

# The 2<sup>nd</sup> Generation Environmental Sample Processor: Evolution of a Robotic Underwater Biochemical Laboratory

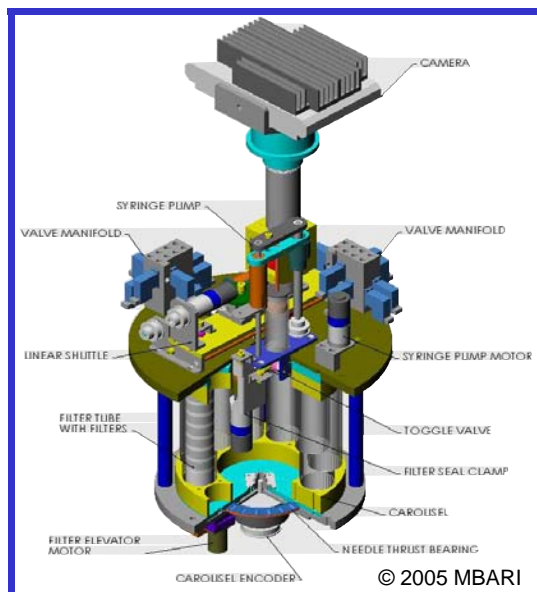
Brent Roman, Christopher Scholin, Scott Jensen, Roman Marin III, and Eugene Massion  
of the Monterey Bay Aquarium Research Institute

Jason Feldman of NASA Jet Propulsion Laboratory

Many of the diagnostic protocols for detecting molecular signatures share an overlapping set of requirements, such as collecting, concentrating, preserving and disrupting cells, applying a series of reagents in a timed sequence, removing particulates, and applying solid phase extraction chemistries, among others. Operationally, these methods can be divided into those that depend on intermolecular reactions, such as antibody/antigen binding, nucleic acid hybridization and enzyme mediated processes, and those that rely on separation and identification of target molecules based on their inherent physical properties. Optical and electrochemical transducers are used commonly to detect and quantify numerous molecular signatures. By developing a relatively low-power, compact instrument that meets this core set of functional requirements, it would be possible to conduct many commonly applied molecular biological analyses remotely, *in situ*.

Toward that goal, a group of scientists and engineers at the Monterey Bay Aquarium research Institute (MBARI) undertook development of the Environmental Sample Processor (ESP; Fig 1; Scholin et al. 1998, 2001, 2005)<sup>1</sup>. The ESP is an electromechanical/fluidic instrument system designed to collect discrete water samples from the ocean subsurface, concentrate microorganisms (particulates), and automate application of molecular probes to identify microorganisms based on ribosomal RNA (rRNA) signature sequences. In addition, the ESP archives discrete samples for a variety of nucleic acid analyses, microscopy and other types of analytical procedures after the

instrument is recovered. To date, applications of the ESP team has focused on detecting a suite of marine planktonic organisms ranging from heterotrophic and photosynthetic bacteria, archaea and eucarya to small invertebrates found in the upper ocean. “First generation” (or “1G”) prototypes of the ESP have been deployed in Monterey Bay and the Gulf of Maine (e.g., Goffredi et al. *in revision*, Scholin et al. 2005).



**Figure 1.** Solid model of the core sample acquisition, processing and storage system of the 1G ESP. The ESP is protected under US Patent No 6187530.

The 1G ESP consists of five major subsystems: carousel, shuttle, clamp, syringe pump, and CCD camera (Fig. 1). The carousel stores up to 100 “pucks”, or reaction chambers, which accommodate a wide variety of user-defined 25 mm diameter filters or

