STONY BROOK
SOLAR RACING

Boat #3: Solar Seawolf

April 28th, 2025

Team Members:

Wesley Ng, Matthew Leo, Steven Chen, Nick Zhang, Kumpu Ide, Ryan Tang, Md Zaman, Bill Zhang, Maggie Tan, Chloe Jiang, Owen Chen, Anton Yanaky, Suhayla Yousuf, Ivan Yu, Daniel Zou, Wilson Lin, Daniel Lew, Nancy Zhu, Wing Chan, Allen Li, Kayla Rodriguez, Yaozhu W, Ryan (Jianyang) Tang

Faculty Advisors:

Austin Giordano and Professor David Westerfeld

Executive Summary

The Stony Brook Solar Racing Team is a student-run club supported by the university and various sponsors. The 2024 Solar Splash was relatively successful compared to previous years. However, we had noticed how slow our vessel was despite having similar hardware to other teams. The team's goal for the 2025 Solar Splash is to reduce the weight of our current systems and unlock the full potential of our motor through further tuning.

The Solar Racing Team is divided into three groups: Mechanical Engineering, Electrical Engineering, and Computer Science. Each team has worked throughout the semester for hours to complete our goals this year and produce a faster boat.

The mechanical team this year was responsible for improving the dashboard, outboard, and steering. Their work also included designing a new water cooling system to accommodate the motor controller and the motor. These new designs focused on minimizing weight by using lighter materials and reducing the complexity of previous systems including the steering and the outboard.

The electrical team this year was responsible for the integration of all electrical components of the boat including but not limited to the motor, motor controller, batteries, solar panels, solar chargers, and dashboard controls. The data acquisition system circuitry was also designed by the electrical team in collaboration with the software team.

The software team this year was responsible for programming the data acquisition system and creating the interface for the dashboard display. Using data gathered from tests and the upcoming competition, they aim to create a model to identify trends and offer insights for optimizing the boat's performance in the future.

The major developments of the boat this year are mainly related to motor controller tuning and steering. Levers and push-pull cables were used to create a steering system that is lighter and much more responsive than our previous hydraulic steering. With the goal of utilizing our motor's full power, we have acquired a new motor controller from inMotion, and extended discussions with them have led to a more optimized tuning of our motor.

The team is very thankful for the help from our sponsors, who enabled us to improve the boat this year. This is the second time our club has participated in two competitions in a single school year, having participated in the Promoting Electric Propulsion for Small Crafts (PEP) competition in Virginia Beach earlier in mid-April. We faced plenty of challenges already this year and are ready to see what we will face in Solar Splash this year.

Table of Contents

I. Project Objectives

Our goal this year was to build a more efficient and reliable boat. We are continuing to use our DHXmachines PEREGRiNE P40 motor, but we have encountered several issues involving our inMotion ACS48M55 motor controller that have prevented us from running at full power. Our solar system was also hastily put together before competition, which limited our energy gain despite incorporating new panels. Steering for the slalom race and in general was lacking in responsiveness. By following the objectives listed below, we hope to perform better for this year's Solar Splash:

- Propulsion
    - Acquire the ACS48L70-W35T inverter from inMotion
    - Improve water cooling for the motor and add water cooling to the motor controller
- Steering
    - Replace the hydraulic steering system with a lever and cable-based system
    - Redesign the dashboard, eliminating the previous steering system's helm pump, to reduce weight and size and improve driver ergonomics
- Solar Panels
    - Acquire more new MPPT solar chargers endurance race and overall charging
    - Reduce weight of solar panel frames
- Data Acquisition System
    - Create a system capable of measuring battery and solar panel voltage and current
    - Allow the driver to easily read acquired data through the dashboard
    - Program a model to optimize performance of the boat in endurance configuration
- Organization
    - Acquire Deutsch connectors and wire sleeving
    - Avoid excessive use of lever nuts and crimp connectors when possible
    - Maintain standards throughout all stages of the design process
    - Neaten electrical box for easy troubleshooting

II. System Design

A. Solar System

1) *Current Design:*

Our system consists of four Renogy 175W monocrystalline solar panels connected in a combination of parallel and series configurations, totalling 700W. These four panels are mounted in two sets of two, with one set placed above the driver's seat and the other in front of the driver and our dashboard, as shown in **Fig. 1**. One set is connected in parallel, and the other in series. Each panel weighs 10 kg and measures 49.7 inches by 27.5 inches by 1.4 inches. The electrical and physical characteristics of the panels are shown in **Fig. 2**.

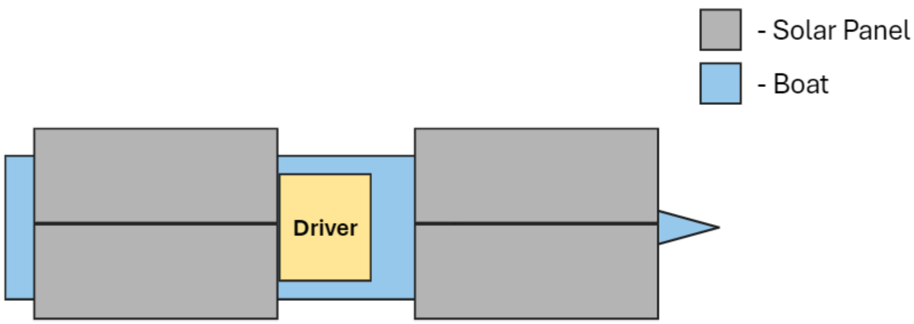


Fig. 1. *The 2024 Solar system setup.*

Maximum Power at STC*	175 W
Optimum Operating Voltage (V_{mp})	18.1 V
Optimum Operating Current (I_{mp})	9.67 A
Open Circuit Voltage (V_{oc})	21.6 V
Short Circuit Current (I_{sc})	10.31 A
Module Efficiency	19.8%
Maximum System Voltage	6000 VDC
Maximum Series Fuse Rating	20 A
Solar Cell Type	Monocrystalline (6.5 x 2.9 in)
Number of Cells	64 (4 x 16)
Dimensions	1262 x 699 x 35 mm(49.7x27.5x1.4in)
Weight	10 kg(22lbs)
Front Glass	Tempered Glass 0.13 in (3.2 mm)
Frame	Anodized Aluminum Alloy
Connectors	Solar Connectors
Fire Performance	Type 1

Fig. 2. *Electrical and Mechanical data specifications of solar panels.*

Half of the solar panels are equipped with a water cooling system that cycles water along the back of the panel to lower their temperature and improve efficiency. Aluminum cooling blocks are attached to the back of the panels using thermal epoxy and connected using rubber tubing, and a DC water pump provides the coolant flow.

2) ***Analysis of Design Concepts:***

The majority of the budget of the electrical team went into purchasing these panels from Renogy to meet the new 720W rule of Solar Splash. As a result, the choice of a solar charge controller was heavily restricted. We ended up settling for a 1000W PWM solar charge controller which was oversized for our charging configuration. It ended up helping us complete the endurance race, but we believe that our performance could be better if we utilized a better charge controller to unlock the full potential of these panels.

This year, we purchased two Genasun 325W MPPT solar chargers. It is important to consider the sizing of any MPPT solar charge controllers for solar panels. In order to optimize output power, an MPPT charger will adjust its variable load accordingly. When input power is lower, the variable load will reduce itself. However, the problem begins when the input power is so low that even the lowest variable load setting on the charge controller consumes most of the output. To mitigate this, we have to pick solar charge controllers with a maximum power input that is close to the maximum output of our solar panels. That way, even under the worst performing sun conditions, the variable load of the charge controller is within the engineered range so the load will not consume too much of the output power.

We are continuing to use our 175W Renogy panels, which means that in order to avoid the variable load problem, we will need to wire two of our panels in series to achieve 350W for a single charger. Although the charger is rated for 325W, Genasun has noted that their chargers are capable of handling a 9A input, meaning a maximum input of 370W, which will be sufficient for our panels. This means that we can continue using the configuration of two sets of panels in series, wired together in parallel.

3) ***Design Testing and Evaluation:***

Despite the panels being rated for 10A, which is above the maximum 9A rated current of our charge controller, we found them rarely operating at such numbers. In the event that the current ever exceeds the rated limit, we will install 9A breakers in series between the panels and the controller. That way, there should never be enough current to damage the charge controller, and we should easily be able to reset it when out on the water. Further testing and documentation will be done when we receive the solar chargers.

B. Electrical System

1) Current Design:

Our current design operates at 36V, utilizing three 12V 50Ah sealed-lead-acid (SLA) rechargeable batteries from UPS Battery Center connected in series. The design includes the motor controller, contactor, and motor. The motor controller is an ACS48M55 inverter from Inmotion that produces a 3-phase AC power to drive the DHXmachines PEREGRiNE P40 motor mounted on our propulsion system. Located on the dashboard are switches for boat power, the bilge pumps, and the linear actuator to lift and lower our propulsion system. Our dashboard contains a forward/reverse switch used to reverse the direction of the motor if needed. To power our linear actuator and data acquisition unit located in the electrical box, we implemented a 12V step-down converter. The bilge pumps are powered by an auxiliary 12V battery. We have a potentiometer mounted to the dashboard to act as our throttle, and the deadman's switch remains in the dashboard as it is the best position for a safe disconnect.

2) Analysis of Design Concept:

Previously, we used a 1000 A, 50 MV shunt in the electrical box, as its resistance was insufficient to gather accurate voltage readings. Our new design no longer implements this, instead, the team developed a data acquisition system (DAQ) that reads voltage and calculates current of the boat. The DAQ contains an ADS1115 chip connected directly to the boat's true ground and power. However, as direct battery voltage exceeds the ADS1115 maximum voltage rating, we incorporated a voltage divider composed of 100kΩ and 1MΩ resistors. A 1.22µF capacitor was added to create a low-pass filter for removing unnecessary noise.

