Technical Design Report of Agastya-1, Remote Operated Underwater Vehicle

Akshar Chauhan, Nisarg Panchal, Ishika Saijwal, Krunal Bhatt, Kathan Patel,
Masoom Lalani, Sanskruti Gondaliya, Akshat Patel
Faculty Advisors: Dr. Mihir Chauhan and Dr. Akash Mecwan

Abstract

Agastya is an Autonomous Underwater Vehicle (AUV) and/or Remote Operated Vehicle (ROV) being developed at the Institute of Technology, Nirma University, Ahmedabad by the students of the Institute. It is developed for various competitions of national and international levels. This report mainly focuses on the design of ROV developed for a national level competition AMUROV. The ROV is featured with various mechanisms that enable it to complete various tasks of the competition.

Introduction

Underwater ROV stands for Remote Operated Vehicle that is designed to work underwater and is controlled remotely by an operator. It is connected to the remote operator through a series of cables for transmission of data. It can be deployed underwater in place of human divers to carry out some specific tasks. It is equipped with a variety of sensors like cameras, lights sensors, SONAR, etc and may have one or many manipulators to grab things, cut the objects or lift the stuff. It can be used in shipwreck inspection, ship hull inspection, deep-sea study, ocean exploration, oil rigs, and oil spills to reach areas where humans' reach is difficult.

For the development of the ROV which can complete the competition tasks, an interdisciplinary team of students is formed which is mainly structured into three divisions like Mechanical, Software, and Electrical. The mechanical division is responsible for hull design, waterproofing, gripper design, and fabrication of the structure. The software division is responsible for developing software such as GUI interface, ROS programming, establishing communication, gaining remote access to onboard computers, and underwater image enhancement. While the electrical division looks after power delivery to each component, PCB designing, and development of the kill switch. All the divisions have their assigned tasks but still they are integrated and united in one team with one goal.

I. Competition Strategy

The vehicle is developed for AMUROV competition to be held at Aligrah Muslim University, UP, India. There are majorly three tasks in the competition - surface maneuvering, vision control and depth control. However the tasks are further categorised into various sub tasks like surface manoeuvring, self-stability and depth control with PID, key mapping for movement, establishing communication between the onboard computer and remote PC, integration of sensors, programming of microcontrollers, etc. Later on, all these tasks were integrated such that the ROV can be operated to perform various tasks of the competition as mentioned below:

Task 1 is about surface manoeuvring through a hexagonal path of 80 cm width without touching the sides of the hexagonal path.

Task 2 is about vision and control where a specific shape is assigned to each team and the team has to locate a balloon under the flag of the assigned shape and burst it after passing through the gate.

Task 3 aims to find and acquire the balls in the target zone which is formed by placing 20 red balls at the surface of the arena, and a golden ball submerged in water near the aim zone. The ROV should first pass through gate 1 and then proceed to the second gate, which is placed approximately 100 cm away from gate 1. The lower half area is blocked, thus ROV has to pass through the upper half area of gate 2. Then it has to gather as many floating red balls as possible. Bonus points will be awarded for picking up a golden ball submerged in water.

Strategy plays an important role for success in any competition. The approach for the competition is to complete all the tasks sequentially in the best time possible. Given the sequence, it is presumed that the difficulty of the tasks to be the same as their order. First, the hexagon task is practiced and tried to complete it in the best time with the least errors. For the gate task, a much smaller gate is used for the practice than the actual dimensions mentioned in the rule book. And in the end, the task of acquiring balls is worked out.

II. Vehicle Design

A. Mechanical Sub-system:

The main outcome expected from the mechanical team is to design a hydrodynamic vehicle that is compact, stable, modular, and can perform all the required tasks. To accomplish all the three tasks of the competition, the ROV should have five independent motions, i.e. three linear motions along the principal axis as well two rotational motions i.e. pitch and yaw. So it needs a minimum 5 Degrees of Freedom(DOF). To achieve this, 8 thrusters are used and arranged such that 6 DOF is achieved. The vehicle is also equipped with a gripping mechanism. The design is made such that the components can be added later, as per the requirement of the competition. The vehicle design is bifurcated into three sections: Hulls, Frames, and end effectors. The CAD model of the ROV is shown in Figure-1.

