Title

A Small Submarine Robot for Experiments in Underwater Sensor Networks

Permalink

https://escholarship.org/uc/item/5t57f9d4

Authors

V. Bokser
C. Oberg
G. Sukhatme
et al.

Publication Date

2004

Peer reviewed

A SMALL SUBMARINE ROBOT FOR EXPERIMENTS IN UNDERWATER SENSOR NETWORKS

Vitaly Bokser, Carl Oberg, Gaurav S. Sukhatme,
and Aristides Requicha

Department of Computer Science
University of Southern California
Los Angeles, CA 90089
Email: vbokser@usc.edu, oberg@usc.edu, gaurav@usc.edu,
requicha@usc.edu

Abstract: This paper describes a small underwater robot designed for experiments with sensor-actuator networks. The robot is based on the mote platform, which is used extensively in the sensor networking community as an experimental testbed. The components and construction of the robot are described. Preliminary tests of depth regulation and temperature measurement are reported and analyzed. *Copyright ©2003 IFAC*

Keywords: marine and underwater systems, sensor networks, sensor-actuator networks, multi-robot, small robotic submarine

1. INTRODUCTION

The ocean is a fascinating domain, and relatively unexplored compared to the land masses. Studies of biocomplexity, particularly those focusing on the life cycles of underwater micro-organisms are the subject of extensive investigation by researchers in Marine Biology (Caron, *et al.*, 2000). However the state of the art of sampling equipment in such studies is usually a single monolithic system which is lowered into the ocean at various depths, and transported using a ship (Blidberg, 2001; Blidberg, *et al.*, 1998; Brutzman, *et al.*, 1998). We posit that a different kind of 'instrument' - a distributed collection of sensors which are networked and can move autonomously, is a useful technology for research in Marine Biology. Such an instrument is effectively a sensor-actuator network, or an underwater multi-robot system.

Fig. 1. Mica2 Mote

Although in recent years a flurry of sensor network devices have been constructed and new algorithms designed (Batalin and Sukhatme, 2003; Li, *et al.*, 2003), research has yet to focus on what useful techniques can be developed with mobile robots and sensor networks in the oceans, with only a few groups working in this domain (McFarland, *et al.*, 1998; Doty, *et al.*, 1998). A popular platform choice in the sensor networking community is the Berkeley mote platform (Pister, *et al.*, 1999). These new devices (Figure 1), miniature in size, have the ability to process data, communicate with others through onboard radio, contain vari-

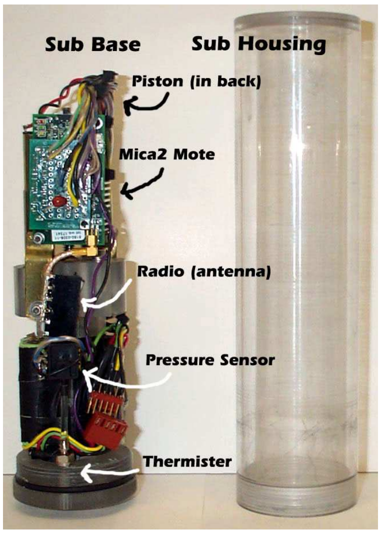


Fig. 2. The robot base and waterproof housing

ous sensors, all designed to fit within a few square inches of silicon. These wireless devices are being used today in a multitude of sensor networks research, in a plethora of fields, ranging from habitat monitoring (Mainwaring, *et al.*, 2002; Cerpa, *et al.*, 2001), to tracking moving objects (Brooks, *et al.*, 2002; Yank and Sikdar, 2002), and vehicle classification in wireless sensor networks (Duarte and Hu, 2003). We have chosen to employ these devices as the base platform used for controlling a small robot submarine. This is primarily because they are small and there is an extensive community that already uses them in sensor networking. These devices have hard limitations on processing and communication, which forces algorithm designers to develop lightweight strategies for robot control and coordination, suitable for miniaturized robots of the future.

2. SYSTEM DESIGN

The physical system is depicted in Figure 2. The robot is composed of two parts: the base on which all the electronics are mounted, and the housing which is a protective enclosure. The robot is a cylinder standing 23.5 cm high and 6 cm in diameter. Figure 3 shows a schematic of the robot's hardware which is described below.

