SRM ASV - Trishul

Aditya Tandon, Arun Gunda, Aadinath Mishra, Akilan G, Ajith Raj, Sudeep Nigam, Harshan Chellamal,
Shreyas Renukuntla
SRM University, Kattankulathur
Tamil Nadu, India

***Abstract*—This paper details the process of building the hardware and software of Trishul, an autonomous surface vehicle and an autonomous underwater vehicle, designed by SRM ASV, a group of 18 students from SRM University, India. It also discusses hurdles the team has faced during the development process and the modifications made to the team structure to overcome them. The vehicles, Trishul and Dolphin, have been built for the purpose of participating in AUVSI Foundation's 9th International Roboboat competition at Virginia Beach, USA. Being the first time SRM ASV is participating in the competition, emphasis has been placed on the completion of hardware and the integration of software for the construction of a stable platform capable of supporting further research and development.**

I. INTRODUCTION

SRM ASV is a research group from SRM University and is currently the only team from India participating in the AUVSI Foundation's 9th International Roboboat Competition. The team currently consists of 18 members from various engineering disciplines, working under five domains - Mechanical, Electronics, Coding and Integration, Computer Vision and Corporate Relations. Established in April of 2014, the team has grown from a 5-member group, largely self-funded to being sponsored by multiple companies and becoming the official representative from SRM University for the competition.

The team is participating with the primary intent being to produce a working platform that can be modified and developed with ease.

To achieve this, the team has built Trishul, the third and final iteration of a hull that has had a year of proto-typing and design changes. For the interoperability challenge, the team has built an under-water vehicle capable of deployment from Trishul. The under-water vehicle was designed and fabricated with reference to existing ROV technology and takes primary inspiration from BlueRobotics' BlueROV.

The content of the following paper is ordered with loose emphasis on domains, and greater importance given to technical challenges that were faced and overcome.

II. DESIGN STRATEGY

When begun in April 2014, the team operated with emphasis given to functioning within the domains, sans a view of the state of progress overall. This voluntary tunnel-vision soon led to a mismatch in work timelines that quickly led the team members to abandon working in separate groups and work together. *Domain* soon became a convenient word to recruit people with broad interests.

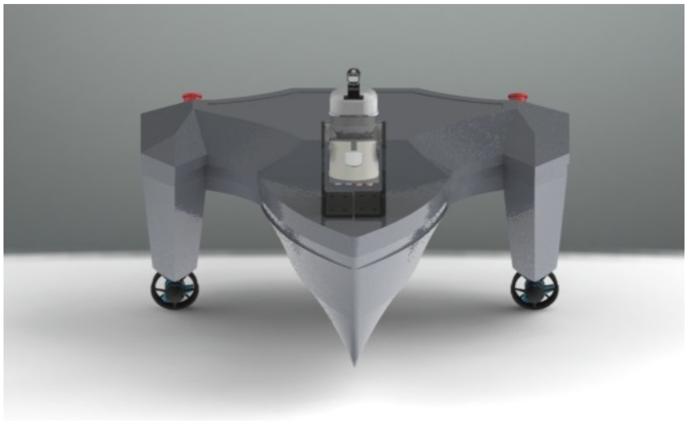


Fig. 1: Render of Trishul

Design became a collaborative process, with constant inputs from each domain to every other. A near-instant improvement with this new mode of project management were the insights into the working of other domains. Knowledge-sharing within the team streamlined the flow of work and offered a chance to move outside one's department and experiment with unfamiliar technology.

The team's design strategy varied in the first few months, until it eventually coalesced to its present form. The final modus operandi can be distilled to the following series of steps.

The first step involved the domain heads deciding the type of platform that is optimal for testing. *Platform* refers to the configuration of the boat, i.e., mono-hull, or multi-hull. On deciding the configuration, the type of hull is chosen. As sensors require a stable platform with least amount of noise, displacement hulls were chosen to be the main class of hulls to be used. Planing hulls usually introduce noise when the hull rises above the water and when it settles back down.

The coding and electronics domain decide the number and type of sensors that go on board. The power supply for these sensors and the control electronics would be decided by the electronics domain and their dimensions estimated. These dimensions are used by the mechanical team to decide placement and get an approximate idea on the dimensions of the hull. Placement of the sensors is decided based on separate requirements from the electronics team.

Simultaneously, the mechanical team chooses the material to be used to construct the hull. Once the initial requirements are finalized, the hull is modelled using SolidWorks. The model is tested virtually using ANSYS Workbench's CFX and Static Structural modules for drag and structural rigidity. The electronics boards are prototyped on Proteus or Multisim and tested before soldering is begun.

