

UAV-UGV Docking and Charging System

A Major Qualifying Project (MQP) Report
Submitted to the Faculty of
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements
for the Degree of Bachelor of Science in

Robotics Engineering,
Computer Science

By:

Albert Lewis
Istan Slamet
Michael J Monda
Myrrh Khan

Project Advisors:

Prof. Nitin Sanket
Prof. Kevin Leahy
Alex Woodyard

Sponsored By:

Draper Laboratory

Date: May 1, 2025

This report represents work of WPI undergraduate students submitted to the faculty as evidence of a degree requirement. WPI routinely publishes these reports on its website without editorial or peer review. For more information about the projects program at WPI, see <http://www.wpi.edu/Academics/Projects>.

Abstract

Unmanned Aerial Vehicles (UAVs), specifically quadcopters, are increasingly used for autonomous tasks, but their limited battery life restricts operational duration. In low-infrastructure environments such as during disaster response or search and rescue operations, short battery life jeopardizes missions.

To address this problem, we present a UAV-UGV docking and charging system in which an Unmanned Ground Vehicle (UGV) serves as a mobile charging station for a UAV.

The docking station uses spring-loaded contacts to supply a 1C charging rate to the UAV, and employs a passive landing mechanism to enable reliable landing despite misalignment by the UAV. The UAV and UGV communicate using ESP32 microcontrollers, allowing the UAV to autonomously initiate and manage charging upon docking. Electromagnets secure the charging connection despite perturbations, and a vision-based guidance system using AprilTags and an OpenMV camera allows the UAV to land autonomously.

We conducted 100 landing trials and achieved a 91% success rate, validating the system’s reliability. The design was further stress-tested on 45° inclines across multiple directions, confirming its robustness. This lightweight, adaptable system demonstrates a promising step toward resilient, infrastructure-free UAV operations and sets the foundation for future field-deployable UAV-UGV cooperative missions.

Acknowledgements

We would like to extend our sincere gratitude to our advisors, Professor Nitin San-
ket and Professor Kevin Leahy, for their invaluable guidance, support, and encouragement
throughout the duration of this project. Their expertise, insightful feedback, and dedication
were instrumental in shaping the direction and success of our work.

We would also like to express our appreciation to Alex Woodyard for his unending
support and for serving as a vital liaison to Draper. His technical guidance, timely assistance,
and facilitation of communication played a critical role in aligning our efforts with project
goals and ensuring a smooth collaboration with our industry partner.

Contents

1	Introduction	1
1.1	Background	3
1.2	Project Objectives	6
2	System Overview	7
3	UGV Design	8
3.1	Landing Zone Design	8
3.2	Electrical Design	10
3.3	UGV Software Functionality	12
3.4	Finalization	14
4	UAV Design	15
4.1	Chassis and Flight Control Overview	15
4.2	Foot Design	16
4.3	Electronics	18
4.3.1	Charging Infrastructure	18
4.3.2	Vision	19
4.4	UAV Software Functionality	19
5	Results and Discussion	21
5.1	Docking and Charging Reliability	21
5.2	Resilience to UGV Perturbations	23
5.3	Design Reflection	24
5.3.1	UAV Vision Tracking	24
5.3.2	Manufacturing	25
5.3.3	Pogo Pins	26
5.3.4	Electromagnets	27
6	Conclusion	29
6.1	Future Work	29
	References	31
	Appendices	34
A	Instructions for Use	34
A.1	Safety Procedures	34
A.2	Starting the UGV	34
A.3	Starting the UAV	35

A.4	Flying the UAV	35
A.5	Landing the UAV	35
A.6	Recharging the UGV Batteries	36
B	UML Diagrams	37
B.1	ESP32 System Class Hierarchy	37
B.2	Board (Abstract Base Class)	38
B.3	UGVBoard (Derived Class)	39
B.4	UAVBoard (Derived Class)	40
B.5	RemoteBoard (Derived Class)	41
B.6	Complete System Overview	42

List of Tables

- | | | |
|---|--|----|
| 1 | Successful UAV charging attempts at a 45° incline in all directions. The “front” of Husky refers to the yellow panel which does not contain the power switch. | 23 |
|---|--|----|

List of Figures

1	Aerial footage of Wajima in the aftermath of the earthquake [1]	1
2	Road destroyed by landslide due to earthquake [2]	2
3	Various commercial docks used for UAV systems today. DJI Dock 1 and 2 (left), Hextronics Universal Dock (center), IDIPLOYER Nexus Dock (right)	3
4	SkyCharge Drone Bolo S1 Docking System. This system utilizes spring-loaded pins on the bottom of the drone to make contact with metal charging pads. .	4
5	UAV prototype developed by Jain and Mueller showcasing the designs for a UAV with a docking system (top) to secure and draw current from a UAV with an external battery for charging (bottom).	5
6	Full UAV-UGV docking and charging system, with our system's UAV docked and in charging mode. ClearPath Robotics Husky A100 UGV also pictured.	7
7	UGV dorsal landing zone (DLZ), as mounted on the Husky A100.	8
8	A previous dorsal landing zone (DLZ) iteration employing neodymium magnets to secure the UAV in place during charging.	9
9	UAV underside. Visible is the lithium-ion battery in the center, the protruding nipple outfitted with female pogo pin contacts on the right, a small hole for the OpenMV camera lens on the left, and washers in the corners of the foot to interface with magnets.	10
10	Power Distribution Module (PDM) within the Husky A100 UGV that contains 24V, 12V, and 5V voltage outputs.	11
11	Flow diagram of the looping process of our code, displaying the four states of our Board class including FLYING, LANDING, DOCKED, and CHARGING.	13
12	Callback setup diagram demonstrating how the Board class sends and receives messages between the UAV and UGV ESP32 microcontrollers.	13
13	UGV's state machine, demonstrating how WiFi communication with the UAV moves the UGV between states Flying, Docked, and Charging.	14
14	Our system's UAV: an 8-inch, lithium-ion powered quadcopter with a custom 3D-printed landing foot to interface with the UGV's dorsal landing zone (DLZ)	15
15	7th-iteration CAD model of the UAV foot. Protoboard, relay, ESP32, and electrical wiring not depicted.	16
16	Combination circuit/control diagram for the UAV-UGV system. Solid lines denote wired connections, with vectors representing control or informing between components. Toggleable connections are represented by the dotted lines.	18
17	Diagram of the UGV's and UAV's communication process.	20
18	UAV's state machine	20
19	Successful landing percentages, teleoperated and autonomous landings	22

20	OpenMV Camera output view, at the resolution which that camera renders images. The green square indicates that the camera script is successfully identifying an AprilTag on the DLZ.	24
21	High-level cost breakdown of our system including costs for 3D modeled components printed in polycarbonate, carbon-fiber quadcopter chassis parts, electronics, and electromagnets.	25
22	A MillMax 0868-0-15-20-82-14-11-0, male pogo pin spring-loaded connector, which was used for this project.	26
23	35 * 30mm cylindrical electromagnet, capable of producing 300N (66lbs/30kg) of magnetic force. Four of these were used in the DLZ, meaning a 120kg UAV could be suspended upside down.	27

1 Introduction

As families were celebrating the New Year on January 1, 2024, a 7.6 magnitude earthquake struck the Noto Peninsula, in the Hokuriku region of Japan, killing 489 people and injuring almost 13,000 [3] [4]. The aftermath left thousands of homes completely destroyed, and even more damaged [1]. A tsunami followed by heavy rainfall caused more devastation. Authorities had to react quickly to save lives.



Figure 1: Aerial footage of Wajima in the aftermath of the earthquake [1]

Response teams worked tirelessly to rescue those who needed help and established emergency infrastructure. However, landslides cut off major roads, and the mountainous terrain made it even more difficult to transport resources and aid into the isolated peninsula. To address these challenges, Japanese authorities deployed advanced technology such as Unmanned Aerial Vehicles (UAVs). The UAVs allowed authorities to conduct damage assessment surveys, deliver supplies, and even set up as mobile telephone networks [5].



Figure 2: Road destroyed by landslide due to earthquake [2]

However, the UAVs faced issues, most notably limited flight duration, due to a short battery life. This made it difficult for UAVs to operate for extended periods of time [4].

In other UAV applications such as agricultural and industrial site monitoring, the short battery life of the drone is also an issue. In cases like these, where there is insufficient infrastructure required for charging, a drone would have to return to a depot in order to recharge. In situations where time, efficiency, and endurance is required, being able to quickly and consistently recharge UAVs can be the difference between a successful and unsuccessful operation.

Unmanned Ground Vehicles (UGVs) operate well in these low-infrastructure settings, carrying a large battery which affords them hours of operation, and by utilizing wheels, tracks or even legs they can traverse well over uneven terrain. By creating a system utilizing a UAV and UGV, the aerial mobility and birds-eye-view of the UAV can be combined with the longevity and robustness of the UGV.

1.1 Background

There are two standard methods of charging: wired charging and wireless power transfer (WPT). Although WPT is increasingly becoming an industry standard, it is more inefficient than wired charging, causing increased charging times of up to 50%, while also achieving a charging efficiency of 65%. Wired charging, although more difficult to implement, is generally more effective for safer and faster charging. [6].



Figure 3: Various commercial docks used for UAV systems today. DJI Dock 1 and 2 (left), Hextronics Universal Dock (center), IDIPLYER Nexus Dock (right)

There are several existing commercial and academic UAV docking and charging systems. One current solution is the DJI Dock 2 (Figure 3), compatible only with other DJI products. It takes the shape of a box, with the top half splitting open and expanding outwards to reveal a platform on which UAVs can land. However, the platform has a wide footprint to make landing easier, and the system is heavy, weighing 34 kilograms (74.96 pounds). Additionally, when closed, it is 570 mm long, 583 mm wide, and 465 tall [7].

A company called Hextronics also offers docking solutions. Their Hextronics Universal dock offers an enclosed space, where the UAV lands on a tray that slides out from the inside of the dock. The product has an AprilTag printed on its tray, allowing UAVs to land on the tray by using the AprilTag as a fiducial marker [8]. However, the docks require the UAVs to land perfectly vertically onto the system, without providing a mechanism to catch the UAV if it is ever misaligned. Additionally, like DJI’s Dock 2, the weight is still large, with the Universal weighing 31.75 kilograms (70 pounds) [9] [10].

