Preliminary Design Report

Submitted to:
Inst. Dick Busick
GTA Melissa Hrivnak

Created by:
Team O

Melanie Gross
Katie Gonsoulin
Jennifer Bertrand
Jessica Hudak

Engineering 1182
The Ohio State University
Columbus, OH
27 March 2017

Executive Summary

After the destruction of the Death Star, the galactic empire is working to rebuild their army. In order to make sure that the galactic empire does not suspect any activity, the rebel alliance must prepare for war on remote planets, where power is limited. The alliance is searching for an efficient yet effective monorail network system that can transport constructed R2D2 units from one side of the land to where the interceptor aircrafts are being assembled. Because the AEV mission takes places on a remote planet where power is a luxury, the AEV will focus on energy management, operational efficiency, and operational consistency. The objective is to successfully pick up and deliver cargo stored on a caboose while minimizing the energy to mass ratio and meeting the operational requirements and design constraints.

The AEV must start at the drop-off area, travel to the gate to be checked in, activate a sensor, wait seven seconds to go through the gate, navigate to the cargo area, and then stop at the cargo area. The AEV must then pick up an R2D2, wait five seconds to make sure that the cargo is loaded, and then travel back to the drop-off area. Since the planet is remote and unstable, the monorail may have slight variations over time, so the AEV should not depend on one specific track. Additionally, the AEV should have operational consistency so that the AEV performs consistently no matter what type of cargo it must carry. As contractors, the team was asked to create an efficient AEV to perform the Scenario. The team will present this final product to the rebel alliance.

The team began to work on the mission by constructing a concept AEV and familiarizing themselves with the external sensor hardware components, troubleshooting techniques, and various program function calls. The members then exercised their creativity skills as they each brainstormed initial AEV designs. The members shared their individual designs with the rest of the team, performed Concept Screening and Scoring to clearly outline the strengths and weaknesses of each design, and utilized the results to create the overall best AEV design. The team evaluated the efficiency of one of the top brainstormed designs by programming the constructed AEV with a specific scenario and testing it on straight tracks. The team used a design analysis tool in MATLAB to extract the data that the arduino collected during the run. This tool allowed the team to analyze the data, understand the level of efficiency of the assembled AEV, and brainstorm additional modifications to improve the design.

The team will continue to refine the code and improve the overall design of the AEV. The team will pay particular attention to making the AEV balanced, aerodynamic, and as energy efficient as possible. Another important area that the team will focus on is achieving consistency. The team will work to make the AEV travel and stop at the appropriate places and times, as laid out in the Scenario. As the team works on these aforementioned tasks, the team will develop an efficient product to eventually present to the rebel alliance.

Table of Contents

Introduction 3

Experimental Methodology 3

Results 3

Discussion 5

Conclusion & Recommendations 8

Appendix 10

List of Tables and Figures

Table 1: Projected Team Schedule 10

Figure 1: Test Run One: Start to Gate Lab 8A 11

Figure 2: Test Run 1B: Start to Gate Lab 8A 11

Figure 3: Lab 8A Test Run in 1 Phase 12

Table 2: Lab 8A Run 1 Phase Energy Breakdown 12

Figure 4: Test Run 2A: Start to Gate Lab 8B 13

Figure 5: Test Run 2B: State to Gate Lab 8B 13

Figure 6: Lab 8B Test Run 2 in Phases 14

Table 3: Lab 8B Run 2 Phase Energy Breakdown 14

Table 4: Concept Screening Scoresheet 15

Table 5: Concept Scoring Matrix 15

Figure 7: Parallel AEV Design Top View 16

Figure 8: Parallel AEV Design Front View 16

Figure 9: Vertical AEV Design Top View 17

Figure 10: Vertical AEV Design Front View 17

Figure 11: Projected Team Schedule Timeline 18

Figure 12: AEV Track Layout 18

Figure 13: AEV Track Layout and Beginning Gate Specifics 19

Figure 14: 3030 Second Pusher Propeller Advance Ratio vs Propulsion Efficiency Graph 19

Figure 15: 2510 Puller Propeller Advance Ratio vs Propeller Efficiency Graph 20

Table 6: Wind Tunnel Data Analysis First 3030 Pusher 20

Figure 16: Primary Orthographic Views and Dimensions of Parallel AEV Design 21

Figure 17: Bill of Materials for Parallel AEV Design 22

Figure 18: Primary Orthographic Views and Dimensions of Vertical AEV Design 23

Figure 19: Bill of Materials for Vertical AEV Design 24

Introduction

The galactic empire is working to rebuild their army after the destruction of the Death Star. To make sure that the galactic empire is unaware of any activity, the rebel alliance must prepare for war on remote planets, where power is limited. The alliance needs an efficient monorail network system that can transport constructed R2D2 units from one side of the land to where the interceptor aircrafts are being assembled. Since power is a luxury on the remote planet where the AEV mission is taking place, the AEV will focus on energy management, operational efficiency, and operational consistency.

The objective is to have the AEV start at the drop-off area, travel to the gate, activate a sensor, wait seven seconds to go through the gate, go to the cargo area, and then stop at the cargo area. The AEV must then pick up an R2D2, wait five seconds to make sure that the cargo is loaded, and then travel back to the drop-off area. Because the planet is unstable, the monorail may have slight variations over time. Therefore, the AEV should not depend on one specific track. Also, the AEV should have operational consistency so that the AEV performs consistently no matter what type of cargo it must carry. As contractors, it is the team's job to create an efficient AEV to perform the Scenario. The team will present the final product to the rebel alliance.

Experimental Methodology

The team began working on the mission by assembling a concept AEV and familiarizing themselves with the various hardware components, troubleshooting techniques, and numerous program function calls. After becoming familiar with working with the concept AEV, each member brainstormed initial AEV designs. The members then shared their individual designs with the rest of the team, and the team performed Concept Screening and Scoring to clearly represent the strengths and weaknesses of each design. The team gathered the highlights from the brainstormed ideas to create improved designs that efficiently and effectively met the requirements and goals from the Mission Concept Review. The team evaluated the efficiency of one of the top brainstormed designs by programming the constructed AEV with a specific scenario and testing it on straight tracks. The team got familiar with utilizing a design analysis tool in MATLAB to extract and analyze the data that the arduino collected during the run.

In Performance Test 1 (Labs 08A/B/C), the team tested and compared two different AEV designs. Each AEV was assembled, programmed with a certain code, and tested on the outside track. After the test runs, the team extracted and analyzed the data (that the arduino collected) with the help of the design analysis tool in MATLAB. This tool allowed the team to visualize and understand how efficiently each AEV was employing the supplied power during the run. The team additionally developed a code that satisfied the goal mentioned in the Mission Concept Review of having the AEV arrive at and stop at the first gate for seven seconds before proceeding through it.

Results

The first two labs focused on introducing the team to the AEV and all the materials and codes that go along with that. The team began brainstorming ideas for ideas for the AEV design and set up team communication links in order to maximize success over the course of the project. Next, the team learned how to use and write basic code commands for the Arduino. A set of instructions was given and the team translated this into code, loaded it onto the Arduino, and ran it to see how the code will translate into real actions.

The next two labs included learning about the reflectance sensors, investigating propulsion efficiency, and discovering the creative thinking process. The team ran tests with basic code written to see how the reflectance sensors worked and how they could be used to control the motion of the AEV. Next, tests were performed on different propeller configurations, pusher vs. puller and 2.5in vs. 3in blade diameter. It was determined that the pusher configuration with the 3in was the most energy efficient configuration to use, compared to the puller configuration or the 2.5in blade. The team then proceeded to come up with individual AEV designs and compare these designs to construct an initial team design.

Labs 4 and 5 involved extracting data collected on the Arduino to calculate information more useful to developing the AEV programming code and scoring different AEV designs to determine which design to continue to develop. The Arduino collected data in ADC counts, such as time, current, voltage, and marks, which the team converted into physical parameters so it was easier to interpret. These converted measurements, such as amps, volts, and meters, allowed the team to graph the results of the run to determine the energy used, as seen in Figures 1 through 6 in the appendix. Next, the different AEV designs were screened and scored on two different matrices to determine which design to pursue. The initial screening process used +, -, and 0 to determine which designs to immediately trash, which to combine into one design, and which to keep for the time being. The scoring process weighted the different grading criteria based on importance and determined the 'Team' design was worth developing more with the highest calculated score of 3.6.

The final three labs included an open work day to work on whatever the team needed, determining the error in the wheel count sensors, calculating the propeller and friction force, and testing two different AEV designs to see which was more energy efficient. The team took advantage of the open lab to develop plans and code for a design the included the use of a Servo motor to rotate the body of the AEV. Next, in order to determine the error in the wheel count sensors and calculate the propeller and friction force, the AEV was run on the straight track so the distance travelled could be manually measured. The wheel count sensors were calculated to have an error of 7 marks and the AEV has a propeller force of 9.4gmf and friction force of 2.5gmf, resulting in a net force of 6.9gmf. Finally, two different designs, one with a parallel body and one with a vertical body, were tested to determine which was more energy efficient. It was discovered the parallel body design was more energy efficient using 67.4137J of energy versus the vertical body design which used 68.5575J of energy.

The first team AEV design, as seen in Figures 7 and 8 in the appendix, featured two propellers coming off the back and a T-shaped body parallel to the ground. This design had the battery secured under the the body with the Arduino above it on the top side of the body and the arm behind the Arduino. A combination of plastic and metal clips were placed on the front of the AEV to be used to pick up the R2D2 unit during the run. The second AEV design, as seen in Figures 9 and 10 in the appendix, is very similar to the first design. All the major components, such as the battery and propellers, are located in the same position as in the first design. The major difference between these two designs is the positioning of the body. The first design featured a body parallel to the ground while the second design flipped the body vertical so it was perpendicular to the ground. The parallel design was developed from the four designs created in Lab 4. All four designs were quite similar in the general design, so the best elements of each were pulled and combined to create the parallel design. The vertical design came after Lab 4 as the group wondered if there was a way to shift the center of mass of the AEV to allow for better handling around the corners of the track.

