# Aquapod: The Design of Small Amphibious Tumbling Robot

**A THESIS**
**SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL**
**OF THE UNIVERSITY OF MINNESOTA**
**BY**

**ANDREW JAMES CARLSON**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS**
**FOR THE DEGREE OF**
**MASTER OF SCIENCE**

**Nikos Papanikolopoulos**

**June, 2012**

**Aquapod: The Design of Small Amphibious Tumbling Robot**

**by ANDREW JAMES CARLSON**

## ABSTRACT

In the area mobile of robots, real-world applications present a number of challenges in regard to traversability. This term represents a robot's ability to navigate irregularities in an environment and is directly related to the scale of the robot with respect to the terrain. Therefore, a robot with a high mobility-to-size ratio is desired among smaller platforms. Tumbling locomotion is a relatively new and unexplored method of mobility that has shown promise in pushing beyond the limitations of wheeled robots. A tumbling robot is any platform that uses its body to assist with gripping and moving over terrain.

The project presented in this thesis expands on the idea of a tumbling robot by placing it in the context of environmental monitoring. The original research platform, named the Adelopod, was the starting point for the work described herein. From this robot, a number of design requirements were produced that described the desired qualities in an environmental monitoring robot. In addition to being more rugged, more mobility was included in the form of buoyancy control to move vertically within the water column.

In order to accomplish the design requirements, three distinct prototypes were developed throughout the course of the project. The first prototype was a simple proof-of-concept, radio-controlled device to ensure the mechanical systems of the robot were functional and waterproof.

Once the concept was proven, a new task to collect water samples was presented. This required developing the electronic components, improving the chemical resistance, and increasing maximum achievable depth, which resulted in the second prototype iteration. Finally, the last prototype made minor improvements to the previous designs by lessening the number of ports, removing external wires, and making a modular backpack system.

Though none of the designs built in this thesis were meant to be full-fledge monitoring devices, future iterations could be used in interesting scenarios. The buoyancy control unit developed for this robot allows many unique applications. Stream travel, lake deployment, tide advancement, and mud flat navigation are just some of the areas where the robot would be useful. In the meantime, the prototypes developed herein will further the research in path planning and locomotion which is still largely unsolved, especially for dynamic terrain.

This thesis details the development process of creating these prototypes. Throughout every iteration the challenges and lessons learned from previous designs were addressed and improved. Careful consideration was taken at every stage in the design to ensure the platform worked as expected and furthered the understanding of tumbling robots.

# Acknowledgements

I would like to thank all of the people at the Center for Distributed Robotics for their support and help. In particular, I would like to thank Brett Hemes for all the information and work relating to the Adelopod. Also to Alex Kossett for the continued design advise; this project would not have succeeded without his help.

Thank you to Dario Canelon-Suarez, Justin Dancs, Sandeep Dhull, Apostolos Kottas, Nick Walczak, and Joshua Fasching who put in a tremendous amount of time and effort into designing the Rapid MRI Aquapod while I was abroad. And again to Dario, Nick, and Josh for their continued support on subsequent designs.

Thank you to the undergraduate students that helped with various aspects of testing and assembly. In particular, thank you to Alexander Hambrock.

Thank you to Richard Voyles for his support and backing of the National Science Foundation Rapid MRI Grant. And to Steven Pennings and his team in Houston for their input on the project.

A big thank you to Nikos for his unrelenting work behind the scenes and for being one of the best advisors a student could ask for.

Finally, I would like to thank my parents and family for their support and simply being amazing.

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

This thesis presents the Aquapod: a small, amphibious, tumbling robot to be used in environmental monitoring. The Aquapod is the second generation of a research robot called the Adelopod [2]. Developed at the Center for Distributed Robotics, the Adelopod uses two rigid arms to push its body out of equilibrium and induce a tumble to move forward. By using its own body as a method of moving, the Adelopod increases its mobility-to-size ratio and can navigate more dynamic terrain than wheeled robot of similar size. However, the increase in mobility with low hardware complexity comes at the cost of non-intuitive motion planning.

This project took the basic design of the Adelopod and ruggedized it to be water-proof. Additional systems, such as the dual-mode buoyancy control, were also added to the robot to assist in environmental tasks. This dual-mode system allows for some unique and innovative scenarios. Three built prototypes are presented in this paper, but none are expected to be a finished consumer product for real-world monitoring settings; rather, these iterations should be considered steps in the overall design process and proof-of-concept prototypes. Moreover, the prototypes were–and will continue to be–meant to determine how useful the Aquapod will be to researchers and to how adapt the design to be a useful tool.

The purpose of this thesis is to provide a detailed description on the design process used to create these prototypes. It is hoped that the chapters herein prove both the functional operation of the prototypes and the completeness of the development process. The main focus is the mechanical and structural design of the robot with possible applications. Though the final prototype is capable of control programing, solving the complicated motion planning algorithms involved in tumbling ambulation is a continuous project and outside the scope of this work.

This thesis begins with the introduction present. Current methods and technology behind environmental monitoring are discussed along with a summary of projects which relate to the numerous research areas this project touches. After sufficient background information has been covered, the idea of the amphibious tumbling robot is developed.

Chapter 3 goes through the history of the robot from its initial conception through many of the design sketches and rejected ideas. Two key designs are presented which move into the prototyping and testing phase. During the design process, many different subsystems were physically tested to ensure the success of the first prototype. The testing methods and models used are presented. At last, the first proof-of-concept, remote-controlled prototype is shown with all the systems and components described in detail.

After the remote controlled prototype was created, a second iteration, done by other graduate students in the lab, included the necessary sensors and equipment for use in the Gulf Coast in order to analyze oil spill damage. Since these contributions were not done by this author, they are quickly summarized to allow for a discussion about the results from the field tests and programming. The final prototype chapter converses the improvements made on the Gulf robot for the final Japan iteration, (see Figure 1.1).

Finally, the future work, both for the project and the current Aquapod, is suggested along with some of the benefits the ruggedized tumbling robot provides to the robotic community and motion planning development. This thesis then concludes with a recap and comparison of all the prototypes developed, how they relate to the overall project, and concludes with some final remarks. All pictures, figures, and tables are the creation of the author unless otherwise noted.

Figure 1.1: Final Aquapod Iteration

# Chapter 2

# Previous Work and Motivation

This chapter presents some of the past and current work in various areas of robotics and environmental monitoring. This section should be viewed as an extensive, not comprehensive, literary review to the best of the author's knowledge. Since this project covers a variety of subjects, research was gathered from many different fields, including, environmental monitoring (both land and water based), general amphibious robots, and tumbling robots. A major topic of this chapter is the Adelopod: the design, motion planning, and the current state at the time of this thesis. Finally, the last section concludes with the need for a solution to bridge the gaps discovered in the applications of these robots.

## 2.1 Environmental Monitoring and Surveillance

This section discusses the current state-of-the-art environmental monitoring through the use of robots; stationary, mobile, or otherwise. Though some of the robots mentioned in Section 2.2 are also used for monitoring, the focus in this is monitoring regardless of the means, while Section 2.2 reverses the focus to the role of the vehicle itself rather than its purpose.

### 2.1.1 Land-based

Monitoring of land environments has been done for a number of years. Most commonly, a network of stationary sensor nodes are stragtigically placed to gather information. These sensors are generally simple (e.g. video cameras, thermometers, etc.), but the amount needed to sample a location and physical installation these nodes can be expensive. Therefore, a number of efforts are underway to develop mobile sensor networks. University of California, Los Angeles (UCLA) has an ongoing project to develop an autonomous-articulated sensor node network for three-dimensional spacial sampling [3, 4]. This project uses mobile sensors attached to aerial cables that can move in a vertical plane with various probes. The moving sensor autonomously follows gradient changes in phenomenon to selectively sample the plane in areas that require more resolution than others, reducing the total number of samples and data collected with the same general information obtained.

Similarly, UCLA has another project to do the same gradient searching on a horizontal plane. The idea is based on bacterium and biased-random walking [5]. By randomly picking a direction to travel and sampling the environment, this algorithm hunts for positive changes in the intensity of the source. The project has proven to work with multiple sources and time-variant intensities better than other methods of source finding.

### 2.1.2 Water-based

Because of the increasing concern for the health of coral reef environments, extensive work has been done in underwater monitoring and the design of Autonomous Underwater Vehicles (AUV). Many projects are very similar, for instance, the AQUA platform (described in detail below) has many of the same features as the strictly underwater project "Starbug." Developed at the CSIRO ICT Centre in Australia, the Starbug is equipped with a stereo camera, GPS, pressure sensors, and more; it was designed to be a low cost, durable, and maneuverable platform [6]. Another is being developed for use in the Baltic Sea for similar autonomous monitoring [7]. A smaller project at Michigan State University uses small plastic fish equipped with a camera and GPS to swim near the surface of the water [8].

## 2.2 Amphibious and Tumbling Robots

This section dives into the world of amphibious and tumbling robots. The bulk of work presented here encompasses current research efforts to design different amphibious robots. These robots fit into two categories: completely submersible or surface swimming. The former implies the robot is capable of swimming underwater, while the latter simply floats on top like a boat; both categories are relevant to the current work. Finally, a few robots which use tumbling as their form of ambulation are discussed. Though the tumbling aspect of the current work is not the main focus, it is important to reveal similar work in progress.

### 2.2.1 Surface Swimming Amphibious Robots

At the NanoRobotics Laboratory at Carnegie Mellon University, research is being done to simulate the water-running abilities of the basilisk lizard [9]. A four-bar mechanism was modeled to simulate the running motion of the lizard. Computer simulations were done to optimize the design for a greater lift to power ratio. Finally, a physical prototype was created to confirm the calculations and computer simulations.

Continuing off the successful Whegs$^{TM}$[10] project, A. Boxerbaum, et al. are working on an amphibious extension. The Whegs$^{TM}$design is inspired by the cockroach and uses a combination wheel-leg to surmount relatively large obstacles and navigate rough terrain. The work proposed in [11] describes the amphibious design and the new features for the platform. A final version would be capable of walking on land or water body floor, and the addition of new propeller-legs would allow the robot to swim, giving it a new dimension of navigational freedom. Complimentary work in [12] discusses the control and autonomous features of the amphibious Whegs$^{TM}$.

Moving away from wheels and legs, the Groundbot is a spherical amphibious robot [13]. Initially developed by the Swedish Defense Research Agency, the 60cm diameter ball can navigate on land and on the surface of water. Two actuators, internal to the robot, displace a central mass causing the ball to roll. The Groundbot is also equipped with two water sealed, forward facing cameras. Though mobile on a variety of terrains with a maximum speed of 3m/s, quick maneuvering and turning still pose an issue with the overall design.

### 2.2.2 Submersible Amphibious Robots

The RHex platform was the first significant innovation relating to legged robots for use in difficult terrain. The project was started in 1999, and lasted five years through the support of DARPA and various other grants including the NSF [14, 15]. The attributing factor to the RHex platform is the hexapod design, which uses compliant legs. These arched legs can be put into different phases to produce a variety of gaits for running, jumping, climbing, or walking. Considerable effort was put into the Rugged-Rhex to achieve a level of water proofing up to 5 meters.

The AQUA robot is an extension of the RHex design, using the same hexapod leg structure. A new set of amphibious legs were designed to assist in traversing sand and surface swimming in shallow water. Another set of fins is used when the AQUA is in open water mode. These allow control over five of the six degrees of freedom inherent in underwater motion. The robot can reach depths of up to 15m [16] and uses vision to semi-autonomously control motion with human input.

The purpose of the AQUA project is to monitor shallow depth aquatic environment such as coral reefs. The advantage of the oscillating fins, as opposed to thrusters in shallow depths, is to avoid the disturbance of sand and debris which could be harmful to the marine life. To conduct environmental monitoring tasks, the AQUA is equipped with a trinocular sensor package that can be used for 3-dimensional reconstruction. [17] mentions the development of a buoyancy control device, but no subsequent publications reference any such control.

Another alternative to traditional mobility methods are serpentine robots. A small number of snake robots exist, a few with swimming capabilities. The AmphiBot series [18] has a total of seven, single-axis segments for a total length of 80cm and a diameter of only 5cm. Its density is lower than water, so the robot surface swims. It also needs to be externally over-pressurized in order to maintain a water-tight seal. The ACM-R5 [19] can operate underwater and uses a 2-DOF bellow to actuate the joints for a more fluid motion. However, motion control and path planning in the underwater environment still remains a complicated issue. At almost 2m in length, the ACM-R5 is still large with complex mechanisms.

### 2.2.3 Tumbling Robots

Continuing with the biologically-inspired robots, the TeTWalker discussed in [20] relates the telescoping mechanism of the robot to various analogs found in nature. The robot is a tetrahedral frame with actuating sides which allow the structure to change its center of mass and flip to move. The purpose of the research is to find a stronger and more efficient telescoping mechanism by looking at various animals that perform similar tasks, such as the tongue of a salamander. Once a suitable mechanism is found for the sides, more complex shapes can be assembled to create larger and more capable structures. These continuously morphing structures have been simulated to almost effortlessly navigate complex terrain [21].

## 2.3 Adelopod

Developed by Brett Hemes at the Center for Distributed Robotics at the University of Minnesota, the Adelopod is a new robotic platform based on tumbling. In the following chapters it will be evident that this thesis is strongly related to the Adelopod. This work should be viewed as an extension or second iteration of the Adelopod platform. And so, it makes sense to spend considerable time with background information and the current state of the Adelopod. The focus of the parallel research is to examine the motion planning algorithm which determines how to get from point A to point B with this non-holonomic system.

### 2.3.1 Motivation for Tumbling Robots

There are a variety of reasons why miniature robots are useful. However, as the size of the robot decreases the utility and mobility of the robot also tends to decrease. Therefore, a high mobility-to-size ratio is often desired among smaller robots. This is harder to overcome with conventional wheeled robots since the largest scalable object is generally directly proportional to the size of the wheel. Therefore, other methods of mobility must be explored.

Many efforts are being put into closely mimicking mechanisms found in nature. The attraction of these designs is the high efficiency of the mechanism, however this is at the cost of complex hardware or intensive low level control. This was partially overcome in the Rhex platform, however, the robot still used a total of six actuators to move. One relatively unexplored method of mobility is tumbling. Using simplified hardware, high mobility can be maintained by using the body of the robot to assist in motion. A definition of tumbling and tumbling robots is taken from [2]:

> **Definition** A tumble is a dynamic state of instability during which the robot pivots about an axis formed by two or more contact points with the ground, accelerating downward with gravity and thus behaving as an inverted pendulum.
>
> **Definition** A tumbling robot is any robot that, by the previous definition, tumbles as its primary means of locomotion with the body playing an active role in achieving such motion.

When these definitions are used in conjuncture, only a few robots qualify as tumbling; the main differentiator being the use of the body for motion. The TeTWalker falls under this category, but the total research being done on tumbling robots is scarce. It is believed that the reason behind the lack of interest is that no natural analog to tumbling can be found in nature. However, tumbling has proven to be very effective in dynamic terrain where a similar sized wheeled robot would fail [22]. Thus, the Adelopod was created to explore the potential of tumbling robots and to determine where, if at all, their place would be in a larger team of robots.

### 2.3.2 Hardware

The Adelopod is a small trapezoidal robot with two rigid arms as seen in Figure 2.1. The dimensions with the external casing are approximately 16cm long, 7cm wide, and 2cm deep. The overall mass comes to 400g and has a battery life of 25-30 minutes. The Adelopod uses standoffs and two PCB boards as its main structural components. The casing adds a lightweight protection for the on-board electronics. The arms are 12cm long, made from steel, and connect to servos inside the body.

Because tumbling is a relatively unexplored method of locomotion, the optimal angle of the arms was unknown. Therefore, additional shoulder actuators allow the arms to change their angle which provide a total of four degrees of freedom for the robot. After some initial experimentation, slight variations in the arm angles where found to have no significant influence on tumbling, so to simplify motion planning the shoulders are generally disabled on the Adelopod. Therefore, future iterations do not include this feature to cut back on weight and complexity.

### 2.3.3 Motion Planning

Though the development of the Adelopod hardware was an important step in the process, the main research was focused on finding a model for motion. On the ground, any object will have three degrees of freedom. However, there are only two actuators on the Adelopod which

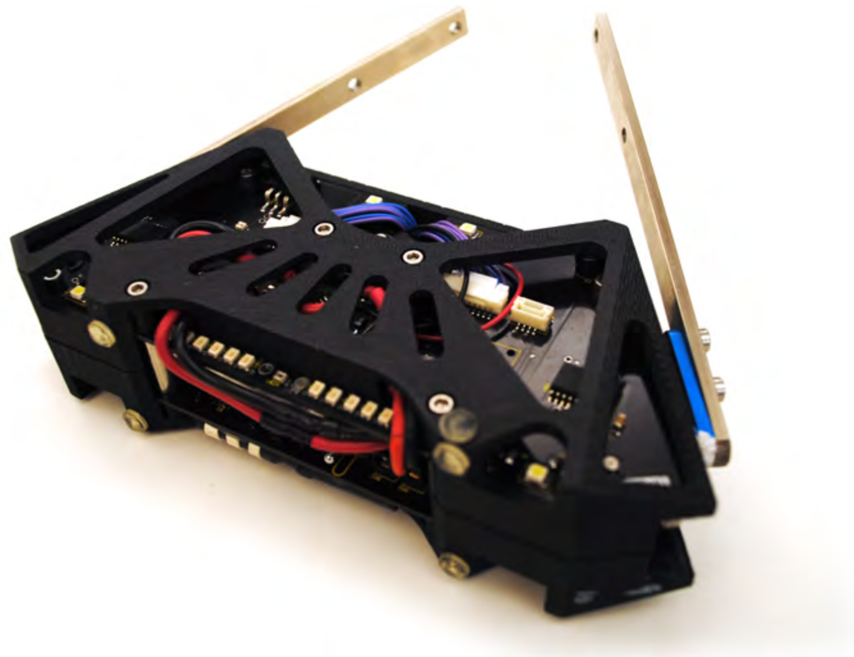


Figure 2.1: Picture of the Adelopod

creates a non-holonomic system. In a fully holonomic system, a robot would be able to move in any direction with any orientation from any location. However, when the system has less controllable degrees of freedom than the environment provides, the system is non-holonomic, and is limited to a finite number of paths.

To complicate matters further, the under actuation for the Adelopod is the not a typical system found in a car or wheeled robot (i.e. front/back and rotation). The movement of one arm does not correspond to a one-to-one motion of the robot and is dependent on the configuration of body and other arm. Therefore, a more complicated motion planning algorithm must be developed to get the robot from point A to point B.

The current system in place separates a finite number of configurations for the arms and body into different states. Each state is connected to another state by the movement of one of the arms (e.g., rotating an arm by 90°). Through experimental results, the transition of these states is connected to a net displacement of the robot.

When a desired end point is given to the Adelopod, a searching algorithm links together a number of these state transitions, adds the net displacements, and attempts to find an optimal route to the goal. Currently, the computational complexity of the search, the compounding errors of the net displacements due to friction variances, and other real-world factors limit the speed and accuracy of the robot. More advanced techniques are being developed, and will hopefully yield a dynamic and accurate model for practical use.

### 2.3.4 Treaded Adelopod

In an effort to simplify the path planning, a major modification was added to the Adelopod. Tank treads were added over the outside shell to allow the robot to travel more like a car (see Figure 2.2). The addition of the treads was two fold: first, as mentioned, was to allow the robot to move smoothly over flat surfaces. The second was to aid the robot's effort while traversing slightly taller obstacles. The treads can grab onto the edge and assist in pulling the robot on the ledge. This also prevents the robot from getting stuck on a tall, thin obstacle if the arms can no longer push against ground. The treads were not meant to replace tumbling as the main form of ambulation, but rather to provide minor adjustments when tumbling over an object.

## 2.4 Filling the Gaps

This chapter presented a number of different robots being developed, some with uses in environmental monitoring, while others just being novel forms of locomotion. For the purposes of specific applications, there appears to be a gap in environmental robots for shallow, fresh water monitoring. Many aquatic robots are designed for use in coral or deep sea environments

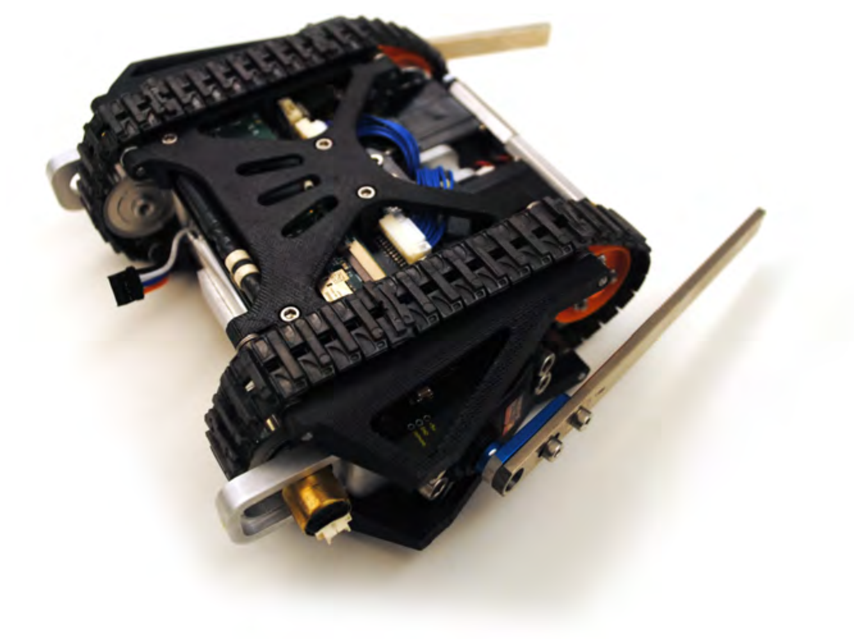


Figure 2.2: Treaded Adelopod

and have difficulties in shallow water. Most of these robots are also very large and difficult to transport, as well as expensive. This thesis aims to fill the gap by providing a highly mobile, inexpensive, and easy-to-transport robot, which is still capable of performing monitoring tasks and data collection. Making this robot amphibious will also allow it to be used in both shallow water and on land. The next chapter will start the development of such a robot.

