6-5-2015

AquaSift: point-of-use microfluidic detection system

Daniel Beyers
Santa Clara University

Jasper Tan
Santa Clara University

Brandon Young
Santa Clara University

SANTA CLARA UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING

Date: June 05, 2015

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

Daniel Beyers
Jasper Tan
Brandon Young

ENTITLED

AquaSift: Point-of-Use Microfluidic Detection System

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREES OF

BACHELOR OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING
BACHELOR OF SCIENCE IN ELECTRICAL ENGINEERING

Thesis Advisor

Thesis Advisor

Department Chair

6/9/15

Department Chair

AquaSift: Point-of-Use Microfluidic Detection System

by

Daniel Beyers
Jasper Tan
Brandon Young

Submitted in partial fulfillment of the requirements
for the degrees of
Bachelor of Science in Computer Science and Engineering
Bachelor of Science in Electrical Engineering
School of Engineering
Santa Clara University

Santa Clara, California
June 05, 2015

AquaSift: Point-of-Use Microfluidic Detection System

Daniel Beyers
Jasper Tan
Brandon Young

Department of Computer Science and Engineering
Department of Electrical Engineering
Santa Clara University
June 05, 2015

ABSTRACT

AquaSift is a portable, affordable, point-of-use system that performs microfluidic detection of contaminants in drinking water. It comprises four main components: a three-electrode sensor, a potentiostat circuit device, an Android application, and an online database. It utilizes three-electrode voltammetry by applying a voltage stimulus across the electrodes and reading the induced current on the water sample. Testing has shown that our system is able to detect arsenic in solution samples. The Android application serves as the user interface to the system, and the online database allows the mapping of test results on an easy-to-use website.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Statement

Being able to detect contaminants in drinking water provides many benefits. Arsenic is one such substance that serves as a good example. According to The World Health Organization (WHO), an estimated 200 million people are exposed to drinking water sources contaminated with arsenic [1]. Long-term exposure is linked to lung, bladder, skin, kidney, and liver cancers. Arsenic is difficult to detect because it is tasteless, colorless, and odorless, which prevents human senses from identifying it. As a result, many do not know that they are drinking contaminated water.

Currently, solutions exist to detect contaminants, but they lack certain features beneficial for the emerging market. Large, expensive lab equipment (costing thousands of dollars) is needed to perform electrochemical detection using electrodes. This equipment requires extensive training, is not field-portable, and may cost too much for deployment in multiple areas. The colorimetric method offers a less expensive alternative, but it does not provide a quantifiable measurement of arsenic. If arsenic is present in the water, the test paper changes color, but it is difficult to map a certain color to an exact concentration of arsenic. Furthermore, the end result of a test is simply a colored piece of paper; the test does not produce digital data that can be stored for further analysis.

A research group in the bioengineering department of Santa Clara University has been able to provide a new method for detecting contaminants in water using a disposable three-electrode sensor system. Such a device needs both hardware and software support for its operation.

We have thus designed and implemented a system comprising an affordable, portable, and simple circuit board that can perform simple microfluidic electrochemical detections, as well as an Android application that interfaces with the bioengineering sensor. The circuit device is connected to an Android device for both power and control. The Android application is able to perform multiple functions, such as reading and analyzing test data, controlling the circuit board's operations, and storing values on a global database. Our system maintains its low cost and convenience while still providing accurate concentration values in its readings.

1.2 Benefit

Our system makes the detection of microfluidic substances easier. It can also be used for bookkeeping of contaminated sources through the database. While our work does not provide methods of filtering the contaminants out or of cleaning the water source, it provides information as to which sources need treatment. This can potentially save lives, as people will be more able to avoid drinking from contaminated water sources. We intend our product to be used by social enterprises and organizations for convenient management of water sources they are responsible for.

Aside from its social implications, our system also has academic value, as it can be used for electrochemistry classes in school labs instead of costly equipment. Studying concepts such as voltammetry is difficult because schools may have reservations about allowing each student to operate an expensive electrochemical analyzer, or may lack the resources to provide enough analyzers for an optimal classroom experience. An affordable and portable potentiostat makes it much more feasible for schools to have lab sessions where students can perform voltammetry and analyze results.

1.3 Three-Electrode Voltammetry

1.3.1 General Overview

The method AquaSift uses to detect contaminants is three-electrode voltammetry. This procedure from electrochemistry utilizes three metal electrodes, called the working, counter, and reference electrodes, that are submerged in the water sample being analyzed. An electrochemical analyzer is then attached to the three electrodes. The process is two-step: voltage stimulus and current measurement.

The electrochemical analyzer first provides a certain voltage input, dependent on the specific contaminant to be detected, across the working and reference electrodes. This voltage stimulus causes a reaction to occur in the sample if the contaminant is present. The reaction then produces an electrical current; the higher the concentration of the contaminant, the higher the current produced. Next, the electrochemical analyzer measures the amount of current produced in the reaction, and one can then examine the data to determine the concentration of the contaminant in the water sample.

1.3.2 Arsenic Detection

While we designed our circuit with the goal of having it be versatile for the detection of multiple contaminants, all of our testing this year has been with arsenic detection. For arsenic, the voltage stimulus required is a linear voltage sweep with two steps. The first is the deposition stage, where the voltage across the working and reference electrodes is maintained at -0.5V for two minutes. This low voltage attracts the arsenic ions, which appear positively charged in their aqueous form, to the working electrode. The ions are then joined with electrons, resulting in a solid form of arsenic. After the two minutes, a linear voltage sweep from -0.5V to 0.6V is applied over 22 seconds. At arsenic's oxidation potential of around 0.1V, the arsenic gives up its electrons, which produces a current that can be read through the working electrode. This is summarized in the plot seen on Figure 1.1.

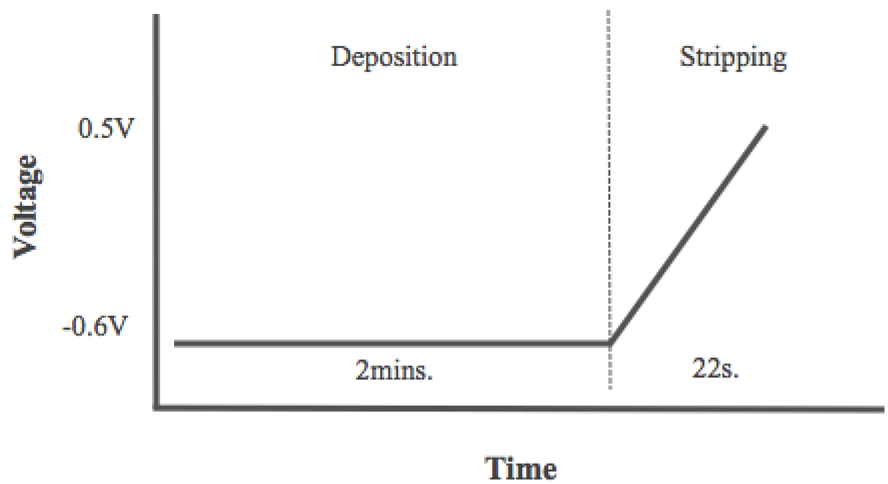


Figure 1.1: Necessary voltage input to detect arsenic

1.3.3 Discrimination

One question that may arise is how the electrochemical analyzer can discriminate among various contaminants. In other words, how can one determine which contaminant is producing the measured current? A certain contaminant would cause a reaction only for a certain voltage stimulus. For example, the voltage input used to detect arsenic may not cause reactions for other substances, which would then not produce any current. Furthermore, the current seen for a certain contaminant occurs at a precise voltage value during the stimulus. For arsenic, this is at 0.1V. Thus, it is safe to say that a current measured at 0.4V is not caused by arsenic contamination.

1.4 Existing Solutions

Water contamination is a problem that has existed for a long time. Thus, there are numerous solutions present today that aim to detect these substances. However, each of them has disadvantages that we aim to address with AquaSift.

1.4.1 Lab-Grade Electrochemical Analyzer

One of the standard methods for accurate detection of contaminants is using a lab-grade electrochemical analyzer, which utilizes voltammetry. It is attached to the set of electrodes that would be placed into the water sample, and it is also connected to a computer. The computer must contain the analyzer's accompanying software tool, which is used to operate the device. The user also uses the software tool to read and analyze the measurements made by the analyzer. Our lab contains the 730D from CH Instruments, as seen in Figure 1.2.

Figure 1.2: The CH Instruments 730D Electrochemical Analyzer

Such a tool provides strong accuracy in detecting contaminants. It is also versatile, as it is able to provide a variety of voltage stimuli for the detection of different types of contaminants. Also, digital data representing test results is readily available to be reviewed and analyzed again. This allows easy management, communication, and storage of information.

However, lab-grade electrochemical analyzers still come with disadvantages, especially for emerging communities. First, these devices typically cost more than a thousand US dollars, which may not be financially feasible for poorer communities. Also, because of its size and software support requirements, the system must be housed in a laboratory instead of being able to be brought into the field for point-of-use testing. In line with this, such a tool requires technical knowledge and training in order to be used properly. For emerging markets and developing nations, such resources are typically not available.

1.4.2 Colorimetric Detection

Colorimetric detection can also be used to detect contaminants in drinking water [2]. In this method, specifically chosen pieces of paper are dipped into a water sample. The presence of the contaminant under investigation changes the color on the piece of paper. Higher concentrations would provide more drastic color changes.

This method addresses the disadvantages of the lab-grade potentiostat. First, it is a portable kit that is easily brought into the field for point-of-use detection. Second, a kit typically costs less than 50 US dollars, which makes it more accessible for developing nations. Third, the method is easily performed by people without technical knowledge, which eliminates the requirement for having technically trained people present to perform tests.

However, colorimetric detection also has some disadvantages. First, it is difficult to determine the specific concentration value of the contaminant present with just a color shade. The method does not return numeric values, but simply a visual cue. Thus, a user would have to restrict his/her analysis to how drastic he/she perceives the color change to be. Also, there would be no digital data that can later be analyzed or shared.

1.4.3 CheapStat

The existing solution that shares many similarities with ours is the CheapStat, a potentiostat device designed by a research group at University of California Santa Barbara [3]. This device is a potentiostat with a size slightly larger than a deck of cards that can perform basic voltammetric sweeps for the detection of contaminants in liquid samples. It consists of a printed circuit board housed in a plastic container to which one can attach three electrodes to perform tests. The device is shown in Figure 1.3.