The screening and scoring matrix, Tables 3 and 4, respectively, in the appendix, was used to support why each design by the team. The initial screening used a +,-, and 0 system of measurement to determine if different aspects of the AEV would help or hurt the finished product. The results of the screening process then funneled into the scoring process which weighted the categories being assessed. The design that was picked to continue to develop (the team design) scored highest in both testing processes.

The two concepts were very similar while running with the same control program. The team expected to see a greater difference in balance and amount the AEV swayed while on the track. While the vertical design did show slight increase stability, the change did not seem noticeable enough on the code being run to switch the orientation of the body. There was not a significant difference in energy consumed by the two concepts. As found in Tables 2 and 3, the parallel design used 67.4137J of energy and the vertical design used 68.5575J of energy, a difference of only 1.1438J. The group was not expecting a large difference in energy consumption. The main reason for switching the orientation of the body was to help with the swaying of the AEV, which was especially important when picking up the R2D2 unit. The vertical design did not prove to improve the overall functionality enough to compensate for the excess energy used.

The Performance Test 1 slightly altered the team's design process. After completing this test, it allowed the team to rule out another design of the AEV. The team had settled upon the parallel design after performing the screening and scoring tests. After seeing some other groups changing the orientation of the body, the team also wondered if switching the direction of the body would improve the functionality of the vehicle. By working through the performance test and comparing the energy usage of each design, it allowed the team to put numbers to previous inquiries and make a definitive decision regarding the usage of different design features, mostly the orientation of the body. In System Analysis Test 1, the team ran tests on the AEV to Figure out how the wheel reflector sensors worked and tested different propeller configurations in a wind tunnel to see how that affected efficiency of the AEV. Knowing this information, the team was able to determine how far the AEV will travel using a command like goToAbsolutePosition(), instead of basing the movements on time. In System Analysis Test 2 the team discovered how to extract data from the Arduino and run it through a MATLAB program to take the collected data and calculate other values useful for analyzing the energy efficiency of the AEV. By having the understanding and programs to take raw data and calculate into more useful information, the team was able to analyze the AEV in a different way than before. The team was able to break up the AEV's runs into different phases, see how much energy each command used, and implement this gathered information when writing a new code that is more energy efficient.

Discussion

The track on which the AEV ran consisted of a full inner and outer loop each divided into two sections. Each team was responsible for successfully designing the AEV to complete one of these sections, or one half of the track (Figure 12). Each AEV run consisted of the AEV starting at the beginning and then traveling to the gate located at the halfway point of the team's assigned section (Figure 13). The AEV had to trigger the first sensor at the gate and pause for seven seconds. The first sensor was located 11.25 inches from the gate, allowing a buffer region of approximately eight inches in which the AEV could stop before the gate. After stopping at the gate, the AEV continued the run on the remaining half of the assigned portion of track until it reached the loading area. This area was where the cargo was located for the AEV to pick up. Once the cargo was successfully loaded, the AEV had to wait for five seconds and then begin the journey back along the track, stopping again at the gate for seven seconds. Once the gate was lifted, the AEV traveled the remaining portion back to the starting location in order to successfully complete the run.

Of the first two labs, the bulk of the work involved writing and testing new commands for the AEV code. Commands such as celerate(m,p1 ,p2 ,t) and goFor(t) were written and observed to complete exactly what they were intended to do. While the program did take a few seconds to actually start running on the AEV, there were no further complications with the program and what actually happened when it was run. The team discovered while some commands like brake(m) and motorspeed(m,p) may immediately cause changes in the motors, the motion of the AEV is not as sudden of a change.

The next two labs mainly tested the variation in propeller configurations and taught about the creative design process. Based on the testing that was performed in this portion of lab, the team decided to use the puller configuration and 3in blade on both concept designs. When looking at the propulsion efficiency versus advance ratio, the 3030 setup (3in blade, Figure 14) had a steeper slope on the graph than the 2510 setup (2.5in blade, Figure 15), meaning it was more efficient. Also, the pusher configuration proved to be better than the puller configuration since it generated more thrust and had higher power output values, as seen in Table 6 and 7. As for the creative design process, the team took an open approach to this by clearly communicating the strengths and concerns found in each design. While working through this process, the team was able to talk through everyone's ideas and combine many different options into one final design that seemed to address all of the important aspects that the AEV must deal with while running on the track.

The following two labs focused on converting the data collected by the Arduino into useful information and finding a way to score the concept designs to determine which is the best to continue to develop. The Arduino collects data for current, voltage, time, and distance. A MATLAB program was written to convert these values into amps, volts, meters, and seconds. This data was plotted to see the energy consumption over time or distance, Figures 1 and 4 in the appendix. The graphs were then divided into phases based on the different commands in the code, Figure 6. By constructing these plots, the team was able to easily visualize how much energy was being consumed during each part of the run. While the overall energy could be calculated, using the graphs provided a more detailed description of when the energy was being used. The parallel design, Figure 1, used less total energy versus the run of the vertical design, Figure 6, therefore the group decided to continue developing the parallel design. The second section scored the AEV designs on preliminary categories and then again with weighted categories. The reason for splitting the scoring into two sections was to eliminate the excessively poor design before wasting time performing more tests on them. The first screening was general and helped separate the good designs from the bad. The scoring process allowed the team to decide what was important to incorporate into the AEV design and implement those categories when selecting the best design.

Next came a open lab, determining the error in the wheel count sensors, and calculating the propeller and friction force. During the open lab, the team worked on developing the program to incorporate the Servo motor into the design of the AEV. By running tests on the Servo and how it moved based on the code it was running on, the team was able to determine what commands need to be called to position the Servo in the proper direction so it will move properly while running with the AEV. The error in the wheel count sensors was then determined. By knowing the magnitude of this error, the team is able to incorporate this into the calculations while determining the distances the AEV must travel. Also, the propeller and friction force were calculated using data from the same run. Having a value for the force of friction will let the team to determine when to cut the power to the motors to allow the AEV to coast to a stop instead of alternative braking methods, After these two forces were calculated, the net force was then determined. Knowing this net force will help the team when writing code for the AEV so it is known how a certain motorspeed will impact the velocity of the vehicle.

In lab 8A, the team tested the first AEV design. This design included a body parallel to the ground with a perpendicular arm. This differed from the second design in that the second design was entirely vertical (testing in lab 8B) (Figures 7,8,9,10). The team developed a code that successfully stopped the AEV close to the gate and collected data from these runs (see appendix). In order to accomplish this task, the motors were reversed at a high percentage power for half of a second in order to bring the AEV to a swift halt. The AEV traveled at a constant speed for a designated amount of time and then this surge of energy was implemented to stop the vehicle (Table 2). While this method was effective, the surge of energy required to stop the vehicle was large (Figure 1,2,3). The largest quantity of energy used was in the segment where the AEV was traveling at a constant speed in the beginning, which consumed about 58 joules of energy, while the power surge of energy to stop the vehicle only consumed about 9 joules of energy. In total, the AEV consumed about 67.4 joules of energy in completing the track up to the first gate (Table 2).

In lab 8B, the team tested the second design (Figures 9,10) in which the entire AEV was vertically oriented. In comparison to the design tested in lab 8A, the two vehicles presented only minor energy differences. The same strategy was used in which the AEV traveled at a constant speed until it was close to the gate at which point a surge of energy was used to stop the vehicle. The constant speed phase of the AEV consumed about 60.6 joules of energy while the surge of power to stop the vehicle used about 7.9 joules of energy (Table 3). The energy consumption of the vehicle from this run was about 68.5 joules, which was about 1 joule higher than the first design. While the results were very similar, the first design was chosen to move forward with due to having slightly less energy consumption. Since only one quarter of the track was completed, the difference in energy would be cumulative as more segments of the track were added onto the journey. Therefore, the first design would end up being more energy efficient than the second design.

In labs 8A and 8B, the team's reflectance sensors did not work on the track due to confusion with which way the wheels should be spinning. To get past the issue, the team used a code that relied on time rather than position of the AEV. Quickly realizing that this method was highly ineffective, the team did some troubleshooting on the reflectance sensors, isolated the issue, and resolved the problem. For lab 8C, a code was successfully developed that relied on the position of the AEV rather than on the time. However, due to time constraints caused by the reflectance sensor issues, the team was unable to download any data from a full run based on the position-reliant code. This code was written to complete a full track, and the AEV successfully completed half of the track before issues arose that prevented the data from being downloaded. The code for the first and second run of lab 8C can be found in the appendix.

A complete description of the two concept AEV designs can be found in the Results section. The parallel body design was selected over the vertical body since it was more energy efficient when running on the same code (see Tables 2 and 3 in the appendix to compare the values of the total energy consumed during each run). Another design feature that was decided on was how many motors and propellers to use. Design Melanie (in Table 4) only featured one motor, so it received a "-" in the thrust/power category. The team decided to instead use two motors and propellers. A third decision made regarding the design of the AEV was where the Arduino, battery, and arm were attached to the body. Design Team (also in Table 4) kept all of these elements close together, improving the center of gravity of the AEV. This better center of gravity was one reason why the team chose to continue to pursue the parallel design.

There were several sources of error that contributed to the inaccuracies in the lab. One such error was the inaccuracies in the external sensor readings. In lab 7, tests were run to determine the accuracy of the sensor readings by comparing the calculated value of how far the AEV should travel to how far it actually traveled. It was determined that at a motor speed of 30%, the AEV had an error of seven marks. The AEV should have traveled a calculated 721 marks, but it only traveled 714 marks. This error affected the AEV runs and had to be accounted for in subsequent runs in order to calculate an accurate location of where to stop the AEV.

An additional source of error was due to inconsistencies of the track. Due to the differing rooms, the tracks had slight differences in their layout. The distances differed by small degrees and there were small bumps in some of the tracks. Due to these differences, errors were produced in the AEV runs that had to be discovered and corrected. A code was developed for the upstairs track versus the downstairs track to minimize inconsistencies and preserve the integrity of the code.

