Laboratory 5 Report

Embedded Control, Rensselaer Polytechnic Institute, Summer 2019
Daniel Alegria, Ankur Singh, and Martin Xu
Section 1, LITEC Side B
6-28-2019

Introduction

Purpose/Objectives

The main focus of Laboratory 5 was for the students to learn and gain experience with controlling the steering and speed of a car based off of the data values an accelerometer is giving. The main goal was that when the car was placed on some point on the ramp and instructed to begin moving, the car should go up or down the ramp until it approaches a flat surface perpendicular to the slope, and then stops.

This lab also had the team gain more experience with proportional gain, especially setting the right gain value for drive gain (how sensitive the drive motor is to tilt in both the x and y axis, independently), and steering gain (how hard the car turns, based on sideways tilt).

Overview of Accelerometer Feedback Control

An accelerometer is an instrument that can measure acceleration, and in this lab the team used an accelerometer to measure the acceleration due to gravity as the accelerometer is rotated through space—pitch, which is rotation about the axis going across the width of the car (which corresponds to the x axis on the accelerometer), and roll, which is rotation about the axis across the length of the car (which corresponds to the y axis).

The accelerometer returns signed 16-bit readings, with negative values being tilt in one direction (relative to zero), and positive values tilt in the other direction. However, because of a large amount of noise, the low byte of the reading is dropped, and only the high byte is used.

The equations used to calculate the pulsewidths for the drive motor and the steering use proportional control; the accelerometer readings are multiplied by a gain value, and the result added or subtracted to the respective neutral pulsewidth of both systems.

For the drive system, there are two sets of pulsewidth equations—one set for forwards, which adds the value calculated from a gain value and the accelerometer reading, and one for backwards, which subtracts the calculated value. Each set has two equations; one takes in the accelerometer x value, and the other takes the y reading.

The steering system similarly has an equation for forwards driving, and an equation that reverses steering for backwards driving. However, the steering system only takes in readings for sideways, x axis tilt.

Results, Plots, Analysis of Plots, & Conclusions

Verification

The required specifications are as follows:

When the run/stop switch is off:

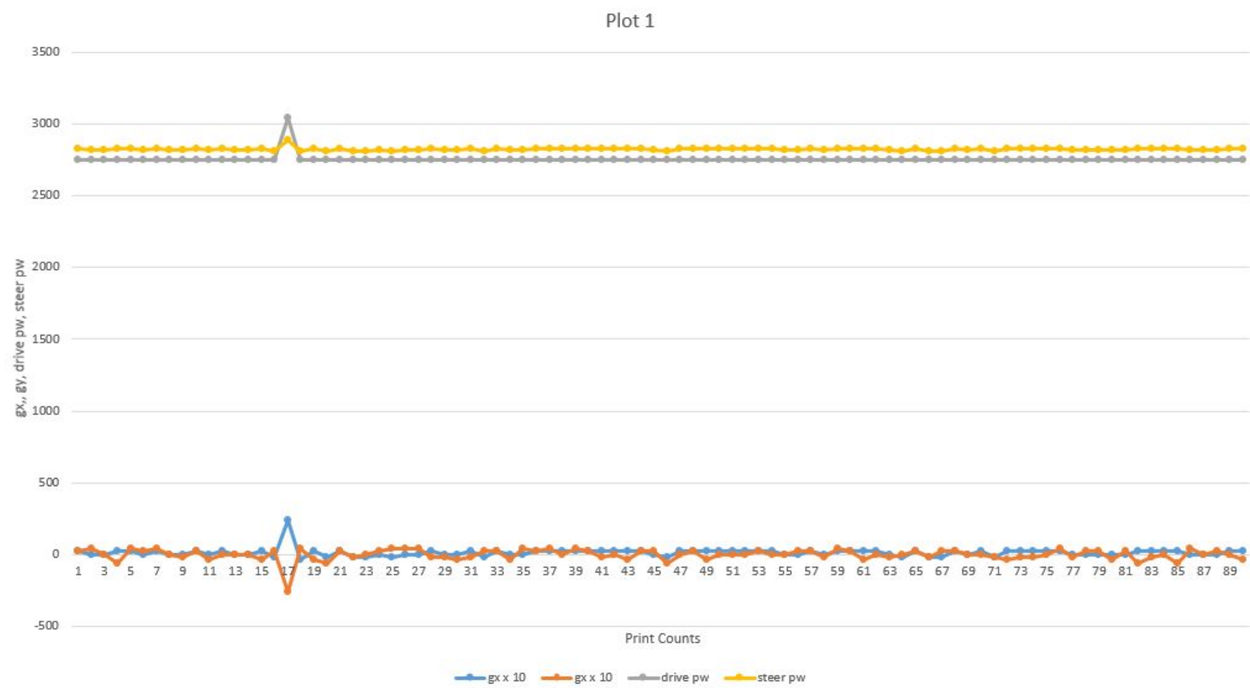
>The user can input the drive x gain, the drive y gain, and the steering gain through the keypad.
>Decide whether the car will go forward or in reverse with a slide switch.
>The car calibrates its current x/y axis tilt orientation as a neutral accelerometer reading.