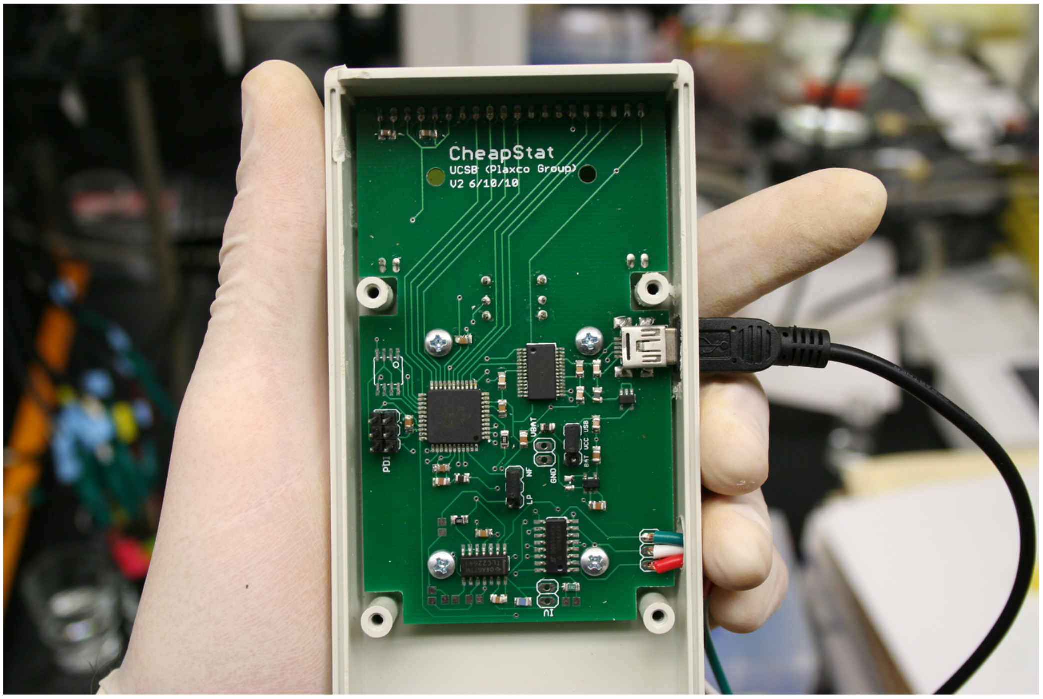


Figure 1.3: The CheapStat device from University of California Santa Barbara [3]

This device has the portability that allows easy point-of-use detection for in-field testing. It is also plugged into a computer, which then yields digital data and precise measurement values for analysis. Furthermore, its price tag of less than 80 US dollars makes it more feasible for developing nations to purchase compared to a lab-grade potentiostat.

This device, however, does not contain some features which we deem important. First, it needs to be plugged into a computer for data analysis. This diminishes the portability value, as even laptop computers carry some extra bulk. Furthermore, computers increase the price for the entire system if they are a requirement for proper operation. Also, the CheapStat does not provide a database feature where results can be stored for easy management of test data. Lastly, some technical knowledge is still required to be able to use the CheapStat properly.

While these existing solutions are able to address some issues with the detection of contaminants in drinking water, there are also still some opportunities for improvement. Our device provides features that none of these existing solutions have.

1.5 Previous Work

In the past few years, other student groups have completed work on this project, and great progress has been made towards a deliverable system. This year, we took their work and improved on it, adding features and implementing designs.

1.5.1 Sensor Development

Bioengineers at Santa Clara University have conducted research on a voltammetric method to detect arsenic using a three-electrode sensor. The end product of such research was a portable sensor prototype that contained three painted electrode strips on which water samples could be placed. The bioengineers tested their sensor by connecting it to a CH Instruments lab-grade potentiostat and its accompanying PC software. Their research has also given us the details regarding the voltage stimulus one must apply to a water sample in order to detect arsenic. We used this information to design our system.

1.5.2 Circuit Development

Past electrical engineers at Santa Clara University have developed a potentiostat circuit to support the bioengineering sensor. A basic and functional potentiostat was implemented on a breadboard. It is able to perform linear sweeps with the Texas Instruments MSP430 as its microcontroller. In our project, we added features to this basic circuit such as sensitivity switching and implemented it on a printed circuit board. We also enhanced the code of the microcontroller to handle sensitivity selection input.

1.5.3 Mobile Application Development

Previous Santa Clara University computer engineers have developed an Android application that integrates with the potentiostat circuit. This mobile app is able to accept test results from the circuit through UART communication. The application also provides a simple algorithm to plot the results. We enhanced the mobile application to include features such as sensitivity switching, geolocation, communication with our online database, a calibration option for visual comparison to clean water samples, and more precise test results displayed to the user.

1.6 Target Specifications

In order for our product to be successful, we determined target specifications that our device must meet.

- The device must be able to tolerate temperatures from 30°F to 120°F to be durable for field testing.
- The target cost for one device is $50 (US). This ensures that developing nations will be able to afford the system.
- To ensure portability, our device must have size dimensions close to 2in x 3in x 1in (about the size of a pack of cards).
- For arsenic, the device must be able to detect as low as 10ppb, as this is the threshold set by the World Health Organization for water safe from arsenic contamination.

1.7 General Overview

AquaSift consists of four main components: a three-electrode sensor, a potentiostat device, an Android application, and an online database. The three-electrode sensor is the product of research done by bioengineering students. Its materials and geometry are carefully chosen for each contaminant. The sensor is where the water sample is placed for analysis. The other three components, which are seen in Figure 1.4 are the products of this research project.

Figure 1.4: AquaSift's potentiostat device, Android application, and online database (L-R)

A potentiostat is a device that utilizes voltammetry by being able to provide a voltage stimulus and to measure a read current. Our potentiostat device contains analog circuitry and a digital microcontroller to accomplish these needs. The Android application serves as the interface for the system. It allows the user to choose the type of voltammetric sweep he/she wants to perform, as well as shows the user the data from the test. The Android application also allows users to take readings from a clean water sample for visual comparison to contaminated samples, and to send test data, water source depth, and geolocation information to an online database. Aside from housing measurements, the online database also presents a clear map-based interface for easy reading of information.

The components are connected with specific interfaces to handle the flow of data and signals. The three-electrode sensor and the potentiostat device are connected electrically using wire connectors such as alligator clips or hooks. A USB cable is used to connect the potentiostat circuit to the Android device. UART is the communication protocol used by the potentiostat's microcontroller, while USB is that used by the Android device. Thus, a USB-UART interface device is used between the two. Hypertext Transfer Protocol (HTTP), the standard protocol for transferring information over the Internet, is used to allow communication between the Android application and the online database.

For a user to run a test, he/she must first place a water sample on the three-electrode sensor. Afterwards, he/she can determine the parameters for the voltammetric sweep to be run and input the depth of the water source. Then, the user starts the sweep with a button on the Android application. After a few minutes, the Android application will show the results of the sweep and upload these results, the water source depth, and the Android device's current GPS location to the online database, finishing the test.

On the system side, the test starts once a signal is sent from the Android application to the potentiostat device. The potentiostat then provides the controlled voltage on the three electrodes of the sensor. This voltage stimulus causes a reaction in the water sample if the contaminant is present, which then causes a current. The current is read by the potentiostat device and converted into a voltage. This voltage is converted into digital data via the microcontroller, which then sends this data back to the Android application. The Android application performs calculations and analyzes the data to produce the final results to be shown to the user.

Chapter 2

Potentiostat Circuit Device

2.1 Overview

The purpose of the potentiostat device is to implement three-electrode voltammetry by being able to provide voltage stimuli and to read induced current. The device is to be connected to the three electrodes that would be placed into the water sample. To accomplish this, it comprises a digital microcontroller and an analog interface. The microcontroller is used to provide voltages, and it also allows communication between the potentiostat device and the Android application. The analog circuitry is used to interface between the sensors and the microcontroller, and it also maintains signal integrity by containing filters to eliminate noise.

Our device also has non-functional requirements. It must be portable enough for convenient point-of-use testing. We aimed to have our device smaller than the size of a regular deck of cards. Also, the device must have a maximum cost of $50. Lastly, the device must be able to be powered by the Android device alone to eliminate the need for external power sources.

2.2 Microcontroller: Texas Instruments MSP430G2553

Our microcontroller choice is the Texas Instruments MSP430G2553. Such a chip contains features that are valuable for our requirements. First, it has a low cost with its through-hole and surface mount versions available on Digikey for $2.80 and $1.01 respectively. It is also small in size, as its through-hole version comes in the standard 20-DIP package. The microcontroller contains a 10-bit analog-to-digital converter, which we require to have digital data available to communicate to the Android application. The MSP430G2553 supports UART communication, which we use for data transfer between the microcontroller and the Android device. It does not have a digital-to-analog converter, and it can only output either its supply or ground voltages. Lastly, the microcontroller runs on low power, requiring a supply voltage of 1.8 to 3.6V, drawing only 230μA while active and

0.5μA on standby. USB connection to a host device is able to provide 3.3V, which is enough to power the microcontroller. These specifications are summarized in Table 2.1.

Table 2.1: Microcontroller Specifications [4]

Microcontroller	Texas Instruments MSP430G2553
Cost (Digikey)	Surface mount: \$1.01 Through-hole: \$2.80
Analog-to-digital converter	10-bits
Digital-to-analog converter	None
Communication protocols	UART, SPI, I2C
Voltage supply	1.8-3.6V
Operation current	Active: 230μA Standby: 0.5μA

2.2.1 State Diagram

Figure 2.1 shows the microcontroller's state diagram, which depicts the standby, sweep, and data transfer states. The details for each state are listed after the state diagram.

