

## **Week 10 (Backwards Looking Summary)**

### **Situation**

The objective of lab 9 was for us to improve the codes in order for the AEV to complete its mission; however, our group encountered a massive obstacle. During the first and second run, the AEV flew off the track due to what we suspect as a design flaw. After a short review, the team decided to add a servo motor to act both as a counterbalance and as a braking mechanism to the wheels. Although this idea has yet to be confirmed and finalized, the team is encouraged to find a way to make this a success in order to make the AEV more efficient and stable in the future.

For the codes the team had tested on, instead of goFor(2), goFor(3) will be used in lab 10. The team also discovered that when handed with different batteries, the code goFor(n) will provide a different travel distance. This will also be another problem that the team should review on.

### **Results & Analysis**

This week of AEV testing has been in its majority, trial and error. The group has had some issues with the AEV's consistency in performing the tasks listed in the lab manual. From the starting position the AEV has to move along the first section of the track and then slow down and stop within a couple inches of a sensor along the track. This sensor, when activated, will wait for 7 seconds and then a gate just in front of the sensor will open allowing the AEV to pass through. The group's AEV will have no issue getting to the sensed area but it will often times pass just inches in front of the last sensor and the gate countdown won't start. The group has attributed this flaw in the code of the arduino. Initially the code used the absolutePosition() command to have the AEV move a certain number of marks and then slowed down using the brake() and reverse() command to come to a complete stop. The main issue is with the absolutePosition() command, this command seems to be very inconsistent after dozens of tests and not optimal for the project. Instead the group had decided to use the goFor() command. After just a few test runs the group was able to get the code to within the sensed range multiple times in a row. This allows the group to check this task off and begin working on new code.

The current AEV design and code uses more power than the group would like. Once the power is cut the AEV tends to coast a great distance. This requires some additional code and therefore additional energy to counteract the movement and slow the AEV. As stated before the group was using the absolutePosition() command over the goFor() command this code often had the AEV moving further than expected and using more energy in the process. To solve this issue the group had made the change to the goFor() and also the group began working towards installing a servo motor on the arm of the AEV. The point of the servo motor is to attach a rubbery object to the end of the motor arm that will stop rotate to where it makes contact with the

wheel to slow it down. This is basically designed as a brake. The point of this is completely substitute the energy used in reversing and braking the AEV to just the servomotor. This will allow the AEV to consume less energy because it only requires the servomotor to move the arm 15 degrees instead of slowing a fast moving object through reversing motor power. From Lab 9 the group learned that using the goFor() and servomotor will improve our project and actually make the code easier to expand upon because of the increased consistency.

### **Takeaways**

The first takeaway that the group learned from the lab was to distinguish the differences between “goFor()” and “goToAbsolutePosition()”. “goFor()” command allows AEV to travel certain time and “goToAbsolutePosition()” allows it to travel some certain marks on the track. The group learned that the AEV should be fixed at the same starting position when using “goToAbsolutePosition()” command otherwise it would be hard to tell how much it would travel and if it would right stop at sensored area. In addition, the group had to take the speed of the AEV and the specific “goFor()” command because the group should convert the distance to the specific time. The second takeaway was to learn how to use micro servo. The group found out that micro servo could be used as a brake to control the speed and the weight to balance the AEV as well. The function “rotateServo()” was used to change the angle that the servo turned. The general project learning from this lab was that the group discussion could help group members to know other members’ ideas better. In addition, team collaboration could help the group improve work efficiency. For example, some team members took responsibility for coding and others were in charge of assembly and testing. What is more, the team should be open to try some novelties like the micro servo in order to improve the AEV.

### **Tables & Figures**

Phase	Arduino Code	Distance (inches)	Wait Time(seconds)
1	motorSpeed(4,25); goToAbsolutePosition(261); reverse(4); motorSpeed(4,30); goFor(3); brake(4); goFor(10);	20.56	16.5 (0 -16.5)
2	reverse(4); motorSpeed(4,30); goToRelativePosition(250); reverse(4); motorSpeed(4,30); goFor(1); brake(4); goFor(3);	34.56	20.7 (16.5 - 37.2 )
3	reverse(4); motorSpeed(4,25); goToAbsolutePosition(261); reverse(4); motorSpeed(4,30); goFor(3); brake(4); goFor(10);	20.56	16.5(37.2 - 52.7)
4	reverse(4); motorSpeed(4,30); goToRelativePosition(250); reverse(4); motorSpeed(4,30); goFor(1);	34.56	20.7 (52.7 - 73.4)

---

	brake(4);		
--	-----------	--	--

*Table 1: breakdown of the distances*

## **Week 11 (Forwards Looking Plan)**

### **Situation**

In the upcoming week the team will finalize the design of the AEV with the inclusion of the servo motor. This is important because it will allow the team to move forward into the performance aspects of the vehicle and allow the team to make more consistent stops. The team will accomplish this by performance multiple tests the the servo motor. Also in the upcoming week the team will use the MATLAB program and run an analysis on the performance of the AEV. This is important because it will allow the team to determine which aspects of the vehicle needs to be changed before performing the final run. The team will accomplish this by performing multiple runs with the newly designed AEV, then analyze the data using the scoring sheet the team prepared in previous labs.