A final source of error was the charge of the batteries. If the battery acquired was not fully charged, the AEV did not move as fast as it would have with a fully charged battery. Because of the major energy differences due to this source of error, it was essential to make sure that a fully charged battery was obtained less the code become drastically different in order to compensate for the lack of energy. If the battery was near capacity but not quite fully charged (as was sometimes the case after multiple trial runs of the AEV), the AEV did not always make it to the precise location as was specified. Minor adjustments were made in the code to compensate for this error.

Conclusion and Recommendations

After thorough testing and analysis, the overall team design (Figures 7,8) was determined to be the one that will be continued and further developed for the remainder of the lab. A few modifications may be added as the lab proceeds, but the basic design will remain the same. A 3D-printed part was designed by the group and recently obtained. However, the part was not delivered in sufficient time to test and collect data with the part incorporated into the design. Future tests will be run incorporating this part into the design to see how the overall efficiency compares to the team's current design.

The current design (Figure 7,8), had the most positive features based on the concept screening and scoring matrices (Tables 4,5). Additionally, this design was the most energy efficient based on the data collected from sample runs. This design had an overall energy consumption of about 67 joules for one quarter of the track as opposed to 68 joules of energy for the other design (Tables 2,3). While this difference was small, the test only included one quarter of the track. By extrapolation, the energy gap would increase as the length of the run increased.

While the team successfully obtained enough experimental results to select a design to proceed with, there were several portions of the lab in which more work needs to be completed. Due to issues with the external reflectance sensors in labs 8A and 8B, the code was written based on timing rather than on positions. This was an inaccurate method for stopping the AEV at the correct position. It was eventually discovered that the external sensors were plugged in backwards and therefore never stopped the AEV at the position written in the code. For this reason, a large portion of time was devoted to troubleshooting the code and resolving the error with the reflectance sensors. This error was effectively resolved and a code was developed that relied on positioning rather than time. However, there was inefficient lab time to test this code and obtain data results to analyze.

Additionally, the team did not recieve an integral part of the AEV design until lab 9A due to the high volume of parts being printed on the 3D printer. Since the design of the AEV that the team wished to proceed with heavily relied on this part, the tests were run with another AEV design rather than the desired design (Figures 7,8). Further tests will have to be completed for the design in which the 3D printed part is incorporated in order to determine the energy efficiency of the design and compare it to the current design. A final design will then be selected to proceed with.

In the upcoming labs, the team will ensure all the components are correctly assembled in order to reduce error and create a working concept. The team will run tests on the new design (now that all of the components have been obtained) and collect numerical data to compare the new design's efficiency to the previously obtained data. Multiple codes will be developed to test the efficiency of the code now that data has been obtained for the physical designs.

Appendix

Table 1: Projected Team Schedule

Task Mode	Task Name	Duration	Start	Finish	% Complete	Jennifer Bertrand	Melanie Gross	Katie Gonsoulin	Jessica Hudak
	Write Code for Testing	11 days	Thu 3/9/17	Thu 3/23/17	75%		30 minutes		
	3D Printed Part Design	3 days	Wed 3/8/17	Fri 3/10/17	75%		1 hour		
	AEV Design 1 Build	11 days	Thu 3/9/17	Thu 3/23/17	0%	30 minutes	30 minutes		
	AEV Design 1 Test	1 day	Thu 3/23/17	Thu 3/23/17	0%				
	AEV Design 2 Build	11 days	Thu 3/9/17	Thu 3/23/17	0%				
	AEV Design 2 Test	11 days	Thu 3/23/17	Thu 4/6/17	0%				
	Progress Report Lab 9	3 days	Thu 3/23/17	Mon 3/27/17	0%				
	Write Code 1 for Testing	11 days	Thu 3/9/17	Thu 3/23/17	0%				
	AEV Final Design Completion	11 days	Thu 3/9/17	Thu 3/23/17	0%				
	Write Code 2 for Testing	11 days	Thu 3/9/17	Thu 3/23/17	0%				
	Test Code 1 and 2	1 day	Thu 3/23/17	Thu 3/23/17	0%				
	Progress Report Lab 10	3 days	Thu 3/23/17	Mon 3/27/17	0%				
	Performance Test 3 (test smaller portions of chosen code on chosen design)	1 day	Thu 3/30/17	Thu 3/30/17	0%				
	Progress Report Lab 11	3 days	Thu 3/30/17	Mon 4/3/17	0%				
	Final Testing	1 day	Thu 4/6/17	Thu 4/6/17	0%				
	Update Website	21 days	Thu 3/9/17	Thu 4/6/17	0%				
	Team Meeting Notes	21 days	Thu 3/9/17	Thu 4/6/17	0%				
	Critical Design Review Report	6 days	Thu 4/6/17	Thu 4/13/17	0%				
	CDR Oral Presentation	11 days	Thu 4/6/17	Thu 4/20/17	0%				