# Chapter 3

# Concept Generation

This chapter starts with an introduction to the Center for Distributed Robotics and discusses some of the goals for hybrid robots. The origin of the project is outlined from the conception of the idea to the first computer model. Initial design considerations and requirements are presented with a detailed description of the requirements, both for short term and long term success. The first fully-developed design is shown in detail, as well a quick description of its limitations and the desire to move to a new design.

## 3.1 New Robotic Teams

The main focus at the Center for Distributed Robotics (CDR) is to create teams of robots which work together to achieve a common goal. Generally, this is done by bringing a number of small-to-mid size robots into an unknown environment. Smaller robots reduce costs, energy consumption, and transportation resources, while increasing the total area searched in a shorter period of time. Many efforts at the CDR, among other places, are focused on the control and coordination of these teams. However, this research generally uses simulations with anonymous or simplistic models for the physical robots.

To deal with this anonymity, a second branch of the CDR is dedicated to the design of robots for use in distributed research. Dealing with unknown environments, a truly versatile platform would be completely configurable from a set of tiny actuated modules much like LEGOs. This approach is currently under development at a variety of places [23, 24], however, the technology and research has yet to produce a useful platform. The smallest size of these modules is currently a 4.6cm cube [25], and only a few dozen have been made. Until these cubes are on the order of sub-centimeters and produced in the hundreds of thousands, an environment interacting robot will be difficult to create. This technology will be achieved eventually, however, in the meantime, other platforms must be explored.

Since the unknown environment is still an important factor, another method is to combine many single use platforms into a multi-functional hybrid robot. The general idea is to create a robot that is capable of going anywhere by a variety of means. In the end, this 'super hybrid' would be able to drive, walk, climb, swim, fly, or use other means of transportation in any given situation, depending on the current need. But, like the modular robots, this technology is far from complete. Which is why taking the smaller steps of combining a few of the features at a time must be made.

In efforts to expand the research at the CDR, brainstorming sessions were held to generate new ideas for platforms. The meetings would start by listing any and all forms of locomotion, and then randomly combining two or more to create a new hybrid. Every combination was examined and quick sketches were done to explore how the hybrid could work. Every idea was considered, no matter how trivial, impracticable, or undesired. Through the meetings, one idea continued to resurface as a viable option to pursue with a variety of applications. This idea was simply a new amphibious robot.

## 3.2 Why?

Before pursuing the idea of an amphibious robot, the first question to answer was 'why?' Why would someone want this robot? What makes it useful or better than what is currently available? When would this robot be used, and who would use it? There are many uses for mobile robots and teams of distributed robots. Search and rescue operations, surveillance, and reconnaissance all play vital roles for keeping people safe. Mobile robots also provide alternatives by taking humans out of dangerous or hostile situations.

One rising global concern is the health and status of the natural environment. Chapter 2 listed the current state of environmental monitoring, which is the focus of this robot. The remainder of the report will discuss the 'how' and technical details behind how this robot solves the current issues. In regards to this concept, a final brainstorming meeting was held to generate a list of uses for this robot. A variety of situations for motivation were generated including the list below. Since the meeting, many of these ideas were explored and researched, while adding others to the list.

- Sensor Deployment in Rivers/Streams
- Coral Monitoring
- Fish Counting
- Temporary Invasive Species
- Automated Retrieval

- Environmental Recharging
- Measuring Water Quality

## 3.3 From Wheels to Arms - Initial Design Considerations

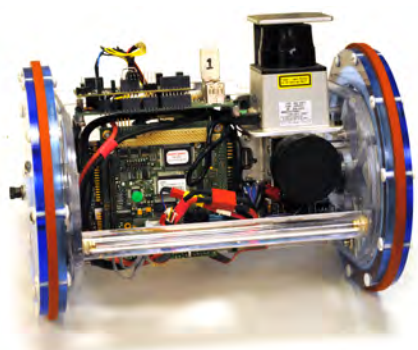
The CDR garnered it reputation from the Scout robot, a two-wheeled plus tail design that was extremely robust. Capable of being thrown through a window or shot from a grenade launcher, the Scout can handle just about any abuse. Though the two wheel design was not innovative, it cut down on hardware complexity which made the design easier to ruggedize. A large number of publications were made about the Scout, and so the CDR continued this trend of design for many years. The eROSI, Explorer, and MicroVision (see Figure 3.1) all share similar design features. Therefore, the earliest designs for the amphibious robot tried to conform to this structure (see Figure 3.2 and 3.3).

However, after some consideration, the two-wheeled design was abandoned for a completely different approach. This was to avoid the complications involved with under-actuated six-degree of freedom motion inherent for swimming. The new idea was to use the floor of the water body

(a) Scout

(b) eROSI

(c) Explorer



(d) MicroVision

Figure 3.1: Different Robots from the CDR

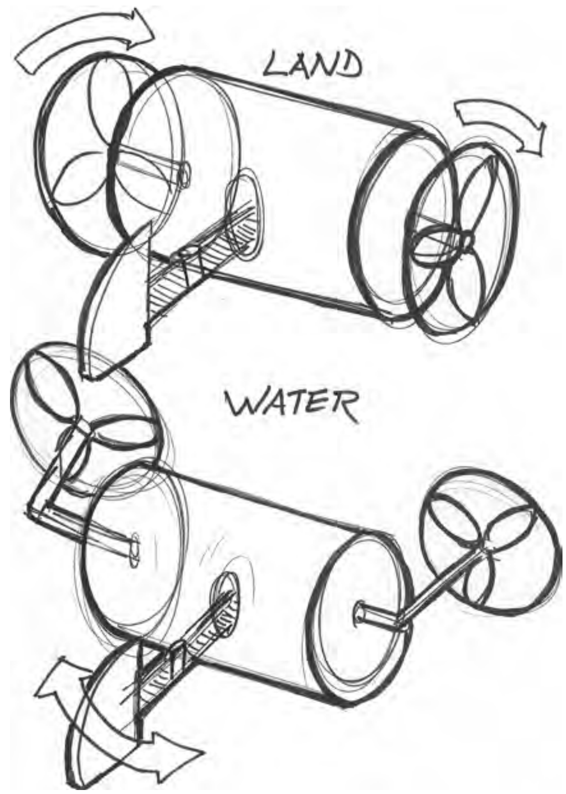



Figure 3.2: First Concept, Based on the Scout

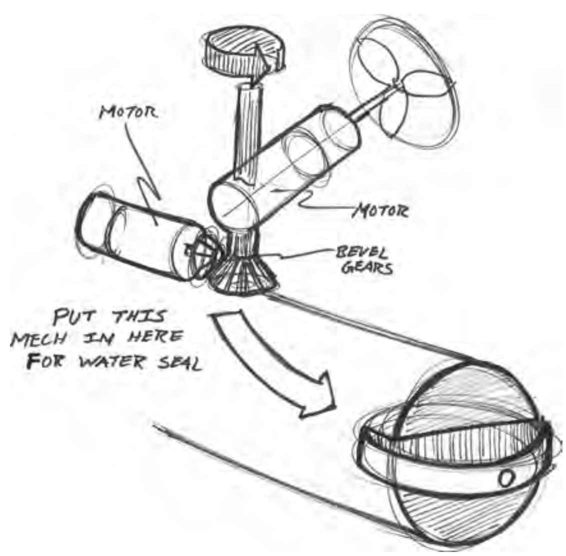



Figure 3.3: Internal Component Concepts

for ambulation. The CDR was developing a completely new mode of locomotion, tumbling. Since wheels would most likely prove incapable of working on muddy and vegetation carpeted floor, tumbling offered a new solution. After considering the advantages of tumbling explained in Section 2.3, an amphibious design was a natural extension of the Adelopod. Finally, with a basic structure of the design, a list of design requirements was created to measure the success of the robot.

### 3.3.1 Proof-of-Concept Design

These requirements were based on the performance of the original Adelopod and the additional need of being waterproof. The requirements are placed into two lists, the first describes a successful proof-of-concept design which is the focus of this work.

- **Small size (under a 50% increase from the original Adelopod):** In order to maintain the title of a small robot, the overall size must be kept to a minimum. Since additional features have been added in comparison to the original Adelopod design, some volume increase is expected, though too much is undesired. This also brings up an important aspect of tumbling, scaling. Wheeled robots can get progressively bigger as long as the actuators are powerful enough to move the wheels. Since tumbling is essentially an inverted pendulum, larger structures will increase the falling distance and impact force, causing larger shocks to the internal components. The actual scaling aspect of tumbling robots has not been explored on a research level, so the smallest size possible is desired.

- **Seamless land to water transitions:** This is the key feature of the amphibious design. Ideally there should be no human interaction with the robot to assist the transition from land to water. The robot must be able to tumble into a stream or lake from the shore, or be dropped into water and continue without adjustments. This is in response to the AQUA design which required that new legs were physically changed between the water and land modes. For use in environmental monitoring, it should not be expected that human interaction is available. As an additional requirement the Aquapod must be able to tumble in and out of streams or lakes. It is unexpected that the robot will be able to climb an undercut or vertical stream bank, but a steeply sloping bank (up to 30 degree slope) should be traversable.

- **Water Surface Movement:** If deployed in the middle of a lake or other body of water, the robot should be able to move into position before diving. This will greatly aid in locating the robot via GPS, since the signal will be lost once underwater. It will also allow the robot to travel to a stream shore when it surfaces and use the GPS to aid in direction.

- **Internal buoyancy control:** Tumbling on the floor of a lake or stream might not always be the most efficient means of traveling, and could at points be impossible due to a large obstacle. Therefore the ability to float to the surface and flow with the current is important. This feature can also be used as a means of retrieval or long-range transportation.

- **Wireless communication and on-board processing:** In order to organize a mission or relay information to other robots it must have some form of wireless communication. This becomes more of an issue when working underwater since many forms of wireless are attenuated. Also to do motion planning or vision detection there must be an on-board processor to handle the information and compute motion planning algorithms.

- **Water proof:** This design requirement is self-explanatory. However, because of the structure of the buoyancy control, the robot must also be air-tight and capable of handling increases in internal pressures. The degree to which the robot can handle is dependent on how far underwater the robot will go. For the purposes of the proof-of-concept, a depth of around 5 meters is acceptable.

### 3.3.2 Real-World Design

This second list expands on Section 3.3.1 to include features which will showcase a successful amphibious robot for use in environmental data collection.

- **Water proof up to 25m:** To operate at the deepest point in most small- to mid-sized lakes (as observed from a wide variety of topographical lake maps), the final design must be capable of handling depths of around 25m. Though it is unanticipated that the robot will be able to preform any useful tasks at this depth, it is important that the components are capable of handling the worst and still be retrievable.

- **On-board sensors:** In order to take readings or collect data in an environment, the robot will need some sort of sensor suite. This could include anything from pressure to conductivity, laser range-finders, cameras, or IR. This category is beyond any sensors the robot will most likely need for navigation and path planning such as accelerometers. The sensors must be small enough to fit inside the the robot and have minimal power drain.

- **GPS:** Including a global positioning system device on the robot will be essential for localizing and recording data. However, GPS is a high frequency transmission and therefore only works a few inches under the surface of water. In order to receive data, the robot must either be on land, surface from the water, or relay the information to other robots in the form of lower frequencies such as sonar.

- **Power Sourcing:** One feature which could prove to be extremely useful is the ability to recharge a battery, or other consumable resource, from the environment. By including a solar panel, or other renewable energy generator, on the robot it could 'bask' for energy from the environment then return to monitoring once charged.

- **Manipulators:** The usefulness of the robot could be greatly expanded by including some form of manipulator. This would most likely be part of the arms and simply consist of a clamp or grasping mechanism. However, other manipulators could include dirt/water sampling devices.

## 3.4 Buoyancy Control

The innovative feature of this robot is the idea of buoyancy control. Some of the unique uses of the buoyancy control are described in detail in Section 8.2. The topic of this section is to describe the physical process of buoyancy control, the calculations involved, and the variety of solutions explored to achieve a change in density.

### 3.4.1 Calculations

An over-simplified, but still accurate, model of buoyancy indicates that an object submersed in a fluid will have a relative weight proportional to the volume of fluid that has been displaced. This can be written in terms of Equation (3.1):

$$m_{relative} = m_{abs} * \left(1 - \frac{\rho_{fluid}}{\rho_{obj}}\right), \tag{3.1}$$

where $m_{relative}$ is the effective mass of the object that forces (e.g. gravity) act upon, $m_{abs}$ is the normal mass of the object, $\rho_{fluid}$ is the density of the surrounding fluid, and $\rho_{obj}$ is the average density of the submerged object. This equation shows that if the average density of the object and the fluid are the same, there will be no net force on the object due to gravity and it will be neutrally buoyant. Furthermore, this proves the trivial conclusion that a less dense object floats since it will have a negative relative mass.

According to the theory, any slight variation from neutral buoyancy will allow passive forces to make the robot sink or float. Thus, any concept should be developed with the idea of being as close to neutrally buoyant as possible. This requires the least amount of density change which takes energy to accomplish. For neutral buoyancy ($m_{relative} = 0$), Equation (3.1) reduces to

$$m_{abs} = \rho_{fluid} * V_{obj}, \tag{3.2}$$

where $V_{obj}$ is the volume of the object. Working in the metric system, fresh water has a density of $1.00 g/cm^3$. The mass of the robot in grams must be equal to its volume in cubic centimeters. From this point, by changing the density, two distinct modes can be achieved, floating or sunk.

There are two ways to think of the density change. Either, keeping a constant volume and changing the mass (i.e. the added weight becomes part of the robot) or keeping a constant mass and changing the volume (i.e., the added weight is not part of the robot; rather the available air-space inside the robot has decreased). Both ways produce the same system, and change the density by the same amount; it is only a matter of where the system boundary is drawn and what is convenient to keep track of.

For example, consider a $2.5kg$ rubber ball, inflated with $0.003m^3$ of air at standard pressure and temperature. The density of the ball is $0.83g/cm^3$, and will have a relative mass in water of $-0.5kg$, indicating that the ball will float. Now, if the rubber were able to contract and decrease its volume to $0.002m^3$, then the new density would be $1.25g/cm^3$, and will have a relative mass in water of $0.5kg$, indicating that the ball will sink. However, because of the decrease in volume, the pressure of the air has also increased by $50kPa$.

### 3.4.2 Concepts

The following pages are a handful of the concepts that were generated to accomplish the change in density. The change in density can be done in a variety of ways, each falling in one of the three major categories: electrical, mechanical, or chemical. As a starting point, it was desired that the robot was capable of undergoing this transition many times, and should therefore use as little consumable resources as possible. This excluded many, but not all, chemical methods; and so the focus was on mechanical and electrical means.

#### Previous Work on Buoyancy Control Mechanisms

The idea of buoyancy control came from the Argo company and their continuing research of ocean currents using neutrally buoyant floats [26]. The original float created by John Swallow in 1954, was designed to stabilize at a depth of 1000m with no means of returning to the surface. Recently, to cope with transmitting data underwater, new designs have been equipped with

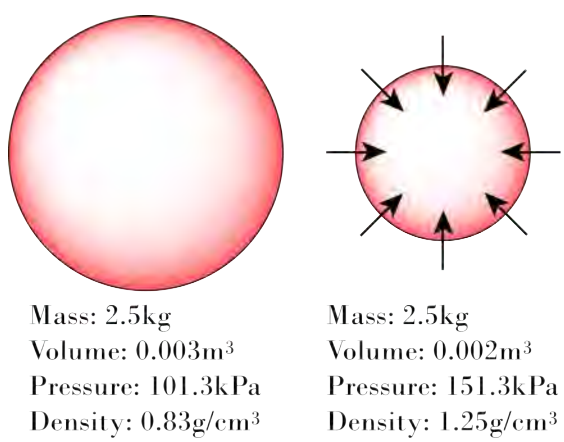



Figure 3.4: Density Example

buoyancy control features. These floats, illustrated in Figure 3.5, work by pushing a pneumatic cylinder full of oil into an external bladder increasing the overall volume of the buoy just enough to create a negative force and float to the surface.

Another concept, based on the head of sperm whales, uses a container of oil which expands when heated and pushes a small volume of water out of a chamber to decrease density [27]. The purpose of the this robot, called DIVEBOT, is similar to the Argo floats for measuring deep ocean currents. By using oil as the variable medium, the incompressible nature of liquids allows for higher pressures to obtain deeper depths.

The most closely related robot using buoyancy control is developed by the University of Southern California [28]. This cylindrical robot uses a simple piston to vary the available water space inside the robot. Designed for use in underwater sensor networks, the idea for the buoyancy control is to maintain a particular depth which is measured by the external pressure of the water. A physical prototype and testing has been done to confirm the design.

Building on this concept for buoyancy control, the AMOUR V robot [29] uses a similar piston for volume alteration, but also moves the battery pack to change the center of mass. The purpose of this robot was to test varying payloads during underwater missions while trying to maintain a stable orientation.

Using these examples as a starting point, many different designs were considered to achieve the same behavior. Though the mechanisms in the above examples have been tested and proven to work, their size and power requirements are larger than what was capable of fitting in the

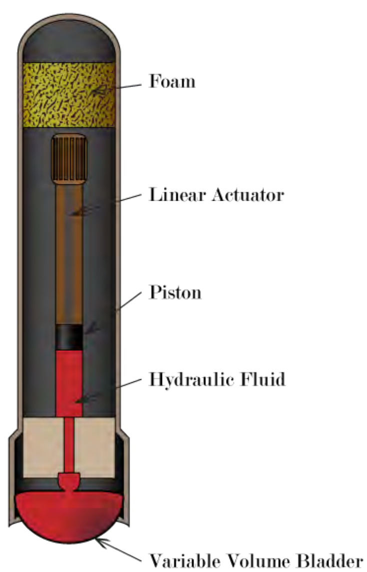



Figure 3.5: Argo Buoy

size constraints of this robot. In addition, the above buoyancy controls were used to maintain a particular depth and required precise variations in density. The robot in this thesis had only two states that required one large difference in density, rather than fine resolution over a small range.

### Variable External Volume

The following concepts took the idea of having an external variable volume similar to the Argo floats. As mentioned in Section 3.4.1, there are two ways to track the system for a change in density, for these concepts it is convenient to think about a fixed mass with a changing volume. The first concept in Figure 3.6 is closely related to the floats, but uses an air pump instead of a linear actuator to expand the bladder.

The next concepts in Figure 3.7 use an accordion mechanism to change the entire size of the external shell. These concepts would require a flexible membrane material for a portion of the shell that could also withstand pressure changes and sharp objects. For this reason, these ideas were rejected. In all of the concepts, it was desired to keep the required actuators to one; this reduces the weight and complexity of the robot overall.

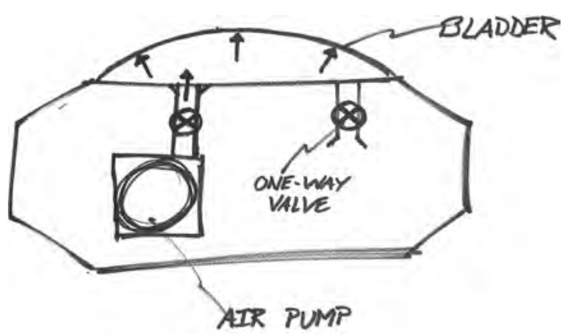
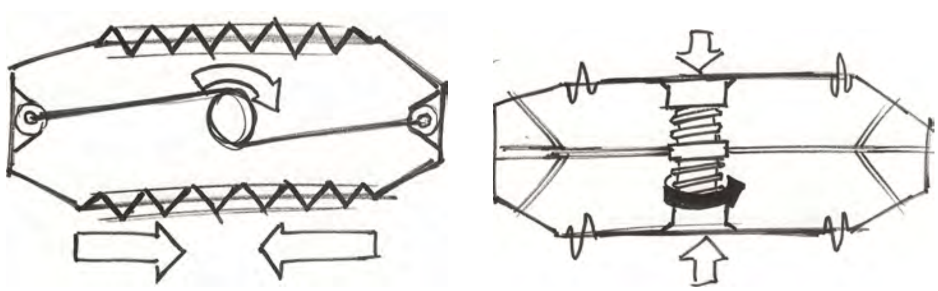



Figure 3.6: Air Bladder



(a) Horizontal Displacement (b) Vertical Displacement

Figure 3.7: Accordion Style

**Variable Internal Mass**

The next concepts have a fixed external shell volume, but take in water from the environment to change the overall mass of the robot. This effectively increases the overall mass of the robot by the amount of water taken in. The first concept in Figure 3.8 uses an accordion tube to pull water into the shell with the aid of pulleys and wire. The natural pressure of the water would be required to fill the bag, keeping a state of equilibrium pressure between the inside and outside. However, this system did not work as anticipated and was rejected.

The next concepts (see Figure 3.9) used a couple different methods for actuating a piston/cylinder combination to pull water directly into the cylinder as opposed to using the bladder medium. Though the bladder may help to keep the robot separate from the external environment and aid in the precision of the density, the extra medium took up too much volume and mass; it was eliminated to conserve space and reduce complexity.

After considerable thought, and some circular idea generation, a good solution was found to be a piston/cylinder combination. However, this was combined with the pulley mechanism and an additional spring to have both in and out control of the cylinder. This concept is talked about in detail in the following section.

## 3.5 First Model

Through a variety of iterations, the pneumatic cylinder of the Argo floats turned into a plunger and cylinder exposed to the outside environment. This concept was given a substantial amount of attention as it appeared to be the most viable option at the time. Figure 3.10(a) shows the outside shell of the robot with the input port for the cylinder, while Figure 3.10(b) shows the internal components of the robot. All of the components in the model were carefully chosen and theoretical calculations were done to ensure the design was valid.

