

Final Lab Portfolio

EEET-427-01: Controls Systems Lab

Blizzard MacDougall, rmm3470

Abstract—In an effort to further understanding of controls systems, students were instructed to construct a system capable of balancing a ball on a beam. This system was to be able to react to outside stimuli, and keep the ball as stable as possible.

I. INTRODUCTION

THE purpose of these experiments was to provide real-world experience in implementing the controls systems concepts described in lecture. In addition, the labs also provided experience with troubleshooting both systems and Arduino code. Background knowledge necessary for this lab includes basic calculus skills and physics (specifically two-dimensional motion and forces). These labs make extensive use of an Arduino (purchased at the start of the course), primarily to control a motor. This is done using code provided by the instructor, which is modified to suit each student's particular hardware. The labs also make use of soldering skills, as well as basic hot-glue and foam-core working skills. Each portion of the lab received a brief description, and were roughly based off of the work in lecture that week. Moreover, each lab built upon the previous lab. The final lab project is pictured in Figure 1.

II. PARTS AND MATERIALS

There are several major components in this lab. First and foremost, the RoboRED YourDuino board (hereafter referred to as "the Arduino"), which is the microcontroller the labs are based in. This board controls all of the following devices, either directly or indirectly. The primary motor used in these labs is the DFRobot FIT0450 DC motor (hereafter referred to as "the motor"). The ultimate goal of the motor is to control the beam on which a ball will balance. The motor is controlled by a Sseed studio MIX1508 H-Bridge Motor Driver (hereafter referred to as "the motor driver" or "the motor driver board"). This motor driver takes power from a steady power source, and controls the motor based on PWM signals sent from the Arduino. The power for the motor driver in all labs is provided by an AC-to-USB power supply. In later labs, a Pololu carrier board containing a STMicro VL53L1X Time-of-Flight sensor (hereafter referred to as "the TOF sensor" or "the time-of-flight sensor") is used. This is used to find the ball's position on the balancing beam. The lab also makes use of foam core and hot glue, for the balancing beam and for connecting components. All code for each lab was provided by the professor at the beginning of lab.

III. RESULTS

A. Lab 1

The first lab's instructions were to create a standard transfer function and block diagram for the motor distributed at the beginning of lab. This was done by sending a constant voltage to the motor, and graphing its velocity over time, as seen in Figure 2. Values of a and b can be found in Table B. The transfer function for the motor can be represented as $\frac{a}{s+b}$ in Figure 7, and is calculated in Equation 2. These values are key for all following labs, as every subsequent lab's code uses these values to control the motor.

B. Lab 2

This lab's purpose was to compare different ways of controlling the motor. This was done by giving the motor a trapezoidal reference signal, and comparing the response with different control schemes. The control schemes tested in this lab were using closed-loop feedback and feed-forward. The graph in Figure 3 shows the final comparison of all of these control mechanisms. This lab found that using both closed-loop feedback and feed-forward in conjunction gave the fastest response time with the most accurate steady-state speed.

C. Lab 3

The purpose of this lab was to introduce how to decouple the input, and how to ensure that the motor works properly when programmed to turn backwards. This used an expanded version of the trapezoidal reference signal used in the previous lab, including a negative portion of the signal. Figure 4 shows both the input reference signal, and that the output signal is close to the same path. In addition, this lab ensured that the motor never tried to use more power than was available (in this case, 5V). This can be seen in the yellow line in Figure 4.

D. Lab 4

This lab's instructions were to take the previous control schemes, and add integral control on top of them. This used the same trapezoidal reference signal as in Lab 3. This resulted in a motor response curve that was more accurate to the input signal, and was more accurate in position. Using the control system consisting only of proportional control and feed-forward produced an error in speed of 35.49rad/sec , while the control system including integral control produced an error of only 11.82rad/sec . This can be seen in comparing Figure 4 and Figure 5, and comparing each motor's speed plot with the `refVel` or `Wref` plots.

E. Lab 5

In this lab, students moved from tuning the velocity error to tuning the position error. This is crucial, as a high error in the position of the motor will mean an even higher error in the beam's position, making it impossible to balance the ball properly. To do this, the control system was given a constant position, and the motor's position over time was plotted. After tuning, the motor's error in position was reduced to 1.20rad , with a response time of $< 500\text{ms}$, as shown in Figure 6.