Other products include the IDIPLOYER Nexus, which sports a similar design to the Hextronics Universal, the Heisha D135, which consists of a plate and a canopy, and the Airscort Flyt Mini (similar to the Hextronics Universal, but with battery swapping included).

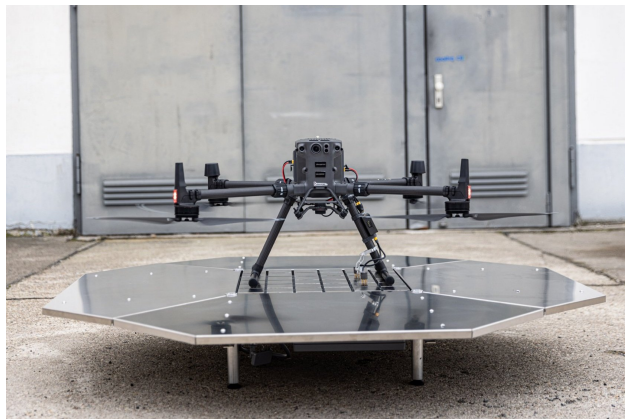


Figure 4: SkyCharge Drone Bolo S1 Docking System. This system utilizes spring-loaded pins on the bottom of the drone to make contact with metal charging pads.

Spring-loaded pins mounted on the bottom of the drone and metal pads acting as charging plates are used in the SkyCharge Drone Bolo S1 Docking System [11]. Although this system proposes a more novel charging concept compared to other charging docks, it does not have a way to secure the drone to the charging pad.

Notably, the solutions listed above do not offer a mechanism that allows the dock to effectively catch the UAV. They instead rely on the control and vision system of the UAV to make a proper landing. In situations where the UAV may miss the landing zone, such as due to wind or faulty control, the UAV may not be able to dock properly and may risk damage or need to reattempt landing. The lack of a passive landing mechanism makes these systems less reliable, as they can not account for potential failures by the UAV. The docks also have large weights and footprints, making them less adaptable for mounting onto a variety of UGVs.



Figure 5: UAV prototype developed by Jain and Mueller showcasing the designs for a UAV with a docking system (top) to secure and draw current from a UAV with an external battery for charging (bottom).

A 2019 academic paper by Jain and Mueller discusses a flying battery experiment (Figure 5), where a custom-made UAV features a bowl design for a flying external battery to land on top of and passively slide into place, allowing the external battery to charge the UAV. The bowl features spring-loaded connectors to connect the battery to the UAV. Users teleoperate the battery, and the battery cannot land autonomously. However, the UAV's passive catching mechanism allows the flying battery to dock without having to be directly aligned with the pins in the bowl, making it easier for the UAV to land without necessitating extremely accurate landing to align the charging pins [12].

1.2 Project Objectives

Inspired by combining the potential of UAVs and UGVs and addressing the downsides of current docking systems, the goal of our project was to create a UAV-UGV docking and charging system that would be reliable, safe, and adaptable. To fulfill the objectives of reliability, safety, and adaptability we came up with the following objectives:

Reliability

- Design a docking system that allows the UAV to successfully dock on to the UGV and successfully charge during 90 percent of all attempted teleoperated landings.
- The docking system will maintain UAV position/orientation and charging connection while the UGV maneuvers over rough terrain.
- The docking system will be able to maintain a secure connection between the UAV and UGV when the landing platform is inclined 45 degrees from the horizontal in any direction.
- Enable the UAV to perform autonomous landing into the dock.

Safety

- A charging circuit on the UGV will charge the UAV battery at a rate of 1C.
- The charging circuit will utilize balance charging to ensure battery safety and longevity.

Adaptability

- The charger will be powered by any 12V or 24V battery.
- The docking system will be designed irrespective to a specific UGV.

2 System Overview



Figure 6: Full UAV-UGV docking and charging system, with our system's UAV docked and in charging mode. ClearPath Robotics Husky A100 UGV also pictured.

The ground vehicle allotted to us was the Clearpath Robotics Husky A100, which was recycled from a previous team's Robotics Engineering Major Qualifying Project (MQP). Although it was approaching a decade and a half of service, and missing most of its sensor suite and motor function, it would adequately serve in its capacity as a portable charging base.

The UAV skeleton was designed and built using open-source software and commercially available drone components. These were selected in accordance with advisor recommendations, and allowed us to easily tailor the system to the project objectives.

The UAV uses a camera fielding open-source software to track the landing zone of the UGV during the autonomous landing process. In the moment in which the UAV foot contacts the dorsal landing zone (DLZ) of the UGV, pogo pins inside of the landing zone make contact with pins on the underside of the UAV. A gyroscope detects a change in the UAV z-axis, communicates to the UGV that a landing has occurred, and request the UGV for a charge. The UGV then confirms the origin of this request, and begins the charging process.

3 UGV Design

3.1 Landing Zone Design

Our dock design was inspired by the bowl design from Jain and Mueller’s research [12]. The DLZ was designed as an inverted square pyramid with rounded vertices that protrude from a rounded square base (interface). The interface is slightly inset, so the UAV foot slides gently down the 35-degree grade and orients itself in one of four target directions.

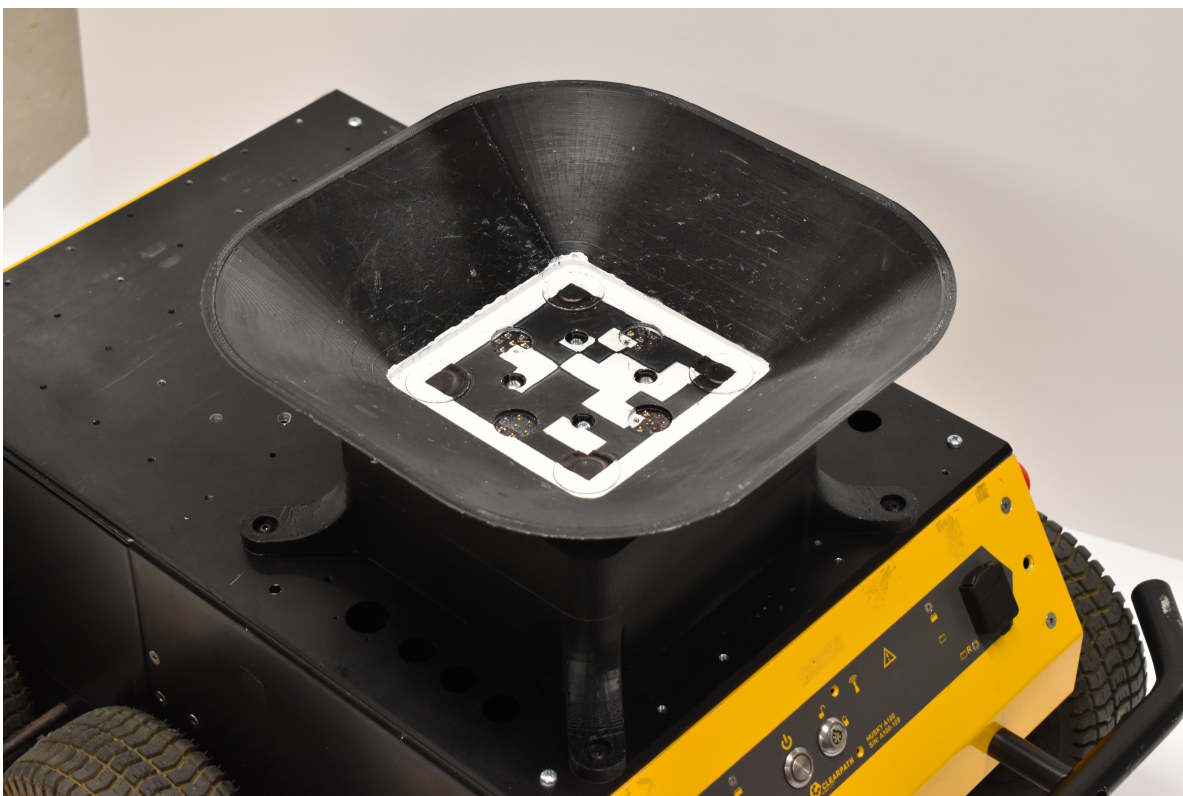


Figure 7: UGV dorsal landing zone (DLZ), as mounted on the Husky A100.

Since the UAV predictably orients upon every landing, we could design a docking mechanism with four-way radial symmetry along the UAV’s vertical axis. First iterations used neodymium magnets, as their extremely strong polarization allowed them to project a sufficient field to draw the robot in upon landing.



Figure 8: A previous dorsal landing zone (DLZ) iteration employing neodymium magnets to secure the UAV in place during charging.

We then progressed to the use of electromagnets. The team replaced the neodymium magnets on the bottom of the drone foot with iron alloy contacts, which would become polarized and magnetize in the presence of the electromagnet's magnetic field.

When designing the electrical interface between the UAV and UGV charging circuits, the team needed a unique 7-wire design to account for the additional output signals of voltmeters on each of the four lithium-ion cells of the UAV battery. The final implementation is an arrangement containing seven male pogo pins in a hexagon pattern with one located at the centroid. The arrangement is slightly inset into the interface, to allow the UAV foot to slide harmlessly over them during landing, but not so inset that contact cannot be maintained with the female pins on the UAV foot. There are four of these sets of pogo pins in the bowl, to account for the four orientations that the UAV can land in.