2.1 Mote Hardware

The mote we are using is the Mica2, which contains the Atmel ATmega 128L microprocessor. The Mica2 hardware contains Digital IO lines, SPI, UART, USART, 10-bit ADC channels, 128K Bytes Program Flash Memory, and 4K Bytes EEPROM. The radio transceiver operates at 433 MHz for a maximum outdoor range of 1000 ft. The Mica2 runs its own specialized embedded OS called TinyOS which is an "event based operating environment designed for use with embedded networked sensors." (TinyOS Website Definition, 2003).

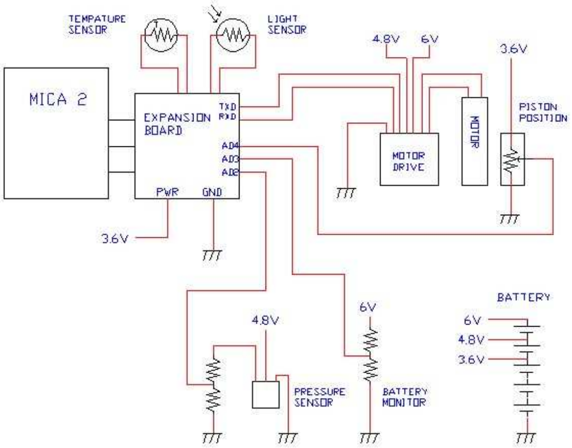
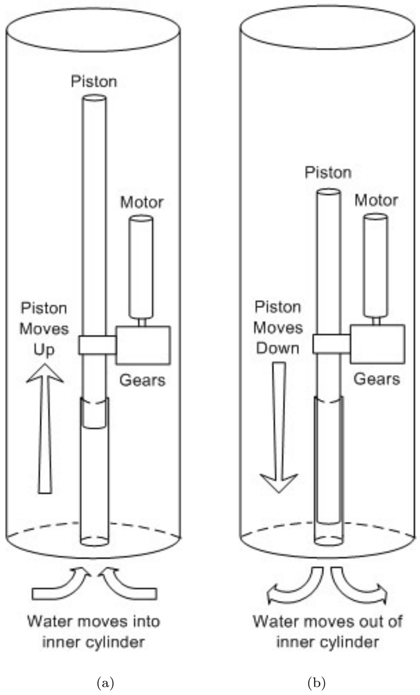


Fig. 3. Circuit schematic of the robot



(a)

(b)

Fig. 4. 4(a) Water being drawn in as the piston moves up, thus causing the robot to sink. 4(b) Water being forced out as the piston moves down, thus causing the robot to rise (float) in the tank.

2.2 Propulsion system

In Figure 4, a simplified view of the propulsion system for the robot is shown. As seen in the figure, the robot uses a simple mechanism which functions by changing the volume of the robot using a piston. This change in piston position leads to a change in volume which varies the robot's buoyancy. As shown in Figure 4(a), when the piston moves up, water is pulled in, increasing the robot's total volume and thereby causing it to descend since its buoyancy has decreased. As shown in Figure 4(b), as the piston moves down, water is pushed out thereby decreasing the robot's total volume causing it to ascend due to increased buoyancy.

2.3 Motor Driver

The motor driver is manufactured by Solutions Cubed and accepts serial commands from the Mica2 UART at 2800 bps. The motor driver controls the direction and speed of the motor, and has some feedback mechanisms which are currently not being used.

2.4 Sensors

2.4.1. Pressure Sensor Manufactured by Invensys, the pressure is valid from 0-5 PSI, with an accuracy of +/-2.0%.

2.4.2. Light Sensor Manufactured by Clairex under part number CL94L, this sensor is a simple CdSe Cell. The max sensitivity occurs at wavelength of 690nm (XBOW Sensor Info, 2003).

2.4.3. Thermistor Manufactured by YSI under part number 44006 with a possible accuracy of 0.2°C if the sensor is calibrated (XBOW Sensor Info, 2003).