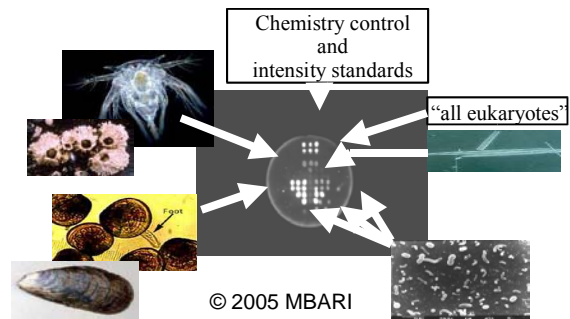
1 <http://www.mbari.org/microbial/ESP>

chemically adsorptive media (Fig. 2). An elevator and linear shuttle are used to move a puck from the carousel to the processing position where it is sealed in a clamp, thus providing connections to the sample port and reagent valve manifolds. The seals used in the clamp have embedded heater pads to regulate the temperature of liquid in the clamped puck from ambient to  $\sim 100^{\circ}\text{C}$ . The shuttle is also used to move pucks to an imaging station where a CCD camera records results of DNA probe array assays (Fig. 3). A syringe pump draws in seawater samples and dispenses the required reagents. Modular valving supports use of up to 16 different custom-defined reagents. Additional valves allow the syringe pump to pull reagents “top-to-bottom” or to push reagents from “bottom-to-top” across the filter or adsorptive medium contained in a puck. Reagents can also be trapped in a puck so that specific reactions can occur over an extended period at a defined temperature. Target molecules eluted (rinsed out) from one puck may be pulled into the syringe pump and used in a protocol requiring a different solid phase medium. Processed or spent pucks are unloaded into an empty carousel tube. Waste may be discharged overboard or captured on-board in a dedicated reservoir. Users control the 1G ESP through a simple ASCII text-based language that defines the sequence of steps to be performed by the instrument, including a “power down” step until the next programmed sampling period or external wake-up command. The current ESP is deployed at a fixed depth on a compliant, subsurface mooring that has an electromechanical cable attached to a surface float and radio modem.



**Figure 2.** Photograph of ESP (1G) Pucks

In order to detect target organisms in near real-time, the ESP employs custom DNA probe arrays for ribosomal RNA (rRNA) sequences that are indicative of specific species or groups of species (e.g., Fig 3). To develop a probe array the ESP breaks down cell membranes, homogenizing a sample using a chaotrope and heat. The resulting filtered sample homogenate is retained. The sample collection puck is replaced with an array puck and the homogenate is applied, followed by a sequence of reagents that reveal rRNA molecules bound at specific locations on the array grid using the principles of sandwich hybridization (Scholin et al. 1996, 1997, 1999). An image of the resulting array is transmitted to a remote location for interpretation. Different arrays are tailored to specific groups of organisms such as “planktonic microbes”, “harmful algae,” or “invertebrate larvae,” etc. In this fashion the specificity of the arrays can be readily altered. Thus, the ESP supports detection of many different target molecules using a common methodology, suite of reagents and core sample processing instrumentation.

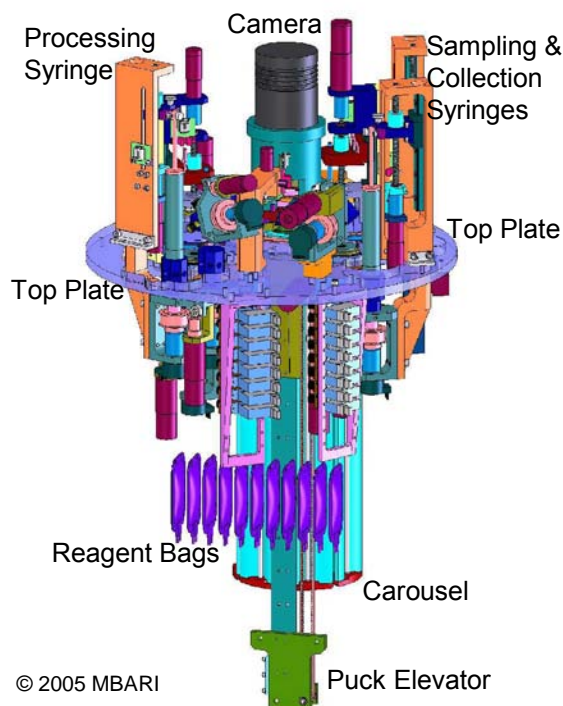


**Figure 3.** Example 25cm diameter DNA probe array for the ESP. This array carries probes for a variety of toxic phytoplankton present in Monterey Bay of California.

