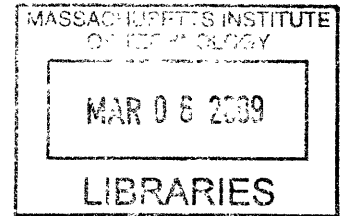


XAUV: Modular High Maneuverability Autonomous Underwater Vehicle

by
Daniel G. Walker

S.B. Mechanical Engineering
Massachusetts Institute of Technology, 2005



SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENGINEERING IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN MECHANICAL ENGINEERING
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

FEBRUARY 2009

©2009 Massachusetts Institute of Technology. All rights reserved.

The author hereby grants to MIT permission to reproduce and distribute publicly paper
and electronic copies of this thesis document in whole or in part in any medium now
known or hereafter created.

Signature of Author: _____

Department of Mechanical Engineering
December 16, 2008

Certified by: _____

Franz Hover
Assistant Professor of Mechanical and Ocean Engineering
Thesis Supervisor

Certified by: _____

John Leonard
Professor of Mechanical and Ocean Engineering
Thesis Supervisor

Accepted by: _____

David E. Hardt
Graduate Officer, Department of Mechanical Engineering

ARCHIVES

XAUV: Modular High Maneuverability Autonomous Underwater Vehicle

by

Daniel G. Walker

Submitted to the Department of Mechanical Engineering
on December 16, 2008 in Partial Fulfillment of the
Requirements for the Degree of Master of Science in
Mechanical Engineering

ABSTRACT

The design and construction of a modular test bed autonomous underwater vehicle (AUV) is analyzed. Although a relatively common stacked-hull design is used, the state of the art is advanced through an aggressive power plant, with capability to support azimuthing thrusters and a 2DOF front sensor assembly. Through an application of lean principles to developmental hardware, the notion of a delayed differentiation is isolated as a key to minimizing rework and creating essentially transparent electronic hardware. Additionally, the use of bus-modular structural and electronic interconnects facilitates reconfiguration of the vehicle across a large range of components, allowing the rapid development of new sensors, control algorithms, and mechanical hardware. Initial wet tests confirm basic data acquisition capabilities and allowed sensor fusion of scanning sonar returns at the beam level with data from an IMU for an orientation-corrected sonar mosaic.

Thesis Supervisor: Franz Hover

Title: Assistant Professor of Mechanical engineering

Thesis Supervisor: John Leonard

Title: Professor of Mechanical and Ocean Engineering

Acknowledgements

This would not have possible without the support of many people. First and foremost, I would like to thank the bosses, John and Franz. Only through their guidance and Herculean patience have I come to understand the true craftsmanship involved in juggling both teaching and research.

Thanks also to my family for their support throughout this process, especially Kate and Rosie.

Thanks to all the UROPs throughout the years: Matt, Joe, and Kenton, you guys were the reason we came up with this project in the first place. Best of luck to you. Devin, Megan, Brooks, Corinna, you all did great mechanical work; there are good things ahead of you. Xiawa, and Rachel, I hope you keep at the electronics, it'll come in handy one day. It's been a pleasure to work with all of you.

Thanks to my classmates, both from the good old days, and the newer days. The long hours will probably be worth something at some point. Thanks also to my labmates, I can't think of a better beginning to a successful Hovergroup.

Thanks to MITOC for all the cold icy nights.

Research was sponsored by the Office of Naval Research Grant N00014-06-10043, monitored by Dr. T.F. Swean.

Table of Contents

Acknowledgements.....	5
List of Tables	8
List of Figures.....	9

Chapter 1: Introduction

1.1 Background.....	12
1.2 Requirements Identification.....	13
1.3 Relevant work	14
1.4 Outline.....	15

Chapter 2: System Level Design

2.1 Lean concepts.....	17
2.1.1 Model considerations	20
2.1.2 Intermediate Hardware Details	21
2.1.3 Delaying differentiation.....	22
2.2 Parameter Identification.....	25
2.3 Overall Layout and Mechanical Design	27
2.3.1 Hydrodynamic Analysis.....	29
2.3.2 Structural Backbone.....	33
2.3.3 Modular Capabilities and Concepts	34
2.4 Overall Electrical design.....	38
2.4.1 Data transmission.....	38
2.4.2 Power System.....	42
2.5 Summary	43

Chapter 3: Specifics and Implementation

3.1 Mechanical Details.....	46
3.1.1 Flotation	46
3.1.2 Thruster mounts	50
3.1.3 Pressure Housings.....	52
3.2 Electronic Details.....	55
3.2.1 Computer Stack.....	56
3.2.2 Interconnects	57
3.2.3 Isolation Board.....	58
3.2.4 Analog Signal Board.....	61
3.2.5 Auxiliary Power Switching Board.....	63

3.2.6 Power Bus Brick	64
3.2.7. External Control Input	65
3.2.8 Battery System	66
3.3 Sensor Payload.....	73
3.3.1 Blazed Array	73
3.3.2 Camera	74
3.3.4 IMU	74
3.3.5 Depth Gauge	75
3.3.6 Scanning Sonar	75
3.3.7 Hokoyu Laser Scanner.....	77
3.3.8 GPS	78
3.3.9 DVL	78
3.4 Summary	79
Chapter 4: Control System Design	
4.1 High Level Control System	80
4.1.1 Linearized Kalman Filter	81
4.2 Low-level Control Structure	85
4.2.1 Decoupled Dynamics	85
4.1.1. System Model	86
4.1.2 Closed loop depth and pitch.....	88
4.1.3 Closed-loop heading and speed	91
4.1.4. Azimuthing Thruster Control.....	93
4.2 Software	96
4.3.1 VB Interface.....	97
4.3.2 Slew Limit.....	99
Chapter 5: Experimental Results	
5.1 Leak Testing.....	102
5.2 Data Capture	102
5.3 Sonar Mosaic/IMU correction	105
Chapter 6: Conclusion	
6.1 Summary	108
6.2 Future Work.....	109
Bibliography	111

List of Tables

Table 2-1: Comparison of Intermediate Hardware Platforms	22
Table 2-2: Algorithm module differentiation points	23
Table 2-3: Mission Strategy - Functional Requirement/ Design Parameter Mapping.	26
Table 2-4: Reorganized parameter mapping	27
Table 2-5: Strategy 2 - Motion Iteration	28
Table 2-6: Expanded Iterations	29
Table 2-7: Architecture tradeoffs and examples	33
Table 2-8: Widely-used data interfaces (39)	40
Table 2-9: XAUV Power Requirements	42
Table 2-10: Cell Chemistry	43
Table 3-1: Potential flotation trade-offs	47
Table 3-2: Computer Specifications	56
Table 3-3: Control System Peripherals	57
Table 3-4: Devices, power, and signal requirements for connectors	58
Table 3-5: Battery Characteristics	66
Table 3-6: Comparison of differing charge control electronics and connectors16.....	70
Table 3-7: P900 blazed array properties (3)	73
Table 3-8: IMU characteristics (54)	75
Table 4-1: Cubic velocity term linearization	86

List of Figures

Figure 1-1: Bluefin/MIT Hovering AUV	13
Figure 1-2: Testing tank at the MIT Marine Robotics Lab.....	14
Figure 2-1: Successful path from simulation to hardware leverages low cost, fast cycle time of simulation to reduce chance of hardware system failure.	17
Figure 2-2: Typical Development Map with large barrier between simulation and hardware	18
Figure 2-3: Development map using a common MATLAB framework and small, intermediate hardware	20
Figure 2-4: Minimization of rework depends on transporting functional blocks (whether code or code-bearing hardware) between development and target environments.	24
Figure 2-5: The X-29's unstable design requires a complex control system, but allows highly acrobatic maneuvers. (29)	30
Figure 2-6: Righting moments and centers of buoyancy and gravity for one and two-tube designs	31
Figure 2-7: Intersection of drag and thrust force curves determines the maximum vehicle speed. The above graph shows forward velocity.	32
Figure 2-8: front sensor platform CAD model and static mockup	34
Figure 2-9: XAUV concept with DVL in bottom/top scan mode.	35
Figure 2-10: Odyssey IV at field operations in Georges Bank. The two main thrusters are controlled by a single rotating thruster unit, incorporating a brushless kit motor and resolver position sensor. This configuration allows high power vertical and forward motion with the same thrusters.	36
Figure 2-11: 1DOF thrusters concept	37
Figure 2-12: a) Experimental ROV with 2DOF thusters b) XAUV concept with 2DOF thrusters	38
Figure 2-13: Module Relations	39
Figure 3-1: Foam pressure testing chamber. (a) main pressure chamber, filled with water and foam chunk. (b) valve to maintain pressure, (c) barb fitting for pump. While under pressure, the chamber was housed in a secondary pipe for safety, in the event of catastrophic failure of the chamber.	47
Figure 3-2: Foam casting process. (a) empty mold; (b) as cast; (c) demolded; (d) cored and prepped for painting	49
Figure 3-3: Tunnel thruster mount assembly	50
Figure 3-4: Finite element deflection analysis on fixed thruster mount. (exaggerated in rendering) The maximum displacement is 1.4mm, resulting in an angle of about 1° toward the vehicle midplane.	52
Figure 3-5: Pressure tolerant housings. Left: atmospheric housing. Right: oil pressure compensated.	53
Figure 3-6: Computer housing (top) and battery housing (bottom). Both use simple double o-ring aluminum plugs inside polycarbonate tubes. This allows the housing to be lengthened by replacing the tube only.	54
Figure 3-7: Main Junction Box	55

Figure 3-8: Main electronics Stack: (a) hard drive (b) CPU card (c) Data acquisition card (d) power conversion and switching cards (e) sensor interface and isolation cards (f) LCD display (g) power bus brick.	56
Figure 3-9: Analog Isolator Topology. Two identical photodiodes allow linearization with one diode and electrical isolation with the other.	59
Figure 3-10: Isolation board circuit	60
Figure 3-11: Isolation Board Hardware	60
Figure 3-12: Standard bridge input instrumentation amplifier. The bridge is essentially a pair of voltage dividers. Each is fed to a separate non-inverting amplifier, with a single balancing gain resistor in between. The outputs of these feed into a differential amplifier. A typical implementation allows the precise measurement of resistive sensor while maintaining low input impedance.	61
Figure 3-13: Op Amps. (a) Physically protected buffer (b) actively protected buffer	62
Figure 3-14: Analog signal board hardware	63
Figure 3-15: Auxiliary power switch hardware	64
Figure 3-16: Power bus brick. The power bus brick incorporates (a) filtering capacitors, (b) mainline fusing, (c) thruster breakout, fusing, and MOVs, (d) main power relay, (e) fused main bus, (f) switched main bus, (g) ground bus	65
Figure 3-17: XAUUV Power System	67
Figure 3-18: Typical charging profile, measured and normalized from a 16S2P battery configuration. Charging current is total current, thus per string current is 1.5A in this case, half the nominal value.	68
Figure 3-19: Individual cell charger Topologies: a) hi-lo switch b) isolated power supply c) bypass/burn off	69
Figure 3-20: rendering of XAUUV shiphull inspection, showing field of view of blazed array.	73
Figure 3-21: Blazed array image from test tank - tank walls(16ft-26ft), cinderblock (~12ft), and gantry head (~16ft) visible	74
Figure 3-22: Imagenex 852 Scanning Sonar	76
Figure 3-23: Inline adapter to allow sonar to interface with standard USB/12V connector. ...	76
Figure 3-24: Scanning Sonar Image, showing sides of MIT Towing Tank facility.	77
Figure 3-25: (left) Hokuyo Laser Scanner. Image courtesy of M. Kokko. (right) Scanning sonar data plotted in MATLAB.	78
Figure 4-1: Block diagram of the core linear Kalman filter process.	82
Figure 4-2: Operational block diagram for linearized Kalman filter process.	84
Figure 4-3: LKF estimate (green) tracking actual position (blue), some distance from planned trajectory (red). Measurement is a corrupted range-bearing of two targets at $x=5, y=+2$. The estimate moves from overconfident initial incorrect estimate to actual path. Performance degrades as distance from targets begins to increase.	84
Figure 4-4: LKF estimator (green) tracking actual position (blue) and attempting to servo to planned trajectory (red). The estimate quickly tracks the actual position, allowing the controller to bring the system back to the planned trajectory. As before, the performance and certainty degrades as the vehicle moves away from the target.	85
Figure 4-5: Depth and pitch model	88
Figure 4-7: Uncompensated Root locus and Bode frequency response in heave	89

Figure 4-6: Pitch-Depth Control Topology. The controller uses a change of variables to control pitch and depth. Plant outputs are measured directly.	89
Figure 4-8: compensated time response in heave for two possible controllers	90
Figure 4-9: Uncompensated Root Locus and Bode system response for pitch.	90
Figure 4-10: Heading and speed model.	91
Figure 4-11: Heading-Speed Control Topology. Again, the controller uses a change of variables to independently compensate pitch and depth. Heading is measured directly, while velocity must be estimated from acceleration.	92
Figure 4-12: Uncompensated system response for yaw	92
Figure 4-13:Azimuthing thruster layout, top view. Two thrusters are positioned at a distance d from the centerline.	93
Figure 4-14: Phase degradation from various transit delays. As phase quickly rolls of with increasing slope, compensation quickly becomes impossible	95
Figure 4-15: Vehicle software running on the XAUV, viewed on a laptop through remote desktop	96
Figure 4-16: VB control software display. (a) joystick input; (b) DAQ: D/A control and A/D display; (c) DIO: Power switch board; (d) IMU orientation and control reference display; (e) Horizontal closed loop thruster display and parameter input; (f) Vertical closed loop thruster display and parameter input; (g) debug display area; (h) servo output control; (i) serial port enable; (j) battery level display; (k) thruster saturation and scaling input; (l) thruster output visualization; (m) operating frequency input.	98
Figure4-17: Interface block diagram	99
Figure 4-18: Simple analog and digital slew limit implementations	100
Figure 5-1: XAUV during in-water tests	101
Figure 5-2: Scanning sonar image of Brendan Englot, highlighted by the ellipse. A scanning discontinuity (area of update) is visible at ~ 15 deg right of vertical from slight movement of the vehicle , but the overall image quality is good.	103
Figure 5-3: Example data capture: vehicle recovery from a pitch disturbance	104
Figure 5-4: Calculated step response exhibits similar behavior to observed measurements, including $\omega_n \sim 6$ rad/s and $T_s \sim 10-15$ seconds	104
Figure 5-5: Distorted image of tank geometry. Moderate motion of the vehicle creates a misleading image of the surrounding environment.	105
Figure 5-6: Raw sonar data plotted in MATLAB. In addition to the obvious offset from vehicle roll, there is also disagreement between successive scans of the sonar. This is the result of the time required for the sonar to scan a full rotation.	106
Figure 5-7: Corrected sonar data. The sonar data is corrected for roll on a beam by beam basis. The cross-section of the MIT Towing tank is seen as the inner-most rectangle of strong returns.	107

Chapter 1

Introduction

Autonomous shiphull inspection is the seemingly straightforward problem of looking at a ship's hull for damage, fouling, or undesirable objects (bombs). In clear water, this can be accomplished visually, either with divers or underwater cameras. Unfortunately, most US ports have murky water, with visibility as short as a foot. The current solution in this situation is to send down a team of divers to feel every section of hull, looking for anything noteworthy. This is time-consuming and dangerous for the divers [1]. A preferred solution may be to send down a robot to do the job. A robot under remote control is better than sending a diver, but the optimum solution would be to drop a robot in the water, have it swim around the entire hull, and return once it was certain it had inspected every inch. In its UUV Master Plan, the Navy has called for mine reconnaissance as its top priority. Although initially referring to seafloor and tethered mines, it also has applications to the antiterrorism efforts of shiphull inspection [2].

1.1 Background

Current research in the MIT Marine Robotics Lab shiphull inspection group revolves mainly around the Bluefin Hovering Autonomous Underwater Vehicle, or HAUV. This is a large (1m cube) vehicle is designed to move in any direction, allowing its large sensor payload to always be oriented to the ship hull being inspected. The HAUV carries a Dual frequency IDentification SONar (DIDSON) as well as a Doppler Velocimetry Log (DVL). Recent developments in sonar technology have allowed much smaller sensors than the DIDSON, while still providing acceptable image quality. One of the more recent sensors is the blazed array sonar, roughly 1/4th the size and weight [3] [4]. As a result, large size of the HAUV may not be necessary when focusing on blazed array imaging. This has led to the opportunity for a new, smaller vehicle outlined in this research.

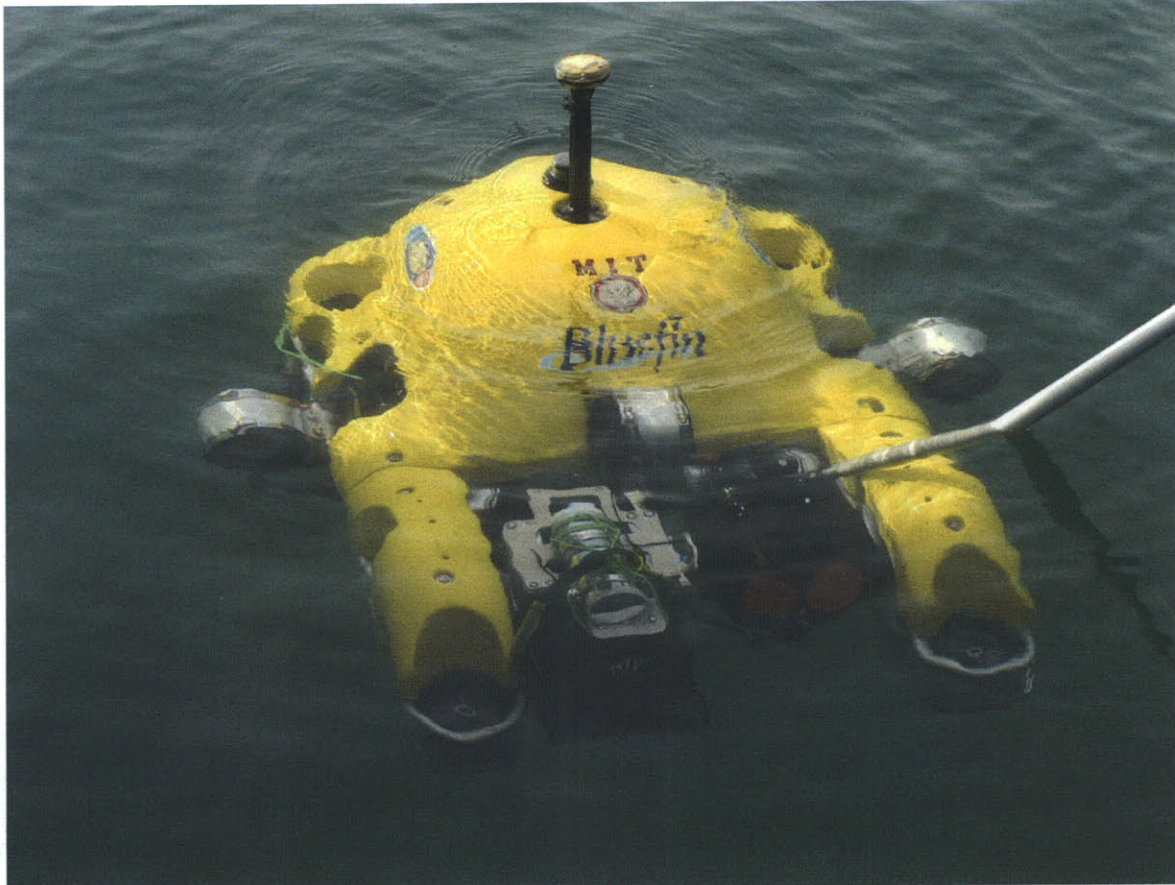


Figure 1-1: Bluefin/MIT Hovering AUV

This research also addresses an unmet need in the Marine Robotics Lab for a system with easily modifiable hardware and software that could be used to test new SLAM/control algorithms and sensors. This would ideally be a small system capable of operating in the lab tank and campus pools, for use when field deployment is impractical.

1.2 Requirements Identification

In order to fulfill the basic needs described above, it is necessary to develop a series of specific functional requirements from which to develop a solution. From the available on-campus pool facilities, it is apparent that the vehicle should be able to operate in small tanks and pools. Next, the specific interest in the blazed array mandates that the new vehicle should be able to carry one. As additional inspection sensors will likely become available in the future, the integration of these as additions or replacements should be as easy as possible. Similarly, it may prove equally useful for the hardware to have similar flexibility in evaluating different arrangements of propulsion. The vehicle will be used in testing SLAM code, likely both in education and development. Also there may also be occasion to operate the vehicle in the application environment, namely shiphulls or piers/pilings. In order to differentiate itself from the HAUV

band thus not simply duplicate its capabilities, the vehicle will attempt operations at a higher speed.



Figure 1-2: Testing tank at the MIT Marine Robotics Lab

1.3 Relevant work

Over the last decade, the depth and breadth of unmanned underwater vehicles has seen an significant increase. Whether result of increasing need and awareness for the application, or a decreased cost of development from a proliferation of supporting technologies, both off-the-shelf and custom platforms appear readily available.

Many large ROVs and AUVs have been developed by the groups such as the MIT AUV Lab, WHOI, MBARI, URA Lab, the Naval Postgraduate School, for cruising and hovering applications. Of greater interest, however, is a selection of smaller vehicles which have recently been developed. A first group of AUVs uses a torpedo shape to takes advantage of the long mission times for survey missions, while still allowing one-person deployment. The Bluefin 9 [5], the Gavia AUV [6], the Nekton (now iRobot) Ranger [7], and Virginia Tech 475 [8] each offer their own take on combining easy deployment with hardware modularity. Another set of small inspection-class ROVs has started to flourish. In the smallest, such as the AC-CESS ROV, VideoRay, and SeaBotix LBV, much creativity is required to modify the payload to carry large sensors. These generally have a dense layout, with a single molded housing, making them less suited for hardware development and sensor experimentation. Just larger than these, however, open frame designs become the norm, as in the SeaEye (Saab) Falcon. These generally have large buoyancy packs on top, with pressure housings, sensors, and actuators slung separately underneath, within the frame. These systems are readily modified to carry different sensors or actuators, but are neither as compact nor streamlined as their smaller cousins. Understanding the design tradeoffs for these systems will help evaluate the new proposed system.

Looking more closely at the shiphull inspection application, a few vehicles have also been adapted or specifically developed for the problem. Both the HAUV and the HAUV 1B were

developed specifically for shiphull inspection, and have been successfully fielded [9] [10] [11]. The REMUS 100 vehicle has been adapted for low-speed maneuvering by adding a pair of tunnel thruster sections at each end. Although still in development, this approach gives the interesting combination of fast, efficient transit and cruising behavior of a torpedo, with precise the sensor and vehicle positioning of a full hovering vehicle [12]. The Lockheed CETUS vehicle, developed by the AUV Lab, has also found use as an inspection platform [13] [14]. Although not specifically tasked for shiphull inspection, the MIT SeaGrant Odyssey IV [15] and WHOI Sentry [16] take an interesting approach to maneuvering by using vertically rotating thrusters for both diving and forward flight.

Lastly, the increase in popularity of underwater robot competitions at the High School and University level has created a large pool of small developmental robots. The AUVSI competition has been facilitating autonomous vehicle competition for over a decade [17]. An analysis of various robots as it relates to system architecture is presented in chapter 2.

1.4 Outline

Chapter 2 covers broad, system level considerations. Initially, an analysis of a developmental platform is conducted, using concepts from lean product design to provide insight. After identifying critical functional requirements at the strategic level, a series of potential design parameters are evaluated. First, the mechanical system and architecture are evaluated, and high-level strategies are selected. Then the electrical system is evaluated in a similar manner.

Chapter 3 looks into detail-level design of the system. Specific mechanical layout and components are considered and developed, within the strategic framework developed in the previous chapter. Then the specific electrical and electronic components are developed.

Chapter 4 covers the control system. First some overall approaches to vehicle control are discussed, including an overview of the linearized Kalman filter. Next the low-level system modeling and control are discussed. Last, details on the implementation of the preliminary user interface are provided.

Chapter 5 summarizes the experimental portion. The procedures required to bring the system online are summarized, including component and full system pressure tests. Also, the scanning sonar data are presented, as well as the approach and result from a beam-by-beam mosaic that combines the sonar with the IMU.

Lastly, Chapter 6 draws conclusions concerning the present and future state of the work.

Chapter 2

System Level Design

This chapter looks at the overall system architecture, starting from the concept of a developmental platform, and narrowing to architectural considerations. A series of necessary requirements is derived, partially from this analysis. Then the overall layout and mechanical design is considered, followed by a similar treatment of the electrical system.

In algorithm development, a simulation is typically developed for the hardware, sensors, and environment. In simulation, a large number of iterations can be executed with no risk to hardware. Because the hardware, sensors, and environment are all models inside the computer, the system can be simulated at many times faster than real-time. This is nothing new, and depending on the intent, can deliver results significant enough to eliminate the need for hardware trials. In many cases, however, it is desirable to validate the approach on test hardware prior to full implementation, in the case of industry, or before publishing, in the case of academia. So motivated, the hardware must then be powerful enough to support this terrain testing, or at least large tank testing; while not explicitly a size requirement, the ability to carry certain sensors, and operate in certain conditions, does suggest a certain size and power.

In designing a developmental system, full advantage should be taken of the fast cycle time for simulation. By resolving as much as possible when the cost of failure is essentially zero, the risk of implementing algorithms on a hardware platform is greatly reduced. Since the cost of failure for a hardware platform might be catastrophically high, some intermediate steps might prove relevant, where the cycle time is relatively high, risk moderate, and cost moderate. In any case, a well balanced development track will combat the increasing cost of failure with a reduction in the likelihood of failure through adequate preparation.

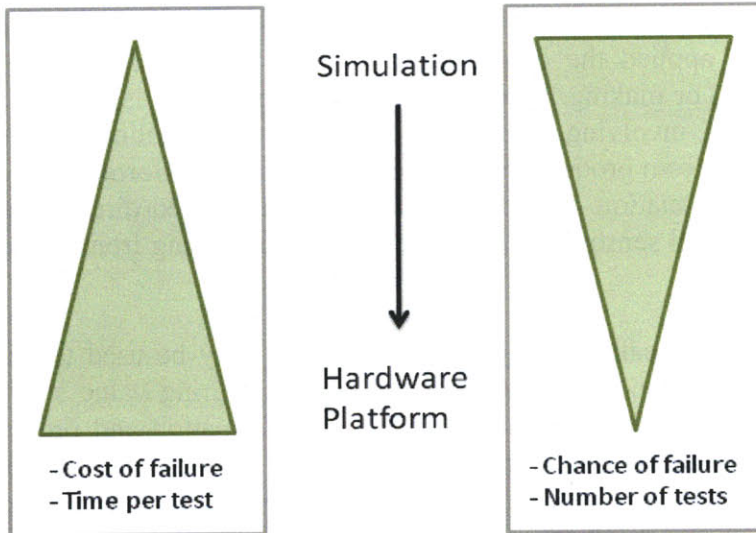


Figure 2-1: Successful path from simulation to hardware leverages low cost, fast cycle time of simulation to reduce chance of hardware system failure.

To validate complex theory, or test algorithms for complex environments, a complex testbed is typically required. The system must be sufficient to execute, or at least simulate the conditions for which the algorithms were developed. In the case of underwater navigation, terrain such as piers, shiphulls, reefs, and open water may be expected. While much of this may be simulated in a pool or tank, if the hardware is additionally capable of operating in this environment, the transition from tank to target environment will be much simpler. The step between tanks and target environment involves complications including recovery of failed vehicles, removal of the operator from lab resources, as well as management of logistical and scheduling problems.

In this section, the primary contribution comes through the analysis of developmental platform attributes. This consideration provides insight into the propulsion system, main power system, control system, vehicle layout, and interconnect system. Looking at the concept of a developmental platform, we come across the notions of flexibility and ease of use. The system should allow the user to focus more on the new elements under consideration, and less on integration with existing hardware. But first, some deeper insight into system development is helpful.

2.1 Lean concepts

From the area of lean manufacturing, we borrow a few concepts on efficiency in order to focus the efforts. First is *muda*, or waste. The underlying idea is that most of the effort in completing a product offers little direct value to the customer. Although the value-adding process itself can be improved, gains are much easier in auxiliary work or *muda*. Auxiliary work is the support effort that allows the value-added process, such as setting up a part in a machine, or conducting maintenance on a tool. *Muda* offers no additional value to the product or process, nor does it support the production process. Examples might include correcting errors, delays, or moving the product from place to place between processes [18 pp. 20-24]. In the context of robotic system development, some flexibility must be used in interpretation, as the process itself is the creation

of new software, sensors, etc, rather than the repeated production of identical parts. The MIT Lean Aerospace Initiative (LAI) has applied the classic wastes to the product development process. For example, overproduction, or making more parts than are being sold, is applied as designing a product in too much detail, involving unnecessary information, or failing to reuse prior development [19]. The adaptation from product design to developmental platform design is much easier. The product design interpretation of overproduction can be applied directly: too much detail in a simulation, a micron-level sensor for a meter-level task, or starting from scratch for every platform and iteration.

Another concept from lean manufacturing called Value Stream Mapping may be used to help identify the sources of waste, as well as generate solutions. In a manufacturing value stream map, specific symbols are used to summarize the process, including information and product flow, inventory, truck shipments, and machine operators, among others [18 pp. 87-91]. For the development of robotic applications, the details are changed, but the method, looking at the main processes and relations between steps, is retained. While manufacturing VSMs are highly detailed, the processes are well defined, and well-suited to such granularity. Product design VSMs may or may not be less detailed, depending on the task and level.

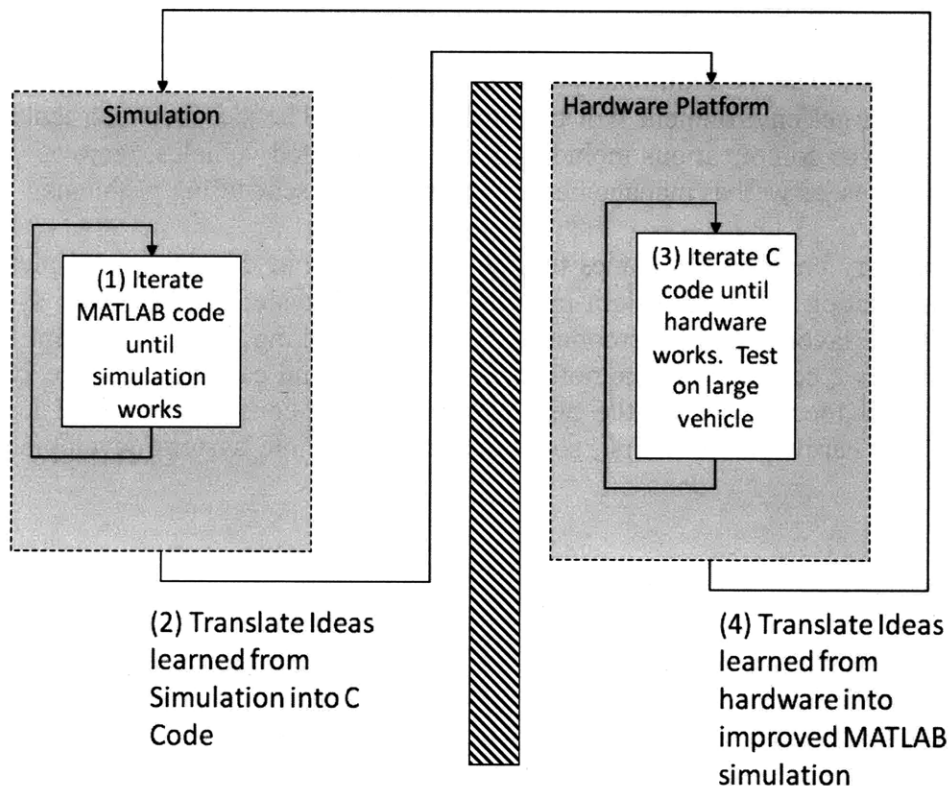


Figure 2-2: Typical Development Map with large barrier between simulation and hardware

Although the step between tank and world offers challenges, the simulation to hardware step is much more significant. Sophisticated simulations would include elements such as discrete time

processing, loop control jitter, software overhead and reduced bandwidth, sensor corruption, and other noise and asymmetries; however, doing so explicitly is difficult and adds considerable distraction from the core control or navigation study. Additionally, the difficulties in the step between a simulation and a large hardware platform may prevent rapid testing and recoding.

A better approach might involve a more incremental approach. If an intermediate hardware platform were available, one with similar or compatible sensors, but more suited for bucket-level testing, a development system could be proposed that would allow smoother exchange to both simulation and full testbed deployment. The goal is to lower the initial hurdle from simulation to hardware. The caveat is that this would add additional hardware, requiring its own complement of knowledge, maintenance, and support. The goal of this thought experiment is thus to find an implementation that provides sufficient benefit in implementation to outweigh its cost in overhead.

Because the initial hardware is used for lab tests only, many of the difficulties associated with a vehicle are eliminated. The vehicle does not need to carry a full complement of sensors, nor does it require an onboard power system. In fact, it does not even need onboard processing capabilities. A tethered system with only actuators and sensors onboard allows the control hardware to be standard, low cost components, without waterproofing. This concept is not new, and has been successfully applied in the form of actuated towed arrays, and even full vehicles [20]. Additionally, this can be thought of as a type of hardware-in-the-loop simulation. The automotive industry has made extensive use of HITL with flexible DAQ hardware and rapid user interface development tools, as with the MATLAB dSPACE platform [21].

The revised map does two main things. First, the barrier between hardware and simulation is eliminated by some means. The recommended method is delaying differentiation, a concept explained in greater detail below. The key concept is to retain as much of the effort as possible between simulation and hardware implementations. Second, it offers the possibility for an inner-loop using intermediate hardware to refine the algorithm.

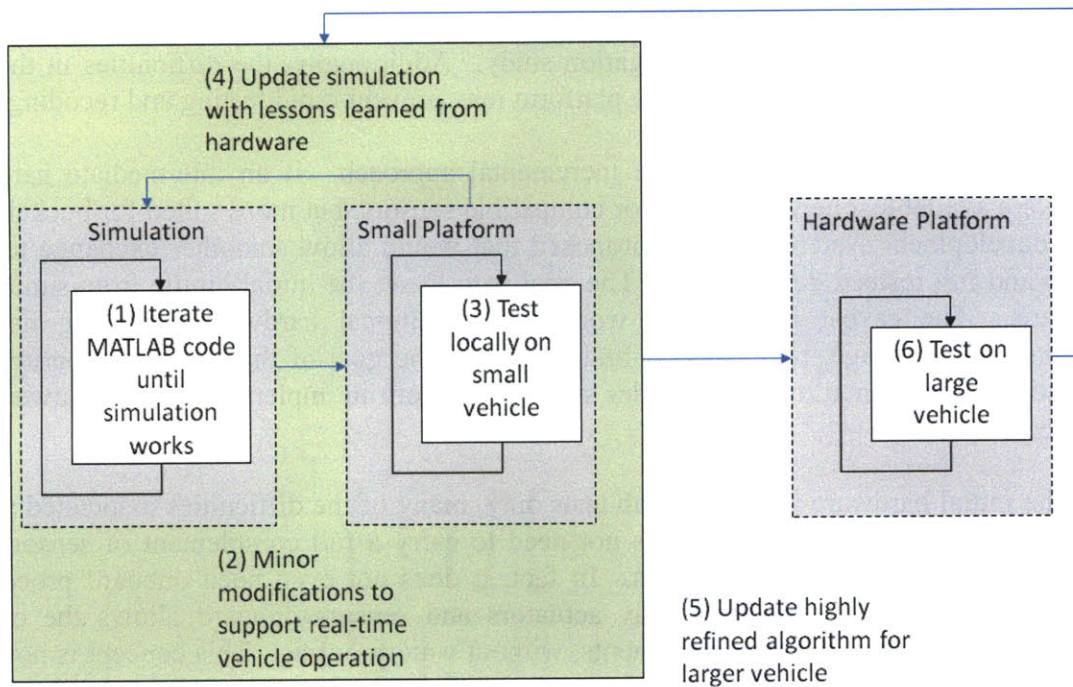


Figure 2-3: Development map using a common MATLAB framework and small, intermediate hardware

2.1.1 Model considerations

Before the notional hardware can be developed too thoroughly, a better understanding of the algorithms under consideration is required. In much of the author's work with using Kalman filters for navigation, a two-dimensional approach is used. While 3D algorithms are the eventual goal, much can still be learned from a 2D approach. Additionally, there is a class of problems that can be solved as a 2D problem or simplification. In the case of surface navigation, the air-water interface provides a convenient planar surface, assuming local or temporal disturbances like waves and tides can be accounted for or neglected. Open seafloor can also be treated as two-dimensional, as long as the local features are simple. If the bottom terrain is too complex for a simple 2D consideration, depth may be used to operate in slices. As essentially the only absolute position sensor available to an underwater vehicle, pressure or depth can be used in a separate closed loop process to decouple the complex horizontal planar navigation from vertical. More details on the various control approaches are available in chapter four.

With a planar system sufficient, or even desirable for the intermediate hardware, suitable systems are now considered. Although wheeled vehicles are common in the research field, they do not offer free-floating, or double-integrator behavior. Put simply, when a wheeled platform is commanded to stop, it doesn't drift. The kinematic relation between wheel motion and vehicle motion make them somewhat less relevant to free-flying underwater vehicles. Mathematically speaking, the vehicle position can be as simple as a function of the control output wheel position θ_{wheel} , and the wheel circumference c .

$$X = c * \theta_{wheel}$$

Alternately, if motor velocity is controlled, a reasonable approximation for motors with simple speed controls, the position is the first integral of some controlled wheel velocity $\dot{\theta}_{wheel}$ and the wheel circumference c .

$$\begin{aligned} \dot{X} &= c * \dot{\theta}_{wheel} \\ X &= \int c * \dot{\theta}_{wheel} * dt \end{aligned}$$

Certainly some wheel-ground interactions in coarse media like sand could cause significant slip, but extreme conditions would be required to recreate the same effect. Double integrating platforms offer no guarantee that they will stay where they are stopped. This is because they are controlled as a force acting on a free mass. Only the acceleration is directly controlled. This is a function of the actuator type, such as thrusters or air jets, which exert a force F on the controlled mass M . Thus the position of the vehicle is the double integral of the control input.

$$\begin{aligned} \ddot{X} &= F/M \\ X &= \frac{1}{M} \int \int F * dt dt \end{aligned}$$

Additionally, it is necessary to consider order of the noise. In the case of wheeled vehicles, if the wheels are ‘locked’ to the ground, any external forces are passed from the vehicle through the wheels into the ground. In effect, nothing happens. Alternatively, for a free swimming vehicle, any disturbance forces are accumulated in the control mass. Intuitively it is expected that a free-floating body will start to drift randomly over time, exhibiting a random-walk type behavior. More exotic types of disturbances are also encountered, but are not considered in this summary analysis.

2.1.2 Intermediate Hardware Details

With the understanding that a free-floating planar vehicle is desired, it is easiest to take advantage of a system passively held in plane, as by gravity. A hovercraft floats just above a solid or liquid. Similarly, a raft floats at the water-air boundary. An underwater hovercraft platform could be envisioned to operate at the water-solid interface at the bottom of a tank. The hovercraft offers the benefit of no water involved, requiring minimal protections for the hardware. This does preclude the use of water-based sensors, namely sonar; however similar sensors, such as laser scanners and cameras can be used instead. It also has a higher cost for payload. Any additional weight must be offset by larger horizontal size or stronger fan. A raft adds payload capacity very simple, by adding more flotation. Additionally, both land and water sensors may be used: sonar underneath the raft, and lasers above.

