Pioneer Rocketry

University of Wisconsin-Platteville

Team High Powered Rocketry Mentor: Adrian Guither

Team Faculty Advisor: Duane Foust (foustd@uwplatt.edu)

Student Team Lead: Jacob Ellenberger (ellenbergerj@uwplatt.edu)

Members: Jacob Napp, Adam Nielsen, Grant Oberhauser, Jane Dickler

Midwest Rocket Launch Competition

2015 – 2016

Contents

Summary of Rocket Design

General Design:

The goal of the competition is to use an active drag system to accurately control apogee. An initial flight will be launched to set an apogee baseline, and the drag system will then be utilized in a subsequent flight to reach a height that is 75% of this baseline. To achieve this goal, the rocket "Skybreaker" was designed.

Nose cone:

The nose cone selected for this rocket is a 20in ogive nose cone. An ogive was selected due to its simplicity and good aerodynamic properties. The nose cone is made from PLA plastic that has been 3D printed. As a team, Pioneer Rocketry has extensive experience with 3D printing nose cones, and they have proven to be very strong. It is covered in fiber glass to ensure that it has maximum strength. The nose cone has an aluminum tip with a threaded hole for the quick swapping of pitot tubes. The Pitot tube has threads so that if it gets damaged during the landing of the rocket it can be replaced quickly. The nose cone also contains a small electronics bay which is a change from the preliminary design report. This houses a Micro Arduino that reads and the pressure sensor connected to the pitot tube. This is connected to the main electronics bay via a wire that travels through the upper body tube.

Body Tube:

The body tube selected for this rocket is 4in diameter fiberglass. The fiberglass was chosen for many reasons. The strength and rigidity of the fiberglass provide high tensile strength and resistance to elastic deformation. In addition, its weather resistance and dimensional stability are beneficial. A material like Blue Tube is highly sensitive to humidity and will expand and contract causing alignment issues. Fiberglass was chosen over the more exotic carbon fiber because it does not impede the radio waves needed for the telemetry system.

Airbrakes and Avionics Bay:

The airbrakes and avionics are housed together in the middle of the rocket, inside a single piece of coupler tube. The avionics bay starts at the separation point for the upper and lower sections of the rocket. Refer to the sections for the avionics and airbrakes under the Drag System for a more complete description of their design and functionality.

Recovery:

The recovery system on this rocket is a dual deployment system. The drogue parachute is a 24in, hemispherical parachute, and the main parachute is 72in and also hemispherical. The shock cord selected for this rocket is 0.25in diameter Kevlar with a breaking strength of 3600lb. To help absorb the shock of parachute inflation, the shock cord has an elastic cord attached. To ensure the parachutes are ejected properly, a redundant altimeter setup was utilized. Two Stratologger CF's were used in this system. The primary system deploys the drogue at apogee and the main at 700ft while the backup deploys the drogue at apogee plus one second, while the main is deployed at 500ft. The ejection charges were ground tested, and the rockets separation was verified.

Fins:

The fins on this rocket are made of 0.125in thick G10 fiberglass. The shape of the fin is a symmetric trapezoid and dimensioned to maximize durability and prevent damage on landing. The symmetric fin is 3.5in tall with a tip that measures 3in and a base that is 5in. The fins have a relatively high aspect ratio (height/width) as this creates both better stability forces and positions the center of pressure farther back on the airframe. The fins were placed above the tail cone to reduce the chance that the rocket lands on its fins.

Camera Pod:

To visually verify the deployment of the airbrake system, a camera pod was designed to attach to the side of the rocket. This component was 3D printed using PLA Plastic, and was fixed to the top of the rocket to allow for a better view of the airbrakes throughout the flight.

Rocket Diagrams

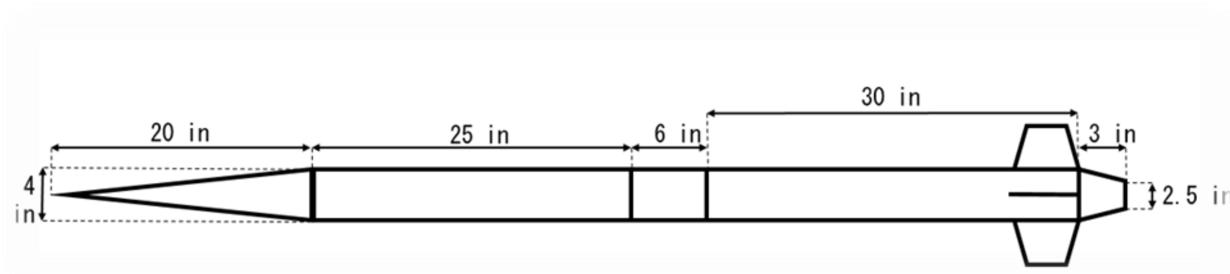

Figure 1: Dimensional drawing of Skybreaker

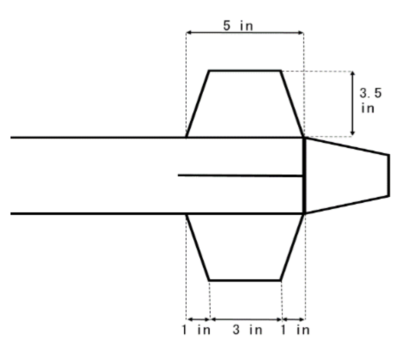

Figure 2: Dimensions of the fins.

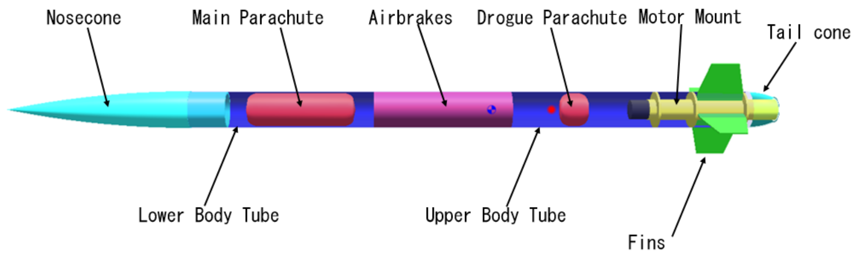

Figure 3: Layout of Skybreaker.

Stability Analysis

Based on the OpenRocket model, the stability of Skybreaker is 2.3cal. This increases to 3.0cal after motor burnout. The rocket was designed with additional stability so that the rocket remains stable even under the influence of the airbrake system.

Propulsion Specifications

Pioneer Rocketry will be launching Skybreaker on a Cesaroni K2045. The motor is 54mm in diameter and 404mm in length. The motor contains 4-Grains of Vmax propellant and has a total burn time of 0.7s. The motor will produce a total impulse of 1407.6 Ns, a maximum thrust of 2231.2N. The Vmax propellant is a good choice for several reasons. The competition requires that the airbrakes come out after motor burnout. Because the Vmax Motor has a short burn time, the airbrakes can be deployed earlier in the flight meaning more control over the rocket's acceleration and altitude. Another benefit of a Vmax motor is that it has a large velocity off the rail. This higher velocity, reduces the effect of wind on the path of the rocket thus making the rocket more predictable and easier to control. See Figure 4 below for the thrust curve for the K2045.

