Foilboat Project Summary

For 8 months between January and August 2021, I worked for Greg Marshall Naval Architecture designing a hydro foiling boat. As a strictly research and development craft, the primary goals involved testing a possible system architecture for such a craft. Hydrofoiling boats are desirable because they operate with much less drag than conventional ships. This would be advantageous for next generation electric vessels which are limited by battery pack volume and weight.

Project Goals and Constraints

There were a few primary goals:

- Build a hydrofoiling catamaran that maintained bilateral symmetry.
- Use electric propulsion
- Carry at least one human operator
- Use fly-by-wire steering and control.
- Maximize range

There were some constraints as well:

- System must use Waydoo electric motors
- System must be flight ready in 3 months.

Mechanical Design

After researching existing hydrofoiling vessels a fully submerged foil architecture was chosen instead of angled foils partially submerged foils. Although partially submerged foils provide some passive roll stability, they are less efficient.

To assess the ease of control during the initial design phase, Python IVP simulations were performed regularly as the design changed to assess the effects of different aspects of the design.

After a few different design concepts such as large central wing with a multiactuated rudder and surface piercing foils with surfaces, a balanced four foiled configuration was selected. Two identical kayaks were chosen as the hulls for this project. In total the foilboat is 6m long, 2.25m wide and weighs 150kg.

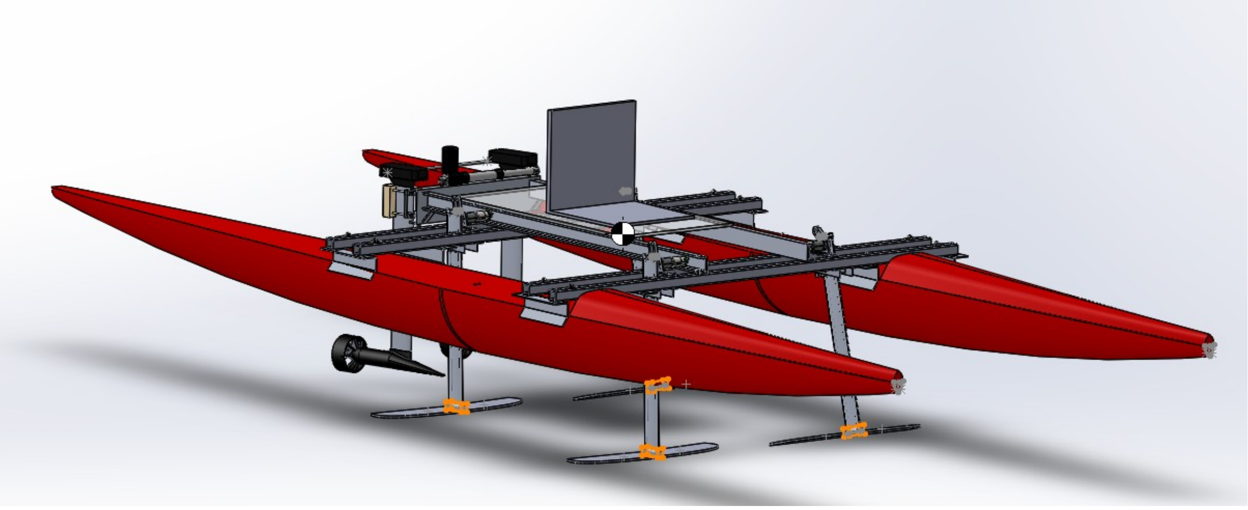


Figure 1: Foilboat with four fully actuated foils and the rudder motors in the rear

