

**pH Limitations in Environmental Geochemical Monitoring**

**Rebekka Gould<sup>1</sup>**

**UW School of Oceanography Undergraduate Research Thesis**

**Spring 2015**

**<sup>1</sup>University of Washington School of Oceanography, Box 357940 Seattle, WA 98195**

**Contact Information:**

**Rebekka Gould**

**rhgould@uw.edu**

---

## Acknowledgements

A sincere thanks is owed to my mentor and adviser Rick Keil, PhD. Thank you for all of your help over this past year and for remaining enthusiastic even when my resolve wavered. I am extremely grateful to Wendi Ruef for her patience and insight into the Arduino IDE, design and build process, and what it means or, more aptly put, takes to learn. To Miles Logsdon, PhD., I am entirely appreciative of your help and guidance aboard the R/V Thompson –although it was exemplified aboard the ship, you always provided me clear and honest guidance, while maintaining my faith in my own abilities. I would like to thank Jim Murray, PhD., for helping me to get in contact with Senior Research Specialist Hans Jannasch, PhD., of the Monterey Bay Aquarium Research Institute, and Hans for his commitment to developing cutting edge technology that will (and has) enable scientists to explore our world's oceans and provide data to inform conscientiousness global citizens. Hans went above and beyond to help me succeed in understanding the engineering of the DuraFET and how to interface the DuraFET with the Arduino. Arthur Nowell, PhD., was an invaluable mentor whose patience and enthusiasm never ran short, and also provided great London travel tips! Lastly, I would like to thank Al Devol, PhD. for his assistance with my project, all of the professors who taught Senior Thesis, River Systems Research Group for their support and encouragement, my parents (and entire family) for helping me defeat writer's block with a week back at home in California, and the wonderful, beloved graduating class of 2015 –Congrats!

## Abstract

Development of methods for measuring *in situ* remineralization rates of carbon in the ocean is necessary to accurately model the ocean's capacity to sequester atmospheric carbon dioxide and the future ramifications of climate change, including ocean acidification and increases in sea surface temperature (Devol and Hartnett 2001; Boyd and Trull 2007; Buesseler et al. 2007; Workshop 2008; Honjo et al. 2008). Currently, the Keil Lab has developed PHORCYS –PHotosynthesis, Respiration, Carbon balance Yielding System—a sediment trap incubator that measures *in situ* respiration rates. PHORCYS is restricted to a maximum depth of 200 meters due to the rating of the pH sensor and data logging package. Thus, this study seeks to improve Phorcys' system so that *in situ* respiration rates can be measured up to 1000 meters below the ocean's surface. This will be accomplished by retrofitting PHORCYS with a DuraFET iii pH electrode, and switching from a YSI EXO2 Multiparameter Data Sonde ® to an Arduino-based data logger that will serve as the communication center between the instrument and probes. These changes will enable greater sensitivity in pH measurements (from .01 to .001 pH units precision), and allow measurement of *in situ* remineralization rate at depths up to 1000 meters.

## Introduction

Within the last century, anthropogenic carbon dioxide emissions have greatly perturbed the natural equilibrium and disrupted global climate (EPA 2014). While the relationship between increased carbon dioxide and the greenhouse effect is well understood, less constrained is the effect of global ocean biology, chemical equilibrium and physical forcing on carbon sequestration within the world's oceans (Jin et al. 2008; Schneider et al. 2008; Iversen and Ploug 2010). Development of instruments capable of measuring the remineralization process *in situ* is key to better modeling the

biological pump and gauging the ocean's potential to sequester carbon dioxide (Boyd and Trull 2007; Cooley et al. 2007; Schneider et al. 2008). To address this concern, this study seeks to improve upon the Keil Lab's sediment trap incubator: Phorcys (PHOTOSynthesis, Respiration, Carbon balance Yielding System). Phorcys is a device capable of measuring organic carbon remineralization and conducting experiments *in situ* without the need to bring water or samples aboard ship.

During November and December of 2014, lab and field experiments were performed to evaluate the existing system. These tests and their results, along with the design specifications for Phorcys, are described in appendix A. After assessment, three main issues with the current YSI EXO2 pH probe were identified: 1) pH sensor drift, 2) sensor accuracy, and 3) sensor sensitivity. While drift can be addressed through calibration and accuracy is not an overarching concern because calculations are contingent upon the change not the unit of change, the issue of sensor sensitivity is a major concern. pH probe sensitivity is important when trying to assess remineralization rates in anoxic environments where discrete changes in oxygen concentration are too small to be registered by modern oxygen probes (Glazer et al. 2004; Edwards et al. 2010). Sensor sensitivity is also an issue in open ocean oligotrophic waters where changes are small. Thus, a probe with greater sensitivity is preferable when working in oxygen limited environments, such as the bathypelagic zone or fjord environments.

An additional concern is that the YSI EXO2 Multiparameter Data Sonde is only rated to 200 meters depth. Phorcys needs to be able to operate deeper in the ocean. This is because remineralization of organic carbon in shallow water leads to degassing of CO<sub>2</sub> back to the atmosphere through physical processes, but remineralization deeper in the ocean stores CO<sub>2</sub> in the ocean for hundreds to thousands of years (Lutz et al. 2002; Boyd and Trull 2007; Buesseler et al. 2007; Herndl and Reinthaler 2013). Thus, the ability to measure *in situ* respiration and effective

remineralization rates at greater depths is vital to modeling future atmospheric CO<sub>2</sub> levels (Cooley et al. 2007; Schneider et al. 2008).

