Developing an AUV Manual Remote Control System

Ann Marie Polsenberg, MIT

Mentor: Drew Gashler

Summer 2000

Keywords: AUV, manual control, remote control

ABSTRACT

This paper discusses the design and creation of a remote control system for MBARI's fleet of Autonomous Underwater Vehicles. Designed for use both at sea and in the lab, the system consists of a lap box and a communication protocol between the lap box and software on the vehicle.

A manual control lap box allows an AUV to be controlled at the water's surface.

INTRODUCTION

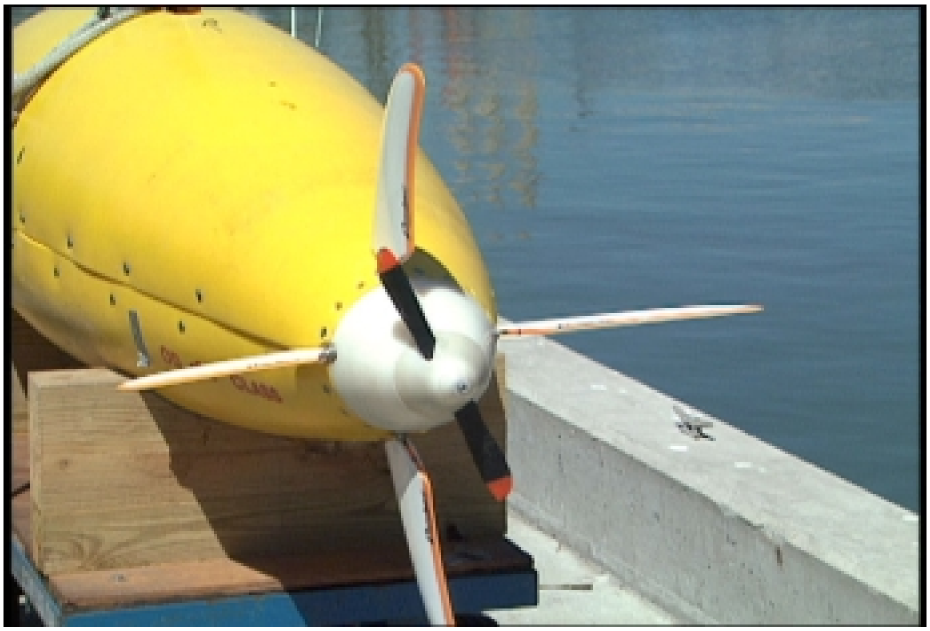
The Monterey Bay Aquarium Research Institute (MBARI) has been working on Autonomous Underwater Vehicle (AUV) design and operation for the last two years. Currently the Institute has three AUVs, and is looking to increase this fleet. An underdeveloped aspect of AUV operations is the launch and recovery of the vehicles. MBARI's AUVs are launched from research vessels. At the end of a mission it is necessary to recover the vehicle, which is floating at the water's surface. Rough seas, and the difficulty of maneuvering large research vessels, make reaching out and hooking the vehicle a challenging endeavor. On many boats used for AUV operations, including the boat MBARI relies upon for most of its AUV operations, vehicles are launched and recovered from the aft of the boat. This means that the research vessel must maneuver to the vehicle, often numerous times, in reverse. This recovery method increases the possibility of collisions between the AUV and the research vessel.

To aid in vehicle launch and recovery, as well as to provide a convenient in-lab vehicle testing method, a manual control system was created. The idea behind this system is to bring the AUV to the research vessel, as opposed to the other way around. This system consists of a lap box with vehicle controls, and an interface protocol for communicating between the box and the manual control software (written by Amanda Green) on the vehicle's main computer.

Design requirements for the full system included compatibility with the vehicle main computer and its existing specifications, the prevention of inadvertently commanding the vehicle to dive while at sea, and a trustworthy exit system for manual control.

Design requirements for the lap box included an easy to use mechanical system, the ability to translate the operator's desired vehicle commands into a format that is readable by the vehicle, and the ability to send these translated commands to the vehicle at the appropriate baud rate.

