Autonomous Parafoil Glider

Executive Summary

The recovery of high-altitude payloads shows a significant logistical challenge, hindering the reuse of valuable scientific equipment. The World Meteorological Organization reports that of the 900-1000 radiosondes launched daily, up to 80% are never recovered[14]. For higher-value assets, they often rely on costly and complex naval “splash-down” operations, which are economically unviable for the growing market of smaller payloads such as CubeSats and university led research instruments. The issue of expendable or expensively recovered hardware shows a need for a reliable, low-cost system capable of autonomously delivering payloads to precise, accessible landing locations.

This proposal outlines the design, development, and validation of a small-scale, autonomous parafoil guidance system engineered to address the technological gap. The system will leverage recent advancements in control systems, complex sensor fusion algorithms, and modern avionics to make precision guidance technology, once reserved for high-budget government programs like NASA’s X-38 Crew Return Vehicle[15], feasible at a university project level. The proposed solution consists of two primary components: an Airborne Guidance Unit (AGU) mounted on a payload up to 5kg, and a portable Ground Control Station (GCS) for mission management and telemetry monitoring. The AGU will integrate a high-precision GPS receiver and an Inertial Measurement Unit (IMU), fusing their data through an Extended Kalman Filter (EKF) to maintain a continuous and accurate state estimate. A guidance, navigation, and control (GNC) architecture employs Model Predictive Control for terminal guidance, commanding servo-actuated brake lines to dynamically steer the parafoil toward a specified landing zone.

The central objective is to design, build, and validate a prototype autonomous parafoil guidance system capable of landing a 5kg payload within a specified error of probability of a designated target from a release altitude of at least 50 feet.

The project will be validated through a series of field testing, beginning in controlled environments such as the University of Michigan’s M-Air Net facility and progressing to open-air locations. Successful completion will show a viable, scalable, and cost-effective solution for precision aerial delivery, with significant potential ranging to the recovery of small satellite and scientific research missions to enhancing atmospheric research and enabling targeted supply drops.

High Level Description

Design Intent and Background

The World Meteorological Organization (WMO) reports that 900 - 1000 radiosondes (weather balloons) are launched daily worldwide, totaling approximately 300,000 launches per year[1]. At the same time, NASA reports anywhere from 15 - 20 yearly scientific balloon launches[2]. Regardless of mission objectives, the end-of-life of these balloons typically involves a simple freefall under a parachute, leaving the payload at the mercy of the jet streams. Currently, the average cost of a WMO compliant weather balloon ranges anywhere from $250 - $500; however, these are single use devices that have a 80% rate of unrecoverability[7]. Some countries have incentives to entice civilians to return the instrumentation if they find it, however, they are considered a collectible in Germany resulting in most never being recovered.

As a result, the recovery of scientific mission payloads is a critical part of mission planning, as the payloads cost in the range of hundreds of thousands, to millions of dollars[3]. However, current recovery methods, which rely on freefall landings, often place payloads in extremely remote or difficult-to-access locations. For example, NASA's Gamma-Ray Imager/Polarimeter for Solar Flares (GRIPS) mission ended up 400 km inland in Antarctica[4], requiring a year of planning and a recovery effort involving hundreds of personnel, naval support, and heavy equipment to traverse the continent.

To extend the scientific mission payloads, it is not just NASA that is looking for a solution to payload recovery. On a smaller scale, University rocketry programs, like the Michigan Aeronautical Science Association (MASA) and the Women in Aeronautics and Astronautics Rocketry Program (WAARP), currently rely on the freefall method described above. With the death of NASA's CubeSat Launch Initiative, research laboratories like the Michigan Exploration Laboratory have to use weather balloons to test satellite technology, and now have to consider the descent of hardware that primarily used to burn up in the atmosphere. There have already been examples of this hardware landing in unrecoverable locations, due to the fluctuating nature of wind, where they still remain to this day.

There are commercially available solutions to this problem, namely Picatinny Arsenal's Joint Precision Airdrop System (JPADS) which can deliver 500 lbs payloads to a landing zone with an accuracy of 150m[5]. Unfortunately, their high price means that they are unviable for University missions and WMO missions, and unattractive for NASA missions that already strain their budgets.

It is clear that there is a pressing need for a low cost autonomous system that can guide missions to landing zones for easier recovery. This would eliminate a large portion of mission planning, wasteful budget, and allow for the WMO and universities to reuse hardware.

Our team proposes that the solution to this problem is a controlled descent using an autonomous parafoil. A parafoil is, in its most basic form, a parachute made in the shape of a basic wing.

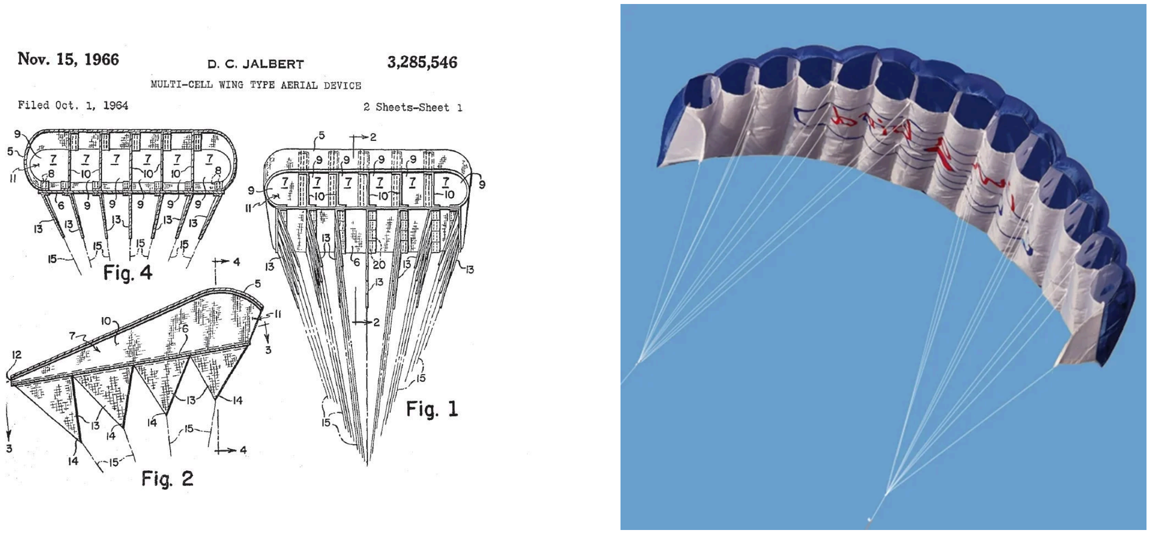


Figure 1: Parafoil Diagram + Picture
Left: Drawing of a parafoil. The bottom left image shows the parafoil's relationship to a wing.
Right: An image of a parafoil in flight.
Source Left: US patent 3285546
Source Right: "1.4 Meters Parafoil Portable Inflatable for Beach Surfing Outdoor Game Toy, Blue." Walmart, Walmart Inc., https://www.walmart.com/ip/1-4-Meters-Parafoil-Portable-Inflatable-for-Beach-Surfing-Outdoor-Game-Toy-Blue/2887870985

If we look at the bottom left of the left image, we can see a side view of a parafoil. In aerodynamics, that shape is commonly referred to as an airfoil. It produces lift as it moves forward through the atmosphere, and a parafoil produces lift in much the same way. The main difference is the strings attached to it. By pulling on the strings, the shape of the airfoil changes, resulting in a change in the aerodynamic profile altering how and where it produces lift. This allows for precise control of the direction of descent.