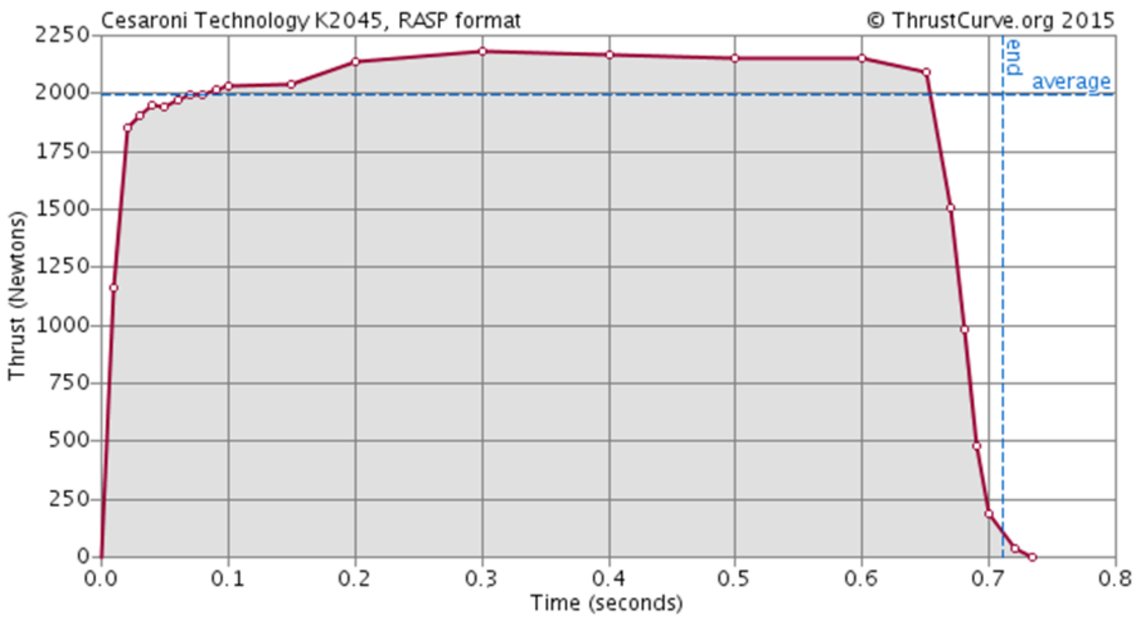

Figure 4: Thrust Curve for the Cesaroni K2045.

Materials Testing

The design of Skybreaker relies on several custom made 3D printed parts such as the nose cone, tail cone, and support material in the avionics bay. Pioneer Rocketry has relied extensively on 3D printed components in the past and has been impressed by their durability. A boosted dart descended from an altitude of around 4,000ft without a parachute and imbedded itself close to 10in into the ground. All the paint was chipped off of the cone, but the actual nose cone itself was completely intact. This club has flown nine other successful flights with 3D printed nose cones, centering rings, or tail cones. Because of these past successes, the team is confident in the strength of 3D printed components.

Construction Techniques

The main design philosophy in the design of Skybreaker is to make the rocket as durable as possible. The main area of focus was the fin can section. The cantilevered nature of fins makes them prime locations for damage during flight. To make the fins as strong as possible, the fins are through-the-wall style with the tab of the fin sliding into a slot that is cut into the centering ring. This slot ensures that the fin is aligned and perpendicular to the body tube. To ensure that the fins are straight, the slots cut into the body tube were cut using a Bridgeport milling machine. The Figure 5 below shows slots being cut on the mill. The fins are then reinforced with internal and external fillets. The fin can was finished with a layer of fiberglass that runs from fin tip to fin tip.

Figure 5: Milling the fin slots.

While in the past 3D printed parts have proven to be very strong they were also reinforced for this rocket. Both the nose cone and tail cone of the rocket have a layer of fiberglass on the surface of the part. This fiberglass was form fitted to the surface of the part using a vacuum food saver. Using the vacuum, even pressure could be applied to all areas of the part and force the fiberglass cloth to conform to the surface of the part.

Drag System

In order to accurately control the height the rocket flies, the control system has to be flexible and able to adapt to differences between flights such as differences in wind speed. The system is designed to get this flexibility through a feedback control loop. The Avionics first predict the apogee of the rocket, then compares this to the desired apogee. The result of this comparison is what controls the airbrake system.

Apogee Prediction

The first step in this control system is the prediction stage. The system works by using the current velocity to predict how much further from the current altitude the rocket will travel. This remaining distance is then added to the current altitude to get an estimate how high the rocket will travel. The two methods that have been explored in the finding the relationship between current velocity and remaining altitude are an analytical and empirical method. A comparison of these methods with flight data can be seen in Figure 6.

The first method explored was an analytical solution of the differential equation. The method made the assumption that the drag coefficient of the rocket remained constant. This equation was solved to find the relationship between velocity and height the rocket will travel and how that is affected by the rocket's mass and drag coefficient. The differential equation and its solution are shown below in equation of 1 and 2. It was found that the analytical approach was not very flexible to modify as more data from flight were acquired.

$$v * \frac{dv}{dx} = \frac{-cv^2}{m} - g \tag{1}$$

$$c = \frac{1}{2}\rho C_d A$$

$$x(v) = \frac{m}{2c} * \ln(1 + \frac{cv^2}{mg}) \tag{2}$$

v	Velocity
x	Remaining distance
m	Rocket Mass
g	Gravity
ρ	Air Density
C_d	Drag Coefficient
A	Cross-Sectional Area of Rocket

To fix the issues experienced in the analytical approach, a more flexible empirical approach was used. This approach took data from both OpenRocket simulations and from test flights to understand the relationship between velocity and remaining attitude. This data was acquired by subtracting the current altitude of the rocket from the measured apogee. This data was then plotted vs the current velocity.

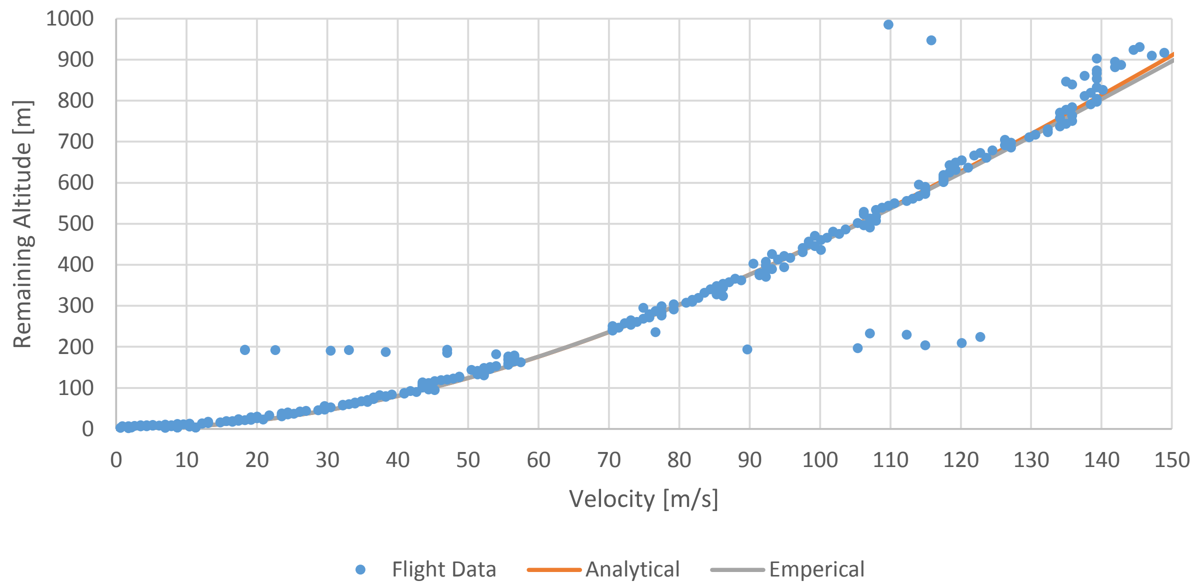

Figure 6: Comparison between the Empirical Look-Up table and the Analytical Solution

Apogee Control

With the predicted apogee known, a feedback control loop could be used to guide the rocket to the desired altitude. The method chosen to do this was a proportional derivative control loop. This control takes the difference between desired and predicted apogee and opens the airbrake proportional to this error. Since the system can only reduce the apogee of the cost of overshooting the desired apogee is large. To prevent this a derivative term was added to the control system. This term looks at how fast the predicted apogee is approaching the desired apogee and opens the airbrakes proportional to this rate. This acts to dampen the system and cause it to smoothly approach the desired apogee.