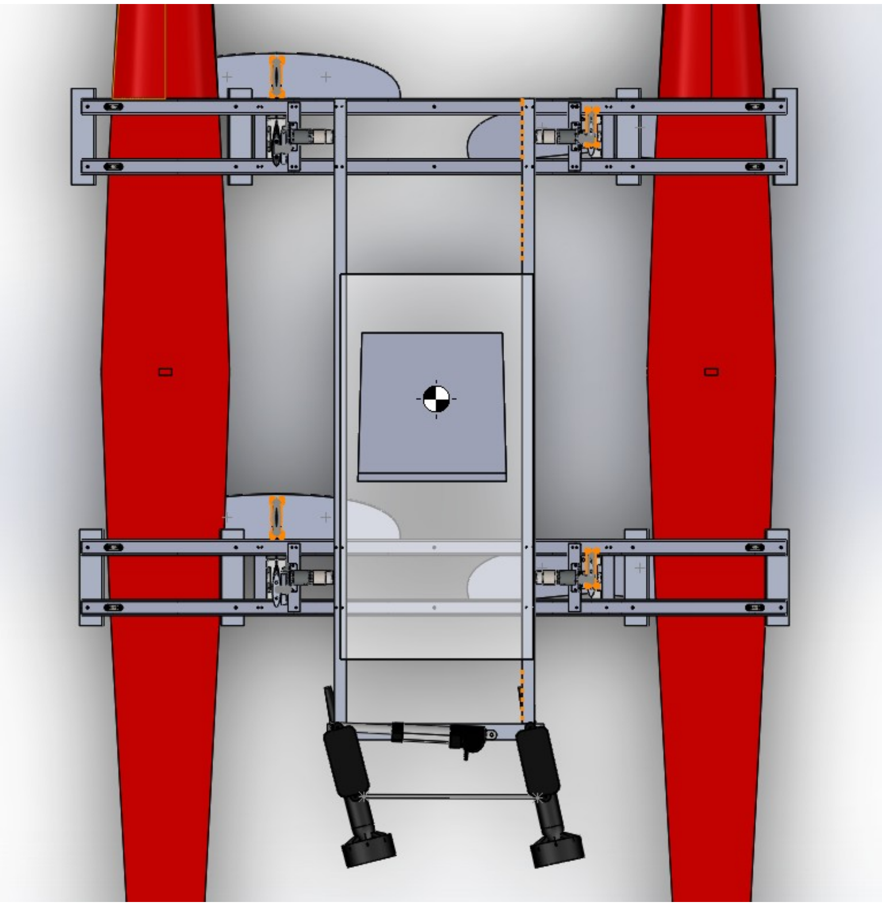


Figure 2: top view

FEA was performed using filament elements for all of the major chassis components to ensure they would not break, nor deflect significantly under the dynamic loading of the system. The chassis is entirely aluminum with some welded joints and others held together with bolts. An electric linear actuator was attached to the thrust motors in the rear to steer them. All four foil subsystems used a served crank arm to change the angle of attack of the foils independently. This independent control allows the boat to vary the lift force, pitch moment and roll moment to stabilize the boat while it is underway.

Figure 3: Close-up view of the foil crank assembly

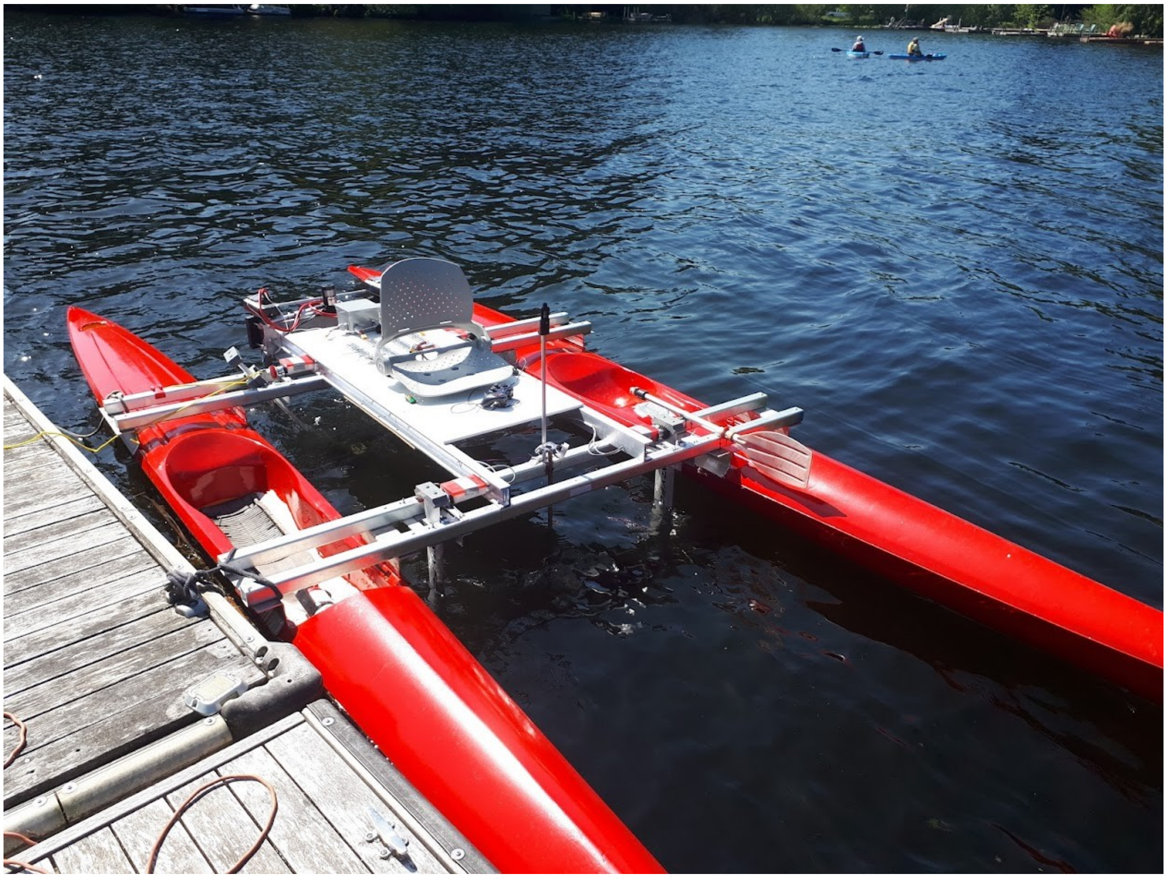
The motor is chosen to have a high torque output at low speed. The built in encoder allows the position of the motor to be reliably determined. By extension this corresponds to the angle of attack of the foil. The motor must overcome the torque cause by drag while the boat is in motion as well as to accelerate the mast and foil which are about 75cm long. The aluminum components were all CNC milled custom for this project.