Our device will primarily be targeting the small-scale scientific community, specifically in atmosphere weather balloon scientific instrument recovery. The goal is to prove the viability of the system that could be scaled to larger payload weights with just an upgrade in parafoil size. Focusing on the high-level mechanical requirements of our system, we will be targeting a controlled descent of 1.5m/s of a payload that weighs a maximum of 5kg. The height of the fall will be discussed in a further section.

Our design proposes a two-part system: the Ground Station and the Airborne Guidance Unit. The Ground Station will be responsible for sending the coordinates of the landing zone to the Airborne Guidance Unit. The Airborne Guidance Unit will be responsible for receiving GPS coordinates from the Ground Station and then continually calculating and executing the best flight path to reach it through the use of a Guidance, Navigation, and Control algorithm.

The Ground Station will send (uplink) a GPS coordinate specified by the user to the Airborne Guidance Unit via Xbee using DigiMesh communicating at 900 MHz for its 22 - 45 km outdoor communication range. Our original plan was to communicate over LoRa using an ESP32, but we decided that in order to add manual control for debugging, we needed a faster communication rate (about a 10x improvement in bps using DigiMesh over LoRa)[6].

The Ground Station will allow the user to send new landing coordinates mid-flight if there is a change in destination. This will also allow for manual control of the flight path if the user wishes to do testing. We will also allow the user to establish a GEOfence that creates no-fly zones for the Airborne Guidance Unit. This will allow the user to case-by-case identify objects that could potentially be hit during flight.

The Airborne Guidance Unit will be an addon that attaches to the top of the payload. In order to minimize space, we have decided to prioritize functionality over appearance. It will be responsible for storing the parafoil during flight. At the end of the mission, the payload will terminate the weather balloon, or it will burst due to expansion at altitude. Our system will use a suite of sensors paired with an extended kalman filter (EKF). The EKF will not only be the primary Guidance, Navigation, and Controls (GNC) algorithm for our system, but for portability we will also be using it to detect when descent begins. It does this by using data from an accelerometer, gyroscope, GPS, and altimeter to estimate linear acceleration, velocity, and orientation to detect a fall[7].

Looking at the technical details of the mechanical system of the Airborne Guidance Unit, our system is responsible for storing the parafoil during the mission and deploying it at the beginning of descent. The parafoil will be stored in such a way that it does not tangle before or after deployment. How we accomplish this is still under discussion and will most likely be determined through trial and error or with expert guidance from Dr. Cutler and the Michigan Exploration Laboratory, who have experience in this area. The control lines attached to the parafoil will be actuated by servo motors that can generate the necessary 3.1kg/cm of torque, which is approximately 12% of total suspended weight[16]. We will be including four M4 bolt holes on the bottom of our device for attachment to payload. These have been chosen to meet the estimated 1.96 kN the connection will be experiencing during flight (See math below).

$$F_{total} = 5kg \ * \ 9.81m/s \ * \ 20\,kg \ = \ 981N$$

With a safety factor of 8, each bolt needs to support:

$(F_{total} \ * \ 8)/4 \ = \ 1962\,N$ per bolt.

Getting into the technical details of the electronics of the Airborne Guidance Unit, we plan to have a “navigation box” that houses the microcontroller and sensors necessary for determining the position state of the system. This includes: an Inertial Measurement Unit (IMU) to provide information on acceleration, attitude, and orientation in Earth’s magnetic field; a GPS for an accurate sense of location and destination; and a pressure altimeter to gauge approximate altitude in the atmosphere using the 1976 Standard Atmosphere model.

Our system will also be powered by an on-board battery that provides power to the system during the duration of the missions. This battery will be selected to provide the maximum voltage used out of all our components, and with an Amp-Hour capacity to support ideally 2-3 hours of sustained use. Based on our current measurements, the highest voltage our components require is 5 Volts, with a total maximum current draw of 2A. This means that our battery must provide at least 5 Volts, have 4000 to 6000mAh, and support 1 Amp of concurrent current draw.

We have consulted a number of industry experts to get an idea of good testing methods. Jimmy Lowe from Vast who worked on MXL's attempt to make a similar system was one of them. He recommended we check out the documentation that was still available from Dr. Cutler, and gave us a list of testing options. More on this can be found in the Reliable Testing section.

We have identified two possible outcomes that must be accounted for in our GNC algorithm. The first case is that the payload can reach the GPS coordinate without issue. In that event, it should do just that. The second case is that the payload cannot reach the GPS coordinate. An extreme example of this would be dropping the payload off the roof of the FXB and setting the GPS coordinate at the Big House—this route is impossible. The payload would not have the altitude to accomplish that trajectory and would therefore follow the best possible path until it eventually touches down.

Since the primary purpose of our design is to eliminate the need for costly recovery methods for scientific missions, the main concerns for our design are cost and size. Size has been addressed throughout the previous paragraphs, as we have been designing our system around space and mass constraints. In terms of cost, we have identified four major contributors:

1. **The IMU:** The cheapest surface-mount IMU on DigiKey is a little under four dollars. Since our system is autonomous and requires real-time control computations, the frequency of measurement and accuracy must be high. Thus, we need a reliable—most likely more costly—IMU. The high-precision 6-axis Motion IMU (BNO085) costs $13.05 with applicable tariffs and will provide more accurate measurement, thereby improving our calculated flight path.
2. **The GPS:** Similarly, the cheapest surface-mount GPS on DigiKey is $11, which will almost certainly not be the GPS we select, as we again require high accuracy and measurement frequency. A highly reliable, low-error, fast GPS, such as the HL7688_1103055 by Sierra Wireless, costs $58.54 with applicable tariffs. But seeing as our system will be designed for in atmosphere drops, a slightly lower accuracy, cheaper GPS can be used like the NEO-M9N module at $27.00.
3. **The Wireless Communication:** There are many methods of wireless communication that can be utilized, but long-range is needed to properly communicate from ground station to parafoil. LoRa radio communication is a valid option, but it lacks high data-rate and low latency which are desired to allow for manual control of the parafoil from the ground station. Utilizing an XBee module (XB3-24DMUM-J) will combine the long range communication with the desired low-latency at a reasonable price of $18.99.

