The Nano-drifter Project

Abstract

We have designed and produced pilot-quantities of very small, very cheap, drifters which we have called “nano-drifters.” The objective is to use them to study the current flow in estuaries where high spatial and temporal resolution is necessary. The cost goal is to reach the point where they can be regarded as expendable, even though every effort will be made to recover them. Here we discuss the design, assembly and operation of a small nano-drifter swarm, and the prospects for future development and deployment.

Introduction

The project is driven by two primary factors: first, the scientific need for high resolution observations of current in estuaries; second, the emergence of an open-source hardware development environment build around the “Arduino” micro-controller.

The scientific need can be seen by example in the management of an oyster farm. Here …..

The importance of the Arduino community can be seen by comparison with a smart phone. The smart phone is vastly more complex than a drifter, yet it can be sold for < $100. However the engineering overhead to develop an drifter in such technology would have a prohibitive cost. Fortunately the necessary components can be purchased off-the-shelf in unit quantities and driven with open-source software produced by the Arduino community.

Another important factors, tangential to the development, is educational outreach. Scientifically, estuarine research is far more accessible to school children than most areas of geophysics. Tens of millions of school children live within easy reach of the great estuaries on the eastern seaboard such as the Chesapeake Bay. The “Arduino” hardware development environment was setup for educational purposes in 2005 (by an inspired group of design instructors at the Interaction Design Institute in Ivria, Italy) and has been expanded world-wide in a very short time. It is now widely used in the USA and is supported by two well-established private sales organizations, Adafruit Industries and Sparkfun Electronics. There are other suppliers and components can also be purchased through Amazon with very short delivery times.

Architecture

The basic requirement for a drifter is that its position be measured. At present a complete GPS receiver suitable for the Arduino environment can be purchased for less than $30 in unit quantities. An open-source software library is available and school child can have a time-position display on a computer screen in a few hrs starting from zero. The data can be written on an SD card with little additional work. This makes the project feasible.

Battery power is always an important factor for field instrumentation. In this application battery weight is useful ballast so we can focus on cost and availability. The most cost-effective batteries are AA alkaline, but AA NiMH are only about 4 times more expensive. So if we lose fewer than 25% of the drifters we can use rechargeable NiMH batteries and avoid the waste stream of alkaline cells. This has certainly been true in development experiments, where we have lost no drifters, and we believe it will be true in large scale practice.

The most attractive shape for a drifter is a “spar-buoy” so we structured the nano-drifter on a spine of fiber-glass board 1” x 12” x 0.062”. A group of 8 AA cells are attached to the bottom of the spine and the GPS antenna is attached to the top of the spine. A high-efficiency DC-DC converter and an Arduino controller with an SD card are attached to the spine near the top.

The water-proof housing is a major factor. We had originally intended to use 125 psi PVC irrigation pipe, but we found that 8 AA cells were not sufficient to stabilize such a housing. We finally discovered 0.004” polyethylene plastic tubing which is available in suitable widths at extremely low cost, and was easily heat-sealed with inexpensive tooling. This makes a cheap and robust housing with the advantage of being transparent. We replace the housing after every deployment (and recycle the remains).

The drifters are shown housed and naked in Figure 1. These were built, and the software developed, by one of us (RH), a high-school student with no previous experience in electronics, software, or use of hand-tools. Very loose supervision was provided by another of us (WAC) a very experienced electrical engineer who was not familiar with the Arduino development environment.

Figure 1.

Since we are regarding the drifters as “expendable” we would like to retrieve the data even if we can’t recover the drifter. For this purpose the most cost-effective backup is to have each drifter record its observations on non-volatile storage and also radio them to a master drifter or gateway which is part of the drifter swarm. There are several RF transceivers available for the 915 MHz band with ranges of 500m to a few km that are suitable. The ones we have been testing use a packet switched data protocol referred to as RFM69. Ideally this gateway drifter will also have a cellular modem so the data can be saved in the cloud. This has not been implemented in our swarm as yet

Detailed Design

A block diagram of the drifter is shown in Figure 1. Design began with the battery pack. It is attractive to run the batteries in series providing 12 v if alkaline are used and 10.6 v if NiMH are used. This can be down-converted with a DC-DC buck regulator at very high efficiency. Fortunately such regulators are available at the current rating we need for less than $6 in unit quantities. This expense can be eliminated when the design stabilizes, by connecting the battery array in a series-parallel configuration that will provide close to 3.6 v DC. We also included a switch between the battery array and the load, after some reflection. The extra component is a nuisance and potential failure mechanism, but it is convenient to be able to close up the drifter housing and turn the unit on or off through the housing. Switches cost $0.50 in quantities of 20.



Figure 2.

The Arduino variant that is most attractive for the drifter is called a Moteino. It can be purchased with an RF transceiver, a serial flash RAM, and a built-in “inverted-F” antenna. The antenna is particularly useful because wires to the GPS receiver can be lead past the RF transceiver antenna on the shorted side of the inverted-F without disturbing the antenna pattern, and the antenna is not sensitive to the GPS unit above it on the spine of the drifter. The minimum Moteino is sold for $22 in unit quantities, but we have used a slightly larger MEGA unit in development which costs $9 more than the minimum unit. The flash RAM (4 Mb) increases the cost by only $3 but provides a very useful backup and test facility. An SD card compatible with Arduino adds $4 to the cost and requires that a separate chip, about the size of an SD flash chip, be added to the spine and interfaced with the Moteino. This interface must be done carefully because it uses the SPI bus which is fast enough to cause trouble if the wiring is not done carefully.

The GPS card we have used is the MTK3339 unit from GlobalTop packaged by Adafruit and sold at $30 in unit quantities. It has a built-in ceramic antenna and autologging capability. It includes a real-time clock and can sleep between samples to minimize power consumption. When on it draws 20 ma at 3.3 v. It is a major contributor to the power load.

The software uses existing libraries to interface with: the GPS card, the SPI bus, the SD card, and the RF transceiver. Examples of code that covers the drifter applications are readily available. The only difficulty that we had was operating the RF transceiver and the SD card simultaneously. This was a problem because the RF receiver runs in interrupt mode. So it was possible for an interrupt to occur during an SD card access. Since the two devices shared the SPI bus this was a problem. Once it was recognized it was easily solved by turning interrupts off during the SD card access.

The total parts cost for our developmental units is about $70. We can reduce it to ~$50 by putting an SD card only on the master units, using the minimum Moteino, and eliminating the DC-DC converter.