

Andrew Barnard, WET Labs

Instrument puck/SIAM design review
MBARI, 11/17-11/18/03

Thank you very much for inviting WET Labs your instrument interface and SIAM design reviews. Your development efforts are impressive and your open and forthcoming presentations show a real commitment to providing improved oceanographic data systems for ocean observing.

Design review comments on Instrument “puck”:

The main comments I have are in the structure of the memory space on the puck and how the data is interpreted by the host. You currently have two identifiers encapsulated on the puck, the UUID and the ISI ID. I’m still a bit unclear on the difference between these two IDs, but I gathered that the UUID is a unique identifier for the puck itself as well as the contents of the puck metadata. I assume that the ISI ID is more specific to the instrument itself as well as its associated driver/calibration file. Both values should be static during the deployment. However at one point in the presentation the scenario was given that if a user changed the instrument service (without connecting a new puck or instrument) through the host, say for example to increase the sampling rate, a new dynamic UUID would be assigned in the instrument service without a change to the puck metadata file. To me the ID’ing implementation is overly complex, and it is not clear how these two IDs are linked together. I am concerned about how we would maintain and track the UUID’ and ISI ID’s (which we would need to provide to the customer if they purchased one with one of our instruments). Who controls the ID’ing, the retailer or the consumer? What would make this much easier for us to implement would be a more standardized format for the pucks memory space. Currently, no presentation of IDing, driver naming convention, etc. have been specified. You may want to have a persistent memory that contains at the very least a unique puck ID (which cannot be reset without hardware intervention).

The other general comment I have is in the “standardization” or protocol purpose of the puck. This comment pertains more to how a manufacturer such as WET Labs if licensed to sell the puck would implement this technology for its consumers. For the MOOS purpose, I agree that the memory map and jar file implementation you have developed is well thought out. However, if the puck technology is to be transferred to the greater oceanographic community, I doubt that most other observatories are going to be using a similar system as SIAM (i.e. jar files, RMI’s etc.) Thus, for each user groups purpose, new formats/descriptors and puck file interpreters will have to be developed such that the contents of the puck metadata file can be read and interpreted by whatever “host” or mooring/observatory controller software is used. A puck developers kit would certainly help with this, but I am still worried about the manufacturer having to continually develop custom interfaces for each of its users of the puck, as well as maintaining a master database of UUID’s assigned to each puck. Given that the idea is to make the puck as easy and inexpensive as possible to implement, I wonder how much cost will be

incurred by just developing possibly many custom interfaces and configuring various puck metadata files.

Other comments/questions:

- In an isolated power supply version of the puck, what would happen if the pucks power supply or microprocessor failed during deployment? In other words is there a fail safe to allow instrument data to be passed through if the puck itself fails?
- Virtual Puck: WET Labs currently sells a device called a DH4 which does many of the same functions as a puck. This device, which is really a data acquisition/controller for up to 8 serial and 8 analog instruments, has built in functions for instrument recognition, processing (applying user supplied calibrations), storage, and instrument sampling. How are systems such as this affected the pending patent? Whats the difference between an external and embedded puck?
- The command set for interfacing with the puck is very simplistic (both good and bad). Could the command set be changed to more uniquely identify them as puck commands (i.e. adding puck_ to them) to avoid possible confusion with any instrument commands?
- What interface is used to write the pucks contents?
- Any support for instruments that use Ethernet?
- Hardware design calls for 8 pin connectors, yet most oceanographic sensors are 6 pin or less. By adding an external puck, a new and different cable must now be provided (in addition to the one that connects the sensor to the puck). Unless the puck is embedded in the instrument or a cable, this will definitely impact the cost.
- Interfacing the puck with SeaBird's inductive modem technology, is this even possible?
- What is a sensor? What "instruments" should have pucks? Does a shutter really need a puck?
- There are still many steps required: configuring puck as well as instrument, and thus there is STILL many chances for user error (one of the main reasons for the puck was to reduce the chances of a user mistakenly loading or using the incorrect calibration file). Are there any fail safes for an incorrect puck/instrument pairing? Does the software check the data stream and settings for validation?