Seawolf V
North Carolina State University
RoboSub 2012

Michael Brown, Jeffrey Fowler, Alex Hertzog, Brent McMiller,
Tim Mervine, Taylor Stiles, Chris Thunes, Carsi Tong,
David Vanleeuwen, Stu Walls, Matt Wiggins, Robert Younts

June 25, 2012

Abstract

Seawolf V is the fifth in a series of submersible autonomous robots developed by the North Carolina State University Underwater Robotics Club (URC). The URC is a group of undergraduate students whose goal is to give students an opportunity to apply classroom knowledge to solve real-world problems. Seawolf V is designed to compete in the 2012 Robosub Competition, which consists of an underwater obstacle course of vision and acoustic tasks which the robot must navigate autonomously.

We chose to start a new robot from the ground up for the 2012 competition in order to give new members a chance to learn from the beginning of the process. Learning from previous robot designs, we developed the mechanical and electrical systems to be as straightforward as possible, minimizing potential problems. We kept the software architecture of a modular application based approach from previous Seawolf designs, but complemented it with components to make operation and testing smoother, such as a video stream server (SVR), and a simulator. In this paper we will describe the different components of Seawolf V and the motivation behind the design.

1 Introduction

The 2012 Robosub competition is hosted by the hosted by AUVSI and ONR. It will take place from July 17 to 22 at the TRANSDEC facility in Point Loma, California. The theme for 2012 is *Ides of TRANSDEC*. This year's mission obstacles are:

- Path
- Training (Buoy)
- Obstacle Course (PVC to pass over)
- Gladiator Ring (Bins)
- Et Tu Brute? (Window Cutouts)
- Feed Emperor Grapes (Manipulation Task)
- Laurel Wreath (PVC Recovery)
- Emperor's Palace (Octagon)

The details of these tasks can be found in the Robosub competition rules.

Figure 1: Seawolf V

The North Carolina State University Underwater Robotics Club (URC) designed and built Seawolf V to complete these tasks. Seawolf V has been built from the ground up utilizing lessons learned from previous Seawolf designs. We developed the mechanical and electrical systems to be as straightforward as possible, minimizing potential problems. We kept the software architecture of a modular application based approach from previous Seawolf designs, but complemented it with components to make operation and testing smoother, such as a video stream server (SVR), and a simulator.

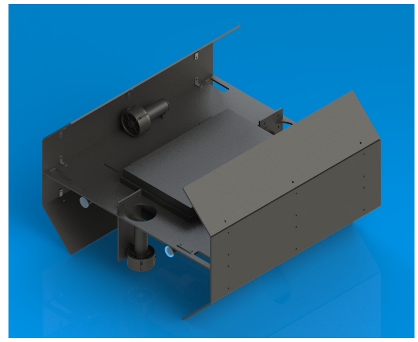

Figure 2: Solidworks Rendering

2 Mechanics

2.1 Frame

The frame was designed in SolidWorks and incorporates sheets of ABS plastic, reinforced with a combination of T-slotted 8020 extrusions and bent stock aluminum. ABS plastic was chosen because it is easy to work with, lightweight, and just slightly negatively buoyant in water. It also has pleasing aesthetics. We used a ShopBot CNC router to precisely cut the sheets of ABS. On the bottom side of the frame is a movable weight system that allows the robot to be finely balanced. A Solidworks rendering of the frame is shown in figure 2.

2.2 Waterproof Case

The housing for the computer and electronic hardware is enclosed in a plastic 1490CC Pelican case. The off-the-shelf Pelican case is durable, waterproof, and easy to modify. The case adds roughly 40 pounds of buoyancy, which is offset by lead acid batteries, thrusters, and adjustable weights. The case is seated into the frame and secured with nylon straps. All of the components outside the Pelican case are connected with dry-matable, hermetically sealed 104 series Fischer Connectors.