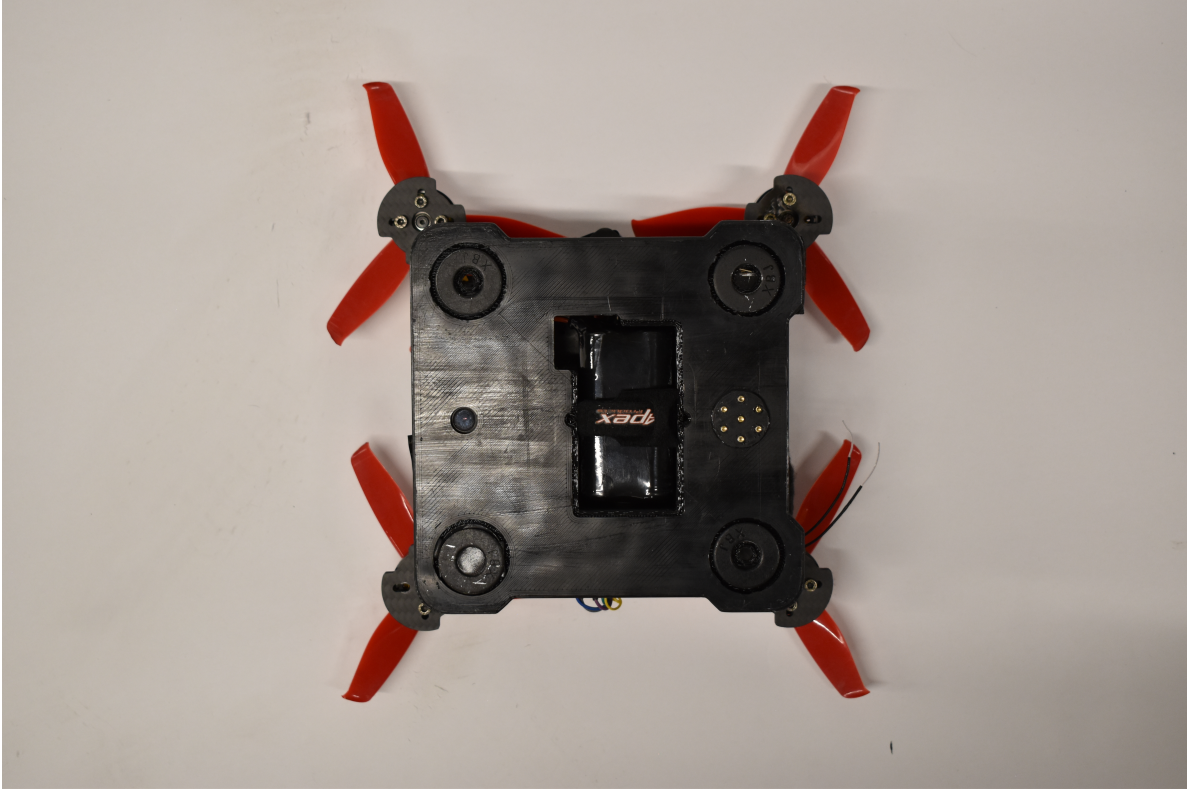


Figure 9: UAV underside. Visible is the lithium-ion battery in the center, the protruding nipple outfitted with female pogo pin contacts on the right, a small hole for the OpenMV camera lens on the left, and washers in the corners of the foot to interface with magnets.

Additionally, the landing zone was raised up from the Husky A100 chassis by a hollow box, which housed electrical wiring and served as an easy way of adapting the bowl to the already-threaded dorsal plate of the UGV.

The current implementation is constructed from PETG 3D print filament, manufactured by a Voron 2.4 3D Printer at 0.6mm precision [13].

3.2 Electrical Design

The backbone of the UGV circuit is the Husky A100's built-in PDM. This module offers 5-, 12-, and 24-volt terminals which we can link directly to the electrical sub-components.

One 5V connection powers an ESP32 microcontroller [14], which is responsible both for starting and stopping the charging process, and toggling power to the electromagnets.

One 12V connection is used to power the electromagnetic locks, and another 12V connection is connected to the Tenergy TB6AV UAV battery charger which we use to monitor the UAV battery during charging [15]. The Tenergy requires a positive and negative lead from the li-on battery, as well as 5 leads that facilitate balance charging for the battery. This means that there are 7 pins in total required to charge the UAV.

Since the UAV's li-ion battery is really a battery pack made up of multiple smaller 18650 li-ion batteries combined, it must be made sure that each individual battery is balance to the proper voltage of 3.7-4.2V. This makes it so that the total voltage of 14.2-16.8V is distributed evenly across the battery pack, and no single 18650 battery is overcharged [16]. Over charging could cause severe safety issues such as having the battery pack light on fire or explode.

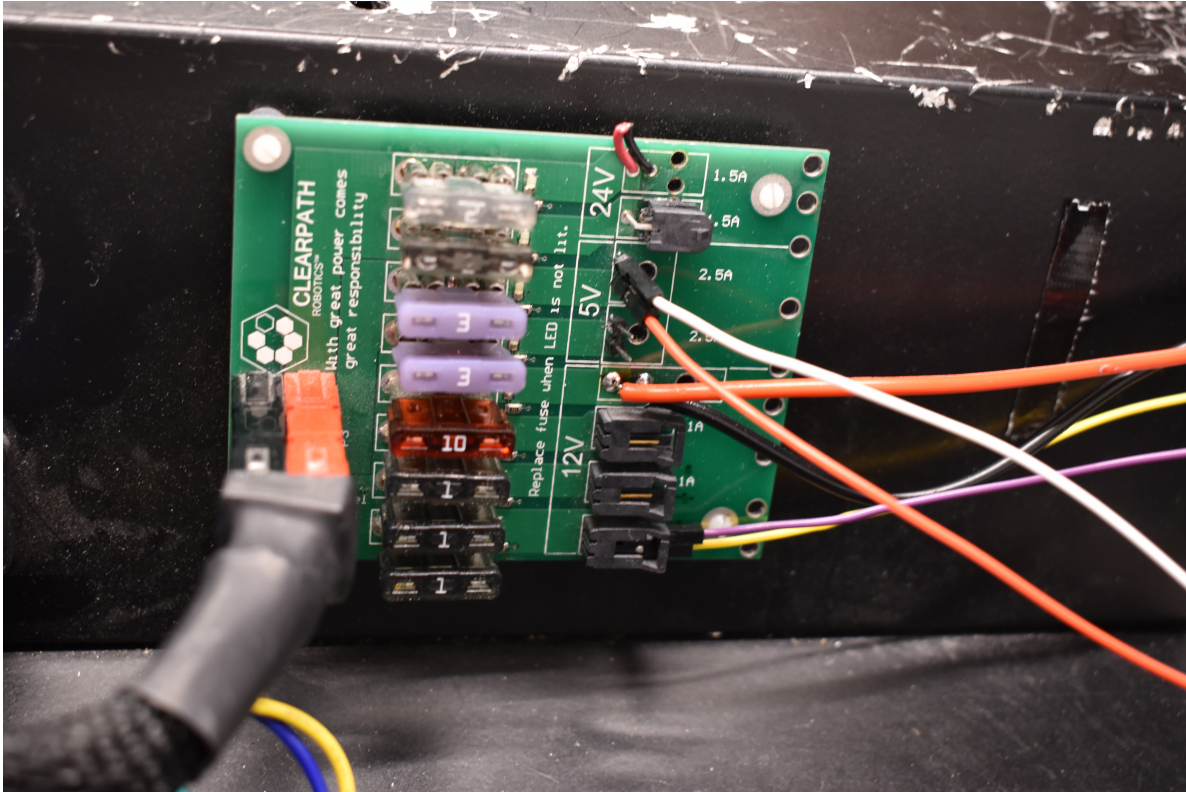


Figure 10: Power Distribution Module (PDM) within the Husky A100 UGV that contains 24V, 12V, and 5V voltage outputs.

The TB6AV is operated by four buttons on the main circuit board. By soldering direct connections from reference voltage pins of each button to the signal pin of the same button, we can change the state from “unpressed” to “pressed.” Running these direct connections through relays allows us to simulate the action of a human pressing a button on the TB6AV. The team created an uninterruptible sequence which closes and releases the relays in the specific sequence required by the TB6AV to analyze the battery, and begin charging.

The electromagnets have no form of signal input to control the strength of their attraction, so they are therefore controlled by an even simpler use of the relays. The positive input of the electromagnets is wired through the normally open configuration of a relay, while the negative terminal is connected directly to the ground. To make the circuit layout more manageable from an assembly and maintenance perspective, the locks are wired in parallel to each other but in series with the relay such that one relay controls all four simultaneously.

For debugging purposes, one colored light emitting diode (LED) is connected in parallel to each relay to confirm that the ESP32 is outputting voltage to the required pins.

3.3 UGV Software Functionality

Both the UAV and UGV’s communication is structured around a C++ Board class, which acts as an abstraction for ESP32-based communication. It manages the wireless communication stack, including initializing WiFi in station mode, configuring ESP-NOW, and handling data transmission and reception.

ESP-NOW is a protocol that operates over WiFi, has a range of around 220 meters, and is designed for point-to-point communication with small payload sizes [17] [18]. The Board class starts setup in its constructor by connecting to a peer via its MAC address. The code also registers static callback functions that run whenever a message is sent or received. In our code, the OnDataSent() and OnDataRecv() callback functions reference an *instance* variable, which serves as a pointer to an instantiated Board object. The callback functions call the instance’s HandleDataSent() and HandleDataRecv() methods, which in turn modify

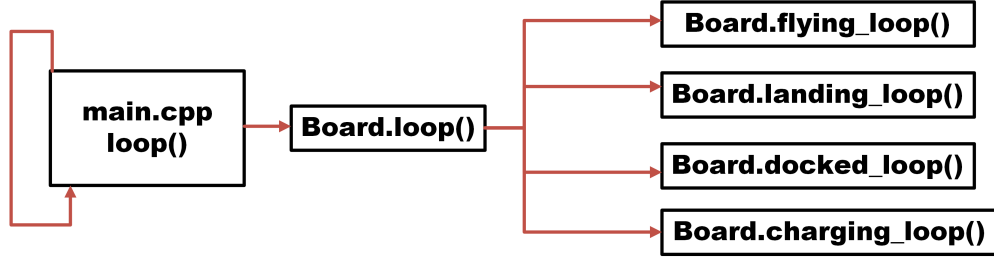


Figure 11: Flow diagram of the looping process of our code, displaying the four states of our Board class including FLYING, LANDING, DOCKED, and CHARGING.