In addition, significant bearing calculations were required to ensure that the thrust of the motors, which is applied 60cm below the bearing block would be carried safely. The Waydoo motors were carried successfully and the bearings turned smoothly.

Figure 4: Thrust motors mounted to the rear chassis block

The completed boat was tested in Prospect Lake.

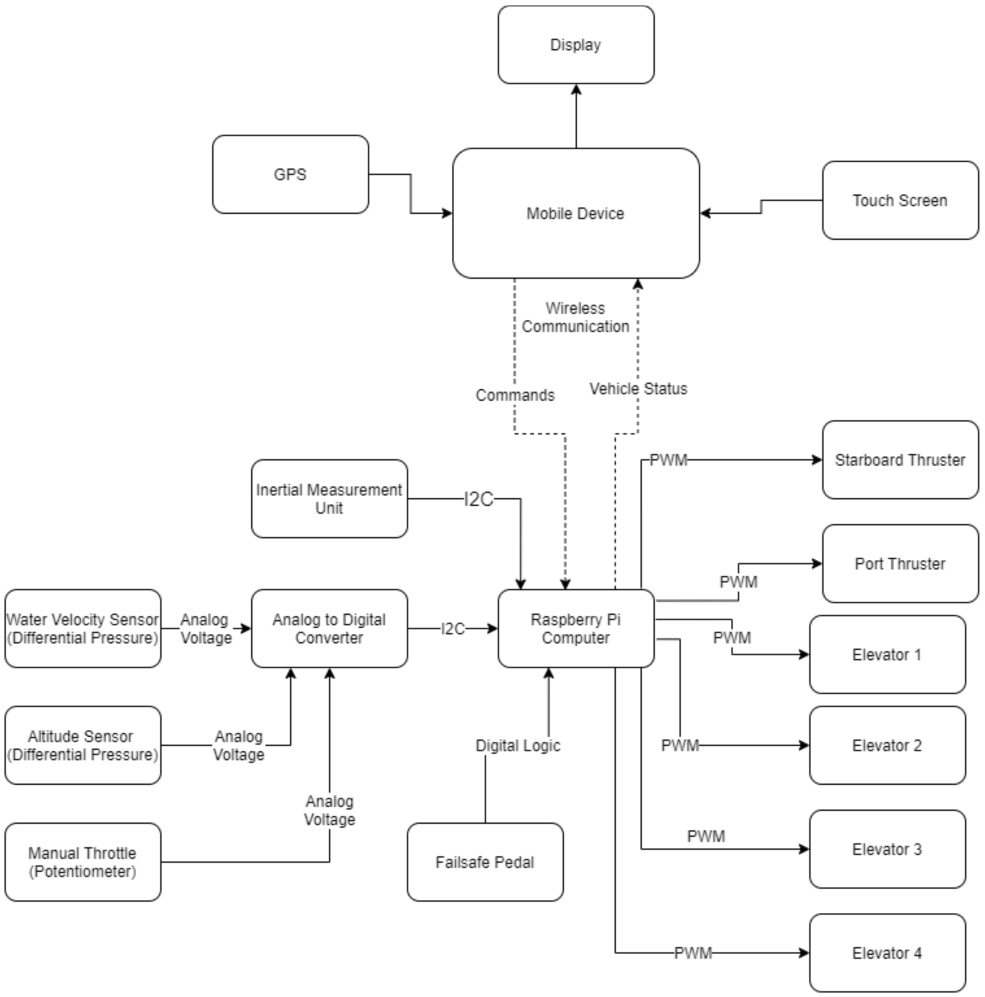
Figure 5: Rear quarter image complete with rider

