las Vegas, July 2010.

[6] B. S. Heck, N.S. Clements, and A.A. Ferri, "A LEGO Experiment for Embedded Control System Design," *IEEE Control Systems Magazine*, vol. 24, no. 5, October 2004, pp. 61-64.

[7] G.F. Franklin, J. Powell, (2001*). Feedback control of dynamic systems.* Array Upper Saddle River, N.J.: Prentice Hall PTR.

[8] K. Wage, J. Buck, T. Welch, C. Wright, "The Signals and Systems Concept Inventory", *2002 Proceedings of the ASEE Conference*.

[9] J. Auerbach and B. Ferri, "Work in Progress - The Costs and Benefits of Using Alternative Approaches in Lecture-Based Courses: Experience in Electrical Engineering," *2010 ASEE/IEEE Frontiers in Education Conference, Washington DC, Oct 2010.*