Trishul is a platform that was originally designed with the capability of deploying a UAV (Unmanned Aerial Vehicle). When the 2016 Primer was released with the replacement of a UAV with an AUV, it was deemed impractical to construct a new hull from scratch because of logistical issues and time constraints. It was decided that the same hull would be used for deploying the AUV. A drawback that the team decided to manage with was a higher waterline (because of the unaccounted AUV weight) and a pitching imbalance (eccentric loading of the AUV's weight).

During the testing process of the trimaran proto-type (henceforth referred to as Lakshya), the team found that the low streamlining of the hull counteracted the high thrust provided by the two BlueRobotics T200 thrusters. The addition of a third thruster and the introduction of streamlining curves in the structure of Trishul was due to the lessons learnt in the tests conducted on Lakshya. This lesson, though, was not without its own drawbacks. The volume of the hull decreases with the angle of attack, and this affects the placement of components inside the hull as well as the waterline. Wiring for the electronics grows tangled as the working space decreases and troubleshooting grows harder as access to the wiring grows more cramped. These potential difficulties were predicted and were accounted for during the design of Trishul.

III. VEHICLE DESIGN

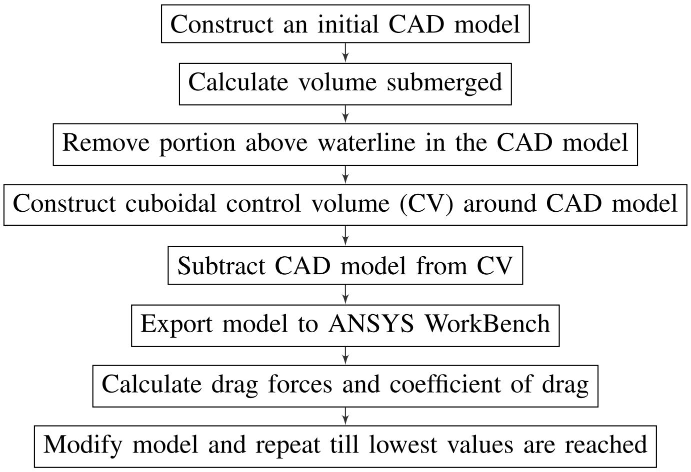
This section is divided into two major subsections: Mechanical and Electronics. The Mechanical subsection details the design and fabrication of the ASV's outer structure and the mounting of the electronics. It also discusses the process of building the AUV. The Electronics subsection consists of PCB fabrication and the wiring circuitry within the ASV and AUV. It also discusses the processor used and the functions of the lower level microcontrollers.

A. Mechanical

The foremost concern for the mechanical domain was the design of an efficient hull structure that had the capability to navigate through the water with least amount of energy expended but also have the strength to handle the stresses a naval vessel is usually subjected to. Furthermore, the domain was also tasked with the design and fabrication of an AUV and an efficient deployment mechanism.

1) Hull Design and Construction: : The hull was designed using the SolidWorks 2013 CAD modelling software and analysed using ANSYS Workbench. The algorithm used to extract comparison data from each iteration of the hull is represented in the following flowchart.

The final results extracted from the analysis include the overall drag force (F_d) experienced by the hull and the coefficient of drag (C_d).



The drag equation is as follows:

$$C_d = \frac{2F_d}{\rho A v^2} \tag{1}$$

In Equation 1, C_d represents the drag coefficient, F_d is drag force, ρ represents the density of fluid, A the wetted area and v

The C_d is a factor of the wetted area and therefore can only be used when comparing models that are very similar to each other, i.e., during fine optimization. F_d can be used for both coarse and fine optimization.

This model of analysis relies on the assumption that the waterline is parallel to the top of the hull, i.e., there is no pitching imbalance. This assumption speeds up the analysis process and provides an element of constancy in all the models. A tradeoff is that the theoretical value might be different from the actual value if the ASV's orientation is pitched differently in the water.

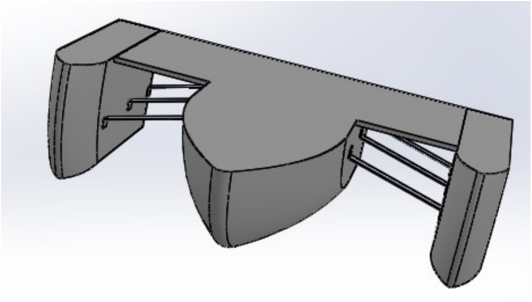


Fig. 2: Initial hull

Figure 2 shows view of the initial hull design. It was made using a center-to-extreme modelling strategy, i.e., the central plane's profile is lofted to the extreme plane's profile. This provides an acceptable amount of control as the lofting curve and the plane profiles can be lofted. This profile proved to have the maximum amount of drag and is highly inefficient, wasting most of the thruster's energy to push water aside.