The goal of this study is to address the sensitivity limitation of Phorcys' pH meter and the depth limit of the sonde. This requires use of a new pH sensor (Honeywell DuraFET iii), and integration of the new sensor and a new data storage unit into an existing Arduino-based microcontroller system. Thus, this study is the first step toward engineering a completely new data sonde to replace the YSI EXO2. This will demonstrate the capacity of future integration of the sensor, burn wire, injection and data storage into one Arduino board operating system (see Appendix A), thereby extending the operating depth of Phorcys from 200 meters to 1000 meters and enabling the measurement of *in situ* remineralization at greater depths. Retrofitting Phorcys with an integrated Arduino based data management and acquisition system will provide a cost effective alternative to the current system and provide complete control of data stream processing. This study aims to elucidate current limitations, as identified by Boyd and Trull 2007, of modeling the biological pump and suggest a means of data acquisition to better gauge the ocean's potential to sequester CO<sub>2</sub>.

## **Materials and Procedures**

Replacement of the pH sensor required a three-pronged approach: a) the design requirements needed to be assessed and a power supply and data logging system had to be engineered to meet the design specifications; b) the Arduino microprocessor and Software (IDE) needed to interface with the DuraFET iii and the code had to be altered to convert the output signal to pH values, and c) the electrical noise associated with the SD shield needed to be assessed. Appendices for each step of the three-pronged approach are provided for reference. These Appendices contain code with comments, circuit diagrams, and any supporting data relevant to the design process.

### ***Phorcys and Design Parameters***

The ultimate goal of this study was to construct the initial code and circuits for integration of the sensor powering, data acquisition, mechanical operations and injection system of Phorcys. The code constructed to acquire data from the DuraFET iii followed the same logic as the existing code for Phorcys. The subsequent outline of data acquisition was as follows: power probe, collect and store data, turn of probe and supply power elsewhere. Following this general outline, time statements were inserted and Boolean logic was used to establish the duration of power supply and data collection from the probe.

Initially, a SparkFun Electronics Temperature Probe was used as a proxy for the DuraFET iii. This probe outputs a voltage signal, which is the same as the DuraFET iii, and its leads could be inserted directly into the bread board of the circuit making it a convenient analog to the DuraFET iii. This probe was also selected because of the wealth of code available for it within the Arduino IDE. Lastly, an Arduino Micro SD Shield and a one gigabyte Toshiba Micro SD card were used for data storage. Further development of the proxy circuit and DuraFET iii initial code, including diagrams, coding with comments and schematics, can be found in Appendix B.

### ***DuraFET iii Interface and Calibration***

The DuraFET iii pH Electrode with Vario Pin Connector was selected to replace the current YSI pH electrode because there is a standardized method of calibration and deployment, peer-reviewed studies demonstrating a reproducible sensitivity of .001 pH units, and evidence of extending the operational depth of the DuraFET iii to below 200 meters (Martz et al.. 2015, PC Hans Jannasch). The connection between the DuraFET iii and Arduino Microcontroller was made with a Honeywell CAP Adapter. The CAP Adapter serves as a preamplifier and conditioner. Housed within it is an electronic chip that ensures a constant current is supplied to the probe. It is important to note that the CAP Adapter plays an integral role in maintaining quality and reproducible pH

measurements. The CAP Adapter mates with the Vario 11 pin connector, and ten wires exit the connector housed within a 20 foot cable. The CAP Adapter used in this study was provided by Senior Research Specialist Hans Jannasch, PhD., of the Monterey Bay Aquarium Research Institute, and was supplied with the wires exposed and with a key for the color-coded wires. Table 1 provides information regarding the function and power requirement for each wire. It is important to note that without the proprietary information for the Vario 11 Pin Connector, there is no way to reproduce the functionality of the CAP Adapter and acquire sound data.

Table 1. Proprietary connection codes for the cap adapter cable

<b>Color:</b>	<b>Orange</b>	<b>White/Black</b>	<b>Black</b>	<b>Blue</b>	<b>Green</b>
<b>Function</b>	Signal Voltage	Signal Voltage	Ground	Positive Voltage	Negative Voltage
<b>Power</b>	-0.2 to +0.2 V output	-0.8 to -1.2V output	NA	5.2 V input	-5.2 V input

The DuraFET iii requires a bipolar power supply that can supply both positive and negative voltage and can act as a sink or source for current. Since a bipolar voltage supply was not available, one was constructed for this study using two MPJA DC Power Supply 14602PS voltage supplies that shared a common ground. One supplied the +5.2 V and the other supplied -5.2 V to the corresponding Cap Adapter wires. The Cap Adapter, Arduino, and both voltage supplies also shared a common ground.

The Arduino microcontroller can only receive positive signal voltage ranging from 0 to 5V. Negative voltage output could damage the Arduino or CAP Adapter. To address this concern, a GB Instruments GDT-11 volt meter measured output voltage and if the voltage output was positive, a secondary reading of the pH was recorded using the Arduino. For conditioning and calibration of the probe, the Arduino received the signal voltage output with an upper limit of +0.2V.