4. **The Parafoil:** Ripstop nylon, the most common parachute and parafoil material, averages about $8.75 per yard. Although, a commercial hobby parafoil should be sufficient for our system, and we could purchase one on Amazon for anywhere between $20 and $60.

With these four components, assuming the highest cost for all three with an applied 30% tariff (even though current reports indicate that microelectronics do not have a tariff applied), we find that the cost of the Airborne Guidance Unit will be upwards of $155. There are other smaller costs not included in this number, but even with those factored in, it remains a vastly cheaper and more reliable method for payload recovery. Although the cost of a regular parachute can be anywhere from $10 - $188 and beyond, the real benefit of our system comes from the elimination of expensive recovery plans, which cost an estimated $400K[4].

As we discussed our project, we considered different methods for controlling descent. We identified three approaches and ultimately settled on continuing with the parafoil. One of the more promising of the discarded methods was a morphable glider. Essentially, this would involve attaching a system that deployed fold-out wings at the start of descent. The reason it would need to be “morphable” is due to needing to limit the induced aerodynamic drag on the weather balloon. A glider would provide more responsive control and access to a much larger knowledge base, as airplanes and gliders with solid wings have been modeled and simulated for well over a century. The controls for such a device would also be simpler for the same reason.

Unfortunately, we encountered issues such as: how to store usable wings in the confined environment of a weather balloon attachment; how to limit the weight of the wings while maintaining structural integrity; and, most importantly, how to avoid trouble with the FAA for violating International Traffic in Arms Regulations (ITAR). The last point was the primary reason we chose to continue with the parafoil. Adding wings to a system with a guidance algorithm comes dangerously close to Unmanned Aerial Vehicle (UAV) and missile territory, which the FAA would not be lenient about. Under U.S. Code (49 U.S.C §44809) UAVs are only allowed for recreational flying, meaning flying for school or work are regulated under Part 107 Code[8].

Reliable Testing

The nature of our project requires that we be able to repeatedly drop the payload from height in order to test our guidance system. We reached out to Dr. James Cutler, the lead of the Michigan Exploration Lab (MXL) and teacher of the CubeSat development course in the Aerospace Engineering department, as well as Jimmy Lowe from Vast who worked on MXL’s attempt to make a similar system. Both provided invaluable feedback and advice for testing this system.

The most important advice that was given was on unit testing and general testing. Both Dr. Cutler and Jimmy recommended doing bench tests on the system to ensure the GNC algorithm is working properly. Since we know how the system needs to react in order to move left or right, we can set up the Airborne Guidance Unit on a bench and send it a coordinate via the ground station to ensure that the servos move in such a way that if the parafoil were attached it would make the correct adjustments to its heading. This way, we can prevent wear on the system by avoiding drop tests until we are sure that the GNC works in a controlled environment.

They both also agreed that doing drop tests from a drone in M-Air was a good idea, as it is a relatively controlled environment. This would allow us to see how the GNC algorithm works during a live free fall and get data for optimizations and corrections.

Dr. Cutler was also very adamant that there are several parks and nature areas around the Ann Arbor area that have places where a test like this can be held. He speaks from experience on this point and was keen on letting us know that the previous group, Jimmy Lowe's group, did test at Peach Mountain by dropping it off the antennae that they have there that UofM owns and operates.

We also reached out to several small aviation companies that have access to airports or airstrips to see if we could do testing at their facilities. So far we have heard from Hans Masing, vice president of operations at Solo Aviation Services who let us know that Ann Arbor Municipal Airport is a controlled airspace that does not allow drone flights. He did also let us know of Scio Flyers, a very active radio control aircraft club in Scio Township that has a dedicated drone and RC airplane flying park. We have reached out to them to see if we can use their facilities for testing.

Kenney Farmer, the owner of Belleville airport, responded to our request to use his airport. We believe it is the perfect location for drone testing. In his response, he indicated that he is confident that he can help us and we are currently working to schedule a conference call with him.

Two other possible opportunities for testing come from the CubeSat Flight Laboratory (CFL). Dr. Cutler is trying to get a reusable helium balloon that can be used to test these kinds of systems. We are also launching CubeSats on a high altitude weather balloon around the time of the Project expo, so there is a possibility that we can include our hardware in that flight to have a test on an actual descent.

Although helium weather balloons would be ideal for our testing, it is unfortunately not a viable option for our budget. To lift a 5kg payload, we would need a little less than 3 m^3 of helium, which costs on average $190 without including shipping, handling, storage, and any other misc fees. This would only be enough for a single launch - so essentially a fifth of our budget would be used on a single test flight.

To ease our testing, we have reached out to several building managers to request permission to their spaces for performing flight tests and demonstrations under their building's necessary safety guidelines. These sites include the FXB, BBB, EECS, and Ford Robotics buildings on North Campus. As of right now, we have received confirmation that EECS will grant permission given we follow their safety guidelines. However, we are still waiting on hearing back from the building managers of the FXB, BBB and Ford Robotics Buildings. We are confident that with the encouragement and methods outlined by Jimmy and Dr. Cutler that we can sufficiently test our system.

Schedule

9/18/2025 - 9/31/2025	*Design and part selection* Once the proposal meetings are done, assuming we get the all clear to continue, order at least two radios and the selected GPS. Select the IMU solution and order it. Select a processor and order a dev kit if it is available. Construct a prototype/board that can talk to the radio and GPS. Submit the final proposal by the 25th. Order all of the parts needed for the prototype by the 27th.
10/1/2025 - 10/15/2025	*Prototype and PCB Design* Get prototype for payload and ground station functional. This means that the payload and ground station should be able to communicate and the payload should be able to track its position with GPS. Payload should also be able to monitor its position state using the IMU. Order the PCBs by the 15th (more than a week before the deadline to account for issues).
10/15/2025 - 10/29/2025	*Software development and testing on prototype* While waiting for PCBs to arrive, the communication between payload and ground station should be completed using the dev kit and prototypes. Get GPS working 100%, including the ability for the payload to track its position and for the ground station and payload to track target position. IMUs should be integrated 100% to allow measurement of the vehicle's motion and orientation. Final date to order a PCB: 10/28 in the event of issues. Test prototype in Aero department wind tunnels.
10/30/2025 - 11/19/2025	*Software development and testing on PCB-based system* Test the PCB. Accuracy of GPS and IMUs will be tested. Communication protocol will be developed to send packets between payload and ground station. Bugs will be found and fixed. PCB issues resolved. Any additional features and stretch goals will be decided on at this time. PCB should be fully assembled by 11/3.
11/20/2025 - 12/4/2025	*Final testing, additional features, poster and expo* Targeting a completion date of 12/1. We will then get our poster done and start on the final report.
12/5-12/6	*Finish final report*

