

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/323325416>

A low-cost feedback control systems laboratory setup via Arduino-Simulink interface

Article in *Computer Applications in Engineering Education* · February 2018

DOI: 10.1002/cae.21917

CITATIONS

2

READS

455

2 authors:



[İsmail Uyanık](#)

Johns Hopkins University

16 PUBLICATIONS 57 CITATIONS

[SEE PROFILE](#)



[Bahadır Catalbas](#)

Bilkent University

6 PUBLICATIONS 3 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Feedback Control Systems Course Low Cost Laboratory Experiment Kits Development Project [View project](#)

A low-cost feedback control systems laboratory setup via Arduino–Simulink interface

Ismail Uyanik¹  | Bahadir Catalbas² 

¹Laboratory of Computational Sensing and Robotics, Johns Hopkins University, Baltimore, Maryland

²Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey

Correspondence

Ismail Uyanik, Laboratory of Computational Sensing and Robotics, Johns Hopkins University, Baltimore, 21218 MD. Email: uyanik@jhu.edu

Funding information

Scientific and Technological Research Council of Turkey (TÜBİTAK)

Abstract

Control theory education, when supported by practice, becomes more comprehensible for students and useful for their professional career. This paper presents low-cost experiments for laboratory sessions of a feedback control systems course, which introduces them modeling feedback control systems, proportional-integral-derivative (PID) controller design, root locus and Bode plots. The experiments are organized around the Arduino-based identification and control of a DC motor via Matlab/Simulink. The objective of this laboratory session is to support teaching feedback control systems via experimental investigations on a low-cost laboratory kit. The built in-house setups support Arduino–Simulink interface, so that students can download their control diagrams in Simulink to the Arduino board directly. This interface allows students to utilize high level control design tools, such as Matlab/Simulink while working on a low-cost hardware laboratory setup. Students' performance in the written exams before and after the laboratory setup were reported to evaluate the instructional effectiveness. Besides, student feedback for four semesters are also presented to evaluate the effectiveness of the laboratory experiments.

KEYWORDS

control education, education, laboratory education, laboratory techniques, learning control

1 | INTRODUCTION

The importance of hands-on experience for engineering education has long been established [9,20]. Literature studies suggest that practice is fundamental for complementing strong theory background [15] as well as developing new theories [6], since working with tangible components supports active and sensory learning for students [12,22]. Besides its substantial advantages in learning, the laboratory modules became a must for engineering colleges after ABET's (Accreditation Board of Engineering Technology) requirement for adequate laboratory practice for students [1]. However, designing and integrating such laboratory sessions to the undergraduate curriculum is very challenging in newly

developing countries such as Turkey due to the high student counts and expensive prices of the commercial educational experimental kits. This is why we used to perform simulation-based laboratory sessions in Department of Electrical and Electronics Engineering of Bilkent University, Turkey.

There are three common ways to introduce practical experience to students in engineering education; simulation, emulation, and use of physical test hardware. Among these three, the highest educational benefit can be achieved by using pilot plants which means using real size pilot hardware for experiments [15]. However, this choice is not feasible for many colleges in terms of cost and large area requirements [15]. At this point, the use of small-scale plants (such as [11,13,23]) can be considered as simple ad-hoc solutions.

For example, the use of a small-scale wind turbine for control theory education has been presented in [13]. The authors developed an educational software tool to familiarize graduate level students to the concept of decoupling control strategies for wind turbines via small-scale plant models [13]. There are some studies which develop experiments based on control and monitoring of an experimental biomass combustor via LabVIEW on a small-scale experimental plant [23]. Similarly, some PC-based electro-hydraulic system control experiments has also been proposed in [11]. This paper presents a good example of training students/researchers with PC-based control system design tools and fluid power systems. The mathematical model of the electro-hydraulic system is derived from first principles and the model is implemented in Matlab/Simulink environment. The controller designed on the Matlab/Simulink environment has been applied on a physical test bench by which students' experience with the physical hardware has also been ensured [11]. However, [13] and [23] present graduate level experiments involving deeper theoretical insight in control theory which goes beyond the scope of introductory level feedback control theory courses for undergraduate education that we focus on this paper. On the other hand, despite the fact that [11] presents experiments for undergraduate students (in addition to graduate students), it is still challenging (in terms of cost) to reproduce this kind of small scale test benches when the goal is to provide a test setup for each student (or pairs).

Different than these studies, remote laboratories [10,17,18,24,31] have also widely been used in the control theory education literature. Remote laboratory concept becomes a cost effective solution for practice needs and it may satisfy the feeling of working with a real hardware up a certain level. For instance, a Wiki-based remote laboratory platform has been developed to provide experience to students about characterization of variations of PID control techniques and remote tuning [31]. There is also web-based remote learning systems, such as NeTRe-LAB [10], which provides an opportunity to reach DC motor exercises in conventional laboratories without place and time requirements. Even though there are favorable features of remote laboratories such as low cost as compared to physical test benches, there are some important drawbacks. It has been shown that majority of the students prefer hand-on experiments with respect to remote ones [31]. One reason of this tendency may be related with lack of physical interaction in remote laboratories.