Table 2-1: Comparison of Intermediate Hardware Platforms

Platform	Advantages	Disadvantages
Land Hovercraft	<ul style="list-style-type: none">- no waterproofing- very low drag (air dynamics dominate)	<ul style="list-style-type: none">- no water-based sensors- limited payload- dynamics different from water
Raft	<ul style="list-style-type: none">- land and water sensors possible- extensive payload- water dynamics (closer to target system)	<ul style="list-style-type: none">- some waterproofing required- pool or tank required

The main differentiation is between sensor capability and tank requirements. Although a hovercraft's ability to operate without a water tank is compelling, the difference in physical dynamics and sensor capability will ultimately create a more complex transition from simulation to full platform. Its utility should be remembered for development not requiring a full platform implementation. For the purpose of full underwater system deployment, or simply an underwater sensor, the raft wins out. In his thesis, Mike Kokko developed a raft for testing SLAM algorithms [10]. Since dubbed the KSV, the raft is a simple design with four thrusters and a central sensor mount. Power and control electronics are off-board. Work with the KSV is ongoing by Brendan Englot in his integrated control research. As the KSV has already been developed, it will not be covered in this paper, except where needed to discuss or evaluate a common software platform or sensor physical layer.

Taking a step back, the development and introduction of this intermediate hardware can be evaluated as a magnifying lens for the benefits and difficulties of streamlining a developmental process. With a single step between simulation and hardware, some rework at each step will likely be tolerated. In a three-step process, however, the added difficulty of adapting the high-level ideas a second time for the second platform will likely result in the intermediate hardware becoming the primary development platform, and may ultimately slow progress. Alternatively, and from a positive light, a successful inclusion of intermediate hardware will highlight the ease with which the high-level ideas can be moved from platform to platform.

Similarly, the inclusion of the intermediate hardware serves as a crucible for extracting the key needs for the system. The primary purpose for the developmental hardware is to develop and test new sensors and control/mapping schemes. The main obstacles are associated with moving the key information within and between hardware and software. The point of interest from a system point of view is the point of differentiation between the implementations.

2.1.3 Delaying differentiation

The point of differentiation is the level at which multiple implementations are different. Anything upstream of this point is the same between simulation and various hardware implementations. Anything downstream is different, allowing application for different scenarios. One key principle of lean product development is delaying this point of differentiation [22 p. 3.2]. From the perspective of *muda*, anything after this point of differentiation must be solved multiple times, and is overproduction [23 p. 4.3].

A classic example is a laptop power supply. With an integrated power supply, each model must be designed to accommodate each of the AC/DC converters for the various customer countries. This will likely impact the chassis layout, and will create variants of an otherwise identical computer. Production estimation and inventory are now more complicated, because each laptop can only be sold to a particular country set. As an alternative, consider an external power supply. The supply input is customized to each country set, but the output is identical. Now, only one expensive laptop is manufactured, while several inexpensive power supplies are produced. The production estimation and inventory problem is much simpler, because regional differences account only for a small percentage of the total cost, and represent a self-contained module that may even be shipped separately[24].

Applying the concept to developmental platform creation, the point of differentiation is the point at which concepts, code, or hardware must be recreated in a separate module. Looking at the possible break points, the most obvious are at the core approach, at the raw code, at a self-contained code module, or at the hardware level. In general, if the point is farther towards hardware, there is reduced rework, and reduced number of data transport channels.

Table 2-2: Algorithm module differentiation points

Differentiation point	Description	Analysis
Strategy: Core Approach	Basic math structures and information flow. Recode on each platform for each language and hardware.	Applicable to all systems, but requires the most rework.
Software: Raw Code	Block of code that is reinserted into main operating code for each platform.	Requires same or similar languages, but prevents rework of core algorithm section
Software: Code Module	Precompiled module of code that shares a common external software interface to hardware operating software. May be implemented as a .dll on a windows machine, for example.	Allows different languages for high-level algorithm and hardware operating code. Requires a standardized set of inputs and outputs between modules.
Hardware: Internal to Control Stack	A separate computing component that allows dedicated high-level processing, but requires low-level control stack electrical resources	Allows physical transport of computing module, but requires an equivalent interface at each platform
Hardware: Separate Module	Fully separate hardware control module that only requires a high-level communications interface, such as Ethernet.	Allows greatest flexibility in host system configuration, as any control module can communicate with any host, as long as each shares the same communication line. This also results in additional overhead on each system.

In lean thought, the goal is to delay differentiation as late as possible. The framework developed above introduces a slight twist. Using the laptop analogy, which component is the laptop, and which is the power supply? From the point of view of the algorithm, the actual algorithm is akin to the laptop, while the various hardware platforms represent the regional power standards. The analogy is not perfect, since although the main deliverable is the algorithm, the hardware platforms are complex and expensive, unlike the power supplies. If used anyway, this perspective suggests that the best differentiation points are farther toward the hardware.

If considered from the opposite end, the developmental platforms represent the laptop, while the different algorithms represent a number of power supplies. The intent behind this approach would be to give emphasis on the time and cost behind the hardware platform development. From this perspective, as before, the purpose is to minimize the cost associated with integrating things that are changing. With three static development platforms, but one widely varying algorithm, the best differentiation points are again farther toward the hardware level.

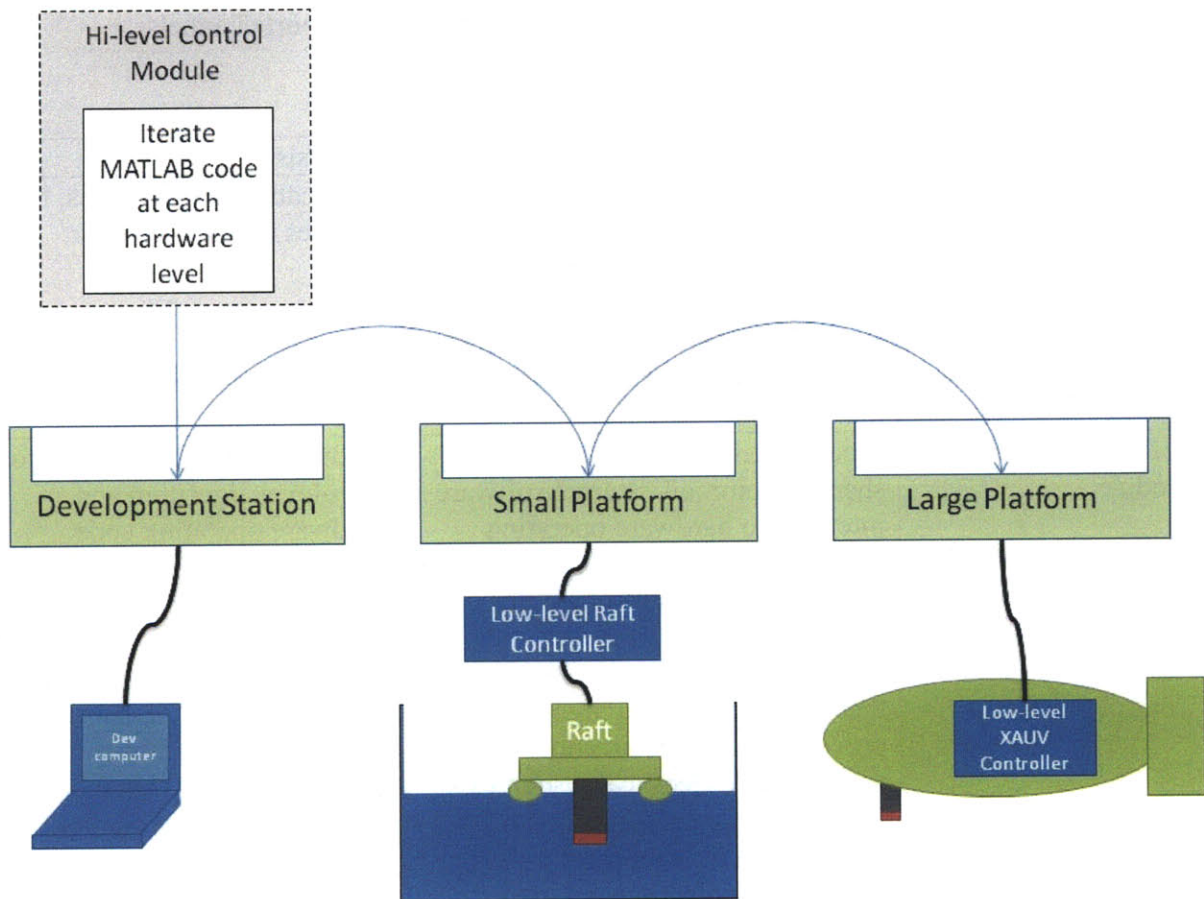


Figure 2-4: Minimization of rework depends on transporting functional blocks (whether code or code-bearing hardware) between development and target environments.

Two likely implementations would be either the fully contained code module, or fully contained hardware module. Although a semi-integrated hardware module has a fully separate software

level, insertion and removal of a PC104 card, for example is a drastically more complicated process than either replacing a code module or unplugging an Ethernet cable. Some clarity is gained in specific example solutions. In the software case, a single algorithm code module is loaded onto the separate hardware platform, using a common interface to talk to simulated or real sensors and actuators. In a MATLAB system, each platform would be required to run MATLAB, or at least a compiled output. The hardware interface segment might be one or more dynamic link libraries (.dll files) that could be called directly from the MATLAB main program. Alternatively, a LabVIEW system could use the graphical programming environment to handle sensor and actuators, while directly inserting a MATLAB m-file script to handle the algorithm side [25]. For a ground up system design, National Instruments offers tight integration between its sensors and software. COTS hardware-in-the-loop development platforms like xPC Target and dSPACE offer a slight twist, where development is done on one module and interfaces to another for operation. In practice the development station compiles and embeds the code into the hardware platform, so there some flexibility in classifying this approach. In any implementation, the key is separated and self-sufficient algorithm code and hardware/hardware driver code.

In the hardware case, a physical high-level controller is moved between each platform. Both the algorithm module and hardware controllers might be embedded controllers, Windows computers running MATLAB or C, Linux computers running MOOS or C, or even something else. There is full flexibility on the part of the developer, as long as the hardware controller can talk to its attached sensors and actuators, and the algorithm module can effectively develop and run the desired algorithms. Whatever specifics are used, the algorithm module is physically separated from the hardware driver firmware.

In summary, in order to reduce the amount of effort that must be recreated between multiple platforms for a common algorithm, the point of differentiation should be delayed as much as possible. First, this reduces the work on the algorithm side by allowing a common framework to communicate with the hardware. Second, it quarantines any differences between hardware platforms within those platforms. Two identified implementations use self-contained software or hardware modules to facilitate this information transport.

2.2 Parameter Identification

As described in the previous chapter, the basic set of functional requirements outlines the overall objective of the vehicle. In other words, if a design meets all functional requirements, it will achieve everything that it is intended to do. For this list of functional requirements (FR), there must be another set of design parameters (DP) that specify how to achieve these. The design process can be thought of as a successful matching of the what (FR) to the how (DP) [26]. This design process begins at the system or strategy level, where high-level objectives are identified. An initial pass at requirement and parameter mapping is executed below, and offers some insight into the development to come.

Table 2-3: Mission Strategy - Functional Requirement/ Design Parameter Mapping.

FR	DP
Operate in Small Tanks (20'x10'x4')	<ul style="list-style-type: none"> - function at low/no speed - independent vertical motion - turn in place - tethered ok
Carry Blazed Array/Similar	<ul style="list-style-type: none"> - get blazed array looking in needed direction - flexible sensor mount and interface
Test New Slam Code	<ul style="list-style-type: none"> - fast computer - easy access for reprogramming - incorporate failsafe/recovery - allow incremental implementation
Ship Hull/Pier + Piling Inspection	<ul style="list-style-type: none"> - open water operation - (relatively) long mission time - untethered (no tangles)
Maneuverability at speed	<ul style="list-style-type: none"> - control surfaces/propulsors providing sufficient control authority at speed
Useful in Education	<ul style="list-style-type: none"> - easy development of code and hardware - robust mechanical system

A complex system like an underwater robot will often have interrelated, complementary, or even conflicting requirements. This can be used to the benefit of the design process: some of the above design parameters offer a basis for evaluating other categories. For example, flexible hardware does not immediately suggest one particular thruster layout. When considered in context of the propulsion system, however, it will provide greater insight into the degree of integration required of the best solutions. A design parameter sorting process will help by arranging the parameters based on their region of influence over the design. This approach will facilitate the generation of consensus specifications.

Table 2-4: Reorganized parameter mapping

Main system ==>	Design parameter ==>	Likely subsystem relevance
Mechanical Layout	<ol style="list-style-type: none"> 1. function at low/no speed (turn in place), independent vertical motion 2. flexible sensor mount and interface 3. open water capable 4. robust mechanical system 5. incorporate failsafe/recovery 6. allow incremental implementation 	<ol style="list-style-type: none"> 1. propulsion system configuration 2. overall layout, sensor mount designs 3. overall layout, propulsion system configuration + size 4. overall layout 5. buoyancy, aux systems 6. overall layout
Electrical System	<ol style="list-style-type: none"> 1. fast computer 2. easy access for reprogramming 3. incorporate failsafe/recovery 4. flexible sensor interface 5. allow incremental implementation 6. easy development of code 7. tethered ok in tanks, but untethered (no tangles) in free water 	<ol style="list-style-type: none"> 1. control stack 2. control stack, architecture, connectors 3. aux electronics 4. connectors, architecture 5. electronic architecture 6. control stack 7. auxiliary systems
Control/HMI	<ol style="list-style-type: none"> 1. easy development of code 2. easy access for reprogramming 3. tethered ok in tanks, but untethered (no tangles) in free water 	<ol style="list-style-type: none"> 1. operating system 2. user interface 3. operating system, level of autonomy, user interface

Now, with design parameters organized by relevant system, the design of those systems can begin. The specific requirements from the original functional requirements will be combined with the more general lessons learned from the lean study of platform design.

2.3 Overall Layout and Mechanical Design

The overall layout of the vehicle has considerable impact on the types of actuators and sensors it can carry, and so the types of missions it can run. Also, the method by which these components are connected and integrated has a direct impact of the ease with which additional components may be added in the future. Because the vehicle is designed to accommodate a range of different payloads, the actuation system becomes the driving force behind the initial layout.

Specifically, the quantity, type, and arrangement of propulsors both determines and is determined by the overall layout and structure of the vehicle. In order to limit the scope of this work, novel or exotic types of thrust generators were not considered for the baseline configuration, with the understanding that a successful design will allow a future researcher to easily incorporate them at a testbed level of integration. Although a full planar hovering capability is highly desirable, meaning actuation in surge, sway, heave and yaw, it is not

explicitly required. Active control of pitch and roll are also not required. Jumping ahead to the result, the eventual baseline design uses two vertical thrusters and two forward thrusters to provide tank-like steering with independent vertical control

In order to put numerical parameters to the general specifications, some idea of the maximum operational environment is required. From the NOAA current charts, East coast currents rarely exceed 2kts [27]. This remains true within Boston harbor, the most likely operational environment. Operating in currents in the 1-2 kts range is therefore a target specification. In order to operate with any hope of making forward progress, the maximum vehicle speed must be greater than the operating currents. If full hovering capability is desired, this performance must extend in all relevant directions. In other words, being able to hover broadside to a strong current may pose a very different and more difficult problem than nosing in.

Looking at the functional requirements related to propulsion, a series of variations is possible, using the number of thrusters, degrees of freedom for the thrusters, and additional fins as variables. Although the minimum number of actuators or thrusters can be determined as a function of the desired output degrees of freedom [28], this approach may introduce geometries which are inconvenient for a broader platform.

Table 2-5: Strategy 2 - Motion Iteration

FR	DP
Independent vertical motion, turn in place, (maneuverability at speed)	<ul style="list-style-type: none"> - 5 Thrusters - 4 Thrusters - 4 Thrusters + Fins - 3 Thrusters - 3 Thrusters + Fins - 2 Thrusters + Fins - 1 Thruster + Fins Multiple thruster DOF
Blazed array in needed position	<ul style="list-style-type: none"> - roll passive stable, fixed position - roll passive stable, active positioning - roll active control, fixed position - roll active control, active positioning - roll unstable

For the baseline configuration, static thrusters are desired, in order to make the development as incremental as possible. An assortment of possible motion strategies are selected from the initial list in order to identify and extract positive attributes.

Table 2-6: Expanded Iterations

Strategy A: Hovering	
FR	DP
Independent vertical motion, turn in place	- 5 Thruster (2Fwd 2Vrt 1Lat)
Blazed array in needed position	- roll passive stable, fixed position - roll passive stable, active positioning
Strategy B-NF: Basic	
FR	DP
Independent vertical motion, turn in place	- 3 Thrusters only (2F 1V)
Blazed array in needed position	- roll passive stable, fixed position - roll passive stable, active positioning
Strategy B-F: Basic Finned	
FR	DP
Independent vertical motion, turn in place	- 3 Thrusters (2F 1V) - Front + rear control surfaces
Blazed array in needed position	- roll active control, fixed position - roll active control, active positioning
Strategy C: Max Finned (outside FR)	
FR	DP
Independent vertical motion (unmet), turn in place	- 2 Thrusters (2F) - Front + rear Fins
Blazed array in needed position	- roll active control, fixed position - roll active control, active positioning

The hovering capable vehicle offers the ability to move sideways by use of a tunnel thruster. This is a highly desirable capability, but may put significant restrictions on the vehicle geometry. Both the basic vehicle and the max finned vehicles have small number of thrusters, but lacks full planar capability. The both finned vehicles gain some sideways maneuverability when at speed, but lose this when hovering. For the second level parameters, there is a two way choice for active or passive roll control and sensor positioning. The hovering vehicle has a great advantage in positioning the sensor without active control; the other designs, however, greatly benefit from being able to position the sensor in a direction other than the direction they are headed. As an initial design, the basic un-finned layout is selected, with provision to add an actively positioned sensor array, as well as low-priority development of either fin modules or a cross-body tunnel thruster. Should geometry prevent a single vertical thruster, a pair may be used instead.

2.3.1 Hydrodynamic Analysis

In the requirement-parameter mapping process, it was determined that passive upright stability was desirable. This means that the vehicle should float upright, even in the presence of minor disturbances. A stacked-hull, multi-thruster layout was selected for its combination of high passive roll stability and damping, easy battery switching, and adaptability to larger payloads. The following analysis outlines the relevant hydrodynamic concerns.

Righting Moment and Damping:

The righting moment is the torque that the system applies to return itself to equilibrium when disturbed in an angular direction. If the system is to be passively stable, this should be as high as practical. A larger righting moment for the same angular displacement means that the system will return to equilibrium faster and earlier; additionally, it will be better able to reject disturbances in these directions. The righting moment is proportional to the sine of the angle between vertical and the axis between the center of gravity and center of buoyancy, θ_{cg-cb} . The moment is also a function of the half-distance between the centers R , and the buoyant and gravity forces F_g, F_b acting on the centers.

$$M_{righting} = (F_b + F_g) * R_{cg-cb} * \sin\theta_{cg-cb}$$

In some systems, passive stability may not be desirable. If the vehicle is to perform acrobatic maneuvers, it will have to exert forces greater than the righting moment in order to produce a given angle. For highly acrobatic vehicles, the system may even be unstable. If the system is to have reduced passive stability, the system will require more complex control algorithms, and possibly more control surfaces/thrusters.



Figure 2-5: The X-29's unstable design requires a complex control system, but allows highly acrobatic maneuvers [29].

The righting moment is generated by the difference between the center of buoyancy and center of mass. The center of buoyancy is located at the average point of the displaced volume.

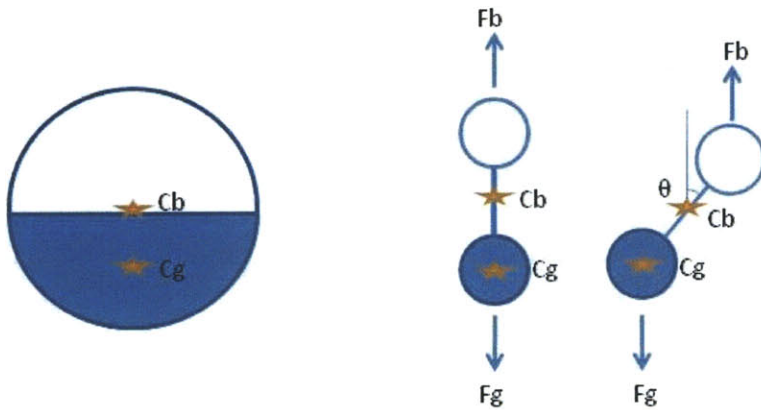


Figure 2-6: Righting moments and centers of buoyancy and gravity for one and two-tube designs

In a single tube design, the center of buoyancy is on the centerline. As a result, even for an 8” tube, the maximum difference between center of mass and center of buoyancy is 4” if everything is inside the tube. More likely, the mass will occupy some volume, and would be closer to 2” away. In the two tube case, the center of righting moment can be increased by changing the separation between the tubes. Because of this, the two hull design provides an easier approach to passive roll stability.

Without any damping, the vehicle would oscillate back and forth as a result of the righting moment. The damping of this process results from water moving across the surface of the vehicle in the roll or pitch direction. Long, slender vehicles typically have plenty of damping in pitch and yaw. In the roll direction, however, these cylindrical designs have almost no damping; this can lead to undesirable performance. Fortunately, drag can be quickly added by adding protrusions. Even simple, un-actuated fins or external structures can greatly increase the settling speed. If the protrusions take the form of actuated fins, they can alter the righting performance of the vehicle, allowing fast settling or dynamic maneuvers.

A twin hull design, on the other hand, must present two cylinders translating through the water, rather than one simply rotating.

Thrust and Drag:

Put simply, the interaction of thrust and drag set the upper limit of speed for a vehicle. Additionally this dictate the maximum current in which a vehicle can operate. From the operating area requirements, the vehicle should be able to operate in up to 2kt currents.

The general equation for drag relates the drag force F_d to an empirical drag coefficient C_d based on shape, the fluid density ρ , frontal surface area A , and forward velocity V .

$$F_d = C_d \frac{1}{2} \rho A V^2$$

For the XAUV, a drag coefficient of 1 is assumed. Highly streamlined vehicles can achieve coefficients less than 0.1 [30]. Assuming that relatively little effort is put forth into streamlining

the vehicle, the choice of a C_d of 1 is reasonable. By assuming this high drag, appropriately sized thrusters can be selected in order to meet the target performance. In order for the vehicle to operate in 2kt currents nose in, the top speed must exceed 2kts. Top speed or terminal velocity for a vehicle is reached when the drag force equals the thrust output of the thrusters. With the above parameters, this is achieved with thrusters generating less than 50N. In the event that hovering may be required, the worst case scenario must be accounted for: a broadside current. With a vehicle side area of approximately 0.3m^2 , and a drag coefficient approaching 1.6 [30], the force required of the thrusters leaps to 235N. The class of thrusters capable of this performance is much smaller than the earlier specification. A pair of 170N thrusters by Tecnadyne are selected for their capability, as well as onboard power electronics and magnetic safety coupling.

Thrusters put out the strongest force at no forward speed; this is called bollard thrust output. When the thruster is moving through the water, however the output jet is slower relative to the surrounding water than in the bollard case. Essentially, if the thruster can generate e.g. a 20kt stream of water, the momentum transfer is reduced when the surrounding flow is e.g. 5kts, for an effective jet velocity of 15kts. The 340N pair Tecnadyne thrusters that are implemented de-rate at approximately 9% per knot forward speed [31]. This results in a maximum forward speed of approximately 3.5m/s, illustrated below.

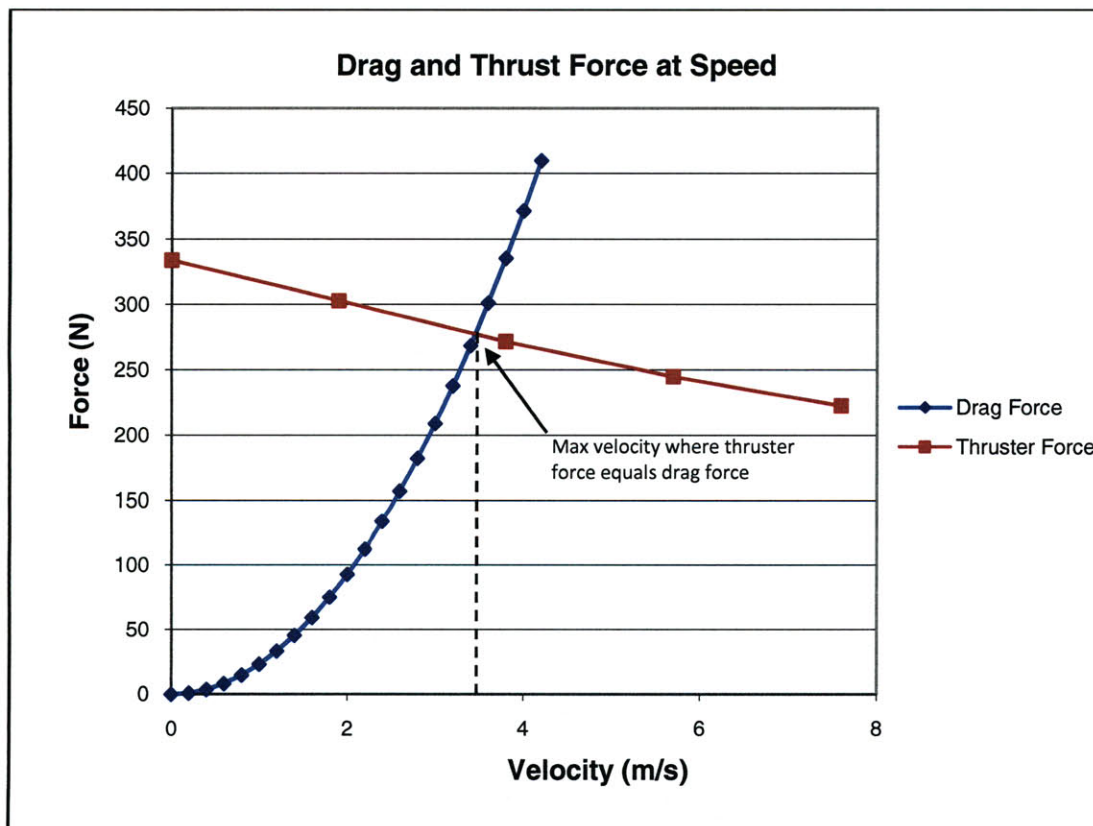


Figure 2-7: Intersection of drag and thrust force curves determines the maximum vehicle speed. The above graph shows forward velocity.

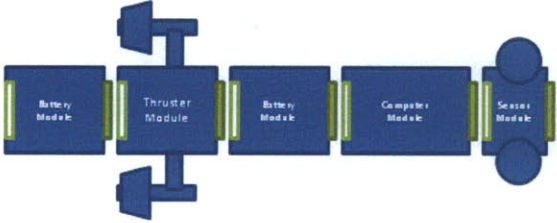
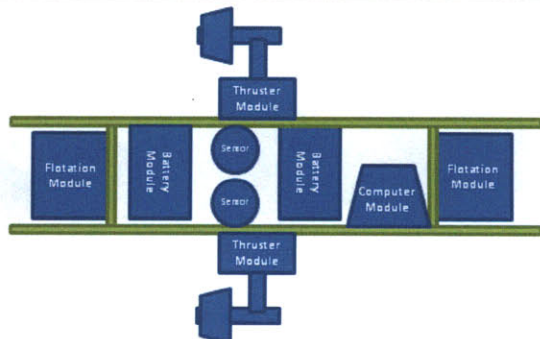
In the worst case scenario of hovering broadside to a current, the thrust capability gives a more modest performance. At approximately 1.1m/s, or 2.2kts, the drag force equals the derated thruster force.

2.3.2 Structural Backbone

Because the vehicle is designed for the addition and subtraction of components, a single hull and faring design is less practical than a modular frame design. In determining the structure and component interfaces, two main alternatives presented themselves. The first uses the components and mechanical connections as the main structure. This design is well-suited to vehicles with well-known configurations, and single-level inline hulls. Several commercial AUVs, such as the Remus [32], Bluefin-21 [5], Gavia [6], have used this with great success. The design allows different parts, such as DVLs, ADCPs, etc. to be added, while still maintaining a clean hydrodynamic profile. For payloads that can be naturally configured into similar interface geometries, this is an excellent design that eliminates additional complexity.

The second option uses an external structure, to which components are attached. This is a more common design approach for ROVs. These options can be thought of lending themselves to sectional-modular architecture and slot/bus modular architecture respectively, as described by Ulrich. In a sectional architecture, each component interfaces with a single standard interface to any of the other component. In bus or slot architecture, each component connects to a single bus or body. In bus architecture, these interfaces are identical; in slot, they are not [24 p. 166].

Table 2-7: Architecture tradeoffs and examples

	Structural Components (sectional)	External structure (slot/bus)
+	<ul style="list-style-type: none"> - very easy from operator standpoint - easy to increase overall size 	<ul style="list-style-type: none"> - easy to add components of different size
-	<ul style="list-style-type: none"> - heavy frontside engineering - difficult to add components outside of scope of original paradigm - smaller components difficult to add 	<ul style="list-style-type: none"> - less streamlined interface - difficult to alter overall size
Ex		

Although the sleek arrangement of structural components is tempting, a fixed size limits the flexibility of the designer for new modules and sensors. In order to add new modules, the full sectional interface must be created and adapted for the new sensor. In the slot/bus physical architecture, a much smaller portion must be recreated or adapted to the sensor. The vehicle core purpose is for rapid testing of new and not-yet-conceived sensors and components; this goal is most easily met through an external structure.

The initial design uses threaded rod as an external skeleton, to which other modules can mount. Although effective, it does not provide strong rotational resistance if only a single frame member is used; for two adjacent members must be used. Additionally, the hydrodynamic profile appeared to present excessive drag. A potential improvement would be to use an extruded form, like 80-20 rod. With this, components can be mounted to the frame using only a single frame member, while still providing significant resistance to motion in all directions. Several university robots have successfully taken this approach, including UOttawa [33], MIT Orca [34], and Georgia Tech [35]. The mounting locations on the 80-20 would also facilitate adding plate fairings to the sides of the vehicle. Through a SolidWorks FloWorks simulation, however, it was determined that the threaded rod approach added minimal incremental drag, when compared to the overall geometry of the vehicle.

2.3.3 Modular Capabilities and Concepts

In order to fulfill its mission of a readily adaptable platform capable of simulating a number of existing and proposed underwater vehicles, a set of modular capabilities have been conceived and designed into the framework of the vehicle. Each of these add a performance capability to the system that is both desired and incremental, but not required for baseline functionality of the vehicle.

Front Sensor Platform

The front sensor platform is an actively servoed, 2DOF positioning system for sensors. This adaptation decouples the orientation of the vehicle from the orientation of the sensors. Because of this, it would then be possible to scan a shiphull side or seafloor while traveling parallel to the surface. The front sensor platform has been redesigned and fabricated by SB student Devin Neal and UROP student Brooks Reed.

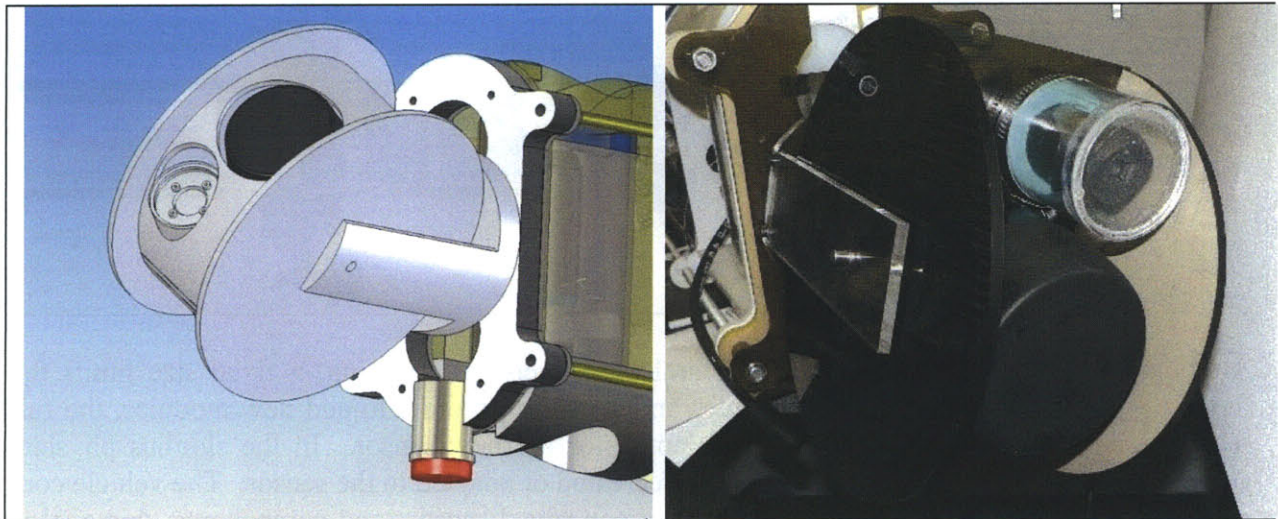


Figure 2-8: front sensor platform CAD model and static mockup

Front and Rear Fin Sections

Separately actuated fin sections are available for front and rear mounting. Each section consists of four independently steerable fin module. These fins provide roll damping, as well as fast turning at high speed. As a result, they allow hardware simulation of REMUS/Nekton Ranger style vehicles which operate with forward thrusters and tailfins.

The fins are actuated with oil pressure compensated RC servos. An additional gear reduction stage is built inside the housing to allow increased torque and higher resolution motion, at the expense of actuation speed. The drivetrain is built with a 1" spacing between drive and driven gears, so several gear ratios can be achieved between 1:3 and 3:1. The fin design would be graciously borrowed and cast in urethane as developed by Jordan Stanway and the Marine Hydrodynamics Lab. The fins can be interchanged by a simple shaft coupling. A prototype waterproof actuator has been revised and fabricated by SB student Megan Roberts.

Large Payload Capacity

In order to carry large sensors on the order of a DVL, the vehicle is flipped sideways, and the sensor is mounted on the new bottom (old side). Additional flotation is mounted on the new top side. The vehicle is now capable of full planar motion. A typical implementation of this allows the vehicle to stay on the surface, and uses the DVL for bottom scanning. Depth can only be controlled through fins, requiring forward motion, or the actuated thrusters.

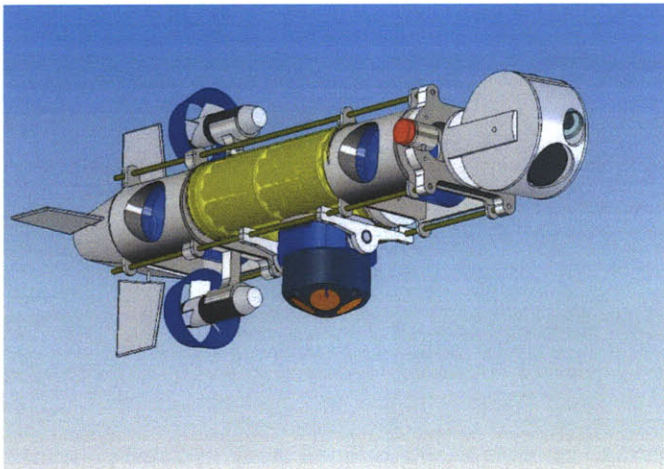


Figure2-9: XAUV concept with DVL in bottom/top scan mode.

IDOF Actuated Thrusters

By actuating both main thrusters in a dual z-drive arrangement, it is possible for the vehicle to gain full hovering capability. This is a similar approach to the MIT AUVLAB Odyssey IV vehicle, only rotated 90°. The OIV has a pair of thrusters mounted on a single rotating cross beam mounted in the center. When pitched up or down, the thrusters control heave; when

pitched forward, the thrusters control forward velocity and steering control. A second pair of cross-body thrusters allow sway control as well as redundant yaw control.



Figure 2-10: Odyssey IV at field operations in Georges Bank. The two main thrusters are controlled by a single rotating thruster unit, incorporating a brushless kit motor and resolver position sensor. This configuration allows high power vertical and forward motion with the same thrusters.

In the XAUV implementation, the actuated thrusters move independently in the yaw direction (azimuthing). This allows forward and steering control while directed forward, but can also rotate outward to control sway. This configuration is similar to the podded propulsors found on many large commercial surface vessels, especially cruise ships [36].

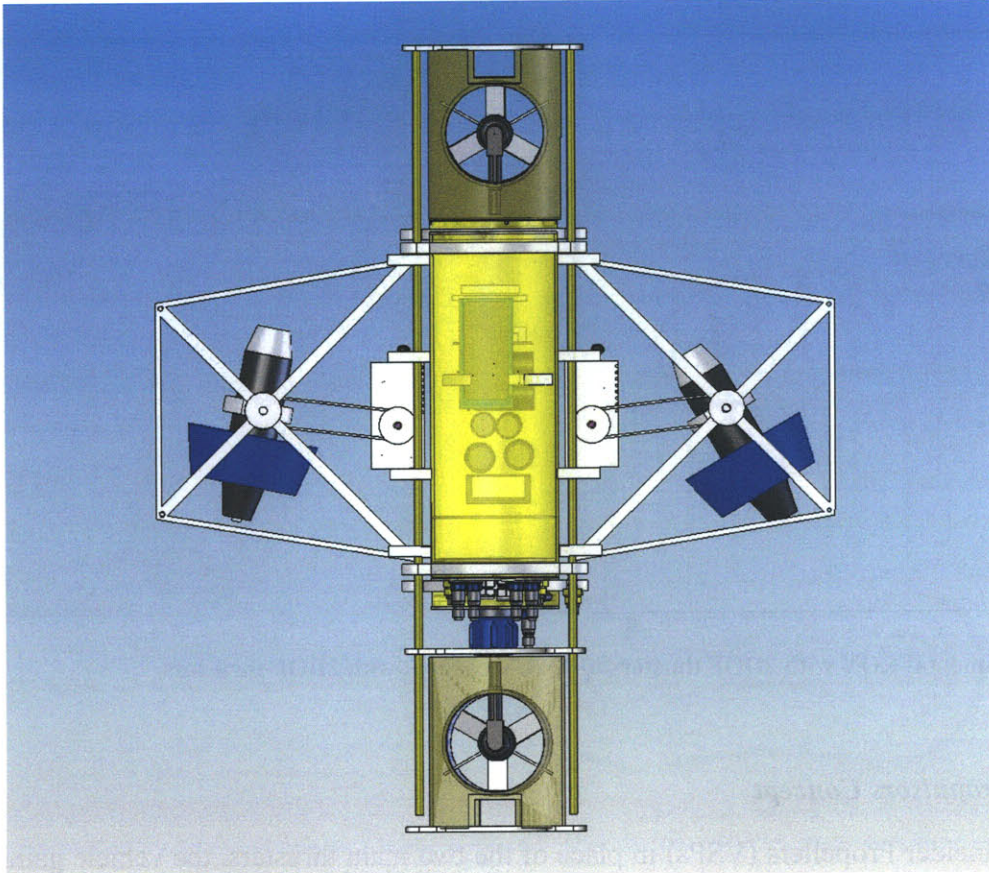


Figure 2-11: 1DOF thrusters concept

In this configuration, the vehicle is capable of operating like a tank-steer vehicle, taking the greatest advantage of the reduced drag and added mass associated with forward flight. While hovering and moving laterally, the large surface area of the sides presents a much more challenging performance. However, the improvement over a non-holonomic system in terms of ease of control and path planning may be sufficient to justify using this hovering approach at a strategic level. For low-level tactics, the hovering approach provides superior performance. Additionally, the thruster selection included provision for functioning against a broadside current. An actuated thruster system can use the main thrusters from the existing baseline configuration and still meet operational requirements.