Figure 3 shows a CAD rendering of Trishul's hull. This was modelled using a top-down approach, i.e., the boat is split into planes from the deck to the keel. Each plane contains a sectional sketch which are lofted individually. This approach

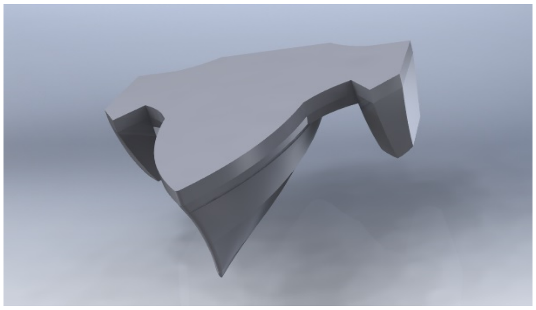


Fig. 3: Trishul's hull

is labour-intensive, but it provides more control over the hull shape.

To provide more clarity in the design iteration process, drag values calculated for various iterations of the hull are shown in Table I. The analysis is conducted under the following conditions:

- Weight of hull = $20kgs$
- Velocity of flow (inlet and outlet) = $3m/s$
- Liquid used = Water at $298K$
- Density of liquid = 997 kg/m^3
- Turbulence = 5%

TABLE I: Analysis values

Iteration	Drag Force [N]	Wetted Area [m^2]	Drag Coefficient
1	169.048	0.540956	0.069653
2	109.765	0.509745	0.047996
3	83.5282	0.51111	0.036426
4	74.4919	0.49852	0.0349263

The first iteration contains the values extracted from the analysis of the initial hull design represented in Figure 2. The subsequent iterations represent modifications done to the final hull represented in Figure 3 with the last iteration being that of the final hull. The wetted areas for the above iterations fall very close to each other. This condition allows us to compare them using the calculated drag coefficients. It is evident that there is a marked decrease in the drag coefficient from the initial to the final hull. The introduction of complex curves into the design eliminated 56% of the drag force exerted on the hull for the same wetted area. Note that although the area might be comparable, the waterlines are not. The waterline for the final hull is higher than that of the initial hull due to the streamlining.

The bow wave generated by the boat leaches a minimal amount of energy from Trishul's momentum. This is primarily due to the passage provided from bow to stern and the surface finish of the hull. The trade-offs for this degree of streamlining is the higher waterline, and the tendency of the boat to pitch forward. The primary reason for positioning the middle-hull thruster closer to the bow is to counteract this forward pitch. The bow is also designed so that water is pushed down at higher thrust; this tends to lift the bow up out of the water during high thrust manouevers. A downside to this is that when the boat runs into choppy water or travels upwind, there is a possibility of water spraying on to the deck of the ASV. Although the ASV is itself waterproof, the deck is best left unwetted as the external sensors are only splash resistant.

An important lesson learnt from Lakshya (trimaran prototype), was the importance and the difficulty of waterproofing. Leaks were continuously encountered due to Lakshya's construction material and methodology; it was built with wooden planks stuck to a central keel and had numerous gaps between each plank that was later filled using wood filler. Waterproofing the gaps proved to be a near insurmountable challenge, and multiple tests were conducted just to check if the hull's integrity was breached. It was decided very early that Trishul would be built with the least probability of water entering, both through material seepage and through interface holes. Each thruster has two interface holes for integration with the power and signal electronics inside and 2 mounting holes. On sealing with silicon sealant (on Lakshya), these holes would start to leak after a few hours when the sealant separates from the uneven surface of the plywood.

Fiberglass was the material of choice to build Trishul for multiple reasons, foremost being the ease with which it can be made to follow complex curves. It's strength and its light weight make it the perfect engineering material for boat fabrication. As it is an impermeable continuous surface, it negates the problem that the team had faced when working with Lakshya.

The hull was designed with flat surfaces provided for the thrusters to be mounted and for through-hull wiring. This was made to ensure that the thrusters could be fixed rigidly at an orientation that is precisely parallel to the central axis of the hull. For the interface holes, cable penetrators were custom-made for each thruster cable (power and signal). With the incorporation of O-rings at the penetrator-fiberglass juncture, an effective seal was made for the through-hull wiring.