With the requirements of the control system known, the actual systems needed to implement it could be designed. The following sections will go into the electronic, software, and mechanical systems that make Skybreaker possible.

Electronics Design

Physical Components

The basis for the electronic payload is an Arduino Due. The Arduino was powered by a 2 cell Lithium Polymer Battery. All components interfacing with the Arduino Due were soldered to a perfboard Arduino shield to ensure that they are adequately secured throughout the flight. When a 3 volt sensor for the pitot tube did not provide adequate precision, a 5 volt sensor was used instead. This pitot tube was implemented by placing an Arduino Micro in the nosecone of the rocket. This reads the pressure difference from the pitot tube and sends it over I^2C to the Arduino Due. A Stratologger and an accelerometer also provide data throughout the flight.

Otherwise, the physical construction of the electronics have not changed since the design report with the exception to the pitot tube changes detailed above.

Skybreaker Software

Software High Level Architecture

The code running the Arduino was written in C++. This code was developed on the Arduino IDE and in Visual Studios. The Arduino IDE develops code that runs directly on the Arduino. It uses a modified version of C++, and allows for incredible easy interfacing with standard C++ classes. Most of the code was developed in Visual Studio using standard C++, then interfaced with the Arduino control loop. This approach was selected because the Arduino IDE already works directly with the Arduino Due. The remaining code was developed in standard C++ using the principles of Object Orientation to provide high cohesion and low coupling.

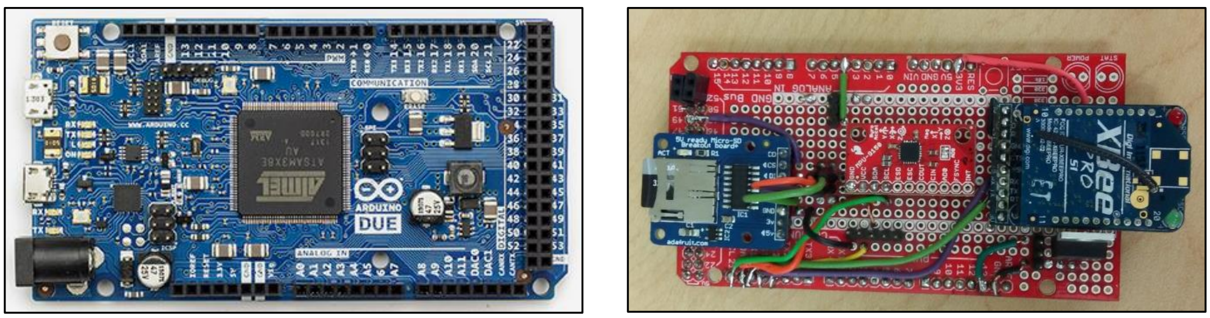

Figure 7: Left is the Arduino Due. Right is the custom shield created for this competition.

Object Oriented programming is a programming language paradigm that uses objects which contain data and procedures. Objects can perform each other's procedures, however they can rarely access each other's data directly. This helps maintain high cohesion and low coupling. In computer programming, cohesion is the amount procedures in an object belong with each other. The procedures of an object with low cohesion seem to be thrown together by coincidence, while the procedures of an object with high cohesion all contribute to a single well-defined task. On the other hand, coupling refers to the amount procedures in objects rely on the procedures in other objects. High coupling happens when one object relies on the internal workings of procedures in another object, the dependent object then has to be changed whenever the independent object is changed. Low coupling happens when object have less coordination, data flow, and interdependency. Appendix II provides an Analysis Level Class Diagram of the software implemented in this rocket.

States

The code has three states, coast, descending, landed which all have unique behavior.

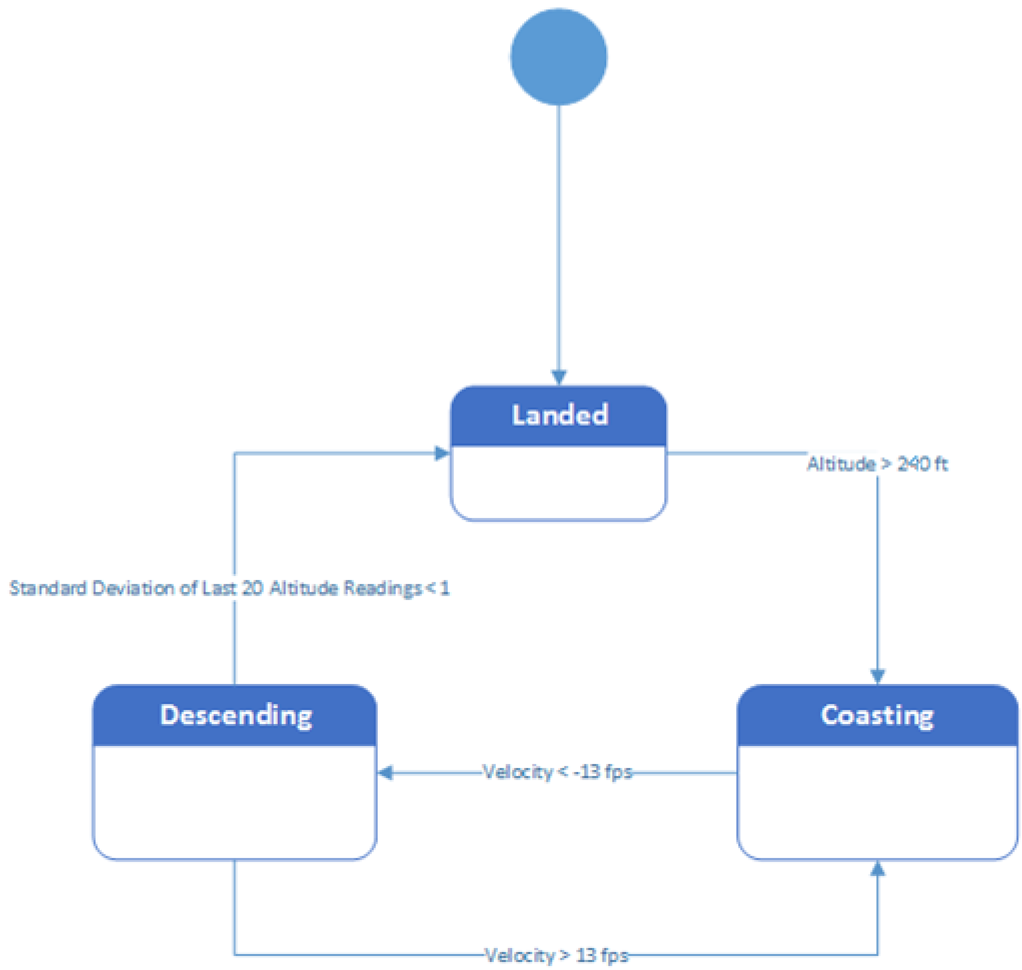

Figure 8: is a State Diagram of the flight states for the flight control system.