MBARI's AUVs (Dorodyssey, Dorado, ALTEX) are all propelled by a single rear thruster. While the specific design of this propulsor varies (Dorodyssey is propelled by a stationary propeller and adjustable fins, while Dorado and ALTEX are both propelled by maneuverable ducted propellers), all three vehicles move based on three factors: rudder angle, elevator angle, and propeller speed. Thus, it is possible to design a single lap box that is compatible for any of the vehicles. There are differences in the allowable range for elevator angle and rudder angle on the different vehicles. The lap box, however, is designed to cover the widest range of possible angles. Software on each of the AUVs' main computer is capable of dealing with values outside of the acceptable range. In this situation the computer will set the angle to the largest allowable value.



The Dorodyssey vehicle maneuvers using a propeller and fins.

MATERIALS AND METHODS

GENERAL BOX DESIGN

The lap box uses joysticks, switches and liquid crystal displays (LCDs) as the interface between user and vehicle. The three switches include a Power switch (which turns the lap box's power source on and off), a Manual Control on/off switch which turns the serial connection on and off, and a Mode switch which allows the user to specify whether the vehicle is being used at sea or in the lab, a distinction that will be discussed later. The two joysticks are used to control rudder angle, elevator angle and propeller speed. The LCD displays allow the user to see the angles and speeds that are being sent to the vehicle, as well as see which mode the box is in. The lap box is attached, by a serial cable, to an RF modem which communicates with the vehicle's main computer.

The finished lap box.

HARDWARE

The microprocessor in the lap box is a Basic Stamp II-SX, manufactured by Parallax. Characteristics of this processor that make it well suited for this application are it's speed, large memory, and ability to send information at baud rates of up to 115K. This last characteristic is especially crucial, as the lap box must be able to communicate at 57.6K baud in order to be compatible with the main vehicle computer.

There are other microprocessors which would be able to control this system, however the low cost of the BSII-SX makes it an appropriate choice for this system. The current lap box design uses only 8 of the 16 Input/Output pins. This makes it ideal for future expansion of the system. One suggested future improvement of the system would be the addition of LCDs displaying data from the vehicle once telemetry is established. Two of the remaining 16 pins are already wired for an LCD connection. Thus it is extremely easy to add two more LCD's if it is decided that additional readouts are desired.

Power is delivered to the lap box using six rechargeable Nickel Metal Hydride C–cell batteries. The lap box system runs at 65 mA. A low drop-out voltage regulator converts the input voltage into the 5 volts needed by the circuit, allowing the system to function even as the voltage begins to drop in the batteries. Fully charged, each battery can deliver 1.2 Volts. By using 6 batteries (providing a total of 7.2 Volts when fully charged), the system uses approximately half a Watt. Given that each battery provides 9.4 Watt hours, a set of six fully charged batteries should last over 100 hours. To prevent damage to the system, a fuse is included in the power component.

COMMUNICATION PROTOCOL

The lap box uses a specific protocol to send commands to the vehicle over an RS-232 connection. When manual control is enabled, the lap box sends a constant stream of ASCII packets to the vehicle computer. These packets consist of five pieces of information, separated by underscores. These pieces of information are: manual control status, propeller speed, rudder angle, elevator angle, and a check sum.

Manual Control status is denoted by a 0 or a 1. If Manual Control is enabled, a 1 is the first element of the packet. If Manual Control is disabled, a single packet of five zeroes, (separated by underscores) is sent to the vehicle, and then the box stops streaming data.

Propeller speed is sent to the vehicle as a signed value, between –300 and +300. This represents propeller speeds between –300 rotations per minute and +300 rotations per minute.

Rudder angle and elevator angle are both sent to the vehicle as signed values, ranging from –4500 to +4500. These numbers represent hundredths of degrees, ranging between –45 degrees and positive 45 degrees. Positive rudder angles cause the vehicle to turn to port when moving forward. Positive elevator angles cause the elevator to go trailing edge low, causing the vehicle to dive.

The last element of the packet is a checksum. The checksum is calculated by summing the status, the absolute value of propeller speed, the absolute value of rudder angle, and the absolute value of the elevator angle. The software on the vehicle also sums these values, and if the solution that it calculates is different from the sent checksum, the packet is ignored.