### Design Evolution

Development of a second generation, or 2G, ESP (Fig. 4) is being carried out with support from the National Science Foundation (NSF, OCE-0314222) and by MBARI. The prototype 2G ESP is currently being tested in MBARI's labs and surface ocean (<50m depth) deployments are scheduled for late 2005. The

2G ESP design goal is to develop a modular core instrument system that can be easily reconfigured to suit a wide variety of deployment and analysis scenarios to make it more accessible to outside research groups. The detailed, technical goals for the 2G ESP were to lower electrical power consumption so conventional rechargeable batteries could be used for 6 to 12 month deployments, to reduce size to allow deployment and recovery by smaller ships, and to improve the serviceability, reliability and flexibility of mechanical systems and software.

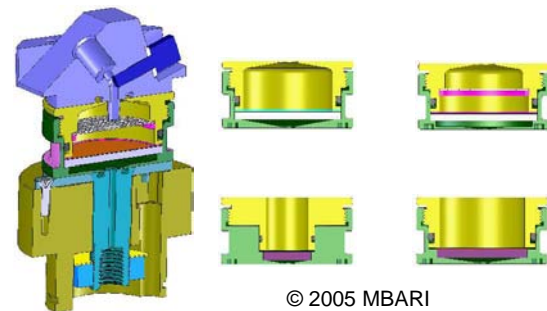


**Figure 4.** Solid model of the 2G ESP depicting rotating carousel, CCD camera, puck clamps, syringe pumps, puck manipulator, reagent bags and valves (some structural elements omitted)

The core of 2G's modular design consists of three syringe pumps and an array of rotary and solenoid valves for sampling, concentrating small to moderate sized raw water samples (10ml to 2L), archiving, and development of DNA probe arrays or competitive ELISAs. This core also provides well standardized fluidic, power, and communications interfaces

for optional internal “analytical modules”, and an interface to an external sampler to meet specialized requirements such as deep-water (up to 4000M) and/or larger sample volume concentration. Development of such a “Deep ESP”, funded by the Keck Foundation, underway now at MBARI.

The basic operations of both the 1G and 2G ESP are very similar. A raw water sample is concentrated by pumping it through a filter sealed in a metal “puck” by halves that screw together around the filter's perimeter. Some pucks variants have large internal volumes (~4ml) to keep samples wet. Others have small internal volumes (~0.5ml) to conserve processing reagents. Imaging pucks have opened tops so that the filter medium can be photographed. All pucks are of the same height and diameter to ease robotic handling and storage in the puck carousel. (Fig. 5)



**Figure 5.** Models of four 2G ESP Puck variants and a cutaway model (left) of clamped puck including a pressure sensor (in dark blue)

### Mechanical Design

Both ESP designs employ a rotating carousel containing filter pucks stacked in vertical tubes and elevators that lift a stack of pucks in a carousel tube to the top plate where all processing stations are located. The first generation prototype features a shuttle that is essentially a puck diameter hole on a linear track driven by a ball screw actuator. After an elevator has lifted the puck stack in the selected carousel tube such that the bottom of the top puck is flush with a hole in the top plate, the 1G's linear shuttle simply slides the protruding puck off the stack and onto the

plate. It then drags the puck further along the top plate to each processing station in turn. After the puck has been completely processed, the 1G shuttle continues over another hole at the opposite side of the top plate, where it is lowered back into the opposite side of the carousel by another elevator.

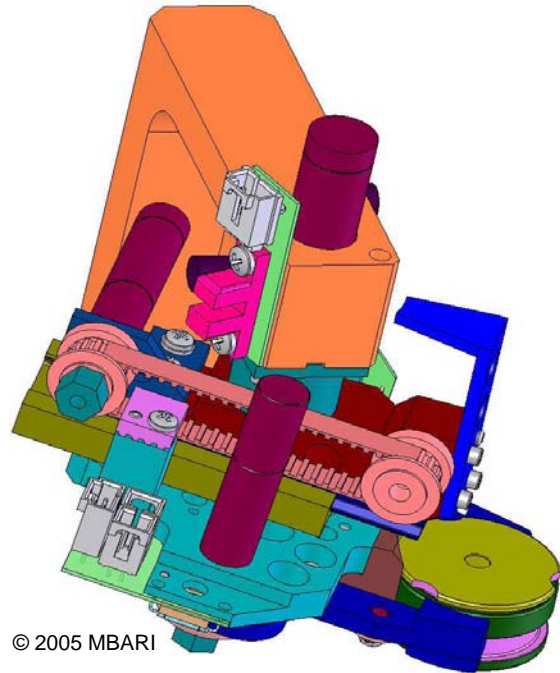
It was eventually determined that dragging metal pucks across the aluminum top plate resulted in contamination of their contents with aluminum dust that reacted with reagents used in subsequent processing. To avoid metal contamination, in the 1G prototype, mylar adhesive backed tape was applied over the area where pucks slid on the top plate.