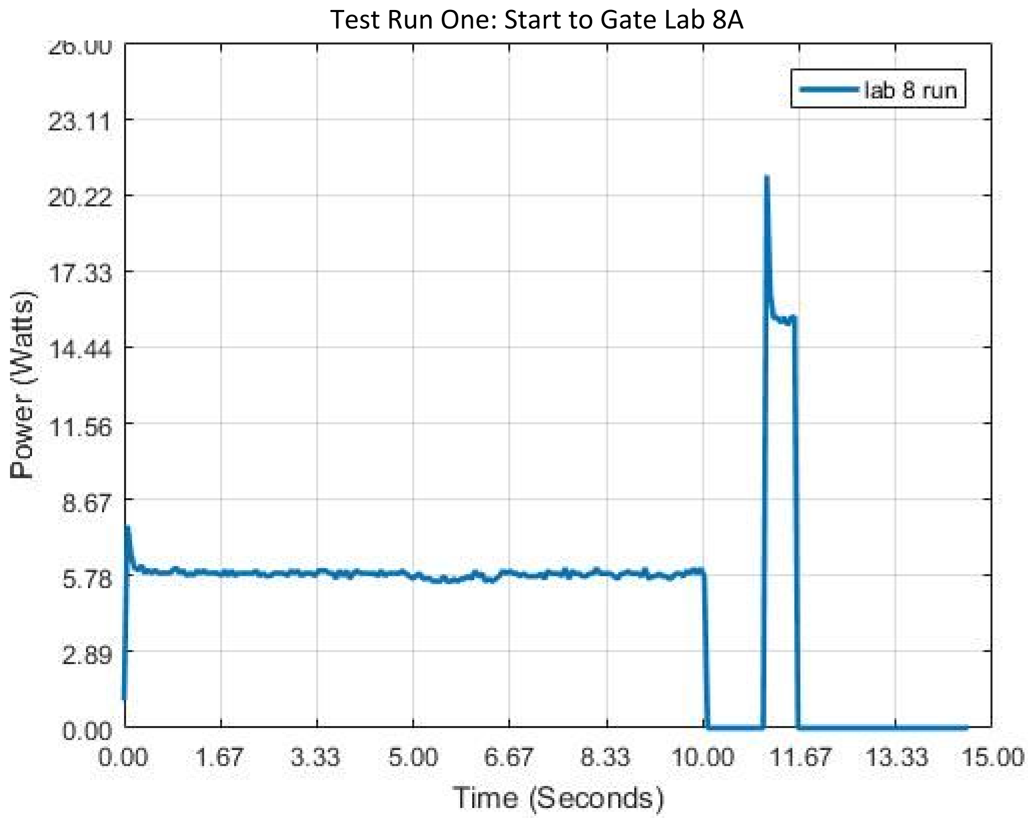


Figure 1: Test Run One: Start to Gate Lab 8A

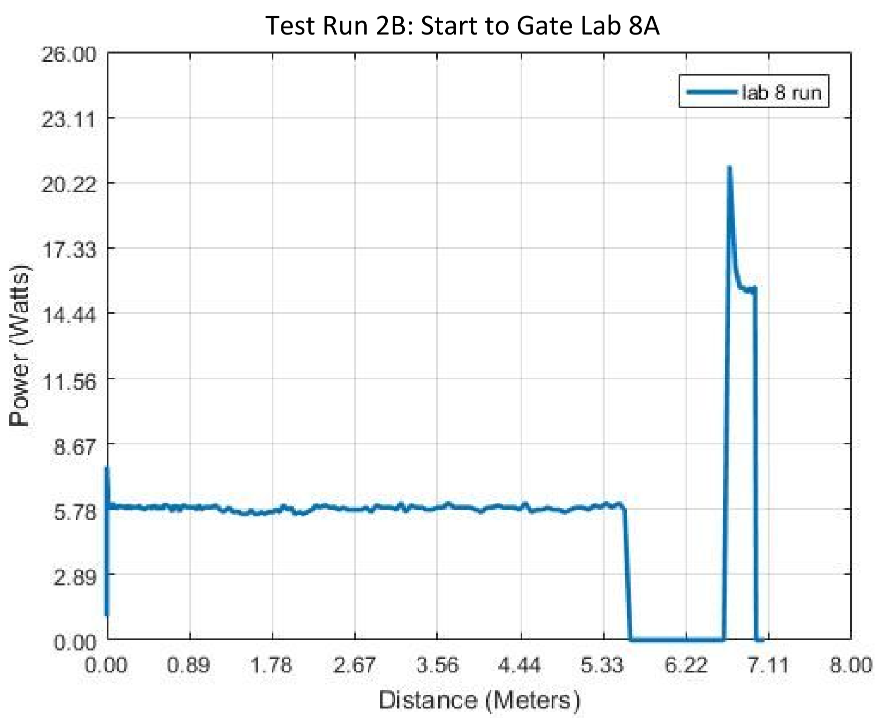


Figure 2: Test Run 1B: Start to Gate Lab 8A

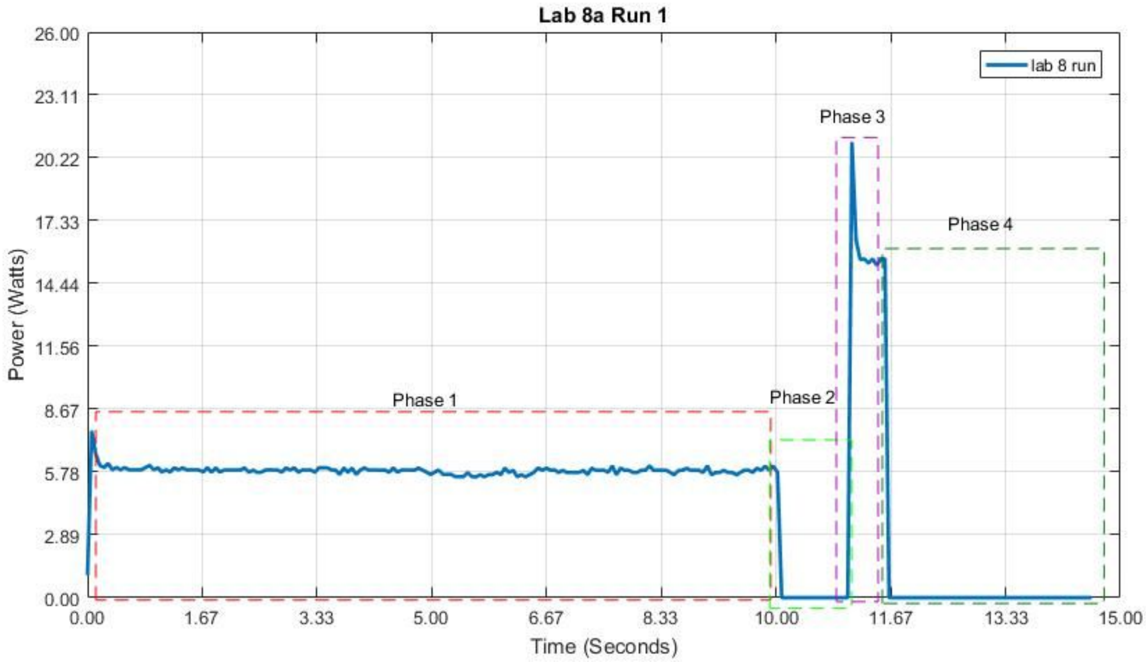


Figure 3: Lab 8A Test Run 1 in Phases

Table 2: Lab 8A Run 1 Phase Energy Breakdown

Phase	Arduino Code	Time(seconds)	Total Energy (joules)

1	reverse(4);
	motorSpeed(4,25);
	goFor(10);
		10	58.658

2	brake(4); goFor(1);	1	0

3	reverse(4);
	motorSpeed(4,60);
	goFor(.5);
		0.5	8.7557

4	brake(4);	4	0
		Total Energy:	67.4137

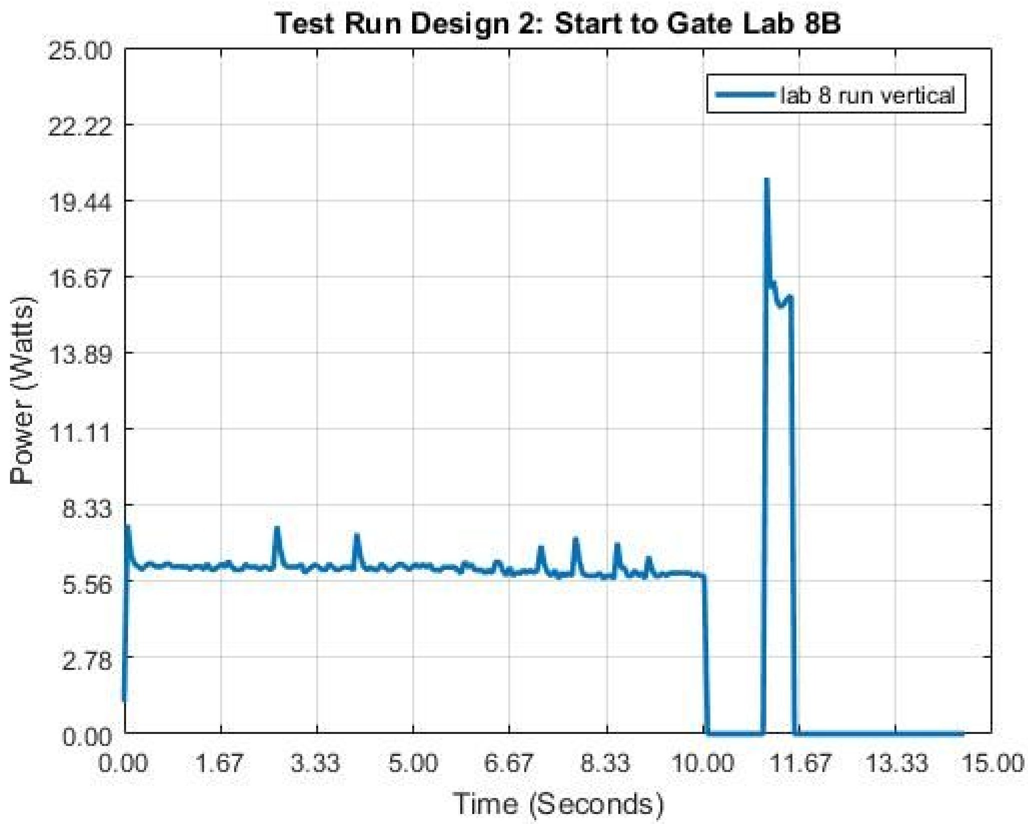


Figure 4: Test Run 2A: Start to Gate Lab 8B

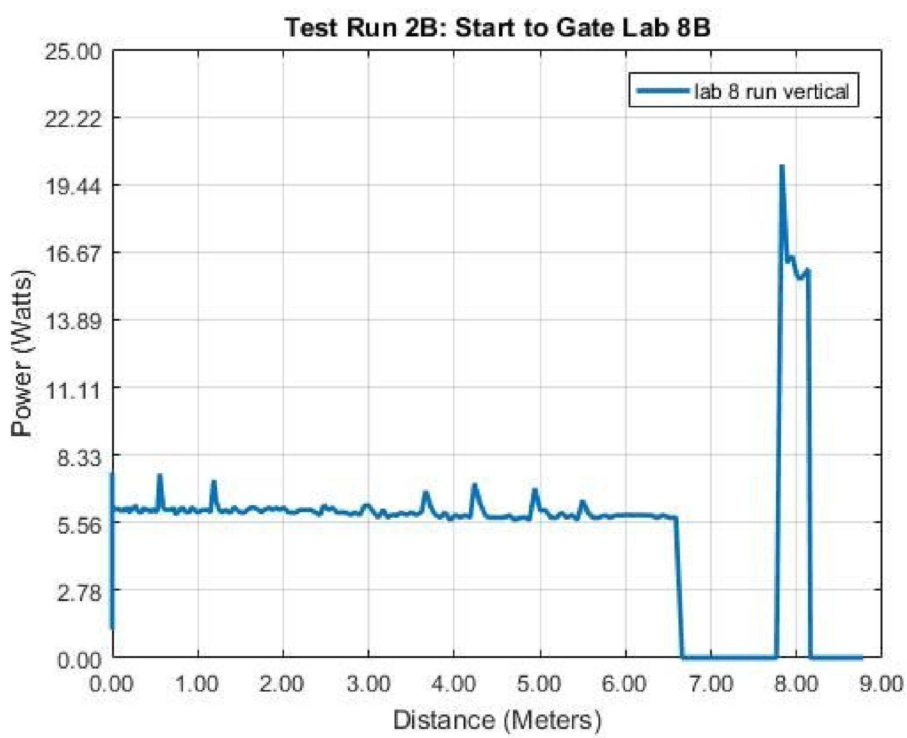


Figure 5: Test Run 2B: Start to Gate Lab 8B

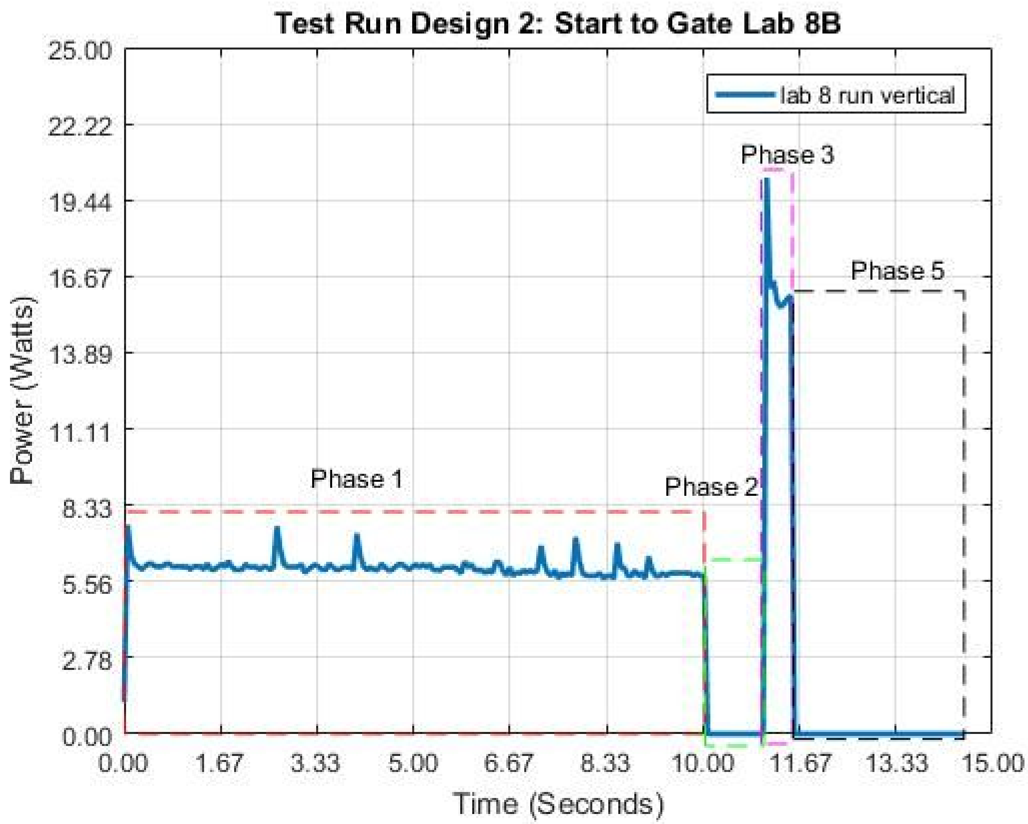


Figure 6: Lab 8B Test Run 2 in Phases

Table 3: Lab 8B Run 2 Phase Energy Breakdown

Phase	Arduino Code	Time(seconds)	Total Energy (joules)