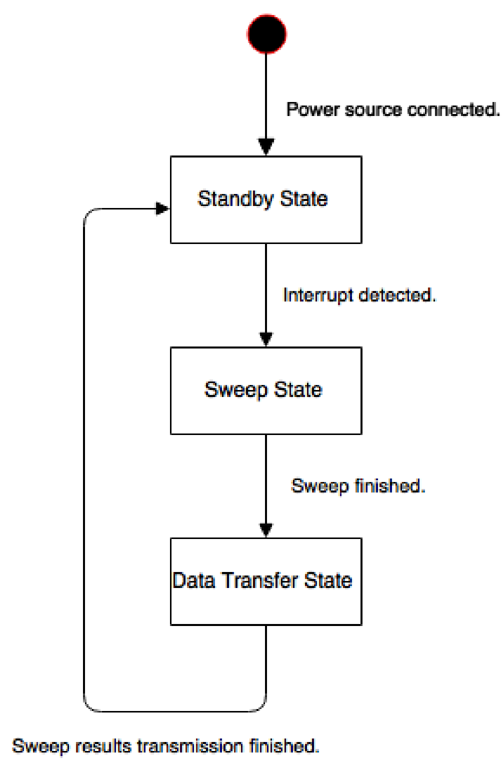


Figure 2.1: Microcontroller State Diagram

- Standby State
  - Initiation: The standby state is entered when the microcontroller is first powered on and whenever data transfer is completed.
  - Function: This state uses low-power mode when waiting for the user to start a given test sweep using the Android application.
  - Interaction: This state interacts with the Android application.
- Sweep State
  - Initiation: The sweep state is entered once an interrupt sent from the Android application has been detected.
  - Function: This state sends the appropriate voltage stimulus to the pathogen sensor for detecting a particular contaminant and reads the results.
  - Interaction: This state interacts with the pathogen sensor.
- Data Transfer State
  - Initiation: The data transfer state is entered after a test sweep has been completed.
  - Function: The microcontroller must send the test results to the Android application for further analysis and processing.
  - Interaction: This state interacts with the Android application.

2.3 Providing Voltage Stimuli

Three-electrode voltammetry requires an input voltage stimulus to start the reaction in the chemical sample. Our device uses both the microcontroller and the analog interface to provide this voltage input. A schematic representing the analog interface can be found in Appendix A.

2.3.1 Pulse-Width Modulation

Since the MSP430G2553 does not contain a digital-to-analog converter, the only voltages that it can provide are its supply voltage (3.3V for our system) and 0V. To be able to provide voltage values in between, pulse-width modulation is used. In this technique, the voltage is kept high for a certain amount of time, and then low for another period of time. The duty cycle is the proportion of the max voltage value that we desire for the output. It also describes how long the voltage should be kept high. For example, to achieve 80% of 3.3V, the voltage should be kept high for 80% of the time, and low for the remaining 20%. This behavior is kept in a cycle for the duration of time we desire the voltage value. This voltage is then fed into a low-pass filter in order to eliminate the high frequencies of switching between high and low voltage values. The final result is the average of the duty cycle, which then allows any value from 0 to 3.3V to be outputted.

2.3.2 Circuit

For our system, we are controlling the voltage between the working and reference electrodes. The working electrode is the input of an op amp, with half of the voltage supply being the other input. Thus, the working electrode is kept at a constant voltage value of 1.65V, and it is the reference electrode's voltage value that is determined by the pulse-width modulation signal. This allows us to provide any voltage value from -1.65V to 1.65V, which is necessary because some contaminants, such as arsenic, require negative voltage values for detection.

A second-order passive RC low-pass filter is used to clean the PWM signal fed in. Afterwards, a unity-gain op amp buffer is used. The output of the op amp serves as the voltage value for the counter and reference electrodes.

2.3.3 Result

Figure 2.2 shows two examples of voltage sweeps by the potentiostat circuit device as measured on the oscilloscope. The linear voltage sweep has a duration of 22 seconds and starts at -0.5V and ends at 0.6V. These are the parameters required for arsenic detection. The square wave voltage sweep was implemented to serve as a proof of concept. While it was not designed with specific parameters in mind, research suggests that square wave voltammetry is necessary for the detection of nitrates. An issue that arises with implementing square wave voltammetry on AquaSift's current design is that the square wave cannot accommodate high frequencies because of the included rise and fall times.

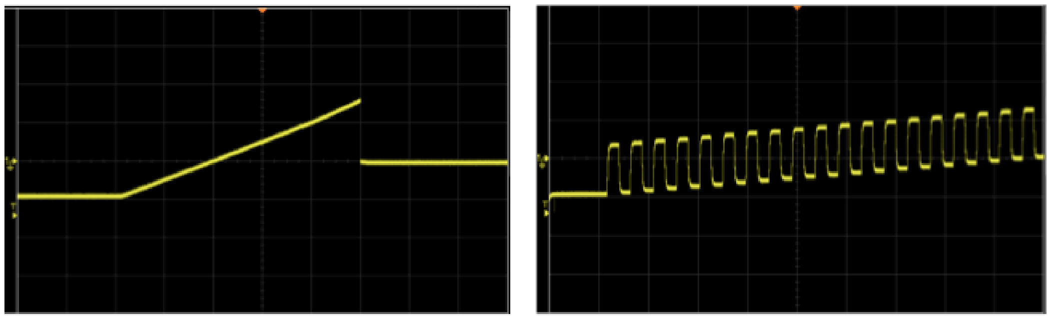


Figure 2.2: Two voltage sweeps: linear voltage sweep for detecting arsenic (L) and square wave voltage sweep (R)

2.4 Measuring Current

2.4.1 Analog Reading

Current is read through the working electrode. The end value is to be read by the microcontroller, which can only read voltage values. Thus, a transimpedance amplifier is used to provide a voltage value that is the product of the current value and the resistor value. It is important to remember that one end of the resistor is kept constantly at 1.65V and not 0V. There are two issues that may arise when mapping the current to voltage. First, the conversion may not be sensitive enough, producing a voltage value that is too small to extract relevant information from. Second, the correct resulting voltage value may be larger than 3.3V, which would saturate the op amp. Such a value would not be able to be read as the output could not exceed 3.3V. Figure 2.3 shows these two issues as seen on an oscilloscope.

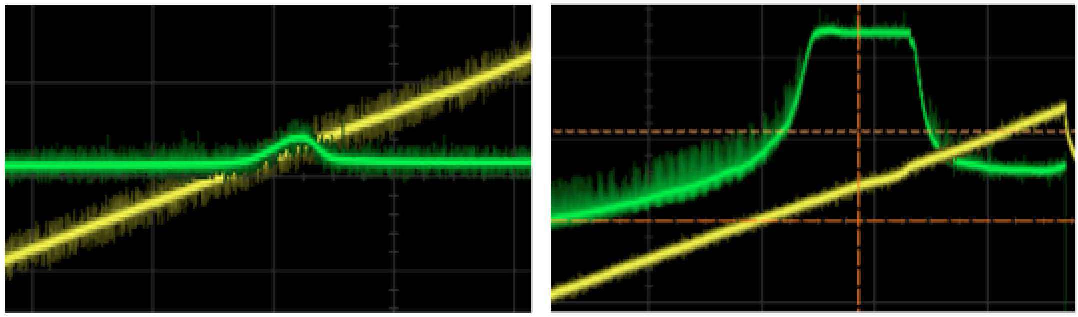


Figure 2.3: Two possible issues: a voltage peak that is too small for analysis (L) and a voltage peak that saturates the op amp (R)

2.4.2 Sensitivity Switching

To address such an issue, our device contains a sensitivity switching feature. Since the resistor value is used as the scaling factor for converting current into voltage, it is the factor that may cause the two issues. Thus, our device contains three different resistor values, each activated by a switch. This allows the user to choose which resistor value to activate for each test. If a low current value is expected, then the larger resistor should be chosen. Included in the transimpedance amplifier is a capacitor in parallel to the resistor to provide filtering. Each resistor can be accompanied by its own capacitor in order to maintain the same RC value regardless of the sensitivity.

The microcontroller selects which of the switches to turn on. It contains three different general purpose input output (GPIO) ports that are each connected to one switch. The initiation of the whole process is triggered by interrupts on the microcontroller, which handles input characters of 'a', 'b', and 'c'. If it receives an 'a', it would turn on the switch with the large resistor value by providing a high voltage to that switch and providing ground voltages to the remaining ones. The characters 'b' and 'c' do the same for the medium resistor value and the low resistor value respectively. Thus, one is able to start the test and to choose the sensitivity by simply sending these characters from the Android application, which, discussed in later sections, masks these operations with pushbuttons on the screen. Figure 2.4 shows the transimpedance amplifier with the sensitivity switching feature.

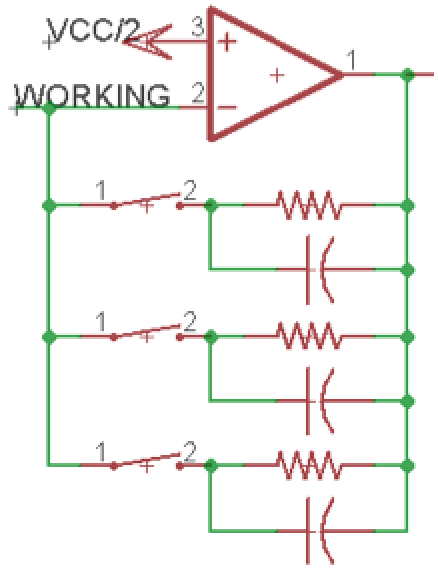


Figure 2.4: The transimpedance amplifier contains three different possible RC configurations to allow different sensitivities

2.4.3 Analog-Digital Conversion

At the end of the whole process, the circuit produces an output voltage from 0 to 3.3V depending on the concentration of the contaminant. The microcontroller then maps this into a 10-bit digital value to be communicated to the Android application. The digital resolution then is 3.3V/1024 = 3.22mV per data point. Our system takes these 10-bit samples 150 times during the duration of the whole sweep, each at different times in order to get a comprehensive image of the output. These 150 values are then sent to the Android application.

2.4.4 Selecting Resistor Values

One can calculate the three resistance values to be able to meet a certain current range. There are 1024 possible data point values for the microcontroller. However, since one end of the transimpedance amplifier is fixed at half the supply voltage, the data point that represents zero current is 512. One could then choose the desired minimum size of the smallest voltage peak and map the lowest expected current value to that. For example, one can choose 150 data points to be the desired smallest peak (for an absolute digital value of 662) and then obtain the voltage value this point represents. Using this as the minimum voltage value and given the minimum expected current value, one can calculate the resistance using Ohm's Law.

$$V_{min} = (DataPoints + 512) * \frac{VDD}{1024} \tag{2.1}$$

$$R = \frac{V_{min}}{I_{min}} \tag{2.2}$$

The maximum current this resistor would be able to read is half the voltage supply divided by the resistance value.

$$I_{max} = \frac{VDD}{2R} \tag{2.3}$$

This process can then be repeated for the other two resistors, ensuring that there is enough overlap between the currents read by two adjacent resistors for each value in the range to be read. Note that these equations make ideal assumptions. They do not take into account the current into the op amps. Thus, for small current values, these equations would not be sufficient.