Once power was supplied to the DuraFET iii, the electrode was allowed to condition for 24 hours, in accordance with the method provided by Martz et al. (2010). The probe was conditioned in a pH 7 buffer solution and allowed to stabilize for 24 hours; initial and final pH values were recorded. A VWR symphony SP70P pH meter and electrode were used to make reference pH measurements. This electrode was calibrated using the two-point calibration method provided by VWR (VWR International 2008). After 24 hours, the DuraFET iii was removed from solution and rinsed with DI water. The pH of a pH 10 buffer solution, pH 7 buffer solution and pH 4 buffer solution were measured using the same method and in that respective order. The solutions were at room temperature, 21.4 °C, and it was assumed that this temperature did not vary over the course of the calibration period.

Measured pH values were obtained by inputting the output voltage into the MATLAB script provided by Bresnahan et al. (2014). This script is specific to SeaFET, which is a pH sensor that utilizes two pH probes: an external probe that is chloride specific and an internal DuraFET. To make the script DuraFET specific, a value of 0 was used for the external electrode potential.

The standard electrode potential was calculated using pHCalib.m and the measured voltage output for initial measurement of pH 7 buffer solution. The measured output voltage value of 0.014 volts was input in to the script, along with a temperature value of 20°C, a CalExt value of 0 volts, and a salinity value of 0psu. This resulted in a standard cell potential value of -0.3987. The measured voltage output of the DuraFET iii for the subsequent solutions was input into pHCalc.m, which references the calibration coefficients calculated in pHCalib.m.

The addition of the SparkFun Electronics microSD shield and incorporation of a 1 GB Toshiba microSD card enable data logging and storage. The proxy code developed for the temperature probe was altered to enable the acquisition and storage of voltage and pH values. pHCalc.m was altered to enable the direct conversion of voltage to pH within the Arduino IDE. The

commented code, *SD Logging*, can be found at the end of appendix B. A MPJA DC Power Supply 14602PS voltage supply supplied direct current of .16 volts to Arduino analog 0 pin, and the program was run to verify the accuracy of the calculated pH.

### ***Electrical Noise of MicroSD Shield***

The Datalogger.pde, housed within the SD Examples, was accessed from the Arduino IDE and used to assess the electrical noise associated with the addition of a SparkFun Electronics MicroSD shield to the Arduino Uno microcontroller. Pins were attached to the MicroSD shield and a MPJA DC Power Supply 14602PS supplied 00.1V to analog pin 0. The power supply was shared a ground with the Arduino. Raw output was stored on the microSD card as a comma separated variable file. The file was imported into excel and the first column of data, corresponding to output from analog pin 0, was converted to voltage using the formula provided within the USK Circuit 7 code. A total of 2466 samples were made, and the calculated output voltage was averaged for the sample population. Standard deviation was calculated utilizing the Excel formula STDEV. Variance was calculated by dividing the standard deviation by the square root of the number of samples. A GB Instruments GDT-11 volt meter was used to make a reference measurement of voltage output from the MPJA DC Power Supply 14602PS.

### **Assessment**

Calculated pH values from the measured voltage output of the DuraFET iii were consistent (+0.03pH units) with the pH buffer solutions expected values (Table 2). The measured pH returned by the SympHony pH electrode were consistently less than the expected pH; this is most likely due to the temperature dependence of pH. However, assessment of sensitivity for the DuraFET iii is beyond

the scope of this paper, as it has been thoroughly addressed within contemporary literature (Martz et al. 2010; Bresnahan et al. 2014). Discrepancy between the calculated and measured pH most likely originates from the assumption that temperature remained constant during calibration of the DuraFET iii. Since pH is dependent upon temperature, any unaccounted for change in temperature of the buffer solution would alter the accuracy of the probes.

Table 2. Calculated pH from DuraFET iii calibration

<b>Solution:</b>	<b>pH7 (Conditioning)</b>	<b>pH 10 (Calibration)</b>	<b>pH 7 (Calibration)</b>	<b>pH 4 (Calibration)</b>
<b>Initial Voltage (mV)</b>	27	199	021	-163
<b>Final Voltage (mV)</b>	14	192	015	-160
<b>Calculated pH</b>	6.9936	10.0392	7.0107	4.0164
<b>Measured pH</b>	n/a	9.90	6.98	3.93

Table 2: Calculated pH from DuraFET iii calibration using the final voltage reading from the GB Instruments GDT-11 volt meter after a 24 hour soak in solution. Calculated pH was obtained using MATLAB script provided by Bresnahan et al. (2014). Also included are the measured pH values acquired from the VWR Symphony pH electrode.

Furthermore, when output voltage from the DuraFET iii was plotted against pH of the buffer solution, measured pH, and calculated pH a linear slope of 58.667mV/pH, 58.95mV/pH, and 58.445mV/pH, respectively, was obtained. Thus, the observed Nernst behavior of the DuraFET iii is comparable to the literature value of 59mV/pH (Martz et al. 2010).