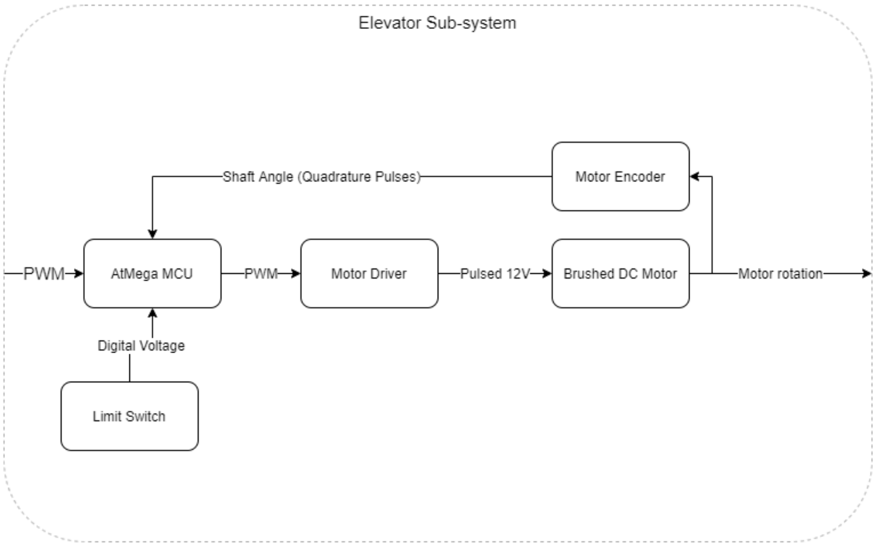


Electrical & Software Systems

The foilboat had many simultaneous responsibilities. The primary one was to manage sensor data and choose an appropriate foil angle to maintain stable flight. However, it also needed to handle fly-by-wire input requests through a handled controller, log data, and send information to the operator to the web app hosted on a mobile device.

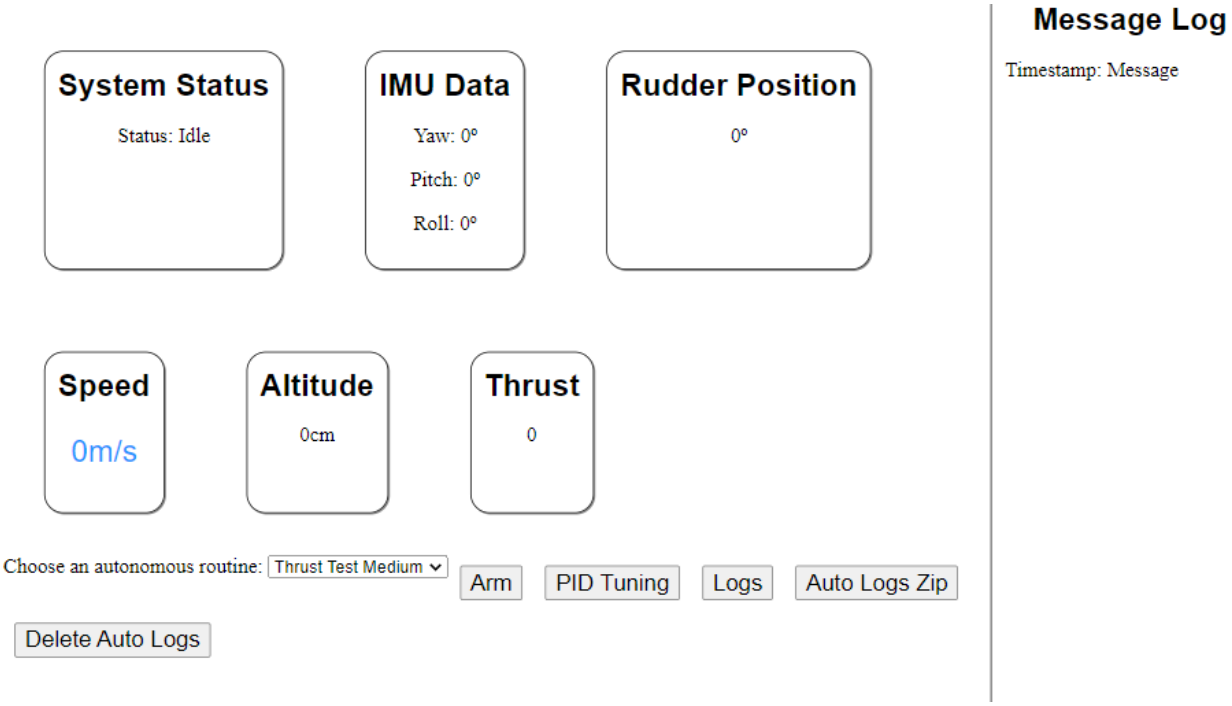
The overall system architecture is shown below.