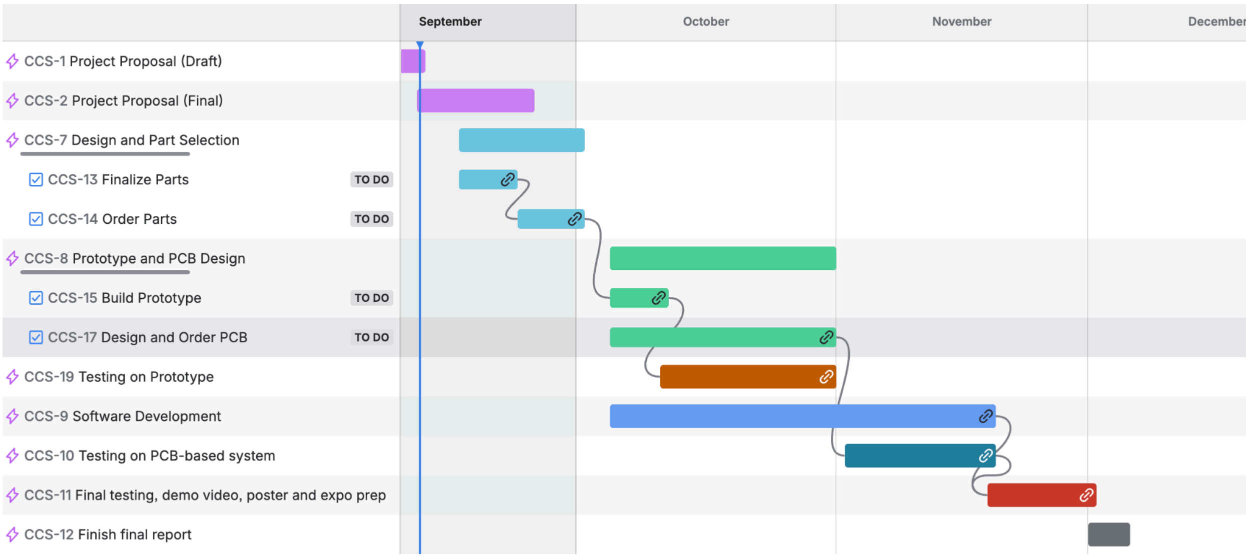


Figure 2: Gantt Chart

The major parts of this project involve ordering the parts and testing, designing the PCB and ordering it, and writing the software. Dependencies exist between finalizing our parts after having a meeting with staff, ordering the parts, and building the prototype. Another dependency is between designing and ordering the PCB and testing it. The software and completed PCB build needs to come together for final testing all together in November. Buying and ordering parts are what's most likely to cause problems. Whether that be shipping delays or parts becoming sold out when we go to order them. Our schedule has generous amounts of time for the sections that involve ordering parts in hopes of ordering early and being able to start early but also being able to wait a few days if things get delayed.

Milestone 1 - Prototype

By the end of building and testing of the prototype, we hope to have functioning XBee communication, GPS tracking, and servo/motor control working. XBee communication will be considered successful if we can send messages from the EECS 473 lab to the South West entrance of the Duderstadt which are approximately 230 meters apart from each other (according to Google maps). This allows us to recreate communication from the distance of the payload in the sky to the ground station. We will test GPS by printing out changes in the GPS to a monitor to demonstrate that the payload is getting updated positioning. Additionally, we will implement a way to control the servos manually using a controller or laptop to ensure that the servos are working as expected before attempting to control them via GPS. Then we can begin integrating the parts together. We can use the GPS to control the servos by moving the payload relative to the ground station and watching the servos move accordingly and we can send commands and updated GPS positions between the payload and a computer with XBee.

Milestone 2 - PCB Build

This milestone will be the working combination of all the components into one device that is able to successfully land within a specified target. This will involve being able to drop our device from any height and have it land within a reasonable range of the ground station. The acceptable range depends on how far away and from how high the payload is dropped from the destination location. Our function for determining the acceptable radius is:

$$radius = \sqrt{\left(\frac{1}{10} \cdot height\right)^2 + \left(wind\ speed \cdot \sqrt{\frac{distance}{9.81}}\right)^2}$$

Height and distance will be measured in meters and wind speed will be in meters per second which can be determined on the days of testing. Wind speed is important for determining the landing radius as it will affect our parafoil's ability to travel. A drop will be deemed successful when our guidance algorithm is able to navigate the payload to a destination location and land within the calculated range. We will be testing our device from different heights, distances from destination, and in different weather conditions to see how our device manages. As mentioned in the testing section, we have the ability to test outside, in the elements, as well as in the EECS building which eliminates the wind concern. Testing in the EECS building will benefit us early on as we are testing the guidance and we can be sure that any movements are a result of our program and not wind.

Design Expo and final presentation

For the design Expo and final presentation, we intend to at least demonstrate the device's ability to control servos based off of the payload GPS relative to the ground station GPS without physically being dropped. Our fully functioning device needs to be dropped from a high place in order to deploy the parafoil and control the payload. Ideally, we are able to get a sectioned off area in EECS to do at least one demonstration, but we know that this is unlikely. Alternatively, we hope to schedule a live demo at some point before the Expo for staff and anyone interested to see. During the Expo, we will have a video of the parafoil device being dropped to show that the device does work as intended despite not being demonstrated at the Expo.

Budget

While this early on a realistic detailed budget is difficult to create, a "line in the sand" projection is possible. The numbers below are reasonable estimates at first glance.