1	reverse(4);
	motorSpeed(4,25);
	goFor(10);
		10	60.6531

2	brake(4); goFor(1);	1	0

3	reverse(4);
	motorSpeed(4,60);
	goFor(.5);
		0.5	7.9044

4	brake(4);	4	0
		Total Energy:	68.5575

Table 4: Concept Screening Scoresheet

Success Criteria	Reference	Design Team	Design Melanie	Design Jennifer	Design Katie	Design Jessica
Balanced in turns	0	0	0	0	0	0
aerodynamics	0	+	+	+	+	+
center of gravity	0	+	0	0	0	0
maintenance	0	-	-	-	-	0
durability	0	0	0	0	0	0
cost	0	0	0	0	0	0
thrust/power	0	+	-	0	0	0
environmental	0	0	0	0	0	0
sum +s	0	3	1	1	1	1
sum 0s	8	4	5	6	6	7
sum -s	0	1	2	1	1	0
Net score	0	2	-1	0	0	1
Continue?	no	yes	no	combine	combine	yes

Table 5: Concept Scoring Matrix

		Team Reference		Jennifer/Katie		Jessica	
Success Criteria	weight	rating	weighted score	rating	weighted score	rating	weighted score
Balanced in turns	15	4	0.6	3	0.45	4	0.6
aerodynamics	25	3	0.75	4	1	4	1
center of gravity	15	4	0.6	3	0.45	3	0.45
maintenance	5	3	0.15	3	0.15	3	0.15
durability	10	3	0.3	4	0.4	3	0.3
cost	5	5	0.25	5	0.25	5	0.25
thrust/power	20	4	0.8	3	0.6	3	0.6
environmental	5	3	0.15	3	0.15	3	0.15
Total Score			3.6		3.45		3.5
Continue?		Develop		No		No	

