Team Circle by Republic Polytechnic

Team Circle consists of Staff, Students, Alumni and Advisors from Republic Polytechnic.

Low Wei En, Lee Feng Chen, Lau Jia Yu Amus, Li Zi Feng, Shafriel Nur Iman Bin Shammuain, Gong Xichen, Isaac Chan Gai Tza, Lim Kai Yao, Mr Mathivaanan Vedaraj, Mr Goh Yi Ming, Mr Ng Wei Jye

***Abstract*—Republic Polytechnic's Team Circle joins the Maritime RobotX Challenge for the first time. As a new team, we explore how to develop and further the research and direction of Uncrewed Surface Vessels (USV), Uncrewed Underwater Vehicles (UUV) and Uncrewed Aerial Vehicles (UAV). This paper sets out how the team's strategy of reliable Core-tier performance across the 2026 mission tasks shapes the design of its BlueROV2-based surface and underwater vehicles and its beacon-relaying UAV, and how a staged simulation-to-water testing programme is used to validate them.**

I. Overview

Team Circle is a mixed and pioneering team of diploma students, staff and alumni of our school who are research assistants from Republic Polytechnic, a Tertiary-Level Education Institute in Singapore. This is Team Circle's debut in the RobotX Challenge, marking the first time a Singaporean polytechnic has participated in the challenge. By participating in the challenge, we hope to gain a deeper understanding of utilising uncrewed systems in our daily lives. Team Circle also hopes that, apart from purely competing in a new challenge and domain, the members of the team would be able to learn, adapt to and develop designs and fundamentals beyond what the team learns on a day-to-day basis in school. This is to hopefully propel the team in preparation for further research and competitive experiences. This paper will showcase not only Team Circle's strategies in developing the uncrewed systems, but to also allow readers to explore the learning curves, difficulties faced and the team's enjoyment in developing an uncrewed application on new ground while showing how Team Circle is working towards the objectives of the 2026 RobotX Challenge.

A. What We Are Building

For the 2026 Maritime RobotX Challenge, Team Circle is developing a multi-domain system of systems comprising an Uncrewed Surface Vessel (USV), an Uncrewed Underwater Vehicle (UUV), and an Uncrewed Aerial Vehicle (UAV). The surface and underwater vehicles are built on the Blue Robotics BlueROV2 platform [1], which the team has converted from remote operation to autonomous control. At present a single six-thruster BlueROV2 serves as the development platform for both the surface and underwater roles. The competition UUV will use an eight-thruster layout with four vertical thrusters; it has not yet been received, and the software, safety and electrical design developed on the current platform is intended to carry over to it.

The UAV operates as the aerial component of the system of systems. Its mission is to maintain an aerial view of the operating area, identify the coloured light beacons marking the course, and pass the position of those beacons to the surface vessel so that the USV can navigate with reference to them. The three vehicles are not developed as independent projects but as cooperating elements of one mission architecture, which is required by the 2026 competition's shift from optional to mandatory multi-vehicle autonomy.

B. Our Overall Philosophy

The team adopts a systems-based approach to development, where mechanical, electrical, and software systems are designed to work together towards the mission objectives. Rather than pursuing unnecessary complexity, the team prioritises solutions that are safe, modular, simple to implement, and directly aligned with the requirements of each competition task. This allows the team to focus its resources on capabilities that contribute directly to mission completion while maintaining a system that can be tested, maintained, and improved efficiently.

This paper will highlight how Republic Polytechnic's design strategy guides the development and integration of the team's mechanical, electrical, and software systems, how its testing strategy is used to validate system performance and compliance, and how the resulting system is applied to the task objectives of the 2026 RobotX competition.

II. Competition Strategy

A. Mission Overview

The 2026 Maritime RobotX Challenge is themed "Storm Response: Technology in Action for Recovery and Relief". The scenario is a major storm affecting a coastal port facility. Teams deploy autonomous systems to support recovery and restoration of maritime operations. The mission requires coordination between Uncrewed Surface Vehicles (USVs), Uncrewed Underwater Vehicles (UUVs), and Uncrewed Aerial Vehicles (UAVs) across multiple maritime domains.

The mission broadly involves four operational objectives:

- Assessment and Discovery. Survey the affected area from air and sea to identify safe passage areas and assess damage.
- Recovery and Repair. Inspect and repair damaged underwater infrastructure such as cables or pipelines.

- Logistics and Delivery. Coordinate autonomous systems to deliver materials or resources required for recovery operations.
- Detection and Tracking. Detect and track a rogue or unidentified element, such as a suspicious vessel or floating debris.

These activities are intended to represent real-world applications of autonomous systems in post-disaster port recovery and resilience. They are realised in the competition as four scored mission tasks, supported by four cross-cutting capabilities that apply to every run [2], [3]. Because the 2026 edition makes multi-vehicle operation mandatory rather than optional, and requires all systems to be in autonomous mode before a collaborative run begins, Team Circle treats inter-vehicle collaboration as a baseline system requirement rather than an enhancement to be added once single-vehicle autonomy is mature.

B. Task Analysis

Each mission task is graded across three capability tiers [2]. Core capabilities are the fundamental competencies required for safe baseline autonomous operation. Advanced capabilities reflect growth in autonomy sophistication, and in practice are where multi-vehicle collaboration becomes mandatory. Disruptive capabilities push the boundary of autonomous systems further still. Table I summarises the tasks, the vehicles each requires, and the tier Team Circle is targeting in its debut year.

TABLE I. ROBOTX 2026 TASK SET AND TEAM CIRCLE TIER TARGETS

Task	Vehicles	Team target	Pri.
Cross-cutting			
RoboCommand reporting	All	Required	1
Obstacle avoidance	All	Required	2
Inter-vehicle comms	All	Advanced	5
Dynamic adaptability	All	Core	7
Mission Tasks			
1: Safe Passage	USV/UUV, UAV Adv.	Core, Adv. if UAV ready	3
2: Infrastructure Survey	UUV req., UAV Adv.	Core	4
3: Coordinated Logistics	USV and UAV primary	Core	6
4: Dynamic Incident Response	USV, UAV	Core	8
Mandatory Activities			
Static safety inspection	All	Pass	1
Weight measurement	All	Pass	1

[a.] Tier definitions and task content from the 2026 Team Handbook [2]. Point values had not been published at the time of writing.