Total Budget: $200/person x 5 people = **$1000 budget**

Item	Quantity	Cost per	Cost Total
STM32H743ZIT6	3	$16.37	$49.11
NUCLEO-H753ZI	1	$27.58	$27.58
XB3-24DMUM-J	3	$18.99	$56.97
SparkFun XBee 3 Wireless Kit	1	$87.95	$87.95

BMP581	5	$2.60	$13.03
Shuttle Board 3.0 BMP581	1	$16	$16
BNO085 Adafruit IMU	1	$24.95	$24.95
BNO085	5	$11.85	$59.26
NEO-M9N Module	5	$27	$135
Sparkfun GPS Breakout - NEO-M9N	1	$71	$71
Parafoil	5	$19.99	$99.95
Airborne Guidance Unit PCB	5	$20	$100
Servos	4	$20	$80
Battery	2	$30	$60
Buck Converter / Voltage Regulator	1	$17	$17
Antennas	3	$2.50	$7.00
Antenna for GPS	1	$32.30	$32.30
Passive components, connectors, misc.	-	~$10	~$10
Estimated Total			$947.10

Implementation Analysis & Issues

An analysis for each of our design features are addressed as follows:

- **Processor: STM32H743ZIT6**
    - **Description:** This processor provides the necessary digital interfaces for the project. It provides PWMs for controlling our motors, as well as the necessary communication protocols. Ordering this component by itself is necessary for our PCB mounting.
- **Processor Development Board: NUCLEO-H753ZI**
    - **Description:** Development board for STM32 processor to allow for rapid testing and prototyping of our system. Includes useful STM32 Cube software libraries and easy to use debugger.
- **Wireless Communication:XB3-24DMUM-J**

    - **Description:** Long-range, high data rate wireless communication that is ideal for our applications. The primary function of the wireless link is to transmit the target landing coordinates from the ground station to the payload. We may also look into receiving real-time telemetry for monitoring during testing.
- **Wireless Communication (Development Board Kit) :** [SparkFun XBee 3 Wireless Kit]
    - **Description:** This kit includes two XBee module dev boards with a built-in microcontroller for easy programming. This will allow for rapid prototyping before we develop our PCB. The kit also includes an XBee Explorer which will convert from USB to serial and interface with the XBee module. This will be useful to allow our ground station computer to send and receive data wirelessly.
- **IMU:** BNO085 Adafruit IMU
    - **Description:** The BNO085 is a 9-axis System in Package (SiP) which integrates a triaxial accelerometer, gyroscope, and magnetometer with an ARM Cortex M0+ MCU running a powerful sensor fusion firmware. This offloads complex calculations from our main processor for guidance and control tasks.
- **Pressure Altimeter (Sensor):** BMP581 Sensor **(Requires: SPI, I2C)**
    - **Description:** A Barometric Pressure to read air pressure while the device is in flight. The main purpose of this component is to convert the pressure reading into height using the 1976 U.S. Atmospheric Model, which accurately estimates altitude based on pre-established pressure estimates.
- **Pressure Altimeter (Development Board):** Shuttle Board 3.0 BMP581
    - **Description:** It will likely be necessary to learn the interface of this sensor in a setting for rapid prototyping and rewiring. As such, we will also purchase a development board from the same manufacturer to ensure ease of transfer when it comes time to design the PCB
- **GPS Communication:** NEO-M9N Module **(Can Use: UART, SPI, I2C)**
    - **Description:** The NEO-M9N provides superior positioning reliability because it can receive signals from four satellite constellations, GPS, Galileo, GLONASS, BeiDou, at the same time. This allows for a faster and more dependable position lock, which is critical for navigation in environments that obstruct the view of the sky.

Current & Power Details

Component	Sensor / Mode	Current Draw	Low-Power Mode Current	Power Consumption
BNO085	6 and 9-Axis Sensor Fusion	11mA	0.13 mA	35.25 mW

BMP581	High Resolution (30 Hz)	80uA	1.3uA	N/A
NEO-M9N Module	Peak Current	100mA	N/A	N/A
NEO-M9N Module	Continuous Current	36mA	N/A	N/A
STM32H753xI	Scale / VOS 1, Caching On, All Peripherals On, Typical Current Draw	176mA	N/A	N/A
STM32H753xI	Scale / VOS 1, Caching On, All Peripherals On, Max Current Draw	544mA	N/A	N/A
XB3-24DMUM-J	Transmit Current	135mA	2uA	N/A
MG995R Servo	Rotation	~1.2A	N/A	N/A

Max Voltages

Component	BNO085	BMP581	NEO-M9N Module	STM32H753xI	XB3-24DMRS-J	MG995R
Max Supply Voltage (VDD)	3.6 Volts	3.6 Volts	3.6 Volts	3.6 Volts	3.6 Volts	4.8-6.0 Volts

Battery Selection

For our battery, we've decided to go with the**Turnigy Heavy Duty 5200mAh 3S 60C LiPo Battery Pack w/XT90**. This is not only a rechargeable battery, but has been known for its reliability and durability over time in hobbyist electronics. Additionally, it can handle a fair amount of current draw due to having a 60 Coulomb rating. To address the voltage difference from the battery to our max voltage (from 11.1 Volts to 5 Volts), we will be using a **DROK Buck Converter** to convert the high voltage down to what it needed for our design.

