Flinders Robotics Team Technical Design Paper for RobotX 2026

K. Dashora, J. Tassell, L. Will, E. Irankunda, O. Pradhan, C. Mitson, M. Cha, A. Lammas, J. Wheare, T. Chaffre, K. Sammut

***Abstract*— This paper presents the design, integration, and testing strategy of the Flinders Robotics Team's autonomous maritime System of Systems (SoS) for the 2026 International RobotX Maritime Challenge. The SoS integrates an unmanned surface vehicle (USV), an unmanned aerial vehicle (UAV), and an unmanned underwater vehicle (UUV), each providing capabilities suited to its own operating environment. The design prioritises reliable autonomous operation, perception, navigation, communication, safety, and inter-vehicle integration. To manage cost and development time, the team adapts existing platforms rather than building new ones. The USV is a Blue Robotics BlueBoat fitted with a custom superstructure that carries a LiDAR, a camera, and an emergency stop system. The UAV is a SwellPro Splashdrone 4 rebuilt around open-source flight control hardware, and the UUV is a BlueROV2 Heavy with upgraded onboard computing, a Doppler velocity logger, and an imaging sonar. The vehicles share a common open-source software foundation based on ArduPilot, MAVROS, and ROS 2, which simplifies integration across the SoS. The competition strategy focuses first on reliable task completion by the USV and then adds collaborative behaviours with the UAV and UUV. Testing follows a staged pipeline that verifies components and subsystems before full system integration and in-field trials.**

I. Overview

The RobotX Maritime Challenge asks teams to complete a series of tasks in a marine environment using autonomous vehicles that operate both independently and together [1]. The Flinders Robotics Team (FRT) from Flinders University addresses this challenge with a System of Systems made up of three vehicles. A BlueBoat USV operates on the surface, a modified SwellPro Splashdrone 4 UAV provides an aerial view of the course, and a BlueROV2 UUV operates below the surface. Each vehicle covers a domain that the others cannot, and together they allow the team to attempt tasks that span all three environments.

The primary consideration is that of reliability, as performance and capability is worthless without it. The second is the availability and cost of materials and components. For this reason, the team adapts existing platforms instead of designing new ones, and it standardises on open-source software across the vehicles so that work on one vehicle carries over to the others.

II. Competition Strategy

The competition strategy begins with the tasks rather than with the vehicles. Each task describes behaviours that define what the SoS must be able to sense, decide, and do [2].

The UAV can be used for reconnaissance and surveillance, and in advanced versions of the tasks, delivery of a payload. It receives its instructions from the USV through a network and sends its sensor data through the same path.

The UUV is the only way to accomplish tasks that require subsurface operations. It contains an onboard camera and various sensors [4], and has been fitted with a Ping360 scanning imaging SONAR and a Nortek Nucleus 1000 DVL. The UUV connects to the USV through a tether and feeds back sensor data to the processing centre onboard the USV. The team also standardises on a small set of open-source tools. ArduPilot provides low-level vehicle control [5], MAVROS [6] connects that control to ROS 2 [7], and ExpressLRS [8] through SBUS provides the manual control links.

The USV can be the central processing hub for the SoS, since unlike the UAV or the UUV, it is not overly constrained by weight or volume. It carries a sensor array consisting of a volumetric LiDAR, IMU, two cameras, and a compass in addition to standard GPS [3]. Since it has the largest payload and cooling capacities, it hosts the computer for the UAV and the UUV. These two vehicles are effectively treated as advanced sensor arrays.

A. Task 1 – Safe Passage

The main focus for this task is reliable beacon detection, obstacle avoidance, and autonomous path planning. Therefore, the plan for this task is to use the USV and UAV together to identify and navigate through the safe route.The team will first ensure that the USV can complete the navigation reliably before adding more complex communication and collaboration with the UAV.

To navigate through the maze, localization of both platforms needs to be established. A spatial relationship between the USV and UAV will be created using sensor data and the ROS transform tree. This allows the platforms and sensors to be localized within the same abstract 3D coordinate system. The platforms can then coordinate themselves within this system and plan through the data received.