Two features of the task set drive the team's design. First, Task 1 at the Advanced tier makes the safe passage detectable only by the UAV, which must relay the entry and exit points to the surface system [4]. This is the aerial beacon-reporting capability described in Section III-G, and it is the task where the team's UAV concept maps most directly onto a scored requirement. Second, Task 2 requires the UUV to survey and report the condition of an underwater pipeline [5]. This depends on submerged navigation, depth hold and visual inspection, which the team has begun to develop on its current platform but can only complete once the competition UUV is received.

Two separate colour schemes appear in this paper and should not be confused. The course RoboBuoys carry LED light beacons whose states encode navigation meaning: flashing red means pass on the starboard side, flashing green means pass on the port side, flashing blue marks the entry point, solid blue marks the exit point, and an unlit buoy is an obstacle [4]. The team's own vehicles, by contrast, carry a three-colour status tower that reports each vehicle's control mode to judges, described in Section III-I.

C. Task Prioritisation

As a first-time entrant, Team Circle's prioritisation is governed by a single principle: points that are reliably earned are worth more than points that are theoretically available. The team therefore ranks tasks by estimated value rather than by maximum points. Adapting the approach used by Team Minion [6], the estimated value of task i is expressed as

$$EV(i) = c_i \, (core_i + \Sigma_j \, c_j \cdot cg_{i,j}) \quad (1)$$

where c_i is the team's confidence in completing task i, $core_i$ is the points earned on task i independently of other tasks, and $cg_{i,j}$ is the portion of task i's points that is contingent on first completing task j. Point values for the 2026 course have not been published at the time of writing, so (1) cannot yet be evaluated numerically. The team uses it instead as a ranking framework, substituting tier attainment for point value: a Core-tier completion that the team is confident of is preferred to an Advanced-tier attempt that it is not.

Applying that reasoning produces the priority ordering in Table I. The cross-cutting capabilities rank first, because RoboCommand reporting and obstacle avoidance apply to every run and a failure in either affects all four tasks rather than one. Mandatory activities are treated as gating: a vehicle that does not pass static safety inspection does not compete at all, so inspection compliance is designed in rather than demonstrated late. Tasks whose Core tier builds on capabilities already shown on the development platform rank next, with Task 2 re-assessed once the competition UUV is received. Tasks in which the UAV plays a primary role rank lowest for the present development cycle, as the aircraft has not yet been procured.

D. System Requirements

The task set generates the following system-level requirements, which are traced directly into the design decisions described in Section III:

- The system shall report each vehicle's status to RoboCommand and adapt to mission changes issued during a run.
- All vehicles shall detect and avoid obstacles, including unlit obstacle buoys.

- The USV and UAV shall exchange mission data, so that beacon and tasking information observed by one vehicle can be acted on by another.
- The USV shall navigate autonomously between waypoints and hold position sufficiently to permit perception tasks.
- The UUV shall hold depth and heading, and capture imagery of a submerged structure at close range.
- The system shall provide an emergency stop that removes propulsion power independently of the processor for automation.
- Each vehicle's control mode shall be externally visible to judges and safety personnel at all times.
- The vehicle shall be positively buoyant, detect water ingress, and present the tow points, lift points and propeller guards required for static safety inspection.
- The safety, electrical and software design shall carry over to the competition UUV without redesign.

E. Competition Approach

Team Circle's strategic vision for 2026 is deliberately conservative: qualify reliably, score consistently at the Core tier across as many tasks as possible, and use the competition as the foundation of a multi-year programme rather than attempting the full task set in a debut year. Disruptive-tier capability is not pursued in the first year, and Advanced tier is treated as conditional on the UAV being procured and flown in time. This reflects the central trade-off identified in the competition guidance: a fixed number of working hours may be spent either adding capability or improving the reliability of capability already built. The team accepts a lower theoretical ceiling in exchange for a higher probability of scoring the points it does attempt.

The team's members are diploma students, staff and alumni working alongside academic commitments, so available engineering hours are both limited and irregular. Under that constraint, each additional subsystem carries a compounding integration cost: a change in one subsystem propagates into others, and with limited time those propagations become difficult to manage. Team Circle therefore caps the number of concurrently developed capabilities and requires that a capability be demonstrated end-to-end on the water before a new one is started. The dated development record in Section IV-H shows this policy in practice: underwater heading and depth control on the development platform were carried through repeated troubleshooting cycles to a working state before further autonomy functions were opened.

F. Confidence Assessment

To support the prioritisation above, the team characterised its confidence in each tool and capability on a 0 to 100 scale, as shown in Table II. Values for demonstrated capabilities are based on the on-water and in-pool test results described in Section IV. Where a capability has so far been shown only on the development platform and must be re-established on competition hardware, its value is discounted accordingly. Depth hold is the main example: depth control on the competition UUV will depend on four vertical thrusters working together, so the results obtained to date are treated as early-stage. Values for undemonstrated capabilities reflect the team's assessment of development risk, and are expected to change as testing continues.

TABLE II. CONFIDENCE LEVEL IN TEAM CIRCLE'S TOOLSET (0 – NO CONFIDENCE, 100 – PERFECT CONFIDENCE)

Capability	Conf.	Basis
Platform and Handling		
Manual RC control	90	Demonstrated
Tow operation	80	Demonstrated
Lift and recovery	80	Demonstrated
Extended-range operation	65	Demonstrated
Safety Systems		
Emergency stop	95	Demonstrated
Mode status indication	85	Demonstrated
Software failsafes	80	Demonstrated
Leak detection	60	Fitted, untested
Surface Autonomy		
Waypoint navigation	70	Demonstrated
Disturbance recovery	60	Demonstrated
Station keeping	50	In development
Obstacle avoidance	35	In development
Underwater Autonomy		
Depth hold (ALT_HOLD)	50	Early-stage testing
Heading control, 180°/360°	70	Demonstrated
Submerged dead reckoning	30	Requires DVL
Pipeline survey and report	25	In development
Perception		
Live visual feedback	75	Demonstrated
Object detection	30	In development
Beacon colour identification	40	In development
Aerial Systems		
Autonomous flight	40	Prior project
Beacon detection from altitude	35	In development
Pixel-to-position transform	25	In development
Payload delivery	15	Not yet developed
Collaboration and Reporting		
RoboCommand reporting	30	In development
UAV to USV data relay	20	In development
Dynamic re-tasking in run	20	In development

[b.] Confidence values are reviewed after each test cycle and will be re-assessed immediately before competition.