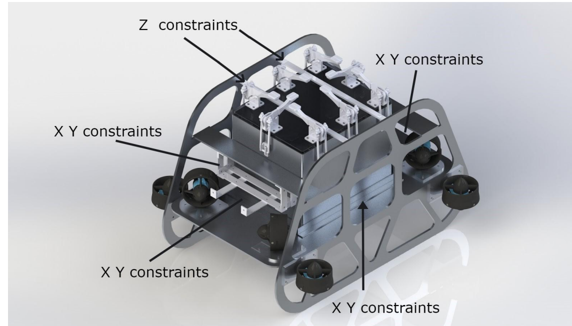

Figure-1: CAD model of the ROV (Agastya-1)

Hulls

Hulls are the storage component where all required components are kept. In case of unmanned underwater vehicles mostly these components are electrical and electronics. Where in case of Manned underwater vehicles (like a submarine) the hull refers to the part where people reside or keep their necessary requirements.

a. **Main Hull:** The main hull of the vehicle is the housing for the most of electrical and electronics components. It is fabricated from the aluminium 6 series sheet that helps in the dissipation of heat effectively. The hull is in the basic shape of a cuboid with space optimization. This hull is fixed with the body with the help of aluminium L-Sections on the middle surface plate. This provides the X and Y direction motion constraints. While in the z-direction, two separate rods are mounted as shown in Figure - 1.
b. **Camera Hull:** There are two camera hulls, one for the front and the other for the bottom side. Both the camera hulls are made up of transparent glass and have a waterproof attached lid. These lids are removable and hence whenever needed the cameras can be removed and/ or replaced. The front camera hull has a larger volume to compensate for the wide-angle view camera. The camera hull at the bottom also follows the same design and has an HD webcam.

Frame

The frame is also fabricated from the same material as that of the main hull i.e. aluminium 6 series sheet. The frame is made up of three

segments (Two side plates and one middle plate) made of aluminium sheets welded together by maintaining the perpendicularity. The side plates have been designed in Solidworks and thoroughly tested for hydrodynamics in Ansys fluent flow work environment.
The side plates are having cut sections mainly for two reasons, one for the outlet flow trajectory from the thruster oriented at different positions and angle, the other reason is to reduce the weight and also make the vehicle have a near to neutral buoyancy point at 2m depth of the center of gravity inside water.

Thruster Orientation
8 thrusters have been mounted on Agastya to provide 6 degrees of freedom. The frame has 2 thrusters attached on each of the side plates facing the downside, i.e a total of 4 thrusters to actuate pitch and roll motion, as well as motion in the z-axis.
The other 4 thrusters are mounted on the corners of the middle plate at 45 degrees in such a way that the vehicle can actuate yaw, sideways lateral movement and forward movement in the water at ease. The fabrication of the side frame is in such a way that it ensures that there is no disturbance in the water flow of these 4 thrusters. Thruster orientation can be properly seen from the top view as shown in Figure-2.