There are also some studies that design simulators for teaching control theory [3,25,26,28]. For instance, a specialized simulation infrastructure called ASPEN HYSYS has been used for control theory education to the undergraduate chemical engineering students. Besides, virtual laboratory tools also provide similar experience of working with a DC motor and controller circuit in simulation environment [3]. However, although the proposed simulation experiments in

these studies are capable of teaching special applications of control theory, these methods do not provide the feeling of working with physical components and do not help students for gaining experience to work on a real hardware system [15].

There are some companies offering commercial products for feedback control systems laboratories. However, such kits are generally quite expensive, especially when the goal is to make sure that all students interact with the hardware. For instance, Quanser develops DC motor control and vertical take-off kits for use in feedback control systems laboratories. However, these kits are very expensive for newly developing countries when the student counts are high to give a lab setup to each one or two students. In order to solve this problem [7] proposes two sets of experiments which include eight hands on experiments that benefit from a wide range of equipment such as use of circuit simulations, circuit realization and control laboratory kits in order to teach control theory notions. Thus, students learn overshoot, rise time, settling time concepts, and use of PSpice and Matlab simulation programs. Besides, there are some experimental setups, which are capable of conducting PID controller and frequency response experiments with a real DC Motor module [19,29]. Unfortunately, most of these examples become expensive for high number of purchases despite their clear advantages on teaching.

Considering the state of the art techniques in the literature, some design decisions are determined for the desired experimental feedback control system setup as

- The students should use high level design tools for controller design.
- Students should not deal with the details of hardware and focus on the principles of feedback control theory.
- Laboratory setups should be low-cost, so that many of them can be simply built and replaced easily in case of failures.

Motivated by these design requirements, Matlab/Simulink environment was used due to its wide spread use in both academia and industry [21,30]. Matlab provides simple and efficient programming environment, especially for educational purposes, for developers from various experience levels. Sometimes, Matlab may be used as sole programming tool for most engineering studies due to its capability to mathematical programming philosophy and integrating different programming platforms [20,21]. In addition, Simulink provides a graphical editor for block diagram programming, which eases the use of custom blocks and reduces the possibility of implementation errors. Currently, Simulink has a wide application area including control systems, digital signal processing and communication fields and there are various built-in block modules that provide simplified and time efficient working environment for professionals as well as students [2]. Among all, Matlab/Simulink provides hardware support for numerous microcontrollers, motor

controllers and sensors. The hardware interface of Matlab/Simulink will be the key functionality that is being used during kit's development stage.

Another key property of using Matlab/Simulink is that students can develop their control diagrams in Simulink and directly download them to a supported hardware without dealing with details of hardware design or micro-controller programming. Among various alternative microcontroller boards that have Matlab/Simulink support, Arduino Uno was used due to its low cost regarding to our third design decision. In addition to its low price, Arduino offers open source hardware and software option for developers with varying backgrounds. This feature makes Arduino a fast prototyping platform for various projects [5,14,16,32]. Also, there are some example uses for education such as laboratory equipment for students [27] and educational mobile robot [4].

Motivated by these, a new laboratory setup has been proposed that is focused on identification and control of a DC motor using Matlab/Simulink environment and Arduino microcontroller. Despite there are a lot of successful systems that are currently used in the literature, the proposed setup differs from the available literature in the following manners:

- The proposed setup is very low cost as compared to literature (below 100 USD), which allows providing one hardware setup for each student (or pairs) during laboratory experiments. The Department of Electrical Engineering at Bilkent University built 100 of these setups and they have been successfully used for the last 4 years.
- Students program their controllers in Simulink environment and download them to Arduino microcontroller. The control diagram works in real-time with minimal delays (10 ms delay due to 100 Hz sampling frequency), which are generally higher for low-cost hardware in the loop systems.
- The experimental investigation of very abstract concepts such as root locus, Fourier analysis, frequency domain identification, Bode plots and time delays (detailed investigation on frequency domain analysis has initially reported in [8]) has been demonstrated on this low-cost setup.
- Most importantly, the paper reports both exam results that illustrate the instructional effectiveness of the course as well as students' surveys that assess the practicality of the lab setups.

2 | DESIGN AND DEVELOPMENT OF EXPERIMENTAL KIT

2.1 | Prior status of the lab course

Feedback Control Systems is one of the must courses in the curriculum of the Bachelor of Science program of the Electrical and Electronics Engineering (EEE) Department of Bilkent University, Ankara, Turkey. The theoretical lectures,

which introduce the principles of feedback control, are supported by three laboratory sessions. Prior to proposed lab kits, students performed Matlab/Simulink based simulation experiments for the identification and control of a DC motor model for several years using a sample DC motor data. They also submitted a post lab report after the lab to discuss and comment on the results they obtained during the lab session.