When the run/stop switch is on,

>If the car is to run forward, it will attempt to steer perpendicular to the slope and drive down it until it reaches a flat surface (usually the ground, as determined by the calibration).
>If the car is to run backward, it will attempt to steer and drive up the slope until it reaches a flat surface (usually the top of the slope).
>A BiLED will shine green when the car is running, and shine red when the car has reached a flat surface, and has stopped driving.

In testing, the team put the car on the ground, with the run/stop switch set to stop, and successfully inputted x gain, y gain, and steering gain values through the keypad. The forward/backward switch was set to forward, and the run/stop switch set to run. The car was then put on the middle of a slope, parallel to the slope, and the car automatically steered and drove down the slope, stopping at the bottom, about perpendicular to the slope. When the car was set to backwards, it steered and backed up the slope, stopping at the top. Multiple angles, positions, and gains on the slope all produced correct results.

Analysis of plots from tabulated terminal data with explanations

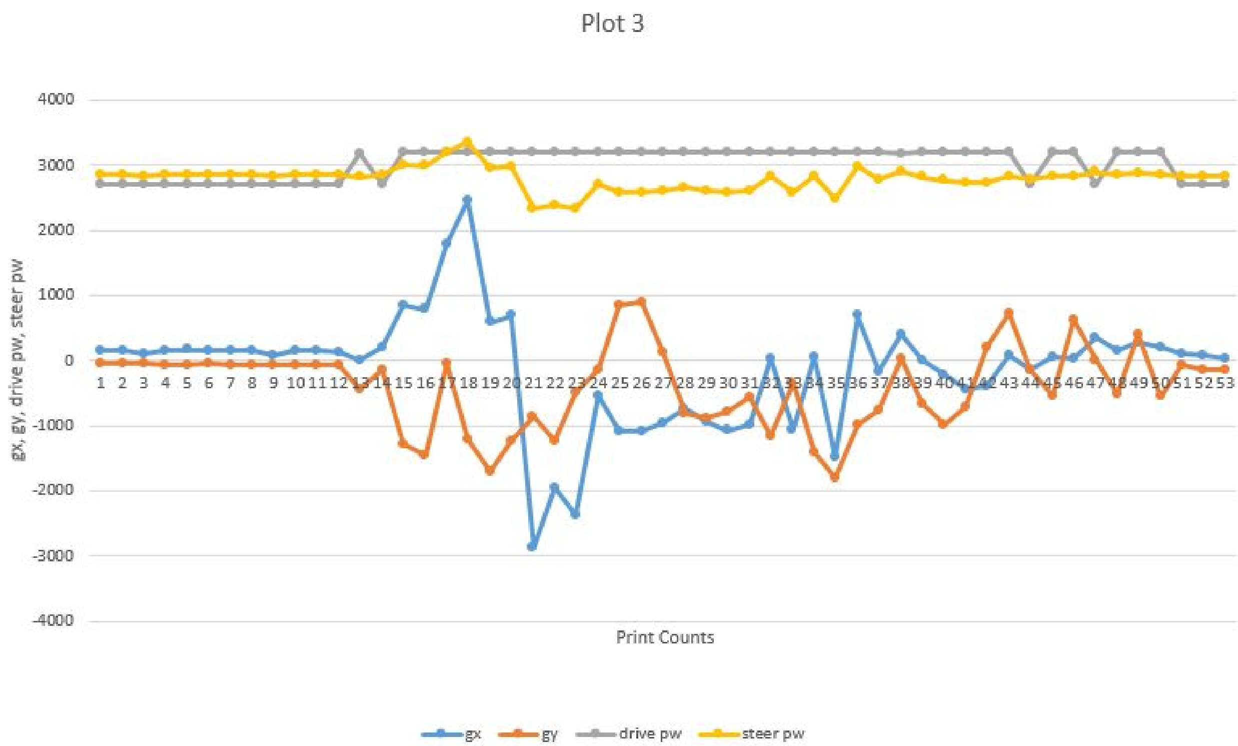


Plot 1: Car on flat ground

Plot 1 shows the default state of the car, the state that the car begins in, and which it strives to reach when running. The accelerometer readings, gx and gy, are 0, and the drive motor pulsewidth and steering pulsewidths are at their neutral state. If gx and gy are moved, the drive and steer pulsewidths will respond accordingly until the car reaches a gx and gy of 0 again.