III. DESIGN STRATEGY

The Republic Polytechnic RobotX team adopts four key principles to guide the development of its autonomous maritime system: Safety, Modularity, Simplicity, and Task Orientation. These principles are applied throughout the design process to ensure that the system remains reliable, maintainable, and focused on fulfilling the requirements of the RobotX 2026 mission.

A. Design Philosophy

1) Safety

Safety is the primary consideration in the design and operation of the system. The team considers the safety of

personnel, equipment, and other vessels throughout the development process. Safety features are incorporated into both hardware and software systems to minimise the consequences of unexpected system behaviour or component failure.

Safety was treated as an architectural constraint rather than a later addition. This is most visible in the emergency stop, which does not depend on software to function. As described in Section III-D, the emergency stop is a manual switch wired directly into the 14.8 V supply feeding the thrusters. Asserting it removes drive power whether or not the processor for automation is running, responsive, or executing correct code. The positively buoyant hull and the leak sensors described in Section III-I follow the same reasoning: neither depends on the mission software working.

2) Modularity

The system is designed using modular components and interfaces to allow individual subsystems to be developed, tested, replaced, and improved independently. It reduces the impact of changes made to one subsystem on the rest of the vehicle and allows the team to troubleshoot problems more efficiently.

A modular approach also provides flexibility during competition preparation, as components can be modified or replaced without requiring significant changes to the entire system. This is particularly important for a first-time team, where iterative development and learning are expected throughout the project.

Three consequences of this principle are worth noting. First, all thrusters are the same T200 model, so a damaged unit is replaced from a common spares pool. Second, mission logic and closed-loop vehicle control run as separate software layers, so the team can change mission behaviour without disturbing the control loop that keeps the vehicle stable. Third, the signal tower and emergency stop were specified with simple electrical interfaces, which allowed both to be integrated without modification to the autonomy software. The same interface discipline is what allows the safety, electrical and software design to be carried over to the competition UUV rather than developed again.

3) Simplicity

The team prioritises simple and practical solutions where they can satisfy the required function without introducing unnecessary complexity. A simpler system is easier to understand, integrate, troubleshoot, and maintain, while reducing the number of potential points of failure.

For example, the signal light and E-Stop systems are designed with simple interfaces so that they can be integrated with different vehicle systems without extensive modifications. This approach allows the team to maintain essential functionality while keeping the overall system manageable during development and operation.

The tow and lift arrangement described in Section III-C is a further example: it meets the competition requirement with fixed hard points and rope harnesses, adding no mechanism that could fail.

4) Task Orientation

The system is designed around the requirements of the RobotX 2026 mission. Rather than developing capabilities that do not directly contribute to the competition objectives, the team prioritises functions that are necessary to satisfy the mission criteria.

Each design decision is therefore evaluated based on its contribution to task completion, system reliability, and available development resources. This task-oriented approach enables the team to focus its efforts on achieving the required mission capabilities while avoiding unnecessary features that may increase system complexity without providing a corresponding benefit to mission performance. Table III traces representative requirements from the competition task that generates them to the design decision that satisfies them.

TABLE III. TRACEABILITY FROM COMPETITION TASK TO DESIGN DECISION

Task driver	Requirement	Design decision
All on-water operation	Remove propulsion power independently of software	Manual E-Stop switch wired in the 14.8 V thruster supply
Static safety inspection	Control mode visible to judges at range	Three-colour status tower driven through a relay bank
Static safety inspection	Vehicle towable and liftable, propellers guarded	Fixed frame hard points, rope harnesses, thruster shrouds
Submerged operation	Fault must not result in loss of vehicle	Positively buoyant hull; leak sensors on Navigator inputs
Unattended autonomous runs	Vehicle must not operate indefinitely on fault	Mission timeout, auto-disarm on exit, operator interrupt
Task 1, Advanced tier	Entry and exit points detectable only from the air	UAV aerial survey with pixel-offset transform to a common origin
Task 2, Core tier	Survey submerged pipeline and report condition	Four vertical thrusters for depth control; downward camera
Cross-cutting	Report vehicle status every run	RoboCommand messaging from the onboard Raspberry Pi
Competition UUV	Design carries over from the development platform	Common safety chain, electrical layout and software layers

Together, these four principles establish the team's overall design philosophy: develop a safe and modular system, keep its implementation simple, and focus engineering effort on the capabilities required to complete the mission.

B. System Architecture

The system of systems comprises three elements, shown in Fig. 1. The USV is the primary mission platform and carries the processor for automation, propulsion, power and safety systems. The UUV performs underwater survey and inspection tasks; until the eight-thruster competition vehicle is received, the six-thruster development platform is used in

this role. The UAV provides aerial coverage and beacon localisation, and feeds beacon positions to the USV.

Onboard processing runs on a Raspberry Pi fitted with a Blue Robotics Navigator board [7]. The Navigator provides the inertial sensors, leak-sensor inputs and pulse-width modulation (PWM) outputs, and ArduSub [8] runs on the Pi to close the vehicle's control loop. The team's mission software runs alongside it as a separate process, handling mission logic, perception and the RoboCommand interface, and issues high-level movement commands that ArduSub resolves into individual thruster outputs. Because the two run separately, an error in the mission script does not stop ArduSub from holding the vehicle's attitude and depth.

Mission data flows in one principal direction. The UAV detects beacons and computes their position relative to a common origin, which is the USV launch point. Those positions are transmitted to the ground station and to the USV, which uses them as navigation references. The USV retains authority over its own motion at all times; the UAV supplies information, not commands. This division keeps the safety-critical control loop on a single vehicle and avoids a failure mode in which loss of the aerial link leaves the surface vessel without control.

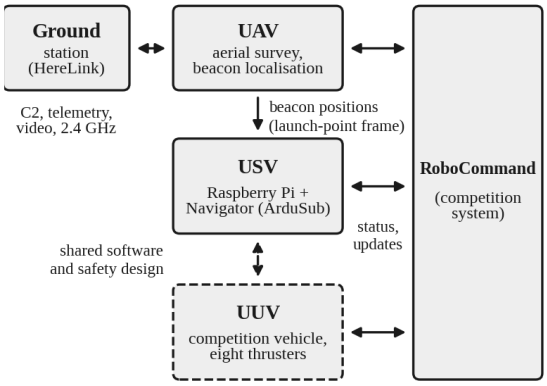


Fig. 1. Team Circle system-of-systems architecture, showing UAV beacon reporting into USV navigation and RoboCommand status reporting.