The 2G ESP replaces the linear shuttle with a 3-degree of freedom, rotary manipulator arm that grips pucks, lifts them off the elevator stack and moves them to processing stations on the top plate. (Fig 6) The arm has a two prong gripper as its end effector. The gripper rides on a precision linear ball bearing slide that retracts and extends. This sliding forearm attaches to an rotating elbow joint and the center of the top plate. It is important to note that the forearm slide is oriented such that, as the puck is retracted toward the center of the top plate, it is also lifted off it. Since the top plate is no longer being contaminated with drips from wet pucks sliding across it, the 2<sup>nd</sup> Generation ESP no longer requires separate load and unload elevators.

Once slid off the top of a puck stack, the 1G's linear shuttle design can only release that puck by dropping it down a hole in the top plate. The 2G's manipulator arm can release pucks at processing stations as soon as they have been clamped into position there, thus enabling the the 2G ESP to process multiple filter pucks on the top plate in parallel. The rotary manipulator also removes the need for the 1G's second puck elevator.

The 2G ESP has three syringe pumps. These facilitate parallel processing, allow syringes and drive motors to be sized appropriately for specific tasks, and, most importantly, eliminate the possibility of the raw sample contaminating reagents via the single, shared

pump. As currently configured, the core includes a high speed 25ml syringe for sampling, a 10ml syringe for collection, and a 2.5ml high accuracy syringe for processing with low volume reagents such as DNA probes.



**Figure 6.** Model of 2G's 3 Degree of Freedom manipulator with filter puck held in gripper.

All valves in the first generation design are solenoid-type flapper valves. Selection among more than two fluid sources requires ganging solenoid valves together into complex manifolds with relatively large “dead” volumes that need repeated flushing. The 2G design replaces most of these with four, six and eight-way rotary valves. (Fig 7) The rotary valve assemblies are much smaller than the equivalent network of solenoid valves and have little or no “dead” volume. Each of these rotary valves utilize a small, very repeatable magnetic angle sensor that was not available at the time the 1G ESP was designed.

Both ESP designs employ many high speed DC brushed servomotors with inexpensive phase-quadrature incremental encoders on

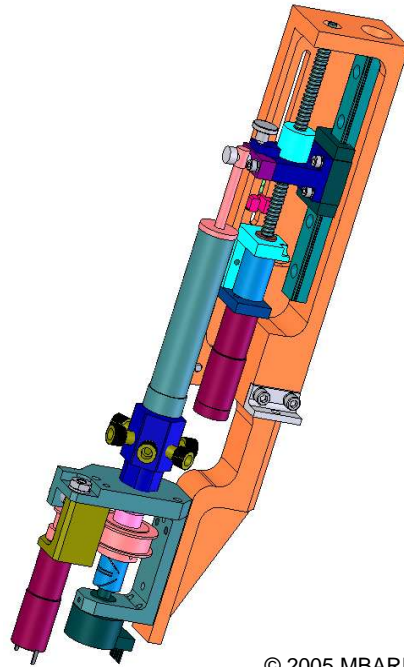
their output shafts for accurate position control. This type of encoder requires external specification of an absolute reference point to calibrate its absolute position. The 1G ESP uses mechanical end-of-travel limit switches for this purpose. However, these occasionally fail – especially when they get even slightly wet. As a result, the 2G ESP design team chose to eliminate end-of-travel indicators entirely. Instead, a Hall-effect “home” flag transitions at some arbitrary reference point in each axis’ travel to calibrate its encoder. The state of the flag always indicates the direction in which the “home” point can be found. Each axis thus “finds home” when reset without being run to its travel limits. Once calibrated, all subsequent positions are expressed as offsets from the home flag’s transition point.