The UAV will provide an aerial view of the task area and assist with identifying the coloured light beacons, while the USV will use its onboard perception and navigation systems to travel through the buoys. For colour identification, the system will use both aerial sensing from the UAV and onboard sensing from the USV. For obstacle identification, the USV will detect

obstacles using vision-based blob detection, with additional sensor data such as LiDAR (Hesai Pandar 40P) providing redundancy.

The data sent back from the UAV is translated into a feature map that contains the position and colour of the beacons, with subsequent pathing being informed by Delaunay triangulation. The UUV will attempt to hold position directly under the USV at all times during this task.

B. Task 2 – Infrastructure Survey and Repair

The main focus for this task is reliable pipeline detection, underwater localisation using the DVL, and autonomous pipeline following.

The DVL will be used to estimate the UUV's velocity and assist with underwater localisation. The Ping360 imaging sonar will be used to detect the pipeline and surrounding underwater features, particularly where camera visibility is limited. The pipeline will first be found using the drone coordinated by the USV main compute system. The UUV will then be sent to the start location and a line following algorithm is used to follow the pipeline. This will be done by converting the pipe and the background as a binary map so the line is made clear for the algorithm.

A downward-facing camera keeps track of the lights. If green, the UUV continues the line following algorithm, but if a red light is encountered, a bump and hold mode will be triggered. The magnetic probe sits at a known position on the vehicle, so it will be used as a coordinate to help with the localisation and odometry of the BlueROV. This along with the downward camera can be used to estimate the position of the light. The downward camera locates the light relative to the vehicle, and the localisation gives the offset the UUV must move to bring the probe onto the magnetic sensor mounted on the pipe. Once the green light is detected, the UUV will be sent to the position it was when the bump and hold mode was activated, then the line following algorithm is continued.

C. Task 3 – Coordinated Logistics

For this task, the USV and UAV will be used together to identify the designated delivery location and complete the required payload delivery. The focus for this task is reliable target identification, autonomous navigation, accurate positioning, and coordination between vehicles.

The team plans to use the USV as the primary navigation and communication platform, coordinating with the UAV to complete the required operations. The USV will use its onboard cameras, LiDAR, and GPS to identify and navigate towards the designated docking area. Sensor data from both vehicles will be used to determine the delivery location and assist with positioning and payload delivery.

The strategy will prioritise reliable USV navigation and docking before introducing more complex collaborative behaviours, including autonomous payload delivery using the UAV. Testing will initially focus on individual vehicle capabilities before progressing towards full SoS integration and coordinated task execution.

D. Task 4 – Dynamic Interruption

This task presents the challenge of dynamic interruptions. Whilst the platforms are completing another task, they must be able to reprioritise and focus on a new objective. This will be achieved by monitoring incoming commands from RoboCommand through the Operator Control Station (OCS). When a request is received, the specified platform must acknowledge the command, interrupt its current task and navigate to the provided GPS coordinates. Once at the location, the platform must hold its position and report its readiness to RoboCommand. A snapshot of the platform's current task state will be stored when the interruption occurs, allowing the previous task to resume from where it was interrupted once clearance is received.

III. Design Strategy

The system consists of a BlueBoat USV, a modified SwellPro Splashdrone 4 UAV, and a BlueROV2 UUV. Each platform provides capabilities suited to a pertinent operating domain while contributing to the team's overall autonomous maritime system.

Design decisions focus on reliable operation, autonomous navigation, perception, communication, and integration between vehicles. The design process was also influenced by the availability and cost of materials and components.