We installed a 12 to 5-volt DC step-down converter to power our liquid crystal display (LCD) screen. The LCD now displays the total remaining battery of the boat and its time until depletion. Our new cooling system incorporates a 12V VEVOR high-pressure diaphragm pump and two 18W high-flow cooling fans. The cooling fans are connected to the 36 to 12-volt DC step-down converter and are powered immediately after contact engagement. At startup, it was found that the coolant pump's high current demand of the main circuit inhibited the activation of the main contractor, thereby preventing current flow to the motor controller and disabling system operations. To address this, the pump was made toggleable using a switch located on the dashboard to allow it to be activated after the contactor had turned on.

3) Design Testing and Evaluation:

We have made improvements to our electrical box, reorganized wires to optimally fit the boat's remodeled dashboard and steering systems, and updated pump configurations. Building upon our previous use of the TE Connectivity Deutsch series automotive connectors and cable glands, the electrical box was made completely watertight. To accomplish this, our team drilled holes along the sides of the electrical box in which we mounted Deutsch connectors, ensuring our connections for the motor control, cooling, dashboard, and pump systems are watertight. Additionally, we have done more wire management with wire sheaths that run from the dashboard to the electrical box on both sides of the boat. The potentiometer was relocated from the dashboard to the right steering lever to make it immediately accessible for the driver. Thus, the potentiometer wires were relocated to its respective location. Two 12V bilge pumps located at the front and rear of the hull were connected in parallel, and will be powered using a separate 12V auxiliary battery and toggleable using a switch on the dashboard when needed. In addition to the performance testing we have done during previous competitions and tests at a local lake, we plan to perform further testing to evaluate our electrical system's reliability.

C. *Power Electronics System*

1) ***Current Design:***

Our system uses a motor controller from inMotion, the ACS48M55 inverter, which produces 3-phase AC power to drive our DHXmachines PEREGRiNE P40 motor. We are using three 12V 50Ah SLA rechargeable batteries from UPS Battery Center to power this system. The system runs the same as previous iterations, with just one motor controller and one contactor to control the system.

2) ***Analysis of Design Concepts:***

This year, we have purchased a new motor controller from inMotion, the ACS48L70-W35T inverter. The ACS48L70-W35T produces the same three phase AC power to drive our motor but offers a peak current of 700A and peak power of 41 kVA compared to the ACS48M55's peak current of 550A and power of 32 kVA. Using a motor controller with these higher current and power ratings ensures our system is able to operate closer to full potential during sprints since our current is no longer limited under peak loads. This motor controller shares the same pin layout as the ACS48M55, which allows for quick and easy transfer of hardware. This system also runs the same electrical configuration as previous iterations, with just one motor controller and one contactor to control the system. We are still using 3 12V 50Ah SLA rechargeable batteries from UPS Battery Center to power this system, but we have three brand new ones for this year.

Due to having a higher-capacity motor controller, we had to provide it with water cooling, which required a redesign of our cooling system. Our new water cooling system uses a VEVOR high-pressure diaphragm pump, an automotive heater core, and two high-flow cooling fans to pump pure distilled water through the motor controller, which then flows through the motor and into the heater core to extract heat.

3) ***Design Testing and Evaluation:***

At our competition in Virginia this year, we once again had motor controller problems which prevented our propulsion systems from functioning properly until the last day of competition. We are planning another wet test at our local lake before Solar Splash to perform motor controller tuning to get the proper output from our motor. The tuning will be done on the water to see real world conditions the motor will have to perform in at competition. We have yet to see the full capabilities of the P40 with our vessel and hope to see it perform at its peak before competition. Wiring diagrams for our 2025 electronics system have been provided in **Appendix E3-4**.

D. Hull Design

1) ***Current Design:***

The hull has worked well and has met the team's expectations of serving as a well-rounded hull that would improve the team's performance in the endurance, sprint and slalom heats. The hull measures 14 feet in length, 40 inches across, and has a waterline 14 inches below the top-most surface. The hull's underside geometry has improved the team's ability to get on plane in sprint races; the longer overall length allowed for a greater hull speed in endurance races; the wider hull provided more stability when taking corners in the slalom races and when faced with turbulent waves. Within the hull, the boat is fitted with 8 foot-lengths of 1.5-inch-wide aluminum slotted extrusions, which provide flexibility in the mounting of the driver's seat and other subsystems. This flexibility has allowed the team to adjust the boat's center of mass, varying the speed at which the boat starts to plane as well as its top speed, and conveniently accommodating future change to the boat's hardware. Please refer to ***Appendix G*** for Hull drag simulation and other relevant calculations.

2) ***Analysis of Design Concepts:*** N/A

3) ***Design Testing and Evaluation:*** N/A

E. Drivetrain and Steering

1) *Current Design:*

Our current design for the drivetrain and steering was created by two mechanical engineering senior design teams. The propulsion team was tasked with designing an outboard-style propulsion system to be integrated with the DHXmachines PEREGRiNE P40 motor. The steering team was tasked with designing the mount for the outboard unit and incorporating a method for steering and trimming the motor. This led to the current design, which was inspired by commercial outboard engines, which uses hydraulics for steering and a linear actuator to raise and lower the motor for ease of launching and loading.

The current drivetrain uses a water cooling loop composed of a 360mm computer radiator, three 120mm computer fans, and a 12V pump for cooling the motor. The ACS48M55 motor controller is passively cooled using an aluminum heatsink.

The current dashboard, onto which the hydraulic helm pump, control switches, gauge, and display are mounted, was constructed of aluminum extrusion, sheet metal, and a transparent polycarbonate front panel.

2) *Analysis of Design Concepts:*

For our new steering design, the hydraulic steering was replaced with a lever and push-pull cable-based system in an effort to reduce weight, improve ergonomics, and quicken steering response. Two levers on the right and left of the driver are used for steering input, and they were designed to move in opposite directions and to allow the driver to make both left and right turns using only one lever. This flexibility allows the driver to use both levers simultaneously, pushing one forward while pulling the other backward, for easy, steady turning in normal situations, or use only one lever in either pull or push in situations where they may need to take one hand off of one lever. These capabilities are accomplished by utilizing one push-pull cable, bellcrank, and linkage rod for each lever and having their outputs meet at the tiller arm attached to the motor.

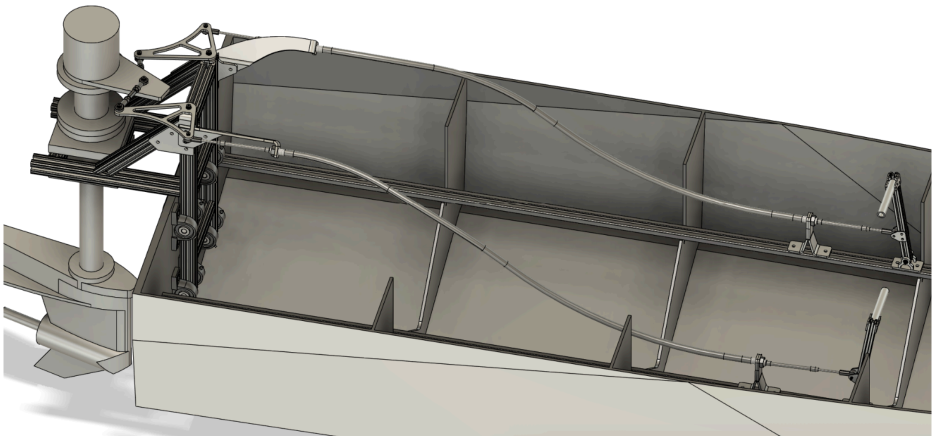


Fig. 3. *CAD model of the cable steering system attached to our boat*

The design process for our new steering components, including cable mounts, bellcranks, bellcrank mounts, and levers, required thorough consideration of the loads they would experience. The steering was designed with the requirement that the system should be able to turn the outboard while at its maximum steering angle of 35° with only one lever. Furthermore, the maximum load transmitted by the push-pull cable should be no greater than its rated tensile and compressive loads, and all custom parts should be designed with a factor of safety of at least two. Lastly, the driver should not be required to exert more than a comfortable amount of force when steering with one lever. Following these requirements, the steering system was designed iteratively, and the strength of all components was verified using static force analysis and finite element analysis (FEA) in Autodesk Fusion 360. By determining the maximum torque necessary to turn the outboard, 200 pound-inches, that value was used to derive the components of the forces on the bellcrank, bellcrank mount, cable mount, and levers, and those force components were finally used to analyze each component individually using FEA. Analyzing the forces revealed that the maximum force required of one cable to transmit in compression would be 104 pounds, which is within its dynamic load capacity of 130 pounds, and the maximum force required by the driver would be 33 pounds, which is comfortably within the strength of our driver. Furthermore, using FEA revealed that all of our components satisfied our requirement of having a factor of safety of two, with the bellcrank having a factor of 12.21, the combined rear cable and bellcrank mounts having 3.05, the front cable mounts having 2.43, and the lever having 6.56.