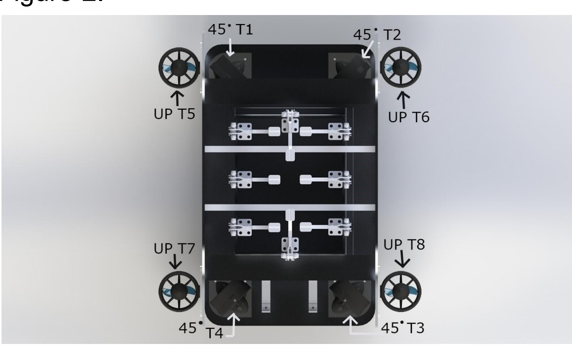

Figure-2: Top view and thruster orientation

Manipulator and End -Effectors:
In task 2, ROV is required to brust the target balloon. To accomplish this, a simple sharp pointed extrusion is rigidly fixed with the front side of the ROV frame. The pointed part when comes in contact with the balloon, it bursts.
As per the Task 3 of the competition the ROV is required to collect the surface balls, collect the golden baal from the bottom and also to burst the balloon. To accomplish all these subtasks, a separate end-effector is designed for each.
A single DOF manipulator, attached with a single DOF three claw gripper, is mounted on the middle plate. The manipulator is actuated through a servo motor to locate the gripper at the required position over the golden ball. A gripper is a three claw gripper actuated by a single servo motor. The three claws provide proper gripping of spherical objects. All the parts of the three claws gripper are 3D printed using Polycarbonate materials which provide high strength to weight ratio.
In task 3, ROV needs to collect the floating balls. For this purpose, ROV is provided with a net unfolding mechanism. The mechanism is initially in a position such that the attached net is in folded condition on the top of the ROV. The mechanism is actuated while the ROV is approaching the target zone from the bottom after collecting the golden ball and unfolding the net. It is aimed to collect the maximum floating balls in the unfolded net while maneuvering towards the surface. After reaching to the surface the mechanism is further actuated to fold the net partially so that the collected balls remain in it.

B. Software Sub-System:

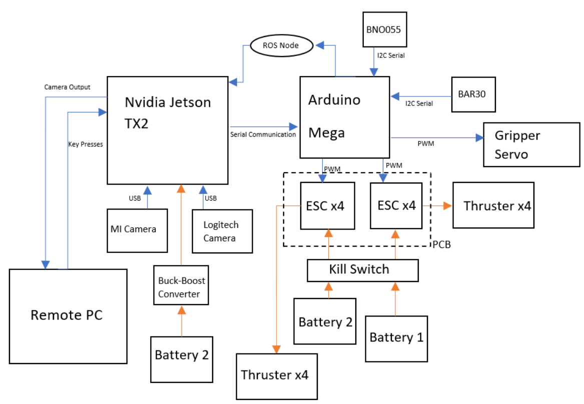

Figure 3: Block diagram of the system

For controlling the ROV underwater, remote access to the Nvidia Jetson TX2 is taken using Virtual Network Computing (VNC) through a LAN cable. Detailed block diagram of ROV is shown in figure 3. The Jetson is placed in the main hull and the input is given from the remote PC to a python program running on the onboard computer. This then communicates the key pressed to the Arduino Mega using serial communication at a specified baud rate. This python program shows a GUI interface of the key pressed and also displays live images from both the cameras mounted on the ROV as shown in Figure 4.

Figure 4: GUI Interface to control the ROV

The left window shows the front camera output and the right window shows the down-facing camera output. While the key presses and sensor data are shown at the top.

This interface is visible to the remote PC. For forward and backward motion ‘w’ and ‘s’ keys are used respectively, for lateral movement in the left and right direction, ‘a’ and ‘d’ keys are used respectively. As for depth control, ‘i’ and ‘k’ keys are used, where the former is used to decrease the depth and the latter is used to increase the same. Moreover, the keys ‘q’ and ‘e’ are used to turn the ROV in the left and right directions.

Underwater images tend to be degraded as components of light with higher wavelengths get scattered near the surface and only higher wavelengths can reach deeper in the water. Thus everything appears to be having a shade of blue and green. This makes it extremely difficult to distinguish between different objects, thus an underwater image enhancement algorithm is integrated. It can be toggled ON or OFF using the ‘g’ key of the remote PC and when it is at ON state this algorithm starts working and when it is on OFF state raw images from the camera are visible. This algorithm uses normalization and histogram equalization while reducing the intensity of blue and green components of the image according to the average value of these components. Comparison between raw images and enhanced images is shown in figure 5.