2DOF Actuated Thrusters

The actuated thrusters are 2-DOF positioning mounts for the two large thrusters. This allows drastically increased maneuverability, at the cost of much trickier control. The actuator assembly leverages previous development of the front sensor platform. It uses a nearly identical gimbal mechanism, with minor modifications in servo layout and hydrodynamic profile. This propulsion builds on earlier work of the author in high maneuverability ROVs [28].

The hydrodynamics of the vehicle in lateral movement present the same problems as with single-degree of freedom thrusters. A thruster solution which is adequate for the 1 DOF case will be applicable to the 2 DOF case.

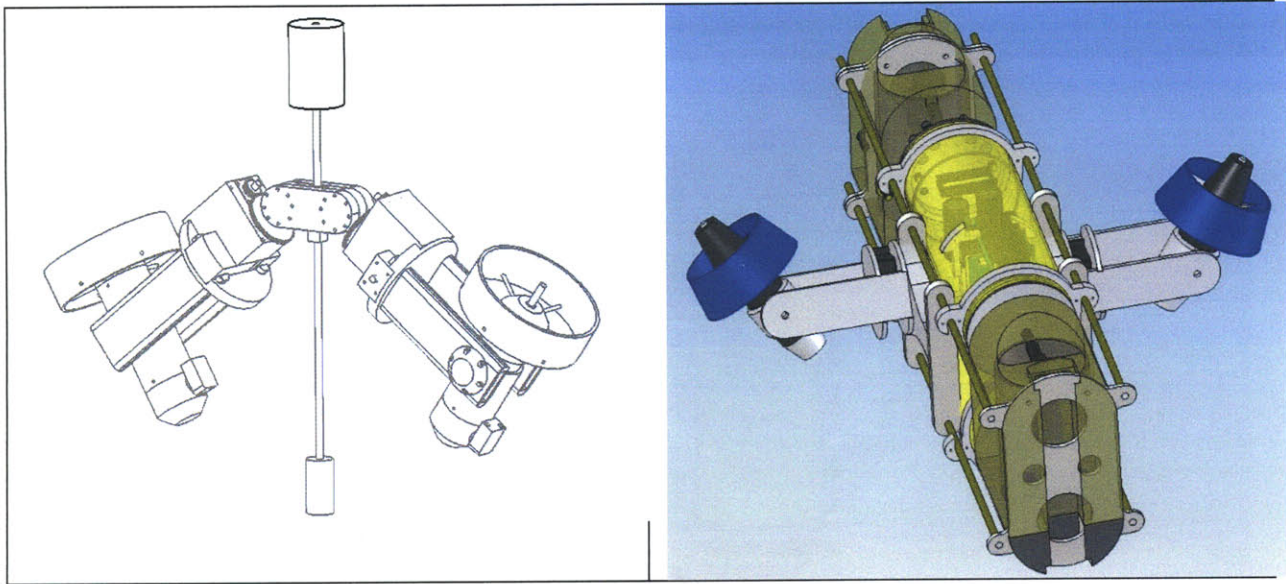


Figure 2-12: a) Experimental ROV with 2DOF thrusters b) XAUV concept with 2DOF thrusters

Voith-Schneider Propulsors Concept

By using Voith Schneider Propellers (VSPs) in place of the two main thrusters, the vehicle gains full hovering capabilities. This is a relatively common drive mechanism on high-performance tugboats, because it allows station keeping and tight maneuvering without the use of a rudder [37]. The VSP system also has the advantage over a dual z-drive arrangement of only moving the blades to change direction, rather than the full thruster unit. This allows faster direction changes for similarly powered actuators. The mechanism itself is somewhat complex, involving a set of revolving linkages to steer the blades. An entrant in the 2007 AUVSI competition used a small pair of VSPs with some success [38].

2.4 Overall Electrical design

Just as the mechanical system offered high-level opportunities for streamlining, so too does the electrical system. First, the topic of data transmission is considered. Here several protocols are considered within the context of the different modules that might be encountered. Drawing from lean thought, the methods which help to create a modular, decision delayed system are preferred over a high-performance single use technology.

2.4.1 Data transmission

Finding a suitable method to interface sensors, components, and data must be a compromise between simplicity and compatibility. Unlike the physical system architecture, where most of

the components or constructions were custom, the electrical problem must accommodate and integrate a wide selection of COTS components. If the arrangement and offering of data interfaces is too small, the system will not be able to support certain components. If, on the other hand, too many are implemented, the system will be cumbersome and may require an excess of connectors. From a lean perspective, this is a question of interface architecture: what elements of sectional, slot, and bus architecture can best be leveraged? As with the physical architecture, a pure bus or slot network is difficult to expand, while a sectional system has a high interface overhead that may preclude small components. Fortunately, there exist several options which allow adding sections of the bus elements together to create a highly scalable network.

Before the methods and strategies are developed, some notion is required of the different modules and how they might interact. Computing modules run software, do significant calculations, and are typified by the computer mainboard. Smart sensors measure some physical parameter, and have sufficient onboard communication electronics to interface directly with a computing module. These may be data-intensive, as in the case of imaging sonar, or they may not, as in the case of a serial compass. Dumb sensors provide only raw analog or digital data, and are not equipped with communication electronics. Similarly, smart actuators are capable of interfacing directly with a computation module, while dumb actuators require an additional translation stage in between. Data acquisition modules read or write analog and digital signals from these dumb sensors and actuators.

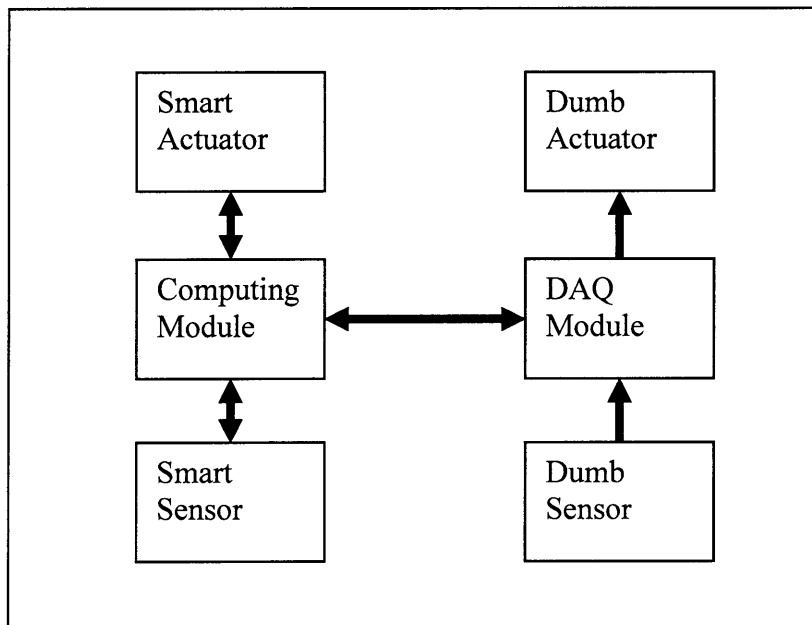


Figure 2-13: Module Relations

With this framework, some generalities are made about the nature of the interactions. Between smart actuators or low-bandwidth smart sensors and the computing module, the physical separation is moderate (still on the vehicle) and the data rates are low. Examples would include a remote r/c servo controller, an IMU, or a user input joystick. A significant proportion of add-on components meet this description; as a result, capability must be built in to allow the addition of an unspecified number of components. Communications between different computing

modules or high bandwidth smart sensors may involve distances that are: short, as within a computing module or cluster; medium, as with a scanning sonar; or long, as with a remote terminal. These connections are much smaller in number. Dumb sensor and actuator connections to DAQ/Control hardware are typically of medium distance, but susceptible to electrical noise, as these analog voltages do not leave the confines of vehicle.

A bus or clustered slot modular system seems appropriate for both smart and dumb peripherals. This means there is a single structure on the upstream component that all peripherals connects to, as opposed to a daisy-chained system where each peripheral plugs into the last one, as with a RS485 or token ring network. In the bus architecture, all connections are identical. In slot architecture, differing components may have differing connections; this would be clustered so that while all components of a same type have the same interface (bus), differing types may have differing interfaces. For a true bus system, all connections would need to be the same, impossible with both dumb and smart components, or have parallel interfaces whereby the peripheral would communicate upstream on the most appropriate one.

Depending on the communication system, the CPU-CPU or CPU-DAQ communications could be implemented as either bus or sectional modules. For example, even within a standard, there is flexibility in implementation that may allow or exclude hardware. In standard desktop computers, such as those with ATX motherboards, additional functionality is added through a row of PCI cards. This is a bus modular architecture implementation of the Peripheral Component Interface standard. It is appropriate for desktop applications where the number of extra capabilities does not exceed the number of slots provided. In a variant architecture, the PC104+ stack-based computer format uses a sectional-modular approach. Used widely in embedded systems, the PC104 system benefits from a compact form factor that expands to accommodate a wide variety of components, including power supplies, data acquisition hardware, and CPU mainboards.

Table 2-8: Widely-used data interfaces [39]

Interface	Characteristics	Examples
Ethernet	10-1000 Mb/s	Computer Networks, Imaging Sonars
Firewire	400-800MB/s	High-speed cameras
USB	1.5-480Mb/s, Serial	
PCI bus	64bit+33MHz , Parallel, Internal	PC104+ , computer cards

Looking at the larger picture, some insight is drawn on the effect of the standards on component arrangement and density. In the case of PC104, the layout of components is dense, but constrained to close physical proximity; the 120pin connection between modules is not at all suitable for cabling. Because of its hardware-level support by operating systems, the interface is, in most cases, seamless to the programmer. The caveat is that if unsupported by the OS, the hardware is very difficult to use, as in the case of xPC Target and some DAQ cards. For the control stack, the sectional modular PC104 interface offers high density. Because it is more difficult to modify than physically separate cabled components, it is well-suited for creating a larger computation cluster that is not physically modified during development. This thus

satisfies the waste analysis parameter that components are easily changed by expanding boundary box for the component.

For the large volumes of data that move between high-bandwidth smart sensors and CPU modules, as well as between multiple CPUs, Ethernet is a convenient data exchange interface. First and non-trivially, most imaging sonar already have Ethernet connections. By using a central Ethernet hub, independent modules may be simply added or removed. Although a bus architecture system, the base hubs can be daisy-chained together in a sectional arrangement to create unlimited module capacity. This trivializes increasing the module count beyond the initial capacity. In the same general trend, this expansion capability allows the vehicle access to much larger Local Area Networks, as well as remote connections through the internet. Because the standard is well-established for such applications, only limited developer resources are required.

For smart components, a bus system is ideal. If each component has the communications brainpower to interface directly, they might as well be the same. The Universal Serial Bus combines a wide array of off-the shelf hardware with reasonable data rates. Additionally, COTS serial adapters allow any device capable of generating asynchronous serial output (e.g. RS232 or similar) to connect via USB. This greatly reduces the complexity of sensor-side communications hardware. Much like Ethernet, USB has the ability to chain hubs together to increase component count. This offers significant benefit by reducing the need for hull penetrations and connectors.

Hub-free networks

Although Ethernet and USB offer many benefits, both require hubs for interconnection between more than two devices. Three multi-drop protocols are considered, each allowing multiple peripheral devices to connect directly to the same. First, the I2C bus is a two-wire master-multi slave network that allows one main device to talk to many connected devices. Although originally designed for short-range communications between integrated circuits, I2C has been successfully implemented as a robot control network between larger, separate modules, as in EPFL's salamander robot [40]. Unfortunately, the master-slave arrangement prevents multiple slave devices from streaming data without complicated buffering schemes. This software would serve the same purpose as the hub in a USB or Ethernet system. Few devices use the I2C protocol for external communications, so an implementation would require extensive adaptation or custom development. As a result, I2C is best left for communications within an embedded module, or in a wider network where smaller packets of command and data are transmitted.

Next, RS485 is an industrial serial protocol. While similar to RS232 serial, it has two main advantages. First, it uses one differential voltage over a pair of wires for transmit and receive, giving it highly noise immune performance (a second pair can be added to allow simultaneous transmit and receive, making the variant RS422). Second, it also is a multi-slave, allowing multiple simultaneous connections to the bus. Although RS485 industrial equipment is common, inexpensive commercial equipment is much more likely to use RS232, USB or Ethernet. This lack of native support for modules makes it unsuitable for a main network backbone.

Last, the CAN bus is a bus designed for communications between modules in an automobile. Without any one master, the bus operates somewhat similar to a MOOS environment, where each module sends out data, and any interested modules listen in. Although relatively slow, and

unsuitable for high volume streaming sensor data, the system is elegant in its use of a priority system to control data flow. Each device is on equal footing, but decides how important its transmission is prior to sending. If two devices begin talking at the same time, the lower priority transmission shuts itself off automatically. This protocol is best suited for a system with many independent modules, instead of a powerful master computer. Because the XAUV is using a master computer to simplify control adjustment and development, the added benefit from the CAN bus is much less.

In summary, the problem of interfaces is simplified by where the boxes around subsystems are drawn. Between smart sensors and actuators and a computing module, either USB or Ethernet is preferred, depending on the data rate required. For dumb sensors and actuators, the immediately proximal connection must be in its native format. By including an interface module such as a DAQ or USB/Ethernet converter, the range of interfaces required of the central computer is drastically reduced. By building and connecting these components with bus architecture system, the number and location of components is easily modified. Through the use of hubs as bus extenders, the typical limitation of a fixed-sized bus is bypassed, and a flexible, scalable system is created.

2.4.2 Power System

The propulsion system of the vehicle presents a difficult challenge for the main power system. With a total maximum draw of 2.6 kW, many off-the-shelf batteries proved inadequate. Keeping with the functional requirements of a scalable system, the power system was attacked from the side of power first, then energy. In other words, the smallest battery possible is used to allow full operation; then additional batteries can be added for energy to increase mission time. Initial parameters for the battery system are tabulated below. The energy calculations are based on 40-50% duty cycle operation, for approximately one hour.

Table 2-9: XAUV Power Requirements

Parameter	Value	Unit
System Voltage	48	V
Peak System Draw	50	A
Typical Draw	20	A
Peak System Power	2.5	kW
System Energy	>1-2	kJ

The high power draw and relatively short required mission time suggests that discharge rates of 1C+ maybe required. This rate is important with the consideration of different battery technologies, as certain batteries, especially standard Li-ion are not capable of continuous 1C+ discharges. Standard secondary (rechargeable) batteries are tabulated below with relevant parameters. All parameters are derived from a survey of actually available cells. Cost and weight metrics are based on raw cells (not including secondary packaging, protection or charging circuitry).

Table 2-10: Cell Chemistry

Chemistry	Abbrev.	Cell Voltage	Discharge Rate	Whr/Lb	lb/in ³
Lead Acid	Pb	2	10C	15	0.19
Nickel-Cadmium	NiCd	1.2	5C	20	0.15
Nickel Metal Hydride	NiMH	1.2	3-10C	30	0.36
Lithium-ion	Li-ion	3.7	1C	80	0.30
Lithium Polymer – Low Power	Li-po	3.7	1C	90	0.08
Lithium Polymer – High Power	Li-po	3.7	5-10C	95	0.12
A123 Li-ion	aLi-ion	3.3	30C+	70	

In order to keep the system weight as low as possible, heavier chemistries such as Pb, NiCd, and NiMH are avoided. NiMH is considered as a backup, as the lightest of the heavy-duty, easily rechargeable chemistries. Lithium batteries offers 2-3x weight savings over more traditional chemistries. Unfortunately, most Li-ion and Li-po batteries are difficult to charge and maintain, requiring specialized monitoring and protection circuitry. Additionally, the maximum current draw from the most inexpensive laptop 18650 cells is limited to approximately 1C. Ideally the draw would be closer to 0.5C to maximize the battery life. Higher power Li-po cells were seen to be prohibitively expensive in the volume required. Alternatively, A123 Li-ion cells offered lower energy density than standard Li-ion, but higher power output than even lead-acid. Unlike most lithium batteries, the A123 cells are safe from explosion and fire, even when punctured or shorted. Additionally, the A123 cells can be charged using only a simple current limited power supply.

From this evaluation of the field, the two main options are standard 18650 Li-ion cells and A123 Li-ion cells. In order to generate the required power using 18650s, over 350 cells would have to be acquired and assembled into a battery. Otherwise, the system would draw more current than the batteries could safely give. For the A123 system, only a single set of 15 cells is required. Although the mission lifetime with this arrangement is very short, for the purposes of this system, it was much more valuable to have a working system with short life, that could then be upgraded by adding more cells in parallel. The A123 system is better in terms of simplicity, scalability, and ease of prototyping. In lean terms, for an equivalent eventual system, the solution avoids unnecessary overwork at the beginning, and puts off mission lifetime work until that phase in the development.

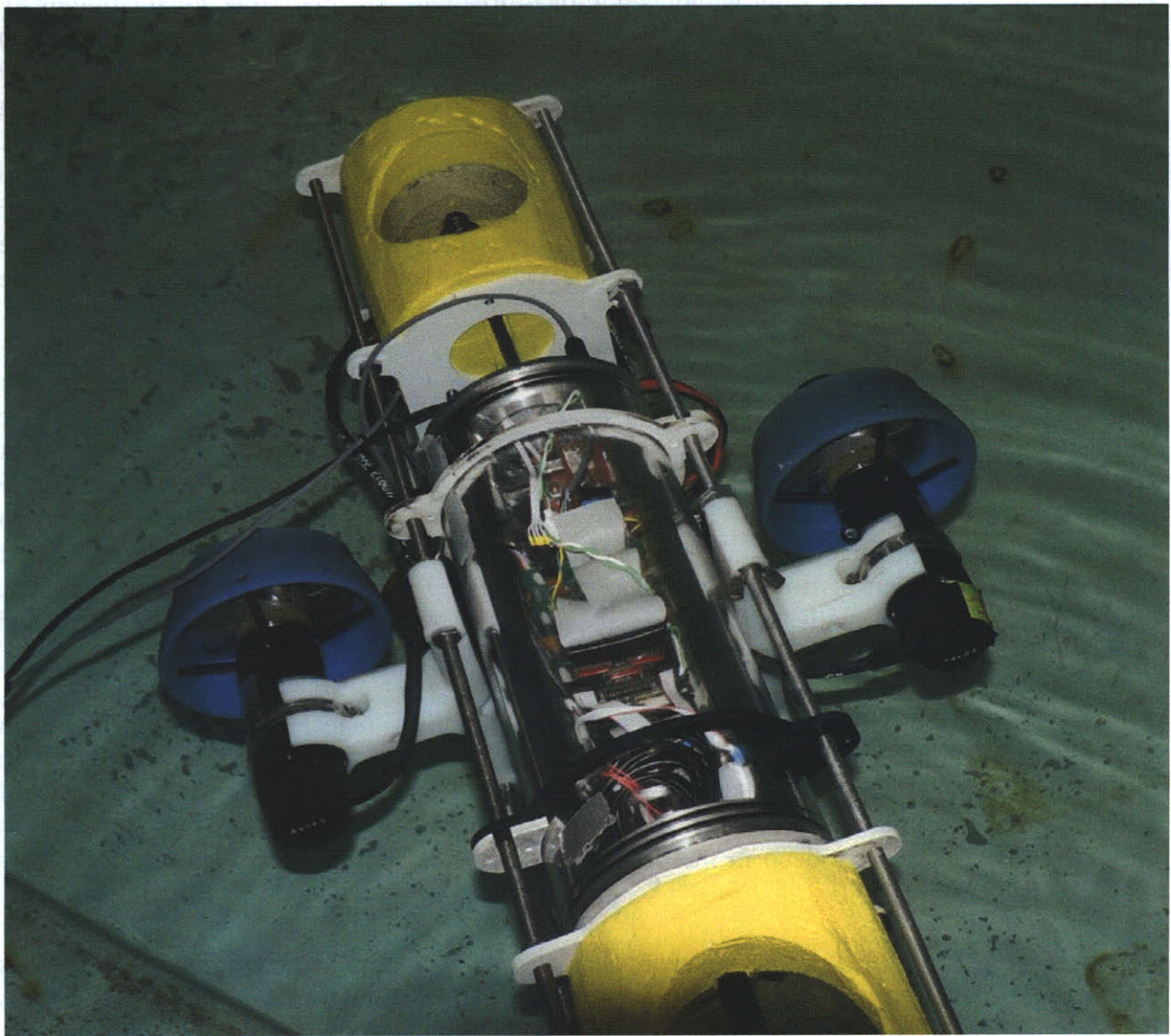
2.5 Summary

An initial exploration of lean concepts has provided a framework of a highly mobile development payload, moving from simulation to hardware. In evaluating these concepts, the

notion of delaying differentiation was selected as a main tool in minimizing the work that must be redone between steps. The essence of this concept is that any common elements between different implementations should be captured in a single module, while only those things that are different should be included in the modules that are different. Although seemingly common sense, this insight proves useful in later analyses. Next the overall mechanical design is considered, with a focus on the physical layout, including thrusters, coarse hydrodynamics, and physical architecture. A relatively standard vehicle layout is selected, with the understanding that the design will facilitate additional performance components. By selecting a slot/bus architecture for the vehicle, some performance and a lot of hydrodynamics are sacrificed for easy integration of a wide array of components. Once the overall geometry is identified, a series of modular components is considered which increase the scope of operation or performance of the vehicle. Next, the electrical system is considered, looking at the signal architecture through the same lens as was used on the physical architecture. From this, two extendible-bus architectures for high and low data rate transmission are selected, specifically USB and Ethernet. This also has the effect of encapsulating any dumb sensors or actuators into modules with converters or DAQs that can speak one of the two protocols. Last, the power system is considered, by looking at the potential batteries from the perspective of the implementation order. The chemistry selected allows the full performance of the vehicle for a short period of time with its minimal implementation.

Chapter 3

Specifics and Implementation



3.1 Mechanical Details

After the overall architecture and geometry of the vehicle has been decided, several subsystems must be developed. The flotation for the vehicle will ensure that the vehicle remains properly oriented during operation, and that it will rise to the surface in the event of a system failure. A material and method for casting the flotation is summarized. Next the thruster mounts are designed, with some first-order calculations to validate their performance. Last, the pressure housings are discussed, eventually settling on 1 atmosphere pressure vessels.

3.1.1 Flotation

Although the two main pressure housings and other components displace a large volume, the vehicle is still negatively buoyant without additional flotation. The main functional requirement of the flotation is to displace as much water as possible with minimum weight or modification to the vehicle shape. Additionally, the flotation is used as the primary failsafe or recovery system. Even if all systems fail, a positively buoyant vehicle will rise to the surface, allowing recovery. A fully failsafe system would return to the surface even if all watertight housings were compromised. Unfortunately, this additional buoyancy would require an excessive downward thrust to overcome during mission operation. Although systems such as dissolving links for ballast, or remotely triggered drop weights could be used to offset the additional flotation, the XAUV will be positively buoyant only in the case that neither of the main housings are compromised.

Several options present themselves for flotation. First is to use hollow objects, such as tubes or spheres. These can be designed to go very deep, as in the case of glass or ceramic spheres, while still remaining relatively lightweight. The factors of wall thickness and diameter have significant impact on weight, buoyancy, ruggedness, and ease of implementation. For resistance to breakage, from depth or impact, a thicker wall or sturdier material is desired. This is likely to increase the weight however, forcing the use of a larger diameter object. Similarly, for the most flexibility on the layout of flotation, small overall diameter is desirable. Smaller overall size, however, will require a smaller wall thickness to give the desired buoyancy. Also, too small, and the flotation objects will be difficult to mount.

Syntactic foam takes desirable attributes from the hollow objects, and puts them into a geometrically flexible form. This foam takes tiny glass balloons, and holds them in place with an epoxy matrix. In this manner, the foam can be cast or machined into a wide array of geometries. Unfortunately, it also is relatively heavy for its buoyancy. While the microspheres allow the foam to go very deep in the ocean, for shallower applications, a simpler foam may suffice.

Polyurethane casting foam offers the advantages of a bulk flotation, where complex geometries can be cast or machined, without damaging the pressure tolerance. Instead of the glass spheres of syntactic foam, polyurethane foam simply has air bubbles trapped inside a polyurethane matrix. It is then the properties of the matrix, rather than the bubbles that determine the pressure

rating. The more bubbles, the more buoyancy; however, this also means that there is less matrix, and will be less resistant to crushing.

Table 3-1: Potential flotation trade-offs

Flotation	Pro	Con
Syntactic Foam	- Structural - no crush risk	- Heavy (sg. ~0.5) - Expensive
Casting Foam	- very light (sg. ~0.12) - Inexpensive	- Possible Crush risk - may be non-structural
Hollow Glass/Plastic Spheres/Tubes	- No crush risk - inexpensive - structural	- limited flexibility - reduced effectiveness for small spaces

In order to confirm the feasibility of using polyurethane foam, a test piece was cast and sawn to break through the outer skin and expose the inner matrix. It was subjected to immersion in water for one week at elevated pressure (equivalent to 10ft), as well as deeper water equivalent (100ft) for over an hour. The piece showed a slight increase in weight, but no change in displaced volume. The weight increase may be explainable by surface wetting of the sample. After prolonged exposure in this environment, in excess of two months, no significant degradation was evident.

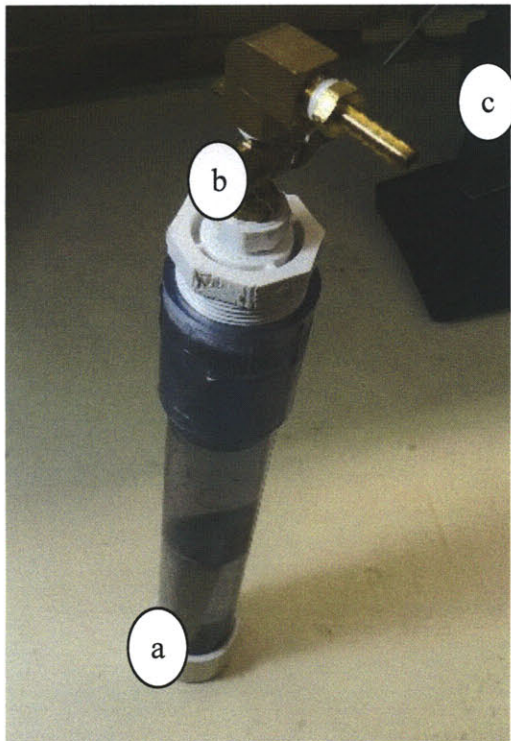


Figure 3-1: Foam pressure testing chamber. (a) main pressure chamber, filled with water and foam chunk. (b) valve to maintain pressure, (c) barb fitting for pump. While under pressure, the chamber was housed in a secondary pipe for safety, in the event of catastrophic failure of the chamber.

The PolyTek R-8 foam selected has reduced carcinogenic and sensitizing agents when compared to other similar polyurethane foams.

The initial approach to foam casting used built-up polystyrene molds for rapid, lightweight buildup. The basic approach was to manually cut out forms from expanded polystyrene sheets, as can be found for insulation. The result is a large, lightweight mold. This had limitations in the complexity of shape and resistance to packing pressure from the foam expansion. Although an automated foam cutter/hot wire could make this process feasible, an alternate approach was developed. This preferred method uses laser-cut foam core to assemble an expendable mold negative. The pieces lock together through castellations and tabs, and eliminate the need for adhesives – an interference fit of 0.005” still allows easily hand assembly in the relatively soft material. Although the board can be bent into curved shapes, they are approximated in this implementation by several small flat pieces. The gaps between the segments can be taped to prevent leakage, although this was generally not necessary. In large flat areas, some reinforcement may be useful in preventing bulges; this can be in the form of additional layers, spars, or a well-placed book.

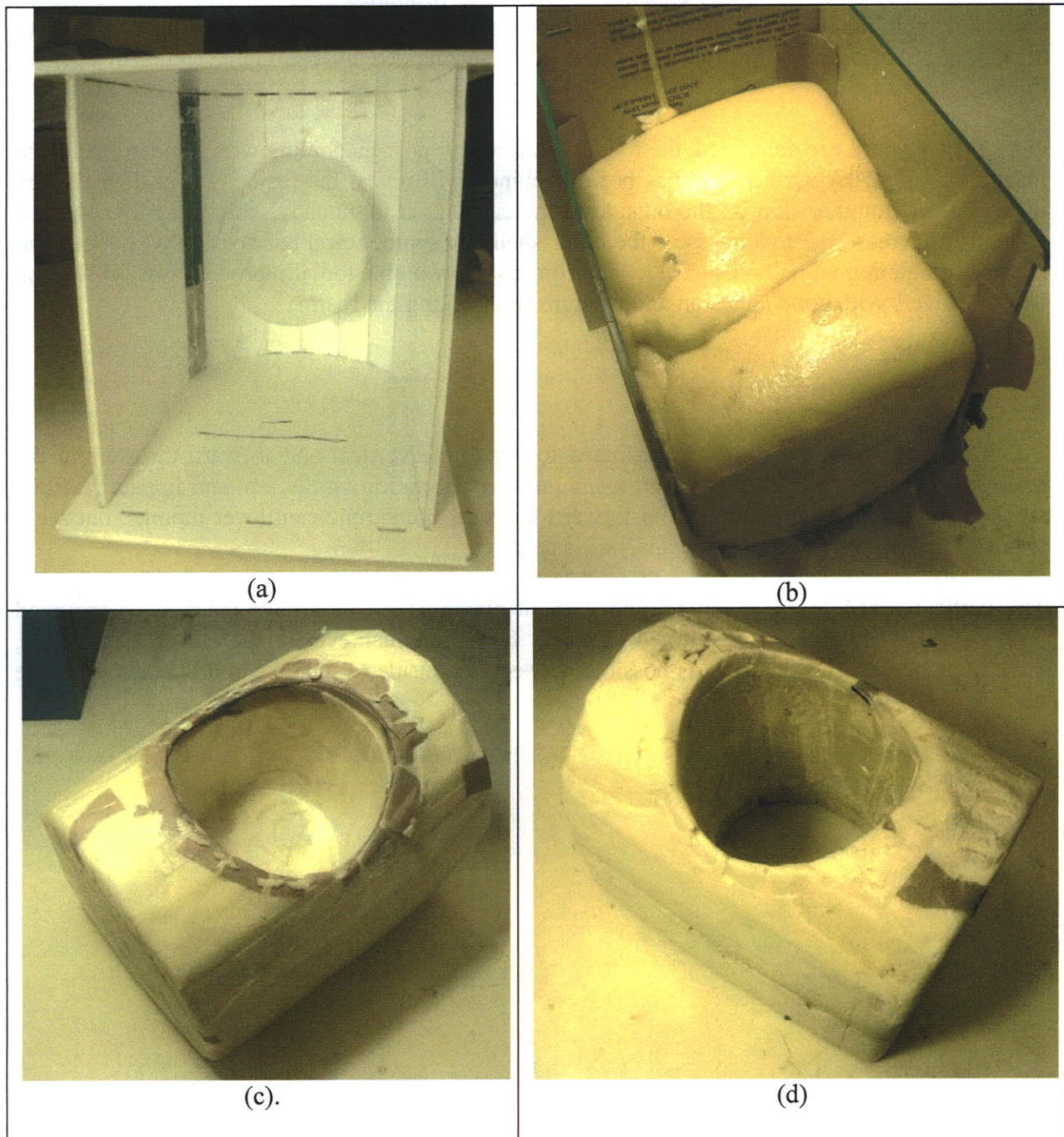


Figure 3-2: Foam casting process. (a) empty mold; (b) as cast; (c) demolded; (d) cored and prepped for painting

Complex geometries can easily be reproduced by inserting a negative. Because the mold is destroyed when the part is removed, no consideration is required for side pulls. Alternatively, objects can be inserted into the mold to reduce the complexity of post-machining. For the main tunnel channel, a plastic cup was inserted into the mold; this reduced the volume of foam that had to be later removed by a boring bar. Because of the size of the resulting part, it was decided not to insert mold the tunnel tube into place. However, this is a valuable technique that should

be considered. When the foam expands in the mold, it applies pressure against any enclosed insert, and will hold the insert in through considerable friction. . Additionally, if incuts or bosses are added to the insert, they will be filled in or surrounded by the foam, adding a positive interlock to the hold.

Once the cast parts are demolded, the foam is suitable for post-machining with standard hand and machine tools. Any surface that has been machined will reveal the inside matrix of the foam, instead of the tougher skin on the outside of the casting. It is this advisable to apply a surface coating to increase the toughness of the outer shell. A synthetic rubber coating was applied to the main floatation with good success. The commonly available tool dip product was brushed on in a few layers with good adhesion and no signs of attacking the foam.

3.1.2 Thruster mounts

The thruster mounts are the structures used to attach the vertical and forward thrusters to the main frame. The vertical thrusters are combined with flotation on the top, and impact resistant striking surfaces on the bottom. The forward thrusters are simple cantilever mounts, but see a significant load. As a result, some analysis is required.

The vertical thrusters are manufactured without cowlings. This allows them to be mounted inside the hull of a vehicle in a tube or tunnel. In order to achieve as much thrust as possible, there must be as little clearance as possible between the inside of the tube and the outside of the blades.

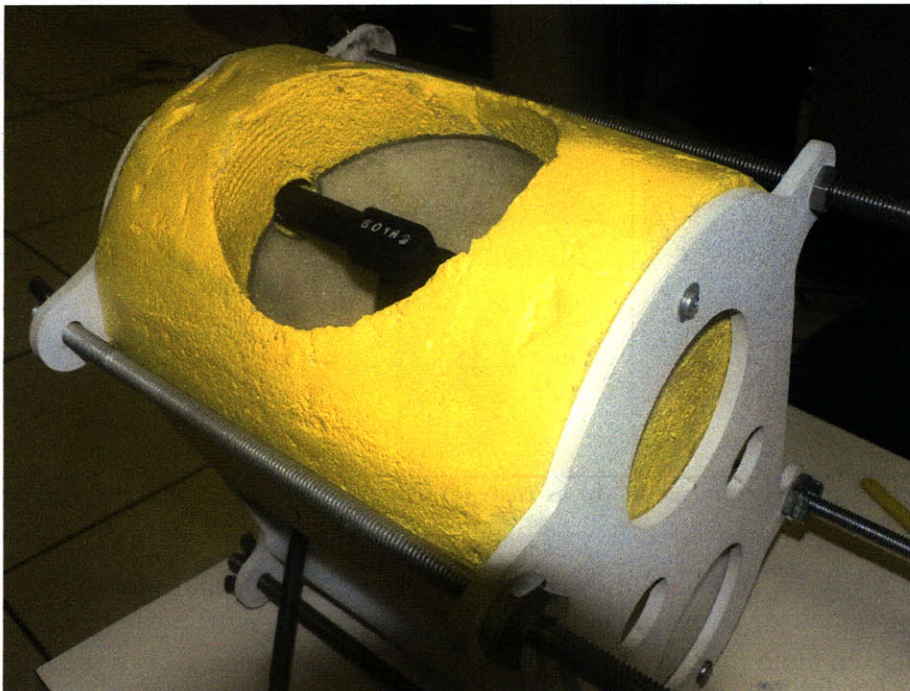


Figure 3-3: Tunnel thruster mount assembly

The fixed forward thruster mounts are bulk-machined UHMW for light weight and superior impact resistance. Each one connects to both of the main structural rails on the particular side. Assuming the loads are small relative to the ultimate tensile strength of the material, the primary concern is the deflection of the components. UHMW is exceptionally resistant to fracture; although difficult while machining, this property makes it suitable for low-stiffness mounts likely to take shock loads or other abuse.

Looking at the geometry, the problem can be decomposed into two primary modes of deformation. The first is the simple cantilever beam bending of the main strut, while the second is the torsional deflection of the support from the rail mounts to the main strut. From basic mechanics, the deflection at the end of a cantilever is [41]:

$$x_{deflection} = \frac{FL^3}{3EI}$$

Where F is the load force from the thruster, L is the length of the beam, E is the Young's modulus for the material, and I is the moment of inertial for the beam geometry. Similarly, the deflection angle is approximated as

$$\theta_{deflection} = \frac{FL^2}{2EI}$$

For a modulus of 2 GPa, load of 200N, area moment of $\frac{bh^3}{12} = 3.5 \times 10^{-8} m^4$, and a length of approximately 6", the deflection is approximately 3mm and the angle is approximately 0.03 rad, or 1.8 degrees.

Looking at the torsional component, the system can be broken into two parallel springs, with a driving moment equal to the load force multiplied by the strut length. The angular deflection ϕ is a function of the torque T , length of torsion member l , shear modulus G , and moment of inertia J .

$$\phi = \frac{Tl}{GJ}$$

Where $T = 200N * 0.15m = 30Nm$, $l = 0.1m$, $J = 5.1 \times 10^{-4} m^4$, and $G = 0.12GPa$ for each of the two arms, the approximate deflection is less than 0.001° . Thus the primary contributor to deflection is the cantilevered component. An FEA simulation was run on the actual part geometry, and computed a 1.4mm linear deflection and a 1° angular deflection.

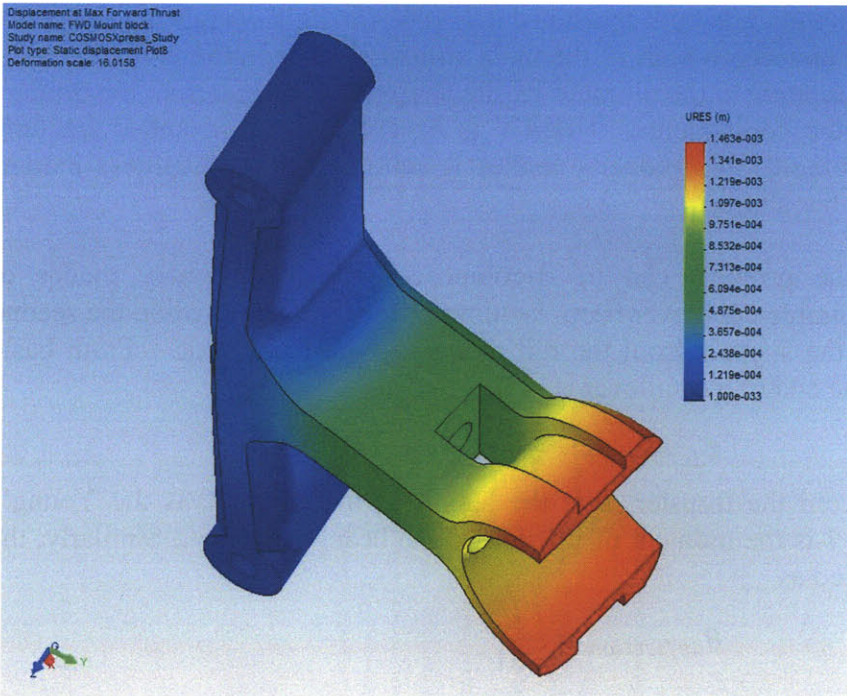


Figure 3-4: Finite element deflection analysis on fixed thruster mount. (exaggerated in rendering) The maximum displacement is 1.4mm, resulting in an angle of about 1° toward the vehicle midplane.

Working backwards, it is possible to determine the maximum allowable length of the main strut before some performance criteria is not met. For now we will assume a 5° deflection before a redesign is required. The length of strut is relevant because it determines the control moment for turning of the vehicle, and determines the balance between overall size and likelihood of the thruster water jet reattaching to the hull. Rearranging the cantilever angular gives

$$L = \sqrt{\frac{2EI\theta}{F}}$$

Using the same constants as before, the maximum length is 0.25m or 10". In summary, the design of the thruster mounts is adequate for the expected loads. Additionally, the material selection will help reduce the damage caused by an impact.

3.1.3 Pressure Housings

The pressure housings keep water and contaminants from sensitive electronics inside the vehicle. While most components would survive a the mission pressure of 50psi, few would survive being immersed in saltwater, even at atmospheric pressure. In designing the pressure housings, the outer diameter is a major design parameter when considering the overall structure, having an impact on both the internal carrying capacity, but also the added displacement and required frame geometry.