The rocket is initialized in the landed state. At this state, the air brakes are actively held closed. The transition between landed and coast is determined by altitude. A running average of 20 readings of the displacement measured by the Stratologger, the velocity measured by pitot tube, and velocity calculated from Stratologger is used to trigger all of the events.

The transition from the landed state to the coast state of the flight occurs at burnout. This height must be determined from the specific design of the rocket and motor being used. When Skybreaker uses a K2046 motor, as it will in the competition, this occurs at 240 feet. So at this point, the program switches from the landed state to the coast state. At this point, the air brakes can be activated. A more detailed description of how the air brakes are deployed and controlled can be found below. When the velocity dips below 20 fps as measured by the pitot tube, the air brakes are closed. This assures that the air brakes are closed before apogee. However, the rocket is still considered to be in the coast stage, and will not transition to descent until the velocity of the rocket as calculated from the Stratologger displacement is below -13 fps. Once again, in this state the air brakes are actively held closed. If the velocity calculated from the Stratologger is measured to be above 13 fps, the event switches back to the coast stage. This means that if the rocket experiences a turbulent flight, and records a false positive apogee, the rocket can still recover and continue to deploy the air brakes for the majority of the remaining coast stage of the flight. The last stage of the rocket is the landed state. This occurs when the standard deviation of among the last 20 displacement measurements is one foot or less.

Filtering Data

Because the inherent noise from the sensors, the rocket implements a Kalman Filter to smooth the data. The Kalman Filter predicts the behavior of the rocket. It then compares this prediction with the sensor data. From this, it actively extrapolates the reliability of the sensors and the predictions. This smooths the data of unnecessary noises and actively merges the three main sensors being used, namely the altimeter, Pitot tube, and accelerometer. By utilizing this filter, the code can determine key events in the launch with greater sensitivity without having to fear that a bad reading could inadvertently trigger a state change such as lift off or burnout. Furthermore, if a sensor where to fail, the filter would automatically disregard the bad readings and calculates it from the remaining two inputs. The three data measurements being filtered are the displacement measured by the Stratologger, the velocity measured by the pitot tube, and the vertical acceleration measured by the accelerometer.

Air Brake Controls

The air brake control system, as detailed above, provided a lookup table of velocities and the remaining altitude the rocket would reach. Linear interpolation was used to more accurately calculate the remaining altitude from the lookup table. This was chosen because it would provide a faster calculation than implementing polynomial function in the code.

The proportional derivative control loop described above was implemented in the code. This output a decimal value between 0 and 1 that was used to control the percent that the air brakes were open.

Testing Code

Because the Arduino does not provide easy debugging features, we had a difficult time properly testing the code. To help with this process, a class was created to read test data off of the SD card to simulate the rocket's sensors. This method was used to test the core functionality of the code such as the filtering, air brake control, and state changes. This also allowed testing of the classes related to data logging and telemetry.

Once the code was thoroughly tested, the base classes responsible for directly reading data from our sensors were tested in various ways. The accelerometer was violently displaced along all three axes of acceleration. The recorded acceleration along each of the axes were then examined to ensure that it was reading the proper values. An automotive test platform was used to calibrate the velocity readings from the pitot tube. The velocity being output by the pitot tube was compared to the test platform's speedometer and adjusted accordingly. The displacement as measured by the Stratologger was calculated by the placing the avionics bay into an altimeter test chamber. This test chamber uses an internal pressure sensor to control the air pump to simulate a realistic rocket flight.

Mechanical system design of air brakes

Design Overview

The airbrake system designed used four plates that rotate outward into the airflow around the rocket and obstruct the airflow around the rockets. The benefit of using plates that rotate normal to the airflow is that the forces generated by the airbrakes plates are not acting to open or close the airbrake system. Because of this the actuator that is used to open the airbrake does not need to be that strong and the airbrakes can open quickly. It was found during design that a quick opening airbrake is much more effective due to the higher velocities at full extension as well as being able to be active for more of the coast period. The quick acting airbrake also has the benefit of being able to quickly adapt to any disturbance

Airbrake Mechanism design

To both ensure that all brakes open simultaneously and to drive the system with a single input all of the plates were geared together. Due to the size constraints present inside of the rocket this gearing was accomplished by using a central gear to drive four pinion gears that are attached to the plates. This central gear is made up of two tiers. The lower gear meshes with the four plates and the upper gear meshes with a pinon on the servo motor. Figure 9 below shows a render of this mechanism with different parts indicated in different colors. The gears indicated with pink are the pinions linked to the plates. The blue gear is the central gear that meshes with the plates. The red gear is the gear driven by the servo motor and is attached to the lower gear. The cyan gear is the gear attached to the servo motor.

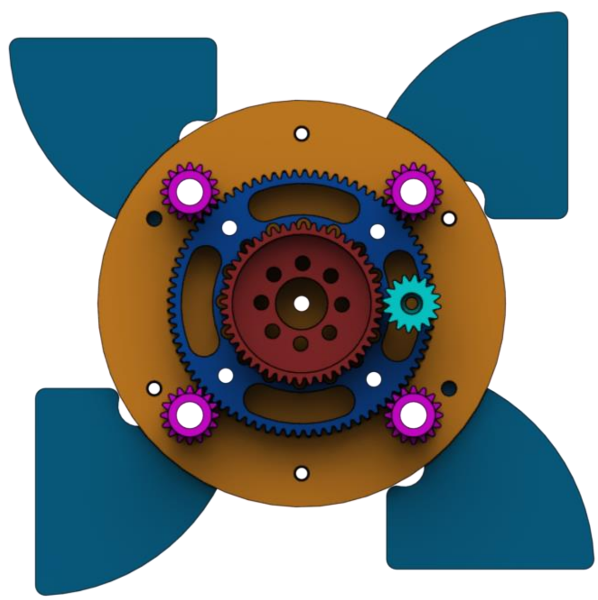

Figure 9: Render of the Airbrake Mechanism.

To make the mechanism as robust as possible the airbrake is made mostly out of aluminum. To accurately make the parts needed in this mechanism a Tormach CNC mill was used. The plates that extend into airflow are made from .25in thick aluminum plate. These plates are held onto a steel shaft and secured with a set screw. These steel shafts are supported by two ball bearings each that go into the center bulk plates. These bulk plates are also made of aluminum with one of them being .25in thick and the other being .1in thick. To fix the airbrake system to the rockets tube large 3D-printed shoulder pieces were used. These shoulders slide firmly into the fiberglass coupler used for this rocket and ensure a rigid connection. To solidify the whole system two bolts and two tie rods go through the airbrake module and secure it together. Since the bulkheads completely divide the upper and lower sections of the rocket, the bolts used to secure the rocket together have hole bored through them to allow for wires to pass through.