Fig. 4: Fabricated hull and top

To make the hull, the mechanical domain made a foam board pattern. Patterning was a labour-intensive process, with multiple implosions of the foam board due to the extreme bending stress it was subjected to. The pattern was overlaid with wall putty and sanded down to a smooth finish. Glass fiber with a mat density of 450 kg/m^3 was used to lay down a mold on the pattern. The pattern was removed and the mold was polished to smoothness again. Two iterations were made of the final hull among which the best one was chosen for commissioning. Table II contains the final specifications of the ASV with the top and access cover affixed to the hull.

TABLE II: ASV Specifications

Length(mm)	1192
Breadth(mm)	847
Height(mm)	385
Weight(kgs)	12.65
Waterline when fully loaded(mm)	242

2) AUV design and fabrication: The design phase of the AUV was not as time-consuming or comprehensive as the ASV. Due to time and funding constraints, proto-typing could not be done to test a proof-of-concept. The presence of extensive online literature and numerous commercial products made it easier for the team to adapt an existing model for competition purposes.

The BlueROV retailed by BlueRobotics was the primary source of inspiration for the team's AUV. The main construction concept that was extracted and used by the team was the usage of threaded rods. Threaded rods can be used to hold together a frame through compressive forces exerted by tightening nuts on both ends. This concept was also used for the construction of the external sensor pod.

It was decided even before the design phase that the AUV would be built using acrylic. This was due to the ease and low cost of machining. The size of the AUV is constrained by the size of the compartment required to contain the electronics. A cylindrical capsule was chosen due its high strength and commercial availability. The dimensions of this capsule was selected by finalizing the electronics required for power and control, as well as communication with the ASV. A model was created to simulate the electronics, and was used to decide the smallest dimensions required to contain them.

The two side panels were then designed so that they could contain the acrylic capsule within their dimensions and also be strong enough to handle the weight of all the electronics, the pressure from the threaded rods and forces exerted underwater. The rest of the acrylic components consist of mounts for the capsule, mounts for the camera and thruster mounts.

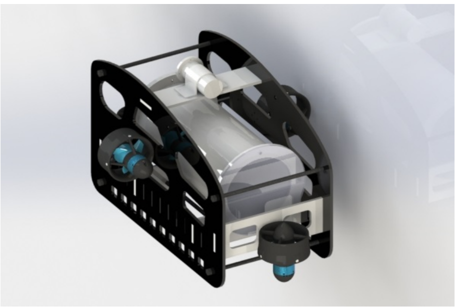


Fig. 5: Render of the AUV

The minimum number of thrusters needed to manouever the AUV is four. A torpedo-like AUV can function with one thruster and multiple control surfaces, but for a slow-moving vehicle it is not practical to use such a configuration. Also, if the center of gravity were precisely known, three thrusters can also be used, but even this is not practical as such precision can't be attained with the available resources.

3) Mandatory Water-based Tasks : The design of Trishul was made so as to exploit the thrust provided by the T200 thrusters. The thrusters used in Seagull(catamaran prototype) were made in-house, with off-the-shelf propellers coupled to a high-torque 500 RPM DC motor. There were multiple operation failures with these thrusters due to water-seepage in the PVC-pipe housing and coupling slippage. Permanently sealing them inside the housing was the only solution that worked satisfactorily. The thrusters used in Lakshya were BlueRobotics T200 thrusters. These proved to be significantly better when compared to the previous thrusters in terms of control and thrust provided. The speeds recorded in the two prototypes and Trishul are tabulated in Table 1. Speeds were calculated by measuring the time required to traverse a predetermined distance (10m).

TABLE III: Velocity measurement (± 0.2m/s)

Voltage (m/s)	Seagull (m/s)	Lakshya (m/s)	Trishul (m/s)
12	0.3	0.9	1.5
16	-	1.2	2.5

Trishul has a configuration of three thrusters facing forward and affixed to each hull. Thrust was measured at 12V and 16V (optimal voltage for maximum thrust) for one, two and thrusters to tabulate the thrust provided and to decide the number of thrusters that would be installed for the competition (two or three). Table IV contains the thrust measurements for each configuration of thrusters. Thrust was measured using a strain gauge attached to eye-bolts fixed to the transom of the hull.