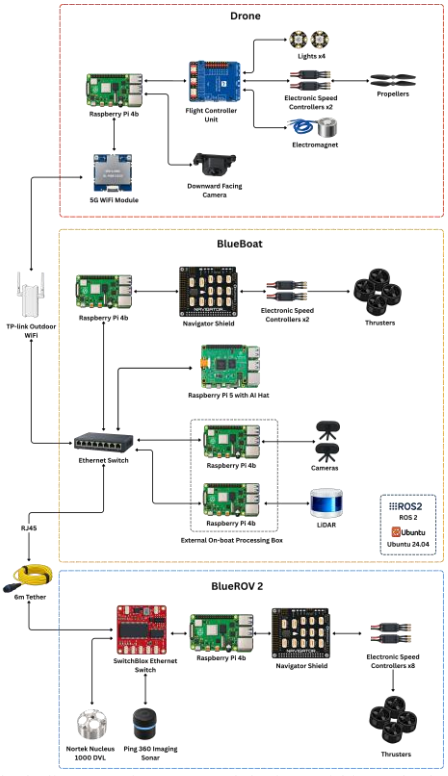


Fig. 1. Block diagram of sensors and devices within each platform

A. BlueBoat (USV)

The modifications to the Blueboat and the design process of additions to the vehicle centered around the need to improve the capabilities of the USV to better align with the functions required to tackle the tasks in the competition and meet the safety requirements.

1) Superstructure

The original BlueBoat platform utilised two cross members to provide structural connection between the hulls, however, this configuration did not provide sufficient mounting locations for the sensing, safety, and computing hardware required for autonomous operation. A dedicated superstructure was therefore developed to integrate the LiDAR, camera, emergency stop system, onboard processing hardware, lightbar support, and eventually area to house the water sprayer subsystem without requiring significant modification to the existing vessel.

The primary design requirements were structural rigidity, subsystem accessibility, sensor field of view, manufacturability, and compatibility with the existing crossmembers. LiDAR placement was prioritised due to its role as a primary perception sensor, with its position selected to maximise environmental coverage while minimizing occlusion from the vessel hulls. Camera and auxiliary hardware locations were then arranged around this geometry to maintain suitable fields of view and allow adequate spacing between subsystems.

The superstructure geometry was iteratively refined in CAD to improve stiffness, maintain sensor alignment under vessel motion, and provide sufficient mounting area for additional hardware. Material selection was also incorporated into the design process, with consideration given to corrosion resistance, cost, availability, and the team's manufacturing capabilities. The current design was able to be manufactured and provides a rigid platform that satisfies the integration requirements of the autonomous subsystems while preserving the original BlueBoat architecture.

2) Propeller Shrouds

Initially the USV had no shrouds fitted to its propellers, which poses the risk of damaging the environment and blades during operation and safety hazards to an individual retrieving the boat. To improve the safety of the vehicle and reduce its likelihood of damaging the environment, propeller shrouds that encapsulated the entirety of the propellers were designed for the USV. The shroud was mounted between the propeller motor and its mount. This mounting point was chosen as it simply required increasing the length of the bolts used for mounting the motor to integrate the shrouds with the existing propeller system.

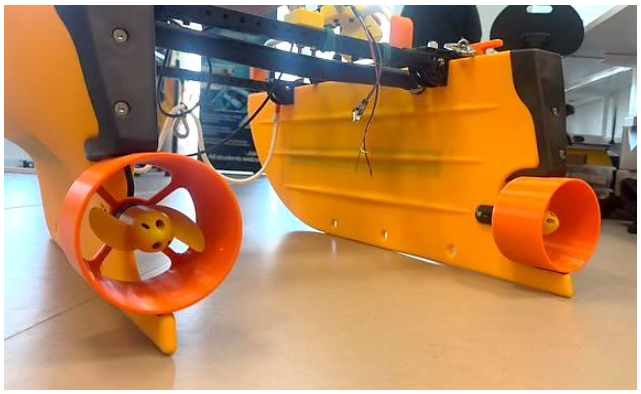


Fig. 2. Final propellor shroud design for the USV

During a review of an updated version of the competition requirements, the shrouds were redesigned. The propeller shrouds were extended so that the end of the shroud would be offset from the blades by 5 cm. Polyethylene terephthalate glycol (PETG) was specifically chosen for the propeller shrouds as it provides significantly greater strength and durability during long-term immersion compared to polylactic acid (PLA) and was readily available to the team.



Fig. 3. Superstructure Initial Design, with light bar visible behind the LiDAR assembly

3) Lightbar Brace Assembly

During construction of the lightbar, it became apparent that a mounting and bracing system needed to be manufactured to integrate the light to the boat. In the design of the lightbar brace assembly, the location of the light relative to the LiDAR superstructure was a key consideration. Placing the lightbar or its bracing assembly close to the LiDAR and inside its field of view would have diminished the boat's sensing capabilities. To avoid this issue, a requirement for the design of the assembly was to place the lightbar far away from the LiDAR as possible. The location of the bracing assembly was also constrained to the space below the field of view of the sensor.