An important consideration when creating this protocol was the requirement that it could be used by other user-interfaces to send information to the vehicle. In Manual Control mode, the vehicle is listening for packets following this protocol. It is not concerned about what is sending these packets. Thus a GUI, or other interface, could be used in place of the lap box without altering the code on the vehicle.

Sample packet:

1_250_1000_-3000_4251*

status ***propeller speed*** ***elevator angle*** ***rudder angle*** ***checksum***

Checksum:

|status| + |propeller speed| + |elevator angle| + |rudder angle|

A sample packet and an explanation of the checksum algorithm.

SOFTWARE

In order to be compatible with the BS2SX microprocessor, the code is written in BASIC. While BASIC has somewhat limited capabilities, it is appropriate technology for this system.

Whenever the microprocessor is powered up, its first task is to calibrate the joysticks. Due to the fact that elevator and rudder angle, as well as propeller speed, are adjusted by moving the joysticks, it is essential that in their neutral position the joysticks are zeroed. While it is possible to do this by selecting an arbitrary potentiometer setting to be the joysticks zero and physically adjusting the potentiometer so as to achieve the selected zero, the nature of this project is such that it is likely that the lap box will be jostled around. If the joystick zeroes were hard coded into the lap box software, normal wear and tear associated with equipment designed for use at sea would almost certainly cause the actual zero values of the joystick to drift, leading to incorrect values being sent to the vehicle.

In order to avoid the situation described above, the lap box recalibrates itself whenever it is turned on. To accomplish this task the processor queries the potentiometers in the joysticks and sets their position at that instant to zero. This is very effective, unless the joysticks are not in their neutral position when the box is turned on. If this is the case, characterized by joystick zeros which are unreasonably large or small, a predetermined estimated zero is used. In order to optimize the control of the vehicle, however, it is desirable to allow the processor to calculate the zero. Thus, care should be taken to ensure that the joysticks are in their neutral position when the box is turned on.

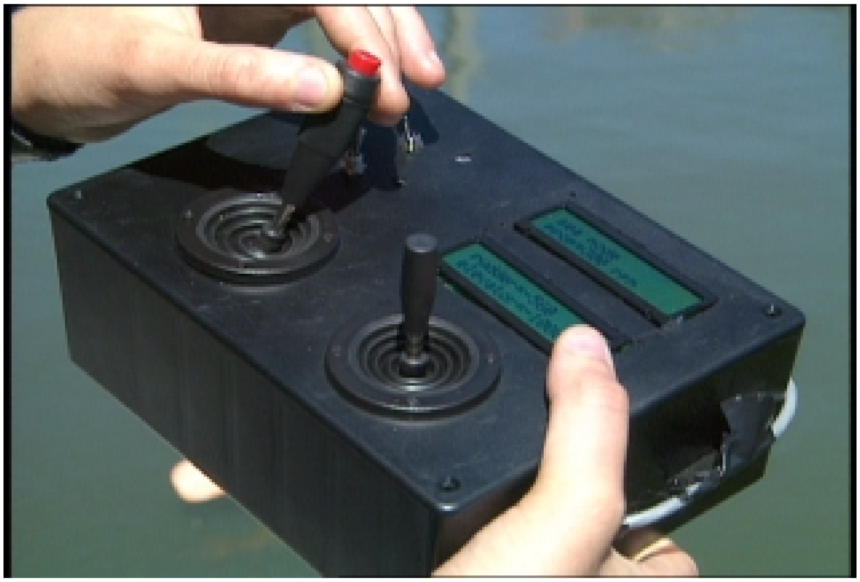
After calibration, the processor checks the state of the Manual Control On/Off switch. If the switch is in the Off position, a single Manual Control off packet (described later) is sent to the vehicle, and then an off loop is entered. In the off loop, the processor sends no data to the vehicle, and keeps querying the state of the Manual Control On/Off switch. If the switch is set to Manual Control On, the On subroutine is run.