In an effort to make the ESP operate faster and be less likely to damage itself if a sensor fails, the 2G team chose motors and gearheads such that the stall torque of each motor is only two to three times the peak needed for normal operation. In practice, this results in a dramatically more responsive instrument that does not damage itself if an engineer or technician does something unthinkable, like forget to plug in a home sensor.

The 1G ESP lacks pressure sensors, making it necessary to sample very slowly and to stop early to avoid any reasonable possibility of the filter clogging. Its sampling rate is fixed at 4ml / minute. The 2G’s dedicated sampling syringe can pump up to 50ml/minute. While sampling, the 2G ESP queries pressure sensors above and below the clamp that seals the filter puck (see Fig. 5) to maintain the maximum flow rate across the filter medium that does not exceed its design pressure gradient. Thus, the sampler automatically adapts to the turbidity of the raw water. Sampling slows gradually as the filter loads with particles and ends when the maximum flow rate falls below a preset threshold, before the filter becomes completely clogged.

As in the original ESP design, at the end of the sampling procedure, particles on the filter medium are either chemically preserved for whole cell archival or homogenized for

further processing, such as DNA species identification.



© 2005 MBARI

**Figure 7.** Solid model of ESP 2G syringe pump with rotary valve at its base.

### Electrical Design

The 1G ESP controller consists of a single, centralized stack of eight off the shelf PC/104-bus based boards. This includes the main Intel 486 based processor and dedicated servo control boards for each pair of motors. Commercially available hardware minimized the in-house electronics design needed to bring up the initial prototype. However, this commercial PC/104 stack has a relatively high constant power consumption: nearly 20 watts of “hotel” load whenever the ESP is active. This high, constant load, well over 75% of the total power consumption, precludes the use of inexpensive rechargeable batteries. A custom 300 Amp/Hr, 14.4 volt primary lithium pack is typically used for each deployment.

One of the primary goals for the 2G ESP was to reduce this hotel load to less than 4 watts when active and less than 10 milliwatts while sleeping between samples. This would allow a

six month deployment powered by a single 100 Amp/Hr lead-acid battery.

Five years between designs helped to meet this goal. The main processor is now an ARM-based PC/104 board roughly equivalent in speed to a 400Mhz Pentium but consuming less than 2 watts. This standard, off-the-shelf PC/104 board plugs into the underside of a 15inch diameter, round custom main board near the top of the ESP pressure housing. The main board also contains: DC-to-DC power supplies allowing the ESP to run on between 10 and 16 volts, eight serial channels with isolation for external “contextual” sensors like CTDs or fluorometers, and a single, ultra-low power microcontroller that controls power to all other subsystems. The main board's microcontroller is the only one that remains powered when the ESP is “asleep” between samples. While sleeping, the 2G ESP draws less than 15mW.

Five MBARI designed, interchangeable servo controllers are mounted above the top plate (see Fig. 4). They are positioned near the actuators they control to minimize the cabling length. Each servo controller consists of a common core board, with one or two plug-in daughter boards. Each core board includes a TI MSP4300 microcontroller and a Xilinx CPLD. It provides for two channels of DC PWM servo motor control with incremental encoder inputs, plus one PWM output for a heater and associated analog inputs for temperature and pressure sensing.

These core servo controllers communicate via a built-in I<sup>2</sup>C 100kbit/s serial bus<sup>2</sup> on a single, common ribbon cable that runs around the perimeter of the inside of the pressure housing and terminates at the main board's microcontroller. Optional daughter boards, each with its own CPLD, control 8 high-current solenoids (valves) and 4 rotary valves. Daughter boards do not have their own microcontrollers. The total power consumed by all five servo controllers is slightly less than 2 watts.

---

<sup>2</sup> <http://www.semiconductors.philips.com/markets/mms/protocols/i2c/>

Due to the the reduced hotel load of the 2G, power draw while operating varies greatly. It averages roughly 10 watts, with the 4 watt hotel load making up less than 50% of the total. Furthermore, the dramatic speed improvement of the 2G design over the original and its ability to process samples at multiple stations simultaneously should confer a significant power savings simply by allowing it to sleep a far greater percentage of the time between samples.