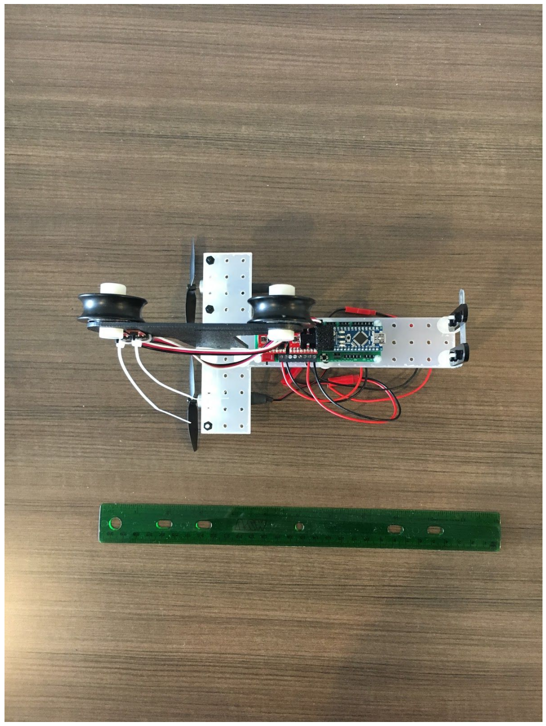


Figure 7: Parallel AEV Design Top View

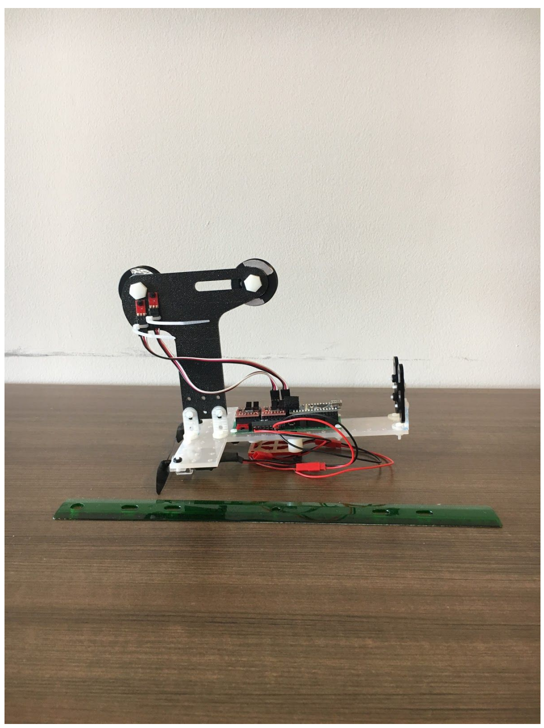


Figure 8: Parallel AEV Design Front View

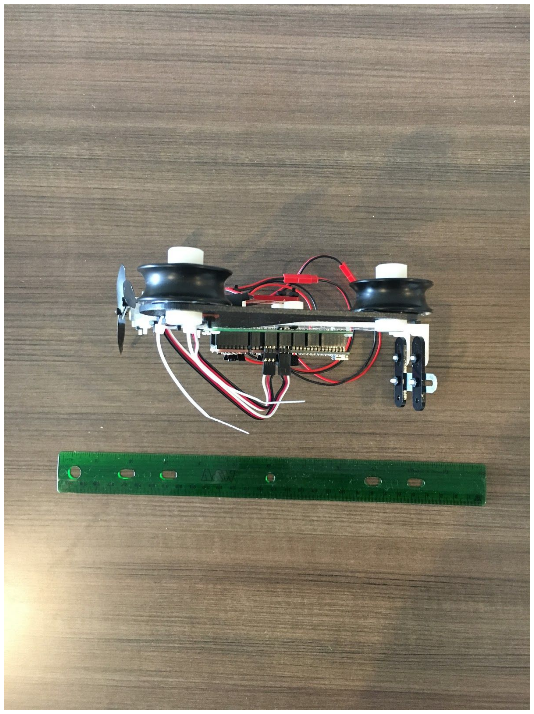


Figure 9: Vertical AEV Design Top View

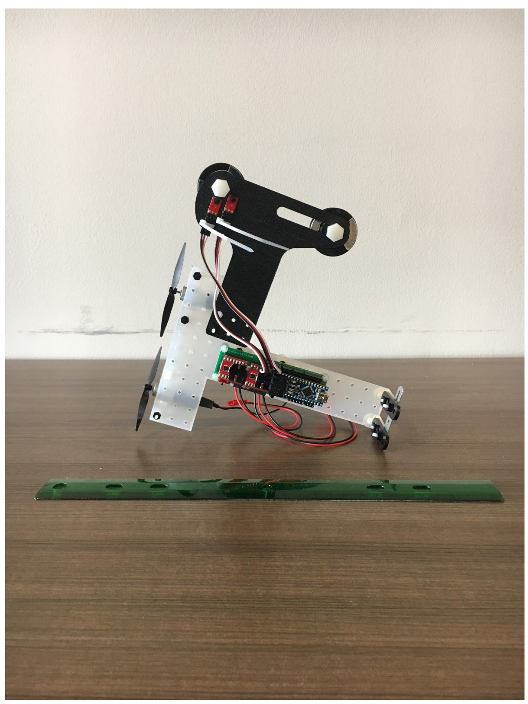


Figure 10: Vertical AEV Design Front View

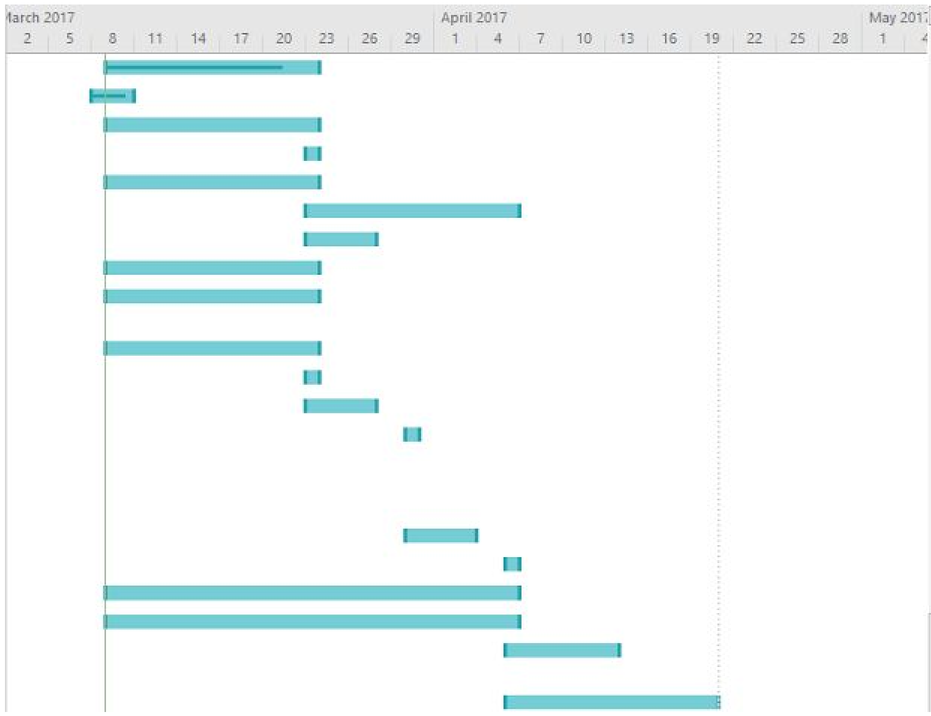


Figure 11: Projected Team Schedule Timeline

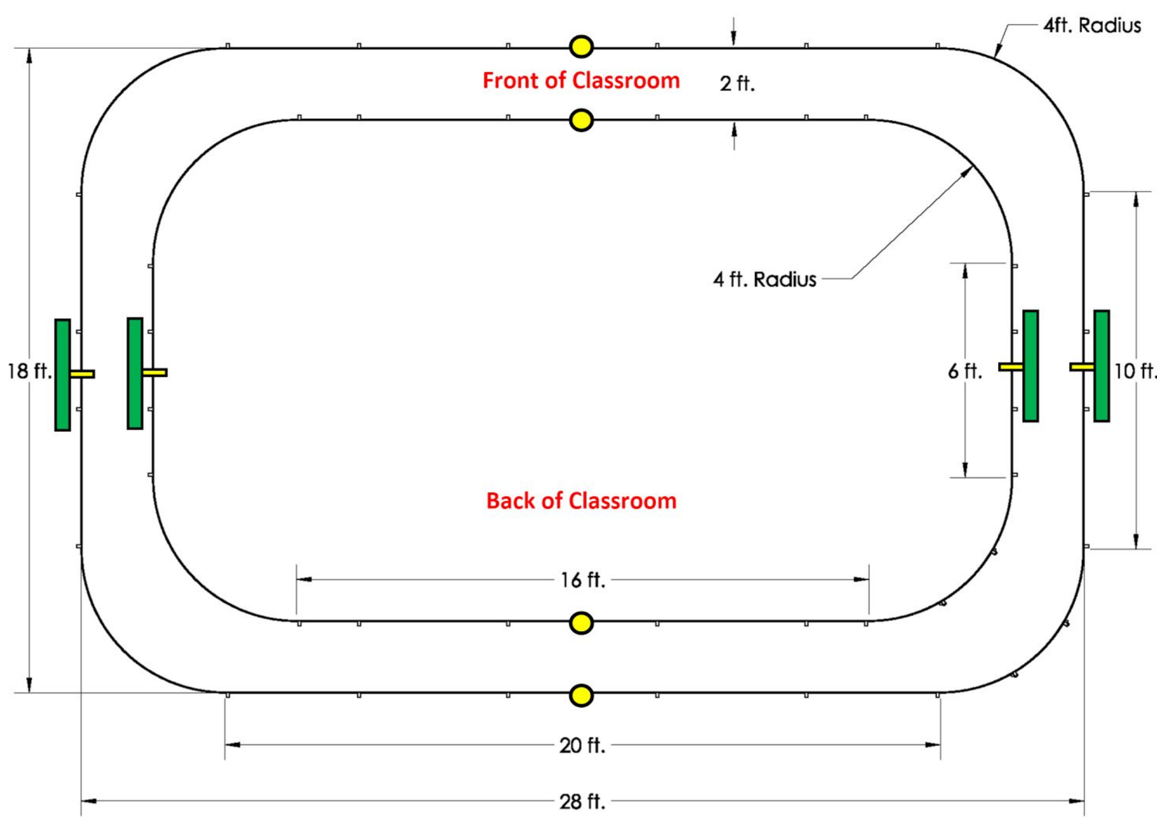


Figure 12: AEV Track Layout

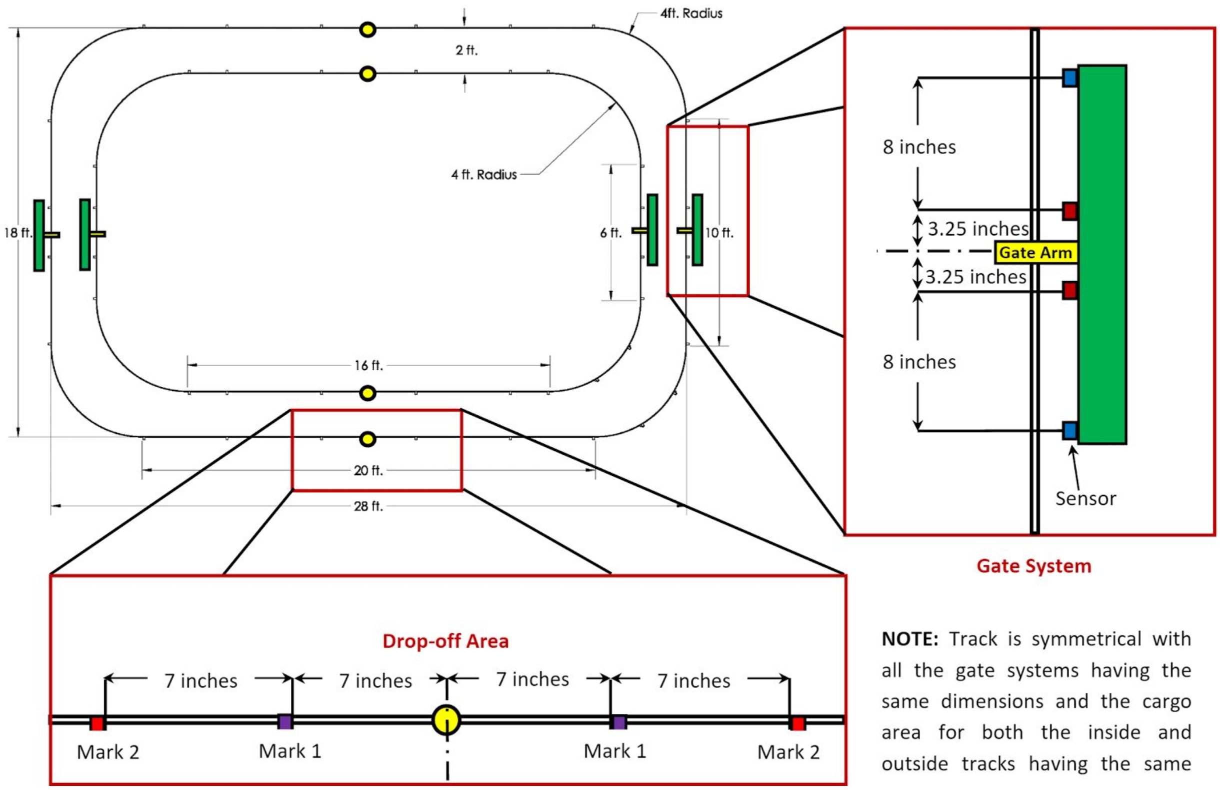


Figure 13: AEV Track Layout Beginning and Gate Specifics

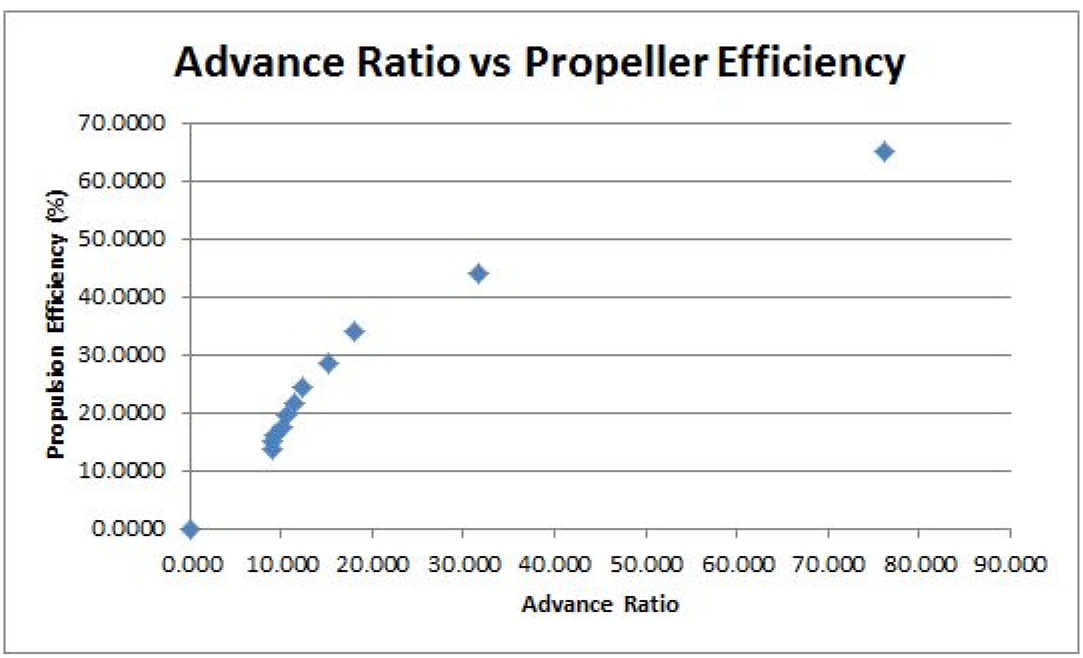


Figure 14: 3030 Second Pusher Propeller Advance Ratio vs Propulsion Efficiency Graph