Student evaluations showed that the simulation based lab sessions did not help for the course topics. This result raised two fundamental problems. First, students cannot integrate the theoretical courses to the practice. Second, the lab sessions do not contribute to their understanding of course topics.

2.2 | Proposed hardware setup

This section focuses on the proposed lab kit for the feedback control systems experiments (an exploded view is illustrated in Figure 1). As briefly mentioned in section 1, hardware setup concentrates on Arduino-based identification and control of a DC motor. The proposed kit consists of an Arduino Uno microcontroller, Arduino motor driver shield, a DC motor with encoders, a power adapter and a USB cable for computer (Matlab/Simulink) interface. Arduino Uno microcontroller and associated motor driver shield are commercial off-the-shelf products. The motor driver shield supports up to 12 V and 2 A supply voltage and current for the target motor with a power adapter, respectively. The motor is 12 V Pololu DC motor with gearhead and integrated quadrature encoder. 12 V power adapter with maximum 3 A current supplies necessary power for the lab kit. Finally, a standard USB cable is used to interface Arduino with computer and hence Matlab/Simulink through serial communication. In addition, several consumables are used for the preparation of the lab kit.

Table 1 lists all equipment used in the developed experimental kit with their current prices. Notice that total cost of a single kit is only about 97 USD, which is very cheap when compared to commercial laboratory kits. This is very important when the goal is to build many of these kits to make sure that all students interact with the hardware during the lab sessions. To accomplish this, 100 kits are built assuming that the students will work in pairs to complete the lab sessions (for a maximum student count of 200). Note that although the students work in pairs to collect data and build controllers in lab, they write individual post lab reports to make sure that they interpret the results based on their own understanding.

On the software side, Matlab/Simulink is used as fundamental programming interface to program the experimental kit. In addition, these computers should have a C compiler compatible with Matlab, Arduino USB driver and Arduino target installer. Fortunately, Matlab/Simulink is a key programming tool for most electrical and electronics engineering departments. Having a licensed Matlab, Arduino target installer can be freely obtained from Simulink support

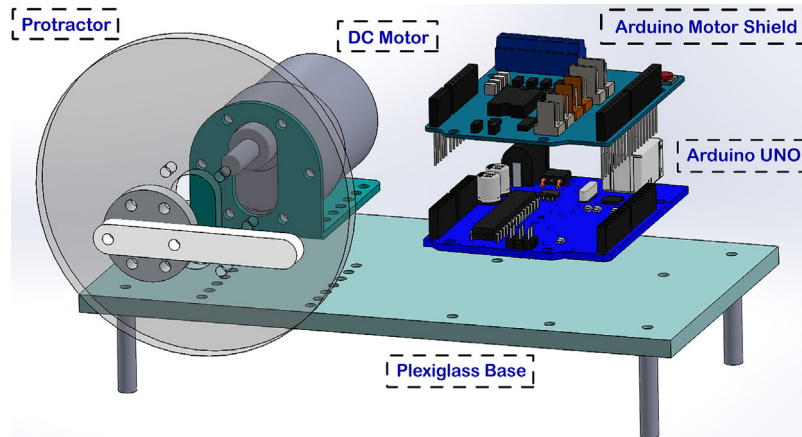


FIGURE 1 Proposed lab setup for feedback control systems course laboratory sessions

TABLE 1 Equipment list

Equipments	Price (USD)
1. Arduino Uno	15,49
2. Arduino Motor Shield	21,99
3. DC Motor with Encoder	39,99
4. Power Adaptor	2,3
5. Mechanical Components	10,29
6. Plexiglass Base	6
7. Consumables	1
Total Cost:	97,06

packages library. C compilers and Arduino USB drivers can also be obtained as free versions from web. Therefore, proposed setup does not require additional paid software when Matlab is available.

In addition to having a computer with the software mentioned above installed, some previously compiled code blocks are supplied to students to make sure that they don't get distracted with the details of microcontroller level programming. The first code block supplied is the *Actuation* block which converts the input voltage command generated by the controller to direction and pulse-width-modulation signals, so that Arduino and the motor driver shield feeds desired supply voltage to the motor terminals. The second block supplied is the *Sensing* block that measures the speed of the motor by counting the quadrature encoder pulses. The two software blocks are combined in a single Simulink block named as *DC Motor Plant* (illustrated in Figure 2), so that students can use this block as a plant model in their Simulink diagrams. This way, complexity of dealing with the details of hardware for the lab course is reduced significantly. Interested students can later go into the details of these code blocks to modify them for further use in their different projects. One advantage of using Arduino and a DC motor in the lab sessions encourages students to work with hardware and notice that controlling a DC motor via feedback has various applications in different robotics and control theory projects.