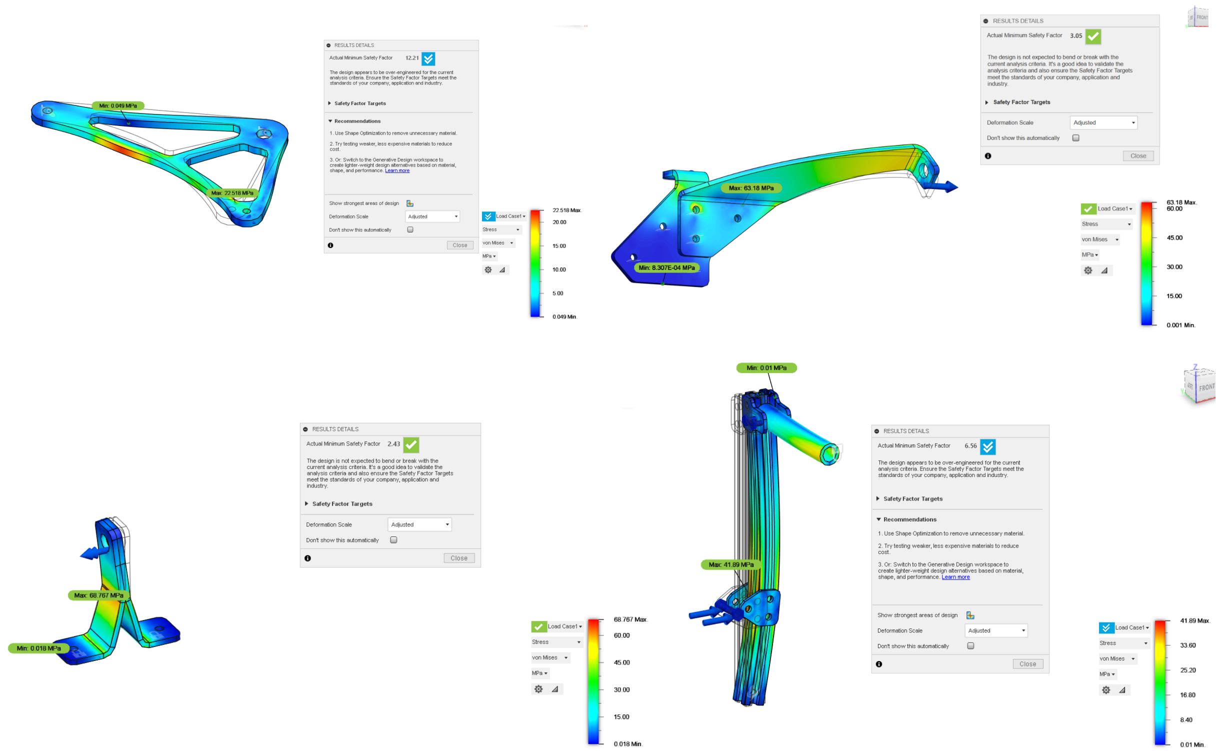


Fig. 4. *Autodesk Fusion finite element analyses of the bellcrank, combined bellcrank and rear cable mount, front cable mount, and steering lever*

In our previous design, a linear actuator and linear rails were implemented to raise and lower the outboard. While functional, the actuator and its supports proved to be excessively heavy, so it was decided to transition to a hinge-mounted outboard assembly, reducing weight and removing reliance on electric power. The outboard is now mounted to eight pillow-block bearings, which are bolted to a custom transom plate. The outboard hinges about the bottom four pillow-block bearings using ½” shoulder bolts to allow the assembly to be tilted and raised for transportation. The four lower bearings remain bolted to the outboard while the upper four bearings are disengaged from the outboard for transportation and reengaged once the boat is in the water. ½” shoulder bolts are then used to fix the outboard into its upright position. Using the weight of the outboard of approximately 165 pounds and the known dynamic thrust produced by the powertrain of 110.5 pounds, the maximum reaction forces at each lower pillow blocks was calculated to be 117 pounds and the maximum reaction forces at each upper pillow blocks was calculated to be 82 pounds, which is well within the pillow blocks static load rating of 1500 pounds. Furthermore, as the transom was constructed from ¾” thick solid fiberglass board laminated with four layers of bi-axial carbon fiber and reinforced with an aluminum plate, the load from the pillow blocks poses little concern to the integrity of our hull.

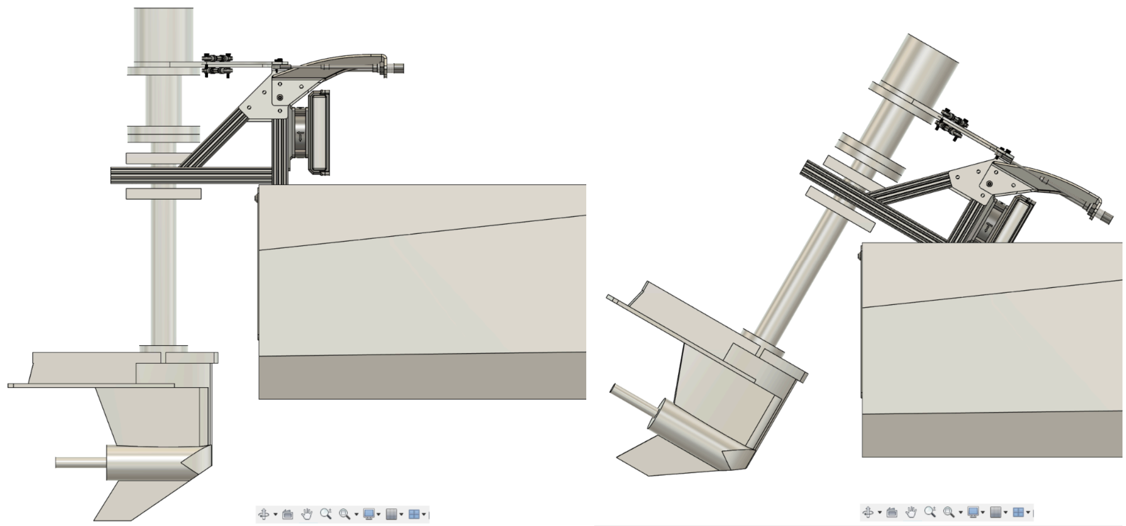


Fig. 5. *Cad model shows the lowered and raised positions of the outboard.*

Removing the hydraulic steering system eliminated the need for a helm pump and steering wheel to be attached to the dashboard, allowing the dashboard to be redesigned. The old dashboard, constructed of heavy aluminum extrusion, sheet metal, and polycarbonate, was replaced with a compact plywood dashboard to reduce space and remove unnecessary weight while retaining sufficient integrity. The size of the front panel of the dashboard was reduced from 23 x 14.5 inches to 20.375 x 7 inches, the overall height was lowered from 12 inches to 9 inches, and the angle of the front panel was lowered to improve driver ergonomics. On the front panel of the dashboard, the throttle was removed and placed onto one of the steering levers, and a switch was added to control the cooling pump. Additionally, we rearranged the placement of the components connected to the dashboard to streamline wiring.

From previous years, it was observed that the old cooling system provided insufficient cooling to the motor, and with the procurement of a new motor controller that draws significant amounts of power, a new cooling loop was necessary. The new cooling loop utilizes a VEVOR high-pressure diaphragm pump, an automotive heater core, and high-flow cooling fans. The cooling loop also incorporates various barb fittings that match the inlet and outlet dimensions of respective components (motor, motor controller, and heater core) and different hoses to accommodate those changes. The cooling solution chosen for the system was pure distilled water due to the large heat capacity, which allows for greater rates of heat transfer. Furthermore, it would be unlikely for the vessel to operate near freezing conditions to necessitate the use of a water-glycol solution.

Fig. 6. *Cooling System installed on the boat*

As for the details of the cooling loop itself, it begins at the pump, where the coolant is first pumped to the motor controller. The coolant is then pumped to the motor and then the heater core, where it begins to cool. The heater core is mounted on the top half of the outboard and has fans to increase air intake, resulting in increased cooling for the system. The output coolant then flows to the overflow reservoir, and the process is repeated. This order was determined based on how much heat each component generates in use. The DHX motor generates approximately 2.3kW of heat and will likely raise the coolant temperature beyond the maximum inlet temperature of the motor controller of 60 °C. The maximum continuous power output of the system is the motor output, which is rated for 23kW. The motor is rated for 95% efficiency, but due to a reduced operating voltage, the team estimates a 90% efficiency for the motor. The motor controller is estimated to be 95% efficient, and thus, a total system efficiency of approximately 85% was determined. Finally, the total heat generated by the components is then calculated using this efficiency and maximum continuous power output to obtain 3.3kW of heat.

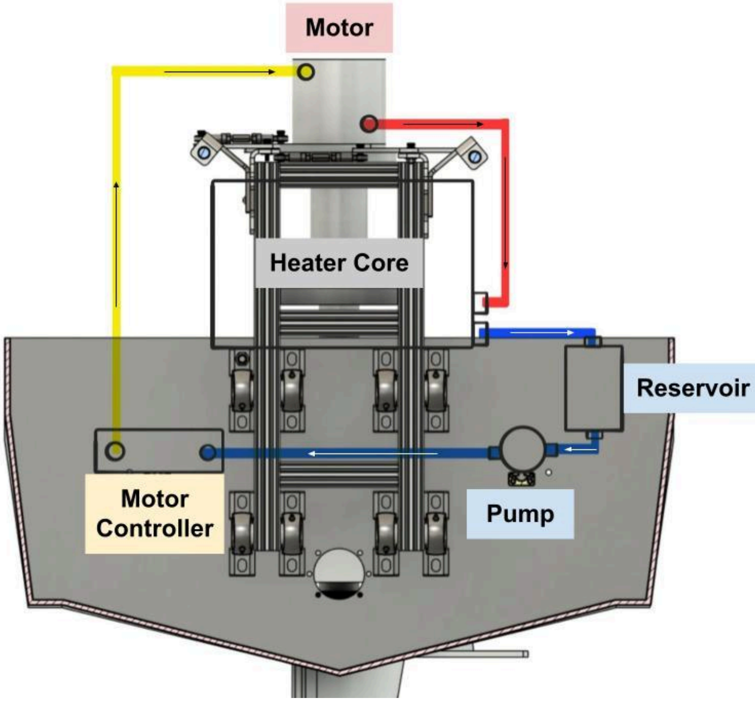


Fig. 7. *Cooling Loop Schematic for Propulsion Cooling*

3) ***Design Testing and Evaluation:***

Testing the new steering system at the PEP competition earlier this year, it proved to function exactly as designed and offered excellent steering response and ergonomics, greatly improving the maneuverability of our boat and comfort of our driver. With the outboard now mounted on pillow blocks and raised and lowered by hand instead of with a linear actuator, it was found that manually raising the outboard was more physically demanding than expected, mostly due to the considerable weight of the assembly. Although we found that this was a worthy trade off to reduce some weight, we will look into more convenient solutions for raising and lowering the outboard in the future.