C. Mechanical System

The current development platform is built on the Blue Robotics BlueROV2 frame and enclosure [1] and driven by six T200 thrusters [9] in the standard vectored configuration, two of which act vertically. The competition UUV will use eight thrusters, four of them vertical. This doubles the thrusters available for depth control and changes how vertical thrust must be distributed, so the vehicle's weight, buoyancy and balance will be characterised again once it is received.

The mechanical decisions confirmed to date follow the requirement-to-result pattern used throughout this section. The requirement for lift and tow handling admitted several options, including dedicated lifting eyes, a cradle, or reuse of existing frame structure. The team selected fixed hard points on the existing frame with rope harnesses, on the grounds that the frame already carried the necessary load path and the solution added no mechanism that could fail. The result has been verified by lift and tow trials described in Section IV.

The requirement to guard rotating machinery near personnel was met with shrouds fitted to all thrusters. Because every thruster is the same model, a single shroud design serves the whole vehicle and spares are interchangeable, which supports both the safety and modularity principles at once. Buoyancy is trimmed so that the vehicle is positively buoyant in the operating condition, which satisfies the recovery requirement described in Section III-A. The assembled USV is shown in Fig. 2.

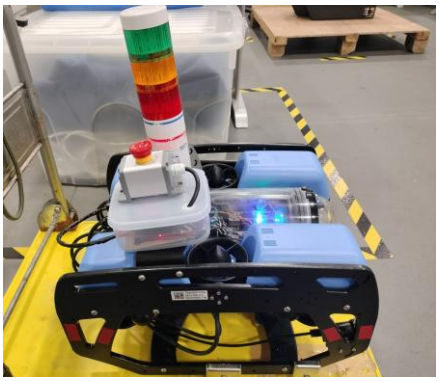


Fig. 2. Assembled USV showing the E-Stop switch, three-colour status tower and relay enclosure on the BlueROV2 frame.

D. Electrical System

The electrical architecture of the current USV, shown in Fig. 3, is a tentative configuration. The team is still experimenting with power distribution and component selection, and the values given are current working figures rather than a frozen design.

Power is supplied by a four-cell lithium polymer battery at a nominal 14.8 V, which feeds the six thrusters through their electronic speed controllers and, via a regulated 5 V rail, the Navigator board and Raspberry Pi. The emergency stop is a manual switch placed in the 14.8 V thruster supply, so that asserting it removes drive power at source. Control signals pass from the Navigator's PWM outputs to the speed controllers.

Mode indication is handled by a separate signal chain. Three single-channel relay modules and a four-channel relay module are driven from the Navigator's signal pins and switch the three-colour status tower through a common terminal block. Keeping the tower behind relays separates the indicator circuit electrically from both the control electronics and the propulsion supply, so a fault in the indicator cannot propagate into either.

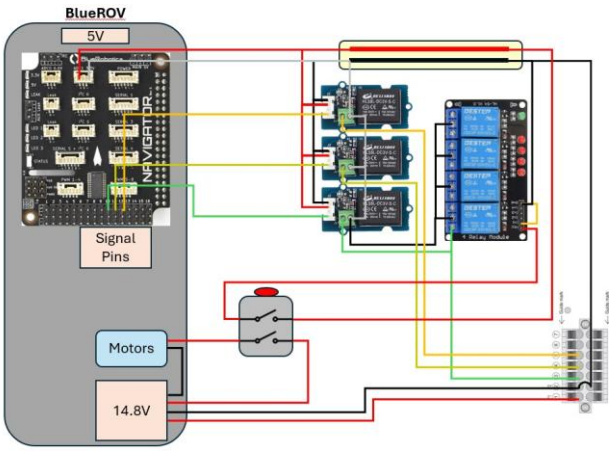


Fig. 3. USV electrical circuit: 14.8 V battery, 5 V rail, Navigator board, thrusters, manual E-Stop in the thruster supply, relay bank and three-colour status tower.

Protection measures comprise the external kill switch, leak sensors on the Navigator's dedicated leak inputs, electrical isolation between the propulsion and computing rails, and sealing of enclosures and connectors against water ingress. Fusing values and final isolation arrangements are being finalised as the configuration is validated through testing.

The onboard sensor suite is listed in Table IV. The vehicle carries no GPS receiver; as satellite positioning is unavailable to a submerged vehicle in any case, the team is developing a single position-estimation approach for both the surface and submerged roles, described in Section III-E.

TABLE IV. USV/UUV SENSOR SUITE

Sensor	Model	Function	Status
IMU	Navigator board: gyroscope, accelerometer, magnetometer	Attitude and heading	Fitted
Pressure	Blue Robotics Bar30 [10]	Depth and water temperature	Fitted
Camera	Blue Robotics low-light USB camera [11]	Visual feedback	Fitted
Leak	Blue Robotics leak sensor	Water ingress warning	Fitted
GPS	Not fitted	Not used; see Section III-D	N/A
DVL	Doppler velocity log	Submerged dead reckoning	Planned
Camera	DeepWater Exploration exploreHD USB [12]	Inspection imagery	Planned
Sonar	Blue Robotics Ping360 [13]	Obstacle and structure detection	Planned

[c.] Planned items are required for Task 2 pipeline survey and for obstacle avoidance; procurement is in progress.

E. *Software System*

The software is organised into two layers. The sensing and perception layer continuously polls the inertial measurement unit (IMU), the water temperature gauge and the depth pressure sensor, and maintains the vehicle's estimate of its own attitude, heading and depth. The control and actuation layer implements a closed-loop controller, presently proportional-integral-derivative (PID) with model predictive control under evaluation, which computes instantaneous thrust corrections across all thrusters. When a current pushes the vehicle off station, it is this layer that resolves the disturbance into per-thruster corrections to hold physical position. Gains tuned on the six-thruster development platform will not transfer directly to the competition UUV, where depth is held by four vertical thrusters acting together, so thrust allocation and tuning will be repeated on that vehicle.