On the other side, the disadvantage of building own lab kits is that it requires some man power. The nice thing about the proposed setup is that it uses off-the-shelf components and do not require so much design and production processes. Based on the experience for building 100 kits for Bilkent University, two teaching assistants can develop the software for actuation and sensing blocks by spending ten hours each. One teaching assistant spent four hours for 3D-CAD design of the main body frame and motor inertia. The most time-consuming part was the integration and test of the whole setup before giving them to students. Five teaching assistants spent eight hours each for integrating and testing all 100 setups. There is also a plenty of time spent for the design of the lab questions. However, only the time spent required for preparing the lab kits is reported, since each instructor will most probably prepare different lab questions once they built this setup.

2.3 | Intended outcomes

The feedback control theory topics, modeling control systems, computing performance characteristics, root locus, Bode plots, and Nyquist stability are taught in class as a single condensed course in one semester. The lectures are explained in classroom by the course instructor and students worked on several theoretical questions as well as some Matlab simulations in class. The three simulation-based lab experiments were aimed to support students' learning. However, the simulation-based experiments did not motivate the students to integrate their theoretical knowledge with real life problems. Hence, students were not able to grasp the fundamentals of the course.

With this in mind, the intended course outcomes with the new laboratory experiments are designed as follows.

1. The lab setups should allow high level control design without distracting students' attention with the details of microcontroller level programming.
2. The lab experiments should support teaching fundamental control theory topics such as system modeling and

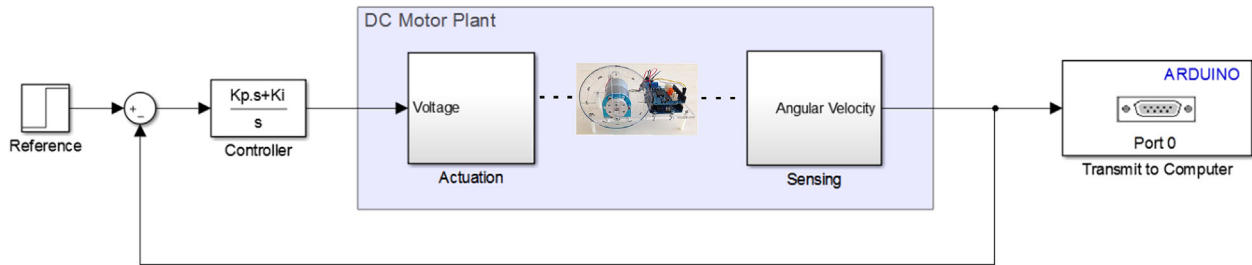


FIGURE 2 A sample feedback control system block diagram that is downloaded to Arduino to work inside the microcontroller. DC Motor Plant block is supplied to students to be used as a plant transfer function block in their control diagrams. A serial transmitter block is attached to the output signal to transmit measured outputs back to the computer via serial communication

characterization, root locus and Bode plots and they should increase students' understanding of these topics by utilizing them on a physical system.

3. The simple nature of proposed lab kits should encourage students for working with hardware.

3 | DETAILS OF INDIVIDUAL EXPERIMENTS

This section details the three individual lab experiments, each of which is designed to support teaching of different feedback control theory concepts.

3.1 | Lab 1: DC motor identification and velocity control

This lab experiments aims to present the concept of identification and control for a DC motor. To this end, students are given the *DC Motor Plant* block explained in section 2.2 to interface their control diagrams in Simulink to the hardware. Note that students program Arduino by designing Simulink blocks and downloading to hardware. Since the blocks run on Arduino, Serial Communication block is used to transmit the measured output to computer (Matlab/Simulink). A sample feedback control diagram (designed in Simulink and installed to hardware) is illustrated in Figure 2. The lab experiments require the completion of the following tasks:

- 1 Perform parametric system identification for a DC motor to obtain its transfer function from input–output data.
 - i. Design a Simulink block diagram to apply 10V step input to *DC Motor Plant* and record the output velocity (shown with red in Figure 3 (a)).
 - ii. Fit a first order transfer function model to velocity response (shown with blue in Figure 3 (a)) and obtain the transfer function via Laplace transform.
- 2 Design a controller to regulate the output of the identified plant in simulation.
 - i. Design a PI controller for the estimated transfer function in Simulink and tune the control parameters manually to

achieve zero steady state error with maximum percentage overshoot 10% and settling time less than 1.5 s.

- ii. Record the velocity response of the controller in simulation (shown with blue in Figure 3 (b))

- 3 Test the controller on hardware.

- i. Implement the same controller for *DC Motor Plant* and download the code to Arduino and record the velocity response (shown with red in Figure 3 (b)).
- ii. Compare the velocity response of the same controller both for the estimated transfer function and real hardware.

3.2 | Lab 2: Controller design via root locus

This lab experiment aims to present the concepts of controller design via root locus as an example of adjusting control parameters with respect to a design criterion without manual tuning. Students use the proposed setup to design position controller for the DC motor considering the sampling delay in the system. The lab experiment requires the completion of the following tasks:

- 1 Use Pade approximation to model the sampling delay.
 - i. Use Pade approximation transfer function to approximate 10 ms delay due to 100 Hz clock frequency.
- 2 Plot the root locus of a closed-loop system.