At the PEP competition, the cooling system worked as intended and kept the motor and motor controller at around 35°C which was well within the acceptable operating temperature range for both components. Prior to running our heat at the PEP competition, the pump was tuned to a lower pressure setting as the additional flow rate was not necessary at the time. It is worth noting that the motor was not pushed to its maximum load and the higher temperatures at the Solar Splash competition can significantly impact the temperature of these two devices. The pump’s pressure and flow rate will be modified accordingly to meet the specifications necessary to do well at the Solar Splash competition. One problem that was encountered at PEP was the compatibility of the cooling system with the electrical subsystem as it drew too much power upon startup, and insufficient power was delivered to the motor controller that switches on the external contactor. To resolve this issue, the cooling pump was wired to a switch on the dashboard so that sufficient power can be supplied to the motor controller upon startup.

F. Data Acquisition and Communications

1) ***Current Design:***

The previous data acquisition used 3 ADS1115 breakout boards measuring solar charging current and voltage, motor current and voltage, and battery current and voltage. The data from the 3 boards was collected by an Arduino Uno via I2C, then relayed to our Raspberry Pi 3B+ via USB serial. Current was measured using shunt resistors throughout our system to measure the voltage differential between the two resistors. Both the current and voltage data were sent to a display on the dashboard for the skipper to view live. Unfortunately, the 3 boards stopped working due to a short and had been rendered unusable.

2) ***Analysis of Design Concepts:***

The new data acquisition system should monitor power consumption, calculate real-time Time-To-Depletion (TTD) and battery percentage, throttle, and relay this information to the dashboard. Voltage is measured using an ADS1115 ADC, read by an Arduino UNO after being stepped down and filtered. Current is measured by a Honeywell CSNV500M Hall-effect sensor connected via a USB2CAN interface. The throttle is read by another ADC connected to the potentiometer. A Raspberry Pi 5 runs Python scripts to collect this data, perform calculations, and manage storage. Raw sensor data is logged locally to a SQLite database for analysis, with calculated metrics cached in Redis for fast dashboard retrieval.

Data collected includes voltage, current, throttle, battery percentage, and time-to-depletion (TTD). These parameters are displayed prominently on the dashboard, allowing the driver to monitor energy usage actively during operation. The data processing is handled by formulas defined as:

$$Battery\,\% = \frac{V_{measured} - V_{min}}{V_{max} - V_{min}},\ V_{min} = 34V,\ V_{max} = 38V$$

and

$TTD = \frac{50Ah}{I}$, where I represents the current measured in amperes.

In reviewing potential design concepts for data acquisition and communication, we considered several options. Initially, integrating a single-board microcontroller such as the Arduino alone was explored; however, the requirement for more extensive computational capabilities and storage for logging and real-time calculations necessitated the inclusion of a Raspberry Pi. The decision to combine an Arduino and a Raspberry Pi created a robust, modular solution. This modular design ensured that if one component died it could be quickly replaced without harming other parts of the system. The Arduino efficiently interfaces with the ADS1115 ADC for precise voltage measurements, while the Raspberry Pi leverages processing power and storage capabilities to handle real-time data interpretation, communication with the USB2CAN interface, and management of the databases.

For current sensing, the selection of the Hall-effect current sensor CSNV500M was driven by its reliability and ease of integration with existing CAN infrastructure. The utilization of Redis for caching data ensures rapid retrieval of calculated metrics by the dashboard, enhancing the responsiveness and reliability of real-time feedback to the driver. SQLite was chosen to store raw data for detailed post-operation analyses aimed at continuous optimization.

3) ***Design Testing and Evaluation:***

Testing of the data acquisition and communication system involved both bench tests and operational validation on-water trials. Bench tests were conducted initially to validate sensor accuracy, communication integrity, and algorithm correctness in controlled conditions. Voltage measurements from ADS1115 chips and current data from the CSNV500M sensors were cross-verified using precision multimeters and known load resistances to ensure accuracy within acceptable tolerances.

Operational testing occurred during wet tests conducted at our local lake. These tests validated real-time data acquisition and communication robustness under actual conditions similar to competition settings. Issues identified during these tests included minor discrepancies in current sensor readings. This was addressed by improving shielding and grounding techniques, enhancing signal integrity.

The integration between Arduino and Raspberry Pi was thoroughly tested to ensure consistent, uninterrupted serial communication. Additionally, database performance was evaluated under prolonged use conditions, confirming that Redis provided optimal performance for real-time data access, while SQLite effectively managed larger data sets for subsequent analysis. For the Solar Splash competition, further testing is planned to optimize the predictive accuracy of the battery percentage and time-to-depletion estimations as well as the effects of solar power on the readings.

III. Project Management

A. Team Organization

The 2024-2025 Stony Brook Solar University Solar Racing team continues to be led by our Executive Board, which handles the administrative duties of the team. This includes maintaining the club's budget, increasing membership, overseeing the three sub-teams, maintaining communications with faculty advisors, and more. These subteams are mechanical, electrical, and software, each managed by its own team lead who is responsible for their portion of the boat. Projects and deadlines are assigned by leads to their respective sub-teams each week. Sub-teams now meet together and collaborate four times a week, which is the most in the history of our team. It's allowed

B. Project Plan and Schedule

The 2024-2025 school year was the first time since we rejoined Solar Splash four years ago that a complete rework of the boat was not required. Because of that, we were able to hold more wet tests at the beginning of the fall semester, which helped us fix any of the problems that plagued us during 2024's Solar Splash. This year, we competed in the American Society of Naval Engineers' Promoting Electric Propulsion for Small Crafts (PEP) competition in conjunction with the Office of Naval Research and the US Department of Navy.

The competition took place from April 15 to 17. By that time, we had conducted several successful wet tests and felt confident in the boat's functionality. However, part failures and loose connections made the lead-up stressful until we finally qualified. Thanks to those early wet tests, the boat arrived at the competition in a condition where—even if damaged—repairs were not an impossible undertaking. The boat is now in a functional state for Solar Splash.

We intend to hold a wet test before the end of our school semester to finalize the tuning of the motor controller with InMotion. The team's designated skippers also use this time to familiarize themselves with the boat. Observations during this wet test will be used to make any remaining optimizations of the boat.

C. Finances and Fundraising

The Stony Brook Solar Racing Team receives a large portion of its funding from the university's Undergraduate Student Government (USG), which attains its funding for clubs like ours from the Student Activity Fee. For the 2024-2025 academic year, we fully utilized a ~$5,200 budget, which went towards boat materials, tools, and other equipment for our shop as well as school events, which helped with the promotion of our club and its purpose. New this year, the Team received two $5,000 grants from the Office of Naval Research as a returning team and a new unmanned team partaking in the Promoting Electric Propulsion for Small Crafts competition. We will be saving one of the $5,000 grants for next year to use as funds for the construction of the new hull. The majority of the available funds went towards travel expenses, but the remainder was used to upgrade systems of the boat.

In addition, Solar Racing receives funding from Stony Brook's College of Engineering and Applied Sciences (CEAS), which funds Solar Racing on a request-by-request basis. Solar Racing also has funds donated by recent alumni. Furthermore, the establishment and preservation of relationships with sponsors and alumni are essential to the continued growth and success of the team. Reflecting the importance of this is the Solar Racing Team's Business Lead, who is responsible for updating and distributing a quarterly newsletter, which updates sponsors on team progress.

D. Team Continuity and Sustainability Strategy

Historically, our team's membership has been small, and continuity was a concern for the executive board. This year, however, we're proud to have the largest active membership in the club's twenty-seven-year history. Most of our thirty new members are freshmen, with a mix of sophomores and juniors. Many—if not all—are highly involved, so we're confident the team won't need to worry about recruitment in the near future.

The beginning of the semester is a crucial opportunity to introduce potential members to the team. By assigning tasks to new recruits, we have significantly improved retention. We also promoted our Vertically Integrated Projects class, allowing students to earn credit for working on the boat, and publicized the PEP competition's workforce development program. Additionally, we continued participating in campus-wide events, and our weekly social gatherings have fostered team bonding and provided a chance to decompress.

E. Discussion and Self-Evaluation

The team has attended the competition for three years in a row, we attempt to improve on the systems that we had tested in competition for the previous year. At the beginning of the year, we reflected on the changes we worked on and what we wished to change about the boat, and communicated the changes that we wished to see with the subteams that would be responsible for each new change. The increased shop hours, collaborative days, and members allowed for more work to be done on the boat when needed. For the future, our members should work on setting hard deadlines for specific tasks to be completed, so there is time to ensure a functioning system before a wet test or PEP.

IV. Conclusions and Recommendations

The boat has faced many setbacks this year due to damage, shipping delays, and programming issues. Despite all this, our team has managed to build a lighter, faster vessel compared to all our previous iterations. The reduction of the complexity of our boat was pivotal in making it more reliable and responsive for the driver. When compared to last year, our boat was able to qualify and compete in PEP 2025 thanks to our team's hard work and perseverance in resolving these challenges.

Through multiple wet tests and a competition, our team has verified the integrity of our new updated systems on the boat. The handling and speed of our new bell crank steering system provide a more instantaneous turning of the outboard compared to our slower, heavier hydraulic system. Our wire management has become neater, and all wires are sheathed together along the rails of the boat, out of the driver's way. We have yet to see how well our system works with our solar panels mounted, as they were not used in our previous competition. Our DAQ is currently still a work in progress that we expect to be ready for Solar Splash, as it will be vital to the endurance race. We plan to test both during our last upcoming wet test before the competition.

During PEP, getting our motor controller and motor to function was the biggest challenge we faced. Our SinCos sensor on our motor was faulty and needed to be replaced to get our motor running again. After the replacement, the programming of the motor controller contained incorrect values for specific parameters related to our potentiometer and our forward/reverse switch. This prevented us from controlling our throttle, meaning our boat was stationary. Despite fixing these errors, our boat was performing suboptimally since it had yet to be tuned for our system. We hope to settle this during our last wet test as well by performing tuning on the water.

For next year, our team would like to create a new hull design, which we were unable to perform this year due to budget and time constraints. Our current hull has gone through multiple iterations for the last 7 years, and is inconvenient for our team to test due to its size and the location of our shop being in a basement. Next year, a senior design team will be responsible for the design and construction of the new hull. We hope this hull will be ready in time for next year's Solar Splash.