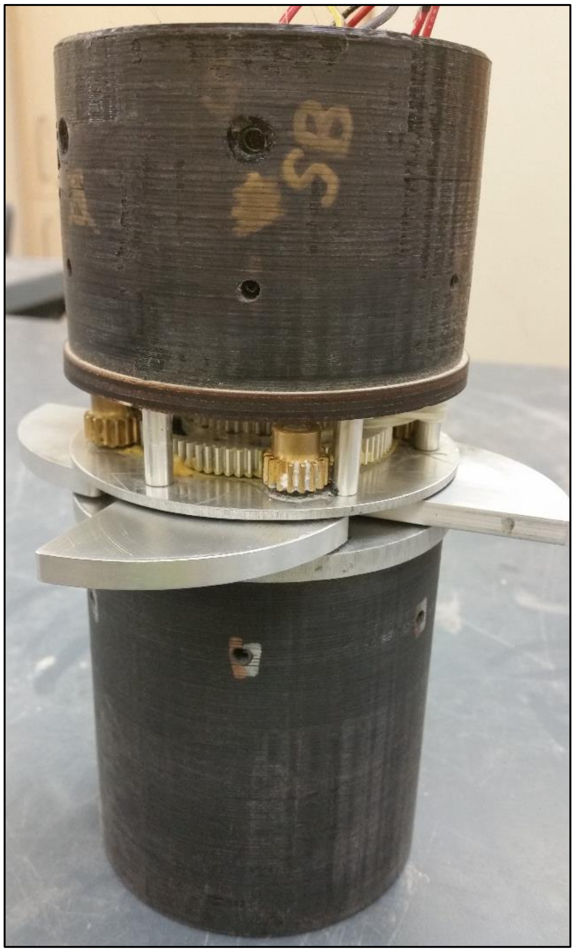

Figure 10: A photo of the Air Brake assembly with the 3D-printed shoulders.

Airbrake System Performance

To test the drag performance of the airbrake system, tests were performed in the school's wind tunnel. Because of the size of Skybreaker only a portion of the rocket could be tested in the wind tunnel. To accomplish this the airbrake section was mounted at the base of the nose cone. To get force measurements the airbrake section was mounted to a sled that was free to slide on rails and push against a force transducer. To characterize the performance of the airbrake system 10 different amounts of deployment were tested at 4 different airspeeds. The apparatus is shown below in figure 11.

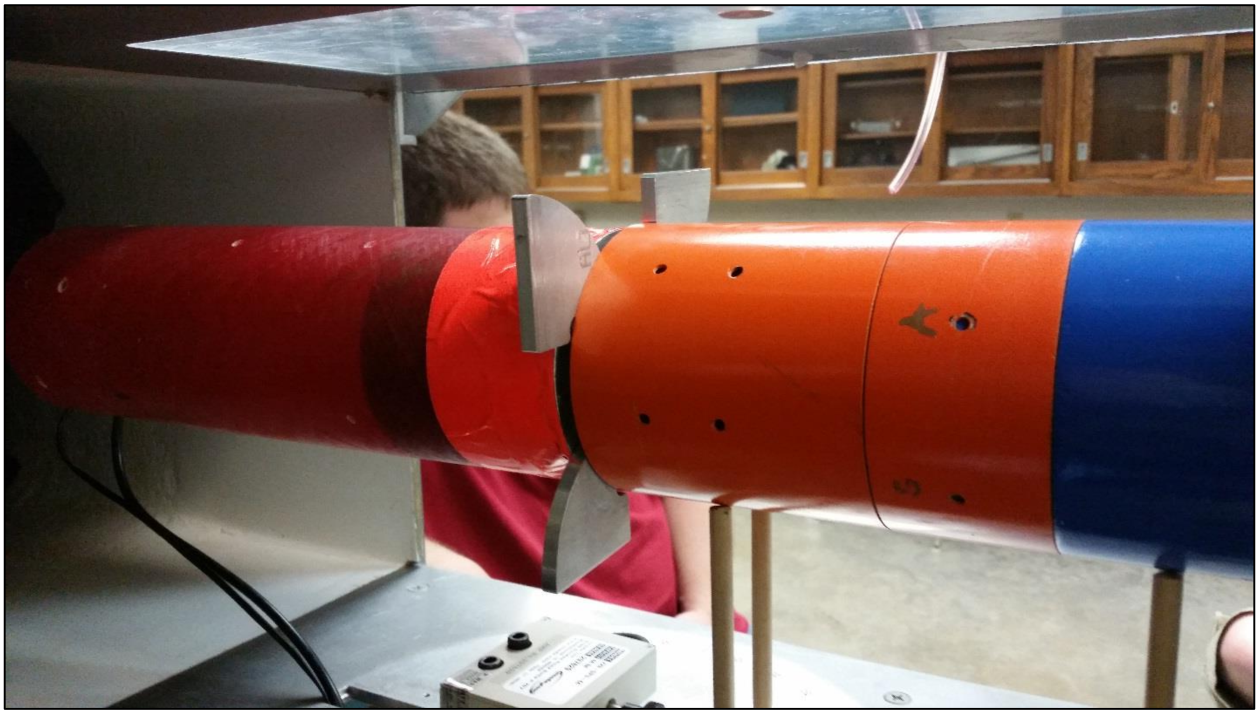

Figure 11: Airbrake being tested in the wind tunnel.

The data collected showed that the airbrakes performed very well. The Airbrake system designed can roughly double the force on the rocket at the airspeeds tested. The drag coefficients were calculated using the relative force between the extended and unextend positions. The drag coefficients are all calculated in reference to the area of the fully deployed brakes. Figure 12 below shows the plot of force vs airbrake deployment and angle.

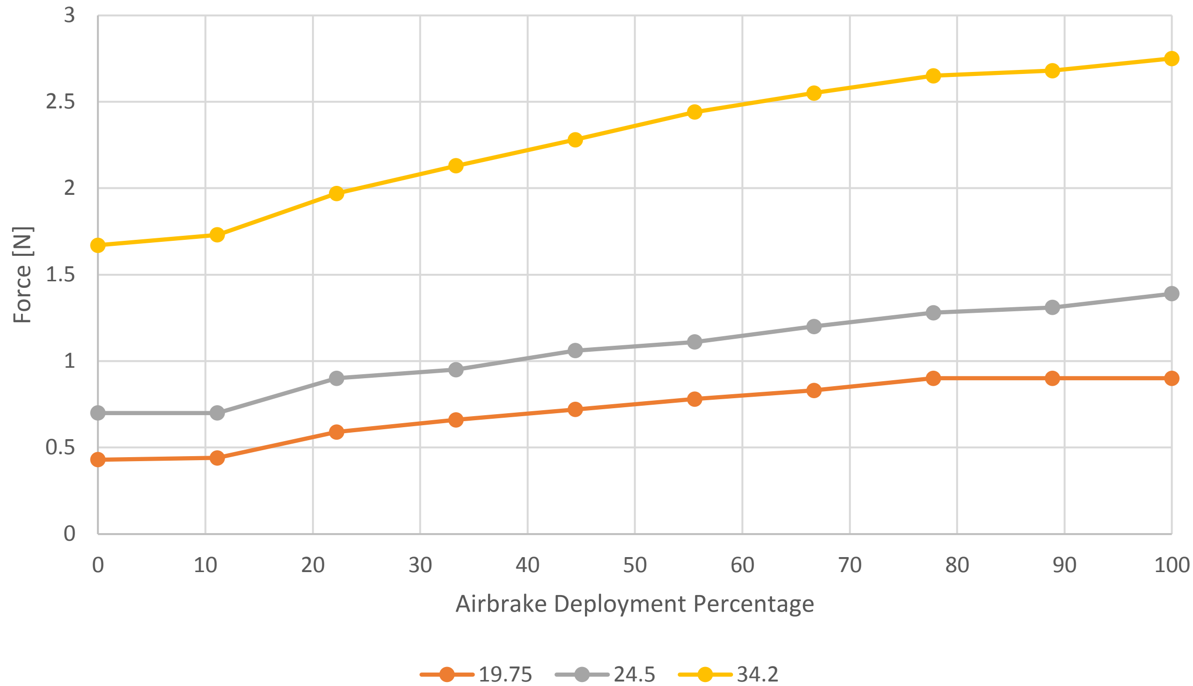

Figure 12: Plot of force vs Airbrake Deployment Percentage. Data from three velocities are shown.