2.4.4. Health Monitoring We have the ability to monitor the charge level of the main battery system through a simple voltage divider and an ADC port of the Mica2. This can, in principle, enable the robot to move to a recharge station at the surface when battery power runs low. Another possibility is to go into low power mode where a particular node in the network will do less work (i.e. turn off non critical subsystems) in order to conserve energy.

2.4.5. Linear Potentiometer Measures the position (location) of the piston along the potentiometer through a simple resistance measurement read by an ADC port on the Mica2.

3. SOFTWARE DESIGN

3.1 General Architecture

The software architecture is shown in Figure 5. The main control (MC) program of the robot is the control center. All other significant subsystems are separate modules. The depth controller (DC) is responsible for regulating the robot's depth in response to depth commands from the MC. The MC program clocks the DC as to how often a reading of depth and a correction is made. In our experiments the DC was clocked at roughly 200ms. This was done for two reasons, the first being that the timers in TinyOS are still not fully mature as of this writing and experience some problems, while the single clock mechanism of TinyOS is fully functional (which we are currently using). The second reason is that it gives the implementer of sensor network algorithms more control over how many readings are taken (to conserve some cpu cycles) and what mechanism is chosen to use for clocking the DC. As the DC is clocked, it sends an event back to the MC program informing it with the current depth of the robot.

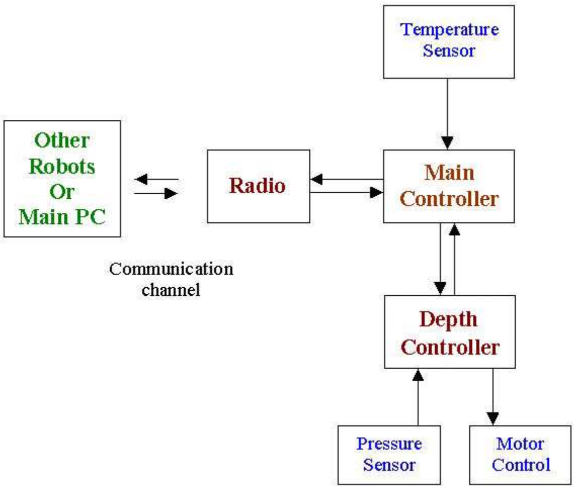


Fig. 5. Software architecture schematic

When the desired depth is reached, it is still necessary to clock the DC so it can continue to perform its job of regulating the robot's depth. This is because even in our experimental testbed (a tank) the robot is constantly buffeted by small eddies. The DC uses the pressure sensor for depth and the motor control module (MCM) to move the piston accordingly. This isolates the DC from the sensors used to measure depth.

3.2 Control

The DC control mechanism is a proportional (P) controller. Depending on the error (i.e. desired depth - current depth) the control system either increases or decreases the velocity of the motor. The code is similar to that shown below.

```
error = desiredDepth - currentDepth
deadBand = desired accuracy in cm
pGain = f(motor, deadband)
```

```
pTerm = pGain * error
motorSpeed = pTerm

IF(abs(error) < deadBand)
    motorSpeed = 0;

IF (motorSpeed == 0)
    Stop robot
ELSE IF (motorSpeed < 0)
    Move robot up
ELSE IF (motorSpeed > 0)
    Move robot down
END
```

The deadband is the region where the motor is commanded to turn off. This is governed by the acceptable error range for our experiments. In the experiments reported here, the deadband was chosen to be 5 cm. The pGain is a function of the motor and the deadband and was determined experimentally to be 5. This means that at a distance of 5 cm from the desired depth and at a gain of 5 the commanded motor speed would be approximately 25 which is basically the bare minimum for our motor to begin movement of the piston.

Depth regulation is important because the robot needs to maintain a stable position underwater so that an accurate measurement can be made by the thermistor at the commanded depth. Currently, this measurement takes roughly 3 min. in order to get an accurate reading. In the experiments reported here we show the temperature readings taken by the robot as it dives. An interesting, well-known, feature of water is its columnar structure with a varying temperature gradient, particularly a region of sharp change in temperature, called a thermocline. Such a region may form a natural barrier between sunlight (available from above), and nutrients (available from below), thus making it an interesting region for a sensor network to focus on.