As previously mentioned, the sensitivity of the DuraFET iii is not within the scope of this study; however, it is worth noting that within the study's experimental design there is inherent data degradation. The largest contributor to signal loss is the 10-bit A/D converter that is built into the Arduino Uno Microcontroller. Literature based assessment of DuraFET iii sensitivity utilizes a 24-bit A/D converter, which results decreased sample resolution. Sample resolution can be enhanced by

increasing the bits of data transmitted for each sample. This could simply be addressed by utilizing a Raspberry Pi® that has a 17bit A/D converter; however, it is recommended to incorporate a 24bit A/D converter to remain consistent with reviewed studies.

The small data transmittance enabled through the Arduino Uno A/D converter results in a limit of two significant figures for sample measurements. Assessment of the noise associated with the SparkFun Electronics MicroSD Shield indicated no signal loss. The calculated average output voltage was .12mV, with a standard deviation of .003 and sample variance of .002. The measured output voltage was .12mV. However, as previously discussed, the data acquired by the Arduino is limited to two significant figures and, thus, electrical noise attributed to the MicroSD Shield may be washed out by the low data transmittance.

Despite these inherent technical limitations, the Arduino Uno consistently reported precise values for the calculated pH, as verified with pHCalc.m. Both the measured voltage and calculated pH values precise to two significant figures, in accordance with the limitation of the A/D converter.

## **Comments and Recommendations**

The integration of the DuraFET iii and development of an Arduino based data logging system is a suitable replacement for the YSI EXO2 Multiparameter Data Sonde ®. However, it is imperative to note that retrofitting Phorcys with an Arduino Uno microprocessor limits the sensitivity of the DuraFET iii due to the 10 bit A/D converter that is a part of the integrated board. The incorporation of an integrated circuit for power, sensor control and data logging has proven successful, and could be improved with incorporation of a 24bit A/D converter.

Further consideration should be given to the compatibility of the DuraFET iii with Phorcys and mooring methods. Martz et al. (2015), clearly outline concerns and considerations for obtaining

reliable and quality data. The largest concern in regards to the integration of the DuraFET iii with Phorcys is the conditioning and equilibrating period recommended for the DuraFET. It is recommended to equilibrate and condition the DuraFET in an environment similar in pressure, salinity and temperature to that of the field location for five days prior to deployment.

For deployment with Phorcys, soaking the DuraFET iii in standard sea water for 4 days and allowing for the remaining day of equilibration to occur *in situ* would be appropriate. This recommendation is based upon the fact that Phorcys sits passively in the water column once moored. This passive period, before the burn wire system closes the gate valves, would allow for equilibration and conditioning at depth for the recommended time associated with depth-related concerns.

Once properly equilibrated and conditioned, the DuraFET should not lose power. A constant power supply ensures the chip, which houses the signal conditioning components, is functioning. A disruption to the chip's power supply would result in a temporary lag in ion exchange across the liquid junction. Thus, if incorporated, a power system would need to be engineered to maintain power to the chip.

Lastly, the goal as described by Martz et al. (2011), is to engineer a housing for the DuraFET chip that will mitigate the mechanical strain associated with pressure. Currently, SeaFET is only recommended for shallow deployment (20 meters) and is not recommended for profiles (0 to 80 meters). This hinges on the incorporation of a solid-state chloride specific external reference electrode in SeaFET and strain on the chip (Bresnahan et al. 2014). The chloride specific electrode is inherently sensitive to haloclines and slow ascension is necessary to accommodate for the slow response time of the electrode. When considering the mechanical limitations discussed by Bresnahan et al. (2014), and Martz et al. (2015), it is key to note the SeaFET is the model instrument being discussed, which incorporates the DuraFET as the internal electrode. Thus, the main concern with

increasing pressure on the DuraFET is the mechanical strain on the chip. Engineering a pressure case and housing specifically for the chip is necessary to minimize this strain.

Incorporation of the DuraFET iii could be a viable long-term option for improved sensitivity of Phorcys. However, over the duration of this study, a new Ion Sensitive FET, Deep-Sea DuraFET, underwent testing in the Wendy Schmidt Ocean Health Xprize. Honeywell's Deep-Sea DuraFET is not yet commercially available, but has performed at depths as great as 3000 meters

(<http://oceanhealth.xprize.org/news/deep-sea-trials-conclude-wendy-schmidt-ocean-health-xprize> ,

May 22, 2015). Contingent upon cost, the Deep-Sea DuraFET may prove a better candidate for integration with Phorcys, and further investigation into the performance and reproducibility of the Deep-Sea DuraFET should be performed.

## References

- Boyd, P. W., and T. W. Trull. 2007. Understanding the export of biogenic particles in oceanic waters: Is there consensus? *Prog. Oceanogr.* **72**: 276–312.
- Bresnahan, P. J., T. R. Martz, Y. Takeshita, K. S. Johnson, and M. Lashomb. 2014. Best practices for autonomous measurement of seawater pH with the Honeywell Durafet. *Methods Oceanogr.* **9**: 1–33.
- Buesseler, K. O., C. H. Lamborg, P. W. Boyd, P. J. Lam, T. W. Trull, R. R. Bidigare, J. K. B. Bishop, K. L. Casciotti, F. Dehairs, M. Elskens, M. Honda, D. M. Karl, D. a Siegel, M. W. Silver, D. K. Steinberg, J. Valdes, B. Van Mooy, and S. Wilson. 2007. Revisiting carbon flux through the ocean's twilight zone. *Science* **316**: 567–70.
- Cooley, S. R., V. J. Coles, a. Subramaniam, and P. L. Yager. 2007. Seasonal variations in the Amazon plume-related atmospheric carbon sink. *Global Biogeochem. Cycles* **21**: n/a–n/a.
- Devol, A. H., and H. E. Hartnett. 2001. Role of the oxygen-deficient zone in transfer of organic carbon to the deep ocean. *Limnol. Oceanogr.* **46**: 1684–1690.
- Edwards, B., D. Murphy, C. Janzen, and A. N. Larson. 2010. Calibration, response, and hysteresis in deep-sea dissolved oxygen measurements. *J. Atmos. Ocean. Technol.* **27**: 920–931.
- EPA. 2014. Climate Change Indicators in the United States: Ocean Acidity. 1–6.