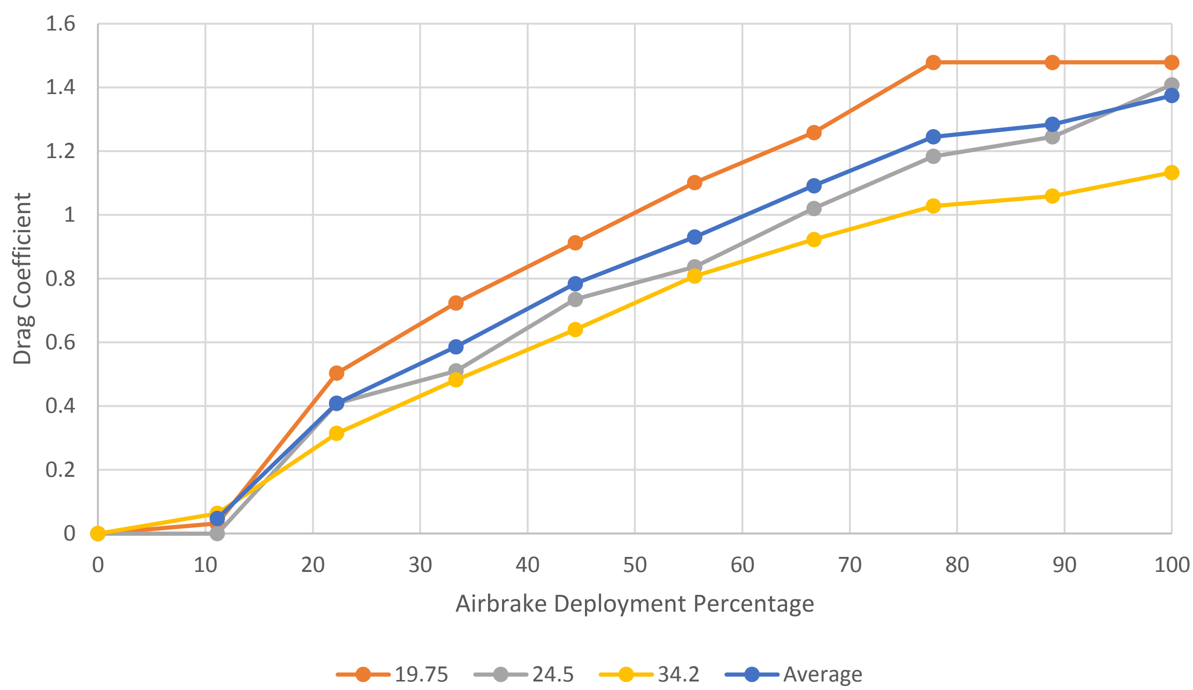

Figure 13: Plot of Drag Coefficient vs Deployment Percentage. Data from three velocities are shown along with their average.

Assessment of Rocket Operation

At the Wisconsin Collegiate Rocketry Launch, the Skybreaker rocket was launched to test the design and air brakes. Before the launch, a safety check was performed on the rocket to ensure it was correctly assembled and ready for the flight. The flight was successful and provided a great deal of information. Off the launch rail, the rocket began to turn but quickly corrected itself and was flying nearly perfectly straight after about 100 feet. Unfortunately, the main deployed at apogee. It appears that this occurred because of the mass of the nose cone. It had steel weights, a pitot tube, and a small electronics bay. When the ejection charge for the drogue separated the rocket, the nose cone at this point appeared to have enough momentum to break the shear pin thereby deploying the main parachute at apogee. To remedy this situation, a second shear pin will be inserted to keep the nose cone attached. The rocket floated much further than was desired, but landed in a set of trees completely undamaged. After the launch the rocket was examined and all of the data was taken from the sensors.

The peak altitude was considerable lower than anticipated. Because of this, the air brake did not need to open as early or as long as had been hoped. It flew to roughly 80% of the predicated apogee when the impact of the air brakes were subtracted from the flight.

There were some issues faced by the electronics however. The data logging to the SD card failed to record any meaningful data. This issue was fixed for future launches, but did mean that no acceleration data was recorded. Furthermore, the telemetry from the rocket went out of range much earlier than was anticipated. This meant that a considerable number of the telemetry was lost. To minimize this in the future, the radio receiver will be suspended higher in the air to prevent any nearby interference on the ground.

Furthermore, because the rocket did not perform as well as had been expected, the target apogee was very close to the actual apogee. This meant that the air brakes only deployed for a very short time and did not provide data on how the airbrakes being opened affected the flight when opened for an extended period of time.

Because no acceleration data was recorded, the coefficient of drag was not able to be calculated. Thus the coefficient of drag vs. time and the coefficient of drag vs. velocity could not be calculated from this test flight.

However, the new Foxeer Legend 1 HD Camera performed flawlessly providing a great deal of data about the air brake performed. This was a significant improvement over the previous sports camera as it provided much higher quality footage. Unfortunately, the data recorded from the flight was poor or nonexistent for some of the sensors so it could not realistically be compared to the camera footage.