TABLE IV: Thrust measurement (± 0.3kgf)

Voltage (V)	1 thruster (kgf)	2 thrusters (kgf)	3 thrusters (kgf)
12	3.2	6.2	9.2
16	4.4	8.7	13

The final decision regarding the number of thrusters was made favouring three thrusters for the following reasons:

- To oppose forward pitch
- To achieve greater thrust for the thrust test
- To traverse the speed gates in lesser time
- To consume lesser power when slow movement is desired (only central thruster functions when the boat needs to move at low speeds, with side thrusters controlling direction)

The weight of Trishul in water is contributed to significantly because of the addition of the AUV. Further, due to excess resin being added in the fabrication process of the ASV, the weight of the hull itself is higher than optimal. Sensor and electronics mounts have been made using light-weight acrylic and plywood to keep weight to a minimum. Also, following the primary philosophy of construction, every component is fixed to its mount using fasteners so that each part is modular and can be removed easily for replacement, repair or servicing. The final weight of the ASV is measured to be 38kgs±0.2kgs.

B. Electronics

The core philosophy that was followed during the design and construction of Trishul's electrical hardware was modularity which enabled a plug-and-play configuration. The two prototypes, Seagull and Lakshya, were planned without ease-of-troubleshooting as a characteristic. This led to greater troubleshooting time in preparing the boat for testing whenever there was a failure.

This problem was solved in Trishul by incorporating discrete modules in the electronics compartment. A vertical mounting system (VMS) was developed with the vertical card guides to improve the stability of the mount so that PCBs could be removed from the circuit without affecting the mechanical integrity for the rest of the electronics.

1) PCB design: All the PCBs onboard have gone through multiple iterations of circuit design and testing, both virtual as well as hardware prototyping. The virtual prototyping was done on Multisim and Proteus. Eagle CAD was used for virtually proofing each board before soldering onto a PCB. The PCBs were completely designed on Eagle CAD and were also exported to SketchUp to create a 3D model. These were then used in SolidWorks to decide the optimal placement of each board to consume the least space and also for heat dissipation purposes.

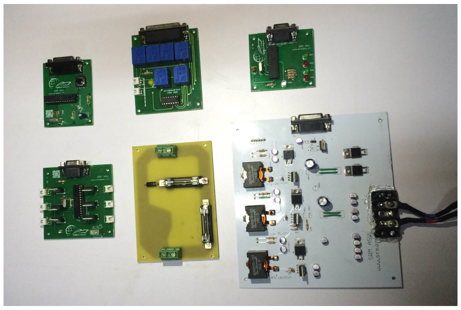


Fig. 6: Printed Circuit Boards

In total there are seven different PCBs on Trishul. The functions of these boards are itemized below:

- Regulation Board - The Regulation board essentially consists of buck converters which are responsible for providing +3.3, +5 and +12 voltage levels, in addition the board also provides a variable output voltage which can be set using an onboard potentiometer. Due to their inherent nature the TI buck converters have a efficiency level of more than 90% which enables the judicious use of the onboard power supply. One disadvantage of using these regulators is the necessity of inductors. The inductors generate an EM field that are strong enough to disrupt the operation of delicate components like the digital compass and GPS. This causes the placement of components to change, with the Buck kept at a distance of at least 35cm from the sensors.

  The Buck draws its input from Trishul's 14.8V battery array. It can function with an input range of 12-24V.

- Display Parameters - Onboard diagnostics is achieved using the display parameter board. Presently, two sets of values are displayed on a 16x2(characters) LCD screen. The screen is mounted under the access port cover and shows the current consumption by each thruster as well as the temperature of the internal volume of the ASV. Local temperatures can also be monitored in sensitive areas (Motherboard, Buck Distribution) if required. A thermocouple is used for temperature sensing, and the placement of this component decides the temperature being displayed.
  A Hall-effect current sensor module is used to monitor real-time individual thruster consumptions.

- Cooling System - The primary reason for the inclusion of this board is due to the temperature sensitive nature of the onboard electronics and sensors. Two types of cooling were considered, water- and air-cooling. In terms of cooling power, water-cooling is more localized than air-cooling. Separate cooling modules need to be deployed for effective temperate control, i.e., each sensor requires a customised cooling surface. In lieu of this, air-cooling was used to exhaust the heat generated through vents. The output from the thermocouple is used to drive a proportional control based system. The speed of the inlet and outlet fans are PWM controlled to maintain the temperature at an optimum level.
  In the interest of symmetry, two inlets and two outlets were installed. Inlets are placed in the front, and outlets at the back. The vents are assembled using PVC connectors, and are orientated to prevent water from entering, both from splashes and rain.
  Each vent has a 12V Brushless DC-motor PC fan to suck-in or exhaust air. A fifth fan is positioned next to the Buck Distribution board to encourage airflow over the inductors. The sixth fan is powered by the motherboard for cooling the processor. The motherboard is positioned under the exhaust outlets to ensures that there is no heat build-up over the processor.