4. EXPERIMENTS

4.1 Depth Measurement

Depth measurement was accomplished through the use of the pressure sensor. The testbed used was a large tank shown in Figure 6 which currently contains fresh water for the experiments, but will be switched to salt water in a few months to better simulate oceanic conditions.

Data from the pressure sensor were recorded at various depths (measured manually). These are shown in Figure 7. The relationship is nearly linear, and the best fit is given by Eq. 1.

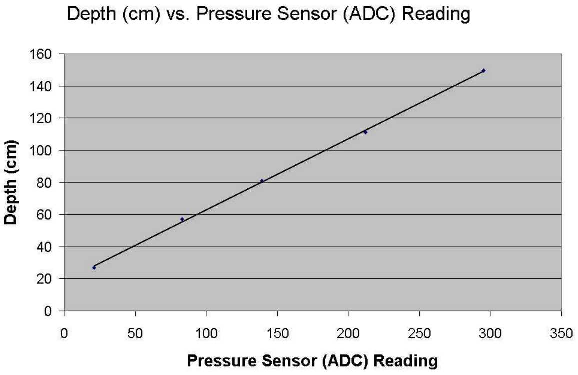


Fig. 7. Depth vs. Pressure Sensor (ADC) Reading

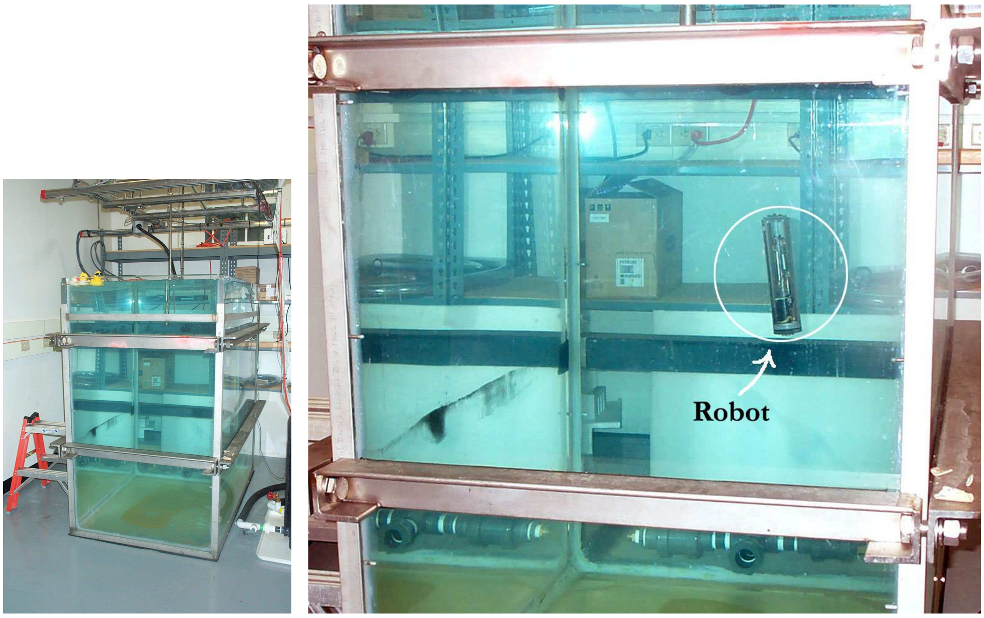
$$Depth(cm) = 0.4432 * PressureReading + 18.587 \quad (1)$$

4.2 Depth Regulation

As described in the previous section, the robot regulates its depth through a piston mechanism which increases and decreases its volume, thereby increasing or decreasing buoyancy. As buoyancy increases the robot rises to the surface and as it decreases, it sinks to the bottom of the tank. The piston is actuated by a motor. The motor is controlled by a motor driver which obtains serial commands from the Mica2 mote. The piston is connected to a linear potentiometer, thus when the piston moves, the position of the linear potentiometer changes in one to one correspondence with the distance travelled by the piston. Currently the linear potentiometer is used to make sure that the piston is stopped if it reaches its limit in either direction. But in the future the potentiometer readings of the piston's position will actually be used to estimate the velocity of the piston and an estimate of the volume displaced.

