John Jeong, Noah Teal, Tim Regrut

Professor Jolanta Janiszewska

February 2, 2016

AEV Lab 2

Introduction

The purpose of this lab was to familiarize the group with the Arduino and the formatting of the program. The group learned the very basic controls for the AEV and practiced a short list of commands. Throughout the lab, the team ran into a minor problem trying to figure out the syntax of the program, but that was an easy fix. The team also ran into a formatting problem, but Professor Janiszewska pointed out the simple fix, which was just a misplacement of the code.

Experimental Setup

Setup consisted of preconditioning the settings within the Arduino software to be compatible for the specific Arduino board that is used in the AEV project. First, the student had to follow procedures of changing software directions to the course's file sketchbook, where direction and code specifics are located. Then, the student went to the tools tab of the software and changed the board type into "Arduino Nano". Codes were given in the AEV Lab Manual and Instructions that controlled the speed and other functions of the propellers. The previous lab instructed the students to construct their AEV, and the propellers were already attached for testing. Uploading codes into the arduino board required a simple USB to microUSB connection.

Results

During the process of the lab, the propellor motors seemed to work really well without delay, but it was noticed that some other teams had trouble spinning the propellers at low speeds without any assistance.

The commands used in this lab could limit the success of the AEV because although the commands are very simple, they also have drawbacks, such as the "brake(m);" command which stops the propellers but doesn't stop the AEV right away. This wouldn't cause a huge problem, it would just take the AEV a little longer to stop on its own. A simple fix would be to turn the propellers on reverse for a short amount of time to make it slow down faster. The drawback of using the reverse command is that the tourists would not have as smooth of a ride because they would be stopped more abruptly.

The two scenarios are very different and the first scenario would be more practical for the task at hand because scenario two has a lot of unneeded braking and speeding up. Neither scenario is particularly smooth, but one could argue that scenario two is smoother because it slows down in smaller increments compared to scenario one, making for a smoother ride for the tourists.

The team made very few errors, but the scenario worked first try, the only errors were forgotten semicolons at the end of commands, but other team members caught the mistakes as progression continued. The team only completed scenario one because it was very understandable so the extra practice did not feel needed. Scenario two was just extra practice for fun that was supposed to remind the group of Star Wars.

The team did not need much help during the lab, except for when the team could not figure out how to run the code, but it was only being typed in the wrong place, which Professor Janiszewska helped solve. Other than that, the lab was very straightforward and easy to understand, so the team does not have any future recommendations.

Conclusion

During this lab the team was able to familiarize themselves with the various aspects of arduino programming. The team was then able to perform the two scenarios successfully. The team learned that the program with the commands has to be placed in a very specific place otherwise the program will not run, it will not upload to the arduino because it has syntax errors, and the team would have been unable to program their design.

Code (with Comments):

```
// Scenerio 1: Accelerate motor one from start to 15% power in 2.5 seconds

celerate(1,0,15,2.5);

// Scenerio 2: Run motor one at a constant speed (15% power) for 1 second

motorSpeed(1,15);

goFor(1);

// Scenerio 3: Brake motor one

brake(1);

// Scenerio 4: Accelerate motor two from start to 27% power in 4 seconds

celerate(2,0,27,4);

// Scenerio 5: Run motor two at a constant speed (27% power) for 2.7 seconds
```

```
motorSpeed(2,27);

goFor(2.7);

// Scenerio 6: Decelerate motor two to 15% power in 1 second

celerate(2,27,15,1);

// Scenerio 7: Brake motor two

brake(2);

// Scenerio 8: Reverse the direction of only motor 2

reverse(2);

// Scenerio 9: Accelerate all motors from start to 31% power in 2 seconds

celerate(4,0,31,2);

// Scenerio 10: Run all motors at a constant speed of 35% power for 1 second

motorSpeed(4,35);

goFor(1);

// Scenerio 11: Brake motor two but keep motor one running at a constant speed(35% power) for 3 seconds

brake(2);

motorSpeed(1,35);

goFor(3);

// Scenerio 12: Brake all motors for 1 second

brake(4);

goFor(1);

// Scenerio 13: Reverse the direction of motor one

reverse(1);

// Scenerio 14: Accelerate motor one from start to 19% power over 2 seconds

celerate(1,0,19,2);

// Scenerio 15: Run motor two at 35% power while simultaneously running motor one at 19% power for 2 seconds
```

```
motorSpeed(2,35); motorSpeed(1,19);

goFor(2);

// Scenerio 16: Run both motors at a constant speed (19% power) for 2 seconds

motorSpeed(4,19);

goFor(2);

// Scenerio 17: Decelerate both motors to 0% power in 3 seconds

celerate(4,19,0,3);

// Scenerio 18: Brake all motors

brake(4);

// Scenerio 19: Save Program as (Save As:)

//save me
```