With the help of the Remote Operating System (ROS), a node was created that publishes IMU, pressure sensor data along the final PWM that is given to the thrusters from Arduino. This could also be done with the help of serial communication but it led to delayed outputs thus to fasten the process ROS is used. To achieve stability underwater and make sure that our ROV moves according to the given input used PID control that is described below.

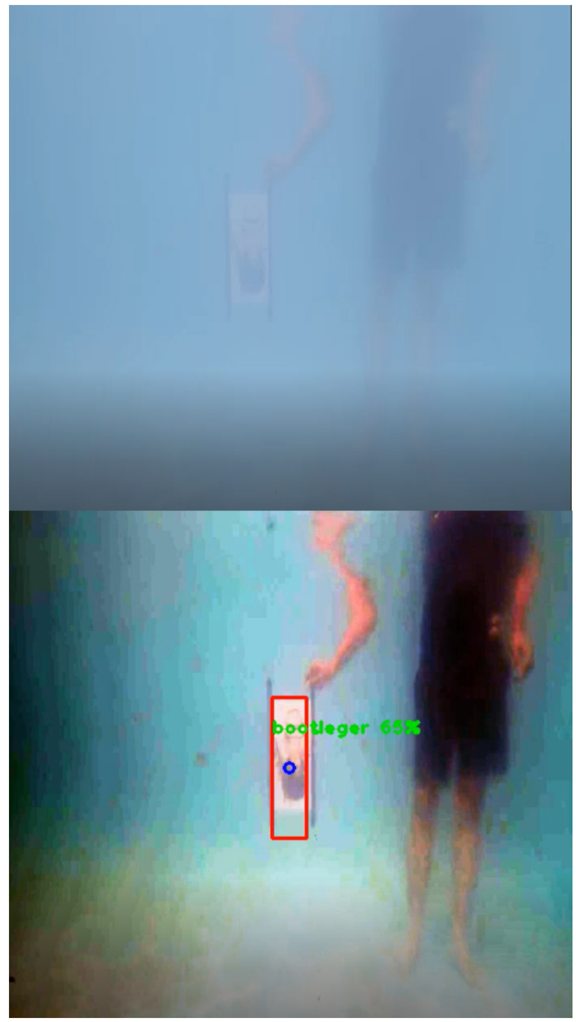

Figure 5 : Comparison between original and enhanced image

PID Control: BNO055 IMU is used to get the orientation (yaw, pitch, and roll) of ROV and BAR30 pressure sensor to calculate its depth. Initially when the ROV is powered on it is kept aligned to the pool and the yaw, pitch, and roll angles are stored in a variable as a set point. This setpoint is then compared to readings taken rapidly and then the error is calculated in orientation as the difference between current angles and the set point. This error is multiplied by the proportional constant ‘kp’ that is tuned during testing. For the differential part previous error is compared with the current error and divided by the time difference between two readings to get change in error with time. This is then multiplied with the differential constant ‘kd’. A condition is kept such that the integral part comes into action only when the error is less than plus or minus three degrees for fine-tuning. Such PID is also applied for depth control.

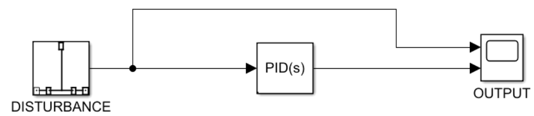

Figure 6: PID output to disturbance

If key ‘q’ is pressed then the set-point is decreased by one degree and thus long-press results in a left turn. Now the ROV stays aligned to this new set point. By following a similar procedure, the ROV turns right. For depth control when ‘i’ is pressed at the remote PC then the depth setpoint is decreased by 10cm and thus error also changes and the ROV automatically aligns to the new setpoint with the help of the PID loop. Thus the ROV can dive and surface.