2.5 Printed Circuit Board Rev. 1

We were able to produce a prototype printed circuit board (PCB) of our system. The prototype does not contain the sensitivity switching feature, only allowing one resistor to be used for the transimpedance amplifier. In addition to the analog circuitry discussed, it also contains other components to support the microcontroller such as power supply filters, LEDs, pushbuttons, etc. The layout and schematic are found in appendices B and C.

The total cost of the board is $125.34, which is much higher than our goal of $50. The breakdown of the cost is seen on Table 2.2. This, however, does not indicate that the goal cannot be reached. First, the prototype board is much larger than necessary to ensure easy testing and low interference. It can be made smaller for the final version, greatly reducing the fabrication price. Furthermore, numerous components were included in the prototype board for the sole purpose of testing the device. These are unnecessary for the productized version. Lastly, a purchased breakout board was used for the UART-USB interface. The cost can be lowered by including the chip into the prototype board itself.

Table 2.2: PCB Rev 1 Costs

Component	Price
Resistors	$1.67
Capacitors	$4.69
Op-amp	$0.98
4-pick socket	$2.16
Microcontroller	$2.80
FTDI USB to Serial Board	$14.95
Testing components	$10.59
Board fabrication	$87.50
Total without test components	$114.75
Total with test components	$125.34

2.6 Noise Issues

Multiple tests with the potentiostat circuit device have revealed that certain factors can cause high amounts of noise in signals. Using high transimpedance resistance values and obtaining simultaneous running of the Android application cause increased levels of noise that may be deemed unacceptable.

The microcontroller only takes 150 samples of the sweep. Noise introduces large spikes in the signal, and if a sample is taken right at the point of where the noise is produced, then that sample would contaminate the result, as it reveals a large voltage value where one is not supposed to be seen. With only 150 samples, there is not enough data to determine which sample contains noise, as a large number of samples would need to be taken to be able to filter out undesired frequencies as discussed by the Shannon-Nyquist Sampling Theorem.

2.6.1 High Resistance Value

High values for the resistor on the transimpedance amplifier introduce additional noise to the signals. Typically, values of 100kΩ and above yield unacceptably noisy signals. For arsenic detection, 2μA is the minimum value of current to be detected, and as such, a maximum resistance value of 50kΩ would be sufficient. However, if future purposes need smaller values of current to be read, then the required large transimpedance resistance may cause noise issues.

2.6.2 Simultaneously Running Android Application

Having the device powered by the Android application does not cause significant levels of noise. Plugging the Android device into the potentiostat circuit device and running a sweep through a hardware interrupt (such as a pushbutton on the circuit) instead of through the Android application does not indicate high levels of noise if the application is not running. However, observing a sweep on the oscilloscope while the Android application is running reveals high levels of noise, as seen in Figure 2.5. Thus, there is some issue that arises when the Android application is running.

We were not able to determine the exact cause of this noise. It is possible that it is because the Android application constantly sends messages to the microcontroller, waiting for the expected data points. If this is the cause, having the application enter standby mode could potentially eliminate the problem. However, there may also be other causes to the noise.

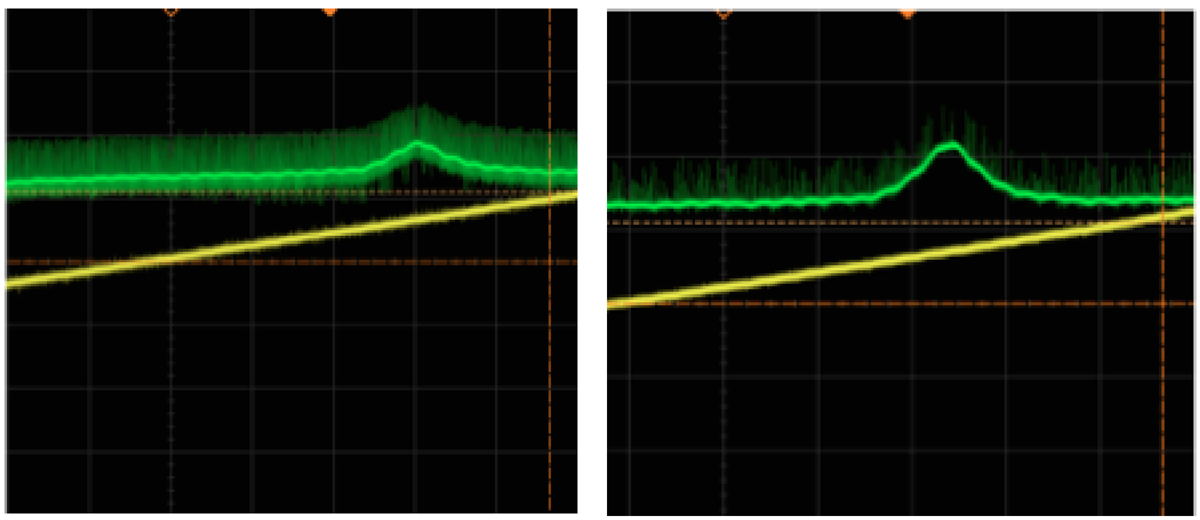


Figure 2.5: Oscilloscope signal of high resistance when the sweep is initialized by the Android application (L) and by a hardware pushbutton interrupt without the application (R) with a 330K transimpedance resistor

2.7 Testing and Results

In order to examine the validity of the potentiostat circuit device, tests have to be performed on it. During the span of our project, we were only able to test our device with arsenic. This is because during the start of our project, there was not enough significant research yet performed by the bioengineering team on how to detect other types of contaminants. The main attributes being tested are the device's accuracy, repeatability, and sensitivity.

2.7.1 Accuracy

The potentiostat circuit device is accurate if it gives the expected results: increased resulting voltage peaks for increased concentrations of contaminants. To test this, we run the procedure on various concentrations. We use an oscilloscope to observe the input and output voltages of our potentiostat.

The oscilloscope can also provide csv values, which we can analyze and plot on Excel.

The control sample we use is 0.1M Hydrochloric Acid, which conducts electrical current much better than water does. The final bioengineering sensor would have the electrodes pre-treated with this acid to allow water samples to be analyzed. Our device should not give any significant voltage peak with this control sample. We then perform tests on 0.1M Hydrochloric Acid with different concentrations of arsenic. As the concentration increases, the voltage peak is expected to increase as well. Figure 2.6 shows the values of the voltage peaks we see for the different concentrations using a 110k resistor for the transimpedance amplifier. As shown, the voltage peak does increase with concentration, verifying the correctness of our device.

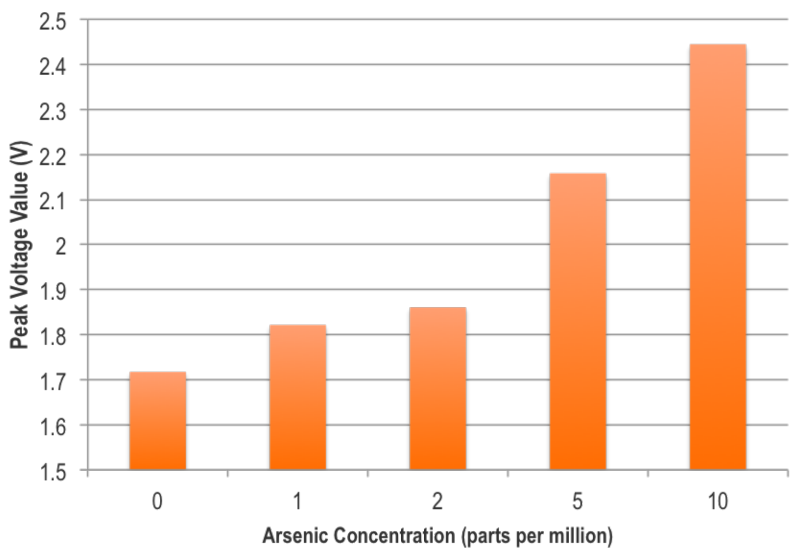


Figure 2.6: Peak Voltage Value vs. Arsenic Concentration

To further analyze accuracy, we compare our results to those obtained with a lab-grade potentiostat, the CH Instruments 730D Electrochemical Analyzer. To test this, we run an analysis on a certain solution using the 730D. Afterwards, we run the operation with our potentiostat circuit device on the same solution. Since the 730D displays current while our device displays voltage, the values will be on different scales. Because voltage and current are linearly related, we can apply a scalar multiple to one in order to compare the results. Figure 2.7 shows that the general trend of our results is similar to the results of the 730D.

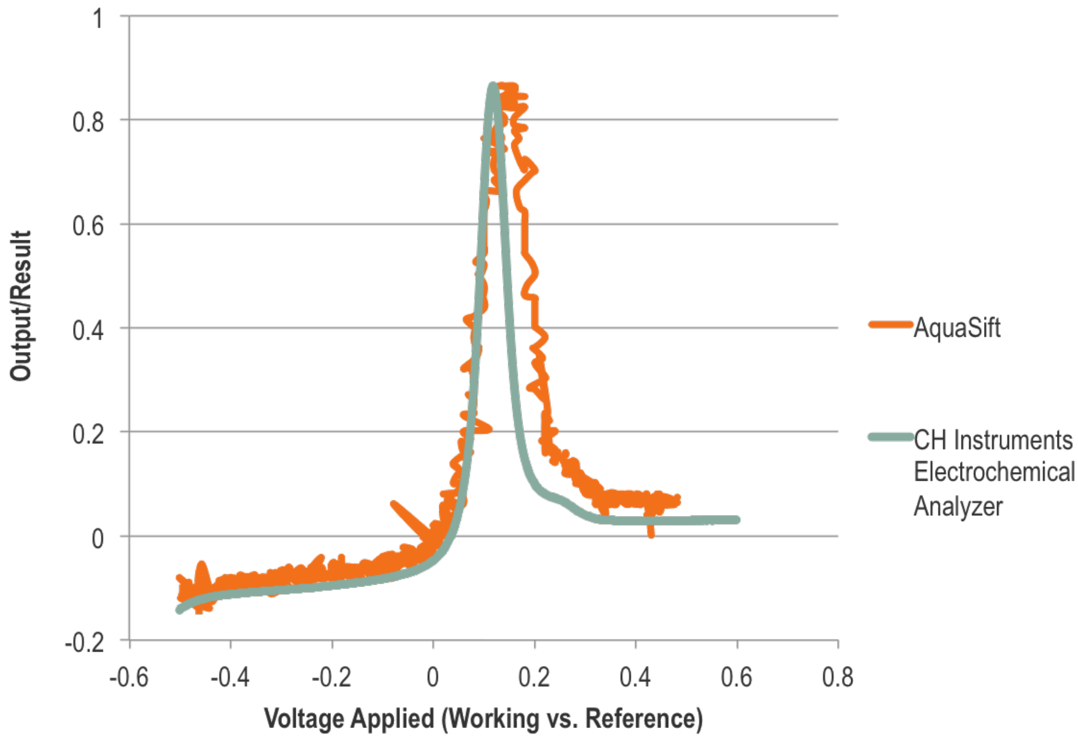


Figure 2.7: Result taken from AquaSift and from CH Instruments Electrochemical Analyzer

It is important to note that the equations provided in Section 2.4.4 did not match our results. That is, the current value seen on the CH Instruments Electrochemical Analyzer, the voltage values seen on the oscilloscope with AquaSift, and the resistor value used do not form the equality described by Ohm's Law. There are a number of possible reasons for such discrepancies. First, with currents on the range of microamps, the current that enters the op amps, which we did not take into account, may have contributed. Second, perhaps the current that the CH Instruments Electrochemical Analyzer reads is not similar to the current that enters the AquaSift. There are other possibilities, but we were not able to fully identify the source of the problem.