In the design and manufacturing of the lightbar housing structure, the use of offcut materials from the LiDAR superstructure was set as a key but optional deliverable of the system. This consideration came into the design process because of the limited budget of the team and of the team's collective desire to reduce the waste produced thus far from the design process. Despite the short time limit for the design and implementation of this structure, a solution that met both requirements effectively and functioned well as a housing and bracing assembly for the lightbar was achieved.

When integrating the e-stop system to the boat, the design of the lightbar bracing assembly was changed. The e-stop buttons for the emergency stop system required an easily accessible and visible location on the vehicle. Its integration to the vehicle was also required on short notice. To effectively meet these requirements, the design of the connectors mounting the lightbar bracing assembly to the LiDAR superstructure was changed to additionally house the e-stop buttons. These were chosen as the location of the connector would allow the e-stop buttons to be highly visible and accessible. The connectors could also be easily and quickly changed because of their 3D printed nature. Changing the design of the connectors took a few iterations to properly house the buttons, but an effective design was eventually reached before the boat was tested in the water.

4) Mooring and Towing Structure

One of the key structures or systems that needed to be added to the vehicle during the team's initial work breakdown discussion was the mooring and towing system. Initially, the USV had no boat cleats or towing points that would allow the vehicle to be moored or towed. As such, entirely new mooring and towing structures had to be designed and implemented.

As the vehicle would be lifted into the air when it was towed, carabiners were selected to connect the vehicle to the lifting harness. These components were also selected as the vehicle had threaded holes where carabiners could be easily mounted to. Marine grade 316 stainless steel was specifically chosen due to its high corrosive resistance and strength, which would be appropriate for the vehicle's long-term exposure to marine environments. The placement and number of the tow points were chosen with care as well to ensure the stability and balance of the vehicle during lift.

To moor the USV, boat cleats, based on open-source designs and those available on the market, were designed and manufactured. Initially, the boat cleats were incompatible with the threaded mounting points on the vehicle and too small for the rope to wrap around the cleat effectively. Based on these issues during testing, iterations of the design were devised and implemented to address them. Eventually a design for the boat cleats that were big enough for the rope used to moor the vehicle and compatible with the mounting points on the boat was achieved.

During manufacturing of the boat cleats, it became apparent that the manner in which the cleat was printed required consideration. When testing the deflection and strength of the cleat, it was determined that printing the boat cleat such that its height was perpendicular to the heat bed resulted in greater deflection in the cantilever sections of the design. To overcome this issue and reduce the likelihood of the cleats fracturing, the cleats were printed such that its height was parallel to the heatbed. This printing method improved the strength of the boat cleats and reduced the deflection of its cantilever sections.

5) Emergency Stop System

The BlueBoat has independent onboard and handheld emergency stops that isolate propulsion power [4].

The handheld unit comprises of an ESP8266, an RFD900x radio, illuminated ARM and DISARM buttons, a power switch, and two 18650 cells.



Fig. 4. Emergency stop remote controller

These components are mounted on a board that handles USB Type-C charging and 5 V regulation, all housed in a 3D-printed case. A dedicated RFD900x link connects it to an ESP32 on the boat. The ESP32 drives a transistor-switched 5 V relay module that enables the motor-relay coil loop. The receiver releases the loop if valid radio traffic is absent for 500 ms, so that the system fails safe under signal or power loss from the transmitter.