By combining these two pieces, the calibrated pressure sensor readings and piston controller, the robot can dive to a desired depth. The discussion of the DC component in the previous section explained this process.

In Table 1 are the results of the depth regulation tests in the tank. We performed 5 trials for each depth listed. In each trial there were 10 readings taken 15 seconds apart after the robot had settled into the deadband region. This settling was determined complete if the robot remained in the deadband region for over 3 minutes. These 10 readings per trial were then averaged. After 5 trials for each depth the robot was surfaced and placed back into the water to begin the next trial. In Table 2, the two right columns represent the averaged depth readings and averaged standard deviations of all 5 runs for each depth. As can be seen from Table 2, the robot had settled and remained in the deadband region consistently on



(a) (b)

Fig. 6. 6(a) shows a picture of the custom built tank used in all the experiments. 6(b) shows a closer view of the robot diving to a commanded depth.

Table 1. Depth Regulation Data

DesiredDepth (cm)	*Run#*	*AvgDepth* (cm)	*StdDev* (cm)
40	1	38.80	0.79
	2	36.20	0.42
	3	39.40	0.52
	4	35.10	0.32
	5	43.80	0.42
60	1	59.80	1.55
	2	56.50	0.53
	3	59.60	1.78
	4	63.80	0.79
	5	57.30	0.48
80	1	81.40	1.35
	2	83.20	1.23
	3	77.10	2.28
	4	80.70	2.21
	5	78.90	1.85
100	1	104.70	0.48
	2	95.90	1.20
	3	103.90	1.37
	4	99.70	1.70
	5	102.00	0.67
120	1	120.60	3.13
	2	122.30	1.95
	3	121.80	2.30
	4	121.40	2.07
	5	119.10	3.38

Table 2. Consolidated Depth Regulation Data

DesiredDepth(cm)	*AvgDepth(cm)*	*StdDev(cm)*
40	38.66	0.49
60	59.40	1.02
80	80.26	1.79
100	101.24	1.08
120	121.04	2.57

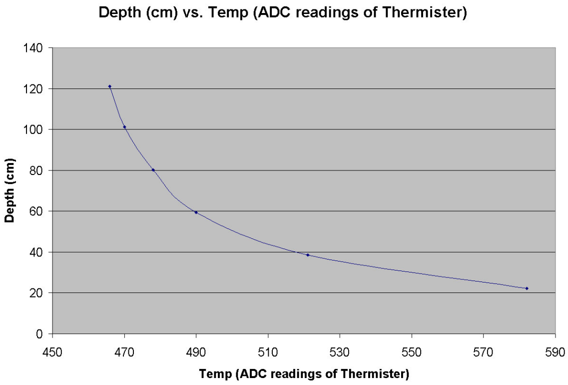


Fig. 8. Depth (cm) vs. Temperature (ADC Readings of Thermistor)

every trial without significant deviation from the commanded depth.

4.3 Thermistor Readings

In Figure 8, the thermistor readings are plotted vs. the depth. The readings were taken in conjunction with each run discussed above. As the Figure shows, the thermocline (greatest change in gradient) is clearly located near the water surface. For our experiments, the thermocline was artificially created by heating the upper region of the tank.

5. CONCLUSIONS AND FUTURE WORK

So far the tests conducted prove the feasibility of the platform. We are poised to conduct further experiments in underwater autonomous sensor networks. The tests with the pressure measurements vs. depth were shown to be linear and are very accurate for our purposes. The robot is able to regulate its depth within 5 cm of the desired depth and consequently we are able to obtain a fairly accurate plot of the thermocline region in our tank. We are in the process of designing a more accurate depth regulation system (+/- 2.5 cm). In order to get more accurate depth control we will decrease the deadband and will experiment with a PD or PID controller since the P controller is unstable at a deadband of less than 5 cm.

Other improvements we envisage include extended battery life, since it is important to keep recharge time in and out of water down to a minimum. A simple technique for power management would be to turn off the power to the motors when not in use, and any other subsystems that are not needed. We plan to implement this in the near future.