IV. DISCUSSION AND SUMMARY

There were some errors in the directions provided by the professor. This resulted in some uncertainty as to how to proceed, the most glaring of which was a discrepancy between the completed system the professor would periodically demonstrate and the specifications for correctly building the system. More specifically, the length of the beam on which the ball sits was significantly longer than the base that supported it. To solve this issue, I shortened the beam to fit the base. This resulted in needing to modify most of the code given by the professor, to account for the differing ratio between beam length and the range of motion of the motor.

A majority of the time, I was working outside of my comfort zone, as the last time I had worked with Arduino code previous to this lab was almost two years ago. This encouraged me to relearn C++ outside of class, and pushed me to learn more languages in preparation for the future to mitigate problems caused by inexperience.

Most of my physical mechanical skills came in handy in this course. Particularly, small arts-and-crafts skills like use of foam core and hot glue. This helped when it came time to troubleshoot the mis-sized beam, as I not only had to change the length of the beam itself, but also account for the beam in the lifting arm. This can be seen in Figure 1 in the triangular notch cut out of the lifting arm on the right side of the device, allowing full range of motion for the motor.

Overall, I learned that I find errors I've made by helping others. There were several times in the process of physically building the system that I had made some measurement incorrectly, particularly the beam length miscalculation mentioned previously. Only by other students asking me questions about their system did I realize I made the measurement error. Overall, I've found that I tend to work best in a small group of people, where we collectively share information back to the larger group.

By the end of this course, I had a good grasp on most first-order system concepts. In addition, I also understood how to control said systems, both mathematically and in Arduino code.

APPENDIX A EQUATIONS AND CALCULATIONS

A. Equations

$$\frac{1}{b} = \tau = t_{65\% \text{ steady-state}} \quad (1)$$

$$\omega_{mtr; \text{steady-state}} \approx 750 \text{ rad/s}$$

$$V_{arm} = 2.5V$$

$$\frac{\omega_{mtr}}{V_{arm}} \approx 299 \frac{\text{rad}}{\text{sV}}$$

$$0.65 * \omega_{mtr; \text{steady-state}} = 450 \text{ rad/s}$$

$$t_{65\% \text{ steady-state}} \approx 120 \text{ ms}$$

$$b = \frac{1}{\tau} = \frac{1}{0.12 \text{ s}} = 8.3 \text{ Hz} \quad (2)$$

$$\frac{a}{s+b} @ \text{ steady-state} = \frac{a}{b}$$

$$299 \frac{\text{rad}}{\text{sV}} = \frac{a}{b}$$

$$a = \left(299 \frac{\text{rad}}{\text{sV}} \right) (8.3 \text{ Hz})$$

$$a = 2491.5 \frac{\text{rad}}{\text{s}^2 \text{V}}$$

APPENDIX B TABLES

Variable	value	unit
a	2491.5	$\frac{\text{rad}}{\text{s}^2 \text{V}}$
b	9.3	Hz
g	9.81	m/s
L_{arm}	10	cm
L_{beam}	36	cm
C_{beam}	5.625	
D_{beam}	20	
C_{ball}	1	
D_{ball}	6	
$K_{p, arm}$	0.1	$\frac{\text{Vs}}{\text{rad}}$
$K_{p, ball}$	1	$\frac{\text{rad}}{\text{m}}$

TABLE I
VALUES FOR VARIABLES IN FIGURE 7

APPENDIX C IMAGES

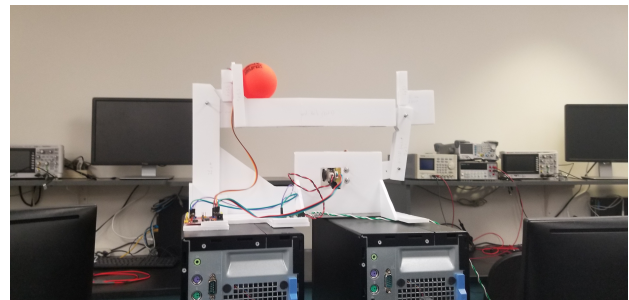


Fig. 1. The completed physical system.