2.3 Thrusters

Seawolf V has four Seabotix DC Brushed Thrusters. They run at 24V which is provided by two 12V sealed lead acid batteries that are connected in series. The thrusters located on port and starboard are used to control forward, backward, and yaw rotation. The thrusters located on the bow and stern are used to control depth and pitch movement.

2.4 Marker Dropper

Seawolf V has a servo-actuated marker dropper that can individually drop ball-bearings, although it only holds 2 per the competition rules. This is used for the Gladiator Ring obstacle. The assembly was machined on MECHA Inc.'s automated mills.

2.5 Torpedo Launcher

The team worked on modifying an off-the-shelf toy torpedo gun to attach to the frame. The design uses a solenoid to activate a spring loaded rubber torpedo. Unfortunately, the design was never implemented due to time constraints.

3 Sensors

3.1 Inertial Measurement Unit (IMU)

The Microstrain 3DM-GX1 Inertial Measurement Unit measures pitch, yaw, and roll data using a three axis gyro and three axis magnetometer. It sends the computer this data through a serial interface. The IMU is shown in figure 3.

Figure 3: IMU

3.2 Pressure Transducer

The Measurement Specialties US300 analog pressure transducer is used to measure depth. It is mounted through a hole in the Pelican case and connected to the electronics board where the microprocessor reads the sensor value.

3.3 Cameras

Seawolf V has three USB HD Microsoft LifeCam Cinema cameras. Previous versions of Seawolf were limited by cameras with a small field of view, but these cameras provide a wide 74-degree field of view. As off-the-shelf webcams they also feature auto exposure to automatically compensate for changing lighting conditions. They communicate directly to the computer through USB. One camera faces down, to be used mainly for the path mission. The other two face forward, giving Seawolf the capability of perceiving distance through binocular vision.

The cameras are custom waterproofed using a combination of silicone based sealant, fiberglass resin, and Plasti Dip. A secondary acrylic lens is attached to the front so the camera can see through the waterproofing. This allows the cameras to be more compact than a simple waterproof camera box would be.

4 Electronics

Seawolf V's custom electronics systems are comprised of two boards, stacked with a high-density mezzanine connector between them. These two boards provide power, thruster control, servo control and sensor input for the craft. This design is a marked improvement from the disarranged multi-board electronic systems used in previous Seawolf designs. The goal of the electronic system in Seawolf V was to reduce cable clutter and produce a more reliable and maintainable design.

The electronics components are shown from an overhead view of the Pelican case in figure 4. The custom electronics boards are shown in figure 5.

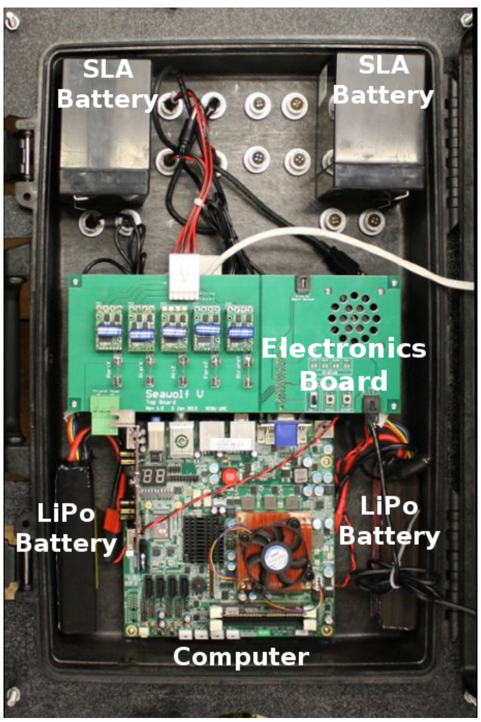

Figure 4: Top view of Pelican case

4.1 Microprocessor

An Atmel ATxmega32A4 microprocessor on the bottom electronic board interfaces to sensors and motor controls. The microprocessor communicates to Seawolf V's computer over a serial connection provided by a FTDI FT4232H USB to serial adapter chip. This chip provides four serial connections over a single USB connection, which provides the serial connection to our IMU.