Navigation is limited currently to the vertical dimension. In future work, we plan to add functionality of movement in all 3 translational dimensions. Another step is to replicate the system, in order to realize our goal of a networked testbed of underwater robots for experimental studies.

ACKNOWLEDGMENT

This work is supported in part by NSF grants EIA-0121141 and CCR-0120778 under the ITR and STC programs respectively. The authors thank Prof. David Caron, Amit Dhariwal, Beth Stauffer, and Bin Zhang for their assistance and comments.

REFERENCES

Batalin, M. A., and Gaurav S. Sukhatme, *Sensor Network-based Multi-Robot Task Allocation*, In IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS2003) Las Vegas, Nevada, October 27-31, 2003.

Blidberg, D. R., *The Development of Autonomous Underwater Vehicles (AUVs); A Brief Summary*, ICRA, May 2001.

Blidberg, D.R., J.C. Jalbert, and M.D. Ageev. *A Solar Powered Autonomous Underwater Vehicle System*, International Advanced Robotics Program, (IARP), Feb 17-19, 1998.

Brooks, R. R., P. Ramanathan, and A. Sayeed, *Distributed Target Tracking and Classification in Sensor Networks*, Proceedings of the IEEE, September 2002.

Brutzman, D., Tony Healey, Dave Marco, and Bob McGhee, *The Phoenix Autonomous Underwater Vehicle*, chapter 13, AI-Based Mobile Robots, MIT/AAAI Press, Cambridge Massachusetts, 1998.

Caron, D.A., E. L. Lim , R.W. Sanders, M.R. Dennett, U. G. Berninger, *Responses of bacterioplankton and phytoplankt to organic carbon and inorganic nutrient additions in contrasting oceanic ecosystems.* Aquat. Microb. Ecol 22:175-184, 2000.

Cerpa, A., J. Elson, D. Estrin, L. Girod, M. Hamilton and J. Zhao, *Habitat Monitoring: Application Driver for Wireless Communications Technology*, ACM SIGCOMM Workshop on Data Communications in Latin America and the Caribbean, Costa Rica, April 2001.

Doty, K. L., A. Antonio Arroyo, Carl Crane Scott Jantz, David Novick, Robert Pitzer, Aamir Qaiyumi, *An Autonomous Micro-Submarine Swarm and Miniature Submarine Delivery System Concept*, Conference on Recent Advances in Robotics, FIT, March 26-27, 1998.

Duarte, M., and Y.H. Hu, *Vehicle Classification in Distributed Sensor Networks*, submitted to Journal of Parallel and Distributed Computing, 2003.

Li, Q., Michael De Rosa, and Daniela Rus, *Distributed Algorithms for Guiding Navigation across a Sensor Network*, ACM MobiCom 2003, pages 313-325, San Diego, Sept. 2003.

Mainwaring, A., J. Polastre, R. Szewczyk, D. Culler, J. Anderson. *Wireless Sensor Networks for Habitat Monitoring*, ACM International Workshop on Wireless Sensor Networks and Applications (WSNA), Atlanta GA, September 28, 2002.

McFarland, D., I. Gilhespy, and E. Honary, *DIVEBOT: A diving robot with a whale-like buoyancy mechanism.* Robotica, special issue, July, Vol 21, Part 4, 2003.

Pister, K. S. J., J. M. Kahn and B. E. Boser, *Smart Dust: Wireless Networks of Millimeter-Scale Sensor Nodes*, Electronics Research Laboratory Research Summary, 1999.

Yang, H., and B. Sikdar, *A Protocol for Tracking Mobile Targets using Sensor Networks*, Proceedings of IEEE Workshop on Sensor Network Protocols and Applications, 2003.

TinyOS Berkeley, *What is TinyOS?*, http://webs.cs.berkeley.edu/tos/faq.html#SEC-16”

CENS at UCLA, *CENS*, http://cens.ucla.edu/Aboutus/index.html, 2003.

Crossbow, *CrossBow*, http://www.xbow.com/Support/manuals.htm, MTS/MDA Mote Sensor and DAQ Manual (pdf), 2003.