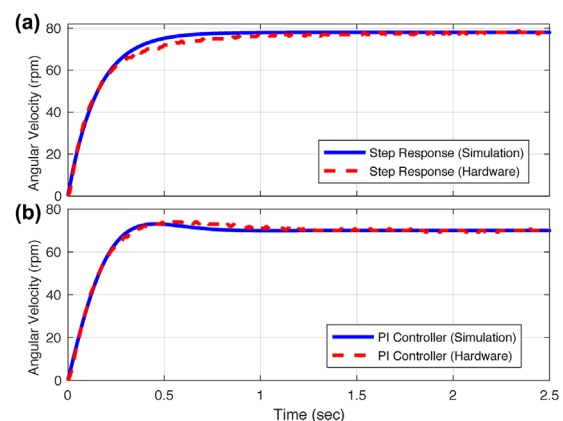


FIGURE 3 Step response and velocity control results for simulation and hardware

- i. Add an integrator block to convert angular velocity output of the *DC Motor Plant* to angular position.
 - ii. Use a PD controller in the form $G_c(s) = K \frac{0.064s+1}{0.0064s+1}$ with a single, tunable gain.
 - iii. Derive the characteristic equation of the closed-loop system and plot the pole locations on the complex plane for different values of K (a sample is shown in Figure 4).
- 3 Find the stability range to choose optimal control parameters
- i. Find the stability range for K using the Routh-Hurwitz and validate on the root locus that poles beyond this gain cross real axis.
 - ii. Choose optimal pole location yielding a stable system with maximum damping for the complex poles.
 - iii. Compute the gain to achieve desired pole locations.
 - iv. Apply the resulting controller to first simulation plant and then to real hardware the compare the controller performance.

3.3 | Lab 3: Frequency domain system identification

The goal of this lab experiment is to introduce frequency domain system identification of a linear time-invariant system from input–output data. Students use the proposed setup to perform data-driven system identification in frequency domain by utilizing principles from linearity and time-invariance. Students complete the following tasks to achieve these goals:

- 1 Use single-sine excitations to obtain Bode plots of an LTI system with input–output data
 - i. Apply 15 sinusoids with each having different frequencies to *DC Motor Plant* and record the angular velocity data.
 - ii. Use Fourier transform to find magnitude and phase changes corresponding to each frequency.
 - iii. Plot the Bode plot by using obtained magnitude and phase changes (a sample is illustrated in Figure 5).
 - iv. Plot the Bode plot of the estimated transfer function obtained in Lab 1 using Matlab's built-in *bode* function and compare it with the experimental one (see Figure 3).

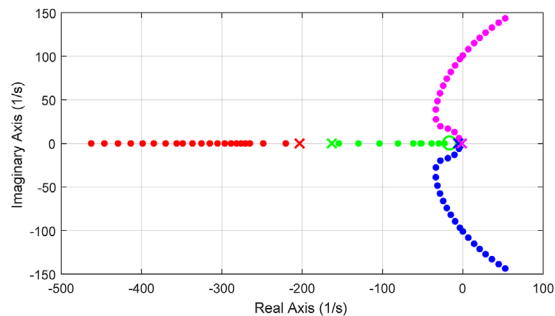


FIGURE 4 Root locus with a template PD controller

- 2 Observe the effect of time delay on frequency response
 - i. Comment on the difference in phase plots.
 - ii. Add a Pade approximation to the estimated transfer function for 10 ms sampling delay and compare the new Bode plot (see Figure 5) with the previous one.

4 | ASSESSMENT ON THE INSTRUCTIONAL EFFECTIVENESS

In this part, some results regarding students' performances in written exams before and after the use of proposed lab setups are presented. To accomplish this, the exams for the last four course offerings before and after the use of new lab setups were collected. In order to obtain a fair understanding of the effects of lab setups, students' marks (which were previously graded by the course instructors) for the topics mathematical modeling and performance characteristics, root locus and Bode plots associated to each lab session are noted.

Of course, each of these topics were not mentioned in all exams and average student marks for each exam was also different based on the difficulty level of the exam. Considering these observations, students' success for each topic as a normalized mean across different exams before and after the use of lab setups has been computed. Therefore, for each topic there was a related question in M exams before the lab setups such as E_1, E_2, \dots, E_M . Let's also define the number of students took each exam as $N^{E_1}, N^{E_2}, \dots, N^{E_M}$. Student's mark on the related course topic as $\alpha_j^{E_i}$ and on the exam as $\beta_j^{E_i}$ for each student and each exam was noted. Note that α and β is normalized, such that full mark for each question and exam corresponds to 1. Hence, students' performance for a course topic in each exam is computed as