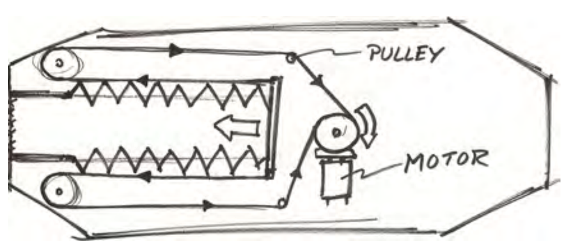
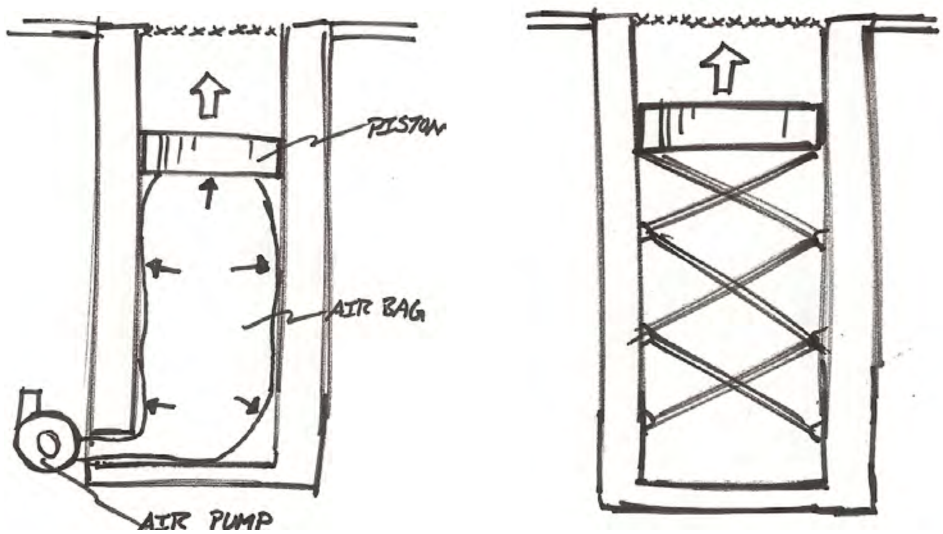


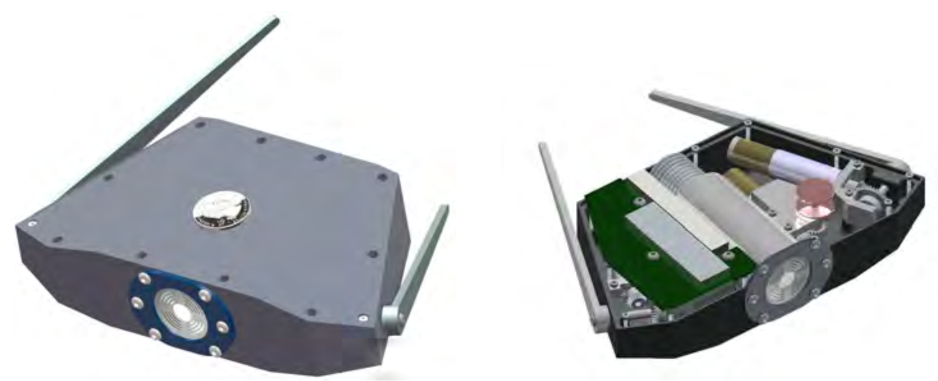

Figure 3.8: Accordion Tube




(a) Air Actuation

(b) Scissor Lift

Figure 3.9: Piston/Cylinder Concepts



(a) Outside View: US quarter on the top shell for reference of size.

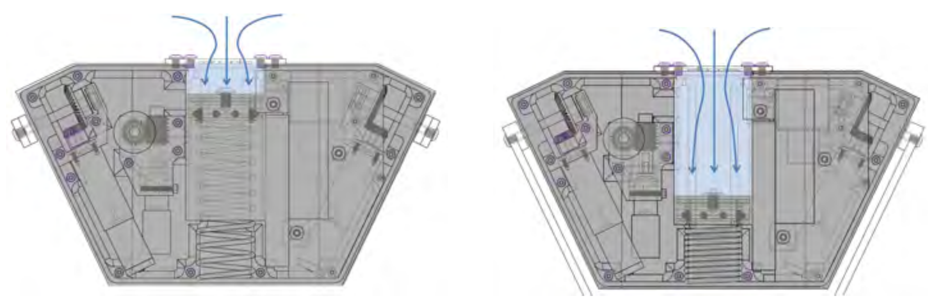
(b) Inside View: Top cover removed to expose internal components.

Figure 3.10: Plunger Design Renderings

### 3.5.1 Plunger/Winch Mechanism

A spring and winch combination was used to push and pull the plunger in the cylinder. The spring was placed inside the shell behind the plunger to maximize the stroke length. A cable and pulley system weaved down the cylinder to provide a greater mechanical advantage reducing the size and weight of the winch motor.

A worm gear was used for added gear reduction and as a locking mechanism. A spool connected to the worm gear winds up a steel wire anchored to the plunger. Since speed was not important for the design, an additional mechanical advantage was added in the form of a block and tackle pulley system. To keep the thickness of the plunger to a minimum, the pulleys were embedded in the plunger (see Figure 3.12). The pulley arrangement, minus the effects of friction, gave the system an additional 4:1 mechanical advantage. This system also reduced the tension in the wire and stress in the pulley components.



(a) Plunger at 25%　　(b) Plunger at 100%

Figure 3.11: Plunger Positions

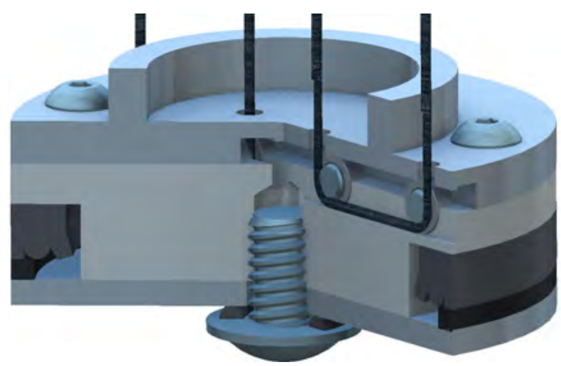


Figure 3.12: Cut-away View of Plunger Assembly

### 3.5.2 Issues

As mentioned in the introduction to this section, the main buoyancy concept used for this design was found to have a variety of issues. The most important was the depth limitations. Since the maximum working depth of the robot relied on the spring, the robot could only reach a theoretical depth of five meters. To go lower, a larger spring was needed, and therefore a larger winch motor, which would increase the mass. And so, though a considerable amount of effort went into this design, it was abandoned for a less expensive and more expandable design.

## 3.6 Moving to Pumps

A new solution was required for the buoyancy control in order to be more robust, reliable, and have a deeper working depth. The idea of using a pump was first discarded when a suitable pump was not found due to size, weight, or maximum pressure inadequacies. However, after extensive searching, a peristaltic pump from a reputable dealer was found. By replacing the tiny custom components and moving parts with a commercially produced product, the overall robot was more robust and less expensive.

### 3.6.1 The Pump

A peristaltic pump works by using rollers to squeeze liquid through a tube without ever contacting the fluid. Though not important for the current project, these pumps are often found in medical devices because of the non-contact feature and precise measuring abilities. The chosen pump was a Series 100 Micro made by Williamson Manufacturing Ltd., see Figure 3.13. Small and lightweight, the pump proved to be a significantly less expensive and more reliable method for buoyancy control. Initial estimates calculated the robot would be able to push water out of the shell from a depth of 15m, based on the maximum pressure rating of the data sheet. The only issue to be addressed was finding a suitable motor capable of turning the internal rollers.

Figure 3.13: Series 100 Micro

### 3.6.2 New Design

Removing the central cylinder from the initial design allowed the new design to reduce the shell thickness by 12%. The width and length of the robot was also reduced due to the freed space inside the robot. However, since the overall volume of the robot was reduced, that also reduced the weight limit. The majority of the remaining components stayed the same for the new design, and were rearranged to fit the pump and motor, this can be seen in Figure 3.14. Removing the cyclinder also eliminated a major sealing issue between the shell halves.

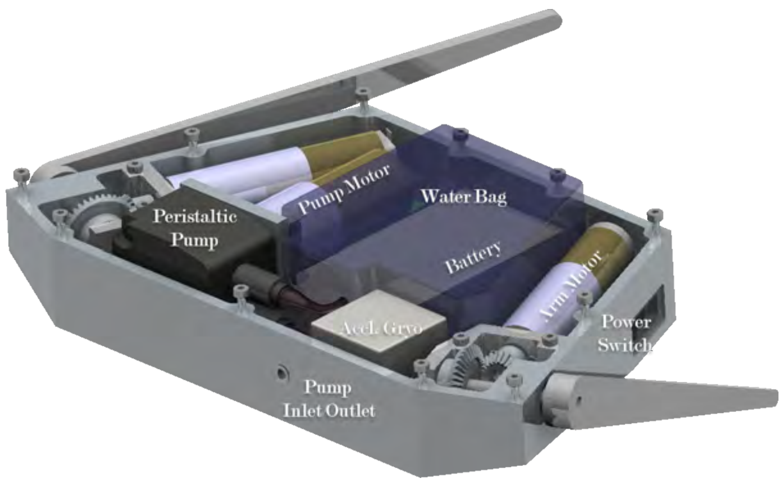



Figure 3.14: New Design with Labeled Components

To accomplish a change in density, the pump took water from the environment through an inlet on the surface of the shell. The tubing went through the pump and ended at a bladder. When the bag expanded due to the increase of water, the available air-space inside the robot decreased, while the overall weight increased. To decrease the density, the pump motor was run in reverse and water was pushed back out into the environment. The advantage to the peristaltic pump was once power had stopped being supplied to the motor, the rollers were held in place by friction. This prevented water from entering or leaving the robot when undesired.

Up to this point, all of the design was theoretical, using PRO/Engineer to model the components and hand calculations to confirm density and power requirements. Very few physical tests had been done to ensure the design would work. The next chapter takes the new design mentioned in this section and starts physical testing and prototyping.

# Chapter 4

# Prototypes and Testing

Throughout the design process, prototypes of various mediums were created to ensure the success of a physical model. The most prolific prototype being the Pro/Engineer 3D model and hand sketches. However, once the design matured to a particular point, paper and computer models were no longer capable of providing valuable feedback. This chapter discusses the various physical prototypes created during the project, their purpose, and what was learned through their use. There was a number of sub-systems in the design that were tested individually to reduce the time spent fixing issues on a full prototype.

## 4.1 Rapid Prototype

The first physical model of the design was made using a Dimension 1200es Series 3D Printer. The purpose of building a rapid prototype (RP) model was to ensure the screw seals, gasket, and lip seals were capable of water-proofing the internal components. Since the scale of the robot fit within the size limitations of the printer, a full size model was made (see Figure 4.1). After some modifications and assembly, the RP model was submerged in water; a successful test was initially designated by no air-bubbles leaving the model. However, this was quickly dismissed as an inaccurate means when residual air pockets in the ABS started to seep through the shell. For a better test of the water sealing, the RP model was left underwater over a period of two days with litmus paper lining the inside shell. After this period of time, the model was disassembled and the paper was inspected for traces of water.

The next step was to test the dynamic quality of the lip seals. For this, two modified servo motors were installed in the shell and connected to mock axles (see Figure 4.2). A radio receiver and battery were also placed in the shell. The model was submerged in water while a radio controller ran the motors. Again litmus paper was used to determine if water was entering the shell. However, though water did not enter through the lip seals, other portions of the shell were

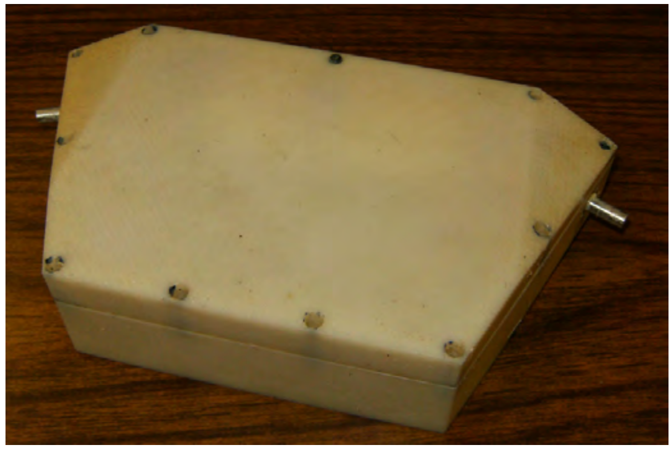


Figure 4.1: Rapid Prototype Model

seeping water.

The RP was a good representation of the design in terms of size, but the printing method and plastic material did not serve as an appropriate tool for seal testing. The ABS did not thread well in terms of screw tapping, and the threads broke down after a few assemblies; this prevented the top and bottom shells from pressing together with enough force to use the gasket material properly. In addition, the layering method used for printing was permeable to water and evident in thin sections of the shell. Though liberal use of lithium grease prevented most of the water absorption, it made it difficult to determine if the air was coming from a seal or through the shell itself.

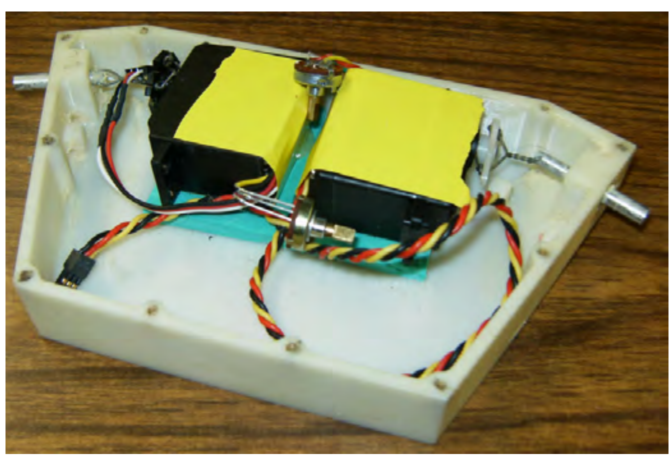


Figure 4.2: Servos Inside the RP Model

## 4.2 Benchtop Buoyancy Testing

The greatest uncertainty in the design was the development of the buoyancy control. With the exception of the methods mentioned in Section 3.4.2, the buoyancy control in this robot was relatively unproven in this type of application. According to the calculations from Section 3.4.1, theoretically, the design would work. Table 4.1 shows the calculations done on the rendered model presented at the end of the previous chapter. However, to avoid any unforeseen issues, the buoyancy needed to be tested in variety of ways. The first was to simply confirm the calculations and the theory of buoyancy itself.

Table 4.1: Relative Weight and Density for Bladder Fill Percent

| **Bladder (%)** | **Vol. Disp. (ml)** | $\rho\left(\frac{g}{cm^3}\right)$ | **Rel. Mass (g)** | **Int. Press. (gpsi)** |
|---|---|---|---|---|
| 0% | 0.0 | 0.97 | -16.1 | 0.00 |
| 26% | 16.1 | 1.00 | 0.0 | 1.05 |
| 50% | 30.0 | 1.03 | 13.9 | 2.09 |
| 100% | 60.0 | 1.10 | 43.9 | 4.88 |

To accomplish this test on a conceptual level, a vessel made from a plastic container was used. The volume of the vessel was found and material was added to mimic the scaled weight and available air volume of the expected model. This was done to create the same density and pressure change that would be experienced in the physical model (see Table 4.2). Though the test vessel did not include a water bag, the amount of water allowed to enter the vessel was controlled in order to achieve the same final density. A picture of the test vessel is shown in Figure 4.3.

Figure 4.3: Test Vessel

The test vessel was placed in a tall tube of water and allowed to float which confirmed the initial density of the vessel was less than the water. The vessel was also connected to a tube

| | Rendered Model | Test Vessel |
|---|---|---|
| External Volume ($cm^3$) | 464 | 156 |
| Internal Airspace ($cm^3$) | 240.7 | 84 |
| Weight ($g$) | 450 | 150.6 |
| Bag Volume ($cm^3$) | 60 | 21 |
| $\rho$ at 0%($g/cm^3$) | 0.97 | 0.96 |
| $\rho$ at 100%($g/cm^3$) | 1.10 | 1.10 |
| Gage Pressure at 100% (psi) | 4.88 | 4.90 |

Table 4.2: Model to Test Comparison

which ran to the pump outside of the water container. To keep the setup neat, a separate supply source of water was used to pump into the vessel. The setup for this experiment can be see in Figure 4.4.

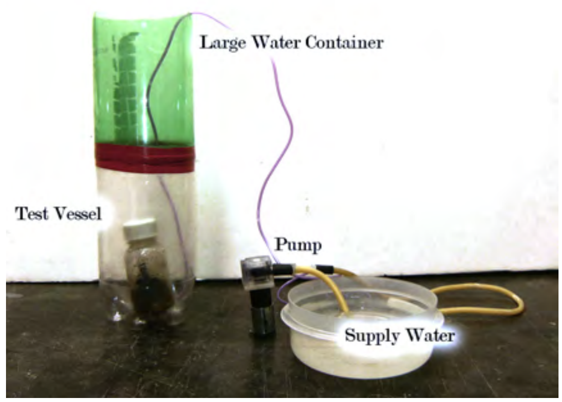



Figure 4.4: Buoyancy Control Concept Test Setup

During testing, the pump was turned on for a specific number of rotations to allow the correct volume of water to enter the vessel. It was observed that once the the vessel went slightly beyond the density of the water, it would rapidly sink to the bottom of the container. The pump was able to perform this transformation in under a minute with the same power input expected in the design. The vessel had no difficulty handling the increased pressure, and additional testing pushed the final pressure up to 11.5psi before the pump stopped increasing the water level.

This test confirmed the buoyancy calculations and the general idea of adding water to a fixed volume to change its density. The test also showed that the pump was capable of handling the required pressure increase inside of the shell.

## 4.3 Peristaltic Pump

Hand-in-hand with the uncertainty of the buoyancy control was the functionality of the pump itself. The manufacturer's data sheet was sparse on detailed information and came in multiple configurations. Testing was done to determine the best combination of settings. The design variables for the testing included: the internal diameter of the tubing, the material of the tubing, and the number of rollers installed. The required results included: the input power to turn the motor, the maximum pressure attained, and the flow rate. Obviously, a low power input, high pressure, and large flow rate were desired, and were of course, inversely proportional to each other. The power input of the motor was also used to determine the minimum required torque; this was helpful while choosing the smallest capable motor for the pump.

## 4.4 RC Prototype

Once all the subsystems were tested, the next step was to build a working prototype of the design. The development of a custom printed circuit board (PCB) with a microprocessor and code would have taken too long for testing the hardware. Therefore, to simplify the electronics, a radio controlled (RC) receiver and electronic speed controllers (ESC) were used to actuate the robot. This section describes some of the modifications made to the initial design along with how the prototype was controlled and assembled. Once the prototypes were built, buoyancy tests, terrain ambulation, and design improvements were performed. See Figure 4.5 for a rendering of the RC prototype.

### 4.4.1 Shell Material Selection

When choosing material for any of the external components, interaction with the environment needed to be taken into consideration. To ensure that the entire robot was designed to the correct density, a plastic was needed for the shell; metal would be too heavy and costly, while other materials would not guarantee a watertight seal, or would be too weak to handle the pressure.

Polymer plastics fit within an acceptable strength/density ratio range to be considered for the main shell. However, polymers have a few negative qualities, mainly cost and water absorption. A table of common plastics along with their qualities and cost was generated to select the best material. This can be seen in Appendix B.1. In the end, polycarbonate was an excellent compromise between density, water absorption, strength, and cost. This material was also widely available and easier to machine.

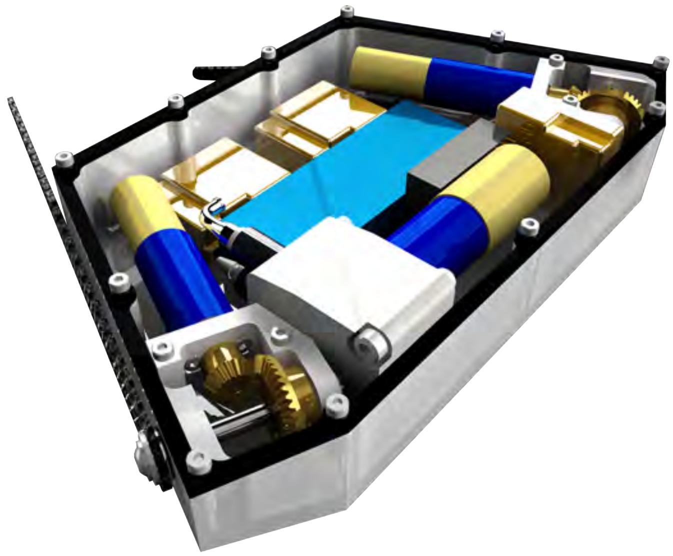


Figure 4.5: RC Prototype Rendering

### 4.4.2 Arms

The RC prototype was designed as a basic proof-of-concept robot, and so it was important that it was able to move and actuate in a similar manner to the original Adelopod. The design of the arms was carefully done to ensure the robot was able to support its own weight while tumbling, and most importantly, keep a watertight seal. This section shows the calculations done to select motors and the sealing methods used on the mechanical seal.

#### Motor Mounts

To maximize the internal space, the arm motors were placed along the angled walls. A bevel gear was required to turn the output shaft along the sides of the robot. Using this gear allowed an additional reduction stage beyond any chosen gearbox. However, these external gears needed to be held in place along with the motor itself. A mount was designed to house the bevel gears with screw holes to attach the motor, see Figure 4.6. The material was chosen as nylon 6/6 to reduce weight and the shape was made symmetrical to be used for both the right and left sides. The mounting bracket also housed one of the support bearings and provided a restraining surface to the E-clip to prevent axial movement of the arm shaft.