Vehicle-side mission control is implemented in Python on the onboard Raspberry Pi, which communicates with ArduSub to command movement. The script is executed onboard rather than from an external laptop, so the vehicle is not dependent on a continuous operator connection during autonomous runs. Control modes exercised to date include MANUAL, STABILIZE and ALT_HOLD, together with commanded heading changes. The team intends to migrate inter-module communication to the Robot Operating System (ROS) [14] as the architecture matures, which would also ease reuse of existing perception packages.

Navigation is waypoint-based. Because no GPS receiver is fitted, the vehicle does not take a satellite fix; position is estimated by dead reckoning from the IMU and, once procured, a Doppler velocity log (DVL), referenced to the launch point. Beacon positions supplied by the UAV enter this pipeline as additional navigation references expressed in the same frame. The team recognises that dead-reckoning error accumulates with time and distance, and treats bounding that drift as the principal open problem in surface and submerged navigation.

Perception presently comprises live visual feedback from the onboard camera in both manual and autonomous modes, which has been demonstrated on the water. Automated object detection and classification, required for beacon identification and obstacle avoidance, are in development. Mission logic is organised as a sequence of task states, with each task entered on satisfaction of a defined starting condition and exited on completion or on timeout. The simulation environment is confirmed as Gazebo [15], which is used as the first stage of the testing pipeline described in Section IV.

F. *Autonomy*

The autonomy architecture answers a single question: how does the vehicle decide what to do without direct human control? Team Circle structures this as a five-stage loop of sense, understand, decide, act and monitor, shown in Fig. 4.

- Sense. Onboard sensors provide attitude, heading, depth and imagery; the UAV contributes beacon observations from altitude; RoboCommand contributes mission updates.
- Understand. Sensor data is resolved into vehicle state and into the positions of mission features in a common reference frame anchored at the launch point.

- Decide. Mission logic selects the active task and the next waypoint according to the task sequence, the current vehicle state, and any re-tasking received during the run.
- Act. The control layer converts the selected waypoint into thrust commands issued as PWM outputs to the speed controllers.
- Monitor. Failsafes run continuously and independently of mission progress. They stop and disarm the vehicle on timeout, operator interrupt or abnormal termination, and raise a warning if the leak sensors detect water ingress.

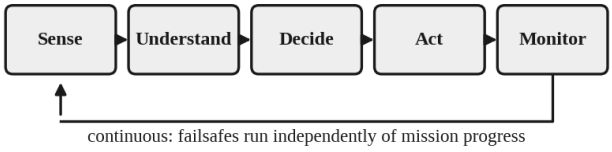


Fig. 4. Autonomy loop: sense, understand, decide, act, monitor.

Autonomous recovery is conservative by design: where the vehicle cannot resolve a situation, it stops safely rather than attempting an unverified manoeuvre. An operator can interrupt the control script at any time, and the team expects to relax this supervised posture as confidence in individual behaviours grows.

G. UAV Concept of Operations

The UAV's mission is to identify beacons and to help the USV and UUV navigate. The concept of operations is to hold an aerial view of the water surface, identify the coloured light beacons within that view, and express their positions with reference to the USV's launch point so that the surface vessel can act on them. This maps directly onto Task 1 at the Advanced tier, where the safe passage is detectable only by the UAV and the entry and exit points must be relayed to the surface system [4]. At the Disruptive tier the route may change mid-transit and the UAV must remain on station to provide updates in real time; the team does not plan to attempt this tier in 2026.

Two methods of obtaining beacon position were considered. In the first, the UAV flies directly over each beacon in turn and records its own position at the moment it is overhead, giving an approximate beacon position. This is simple and accurate per beacon, but it is inherently sequential: the aircraft must visit every beacon, which is slow and scales poorly against a field of ten buoys within a limited mission window.

In the second, the UAV holds a single vantage point covering the whole operating area, with the camera held pointing straight down by its gimbal. The image is treated as a plane in which each pixel corresponds to a known ground distance, set by the aircraft's altitude h and the camera's horizontal field of view θ, as shown in Fig. 5. The width of water covered by the image is

$$W = 2h \tan(\theta/2) \quad (2)$$

and a beacon that appears $u - u_0$ pixels from the image centre, in an image N pixels wide, lies at a ground offset of

$$\Delta x = (u - u_0)\, W / N \quad (3)$$

from the point directly below the aircraft. The offset Δy is found in the same way from the vertical field of view. The two offsets are rotated by the aircraft's heading into the launch-point frame and added to its own position, so every beacon in view is localised from a single observation.

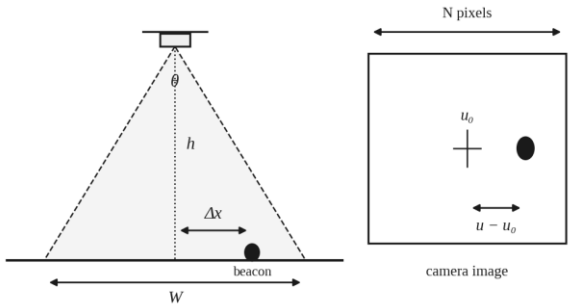


Fig. 5. Pixel-offset beacon localisation: side view of the camera footprint (left) and the corresponding camera image (right).

The team has selected the second method. The trade-off is explicit: the pixel-offset approach introduces a dependency on altitude accuracy and camera calibration that the hover-over method does not have, and its position error grows with distance from the image centre. In exchange it removes the sequential flight time that dominates the first method and allows the whole beacon field to be reported in a single pass. Given that the UAV's output is a navigation reference for the USV rather than a precision fix, the team judged coverage speed to be worth more than per-beacon accuracy. The hover-over method is retained as a fallback for individual beacons where a more accurate position is required.

Tasks 2 and 3 additionally require the UAV to carry and release a small weighted payload onto a marked platform [5], [16]. This capability lies outside the team's first-year Core-tier target. The airframe for competition has not yet been procured; development has used a reference quadrotor with a gimbal-stabilised camera. The design is expressed in terms of requirements rather than a specific aircraft: the selected airframe must provide stable hover at the altitude needed to cover the operating area, carry a downward-facing camera of sufficient resolution to resolve beacon colour at that altitude, report altitude and position accurately enough for (2) and (3), and support the command, telemetry and video link described in Section III-H.

H. Communications

Three communication paths are required. The first is the UAV command and control, telemetry and video link, provided by a CubePilot HereLink Air Unit Set V1.1 [17] operating in the 2.4 GHz band across 2409 to 2459 MHz and 2412 to 2462 MHz, with centre frequencies of 2434 and 2437 MHz. The controller uses one directional and one omnidirectional antenna at 5 dBi and 2 dBi respectively, and the air unit carries two supplied antennas. This link has been declared in the competition radio frequency submission under the identifier RP SkySentinel.