C. Electrical Sub-System:

To control and power the thrusters, a PCB is designed on which all the eight ESCs are mounted along with an Arduino Mega Microcontroller as shown in figure 7. These ESCs are used to control the thrusters. Two four-celled, 10000 mAh 14.8V Li-Po batteries are connected to PCB, where each battery provides electric power to four thrusters. Another similar battery provides power to the Nvidia Jetson TX2 through a buck-boost converter at 10V and 1.5A.

Kill Switch: The kill switch is mounted on the main hull and it is used to kill the power to thrusters while keeping the Nvidia Jetson and Arduino running. It uses two relays where each relay is connected to a battery. The ROV has positive buoyancy so it surfaces on the operation of the kill switch. These relays can also be operated from the remote pc as they are connected to Arduino as well so this operation can be done from outside the water as well as by using the physical switch mounted on the ROV underwater.

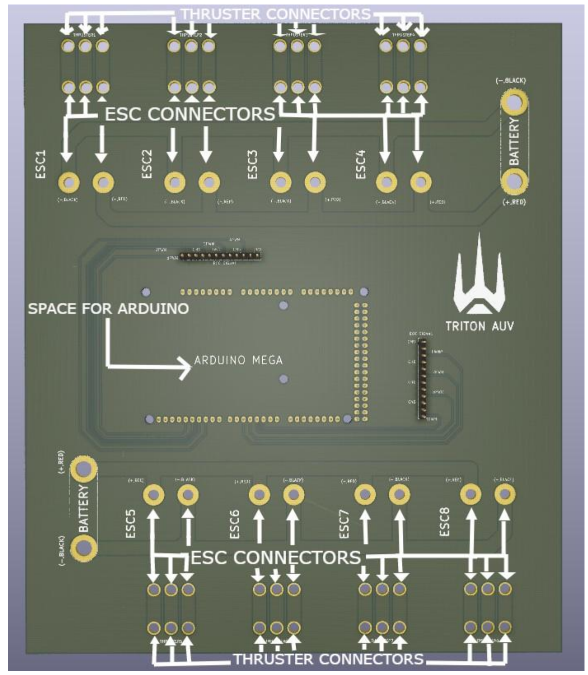

Figure 7: PCB used to mount ESC and Arduino Mega microcontroller

Sensors:

BNO055 Inertial Measurement Unit (IMU): BNO055 IMU is used that have a 3 axis magnetometer, 3 axis gyroscope, and a 3 axis accelerometer, and also a temperature sensor. It gives the following data outputs:

- Absolute orientation in the Euler vector at 100Hz.
- Absolute orientation in quaternion at 100Hz.
- Angular velocity vector: Three axes angular velocity in rad/s.
- Acceleration vector: three-axis acceleration in m/s^2, gravitational acceleration + linear acceleration.
- Magnetic field strength vector in microtesla at 20Hz
- Temperature (1Hz)
- Gravity vector at 100Hz
- For controlling ROV, Absolute orientation in Euler angles and the temperature data is used.

BAR30 (Pressure Sensor): BAR30 High Resolution 300m Depth/Pressure Sensor is used which is a high-measure pressure sensor with a 0.2mbar resolution that can measure up to 300m depth. It has Measurement Specialties MS5837-30BA and communicates over I2C on a 3.3V I2C.

Cameras: This ROV is equipped with two cameras, one primary front-facing camera and another mounted at the bottom of the ROV. The cameras used here are:

- Logitech C615 has 1080p/30 fps with autofocus
- Mi Webcam Hd has 720p with autofocus

III. Experimental results

Software: Given the dynamic complexity of the ROV and how the thrusters are mounted, the PID calibrated should provide the best achievable stability to ensure that the tasks are performed eloquently and no software issues occur. Hence, starting from writing Arduino codes that would give PWM values to thrusters, followed by PID tuning according to Agastya. Further integrating the values from BNO055 and Bar30 and then tuning it's PID by assigning a setpoint to it. If any disturbance is caused, it would come back to its original orientation (set-point). For proper visual representation and also to check whether PWM values are properly given to thrusters, thrusters values are published using ROS where if 'w' is pressed, thrusters of centre-plate (T1, T2, T3, T4) are given an additional PWM of 200.