#### Sealing

The motor electronics needed be kept away from water in order to work properly and prevent short circuits. This meant that somewhere between the motor and the environment a dynamic

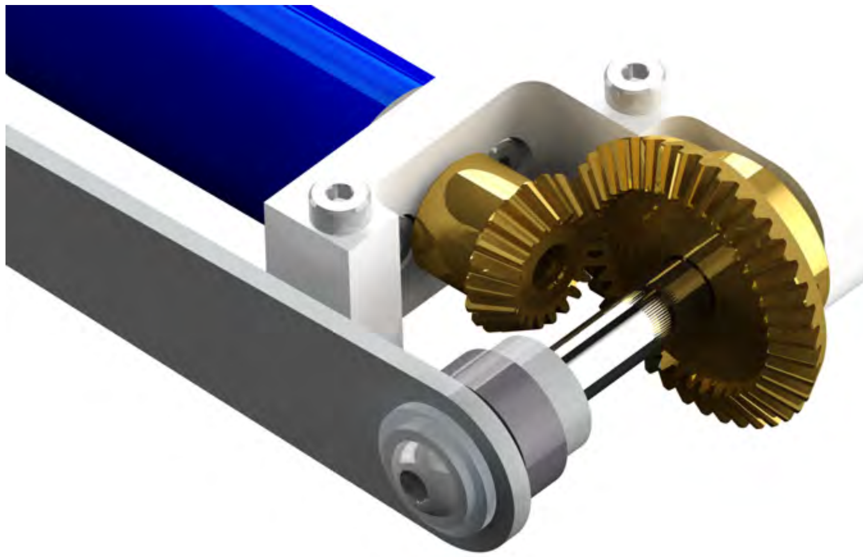


Figure 4.6: Motor Mount

seal was needed. For this design, the seal was chosen to be at the end of the output shaft to keep as much of the moving components internal. Unfortunately, standard mechanical seals are not sold in sizes small enough to fit in the RC model. However, the low rotational speed allowed for simple spring seals to be used. These work by using a rubber sheath with a coil spring on the inside which applies pressure to the turning shaft and outer wall.

Spring seals come in a variety of sizes and are easily available from a number of vendors. The shaft and arm port were designed around a stock seal to prevent trying to customize this component. A diagram of the sealing port is shown in Figure 4.7. To facilitate the transition between the spring seal diameter, the arm connection, and the internal bevel gears, a complex arm shaft needed to be machined (shown later in Figure 4.8).

### Motor Torque Requirements and Selection

During tumbling motion, it is not uncommon for the robot to support its entire weight with a single arm. Therefore, the motors were specified to work under equal or worse conditions. The required torque was calculated by using the rendered model for the center of mass and the design weight as the mass of the robot. The arm creates a moment around the shaft which must be overcome by the motor in order to tumble. Friction from the spring seal was also taken into account. A factor of safety of two was added to make sure the motors were more than adequate

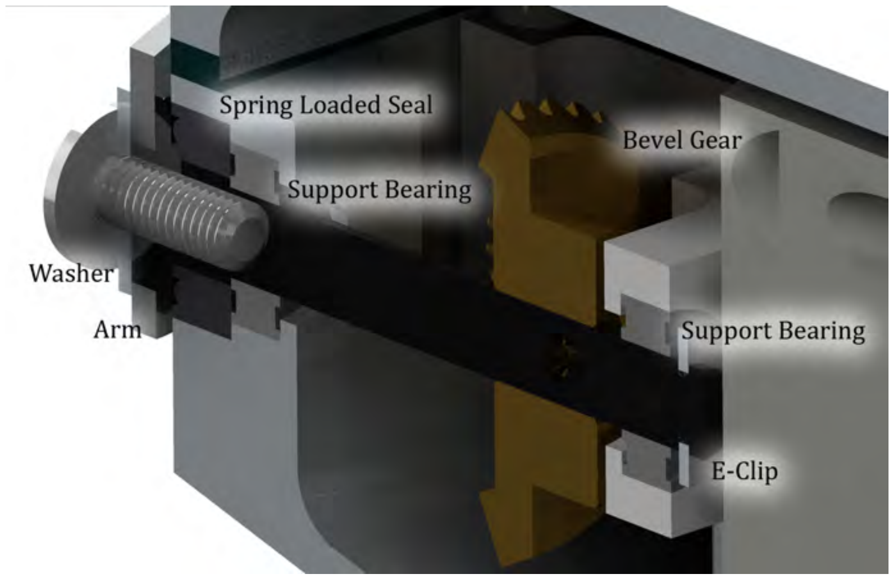



Figure 4.7: Arm Shaft and Spring Seal

for moving the robot, see Equations 4.1 and 4.2.

$$\tau_{Req} = (\tau_{SpringSeal} + (m_{Design} * g * x_{MomentArm})) * 2_{SafetyFactor} \tag{4.1}$$

$$248mNm = \big(12mNm + (0.492kg * 9.81m/s^2 * 23mm)\big) * 2_{SafetyFactor} \tag{4.2}$$

With a calculated torque requirement of 248mNm, the next step was to select a motor. Using the Maxon Motor Selection Guide [30] as a baseline and a spreadsheet created by Alex Kossett, the ideal motor and gear head combination was selected. This spreadsheet can be seen in Appendix B.2. The spreadsheet takes into consideration the nominal input voltage, the torque and speed requirements, and motor and gear efficiency among many other factors.

### Arm Design

The final aspect of the arms was the actual arm design. The original Adelopod used aluminum arms, but again, because of the design weight, aluminum was simply too heavy. Carbon fiber was found to be a suitable substitute. The problem with carbon fiber however, was that it is not easily machinable. One method for customizing carbon fiber is water-jet cutting, which also made the part more simplistic and less costly to produce. Securing the arm to the arm shaft required a key hole to be made in the arm which was chosen as a triangle to minimize play as seen in Figure 4.8.

Figure 4.8: Arm Shaft

### 4.4.3 BCU

The buoyancy control unit was critical in the success of the prototype. Benchtop testing mentioned in Section 4.2 provided the necessary information to design the BCU with enough confidence. This section goes through some additional calculations on the assembly and how it was adapted to fit inside of the Aquapod.

#### Mounting

The stock motor supplied with the peristaltic pump was too bulky to fit inside of the Aquapod. A new motor was selected which was smaller, more efficient, and operated in the correct voltage range for the battery. To interface the pump housing and the new motor, a mounting plate was designed, see Figure 4.9 (some components have been hidden for clarity). A two dimensional part was designed to be water-jet cut which reduced costs. The material was chosen as aluminum in order to make the part as small as possible while still providing enough strength to work against the torque required to turn the pump head.

To secure the pump assembly to the shell, small braces were cut into the bottom shell surface. These provided a planer restraint while the spacing between the top and bottom shell constrained any vertical movement. This prevented additional parts from being created, but did not provide mechanical mating between the pump and shell. However, since the pump outlet was connected to the shell wall through flexible tubing, the sealing was independent of the pump's position. Therefore, minor play in the assembly was tolerable.

#### Motor Torque Requirements and Selection

Various torque requirement calculations were done to determine the best operating conditions for the pump motor. However, in the end, the most reliable choice was to simply find a new motor

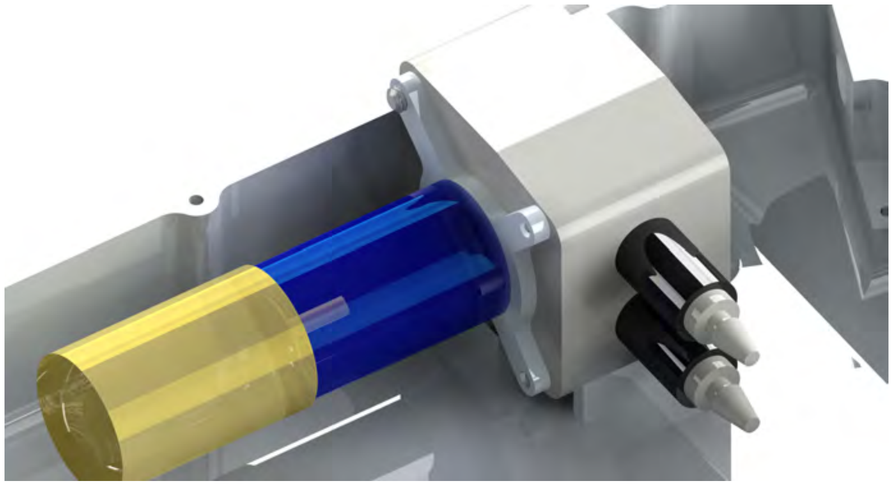


Figure 4.9: Pump Assembly and Mount

which matched or exceeded the specifications of the supplied stock motor. The stock motor was tested to run at a speed of 95rpm with a running torque of 250mNm. Again, using the motor selection spreadsheet, a suitable motor was found, see Appendix B.3 for the spreadsheet. The Maxon 2269571 motor was used with a 84:1 gear head. This combination provided a maximum stall torque of 660mNm which was well above the running torque of the stock motor.

#### Bladder Selection

One of the hardest components of the pump assembly was the bladder. This part simply needed to hold water, expand, and contract without segmenting pockets of water, much like a balloon. Numerous methods were used to create a bladder, everything from heat sealing plastic bags to sewn rubber sheets. Vendors were also contacted to try and manufacture a custom bag, but this method was not feasible for producing a small number of bags.

The most cost-effective solution was to use a normal, latex, party balloon. This was only meant as a temporary solution but proved to be fairly versatile. A more conforming shape would have been preferred, however, a double wrapped balloon was a reliable alternative and had no leaking issues.

### 4.4.4 Electronic Speed Controllers and RC Receivers

To actuate the arms and drive the pump underwater a remote control needed to be added to the design. To penetrate the water, a 72MHz receiver and transmitter were used. This frequency is one of the lowest commercially available and is commonly used for underwater model submarines; it has a range of 10-15 feet of depth in the water, which was more than adequate for the prototype.

The receiver used was built to drive servo motors at 5V, and so a speed controller was needed to turn the servo signal into a pulse width modulation (PWM) and add extra power to drive the arms and pump. A reversible electronic speed controller (ESC) was found and three were modified to run from the same power source with separate input signals. This setup allowed for a standard RC transmitter to control the arms and pump individually, turning the robot into a remote control model. See Figure 4.10 for a wiring diagram of the RC prototype.

## 4.5 RC Testing

The main functional requirement of the RC prototype was to test the hermetic seal design and verify the buoyancy control. Figure 4.11 shows the completed prototype being tested on the shore of the Mississippi River. Various other terrains and conditions were also tested along with the BCU. This section goes into some detail about the different trials through which the prototype was run.

### 4.5.1 Submersion Testing

Just as all the subsystems were tested before the prototype was designed, the prototype was constructed piece-wise to ensure all the components worked independently before being assembled completely. Once the shells and gaskets arrived, the first step was to assemble the shell with dummy internal components and do a submersion test. When no leaks were detected, a process similar to the one in Section 4.1 was followed to test the spring seals. One-by-one, the different components were added until the complete prototype was assembled.

In some locations, additional sealing was required. Places such as the power switch and pump inlet needed epoxy and a small amount of marine sealant before they became hermetic.

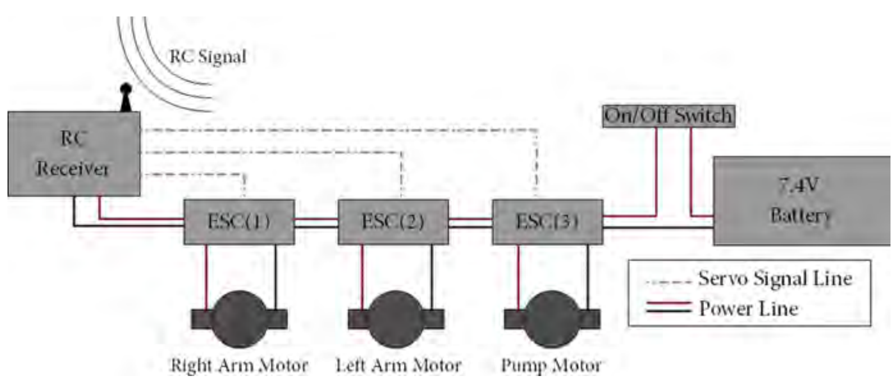



Figure 4.10: Wiring Diagram of the RC Prototype

Figure 4.11: RC Prototype Outside on Mississippi River Shore

The gasket and arms also had a liberal application of lithium grease for sealing protection and decreased friction.

The RC prototype was designed to be water tight, but the exact pressure limits were unknown. The shell and gasket were specified to withstand a depth of around five meters. However, because the prototype was controlled through a radio transmitter, it could only be tested as deep as the transmission would penetrate the water. In the case of the 72MHz transmitter, this was around supposed to be 10-15 feet; however, this is with ideal conditions and powerful transmitters and receivers. The prototype was found to only have a range of a few feet under the water, and even then the signal was unreliable. And so the robot was only tested in shallow water and small tanks where the signal could reach the robot.

Later depth testing revealed that the failure depth of the robot was actually around three meters. This was, in part, due to the deformation of the shell which will be discussed in detail below.

### 4.5.2 Terrain

One of the advantages of having a completely sealed and protected tumbling robot, is the ability to test the mobility in a number of different terrains. Though not documented or tested thoroughly, this section notes some observations of the RC prototype in various types of terrain.

- **Grass:** Much like the original Adelopod, the RC prototype was able to traverse through grass and short vegetation with relative ease. Pinching would sometimes occur when both arms moved back and inward toward the body, essentially hugging blades of grass. This could be overcome by reversing and moving one arm at a time.
- **Packed Sand:** This terrain was similar to soft floor or carpet. The robot could easily tumble over packed sand and left an interesting footprint behind.
- **Loose Sand:** The body of the robot helped traverse loose sand by essentially digging its own path. The tumbling motion grabbed the sand in front of the robot and pushed it underneath. Sloped surfaces in loose sand posed a problem, but the robot could progressively move forward with the digging motion.
- **Mud:** Mudded terrains were similar to sand, however, the lower viscosity of the mud allowed the robot to move better. The thin arms hurt the robot in this terrain when they cut through the surface rather than pushing off of it.
- **Shallow Water:** In very shallow water(<1 inch), the robot performed just as it did on whichever terrain was underneath. In deeper water (1-6 inches) the robot would still tumble as expected but required a full bladder. In submersible water (>6 inches), the weight distribution of the robot caused it to orientate itself to a neutral position. Tumbling was difficult since the buoyancy forces prevented it from flipping.

### 4.5.3 Lessons Learned and Improvements

The RC prototype worked near exactly as expected after some minor modifications during the assembly. However, the model was not perfect and provided a great deal of information on improvements that could be made to the design. This section will cover some of the major lessons learned from the prototype and some possible changes that could have been made to solve the issues.

#### Shells

The biggest issue with the shells was found while pressurizing the inside while taking in water. Not only did the internal pressure increase, but the spherical shape of the balloon applied additional force to the shell. These forces combined to elastically deform the top shell. Normally this would not be a cause for concern, however, the deformation was so severe that the material between screw locations moved enough to cause the gasket to fail. Finite element analysis (FEA) after the fact confirmed this failure, see Figure 4.12

One solution could be in the form of an expanding, but constrained, bladder. The spherical shape of the balloon and the latex material would simply keep growing and apply point pressure


Figure 4.12: FEA Analysis of Shell Deformation

on the shell. A flat, flexible bladder that was self-constrained within the boundaries of the shell would prevent the added pressure and direct those forces to the bladder material rather than the shell. Another solution would be to simply increase the thickness of the shell or strengthen the material.

### Sealing

As mentioned above, when the internal pressure increased too much the shell would deform causing the gasket seal to fail. However, there were problems with the gasket itself as well. The gasket material was a soft Butyl rubber which was very pliable and conformed to the surface of the shells well, but would also largely deform when compressed. Since the gasket was only constrained on the surface plane by the screws, when the shells were pressed together, the rubber between screw locations would push out or in and no longer contact the shells. The effect was worsened when the motion from turning the shell screws would apply a torque to the rubber and twist it out of place.

Figure 4.13 shows an extreme case of gasket deformation, however, since the arm was designed to be within a few millimeters of the body, even a small protrusion of the gasket could get caught by the arm and forced out of place.

### Arms and Motors

The arm motors selected for the RC prototype were more than powerful enough for the tumbling motion. There was no fear of stalling the motors and were generally used at 30-40%

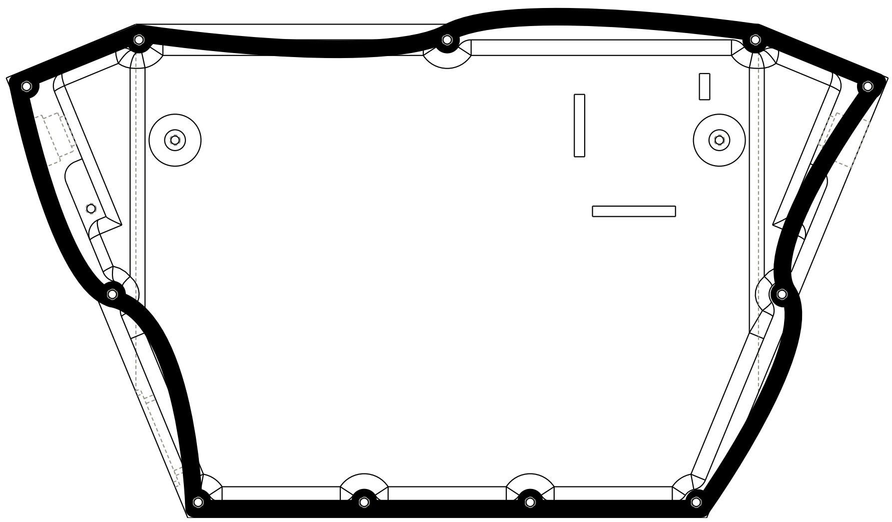


Figure 4.13: Extreme Case of Gasket Deformation

power. The gearbox and bevel gears also performed well, though flexibility in the mounting bracket would cause the bevel gears to slip when the arms got stuck, acting as a terrible slip-clutch.

Between the triangle key hole, bevel set screw, and gear alignment, there was about a 30 ° play in the arms. This was not a problem for the prototype, but would be a concern for encoders. Tightening the tolerances on the set screw and gear alignment would help, also changing the key hole to a hex shape would lessen the play.

**Pump and Motor**

The biggest issue with the pump and motor was speed. A full transition from buoyant to sunk took almost two minutes. The flow rate of the pump simply could not handle much more without increased wear on the internal components. Either a larger pump, or a different pumping method altogether would be the only way to improve the transition time. Faster motors and more flexible tubing could marginally improve the time, but not by an order of magnitude.

# Chapter 5

# Embedded Systems and Depth Design

During the 2010-2011 academic year, the Aquapod project was awarded a Rapid Massive Research Instrument Grant to help assess damage due to the Deepwater Horizon oil spill in 2010. The grant proposal called for a number of semi-autonomous Aquapods to be delivered to researchers with access to affected environments. The robot needed to be capable of collecting water samples, recording temperature and pressure, and withstand depths of up to 10 meters.

A special note about this chapter. During the 2010-2011 academic year, this author was studying abroad and had limited contact with the Center for Distributed Robotics. Since the MRI Grant needed to be fulfilled, a number of graduate students were brought in to work on the project in his place. **The work and information in this chapter should not be considered the accomplishment of the author.** Rather this chapter is included for consistency in the development of the robot. The written text, however, is original work[1].

The final outcome of the grant was the robot design pictured in Figure 5.1. Tested to depths of over 6m, this semi-autonomous iteration was capable of collecting two 10ml water samples. Sensors included both internal and external pressure, GPS, internal humidity, motor encoders, and a detachable HOBO conductivity module.

Since this chapter is not the work of the author, only a simplistic overview of the design will be presented. The hardware will be covered first, and an outline of the major changes made from the RC Aquapod. The next section will cover the electronics and system architecture along with some specifics regarding the sensors and electronic hardware.

[1] Some of Section 5.2 has be adapted from author-written segments of [1].

Figure 5.1: Completed MRI Grant Version

## 5.1 Hardware

There were a number of increased requirements for this iteration of the robot in comparison to the RC prototype. Mainly, the robot had to withstand deeper depths, be resistant to corrosive elements, and in general, be more robust. On a high-level view, the overall structure did not change drastically, but many components were upgraded or replaced to handle the more extreme conditions. This section will walk through each of the individual components of the hardware design that were changed for the Rapid MRI grant.

### 5.1.1 Shell

As mentioned in the previous chapter, the RC Aquapod had a major issue of bulging when the bladder was full. Deflections in the shell would bend the material between screw locations and eventually cause leaks. This problem was present in a few sub-iterations of the MRI robot as well. To solve this, the shell thickness was increased from 4mm to 10mm. This also added mounting support for the various ports that were installed. On the top shell, support ribs were added to lessen stress and direct deflections to areas with more thickness, see Figure 5.2.

Due to the additional shell thickness and new internal components, the overall volume of

Figure 5.2: Top Shell with Ribs

the robot increased by about 300% (including the sampling backpack). The material for the shell remained polycarbonate due to its cost and availability. Mounting brakets on the front surface were added to attach a HOBO sensor, see Figure 5.3, and screw holes were added on the bottom shell for the water sampling backpack. Inside, a dividing wall was added between the main electronics and bladder. This was done to prevent the bladder from interacting with the board components and to add some minor protection against flooding.