2.7.2 Repeatability

To test repeatability, we run multiple tests on one solution. We wanted to see if we could obtain the same results for every test we perform. Table 2.3 shows the peak voltage values on three tests run on one 5μg/L Arsenic solution using a 110k resistor on the transimpedance amplifier. The percent error is smaller than 2%, which is acceptable for our purposes.

Table 2.3: Peak Voltage Values on a 5μg/L Arsenic Solution

Description	Peak Voltage Value
Run 1	2.141V
Run 2	2.158V
Run 3	2.123V
Average Value	2.141V
Range of Differences	0.035V
% Error	1.635%

2.7.3 Sensitivity

We also tested the validity of our sensitivity switching feature. Figure 2.8 shows the oscilloscope results of measuring the output voltage of a 50mg/L sample of arsenic using three different resistor values: 110k, 24k, and 10k. The image shows that even for the same solution, different resistor values can give different sized voltage peaks. Thus, sensitivity switching is proved as effective as it maps the current from one sample into various voltage values depending on the resistance. The values of the results are seen on Table 2.4.

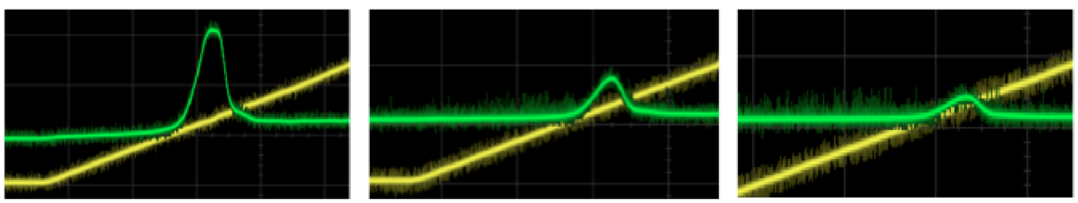


Figure 2.8: Voltage peaks for a 50mg/L arsenic sample using a 110k resistor (L), a 24k resistor (M), and a 10k resistor (R)

Table 2.4: Voltage readings for sensitivity switching on a 50mg/L arsenic solution

Resistor Value	**Peak Voltage**
110k	2.57V
24k	1.94V
10k	1.76V

Chapter 3

Mobile Application

3.1 Overview

The purpose of the Android application is to serve as the user interface for the system. It provides an easy-to-use means for conducting tests on water samples. Its features include the ability to start sweeps of varying sensitivities through resistor selection on the potentiostat device, a calibration function for visual comparison of contaminated water samples to a non-contaminated sample, graphing of test results, displaying contaminant concentration, and the uploading of test results to our online database while including GPS coordinates and the user-inputted depth of a water source.

3.2 Requirements

This section lists the requirements for the Android application. Such requirements are valuable, as they allow easy identification of whether the device has been successfully implemented or not. Included with each requirement is its importance, a value from 1-10 where 10 denotes the most crucial requirements. Our system's requirements are divided into two categories: functional requirements and non-functional requirements.

3.2.1 Functional Requirements

Functional requirements are those that our system must be able to do. They can be evaluated only as either true or false. Our system's functional requirements are listed in Table 3.1.

Table 3.1: Mobile Application Functional Requirements

Name of Requirement	Importance (1-10)
Read data from the sensor via the microcontroller	10
Send a message to the microcontroller to start taking a reading	9
Configure the sensitivity of the sweep the microcontroller uses to take a sensor reading	8
Display a graph of the results read from the microcontroller	8
Display a safe/unsafe result based on the data collected from the microcontroller	8
Display the amount of contaminants present in the water sample	8
Log the results to a map and a database	8

3.2.2 Non-Functional Requirements

Non-functional requirements are those that describe the manner of achievement of the functional requirements. They are evaluated as a degree of satisfaction. The non-functional requirements of the product are shown in Table 3.2.

Table 3.2: Mobile Application Non-Functional Requirements

Name of Requirement	Importance (1-10)
Be easy to use for new users	10
Be quick in displaying its results	8

3.3 State Diagram

Figure 3.1 depicts the various states of the Android application and the actions that lead to each one. The diagram includes a disconnected state, connected state, testing state, and results state. The details for each state are listed after the state diagram.

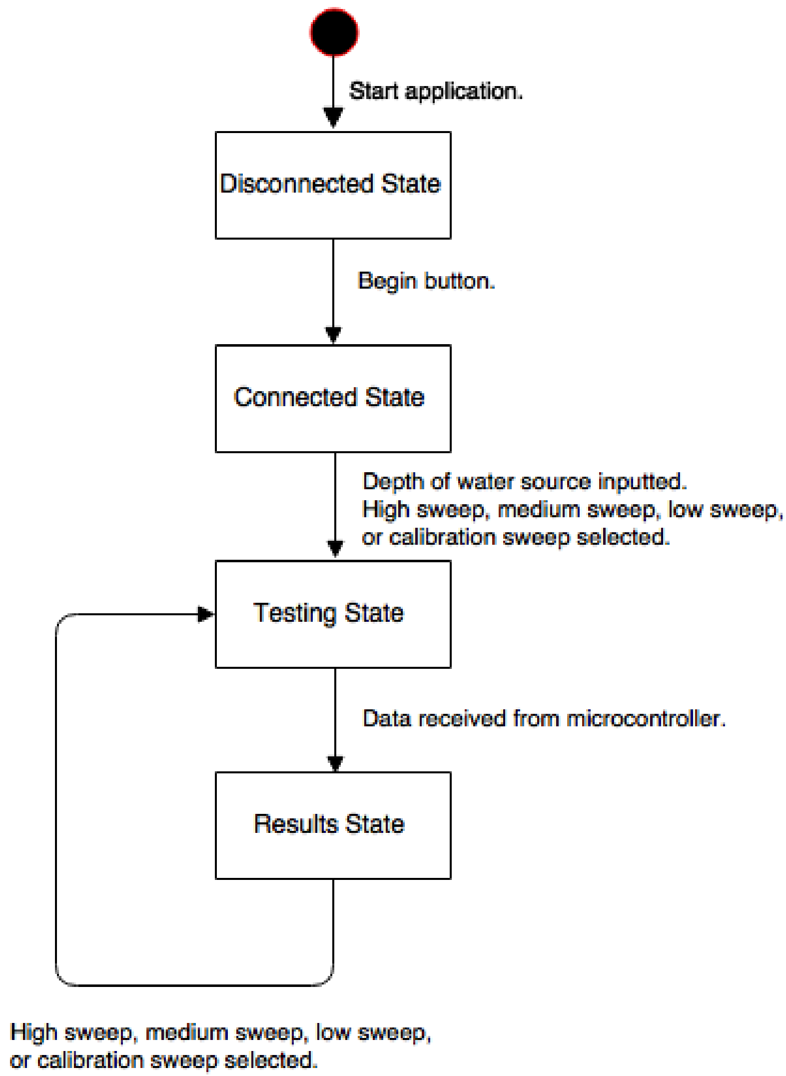


Figure 3.1: Mobile Application State Diagram

- Disconnected State
  - Initiation: The application enters the disconnected state when it is first started.
  - Function: The application must be initialized before anything else can happen. When the user is ready, he/she can connect to the microcontroller using the Begin button.
  - Interaction: This state interacts with the MSP430 microcontroller.
- Connected State
  - Initiation: The connected state is entered once the application has successfully connected to the MSP430 microcontroller.
  - Function: This state enables the buttons for high, medium, and low sensitivity sweeps, as well as the calibration sweep. Once the user has placed the three electrodes in the water sample, he/she can begin a test sweep of his/her choosing.
  - Interaction: This state interacts with the MSP430 microcontroller.
- Testing State
  - Initiation: The testing state is entered after a test sweep has been started.
  - Function: The application must wait for the microcontroller to perform the test sweep and then read the resulting data.
  - Interaction: This state interacts with the MSP430 microcontroller.
- Results State
  - Initiation: This state is entered once the application has finished reading the data from the microcontroller.
  - Function: When this state is entered, the application graphs the test results, displays peak voltage and digital values, and uploads the test results to the online database, as well as GPS location and water source depth. The user can then start another sweep of his/her choosing.
  - Interaction: This state interacts with the online database and the MSP430 microcontroller.

3.4 Use Cases

Our device allows users to perform certain actions such as starting a sweep and selecting sweep sensitivity. Any action that a user might take is considered a use case, which is defined as the steps needed to achieve a goal on our system. A use case contains information on the actor, goal, preconditions, postconditions, sequence of steps, and exceptions associated with the action. It is important to identify these use cases to ensure our device's requirements can be met through a certain sequence of steps. The use cases also shed light on appropriate ways to test the device. Our system must be able to support all these use cases to be successful.