The main computer housing must hold the computer stack. Because the computer stack is based off a PC104 form factor, the minimum inside diameter is 5.2" This comfortably fits within a 6" OD tube. The upper bound on pressure housing size is determined by availability of components

and machining facilities; components larger than 8" become very difficult to source. Additionally, because of in-house machining constraints, 6" is much preferred. This presents a significant buoyancy tradeoff due to the doubling in volume between 6" and 8"; however, the projected volume for a 6" system is sufficient to require only a small amount of additional buoyancy. In this case, the 8" system may actually require additional ballast, thereby increasing overall system weight.

The primary concern for the housings is that they are completely watertight. Since one contains the delicate computer and control system, and the other contains a battery capable of a 90kW dead short, even a small leak could prove disastrous. As a result, the system uses plug-type o-ring seals instead of face seals. Face seals are typically constrained by a bolt circle or thread. Threading is impractical at this diameter. Plug seals use the normal force between the plug and socket to compress the o-ring. No user-applied force is required to maintain the seal preload [42].

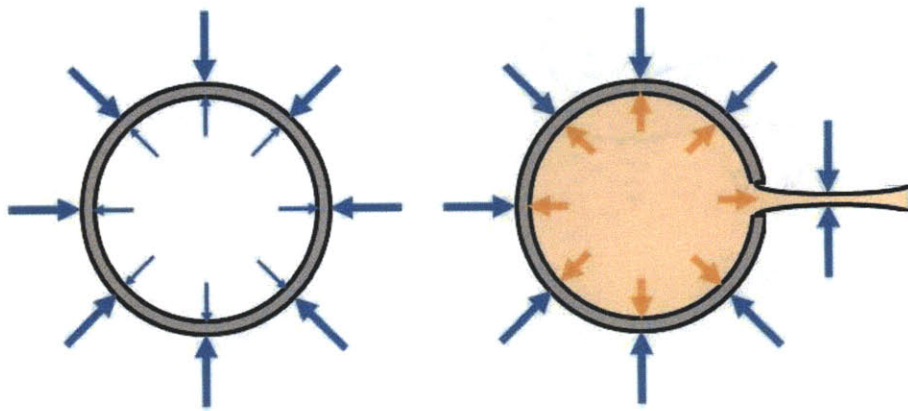


Figure 3-5: Pressure tolerant housings. Left: atmospheric housing. Right: oil pressure compensated.

In order to protect the delicate components from water, the housings must resist the pressure pushing water past the seals. For deep operation, crushing or cracking of components is a concern. For the depths that the XAUV is to operate, this is not the primary concern. The primary concern is preventing leaks, thereby preventing electrical shorts.

The simplest configuration uses a water- and airtight container, sealed at one atmosphere. As the housing moves deeper, an increasing pressure pushes against the container and its seals. If the pressure becomes too great, the seals will fail, and water will penetrate. An alternate configuration uses an oil-filled housing with a flexible member exposed to the surrounding pressure. If no air is present in the housing, the external pressure squeezes the flexible member. Because the oil is incompressible, no actual displacement of fluid is required to pressurize the housing. If some air is trapped inside, the oil will transfer the pressure to the air, compressing it. Through this pressurization process, the difference in pressure between the inside and outside of the housing is very small. As a result, there is very little driving force with which water would penetrate seals or break housings. As an additional benefit, a suitable choice of oil can either repel water, providing an additional source of resistance, or absorb and encapsulate leakage water. Air-tool oils typically have these absorbent additives to protect pneumatic equipment from rust. A hybrid atmospheric-oil system could be envisioned, where the oil serves as a

displacement barrier, preventing a certain total volume of water ingress simply by occupying the space inside the housing. Suffice it to say, any approach using gallons of oil has tradeoffs in increased weight, and outright mess.

Both the battery and computer housings use atmospheric housings filled with air. Each housing consists of a long polycarbonate tube, and two aluminum endcaps. The endcaps have a double o-ring plug seal. Properly designed O-ring seals can tolerate pressures to 5000psi [42], or a depth of 3500 meters. At this depth and pressure, deformation of the housings may cause seal leaks by altering the relative dimensions surrounding the o-ring. Additionally, this pressure is sufficient to crush the XAUV housing tube. Therefore, the o-ring seal is not considered a high-risk element at 50-100psi.

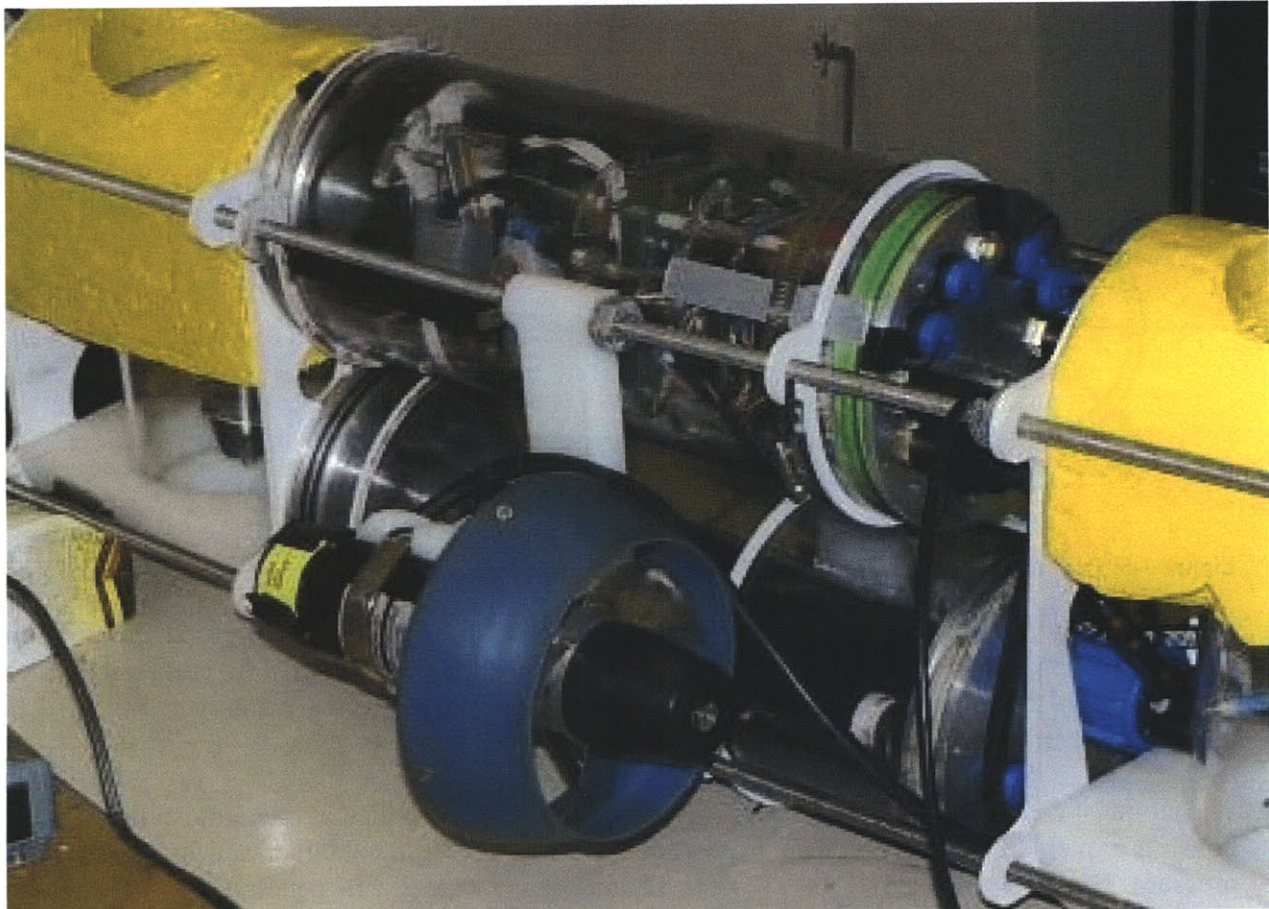


Figure 3-6: Computer housing (top) and battery housing (bottom). Both use simple double o-ring aluminum plugs inside polycarbonate tubes. This allows the housing to be lengthened by replacing the tube only.

Main Junction Box

An early design used a separate main junction box, attached to the end of the computer housing. This is an oil-filled module that allows the operator to modify or add electrical and control connections without modifying the computer housing. This minimizes the risk of flooding the computer housing by limiting the number of times the computer housing is opened, allowing it to

stay sealed and safe. This type of approach was successfully implemented in the MBARI Tiburon's multiplexer modules [43].

The final design removed this feature in order to limit the possibility of an oil leak, while increasing the overall simplicity of the design. The tradeoff is that any future hull penetrations must be made on the main housing, rather than on a more easily replaced junction box flange.

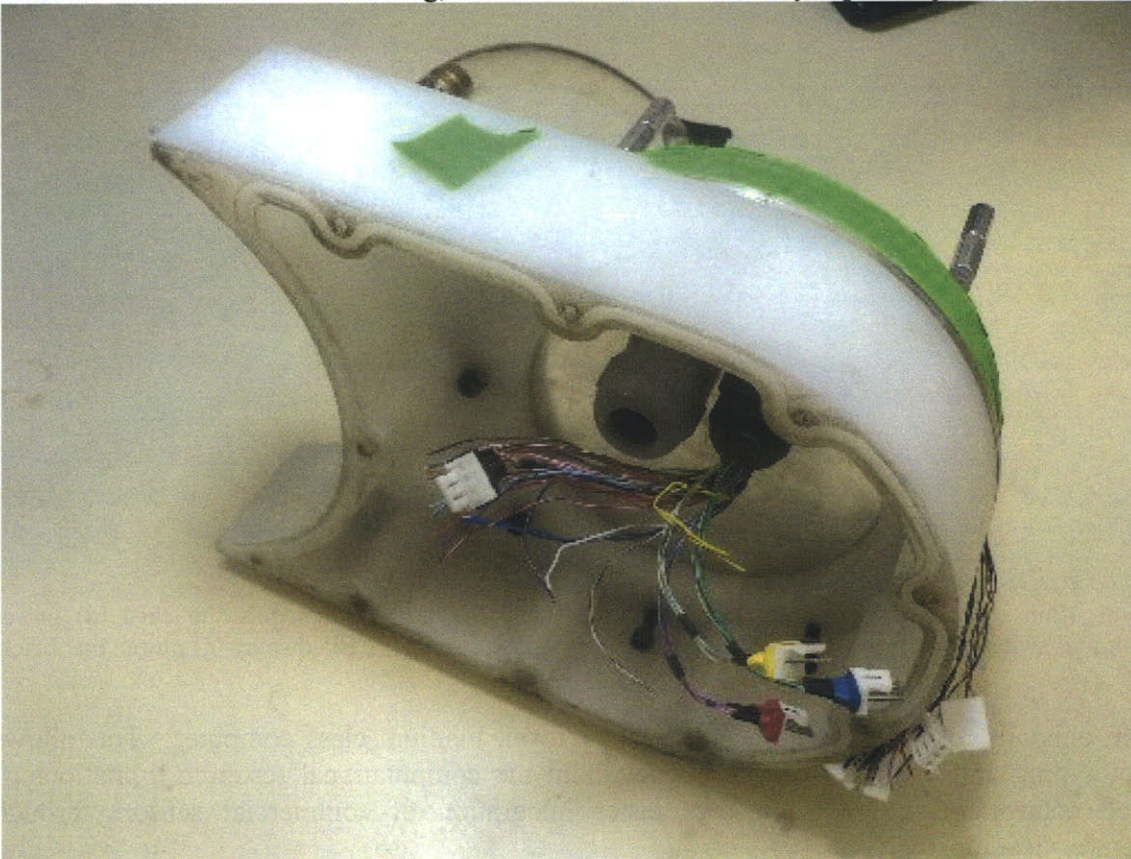


Figure 3-7: Main Junction Box

3.2 Electronic Details

If the mechanical system is the muscle and skeleton of the vehicle, the electronic system functions as the nervous and circulatory system for the robot, generating and controlling the flow of power and information. The various modules take measurements, conduct calculations, store information, control thrusters, and interface with a user.

3.2.1 Computer Stack

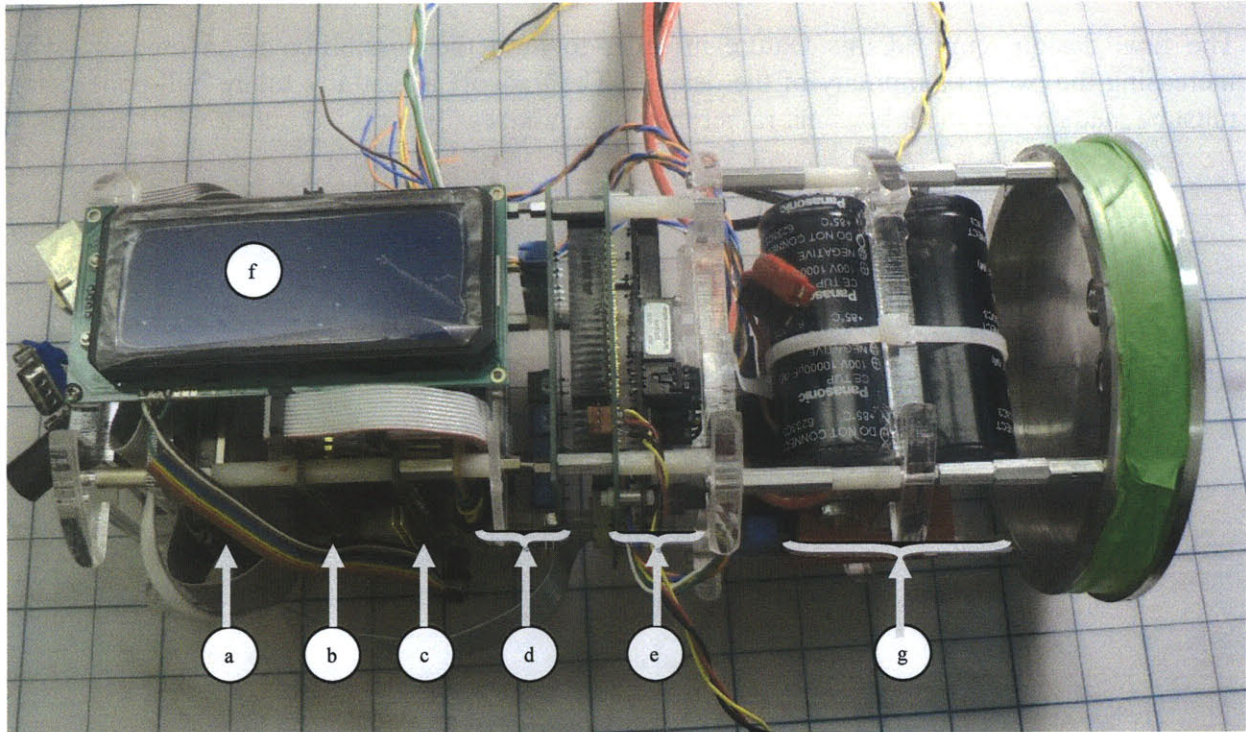


Figure 3-8: Main electronics Stack: (a) hard drive (b) CPU card (c) Data acquisition card (d) power conversion and switching cards (e) sensor interface and isolation cards (f) LCD display (g) power bus brick.

The main onboard control system uses a PC104+ form, Pentium class computer. This allows easy prototyping and modification of code, without the computational resource limitations of embedded controllers. It also allows faster integration of commercial sensors through manufacturer-developed drivers.

Table 3-2: Computer Specifications

<i>Component</i>	<i>Spec</i>
CPU	Pentium III 700MHz
RAM	500Mb
Storage	60Gb Laptop HDD, 8Gb Compact Flash card
A/D	8ch 12bit (Multiplexed) 100kHz
D/A	4ch 12bit
DIO	3 x 8bit ports
Comms	2 USB 1.1, 2 Serial, 1 Ethernet, 802.11b/g

The design attempts to maximize flexibility through the use of USB components. By using USB compatible components, the configuration can rapidly be changed between development and execution. For example, the additional IDE cabling of an internal CD-ROM drive is

eliminated by using a USB CD-ROM. Furthermore, the devices do not require a modification of the control stack; the devices are simply added to a USB hub. Components using a traditional serial connection are put on the USB through a USB-RS232 converter. This allows them to be integrated into the system from any of the hubs on the vehicle.

Table 3-3: Control System Peripherals

<i>Component</i>	<i>Bus</i>	<i>Mode</i>
DAQ	PCI	All
Hard Disk	IDE	All
CD-ROM	USB	Development
Floppy Drive	USB	Development
Keyboard	USB	Development
Mouse	USB	Development
Wireless Joystick	USB	Operation
Wireless Ethernet (802.11b/g)	USB	All
Flash Storage (8Gb CF card)	USB	All
GPS (through converter)	USB	Operation
Servo Controller (through converter)	USB	Operation
Laser Scanner	USB	Operation
IMU	RS232	Operation
Mini LCD Display	RS232	All
Blazed Array	Ethernet	Operation

If modularity becomes an even higher priority, or replacement components are sought, several additional components can be adapted to USB. The hard drive can be adapted through a common external hard drive enclosure. The DAQ system can also be replaced with a USB module. National Instruments has several that have direct driver support with MATLAB [44]. Last, the Blazed Array has options for direct USB interface [3]. In order to handle the significantly increased data moving through the bus, an additional USB 2.0 card PCI may be installed in the computer stack.

3.2.2 Interconnects

In order to achieve the desired flexibility in the system, careful design and layout of connectors is required. Typical small underwater vehicles sport a wide array of high-cost connectors, perhaps a result of interfacing to different modules, each with different interfaces. In order to make the vehicle as hardware transparent as possible, a standardized USB-based connector is used as much as possible. In order to develop the size and types needed, a set of requirements must be developed. Below, some potential external modules are tabulated, with power and communication requirements.

Table 3-4: Devices, power, and signal requirements for connectors

Device	Number	Power Requirements	Conductors	Notes
Thrusters	4	48V 20A	4 Isolated + 2 Power	
Blazed Array	1	24V <1A	Ethernet*	USB models exist
Scanning Sonar	1	24V <1A	2 RS485 + 2 Power	
Servos	2-8	5-7V <10A ea	1 PWM signal + 2 Power	USB adaptors exist
Webcam	1	5V <100ma	4 signal	Only USB
Battery	1	48V 50A	2 Power, 0-8 signal	Signal for battery monitoring, option for intelligent (serial) monitor

Looking at the power requirements as a separating factor, several different solutions may be required. Although all connectors could be standardized to handle the full power, this would require an excessive space. Alternatively, smaller connectors could be put in parallel, spreading a large current over many contacts on several connectors. This does introduce the possibility of hooking up a power connector to a signal circuit. Looking back to the lean concepts, a fool-proof system (poka-yoke) has as many standardized parts as possible, but uses physically incompatible parts for different functions [45]. Taking this approach, the battery and thrusters will have separate connectors from small signal connectors.

The main power connector is a right-angle modified connector with 4x 32A contacts. Using two each for positive and negative connections, the total current carrying capacity for the connector is 64A, well above the nominal max power draw. The thrusters keep their industry standard MCIL-6 connectors, a wet-mate connector by Subconn. The signal connectors are from the Bulgin Buccaneer 400 series connector. For USB and aux power, a six pin connector is used. For Ethernet, an eight pin connector may be used. As before, this prevents incorrectly hooking up connectors. Although intelligent peripherals are desired, any additional ‘dumb’ components such as deadman switches should have a different pin count connector where possible.

3.2.3 Isolation Board

The thrusters required an isolated power supply and analog control output for each. Because of the high powers involved, the potential for damage from ground loops was high. As a result, a custom power and analog isolation board was constructed. The power supplies are common off the shelf isolated power supplies.

The analog isolator uses the HCN2000 opto-isolator. Digital opto-isolation is straightforward, relying on the presence or absence of light from a LED to activate or deactivate a photodiode or phototransistor. As a result, most of the nonlinearities involved can be ignored. In contrast, linear opto-isolation faces non-linear light output from the LED, non-linear electrical output from the transistor and diode, and non constant parameters for both. HP provides an elegant

solution in its datasheet [46]. A single LED shines on two identical photodiodes. The output from one of the diodes is fed back, linearizing the output. Because the two diodes are identical, the output from the other diode is also now linearized, and electrically isolated. The figure below illustrates the topology from a control perspective.

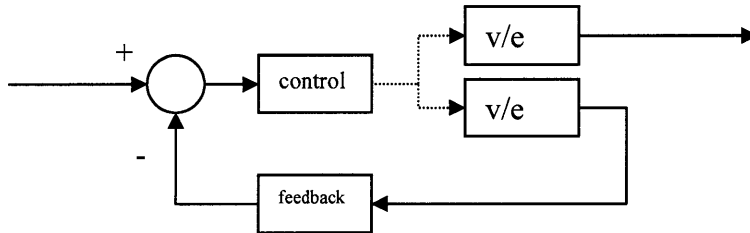


Figure 3-9: Analog Isolator Topology. Two identical photodiodes allow linearization with one diode and electrical isolation with the other.

The measured output is linear up to 100Hz. Above that and up to 2kHz, monotonic, but increasingly severe nonlinearities arise. The thrusters, require a much lower bandwidth than this, however. An additional analog filter acts as a slew limiter at 100ms/full throttle, or 10Hz. This prevents damage to the power electronics, but also sets a low required bandwidth. Low frequency and DC error is more of a concern, therefore. The output follows the input within an accuracy of 1%.

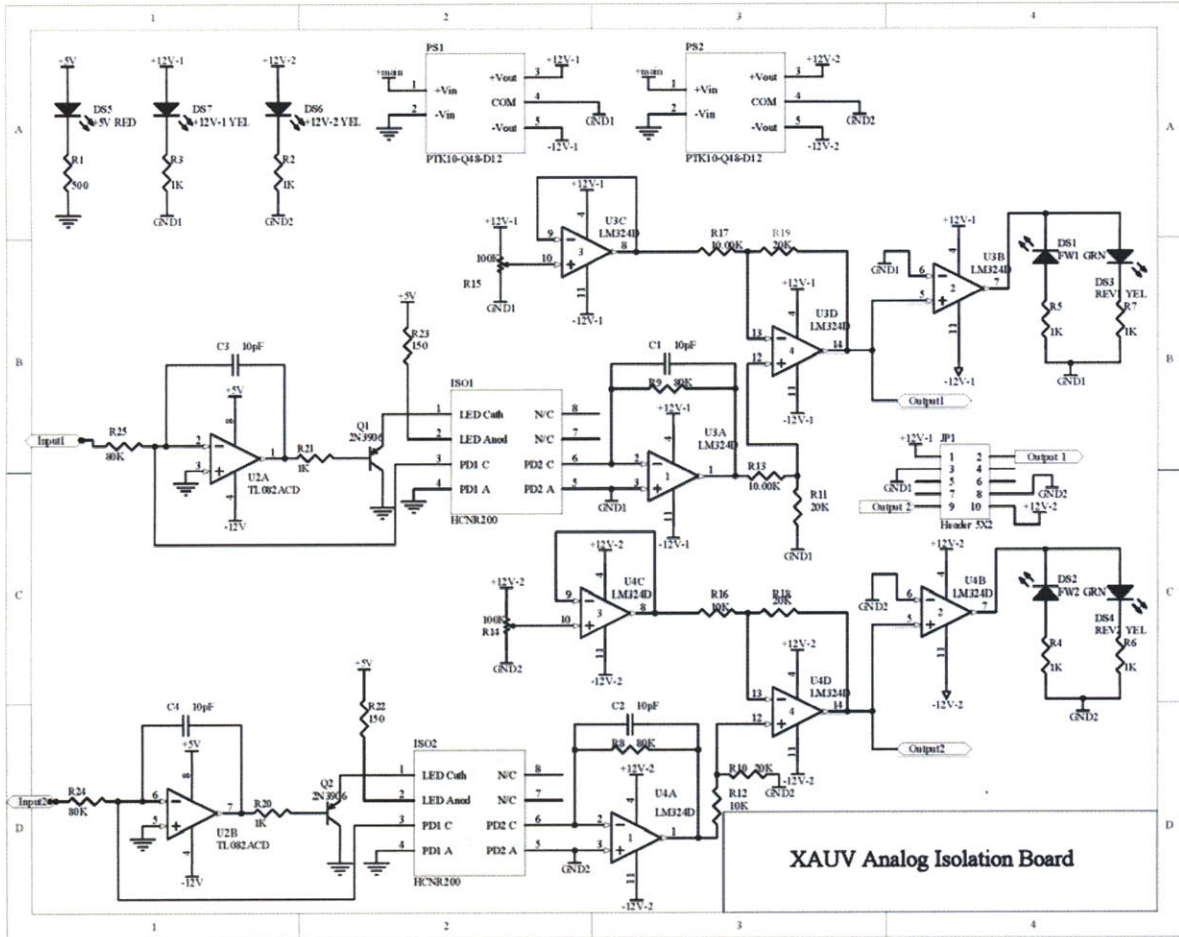


Figure 3-10: Isolation board circuit

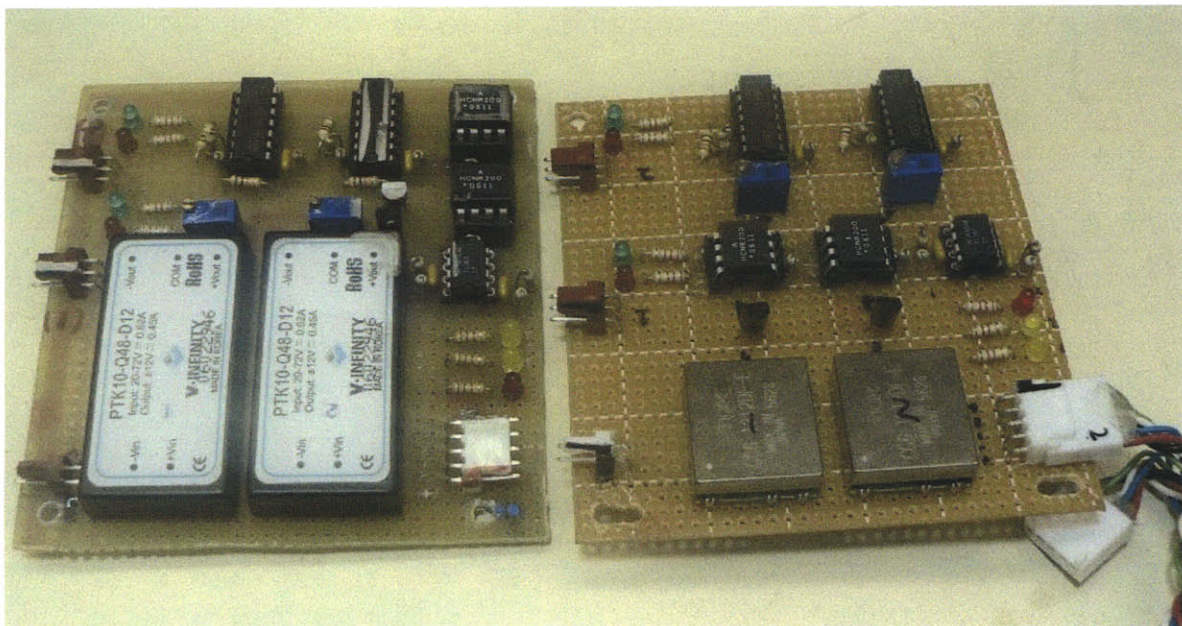


Figure 3-11: Isolation Board Hardware

3.2.4 Analog Signal Board

Although the PC104 stack has an onboard data acquisition card, many sensors are not suitable for direct attachment. The analog signal board serves as an interface between these sensors and the computer, by providing excitation voltages and scaling or filtering outputs, as needed. It also provides a degree of protection through input buffer op amps. A detailed discussion of this protection is provided below. The signal board is designed with a direct interface for the vehicle pressure sensor, two additional bridge input instrumentation amplifiers, and four buffered inputs.

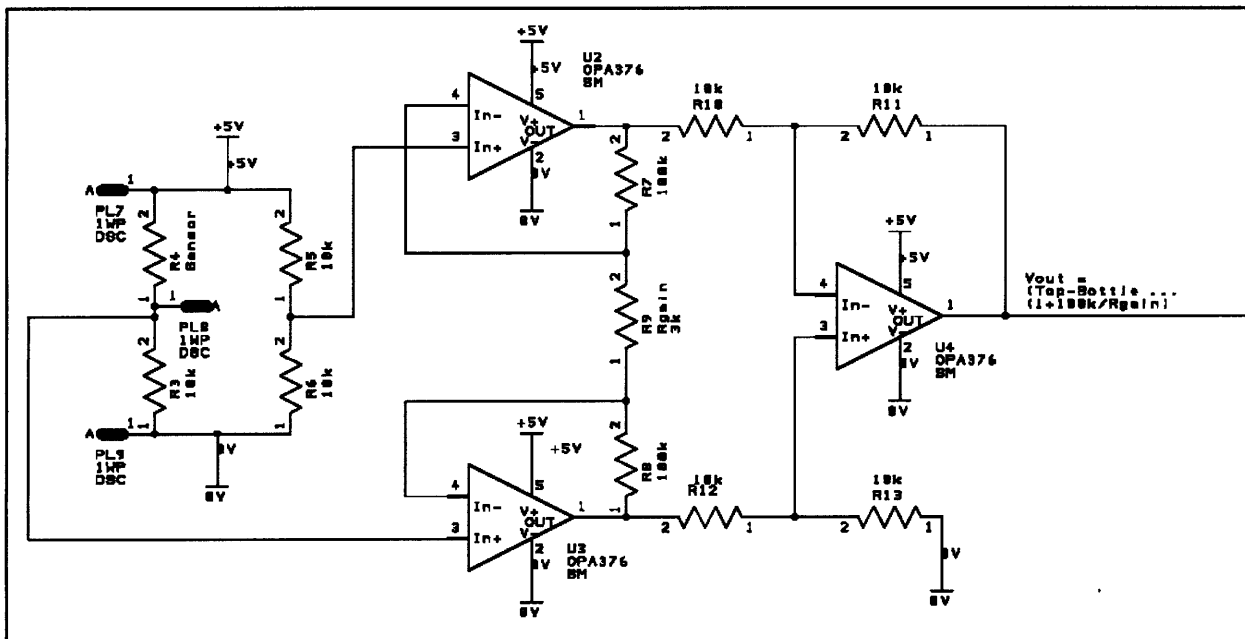


Figure 3-12: Standard bridge input instrumentation amplifier. The bridge is essentially a pair of voltage dividers. Each is fed to a separate non-inverting amplifier, with a single balancing gain resistor in between. The outputs of these feed into a differential amplifier. A typical implementation allows the precise measurement of resistive sensor while maintaining low input impedance.

DAQ Input protection

All sensor inputs go through at least one following op amp (fig below, a). In this configuration, an errant signal must physically destroy enough of the op amp to flow from the input directly to the output. Although not true isolation, this protection offers significantly greater protection than an actively buffered output (fig below, b), as with a typical inverting gain topology. Typical input impedance for an op amp, such as the OPA376 used in the circuit, limits the input bias current on the order of picoamps [47]. The physically buffered circuit allows this small current to flow through the input to ground, then generates an output voltage from the power supply rails. In a typical breakdown, the input stage would short the input to ground. Although potentially dangerous for the sensor, this will likely not generate excessive voltages on the output. In this manner the more important data acquisition card is protected. An actively

buffered circuit, such as an inverting or difference topology, typically found in analog filters, maintains the output voltage by sourcing or sinking current to or from the input or output, depending on voltages; that is a lot of options, some of which lend themselves to danger. The output follows some static (gain) or dynamic (filter) physical equation, based on the application.

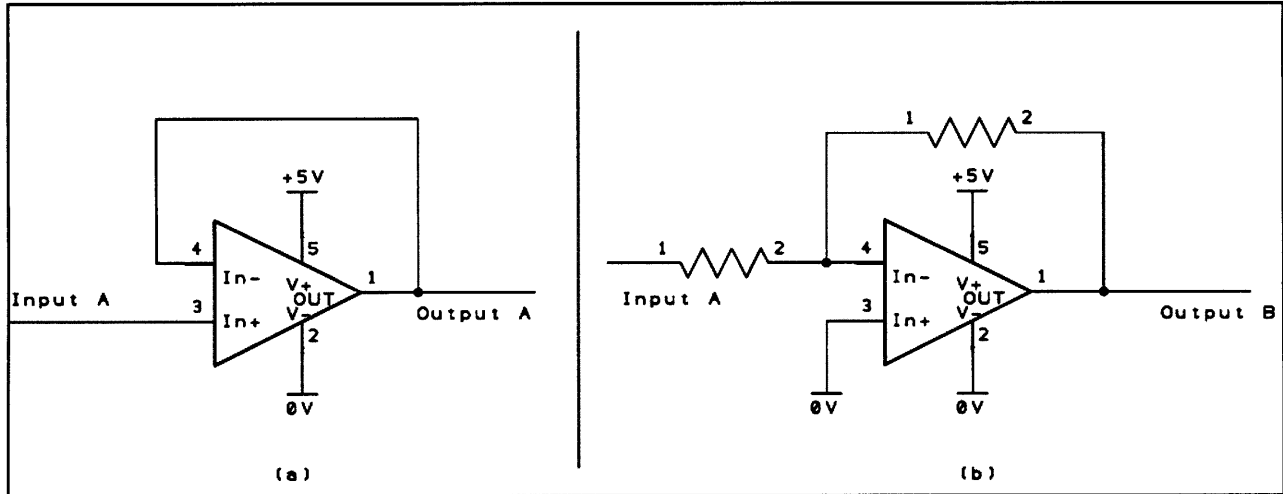


Figure 3-13: Op Amps. (a) Physically protected buffer (b) actively protected buffer

If an input within the op amp nominal ranges is put through an equation which should generate an output beyond its output voltage range, it will saturate. As an example, for with a 3x gain and an output range of 0-5V, an input of 3V will command a $3 \times 3V$ output of 9. The opamp will output 5V instead. This is normal and not damaging. Additionally, and to great benefit, it protects the data acquisition circuitry by limiting the output to an acceptable voltage. If the input voltage to this op amp is much higher, say 12V, the op amp will command $3 \times 12V$, or 36V. If the input voltage does not destroy the op amp, the output saturation will attempt to limit things to 5V. Whether it will be able to or not is a question of its internal current sinking capability. If too much current is drawn, the op amp will be destroyed. If the current draw saturates, instead, the op amp will not be damaged, but the output voltage will exceed 5V. This may cause damage to the data acquisition card. If, on the other hand, the input voltages are sufficient to destroy the op amp, it can do one of several things, none of which are particularly good. In the best case, it shorts everything to ground, possibly destroying the sensor, but also possibly protecting the data acquisition card. It may also stop generating an output altogether. This is especially bad. If the opamp is no longer controlling the output, the circuit (b), the op amp is essentially removed from the circuit, leaving only two resistors in series. Although the two resistors may limit the total current entering the data acquisition system, it does nothing to limit the voltage. If the high input voltages are still present, they will pass directly through, and likely damage the data acquisition card. Because of this risk, a physical buffer is implemented on each channel.

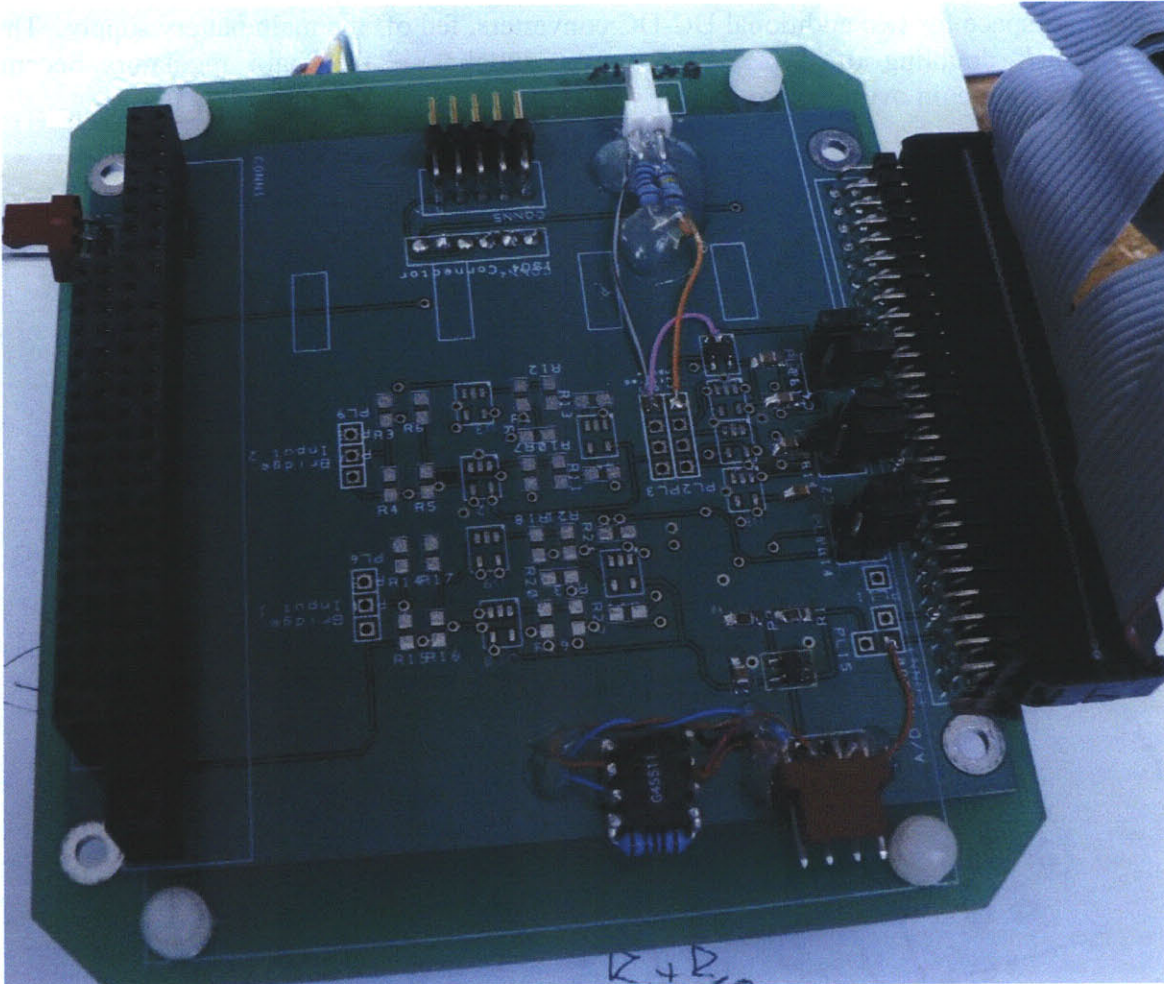


Figure 3-14: Analog signal board hardware

3.2.5 Auxiliary Power Switching Board

The auxiliary power switching board controls the flow of electricity to peripheral electronics. The board has three 5V and three 12V circuits that can be individually controlled by PORTA on the DAQ card digital output. This allows certain subsystems to be deactivated for modification, while allowing others to remain functional.

For instance, one 12V line controls the main thruster power relay. With this relay on, the thrusters are enabled and can spin. While launching/ retrieving or performing adjustments on the vehicle, it is important that the thruster power can be shut off. Because there is a significant inconvenience associated with shutting off the master power, this selective shutdown allows a smoother development process while still maintaining acceptable safety levels. As a similar example, a 5V circuit is used to power the servo control module. With this circuit shutdown, the servos can be backdriven safely for maintenance or passive operation.