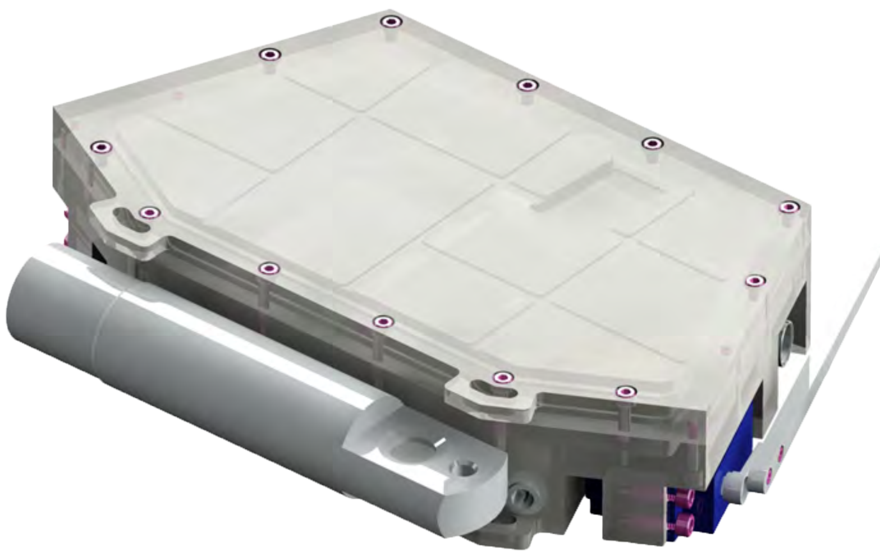


Figure 5.3: Rendered MRI Aquapod with HOBO Sensor Attached

### 5.1.2 Sealing

To withstand a maximum diving depth of 10m all open ports on the Aquapod had to be capable of handling pressures of over 30psi. The main seal between the top and bottom shells was formed with an O-ring and channel. The shell screws were also spaced more evenly to apply a more consistent force on the O-ring. In contrast to a gasket, O-rings apply a linear contact force as opposed to a surface contact, which results in better sealing with less compressive force.

With the recent development of water-proof servos, the arm motors were moved to the outside of the robot. Signal lines to both the arms and the water sampling backpack were done with industry standard push-to-connect (PTC) fittings. These ports allowed for commercial-off-the-shelf (COTS) components to be bought, instead of customizing new parts.

Two programming ports on the backside of the robot were accomplished with Ingress Protection 68 (IP68) mini-USB panel mounts. One port connected to the slave board for low level programing, and the other to the master board for data downloading and routine programming.

### 5.1.3 BCU

The buoyancy control unit (BCU) for the MRI version remained nearly unchanged. The pump tubing was altered to account for compatibility of various forms of crude oil, gasoline, and salt water. Also, a more powerful motor was selected for a higher flow rate. Finally, an angular velocity feedback encoder was added to track how much water was being pumped into the shell. However, the bladder remained a simple latex balloon, which was still robust enough for testing in clean water, but was not chemically compatible and had a number of other issues which will be discussed in the next chapter.

### 5.1.4 Arm Motors

During the development of the MRI robot, a new series of water resistant servo motors came to the market from Traxxas. These servos were specified as water-proof, but with no real maximum depth. They were modified with new O-rings and sealant to be hermetic up to the 10m depth. In addition, the internal components were modified for continuous rotation and new encoders were added. Custom encoder mounts were designed to fit inside of the servos to replace the existing electronics.

As mention above, the servo motors were placed external to the robot shell, so no additional dynamic seals were required. The arms were also changed to aluminum for chemical compatibility and ease of mounting to the servo horn.

### 5.1.5 Sampler

The main functionality of the MRI Aquapod was to dive and collect water samples in the Gulf Coast region. To collect water samples, a backpack device was designed to attach to the backside of the Aquapod. The attachment contained two 10ml vials with caps on all ends. The caps were held in a naturally closed position via an elastic band connected to both ends. The caps were also connected by steel wire to a servo. When the servo motor was run, a center hub pulled on the steel cables which opened the caps and allowed water to flow into the vial, see Figure 5.4.

## 5.2 Electrical

The embedded system design of the MRI Aquapod was split into three main components. Two of the PCBs were custom designed to meet the volume restrictions, while the third PCB was a small, COTS module from SparkFun Electronics. Splitting the electronics into separate boards made the design highly versatile and allowed for expansion with minimal effort. Different sensor suites could be interchanged and made the robot more robust to ever adapting needs.

The initial RC Aquapod had no embedded system design, therefore this iteration had to make a number of drastic changes. A three-PCB design made up of a master board, a slave board, and a fuel-gauge board let the components to be broken apart, better utilizing the space

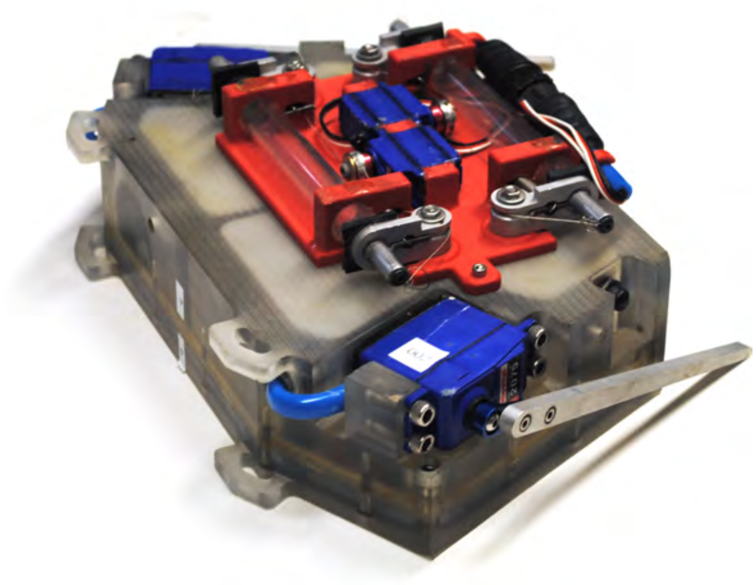


Figure 5.4: Sampler

available. The complete system architecture can be seen in Figure 5.5.

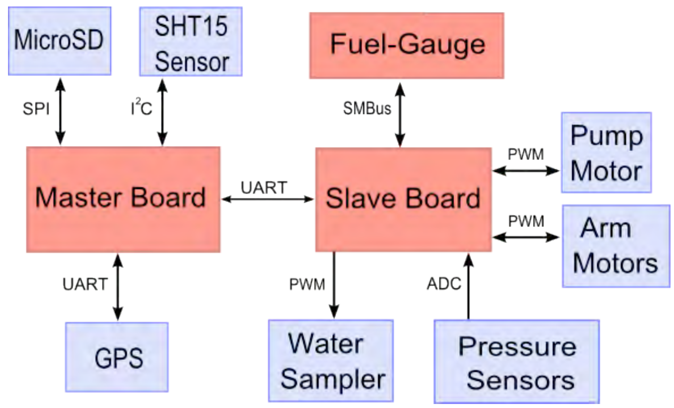



Figure 5.5: Embedded System Architecture (Taken From [1])

### 5.2.1 Slave Board

Due to the custom nature of the hardware, the slave board was designed specifically for the Aquapod robot. The main processor was chosen as the PIC16F1827 from MicroChip based on the number of required input and output pins. The slave board also includes a pressure sensor, charge recognition circuitry, internal communication devices, and the required motor drivers for five separate actuators (two arms, two samplers, and one pump), see Figure 5.6.

Figure 5.6: Slave Board

### 5.2.2 Master Board

In order to facilitate the requirements within the time constraints, a COTS module was used for the master board. The Package Tracker, sold by SparkFun Electronics, runs on an LPC2148

ARM7 microcontroller (see Figure 5.7). This unit was chosen for its 'ready to use' mircoSD card reader, on-board sensors, and GPS interface.

### 5.2.3 Battery Management

The Aquapods's power system comprised of the battery, fuel-gauge PCB, charge recognition circuitry, and voltage converters. A lithium-ion polymer battery (LiPB) was chosen to power the robot due to its thin and light-weight packaging, high energy density, and output power capabilities. When fully charged, it could source roughly 14Wh of energy to provide almost two hours of runtime with all components running. An integrated charge-management circuit was included on the slave board, which allowed the LiPB pack to charge quickly and safely and contained a pre-charge (conditioning) mode for deeply discharged batteries.

### 5.2.4 Sensors

The Aquapod had two absolute pressure sensors with an output range of 0.3V to 4.9V, corresponding to an input range of 3 psi$_a$ to 42 psi$_a$, respectively. These sensors were used to monitor internal and external pressure and act as the main decision points of the state machine.

The two arm motors featured absolute magnetic shaft encoders which reported the current arm position back the slave board which handled the PID control for arm movement. The pump motor also included a digital encoder to determine the total water volume pumped into the bladder. The sampler motors had no feedback, but operated on a standard servo open-loop signal.

The COTS master board contained a number of on-board sensors including temperature, humidity, and a 3-axis accelerometer. The board could also access the SiRF III GPS module from USGlobalSat.


Figure 5.7: Package Tracker

Finally, the Aquapod was designed to hold a HOBO Conductivity Data Logger outside the shell. This sensor operated independently of the robot and had foam insulation added to the casing to make the device neutrally buoyant. It was capable of measuring temperature and conductivity (salinity) from 5 to 40 °C and 0 to 10,000 S/cm respectively.

# Chapter 6

# Programming and Testing

This chapter continues this author's work using the platform developed from the Rapid MRI Grant. Though a majority of the work was done in the development of the robot, it was not complete. Minor hardware changes and programming still needed to be completed to make the robot functional. This chapter goes into the programming and testing that was done on the Rapid MRI platform (which will be referred to as the Aquapod v1.1), though the larger part of the work presented in this chapter is the sole effort of this author, credit should be given to Sandeep Dhull, Nick Walczak, and Dario Canelon-Suarez for their programming contributions.

The purpose of the robot was to collect water samples in Gulf coastal environments. And so the programming and testing of the Aquapod was primarily focused on diving and taking samples. The 'standard routine,' or state machines, that were programmed had the robot take in water until it sunk, monitored the depth, took samples at pre-programmed locations, and finally, expelled water until it resurfaced. Later routines would expand on this idea and include some additional functionality, but the core program was designed to dive and take samples.

## 6.1 Basic Control

Initially, communication problems between the slave and master board prevented the master board from running the state machine. Therefore, all of the functionality was initially programmed on the slave board. This section breaks down the individual functions of the robot and explains how each was programmed. The next section builds on these functions and describes how the state machine was eventually moved to the master board.

### 6.1.1 Diving

Diving was, perhaps, the easiest function to implement, but the most difficult to properly establish. Turning the pump motor on and bringing in water was a simple programming task, however, once the robot started to sink there were a number of problems that could arise. Not over-filling the bladder and knowing when to resurface was not trivial. The robot was equipped with internal and external pressure sensors. The internal pressure sensor was included to determine the bladder expansion by measuring the increase in internal pressure. The external pressure sensor measured the hydrostatic pressure, and in turn, the depth.

Using the manufacturer's datasheet for the pressure sensors, depths were calculated as analogue-to-digital converted (ADC) values and hardcoded into the program. It was later discovered that these values varied fairly significantly with ambient pressure from day to day (detailed below). Using the calculated values pressure, it was possible to determine the depth of the robot, which was the primary driving variable of the state machine.

The internal pressure sensor was a poor method for controlling the cut-off time of the pump. Varying ambient pressure would, at times, over-fill the bladder, or not fill it enough to begin sinking. Therefore, the external pressure sensor was used to determine when the robot started sinking. A threshold of a quarter meter was set as the sinking trigger. Once the robot registered that the external pressure was greater than a quarter meter, the robot was assumed to be sinking and the pump shut off. The internal pressure sensor was still used as a safety feature in case the robot was taking in air, or experience other failures so that the bladder would not continue to expand and subsequently explode.

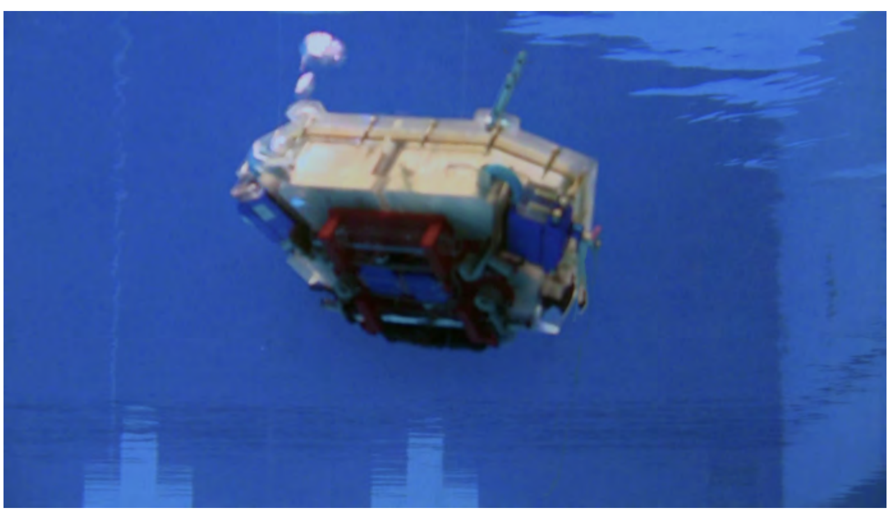


Figure 6.1: Aquapod Sampling During a Dive

### 6.1.2 Sampling

Along with diving, the second major function of the Aquapod v1.1 was to collect water samples. The sampler was a modular backpack that attached to the shell and used PTC tubing to connect with the slave board on the inside of the robot. Two servo motors were the only electronics on the backpack and required a ground, power, and two signal lines. These servos were unaltered and operated on standard servo signal lines. The microcontroller was programmed to simulate the pulse signal to the servo which was interpreted by the microprocessor inside the servo and activated an internal PID controller to move the servo arm.

A servo signal sends a high pulse every 20ms that varies in width from 1 to 2ms. A typical servo has a $180^\circ$ range, and so a pulse width of 1.5ms corresponds to the neutral position ($90^\circ$) with $\pm 0.5$ms to each end of the full range. On the sampler, only two positions were necessary, open and close. To prevent backdriving the servos and to ensure the samplers were initially closed, the program started the motor at the $0^\circ$ position. When a sample needed to be taken, a $180^\circ$ signal was sent and kept for one second. This allowed enough time for the servo to move into the open position and for the air inside the vials to be replaced by water. After the delay, a $0^\circ$ signal was sent to close the sampler again.

### 6.1.3 Tumbling

One of the most basic functions of the Aquapod v1.1 was not programmed until later in the project. Simple tumbling on land was pushed off due to the importance of diving and sampling. Another cause for the delay was the overlapping signal lines of the sampler and arms. The chosen microcontroller did not have enough output pins for separate lines to the arms and sampler motors, so a switch was placed to change the destination of the output pin. In order to control the switch, a signal had to be sent through a multiplexer.

Once the signal line was routed correctly, a standard PWM signal was sent through motor driver which supplied power to the arm motors. Code was written to change the direction and duty cycle of the arm motors using simple function calls. The arm encoders supplied position feedback and were used in control algorithms. To keep calculations and programming space to a minimum, only proportional control was implemented. However, up to this point, any control of the robot was through a pre-programmed routine.

### 6.1.4 RC Control

One of the major concerns about the robot was the lack of remote communication and control. Built into the design of the slave board, but not yet implemented, was the ability to take radio control input. A radio transmitter sends out a signal through a FM channel to the receiver. Since one servo pulse is only 2ms long, but sent every 20ms, multiple signals can be stacked together during one pulse. This PPM (Pulse Position Modulation) signal transmits the Servo A pulse first, then a 0.3ms delay, then Servo B and another 0.3ms delay, and so on, see Figure 6.2. The remaining dead space is the synchronization pulse that allows the receiver to know when the first signal starts.

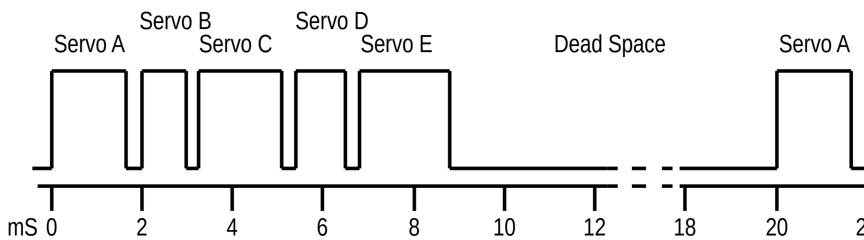



Figure 6.2: PPM Signal Example

The receiver takes this signal, filters it, and separates the different channels. Breakout pins on the receiver allow individual servos to be connected and controlled. However, only one input pin was available on the slave board, so the raw PPM signal needed to be intercepted and decoded. A stock radio receiver was taken apart and probed for the PPM line, which was then connected to the slave board (see the white wire in Figure 6.3). Code was written to separate the signals and control the arms and pump. This allowed remote controlled tumbling on land, much like the original RC Aquapod.

### 6.1.5 Swimming

Another concern about the Aquapod v1.1 was the lack of water mobility. In order to collect water samples, the user would have to deploy the robot in the exact vertical location where the samples were to be taken. This significantly reduced the usefulness of the robot, since the user already had to be where they wanted to take the samples. The original concept of the Aquapod was to sink to the bottom and use the waterbed to tumble into position. However, early tests


Figure 6.3: RC Receiver

showed that tumbling underwater with the Aquapod v1.1 was impractical. Also, it would be impossible to get a GPS signal for path planning. Therefore, a method of water surface mobility was desired.

Without adding additional actuators, the first idea was to simply to use the arms to swim. Various tests with the RC transmitter were done, but the arms were difficult to synchronize by hand. A quick flipper motion appeared to be more effective than a windmill sweep. Code was written to control the motion of the arms and convert the RC input signal to flipper amplitude. Rubber paddles were also attached to the arms for more surface area and to act as a fin, (see Figure 6.4).

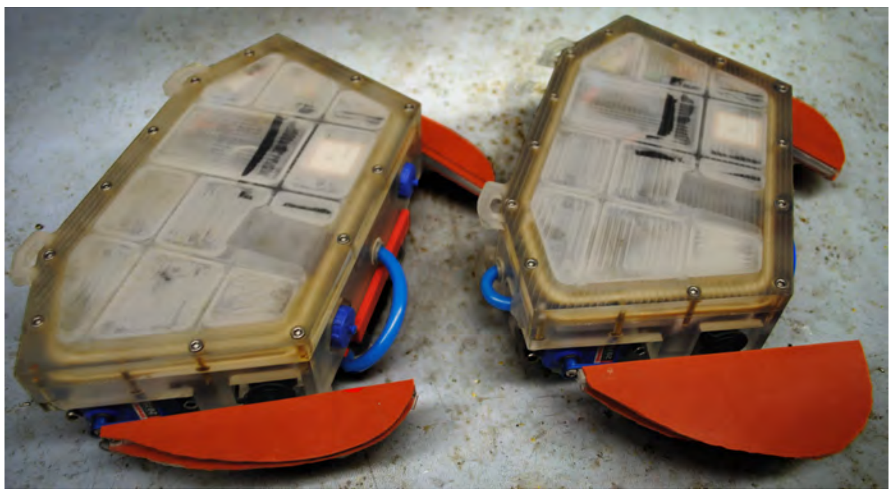


Figure 6.4: Fin Designs

To optimize the swimming motion, a design of experiments was done on a variety of parameters including the amplitude, speed, fin size, and phase. The results from the experiments showed that out-of-phase arms with a small amplitude and high speed produced the best swimming motion. Further testing showed that the neutral position of the arms should be just below the water, but not high enough for the fins to break the surface tension of the water during their stroke.

In theory, the swimming concept was an ingenious method for moving on the water surface. In practice, however, the swimming motion did not produce enough thrust to swim against even the smallest current. Moving at around 2in/sec, it was functional, but impractical.

## 6.2 Package Tracker State Machine

Once all of the low-level controls were programmed and tested, and the communication issues between the slave and master board were resolved, it was time to move the state machine on to the master board. Programming memory space on the slave board was extremely limited, often preventing all the features being included at once; therefore, to run the entire state machine–and also to utilize the GPS, data logging, and other sensors–the master board needed to be programmed as well. For convenience, the master board will be referred to as the Package Tracker (PT).

Any state machine code on the slave board was stripped away, leaving only the basic commands. After a quick start-up routine to ensure all the motors were either off or in a closed position, the slave board sat in a state of continuous waiting and would do nothing without a command from the PT. The PT ran the state machine and sent commands or requested sensor information from the slave board. This communication was done through a custom serial protocol in 9-byte packets, see Table 6.1.

| Start | Cmd | Payload1 | Payload2 | Payload3 | Payload4 | CRC16 | CRC16 | Stop |
|---|---|---|---|---|---|---|---|---|
| Start | Pump | DIR | ONOFF | N/A | N/A | CRC16 | CRC16 | Stop |
| 0x1a | 0x20 | 0x01 | 0x01 | 0x00 | 0x00 | 0xB3 | 0x5D | 0x1b |

Table 6.1: Command Packet Format and Example

Bytes 6 and 7 were checksum bytes to ensure all the information was transmitted correctly and were done in a cyclic redundancy check (CRC). There was also a command acknowledgment system in place to ensure that not only was the command transmitted correctly, but also understood correctly. For every command packet or data request the PT sent, the slave board responded with an acknowledgment packet containing the original command and–if requested–the data, see Table 6.2. Once, the slave board received a command, it would apply the payloads to the motors, or act accordingly, and then send the acknowledgment packet.

| Start | Ack | Payload1 | Payload2 | Payload3 | Payload4 | CRC16 | CRC16 | Stop |
|---|---|---|---|---|---|---|---|---|
| Start | Press | Int HI | Int LO | Ext HI | Ext LO | CRC16 | CRC16 | Stop |
| 0xa1 | 0x22 | 0x00 | 0xF3 | 0x00 | 0xE2 | 0xA1 | 0x03 | 0xb1 |

Table 6.2: Acknowledgment Packet Format and Example

With the communication system in place and all the functionality of the slave board controllable through command packets, the state machine was coded for the PT. More programming space and additional sensors allowed for a more complex state machine and additional fail-safe checks were put into place. With many land-based robots, software errors could prevent the robot from working, but would most likely not destroy the robot. For the Aquapod v1.1, a failure during dive could be the loss of a robot or flooding of the internal components. Therefore it was critical that a number of fail-safe checks were in place to always allow the robot to resurface.