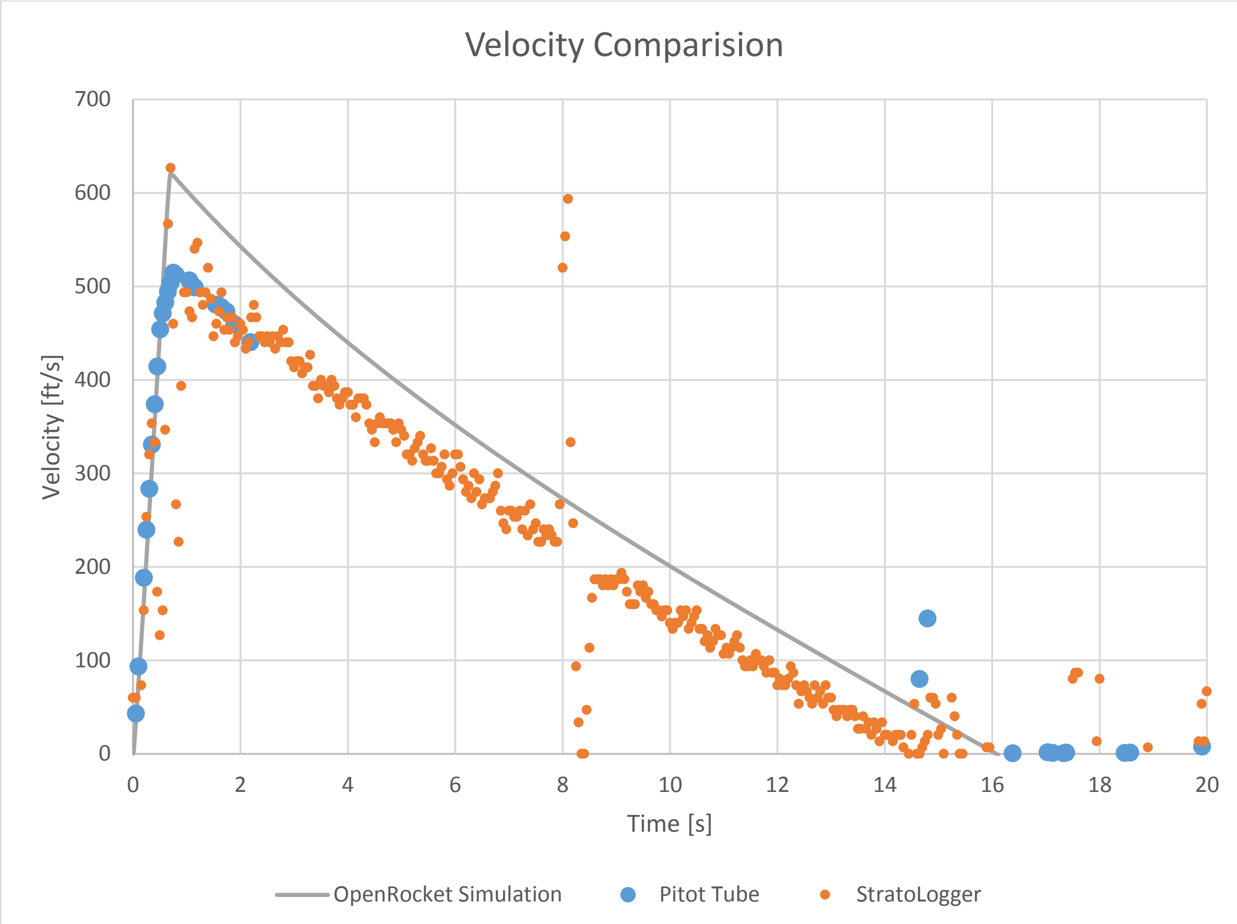

Figure 124 is a comparison of the velocity calculated from the Stratologger, read from the pitot tube, and simulated.

Measurement	Flight Data	Simulation
Peak Velocity [ft/s]	515	620
Apogee [ft]	3553	4436
Mass [lb]	13.8	
Motor	K2045	

Table 1 is a table listing the characteristics of the test flight.

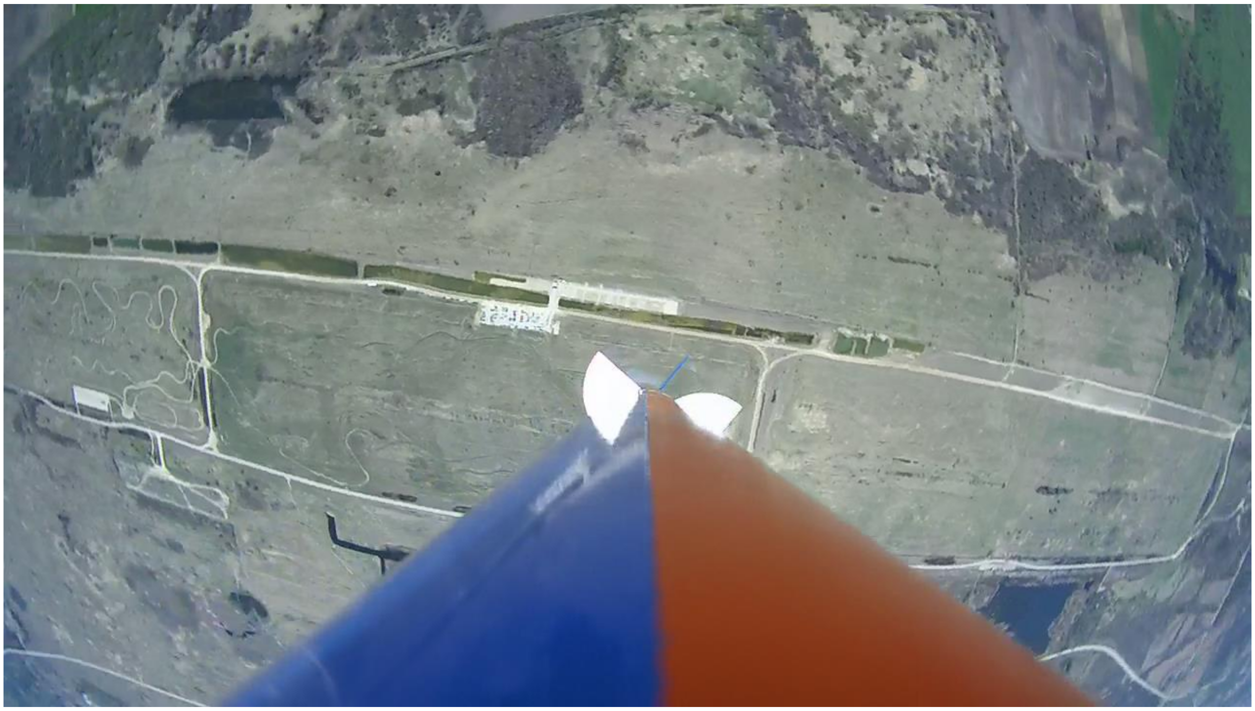

Figure 15 is an image of the air breaks being deployed during the test flight. This image was taken the new Foxeer Legend 1 HD Camera.

Figure 16: Skybreaker after launch.

Figure 17: Team photo before test launch.

Figure 18: Skybreaker after test launch.

Conclusion

Potential improvements to the design would including using an XBee Pro 900 XSC S3B. This would increase the range of the telemetry to 28 miles. Thus eliminating the problem of telemetry being lost when the rocket is at altitude. Furthermore, the components of Skybreaker could be made smaller and lighter to allow the rocket to fly higher on a smaller and less expensive motor.

One of our key findings include the importance of test flights. Variations of the air brake system were test flown four times. Each flight revealed potential problems that were analyzed in order to improve the next iteration of Skybreaker. Another key finding is the importance of thoroughly testing the electronics before flight. The electronic systems on Skybreaker were extensively tested using simulated flights, pressure chambers, wind tunnels, and automotive test platforms. Even then the electronics were not guaranteed to function properly during a launch. There were problems with sudden power loss and failure to record flight data that were unforeseen until the rocket was launched. Therefore, even more tests need to be conducted to test for every eventuality.

Pioneer Rocketry is thankful for the opportunity to participate in the 2015-2016 Midwest High-Power Rocket Competition. We have worked closely with our friends, mentors, and school to help make this project possible. We would like to thank Duane Foust for teaching us how to use the mills, laser cutters, and 3D-printers that we use to build parts for our rockets. We would like to thank the University of Wisconsin-Platteville for hosting our team and allowing us to use their farm for a nearby launch site. We would like to immensely thank the Wisconsin Space Grant

Consortium for their continued support of our team. Last but not least, we would like to thank the Minnesota Space Grant Consortium for hosting this competition and giving us the opportunity to compete.