With this being our second year participating in two competitions, our team was far better prepared for the amount of work we would have to complete before the competition. This is also the largest our team has ever been, which has given us plenty of hands to assemble the boat on time. As a team, we could work to improve the delegation and management of tasks with how large our team has grown. Finally, our team has reached out to various new and returning sponsors for parts and discounts towards our boat.

V. References

[1] "Guidelines for Writing the Solar Splash Technical Report." *Solar Splash,* 17 June 2017, https://solarsplash.com/wp-content/uploads/2014/10/TechReportGuidelines_2017_06_17.pdf

[2] "Technical Report 2023." *Stony Brook Solar Racing Team,* https://docs.google.com/document/d/17ge6xxjl54iX8Huk6e-5z-z6_L8Rn2L2qniS5troPj0/edit

VI. Appendices

Appendix A: Battery Documentation

Documentations of the batteries we will be using

Width (mm/inch): 165/6.50
Height (mm/inch): 170/6.69
Total Height (mm/inch): 170/6.69
Approx. Weight (kg/lbs): 14.5/31.9

Performance Characteristics

Nominal Voltage: 12V

Nominal Capacity 77°F (25°C)
20HR: 50Ah
10HR: 45Ah
5HR: 42Ah
2HR: 34.6Ah

Internal Resistance
Fully Charged battery (25°C): Approx. 7mΩ

Capacity Affected by Temperature
40°C: 102%
25°C: 100%
0°C: 85%
-15°C: 65%

Self Discharge (25°C)
3 Month: Remaining Capacity: 91%
6 Month: Remaining Capacity: 82%
12 Month: Remaining Capacity: 65%

Nominal Operating Temperature
25°C ± 3°C (77°F ± 5°F)

Operating Temperature Range
Discharge: -15°C ~ 50°C (5°F ~ 122°F)
Charge: -10°C ~ 50°C (14°F ~ 122°F)
Storage: -20°C ~ 50°C (-4°F ~ 122°F)

Cyclic Charging Method
14.60 to 14.90V @25°C Temperature Compensation: -24mV/°C

Maximum Charging Current: 13.5A

Terminal Material: Copper

Maximum Discharge Current: 675A (5 sec.)

Designed Cycle Life (25°C)
500 Cycles @70% DOD

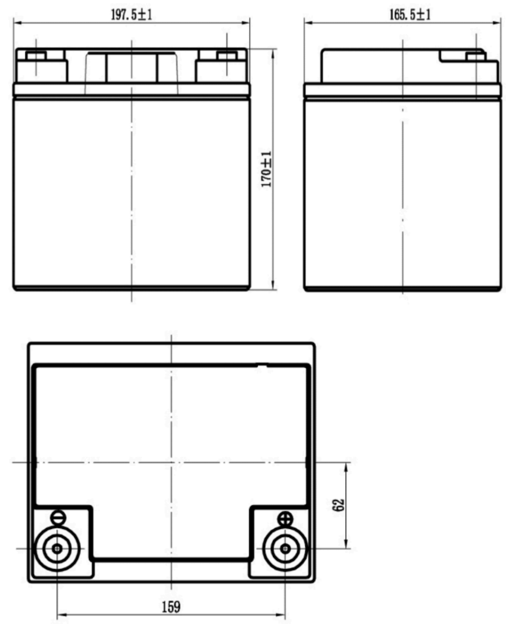
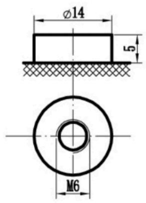
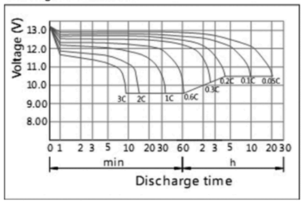


Fig. A1. *12V 50Ah Sealed Lead Acid Battery with F11 Terminals Documentation Sheet Page 1*

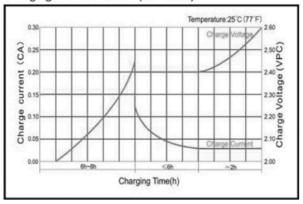


Terminal F11

Discharge Characteristics



Charging Characteristics (IUI mode)



Constant Current Discharge Characteristics

F. V/TIME	5min	10min	15min	30min	60min	2h	3h	4h	5h	10h	20h
9.60V	190	120	94.2	52.5	32.3	17.5	12.6	10.1	8.55	4.69	2.55
9. 90V	185	117	91.9	51.4	31.8	17.5	12.6	10.0	8.55	4.61	2.51
10.2V	177	112	88.5	49.8	31.0	17.3	12.4	9.93	8.47	4.61	2.51
10. 5V	169	107	85.5	48.7	30.4	17.0	12.3	9.90	8.44	4.53	2.44
10.8V	160	101	81.3	46.9	29.3	16.5	12.0	9.60	8.13	4.50	2.40

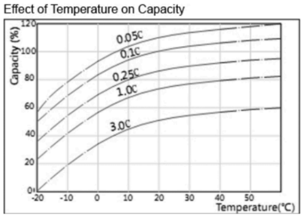
Constant Power Discharge Characteristics

F. V/TIME	5min	10min	15min	30min	60min	2h	3h	4h	5h	10h	20h
9.60V	2123	1347	1069	600	372	206	150	120	102	55.9	30.0
9. 90V	2062	1313	1043	588	368	205	149	119	102	55.5	30.0
10.2V	1977	1257	1001	574	359	203	147	119	101	55.5	29.9
10. 5V	1893	1207	975	559	352	199	146	118	100	55.1	29.6
10.8V	1797	1136	926	537	341	195	143	114	97.5	54.0	29.3

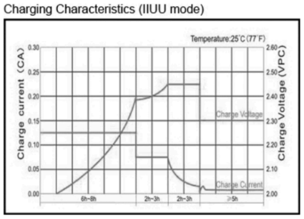
(Note) The above characteristics data are average values obtained within three charge/discharge cycles not the minimum values.

Fig. A2. *12V 50Ah Sealed Lead Acid Battery with F11 Terminals Documentation Sheet Page 2*

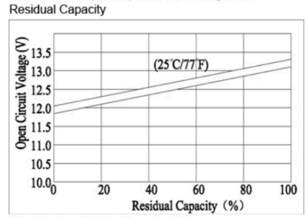
Effect of Temperature on Capacity



Charging Characteristics (IIUU mode)



The Relationship for Open Circuit Voltage and Residual Capacity



The Relationship for Charging Voltage and Ambient Temperature

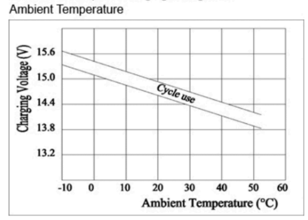
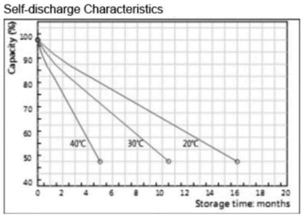


Fig. A3. *12V 50Ah Sealed Lead Acid Battery with F11 Terminals Documentation Sheet Page 3*

Self-discharge Characteristics



Cycle Life@5A Discharge

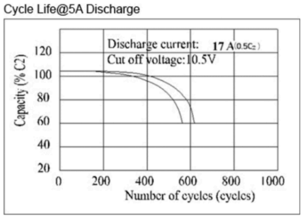


Fig. A4. *12V 50Ah Sealed Lead Acid Battery with F11 Terminals Documentation Sheet Page 4*

Appendix B: Flotation Calculations

Our boat weighs approximately 374 lbs across all sections, with the weight distribution among systems and items shown below in **Table B.1**.

The means of flotation are separated into two parts, one for each section of the boat. The bow and stern portions of the boat, due to being possible to separate into two distinct units, will each have their own means of flotation in the event of separation that will still satisfy the requirements of Rule 7.14.2. The weight distribution for both sections is seen below in **Table B.2**. For all calculations, the specific weight of water, γ_W, will be 62.4 lb/ft^3.

In the bow of the hull is an enclosed volume V_{Bow} of 2.47 ft^3 filled with closed-cell expanding polyurethane (PU) foam with a specific weight γ_{PU} of 2 lb/ft^3. Multiplying V_{Bow} by the difference of specific weights of water and foam, we get a buoyant force $F_{b,\, Front}$ of 149.188 lb.

$$
\begin{aligned}
F_{b,\, Front} &= V_{Bow} \cdot \left(\gamma_W - \gamma_{PU}\right) \\
&= 2.47\, ft^3 \cdot \left(62.4\, lb/ft^3 - 2\, lb/ft^3\right) \\
&= 149.188\, lb
\end{aligned}
$$

The weight of the bow is 36 lb, and with the additional 20% safety factor, results in a minimum required buoyant force of 43.2 lb. Due to the buoyant force for the bow exceeding the minimum required flotation of the boat section with additional safety factor, Rule 7.14.2 is satisfied for the bow.

Along the sides of the stern section, there are four (4) foam blocks made of Owens Corning FOAMULAR 250 extruded polystyrene (XPS) foam, and four (4) flotation airbags made of a durable polymer. A single flotation airbag displaces a volume V_{Airbag} of 0.6 cu. ft. and has a weight w_{Airbag} of 0.33 lb. Floatation force can be calculated by subtracting the weight of the airbag from the weight of water it displaces, resulting in a total buoyancy contribution of 148.44 lb from the four airbags, as can be seen below.

$$
\begin{aligned}
F_{b,\, Airbags} &= 4\left(\left(V_{Airbag} \cdot \gamma_W\right) - w_{Airbag}\right) \\
&= 4 \cdot \left(\left(0.6\, ft^3 \cdot 62.4\, lb/ft^3\right) - 0.33 lb\right) \\
&= 148.44\, lb
\end{aligned}
$$

All foam blocks, combined, make up for a displacement volume V_{Blocks} of 5.10 ft^3. With a specific weight γ_{XPS} of 1.55 lb/ft^3, by multiplying the volume by the difference between specific weights of water and foam, we get a contribution of 310.335 lb of buoyancy force for all foam blocks.

$$
\begin{aligned}
F_{b,\, Blocks} &= V_{Blocks} \cdot \left(\gamma_W - \gamma_{XPS}\right) \\
&= 5.10\, ft^3 \cdot \left(62.4\, lb/ft^3 - 1.55\, lb/ft^3\right) \\
&= 310.335\, lb
\end{aligned}
$$

The weight of the stern is 338 lb, and with the additional 20% safety factor, results in a design weight of 405.6 lb. The combined buoyancy force of the airbags and all foam components is 458.775 lb. Due to the combined buoyancy force exceeding the design weight of the boat section with the safety factor, Rule 7.14.2 is satisfied for the stern section.