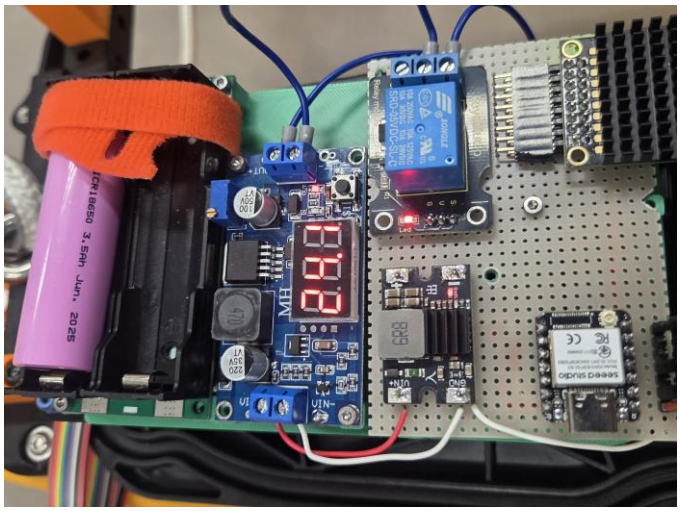


Fig. 5. Emergency stop loop USV

The ESP family of microcontrollers was chosen for its ease of programming and 3.3 V logic output, which the RFD900x requires. The built-in wireless radios of both microcontrollers are disabled, so the RFD900x link is the only wireless path into the emergency stop system.

A DC-DC boost converter powers two Jaycar SY4074

(FRC3) relays, each with a nominal 12 V coil, in series at 24 V [9].

Each relay's normally open contacts interrupt the battery feed to one hull motor's electronic speed controller (ESC). The onboard, normally closed stop button lies directly in the coil circuit, so pressing it releases both relays independently of the ESP32 and radio. Loss of boat-side emergency-stop power has the same effect. The RGB strip indicates vessel state. Direct sensing of an ESC's 5 V battery-eliminator-circuit (BEC) output to confirm ESC power loss is used to put the system in a safe state in case of loss of power to the propulsion from an external cause. This system can be configured to stop just the USV, or when tethered to UUV it stops both in tandem.

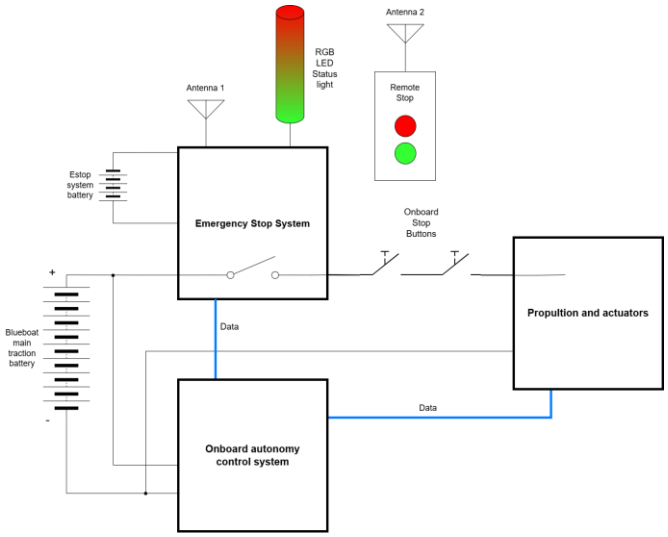


Fig. 6. Block diagram of emergency stop loop for USV and UUV

B. Drone (UAV)

The design of the UAV was driven by the necessity to adapt an existing SwellPro Splashdrone 4 to meet competition requirements. Since the Splashdrone was a water-resistant, highly integrated system with capabilities closely aligned with the challenge outcomes, it was cost advantageous to repurpose the extant system. However, detailed analysis of the system in relation to specific challenge requirements revealed several shortcomings.

A teardown of the drone showed proprietary transmission and control systems that could not easily be controlled except through the SwellPro controller. An open-source application programming interface (API) was found online, however, this software lacked full functionality across systems such as camera gimbal control, lighting control, along with mode switching and emergency stop capabilities. Thus, it was decided that the flight controller and other associated systems would be replaced with open-source hardware.

The main flight controller was replaced with a MatekSys H7A3-WING [10], and the transmission module was replaced with an LB-Link 5 GHz Wi-Fi module . This was paired with a Raspberry Pi 4, used to handle the video feed and transmission duties. A second ExpressLRS (ELRS) receiver, with its own antenna, adds a 2.4 GHz link to the handheld remote. Using two receivers separates basic manual control from the high-level autonomy systems, which helps the UAV meet the competition safety requirements.