Plot 2: X gain = 5, Y gain = 5, Steer gain = 10, driving backwards

When driving backwards, the car will attempt to steer up the ramp. From count 0 to count 20, the car was calibrated on flat ground. From count 20 to count 35, the car was picked up, and set on the slope, parallel to the slope. After it was let go, the drive motor pulsewidth was set to the minimum, causing the car to drive backwards. The steering pulsewidth was set to maximum, causing the front wheels to turn right. The car kept at this until around count 80, where it started to approach the peak of the slope. Here, the pulsewidths started to converge to their neutral state, reaching neutral when gx and yx read 0.

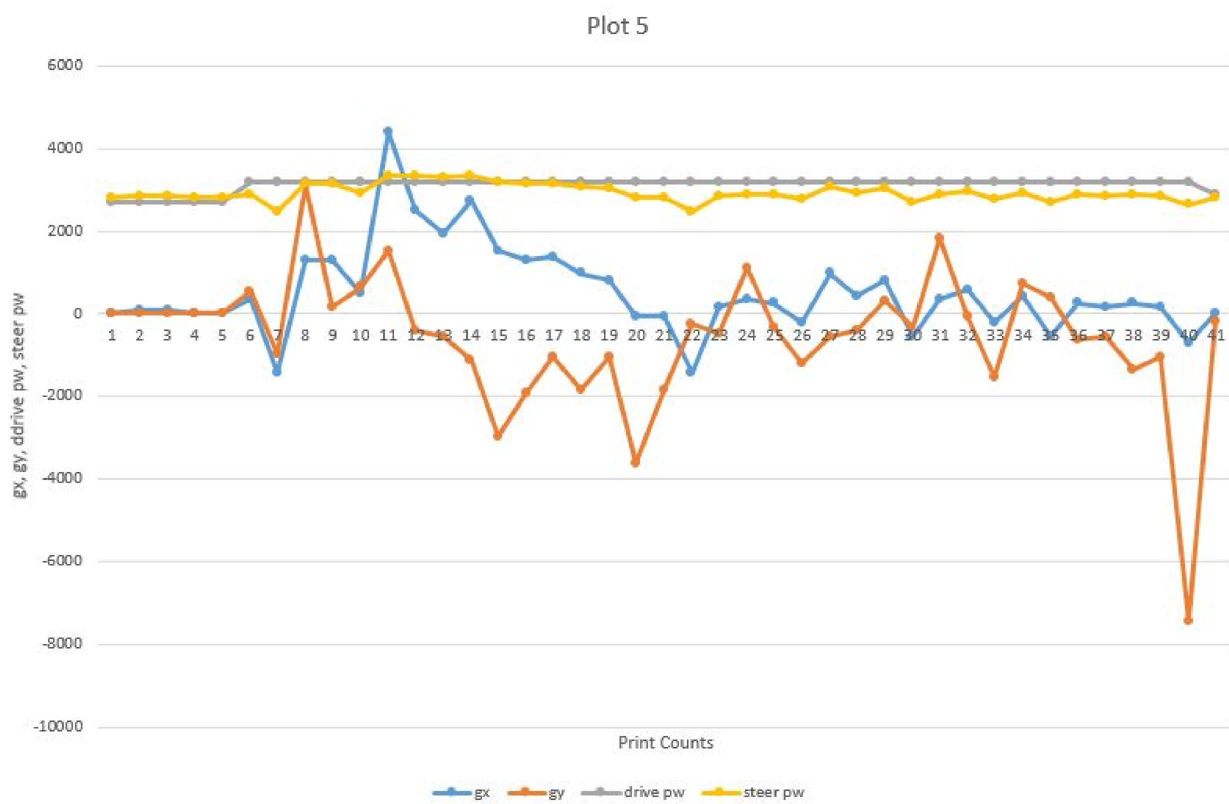


Plot 3: X gain = 10, Y gain = 10, Steer gain = 8, driving forwards

This plot demonstrates a proper run going forwards down the hill at relatively higher gains. The asymptotic trend towards a neutral state (~2700 for the pulsewidth values and 0 for the gx/gy values) is analogous to the car starting at an angle halfway up the ramp and finding level ground using the accelerometer. There are more consistent peaks, especially at the end, because the car is approaching close to flat ground so noise becomes more significant, and is amplified by the higher gain.