The final state machine for the robot is pictured in Figure 6.5. As shown in the state machine diagram, once the bladder is filling, all paths lead to emptying the bladder. This was done to ensure that the robot would never get stuck in a situation where it did not try to resurface. Mechanical failures, environment constraints, or power shortages could still interfere, but programming-wise, the routine would always end with an empty bladder. Additional hardware checks were also added to the slave board to look for mechanical failures in the arms or pressure sensors. These checks would put the robot into an emergency empty bladder state, ignoring any further commands from the PT.

Figure 6.5 shows how the Aquapod v1.1 took samples using the external pressure sensor. While filling the bladder, or while sinking, the external pressure was continuously monitored to determine if the current depth was equal to one of the pre-programmed sample depths set by the user (see Section 6.3.1). If so, a sample was taken, and the robot continued to dive. Once all the samples were full, or the robot was no longer sinking, the bladder was emptied and the robot returned to the surface.

During the entire state machine, every event was logged in central log file. The internal and external pressure, temperature, humidity, and accelerometer were also logged in this file along with a time stamp from when the robot was first powered on. These log files were saved to an

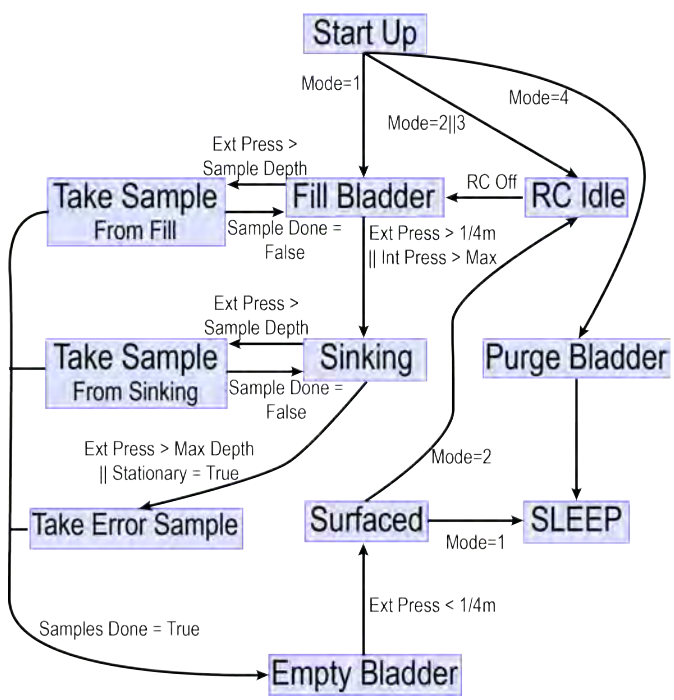



Figure 6.5: Package Tracker State Machine

SD card and could be easily accessed through the USB port on the Aquapod. This process is described in detail in the next section.

## 6.3 User Interface and Data Collection

This section will talk about the graphical user interface (GUI) that was developed for the robot, along with how the user would program a diving depth and access the stored data. This will be accomplished by simply walking through a typical example of using the robot as a water sampling device.

### 6.3.1 Programming the Robot

The first step in programming the robot was to connect the Package Tracker to the computer via the USB port (see Figure 6.6). Once connected, the SD card on the PT would mount on the computer as a mass storage device. The raw data and configuration files could be copied or modified there, however, to make the experience more user-friendly, a GUI was developed in Matlab to set and view these files automatically.

Figure 6.6: USB Port Location

When running the GUI from Matlab, the script would first explore all the available drives to find the location of the Aquapod, see Figure 6.7. Once an Aquapod was detected, the GUI would ask the user if they wished to view the data or configure the robot. To configure the robot to collect a sample at a particular depth, the user would enter information on the screen in Figure 6.8. Here, different operating modes could be selected (see Section 6.3.2), along with setting the ambient pressure, and the sampling depths. Once the changes had been made and

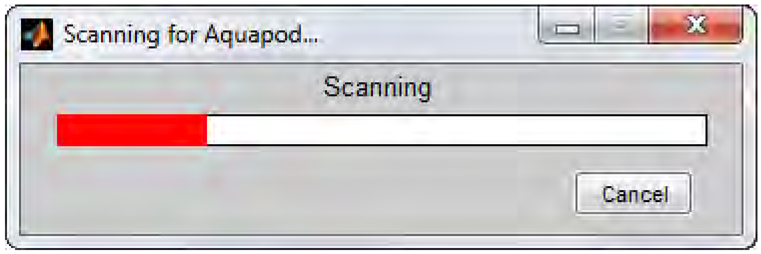



Figure 6.7: GUI Scanning for Aquapod

saved to the robot, the indicator light would turn green, and the user could unplug the USB cable.

### 6.3.2 Operating Modes and Usage

After the robot had been configured and turned on, the robot would enter the state machine described in Figure 6.5. There were four operating modes in which the robot could travel through the state machine.

- **Instant Dive:** Once powered on, the pump would immediately take in water and dive. After collecting samples, it would expel water and return to the surface.
- **Remote Dive:** Once powered on, the RC receiver would be turned on and the robot could be controlled through the RC transmitter. The transmitter could switch between

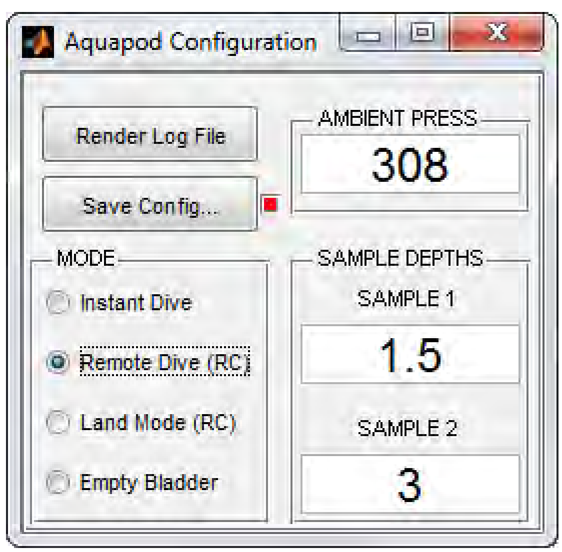



Figure 6.8: GUI Configuration Screen

tumbling and swimming modes and could trigger the dive sequence. Once returned to the surface, the RC mode was again activated to allow the user to drive the robot back.

- **Land Mode:** Here, the robot would only remain in RC mode and ignore any commands to dive. This mode mainly demonstrated tumbling and swimming abilities.
- **Empty Bladder:** This failsafe option allowed the user to bypass the state machine and expel any remaining water from the bladder.

While in any of the first three modes, the Package Tracker would continuously log data from the various sensors on the Aquapod. This data was recorded and stored on the SD card in a text file which could be accessed through USB.

### 6.3.3 Accessing and Visualizing Collected Data

Once the robot was finished collecting samples, the USB cable could be reconnected and the GUI restarted. Selecting to generate the log file would prompt the Matlab script to filter and sort the log file and separate the various sensor data into different graphs. Once finished, the GUI would present the user with the screen as shown in Figure 6.9. This screen contained three graphs, one for the depth and pressure, one for accelerometer, and the last for temperature and humidity. The depth and pressure graph would also display state changes and events. This confirmed when samples were taken (blue vertical lines) and when the pump state was changed (black vertical lines). From here, the user could go back and re-configure the robot for another run, or save the log file for viewing later.

## 6.4 Houston Feedback

On February 24th, 2012, three Aquapod robots were brought to Houston, Texas for a two day demonstration and user testing with a team of research biologists. Steven Pennings and his graduate students were selected as consultants to demonstrate and discuss the capabilities of the Aquapod, its limitations, and get feedback on useful functions and new directions to head in. The meeting consisted of an informal presentation and on-campus demonstration of the robot, along with field testing in various types of marshland terrain.

The robot was first demonstrated in a pond on the University of Houston campus. After gradually swimming to the center of the pond, the robot was commanded to dive and sank to the bottom. Air bubbles confirmed that two samples were taken successfully. However, after retrieving the robot and investigating the log files, it was determined that an internal pinch in the bladder system prevented the robot from completely expelling its water. A second run performed as expected.

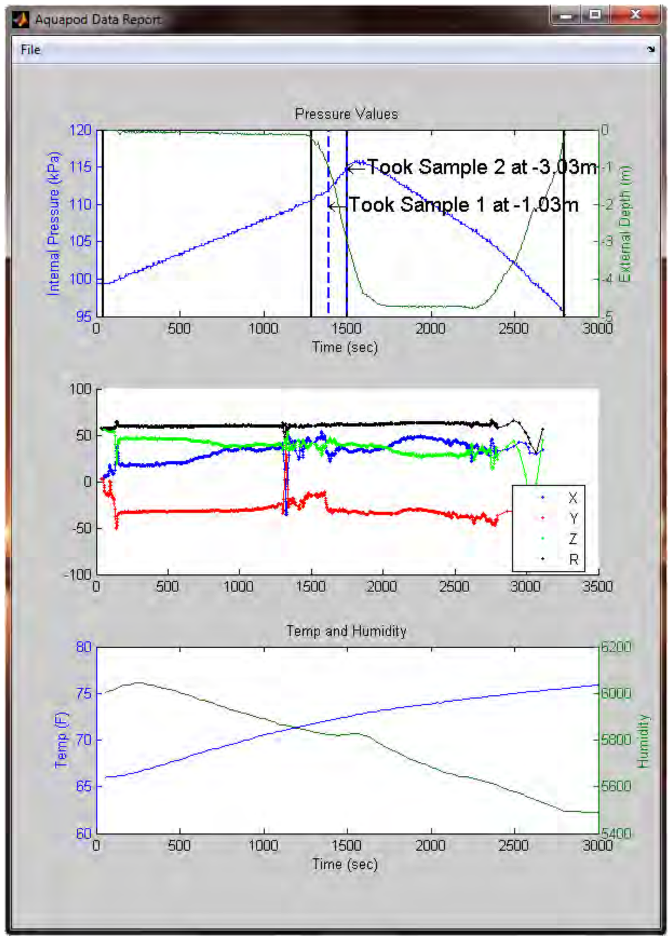



Figure 6.9: GUI Data Graphs

On the second day, the Aquapod was tested at the university field site in four different environments. The first site was a concrete holding with a sloped entrance and little vegetation. The robot tumbled in, swam around for a while, and tumble back out. The concrete slope made an ideal surface for tumbling and allowed the robot to grip the floor. It had some difficulties directing its swimming but was able to get in and out with minimal navigating.

The second site was a small pond with a little vegetation. The Aquapod tumbled in, swam out about 15 feet and then dove. The samples were taken, but after looking at the log file, it appeared to be too heavy to resurface. This was in part due to the added tether, and also the unaccounted collective weight of mud, paddles, and other minor additions to the robot. With a second try, and the sampler disabled, the robot dove and surfaced as expected.

The third site had significantly more vegetation. The sampler backpack was completely removed and the robot attempted to navigate in the weeds and muck. It consistently got stuck on vegetation and could not swim through weeds. Using the paddles made some navigation more difficult but prevented the arms from pushing deep in the mud.

The final site was a mud flat (flat, wide ditch with very mucky, thick, mud). The robot was able to traverse this terrain much easier than all the others, and even better than a human could. The paddles in this environment were an advantage since it prevented the arms from cutting into the mud. Even in thick mud locations the robot was fairly mobile and able to climb slopes and out of puddles. After several hours in discussion with the biologist, a list of user requirements was formed in response to some of the limitations and features of the current Aquapod design. Coming at the end of the project, these user requirements should be viewed as considerations for the next Aquapod project, or perhaps even a completely new solution.

- Robot must last a minimum of 12 hours for a complete tidal cycle, preferably would be able to take readings over the course of 28 days.
- GPS tracking for measurement locations is crucial, could need to be accurate within 5cm for certain applications.
- Temperature, pressure, and conductivity (salinity) are the basic sensors needed for most environmental monitoring tasks.
- Needs a wireless range greater than 50 feet.
- Must be able to sample an area in 5-10 minutes to be considered a single instance in time.
- Must be able to withstand strong currents and waves.
- Capability to house one large (a liter) water sample, or many (30+) small samples.
- Should be compatible with current commercial sensors.

Overall, the Aquapod had trouble in mid- to heavily vegetated areas around and in the water. However, in comparison to similar sized robots, the Aquapod was able to traverse these areas better. It did perform exceedingly well in mud flat areas with thick mud. It was able to navigate, with minimal effort, on mud in which other robots and humans would get instantly stuck. The Houston demo was extremely informative and put the robot in perspective in regards to environmental monitoring, its limitations, but also its novel features and expected applications.

## 6.5 Issues and Lessons Learned

Even after everything that was learned from the RC Aquapod, the Aquapod v1.1 added more hardware complexity and electronics. Therefore, the design had a number of problems that could be improved on future iterations. This section goes through some of the components that caused issues and general areas of shortcomings.

### 6.5.1 BCU

Since the BCU was largely unchanged, the Aquapod v1.1 had the same issues as the RC prototype. Mainly, the latex balloon would often get tangled and pinch when expelling water, and would put added pressure on the shell when expanding in a spherical shape. Also, even with the increased torque on the pump motor, the transition time still took a minute and thirty seconds to complete.

### 6.5.2 Shells

Once the shell thickness increased to 10mm, there were no longer major deflections in the shell casing from increased internal pressure. However, some of the components were moved to outside of the shell and data/programming ports were needed. A total of five holes were added from the RC Aquapod. These ports were occasionally the cause of shell flooding due to a deformed O-ring or user negligence. Ensuring the robot was properly sealed before every test made the robot cumbersome to use and dismantle. Improperly placed shell screws also created the need to unmount the servo arms to gain access to the electronics.

### 6.5.3 Arms

The arm motors were selected as water-resistant servo motors to eliminate the need for a dynamic seal. However, even by starting with a COTS product, several hours of modifications needed to be done to each servo before being installed on the Aquapod. During the installation process, wire terminals could only be connected once the wire tubing was placed through the shell. This prevented simple replacement of servo motors and made swapping out new motors a time consuming process. In addition, running the PTC tubbing from the shell to the servos created two loops on the shell of the robot that would hinder tumbling and get caught in vegetation.

### 6.5.4 Backpack

The original idea of the backpack was to be a modular unit in order to create different sensor suites. However, only the sampler backpack was created and the internal electronics were built specifically to be used with the sampler. The PTC fitting also prevented the Aquapod from not needing the backpack attached at all times. Without the backpack, the tubbing would hang off the robot like a tail and get caught in vegetation or limit tumbling abilities.

### 6.5.5 Package Tracker

The Package Tracker board from Sparkfun Electronics was only purchased due to time constraints on the original MRI Grant. Numerous hardware and firmware bugs forced Sparkfun to discontinue this product, and so even in perfect condition, it was no longer a viable option for the robot. This board also contained no external communication devices for talking to a host or other robots. The only wireless communication was done through the one-way RC transmitter.

In addition to being a discontinued product, the boot-loading firmware of the Package Tracker caused a number of mysterious behaviors. This mainly happened while flashing new firmware onto the chip. Loading the new file would take up to three trials, but would typically hold on the fourth attempt. Slight variations in the code would randomly cause the firmware to break. This greatly hindered the development process since seemingly minor changes (e.g. adding two variables together, or simply declaring a new variable) would prevent the firmware from loading.

# Chapter 7

# Final Prototype

The Aquapod project was awarded a second, less substantial, grant in the aftermath of the 2011 Japan earthquake. This grant called for two additional robots to be made to search for traces of radiation in urban rubble, see Figure 7.1. However, due to time constraints, no drastic changes were made to the design of the Aquapod. Therefore, this chapter covers some of the incremental improvements made to the v1.1 design. This iteration used a majority of the same hardware and system architecture components but addressed the limitations discussed in Section 6.5. Again, though the majority of the content in this chapter is the work the author, credit should be given to Dario Canelon-Suarez for his contributions on the overall design.

Figure 7.1: Rendering of the Japan Robot

## 7.1 Hardware Improvements

As mentioned in the introduction, there was not enough time for a major revision of the design from scratch. However, a number of improvements were made to the overall design. The hardware aspects are discussed in this section including the BCU, shells, ports, arms, and new backpack module. See Figure 7.2 for a view of the inside components.

### 7.1.1 Bladder

The balloon was replaced with a non-latex intravenous bag. The bag was self-restricting in size and had a rectangular shape which conforms better to the inside of the Aquapod. This component was significantly more robust and durable than the party balloon, which removed the need for a separating wall between the bladder and electronic components.

### 7.1.2 Shells

A few minor improvements were made to the shell. Since the HOBO sensor was moved to the backpack (see Section 7.1.5), the mounting brackets on the outside shell were taken away, along with adding rounded edges and taking 4mm off the overall thickness. This was done by re-arranging the internal components to lay flat on the bottom instead of stacked.

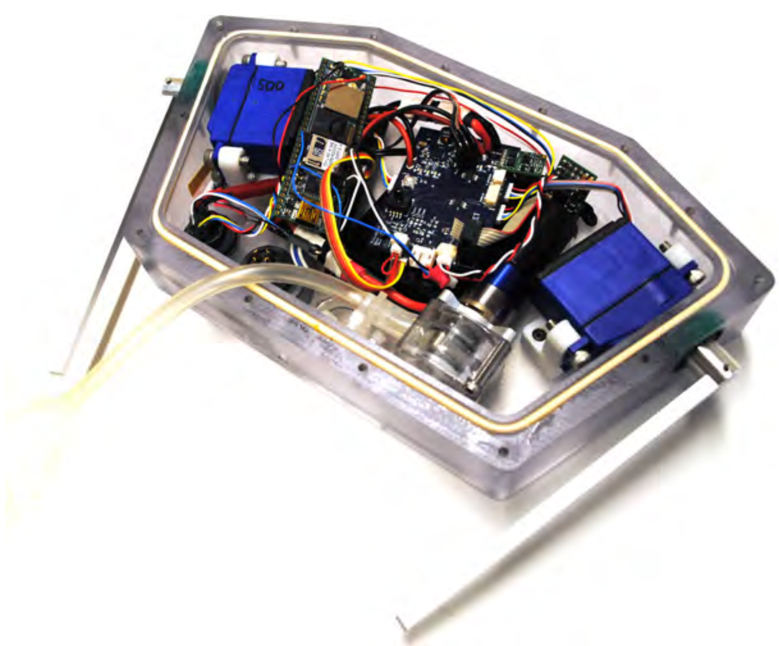
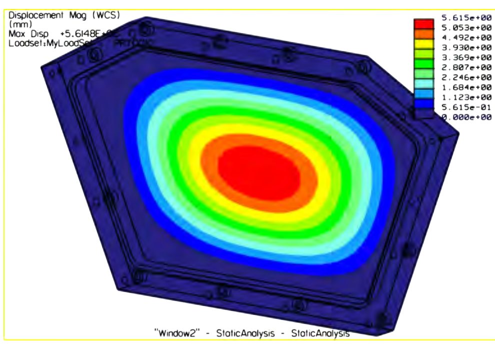


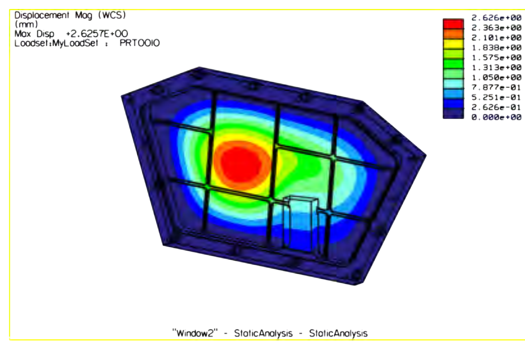
Figure 7.2: Inside the Japan MRI Aquapod

The top shell ribs were also modified to reflect an FEA analysis of the deflection due to internal pressure. Both a model with no ribs and the v1.1 design were simulated with 50psi of internal pressure. The maximum displacement location and stress points were recognized and a new rib structure was designed. The displacement of the shell was reduced from 2.6mm in the v1.1 design to 1.8mm, and the maximum von Mises stress was reduced from 1.53GPa to 0.57GPa.

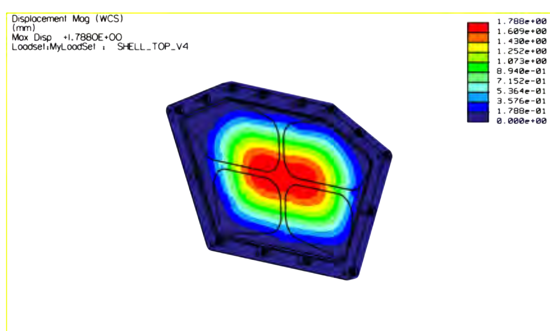
This iteration of the robot was meant to be more robust and modular than the Aquapod v1.1. Inside the shell, four mounting points were added to the bottom surface to install a rapid prototype tray. This tray can change with the addition of new components or adapt to new arrangements. The tray also allowed components to be assembled on the tray before simply inserting into the robot. The tray in Figure 7.2 consisted of two mounting brackets for the servo motors. However, this could be expanded to include mounts for the pump and electronic components as well.




(a) No Ribs




(b) Gulf Ribs




(c) Japan Ribs

Figure 7.3: FEA Displacement Analysis of Rib Designs

### 7.1.3 Ports

Minimizing the number of holes in the Japan Aquapod was a priority for much of the design. The charging port, programing port, and data port were combined into one opening, see Figure 7.4. This allowed for a more robust, IP68 verified, panel mount connector to be used. This connector was copied for the backpack port as well. The overall port number was reduced from 10 to 7 with the new design. It also eliminated all of the PTC connectors which hung off the robot reducing the chance of snags.