The board has space for two additional DC-DC converters, fed off the main battery supply. This allows a quick doubling of the 5V/12V power supplies if the main regulators become overburdened and begin overheating.

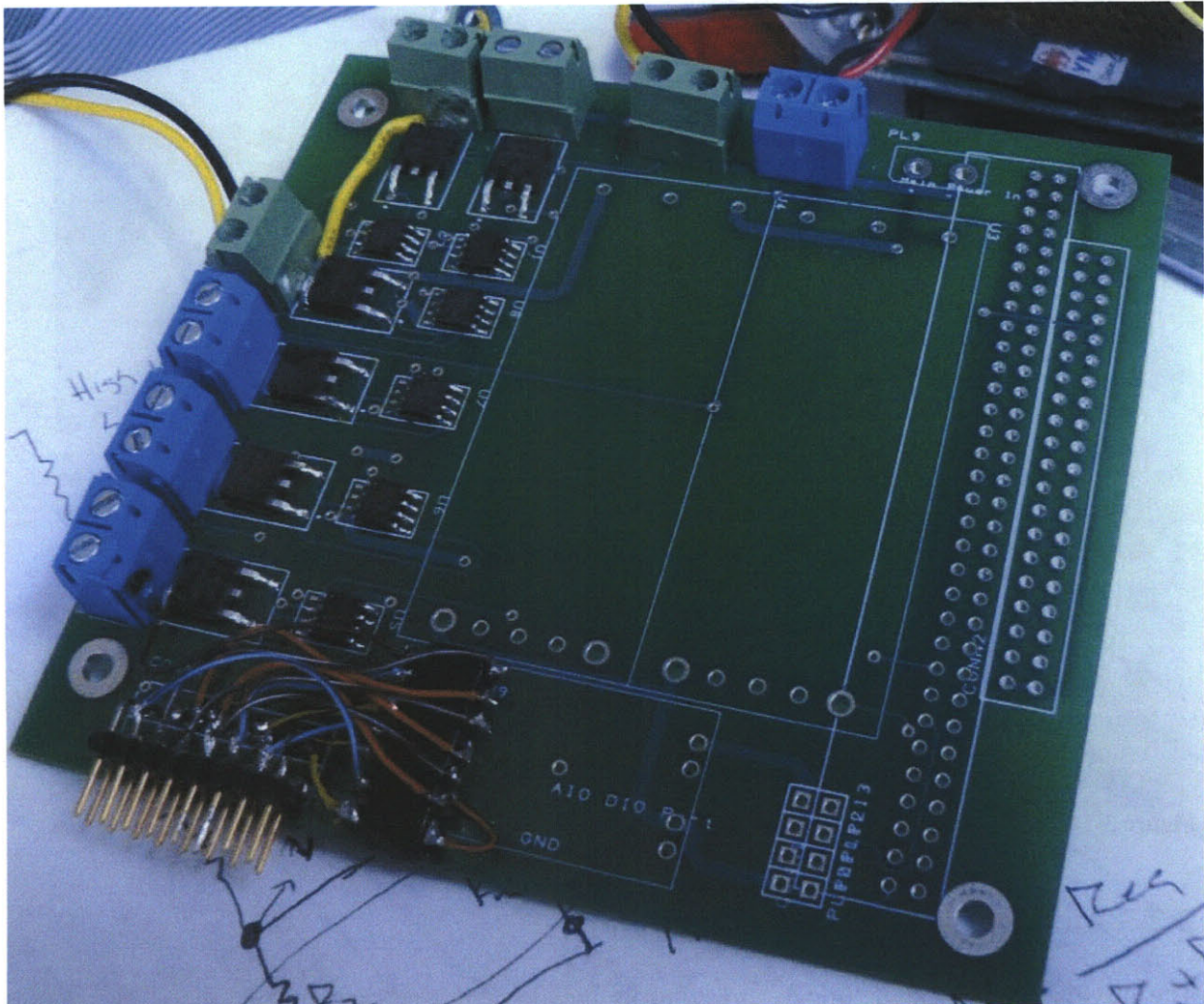


Figure 3-15: Auxiliary power switch hardware

3.2.6 Power Bus Brick

The power bus brick is a main housing module in between the bulkhead and the control stack. It integrates the main power bus and its fusing, power breakout and fusing for the thrusters, power line filtering, and switching for the switched power bus.

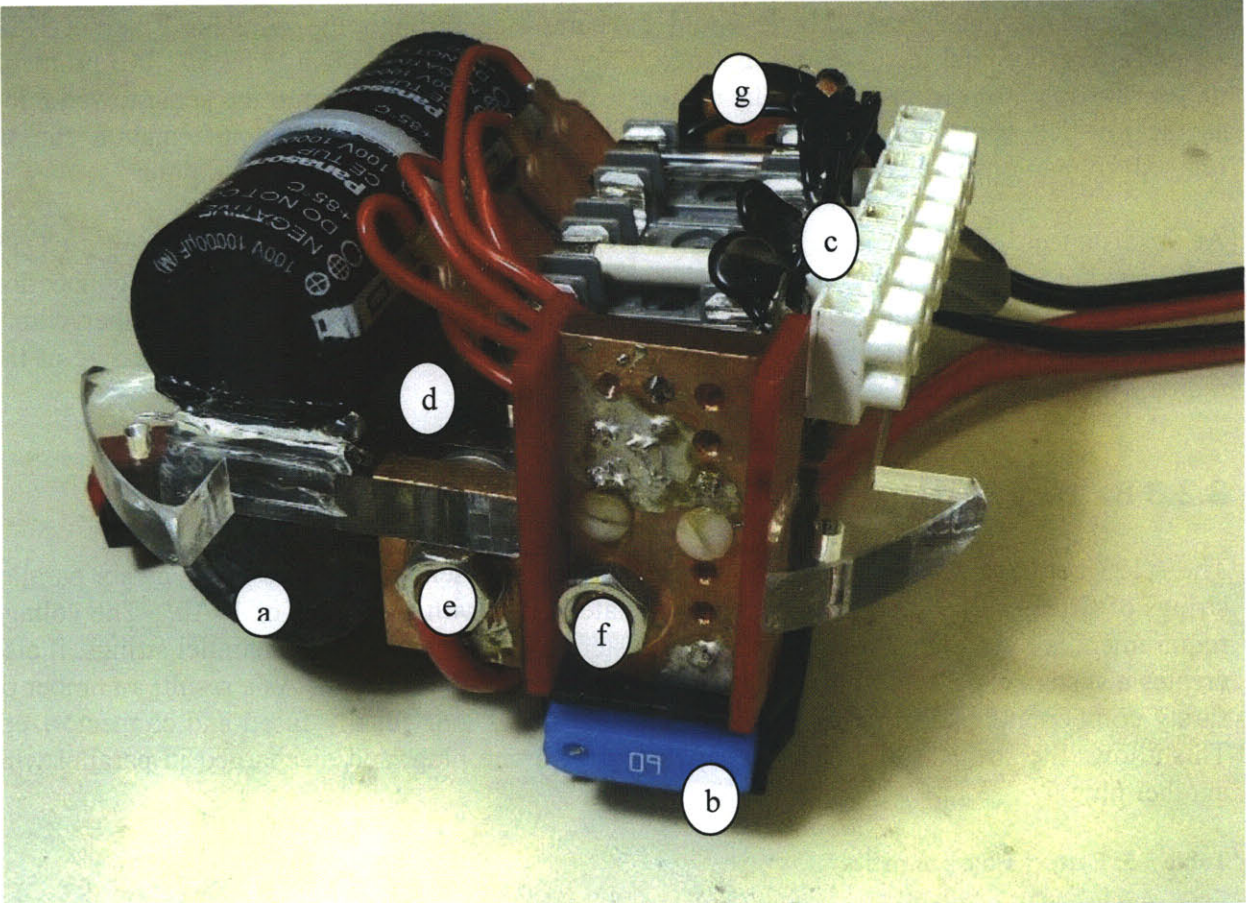


Figure 3-16: Power bus brick. The power bus brick incorporates (a) filtering capacitors, (b) mainline fusing, (c) thruster breakout, fusing, and MOVs, (d) main power relay, (e) fused main bus, (f) switched main bus, (g) ground bus

3.2.7. External Control Input

As a practical matter, the XAUV is designed to be controlled by a user on shore. For small tank testing, a short Cat5 cable is connected directly to the onboard Ethernet hub. By using remote desktop, the onboard control software can be directly monitored, operated, or edited. Although there is a performance degradation associated with this data flow, it is not sufficient to prevent testing and calibration of the system. While tethered in this manner, a hard-kill connection can be made to the main power relay. This requires an external 12V supply on the topside, but allows the user to terminate operation with 100% certainty at any time.

For nearby steering control, a towed buoy can be used to communicate with a USB wireless game controller. This is designed primarily for in-tank usage, as the best-case range is approximately 10m. This is implemented through the standard Windows joystick library 'winmm.dll'. The wireless controller has a cabled USB receiver, which is waterproofed and towed as a buoy. The joystick may also be used on the topside computer

At moderate distances, the vehicle could be controlled through the wireless Ethernet USB adapter. The nominal ideal-condition range is approximately 300m using the 802.11b protocol, or 70m for the 802.11g protocol. The adapter would be incorporated into the same towed buoy as the joystick controller. Using the Winsock API, data and command information can be relayed from an onshore command computer directly to the onboard control computer.

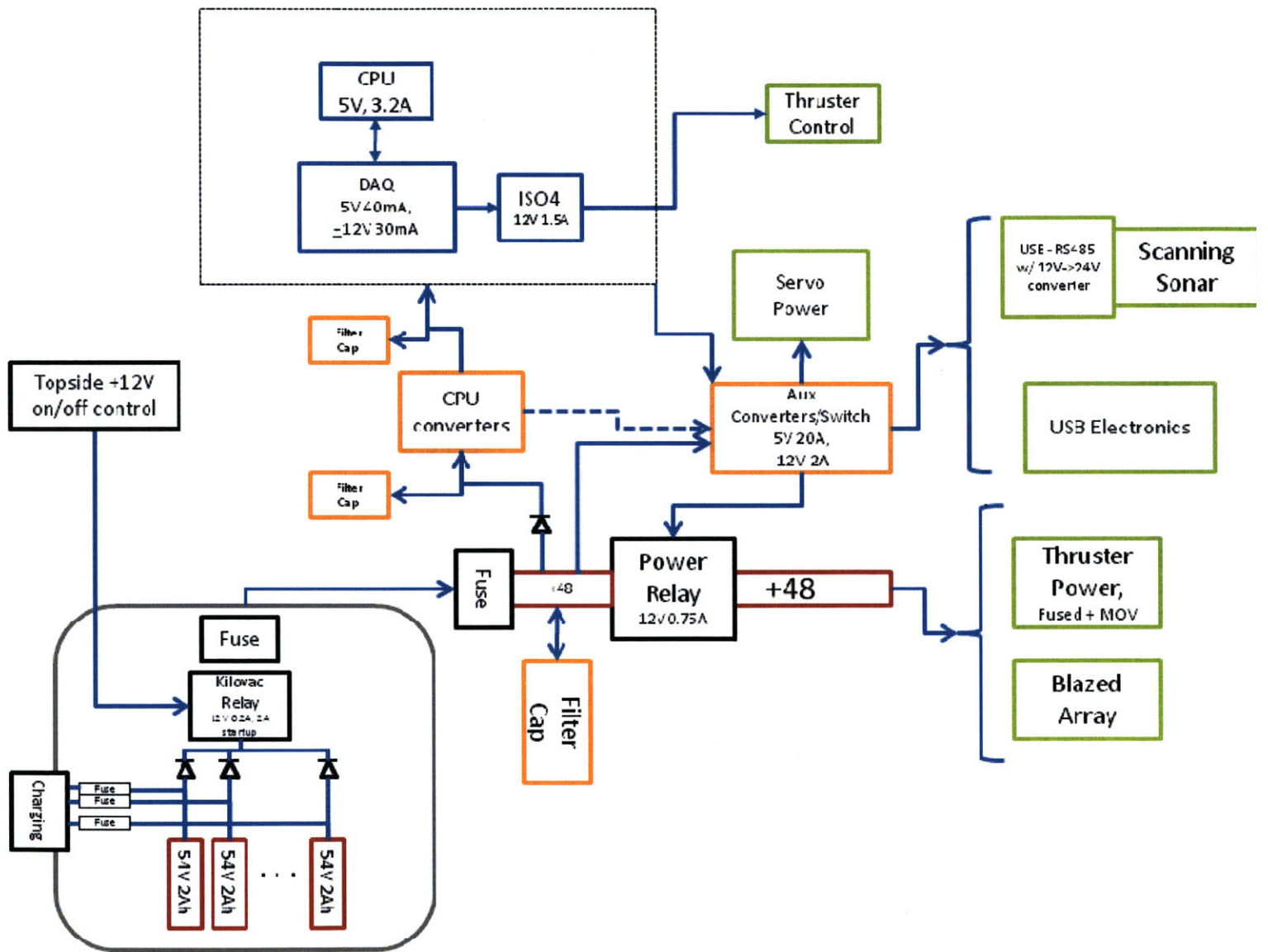
At far distances, a full wireless Ethernet transceiver could be implemented. By using a COTS radio on shore and in a buoy, remote control functionality similar to the direct Cat5 connection may be achieved. Several SeaGrant vehicles use this scheme very successfully, either with a buoy or in the hull for on-surface communications. It should be noted that the large size of the radios would necessitate the fabrication of a larger buoy.

3.2.8 Battery System

The A123 batteries selected in the previous chapter are assembled into as many as six parallel strings of fifteen cells. This provides approximately 50V with a capacity of 14Ah. The voltage requirement is the lowest available for the thrusters selected. With all six parallel strings, it also creates a system capable of delivering a maximum of 1.8kA, or 90kW. As a result, a number of safety components had to be introduced. Each string is individually fused and connectorized. This allows the strings to be charged separately with one plug, and discharged in parallel with another plug.

Table 3-5: Battery Characteristics

Nominal Voltage	49.2V
Voltage Range (full-empty)	54.0V-30V
Capacity	14Ah
Energy	0.688 kWhr
Weight	13lbs



When charging the A123 batteries, a constant-current, constant-voltage (CC-CV) approach is used. A current limited power supply applies 3A per parallel cell until the voltage reaches 3.6 V per series cell. The voltage is held here until the parallel cell current reaches less than 50mA. At this point the charging is complete.

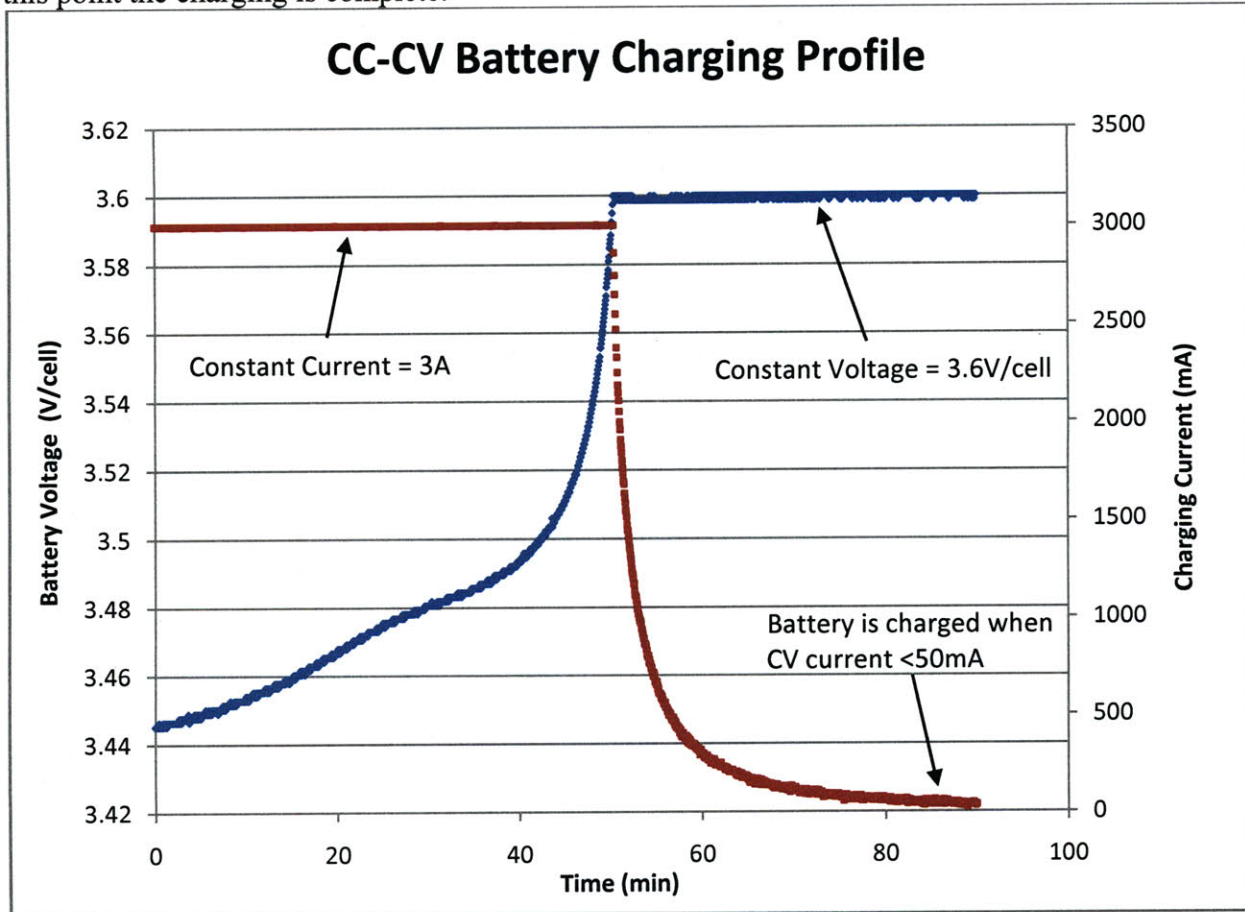


Figure 3-18: Typical charging profile, measured and normalized from a 16S2P battery configuration. Charging current is total current, thus per string current is 1.5A in this case, half the nominal value.

Charging arrangement and method

In safest possible arrangement when charging the batteries, each cell is charged individually. This can be likened to individually charging AA or AAA batteries in a wall charger. Each cell is exactly charged to the required voltage and current limit. If a cell is damaged, the charger alerts the user, and the cell is discarded. Once the cells are individually serviced, they are returned to the device and assembled into a higher voltage battery. This is an acceptable usage model for small numbers of cells, such as the two or four typically used in consumer electronics. The user typically needs only to process a single batch before the device may be used again.

In the case of the XAUUV, disassembling the battery for each charge is undesirable, because it would require opening the seals on the waterproof housing, thus introducing the possibility of seal assembly failure when charging is complete. Additionally, the process would require between 15 and 60 cells to be recharged. The physical process involved in doing this would be

significant. An alternate solution would have charging wires running off each cell to an external connector or charge controller. In this manner, each cell can be hooked up to the charger without disassembling the battery. With this approach, one charger could be sequentially hooked up to each battery. Additional charger blocks would allow multiple batteries to charge at once, up to the limiting case of one charger for every battery. This becomes difficult unless care is taken to isolate each battery charger from the others; otherwise, a short may easily occur. Globally, this single cell approach has the particular advantage of requiring a low voltage. This allows the use of COTS battery charging or power supply components; IC manufacturers have in recent years developed a wide array of low voltage components for digital cameras and similar portable applications. It has the disadvantage of a large number of switches. At the maximum, two switches for each cell multiplied by two switches for each power supply. Clever use of blocking diodes or battery segmentation has the potential to reduce this number. Another option uses a single power supply, and uses a resistor or a bypass on each battery to limit any overcharging of a single cell; if an overcharge condition is identified, the switch prevents any additional charging, or discharges the cell to a safe voltage. A zener diode can be used as both switch and shunt [48]. If the if the approach requires that voltage is burned off, the heat generated must also be safely dissipated.

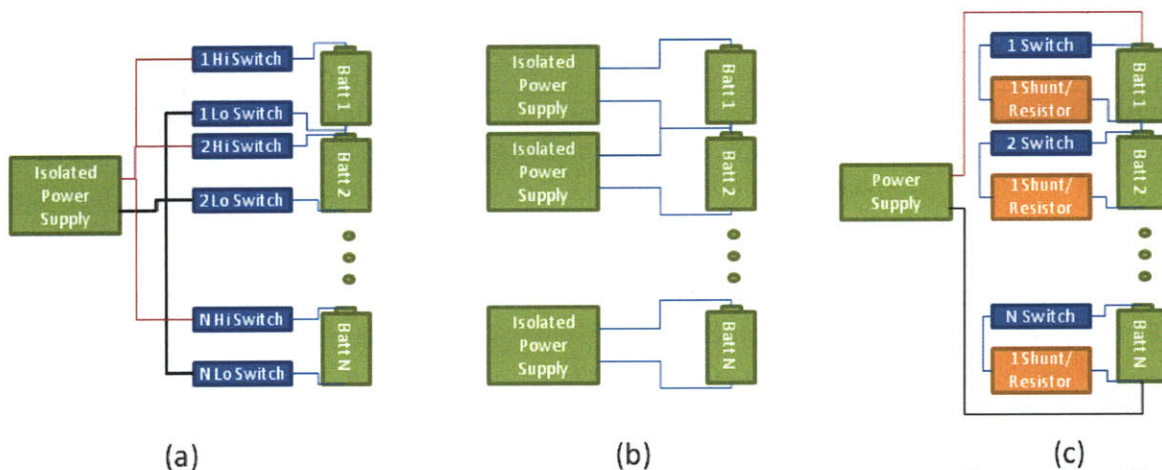


Figure 3-19: Individual cell charger Topologies: a) hi-lo switch b) isolated power supply c) bypass/burn off

In general, however, the A123 batteries are relatively well-behaved, and can be charged together in series. This is the approach used in DeWalt’s current line of power tool batteries with A123 cells. In contrast to the previous approach, the charging system must develop the full battery voltage of 50V. This could be an inconvenience for developing a custom solution, or require harder-to-find parts. Looking at the opposite side, charging all 15 cells at once drastically reduces the number of individual charging blocks required. Additionally, a single charger capable of delivering the full 3A will charge faster than a single one battery charger. A quick power calculation confirms that $3A \times 50V = 150W$ will stuff energy in much faster than $3A \times 3.6V = 11W$. In the XAUUV implementation, a lab power supply is used, thereby eliminating any custom current control electronics from the design.

In charging the parallel strings of batteries, a few of the same issues arise. With a separate charging wire to each of the six strings, only twelve wires (or connector pins) are required. If the ground wires may be connected, only seven conductors are needed. This greatly simplifies

the connector. Several system-level questions arise when considering how to proceed with switching. For the connector, all power can be driven through two large conductors, with internal control hardware to switch between strings. Similarly, a single connector with twelve conductors (each battery string) can be used. When connected to the robot, the six positive conductors are connected, as are the six ground conductors. In either case, the connectors must be able to carry the full 50A system draw. This acts just as the two large conductors did. For recharging, however, the mating connector keeps each string separate, and moves the switching hardware off the robot. In general, the electronics will be cheaper, since lower power, bulkier components like signal relays may be used. Alternatively, the battery can be charged through a separate charging connector. The charging electronics may be internal or external, depending on the connector arrangement. In general, the connectors can be smaller, since only charging currents are required (3A). One additional consideration is that the system may operate off the battery while simultaneously being charged. If the main power connector must be disconnected for charging, the electronics need a separate shore power connector. It is unlikely that the main thrusters can operate off this, however. If a single connector is used, however, the battery can be charged, while a large shore power supply can run the vehicle at the same time. Of course, this same strategy may be applied to the multiple connector approach as well.

Table 3-6: Comparison of differing charge control electronics and connectors

	Onboard Switching	Offboard Switching
Single Connector	<ul style="list-style-type: none"> - shore power thrusters ok - no interrupted operation charging - very large connector 	<ul style="list-style-type: none"> - shore power thrusters ok - no uninterrupted operation charging - several medium connectors
Separate Connector	<ul style="list-style-type: none"> - small conductors - uninterrupted charge - no thrusters on charge 	<ul style="list-style-type: none"> - several small conductors - uninterrupted operation

MBARI developed a switching solution using a FET and relay in parallel. The system combined the near zero on state resistance of a relay with the soft switching of a FET [49]. When switching DC loads, mechanical switches will arc, or spark between the two contacts before they are in mechanical contacts. It is possible to switch AC without this effect, by toggling the switch at the exact point that the voltage passes through zero. Because there is no instantaneous voltage difference, there is no arc. This is called a zero-crossing switch. Solid state switches, on the other hand, transition from a high-resistance ‘off’ state to a low resistance ‘on’ state in a continuous manner, passing through intermediate resistances, although typically for a short period of time. The MBARI approach uses this transition resistance to gently start the flow of current, before switching on the parallel relay to nearly eliminate any steady-state on resistance. Similarly, when switching off, the relay switches off first, transferring the load to the FET. The FET can then turn off with a smooth transition.

In a MOSFET, as charge accumulates on the gate, a conductive channel widens in the base silicon. The more charge, the better conductivity. In the intermediate stage, current I is flowing

through a resistance, thus creating a voltage drop V and dissipating energy as heat. From the definition of electrical power

$$P_{dissipated} = I * V_{drop}$$

$$E_{thermal} = P_{dissipated} * time$$

COTS gate driving chips function to transition through this phase as quickly as possible, thereby limiting the energy dissipated in the FET as heat. For fast, repeatedly switching applications like regulated power supplies or motor drivers, this switching energy dissipation is a major limitation. The timescale that determines what is ‘frequent’ is a function of FET resistance and switching time (which involves gate and gate drive characteristics), but is much less than one second. For infrequent switching, as in the case of a on/off battery control circuit, the main energy dissipation is in the small internal resistance of the fully on FET. Typical values for the ‘on’ resistance range from 1 to 100 mOhm. If the load circuit is drawing a significant current, say 25A, this small resistance can result in a large thermal dissipation. Again, from the definition of electrical power, and assuming all the loss is transferred to heat,

$$P = I^2 R$$

$$P_{thermal} = 25^2 * \{0.001 \sim 0.01\} = 0.6 \sim 6.2W$$

This type of heat dissipation could quickly become a problem in a small pressure housing, if care is not taken to direct it into the surrounding water. In our case, this is a minor contributor to total loss, so the combined relay-FET approach is not needed.

When combining multiple batteries into a large pack, diodes can be used to prevent one pack from charging the others. This is especially important with the low internal resistance of the batteries selected, as large currents could develop. Unfortunately, diodes also have a loss associated with them. Diodes have a set forward voltage drop. This means that any current will create the same voltage drop. This effect can be taken advantage of to make a low-efficiency voltage regulator. In our case, it means that we will have a power loss directly proportional to the current. From power, and with typical voltage drops between 0.1 and 1V,

$$P = I_{load} * V_{diode\ drop}$$

$$P_{thermal} = 25A * \{0.1 \sim 1\} = 2.5 \sim 25W$$

Although the FET has a square law loss function, the diode’s higher effective resistance will cause it to dissipate more energy in the applicable range of currents.

Circuit protection

Additionally, a number of safeguards are implemented to protect the delicate electronics for the thrusters. Fuses are used to prevent fires onboard. The full power output of the battery system is at the upper end of what fuses are able to protect, in terms of electronics [50]. Fuses with a slow enough blow time to allow short spikes for capacitor charging or motor startup loads are also too slow to catch a quick surge capable of destroying control electronics. However, the value of the fuse remains in preventing a continuous discharge of energy into those electronics, thereby

preventing an electronics fire and possible damage to the batteries. If short surge currents can be avoided, a lower fuse trip current can be chosen.

In order to prevent an initial surge of current into the system on startup, a negative temperature coefficient (NTC) thermistor can be used [51]. The internal resistance of a NTC decreases as the temperature increases; sensors use this effect to measure temperature. In a current limit application, the initial high resistance limits the current flow. As the current passes through the resistor, it heats up, also lowering the internal resistance. As more current flows, the resistance drops even more, down to some low nominal value, typically 0.1~0.01. The thermal dissipation here is thus loosely on par or an order higher than that of FETs. It is important to consider, however, that once the device heats up under normal circuit operation, the limiting function is lost.

Another circuit protection element is the Metal Oxide Varistor (MOV). This acts like a bi-directional zener diode, by shorting any voltage above a certain threshold [52]. When placed across the power leads, any downstream circuit element will only see voltages up to the MOV breakdown voltage. These are used to protect from short, sudden spikes in the power supply, but are not capable of sustained over-voltage protection, as the devices will overheat. Because these are parallel with the load circuit, no steady-state current or loss is associated with the elements. For misbehaving power supplies, the dissipation could be estimated from typical noise characteristics.

The current limiting equivalent to a MOV is a polymeric positive temperature coefficient (PPTC) device. Whereas the voltage shunting MOV is placed across the load, the PPTC is placed in line with the load. Up to the trip current, the small internal resistance prevents the device from heating up. Once above the trip current, however, the nonlinear resistance curve allows the device to heat up quickly. As the device heats and expands, the internal resistance drastically increases, and shuts off the flow of current [53]. These devices have the added benefit of being self-recovering. As the current decreases, the device will cool down, shrink, and begin conducting again. These devices are useful in protecting against temporary overloading or surges. If a permanent short from damage or misconnection is present, an additional series one-time use fuse may be necessary to prevent the PPTC from reattempting to current flow.

3.3 Sensor Payload

The XAUV is equipped with several inspection sensors. These can also be thought of as mission-oriented sensors. Although they also are used in navigation, their primary purpose is for object identification and classification. In addition to the main sensor payload, additional sensors are used to assist in navigation.

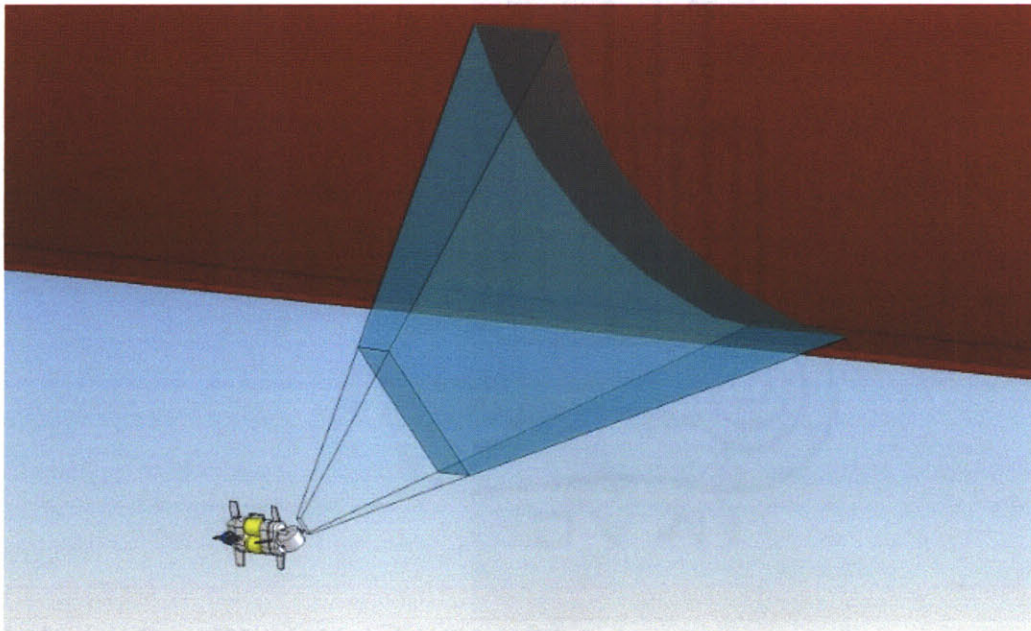


Figure 3-20: rendering of XAUV shiphull inspection, showing field of view of blazed array.

3.3.1 Blazed Array

The XAUV is designed to carry a BlueView Technologies P900 blazed array sonar. This is a new imaging sonar type intended to replace the DIDSON (dual frequency identification sonar). Although the blazed array resolution is significantly worse than the DIDSON, the package is much smaller and the system is less expensive. Additionally, higher frequency versions under development will improve the close range resolution.

Table 3-7: P900 blazed array properties [3]

FOV	45° x 20°
Beam width	1° w x 20° tall
Beam spacing	0.18°
Number of beams	256
Range bins	256

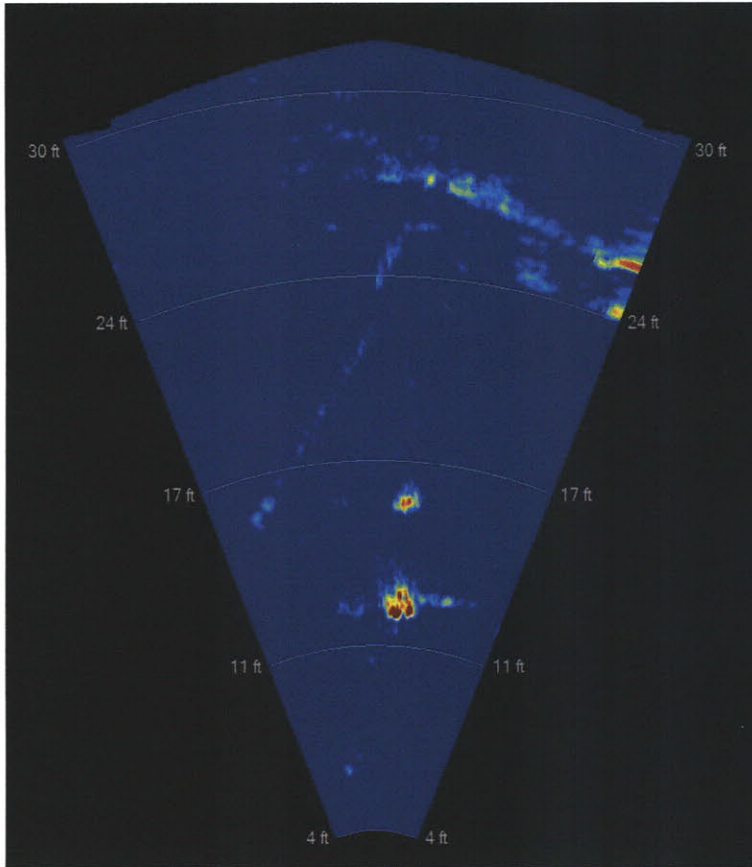


Figure 3-21: Blazed array image from test tank - tank walls(16ft-26ft), cinderblock (~12ft), and gantry head (~16ft) visible

3.3.2 Camera

An ATC2K helmet camera is used on the XAUV for video. This sensor is a ruggedized video recorder that has been miniaturized by eliminating the view finder and other user control options. It also records directly to SD flash memory, instead of larger tapes. The resolution and frame rates are selectable at {640x480, 320x240, 160x120 pixels} and {30, 15 fps} respectively. Although the sensor is not capable of streaming video, it serves as an independent data source for post-processing. For realtime video applications, the XAUV can also use a waterproofed USB webcam by plugging it into one of the standard USB wet ports.

3.3.4 IMU

A Microstrain 3DM-GX2 Inertial Measurement Unit is used for dead reckoning. The IMU outputs 3D accelerations and rotation rates, as well as magnetic north orientation. The accelerations may be integrated twice within the context of the orientation to provide an estimate of position. Because of the relatively low accuracy of this double integration process, this

measurement is best used for fast processes or as a component in a minor-loop feedback scheme. Relevant resolution and accuracy metrics are tabulated below. The IMU also has built-in filtering which combines the gyro and magnetometer measurements to produce an orientation estimate. This gyro-enhanced orientation measurement is the primary control input for pitch and heading.

Table 3-8: IMU characteristics [54]

Accelerometer	5g range, 0.005g stability, 0.2% nonlinearity
Rate Gyro	+300°/sec range, 0.2°/sec stability, 0.2% nonlinearity
Magnetometer	1.2 Gauss range, 0.01 Gauss stability, 0.4% nonlinearity
Filtered Orientation	±2° typical dynamic accuracy, <0.1° resolution, 0.20° repeatability

The interface for the IMU is a virtual serial port through the IMU's built-in USB device controller. The GX2 offers floating point output for reduced information loss when compared to the fixed point 32bit binary output of the previous GX1. The IMU is polled with a single command byte (0xCE). It then responds with 18-byte packet containing a header byte (0xCE), data, and checksum information. Alternate information or reprogramming of calibration values is possible through a similar process with different command bytes.

3.3.5 Depth Gauge

A 100psi strain gauge pressure sensor is used as a depth sensor. The sensor has a metal diaphragm that expands or contracts with varying pressure. Electronically, the sensor acts like an internally referenced wheatstone bridge. With an excitation above 10V, the sensor puts out differential voltage of 0-100mV full scale. This must be put into a differential or instrumentation amplifier in order to eliminate common mode noise or DC offsets. Once referenced to ground, the signal can then be amplified to a 0-5V full scale signal for easy measurement in the DAQ.

3.3.6 Scanning Sonar

A scanning sonar may be used for navigation and obstacle detection. It outputs return strength in all directions in a plane. This has lower resolution than the blazed array, but is useful for use when the main sensor platform is scanning the hull (not looking where it is going). Additionally, this sensor provides obstacle detection within the blazed array dead zone and near-field, or approximately 5m.

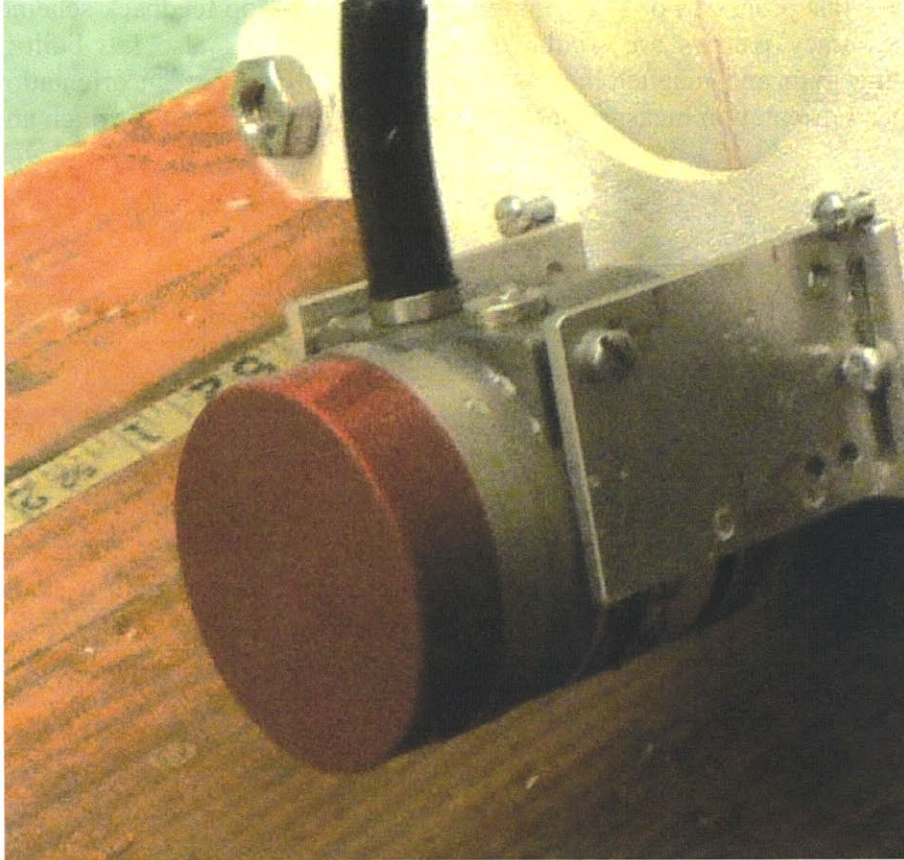


Figure 3-22: Imagenex 852 Scanning Sonar

The scanning sonar speed is range and mode dependent. In a 360° scan at 650kHz with a 40m range, the sonar operates at approximately 0.125 Hz. At 850kHz and 5m range, the system operates at 0.33Hz. The sonar can be placed into a sector scanning mode, where it scans back and forth between any two bearings. For example, with a 72° sector at close range, the system operates at 0.6Hz. In the limiting case, it can be configured to look in only one bearing, thereby functioning as a higher-rate forward looking sonar.