Plot 4: X gain = 8, Y gain = 8, Steer gain = 6, driving backwards

The plot above is another good example of the car finding level ground, this time going backwards to the top of the ramp. The gx and gy values here are correlated to the original position of the car, at a steeper part of the ramp but turned relatively close to the final backward position. The gx and gy values still converge to 0 meaning it successfully found level ground; however it is worth noting that the top of the ramp is noticeably less flat than the floor in some places, so depending on the tolerance of the car it may not detect a level ground. Also, many cars do not have a strong reverse gear, so it may not have the power to make it all the way to the top, which explains the three spikes in the drive pulsewidth at the end where it was assisted by a team member.



Plot 5: X gain = 9, Y gain = 9, Steer gain = 9, driving forward, accelerometer readings averaged 2 times

In Plot 5 the car is again going forward down the hill, but this time the accelerometer is only averaging 2 values instead of 8 values. This is why even though the plot maintains a similar shape as of Plot 3, there are more spikes, and the values on the graph are less consistent. This can be attributed to electrical noise—when less values are averaged, electrical noise has a greater potential to skew the data and make it less consistent. This inconsistency lead to more jittery steering. However, the car still ultimately reached the ground, and error converges to 0 while the pulsewidths converged to their neutral states.

Problems Encountered & Solutions

One problem the team initially had was that when the car travelled up the ramp, it would usually stop before actually reaching a flat surface, but the BiLED indicated that it had not stopped running. The team deduced this was because after traveling up the ramp the car would have to encounter friction, which along with gravity are forces which both work against the car's motion. The team decided that the car needed more power, and therefore a higher (or lower, if it were going backwards) pulsewidth to be able to complete its trip up the ramp and onto the flat surface. This was accomplished by modifying the equations which controlled the drive motor pulsewidth so that an extra value of 50 was added to the pulsewidth, giving the motor a small boost even at minimum pulsewidth. This ensured the car would always have a sufficient pulsewidth to finish its route up the slope and onto the flat surface, while still maintaining a linear relationship between the acceleration reading and the drive motor pulsewidth value.

Proper calibration of the accelerometer was another problem the team had to deal with. The first iteration of the calibration formula only took one reading from the accelerometer, so the car did not calibrate correctly due to accelerometer noise. The next iteration averaged 64 readings, leading to successful zeroing on flat ground.

Apart from these two problems, the team did not face any other major problems. Much time was also devoted to finding appropriate proportional gain values and ensuring the code performs the necessary tasks, but apart from this the team was able to complete the lab without too many hiccups.

Suggested improvements to HW & SW

There are few improvements to be made on the hardware due to the simple nature of the wiring. Built off of the Lab 4 wiring, a few modifications were made and none were sophisticated, with the most notable change being the addition of the accelerometer. While there definitely could be more efficient methods of accomplishing the same tasks in terms of wiring, the design the team utilized was neat and effective.

The car itself presented some limitations that the team had to work around—most notably, the drive motor is weak, with low torque at low speeds, and lower torque backwards than forwards. In addition, the accelerometer's high noise meant that a lot of time had to be spent tweaking the tolerances of what the car defines as "level ground".

In terms of software, there likely are a few ways in which the code could have been more elegant and simplified, but the code used in the lab was easy to read and debug, and so got the job done.

Academic Integrity Certification *(this part is required exactly as stated)*

All the undersigned hereby acknowledge that all parts of this laboratory exercise and report, other than what was supplied by the course through handouts, code templates and web-based media, have been developed, written, drawn, etc. by the team. The guidelines in the Embedded Control Lab Manual regarding plagiarism and academic integrity have been read, understood, and followed. This applies to all pseudo-code, actual C code, data acquired by the software submitted as part of this report, all plots and tables generated from the data, and any descriptions documenting the work required by the lab procedure. It is understood that any misrepresentations of this policy will result in a failing grade for the course.

Participation *(this is only a template; make changes as appropriate or necessary)*

The following individual members of the team were responsible for (give percentages of involvement)

Hardware implementation:
(wiring & pin-out sheet)

Daniel Alegria 50
Ankur Singh
Martin Xu 60

Software implementation:
(pseudo-code & code)

Daniel Alegria 45
Ankur Singh 10
Martin Xu 45

Data analysis (if relevant):

Daniel Alegria 33
Ankur Singh 33
Martin Xu 33

Report development & editing*:
(schematic, diagrams & plots)

Daniel Alegria 33
Ankur Singh 33
Martin Xu 33

The following signatures indicate awareness that the above statements are understood and accurate.

Daniel Alegria
Ankur Singh
Martin Xu

*Note, report development/formatting does not constitute an engineering contribution toward successful laboratory completion.