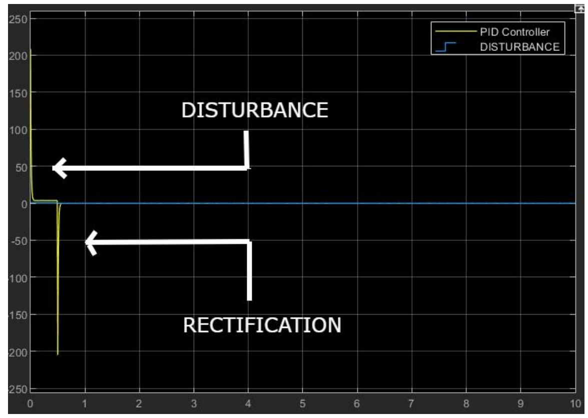

Figure- 8 : PID response to a disturbance

Waterproofing and durability: The first and foremost task was to ensure that the ROV hull remains waterproof throughout. Several days were spent trying to achieve complete waterproofing and once achieved , a final check was done by keeping it underwater at the depth of 2 meters for a day and simultaneously checking how much pressure it can withstand. Additionally, a pair of gaskets (i.e a mechanical seal) were placed which ensured that the lid when closed made the hull completely sealed. Moreover, the clamps of the main hull were also tested to check how much pressure the 10mm acrylic lid can withstand. Upon several trials, an observation was made that the acrylic lid was breaking repetitively as the clamps were hitting the critical point. After finally setting up the lid, one MS sheet lid is kept as a backup.

IV. Procurement Report

Component	Vendor	Model	Specs	Cost (in Rs)
Buoyancy Control	Designed In-House	Dead Weights & Foam	Swimming Pool Noodles	2000
Frame	Designed In-House	Aluminum 6 series	Dimensions: 70cm X 64cm X 45cm	7000
Main Hull	Designed In-House	Aluminum 6 series	Dimensions: 32cm X 25cm X 25cm	5000
Camera Hulls	Signoraware	Glass	Standard sizes	750
Waterproof Connectors	Imported	Plastic	GP7 standard size of waterproof connector	600
Motor Control	Blue Robotics	ESC basic R3	-	14000*
Actuators	Blue robotics	T100	BLDC	120000*
	Robokits	Servo 20kg cm	Waterproof for 20N	7200
Battery	Skycell	LiPo 4S	14.8V, 10000 mAh	34000
CPU & GPU	Nvidia	Jetson TX2	Memory: 4 GB 128-bit LPDDR4 51.2 GB/s Storage: 16 GB eMMC 5.1	60000*
Programming Languages	–	C++, Python	–	–
Inertial Measurement Unit (IMU)	Adafruit	BNO055	It is acceleration Sensor from Adafruit having 9-DOF Absolute Orientation IMU Fusion Breakout	4400*

Component	Vendor	Model	Specs	Cost (in Rs)
Pressure Sensor	Ocean Robotix	Bar30	High-Resolution 300m Pressure/Depth sensor with 2mm resolution	7000
Kill Switch	Ocean Robotics	High pressure waterproof Switch	Can handle upto 5A and 120V AC and 26V DC. Can be used till 1000m depth	1500
Camera(s)	Logitech	C615	1080p/30fps	3000*
	Mi	XMSXT001TM	720p	800
Manipulator	3D printed and designed In-House	Gripper and Arm	3D printed and attached on Aluminum arm	4000
		Ball catcher	Made out of aluminum rods	2000
		Needle	Aluminum	20

**These items were reused from our previous vehicle.*

VI. Acknowledgements

We would like to thank the Institute of Technology, Nirma University for providing all the necessary resources and also the faculty mentors, Dr. Mihir Chauhan and Dr. Akash Mecwan for their valuable insights and guidance in making of this ROV. We would also like to thank Aligarh Muslim University for hosting the competition.