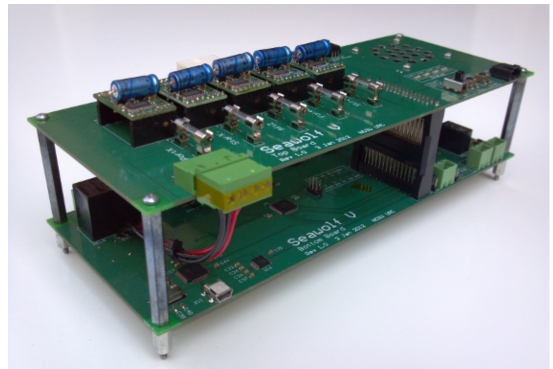

Figure 5: Stacked Electronics Boards

4.2 Power

Power for electronic systems is split into two isolated domains. In one domain, Sealed Lead-Acid (SLA) batteries provide power for thrusters. In the other domain, Lithium-Polymer (LiPo) batteries provide power for Seawolf V's computer and controls. These two power domains are optically coupled and otherwise are entirely isolated. This isolation protects sensitive digital and analog electronics used in Seawolf V's control systems from the noise and transients produced by the thrusters.

Power for the computer and controls are regulated by two 12V switching power supplies capable of providing 72W of power each. Additional regulators provide 9V, 6V, 5V, and 3.3V for peripherals and electronics.

Additional features provided by the electronic system include automatic low voltage cut-off for the LiPo batteries, low voltage warnings for all batteries, hot-switching to external power to conserve battery life, and a diagnostics and control panel for quick debugging.

4.3 Computer

The computer's processor is an embedded Intel Core i7-2710QE on a mini ITX form factor motherboard, donated by Intel. It is shown in figure 6. Most of the computer's speed is put toward image processing, which takes advantage of the processor's four cores by processing images from each binocular camera separately. The two LiPo batteries that power the computer will run for about two hours on one charge.

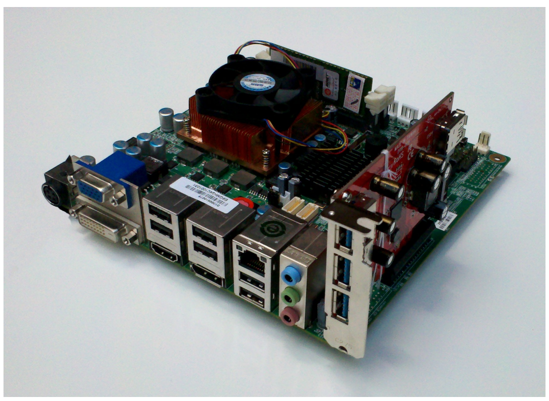

Figure 6: Intel Motherboard

5 Software

5.1 Libseawolf

Seawolf's software is organized into many independent applications which each execute in a separate process. They can be run independently, making for a highly modular design.

To allow these applications to communicate we use a custom interprocess communication (IPC) library called libseawolf. libseawolf was developed in 2008 by the team and has since become a stable independent library. libseawolf allows applications to seamlessly set and access shared variables, send notifications, and wait for events to complete. Each application talks directly with a central server, called seawolf-hub, using TCP/IP, which allows components of the system to be distributed across multiple machines, or even written in different languages. Although libseawolf is written in C, we have written Python language bindings that are used to write some of our applications in Python.

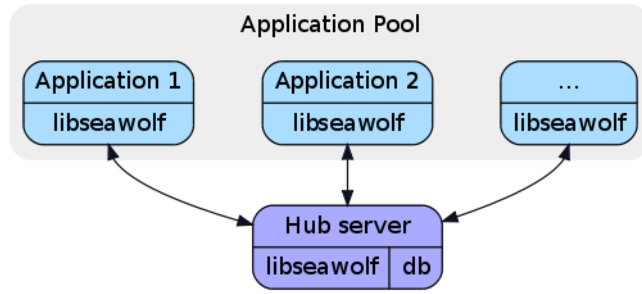

Figure 7: libseawolf Architecture

An overview of the dataflow through Seawolf V's applications are shown in figure 8.