The sensors (compass, GPS, barometer) and power electronics were reusable, as were the ESCs and the motors. These modifications made it possible to use ArduPilot, QGroundControl [11], and ROS 2, greatly simplifying the software integration of the drone into the SoS.

These changes to internal electronics necessitated the expansion of the electronics housing; the bottom plate of the drone was extended to increase the internal space available. This design underwent several iterations to optimise the internal layout of the drone, and to minimise the additional external volume required. Attention was paid to the mounting of internal components to ensure the mounting was sufficiently rigid and that the connections were not stressed. A mounting plate was used both to secure components without modifying the body of the drone and to provide several distinct channels for wire management. Because the housing changes, the whole airframe is re-waterproofed with silicon sealant. Custom lights were constructed using off-the-shelf LED strips of the same size as the original lights to reduce the time and cost of replacement.

The flight controller runs ArduPilot. ArduPilot is preferred over PX4 for its customisability, and many of the competition requirements can be met through its configuration alone. The Raspberry Pi 4 runs Ubuntu 24.04 and uses mavp2p to route MAVLink traffic between the flight controller and the ground control station.

Within the SoS, the UAV acts as a flying set of sensors rather than having any significant amount of compute power. It receives instructions from the USV, flies to waypoints at a predefined altitude, and streams video to the USV using the Real Time Streaming Protocol (RTSP). On request, it can also supply high-resolution still images through the Picture Transfer Protocol (PTP). Conversion between MAVLink and ROS 2 messages takes place on the USV through MAVROS.

The main lesson from the UAV work came from working on closed, proprietary systems. Reverse engineering the Splashdrone's custom transmission and control hardware takes considerable time and still does not give full control of the vehicle. Replacing the hardware with open-source components solves the problem and brings the UAV onto the same software foundation as the rest of the SoS. Future teams adapting a commercial platform should first confirm that it can be controlled through open interfaces, and should plan for replacing its electronics if it cannot.

C. BlueROV2 (UUV)

The team received a BlueROV2 Heavy from Blue Robotics to

serve as the UUV platform for the competition. The original vehicle configuration used a Raspberry Pi 4B with 2 GB of RAM running a 32-bit operating system image [12]. To improve onboard computing capability and provide greater compatibility with the team's software and autonomy systems, the base configuration was upgraded to a Raspberry Pi 5 running a 64-bit image. The upgraded system continues to interface with the Blue Robotics Navigator and BlueOS while providing additional processing capability for autonomous control, perception, and sensor integration. The UUV is also equipped with a Raspberry Pi camera and Ping360 imaging sonar to support underwater perception and pipeline detection required in task 2. GPS is unavailable underwater, so a Nortek Nucleus 1000 Doppler Velocity Log (DVL) measures the vehicle's velocity relative to the seabed to support localisation. For handling and safety, the UUV is also fitted with tow points and emergency lights.

1) BlueROV Emergency Stop Interface

The BlueROV interface uses a 6 m tether with four wire pairs: two for data, one for control, and one dedicated to the emergency-stop circuit. When the tether is connected, this final pair places the BlueROV's magnetic emergency-stop switch and the coil of an FRC6BC-1 relay [13] in series with the two USV relay coils. The boat's boost converter is then configured for a nominal 36 V loop supply. The BlueROV relay's normally open power contact interrupts its propulsion supply. Opening the magnetic switch or disconnecting the tether de-energizes the shared loop, releasing the USV and BlueROV relays.

D. RoboCommand Interface

During the competition, vehicles are required to interface with RoboNation's RoboCommand system. This system allows control of the start and end of runs, while also allowing vehicles to report the tasks they are attempting.

To interface with the provided system, FRT has developed a RoboCommand Interface – this connects via an Message Queuing Telemetry Transport (MQTT) interface to RoboNation's RoboCommand system, while exposing a set of ROS2 interfaces to the robotics vehicles. These interfaces connect directly to the topics published by the MAVROS systems and autonomy software. As such, state changes in the robot will automatically update the RoboCommand interface.

IV. Testing Strategy

The RobotX competition involves multiple autonomous systems using sensing, decision-making, navigation, control, software and mechanical hardware in a dynamic marine environment. To approach this challenge, testing was decomposed into as many minimal functional units as possible consisting of both hardware and software. Each functional unit's behavior was tested in isolation.