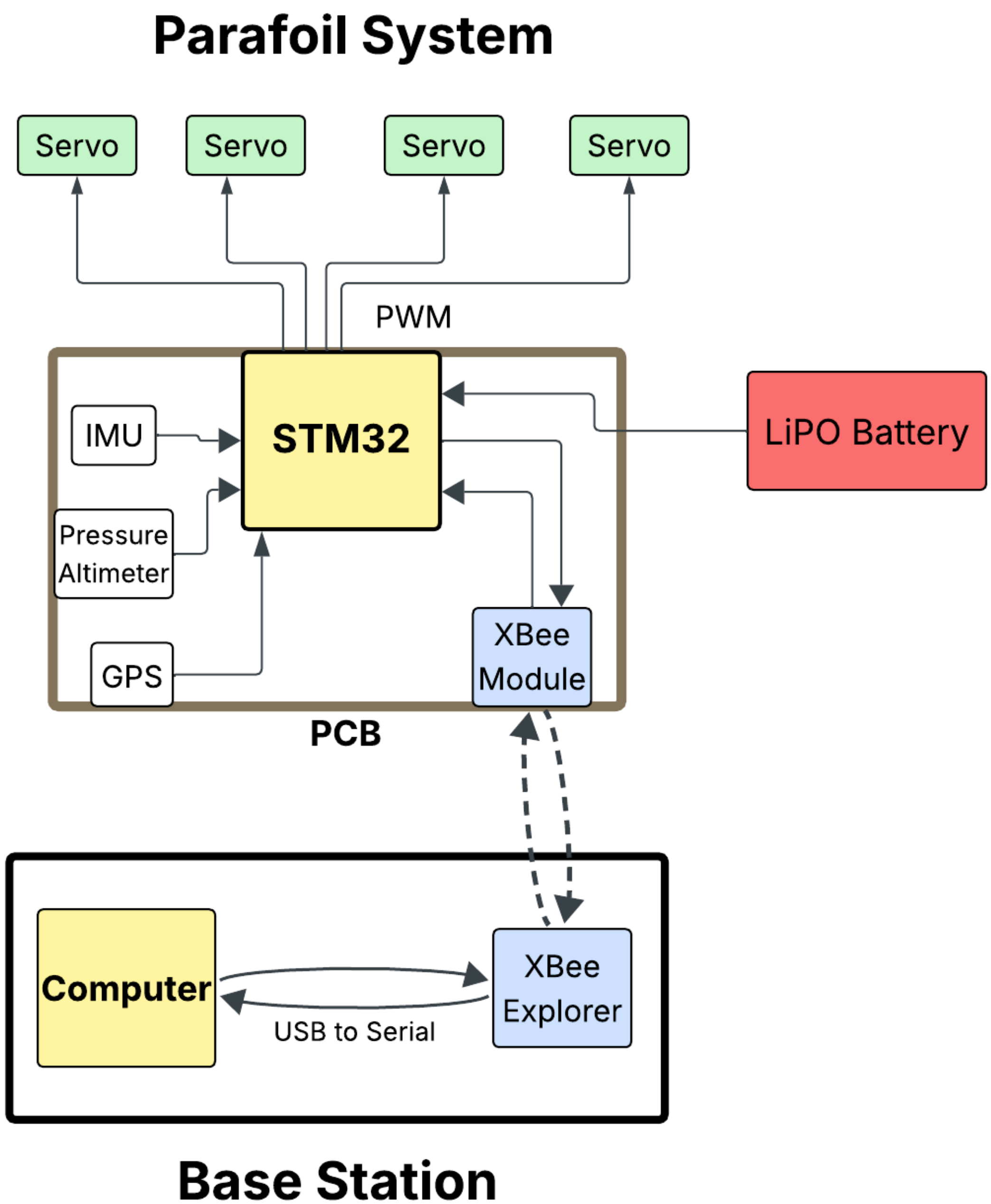


Figure 3: System Component Schematic

The largest concerns at this time are

1. Reliable Testing
2. Aerodynamic control of parafoil
3. Developing Guidance, Navigation, and Control algorithms for precise landing control.

Aerodynamic control of parafoil

As most of us have little to no experience in aerospace engineering, a chunk of the project will be out of our expertise. Our amazing team member Max will be able to help here with his aerospace knowledge, but of course he can't do everything himself so this will require extra effort from the rest of the team. We have done some initial research into the aerodynamic control of the parafoil which are as follows:

Since our system's primary target is CubeSats, we know that the mass of the payload and our system can be no more than 3kg. Knowing this, we can calculate the lift that we need to generate in order to gently fall.

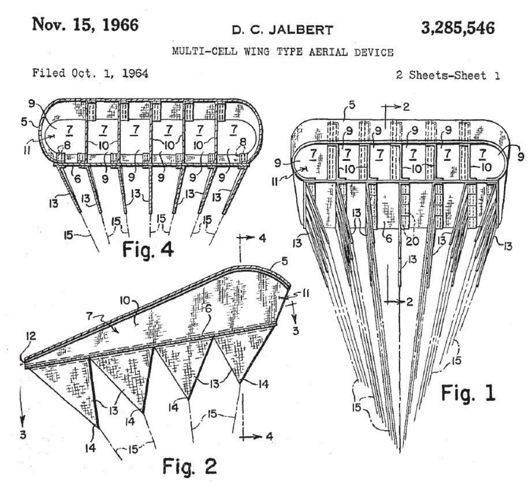


Figure 4: Drawing of a parafoil. The bottom left image shows the parafoil's relationship to a wing.

Source Left: US patent 3285546

As we can see again in Figure 4 above, the side profile of the parafoil shows that it is just a thin airfoil. In order to glide, the lift needs to be approximately equal to the mass times the acceleration through the use of Newton's 2nd law. We can achieve different lifts by changing the size of the parafoil. The area of the parafoil is called the planform area, and can be found by rearranging the equation for the lift of a thin airfoil. Note that we are assuming that we are falling at a speed of 1.5 m/s and moving horizontally forward at a speed of 5 m/s. This also assumes we are falling in steady level flight, meaning that our speeds are not changing and lift is equivalent to the force of gravity (no acceleration downward).:

$$L = \frac{1}{2}\rho * u^2 * S * C_L$$

Where

$\rho = air\ density = 1.225\ kg/m^3$ at sea level

V^2 = forward velocity, which for the purposes of this discussion we will assume is 5 $\frac{m}{s}$

S = planform, or wing area

C_L = coefficient of lift, which depends on the shape of the airfoil. Typically around 0.8 - 1.2

With this information, we can rearrange the formula to solve for the planform area and find the maximum size of the parafoil:

$$S = \frac{2L}{\rho * u^2 * C_L} = \frac{2(5 * 9.8)}{1.225 * 5^2 * 0.7} = 4.57\,m^2$$

We can lower the area of the parafoil required by either flying at a faster speed, or lowering the payload mass.

Developing Guidance, Navigation, and Control Algorithms

The GNC subsystem serves as the autonomous brain of the parafoil system. It is primarily responsible for processing data from onboard sensors, determining the vehicle's current state, calculating the optimal path to land, and executing precise control movements.

State Estimation Framework: Justification for the Extended Kalman Filter

An accurate, continuous, and high-frequency understanding of the parafoil's state, whether it be its position, velocity, and altitude, is the start of any autonomous guidance system. However, no single sensor can provide the full picture. A multi-sensor combination is required, but each component will have its own limitations that require a fusion algorithm to create a unified, reliable state estimate. The proposed sensor combination will include an Inertial Measurement Unit (IMU), a GPS receiver, and a barometric altimeter, each with its own strengths and weaknesses that make them complementary.

- **Inertial Measurement Unit (IMU):** This will provide high-frequency updates of angular velocity and linear acceleration, which are essential for real-time flight stabilization. However, when these measurements are integrated over time, grabbing velocity and position, errors from the noise and bias will accumulate without limits, also known as "drift"[9].
- **Global Positioning System (GPS):** Delivers absolute, drift-free measurements of position and velocity. Its long-term accuracy will be its key strength, but its low update rate is insufficient for high-speed control feedback. GPS signals can also be susceptible to noise and signal loss[9].
- **Barometric Altimeter:** Offers rapid and sensitive updates on the relative altitude, which will be crucial during the final landing phase. However, its absolute accuracy can be poor, and readings drift with changes in atmospheric pressure, making it unsuitable as the only altitude source[9].

The goal of sensor fusion is to combine the high-frequency data from the IMU with the long-term, absolute accuracy of the GPS, as well as the updates on relative altitude to create a single state estimate that is more accurate, reliable, and available at a higher frequency than any individual sensor could provide on its own.

The EKF as the Optimal Solution

The Kalman filter and its other variants are industry standards for sensor fusion in navigation systems, but it is under the assumption that the system dynamics are linear. The flight dynamics of a parafoil are inherently non-linear, with disturbances such as aerodynamic force and 6-DOF motion. Therefore, a standard Kalman Filter is just not suitable for the application[10].
The **Extended Kalman Filter (EKF)** is the logical solution for non-linear systems. It works by linearizing the non-linear models surrounding the current state estimate at each time step using a first-order Taylor series expansion[11]. The EKF operates in a constant two-step cycle:

- **Prediction Step:** Between each GPS update, the EKF uses the system's dynamic model and the high-rate IMU data to predict the next state. This ensures the GNC system always has a fresh estimate to control calculations[9].
- **Update Step:** When a new GPS measurement arrives, it is used to correct the predictive state, removing the accumulated drift from the IMU and preventing the error from growing[9].