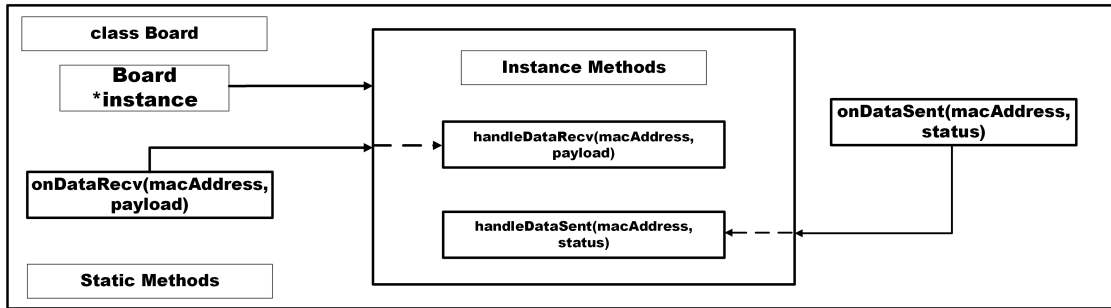


Figure 12: Callback setup diagram demonstrating how the Board class sends and receives messages between the UAV and UGV ESP32 microcontrollers.

mailbox variables within the Board class, saving the message for the methods to handle later.

The Board class defines a finite-state machine (FSM) with four distinct states: FLYING, LANDING, DOCKED, and CHARGING. UAVBoard and UGVBoard classes derive from the parent Board class. A main.cpp file has a loop() function which calls the Board's own loop() method, which then calls flying_loop(), landing_loop(), docked_loop(), or charging_loop().

The flying_loop() method checks for new messages from the UAV via ESP-NOW. Upon seeing a message, it sets the state directly to DOCKED. The docked_loop() method turns on the electromagnets and the charging circuit, sets the state to CHARGING, and sends a message to the UAV. The landing state is skipped as only the UAV needs specialized code to run while landing.

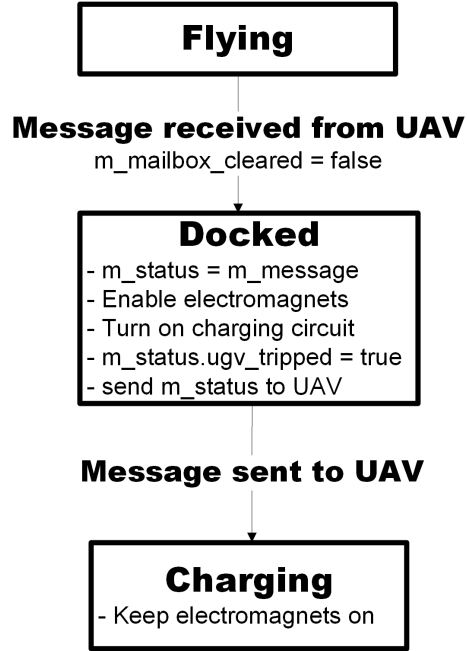


Figure 13: UGV’s state machine, demonstrating how WiFi communication with the UAV moves the UGV between states Flying, Docked, and Charging.

3.4 Finalization

To finalize the design, the team needed to insert a type of fiducial marker known as an AprilTag onto the surface of the interface [8].

We immediately ruled out using any kind of 3D-printed marker, as the placement of the electromagnets limited its size to within the border formed by them. The team therefore opted for using *Rustoleum* [19] oil-based spray paints to finish the project. The paints would protect the PETG structure from surface damage and erosion from the elements, and allow us to expand the size of the AprilTag to the entire area of the interface, since we could simply prime and paint over the electromagnets.

4 UAV Design



Figure 14: Our system’s UAV: an 8-inch, lithium-ion powered quadcopter with a custom 3D-printed landing foot to interface with the UGV’s dorsal landing zone (DLZ)

4.1 Chassis and Flight Control Overview

The UAV is built on an X Style Carbon Fiber frame. It uses four T-Motor F80 Pros and four 3-bladed 5” Propellers. It has a Holybro Pix32 V6 Mini-Set as its flight controller, and a T-Motor Velox V50A 4-in-1 ESC [20]. The firmware that we used to run the UAV was Ardupilot Copter, and we powered it with an Auline 4S Li-Ion Battery. We chose ArduPilot as the firmware because it had multiple features aligned to our goal of autonomous landing [21]. The main feature we were interested in was the Precision Landing mode. This mode allows the flight controller to identify where to land based off an IR-beacon or fiducial marker that is supplied by an outside sensor [22]. For our use case this was a perfect fit, as we planned on using the OpenMV camera to detect the AprilTag in the landing zone, and

used UART communication with the MAVLink Protocol to communicate position offsets from the AprilTag [23]. The other two flight modes that we utilized were Stabilize, which allowed us to pilot the quadcopter easily in an upright manner, and Alt-Hold, which allowed us to hover over the UGV before going into Precision Landing mode.

4.2 Foot Design

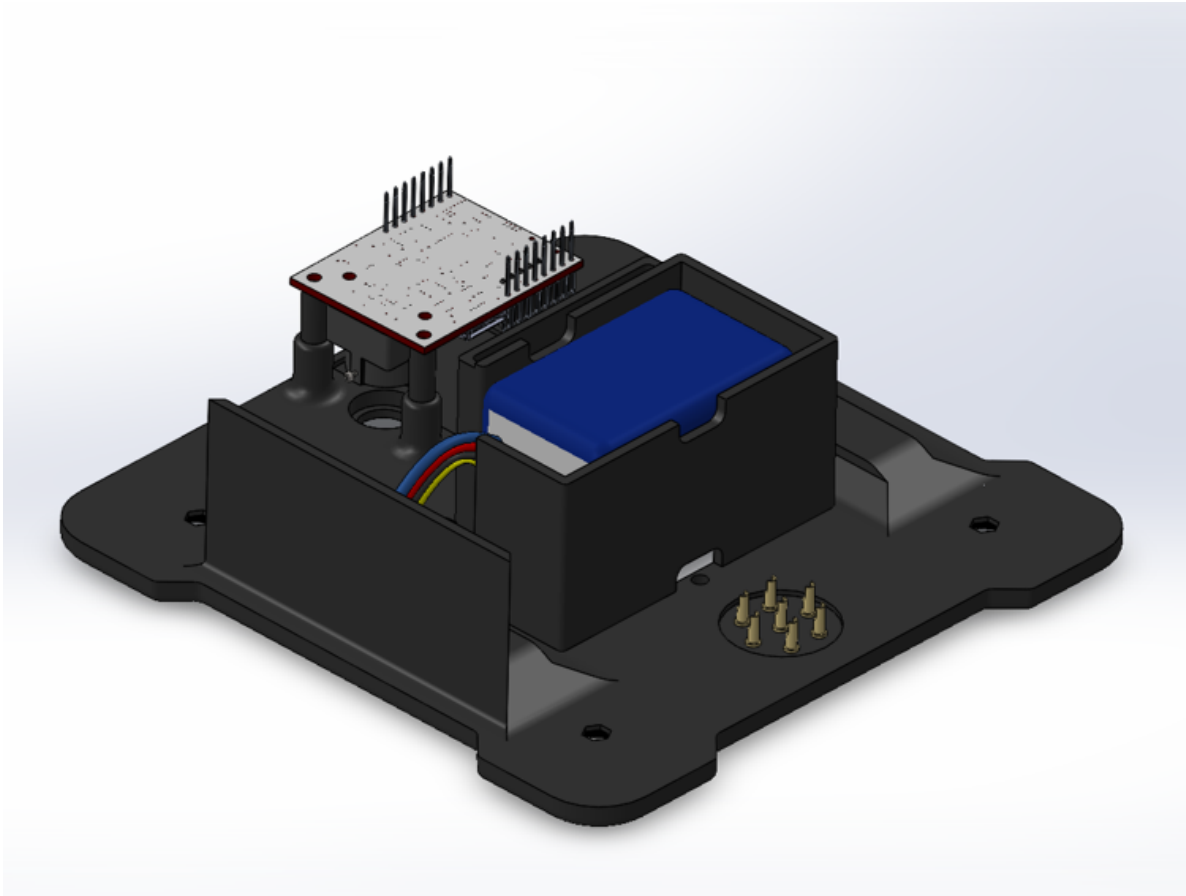


Figure 15: 7th-iteration CAD model of the UAV foot. Protoboard, relay, ESP32, and electrical wiring not depicted.

The UAV foot is a rounded square with the same footprint as the interface at the bottom of the landing zone. It is connected to the bottom of the carbon fiber drone chassis by M3 standoffs, which are anchored into two special “banks” using heat-set thread inserts. The foot contains mounting points for the onboard ESP32 microcontroller, a central battery stowage position, a mount for the camera designed for use with M2-sized components, a

special nipple for the female pogo pins to ensure contact, and impressions for the locations of the relay and transistor protoboard.

The foot, like the UGV dorsal landing zone, is printed using PETG 3D printer filament from the same Voron 2.4 3D printer. The precision of the final model is also 0.6 millimeters. The banks are arranged as a square, plateaued ziggurat shape extruded from the top surface of the foot, which helps distribute the impact force of landing across a wider area. This extends the lifetime of the foot by preventing standoffs from piercing the volume during rough landings. In addition to the charging nipple on the bottom of the foot, there is also a porthole sized for the lens of the OpenMV camera which the team used for tracking the landing zone AprilTag. The camera is inset from the bottom of the foot to prevent the lens from becoming scratched during landing.

The drone foot is not finished with any kind of weather-resistant barrier such as a zinc-based epoxy or spray paint.