The power switch was replaced with a magnetic reed switch. A three-amp reed switch was installed along the perimeter of the shell under one of the arm servos. A non-through hole was drilled into the side of the shell which provided a cavity for a magnetic pin. When the pin came in proximity of the switch, the field closed the circuit and turned the robot on. A small disc of steel was secured to the bottom of the hole which held the pin magnetically in place. A groove cut into the top of the pin allowed for easy removal when turning the robot off.

### 7.1.4 Arms

The arm motors were brought back into the shell to simplify the shell features and remove the need for highly modified servos. Going back to the dynamic spring seal allowed any standard servo to be used without the need to water-proof the casing. This also made replacing the

Figure 7.4: Back Ports

servos much faster. The original RC spring seal was only rated to 10psi, however, a slightly larger, double-spring, lip-seal was found with a maximum pressure of 90psi (static) and 50psi (dynamic)–well above the required 10m depth.

The spring seals were press-fit into counter-sinks on the shell and further secured with sealant. Lithium grease around the shaft hub provided lubrication and additional sealing on the rotating parts. Small, rapid prototyped, mounting brackets secured the servos in place and align the shaft with the spring seal opening. A modified COTS shaft adapter was attached to the servo and the arm outside the shell. The adapter was milled and tapped for the arms to sit inside the shaft and secured by set screws from the outside.

### 7.1.5 Backpack

The backpack support was designed to be open-ended and extremely modular. Simple shells can be designed with specific sensor suites, or new functionality. They can be either self-contained, or interact with the new master board through the second backside port. By using this port, the backpack almost becomes a 'plug-and-play' device that can be exchanged or upgraded without having to change the physical hardware on the Aquapod. New firmware and code would need to be added to interpret the signals coming from the backpack. However, it would be possible to send signals which identify what backpack is currently attached, and let the software act accordingly.

At the time of this thesis, a backpack module was under development that incorporates the components of the HOBO sensor. Since the HOBO sensor is completely self-sufficient, no interaction with the secondary ports was required. However, if a Geiger counter is added to this backpack, the radiation sensor could communicate with either the slave board or directly with the master.

Mechanically, any backpack design attaches to the bottom of the Aquapod through four screw mounts. Once the data ports are aligned and pressed together, the backpack is secured to the shell by these screws (see Figure 7.5). On the v1.1 iteration, the thinnest backpack design was limited to the diameter of the PTC connectors which was 22.3mm, which made the addition bulky and had loose wires hanging from the edges. New backpack designs were limited to length of the SwitchCraft female port which were 15.3mm. This reduced the overall thickness of the robot, conformed better to the shell, and eliminated any loose wires.

## 7.2 Software Improvements

Again, the scope of this grant did not allow for a complete revision of the electronic components. The slave board, power management board, and battery remained the same from the v1.1 Aquapod. The only electronic hardware change was the replacement of the master board

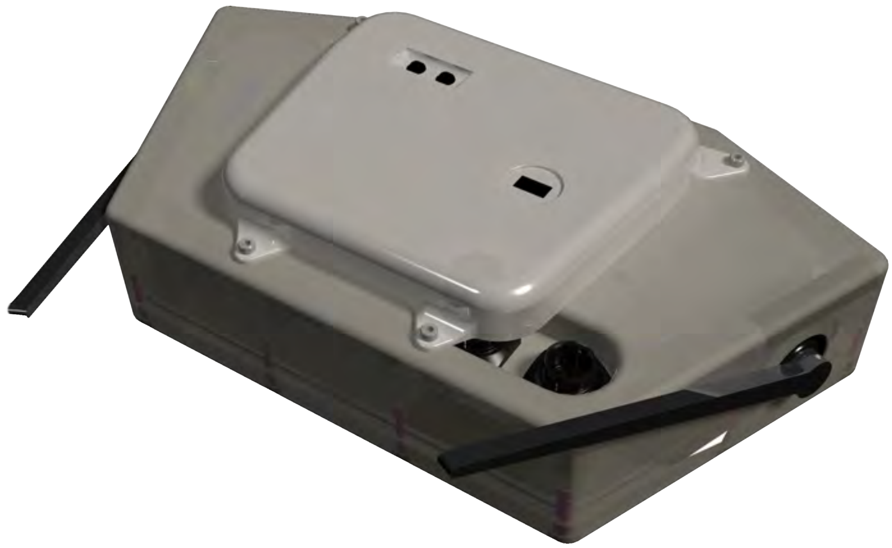


Figure 7.5: Possible Backpack Design

from the Package Tracker to the Gumstix Overo Air. As mentioned in Section 6.5, the Package Tracker was discontinued by SparkFun, so a new replacement needed to be found in order to ensure the future success of the project. However, with the switch came many firmware changes which this section will discuss.

### 7.2.1 Gumstix Overo Air

The Gumstix company produces small form-factor computers which are approximately the size of a stick of gum. The current line of boards at the time of this Thesis was called the Overo COM (Computer-on-module). These boards contained an ARM Cortex-A8 OMAP35xx processor, and could run full versions of certain Linux distributions, including Ubuntu.

The Overo Air (see Figure 7.6) was chosen for the Aquapod because of the 802.11g wireless and Bluetooth connections. Additional breakout boards could be attached to the Overo, which provided access to ports for USB interface devices, such as keyboards, and even HDMI video. The Pinto-TH breakout board was selected due to its small size and gave breakout vias to 60 of the 140 pins, USB access, and level shifting.

By moving to the Overo Air, the robot lost a number of sensors including the 3-axis accelerometer, humidity, and temperature. However, the Pinto-TH board, gave access to a number of PWM and GPIO pins on the ARM processor. Though not ideal, a wide variety of new sensors or devices could be connected directly to the Overo.

Figure 7.6: Gumstix Overo Air with Pinto-TH

### 7.2.2 Wireless Communication

As mentioned above, the Overo Air was equipped with both 802.11g and Bluetooth wireless. At the time of this thesis, the wireless communication was only being used to transfer program files and execute commands. Section 8.1.2 goes into detail on the different features possible through these new connections. By operating under a Linux shell, it was possible to connect with an Aquapod through standard SSH protocols over the broadcasted ad-hoc network. Once connected, command execution and file exchange could be done on a remotely connected device such as a laptop or phone.

### 7.2.3 Programming

Since the Overo board operated with a standard Linux distribution, executing routines and programs was the same as running programs on a standard desktop. Code was written on a Linux desktop computer, cross-compiled for the ARM processor, and then transfered to the Overo. Once on the board, a remote connection or direct serial connection could run the program through terminal commands.

Previously, the slave board and Package Tracker communicated through a 9-byte serial protocol. To prevent a complete revision of code on both boards, code was written for the Overo to duplicate the 9-byte protocol. From the perspective of the slave board, nothing had changed; it still received and executed commands over the serial line. Because of the standardized command set, multiple routines could be added to the Overo board and executed individually without the need to reboot or reprogram the slave board.

However, at the time of this thesis, the first priority was to program the Japan iteration to the same capabilities of the v1.1 robot. Therefore, the same state machine described in Section

6.2 was transfered to the Overo board. However, since the Overo board did not have a wired USB connection and mount as an external mass storage device, the GUI was not able to directly transfer. Instead, the user entered the parameters of the state machine during the program execution. Future work could include adapting or creating a new interface through the terminal window.

# Chapter 8

# Next Steps and Future Work

For as much work that has been done with the Aquapod, there is still much left. This chapter describes two levels of work needed to be done. This first level is immediate items for which there was simply not enough time to complete, but are necessary for the current phase of the project. The second level includes any recommendations for future work should other researchers continue this project. The latter section also includes some of the anticipated uses and further developments of the project as a whole.

## 8.1 Next Steps

This section provides details on some of the remaining work that should be done to improve or complete the final design described in Chapter 7. All of the work described in this section can be done with the current Japan MRI iteration of the robot with no or small modifications. These items have been left out of the project due to time constraints.

### 8.1.1 A New BCU

The biggest personal criticism of the Aquapod is the exceedingly long transition time. The peristaltic pump wastes a lot of energy trying to squeeze water through a tube without contacting the fluid. This is great for medical and food applications, however, it is unnecessary in this situation. A more efficient pumphead from MicroPump was found with a significantly higher flow rate, while keeping a similar maximum pressure rating. Also using a more efficient brushless motor, the MG1000 pump is capable of flow rates up to 1500ml/min which can fill the 100ml bladder in 4 seconds! However, using the brushless motor requires an external motor driver which takes up weight and volume inside the shell, see Figure 8.1. Due to delays in the manufacturer's production, this pump was not available at the time of this thesis. Care was taken to ensure there was enough space and weight for this pump to be added later on. A new tray design would be required to add this component.

### 8.1.2 Advanced Programming

With the addition of the Gumstix ARM processor, a whole new level of programming on the Aquapod robot has opened up. Since the programs are being executed within a Linux shell, multiple routines can be loaded at the same time and run on command without the need to re-flash the entire processor. This could also facilitate remote debugging and self-testing. Many of these features are already implemented on the Adelopod platform which contained a previous version of the Gumstix board.

New methods of control and communication are now possible with the addition of 802.11g wireless and Bluetooth. Programming and data collection can be downloaded to phones, or remote control can be done through touch screens or even Nintendo Wii-motes. Additional sensors that operate on digital signals can be directly wired to the Pinto-TH breakout board. This should include some of the sensors lost in the transition from the Package Tracker, most importantly, an IMU.

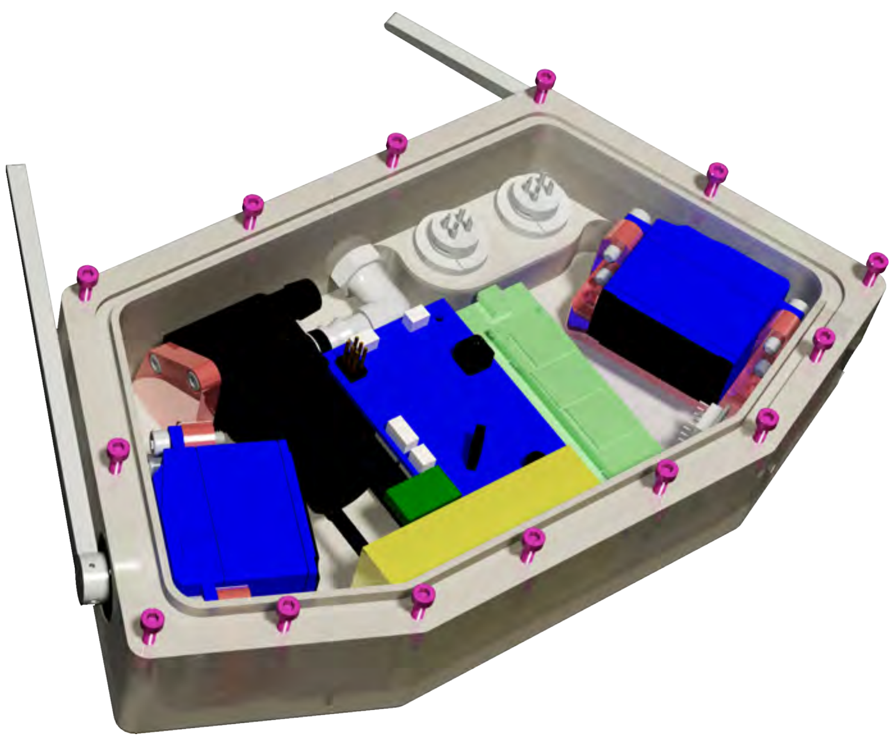


Figure 8.1: New BCU with Expanded Tray

Path planning is also possible with the more powerful processor. Using a combination of an IMU and GPS, it should be possible through trial and error to get the robot from point A to point B. However, by incorporating the work on the Adelopod, more advanced techniques could be developed with this platform. The water-proof shell and BCU would also make for some exciting–but challenging–problems in dynamic terrain and underwater.

The Gumstix board also provides support for video integration. Camera arrays positioned along the exterior of the robot could provide feedback for tumbling and create new ways for the operator to interact with the robot.

### 8.1.3 Backpack Mods

As described in Section 7.1.5, the backpack was designed to be versatile and modular. Completely new sensor suites can be developed for a wide range of applications. This could include everything from radiation detection, to measuring phosphorus and nitrogen levels. As mentioned in Section 6.4, if being used in shallow water monitoring, temperature, pressure, and salinity (conductivity) sensors are the most important.

The current slave board has open 5V, ground, and two PWM lines left over from the original sampler design. This can be used to communicate on a low level to sensors in the backpack while the Gumstix is free to handle higher level operations. These lines can also provide power for small actuators or a separate CPU device. If needed, the four pin backside port can easily be exchanged for up to eight pins. The only design limitations to the backpack is that it fits within the footprint of the Aquapod, has a minimal thickness, and is neutrally buoyant.

### 8.1.4 Fin Design

Unfortunately, there was not enough time to more thoroughly investigate the fin design of the Aquapod. However, through the availability of 3D printing, a variety of fin design iterations can be explored in rapid iteration. Either whole arms with new shapes or sheaths that slip over the current arms can be printed and tested. This will become a high priority if the work described in Section 8.2.3 is undertaken.

### 8.1.5 Docking Station

The bottom ports of the Japan design provided the opportunity to use a docking station with the robot. By simply setting the robot on the docking station and aligning the ports, the robot can be charged, programmed, and debugged. This would be a good accessory to have, both for development and use. A single docking location that both charges and downloads data would create a better user experience and provide easier use for field work.

## 8.2 Future Work

As mentioned, the platforms developed in this work have been mainly proof-of-concept prototypes. There is still much work to be done in order to make the Aquapod a successful tool. This second level of future work involves all the additions to the robot which are outside the scope of the project. As the Aquapod stands now, it provides minimal use in any real monitoring situation. This section deals with a recommended process to get from the prototype to a real-world design.

### 8.2.1 Durability

One factor that was not taken into serious consideration during the design of the Aquapod was the overall durability. In essence, the water-proofing qualities and external shell structure make the Aquapod significantly more rugged than the Adelopod. However, there are still many issues to take care of. The arms are long and unprotected, any back-driving or sudden force creates significant torques on the gears and motors. Adding a slip clutch would prevent the transmission of forces, but would throw off motor controllers.

A real-world design should also be able to handle a fair amount of abuse. Due to the low quantity of prototypes assembled, no specific failure testing was done on any of the iterations beyond typical use and demonstrations. Throughout their use, the Aquapods occasionally had mechanical failures due to wear or excessive forces. Most commonly, gear boxes were stripped. Future iterations should include a more robust drive-train and shock protection. An ideal goal would be to make the robots deployable from a low flying UAV.

### 8.2.2 Underwater Communication

Since most forms of wireless communication take place in the higher frequency range, their penetration depth in water in minimal. Underwater sonar transducers would be best option for communicating between Aquapods in lakes. Most sonar transducers are built for long-range ocean environments which prevents them from working in shallow water, and are also too big to fit on the Aquapod itself. A more feasible solution would be to adapt a commercial fish finding sonar which has a smaller range and works in shallow water. However, even these small transducers, at best, have a 5 foot minimum operating distance, which means that they would not work for streams. In these instances, the robot would have to return to the surface using the BCU and communicate through another method.

### 8.2.3 Full-Range Underwater Mobility

The pump described in Section 8.1.1, has such a high flow-rate that it may be possible to use the pump for jet propulsion. In combination with arm actuation, the robot could propel forward using the pump jet and control pitch and roll with the arms. A bladder by-pass valve would need to be installed in order to have a continuous flow of water. However, if possible, full-range underwater mobility could be programmed into the robot, offering a whole new level of applications and uses.

### 8.2.4 The Bigger Picture

The Center for Distributed Robotics concentrates on teams of heterogeneous robots to perform together to complete a task, and this robot is no different. Because of the small size and complex motion planning, the Aquapod is not fit for long range travel. Therefore, a larger robot is required to carried multiple Aquapods into an environment. Docking station robots, much like the one described in [31] could be used to transport and recharge Aquapods doing prolonged fieldwork. The robots can be distributed throughout a location and perform short-range operations and sensing.

Since space and weight is strictly limited inside the Aquapod, an option would be to have a team of different specialty Aquapods. Each member of the team would be equipped with a different backpack/sensors to accomplish a wide variety of sampling at a time. This could also be used as a method to overcome underwater communication issues.

### 8.2.5 Unique Applications

The first predicted application of the Aquapod, and the main push for including buoyancy control was for stream travel. Continuing from the above scenario, once an Aquapod has been deployed and is done sensing, it can find the closest stream or river and tumble towards it. Once at the stream, the robot can drive in, float, and use the current of the stream to transport to a new location. Once a desired location has been reached, the robot can swim to the side of the stream and tumble out. Or, use the BCU to sink to the bottom of stream and take other measurements until it is ready to be retrieved.

Another use of this setup would apply to tracking a point contamination source. If the source of a contaminate is known (e.g. hazardous chemicals leaking into a water supply) the Aquapod can be deployed in the stream at the source, float and tumble down the stream, periodically take measurements, and determine how far the contamination has spread.

The BCU can also be a means of low energy travel through tidal areas. By floating during tidal changes towards the desired location, and sinking during the reverse tide, the Aquapod can gradually make its way by floating with the current.

Another expressed interest has been the investigation of frozen water environments. By drilling a hole in the ice, an Aquapod can be deployed in the water and use the ice ceiling to do inverse tumbling. This idea has not been tested, but could provide valuable information on fish habits during the winter season which is currently difficult to obtain.

# Chapter 9

# Conclusion

Chapter 8 went through some of the work that still needs to be done on the Aquapod project, however, this author's contributions have ended. This chapter recaps some of the highlights of the project, draws a comparison of the different prototypes, and also provides of list of general design considerations for future tumbling amphibious robots.

## 9.1 Project Overview

The starting point for this project was strongly grounded on the Adelopod platform. The original purpose was to take the Adelopod design and make the robot more resilient to harsh environments. By doing so, the hope was to learn more about how tumbling locomotion could be used in dynamic and rough terrain where a similar-sized wheeled robot would fail. If successful, the design would find many useful and unique applications in environmental monitoring and surveillance. A buoyancy control unit and water-proofing was also added to the list of functions in order to expand the locomotion abilities.

The initial stop in the development was the design of the first, proof-of-concept, remote-controlled prototype. Built on a small budget, this prototype proved the BCU functionality and the ability to water-proof a tumbling robot. However, a number of design flaws and lack of any autonomous capabilities or sensors required the development of a more advanced model.

Using the RC prototype, a substantial NSF-MRI Grant was awarded to the project. The goal of this second design was to add functionality that would enable the robot to gather water samples, semi-autonomously, from a deployed location. It should be restated, that the design of this prototype was not the work of the author, but rather a team of graduate students from a variety of backgrounds. During this phase, the electronic components were created and the core firmware was started.

The final prototype was built with funds from a second NSF grant and improved many of the design flaws from the previous iteration. However, due to time and budget constraints, only minor improvements were made to the v1.1 design. The new master board and module backpack system allowed for continued expansion of the robot's applications.

## 9.2 Design Comparisons

Section 9.1 described the three main prototypes built for this project. However, it is important to look back and evaluate these designs based on the original objectives described in Sections 3.3.1 and 3.3.2. Figure 9.1 shows the progression of the project in the various prototypes. Table 9.1 includes all the iterations and their features for a direct comparison of the designs.

| **Feature** | **Adelopod** | **RC** | **v1.1** | **Japan** |
|---|---|---|---|---|
| Volume | $412cm^3$ | $443cm^3$ | $1350cm^3$ | $1178cm^3$ |
| Weight | 400g | 435g | 1310g | 1100g |
| Max Depth | n/a | 2m | 10m | 10+m |
| Bladder Vol. | n/a | 60ml | 80ml | 110ml |
| Trans. Time | n/a | 2.5min | 1.2min | 1.2min |
| Radio Control | No | Yes | Yes | Yes |
| Wireless Com. | 802.11b/BT | None | None | 802.11g/BT |
| GPS | No | No | Yes | Yes |
| Expandability | No | No | Limited | Yes |

Table 9.1: Design Comparison

For completeness, the objectives in Sections 3.3.1 and 3.3.2 will be covered one-by-one in the following list, with an explanation on how well the final prototype (Japan) fits the design requirements.

Figure 9.1: Rendering of Adelopod, RC Aquapod, v1.1 Aquapod, and Japan Aquapod

- **Small size (under a 50% increase from the original Adelopod):** The final volume of the robot came to $1178cm^3$, which correspondes to a 285% increase from the Adelopod. However, this was partially due to time limitations, where a the shell from the v1.1 design could only be minorily modified without the need for additional pressure testing. Future iterations could keep the same functionality, but be reduced in size.
- **Seamless land to water transitions:** The tumbling locomotion works both on land and in shallow water. In water where the robot can no longer reach the floor, the swimming mode can be activated in the program to swim on the surface instead of tumbling.
- **Water Surface Movement:** Though not the most ideal, surface movement was achieved with paddles.
- **Internal buoyancy control:** The BCU, though slow, worked as anticipated throughout all the design iterations.
- **Wireless communication and on-board processing:** With the addition of the Gumstix Overo Air board, the Japan Aquapod was equipped with three forms of wireless communication and a complete on-board processor.
- **Water proof up to 25m:** The limiting factor on the v1.1 design for depth was the arm servos and USB ports. The Japan model removed these, and though a maximum pressure test was not done, it is assumed that deeper depths can be reached with this design. The exact number is unknown.
- **On-board sensors:** The only on-board sensors included in the Japan robot were the internal and external pressure sensors. However, with the backpack module, a wide variety of sensors could be added to the design.
- **Power Sourcing:** This feature was not explored during the design process and will most likely need a completely new design.
- **Manipulators:** This feature was also not explored, however, just like additional sensors, the backpack module could include some form of manipulator.

## 9.3 Tumbling Amphibious Robots

Throughout the design process many lessons were learned that could be applied not only to future iterations of the Aquapod, but other tumbling and/or amphibious robots in general.

- Tumbling is not a scalable form of mobility. Larger bodies will exert higher impact forces on the internal components during tumbles. However, a detailed analysis has yet to be done to determine the exact size limitations.