3.4.1 Use Case 1: Connect to Microcontroller

- Actor
  - User.
- Goal
  - To connect the system to the microcontroller so that readings can be taken.
- Preconditions
  - The system is plugged in to the microcontroller via microUSB.
- Postconditions
  - The system is connected to the microcontroller.
- Sequence of Steps
  1. User presses Begin button.
  2. System prompts user to allow access to the microUSB port.
  3. User confirms access to the microUSB port.
  4. User presses Begin button a second time.
  5. The system indicates if it was able to connect to the microcontroller successfully or not.
- Exceptions
  - None

3.4.2 Use Case 2: Take Calibration Reading

- Actor
  - User.
- Goal
  - To collect baseline data from an uncontaminated water sample via the pathogen sensor and display the results to the user.
- Preconditions
  - The system is connected to the microcontroller.
  - The microcontroller is connected to the pathogen sensor.
  - The test sample has been placed on the pathogen sensor.
- Postconditions
  - The system displays the results from the calibration reading in a graph.
  - The system indicates the amount of contaminants present in the water sample.
  - The system indicates that the uncontaminated sample is safe to drink.
  - The system has stored the calibration data for later comparison to contaminated samples.
- Sequence of Steps
  1. User enters depth of the water source in meters.
  2. User presses Calibrate button.
  3. System starts a sweep using the microcontroller and pathogen sensor.
  4. System reads data from the microcontroller.
  5. System stores the data for later comparison with sensor readings.
  6. System displays a graph of the data.
  7. System displays the amount of contaminants present in the water sample.
  8. System indicates whether the water sample is safe or unsafe to drink.
  9. System uploads the results, geolocation information, and water source depth to the Lab-on-a-Chip online database using a cell or wi-fi connection, if available.

- Exceptions
  - None.

3.4.3 Use Case 3: Take Sensor Reading

- Actor
  - User.
- Goal
  - To collect data from the pathogen sensor and display the results to the user.
- Preconditions
  - The system is connected to the microcontroller.
  - The microcontroller is connected to the pathogen sensor.
  - The test sample has been placed on the pathogen sensor.
- Postconditions
  - The system displays the results from the sensor reading in a graph.
  - The system indicates the amount of contaminants present in the water sample.
  - The system indicates if the sample is safe or unsafe to drink.
- Sequence of Steps
  1. User enters depth of the water source in meters.
  2. User presses High, Medium, or Low button, depending on the desired test sensitivity.
  3. System starts a sweep at the appropriate sensitivity level using the microcontroller and pathogen sensor.
  4. System reads data from the microcontroller.
  5. System displays a graph of the data, as well as the baseline data from any previous calibration reading.
  6. System displays the amount of contaminants present in the water sample.
  7. System indicates whether the water sample is safe or unsafe to drink.

8. System uploads the results, geolocation information, and water source depth to the Lab-on-a-Chip online database using a cell or wi-fi connection, if available.

- Exceptions
  - None.

3.5 Data Analysis

When checking the accuracy of our application, we compared our peak voltage values for a variety of samples of known arsenic concentrations to the peak voltage values we read from our oscilloscope in the lab. The results of these tests are contained in Table 3.3 below. We found that the percent error between our application and our oscilloscope was less than 2% in all cases.

Table 3.3: Oscilloscope to Application Comparison

Arsenic Concentration	Oscilloscope Voltage	Application Voltage	Percent Error
50 mg/L	1.76 V	1.75 V	0.6%
20 mg/L	1.92 V	1.91 V	0.5%
10 mg/L	2.44 V	2.46 V	0.8%
2 mg/L	1.86 V	1.89 V	1.6%

3.6 Geolocation

One of the necessary features for our application is the ability to acquire the location of the site where the user is performing contamination testing. Android provides a simple interface for acquiring geolocation coordinates called the location manager. As long as the user is within range of a wi-fi or cell network, the Android device hosting the application can connect and use the location manager to produce the appropriate GPS coordinates, which can later be logged as needed.

3.7 Test Plan

The test plan that follows contains the procedures for verifying that each test case is working correctly. The sequences of steps for verifying that the application can connect to the microcontroller, take a calibration reading, and take a sensor reading are in Tables 3.4, 3.5, and 3.6 respectively. Our system is simple enough that testing can be performed manually in a short amount of time.

Table 3.4: Connecting to Microcontroller

Preconditions: The system is plugged in to the microcontroller via microUSB.	
Steps	Expected Results
1. User presses Begin button.	1a. System prompts the user to allow access to the microUSB port.
2. User confirms access to the microUSB port.	2a. Prompt window closes. 2b. System is ready to connect to the microcontroller.
3. User presses Begin button a second time.	3. System indicates if it was able to connect to the microcontroller successfully or not.

Table 3.5: Taking a Calibration Reading

Preconditions: The system is connected to the microcontroller. The microcontroller is connected to the pathogen sensor. The test sample has been placed on the pathogen sensor.	
Steps	Expected Results
1. User enters depth of the water source in meters.	1a. System displays the entered depth in the input field.
2. User presses Calibrate button.	2a. System starts a sweep using the microcontroller and pathogen sensor. 2b. System reads data from the microcontroller. 2c. System stores the data for later comparison with sensor readings. 2d. System displays the results in a graph. 2e. System indicates whether the water sample is safe or unsafe to drink. 2f. System uploads the results, geolocation information, and water source depth to the Lab-on-a-Chip online database using a cell or wi-fi connection, if available.

Table 3.6: Taking a Sensor Reading

Preconditions: The system is connected to the microcontroller. The microcontroller is connected to the pathogen sensor. The test sample has been placed on the pathogen sensor.	
Steps	Expected Results
1. User enters depth of the water source in meters.	1a. System displays the entered depth in the input field.
2. User presses High, Medium, or Low button, depending on the desired test sensitivity.	2a. System starts a sweep at the appropriate sensitivity level using the microcontroller and pathogen sensor. 2b. System reads data from the microcontroller. 2c. System displays graph of the data, as well as the baseline data from any previous calibration reading. 2d. System displays the amount of contaminants present in the water sample. 2e. System indicates whether the water sample is safe or unsafe to drink. 2f. System uploads the results, geolocation information, and water source depth to the Lab-on-a-Chip online database using a cell or wi-fi connection, if available.

Chapter 4

Web

4.1 Overview

The purpose of the web application is to collect all of the data from the Android device and display that data in an appealing medium. The Android device communicates with a database which in turn relays this information to a website for public access. The main goal of the web end is to store the test information in a central location accessible to the general user.

4.2 Requirements

This section lists the requirements for the web application. Such requirements are valuable as they allow easy identification of whether the system has been successfully implemented or not. Included with each requirement is its importance, a value from 1-10 where 10 denotes the most crucial requirements. Our system's requirements are divided into two categories: functional requirements and non-functional requirements.

4.2.1 Functional Requirements

Functional requirements are those that our system must be able to do. They can be evaluated only as either true or false. Our system's functional requirements are listed in Table 4.1.

Table 4.1: Web Functional Requirements

Name of Requirement	Importance (1-10)
Collect and store data from the Android device	10
Display data on a website	10
Allow for data filtration for easier viewing	5

4.2.2 Non-Functional Requirements

Non-functional requirements are those that describe the manner of achievement of the functional requirements. They are evaluated as a degree of satisfaction. The non-functional requirements of the product are shown in Table 4.2.

Table 4.2: Web Non-Functional Requirements

Name of Requirement	Importance (1-10)
Be intuitive for users to access data	10
Be aesthetically pleasing for users	6

4.3 Solution

The web aspect of the project utilizes a PHP framework for the backend with a MySQL database to store data. The Android application communicates with the MySQL database via PHP and HTTP and the data is stored accordingly. The front end consists of a Bootstrap framework for aesthetics and simplicity with the Google Maps API to take the data and display it in an accessible medium. Filters created in Javascript allow for only specific data to be accessed at the user's discretion.

Chapter 5

Overall System

5.1 General Overview

Each major component of AquaSift has been discussed in detail. This section describes how the components are all connected together to form the entire system. AquaSift comprises four main components: the three-electrode sensor, the potentiostat circuit device, the Android application, and the online database. These all need to be connected to each other in order to have information and quantities be transferred among them.

5.2 Communication

The various parts of our system have different types of data to communicate to each other. Also, they all perform this communication in specific circumstances. Thus, each requires certain communication protocols.

The communication between the three-electrode sensor and the potentiostat circuit device is purely electrical. That is, the only quantities being transferred between the two are physical voltages and currents. Thus, the only connection required is physical.

The part of the potentiostat circuit device that communicates with the Android application is the microcontroller. Digital data is transferred between the two. Specifically, the microcontroller sends 10-bit values to the Android device, while the Android device communicates 8-bit ASCII values, which contain the signals to start sweeps.

For the microcontroller side, we chose to use the Universal Asynchronous Receiver/Transmitter (UART) protocol. This protocol allows both the receiving and transmitting of data, and it does not require synchronized clocks. Thus, the microcontroller is able to communicate with various Android devices. However, the Android device itself does not support UART protocol. Instead, it supports Universal Serial Bus (USB). An interface is required for both to be able to communicate with each other using their respective communication protocols. For our project, we used the FT232R USB UART IC by FTDI Chip. SparkFun provides a breakout board that we used for our system.

The Android application must also be able to communicate with the online database. For this, we use Hypertext Transfer Protocol, or HTTP, which is the standard protocol for communicating information over the Internet. When a test sweep has been completed, the Android application can send the results and additional information to the database using an HTTP POST request, and the database will send a message back confirming that the data has been uploaded successfully.

5.3 Physical Connections

The sensor and the circuit require electrical connection via physical contact. Any conducting wire between the two can suffice. If bulk electrodes are being used, for example, alligator clips or hook wires are suitable because of their shape. Ideally, the potentiostat board would contain an adaptor that the sensor could be plugged into. As long as the adaptor can provide a conductor that has robust contact with the electrodes, then the three-electrode sensor will be electrically connected to the potentiostat circuit device. The adaptor would be dependent on the geometry of the sensor.

The potentiostat circuit device and the Android application require a wired communication method. The most suitable output port for the Android device is the USB port. USB is a host-slave connection. In this case, the Android device serves as the host. It is able to provide around 5V and ground, which allows the Android device to power the potentiostat circuit device. USB is a four-pin communication method where two pins are used for the supply voltage and ground and the other two are used for data communication.