Table B.1: *Weight distribution for the boat across all systems.*

System/Item Name	**Weight (lb)**
Hull and Frame	98
Drivetrain and Steering	125
Electronics and Solar Array	41
Batteries	98
Accessories and Equipment	12
Total	*374*

Table B.2: *Weight distribution for the distinct separable sections of the boat.*

Hull Section Name	**Weight (lb)**
Bow Section	36
Stern Section	338
Total	*374*

Appendix C: Proof of Insurance

EXCELSIOR

OFFICE OF GENERAL SERVICES
BUREAU OFRISK AND INSURANCE MANAGEMENT

TO: Whom it may Concern

FROM: Tomlynn Yacono
Director, Bureau of Risk and Insurance Management

SUBJECT: *Statement of Self Retention*

The General Liability exposures of the State of New York as well as those of the State Agencies are self retained. Suits for bodily injury and property damage are brought in the NY State Court of Claims, which is supported by a multi-million dollar annual appropriation.

Employees are protected against suits under Public Officers Law Section 17 for actions or alleged actions that occur while they are acting within the scope of their employment.

If there are any questions or further information is needed, please do not hesitate to contact the OGS Bureau of Insurance at (518) 474-4725.

Appendix D: Team Roster

The list of members who have contributed to the creation of the boat.

First Name	Last Name	Major	Year	Team Role
Wesley	Ng	Electrical Engineering	Senior	President / Electrical Lead
Matthew	Leo	Computer Engineering	Senior	Vice President
Steven	Chen	Mechanical Engineering	Junior	Secretary / Mech Lead
Nick	Zhang	Mechanical Engineering	Junior	Treasurer
Bill	Zhang	Computer Science	Senior	Software Lead
Daniel	Lew	Computer Science and Applied Mathematics	Sophomore	Member
Suhayla	Yousuf	Computer Engineering	Freshman	Event Coordinator
Allen	Li	Economics	Freshman	Member
Wilson	Lin	Mechanical Engineering	Freshman	Member
Kumpu	Ide	Mechanical Engineering	Graduate	Member
Nancy	Zhu	Electrical Engineering	Freshman	Member
Owen	Chen	Electrical Engineering	Freshman	Member
Chloe	Jiang	Mechanical Engineering	Freshman	Member
Maggie	Tan	Mechanical Engineering	Freshman	Member
Md	Zaman	Information Systems	Senior	Webmaster
Ryan	Tang	Mechanical Engineering	Junior	Member
Daniel	Zou	Mechanical Engineering	Freshmen	Member
Anton	Yanaky	Computer Science	Freshman	Member
Ivan	Yu	Mechanical Engineering	Freshman	Graphic Designer
Ryan (Jianyang)	Tang	Applied Mathematics	Freshman	Member
Wing	Chan	Computer Engineering	Freshman	Historian
Kayla	Rodriguez	Electrical Engineering	Freshman	Member

Appendix E: Schematics

Diagrams for the wiring of the motor controller to the motor.

inmotion

8.11.3 ACS 35 pin standalone controller typical wiring

NOTE

The motor temperature sensor must be isolated from other signals in the system to avoid malfunction caused by incorrect wiring.

Connect the motor temperature sensor to pins MOTOR_TEMP+ and MOTOR_TEMP- only.

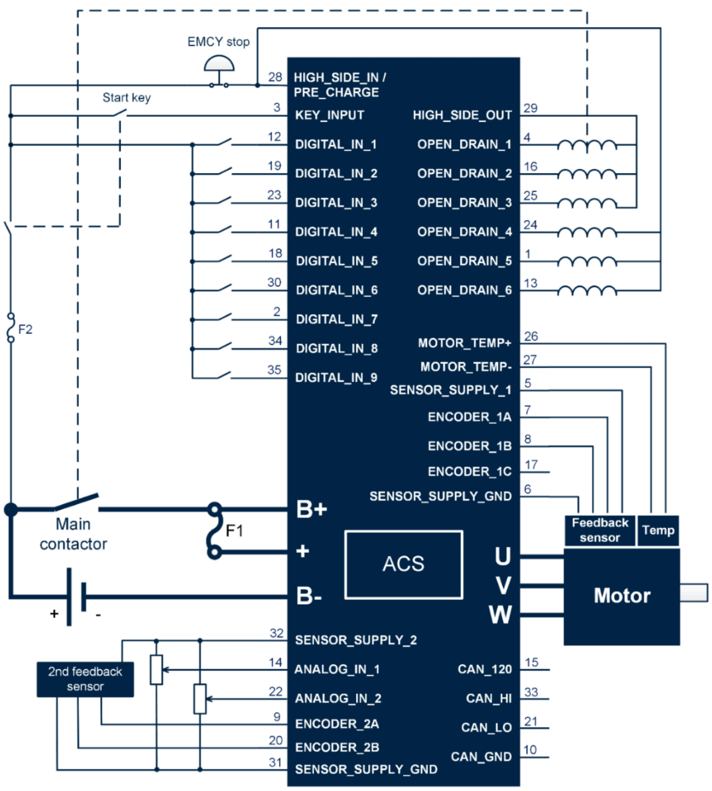


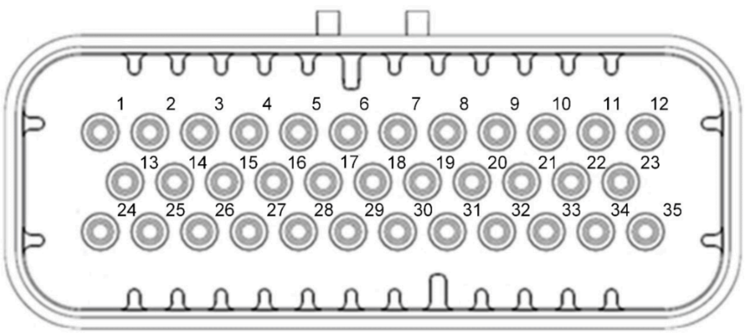
Figure 18 Typical wiring of ACS 35 pin standalone I/O controller with 12 V feedback sensor

43 Product Manual ACS Gen 7

Fig. E1. *inMotion provided motor controller wiring diagram for ACS Gen 7 MCs.*

inmotion

9.4 ACS MD 35 pin pinout



Pin	Name	Chapter	Pin	Name	Chapter
1	OPEN_DRAIN_OUT_2	9.10	19	ENCODER_IN_3B	9.9
2	DIGITAL_IN_3	9.6	20	ENCODER_IN_2B	9.9
3	KEY_INPUT	9.5	21	CAN_LOW	9.15
4	OPEN_DRAIN_OUT_1	9.10	22	CAN_LOW	9.15
5	SENSOR_SUPPLY_1 (+12 V)	9.13	23	DIGITAL_IN_2	9.6
6	SENSOR_SUPPLY_GND	9.14	24	OPEN_DRAIN_OUT_4	9.10
7	ENCODER_IN_1A	9.9	25	OPEN_DRAIN_OUT_3	9.10
8	ENCODER_IN_1B	9.9	26	MOTOR_1_TEMP+	9.12
9	ENCODER_IN_2A/_1C	9.9	27	MOTOR_1_TEMP-	9.12
10	CAN_GND	9.15	28	HIGH_SIDE_IN/PRE_CHARGE	9.11
11	CAN_GND	9.15	29	HIGH_SIDE_OUT	9.11
12	DIGITAL_IN_1	9.6	30	ENCODER_IN_4A/_3C	9.9
13	DIGITAL_IN_6	9.6	31	SENSOR_SUPPLY_GND	9.14
14	DIGITAL_IN_5	9.6	32	SENSOR_SUPPLY_2 (+12 V)	9.13
15	MOTOR_2_TEMP+	9.12	33	CAN_HIGH	9.15
16	MOTOR_2_TEMP-	9.12	34	CAN_HIGH	9.15
17	ENCODER_IN_3A	9.9	35	CAN_120R	9.15
18	ENCODER_IN_4B	9.9			

