

San Diego City College Technical Design Report 2020: Zoidberg Revisited

Jacqueline Alcazar, Regina Bochicchio, Jose Cardoso, Leila Lopez, Lukas Mueller, Liza Porterfield, Juan Rodriguez, Terry Thompson

Abstract—San Diego City Robotics Team is a team of engineering and computer science students who plan to compete in the 2021 Robosub Autonomous Underwater Vehicle (AUV) Competition. The strategy being employed is to continue bench testing and pool testing of the former team’s AUV, to complete the pre-qualification video, and eventually to incorporate vision ability into the present system.

I. COMPETITION STRATEGY

The San Diego (SD) City Robotics Team is part of the community college environment, where student turnover is frequent. None of the 2020 team members were present in 2018, the last time a City College Autonomous Underwater Vehicle (AUV) qualified to compete at Robosub. No members of that team are in contact with the college today, although they left a Github repository with their code. As a team with mostly new members and a limited equipment budget, it was decided the best path forward was to use “Zoidberg”, the 2018 SD City College AUV, without any appreciable changes (see Fig.1). This will allow the team to focus on improvements to the electrical system, further understanding of the other systems and software, and pool testing. The main goals for the 2021 competition are the following:

- Continued pool testing as soon as circumstances allow,
- Completion of a pre-qualification video,

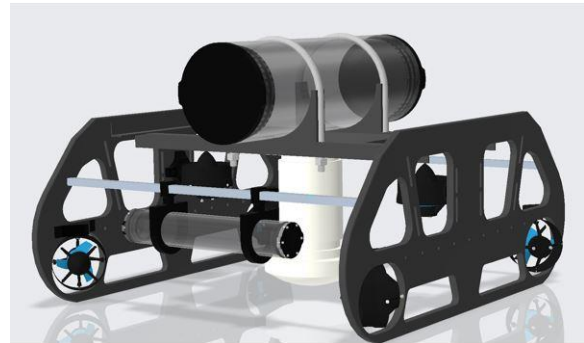


Fig 1. CAD drawing of Zoidberg

- Integration of the Nvidia Jetson TX2 microcontroller with Connect Tech Orbitty Carrier, and the Zed camera, which are ongoing projects for the team.

The eventual aim is to be able to attempt a subset of the tasks that require vision. These include touching the buoys and surfacing in the octagon. If these tasks become possible, the next step is to move on to tasks that require additional equipment, such as the marker drops, arm manipulation and torpedoes. Another goal for the team is to develop robust documentation for Zoidberg’s various systems, to help future City College Robotics Teams.

II. VEHICLE DESIGN

The last major modifications to Zoidberg, the City College AUV, were completed by the 2018 team. The 2019 team made one modification, replacing the magnetic kill switch with a mechanical one. The wiring of that project was never completed. The 2020 team is planning

future enhancements to Zoidberg, which include completing the kill switch mechanism, incorporating a separate watertight enclosure for the battery, so it can be swapped without opening and closing the main enclosure, and a general wiring update and replacement. Future plans also include replacing the present microcontroller with a donated Jetson TX2 computer and a Connect Tech Orbitty Carrier, to facilitate using a camera to complete some of the vision tasks.

A. Mechanical

The material used for the sides of the Zoidberg chassis is a High Density Polyethylene (HDPE). This is a thermoplastic polymer made from petroleum. This material is very resistant to breaking even when under pressure and is a good option for underwater-based projects. There are two aluminum crossbars at each end of the top of the chassis and one on the bottom in the front. Two, additional bracket crossbars, hold a rectangular center platform. At the present time, Zoidberg has 2 watertight enclosures, cylindrical in shape and made of clear acrylic. The main enclosure has a diameter of 15.2 centimeters (cm) and a length of 48.8 cm. Two U-bolts wrapped in rubber and 3D printed cradle mounts secure the watertight enclosure to the chassis, using the crossbars. The electronics board, containing the Pixhawk autopilot, the Raspberry Pi microcontroller and the electronic speed controllers (ESCs) is in this enclosure, along with the LiPo battery. The Pixhawk is connected to the pressure sensor, which is installed into one of the end caps of the enclosure. The smaller enclosure is 30.0 cm long with a diameter of 5.0 cm. It is fastened to the front of the AUV using aluminum brackets and a 3D printed mount. Eventually this enclosure will hold the camera. Neutral buoyancy for Zoidberg is provided by the addition of two foam squares to the

chassis. Each is about 5 cm thick and is about 12 cm on a side. These squares are zip-tied to the front and back of the AUV. The six thrusters are mounted in a vectored configuration, with four used for horizontal motion and two for vertical motion. The horizontal thrusters are mounted with aluminum bracket mounts, while the vertical thrusters have 3D printed mounts.