To manage all of these interests a Raspberry Pi single board Linux computer was chosen. Python’s Asyncio library was leveraged to allow for parallel tasks to be scheduled, created and cancelled. The web framework Quart was used to develop the mobile web application.

A web app was chosen because it provided the most flexibility. By creating a local wifi network hosted by the Raspberry Pi any device can connect.



Status: Idle

Yaw: 0°

Pitch: 0°

Roll: 0°

Rudder Position: 0°

Timestamp: Message

Speed: 0m/s

Altitude: 0cm

Thrust: 0

Choose an autonomous routine: Thrust Test Medium

Figure 6: Web Server sample output

In this case an Android cell phone was used onboard to provide a diagnostic interface for choosing autonomous routines and displaying information. However, this system works with iOS or laptops as well.

The decision to distribute the control for the foils was based on improving performance. Using a dedicated microcontroller for each foil meant that that the system could run at 1kHz, reading encoders and dynamically updating motor power to hold the foil still. The Raspberry Pi simply sent a command PWM signal, and the servo box would hold it. This was quite successful.

The software was built around an asynchronous finite state machine which leveraged Python’s function passing and decoration to define the functionality for states and the acceptable transitions between them.

Electrically, the system had two different voltage levels used to drive the system. A high voltage set of four 52V lithium batteries were connected to the thrusters. The batteries could provide 120A continuously 300A burst for the 2 6kW motors. In normal operation. The motors were usually operated at about 2kW. Each thruster had a 3.3V PWM control signal. All of the sensors and the foil motors ran off of a separate 12V system. That system had a 50A peak capacity. Under normal loads only about 2A was used, but in the event of a foil motor stall, up to 45A could be required. Separating the two systems reduced the electrical noise introduced into the system. However, this electrical noise ultimately proved to be a significant problem. Although the two circuits were electrically coupled the control electronics on the thruster side were continually destroyed by electrical noise. High voltage noise travelling back up the PWM wires destroyed multiple microcontrollers.

Results

The foilboat was ultimately successfully for short periods of time. The quad foil system was effective but the electrical noise caused by the thrusters ultimately the boat from every achieving continuous level flight. The thrusters usually died after about 15-20seconds of used. Sometimes a reboot would start them again, but the control objectives could not ultimately be realized without thrust. However short flights were achieved consistently. With new thruster electronics level flight would be possible.

Figure 7: The foilboat while airborne

Lessons Learned

I learned many different things working on this project. Firstly, because I had little direct engineering support, I gained a better system for managing tasks and staying productive. Firstly, I found it effective to “self-parallelize” my tasks. When I encountered a challenge on one thing like how to cancel autonomous routines safely, I could pivot and work on something else instead. While I would solder boards or machine new components, the original problem percolated in my mind and a solution would often occur. One good example of this was the foil init sequence. Originally, I used a second enable/disable line to tell the MCU to stabilize foil or to leave them slack and unpowered. However, during the boot sequence of the computer, these lines would sometimes take on indeterminate voltages causing unpredictable foil movements. Instead of spending a long time on this problem, I worked on improving the autonomous routines interface in the web app. While working on something else, I had the idea to modulate the PWM frequency to encode system state. This system worked reliably, and the boat was safer with no unpredictable movements.

In addition, I learned the value of components that come with datasheets/support. The Waydoo thrusters were purchased before I began my term but were approximately $2000 less each than an equivalent 6kW brushless system. However, in the end I spent so many hours trying to debug the motors without any useful customer support that it was not worth it; particularly because that was ultimately the failing part of the project.