### Firmware Design

Firmware, written in ANSI 'C', running on the above described distributed microcontrollers closes all real-time servo control loops. This frees the main host processor from all critical real-time processing constraints so that it may, in principle, run any mainstream operating system.

Each of the servo microcontroller's nonvolatile flash memory contains *identical* firmware. This not only eases development, but it also aids troubleshooting and repair, as any core servo board may be swapped for any other after simply setting a DIP (dual-inline) switch to assign it the appropriate address on the I<sup>2</sup>C bus. The host processor configures each servo controller for the actuators they control, as indicated by its bus address, each time system power is reapplied.

The I<sup>2</sup>C bus was originally intended to communicate among chips on a single circuit board or closely connected set of boards. MBARI has extended the low-level I<sup>2</sup>C bus protocol to include a checksum and a single acknowledge bit for all messages so as to detect transient errors (such as those caused by induced electrical noise on a ribbon cable), and retry. These MBARI-specific extensions can be disabled to allow communicating with any standard I<sup>2</sup>C bus peripherals.

Typically, I<sup>2</sup>C peripherals such as memory chips or environmental sensors are slaves on the bus that respond immediately to requests. The requesting master holds the bus for the (very short) time it takes them to process each request. Such a protocol is simple to

implement, but it would be *very* inefficient when applied to servo control as requests to reposition an actuator often take many seconds to complete, during which time no other bus traffic can occur.

The 2G ESP's servo controllers avoid this bus congestion by communicating on the I<sup>2</sup>C bus as multi-mastering peers. Whenever a request arrives that could take time to complete, the servo controller merely acknowledges to the requester that the message was received and starts the requested operation. The servo controller later masters the I<sup>2</sup>C bus to send a completion message back to the original requester after the operation is done.

Each request message contains a tag, assigned by the requester, that the servo controller echos back in its completion reply message. The request tag allows a multi-threaded requester to differentiate among multiple outstanding requests, dispatching each reply its waiting thread as they are received. This combination of multi-mastering and message tags facilitates efficient bus use by multi-threaded applications.

The main board's microcontroller is the only microcontroller running different code than all the servo controllers. Its dedicated RS-232 command port, clocked at 115.2 kbaud, serves as the gateway between the host, the controllers on the I<sup>2</sup>C bus, and the main board's eight serial ports. This gateway multiplexes all control of the 2G ESP onto this single serial port, which is normally connected to the PC/104 ARM CPU host via a short patch cable. However, the host processor can be removed from the ESP chassis by attaching it via a longer serial cable so that a standard laptop may be used in a laboratory setting.

### Software Design

The 2G ESP software redesign initially considered adding standard programming language features such as loops, conditionals, variables, parameterized functions, and multi-

threaded execution to the simple macro language MBARI had developed for the 1G ESP. However, it quickly became apparent that adopting an already developed, general purpose scripting language would be a far easier and maintainable choice than extending our own. Languages whose source code was “open” were clearly preferred as they are generally free of licensing fees and can be modified in-house if the need arises.

This narrowed the field to general purpose, open-source scripting languages such as Tcl, Python, Perl and Ruby. Although less popular than some of the other contenders, Ruby<sup>3</sup> stood out because it combines an intrinsically object oriented, easily extensible language with an expressive syntax that is free of unnecessary punctuation such as leading whitespace, trailing semicolons, and parentheses for arguments in simple function invocations.

Originally, the intent was to extend Ruby with 'C' or C++ code to support low-level operations on the ESP hardware. Engineers on the project also wanted direct, low-level access to servo controllers for testing purposes. The most direct way to provide them this was to represent the entire system as a set of Ruby scripts – from the high-level protocol descriptions characteristic of the 1G's macro language, down to Ruby scripts that actually input and output bytes received from the gateway microcontroller's serial port. As a result, all aspects of the 2G's application software can be easily modified on the embedded host by anyone with an understanding of the Ruby language. This enables even “system-level” code modifications by engineers at sea, who may be without access to specialized compilers and other software development tools.

With the entire system coded in an interpreted scripting language like Ruby, concerns were raised that the software might be too slow to be usable. In practice, the embedded host takes about 5 seconds to start the Ruby application. However, once the application is

---

3 <http://www.ruby-lang.org>

running, commands are seen as running as fast as they are input. Since most commands involve moving actuators that take seconds to complete, the few additional milliseconds delay Ruby imposes to send from the host to the servo controllers them is simply not perceived.