- Switching Interface Board - For safety reasons it is required that the ASV have the ability to instantly switch from autonomous to manual control. This is achieved by the switching logic board. On prompting from the RC, the signals going to the thruster are shifted from processor control to RC control. The switching board consists of LEDs which are provided onboard so that the current status of the system can be monitored in real-time.

- E-kill switch - This board is the ultimate precaution given to Trishul in case of a malfunction. The power lines for each thruster run through a 12V-65A electro-mechanical relay, and these relays are controlled through the E-kill switch, which is connected to the RC. In case of an emergency where the boat has to be stopped immediately, the RC can activate the relays and open the power lines.

- Interface board - The Switching and E-kill boards can not interface directly with the RC. The interface board functions as an intermediary, and converts the analog signal from the RC to a digital signal. These signals can be sent to their respective boards. The interface board was made as a separate module due to its importance in the precautionary circuitry.

- Protection board - Trishul's thrusters are high consumers of power; they are also very susceptible to power surges and do not have any native electrical protection.The protection board isolates the thruster from the batteries and ensures that power surges or drops do not affect the thruster operation.
  The traces use 2 ounces of copper and are rated for 40A. The PCB design used for Trishul is the second iteration of the Protection Board. The first iteration used a H-bridge to ensure that even if the battery were incorrectly connected, the thrusters would still function. The H-bridge was eliminated in the final circuit as the diodes caused a 0.2V voltage drop each.
  There are two types of protection offered by the Protection Board. Reverse-polarity protection is ensured using a diode, and over-current protection using a 35A fuse.

2) Microcontroller: Trishul uses two types of low-level microcontrollers for data acquisition, thruster control and logic-circuits.

- ATmega 328P - The ATmega chip is responsible for the Display Parameter, Switching Interface and Cooling system PCBs. They are bootloaded with the Arduino firmware, for a easy programmable interface using the Arduino IDE and the FTDI module. This enables the ATmega chip to be a standalone system which can handle all the low-level operations.

- Cypress PSoC 4 - It is an ARM Cortex-M0-based, PSoC device that integrates programmable analog front ends and programmable digital logic. The PWM values for each thruster is sent in a custom UART string format, which is parsed efficiently through the PSoC with abbreviated processing time to overcome delays in the execution of the chain of commands.The PSoC also serves as the Primary Thruster Controller with a 16 bit resolution and provides a larger range of control values for the thrusters. This enables a much finer control over the ASV.

3) Central Processing Unit: The onboard system is hosted on an Intel i5 processor which was salvaged from old computer hardware. The processor utilizes a Gigabyte micro-ATX motherboard and uses both USB 2.0 and USB 3.0 ports to interface with the various sensors onboard. The motherboard is also equipped with a Dual band WLAN PCI-Express card to facilitate seamless wireless communication at 5GHz. The motherboard has a dedicated M4-ATX 250 watts Power Supply Unit(PSU) which is powered by two independent 3S 6Ah batteries. This ensures that the main system's endurance is never compromised.

IV. SOFTWARE ARCHITECTURE

The Software architecture uses a cross platform architecture designed to support parallel asynchronous data acquisition from various sensors to estimate the POSE of the ASV at every point during the mission. The software has been designed such that it is event driven. The main event/task list is stored in the Mission Planner which is used as a repository for the complete mission. The Software also involves a Mission Tracker which is used to track the current task that is being performed and this information is used to correlate information from the Mission Planner. The Mission Planner serves as a repository of Mission related information whereas the Mission Tracker functions as the central brain of the software assigning and completing mission tasks.