$$\mu^{E_i} := \frac{1}{N^{E_i}} \sum_{j=1}^{N^{E_i}} \alpha_j^{E_i} \quad (1)$$

Similarly, students' performance on each exam is computed as

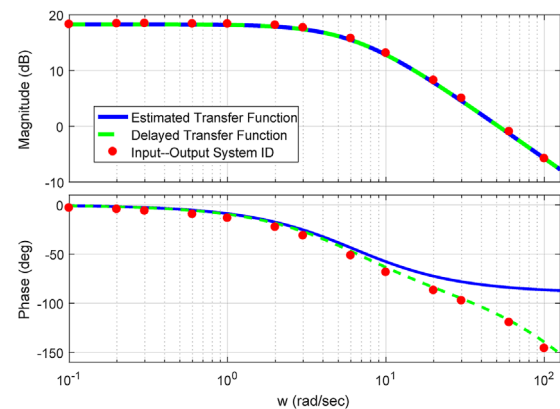


FIGURE 5 Bode plots of the velocity transfer function

$$\Psi^{E_i} := \frac{1}{N^{E_i}} \sum_{j=1}^{N^{E_i}} \beta_j^{E_i} \quad (2)$$

Now, the results, μ^{E_i} to obtain $\Psi^{E_i} = 0.5$ for each exam are normalized to consider the effects of exam's difficulty levels. Hence, the normalized students' success for each exam is computed as

$$\eta^{E_i} := 0.5\mu^{E_i} / \Psi^{E_i} \quad (3)$$

Finally, student' performance on each topic across all exams before the lab setups is computed as

$$P := \frac{\sum_{i=1}^M \eta^{E_i} N^{E_i}}{\sum_{i=1}^M N^{E_i}} \quad (4)$$

Now, using (4), one can compute the students' performance for each course topic before and after the use of lab setups. Figure 6 demonstrates the results of this analysis. As seen in the Table 2, students normalized marks on these course topics increased in a noticeable manner. These results show the instructional effectiveness of the proposed setups for control theory courses.

5 | ASSESSMENT BASED ON STUDENT SURVEYS

In this section, student surveys four the last course offerings regarding the usefulness of the lab setups are presented in terms of relating theory to experiments and difficulty level of lab components.

5.1 | Relating theory to experiments

In this part, students are asked to evaluate if the lab sessions were helpful to understand the applications of theory they learned in class. The four questions in this set specifically asks whether the experiments performed in different lab sessions helped them to understand the use of “System Identification via Mathematical Modeling,” “Controller Design via Root Locus,” and “Frequency Response of Linear Time-Invariant (LTI) Systems.” Finally, as an overall assessment, students are asked to evaluate lab sessions' contribution to their understanding of “Identification and Control of a DC motor.”

In order to avoid the subjectivity of students in a quantitative analysis, students are asked to give their answers in a qualitative manner, which are later mapped to grades 1–5, respectively (as shown in Table 2). Interestingly, during the first semester of the lab sessions, where 90 students answered

the survey yielded an average of 4.58 for four questions, which corresponds to somewhere in the middle of “Good” and “Excellent” in our qualitative survey. For the following semesters, 93, 99, and 58 students yielded an average of 4.07, 4.20, and 4.41, respectively.

5.2 | Difficulty level of lab components

In this part, students are asked to evaluate the difficulty level of different components in the lab session. These questions specifically ask the difficulty level of “Working with hardware components,” “Working with Matlab/Simulink environment,” and “Laboratory Assignments.”

Note that, student answer about difficulty level of working with hardware components for the first semester with 90 evaluations yielded an average result of 3.92 that almost corresponds to easy (4.0) in our survey. When the other semesters are considered, student evaluation averages are 3.56, 3.93, and 3.55. There are two important results that needs to be discussed here. First of all, students find it almost easy (somewhere between fair and easy but close to easy with a weighted average of all semesters as 3.76) to work with hardware components. This supports our goal for building a simple hardware, where students do not deal with the details of hardware and use it as simple as possible for their experiments. The second thing that needs to be comment on is that 3.92 and 3.93 results belong to Spring semesters, when the course is offered in its regular semester for regular students. On the other hand, 3.56 and 3.55 belong to Fall semesters when the course is offered for irregular students. Evaluation results show that regular students find it easier to work with hardware components. Also, only twelve students responded very difficult for this item during all semesters.

The students answer for evaluating difficulty level of working with Matlab/Simulink environment yielded an average result of 4.00 for the first semester and 3.53, 3.89, and 3.52 for the following semesters. Evaluations show that working with Matlab/Simulink is also not a big deal for students. Additionally, bias between regular and irregular semester students are also observed in this item.