The other switch querying routine checks to see whether the user has selected to use the lap box in sea mode or in land mode. This query occurs during each iteration of the program loop, so that the mode can be switched while the box is being used. In Lab mode, the user is able to control the propeller speed, the rudder angle, and the elevator angle. In Sea mode, the user is only able to control the propeller speed and rudder angle, while the elevator angle is set to a constant trailing edge high position. This makes it harder for the vehicle to submerge, which is important as the lap box is communicating with the vehicle over an RF modem. The RF link is only open as long as the vehicle's antenna is above the water's surface.

The three properties that are controlled by joystick deflection (propeller speed, rudder angle, and elevator angle) are calculated using similar subroutines. (It is important to note that the Elevator joystick reading routine is only run when the lap box is in Lab mode. In Sea mode, this routine is replaced by a separate routine which sets the elevator angle to a predetermined negative value.) The joystick reading routines all follow the same process. The capacitor in that potentiometer/capacitor circuit is discharged for 1 millisecond. The charge time of the circuit is then measured. As the resistance of the potentiometer is related to the deflection of the joystick, the degree of deflection can be calculated by finding the resistance of the circuit. The joystick reading routines then check this value, called the RCtime, to see whether it falls within the deadband range. If this is the case, the relevant variable is set to zero. The deadband, a way of discarding erroneous RCtime values, is defined as a range of RCtime units on either side of the zero set for that particular joystick axis during the calibration routine. The next step is to convert the RCtime value into propeller speed, rudder angle or elevator angle (depending on which routine is being run). The routine then checks this scaled value to make sure it fits within the allowable range for that variable. If not, the variable is given the value of the closest allowable speed or angle. Once all of the appropriate joystick readings have been taken and values have been assigned to propeller speed, elevator angle, and rudder angle, the microprocessor sends this information to the LCD screens and to the vehicle.

MECHANICAL DESIGN

Once the hardware and software were created, it was necessary to create a housing for the system. The prototype of this housing (used for the initial water test of the system) consisted of a thin plastic box with holes cut in it. At the time this was created, the vehicle was still running off of an external power source, and thus an additional power cable (and a hole in the box for this cable) was necessary.



The lap box prototype.

This initial housing, while suitable for preliminary testing, had many flaws. The flimsy nature of the box made it prone to damage from a variety of sources, including jostling during transport. The oversized holes for the power and serial cables created an overwhelming possibility of wetting the circuitry. Additionally, the circuit board was not attached to the inside of the box, causing unnecessary stress to the solder connections when the box was transported and handled. Finally, the fully detachable box top meant that wires attached to the board could be yanked when removing the lid. All of the above reasons made this preliminary box unsuitable for use at sea.

The final version of the lap box was a much more robust design. The internalization of the vehicle's power source eliminated the need for a power cable hole in the side of the box. The lap box has only one hole on its side, for the neoprene coated serial cable, and this hole is fitted with a waterproof connector. The box itself is quite sturdy, with a hinged lid and a handle for transport. Holes were milled into the lid in order to attach the switches, joysticks, and LCDs.

The lap box is designed to be transportable and sturdy.

RESULTS

The initial in-water test of this system took place on August 10, 2000. For the test, the Dorodyssey vehicle was deployed in a boat slip at MBARI. Using the lap box, we were able to drive the vehicle around the slip. A simulation of a real situation, pulling the vehicle up to a research vessel, was successful.

CONCLUSIONS/RECOMMENDATIONS

AUV Manual Remote Control is feasible. The prototype version of lap box driven control has proven itself powerful and useable. This system was deemed sufficiently reliable to be brought along on the most recent MBARI AUV operations cruise. As of the writing of this paper, the results of at sea testing of this system are not known. Upon the receipt of these results, there will undoubtedly be parts of this system that can be improved upon.

ACKNOWLEDGEMENTS

I would like to thank Drew Gashler and Paul McGill for all of their help and advice, Amanda Green for writing the vehicle on-board manual control software and all of the members of the MBARI AUV lab for their help and support. I would also like to thank Clark Brecht, Zorba Pickerill, Herb Lundin, Jim Scholfield and Gary Burkhart, for bringing a level of polish to this project that I would not be capable of achieving on my own.

\