- Glazer, B. T., A. G. Marsh, K. Stierhoff, and G. W. Luther. 2004. The dynamic response of optical oxygen sensors and voltammetric electrodes to temporal changes in dissolved oxygen concentrations. *Anal. Chim. Acta* **518**: 93–100.
- Herndl, G. J., and T. Reinthaler. 2013. Microbial control of the dark end of the biological pump. *Nat. Geosci.* **6**: 718–724.
- Honjo, S., S. J. Manganini, R. a. Krishfield, and R. Francois. 2008. Particulate organic carbon fluxes to the ocean interior and factors controlling the biological pump: A synthesis of global sediment trap programs since 1983. *Prog. Oceanogr.* **76**: 217–285.
- Iversen, M. H., and H. Ploug. 2010. Ballast minerals and the sinking carbon flux in the ocean: carbon-specific respiration rates and sinking velocity of marine snow aggregates. *Biogeosciences* **7**: 2613–2624.
- Jin, X., N. Gruber, H. Frenzel, S. C. Doney, and J. C. McWilliams. 2008. The impact on atmospheric CO<sub>2</sub> of iron fertilization induced changes in the ocean 's biological pump. *Biogeosciences* **5**: 385–406.
- Lutz, M., R. Dunbar, and K. Caldeira. 2002. Regional variability in the vertical flux of particulate organic carbon in the ocean interior. *Global Biogeochem. Cycles* **16**: 11–1–11–18.
- Martz, T., K. Mclaughlin, and S. B. Weisberg. 2015. Best Practices for autonomous measurement of seawater pH with the Honeywell Durafet pH sensor.
- Martz, T. R., J. G. Connery, and K. S. Johnson. 2010. Testing the Honeywell Durafet for seawater pH applications. *Limnol. Oceanogr. Methods* **8**: 172–184.
- Mclaughlin, K., S. B. Weisberg, a. G. Dickson, G. E. Hofmann, and J. Newton. 2013. Core Principles for Development of a West Coast Network for Monitoring Marine Acidification and Its Linkage to Biological Effects in the Nearshore Environment.
- Schneider, B., L. Bopp, and M. Gehlen. 2008. Assessing the sensitivity of modeled air-sea CO<sub>2</sub> exchange to the remineralization depth of particulate organic and inorganic carbon. *Global Biogeochem. Cycles* **22**: n/a–n/a.
- VWR International, I. 2008. VWR Symphony Meter Specifications Addendum. 8–9.
- Workshop, E. 2008. Controls on Organic Carbon Export and Twilight Zone Remineralization.

## Appendix A.

### *Phorcys: Understanding Current Limitations*

Figure one

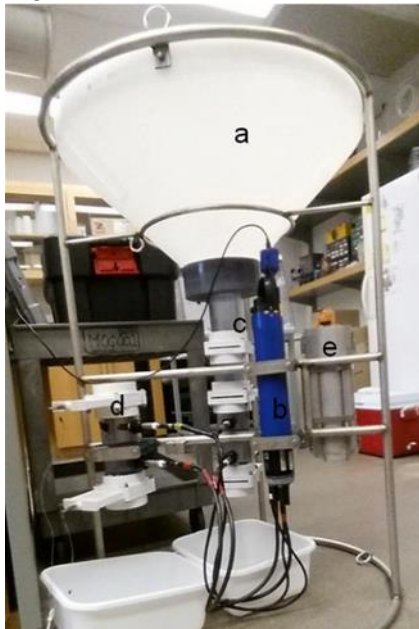


Figure one: Phorcys sediment trap incubator. (a) large funnel, 30 inches in diameter, (b) the YSI EXO2 Multiparameter Data Sonde, (c) sediment trap incubator bottle, (d) serves as the control incubator bottle, (e) pressure case containing the Arduino board.

Fairly simple in design, Phorcys has five main components. There is a 30 inch diameter funnel that directs falling particles into the sediment trap incubator bottle, where Arduino-controlled titanium gate valves compartmentalize sinking particles into two distinct regions or chambers. The lower region is the incubation chamber and pH and DO values are recorded by YSI probes relaying the data to the YSI EXO2 Multiparameter Data Sonde ®. A secondary incubator bottle serves as the control and pH and DO values are also monitored for this bottle. The pressure case contains the Arduino board, which controls the burn system that closes the gate valves. Additionally, syringes may be attached to the instrument and injection of tracers would be timed using the burn system.