Mobile devices are typically slaves in the USB communication system. Thus, it is important to convert them into hosts for our purpose. To accomplish this, an on-the-go cable, which converts USB slaves to hosts, is needed. The microcontroller communicates with the UART protocol. Thus, a USB-to-Serial interface is also required. We use the FTDI 232r chip to fill this purpose. SparkFun Electronics provides a breakout board for easy usage. Its input is the USB cable and its output contains 4 pins: 3.3V, GND, receive data, and transmit data. With these, the connection between the potentiostat circuit device and the Android application is complete. The connections can be seen on Figure 5.1.

The Android application communicates data with the online database through the Internet. With mobile Internet connection, no additional physical component is necessary to facilitate the transfer of data between the two. Existing Internet solutions serve the purpose.

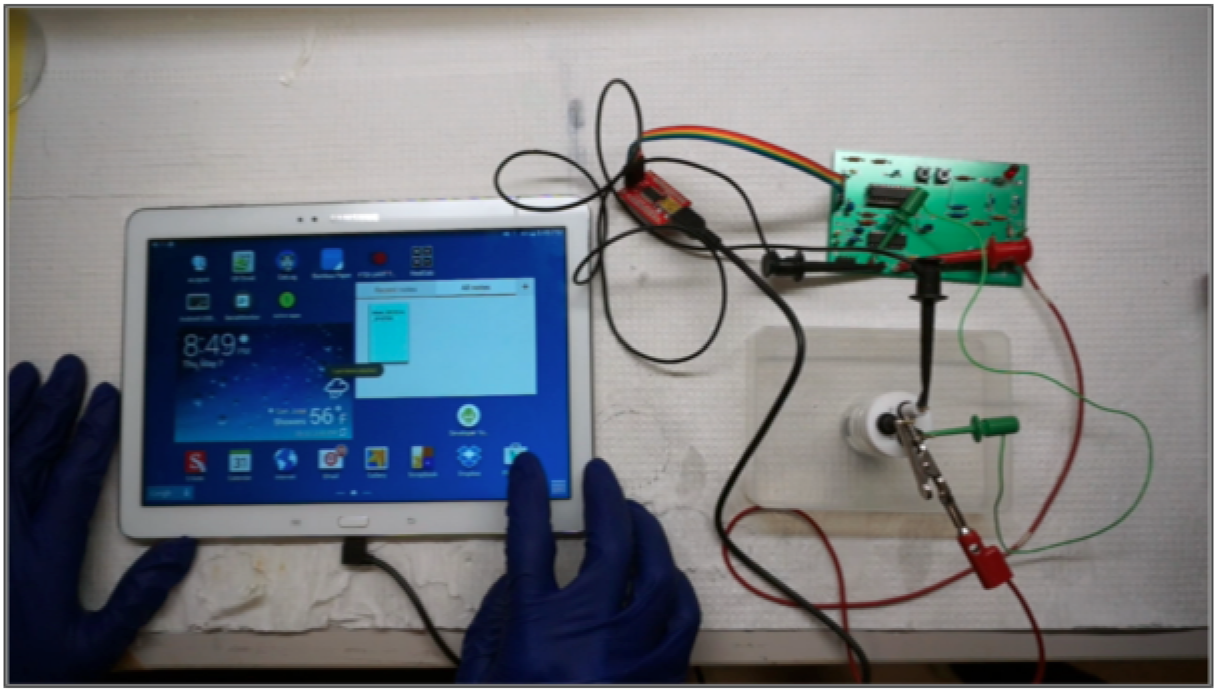


Figure 5.1: The Android tablet uses a USB cable to connect to the FTDI 232r, which is then connected to the potentiostat circuit device using wires. The three electrodes are connected to the potentiostat circuit device using wires.

Chapter 6

Ethical Analysis

6.1 Introduction

For our project, we created a device that can measure the presence of dangerous microfluidic substances, such as arsenic, in drinking water. It will also allow users to upload the data from their measurements into a global database. This device is to be productized and sold to NGOs, who can use it to analyze the safety of drinking water in areas that may not have enough resources to use expensive and complex lab-grade analysis equipment. Since our device has many real consequences, we investigated the ethical ramifications and implications of our product. Furthermore, since the project contains scientific research, we analyzed how we could be ethical in the different steps of the research process.

This section contains our analysis of the ethics as divided into three components. In the first section, we take a general view and analyze the ethics of our project using the various approaches provided by the Markkula Center for Applied Ethics. In the second section, we analyze the product development ethics. This describes how we kept good ethics throughout the designing and implementation of our device. Finally, we analyze the implications of our project for issues of sustainability.

6.2 Markkula Center for Applied Ethics

The center provides various approaches for analyzing the ethical aspects of our device. In this section, we analyze our project using the utilitarian approach, the rights approach, and the common good approach.

6.2.1 The Utilitarian Approach

In this approach, we aim to measure the benefits of our device and investigate if our system does more good than harm. If our device is successful, it provides the benefit of letting people know if their water is safe or not. People may then be saved from sickness if they avoid water that our device measures to be unsafe. Thus, our device provides the benefit of improved health and saved lives.

Furthermore, our device has benefits for the scientific community. There is little information regarding an independent affordable potentiostat device. Thus, we provide advances in the field of electrochemical detection through our work. Our device is also beneficial for people interested in managing the water safety of their respective communities. The device allows them to take readings and place them into the database, giving them an easy-to-use and organized system to store information regarding the state of their water.

The device in itself carries very little harm. The materials used to create it do not bring any significant detrimental effects on the environment. Also, the presence of the device does not pose a danger to people. It can, however, be harmful if used in the wrong way. If people use it to make false claims, such as claiming that a dangerous source of water is actually safe, then it is harmful by allowing deception. Also, if our device is not accurate, it can be harmful to people's health by giving false negatives. An inaccurate device may tell a community that its drinking water is safe when it may actually have arsenic, causing people to comfortably drink the water, which then results in sickness and death. If our device is to be sold to other organizations, we must ensure that it is accurate and reliable.

6.2.2 The Rights Approach

This approach analyzes the way our device promotes human rights. The most basic right that our project aims to protect is the right to human life. Safe drinking water is a human necessity, and everyone should have access to such a resource. However, some believe they have access when in reality they are drinking water that makes them sick. Thus, awareness is the first step in having access to clean water. By letting its users know about the state of their drinking water, our device helps communities know when they need to take steps to clean their water source or to search for a new one. This draws them closer to maintaining a physically healthy life, upholding their right to live.

6.2.3 The Common Good Approach

This approach analyzes whether our device promotes the welfare of the general public. Our device is fair, as it can be used for the benefit of people of any race, gender, or age. In fact, we are building our device specifically to be portable and affordable to help emerging communities. It is unfair that wealthier communities can know if their water is safe or not, while poorer ones do not have such capability. Thus, our device works to level the two, allowing both to have this privilege.

6.3 Product Development Ethics

As we developed our product's design, it was important to consider the integrity of data. There are many designs, schematics, and code present that are open-source. However, some contain copyrights. Before using other people's designs, we first ensured that we had the right permissions. We also attributed credit to others as necessary. There is also the issue of whether we should copyright our design or not. While it is perfectly legal for us to claim ownership of our own work, it could be more ethical to have our work be open-source and to allow others to modify, distribute, and use our work to open doors for more social benefits. Also, our designs and code could be used to advance scientific research in the field of low-cost microfluidic sensing devices. At this point, our code is not yet at the level of a product ready for distribution, and so we have not yet made the final decision on whether it should be open-source or not.

To ensure the quality of our product, we ran multiple comprehensive test cases. We did not ignore any test cases that gave us unexpected results. We had a zero-tolerance policy when it came to bugs in our device. This ensured that our product does not produce false negatives. We also did not cut corners in providing features that allow for a more accurate device, such as making our device larger to avoid noise between components caused by small distances. Such features may have increased costs, but they were the more ethical ones to implement. We also ensured that we recorded all anomalies in our tests and tried to resolve them. If we were unable to eliminate errors in our device, we reported them without concealing details.

It was critical to ensure safety as we worked on our project, which required handling dangerous chemicals in the lab. We kept our work environment safe by neatly organizing our chemicals in secure places. Also, we were responsible in cleaning chemical spills. Other students joined our team in the middle of the year, and we ensured that they went through the necessary safety training programs. We accurately informed them of all our errors, so they could be made aware of what remaining issues need to be solved before releasing the device. This is especially important as we now turn over our project to a new group of students.

6.4 Sustainability

It is important to consider the implications our project and device have for the environment. We made design decisions while considering sustainability, so that our product would not contribute to long-term detrimental effects on natural resources or the global ecosystem.

As we designed our product, we aimed to require as little material as possible. Extra circuitry or electronics, for example, would contribute to electronic waste. Thus, we minimized the number of new physical components for our product, with our entire system only requiring the printed circuit board of the potentiostat circuit device. We chose to use an Android application as the software interface since Android devices are very common, and this would not introduce an additional physical component beyond what people already own. Our database is hosted online, and does not require more physical components for our system.

We also considered the sustainability of our system's power source. We power the microcontroller through USB connection to the Android device. This eliminates the need for added circuitry or external batteries to power the system. Lastly, the potentiostat circuit device is reusable, minimizing disposal of circuit boards.

Chapter 7

Future Work

While the system is functional, other features can still be added to it. Furthermore, it is not at a state where it can be distributed as a product. Thus, each aspect of the system can still be improved by future work.

7.1 Potentiostat Circuit Device

7.1.1 Microcontroller

Other Voltage Sweeps

Currently, the microcontroller is only programmed to provide the necessary voltage sweep for arsenic detection. Other contaminants such as nitrates and fluoride would require other types of voltage stimuli. The research on what type of voltage sweep is required is in the domain of the bioengineering team. Once they complete their investigation, the microcontroller can be reprogrammed to allow other voltage sweeps for the detection of other contaminants.

A PWM signal may not be sufficient in providing these other voltage values. Thus, an external digital-to-analog converter may be necessary. We refrained from using one this year because it provided unnecessary cost and complexity for the detection of arsenic.

Parameter Passing

Currently, the parameters of the voltage sweep, such as its starting voltage, duration, and other such settings, are fixed. The user can only decide when to start a sweep and which sensitivity to use. A helpful feature would be to allow the user to select the values for these parameters and input them through the Android application. These values would then be passed to the microcontroller through UART. The microcontroller code could then be adjusted to use these values to determine the parameters of its operations.