A key advantage of EKF is its efficiency. It is a recursive algorithm, so it does not require a history of past measurements. That makes it memory-efficient and useful for real-time implementation on resource-constrained microcontrollers. While more advanced filters like the Unscented Kalman Filter can offer higher accuracy in very non-linear environments, they often come with higher compute costs that could negatively affect real-time performance on our hardware. The EKF represents an optimal engineering trade-off between performance, efficiency, and making the implementation of our project feasible.

Backing of the Extended Kalman Filter

EKF is not a new or experimental algorithm developed from scratch for this project. It is an 'off-the-shelf' technique in modern navigation and state estimation. It has been a cornerstone of aerospace and robotics research for decades and serves as the basis for sensor fusion systems in most commercial and open-source UAV autopilot systems, such as Ardupilot and TinyEKF [4, 5]. The algorithm is very well documented, and numerous libraries and implementation guides are publicly available. By choosing EKF, we are building upon a well-understood, widely validated technology, which significantly reduces implementation risk.

Methods of Collision Avoidance - Geofencing

We are going to use geofencing as an adaptive way to block off certain areas that we do not want the parafoil to go towards. If there are known buildings, mountains or any other obstacles, we will have a way for them to be blocked off by entering their coordinates and dimensions beforehand so that the guidance algorithm knows to avoid those regions.

Safety Protocol

Safety should be the primary consideration for all of our operations. To ensure a safe testing environment, we will use a protocol of risk assessment, procedural checks, and emergency plans. All controlled and open tests should be governed by a standardized Test Procedure and Report document, which is a pre-test safety review.

General Safety Procedures

- **Personnel Roles:** All field tests will involve designated roles: a Test Conductor, a Pilot/Operator, and a Safety Officer that is responsible for monitoring the area.
- **Pre-Execute Checklist:** A mandatory checklist to be used before every test, verifying hardware integrity, software status, battery percentages, and communication link quality.
- **Protective Hardware:** Payload is to be designed to protect electronic components from impact.

Operational Safety

- **Compliance with Regulations:** All flights should be conducted in accordance with FAA regulations.
- **Weather Monitoring:** Go/no-go conditions for wind speed or precipitation will be enforced.
- **Controlled Airspace:** Initial integration tests will be confined in controlled locations such as UofM M-Air Net facility to mitigate risks due to uncontrolled flight.
- **Clear Landing Zones:** Open-air tests to be conducted in pre-approved locations with landing zones free of property or people.

Failsafe planning

- **Autonomous Failsafe:** The GNC firmware should have a maneuver that can safely slow the descent of the system in a tight movement pattern. This can be activated if the system's sensors fail or if it is in an unrecoverable flight state.
- **Manual Override:** The Ground Control Station should provide capabilities for the operator to manually override the autonomous system in the event of unforeseen circumstances.

Additional Stretch Goal

An additional stretch goal would be enabling manual control of the flight from the ground station. While this sort of defeats the purpose of having an autonomous parafoil, we think it could be useful in certain scenarios, and it would also be fun to have. A scenario where this would be useful is when the payload detects an obstacle that it thinks it can't go around, but you can see clearly that there is enough space to navigate around it and still land on target. Then, you can switch to manual mode and do that navigation yourself. We would also like to find a reliable way to cut the parafoil if it ever gets tangled during descent. This would be useful in times when the payload finds itself stuck in a tree.

Conclusion

Our proposal outlines a plan to address the high costs and logistical hurdles of payload recovery by developing a low-cost, autonomous parafoil system.

While we recognize the challenges related to budget constraints, testing logistics, and the complexities of aerodynamic control and nonlinear scenarios, our schedule includes milestones and strategies to reduce risk to address these issues beforehand. The proposed bench and field testing plan such as testing in the M-Air Net, provides a clear path to validating our system.

The success of this project will demonstrate a viable and cost-effective solution for precise payload recovery. This technology has the potential to significantly reduce the cost for CubeSat missions, atmospheric research, and other small scale aerial deployments, making it a worthwhile and impactful engineering endeavor.

Appendix A:

Design Criteria	Importance	Will/Expect/ Stretch
Parafoil safely lands at desired landing coordinates (within a ~20m radius) starting from various heights reaching ~300m	Fundamental	Will
Allow for manual control of the parafoil from ground station	Important	Will
Base station can reliably change parafoil’s destination coordinates	Important	Expect
Parafoil lands at closest possible point to unreachable destination coordinates	Important	Expect
Parafoil’s location, direction, and speed monitored from base station	Optional	Expect
Allow for the parafoil to be automatically deployable when it reaches a certain altitude/speed	Optional	Stretch
Parafoil avoids collisions with large objects/structures	Optional	Stretch

Appendix B:

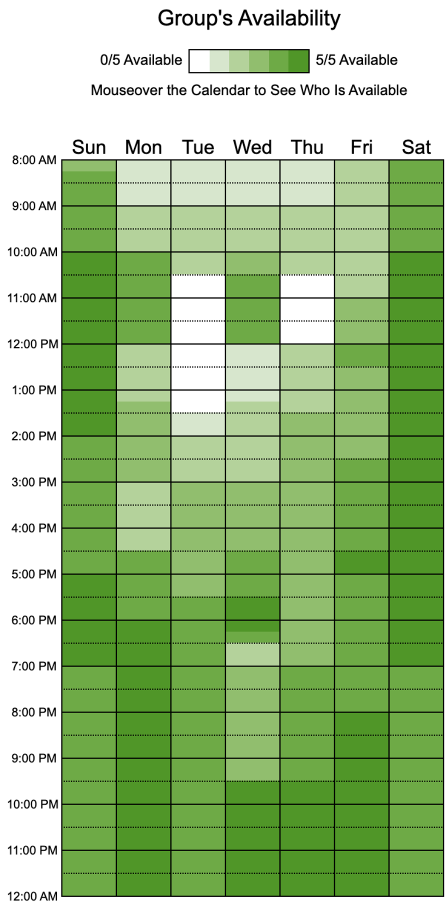
Milestone 1 - Prototyping
Prototype is able to receive GPS information and control servos/motors accordingly.
Information is communicated between ground station and payload via XBee modules. Range is tested across campus. EECS -> Duderstadt is good. FRMCB -> Pierpont would be fantastic
Servo control driver is written, tested, and working (motors move on command)
GPS, motor, XBee interface modules are written and actively being tested
Devices have been integrated with at least one other device.
Milestone 2 - PCB Design
PCB assembled
Payload lands within our acceptable radius of ground station GPS coordinates. $$radius = \sqrt{(\frac{1}{10} \cdot height)^2 + (wind\ speed \cdot \sqrt{\frac{distance}{9.81}})^2}$$
Guidance algorithm is completely written and tested
Full device integration. All parts are able to work together through the PCB

Appendix C:

Header files can be found on our GitHub:
https://github.com/MaxKenny2003/EECS-473-parafoilers

Appendix D:

Generally we are expecting 15-20 hours of solid work committed to the project per week per member. This is expected to be outside of the itinerary meeting planned on Monday evenings. Below is an outline of team availability throughout the week. This is subject to change. Further discussions are needed on converting usually lab times into more time working on the project.