4.3 Electronics

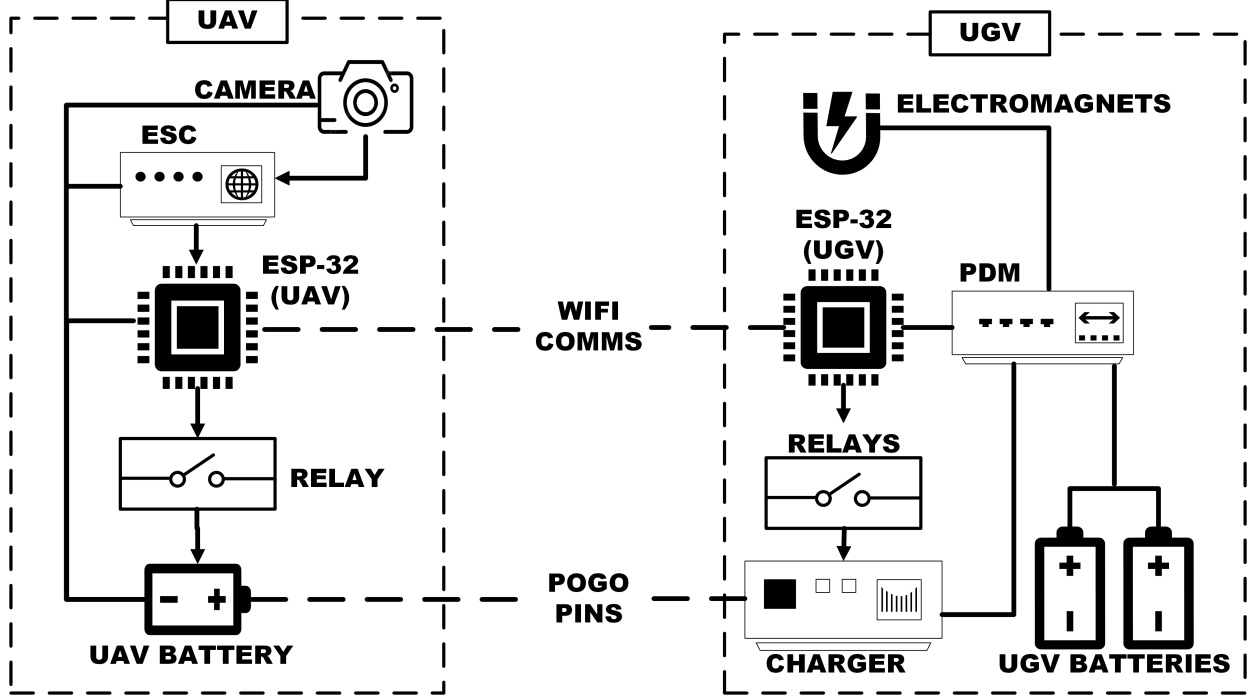


Figure 16: Combination circuit/control diagram for the UAV-UGV system. Solid lines denote wired connections, with vectors representing control or informing between components. Toggleable connections are represented by the dotted lines.

4.3.1 Charging Infrastructure

The UAV contains four main circuits. The first of these is the drone core circuit, powered by a 4-cell lithium-ion battery. This provides direct power to the Electronic Speed Controller (ESC), which regulates rotor function, receives and processes teleop input via antennae, and acts upon visual data provided by the OpenMV camera.

The second is the ESP32 microcontroller circuit, which is solely responsible for communicating over WiFi with the UGV equivalent circuit, and for regulating the third circuit: the relay circuit. Regulation is performed via a protoboard containing an additional driving relay used to direct the required voltage to the charging relay in order to close the battery-charger circuit.

The charging relay circuit directly regulates the connection between the UAV battery,

ESC, and the TB6AV. The connection between the UAV battery and ESC is configured as normally closed, as this is the drone’s default state. When the drone lands, the relay will open to connect the positive terminal of the UAV battery to the positive power cable of the UAV power/signal complex. The negative terminal and the battery signal terminals are always connected to the UAV power/signal complex.

Figure 16 is a combination circuit/control diagram, in which solid black lines denote electrical connections between components and black arrows denote direct control or informing over other components. The dashed blue arrow represents the communication network between the microcontrollers of the UAV and UGV. Note how there are two black arrows from “RELAYS” to “CHARGE CIRCUIT;” this describes how we use one relay each for the ‘start’ and ‘stop’ buttons on the TB6AV.

4.3.2 Vision

The fourth and final circuit is the OpenMV camera circuit. Powered by the flight controller, this circuit uses open-source vision recognition software to identify the distance, relative position, and relative orientation of the AprilTag marker. Based solely upon the position of the AprilTag relative to a specific pixel of the camera, it will inform the ESC to control the motors such that the UAV positions itself to be directly over the target while descending.

4.4 UAV Software Functionality

The UAVBoard class extends Board and incorporates an Adafruit BNO055 inertial measurement unit (IMU) for detecting landing events. Currently, the UAV is always in the LANDING state, but a `flying_loop()` method exists, which can be completed to determine when the ESC converts to landing mode. During `landing_loop()`, the UAV measures acceleration changes along the Z-axis to confirm landing. Once confirmed, it signals the UGV after setting a relay pin to HIGH, disconnecting the motors, and transitions to DOCKED.

The final transition to CHARGING occurs upon receiving confirmation from the UGV that it has engaged its relay.

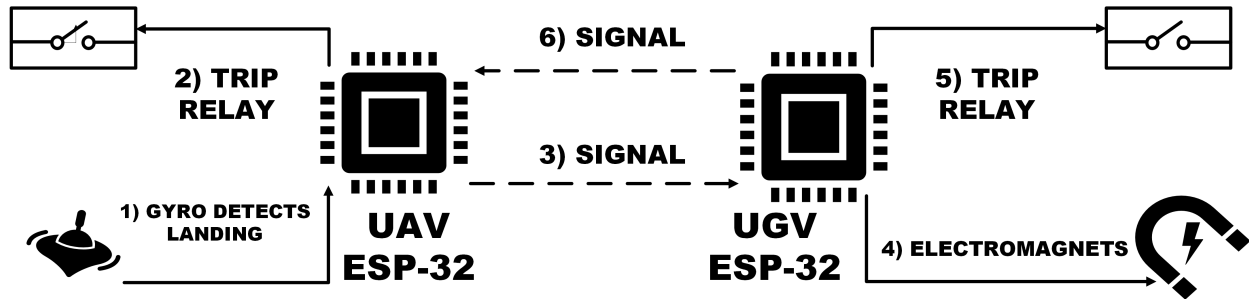


Figure 17: Diagram of the UGV's and UAV's communication process.

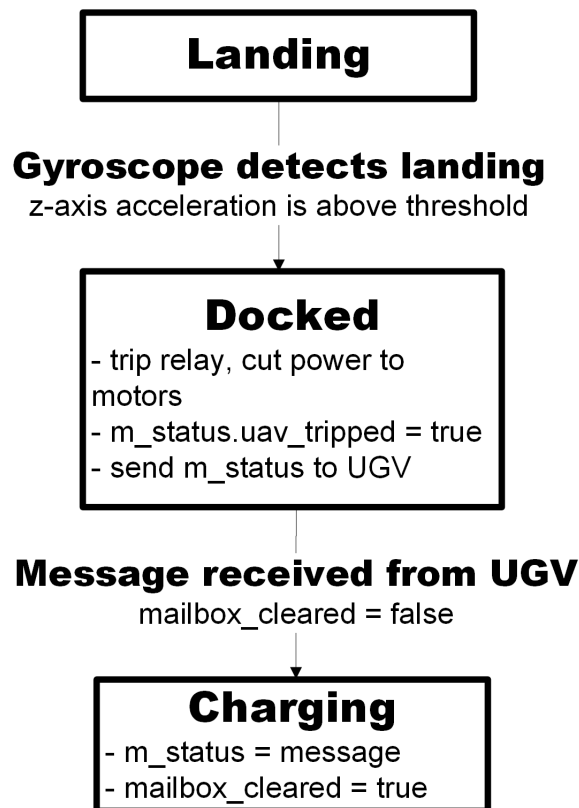


Figure 18: UAV's state machine

5 Results and Discussion

To verify the functionality of our docking and charging system, we designed a series of experiments that evaluated each of the project’s core objectives under controlled and realistic conditions. Our testing was structured to assess the reliability of each subsystem, focusing on meeting the goals outlined in the introduction.

5.1 Docking and Charging Reliability

To evaluate the consistency of the docking and charging process, we conducted 100 landing trials. These were split between autonomous and teleoperated landing modes to assess both operator-assisted and autonomous performance. For each trial, we recorded whether the UAV successfully docked with the UGV, meaning a landing within the landing area of the DLZ, and also whether the UAV and UGV formed a stable electrical connection and initiated charging.

Figure 19 demonstrates our results for these experiments. With a total of 100 landings, split 50-50 between teleoperated and autonomous, we achieved a total success rate of 91%, including a total of 47 and 43 successful teleoperated and autonomous landings, respectively. These results demonstrate that our system is overall effective and reliable. Additionally there was no case in which the UAV, when securely docked, failed to initiate charging.

The failure cases can be attributed to several factors. For teleoperated flight, this mainly came down to human error and imprecision in flying the UAV, causing it to miss the DLZ. There were also some edge cases in which the UAV became stuck diagonally against the base of the charging circuit.

Autonomous flight has a noticeably lower success rate than teleoperated flight. This is a result of the reliability of the OpenMV camera’s vision tracking. It was often difficult to completely line up the camera to properly recognize the AprilTag resulting in failed landings,