The sonar has native support for serial communications through RS485. In order to interface with the USB/+12V Aux connector, a DC/DC converter and USB adapter are used in line with the cable.

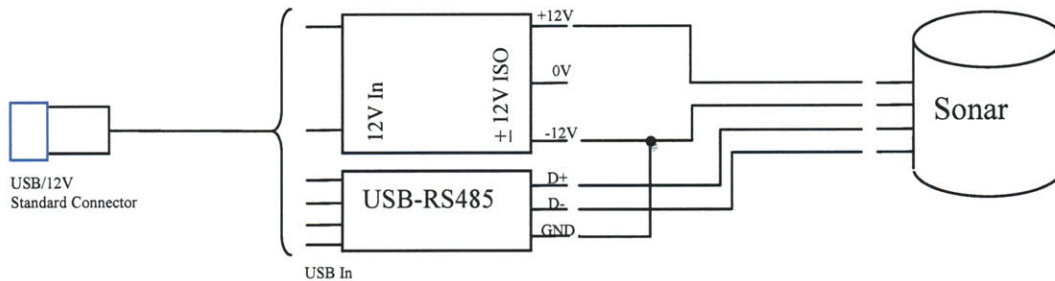


Figure 3-23: Inline adapter to allow sonar to interface with standard USB/12V connector.

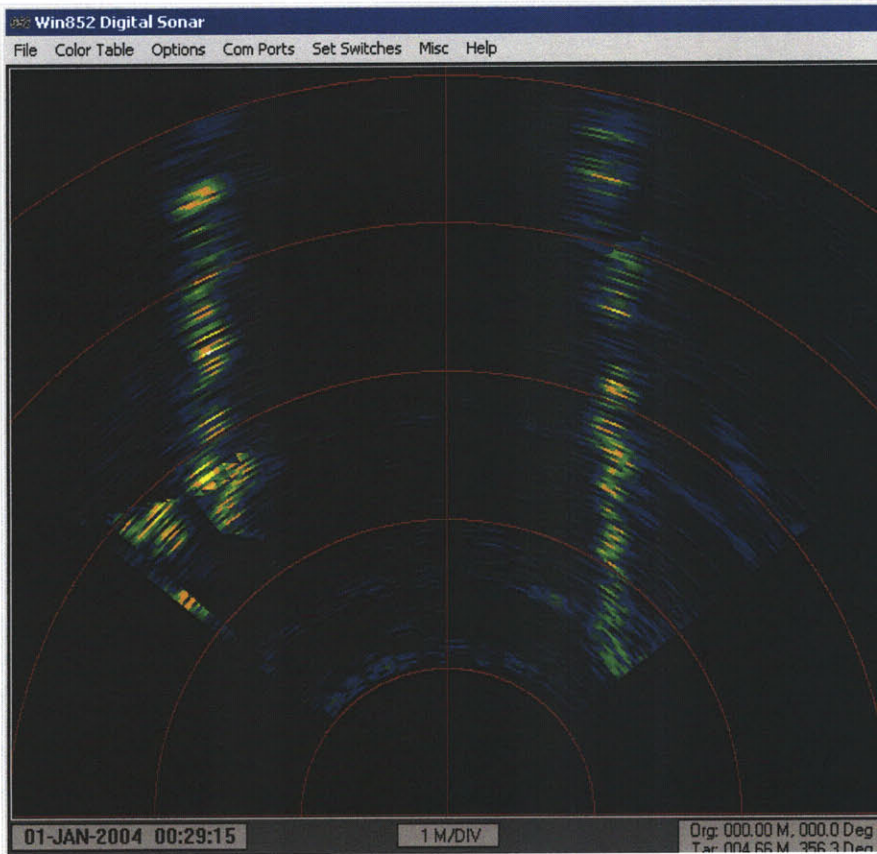


Figure 3-24: Scanning Sonar Image, showing sides of MIT Towing Tank facility.

3.3.7 Hokoyu Laser Scanner

A laser scanner could be waterproofed in an attempt to use it for close-in navigation, similar to a navigation scanning sonar. The Hokoyu has the advantage of good close-in accuracy, with a minimum range on the order of inches, maximum of four meters, and resolution to the millimeter. Because it is optical in nature, it will not function properly in turbid water. Additionally, any distortion between the air-housing or housing-water interface will cause inaccurate or nonsensical measurements. It is likely that an error mapping procedure would be necessary to account for these distortions. If successful, the same sensor could be used for the KSV raft as well as the XAUV. This has not yet been attempted on this project.

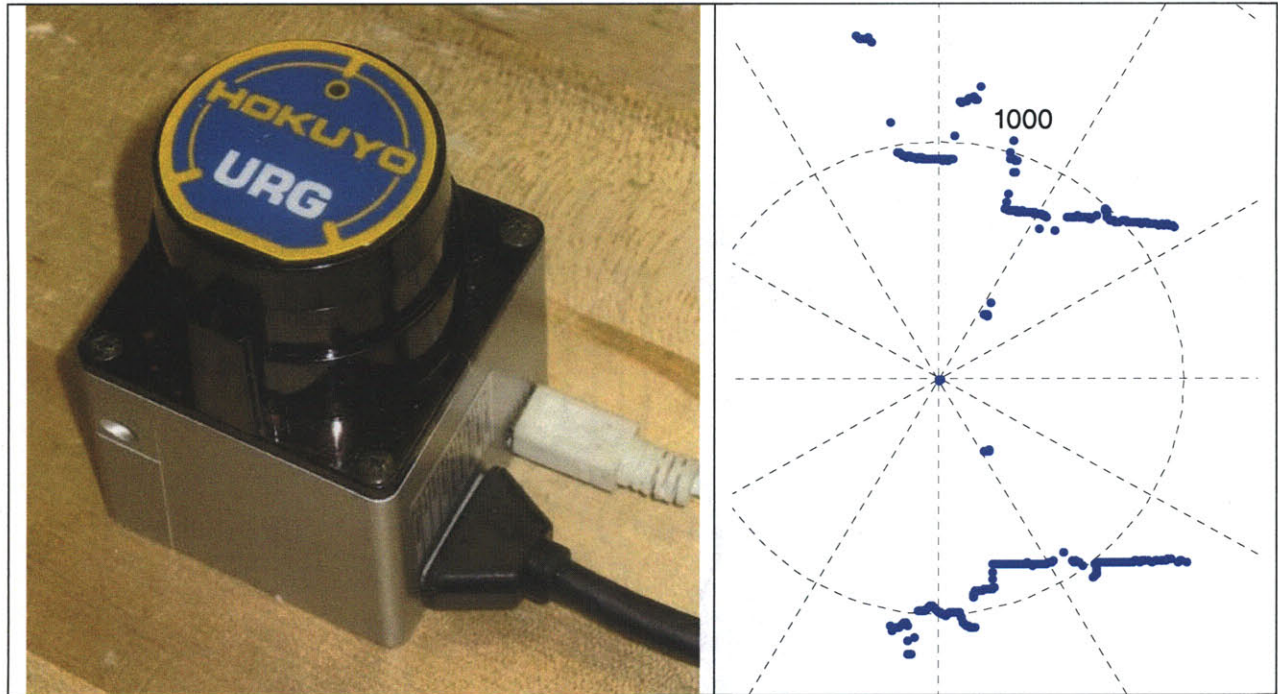


Figure 3-25: (left) Hokuyo Laser Scanner. Image courtesy of M. Kokko. (right) Scanning sonar data plotted in MATLAB.

3.3.8 GPS

GPS is useless under water, as a result of the sharp attenuation of the relevant frequencies in water. However, it is useful for surface operations, and for gaining an absolute reference before and after running a mission. By comparison, this gives a much better understanding of the accuracy of any onboard position measurement while below surface. Additionally, a clever filtering operation can post-process the mission position estimates for a better correspondence of observations to a global map. This, in turn, can be used to better ensure coverage or allow relocation of interesting features. The XAUV could carry a small GPS receiver in a towed buoy, or in the main housing.

3.3.9 DVL

The XAUV can also carry a Doppler Velocimetry Log in the large payload configuration. A DVL uses the Doppler effect to measure velocity with respect to a surface. Because only a single integration is required to compute position, the sensor allows for a relatively accurate position estimate, especially when compared with twice-integrated accelerometer data. By turning the vehicle on its side and changing the flotation and ballast, even a large RDI DVL, such as the one the HAUV carries, can be mounted either on top or on bottom. On top, it can be used to lock onto the bottom of a flat boat, such as a barge, for relative motion and standoff distance. If mounted on the bottom, it can be used to lock onto the seafloor for general

navigation. Because of the size of the vehicle, only a small DVL would be practical for normal operations.

3.4 Summary

After the overall framework of the vehicle is decided, the detail-level design is required to flush out the remaining subsystems necessary to make a viable robot. First, flotation is considered; by making the vehicle slightly positive, it will float to the surface should the vehicle run out of batteries. Additionally, an expanding polyurethane foam is selected for its robustness, high flotation value, and ease of formation. The method by which it was processed is provided. Next, the thruster mounts are designed, using first-order equations and then FEA to validate the design. The vertical thrusters are combined with the flotation system for a tightly-integrated module, while the forward thrusters use a simple cantilever design that facilitates adjusting them forwards and backwards as needed. Then the pressure housings are designed, evaluating several possible approaches, but eventually settling on a standard atmospheric housing sealed by o-rings. The electronic systems are covered, including the control stack. This includes the computer and peripherals, as well as some auxiliary boards that provide analog isolation, sensor interface, and power distribution capabilities. The battery system is also discussed, touching on methods by which the system can be charged and discharged safely. Last the vehicle payload is discussed, including imaging sensors like the blazed array and camera, as well as navigation sensors like the depth sensor, scanning sonar, and inertial measurement unit. From all these details the subsystems of the vehicle are flushed out, while still leaving room for future expansion.

Chapter 4

Control System Design

The control system is responsible for combining sensor data, reference commands, and some notion of its own response to stimuli, into meaningful motion. This can take place at the high-level, where emphasis is on operator commands and mission objectives, and at the low level, where emphasis is on regulating actuators.

4.1 High Level Control System

The high level control system develops navigation strategies by integrating information from several different sensor and actuator sources. It is responsible for developing paths to follow, as well as estimating the vehicle location and/or generating a map of surrounding features. As an output, it provides the reference inputs to the low-level control system for depth, thrust, and orientation. This is a supervisory control strategy. Alternatively, and depending on the implementation, the high-level control system could also provide direct thruster outputs; this would be the case with tightly integrated mapping and control, for example.

At the simplest level, the high level control could consist of direct human input or a G-code –like file of bearing, depth, and velocity. For example, the HAUV is capable of both methods, through joystick control and the simplest level of mission planning. Similarly, the Odyssey IV uses a mission file to execute operator-planned paths. Well designed GUIs, as with the Ocean Server AUV, allow for automatic writing of these mission files. This additional layer of abstraction can be likened to CAM software for automatic posting of G-code.

At a more complex level, the vehicle might make use of its sensor data streams to develop and execute plans on the fly. Although there are wide arrays of both analytical and heuristic approaches to mission planning and high-level control, only a few will be discussed here. One approach expands the 3D physical space to include the dynamics of the vehicle and estimates of the ground conditions. This trajectory space greatly facilitates obstacle avoidance by calculating exactly which trajectories will result in collisions or vehicle roll-over [55]. Several other approaches attempt to gain the maximum amount of information, for the purposes

of exploring, or achieving the best understanding of current location. One approach uses the Fisher information as a way to estimate the information that will be gained from a possible movement [56]. Another approach combines entropic information gain with a grid of trajectories for real-time path planning [57]. Another interesting area attempts to combine multiple, potentially contradictory objectives. This interval programming is used to coordinate mission tasks that depend on position estimates with the task of maintain the quality of the position estimate itself [58]. In this work, nothing so new is added to the body of knowledge, but rather, an existing estimation technique is examined as a potential navigator.

4.1.1 Linearized Kalman Filter

The Kalman filter is a method estimating values for a system. It weights the contribution of different sensor measurements and model-based predictions by estimating the reliability of each. A linearized Kalman filter takes a non-linear process model and precomputes a linearized model for each timestep of a trajectory. As long as the process is near the linearization point, the filter should work reasonably well. In essence, the Kalman filter is a Luenberger observer with a gain tuned for specific process and measurement noise parameters [59].

In order to use the LKF for vehicle navigation, an augmented state approach is used. The vehicle state and the location of every map feature are combined into one matrix [60] [61].

$$X_{augmented} = \begin{bmatrix} X_{vehicle} \\ X_{targets} \end{bmatrix}$$

For a free swimming vehicle, the system is essentially second order from input thrust to output position. As a result, the vehicle velocity is included as a set of states. Looking only at a 2D planar case (surge, sway, yaw), the full augmented state looks something like:

$$X_{augmented} = \begin{bmatrix} X_{veh} \\ \dot{X}_{veh} \\ X_{targ} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} x_{veh} \\ y_{veh} \\ \theta_{veh} \end{bmatrix} \\ \begin{bmatrix} \dot{x}_{veh} \\ \dot{y}_{veh} \\ \dot{\theta}_{veh} \end{bmatrix} \\ \begin{bmatrix} x_{targ1} \\ y_{targ1} \\ x_{targ2} \\ y_{targ2} \\ \vdots \\ x_{targN} \\ y_{targN} \end{bmatrix} \end{bmatrix}$$

Here the targets are simply point features with no quality, shape, or orientation information associated with them.

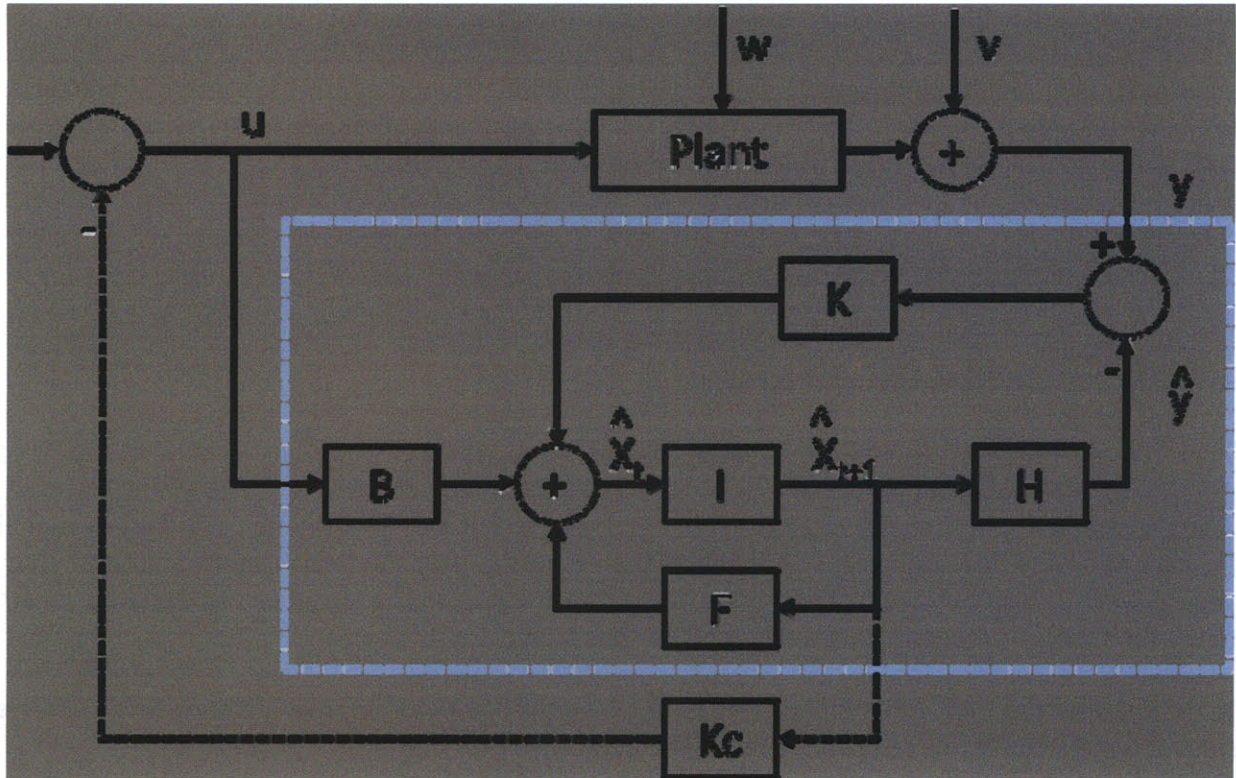


Figure 4-1: Block diagram of the core linear Kalman filter process.

The filter operates just as a standard observer. An controller generated input u acts on the plant, which is also affected by the process noise w . Process noise for a vehicle can include disturbances like waves or currents, as well as modeling errors. The plant responds with some physical motion or change of state. This is measured by some sensors, but not necessarily in the most convenient form. Absolute position can be hard to measure underwater, so distances to map features, or velocities might be measured instead. These measurements are corrupted by a measurement noise v , which can include quantization errors, intrinsic sensor limits, and environmental distortions. The noisy output of the sensors y are one of two inputs to the filter. The other input is simply the controller generated input u . Starting with some previous or initial estimate of the vehicle state x_k , an expected measurement is calculated through the observability process H . This matrix is the linearized relation that relates the state to the measurements.

The difference between this estimate and the actual measurement is then scaled by the Kalman gain. This gain incorporates the different trustworthiness weights for the various measurements and estimates. The process model itself is also used to create a prediction of the next state, based on the current state and controller generated input. The new estimate is the sum of the model prediction and the Kalman-gain scaled measurement difference.

In the above case, where the state vector also includes the locations of map features, the new estimate is the current best-guess for both the entire map, and the vehicle's location on it. Once the position estimate is acquired, full state feedback can be implemented for closed-loop control. Alternatively, one or a few of the estimated parameters can be used in separate classical control loops.

With linear processes and measurements, a standard Kalman filter can be used. Unfortunately, if either the measurement or process equations are non-linear, a more complicated variant must be used. One common approach is to use a linearization near some operating point to create well-behaved equations. The extended Kalman filter estimates the current position based on a linearization around the previous estimate. This approach has the intuitive benefit of 'starting where you left off', but does require the computation of the linearization at every time step. An alternative approach is the linearized Kalman filter, which uses a preplanned trajectory, and generates the linearized equations for each time step ahead of time. This approach can be useful where the intent is to follow a trajectory with some accuracy (instead of simply roving), and where calculating the linearized equations at each time step may be too computationally intensive [62].

Because the transition matrices are linearized ahead of time, the initialization is slightly more complex than an extended Kalman filter. In practice, the process would start with some initial estimate of position. This might be the result of a previous cycle, or a 'wake-up' guess from a few measurements. Next, a trajectory generator lays out the intended path based on some desired behavior, whether obstacle avoidance, target interception, waypoint following, etc. From this high-level path, a set of transition matrices are generated, including the Kalman gain K , the error covariance P , and the process and measurement matrices F and H . Also, the range and bearing from the trajectory positions to each target are precalculated. Now, the filter is ready to run. At each time step, the current measurements are subtracted from the predicted measurements, then run through the Kalman filter with matrices for that particular time step. This generates the state estimate deviation, which can then be added back to the trajectory to find estimated position. There are two main conditions where the filter stops and regenerates the trajectory. First, if the filter executes all trajectory points, it must calculate the next leg of the trajectory, including the matrices and expected measurements. Alternatively, if the number of targets changes, the filter must be recomputed. If a target is lost, the filter has no built-in way of compensating. Instead a measurement must be made up. However, since the trustworthiness of that measurement assumes a sighting, repeatedly making up measurements will hurt the accuracy of the estimate. Similarly, if another target is sighted, the filter cannot add it to the system and gain the benefit of the information it provides. For this to work, a separate process must keep track of the feature to ensure its measurements are not confused with the actively tracked targets. Depending on the feature extraction algorithm, this may or may not already be in place.

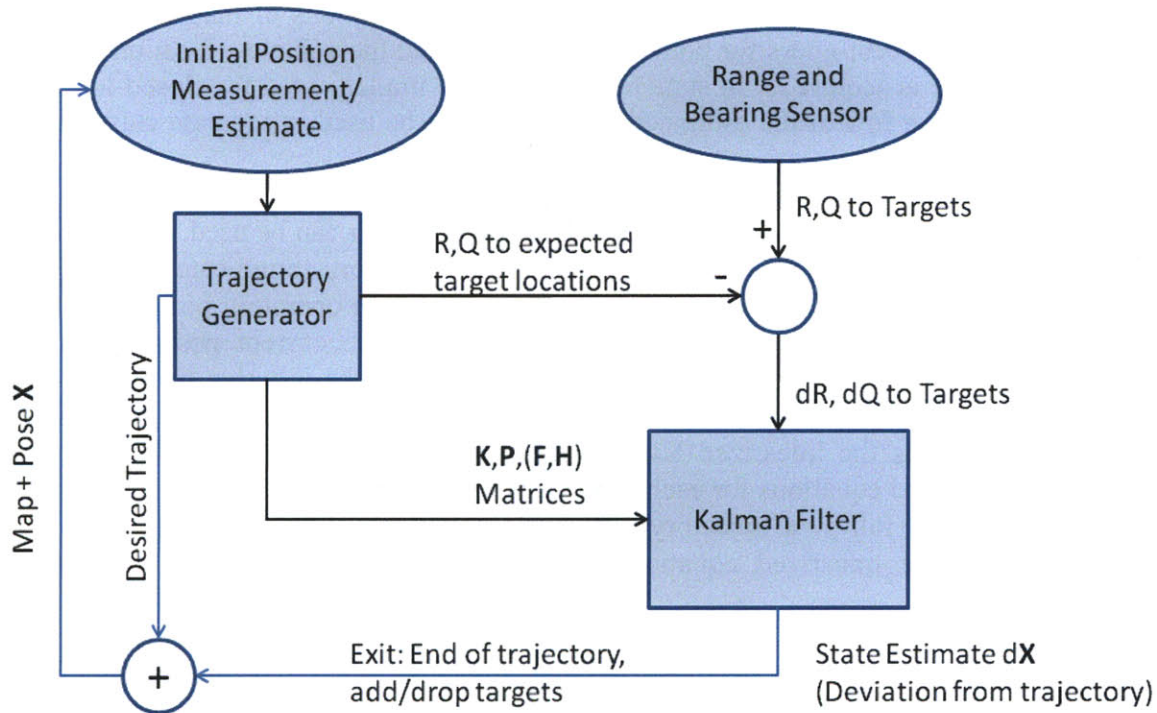


Figure 4-2: Operational block diagram for linearized Kalman filter process.

Overall, the approach offers the benefit of low underway computing costs, at the expense of disallowing a dynamically changing environment. For basic obstacle avoidance through point-like features, however, this may be sufficient. As with most observers, the job of the controller is made easier if the observer is an order of magnitude faster than the controller.

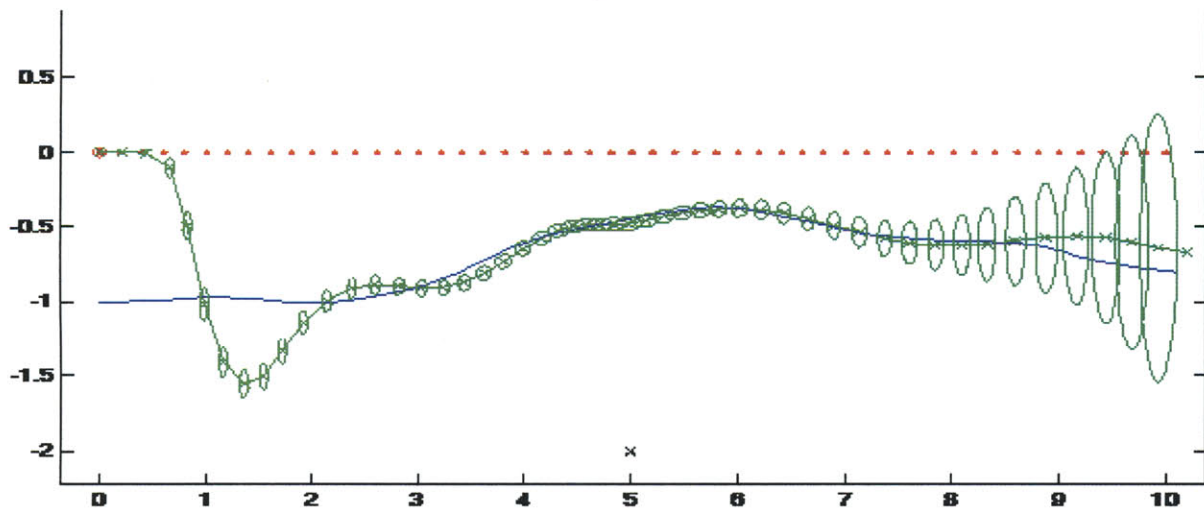


Figure 4-3: LKF estimate (green) tracking actual position (blue), some distance from planned trajectory (red). Measurement is a corrupted range-bearing of two targets at $x=5, y=\pm 2$. The estimate moves from overconfident initial incorrect estimate to actual path. Performance degrades as distance from targets begins to increase.

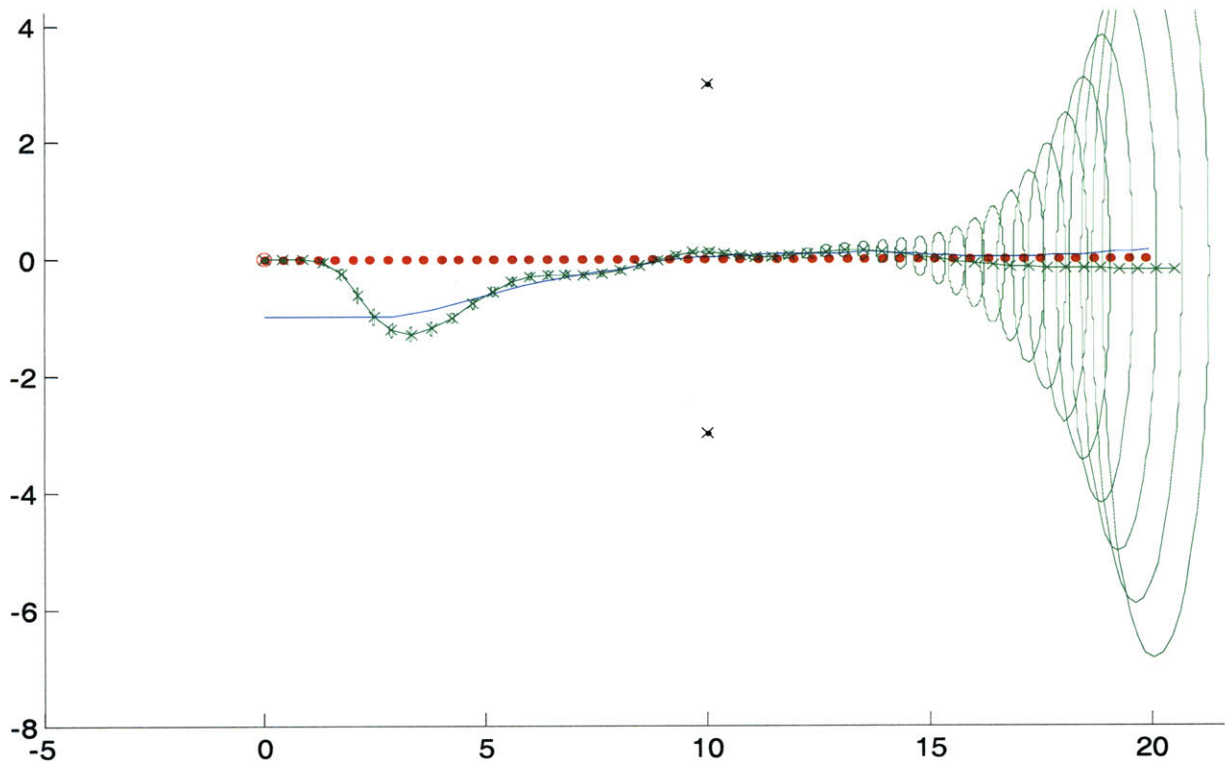


Figure 4-4: LKF estimator (green) tracking actual position (blue) and attempting to servo to planned trajectory (red). The estimate quickly tracks the actual position, allowing the controller to bring the system back to the planned trajectory. As before, the performance and certainty degrades as the vehicle moves away from the target.

4.2 Low-level Control Structure

The low-level control structure translates high-level behavior such as moving forward at a certain speed, or turning to a certain heading, into actuator outputs. The high-level commands are generated elsewhere. In other terms, the low-level controller attempts to fulfill a trajectory generated by the high level control. The first version low-level controller uses standard decoupled diving and heading models for underwater vehicles.

4.2.1 Decoupled Dynamics

In the baseline XAUV configuration (4 thrusters only), the horizontal and vertical motion planes are essentially decoupled. By virtue of the symmetrical layout, vertical thrusters cannot generate yaw or surge. Similarly, the forward thrusters cannot generate pitch or heave. In practice, there may be a slight cross-coupling from off-center thrust and drag loads from minor asymmetries. As an example, many symmetrical-looking vehicles will pitch slightly when moving forward through the water. If this coupling is light, it might be ignored for simplification of the control process. A more complicated surge/yaw control system could inject a feedforward component into the heave/pitch system to correct the disturbances it causes.

The general dynamic model is taken from Fossen and Yuh [20].

$$M\dot{v} + C(v)v + D(v)v + G = \tau$$

Because the system is modeled as decoupled, both the horizontal and vertical planes Expanding in the vertical plane only, the equation reduces to

$$M \begin{bmatrix} \dot{p} \\ \dot{z} \end{bmatrix} + C \left(\begin{bmatrix} p \\ z \end{bmatrix} \right) \begin{bmatrix} p \\ z \end{bmatrix} + D \left(\begin{bmatrix} p \\ z \end{bmatrix} \right) \begin{bmatrix} p \\ z \end{bmatrix} + G = \tau$$

For initial computation, the vehicle is modeled as a roughly elongate cylinder with cylindrical thruster pods. Again, this approach assumes that off-axis contributions are minimal

4.1.1. System Model

Initially, a very quick model and parameterization is used to get the system operational. Many of the assumptions are invalid at high speeds and under complex maneuvers. First, assuming the center of mass and buoyancy are at the midline of the vehicle, approximately 6" apart. Additionally, the vehicle geometry is approximated by a 25kg neutrally buoyant cylinder 8" in diameter by 48" long. The goal for this stage is to develop approximate SISO coefficients for roughing in control equations that can be tuned later.

Surge (X_1):

$$D_1 = \frac{1}{2} C_d \cdot a_{frontal} \cdot \rho \cdot v^2$$

$$c_d = 1 \text{ (flat cylinder, (30))}$$

$$m_a = \frac{2}{3} \pi r^3 \rho \text{ (half-sphere on front and rear)}$$

$$(m + m_a) \cdot \ddot{X} + D_1 \cdot \dot{X} = F_{thrusters}$$

$$\left(m + \frac{2}{3} \pi \rho r^3 \right) \ddot{X} + \frac{1}{2} (\pi r^2) \rho \dot{X}^3 = F_{thrusters}$$

The cubic term causes trouble in quick Laplace domain analysis. A quick least squares regression gives us a reasonable coefficient over a specified linearization region.

Table 4-1: Cubic velocity term linearization

range	V term
-0.5 to 0.5	$\dot{X}^3 \approx 0.153\dot{X}$
-1 to 1	$\dot{X}^3 \approx 0.606\dot{X}$
-2 to 2	$\dot{X}^3 \approx 2.412\dot{X}$

Assuming speeds do not exceed 1m/s for now, the coefficient is set at 0.6, with the understanding that this is overestimating the damping.

$$32.3\ddot{X} + 22.9\dot{X} = F_{thrusters}$$

Heave (X_3):

$$D_1 = \frac{1}{2} C_d \cdot a_{frontal} \cdot \rho \cdot v^2$$

$$c_d = 1.2 \text{ (cylinder, [30])}$$

$$m_a = \pi r^3 L \rho \text{ (circle thin section)}$$

$$(m + m_a) \cdot \ddot{X} + D_1 \cdot \dot{X} = F_{thrusters} - F_g$$

$$(m + \pi r^3 L \rho) \ddot{X} + \frac{1}{2} 1.2 (2rL) \rho \dot{X}^2 \dot{X} = F_{thrusters} - F_g$$

With $L = 1.22m, r = 0.157m, \rho = 1000kg/m^3$

$$1\ddot{X} + 1\dot{X}^3 = F_{thrusters} - F_g$$

Pitch (X_5):

$$D_1 = 2 \int_0^{\frac{L}{2}} \frac{1}{2} C_d \cdot a_{frontal} \cdot \rho \cdot v^2 \cdot x_5 \cdot dx_5$$

$$c_d = 1.2 \text{ (cylinder, [30])}$$

$$m_a = 2 \int_0^{\frac{L}{2}} \pi r^3 \rho x_5 \cdot dx_5 \text{ (circle thin section)}$$

$$m_a = 2\pi r^3 \rho \left(\frac{L}{2}\right)^2$$

$$(m + m_a) \cdot \ddot{X} + D_1 \cdot \dot{X} = F_{thrusters} - F_b$$

$$\left(m + \frac{1}{2} \pi r^3 \rho L^2\right) \ddot{X} + \frac{1}{2} \dot{X} = F_{thrusters} - (F_b + F_g) d * \sin(X)$$

$$1\ddot{X} + \frac{1}{2} \dot{X} = F_{thrusters} - 37 \sin(X)$$

Or for small angles:

$$1\ddot{X} + \frac{1}{2} \dot{X} + 37X = F_{thrusters}$$

Here there is an expected natural frequency of $\sqrt{37}$, and damping of 0.04 This damping may be small enough to warrant additional drag surfaces.

Yaw(X_6):

Yaw is the same as pitch for this approximation, without the contribution from buoyancy and ballast.

4.1.2 Closed loop depth and pitch

The combined depth and pitch control uses the two fixed vertical thrusters to maintain depth and pitch. This allows for variance and uncertainty in the ballasting and trim of the vehicle. Each thruster is capable of 12/14lbs of upward/downward thrust, providing a payload swing of +24/-28lbs.

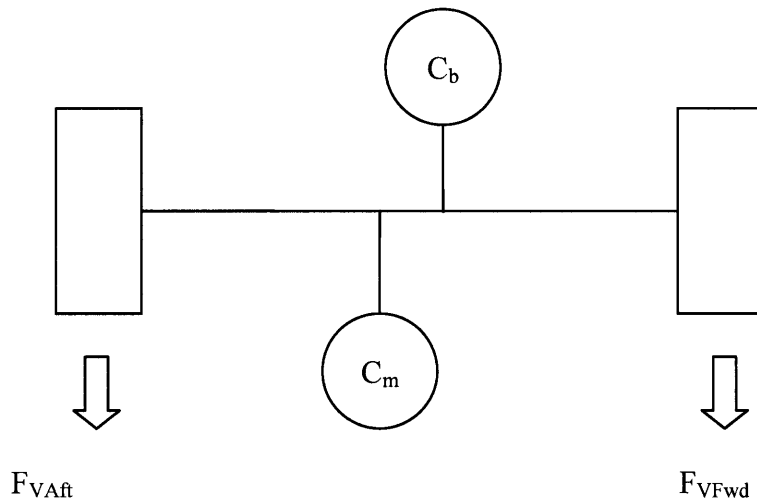


Figure 4-5: Depth and pitch model

The basic model assumes a moment of inertia, plus a point mass and point buoyancy at an unknown location along the centerline. The mass and buoyancy provide a righting moment to an unknown equilibrium position.

The model is derived in terms of downward thruster outputs F_{VFwd} and F_{VAft} . This is useful for direct control output of the thrusters. However, it is less obvious how the system will respond in terms of pitch and depth. A simple change of variables is used to allow SISO tools for the decoupled pitch and depth terms.

$$\begin{bmatrix} F_p \\ F_z \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} F_{VFwd} \\ F_{VAft} \end{bmatrix}$$

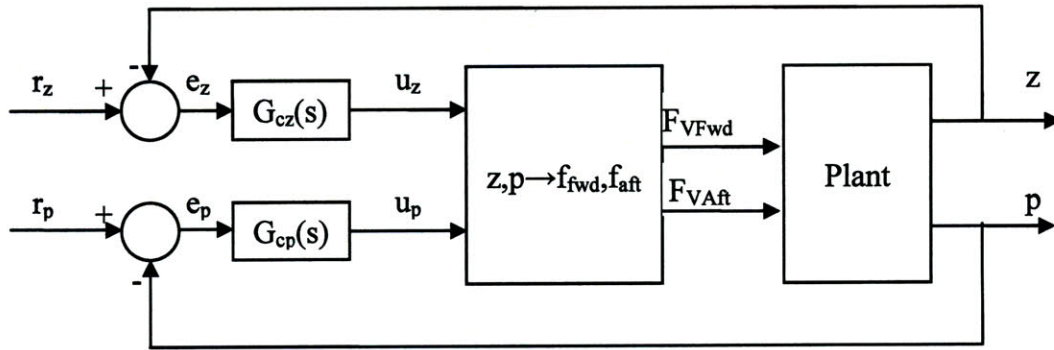


Figure 4-6: Pitch-Depth Control Topology. The controller uses a change of variables to control pitch and depth. Plant outputs are measured directly.

Both pitch and depth are effectively position control on a mass, or a double integration problem. The standard control approach for this type of system uses a lead-lag controller $G_{cz}(s)$ in depth and another lead-lag controller $G_{cp}(s)$ in pitch. The controllers are simulated in MATLAB, then converted to difference equations and implemented.

The decoupled depth model gives two real poles: one at the origin, and one at -0.7226 . Looking at the response with root locus and bode plots, using P-only control will give stable, if slow, performance. With a closed loop performance of 0.15Hz , the system will have about 45° of phase margin. The system will have decent steady-state behavior from the -1 decade/decade slope at low frequencies on the Bode plot (an integrator). The system will also have significant high-frequency noise rejection from the -2 decade/decade slope at high frequencies.