The testing progressed as a bottom-up process where the functionality performing power management, estop, positioning, and manual control was tested first, followed by low level autonomy such as waypoint following and geofencing, continuing to higher level autonomy, such as sensor processing, mapping and, planning. This culminates in the accumulation of the functional units building from the bottom up. In the design strategy, the aerial drone and the ROV would be required to demonstrate low level autonomy including waypoint navigation.

The development of the aerial drone consisted of successful automated waypoint control of the drone using the onboard flight controller, consisting of the following steps which demonstrate:

- RC communication with the flight controller and control mapping
- Reliable MAVLink telemetry from the flight controller to the BaseStation, QGroundControl, via the 5GHz link using the onboard Raspberry Pi 4
- The stable operation of the onboard EKF using the built in IMU, and the onboard GPS, pressure sensor and magnetometer to provide reliable 6DOF pose estimation.
- Uploading and deployment of waypoints
- Optimization of the AltHold mode via manual configuration of the PID controller
- Optimization of the Loiter mode via training of the PID controller using the onboard training mode
- Uploading, deployment and execution of waypoints (see figure)
- Successful imposition failure mode upon loss of control
- Successful imposition of geofencing both horizontally and vertically

The development of the ROV approximately followed the same development pathway as that of the drone, with the addition of testing the e-stop system cutting the power to the motors under mode change or loss of communication, horizontal positioning system, due to the lack of GPS, and the removal of the PID optimization steps, due to the PID loops already being optimized.

The development of the boat followed the same pathway as the drone except for the PID optimization due to the same reasons as the ROV.

A. Hardware Testing Strategy

The Flinders Robotics Team hardware testing strategy involves a pipeline of tests with each step moving towards full system integration. A component may perform correctly in isolation but fail once integrated into the complete system. If there are multiple failures in the complete system, this makes trouble shooting challenging. Our strategy minimizes this challenge and saves time for further development.

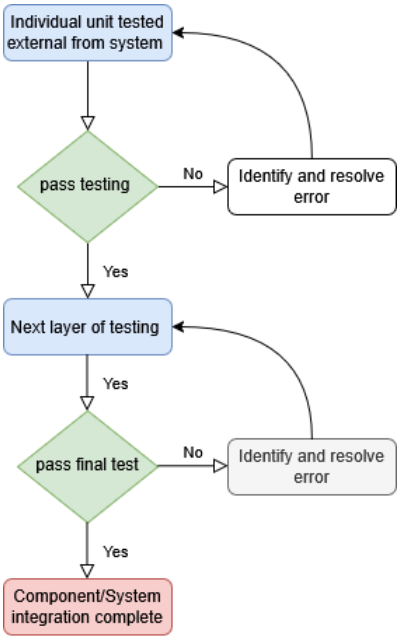


Fig. 7. Testing pipeline for hardware integration

1) Individual components: each component is tested on its own to confirm that it meets its requirements before it is integrated.
2) Subsystem verification: components are combined into subsystems, such as the emergency stop system, and tested as a unit.
3) System Integration: components are combined into subsystems, such as the emergency stop system, and tested as a unit.
4) In-field testing: components are combined into subsystems, such as the emergency stop system, and tested as a unit.

An item that fails at any stage returns for diagnosis and correction, then repeats that stage before it moves on. Four principles apply at every stage. Rapid prototyping, particularly through 3D printing, allows designs to be tested and revised quickly. Unit testing confirms that each element works before it becomes part of a larger assembly. Each test is planned to minimise risk to the vehicles and to team members. Finally, all testing follows university guidelines and safety standards.

B. Emergency Stop Verification

Functional tests demonstrated ESC and thus motor supply disconnection on remote disarm, onboard e-stop actuation, transmitter battery removal, link loss, and removal of boat-side emergency-stop power. The RGB strip showed solid red when stopped via both the remote and onboard estop buttons. Manual and autonomous mode changes were also exercised. The configured link-loss timeout is 500 ms; measured stop latency was not recorded. Before final submission, the team will time ESC-supply removal against the handbook's less-than-two-second limit (RobotX, 2026) and document the installed signal path that produces the red indication.