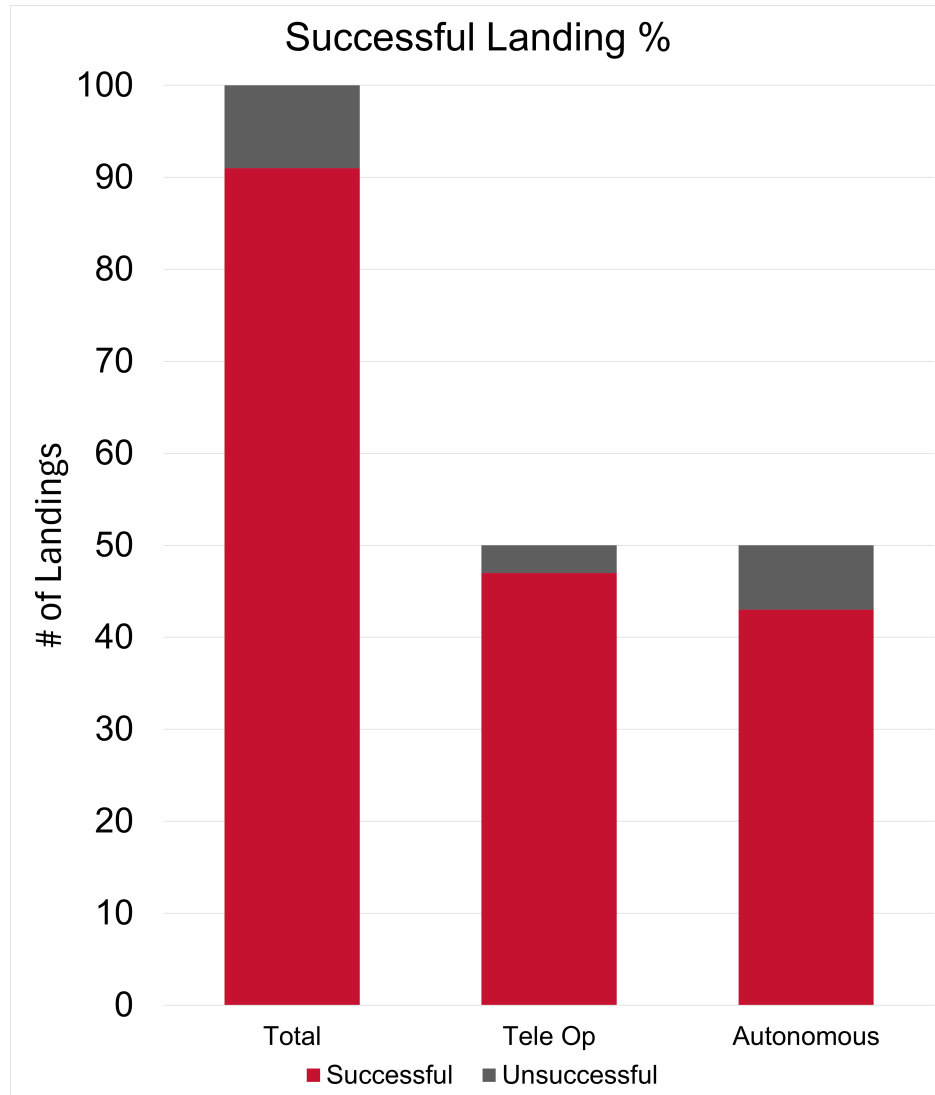


Figure 19: Successful landing percentages, teleoperated and autonomous landings

where the UAV would miss the bowl. This aspect of the system could be improved, but optimizing the UAV's flight capabilities was not a priority of this project.

In addition, the repeated landings took a toll on the integrity of the UAV foot itself. Because the foot is attached to the carbon-fiber frame using solely heat-set inserts and standoffs, over time, the repeated landing impact force caused these connectors grew weaker and weaker causing the UAV to begin falling apart. An improved method of attaching the foot to the frame that could more evenly distribute the impact force upon landing would mitigate these issues.

5.2 Resilience to UGV Perturbations

Tilt Direction	Successful
Forward	5/5
Right	5/5
Backward	5/5
Left	5/5
Forward Right	5/5
Backward Right	5/5
Backward Left	5/5
Forward Left	5/5

Table 1: Successful UAV charging attempts at a 45° incline in all directions. The “front” of Husky refers to the yellow panel which does not contain the power switch.

As our system requirement for charging was to be able to successfully charge the UAV while the Husky was at a tilt of less than or equal to 45° , our testing for resiliency towards UGV perturbations was quite straightforward. For all 8 directions (including diagonals) we performed five tests each of charging the UAV at 45° for a total of 40 tests. Table 1 showcases that in our tests, the UAV did not fail to charge at any of the angles in which the Husky was tilted, indicating that the UAV can reliably charge and remain docked while the Husky is in motion.

The results for charging and secure docking fully satisfied our system requirements. We could certainly expand upon these tests to provide more quantifiable data for charging reliability during Husky movement, such as by including more dynamic experiments involving vibrations or movement over real-world terrain. However, although not formally tested, we found that the electromagnets in our system were strong enough to keep the UAV charging at angles greater than 45° as well, which are tilts that the Husky itself is not even capable of operating on. This indicated to us that the UAV can remain secure and satisfactorily charge on the UGV.

5.3 Design Reflection

Although our system successfully met each of our desired system requirements, there were several aspects of our design that could be improved. These changes, such as improvements to the UAV vision system, manufacturing, and physical design. would make the system more reliable in real-world scenarios and also improve the troubleshooting process.

5.3.1 UAV Vision Tracking



Figure 20: OpenMV Camera output view, at the resolution which that camera renders images. The green square indicates that the camera script is successfully identifying an AprilTag on the DLZ.

The OpenMV Camera's vision tracking worked fairly inconsistently during flight and landings. Although we were still able to land the UAV during the majority of attempted landings, the quality of vision-sensing still left much to be desired. Our initial tests with the camera included pointing it at various test distances from the AprilTag and determining if it sensed it, which proved to work well. However, during flight, especially at distances greater than 10 feet, the camera struggled a lot with sensing the AprilTag, as aligning the UAV in the correct position proved to be a challenge. We altered the camera's frame of view, frame size, and image quality to varying degrees, which significantly improved the sensing during initial testing, but still had little to no effect during flight.

When the UAV enters landing mode after sensing the AprilTag, it uses proportional control to move towards it and land. However, during this period, it does not consider

external perturbations like wind which can direct its landing off-course. The landing sequence can be improved by accounting for this. Implementing controllers or other functionality to improve the UAV’s stabilization during flight and landing would allow it to better sense the AprilTag and land more precisely.

5.3.2 Manufacturing

Both the drone foot and DLZ were manufactured entirely using 3D printing with PETG. The use of 3D printing as our primary manufacturing method allowed us to easily redesign and reiterate our system when required. The PETG surface also did not have too much friction, allowing the UAV to easily slide into the correct charging orientation without getting stuck. The filament proved to be durable enough to withstand all landing sequences, and neither the foot nor the DLZ sustained any critical damage to their function, besides some dents and scratch marks. But we anticipate that continued use over a longer period of time would result in damage to the DLZ that would begin to impact system performance. A high-level cost breakdown of the full system (both UAV and DLZ parts) can be found in Figure 21.

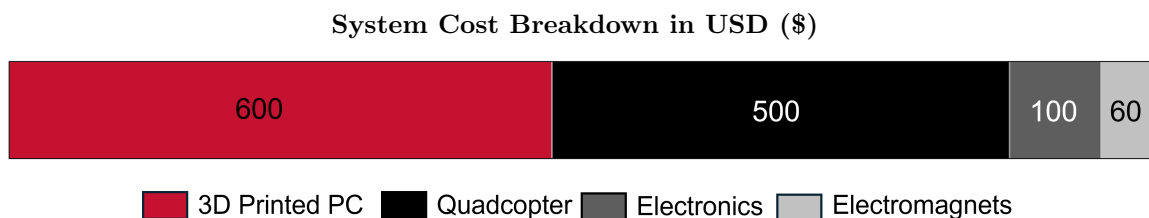


Figure 21: High-level cost breakdown of our system including costs for 3D modeled components printed in polycarbonate, carbon-fiber quadcopter chassis parts, electronics, and electromagnets.

For our purposes, PETG proved to work well for our design process and our overall system. However, marine- or automotive-grade fiberglass production would greatly improve it and add to its capabilities. Fiberglass is extremely resistant to erosion, which helps in weatherproofing the devices, and is also very flexible, which lowers the risk of denting the DLZ and potentially impacting the quality of UAV landings [24]. Its greater strength would

also allow for the casting of more complex shapes ideal for water drainage or improved air flow. It is also easier to produce in mass and has a relatively low per unit cost; once you are beyond the costs of mold fabrication and factory tooling.

5.3.3 Pogo Pins



Figure 22: A MillMax 0868-0-15-20-82-14-11-0, male pogo pin spring-loaded connector, which was used for this project.

Pogo pins facilitated the physical contact between the UAV and DLZ's charging systems. These pogo pins are 0.362" (9.19mm) tall spring-loaded contacts that are capable of drawing up to 9.6A of current [25]. They consistently enabled current flow for charging at a 1C rate, satisfying our desired requirements. The pogo pins are inset into the bottom of the DLZ, with the spring loaded contacts facing upward and set slightly below the DLZ surface so that the UAV does not get stuck on them upon landing. We inset 28 male pogo pins, four sets of seven, on the base of the bowl so that the UAV could land in all four orientations.

Due to the sheer number of pins needed for charging, their implementation on the DLZ was challenging. Individually attaching the 28 pins to the bottom of the bowl and soldering wires proved to be tedious and time consuming. The insert holes for the pogo pins were designed for press-fitting, but due to their small size and the limitations of our 3D

printer, the press-fit did not work properly and we had to manually heat the pins with the soldering iron for them to melt the PETG around the pins. Replacement of broken pins also proved to be a challenge because of this.

Although the pogo pins were able to properly facilitate charging, alternative contact methods could provide simpler installation and troubleshooting. For example, instead of using pins, contact rails that circle the DLZ base could be used to remove the need for a large number of small parts which would also make troubleshooting the system's connectivity issues more efficient. Overall, the use of the spring-loaded contacts provides a unique solution for connectivity, allowing the UAV to effectively land and ensure a strong connection.

5.3.4 Electromagnets



Figure 23: 35 * 30mm cylindrical electromagnet, capable of producing 300N (66lbs/30kg) of magnetic force. Four of these were used in the DLZ, meaning a 120kg UAV could be suspended upside down.

Electromagnets, as opposed to neodymium magnets, proved to be an effective method of securely locking the UAV in place during charging while also allowing the UAV to take off from the DLZ without resistance. The electromagnets used were capable of 300N (66lbs/30kg) of suction force which was a substantial amount of force relative to the UAV's weight. In order to fully lock the UAV in place, the washers on the base of the drone foot

required good contact with the surface of the electromagnets. For this to occur, we had to be very precise with the placement of the electromagnets relative to the surface of the DLZ's base. Insetting the electromagnets higher caused them to stick out more, allowing for better contact with the washers but interfering with the UAV sliding into place, but inseting them lower risked the opposite effect.

Our current electromagnets are powered to activate. Using the alternative of permanent electro-holding magnets, which are powered to deactivate, could be a potential improvement. With permanent electromagnets, the UAV would dock more easily as a result of the magnetic force in the bowl being constantly projected. This would also reduce power consumption of the UGV battery as electromagnets would only draw power when the UAV is taking off, as opposed to the current setup [26]. With the addition of a magnetometer, we could also direct the UAV towards the DLZ's magnetic field as an additional homing mechanism.