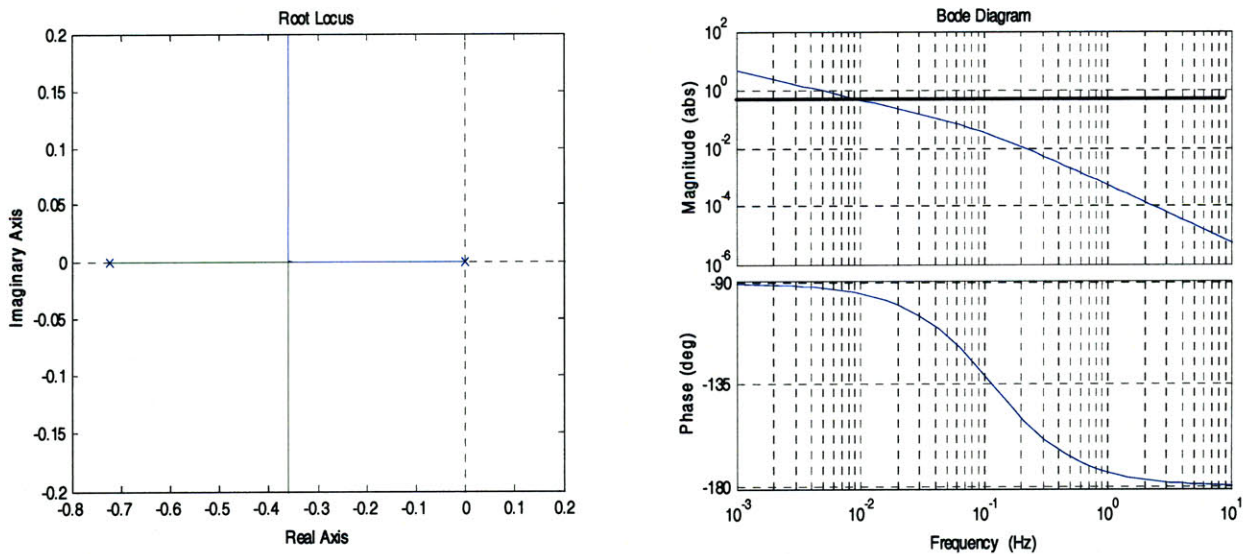


Figure 4-7: Uncompensated Root locus and Bode frequency response in heave

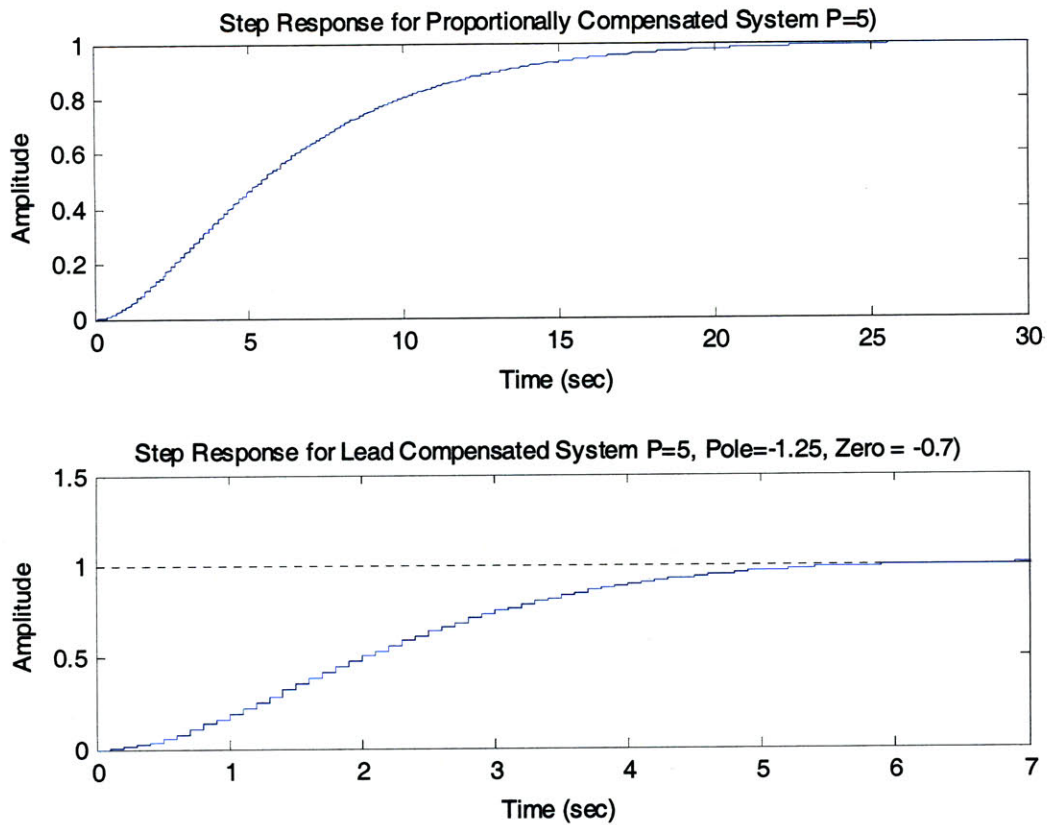


Figure 4-8: compensated time response in heave for two possible controllers

The pitch component proves significantly different, a result of the complex pair of poles representing the oscillatory behavior between the mass of the vehicle and the gravity spring caused by the righting moment.

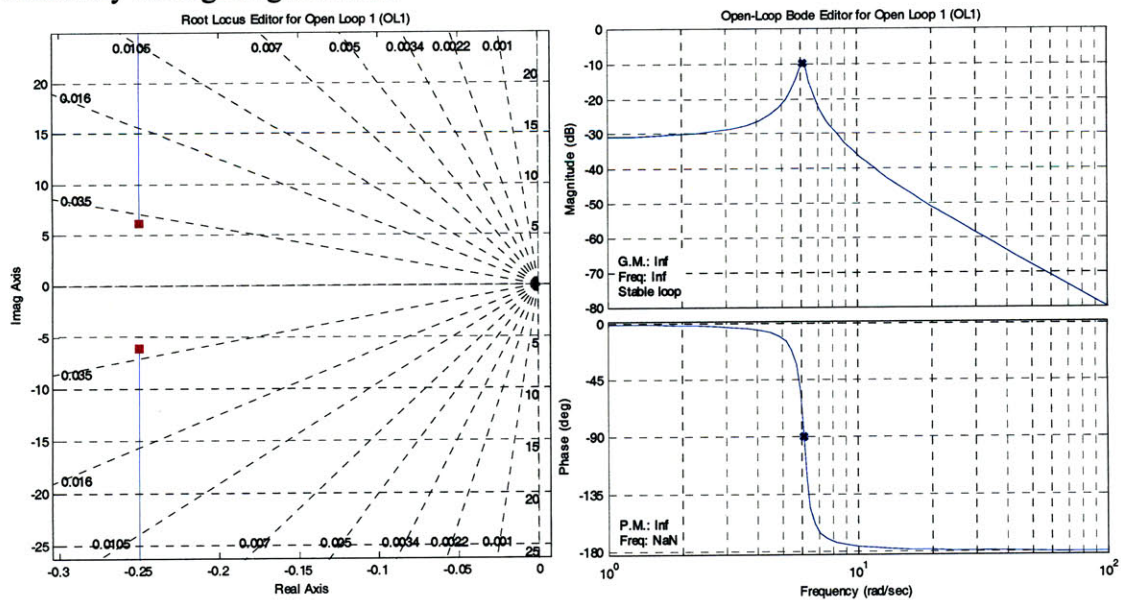


Figure 4-9: Uncompensated Root Locus and Bode system response for pitch.

A lead controller can be used to add phase to the response. As a first pass example, putting a zero at 10rad/s and a pole at 50rad/s, combined with a gain of 1 allows a crossover at 25rad/s, with phase margin of approximately 45 degrees. Care must be taken, however, not to saturate the output of the thrusters, or create a loop behavior that is too fast for the computer.

A more advanced controller can take advantage of the raw pitch rate output from the IMU, either as an input to a velocity inner loop, or as an input to a state-space model.

4.1.3 Closed-loop heading and speed

Taking advantage of the decoupled vertical and horizontal movement planes, the next step in controlling the robot is closed loop heading and speed control. This uses the two forward thrusters in a nearly identical fashion to the vertical ones: the average output creates forward motion, the differential output creates rotation.

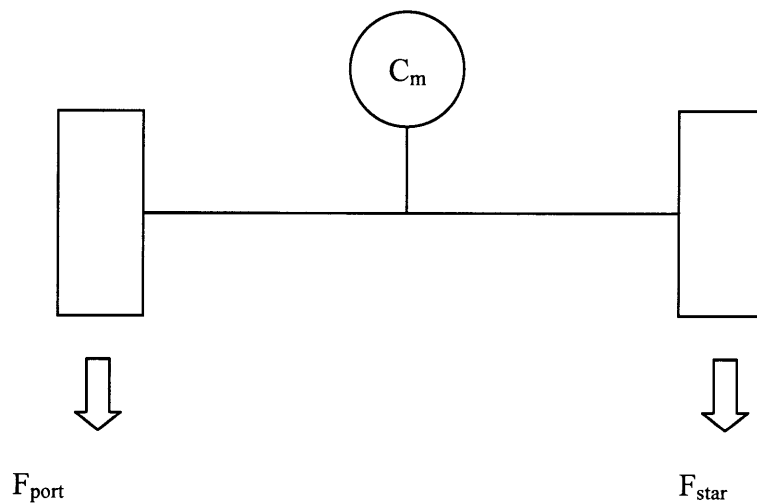


Figure 4-10: Heading and speed model.

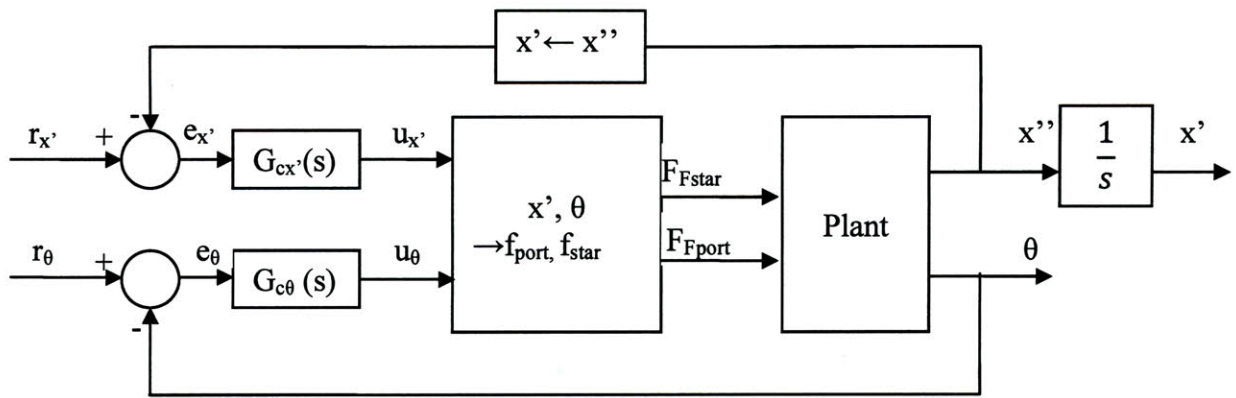


Figure 4-11: Heading-Speed Control Topology. Again, the controller uses a change of variables to independently compensate pitch and depth. Heading is measured directly, while velocity must be estimated from acceleration.

In this case, the geometry is similar to the previous case, heading, like heave, is a position control on a mass, or double integration problem, while speed is a single integration problem. Because there is no reliable speed measurement at this point, an open loop approach is taken, simply adding a scaled value to the thruster output based on the desired speed.

The yaw is also well-behaved, providing reasonable, although not fast behavior when uncompensated. A controller similar to heave may easily be implemented.

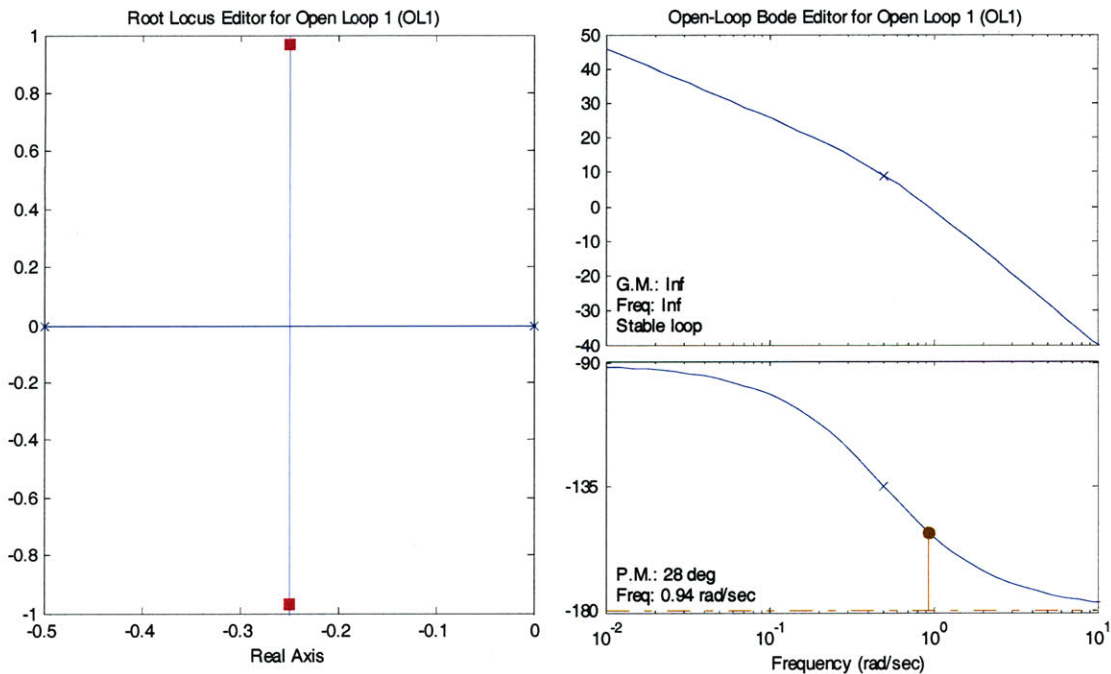


Figure 4-12: Uncompensated system response for yaw

4.1.4. Azimuthing Thruster Control

The control design space for azimuthing thrusters is greatly increased when compared to a fixed-thruster design. At the fast end of the control timescale, the time required to actuate a thruster introduces limitations in the performance. At the long end, the ability to direct thrust in the exact direction required allows increased mission endurance.

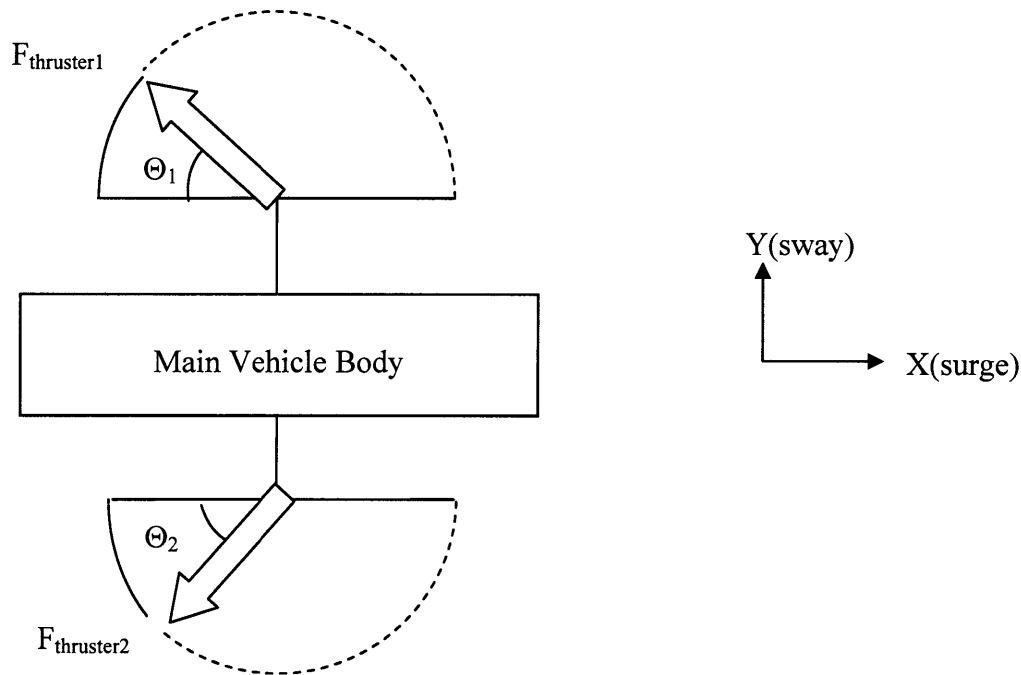


Figure 4-13:Azimuthing thruster layout, top view. Two thrusters are positioned at a distance d from the centerline.

The planar nature of the azimuthing layout allows the pitch/depth half of the decoupled pitch/depth-surge/yaw approach to remain unchanged. With the thrusters locked at 0° , the surge/yaw can remain the same as well. The main advantage of the design, however, is that sway can be controlled as well, with only a single pair of thrusters.

The forward kinematics required to calculate the resultant forces and moments are relatively straightforward. The individual x -components sum to form the surge force. The differences in the individual y -components form the sway force. The difference between the x -components form the yaw moment when scaled by the distance from the midline, d .

$$F_x = F_1 \cos(\theta_1) + F_2 \cos(\theta_2)$$

$$F_y = F_1 \sin(\theta_1) - F_2 \sin(\theta_2)$$

$$M_z = (F_1 \cos(\theta_1) - F_2 \cos(\theta_2)) * d$$

The inverse kinematic problem looks at where to point each thruster to achieve a desired vehicle response. This can be a simple minimization of total thrust, or a complicated optimal or model-predictive control problem. Fossen has suggested a number of relevant techniques for the optimization [63]. For applications where the transit delay of the thruster prevents the execution of the desired trajectories, one of these more complicated approaches is likely required. Here we will first examine a simple scheme based on minimum sufficient thruster output, then examine conditions where this approach may not be sufficient.

Looking quickly at the problem, it is evident that there are multiple equivalent solutions. From a battery endurance standpoint, we desire a solution which uses the minimum energy, which is loosely equivalent to thrust output in the steady-state problem. Looking at the contribution of each thruster:

$$F_{1x} = \frac{1}{2}F_x + \frac{1}{2}\frac{M_z}{d}$$

$$F_{2x} = \frac{1}{2}F_x - \frac{1}{2}\frac{M_z}{d}$$

$$F_{1y} = \frac{1}{2}F_y$$

$$F_{2y} = -\frac{1}{2}F_y$$

Thus:

$$F_1 = \frac{1}{2}(F_x + \frac{M_z}{d})/\cos(\theta_1) + \frac{1}{2}F_y/\sin(\theta_1)$$

$$F_2 = \frac{1}{2}(F_x - \frac{M_z}{d})/\cos(\theta_2) - \frac{1}{2}F_y/\sin(\theta_2)$$

Now the expression for the thruster forces is in terms of the desired X and Y forces, desired moment M, as well as the angles of the thrusters. Unfortunately, there exist certain singular angles for which the required thrust is infinite.

One coarse solution would scale back the scope of the solution to a separate rotating phase and translating phase. This approach is motivated by the insight that in an operational environment, the vehicle will face greater difficulty fighting to hold position in a cross-current than to maintain a heading in a off-angle current. In a 2 knot current, the earlier hydrodynamic analysis indicated that essentially all control authority, or 220N of 280N will be used to maintain position. If the vehicle is at a 45° to the current, the Munk moment on the vehicle will be [64]

$$M = \frac{1}{2} U^2 \sin(2\theta) * (m_{22} - m_{11})$$

Or approximately 62Nm. With the thrusters approximately 0.6m apart, and positioned for maximum moment, the required thrust output is 100N, or less than half of the control authority required for translation. For the translation segment, the thrusters are directed directly in the vector sum of desired X and Y velocities. For the rotational segment, a new differential component is added to the thruster angles, creating the moment. Assuming the perturbations in

the translation resultant are small, a differential thrust may also be added. The smaller this differential, the greater weight is placed on the translation phase.

This approach hasn't yet taken into account the time it takes the thrusters to move from their starting position to the final, desired position. Ongoing research with the Odyssey IV is looking into applying optimal control strategies to rotating thrusters, taking into account this transit time. Rather than a full analytical approach, a delay analysis can be used to determine the limits at which this control method is no longer viable. Assuming that there is a finite delay T , the relation between the input and output is

$$y(t) = u(t - T).$$

The Laplace transform results in a transfer function of [59 p. 155]

$$\frac{Y}{U}(s) = e^{-sT}$$

Looking at the Bode Response, the magnitude is unity across all frequencies. The phase, however, decreases linearly with frequency, appearing exponential on the semi-log phase plot.

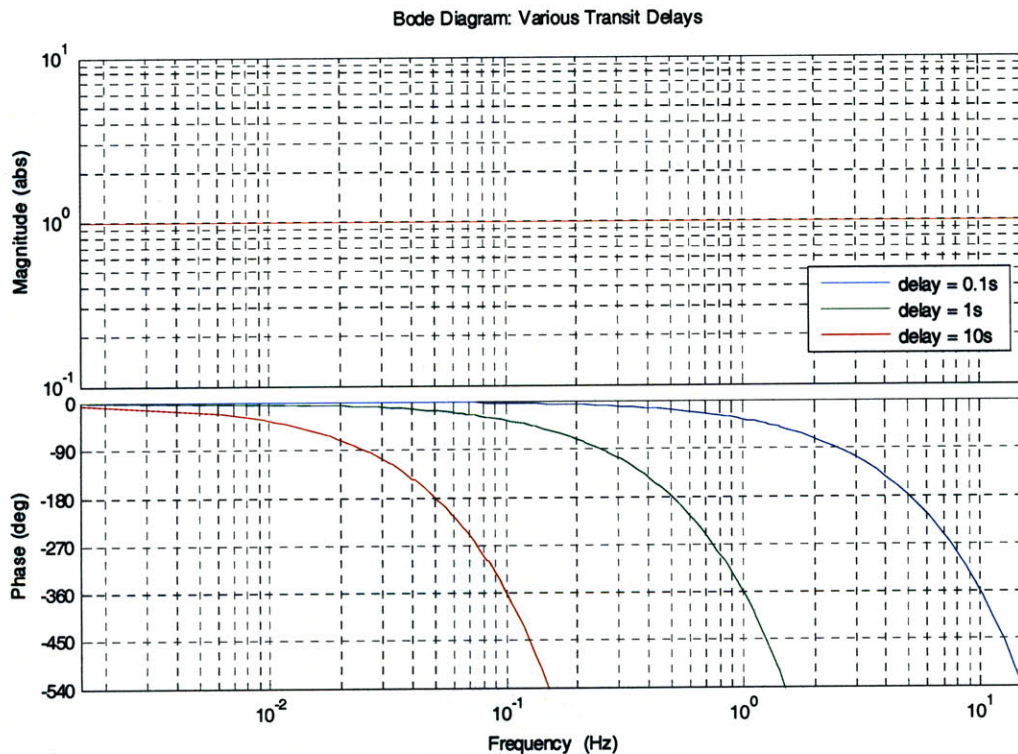


Figure 4-14: Phase degradation from various transit delays. As phase quickly rolls of with increasing slope, compensation quickly becomes impossible

As a coarse estimate, adding more than 270° of phase to the system through compensation will be problematic, as in the case of multiple lead controllers with high α p-z separation. From this number, the maximum bandwidth of the system is 10Hz for the 0.1sec delay, 1Hz for the 1sec delay, and 0.1Hz for the 10 second delay. The actual bandwidth limit may be an order of magnitude lower, depending on the compensation approach used.

Applied to the thruster system, the servos have a maximum angular velocity of 0.14-0.19 sec/60°, depending on the applied voltage. For a full reverse, or 180° transit, the transit delay will be between 0.42 and 0.57 seconds. This puts the bandwidth upper limit at approximately 2Hz. Again, the actual performance may be much lower, assuming a required dwell time at each location.

4.2 Software

The control software is the method by which control schemes and basic function regulation occur. Initial testing and evaluation is done with Windows programming, initially in VB for low level debugging. This allows access to the Windows DAQ board drivers. The DAQ hardware registers are directly accessible through a windows dynamic link library. The ultimate implementation might be a dual-boot Linux/MOOS and MATLAB xPC Target scheme, NI LabVIEW machine, or raw MATLAB. In any case the core lean concepts of chapter 2 are most directly applied to the future software.



Figure 4-15: Vehicle software running on the XAUV, viewed on a laptop through remote desktop

4.3.1 VB Interface

The Visual Basic interface collects and logs sensor and control data. It also computes the difference equations for simpler control schemes, and so can be used for autonomous vehicle operation.

The interface can operate in two main modes: open-loop and closed-loop. The open-loop mode uses one of several human input methods to generate the output commands. At the simplest level, the user enters the 10-bit number corresponding to the D/A converter output. This value is then sent directly to the DAQ with minimal processing. This is useful in the initial stages of development, where interface hardware is being debugged. At the next level of complexity, a human interface device, in this case joystick, has its axes directly mapped to thruster outputs. A separate scaling and saturation panel can be used to reduce the maximum output, while a display confirms the values that the joystick is commanding. This capability is useful when the operator would like to move quickly but smoothly through a range of values, a task which would be otherwise difficult with a text entry interface. A reasonable alternative would be to use slider bars. Several commercial interfaces use this approach. [50] At the highest complexity, the computer mixes the joystick channels to provide a more intuitive control for actual driving. The implemented method is the standard tank steering approach of using forward and reverse on the joystick, $Y_{joystick}$, to generate additive thrust from the forward thrusters F_{left}, F_{right} , while using left and right on the joystick, $X_{joystick}$, to create differential thrust, thereby turning the vehicle.

$$\begin{bmatrix} F_{left} \\ F_{right} \end{bmatrix} = \begin{bmatrix} -a & b \\ b & a \end{bmatrix} * \begin{bmatrix} X_{joystick} \\ Y_{joystick} \end{bmatrix}$$

In this case, the thrusters are assumed to put out equal thrust at full power. Any difference in output power can be adjusted with a simple scaling factor.

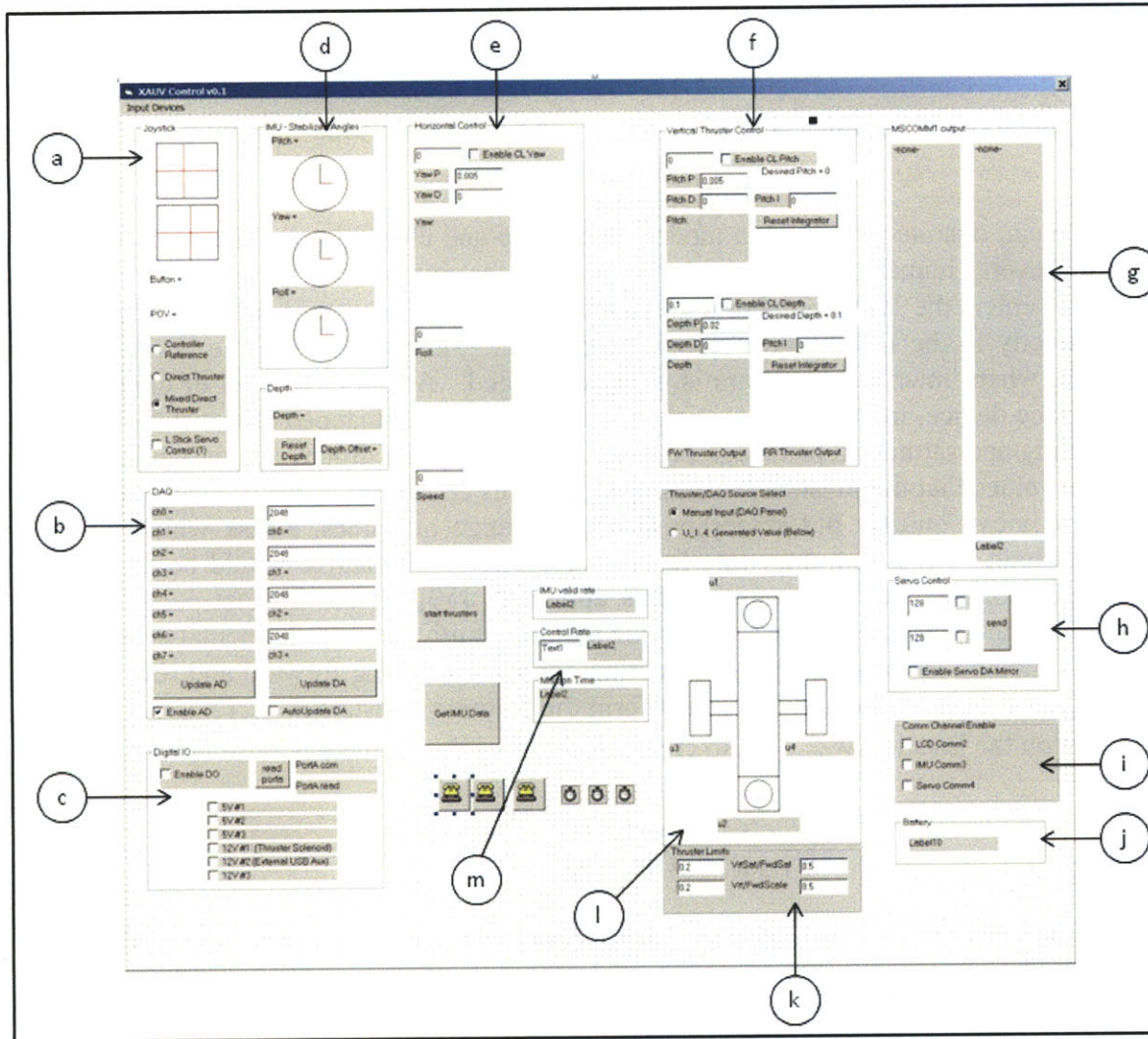


Figure 4-16: VB control software display. (a) joystick input; (b) DAQ: D/A control and A/D display; (c) DIO: Power switch board; (d) IMU orientation and control reference display; (e) Horizontal closed loop thruster display and parameter input; (f) Vertical closed loop thruster display and parameter input; (g) debug display area; (h) servo output control; (i) serial port enable; (j) battery level display; (k) thruster saturation and scaling input; (l) thruster output visualization; (m) operating frequency input.

Although mixed open-loop control is useful for manually piloting the vehicle, extended missions or autonomous tasks will benefit from the disturbance rejection that closed-loop functionality provides. With the basic configuration of the vehicle, this takes the form of heading/speed and pitch/depth servoing. This is controlled from another section of the VB application. In the closed loop vertical or thruster control, there are boxes for inputting control gains in the standard PID form. The integrator is constructed with both anti-windup and reset capabilities, and the derivative is a first-order backwards looking approximation. As an alternative to the PID topology, the difference equations can be directly accessed in the code with minimal difficulty. In either case, the outputs and key internal values, such as error and control effort, are displayed. The control effort outputs are then subject to the same saturation and scaling controls as the open-loop control, described above.

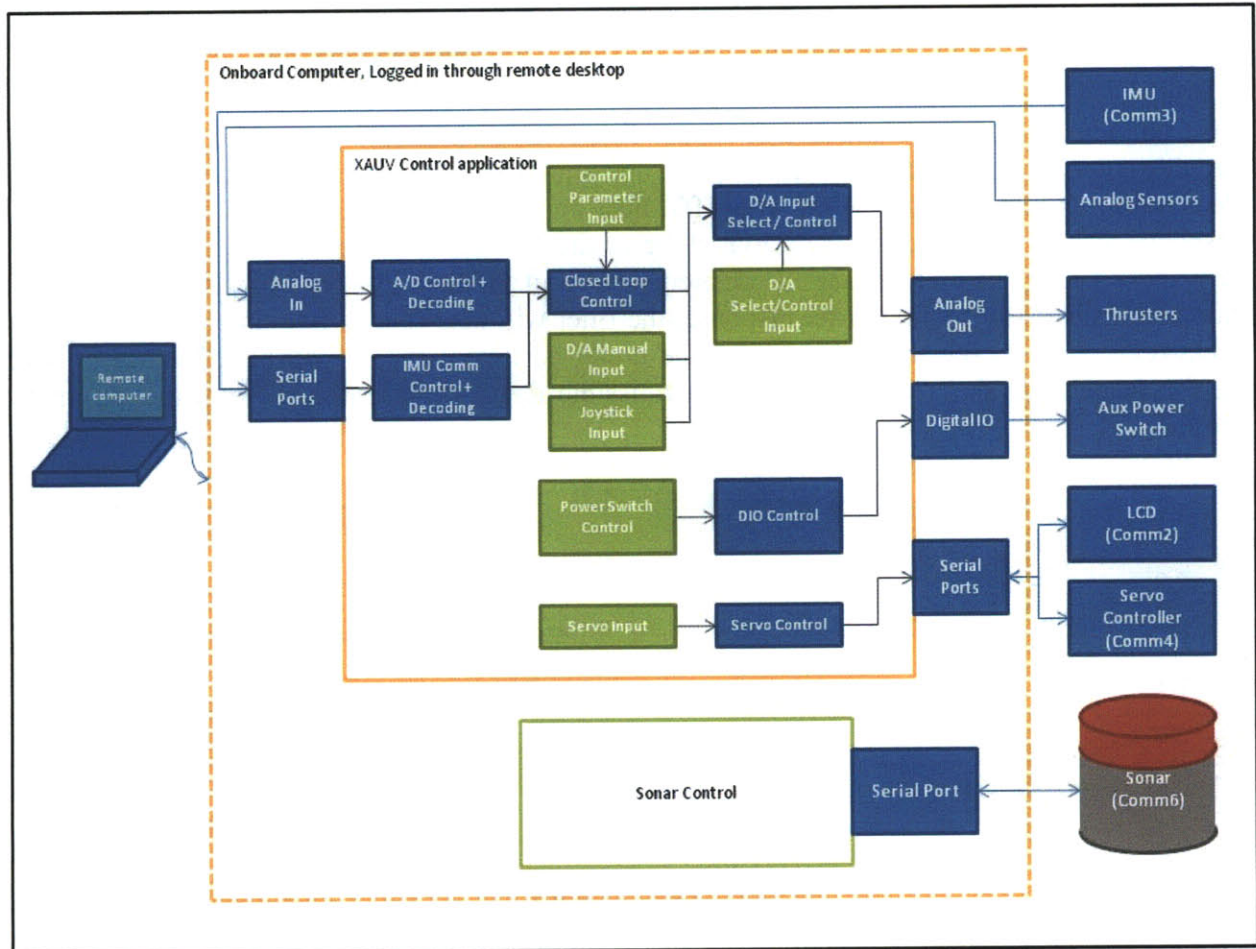


Figure4-17: Interface block diagram

4.3.2 Slew Limit

Slew limiting is a technique used to prevent damage to sensitive components, by limiting the rate of change of current through them. The Tecnydyne thrusters require that the control signal does not change faster than 10ms/V. The control input has a range of $\pm 5V$, giving a minimum rail to rail swing time of 0.1s. In the discrete (clocked) system, the slew limit is simply the max change in output over the clock period. This can be implemented in software as a bound on the next output Y_0 , equal to the present value Y_{-1} plus or minus the slew limit multiplied by the clock period Δt .

$$Y_{-1} - (SlewLimit) \cdot \Delta t < Y_0 < Y_{-1} + (SlewLimit) \cdot \Delta t$$

In pseudocode, this looks can be implemented as a conditional saturation:

```

 $Y_0 = f(u_0, u_{-1}, \dots, y_{-1}, y_{-2}, \dots)$ 
If  $(Y_0 > Y_{-1} + (SlewLimit) \cdot \Delta t)$  Then  $Y_0 = Y_{-1} + (SlewLimit) \cdot \Delta t$ 
If  $(Y_0 < Y_{-1} - (SlewLimit) \cdot \Delta t)$  Then  $Y_0 = Y_{-1} - (SlewLimit) \cdot \Delta t$ 
DA Channel =  $Y_0$ 

```

This method of slew limiting only generates a series of quantized steps of a size consistent with the slew limit. For very short clock periods (high frequency control loops), this is insignificant.

At the other extreme, very large periods will still create large step outputs to the thrusters. In this case, a separate interpolation timing stage or analog back end is required.

The simplest analog back end is a low-pass filter. Even a simple first-order RC circuit will provide a degree of protection from sharp voltage gradients. Because a first-order response has a decidedly non-linear shape, a few tradeoffs must be made. In order for the total time of the response to be roughly equal, the initial slope of the first order response will be approximately two and a half times greater than the desired slew limit. If the ideal response is imposed as the maximum slope for the first order response, the filter will take four to five times longer than the ideal response to reach the full value. The preferred time constant will be a balance of need performance from the settling time and permissible danger to electronics from the maximum slope.

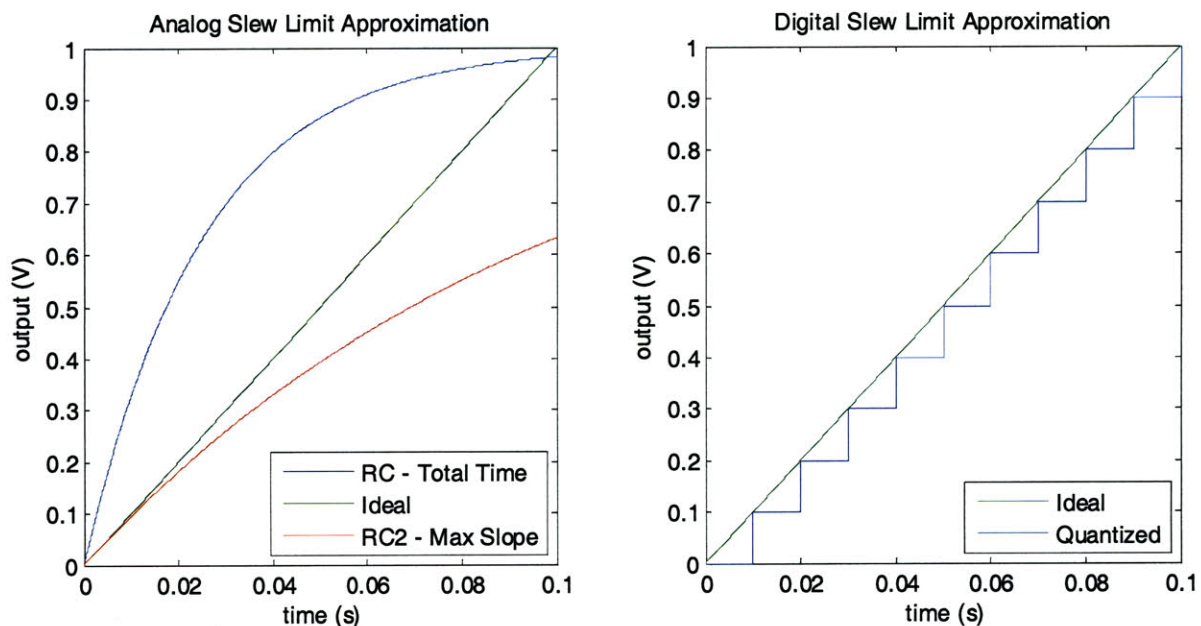


Figure 4-18: Simple analog and digital slew limit implementations

4.4 Summary

The control system is developed as a hierarchy of possibilities, ranging from pure human control to pure autonomous control. Beginning at the high level, several notions supervisory or separated high-level control are considered. Also, a few notable efforts to automate high-level control and incorporate them within the lower-level control loops are briefly discussed. Next, the Linearized Kalman Filter is explained as a possible high-level controller capable of updating its own map. Next, the baseline low-level control loops are discussed, beginning with an approximation of the equivalent SISO parameters, and the generation of rough control equations. Then, a brief aside is taken to examine azimuthing thruster control, where the approach is taken to prioritize translations over rotations. Last, the software interface is described, discussing both the existing driver software and custom GUI, as well as the approach for slew limiting.