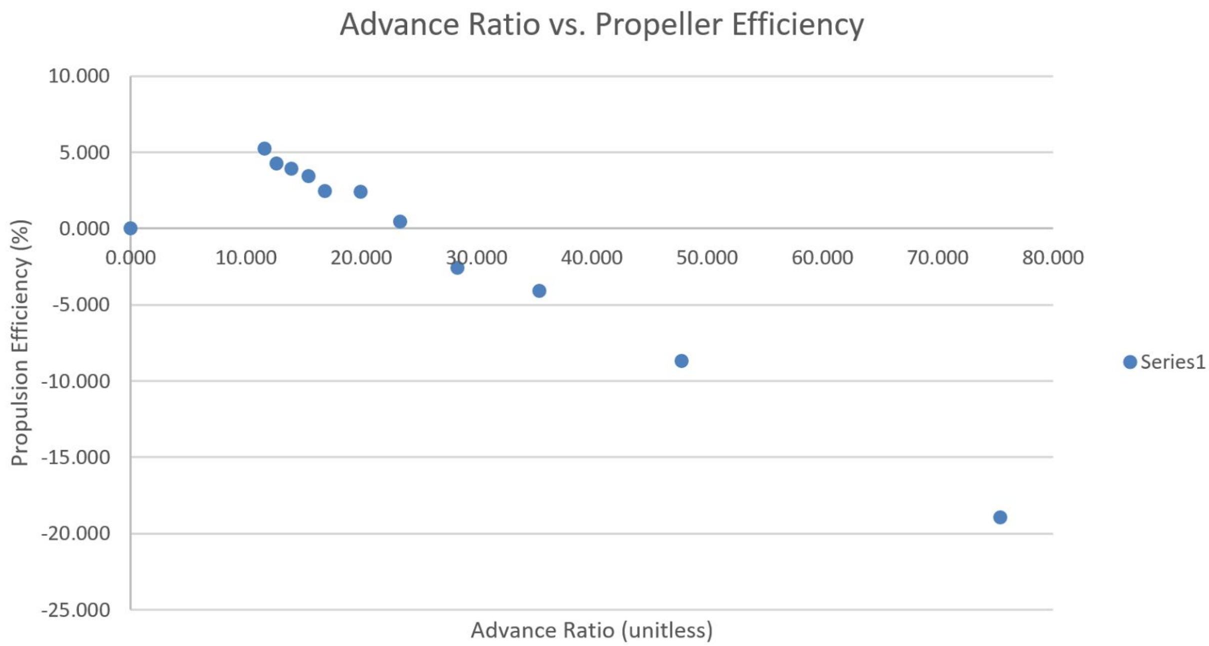


Figure 15: 2510 Puller Propeller Advance Ratio vs. Propeller Efficiency Graph

Table 6: Wind Tunnel Data Analysis First 3030 Pusher

Thrust Calibration	RPM	Power Input	Power Output	Power Output	Propulsion Efficiency	Advance Ratio
grams	*RPM*	*Watts*	*Horsepower*	*Watts*	*%*	*--*
0.0000	0.000	0.000	0.00000000	0.00000	0.000	0.0000
-0.1233	2035	0.000	-0.00000438	-0.00326	0.000	62.6826
-0.9453	3035	0.0999	-0.00003356	-0.02503	-25.053	42.0293
-1.7262	3892	0.2664	-0.00006129	-0.04570	-17.156	32.7747
-2.6304	4610	0.5180	-0.00009339	-0.06964	-13.444	27.6701
-3.9045	5449	0.8436	-0.00013863	-0.10337	-12.254	23.4096
-3.6168	6167	1.2432	-0.00012841	-0.09576	-7.702	20.6841
-3.3702	6826	1.7168	-0.00011966	-0.08923	-5.197	18.6872
-1.8495	7425	2.1978	-0.00006567	-0.04897	-2.228	17.1797
-0.2055	8043	2.8490	-0.00000730	-0.00544	-0.191	15.8596
1.3563	8682	3.5002	0.00004815	0.03591	1.026	14.6924
3.2058	9481	4.3068	0.00011382	0.08488	1.971	13.4542

Table 6: Wind Tunnel Data Analysis Second 3030 Pusher

Thrust Calibration	RPM	Power Input	Power Output	Power Output	Propulsion Efficiency	Advance Ratio
grams	RPM	Watts	Horsepower	Watts	%	--
0.0000	0.000	0.0000	0.00000000	0.00000	0.000	0.0000
0.2055	2035	0.0074	0.00000757	0.00564	76.247	65.0042
1.0275	2994	0.0888	0.00003783	0.02821	31.770	44.1829
1.8495	3892	0.2812	0.00006810	0.05078	18.058	33.9886
2.8770	4610	0.5180	0.00010593	0.07899	15.249	28.6949
3.9045	5389	0.8658	0.00014376	0.10720	12.382	24.5469
5.3430	6107	1.2691	0.00019673	0.14670	11.559	21.6610
6.7815	6766	1.7464	0.00024969	0.18619	10.662	19.5512
8.4255	7485	2.2644	0.00031022	0.23133	10.216	17.6731
9.8229	8143	2.8860	0.00036167	0.26970	9.345	16.2451
11.7135	8802	3.5816	0.00043129	0.32161	8.979	15.0288
15.0015	9760	4.5288	0.00055235	0.41189	9.095	13.5536

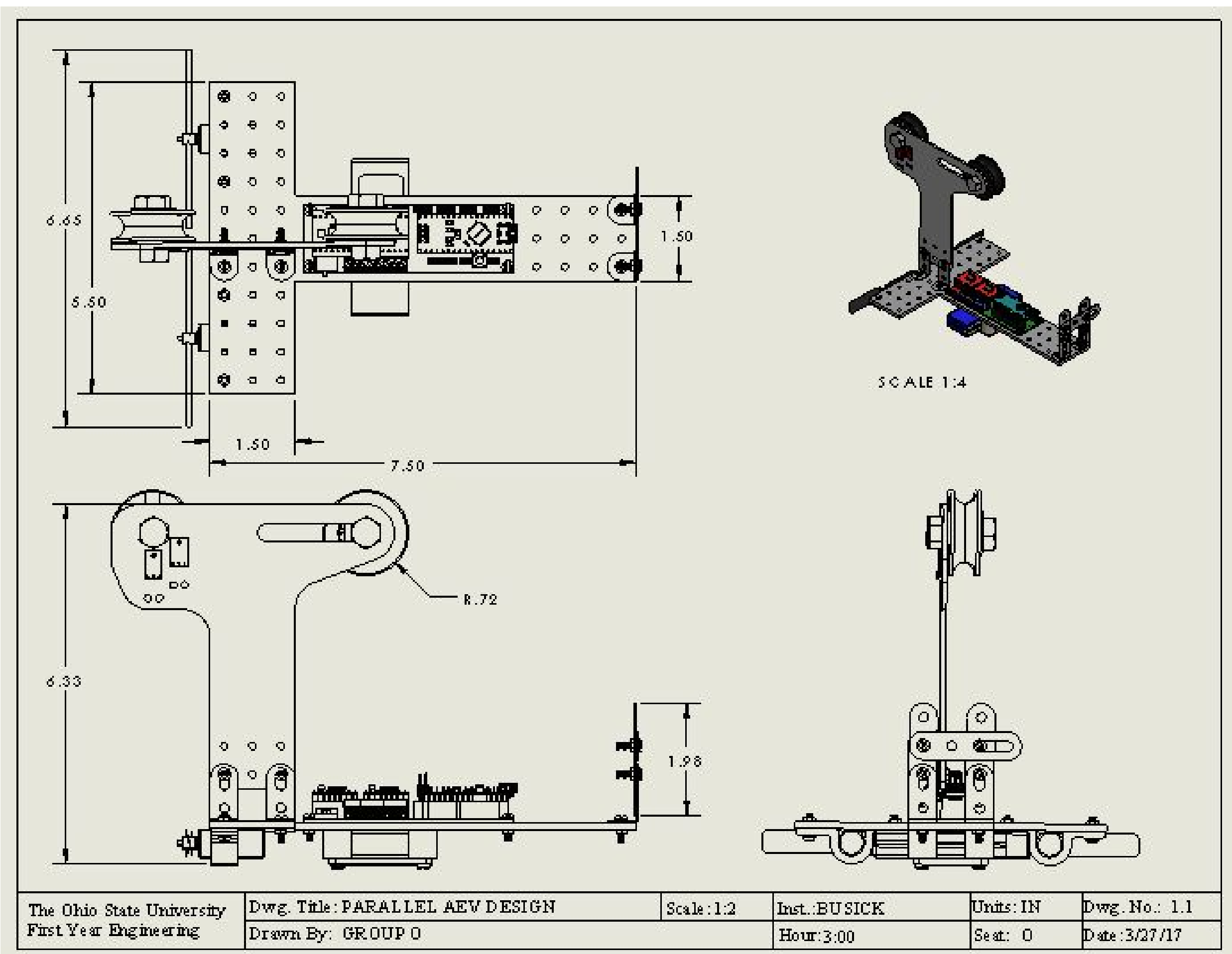
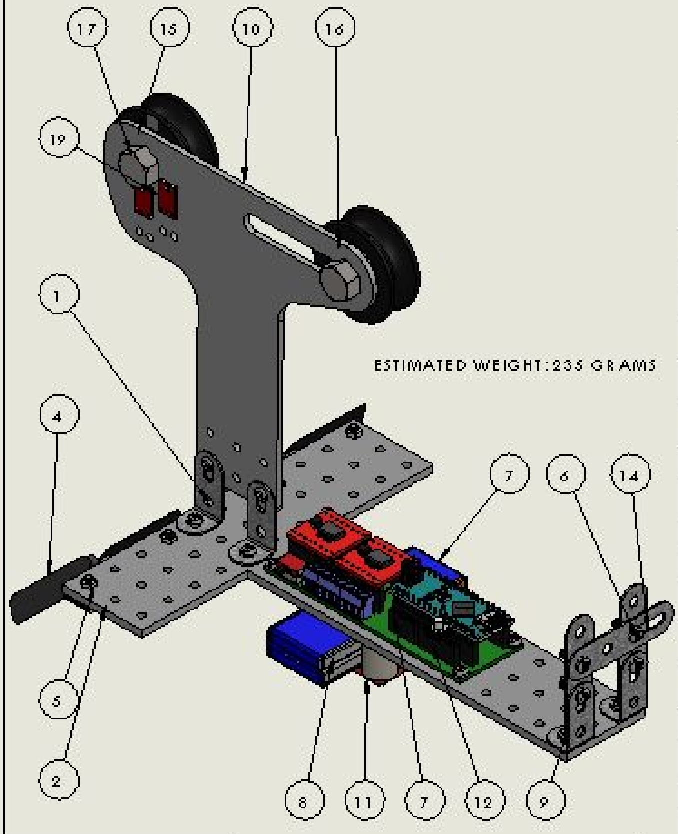