Currently, the main limitations of Phorcys are the YSI EXO2 Multiparameter Data Sonde ® and the YSI pH probe, which, respectively, are restricted to 200 meters and only sensitive to .01 pH units. During November 2014, experiments were performed to evaluate the pH probe. Over an 11 day period, the pH and DO of 35 psu standard sea water were monitored in the incubator chambers and data was logged at a one minute interval. From this experiment, three problems with the current pH sensor were identified (Figure A2). The first problem is a constant drift of both pH sensors over the course of the study. The second is an increase of pH

measured by one of the two probes. This increase is anomalous, as the control water should not be changing and this change, if accurate, should have been measured by both probes since they were operating within the same system parameters. Lastly, there is a clear stepwise function to the graph, in which pH drops at 0.01 units over the course of the study. This constant decrease is equivalent to the 0.01 sensitivity rating of the sensor. Thus there is a clear need for the pH probes to be replaced or improved.

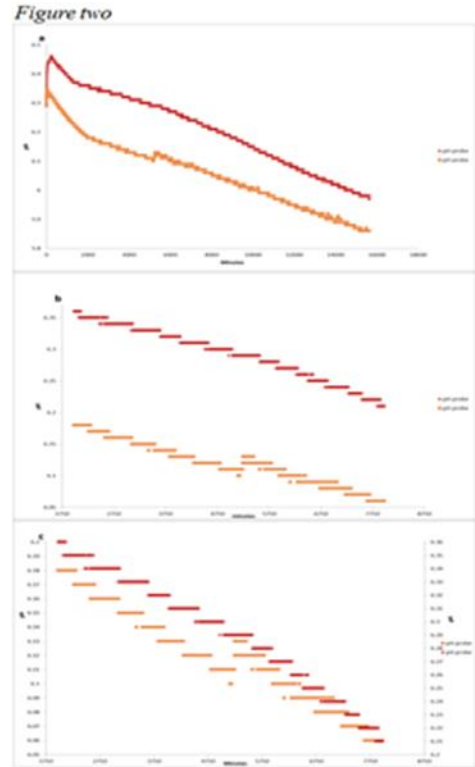


Figure two: The three problems with the YSI pH sensor currently used: (a) drift, (b) inaccuracy evident in a sudden increase in pH measure by one probe, and (c) a stepwise decrease in pH illustrating the sensor limitation of .01 pH units.

During December 2014, a field test to assess the requirements of the mooring method and further assess

Phorcys system parameters was performed utilizing the R.V. Thompson. A sediment trap net was moored in Nootka Sound to collect particles and conduct onboard incubation. A 53  $\mu\text{m}$  sediment trap net was moored in accordance with the Phorcys mooring protocol, such that a 70 lbs anchor was attached to the bottom bridle of the sediment trap. The top of the bridle was attached to two sub-surface vinyl floats. Polydac rope was used for subsurface components, as it remains rigid under strain. The vinyl floats were connected to a surface buoy and small vinyl float with polypropylene rope and a flag and flashers were added for visibility.

The net was deployed on the eastern side of Gore Island in Williamson passage at 49°38.83'N, 126°27.13'W. Sinking particles were collected for a 24 hour period, and the trap was retrieved 28 hours after deployment. Upon retrieval, water from the depth of the mooring (140m) was retrieved using a SBE CTD. The collected water was used to wash the sample from the cod end and

the sample was transferred to Phorcys. The water was transferred to the incubation chamber, carefully poured down the side of the funnel as to limit the amount of bubble injection. A 24 hour onboard incubation of the sample was performed. The control chamber of Phorcys received the same water sample and was filled in the same manner.

Post cruise the data from the onboard incubation was exported and converted into a text file using the KOR-EXO data processor. The data was then imported into Excel©. This completed the assessment of Phorcys.

## **Appendix B.**

### ***Proxy Circuit and Code for DuraFET iii***

A major challenge presented by this study was learning the Arduino IDE and how to develop and write code. Starting with no prior knowledge or experience, this challenge was overcome by carefully following and reconstructing example code and circuits provided housed in the Arduino IDE Library and diagramed in the Vilros Ultimate Starter Kit Guide (SKG). As mentioned before, the design parameters for the circuit and subsequent code were as follows: power probe, collect and store data, turn off probe and supply power elsewhere.

The first step in code development was figuring out how to power and receive voltage output signal from a sensor. A SparkFun Electronics Temperature Probe was selected as an analog to the DuraFET iii for reasons mentioned within the Methods: PHORCYS and Design Parameters. Utilizing the SKG, code from the Arduino IDE Library was uploaded to the microcontroller.

The second step in code development was understand how to supply power to a sensor for a given amount of time and then supply power elsewhere. A relay was incorporated into the circuit to meet this design requirement. Use of a relay was preferred over a push button because the oscillation of power supply and removal needed for autonomous and continuous data acquisition during the

incubation period. To accomplish this, Circuit 7 (Temperature Probe) and Circuit 11 (Relays) from the SKG were integrated. A time statement was constructed utilizing the millis() command, that provided the conditions to regulate the power supply and data streaming, The commented code for this circuit is provided at the end of this Appendix (Integrated Circuit Code).