Another goal for the 2G software design was to add the ability to simulate scripts that describe protocols to check them for errors and estimate the resources they would consume. This simulation is implemented by having the application route the messages that would normally go to servo controllers on the physical I<sup>2</sup>C bus to virtual software models of those controllers instead.

The current implementation of these virtual servos is very crude. Time and power consumption can be estimated and some erroneous operations will be trapped, such as those that cause illegal commands to be output to servo controllers. However, the models cannot yet detect more common errors such as a move that is blocked because one actuator is in the way of another (e.g., an attempt to move the puck manipulator arm while the puck it is gripping is clamped into place).

Activity of the ESP during deployments occurs in relatively brief bursts separated by long periods of sleep. In fact, the duty cycle of waking over sleeping time on long deployments will typically be less than 1%. Although it would be convenient to utilize the host CPU's low-power sleep modes to preserve the application's state while sleeping, the 50mW or so required for dynamic memory refresh would significantly reduce the maximum deployment time possible. Instead, the gateway microcontroller cuts off the host's power entirely to enter a deep sleep state. The gateway restores power to the host after the requested time elapses or, optionally, if some interesting event occurs.

Each programmed power cycle of the host is called a mission "phase". The host preserves important state information in flash memory, just before its power is cut off, so that it can resume the deployment properly when the next

phase begins. In addition, the host's minimally configured Linux operating system manages to start the next phase of the Ruby mission phase within less than 10 seconds of power being applied.

### Current 2G ESP Development Status

All the active electro-mechanical components of 2G ESP prototype are assembled and in use on a daily basis in MBARI's labs. The 2G prototype has run all the protocols that MBARI had previously run on the first generation instrument. The quality of the initial lab results, from the 2G ESP, in terms of reduced filter deformation and improved signal-to-noise ratios in photographic images of spotted filters, already far exceeds the best ever achieved by the 1G and appears to be even better than that achieved by MBARI's own skilled technicians performing (unautomated) bench top reference protocols.

The 2G's initial deployment is schedule to take place in late 2005. Some deployment specific software components, such as mission phase sequencing, remain unimplemented. Also, the new pressure housing and mooring are not yet completely assembled and tested.

In the meantime, MBARI will develop and test the 2G ESP in its labs. In particular, researchers will continue exploring potential benefits of parallel filter puck processing and other performance enhancements made possible by this new design.



## References

- Goffredi, S.K., W. Jones, C. Scholin, R. Marin, S. Hallam, R.C. Vrijenhoek. Molecular detection of marine larvae. *Marine Biotechnology* (in revision).
- Scholin, C.A., G.J. Doucette and A.D. Cembella. 2005. Prospects for developing automated systems for *in situ* detection of harmful algae and their toxins. In: *Monographs on oceanographic methodology* (Babin, M., Roesler, C, and Cullen, J. eds). UNESCO (in press).
- Scholin, C.A., Massion, E.I., Wright, D.K., Cline, D.E., Mellinger, E., Brown, M. 2001. Aquatic autosampler device. US Pat. No. 6187530.
- Scholin, C., R. Marin, P. Miller, G. Doucette, C. Powell, J. Howard., P. Haydock and J. Ray. 1999. Application of DNA probes and a receptor binding assay for detection of *Pseudo-nitzschia* (Bacillariophyceae) species and domoic acid activity in cultured and natural samples. *Journal of Phycology* **35**: 1356-1367.
- Scholin, C.A., P. Miller, K. Buck, F. Chavez, P. Harris, P. Haydock, J. Howard and G. Cangelosi. 1997. Detection and quantification of *Pseudo-nitzschia australis* in cultured and natural populations using LSU rRNA-targeted probes. *Limnology and Oceanography* **42**: 1265-1272.
- Scholin, C., G. Massion, E. Mellinger, M. Brown, D. Wright and D. Cline. 1998. The development and application of molecular probes and novel instrumentation for detection of harmful algae. *Ocean Community Conference '98 Proceedings*, Marine Technology Society, Vol. 1 pp. 367-370.