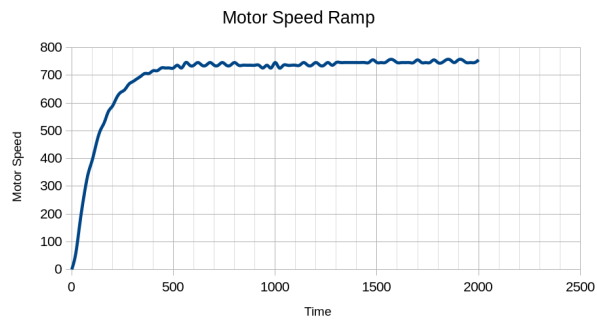


Fig. 2. Motor response to a steady 2.5V input

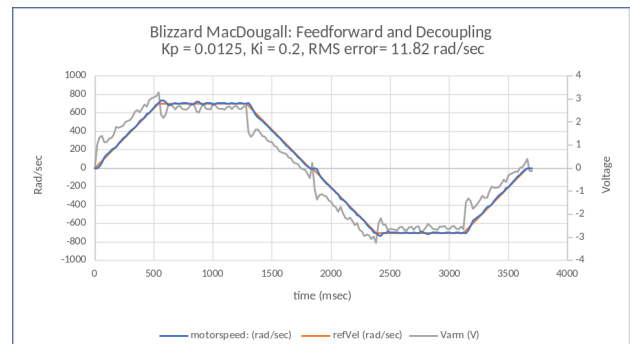


Fig. 5. Trapezoidal reference signal with PI control.

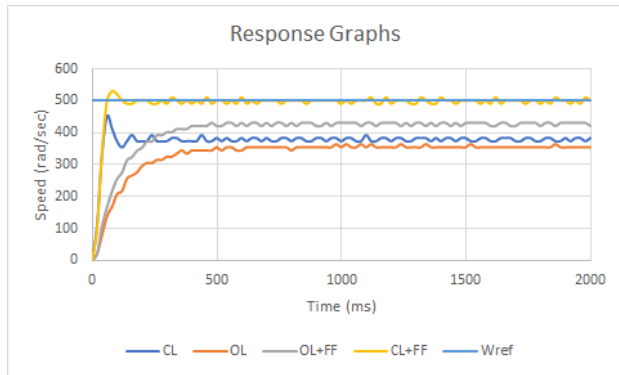


Fig. 3. Comparison of open-loop, closed-loop, and feed-forward systems.

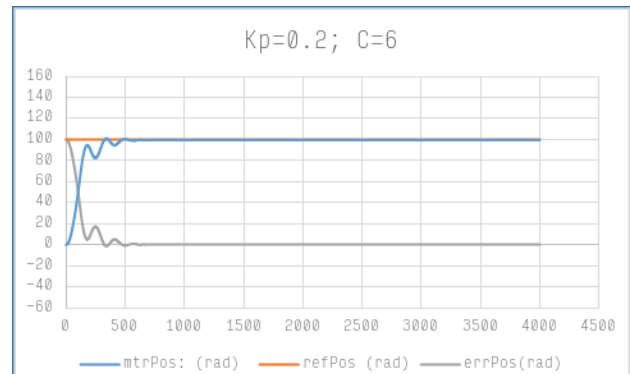


Fig. 6. Static position response curve for the motor.

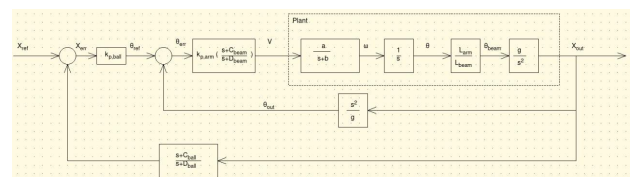


Fig. 7. General block diagram for a Ball-On-Beam system

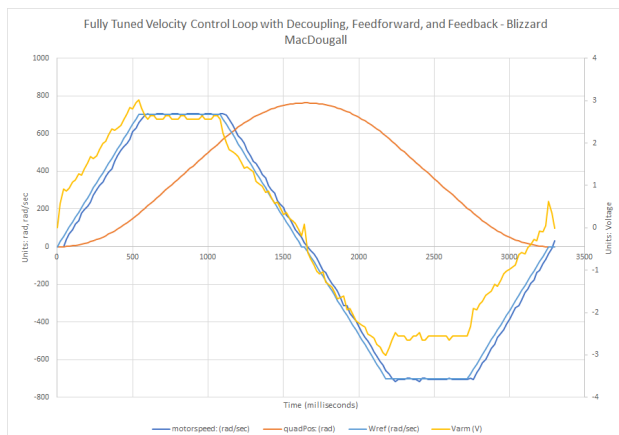


Fig. 4. Trapezoidal reference signal input and output with proportional control.