The controlling circuitry was developed by integrating the SKG Temperature and Relay circuits. Fritzing open-source hardware was used to construct diagrams 1-3, which represent the temperature circuit, relay circuit, and integrated circuit, respectively. Diagram 3 clearly illustrates the integrated circuit. The relay works by having one normally open port and one normally closed port. The normally closed port was selected to power the temperature probe, and, as commented in the integrated circuit code, the relay switches every 6 seconds, powering the temperature every-other six seconds. The normally open port supplies positive current through it, and a jumper wire removes the positive current from the circuit. This wire represents the “power elsewhere” clause designated in the code outline at the beginning of Appendix B.

Diagram 1

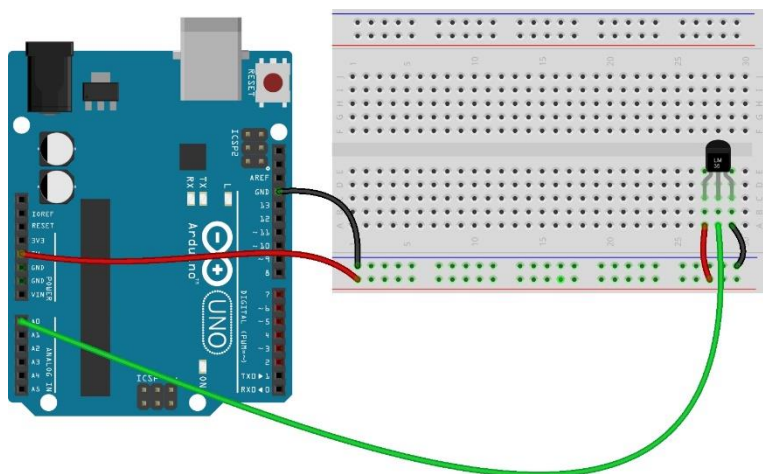


Diagram 1: Temperature Sensor circuit constructed using SKG. Pictured is the Arduino Uno Microcontroller, breadboard half, SparkFun Electronics Temperature Sensor, and jumper wires. The red wires represent input voltage (5V), black wires represent ground, and the green wire represents analog communication with analog pin 0.

Diagram 2

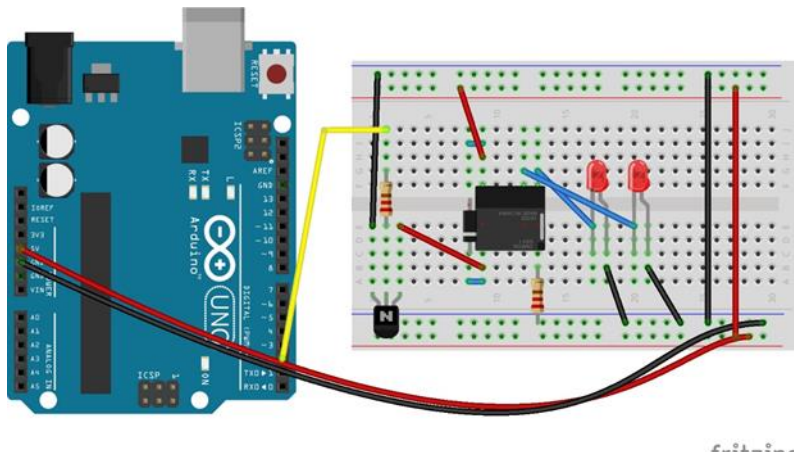


Diagram 2: Relay circuit constructed using SKG. Pictured is the Arduino Uno Microcontroller, breadboard half, SparkFun Electronics relay, two LEDs, a diode, transistor, resistors, and jumper wires. The red wires represent input voltage (5V), black wires represent ground, blue wires represent relay-regulated voltage output, and the yellow wire drives the transistor.

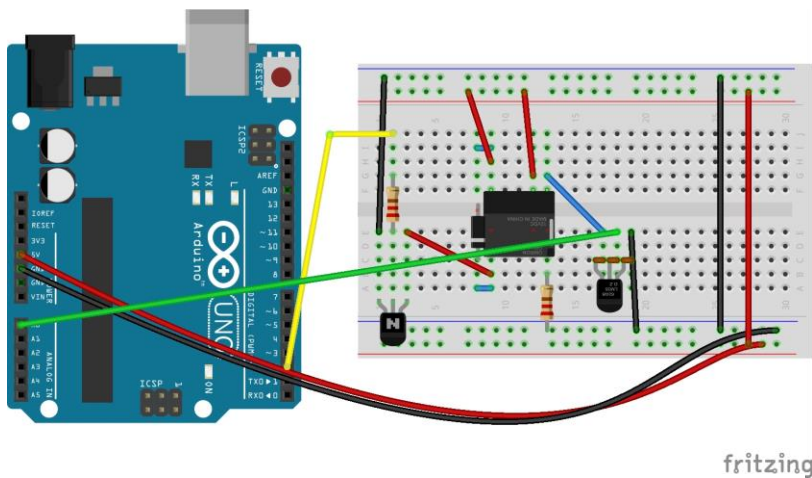


Diagram 3: Integrated Temperature Sensor and Relay circuit constructed using SKG. Pictured is the Arduino Uno Microcontroller, breadboard half, SparkFun Electronics Temperature Sensor, a diode, transistor, resistors, and jumper wires. The red wires represent input voltage (5V), black wires represent ground, the yellow wire drives the transistor, and the green wire represents analog communication with analog pin 0. The relay regulates power output to the temperature sensor through the normally closed port.