- **Meeting Formats:**
    - **Main Meeting Details:**
        - **Summary**: As a group, we plan to hold one weekly all-hands meeting to discuss our individual progress on tasks and plan our goals for the following week. The current planned itinerary discusses pertinent issues for the first hour before breaking off into working on tasks.
        - **Location**: Duderstadt Meeting Room on 3rd Floor
        - **Time**: Mondays 6pm - 9pm
        - **Jobs:**
            - **GPS** - Matteo, Thomas
            - **XBee** - Matteo, Max, Anthony
            - **Servo/motor control** - Shawn, Max, Anthony

            - **Manual Control** - Shawn, Max
            - **Autonomous Control including EKF** - Shawn, Thomas
            - **Altimeter** - Anthony, Matteo
            - **Structural Design** - Max, Shawn
            - **Testing Logistics** - Thomas, Max
    - **Non-Work Meeting Details:**
        - **Summary**: Every other Friday we will try out a new AFFORDABLE establishment for eating. Some ideas already are Panda Express, Olive Garden, Blazin Coop, Burger One, No Thai, and Canes!!!!!! We are starting our dinner traditions this Friday September 26th.
        - **Location**: Affordable Restaurants.
        - **Time**: Biweekly on Fridays in the evenings. Lasting 1-2 hours.
- **Scheduling Conflicts:**
    - **Anthony:**
        - **Thursdays 6:30-9pm: Michigan Hackers Student Organization**
        - **Saturdays OR Sundays: 1.5 Hours for Co-op House Meetings**
        - **Saturdays OR Sundays: 3 Hours of House Labor for Co-op**
        - **Gone (But still in Town) for Thanksgiving**

    Matteo:
        - **Futsal Club - Fridays 6-7 pm**
        - **Thursdays 6:30-8pm IM soccer**
        - **September 20-21 - Visiting family in Canada**
        - **Gone for Thanksgiving**
    - **Maxwell:**
        - **Sundays 10am - 2pm: EECS 485 project team work**
        - **Wedding: Weekend of 19th of September**
        - **Jury Duty: Tentatively starting October 21st in Port Huron, MI**
        - **Gone for Fall Break**
        - **Gone for the Wednesday and Thursday of Fall break**
        - **CubeSat Launch Window: November 17th - 23rd (could be unavailable for a day during this week)**
    - **Shawn:**
        - **Home football games on Saturdays**
        - **Gone for Thanksgiving**
    - **Thomas:**
        - **Sunday 2-4pm: Michigan Aeronautical Science Association**
        - **Wednesday 6:30 pm - 7:00 pm Weekly meetings**

Signatures:

Shawn Trujillo

Matteo Caporuscio

Anthony Young

Thomas Tran

Maxwell Kenny

References

[1] *Amid 'spy balloon' controversy, WMO highlights key role of weather balloons in climate monitoring*, UN News, 17 February 2023, available at https://news.un.org/en/story/2023/02/1133652

[2] NASA — Scientific Balloons, "Scientific Balloons," NASA website, available at https://www.nasa.gov/scientificballoons/

[3] NASA https://www.nasa.gov/news-release/nasa-awards-nasa-balloon-operations-contract/

[4] *SVS – Balloon Mission Trajectory Visualization*, NASA Goddard Space Flight Center, https://svs.gsfc.nasa.gov/12522

[5] *Picatinny Engineers Ensure Software Reliability for Precision Airdrop System*, U.S. Army, https://www.army.mil/article/178707/picatinny_engineers_ensure_software_reliability_for_precision_airdrop_system

[6] *Digi — XBee PRO 900HP RF Module*, Digi, https://www.digi.com/products/embedded-systems/digi-xbee/rf-modules/sub-1-ghz-rf-modules/xbee-pro-900hp

[7] Leela Kumari. B, Padma Raju. K, Chandan V.Y.V, Sai Krishna. R, V.M.J. Rao, *Application of Extended Kalman Filter for a Free Falling Body Towards Earth*, International Journal of Advanced Computer Science and Applications, Vol. 2, No. 4, 2011, pp. 134–140, available at https://thesai.org/Downloads/Volume2No4/Paper%2020-Application%20Of%20Extended%20Kalman%20Filter%20For%20A%20Free%20Falling%20Body%20Towards%20Earth.pdf

[8] *49 U.S. Code § 44809 – Exception for limited recreational operations of unmanned aircraft*, GovRegs, https://www.govregs.com/uscode/title49_subtitleVII_partA_subpartiii_chapter448_section44809

[9] Sensor Fusion for Drone Position and Attitude Estimation https://www.researchgate.net/publication/393590343_Sensor_Fusion_for_Drone_Position_and_Attitude_Estimation_using_Extended_Kalman_Filter

[10] Autonomous parafoil system precision landing via closed-loop guidance and control considering six-degree-of-freedom model https://www.researchgate.net/publication/377206507_Autonomous_parafoil_system_precision_landing_via_closed-loop_guidance_and_control_considering_six-degree-of-freedom_model

[11] High-Order Extended Kalman Filter for State Estimation of Nonlinear Systems https://www.mdpi.com/2073-8994/16/5/617

[12] TinyEKF https://github.com/simondlevy/TinyEKF

[13] Ardupilot

https://ardupilot.org/copter/docs/common-apm-navigation-extended-kalman-filter-overview.html

[14] Radiosonde Recovery

https://www.weather.gov/media/key/KEY%20-%20Weather%20Balloon%20Poster.pdf

[15] NASA X-38

https://ntrs.nasa.gov/api/citations/20100033241/downloads/20100033241.pdf

[16] Gupta, B., Kumar, V., Krishna, R., *Estimation of Load on Control Lines of Ram Air Parachute Designed for Precise Delivery Using 9-Dof Model*, Journal of Aeronautical Science and Technology,

https://www.joast.org/index.php/joast/article/download/534/523/1021