Figure 16: Primary Orthographic Views and Dimensions of Parallel AEV Design



ITEM NO.	DESCRIPTION	QTY.	COST
1	Tee	1	$3
2	Motor Mount Clip Aluminum	2	$0.59
3	AEV Motor	2	$20
4	Prop 3inch	2	$0.45
5	MS HXNUT 0.086-56-S-S	18	$2.88
6	SL-PHMS 0.112-40x0.375x0.375-S	18	$2.88
7	Battery Pack	1	N/A
8	Battery Pack Clamp Plate Narrow	1	$1
9	90-deg bracket	4	$2
10	Support Arm 1 2 Sensor Holes V2	1	N/A
11	Battery Spacer	3	$1
12	AEV Arduino Assembly	1	$100
13	SL-PHMS 0.112-40x0.5x0.5-S	3	$3
14	flat bracket	3	$3
15	Pulley Assembly w-reflective tape	1	$7.50
16	Pulley Assembly	1	$7.50
17	HBOLT 0.3125-18x0.875x0.875-S	2	$2.88
18	MS HXNUT 0.3125-18-S-S	2	$2.88
19	Rotation Sensor Board	2	$4

The Ohio State University First Year Engineering	Dwg. Title: PERPENDICULAR AEV DESIGN BOM	Scale: 2:3	Inst.: BUSICK	Units: IN	Dwg. No.: 1.2
	Drawn By: GROUP 0		Hour: 3:00	Seat: 0	Date: 3/27/17

Figure 17: Bill of Materials for Parallel AEV Design

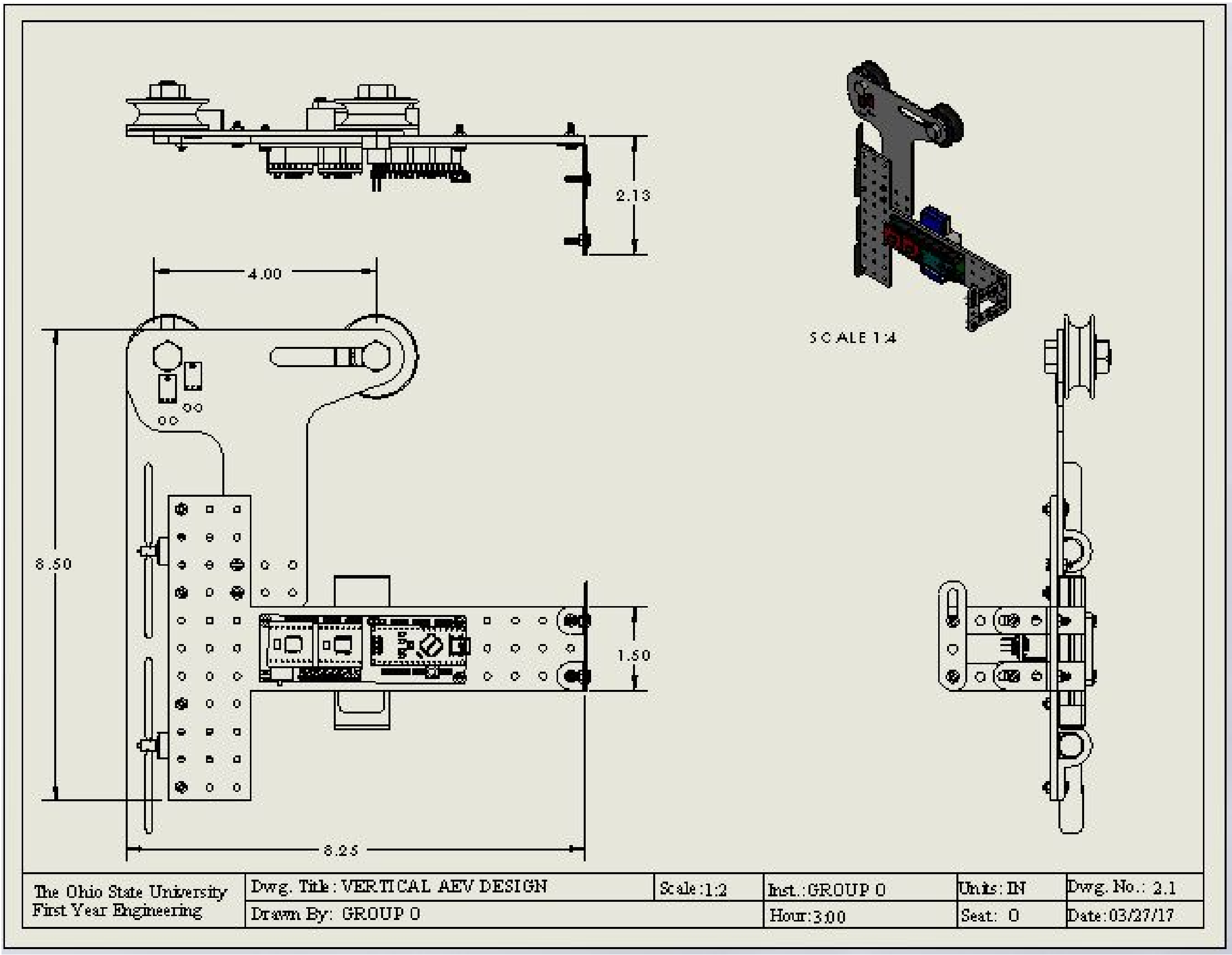
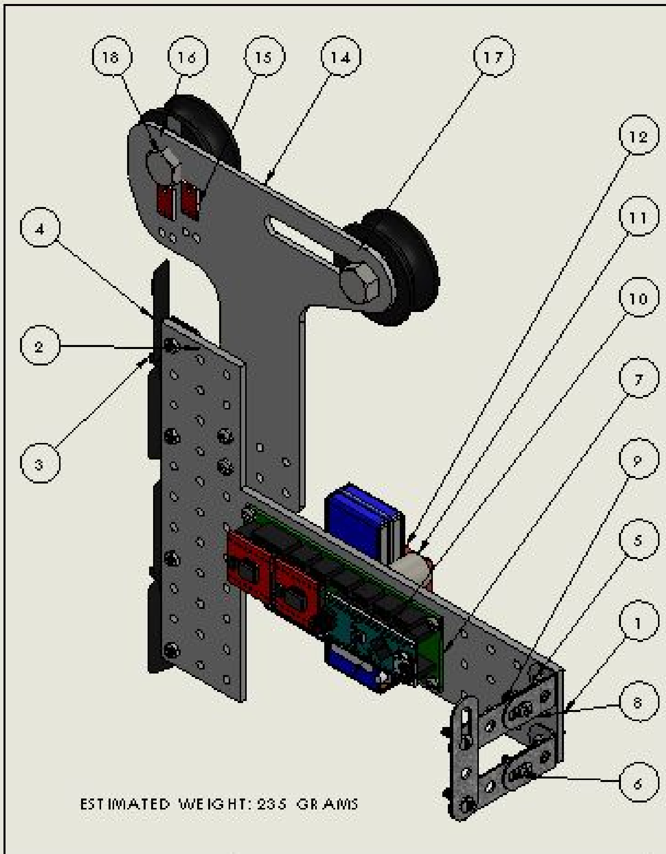


Figure 18: Primary Orthographic Views and Dimensions of Vertical AEV Design



ITEM NO.	DESCRIPTION	QTY.	COST
1	Tee	1	$3
2	Motor Mount Clip Aluminum	2	$0.59
3	AEV Motor	2	$20
4	Prop 3inch	2	$0.45
5	SL-PHMS 0.112-40x0.375x0.375-S	16	$2.88
6	MSHXNUT 0.086-56-S-S	16	$2.88
7	AEV Arduino Assembly	1	$100
8	90-deg bracket	2	$2
9	flat bracket	3	$3
10	Battery Pack	1	N/A
11	Battery Spacer	3	$1
12	Battery Pack Clamp Plate Narrow	1	$1
13	SL-PHMS 0.112-40x0.5x0.5-S	3	$2
14	Support Arm 1 2 Sensor Holes V2	1	N/A
15	Rotation Sensor Board	2	$4
16	Pulley Assembly w-reflective tape	1	$7.50
17	Pulley Assembly	1	$7.50
18	HBOLT 0.3125-18x0.875x0.875-S	2	$2.88
19	MSHXNUT 0.3125-18-S-S	2	$2.88

The Ohio State University First Year Engineering	Dwg. Title: PARALLEL AEV DESIGN	Scale:2:3	Inst.:BUSICK	Units:IN	Dwg. No.: 22
	Drawn By: GROUP 0		Hour:3:00	Seat: 0	Date:3/27/17

Figure 19: Bill of Materials for Vertical AEV Design

Arduino Code

8A and 8B Run 1&2

```
reverse(4);
motorSpeed(4,25);
goFor(10);
brake(4);
goFor(1);
reverse(4);
```

8C Run 1

```
//First quarter
reverse(4);
motorSpeed(4,20);
goToAbsolutePosition(475);
reverse(4);
motorSpeed(4,60);
```

```
goFor(.5);
brake(4);
goFor(7); //stop at gate for 7 seconds

//2nd quarter
reverse(4);
motorSpeed(4,20);
goToAbsolutePosition(950);
reverse(4);
motorSpeed(4,60);
goFor(.5);
brake(4);

//3rd quarter
//start back with r2 unit
//reverse(4); //do this if servo design
motorSpeed(4,20);
goToAbsolutePosition(470);
reverse(4);
motorSpeed(4,60);
goFor(.5);
brake(4);
goFor(7); //brake at gate for 7 seconds

//4th quarter
reverse(4);
motorSpeed(4,20);
goToAbsolutePosition(0);
reverse(4);
motorSpeed(4,60);
goFor(.5);
brake(4);
```

Lab 8C Run 2

```
//First quarter
reverse(4);
motorSpeed(4,20);
goToRelativePosition(470);
reverse(4);
motorSpeed(4,60);
goFor(.5);
brake(4);
goFor(7); //stop at gate for 7 seconds

//2nd quarter
reverse(4);
motorSpeed(4,20);
```

```
goToRelativePosition(475);
celerate(4,20,0,2);
goFor(1);

//3rd quarter
//start back with r2 unit
reverse(4); //don't do this if servo design
motorSpeed(4,35);
goToRelativePosition(475);
reverse(4);
motorSpeed(4,60);
goFor(.5);
brake(4);
goFor(7); //brake at gate for 7 seconds

//4th quarter
reverse(4);
motorSpeed(4,20);
goToRelativePosition(475);
reverse(4);
motorSpeed(4,60);
goFor(.5);
brake(4);
```