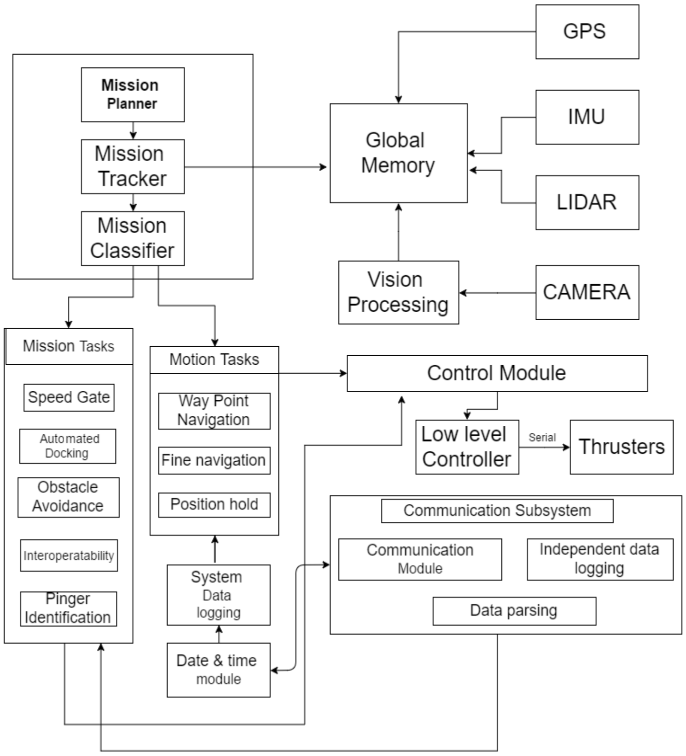


Fig. 7: System Architecture

The complete system has been designed in a way that allows for a global common memory for accessing system variables and all the other modules are run on independent threads to facilitate simultaneous operations. Once the task to be performed has been determined from the mission tracker, the system first assesses what kind of task has to be performed mission or motion. If a motion task has to be performed, the software will then move to realise what sort of movement is required Waypoint Navigation, Fine Navigation or Position Hold. By default if a motion task isnt required, the software will assume Position Hold to be engaged. In case a mission task is required, the software will then call the respective module allocated for every individual task. During the run of the complete software, the communications module is also engaged and is used to send a periodic heartbeat signal to the server. The Communications module is also responsible for reporting all mission data to the server.

The complete system is made to run as an executable on a windows platform and was written with a combination of C and MATLAB code to increase reliability and lower system run time. The most important benefit of choosing such system architecture is that it is modular and scalable. Every time a new module is written or an existing module is modified, the complete code of the system doesnt need to be changed. We have realised that as a system is scaled, modularity is one of the most important features of the software to ensure ease of development and reliability.

A. *Data Acquisition*

The Data Acquisition system has been made to facilitate simultaneous acquisition of data from multiple sensors. All of the sensor data is first put through a base level of processing to parse data and feed only required variables to the primary controller. This allows for a distributed system saving the central processors memory and processing power, making the complete process run time as low as possible. This low level controller then transfers data to the main processor over a serial interface which is then logged into the global memory of the system. After every iteration the system refreshes the global memory values obtained from the sensors. The system acquires data from the IMU to obtain heading information, GPS for getting latitude and longitude data as well as the LIDAR to get environment data.

B. *Navigation and Control System*

The System utilises a PID based Control System to allow for effective manoeuvring and control of the ASV based on the environment it is being subjected to and the tasks it is currently performing. Based on the input received from the Mission Classifier, the motion task is defined which will then utilise data read from the sensors and pass this environment data to the trajectory planner module. This module also works with the Computer Vision system to create a Vector Field Histogram to understand the exact location of the different obstacles that could be present at given points in the environment. This trajectory planner module utilises data from the sensors to estimate the amount of manoeuvring required and then chalks out a trajectory to be followed. This trajectory is then fed to the control system which is responsible for converting this trajectory to control signals for the thrusters which will be used to modify thrust at different intervals from different thrusters.

C. *Communications Sub-system*

The Communication subsystem is an integral part of the complete software architecture and allows for accessing mission data from the server as well as reporting mission data to the server. The Communication sub-system consists of three parts Communications module, Data Parsing module and an Independent Data Logging Module for the communication sub system. The Communications module is responsible for sending the HTTP GET and POST requests as well as Image Upload to the server. This module is also responsible for verifying the SSL encryption certificate in the case of HTTPS requests. The Communications module is also responsible for the heartbeat signal which is sent out periodically over the course of operation of the ASV. All Response data obtained from the server after the GET and POST requests is handled by the Data Parsing Module which is responsible for parsing JSON data from the response. This data is then fed to the mission task during operation. The Data Logging Module present inside the Communications sub system is critical as it enables easier debugging of the communication code by decoupling communication log data from the main system log. This subsystem works in conjunction with the mission tasks module as well as the system date and time module.

D. *Data-Logging*

The data-logging module allows the storage of all data received by the central processor from the peripheral sensor units. This information is used for analysis of the functionality of the calculation and control algorithms being executed. The data being recorded include position data (latitude, longitude) from the GPS and compass data from the IMU. The System data being logged is also all of the data that is written to the actuators after processing that takes place at the end of ever y subsystem or individual module. Analysis of all of the data that was received as well as what corresponding data was written to the actuators aids the process of troubleshooting the system to achieve the required response.