The second is the inter-vehicle path carrying beacon positions from the UAV to the surface vessel, which is what makes Advanced-tier collaboration possible. The third is RoboCommand, the competition-provided system through which every vehicle must report status and through which dynamic mission updates are received [3], [18]. RoboCommand is not optional infrastructure for the team: it is a scored cross-cutting requirement and a dependency of Task 4.

The surface and underwater vehicle radio configuration is being finalised and will be submitted as an addendum to the competition frequency declaration once confirmed.

I. *Safety Systems*

1) Kill Switch and Emergency Stop

An external power switch controls the vehicle's power state during deployment and recovery. The emergency stop, described in Section III-D, removes thruster power at source without relying on the Raspberry Pi or ArduSub, and has been demonstrated on the water.

2) Signal Lighting

The vehicle carries a three-colour status tower that indicates its control mode to judges and safety personnel. Yellow indicates manual control, green indicates autonomous operation, and solid red indicates that an emergency stop or emergency halt is in effect. The tower is switched by a relay bank driven from the Navigator's signal pins, and the emergency stop condition is wired into the same bank so that the red indication follows the stop rather than depending on mission software to report it.

3) Leak Detection and Buoyancy

Leak sensors are fitted inside the electronics enclosure and report to dedicated inputs on the Navigator board, giving warning of water ingress before it reaches the electronics. The vehicle is trimmed to be positively buoyant, so that a total loss of propulsion or power results in the vehicle surfacing rather than sinking. Together these measures mean that the most likely serious fault during submerged operation ends with a recoverable vehicle on the surface.

4) Software Failsafes

Three software failsafes operate on the surface vehicle. A try and finally construct ensures that the vehicle is disarmed before the control script exits, so that the vehicle is left in a safe state even if an unexpected error occurs during execution. A keyboard interrupt allows the operator to terminate the script immediately from the operator side, providing a manual means of stopping the vehicle. A mission timeout prevents the vehicle from operating indefinitely if the control script becomes unresponsive, encounters a logic error, or exceeds its expected mission duration; on expiry the vehicle stops and disarms automatically.

5) Aerial Failsafes

The UAV uses the standard ArduPilot failsafe set [8]. Radio failsafe governs behaviour on loss of the control link [19]. Battery failsafe triggers a defined response when the pack falls below configured thresholds [20]. A geofence constrains the aircraft to an approved operating volume [21]. Mission abort is available to the safety pilot either by pausing the mission from the ground control station [22] or by switching the aircraft to Loiter mode from the transmitter [23]. The aircraft must additionally pass an airworthiness inspection and comply with Singapore Civil Aviation Authority regulations before operating at the venue [24].

6) Handling and Mechanical Safety

Tow points and lift points are provided as marked hard points on the vehicle frame, with rope used as the tow and lift harness. All propellers are fitted with shrouds, and all propellers on the vehicle are of the same model, which simplifies inspection and replacement. These provisions are specified against the static safety inspection criteria, which check emergency stop systems, tow and lift points, propeller guards, component security and recovery light compatibility [24].

IV. Testing Strategy

A. *Testing Methodology*

Testing progresses through six stages: simulation, component, subsystem, system, environmental and mission. Testing is conducted progressively to identify problems early and to reduce the risk of introducing unverified systems into full mission testing. A capability is not advanced to the next stage until it has behaved as intended at the current one.

This ordering is a direct consequence of the team's resource constraints. On-water test time is the scarcest resource available to the team, requiring venue access, safety cover and the simultaneous availability of members who are also full-time students. Every fault that can be found in simulation or on the bench is a fault that does not consume on-water time. The team accepts a slower path to its first on-water demonstration so that each on-water session yields more.

B. *Component Testing (Hardware)*

The team conducts component-level testing to evaluate the capabilities and limitations of newly introduced hardware and sensors before integrating them into the complete system. Testing is conducted progressively at the pool, allowing the team to observe the behaviour of individual components under real operating conditions.

Through these tests, the team identifies the operating characteristics of each component, including its performance, reliability, and suitability for the intended mission. The results also provide the team with a better understanding of how individual hardware components can be integrated to support the RobotX mission objectives.

As the team is participating in RobotX for the first time, this approach allows members to gain practical knowledge of the newly implemented hardware while identifying potential applications and limitations early in the development process. Findings from component testing are subsequently used to guide subsystem integration and further development.

C. *Subsystem Testing*

Subsystem testing verifies that groups of components behave correctly together before they are exposed to full mission conditions. The safety chain was tested in this way: the emergency stop switch, relay bank and status tower were

exercised as one subsystem to confirm that asserting the stop removed propulsion power and that the tower changed to red, independently of the state of the autonomy software.

Propulsion and control were similarly tested as a subsystem under remote control before any autonomous behaviour was attempted, establishing that the vehicle responded correctly to commanded motion and that the control allocation across the six thrusters of the development platform was correct.

D. System Testing (Software)

The team uses the Gazebo simulation environment to test and evaluate software before deploying it to the physical vehicle. Simulation provides a controlled environment where software functionality and autonomous behaviours can be evaluated without requiring immediate access to the physical test platform.

Software is first developed and tested within the simulated environment to determine whether the intended functions and behaviours are feasible. This allows the team to identify and resolve software issues before conducting physical testing at the pool, reducing unnecessary testing time and minimising the risk of introducing untested code to the vehicle.

Once the software demonstrates the intended behaviour in simulation, it can be transferred to the physical system for further validation. The results from physical testing are then used to identify differences between simulated and real-world behaviour and guide subsequent software improvements.

This simulation-to-physical testing approach allows the team to progressively validate its software while reducing reliance on physical testing during the early stages of development.

E. Environmental Testing

Environmental testing moves the vehicle from the controlled pool environment into conditions closer to those expected at Marina Bay. Two aspects have been exercised to date. Operation at extended range was tested at a separation of 28 m to examine control link behaviour and the operator's ability to assess vehicle state at distance. Disturbance response was tested by introducing external disruption to the vehicle while it was operating in autonomous mode, in order to observe whether it recovered its commanded track.