B. Electrical

The team constructed a schematic to facilitate overall understanding of the electrical system (see Fig. 2). A single, four-cell, lithium-ion polymer (LiPo) battery, with a capacity of 10,000 milli-Amp hours, is used to power the entire AUV. Power is controlled and distributed by a power module and distribution board (PM), which is combined in one unit. The PM provides a regulated 5 volts for the Pixhawk autopilot. It allows current and battery to be monitored through the surface computer base station software (when the AUV is tethered). It also provides high voltage/high current to the ESCs, which control the thrusters. The ESCs have a servo-type input which consists of a signal wire and a linear battery eliminator circuit (BEC). The BECs provide 5 volts to the servo rail of the Pixhawk, which in turn powers the Raspberry Pi. The Raspberry Pi uses a USB connection to communicate with the Pixhawk, using python code and the Pymavlink library [1].

C. Software

The present software used by Zoidberg consists of a series of Python programs written by a member of the 2018 team. It was not used during the 2018 competition. This code resides in a Github repository [2].

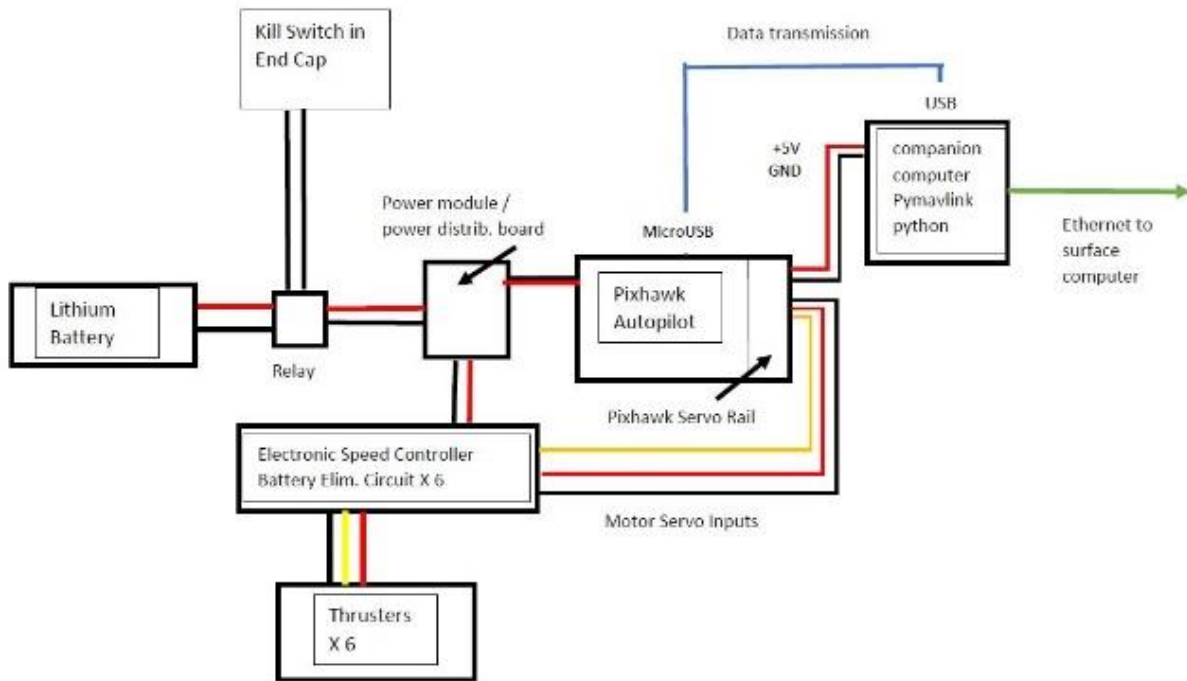


Fig.2 Zoidberg electrical system schematic

There is a navigation node which communicates with the autopilot, as well as a vision node. The navigation node utilizes the Pymavlink library to communicate with the Pixhawk. During testing, the surface computer is used as a monitor/keyboard via the Secure Shell (SSH) protocol.

The present team members have done significant testing of the navigation node (see Experimental Testing, below). The vision node operates the camera, utilizing OpenCV. The present team has not yet had the chance to work with the vision node.

III. EXPERIMENTAL TESTING

The majority of bench testing was done to establish how the software performed, as none of the team members had experience with it. In the fall of 2019, the present team, with the help of the physics lab, designed

and built a small land robot that moved using metal rolling castors, and had the same vectored configuration of motors in the horizontal direction. It was built of wood and 3D printed material. The small robot used the same electrical platform as Zoidberg; Pixhawk Autopilot, ESCs with linear BECs, a power module/distribution board, and power from a LiPo battery. The main advantage of the small robot was that the motors could be tested continuously without being in the water. The team was able to learn about the workings of the Pixhawk autopilot [3], and how QGroundControl base station [1] [4] was used to determine motor direction and how the pressure sensor was calibrated. The next task was to deploy the software. The team acquired two Raspberry Pi 3B microcontrollers and downloaded the pre-configured image file provided by ArduSub

to the Raspberry Pi SD Card. The Zoidberg software was cloned from the Github repository and added to the card. There was a steep learning curve as familiarity with both the Raspberry Pi, Linux and Python was limited. Connectivity and data transmission issues were particularly troublesome. However, the small robot was soon going forward, strafing, and turning with software commands.