C. Software Testing Strategy

The Flinders Robotics Team software testing strategy follows a similar approach to the hardware testing. Individual functions are verified before they're integrated into the complete system. Software that performs correctly by itself may behave differently when integrated into the complete system or tested in real conditions. By testing software progressively, errors can be identified and corrected early, reducing the risk of failures during competition time.

1) Individual algorithms: Individual algorithms and functions are to be tested independently to verify that they produce expected outputs.
2) Software integration: These individual functions are then combined into larger software modules to verify that they can communicate and operate correctly together.
3) Simulation testing: The integrated software is to be tested in a simulated environment representing the RobotX competition. Virtual obstacles, GPS coordinated and mission objectives are introduced to evaluate autonomous navigation and decision making without needing to go through the cumbersome process of physically testing platforms.
4) In field testing: This is the final test for the software. It's deployed in physical platforms and tested under realistic operating conditions. This allows the team to evaluate the performance of the autonomous system using real sensor data and environmental conditions.

Software that fails at any stage is debugged and corrected before being tested again. Previously completed tests are repeated where necessary to ensure that modifications have not introduced additional errors.

D. Balancing Design with Experimentation

Several design decisions in this paper rest on testing and review instead of assumption. Strength and deflection tests on the 3D-printed boat cleats show that print orientation strongly affects stiffness. Cleats printed with their height perpendicular to the heat bed deflect more in their cantilever sections than cleats printed with their height parallel to it. The team therefore prints the cleats parallel to the bed, which reduces deflection and the risk of fracture. Mooring trials also show that the early cleats do not fit the vehicle's threaded mounting points and are too small for the mooring rope, and these findings drive further design iterations.

Review also shapes the design. A check of the updated competition requirements leads to the propeller shroud redesign described in Section III.

V. References

[1] "RobotX Challenge 2026 - RobotX." Accessed: Oct. 10, 2026. [Online]. Available: https://robotx.org/programs/2026/
[2] RoboNation, "RobotX 2026 Team Handbook." Accessed: Oct. 10, 2026. [Online]. Available: https://robonation.gitbook.io/robotx-2026-team-handbook
[3] "BlueBoat Uncrewed Surface Vessel," Blue Robotics. Accessed: Oct. 10, 2026. [Online]. Available: https://bluerobotics.com/store/boat/blueboat/blueboat/
[4] "BlueROV2," Blue Robotics. Accessed: Oct. 10, 2026. [Online]. Available: https://bluerobotics.com/store/rov/bluerov2/
[5] "ArduPilot Documentation — ArduPilot documentation." Accessed: Oct. 10, 2026. [Online]. Available: https://ardupilot.org/ardupilot/
[6] "mavros - ROS Wiki." Accessed: Oct. 10, 2026. [Online]. Available: https://wiki.ros.org/mavros
[7] "ROS 2 Documentation," ROS 2 Documentation. Accessed: Oct. 10, 2026. [Online]. Available: https://docs.ros.org/en/jazzy/index.html
[8] ExpressLRS, "ExpressLRS." Accessed: Oct. 10, 2026. [Online]. Available: https://www.expresslrs.org/
[9] "Horn Relay - 60A SPDT | Jaycar Australia," Jaycar Australia PROD. Accessed: Oct. 10, 2026. [Online]. Available: https://www.jaycar.com.au/horn-relay-60a-spdt/p/SY4074
[10] "(EOL) Flight Controller H7A3-WING – MATEKSYS." Accessed: Oct. 10, 2026. [Online]. Available: https://www.mateksys.com/?portfolio=h7a3-wing
[11] "QGC," QGroundControl - Drone Control. Accessed: Oct. 10, 2026. [Online]. Available: https://qgroundcontrol.com/
[12] "Technical Reference." Accessed: Oct. 10, 2026. [Online]. Available: https://bluerobotics.com/learn/technical-reference/#software-and-firmware
[13] "Automotive Relay 150A SPDT | Jaycar Australia," Jaycar Australia PROD. Accessed: Oct. 10, 2026. [Online]. Available: https://www.jaycar.com.au/automotive-relay-150a-spdt/p/SY4073