Enclosures and connectors are inspected before and after each immersion. The leak sensors described in Section III-I will add an automatic check once their own testing is complete. Further environmental testing in open water, including operation in wind, current and wave conditions representative of the competition venue, is planned as venue access permits; it is also the only setting in which dead-reckoning drift can be characterised over realistic distances.

F. Mission Testing

Mission testing exercises complete task sequences end to end rather than individual functions. Autonomous navigation has been demonstrated with the vehicle executing a commanded route without operator input, with live visual feedback available throughout. Handling operations required by the competition, namely tow and lift, have each been demonstrated as complete operations.

Full mission rehearsals combining aerial beacon reporting with surface navigation have not yet been conducted, as the competition UAV has not been procured. These rehearsals are the team's principal outstanding test activity and are scheduled once the airframe is available. RoboCommand reporting has not yet been exercised against the competition interface; as it is a precondition of every scored run, it is scheduled ahead of the first full rehearsal.

G. Test Results

The team records test outcomes through photographic and video evidence of each test run. Automated data logging of vehicle telemetry is not presently implemented, and no logged flight or run statistics are available; results in Table V are therefore reported as observed outcomes rather than as measured quantities. Introducing onboard telemetry logging is a recognised gap and is identified in Section V as a priority improvement, since it is a prerequisite both for quantitative comparison between simulated and physical behaviour and for characterising dead-reckoning drift.

TABLE V. SUMMARY OF COMPLETED TESTS AND OBSERVED OUTCOMES

Test	Stage	Outcome
Remote control manual handling	Subsystem	Successful
Emergency stop actuation	Subsystem	Successful
Visual feedback, manual mode	Subsystem	Successful
Visual feedback, autonomous mode	System	Successful
Autonomous navigation run	Mission	Successful
Disturbance response, autonomous	Environmental	Recovered
Extended-range operation at 28 m	Environmental	Successful
Tow operation	Mission	Successful
Lift and recovery operation	Mission	Successful
Underwater depth hold (development platform)	System	Successful, early-stage
Underwater 180° and 360° turns	System	Successful

[d] Evidence is retained as photographic and video records; telemetry logging is not yet implemented. Underwater results were obtained on the six-thruster development platform.

H. Design Iterations

The conversion of a remotely operated underwater vehicle into an autonomously controlled platform illustrates the team's iteration process, and is recorded in Table VI. The sequence shows the pattern the team applies generally: establish communication, achieve commanded motion, move execution onboard, then stabilise one control mode at a time.

TABLE VI. DEVELOPMENT RECORD: ROV TO AUV CONVERSION

Date	Activity	Outcome
3 Jul	Establish Raspberry Pi to vehicle communication	Achieved

6 Jul	Manual mode control from Python; verify commanded motion matches vehicle motion	Achieved
7 Jul	Run Python control onboard via vehicle hotspot; build test gate	Achieved
9 Jul	Troubleshoot movement in STABILIZE and MANUAL modes	In progress
15 Jul	Troubleshoot ALT_HOLD depth control	In progress
26 Jul	ALT_HOLD depth control	Successful
27 Jul	180° and 360° turns using onboard sensors	Successful

Two points emerge from this record. First, the decision on 7 July to execute the control script onboard rather than from an external machine removed a dependency on a continuous operator connection, and is the reason the later autonomous runs could be conducted without a tethered laptop. Second, these results were obtained on the six-thruster development platform and are treated as early-stage. Depth control depends on how vertical thrust is distributed and on the vehicle's weight, buoyancy and balance; with four vertical thrusters on the competition UUV, depth-control development and tuning will start again once that vehicle is received, and the confidence assigned to depth hold in Table II has been lowered accordingly.

V. Conclusion

Team Circle enters the 2026 Maritime RobotX Challenge as the first Singaporean polytechnic to compete, with a system of systems built around BlueROV2-based surface and underwater vehicles and an aerial vehicle. The team has demonstrated manual and autonomous surface navigation, live visual feedback in both modes, disturbance recovery, extended-range operation, tow and lift handling, an emergency stop that cuts propulsion power independently of software, mode indication to judges, and early-stage underwater depth and heading control on its development platform.

Several items remain outstanding. The competition UUV, with its eight-thruster layout, has not yet been received, and depth control must be developed and tuned again on it. The competition airframe has not been procured, and with it the aerial beacon localisation and the UAV to USV relay that Task 1's Advanced tier depends on remain undemonstrated. The DVL, inspection camera and sonar required for Task 2 and for obstacle avoidance are still being procured. Automated object detection is in development and RoboCommand reporting has not yet been exercised. Telemetry logging is the team's priority improvement, as its absence currently limits the team to qualitative test evidence. UAV payload delivery, required by Tasks 2 and 3 at the Advanced tier, remains undeveloped.

The system nonetheless reflects the RobotX mission as the team understands it. Safety is architectural rather than retrofitted: the stop is a switch in the power path, the hull is positively buoyant, and leak sensors give warning before ingress becomes loss. The design is modular enough to carry over to the competition UUV, and simple where a more complex option offered no mission benefit. For a debut year, Team Circle regards a smaller set of reliably executed Core-tier tasks, on a platform the next cohort can extend, as the right measure of success.

Acknowledgment

Team Circle would like to thank the Ministry of Defence, Singapore for its support of the team's participation in the 2026 Maritime RobotX Challenge. The team thanks Republic Polytechnic for providing facilities, pool access and test time, and for the institutional support that made a first entry possible.

The team is grateful to Mr Melvin for technical guidance on the aerial beacon localisation concept, and to Mr Goh Yi Ming for review of the vehicle's electrical and software architecture. The team further thanks the staff, alumni and advisors who contributed engineering advice, equipment and supervision during testing.