### **Weekly Goals**

1. The team will incorporate the use of the servo motor to the AEV design to optimize performance. This will allow the team to be able to stop the AEV for efficiently and move the AEV more efficiently. The team will meet during lab this week to complete the installation.
2. The team will test out the performance components of the AEV to make sure it is operating at the efficiency requirements the team set. This will allow the team to complete the AEV project ahead of schedule. The team will meet during the lab times this week to complete the analysis.

### **Weekly Schedule**

Tasks	Teamates	Start Dates	Due Date	Time Needed
Get AEV to consistently attach with magnet	All	4/3/17	4/3/19	1-2 hours
Redesign code to use goFor command instead of absoluteposition	All	4/3/17	4/3/17	1-2 hours

---

Prepare Oral Presentation Draft	All	4/2/17	4/7/17	2-3 hours
---------------------------------	-----	--------	--------	-----------

## **Appendices**

### **Team Meeting Notes**

Date: 4/2/2017

Time: 4:00 PM -- 6:00 PM (Face-to-Face)

Members Present: Nick Waugh, Marcus Williams, Yinuo Wang, Yao Chong Chow

Location: Room 324, Hitchcock Hall

---

#### Objective:

This week meeting was aimed to summarize the work the group did on the Lab 10 and schedule a plan for the next week. In addition, the group talked about the differences between “goFor()” and “goToAbsolutePosition()” commands.

---

#### To do/ Action Items:

- Questions -(All members)
- Lab 10 -PT3 -(All members)
- Discussion on code -(All members)

---

#### Decisions:

- The group decided to use the micro servo
- The group decided to try both goFor() and goToAbsolutePosition().

---

#### Reflections:

- Yinuo talked about “goFor()” command. He held the view that the group should calculate the time to the gate before using “goFor()” command.
- Nick pointed out that the group should make sure the starting position when using “goToAbsolutePosition()” command.
- Marcus thought that the micro servo could balance the AEV since it tilted backwards.
- Yao thought that the micro servo could be used as a brake in order to make sure that the AEV did not slide too long.
- We need to make sure the AEV could run consistently.

## Arduino code

### Code #1

```
motorSpeed(4,25);           //run all motors at a speed of 25%
goToAbsolutePosition(261);   //run it until AEV travels 261 total wheel counts
reverse(4);                  //reverse all motors
motorSpeed(4,30);           //run all motors at a speed of 30%
goFor(3);                    //run it for 3 second
brake(4);                    // brake all motors
goFor(10);                   //run it for 10 second
//After the gate is open
reverse(4);                  //reverse all motors
motorSpeed(4,30);           // Set all motors to 30%
goToRelativePosition(250);   //run it until AEV travels 250 more total wheel counts
reverse(4);                  //reverse all motors
motorSpeed(4,30);           //run all motors at a speed of 30%
goFor(1);                    //run it for 1 second
brake(4);                    // brake all motors
goFor(3);                    //run it for 3 seconds
//After attaching the magnet
reverse(4);                  //reverse all motors
motorSpeed(4,25);           //run all motors at a speed of 25%
goToAbsolutePosition(261);   //run it until AEV travels 261 total wheel counts
reverse(4);                  //reverse all motors
motorSpeed(4,30);           //run all motors at a speed of 30%
goFor(3);                    //run it for 3 second
brake(4);                    // brake all motors
goFor(10);                   //run it for 10 second
//After the gate is open again
reverse(4);                  //reverse all motors
motorSpeed(4,30);           // Set all motors to 30%
```

```
goToRelativePosition(250);    //run it until AEV travels 250 more total wheel counts
reverse(4);                   //reverse all motors
motorSpeed(4,30);             //run all motors at a speed of 30%
goFor(1);                     //run it for 1 second
brake(4);                     // brake all motors
```

## Code #2

```
motorSpeed(4,25);             //run all motors at a speed of 25%
goFor(13);                     //run it for 13 seconds
reverse(4);                     //reverse all motors
motorSpeed(4,30);             //run all motors at a speed of 30%
goFor(3);                      //run it for 3 seconds
brake(4);                      // brake all motors
goFor(10);                     //run it for 10 seconds
//After the gate is open
reverse(4);                     //reverse all motors
motorSpeed(4,30);             // Set all motors to 30%
goFor(15);                     //run it for 15 seconds
reverse(4);                     //reverse all motors
motorSpeed(4,30);             //run all motors at a speed of 30%
goFor(1);                      //run it for 1 second
brake(4);                      // brake all motors
goFor(2);                      //run it for 2 seconds
//After attaching the magnet
reverse(4);                     //reverse all motors
motorSpeed(4,25);             //run all motors at a speed of 25%
goFor(13);                     //run it for 13 seconds
reverse(4);                     //reverse all motors
motorSpeed(4,30);             //run all motors at a speed of 30%
goFor(3);                      //run it for 3 seconds
brake(4);                      // brake all motors
goFor(10);                     //run it for 10 seconds
//After the gate is open
motorSpeed(4,30);             // Set all motors to 30%
goFor(15);                     //run it for 15 seconds
```

---

```
reverse(4);           //reverse all motors
motorSpeed(4,30);     //run all motors at a speed of 30%
goFor(1);             //run it for 1 second
brake(4);             // brake all motors
```