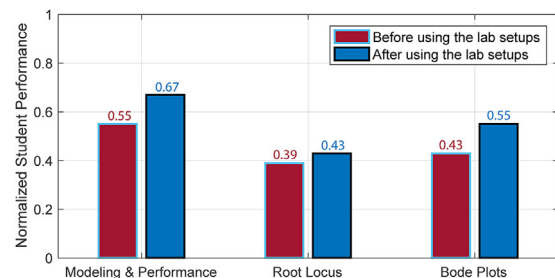


FIGURE 6 A comparative illustration of students' performance on each topic before and after using the lab setups

TABLE 2 Student Evaluations: Part I: Poor, 1; Needs Improvement, 2; Fair, 3; Good, 4; Excellent, 5. Part II: Very Difficult, 1; Difficult, 2; Fair, 3; Easy, 4; Very Easy, 5

Questions	Mean	1 (%)	2 (%)	3 (%)	4 (%)	5 (%)
PART I: Relating theory to experiments						
a. System identification via mathematical modeling	4,39	1,18	1,18	8,85	35,40	53,39
b. Controller design via root locus	4,32	0,89	2,36	11,80	33,92	51,03
c. Frequency response of LTI systems	4,14	1,78	6,21	16,57	27,22	48,22
d. Identification and control of a DC motor	4,36	1,82	1,21	9,39	34,55	53,03
PART II: Difficulty level of lab components						
a. Working with hardware components	3,76	2,65	7,35	29,12	32,94	27,94
b. Working with Matlab/Simulink	3,76	2,95	5,31	31,27	34,22	26,25
c. Laboratory assignments	3,52	2,35	9,41	40,59	28,82	18,83

Finally, students are asked to evaluate the difficulty level of the laboratory assignments regardless of the hardware and Matlab/Simulink to understand their perception of difficulty. Student evaluations for the four semesters yielded average scores of 3.78, 3.21, 3.72, and 3.29 for the same qualitative difficulty measures. First of all, the bias between regular and irregular students is still there for a validation. Second, students find the assignments more difficult than working with hardware or Matlab/Simulink.

6 | CONCLUSION

In this paper, a novel low-cost experiments setup for feedback control systems course is proposed. The proposed experiment kit utilizes simple Arduino micro-controllers with Matlab/Simulink interface. The setup can be programmed easily via Simulink without distracting student focus with the details of micro-controller programming.

Students' performance in written exams are reported to illustrate the instructional effectiveness of the proposed lab setups. Besides, student assessments suggest that the proposed setup is easy to use with plug and play features to Simulink. Students also find it useful for relating theory to experiments, since the setup supports experimental validation of some challenging tasks such as modeling and control of a DC motor, controller design via root locus and frequency domain system identification for an LTI system.

The results show that the proposed lab kit supports both theoretical and practical goals of the lab sessions. In addition, its low price (below 100 USD) makes it an excellent choice for the universities with high student counts, especially when the goal is to give a setup to each one or two students. The Matlab/Simulink interface supports easy programming in a block-diagram based environment. Also, this interface eases the transition to proposed hardware, since most engineering colleges use Matlab/Simulink in their education programs.

ACKNOWLEDGMENTS

We would like to thank Hitay Özbay, Ömer Morgül, Orhan Arkan, A. Bülent Özgüler, Hasan Hamzaçebi, Ali Nail İnal, Mustafa Gül, H. Eftun Orhon, and A. Safa Öztürk for their ideas and support. The authors appreciate the financial support of the Scientific and Technological Research Council of Turkey (TÜBİTAK).

ORCID

Ismail Uyanik  <http://orcid.org/0000-0002-3535-5616>

Bahadır Catalbas  <http://orcid.org/0000-0002-8684-5754>

REFERENCES

1. ABET. (2012). *Criteria for accrediting engineering programs*, available at <http://www.abet.org/wp-content/uploads/2015/04/etac-criteria-2013-2014.pdf>.
2. A. Albagul, O. O. Khalifa, and M. Wahyudi, *Matlab and simulink in mechatronics education*. *Int. J. Eng. Educ.* **21** (2005), 896–905.
3. I. H. Altas and H. Aydar, *A real-time computer-controlled simulator: For control systems*. *Comput. Appl. Eng. Educ.* **16** (2008), 115–126.
4. A. Araujo et al., Integrating Arduino-based educational mobile robots in ROS. In *2013 13th International Conference on Autonomous Robot Systems (Robotica)*, (2013), 1–6.
5. J. A. Arízaga et al., Automatic control for laboratory sterilization process based on arduino hardware. In *Electrical Communications and Computers (CONIELECOMP), 2012 22nd International Conference on* (2012) (pp. 130–133). IEEE.
6. D. S. Bernstein, Control experiments and what I learned from them: A personal journey. *IEEE Control Systems Magazine*, **18** (1998), 81–88.
7. Y. Buğday and M. Ö. Efe, *A set of computer aided automatic control experiments for undergraduate students*. *Comput. Appl. Eng. Educ.* **21** (2013), 300–312.
8. B. Çatalbaş and İ. Uyanık, *A low-cost laboratory experiment setup for frequency domain analysis for a feedback control systems course*. *IFAC-PapersOnLine*, **50** (2017), 15704–15709.