5.2 Seawolf Video Router

New in Seawolf V is a streaming video router called SVR (Seawolf Video Router). SVR allows for efficient handling of multiple video stream sources, and allows for the creation of live debug video streams which can be monitored remotely over a network connection to Seawolf V. Without SVR, viewing these streams would mean using the X11 protocol, which is slow at transferring large images.

5.3 Vision

The vision application takes images from SVR and interprets them by looking for objects. Examples of objects include buoys, the gate, and a path marker. Mission control requests vision to look for a particular object, and vision will respond whenever it sees that object. To recognize objects, we use a set of OpenCV algorithms

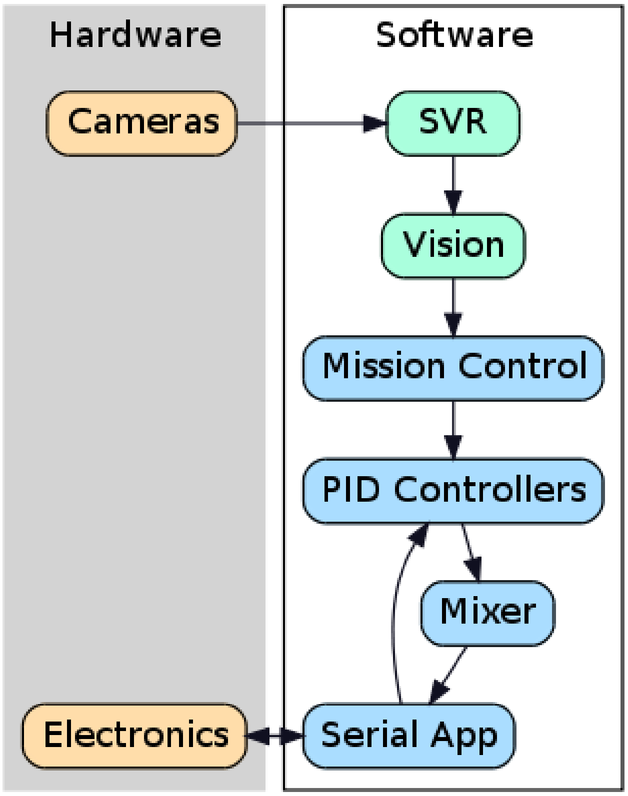

Figure 8: Application data flow

including the Canny Edge Detect, Hough Transform, and Haar Classification.

5.4 Mission Control

Seawolf's mission control subsystem utilizes vision and sensor data in order to make all the decisions necessary to complete tasks. When a task is finished, mission control progresses through each task linearly. Each task has its own logic code that navigates the current task and positions for the next task. These navigation commands are given to the PID controller subsystem, which controls the values to be set each thruster. Figure 9 shows the logic behind the buoy task, an example of how a mission control task works.

5.5 Control

The PID controller applications are responsible for setting thruster values based on a desired heading, which is set by mission control. PID is a common control loop algorithm which uses three terms: proportional, integral and differential, hence the name PID. Seawolf V has three PID controllers: yaw, depth, and pitch. Since there are only two thrusters in the up and down axis, roll cannot be actively controlled. Instead movable weights must be adjusted to balance roll.

The output of each of the PID controllers must be mixed together into actual thruster values,