- Reducing the number of openings and ports on underwater robots should be a priority over user convince or internal layout.
- Passive buoyancy control via ingesting water from the surrounding environment is a viable means of controlling vertical movement in a water column.
- When using passive buoyancy control, the design weight of the robot should be as close to neutrally buoyant as possible, without going overweight. This places strict design constraints on the available weight verses size, but will produce a more efficient design.

As mentioned in Section 2.3, there is a strong case for tumbling robots. However, throughout the development of the Aquapod, the motion planning algorithms have been assumed. At the time of this thesis, controlling the robot from point A to point B has gone largely unanswered. Before the benefits of tumbling can be utilized, a reliable solution to the complicated path planing problem over real-world terrains must be solved. This task was outside the scope of this project, but is essential to the future success of all tumbling robots.

After closely working on tumbling robots for three years, this author has seen a number of advantages to tumbling locomotion. Like any method of ambulation, tumbling has its shortcomings in the form of scaling and controllability. But for small-scale, short-range robots it is a promising and viable form of mobility, with hybrid systems like the Treaded Adelopod most likely being the future of tumbling robots. It is hoped that the detailed development description of this thesis has provided useful information for future ventures, and can prevent a repeat of the lessons learned throughout this project.

## 9.4 Final Remarks

It is the opinion of this author that, although the prototypes developed herein are not yet ready for commercial use, within a few more years of development, the Aquapod could be ready to collect real-world data; benefiting a wide variety of environmental research. The novelty of the buoyancy control unit can provide unique and innovative application scenarios. Even with the unmet expectations of the Houston demonstration, it is believed that by playing on the robot's strengths in a particular application, a very functional and productive product can be achieved.

# References


[1] Sandeep Dhull, Dario Canelon, Apostolos Kottas, Justin Dancs, Andrew Carlson, and Nikolaos Papanikolopoulos. Aquapod: A small amphibious robot with sampling capabilities. *IEEE International Conference on Intelligent Robots and Systems*, Submitted for Publication.

[2] Brett Hemes, Duc Fehr, and Nikolaos Papanikolopoulos. Motion primitives for a tumbling robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008.

[3] M.A. Batalin, M. Rahimi, Y. Yu, et al. Call and response: experiments in sampling the environment. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 25–38. ACM New York, NY, USA, 2004.

[4] M. Rahimi, R. Pon, D. Estrin, W.J. Kaiser, et al. Adaptive sampling for environmental robotics. *UC Los Angeles: Center for Embedded Network Sensing. Retrieved from: http://www. escholarship. org/uc/item/77d882tk*, 2003.

[5] A. Dhariwal, G. Sukhatme, and A. Requicha. Bacterium-inspired robots for environmental monitoring. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1436–1443. Citeseer, 2004.

[6] M. Dunbabin, J. Roberts, K. Usher, and P. Corke. A new robot for environmental monitoring on the Great Barrier Reef. In *Proceedings of the 2004 Australasian Conference on Robotics*, 2004.

[7] M. Listak, G. Martin, D. Pugal, A. Aabloo, and M. Kruusmaa. Design of a semiautonomous biomimetic underwater vehicle for environmental monitoring. In *Computational Intelligence in Robotics and Automation, 2005. CIRA 2005. Proceedings. 2005 IEEE International Symposium on*, pages 9–14. IEEE, 2005.

[8] X. Tan, D. Kim, N. Usher, et al. An autonomous robotic fish for mobile sensing. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5424–5429, 2006.

[9] S. Floyd, T. Keegan, and M. Sitti. A novel water running robot inspired by basilisk lizards. In *Proc. of Intelligent Robots and Systems*, pages 5340–36, 2006.

[10] R.D. Quinn, DA Kingsley, JT Offi, and R.E. Ritzmann. Improved mobility through abstracted biological principles. In *IEEE Int. Conf. On Intelligent Robots and Systems (IROS 02)*, 2002.

[11] A.S. Boxerbaum, P. Werk, R.D. Quinn, and R. Vaidyanathan. Design of an autonomous amphibious robot for surf zone operation: Part I mechanical design for multi-mode mobility. In *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1459–1464, 2005.

[12] R. Harkins, J. Ward, R. Vaidyanathan, A.S. Boxerbaum, and R.D. Quinn. Design of an autonomous amphibious robot for surf zone operations: part II-hardware, control implementation and simulation. In *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1465–1470, 2005.

[13] V. Kaznov and M. Seeman. Outdoor navigation with a spherical amphibious robot. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 5113–5118. IEEE, 2010.

[14] Chris Prahacs, Aaron Saunders, Matthew K. Smith, et al. Towards legged amphibious mobile robotics. *The Inaugural Canadian Design Engineering Network (CDEN/RCCI) Design Conference*, 2004.

[15] Uluc Saranli, Martin Buehler, and Daniel E. Koditschek. Rhex: A simple and highly mobile hexapod robot. *International Journal of Robotics Research*, 20:616–631, 2001.

[16] G. Dudek, M. Jenkin, et al. A visually guided swimming robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1749–1754. Citeseer, 2005.

[17] G. Dudek, P. Giguere, C. Prahacs, et al. Aqua: An amphibious autonomous robot. *AQUA*, 10:43, 2007.

[18] A. Crespi and A.J. Ijspeert. Amphibot ii: An amphibious snake robot that crawls and swims using a central pattern generator. In *Proceedings of the 9th international conference on climbing and walking robots (CLAWAR 2006)*, volume 11, pages 19–27. Citeseer, 2006.

[19] H. Yamada, S. Chigisaki, M. Mori, K. Takita, K. Ogami, and S. Hirose. Development of amphibious snake-like robot acm-r5. *Proc. ISR2005*, 2005.

[20] P.E. Clark, M. L. Rilee, S.A. Curtis, et al. Bees for ants: Space mission applications for the autonomous nanotechnology swarm. *AIAA 1st Intelligent Systems Technical Conference*, 2004.

[21] M. Abrahantes, A. Silver, and L. Wendt. Gait design and modeling of a 12-tetrahedron walker robot. In *System Theory, 2007. SSST '07. Thirty-Ninth Southeastern Symposium on*, pages 21–25, March 2007.

[22] Brett Hemes, N. Papanikolopoulos, and B. O'Brien. The adelopod tumbling robot. *IEEE International Conference on Robotics and Automation*, pages 1583–1584, 2009.

[23] J.W. Suh, S.B. Homans, and M. Yim. Telecubes: Mechanical design of a module for self-reconfigurable robotics. In *Proceedings- IEEE International Conference on Robotics and Automation*, volume 4, pages 4095–4101. Citeseer, 2002.

[24] P. White, V. Zykov, J. Bongard, and H. Lipson. Three dimensional stochastic reconfiguration of modular robots. In *Proceedings of robotics science and systems*, pages 161–168. Citeseer, 2005.

[25] K. Gilpin, K. Kotay, D. Rus, and I. Vasilescu. Miche: modular shape formation by self-disassembly. *International Journal of Robotics Research*, 27(3-4):345–372, 2008.

[26] W.J. Gould. From Swallow floats to Argo: the development of neutrally buoyant floats. *Deep-Sea Research Part II*, 52(3-4):529–543, 2005.

[27] D. McFarland, I. Gilhespy, and E. Honary. DIVEBOT: A diving robot with a whale-like buoyancy mechanism. *Robotica*, 21(04):385–398, 2003.

[28] V. Bokser, C. Oberg, G. Sukhatme, and A. Requicha. A small submarine robot for experiments in underwater sensor networks. In *Symposium on Intelligent Autonomous Vehicles*. Citeseer, 2004.

[29] C. Detweiler, S. Sosnowski, I. Vasilescu, and D. Rus. Saving energy with buoyancy and balance control for underwater robots with dynamic payloads. In *Experimental Robotics*, pages 429–438. Springer Berlin/Heidelberg, 2009.

[30] maxonmotors.com. Key information, May 2009. https://downloads.maxonmotor.com/.

[31] A. Drenner and N. Papanikolopoulos. Docking station relocation for maximizing longevity of distributed robotic teams. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2436–2441, 2006.

# Appendix A

# Acronyms

An effort was made to introduce any acronyms in the written text of this thesis. The list below is included for consistency and to clear any confusion the reader may encounter, especially for acronyms that were created specifically for this paper, or outside of their common usage.

| | |
|---|---|
| ADC | Analog-to-Digital Converter |
| BCU | Buoyancy Control Unit |
| CDR | Center for Distributed Robotics |
| COTS | Commercial Off-the-Shelf |
| ESC | Electronic Speed Control |
| FEA | Finite Element Analysis |
| GPIO | General Purpose Input-Output |
| GUI | Graphical User Interface |
| IMU | Inertial Measurement Unit |
| IP68 | Ingress Protection 68 |
| MRI | Massive Research Instrument |
| NSF | National Science Foundation |
| PCB | Printed Circuit Board |
| PID | Proportional-Integral-Derivative |
| PPM | Pulse Position Modulation |
| PT | Package Tracker |
| PTC | Push-to-Connect |
| PWM | Pulse Width Modulation |
| RC | Radio Control |
| RP | Rapid Prototype |

# Appendix B

# Misc. Calculations

## B.1 Shell Material Selection

| Material | Density [g/cm^3] | Water Absorption At Saturation | Yield [MPa] | Elasticity [Gpa] | McMaster Price 12x12x1 | Shell Weight [g] | Absorbed Weight [g] | Total Weight [g] | Cost/ Weight | Cost/ Strength | Strength/ Weight |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ABS | 1.050 | 0.80% | 41.3 | 2.04 | $44.93 | 99.4 | 0.8 | 100.1 | 0.45 | 1.09 | 0.41 |
| Nylon 6/12 | 1.080 | 3.50% | 55.6 | 2.36 | $181.93 | 102.2 | 3.3 | 105.5 | 1.72 | 3.27 | 0.53 |
| Nylon 6/6 | 1.170 | 7.73% | 84.0 | 3.66 | $82.65 | 110.7 | 7.3 | 118.1 | 0.70 | 0.98 | 0.71 |
| Acetal Copolymer | 1.400 | 0.88% | 61.3 | 2.65 | $81.21 | 132.5 | 0.8 | 133.3 | 0.61 | 1.32 | 0.46 |
| Delrin 570 | 1.560 | 0.80% | 53.0 | 2.05 | $199.74 | 147.7 | 0.8 | 148.4 | 1.35 | 3.77 | 0.36 |
| Delrin | 1.410 | 0.90% | 75.8 | 3.10 | $91.47 | 133.5 | 0.9 | 134.3 | 0.68 | 1.21 | 0.56 |
| Hydlar Z | 1.160 | 6.30% | 110.0 | 0.62 | $345.57 | 109.8 | 6.0 | 115.8 | 2.99 | 3.14 | 0.95 |
| Polycarbonate | 1.200 | 0.37% | 64.1 | 2.39 | $137.00 | 113.6 | 0.4 | 113.9 | 1.20 | 2.14 | 0.56 |
| Polysulfone | 1.360 | 1.27% | 85.3 | 5.31 | $230.69 | 128.7 | 1.2 | 129.9 | 1.78 | 2.70 | 0.66 |
| Polypropylene | 0.912 | 0.07% | 32.7 | 1.42 | $103.97 | 86.3 | 0.1 | 86.4 | 1.20 | 3.18 | 0.38 |
| LDPE | 0.923 | 0.00% | 10.0 | 0.20 | $19.96 | 87.4 | 0.0 | 87.4 | 0.23 | 2.00 | 0.11 |
| HDPE | 0.958 | 0.04% | 23.9 | 0.88 | $34.25 | 90.7 | 0.0 | 90.7 | 0.38 | 1.43 | 0.26 |

## B.2 Arm Motor Selection

**Arm Motor Torque Selection**

| | | |
|---|---|---|
| Minimum Voltage | 7.00 | V |
| Maximum Voltage | 8.40 | V |
| Ambient Temp | 25.00 | °C |
| Assumed percentage of Rth2 | 25% | |
| Design Torque | **248.00** | mNm |

| **Motor Characteristics** | **MicroMo** | | | | **Maxon** | | |
|---|---|---|---|---|---|---|---|
| | **1224 006 S** | **1224 012 S** | **1024 006 S** | **1319 006 SR** | **256101** | **118455** | **265375** |
| **Terminal Resistance, ohms** | 6.60 | 26.80 | 10.80 | 8.26 | 9.09 | 11.30 | 16.00 |
| **Speed constant, rpm/V** | 2318.00 | 1172.00 | 2231.00 | 1110.00 | 2100.00 | 1880.00 | 2190.00 |
| **Torque Constant, mNm/A** | 4.12 | 8.14 | 4.28 | 8.57 | 1.13 | 5.08 | 4.36 |
| Speed/Torque Slope (rpm/mNm) | 3713.30 | 3858.67 | 5629.63 | 1069.85 | 16892.92 | 4181.89 | 8036.70 |
| **No-load current (A)** | 0.013 | 0.006 | 0.008 | 0.015 | 0.011 | 13.800 | 0.009 |
| **Rth1, windings/case, K/W** | 22.00 | 22.00 | 14.00 | 8.00 | 9.00 | 14.00 | 15.00 |
| **Rth2, case/ambient, K/W** | 45.00 | 45.00 | 41.00 | 35.00 | 37.50 | 46.00 | 44.50 |
| **Max winding Temp (°C)** | 85.00 | 85.00 | 85.00 | 85.00 | 85.00 | 85.00 | 85.00 |
| **Slope of n-M curve** | 3713 | 3862 | 5630.00 | 4510.00 | 4180.00 | 4030.00 | 8060.00 |
| **No-load speed (rpm)** | 12700 | 13100 | 13200.00 | 12800.00 | 12400.00 | 6490.00 | 12800.00 |
| **Length (mm)** | 24.20 | 24.20 | 23.70 | 19.20 | 24.60 | 19.20 | 21.20 |
| **Weight (g)** | 13.00 | 13.00 | 8.80 | 12.00 | 10.00 | 15.00 | 12.00 |
| Power at stall at max V (W) | 10.69 | 2.63 | 6.53 | 8.54 | 7.76 | 6.24 | 4.41 |
| Temp at stall (°C, reduced Rth2) | 380.47 | 112.54 | 183.43 | 168.08 | 167.63 | 184.23 | 140.21 |
| | | | | | | | |
| Stall Torque at min V (mNm): | 4.37 | 2.13 | 2.77 | 7.26 | 0.87 | 3.15 | 1.91 |
| Stall Torque at max V (mNm): | 5.24 | 2.55 | 3.33 | 8.72 | 1.04 | 3.78 | 2.29 |
| Max thermally-allowable torque (mNm, reduced Rth2) | 2.15 | 2.11 | 2.05 | 5.64 | 0.68 | 2.32 | 1.65 |
| | | | | | | | |
| **Gearbox Number** | 12/4 64:1 | 12/4 64:1 | 12/4 256:1 | 14/1 159:1 | 218418 | 110316 | 110316 |
| **Gearbox Gearing** | 64.00 | 64.00 | 256.00 | 159.00 | 256.00 | 275.00 | 275.00 |
| **Gearbox Efficiency** | 70% | 70% | 60% | 60% | 65% | &(# | &(# |
| **Gearbox Length (mm)** | 24.3 | 24.3 | 28.9 | 33.3 | 20.4 | 27.6 | 27.6 |
| **Gearbox Weight (g)** | 18 | 18 | 21 | 27 | 8.2 | 20 | 20 |
| Output Gearing | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 |
| Output Gearing Efficiency | 90% | 90% | 90% | 90% | 90% | 90% | 90% |
| Total Gearing | 128.00 | 128.00 | 512.00 | 318.00 | 512.00 | 550.00 | 550.00 |
| Total Gearing Efficiency | &)# | &)# | 54% | 54% | 59% | 62% | 62% |
| | | | | | | | |
| Output Stall Torque, min V, mNm: | 352 | 171 | 767 | 1247 | 261 | 1075 | 652 |
| Output Stall Torque, max V, mNm: | 423 | 206 | 920 | 1497 | )') | 1290 | 782 |
| Output Stall Torque, thermal limit, mNm: | 174 | 170 | 566 | (&( | 203 | 792 | 564 |
| | | | | | | | |
| Motor Seen Required Torque | 3.075 | 3.075 | 0.897 | 1.444 | 0.828 | 0.726 | 0.726 |
| Motor Speed at Peak Torque (rpm) | 1281 | 1223 | 8150 | 6287 | 8939 | 3564 | 6948 |
| Output Speed at Peak Torque (rpm) | 10.0 | 9.6 | 15.9 | 19.8 | 17.5 | 6.5 | 12.6 |
| | | | | | | | |
| Average Stall Torque | 405 | 197 | 882 | 1434 | 300 | 1236 | 749 |
| Total Length | 48.5 | 48.5 | 52.6 | 52.5 | 45.0 | 46.8 | 48.8 |
| Total Weight | 31.0 | 31.0 | 29.8 | 39.0 | 18.2 | 35.0 | 32.0 |

## B.3 Pump Motor Selection

**Pump Motor Torque Selection**

| | | |
|---|---|---|
| Minimum Voltage | 7.00 | V |
| Maximum Voltage | 8.40 | V |
| Ambient Temp | 25.00 | °C |
| Design Torque | 250.00 | mNm |

| **Motor Characteristics** | **MicroMo** | | | | **Maxon** | | |
|---|---|---|---|---|---|---|---|
| | **1331 006 SR** | **1524 009 SR** | **1331 012 SR** | **2224 006 SR** | **110084** | **110063** | **269571** |
| **Terminal Resistance, ohms** | 2.83 | 10.40 | 13.70 | 1.94 | 5.77 | 8.32 | 4.61 |
| **Speed constant, rpm/V** | 1790.00 | 1140.00 | 835.00 | 1380.00 | 1320.00 | 1720.00 | 1270.00 |
| **Torque Constant, mNm/A** | 5.35 | 8.37 | 11.40 | 6.92 | 7.23 | 5.58 | 7.49 |
| Speed/Torque Slope (rpm/mNm) | 946.86 | 1416.49 | 1003.46 | 386.88 | 1053.44 | 2564.59 | 781.67 |
| **No-load current (A)** | 0.022 | 0.014 | 0.011 | 0.029 | 0.019 | 0.047 | 0.037 |
| **Rth1, windings/case, K/W** | 6.00 | 4.50 | 6.00 | 5.00 | 10.50 | 5.50 | 12.00 |
| **Rth2, case/ambient, K/W** | 25.00 | 31.00 | 25.00 | 20.00 | 21.30 | 29.80 | 35.00 |
| **Max winding Temp (°C)** | 85.00 | 85.00 | 85.00 | 85.00 | 85.00 | 85.00 | 85.00 |
| **Slope of n-M curve** | 946 | 1420 | 1000.00 | 387.00 | 1050.00 | 2560.00 | 785.00 |
| **No-load speed (rpm)** | 10600 | 10100 | 9900.00 | 8200.00 | 7780.00 | 9620.00 | 11300.00 |
| **Length (mm)** | 31.20 | 23.80 | 31.20 | 24.20 | 28.90 | 25.40 | 25.40 |
| **Weight (g)** | 19.00 | 21.00 | 19.00 | 46.00 | 33.00 | 22.00 | 26.00 |
| | | | | | | | |
| Stall Torque at min V (mNm): | 13.23 | 5.63 | 5.82 | 24.97 | 8.77 | 4.69 | 11.37 |
| Stall Torque at max V (mNm): | 15.88 | 6.76 | 6.99 | 29.96 | 10.53 | 5.63 | 13.65 |
| Max thermally-allowable torque (mNm, reduced Rth2) | 7.04 | 5.74 | 6.82 | 12.17 | 5.86 | 4.16 | 5.93 |
| | | | | | | | |
| **Gearbox Number** | 13A | 15A | 13A | 22E | **218418** | **327790** | **112864** |
| **Gearbox Gearing** | 64.00 | 102.00 | 50.00 | 28.00 | 104.00 | 141.00 | 84.00 |
| **Gearbox Efficiency** | 72% | 68% | 72% | 77% | 73% | 62% | 60% |
| **Gearbox Length (mm)** | 22 | 22 | 22 | 32.8 | 23.1 | 19.3 | 25.5 |
| **Gearbox Weight (g)** | 5 | 5 | 5 | !" | # | 17.2 | 8.4 |
| Output Gearing | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Output Gearing Efficiency | 100% | 100% | 100% | 100% | 100% | 100% | 100% |
| Total Gearing | 64.00 | 102.00 | 50.00 | 28.00 | 104.00 | 141.00 | 84.00 |
| Total Gearing Efficiency | 72% | 68% | 72% | 77% | 73% | 62% | 60% |
| | | | | | | | |
| Output Stall Torque, min V, mNm: | 609.79 | 390.75 | 209.69 | 538.33 | 665.91 | 410.41 | 573.20 |
| Output Stall Torque, max V, mNm: | 731.74 | 468.90 | 251.63 | 646.00 | 799.09 | 492.49 | 687.85 |
| Output Stall Torque, thermal limit, mNm: | 324.33 | 398.41 | 245.39 | 262.38 | 444.95 | 364.02 | 298.97 |
| | | | | | | | |
| Motor Seen Required Torque | 5.425 | 3.604 | 6.944 | 11.596 | 3.293 | 2.860 | 4.960 |
| Motor Speed at Peak Torque (rpm) | 5468 | 4982 | 2956 | 3713 | 4322 | 2299 | 7406 |
| Output Speed at Peak Torque (rpm) | 85.4 | 48.8 | 59.1 | 132.6 | 41.6 | 16.3 | 88.2 |
| | | | | | | | |
| Average Stall Torque | 701 | 449 | 241 | #!" | 766 | 472 | 659 |
| Total Length | 53.2 | 45.8 | 53.2 | 57.0 | 52.0 | 44.7 | 50.9 |
| Total Weight | 24.0 | 26.0 | 24.0 | 65.0 | 69.0 | 39.2 | 34.4 |