Chapter 5

Experimental Results

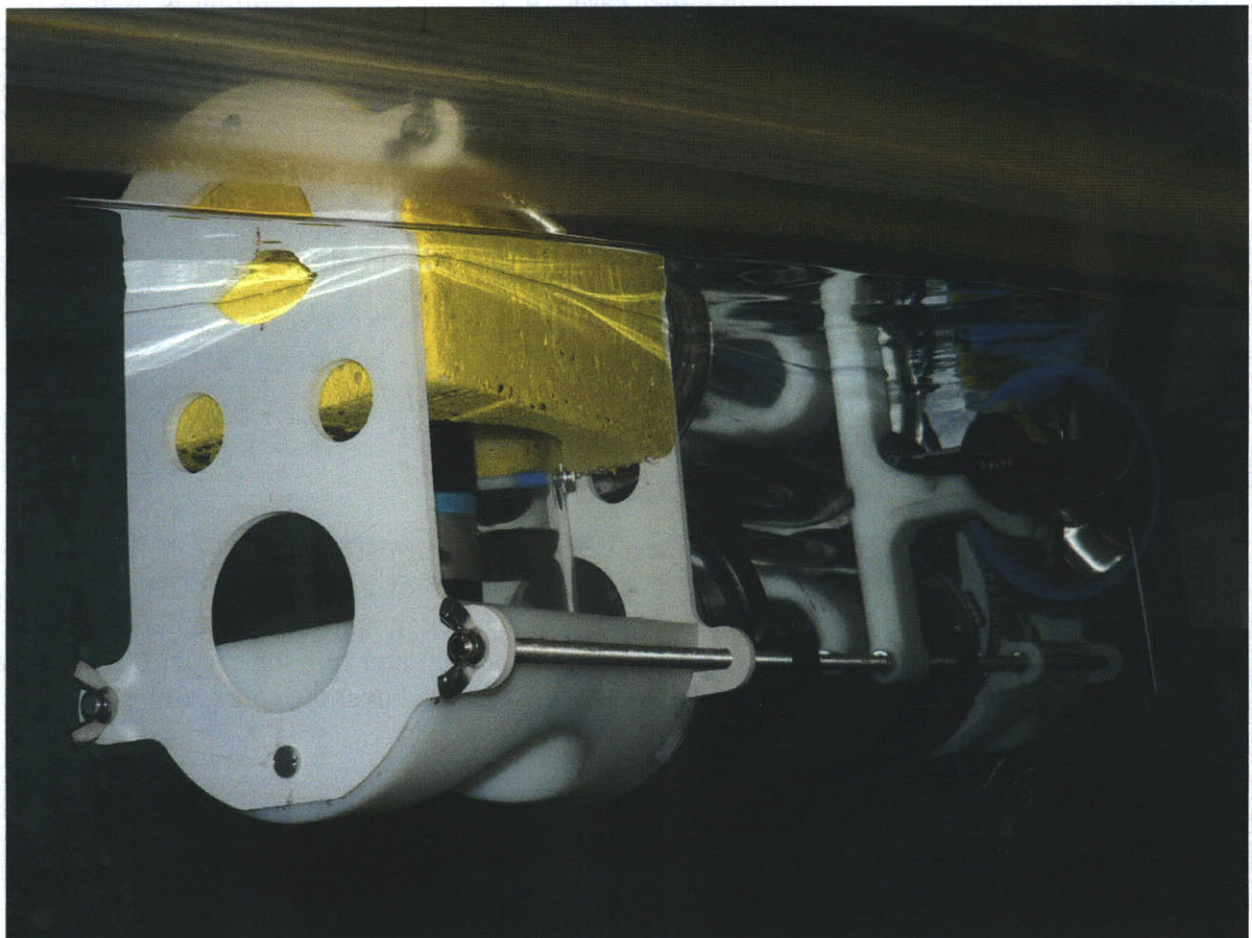


Figure 5-1: XAUV during in-water tests

5.1 Leak Testing

During testing, the vehicle was subjected to simulated and actual depths to evaluate the integrity of the waterproofing. The purpose of these tests was twofold. First, the tests identified and corrected any leaks before sensitive components were installed. Second, the tests provided a degree of leak-proof assurance where the vehicle does not need to be monitored constantly for water ingress. The most basic test involves submerging the pressure housings for a short period of time. Although this may not be sufficient to identify small leaks, large leaks can be quickly detected, if not identified. By packing likely leak areas with paper, the location of the leak can be identified. The paper absorbs the water, and prevents it from pooling somewhere other than the original leak location. As long as the paper does not become saturated with water, a reasonable accuracy may be expected. On the XAUV, a few connectors were identified as improperly installed at this level.

After the ‘dunk’ test, smaller and more stubborn leaks can be identified by simulating depth. Submerging the components to 25 or 50ft is impractical in terms of available water and observability of the components while at this depth. Instead of generating a large external pressure through depth or pressure tank, the internal pressure is reduced with a vacuum pump. The pressure differential is the same in either case. A standard ¼” NPT fitting is used as this pressure testing port or with a plug as a purge/vent port. For proof testing of the seals, the component under vacuum can be placed into a small basin of water, sufficient only to cover the component. This technique proved useful in detecting unexpected leak paths. Specifically, two wires had small cuts in the insulation, which allowed water to seep in between the copper strands, and bypass the cable gland seal. This was fixed by replacing the wires and constructing an epoxy block at the inside connector. Both the main housing and battery housing held a steady pressure at -450mmHg, or 20ft H₂O pressure differential in air. This approached the limit of what the vacuum pump plumbing could hold.

5.2 Data Capture

The primary products for the in water tests were scanning sonar data and IMU orientation data. The IMU data was recorded through the main XAUV control application. The sonar was monitored in real-time and recorded through the Imagenex graphical user interface. The Imagenex GUI also allows basic file playback, noise filtering, and full sonar control. Nearly 200 Mb of CSV data and over an hour of sonar data was recorded over the course of testing.

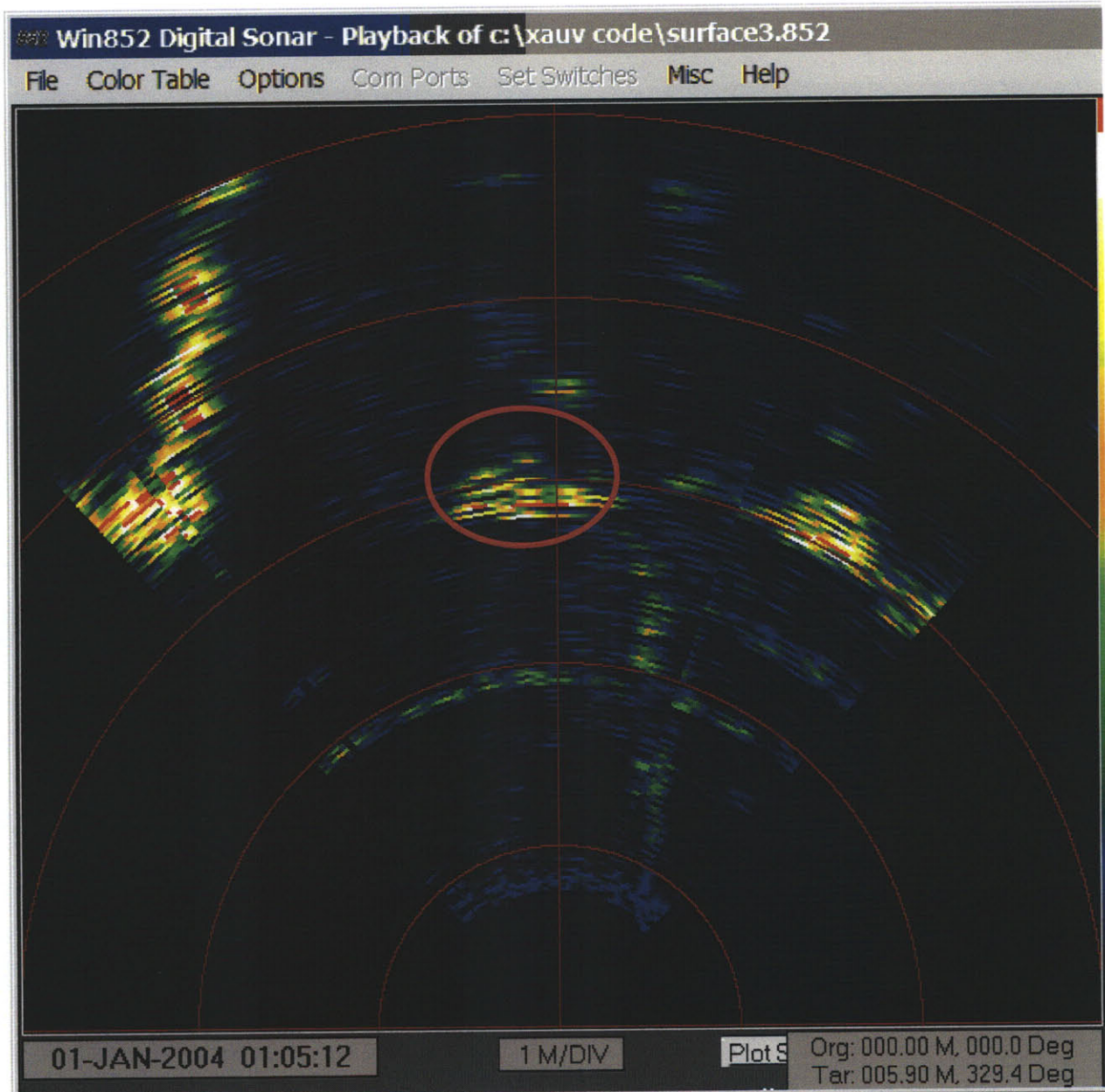


Figure 5-2: Scanning sonar image of Brendan Englot, highlighted by the ellipse. A scanning discontinuity (area of update) is visible at ~15 deg right of vertical from slight movement of the vehicle , but the overall image quality is good.

The image quality of the sonar is not sufficient to identify small objects by their shape. However, it is capable of identifying large objects which would prove to be a navigation hazard. Additionally, the layout of the sonar on the vehicle did not appear to provide interference with the measurements.

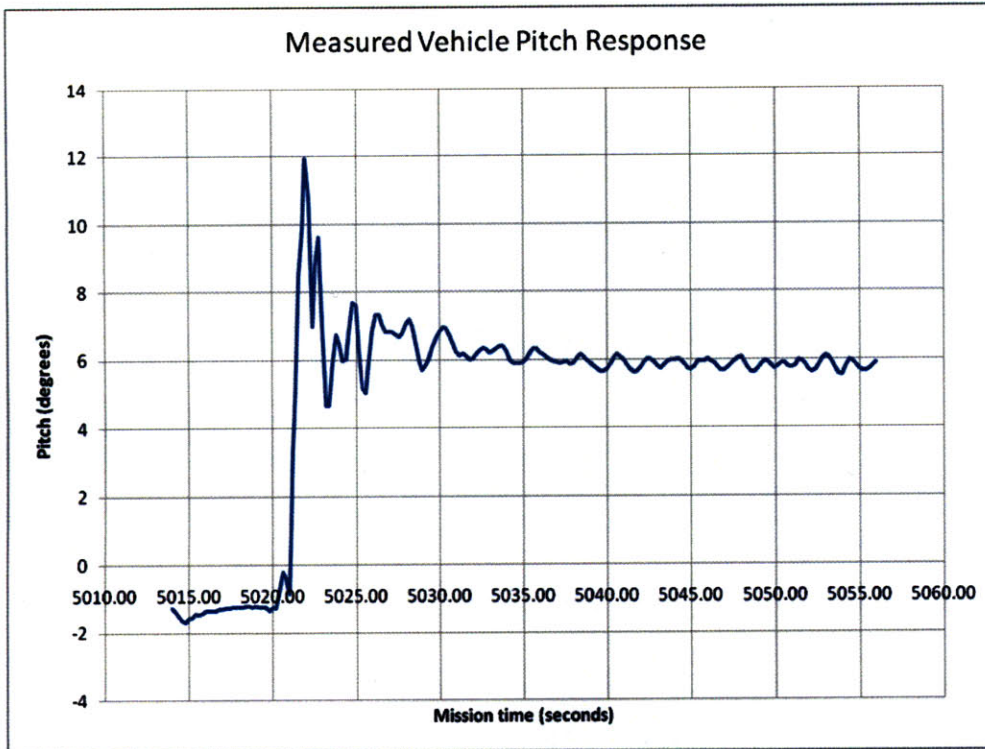


Figure 5-3: Example data capture: vehicle recovery from a pitch disturbance

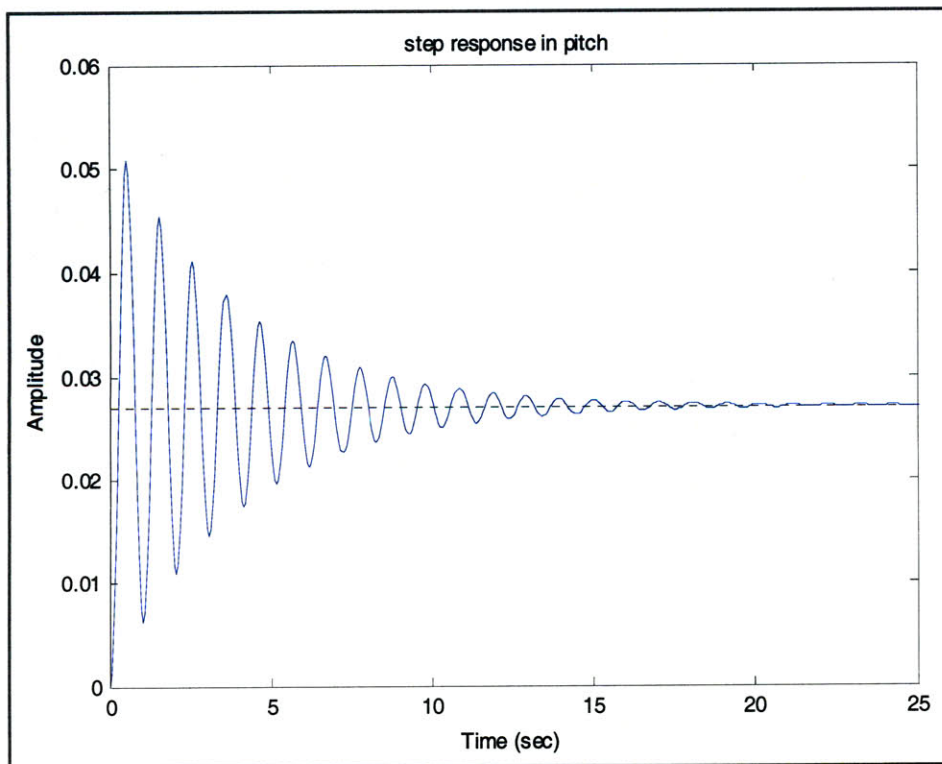


Figure 5-4: Calculated step response exhibits similar behavior to observed measurements, including $\omega_n \sim 6$ rad/s and $T_s \sim 10-15$ seconds

5.3 Sonar Mosaic/IMU correction

While recording sonar data, the vehicle would typically move faster than the sonar head. The viewer, understandably, only updates the area currently being measured. Because the scanning speed is loosely equivalent to the vehicle maneuvering speed, using this data mosaic may lead to inaccurate interpretations of the environment. A script was developed to read the recorded sonar files and mosaic the beam returns using the IMU orientation data.

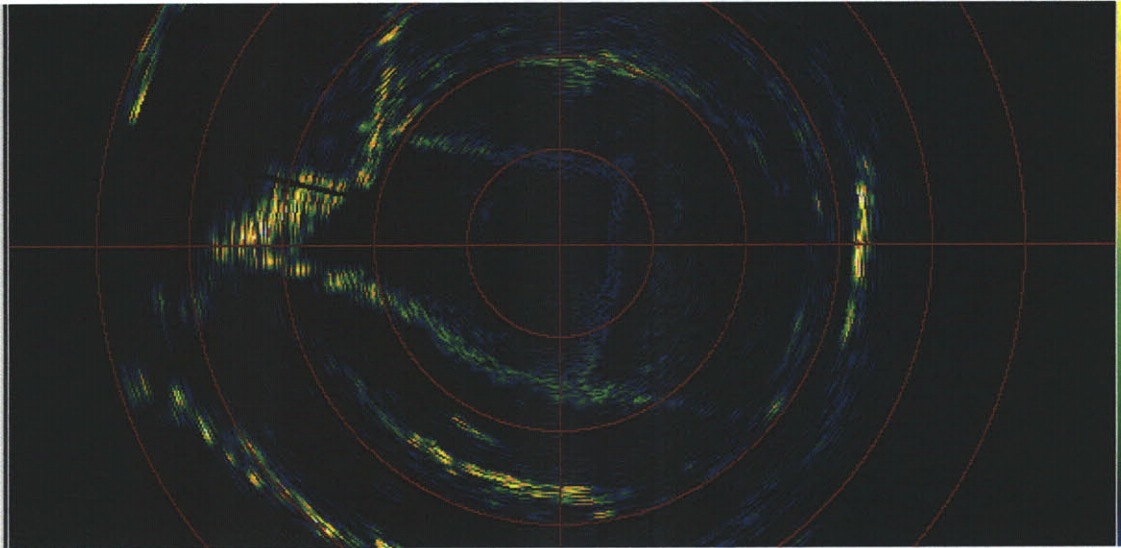


Figure 5-5: Distorted image of tank geometry. Moderate motion of the vehicle creates a misleading image of the surrounding environment.

The script works as follows. First, the file is loaded, and the first beam of interest is loaded. The beam data also carries a time stamp, direction, and scaling parameters. Next, the IMU data is read in. Because the two data files were synchronized, but not running at the same speed, a search function is used to find the closest orientation data to the beam data. The beam is orientation is then corrected for with the IMU data. In the simplest case, only roll or pitch is used for a vertical or horizontal slice, respectively. The same approach is easily used to provide additional correction for all three axes, plus depth. With any transformation approach, the new beam location is overlaid on the map to create the mosaic.

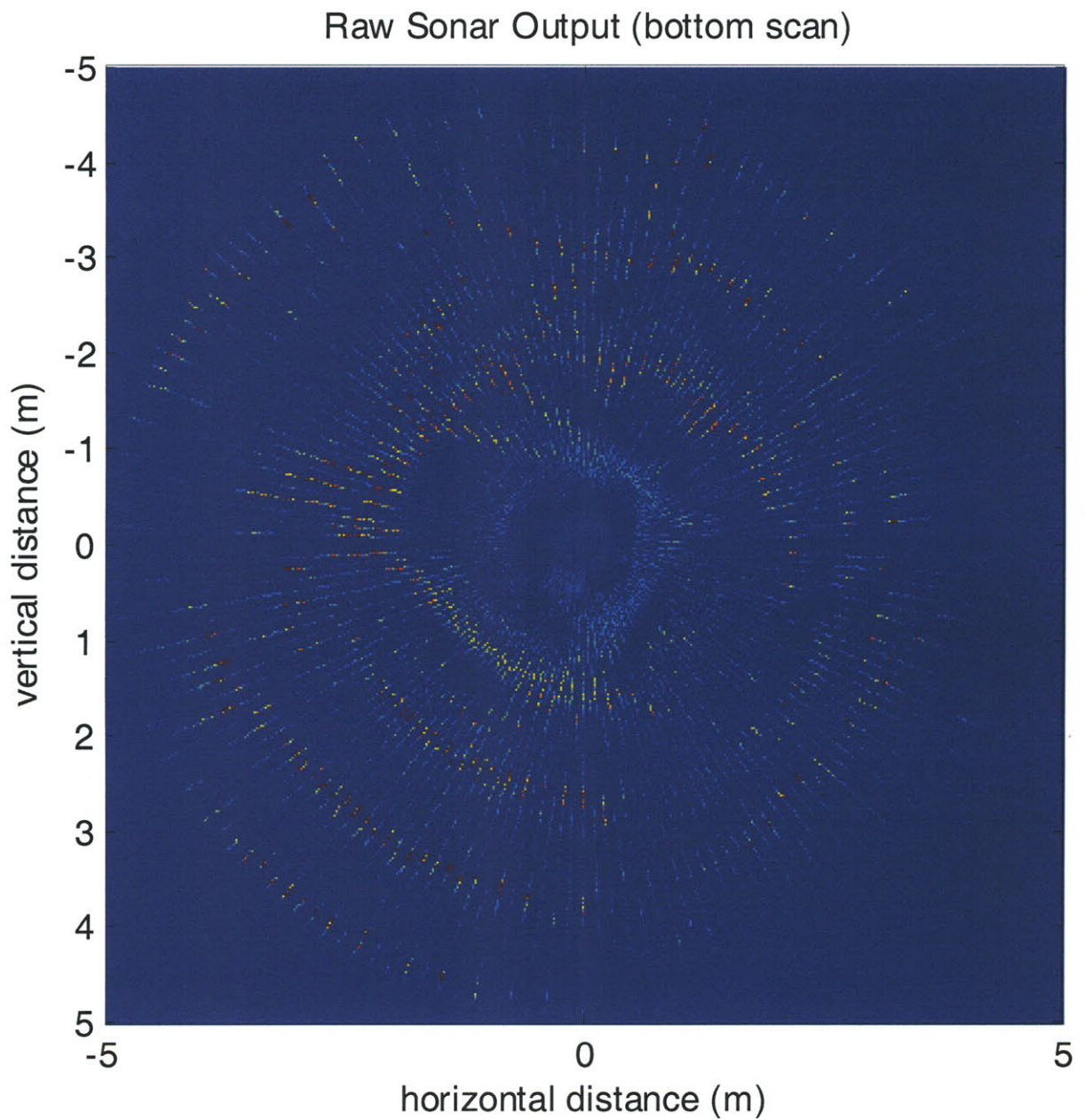


Figure 5-6: Raw sonar data plotted in MATLAB. In addition to the obvious offset from vehicle roll, there is also disagreement between successive scans of the sonar. This is the result of the time required for the sonar to scan a full rotation.

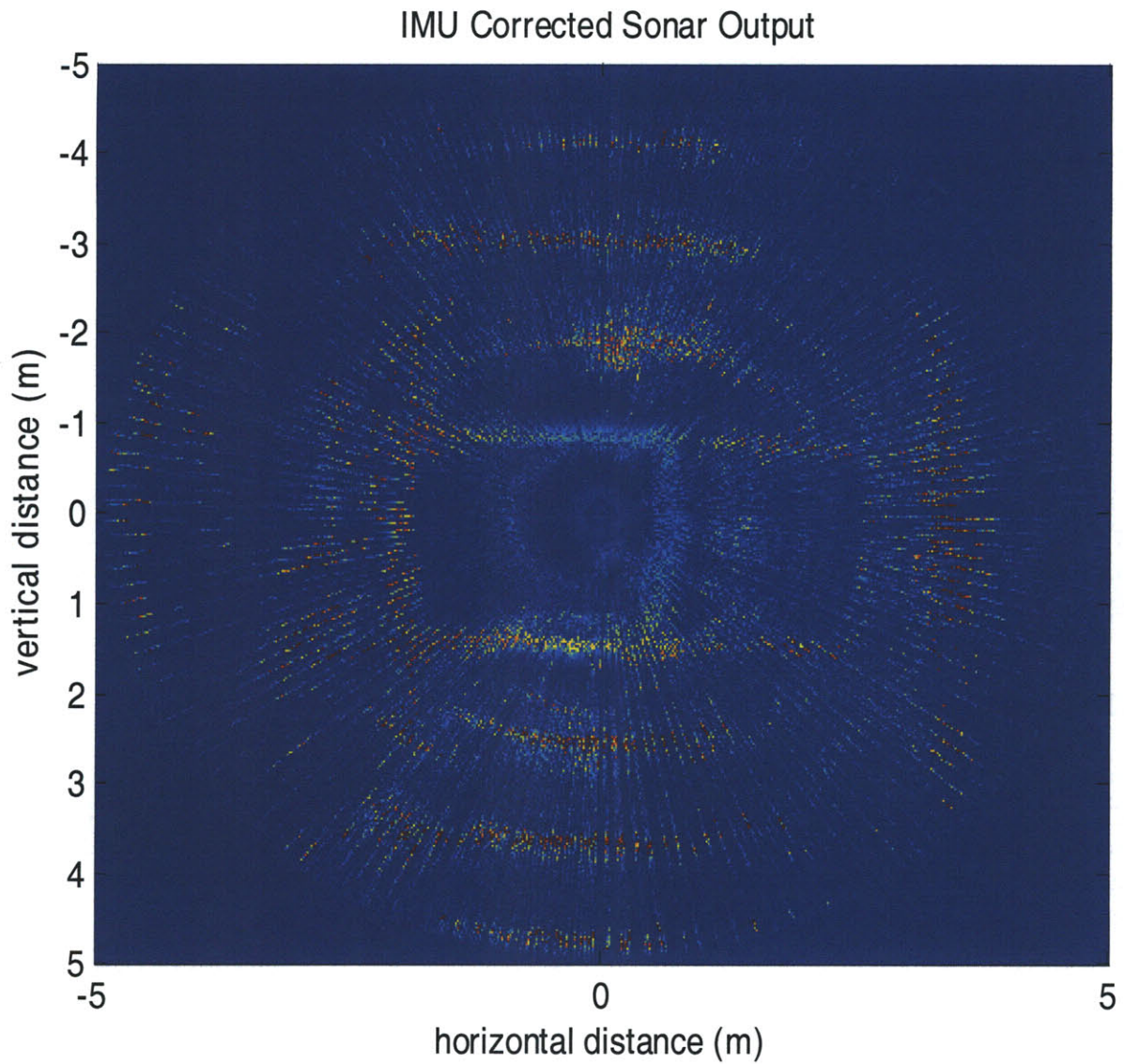


Figure 5-7: Corrected sonar data, plotted in MATLAB. The sonar data is corrected for roll on a beam by beam basis. The cross-section of the MIT Towing tank is seen as the inner-most rectangle of strong returns.

Chapter 6

Conclusion

This work has designed and constructed an underwater platform for testing hardware and software. The system has undergone preliminary testing and validation. The main intellectual contributions are a standardized, expandable connector arrangement, and high maneuverability vehicle design without requiring fairings or extensive streamlining.

6.1 Summary

Chapter 1 introduced the problem. First, shiphull inspection is summarized as the target application. Next, more detail is given on the current work and available resources of the research group are discussed to provide background on the need and potential use. Based on this, some requirements are identified for a potential new vehicle for use by the group. Lastly, a highlight of some existing work in the field is provided.

Chapter 2 covers broad, system level considerations. After identifying critical functional requirements at the strategic level, a series of potential design parameters are evaluated. After reviewing the potential robots developed, academic lean concepts are applied towards the underlying processes of algorithm and sensor development. Some suggestions toward streamlined development flow and standardized interfaces are made. Specifically, the idea of delaying differentiation is used as a method for reducing the rework required between developmental stages. Next a series of high-level mechanical configurations and considerations are evaluated, leading to the selection of a relatively standard layout with both high maneuverability and modularity maintained through an faring-less open chassis design with exceptionally high thrust capability. A series of additional capabilities and configurations are explored, including actuated thrusters and sensor mounts. After an outline for the mechanical system is developed, the high-level electrical aspects are explored. The problem is considered from top-down, examining different classes of component and the relevant transmission of data between. As with the mechanical system, a system with a high degree of integration within modules, but high modularity between modules is selected. Last, the initial design of the power system is developed to meet the unique high-power requirements of the propulsion system.

Chapter 3 looked into the details of implementation for the system. First, the mechanical subsystems is described in detail, including pressure housings, component mounts, and flotation. This includes preliminary testing and manufacturing procedures for the different pieces. Next, the electronic subsystems are detailed. First, the necessary connectors are enumerated and specified, including the development of a power-augmented USB connector. Next, components of the main control stack are developed, including sensor interface, thruster isolation, power switching, and power filtering. Additionally, the battery system is specified, with particular emphasis on safety considerations. Lastly, the sensor payload is described, including video and sonar cameras, navigation sonar, inertial measurement unit, depth gauge, and auxiliary sensors.

Chapter 4 covered the control system. First a coarse model of the system is developed, for drawing general conclusions about the relevant types of control to be applied and the performances that might be expected. With this model, a decoupled control scheme is implemented, separating pitch and heave from surge and yaw. Next, a scheme for controlling the vehicle when equipped with azimuthing thrusters is described. After this, the electronic hardware is described within the context of control. Then, the operating software is discussed. A Visual Basic application is used to control all low-level vehicle functions. Last, software and hardware approaches to slew limiting are compared.

Chapter 5 summarizes the experimental portion. Leak testing methods are described, including a depressurization method for simulating depth. An unexpected leak path inside hull-penetrating wires is identified and resolved. Next, data capture and remote monitoring capabilities are demonstrated, including vehicle pose and sonar images. Lastly, the recorded data is combined to create a mosaic of the MIT towing tank cross section. In this process, the scanning sonar data is corrected and overlaid on an beam-by-beam basis.

6.2 Future Work

A few details remain before the vehicle is ready for use as a widely-capable test platform. Repairs on the thruster power electronics are required, the result of a power surge. A separate power and signal filtering/watchdog module may reduce the likelihood of this happening again. Such a device would prevent activation of the thruster main power supply until the control supplies were active. It would additionally provide analog slew limiting to the control signals, even if the control stack shuts off. Once this is complete, full sea trials and planned validation of the control system will be possible. A conservative progression of these trials would be closed-loop pitch and depth, followed by yaw and forward speed.

With an operational, waterproof base system, the most exciting work remains ahead. The most interesting of the modular hardware options discussed in chapter 2 would be to finalize and install the azimuthing thrusters. This would allow drastic increase in performance, and facilitate the study of a new broad set of control topics.

The Visual Basic interface serves as an easy method to validate hardware, and implement control strategies using simple difference equations. It will very likely become burdensome, however, during the large matrix operations associated with real-time feature tracking. The two main options include adding a separate navigation coprocessor, or changing the operating software to a

better computational platform, such as hard c-code or fully-integrated MATLAB. The first option has the elegance of separating hard-clocked low-level operations from the more varying navigation calculations and image processing. Once the issue of communication and data flow between the processors is resolved, it offers great flexibility. The latter solution offers the benefit of not requiring a hardware modification or the development of communication protocols. It also has a much greater likelihood of developing an interoperable system with the KSV at the low level. However, with some cleverness, the separation of high- and low-level processes can actually facilitate the transfer of code across the path from simulation to KSV to XAUV.

Bibliography

- [1] Department of the Navy. *The Navy Unmanned Undersea Vehicle (UUV) Master Plan*. 2004.
- [2] Department of the Navy *The Navy Unmanned Undersea Vehicle (UUV) Master Plan*. 2000.
- [3] Blue View Technologies. P900-20 Datasheet. [Online] 2007.
http://www.blueviewtech.com/files/products/P900E_Datasheet.pdf.
- [4] Sound Metrics Corp. SPECS for DIDSON Standard and Long Range units. [Online] 2007.
http://www.soundmetrics.com/PRODUCTS/SPECS/PR_SPECS_UNI-STD+LR.pdf.
- [5] Bluefin Robotics. [Online] 2007. <http://www.bluefinrobotics.com/bluefin9.htm>.
- [6] GAVIA Brochure. [Online] 2007.
<http://www.gavia.is/downloads/brochures/GaviaBrochure0402.pdf>.
- [7] *Development of a micro autonomous underwater vehicle for complex 3-D sensing*. Hobson, B. Schulz, B. Janet, J. Kemp, M. Moody, R. Pell, C. Pinnix, H. OCEANS, 2001. MTS/IEEE Conference and Exhibition.
- [8] *Design elements of a small low-cost autonomous underwater vehicle for field experiments in multi-vehicle coordination*. Stilwell, D.J. Gadre, A.S. Sylvester, C.A. Cannell, C.J. 2004. 2004 IEEE/OES Autonomous Underwater Vehicles.
- [9] Robert Damus, Samuel Desset, James Morash, Victor Polidoro, Franz Hover, Chrys Chryssostomidis, Jerome Vaganay, Scott Willcox. A New Paradigm for Ship Hull Inspection Using a Holonomic Hover-Capable AUV. [book auth.] J. Braz et al. (eds.). *Informatics in Control, Automation and Robotics I*. Netherlands : Springer, 2006.
- [10] *Range-based Navigation of AUVs Operating Near Shiphulls*. Kokko, Mike. Cambridge : MIT, 2007.
- [11] *Ship Hull Inspection with the HAUUV: US Navy and NATO Demonstration Results*. Vaganay, J., Elkins, M., Esposito, D., O'Halloran, W., Hover, F., Kokko, M. 2006. Proceedings of IEEE / MTS Oceans 2006.
- [12] Amy L. Kukulya, Mike Purcell, Roger Stokey. AUVFest08 Demonstration. Panama City, FL : s.n., 2007.
- [13] *The Cetus UUV/EOD robotic work package: a low-cost shallow-water UUV system for underwater search and intervention*. Trimble, G.M. Nice : s.n. OCEANS '98 Conference Proceedings.

- [14] *Real-Time Image and Status Transmission from a UUV during a Ship Hull Inspection in a Port Environment using a High-Speed High-Frequency Acoustic Modem*. Beaujean, P.-P.J. Vancouver, BC : s.n. Oceans 2007 Proceedings.
- [15] *Closer to deep underwater science with ODYSSEY IV class hovering autonomous underwater vehicle (HAUV)*. Desset, S. Damus, R. Hover, F. Morash, J. Polidoro, V. Oceans 2005 Europe Proceedings. Vol. Vol. 2, pp. 758- 762 .
- [16] WHOI Media Relations. *News Release : New Robot Sub Surveys the Deep off the Pacific Northwest*. Woods Hole, MA : s.n., August 8, 2008. Science Daily.
- [17] AUVSI. AUVSI and ONR's 11th International Autonomous Underwater Vehicle Competition . *Association for Unmanned Vehicle Systems International*. [Online] <http://www.auvsi.org/competitions/water.cfm>.
- [18] Pascal, Dennis. *Lean Production Simplified*. New York : Productivity Press, 2007.
- [19] *Strategies for Lean Product Development*. Wilson, Myles. Cambridge : MIT Lean Aerospace Initiative, 1999. Working Paper WP99-01-91.
- [20] Fossen, Thor I. *Guidance and Control of Ocean Vehicles*. s.l. : Wiley, 1994.
- [21] *Automated test of ECUs in a hardware-in-the-loop simulationenvironment*. Boot, R. Richert, J. Schutte, H. Rukgauer, A. Kohala Coast : Proceedings of the 1999 IEEE International Symposium on Computer Aided Control System Design, 1999.
- [22] Mascitelli, Ronald. *The Lean Design Guidebook*. Northridge CA : Technology Perspectives, 2004.
- [23] *The Lean Product Development Guidebook*. Northridge CA : Technology Perspectives, 2007.
- [24] Eppinger, Ulrich and. *Product Design and Development*. Singapore : McGraw-Hill, 2003.
- [25] National Instruments. *Working with .m File Scripts in NI LabVIEW for Text Based Signal Processing, Analysis, and Math*. 2006. Application Note, Version 15.
- [26] Suh, Nam P. *Axiomatic Design: Advances and Applications*. New York : Oxford University Press, 2001.
- [27] NOAA. Tidal Current Table. *NOAA Tides and Currents*. [Online] Dec 2007. <http://tidesandcurrents.noaa.gov/currents08/>.
- [28] Walker, Daniel G. *Design and Control of a High Maneuverability Remotely Operated Vehicle with Multi-degree of Freedom Thrusters*. Cambridge : MIT, 2005.

- [29] X-29 Fact Sheet. [Online] <http://www.nasa.gov/centers/dryden/news/FactSheets/FS-008-DFRC.html>.
- [30] White. *Fluid Mechanics*. New York : McGraw Hill, 2002.
- [31] *Model 560 Installation & Repair Manual*. Tecnadyne. RANCHO SANTA FE, CA : s.n., 2006.
- [32] REMUS 100. [Online] 2007. <http://www.hydroidinc.com/pdfs/remus100web.pdf>.
- [33] *University of Ottawa Fourth Generation AUV*. Engineering, Advanced Robotics Innovations Society in. Ottawa : Association for Unmanned Vehicle Systems International, 2008, Vol. AUVSI & ONR's 11th International AUV Competition.
- [34] *ORCA-VIII: An Autonomous Underwater Vehicle*. al., Joshua F. Apgar et. Cambridge : Association for Unmanned Vehicle Systems International, 2004, Vol. 8th Annual International Autonomous Underwater Vehicle Comptition.
- [35] *8th Annual AUVSI Competition: Georgia Tech Marine Robotics Group*. al., Melissa Cataldo et. s.l. : Association for Unmanned Vehicle Systems International, Vol. 8th Annual Autonomous Underwater Vehicle Competition.
- [36] House, D J. *Ship Handling: Theory and Practice*. Oxford : Butterworth-Heinemann, 2007.
- [37] Carlton, John. *Marine Propellers and Propulsion (Second Edition)*. s.l. : Butterworth-Heinemann, 2007.
- [38] *The Stingray*. Society, San Diego iBotics Student Engineering. San Diego : AUVSI, 2007, Vol. 2007 AUVSI Underwater Competition.
- [39] Gook, Michael. *PC Interfaces: A Developer's Reference*. Wayne, Pa : A-List, LLC, 2004.
- [40] *An amphibious robot capable of snake and lamprey-like locomotion*. A Crespi, A Badertscher, A Guignard, A Ijspeert. Paris : ISR2004, 2004.
- [41] S. H. Crandall, N.C. Dahl, and T. J. Lardner. *An Introduction to the Mechanics of Solids: Second Edition with SI Units* . s.l. : McGraw-Hill, 1999.
- [42] Parker Seals. *Parker O-Ring Handbook*. Cleveland : s.n., 2001.
- [43] *Distributed multiplexers for an ROV control and data system*. Mellinger, E., Pearce, A. and Chaffey, M. s.l. : OCEANS '94, 1994.
- [44] *USB-6215 User Manual*. National Instruments. 2006.

- [45] Hirano, Hiroyuki. *Poka-Yoke: Improving Product Quality by Preventing Defects*. Portland : Productivity Press, 1988.
- [46] *High-Linearity Analog Optocouplers*. HP. 1996. 5965-3577E.
- [47] *OPA376: Precision, Low Noise, Low Quiescent Current Datasheet*. TI. SEPTEMBER 2007. SBOS406B.
- [48] Turnbull, Robert. *MULTIPLE CELL AND SERIALLY CONNECTED RECHARGEABLE BATTERIES AND CHARGING SYSTEM*. 5821733 USA, October 13, 1998.
- [49] *Power system for new MBARI ROV*. Mellinger, Ed. Victoria, BC, Canada : s.n., 1993. IEEE OCEANS '93. 'Engineering in Harmony with Ocean'. Proceedings.
- [50] RoboteQ. *AX2500/2850 Motor Controller User's Manual*. Version 1.7. February 1, 2005.
- [51] AMTHERM. *Part Number: MS32 0R536*. Carson City, NV : s.n., Revision Date 3/16/2004.
- [52] Littelfuse. *AN9311.6: The ABCs of MOVs*. July 1999.
- [53] *In-Depth Fundamentals of Overcurrent and Overtemperature Devices*. Raychem. 2004.
- [54] *3DM-GX2 Technical Product Overview*. MicroStrain. Williston, VT : s.n., July 13, 2007.
- [55] *High Speed Hazard Avoidance for Mobile Robots in Rough Terrain*. Spenko, Iagnemma, Dubowsky. 2004. SPIE Conference on Unmanned Ground Vehicles. pp. 439-450.
- [56] *Adaptive Mobile Robot Navigation and Mapping*. Feder, Jeonard, Smith. s.l. : International Journal of Robotics Research, 1999.
- [57] *Active Airborne Localisation and Exploration in Unknown Environments using Inertial SLAM*. Bryson, Sukkarieh. s.l. : IEEE Aerospace Conference, 2006.
- [58] *Multi-Objective Optimization of Sensor Quality with Efficient Marine Vehicle Task Execution*. Benjamin, Grund, Newman. Orlando : International Conference on Robotics and Automation, 2006.
- [59] Friedland, Bernard. *Control System Design: An Introduction to State Space Methods*. New York : McGraw-Hill, 2005.
- [60] *Simultaneous Map Building and Localization for an Autonomous Mobile Robot*. Leonard, Durrant-Whyte. Osaka : Intelligent Robots and Systems '91, 1991.
- [61] Durrant-Whyte, Tim Bailey and Hugh. Simultaneous Localization and Mapping. *IEEE Robotics and Automation Magazine*. 2006, September.

[62] Gelb, Arthur. *Applied Optimal Estimation*. s.l. : MIT Press, 1974.

[63] *A Survey of Control Allocation Methods for Ships and Underwater Vehicles*. Fossen, Johansen. s.l. : 14th Mediterranean Conference on Control and Automation, 2006.

[64] Trintafyllou, Hover. *Maneuvering and Control of Marine Vehicles*. Cambridge : MIT, 2003.

[65] Nise, Norman S. *Control Systems Engineering*. 4th. Pomona : John Wiley & Sons, Inc., 2004.