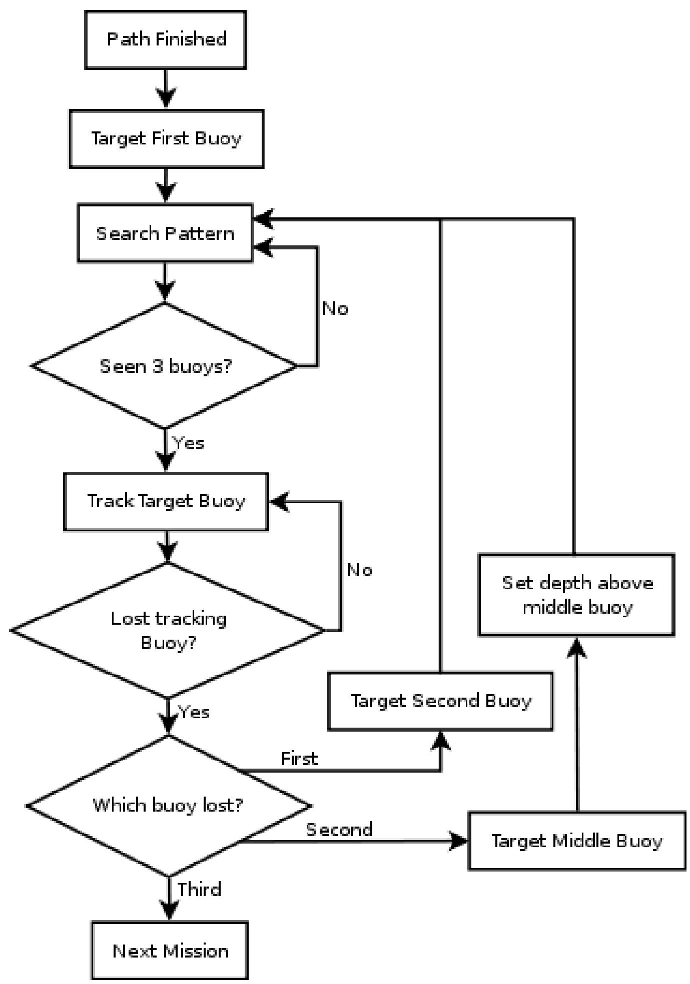

Figure 9: Buoy Mission Logic

which range from -1 to 1. This is the job of the mixer application. The final thruster values are then given to the serial application.

5.6 Serial Application

The serial application handles communication to and from the microcontroller and other sensors except for the cameras, which communicate directly to SVR. The serial application goes through each serial interfaces and deduces which device it is connected to. It then starts the correct driver for each serial device.

5.7 Simulator

A major barrier to getting mission control working correctly is the amount of testing time it takes. Testing usually requires long amounts of time in the water. Even simple problems with mission control can be hard to fix out of water. To alleviate this barrier, we wrote a robot simulator in Python using OpenGL. By replacing applications that interact with the environment, the simulator allows us to test the remainder of our software without getting in the water.

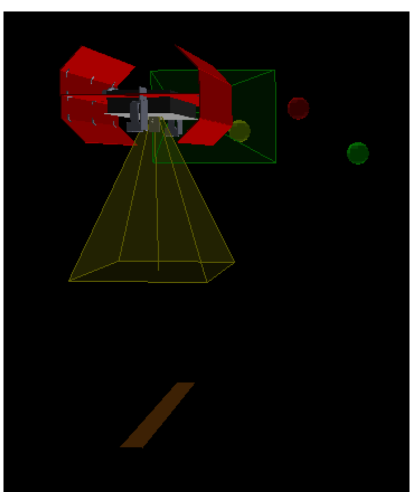

Figure 10: Seawolf Simulator

6 Acoustics

Seawolf V's acoustic navigation system, though incomplete, has a largely complete hardware design consisting of two Analog Devices AD7654 16-bit ADCs providing four channels at 250 kilo samples per second each. These connect to a module containing a Analog Devices BF537 Blackfin DSP running uClinux and custom software for sample collection and signal processing. These components are integrated onto a single board measuring 9cm x 6cm.

7 Acknowledgements

We would like to thank our advisors, Dr. Muth, and Professor Greene. Dr. Muth has generously provided lab space for our team since 2005. We would also like to thank our ECE department staff, especially Ms. Schwab and Ms. Howington; and the NCSU Aquatic Center for use of their diving well.

This year's corporate sponsors include the Dax Foundation, Intel, Northrop Grumman, Pelican, Advanced Circuits, TechShop, Tranquil Hosting, and Digi-Key.

List of Figures