Appendices

Appendix I: Budget

	Item	Company	Unit Price	Qty	Shipping	Total Cost
[redacted]						
	Jolly Logic Altimeter	Apogee Components	$139.90	2		
11/9/2015	SanDisk MicroSD 2GB Memory Card	All-Out Mobile	$4.98	1		
11/9/2015	Pigtail Cable for DTx U	PerfectFliteDirect	$2.51	3	$7.51	$7.53
11/9/2016	Cable	Digi Key	$20.00	1		
	Breadboard Solderless 300 Tie	Digi Key	$4.50	1		
	Logic Level Converter	Digi Key	$2.95	2		$5.90
	Solder 1MM	Digi Key	$6.50	1		
1/11/2016	Screw Switch	Featherweight Altimeters	$5.00	6	$10.00	$30.00
1/11/2016	MicroSD Card	Adafruit Industries	$14.95	1	$13.79	$14.95
1/11/2016	Pitot Tube Kit	Eagle Tree Systems	$9.99	1	$8.00	$9.99
1/11/2016	Threaded Rod	MSC Industrial Supply	$7.08	1		$7.08
1/11/2016	Zippy Flightmax 350 mAh	Hobby King	$3.19	4		$12.76
	Zippy Flightmax 1000 mAh	Hobby King	$6.14	2		$12.28
	IMAX DC Charger	Hobby King	$15.58	1		$15.68
	Turnigy Nano Tech 750 mAh	Hobby King	$3.10	2	$27.35	$6.20
1/13/2016	Arduino	Digi Key	$36.35	1		$36.35
	Sensor	Digi Key	$16.09	1		$16.09
	Wire Jumper	Digi Key	$3.06	1		$3.06
	Breadboard Solderless 300 Tie	Digi Key	$4.50	1		$4.50
	Bergstik	Digi Key	$1.44	4		$5.76

	Conn Header	Digi Key	$3.76	4		$15.04
	Kit LED	Digi Key	$15.60	1		$15.60
1/22/2016	Adata Micoshc	RadioShack	$49.98	1		$49.98
	9V 2 Pk Batteries	Menards	$3.99	3		$11.97
2/8/2016	Adapter Kit	Digi Key	$10	1		$10
	IC Reg LDO	Digi Key	$1.20	3		$3.60
	Mega Protoshield	Digi Key	$17.95	1		$17.95
	Conn Rcpt	Digi Key	$0.18	25		$4.46
	Conn Plub	Digi Key	$0.15	25		$3.52
2/23/2016	Foam	Tap Plastics	$29.95	1		$29.95
2/23/2016	SM Connector Series	Digi Key	$8.69	1		$8.69
	SM Connector Series	Digi Key	$0.07	50		$3.30
	RCY Series	Digi Key	$0.07	50		$3.45
	G12-2.1-24 - 54mm G12 Fiberglass Tube (2 feet)	Wildman Hobbies	$28.80	1		$28.80
	G12-4.0-60 - 98mm G12 Fiberglass Tube (5 feet)	Wildman Hobbies	$116.725	2		$233.45
	G10-1/8 - 1/8 INCH G10 FIBERGLASS SHEET 1 SQUARE FOOT	Wildman Hobbies	$18.00	4		$72.00
	G12CT-4.0-12 - 98mm G12 Coupler 12 inches long	Wildman Hobbies	$31.20	3	$49.91	$93.60
3/4/2016	Aluminum Threaded Standoffs	Servo City	$1.49	1		$1.49
	Flanged Ball Bearing 2 pack	Servo City	$2.40	5		$11.95
	16 Tooth 32 Pitch Gearmotor Pinion Gears	Servo City	$8.00	5		$39.95
	1/4" Precision D- Shafting	Servo City	$1.09	5		$5.45
	16 Tooth 32 P Futaba Metal Gear	Servo City	$15.00	2	$6.99	$29.98
3/8/2016	New Turnigy		$13.37	2		$26.74
3/8/2016	1/4"-20 Rod	Menards	$2.38	1	$0.13	$2.38
	Foxeer Legend 1 HD Camera	SurveilZone	$89.99	1		$89.99

Table 2 is a summary of the budget. The only addition from the Design Report is the Foxeer Camera.

Appendix II: Analysis Level Class Diagram of Software

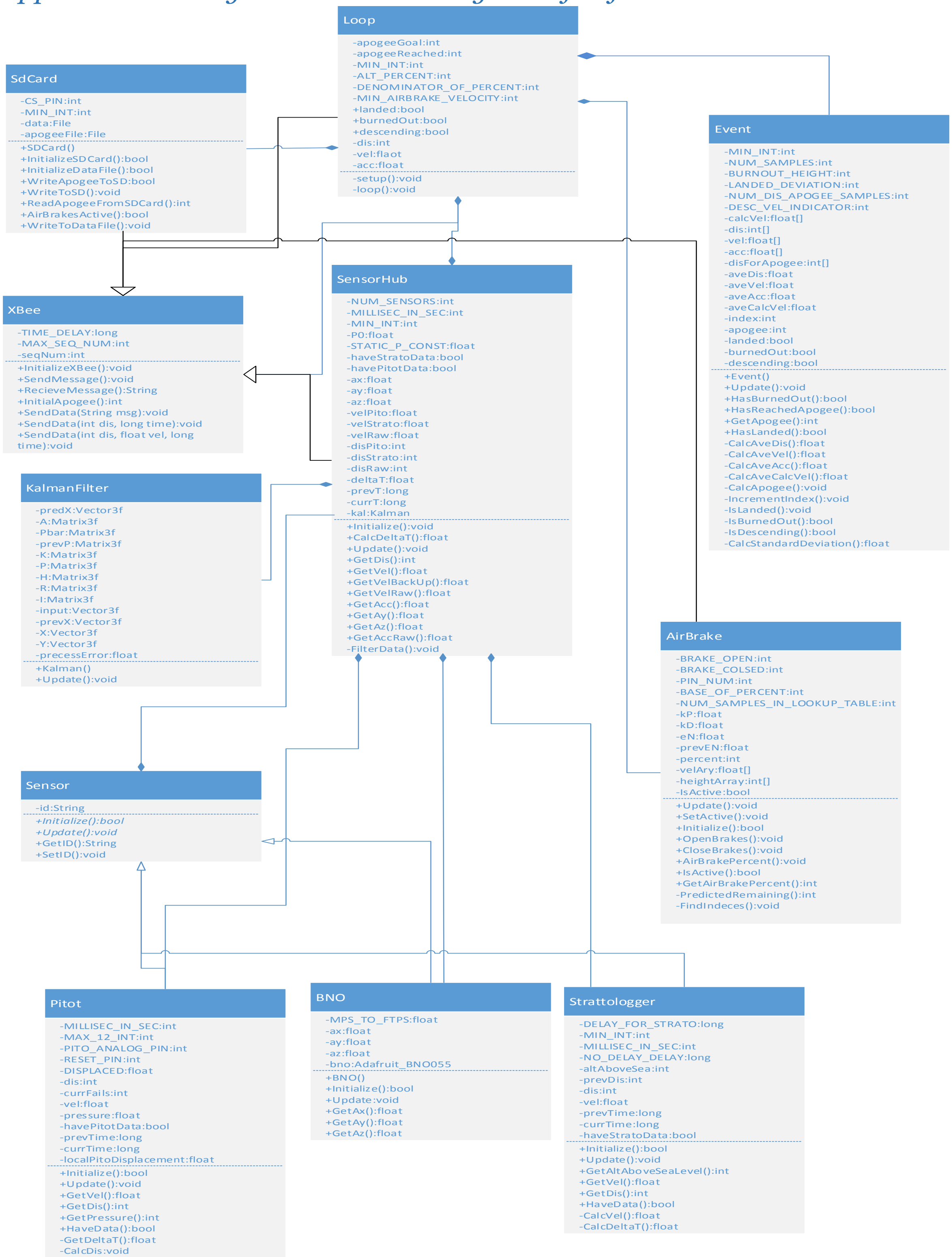

Figure 19 is a Class Diagram of the Control Code for the Air Brakes