6 Conclusion

Our system successfully allows a UAV to land, charge safely, and remain docked to a UGV during charging for the purposes of UAV-UGV collaboration. A passive landing mechanism is used for docking in the shape of a rounded bowl for the UAV to slot into upon landing. Charging is facilitated by pogo pin connections between the UAV and the DLZ's internal charging circuit, which is supported by a relay circuit onboard the UAV controlled by ESP32 microcontrollers using WiFi that trigger the circuit which ensures safe charging at a 1C rate. Electromagnets lock the UAV to the UGV allowing for continuous charging even while the UGV is inclined or experiencing perturbations. Our system achieved a 91% successful landing and charging rate for teleoperated and autonomous landings.

6.1 Future Work

Although we have successfully fulfilled all our desired system requirements, our system can be improved by adding several functionalities. In particular, implementing weather-proofing, alternative manufacturing methods, additional homing mechanisms for UAV landing, and support for simultaneous UAV-UGV operations can result in a more complete and finished product.

The charging dock has not been formally tested against various weather scenarios and real-life conditions. However, due to its bowl shape, sediment and rain can build up inside, potentially causing damage to internal electronics. To avoid this, we could slightly modify the bowl's design by adding drainage holes in the bottom lip of the bowl's base to redirect rain and prevent buildup. We could also manufacture the bowl with different material such as fiberglass to create a more weather-resistant system in addition to the addition of these drainage holes.

In low-visibility cases such as fog, the UAV would find it difficult to spot and land on the AprilTag. Additional mechanisms for the UAV to recognize the DLZ were considered

but not implemented. An IR beacon could provide an alternative to the vision-based sensing so that the UAV could land in tougher, low-visibility conditions. The use of a magnetometer could pose another solution. However, this would require the use of alternative, more expensive electromagnets that are powered to deactivate. The current electromagnets that are powered to activate would create no magnetic field upon landing.

The Clearpath Robotics Husky A100 UGV we were provided with had no supported functionality. In other words, we weren't able to drive it. But for our purposes, and due to our focus on the charging and docking aspect of the system, we considered the UGV's function non-essential to the project. However, we believe that for future work the system should be implemented on a working UGV to better assess its functionality in real-world scenarios.

References

- [1] T. Y. S. E. Board, “64 Confirmed Dead after Jan. 1 Noto Peninsula Earthquake; Fire Razes Famous Wajima Market; Tsunami Hits Suzu Buildings (UPDATE 3).” <https://japannews.yomiuri.co.jp/society/noto-peninsula-earthquake/20240102-159388/>, Jan. 2024.
- [2] D. Petley, “Landslides from the Noto Peninsula Earthquake in Japan.” <https://eos.org/thelandslideblog/noto-peninsula-earthquake-1>, Jan. 2024.
- [3] “One Year On: The State of Recovery on the Noto Peninsula.” <https://www.nippon.com/en/japan-data/h02234/>, Jan. 2025.
- [4] M. Ishiwatari, “Leveraging Drones for Effective Disaster Management: A Comprehensive Analysis of the 2024 Noto Peninsula Earthquake Case in Japan,” *Progress in Disaster Science*, vol. 23, p. 100348, Oct. 2024.
- [5] A. K. T. Martin, “A year after a deadly earthquake, Noto Peninsula still faces long road to recovery.” <https://www.japantimes.co.jp/news/2025/01/01/japan/society/noto-anniversary/>, Jan. 2025.
- [6] T. Ojha, T. P. Raptis, A. Passarella, and M. Conti, “Wireless power transfer with unmanned aerial vehicles: State of the art and open challenges,” *Pervasive and Mobile Computing*, vol. 93, p. 101820, June 2023.
- [7] “DJI Dock 2 - Specs.” <https://enterprise.dji.com/dock-2/specs>.
- [8] “AprilTag.” <https://april.eecs.umich.edu/software/apriltag>.
- [9] “Hextronics Atlas - Specs.” <https://hextronics.com/atlas/specs>.
- [10] “Hextronics Universal - Specs.” <https://hextronics.com/universal/specs>.
- [11] “Drone Charging Pad - BOLO S1.” <https://www.skycharge.de/drone-charging-pad>.

- [12] K. P. Jain and M. W. Mueller, “Flying batteries: In-flight battery switching to increase multirotor flight time,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3510–3516, May 2020.
- [13] “VORON2.4.” <https://vorondesign.com/voron2.4>.
- [14] “ESP32 Wi-Fi & Bluetooth SoC | Espressif Systems.” <https://www.espressif.com/en/products/socs/esp32>.
- [15] “Tenergy TB6B Multifunctional Balance Charger for NiMH/NiCd/LiPo/Li-Fe Battery Packs + Power Supply.” <https://power.tenergy.com/tenergy-tb6b-multifunctional-balance-charger-for-nimh-nicd-li-po-li-fe-battery-packs-power-supply/>.
- [16] “BU-409: Charging Lithium-ion.” <https://batteryuniversity.com/article/bu-409-charging-lithium-ion>, Sept. 2010.
- [17] “Device to Device Communication with ESP-NOW | Arduino Documentation.” <https://docs.arduino.cc/tutorials/nano-esp32/esp-now/>.
- [18] “ESP-NOW Wireless Communication Protocol | Espressif Systems.” <https://www.espressif.com/en/solutions/low-power-solutions/esp-now>.
- [19] “Rust-Oleum | Spray Paint, Wood Stains, Floor Coatings and More.” <https://www.rustoleum.com/>.
- [20] “Holybro Pix32 v6 | PX4 Guide (main).” https://docs.px4.io/main/en/flight_controller/holybro-pix32.
- [21] “ArduPilot Documentation — ArduPilot documentation.” <https://ardupilot.org/ardupilot/>.
- [22] “Precision Landing and Loiter — Copter documentation.” <https://ardupilot.org/copter/docs/precision-landing-with-irlock.html>.
- [23] “Small - Affordable - Expandable.” <https://openmv.io/>.
- [24] “All About Fiberglass | Xometry.” <https://www.xometry.com/resources/materials/what-is-fiberglass/>.

- [25] “0868-0-15-20-82-14-11-0 - Spring-Loaded Pin | Mill-Max Mfg. Corp..”
<https://www.mill-max.com/products/discrete-spring-loaded-pins/spring-loaded-pin-with-solder-cup-termination/0868/0868-0-15-20-82-14-11-0>.
- [26] “Energize To Release Electromagnet | Eclipse Magnetics.”
<https://www.eclipsemagnetics.com/en-us/products/magnetic-tools-and-standard-magnets/energise-to-release-electromagnet/>.

Appendices

A Instructions for Use

A.1 Safety Procedures

Before starting the quadcopter, be sure that the switch of the transmitter is set to disarmed. Starting the UAV while the motors are armed can result in serious injury to the technician and damage of the motors and rotors. It is highly recommended that technicians within 40 feet of the quadcopter wear safety goggles and long-panted legs with closed toed shoes.

Additionally, if the battery is connected to the UAV and it is charging, we recommend the user switch into the disarmed state. While the UAV charging circuit relay will disconnect the drone battery from the motors, it is still important to ensure that the UAV will not begin to spin the rotors while in this state.

All Lithium-Ion and Lithium-Polonium batteries used by the quadcopter must maintain a charge of between 9.8 and 16.8 volts. In the event that a battery is severely depleted, dispose of it in the nearest explosive materials cabinet and notify your supervisor or the relevant maintenance teams. Do not attempt to charge a battery below 9.6 Volts.

A.2 Starting the UGV

To start the Husky A100, simply press the circular power button located on the yellow panel adjacent the landing zone. The UGV will immediately enter its default idle state, in which it waits for an external WiFi signal from the UAV before charging. The UGV can be turned off at any time using the same button, but it will always default to the idle state and not to the state it was in prior to shutdown.

A.3 Starting the UAV

To use our UAV’s teleoperation capabilities bind the UAV’s receiver to a transmitter of your choice – we utilized the FrSky Taranis X9 Lite. Once this is set up, turn on the remote controller, and select the quadcopter’s channel from the available list (on our controller, the transceiver ID is called “charge mqp”). **While the controller is set to “disarmed”,** connect the yellow male battery plug to the female equivalent of the ESC. You will hear a jingle emanate from the controller, which signifies it is powered properly. **The ESC twitches the motors during startup to confirm they are connected.** This is the main reason the motors must be disarmed during startup.

A.4 Flying the UAV

Assuming the transmitter was turned on before the UAV was, make sure stabilized flight mode is enabled by throwing the switch above the left stick to the top position. Set the throttle to 0, and arm the motors. They will begin to spin at a low RPM. From here, you are able to fly the UAV as normal.

If you wish for the UAV to hold altitude, manipulate the selector switch to the middle position. This largely disables user throttle control, but allows you to maneuver the UAV in a plane parallel to the ground.

A.5 Landing the UAV

When landing the UAV, manipulate the selector switch to the down position to enter precision landing mode. The user can manipulate the UAV in this mode, but if left undisturbed it will naturally track the AprilTag in the UGV DLZ.

A.6 Recharging the UGV Batteries

When charging the UGV batteries, simply connect them to a power supply and charge until 14.8V maximum for floating cycle use. Be sure to maintain an amperage lesser than 6.0 Amps to prevent internal damage.