E. *Computer Vision*

The competition places heavy emphasis on detecting multiple objects, multicolored shapes and numeric figures in a highly dynamic environment when it comes to lighting conditions. To face these challenges the team has focused on developing a robust computer vision stack using MATLAB. The algorithms are designed not only to meet the objectives but also to reduce the computational burden as much as possible. Since, most of vision based tasks involve similar pre-processing operations such as normalization for color constancy, special attention has been given towards modularity of the software.

- Blob Detection: Due to varying light conditions the color consistency is compromised, in order to reduce the effect of due dynamic lighting conditions the input image from the camera is first converted to HSV color space where it becomes a function of Hue, Saturation and Value metrics. The object detection requires segmentation of the image in order to distinguish between the object of interest and the background. To achieve this, thresholding operation is performed on the image which rejects the pixel intensities not falling in the range pre-defined. Later the image is obtained in a binarized (Black and white) where desired pixel values are represented by 1s and the remaining are represented as 0s. Hence, a number of Blobs are obtained which are then labeled and then sorted in order to obtain only the desired blob/blobs for further processing and analysis. Once these blobs are obtained they are then analyzed to acquire certain metrics that are then stored for later purposes. The threshold values were obtained after several experiments arranged in the lab.

- Shape Detection The competition requires shapes i.e. triangle, circle and cruciform of different colors (Red, Green, Blue and Black) to be identified prior to docking. The algorithm for shape detection performs somewhat similar operation of blob detection followed by Edge detection. The Contours are then looked for and compared to the ground truth value in order to identify the desired shape.

V. EXPERIMENTAL RESULTS

The results obtained from each iteration of the module include the outputs from the respective function of the module, and the time required to execute the program code. On achieving functionality, extensive optimization was made to the code to decrease runtime and memory used.

Table V contains the times recorded after each iteration of the code for the respective module.

TABLE V: Program Run-Times

Module	I (s)	II (s)	III (s)	IV (s)
Heartbeat	2.2	1.2	1.1	0.8
Communication	5.8	5.7	4.9	4.9
IMU DAQ	1.3	1.2	1	0.2
GPS DAQ	0.2	1	-	-
Computer Vision	62	45	42	35

The above results show a general decrease in runtime after subsequent code optimization. The code was modified using the methods briefly explained below:

- For the heartbeat module, optimization included the elimination of an initial one second delay and the removal of datalogging at every step. This was replaced by consolidated data-logging after every request.
- The Communication module was optimized with methods similar to the heartbeat module. Further optimization was achieved by creating discrete functions for snippets of code that were repeatable in nature throughout the scope of the module.
- Data acquisition from the IMU was initially achieved by opening and closing the COM port repeatedly. Although this ensured that the code was smaller in size, the runtime was relatively high. The final runtime of 0.2 seconds was achieved after the COM port was opened once, and data was continuously read. The port was closed only after boat's run was completed. This functionality was achieved by changing the complete layout of the softwares memory architecture which involves updation of global variables using read data, which is refreshed periodically.
- For GPS DAQ, the initial baud rate was 38400bps, but this led to buffer overflow and eventually caused the system to lag and lose reliability. The baud rate was switched to 4800bps and an update frequency of 1Hz, allowing the system to efficiently process incoming GPS data. Though final DAQ time is higher than the initial iteration, the system shows greater reliability and stability.

VI. ACKNOWLEDGEMENTS

SRM ASV would like to acknowledge SRM University for supporting the research technically as well as financially. We thank our faculty from SRM University and at the National Institute of Ocean Technology, India who provided insight and expertise that greatly assisted the research. We thank Mr. K. Sivanathan, Faculty Advisor - SRMASV for assisting us with all of his experience in managing technical projects. We would also like to express our gratitude to him for his valuable insights in the design of the control system of the ASV.

We would also like to show our gratitude to Sparton, Pulse batteries, National Instruments, TP-Link, Mathworks, Solidworks, IDS GmbH, Cypress, Pololu, Robotshop, Coilcraft and Helidirect for sharing their technology and expertise with us during the course of this research and aiding our team with resources at par with the industry. We are also immensely grateful to Dr. G. Murali, HOD-Mechatronics & Dr. C. Muthamizhchelvan, Director E&T-SRM University for their continual support, making our dream of developing an ASV and participating in the International RoboBoat Competition a reality.