Figure 33 ACS MD 35 pin pinout

Product Manual ACS Gen 7 70

Fig. E2. *inMotion provided 35 pin pinout for controller and encoder signals.*

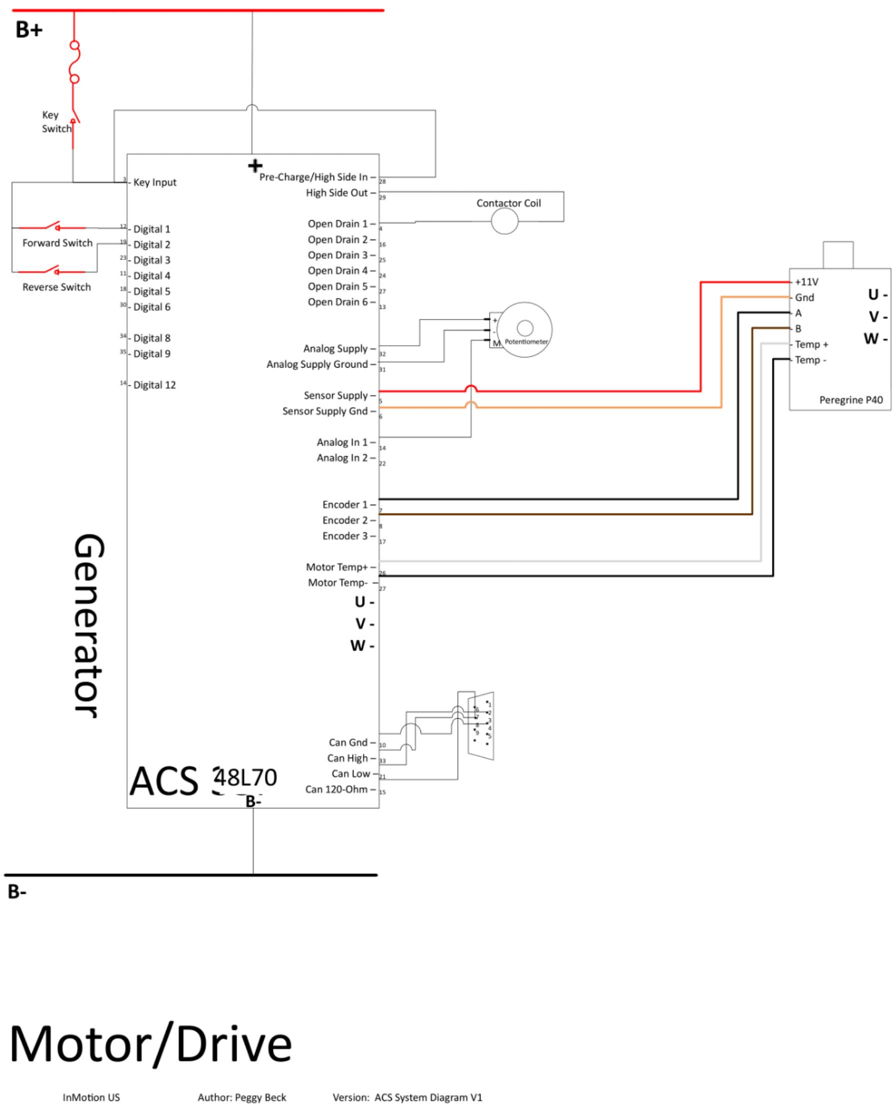


Fig. E3. System Overview Wiring Diagram.

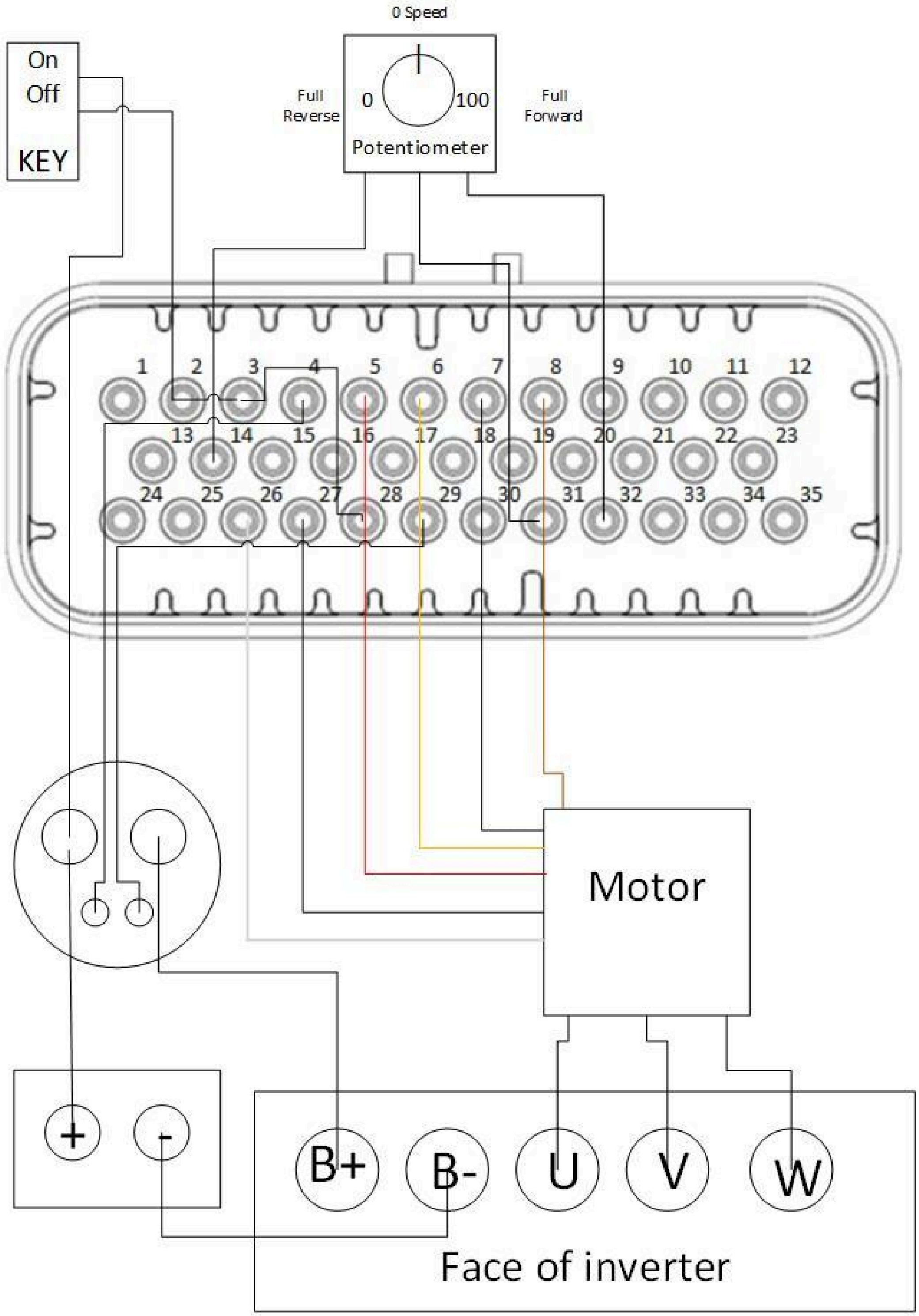


Fig. E4. *35 pin pinout wiring.*

8.11.5 Connections

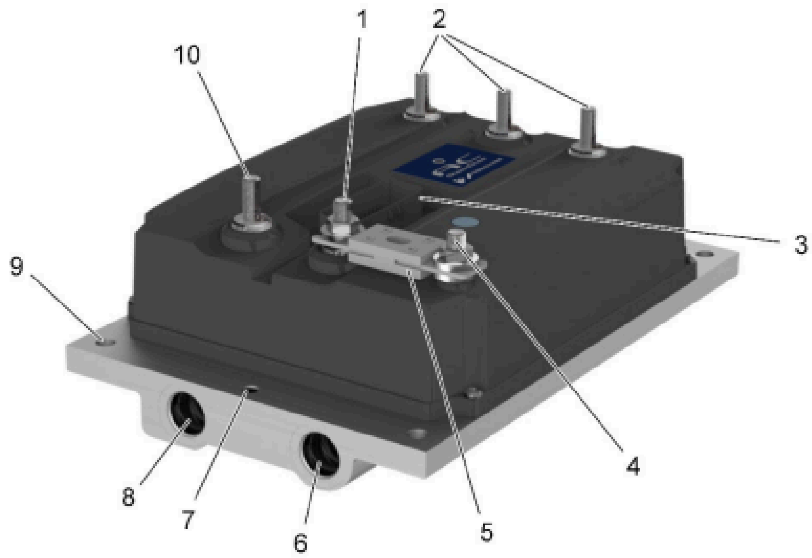


Fig. E5. *Motor controller diagram*

Appendix F: DAQ Code

Code that was written for the Raspberry Pi to communicate with the ADS1115s.

Display.py

```python
import tkinter as tk

class DisplayApp:
    def __init__(self, root: tk.Tk, adcs):
        self.root = root
        root.configure(background='black')
        self.master = root
        master = self.master

        # self.mainframe = tk.Frame(self.master, padx='10', pady='10')
        # self.measures = [tk.Label(self.mainframe, text=f'V: 0') for i in range(adcs)]
        root.attributes('-fullscreen', True)  # Fullscreen mode
        root.bind('<Escape>', self.quit_fullscreen)  # Bind escape key to exit fullscreen

        self.m_labels = []

        self.section1 = tk.Frame(master, background='black', pady=50)
        self.section1.pack(side=tk.TOP, pady=20)
        self.label1a = tk.Label(self.section1, text="0 V", font=("Helvetica", 108), fg='#fff', bg='#000')
        self.label1a.pack(side=tk.LEFT, padx=10)

        self.label1b = tk.Label(self.section1, text="0", font=("Helvetica", 108), fg='#fff', bg='#000')
        self.label1b.pack(side=tk.LEFT, padx=10)

        self.m_labels.append((self.label1a, self.label1b))

        self.section2 = tk.Frame(master, background='black', pady=50)
        self.section2.pack(side=tk.TOP, pady=20)
        self.label2a = tk.Label(self.section2, text="0 V", font=("Helvetica", 108), fg='#fff', bg='#000')
        self.label2a.pack(side=tk.LEFT, padx=10)

        self.label2b = tk.Label(self.section2, text="0", font=("Helvetica", 108), fg='#fff', bg='#000')
        self.label2b.pack(side=tk.LEFT, padx=10)

        self.m_labels.append((self.label2a, self.label2b))

        self.section3 = tk.Frame(master, background='black', pady=50)
        self.section3.pack(side=tk.TOP, pady=20)
        self.label3a = tk.Label(self.section3, text="0 V", font=("Helvetica", 108), fg='#fff', bg='#000')

        self.label3a.pack(side=tk.LEFT, padx=10)
        self.label3b = tk.Label(self.section3, text="0", font=("Helvetica", 108), fg='#fff', bg='#000')
        self.label3b.pack(side=tk.LEFT, padx=10)

        self.m_labels.append((self.label3a, self.label3b))

    def update_measurements(self, data):
        for i, (v,c) in enumerate(data):
            self.mod_label(self.m_labels[i][0], v, 'V')
            self.mod_label(self.m_labels[i][1], c, 'Amps')
        pass

    def mod_label(self, label, measurement, unit):
        label.config(text=f"{measurement} {unit}")

    def quit_fullscreen(self, event):
        self.root.attributes('-fullscreen', False)
        self.root.destroy()
```