The limitations of the preconfigured SD Raspberry Pi card were that it required a static IP address on the surface computer, and it did not allow the Zoidberg software to run with Python 3. At this point, the team acquired 3 Raspberry Pi 3B+ microcontrollers, downloaded the usual Raspbian distribution, and researched various websites [5] [6] to configure it properly. The software was then able to run in Python 3.

In the spring, the software was tested on Zoidberg in the pool with good results. However, the code failed when it came to controlling depth. That was about the time that all on campus activities were curtailed by the coronavirus. Software testing then continued via Zoom. The team determined there are 3 variables in the depth function where sign would affect vertical direction. The team member who had possession of the robot executed the depth function for short periods of time for all 8 different possibilities, eliminating those which caused downward thrust. For the remaining combinations, previously-logged data from a recent pool test were used for actual depths, and the results from the code were calculated using an Excel spreadsheet. This test resulted in 2 discoveries for the team: 1) the correct signs for the variables in the depth function with the current thruster configuration and 2) the fact that the direction of the vertical motors could be switched to get the desired results with all positive values for the depth

function variables. When the team can return to pool testing, this can be verified, and work on maintaining depth while executing the mission will also be continued.

IV. ACKNOWLEDGEMENTS

The San Diego City Robotics Club would not be able to succeed in this endeavor without the assistance provided by many people and organizations. We would like to thank Nvidia and Connect Tech Inc. for their generous hardware donations. We are particularly grateful to Ned Richards for authoring both the navigation software and the vision code we hope to use in the future. A special thanks goes out to Steve Volin and Kathrine Anduaga, who provided invaluable assistance on many of the technical aspects of this project, and Gina Bochicchio, our faculty advisor. We would also like to thank the administrators and lifeguards at the Ned Baumer Aquatic Center for facilitating our pool testing. Finally, we gratefully acknowledge the support and resources provided by San Diego City College, especially Dean Randy Barnes of the School of Engineering & Technologies, Mathematics, Sciences & Nursing, and Fred Julian, Chair of the Engineering Department. Their continued support affords us the opportunity to become better engineers and scientists.

V. REFERENCES

- [1] BlueRobotics, "ArduSub and the ArduPilot Project". [Online]. Available: <http://www.ardusub.com/> [Accessed 17 July 2020]
- [2] San Diego City College Robotics, "sdcityrobotics/zoidberg". [Online]. Available: <https://github.com/sdcityrobotics/zoidberg> [Accessed 17 July 2020]

[3] Ardupilot, “Pixhawk Overview”. [Online]. Available: <https://ardupilot.org/copter/docs/common-pixhawk-overview.html> [Accessed 17 July 2020]

[4] Dronecode Project, Inc., 2019, “QGroundControl”. [Online]. Available: <http://qgroundcontrol.com/> [Accessed 17 July 2020]

[5] Ardupilot, “Communicating with Raspberry Pi via MAVlink”. [Online]. Available: <https://ardupilot.org/dev/docs/raspberry-pi-via-mavlink.html#> [Accessed 17 July 2020]

[6] The Robotics Back-End, “Raspberry Pi 4 Pins – Complete Practical Guide”. [Online] Available: <https://roboticsbackend.com/raspberry-pi-3-pins/> [Accessed 17 July 2020]

Appendix A: Component Specifications

Component	Vendor	Model/Type	Specs	Cost (if new)
Buoyancy Control				
Frame	N/A	Custom	HDPE/Aluminum	
Waterproof Housing	BlueRobotics	6" enclosure 2" enclosure	49cm L, acrylic 29cm L, acrylic	\$317 \$93
Waterproof Connectors	TE Connectivity	WET-CON Connectors	1-16 contacts, 19A, 20K psi max	Donated
Thrusters	BlueRobotics	T200 x6	watercooled	\$1014
Motor Control	Unknown	ESC	Linear BEC	Unknown
High Level Control	BlueRobotics	Pixhawk1 with FPU	32-bit ARM Cortex M4 core	\$139.00
Actuators	Walmart	JD1914	12 V, 100A	\$7.49
Propellers	BlueRobotics	for T200	3CW, 3 CCW	\$18.00
Battery	Turnigy	Multistar	4S,10 Ah,10C	\$61.94
Converter				
Regulator	mRobotics	AUAV ACSP4	5V, 12V regulated power module	\$37.90
CPU	RaspberryPi	Model 3B+	ARM Cortex-A53 1.4GHz	\$41.48
Internal Comm Network	USB			
External Comm Interface	Ethernet			
Programming Language 1	Python			
Programming Language 2				
Compass	BlueRobotics	ST Micro LSM303D	accelerometer and magnetometer	see Pixhawk
InertialMeasurementUnit(IMU)	BlueRobotics	Invensense MPU 6000	3 axis gyroscope/ accelerometer	see Pixhawk
Doppler Velocity Log (DVL)				
Camera(s)	StereoLabs	Zed 1	3D, Dual 4MP	\$449.00
Hydrophone				
Manipulator				
Algorithms: vision	Open CV			
Algorithms: acoustics				
Algorithms: localization/mapping	custom			
Algorithms: autonomy	custom			
Open source software	pymavlink			
Team size (number of people)	15			
HW/SW expertise ratio	5:1			
Testing time: simulation				
Testing time: in-water	10 hours			