B UML Diagrams

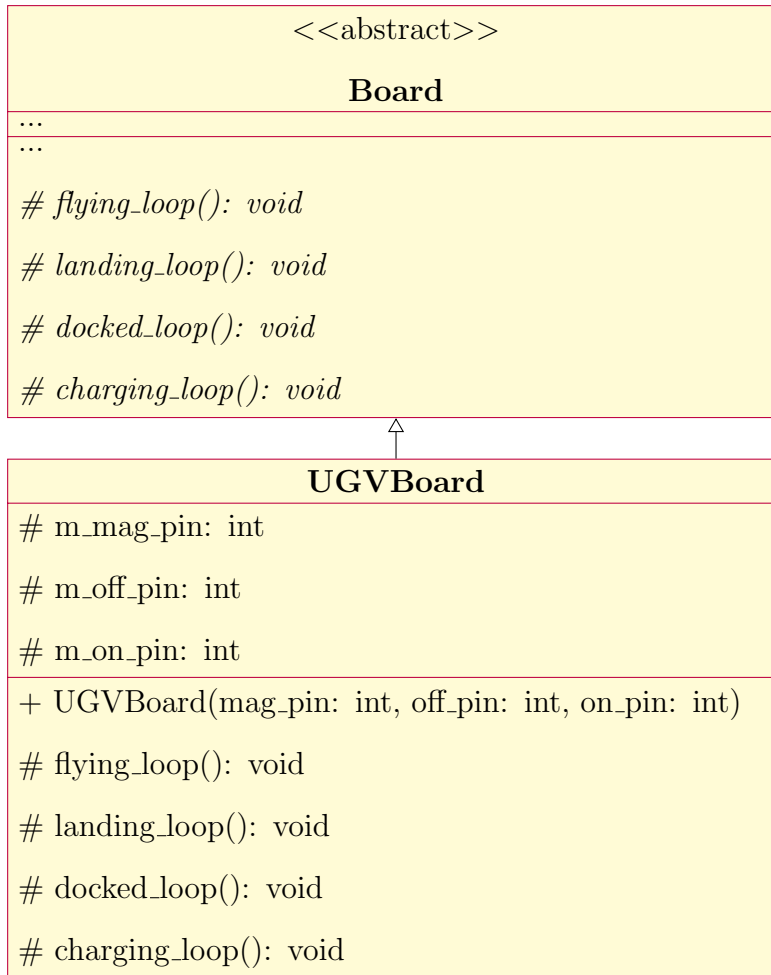
B.1 ESP32 System Class Hierarchy

This document shows the class diagram for an ESP32-based system with UAV, UGV, and Remote control boards. The diagram is split across multiple pages for readability.

B.2 Board (Abstract Base Class)

<<abstract>>	
Board	
+ <u>m_instance</u> :	Board*
# m_status:	Status {bool uav_landed, bool uav_tripped, bool ugv_tripped}
# m_mailbox_cleared:	bool
# m_boardState:	BOARD_STATE {FLYING, LANDING, DOCKED, CHARGING}
- m_peerAddresses:	uint8_t[][6]
- m_peerInfo:	esp_now_peer_info_t[]
- m_peerCount:	int
- m_print_count:	int
- M_PRINT_COUNT_RESET:	const int
+ Board()	
+ loop():	int
+ connect_peer(mac: uint8_t[6]):	bool
+ connect_multiple_peers(mac: uint8_t[][6], count: int):	int
# send_msg(peer_idx: int = -1):	bool
# <i>flying_loop()</i> :	<i>void</i>
# <i>landing_loop()</i> :	<i>void</i>
# <i>docked_loop()</i> :	<i>void</i>
# <i>charging_loop()</i> :	<i>void</i>
# set_state(newState: BOARD_STATE):	void
# print_state():	void
- <u>OnDataSent</u> (const uint8_t*, esp_now_send_status_t):	void
- HandleDataSent(const uint8_t*, esp_now_send_status_t):	void
- <u>OnDataRecv</u> (const uint8_t*, const uint8_t*, int):	void
- HandleDataRecv(const uint8_t*, const uint8_t*, int):	void
- init_esp_now():	bool

B.3 UGVBoard (Derived Class)



Constants:

SHORT = 200

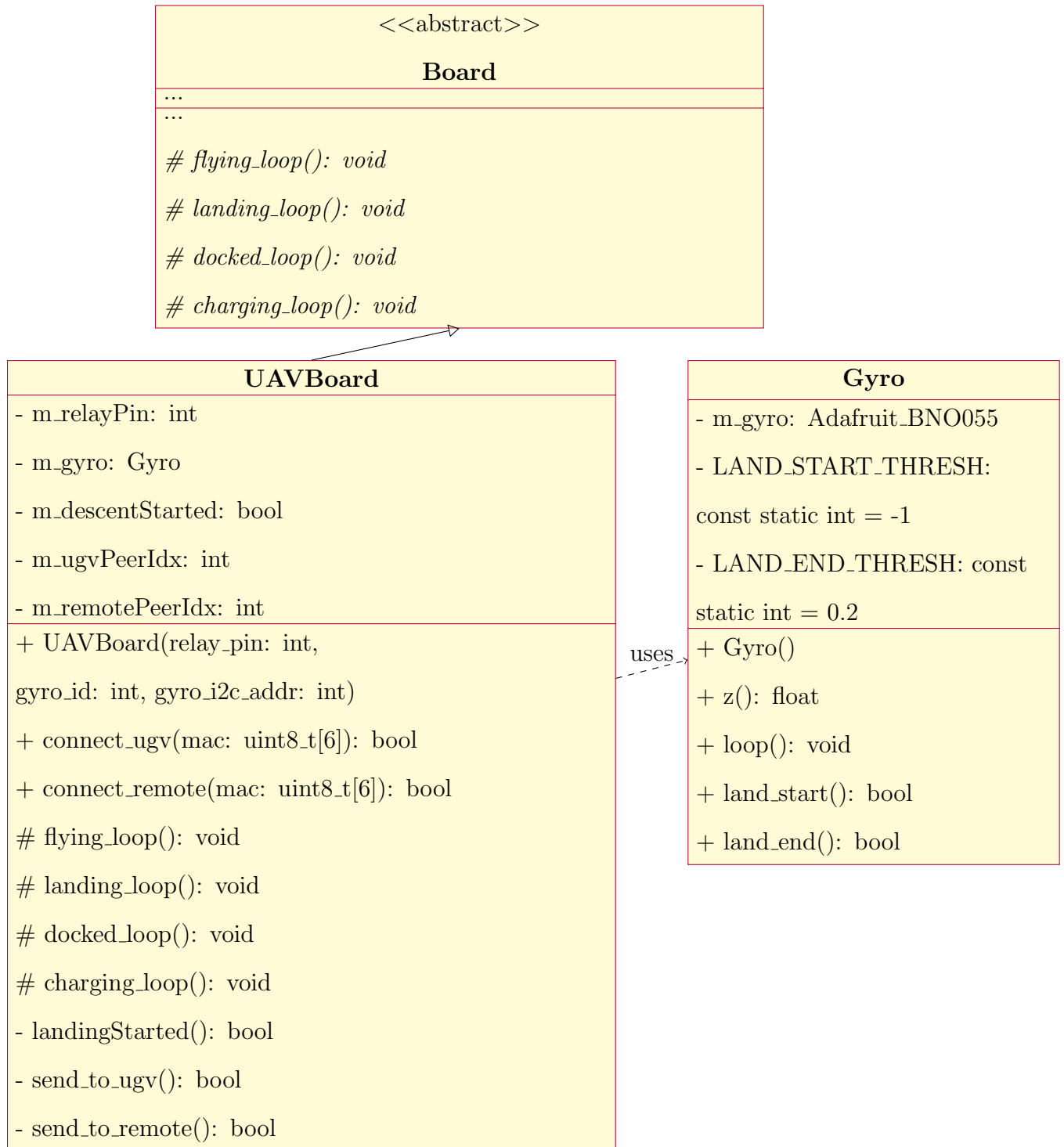
LONG = 2750

GAP = 300

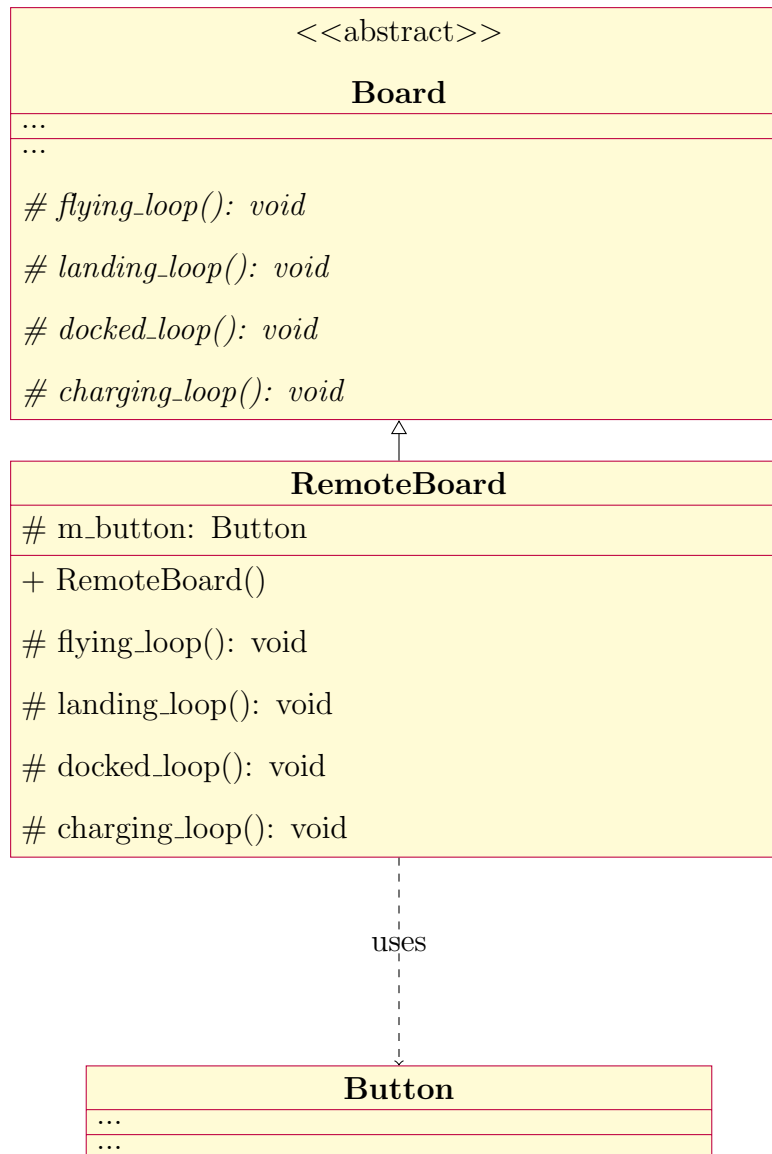
SUPER = 3000

SUPERLONG
= 1700

B.4 UAVBoard (Derived Class)



B.5 RemoteBoard (Derived Class)



B.6 Complete System Overview