### [Integrated Circuit Code]

```
const int relayPin = 2; // use this pin to drive the transistor
const int timeDelay = 6000; // time delay is in milliseconds
const int temperaturePin = 0; // analog pin 0 is used to transmit data stream to com port
unsigned long startTime; //time since Arduino first received power
long relayTime; //time relay state reads HIGH
long elapsedTime; //time elapsed since relay last read HIGH

// timeDelay can be shorter, but note that relays will wear out quickly if you try to drive
// them too fast.

void setup()
{
  pinMode(relayPin, OUTPUT); // set pin as an output
  Serial.begin(9600); // baud rate: 1 transmission/sec
  startTime= millis(); // startTime variable equals the number of milliseconds since the Arduino began
  // running the current program
}

void loop()
{
  relay(); //first run relay
  temp(); //then run temp
}

void relay()
{
  {
    relayTime= millis(); //time relay is on
    digitalWrite(relayPin, HIGH); // turn the relay on

    delay(timeDelay); // wait for six seconds

    digitalWrite(relayPin, LOW); // turn the relay off
  }
}

void temp()
{
  int relayState; // relayState represents the output of the relayPin
  relayState = digitalRead(relayPin); //read relay pin outout, can either be HIGH or LOW
  for (elapsedTime= 5999; elapsedTime <= 12000; relayState == LOW) //if the elapsed time is 5999ms //and less than or equal to 12000ms, the
  //relayState is LOW and the loop should be executed until false
  {
    float voltage, degreesC, degreesF; // floating values can be
    voltage = getVoltage(temperaturePin); //subroutine declared below
    degreesC = (voltage - 0.5) * 100.0; //convert voltage to degree c
    degreesF = degreesC * (9.0/5.0) + 32.0; //convert degree c to degree f
    elapsedTime = millis() - relayTime; //elapsed time is current time minus the last time relay read high
    Serial.print("voltage: ");
    Serial.print(voltage);
    Serial.print(" deg C: ");
    Serial.print(degreesC);
    Serial.print(" deg F: ");
    Serial.print(degreesF);
    Serial.print(" time: ");
    Serial.print(millis()); //time since arduino turned on
    Serial.print (" duration sensor on: ");
    Serial.println (elapsedTime); //how long the temperature probe is active
    delay(1000); //repeat every 1 sec until false
  }
}

float getVoltage(int pin) //subroutine
{
  return (analogRead(pin) * 0.004882814);
}
```

```
}
```

[SD Logging]

```
#include <SD.h>
#include <SPI.h>
const int chipSelect = 8; //CS pin for Shield, regulates I/O from shield
const int relayPin = 2; // use this pin to drive the transistor
const int timeDelay = 6000; // time delay is in milliseconds
const int temperaturePin = 0; // analog pin 0 is used to transmit data stream to com port
unsigned long startTime; //time since Arduino first received power
long relayTime; //time relay state reads HIGH
long elapsedTime; //time elapsed since relay last read HIGH
// timeDelay can be shorter, but note that relays will wear out quickly if you try to drive
// them too fast.

void setup()
{
  pinMode(relayPin, OUTPUT); // set pin as an output
  pinMode(chipSelect, OUTPUT);
  Serial.begin(9600); // baud rate: 1 transmission/sec
  startTime = millis(); // startTime variable equals the number of milliseconds since the Arduino began
  // running the current program
  if (!SD.begin(chipSelect)){ Serial.println ("initialization failed"); //initialize SD card
  return;
  }
  Serial.println("initialization complete");
}

void loop()
{
  relay(); //first run relay
  temp(); //then run temp
}

void relay()
{
  relayTime = millis(); //time relay is on
  digitalWrite(relayPin, HIGH); // turn the relay on
  delay(timeDelay); // wait for six seconds
  digitalWrite(relayPin, LOW); // turn the relay off
}

void temp()
{
  int relayState; // relayState represents the output of the relayPin
  relayState = digitalRead(relayPin); //read relay pin output, can either be HIGH or LOW
  for (elapsedTime = 5999; elapsedTime <= 12000; relayState == LOW) //if the elapsed time is 5999ms //and less than or equal to 12000ms,
  the relayState is LOW and the loop should be executed until false
  {
    elapsedTime = millis() - relayTime; //elapsed time is current time minus the last time relay read high
    File dataFile = SD.open("datalog.txt", FILE_WRITE);
    if (dataFile){
      float voltage, degreesC, degreesF; // floating values can be decimal
      voltage = return (analogRead(temperaturePin) * 0.004882814); //subroutine declared below
      [code]
      pH = (voltage + 0.3932) / 0.058166; // for temperature of 20C, to change the generic equation is pH= Voltage - (-0.3987+(-
0.001101*(tempC-25)))/((8.314*tempK)/96487)*ln10
      dataFile.print("voltage: ");
      dataFile.print(voltage);
      dataFile.print(" pH: ");
      dataFile.print(pH);
      dataFile.print(" time: ");
      dataFile.print(millis()); //time since arduino turned on
      dataFile.print(" duration sensor on: ");
      dataFile.println (elapsedTime); //how long the temperature probe is active
      delay(1000); //repeat every 1 sec until false
    }
  }
}
```