Fig. F1. *The code for displaying the battery voltage on the dashboard screen.*

Logger.py

```python
# import matplotlib.pyplot as plt
# import matplotlib.animation as animation
import os
import tkinter as tk
# import pandas as pd
import threading
import signal
import glob
from serial import Serial
from voltage_reader import VoltageReader
from display import DisplayApp

class Logger:

    app_root = None

    def get_pi_port():
        res = glob.glob('/dev/ttyACM*')
        return res[0]

    PI_PORT = get_pi_port()
    WINDOWS_PORT = 'COM3'
    reading_thread = None

    def __init__(self):
        pass

    def update_plot(readings, lines_data, axs):
        print(readings)

        for i in range(len(readings)):
            lines_data[i].set_data(readings[i][0], readings[i][1])
            axs[i].set_xlim(min(readings[i][0]), max(readings[i][0]))
            axs[i].set_ylim(min(readings[i][1]) - 1, max(readings[i][1]) + 1)

    # def plot_animation(i, v_data: VoltageReader, lines_data, axs):
    #     v_data.readings.use_readings(update_plot, lines_data, axs)

    def int_handler(self, s,t):
        # print(s, t)
        print('Stopping')
        VoltageReader.stop_reading.set()
        Logger.reading_thread.join()
        Logger.app_root.destroy()
        # plt.close()

    def main(self):
        adcs = 3

        # defines DISPLAY variable for tkinter if needed
        if os.environ.get('DISPLAY','') == '':
            print('no display found. Using :0.0')
            os.environ.__setitem__('DISPLAY', ':0.0')

        Logger.app_root = tk.Tk()
        app = DisplayApp(Logger.app_root, 3)

        arduino = Serial(port=self.PI_PORT, baudrate=9600, timeout=1)
        # fig, axs = plt.subplots(2, 1)
        # lines_data = [axs[i].plot([], [])[0] for i in range(plots)]

        v_data = VoltageReader(arduino, adcs, app)

        Logger.reading_thread = threading.Thread(target=v_data.read_voltages)
        Logger.reading_thread.start()

        signal.signal(signal.SIGINT, self.int_handler)
        # reading_thread.join()

        # fig.suptitle('Voltage Measurements')
        # frame_count = 0

        # plot_ani = animation.FuncAnimation(
        #     fig,
        #     plot_animation,
        #     fargs=[v_data, lines_data, axs],
        #     interval=1000,
        #     save_count=frame_count,
        #     cache_frame_data=True
        # )
        # plt.plot([1], [2])
        # plt.show()
        Logger.app_root.mainloop()

    def start(self):
        self.main()

if __name__ == '__main__':
    Logger().start()
```

PDF document made with CodePrint using [Prism]

Fig. F2. *The main file used to run the code on the pi.*

Record.py

```python
import csv
import os
import time

class Recording:

    def __init__(self, cols) -> None:
        # remove newline when deployed
        # os.path.dirname(os.path.abspath(__file__))
        dir_path = os.path.dirname(os.path.abspath(__file__))
        self.csv = open(dir_path + f'/logs/{time.time()}-readings.csv', mode='w', newline='')
        self.csv_writer = csv.writer(self.csv)
        self.cols = cols
        self.setup_csv()

    def setup_csv(self):

        header = ['Time']

        for i in range(self.cols):
            header.append(f'Voltage{i+1}')
            header.append(f'Current{i+1}')
        self.csv_writer.writerow(header)

    def write_data(self, data):
        row = [time.time()]
        for v,c in data:
            row.append(v)
            row.append(c)

        # Flushes data out to file immediately
        self.csv_writer.writerow(row)
        self.csv.flush()
        os.fsync(self.csv)

    def close(self):
        self.csv.close()
        print('Closed')
```

Fig. F3. *Code for saving the data within the raspberry pi.*

Voltage_reader.py

```python
# import time
import threading
# import pandas as pd
from serial import Serial
from record import Recording
from display import DisplayApp


class Readings:

    def __init__(self, adcs, app: DisplayApp) -> None:
        self.sema = threading.Semaphore()
        self.counter = 1
        self.app = app
        # self.readings = pd.DataFrame()
        # df_index = pd.Index([i for i in range(plots)])

        # self.readings.set_index(df_index)
        # [timestamp, voltage]
        self.readings = [
            [[0],[0]] for _ in range(adcs)
        ]
        self.csv_record = Recording(adcs)

    def update_readings(self, filtered_data: list):
        # print('filtered:', filtered_data)
        self.sema.acquire()
        try:
        # updating readings
            # for i in range(len(filtered_data)):
            #     reading = self.readings
            #     reading[i][0].append(self.counter)
            #     reading[i][1].append(filtered_data[i])
            #     if (len(reading[i][0]) > 10):
            #         reading[i][0] = reading[i][0][-10:]
            #         reading[i][1] = reading[i][1][-10:]
            # self.counter += 1

            # recording readings
            # filtered_data.insert(0, time.time())
            # pdb.set_trace()
            # print('filtered data: ', filtered_data)
            self.csv_record.write_data(filtered_data)
            self.app.update_measurements(filtered_data)

        except:
            print('Could not update readings')
            self.sema.release()
            # self.shutdown()

        self.sema.release()

    def shutdown(self):
        print('Stopping logging')
        self.csv_record.close()

    def use_readings(self, action, lines_data, axs):
        self.sema.acquire()
        action(self.readings, lines_data, axs)
        self.sema.release()

    def __str__(self):
        return str(self.readings)

class VoltageReader:

    stop_reading = threading.Event()

    def __init__(self, arduino: Serial, adcs, app: DisplayApp) -> None:
        self.counter = 0
        self.readings = Readings(adcs, app)
        self.arduino = arduino
        self.adcs = adcs
        # self.exp_cols = adcs * 2

    def line_filter(self, v):
        if v != '':
            return True
        return False

    # could set the serial to have no timeout
    def read_voltages(self):
        arduino = self.arduino
        arduino.reset_output_buffer()
        arduino.reset_input_buffer()

        while not VoltageReader.stop_reading.is_set():
            read_lines = []
            while (len(read_lines) < 3):
                line = arduino.readline()
                read_lines.append(line)

            lines = [line.decode().strip() for line in read_lines]
            print(lines)
            filtered_lines = [(float(v), round(float(v) / 0.05, 2))
                              for v in filter(self.line_filter, lines)]


            # while (not not read_line):
            #     print(lines)
            #     lines.append(read_line)
            #     time.sleep(0.01)
            #     try:
            #         read_line = arduino.readline().decode().strip()
            #     except:
            #         print('Could not read line')
            #     if not read_line:
            #         lines.append(read_line)

            #     if VoltageReader.stop_reading.is_set():
            #         break

            if (len(filtered_lines) == self.adcs):
                # filtered_lines = [float(v) for v in filter(self.line_filter, lines)]
                # print(filtered_lines)
                self.readings.update_readings(filtered_lines)
            if VoltageReader.stop_reading.is_set():
                break

        self.readings.shutdown()
```

PDF document made with CodePrint using Prism

Fig. F4. *The code to read the voltage from the Arduino.*

Appendix G: Hull drag analysis

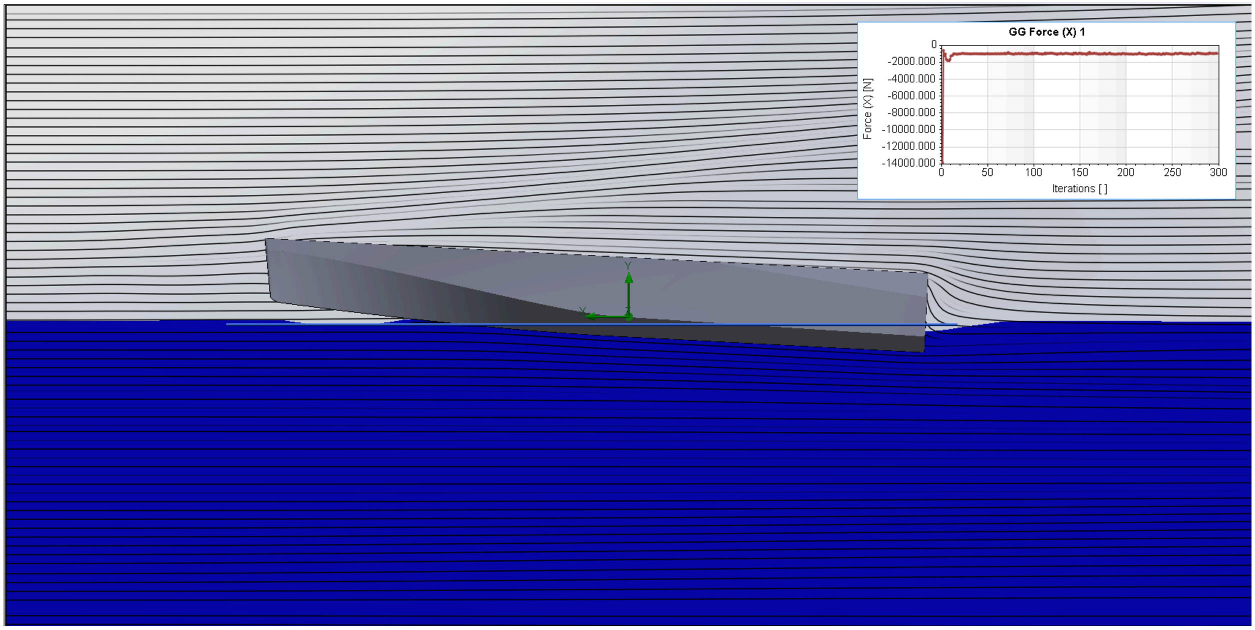


Fig. G1. *Solidworks free surface analysis of hull at 3-degree incline at lowest anchored point*