References

[1] Blue Robotics, “BlueROV2 underwater ROV.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://bluerobotics.com/store/rov/bluerov2/

[2] RoboNation, “Maritime RobotX Challenge 2026 team handbook.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://robonation.gitbook.io/robotx-2026-team-handbook

[3] RoboNation, “3.3.1 Cross-cutting tasks,” in Maritime RobotX Challenge 2026 team handbook. Accessed: Sep. 22, 2026. [Online]. Available: https://robonation.gitbook.io/robotx-2026-team-handbook/section-3-autonomy-challenge/3.3-mission-task-descriptions/3.3.1-cross-cutting-tasks

[4] RoboNation, “3.3.2 Mission task 1: Safe passage,” in Maritime RobotX Challenge 2026 team handbook. Accessed: Sep. 22, 2026. [Online]. Available: https://robonation.gitbook.io/robotx-2026-team-handbook/section-3-autonomy-challenge/3.3-mission-task-descriptions/3.3.2-mission-task-1-safe-passage

[5] RoboNation, “3.3.3 Mission task 2: Infrastructure survey and repair,” in Maritime RobotX Challenge 2026 team handbook. Accessed: Sep. 22, 2026. [Online]. Available: https://robonation.gitbook.io/robotx-2026-team-handbook/section-3-autonomy-challenge/3.3-mission-task-descriptions/3.3.3-mission-task-2-infrastructure-survey-and-repair

[6] A. Lachguar et al., “Minion: Design and competition strategy for the 2024 Maritime RobotX Challenge,” Embry-Riddle Aeronautical University, Daytona Beach, FL, 2024.

[7] Blue Robotics, “Navigator flight controller.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://bluerobotics.com/store/comm-control-power/control/navigator/

[8] ArduPilot Development Team, “ArduSub and ArduPilot documentation.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://ardupilot.org

[9] Blue Robotics, “T200 thruster.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://bluerobotics.com/store/thrusters/t100-t200-thrusters/t200-thruster-r2-rp/

[10] Blue Robotics, “Bar30 high-resolution depth and pressure sensor.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://bluerobotics.com/store/electronics/bar30-sensor-r1/

[11] Blue Robotics, “Low-light HD USB camera.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://bluerobotics.com/store/sensors-cameras/cameras/cam-usb-low-light-r1/

[12] Blue Robotics, “DeepWater Exploration exploreHD USB camera.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://bluerobotics.com/store/sensors-cameras/cameras/deepwater-exploration-explorehd-usb-camera/

[13] Blue Robotics, “Ping360 scanning imaging sonar.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://bluerobotics.com/store/sonars/imaging-sonars/ping360-sonar-r1-rp/

[14] Open Robotics, “Robot Operating System (ROS) documentation.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://docs.ros.org

[15] Open Source Robotics Foundation, “Gazebo simulator.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://gazebosim.org

[16] RoboNation, “3.3.4 Mission task 3: Coordinated logistics,” in Maritime RobotX Challenge 2026 team handbook. Accessed: Sep. 22, 2026. [Online]. Available: https://robonation.gitbook.io/robotx-2026-team-handbook/section-3-autonomy-challenge/3.3-mission-task-descriptions/3.3.4-mission-task-3-coordinated-logistics

[17] CubePilot, “Herelink V1.1 user manual.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://docs.cubepilot.org

[18] RoboNation, “3.3.5 Mission task 4: Dynamic incident response,” in Maritime RobotX Challenge 2026 team handbook. Accessed: Sep. 22, 2026. [Online]. Available: https://robonation.gitbook.io/robotx-2026-team-handbook/section-3-autonomy-challenge/3.3-mission-task-descriptions/3.3.5-mission-task-4-dynamic-incident-response

[19] ArduPilot Development Team, “Radio failsafe.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://ardupilot.org/copter/docs/radio-failsafe.html

[20] ArduPilot Development Team, “Battery failsafe.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://ardupilot.org/copter/docs/failsafe-battery.html

[21] ArduPilot Development Team, “Geofencing.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://ardupilot.org/copter/docs/common-geofencing-landing-page.html

[22] QGroundControl, “Fly view user guide.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://docs.qgroundcontrol.com/master/en/qgc-user-guide/fly_view/fly_view.html

[23] ArduPilot Development Team, “Loiter mode.” Accessed: Sep. 22. Accessed: Sep. 22, 2026. [Online]. Available: https://ardupilot.org/copter/docs/loiter-mode.html

[24] RoboNation, “3.2 Mandatory activities,” in Maritime RobotX Challenge 2026 team handbook. Accessed: Sep. 22, 2026. [Online]. Available: https://robonation.gitbook.io/robotx-2026-team-handbook/section-3-autonomy-challenge/3.2-mandatory-activities

APPENDIX A: COMPONENT LIST

TABLE VII. COMPONENT LIST

Component	Vendor / Model	Custom / Purchased	Reasoning
Platform	Blue Robotics BlueROV2 frame and enclosure	Purchased	Development platform; converted from ROV to autonomous control
Competition UUV	Eight thrusters, four vertical	Pending receipt	Depth control to be developed and tuned on this vehicle
Propulsion	6 x Blue Robotics T200 thruster	Purchased	Common model; interchangeable spares
Battery	4S LiPo, 14.8 V nominal	Purchased	Tentative; under evaluation
Flight controller	Blue Robotics Navigator board	Purchased	Raspberry Pi HAT: IMU, leak inputs, PWM outputs
Onboard computer	Raspberry Pi	Purchased	Runs ArduSub and the mission software
Emergency stop	Manual switch in 14.8 V thruster supply	Custom	Cuts propulsion power independently of software
Kill switch	External power switch	Purchased	Manual power state control
Mode indication	Three-colour status tower	Purchased	Control mode visible to judges
Relay bank	3 x HLS8L-DC3V-S-C, 4-channel relay module	Purchased	Isolates status tower from control and propulsion circuits
Leak sensor	Blue Robotics leak sensor	Purchased	Water ingress warning to Navigator
IMU	Integrated on Navigator board	Purchased	Attitude and heading; no GPS fitted
Pressure sensor	Blue Robotics Bar30	Purchased	Depth and water temperature
Camera	Blue Robotics low-light USB camera	Purchased	Live visual feedback
DVL	Doppler velocity log	Planned	Bounds dead-reckoning drift; required for submerged navigation
Inspection camera	DeepWater Exploration exploreHD USB	Planned	Task 2 pipeline survey imagery
Sonar	Blue Robotics Ping360	Planned	Obstacle detection and structure location
Propeller shrouds	Common design	Custom	Personnel safety
Tow / lift points	Frame hard points, rope harness	Custom	No added mechanism
UAV data link	CubePilot HereLink Air Unit Set V1.1	Purchased	Command, telemetry and HD video
UAV airframe	Selection in progress	Pending	Requirements defined in Section III-G
Simulation	Gazebo	Open source	Pre-deployment software validation
Framework	ROS	Open source	Planned migration for inter-module comms

[e.] Electrical configuration is tentative pending completion of the team's experimentation phase.