The difficulty is that UART has delays when communicating values. It can only pass one ASCII character at a time, and if a user tries to submit more than one, the others may be lost. Thus, even passing in a 2-digit value would pose problems, as the second character may not be transmitted in time. A possible workaround would be to provide delays in transmitting each ASCII value. Thus, the Android application could ensure each ASCII value is sent before attempting to pass the next one.

7.1.2 Analog Interface

Programming the microcontroller for other voltage sweeps may require changes on the analog interface. For example, the current low-pass RC filter following the PWM signal may not be able to provide the necessary filtering. If an external DAC is to be used, then the analog interface would have to be set up for such adjustments.

Noise is an issue that has been observed on our results, as discussed in Section 2.6. Such noise is caused by factors such as having the microcontroller powered by an Android device and by large resistance values for the transimpedance amplifier. Possible solutions to this noise issue may include redesigning the filters on the analog interface or examining other methods of supplying power to the microcontroller.

7.2 Android Application

While we were able to successfully add many features to the Android application this year, there is still more work to be done in the future. First, a UI redesign could be helpful in making the application more aesthetically appealing and reducing the amount of space some of the current buttons take up. Next, test sensitivity selection is currently done manually by the user, but adding some sort of automatic retest feature for peaks that are either too low or that saturate the system could be desirable, if the need becomes apparent. The ability to select additional sweep types will also need to be added when the system is expanded to detect other contaminants beyond arsenic. As far as test results, the application only displays digital, voltage, and current values from test sweeps at the moment. More testing needs to be done by the electrical engineering and bioengineering teams to acquire formulas for converting these values to contaminant concentration measured in mg/L or parts per billion when using the bioengineering sensor. Lastly, additional database operations could be added to load test results onto the phone from the online database, or to store multiple tests. Currently, the application does not save data from tests on the Android device itself, it only tries to upload the results to the web. Adding a small database on the application side could help if users do not have an internet connection when they are performing tests and want to save the results for later.

Chapter 8

Lessons Learned

Throughout the duration of our project, we have learned both technical and non-technical lessons that will be relevant to our careers. This is this result of our accomplishments and failures as we worked on our system.

8.1 Team Accountability

One lesson we learned is that it is very important to maintain accountability with each other. During the fall quarter, we met weekly to discuss our progress and goals. These meetings helped us always be aware of the recent accomplishments and problems each member had with the project. Thus, we always knew what to expect in the upcoming weeks from our teammates. Also, we were able to set weekly requirements, which provided enough pressure for us to make progress regularly. During the winter quarter, we stopped meeting every week (instead meeting every two weeks or so), and we progressed at a much slower pace. The accountability that comes with more frequent meetings really helps in moving a project forward.

8.2 Abstraction

Our project delved into a variety of topics, from electrochemistry to analog circuitry to Android programming. We were able to efficiently divide our tasks by having each member work on different aspects of the project. As such, it was important to practice abstraction. For example, the member programming for the Android device did not need to understand how the circuit works. He simply needed to know how his application would communicate with the circuit. This kind of abstraction saved us a lot of time, as we focused only on what was important for our individual tasks.

Similarly, while we were working towards the same general goal as the bioengineering research team, we still had very different assignments from them. If a problem with the system occurred, it needed to be clear whose domain it fell into. For example, if the tests were not repeatable, we identified first if that was because of the bioengineering sensor or the electrical engineering circuit. After identifying who was in the appropriate position to tackle the problem, the other party did not waste time and effort by trying to simultaneously solve it. This abstraction is relevant as many research projects are interdisciplinary, and an individual cannot master every aspect.

8.3 Big Picture

It is important never to lose sight of the big picture. It is the overall goal of a given project that dictates how each individual component has to be constructed. For example, we have done multiple tests with bulk electrodes. At a point, we decided we wanted to be able to convert the voltage values we obtained into the concentration values of arsenic. In the long run, this really is our goal because users do not want to see voltage values; they want to see how contaminated their water is. However, considering the big picture reminded us that we are building a system to be used with the bioengineering sensor, not with bulk electrodes. Thus, working towards formulating a conversion factor for the bulk electrodes would be useless in the long run because the bioengineers' three-electrode sensor would behave differently.

Remembering the big picture also reminded us of our purpose for working on the project. It is easy to get discouraged by the little details of noise seen on the circuit. However, remembering that the device we are creating can provide many benefits for emerging communities and enhance the scientific body of knowledge energized us to persevere and focus on the task.

References

[1] Naujokas, M. F. et al. The broad scope of health effects from chronic arsenic exposure: update on a worldwide public health problem. *Environ Health Perspect* 121, 295-302, doi:10.1289/ehp.1205875 (2013).

[2] Richardson, S. D., & Ternes, T. A. (2005). Water analysis: emerging contaminants and current issues. *Analytical Chemistry*, 77(12), 3807-3838.

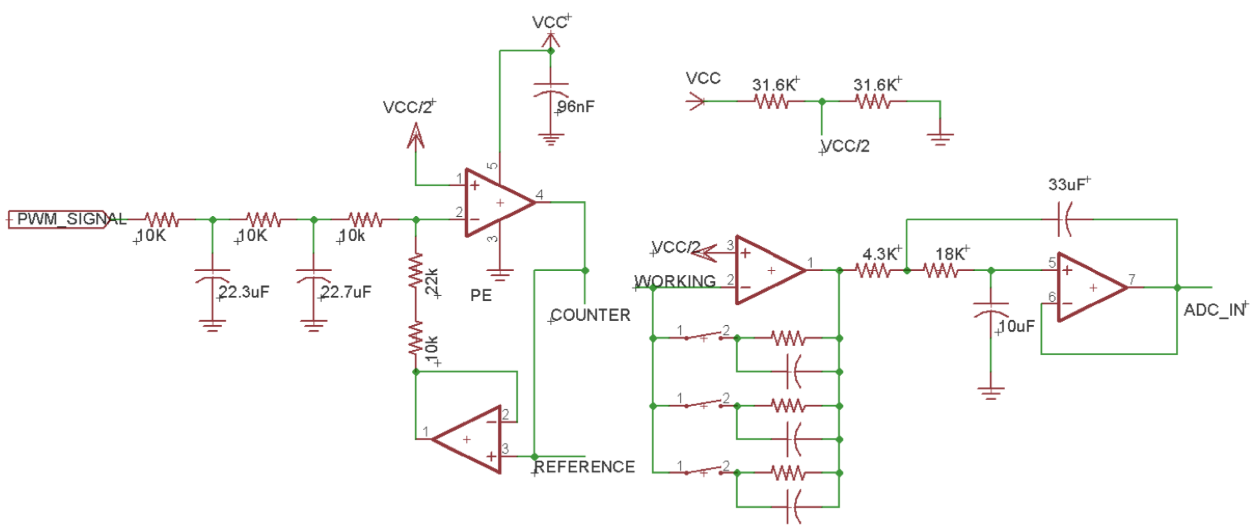
[3] Rowe, A. A., Bonham, A. J., White, R. J., Zimmer, M. P., Yadgar, R. J., Hobza, T. M., ... & Plaxco, K. W. (2011). CheapStat: An open-source,"do-it-yourself" potentiostat for analytical and educational applications. *PloS one*, 6(9), e23783.

[4] Texas Instruments. *MSP430G2553 Datasheet.* 2011. URL: http://www.ti.com/lit/ds/symlink/msp430g2553.pdf.

Appendices

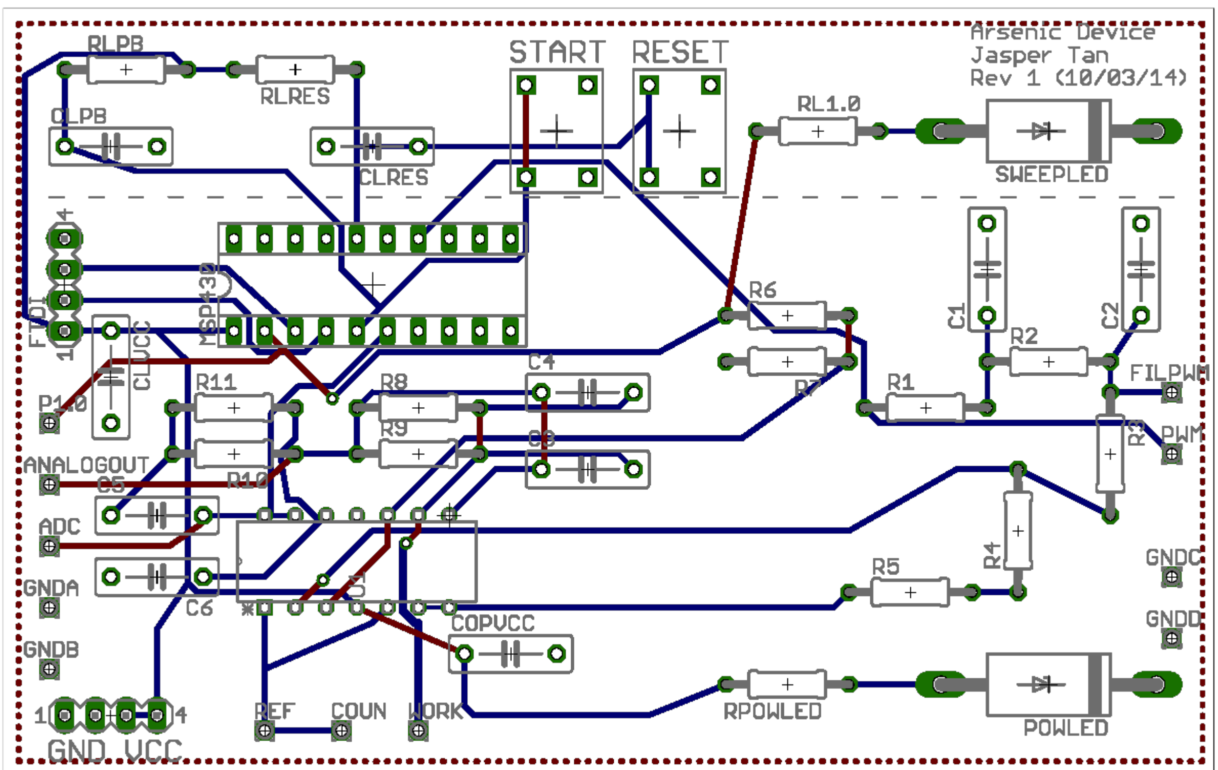
Appendix A

Potentiostat Schematic



Appendix B

PCB Rev 1 Gerber



Appendix C

PCB Rev 1 Schematic