9. Y. C. Chen and J. M. Naughton, *An undergraduate laboratory platform for control system design, simulation, and implementation*. *IEEE Cont. Syst.* **20** (2000), 12–20.
10. I. Colak et al., *A novel web-based laboratory for DC motor experiments*. *Comput. Appl. Eng. Educ.* **19** (2011), 125–135.
11. E. Erzan Topçu, *PC-based control and simulation of an electro-hydraulic system*. *Comput. Appl. Eng. Educ.* **25** (2017), 706–718.
12. R. M. Felder, *Reaching the Second Tier: Learning and Teaching Styles in College Science Education*. *J. Col. Sci. Teach.* **23** (1993), 286–290.
13. S. Frago et al., *Educational software tool for decoupling control in wind turbines applied to a lab-scale system*. *Comput. Appl. Eng. Educ.* **24** (2016), 400–411.
14. F. D. Gonzalez, J. G. Guarnizo, and G. Benavides, *Emulation system for a distribution center using mobile robot, controlled by artificial vision and fuzzy logic*. *IEEE Latin America Transactions*, **12** (2014), 557–563.
15. G. C. Goodwin et al., *Emulation-based virtual laboratories: A low-cost alternative to physical experiments in control engineering education*. *IEEE Transactions on Education*. **54** (2011), 48–55.
16. G. Guo and W. Yue, *Autonomous platoon control allowing range-limited sensors*. *IEEE Trans. Vehi. Technol.* **61** (2012), 2901–2912.
17. E. Guzmán-Ramírez et al., *An educational tool for designing DC motor control systems through FPGA-based experimentation*. *Int. J. Elec. Eng. Educ.* **52** (2015), 22–38.
18. M. Kalúz et al., *ArPi lab: A low-cost remote laboratory for control education*. *IFAC Proc. Vol.* **47** (2014), 9057–9062.
19. Z. Kamis, E. E. Topcu, and I. Yuksel, *Computer-aided automatic control education with a real-time development system*. *Comput. Appl. Eng. Educ.* **13** (2005), 181–191.
20. N. A. Kheir et al., *Control systems engineering education*. *Automatica*. **32** (1996), 147–166.
21. T. Michałowski, *Applications of MATLAB in Science and Engineering*. Published by InTech Janeza Trdine, **9** (2011), 51000.
22. S. Papert, *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc. (1980).
23. A. Regueiro et al., *A practice for engineering students based on the control and monitoring an experimental biomass combustor using labview*. *Comput. Appl. Eng. Educ.* **25** (2017), 392–403.
24. P. Reguera et al., *A low-cost open source hardware in control education. case study: Arduino-Feedback MS-150*. *IFAC-Papers-OnLine*. **48** (2015), 117–122.
25. A. Rmilah et al., *A PC-based simulation platform for a quadcopter system with self-tuning fuzzy PID controllers*. *Comput. Appl. Eng. Educ.* **24** (2016), 934–950.
26. J. Sánchez, S. Dormido, and F. Esquembre, *The learning of control concepts using interactive tools*. *Comput. Appl. Eng. Educ.* **13** (2005), 84–98.
27. J. Sarik and I. Kymissis, *Lab kits using the Arduino prototyping platform*. In *Frontiers in Education Conference (FIE), 2010 IEEE* (2010) (pp. T3C-1). IEEE.
28. A. A. Taimoor, *Virtualization of the process control laboratory using ASPEN HYSYS*. *Comput. Appl. Eng. Educ.* **24** (2016), 887–898.
29. E. E. Topcu, Z. Kamis, and I. Yuksel, *Automatic control training on a DC motor control module*. *IFAC Proc. Vol.* **36** (2003), 269–274.
30. S. Uran and K. Jezernik, *MATLAB web server and M-file application*. In *2006 12th International Power Electronics and Motion Control Conference, IEEE, 2006*, pp. 2088–2092.
31. N. Wang et al., *Integration of a remote PID motor speed control experiment with teaching in engineering education*. *Eng. Edu. Lett.* **2017** (2017), 1.
32. M. Zolkapli et al., *High-efficiency dual-axis solar tracking development using Arduino*. *2013 International Conference on Technology, Informatics, Management, Engineering and Environment, IEEE, 2013*, pp. 43–47.



I. UYANIK is a postdoctoral researcher in Laboratory of Computational Sensing and Robotics (LCSR) at Johns Hopkins University. He received his PhD degree in Electrical and Electronics Engineering from Bilkent University in May 2017. He also received his BSc and MSc degrees from

the same department in June 2009 and August 2011, respectively. His research focuses on discovering the principles of animal locomotion by developing novel techniques in the areas of system identification theory, computational neuroscience and robotics. He is also the recipient of the Aselsan PhD Fellowship.



B. CATALBAS is a PhD candidate in Electrical and Electronics Engineering Department at Bilkent University. He also received his BSc and MSc degrees from the same department in June 2013 and September 2015, respectively. He is working in developing novel deep learning

methods and he also interests in legged locomotion applications of deep learning. He is the recipient of the Scientific and Technological Research Council of Turkey (TÜBİTAK) Graduate Scholarship.

How to cite this article: Uyanik I, Catalbas B.

A low-cost feedback control systems laboratory setup via Arduino–Simulink interface. *Comput Appl Eng Educ.* 2018;1–9.

<https://doi.org/10.1002/cae.21917>