Development of a Flame Atomic Emission Spectrometer for Lab on Chip Applications

by

Victor Grant Shadbolt

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering (Nanotechnology)

Waterloo, Ontario, Canada, 2015

Authors Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Victor Grant Shadbolt

Abstract

Flame Atomic Emissions Spectroscopy (FAES) is a powerful spectroscopic technique for trace metal analysis, however, the applications of the method is severely curtailed by its inability to deal with complex samples unless substantial sample preparation, typically elaborate and time consuming, is performed. Some examples of interferences include the classical spectral interference between Sodium (Na) and Calcium (Ca) and the prominent matrix interference between Aluminum (Al) and alkaline earth metals, in either case leading to an inability to detect the weaker spectral signature. As a result, despite being the one of the oldest chemical analysis techniques, the use of FAES in an industrial setting is primarily limited to the detection of Na, Ca, Potassium (K), Lithium (Li), and Magnesium (Mg) within simple samples.

In addition, the future of health care sees the use of Point-of-Care (POC) Lab on Chip (LOC) devices integrating one or more laboratory functions onto a single platform; typically utilizing microlitre sized volumes. These microfluidic devices, when fully developed, may contain the ability to acquire, pretreat, separate, post-treat and detect a variety of samples. This has great potential to revolutionise the health-care industry by providing fast, automated, and inexpensive POC diagnostics. A POC implementation of a FAES and LOC device would be of great importance to the medical industry.

Here, we develop the infrastructure to fabricate an ElectroThermal Vaporizer (ETV)-FAES for the purpose of addressing this limitation with complex samples; we also prepare the device for coupling to these LOC POC devices. This incorporates the engineering design surrounding the electrical hardware required for sub-micromolar detection capability on microlitre sized samples with a total expense of less than $500 CAD; excluding the spectrometer and computer. In addition, this hardware has been designed for, and is being utilized to fabricate, a LOC POC diagnostic instrument.

The addition of an ETV to the FAES improves the sample introduction efficiency, enables smaller sample volumes, improves limits of detection, and enables the ability to process complex samples while minimizing interferences.

The software is predominantly written using Python 2.7 to enable portability between operating systems. Modern engineering tools are utilized to fully test and analyze the hardware and software timings to confirm proper operation. Limitations in commercial hardware are identified, and presented; and appropriate fixes for these limitations are implemented.

This work gives a preliminary demonstration of an inexpensive, propane-only, FAES system that could be adapted for use with the small sample volumes required for LOC applications. The detection of standard targets of Li, K, Na, and Ca is characterized and found to have detection levels in the 10-100 nM regime. The system is coupled with an electrothermal source that allows selective desorption of species, partially solving the spectral overlap problem and allowing the analysis of larger dilute samples. Ultimately the system demonstrates comparable performance to commercial non-oxygen assisted systems. Although not integrated, an infrastructure was developed for coupling with LOC. Future work with an oxygen assist and LOC integration might fully address most barriers to the use of FAES in important applications.

Although the hardware is designed to be integrated in order to offer a joint LOC POC diagnostic instrument with an ETV-FAES, this thesis focuses primarily on the ETV-FAES and its development, characterization, and limitations.

Acknowledgements

Firstly, I would like to thank my supervisor, Dr. Chris Backhouse, without whom none of this would have been possible. I would also like to personally thank Dr. Cathy Gebotys, who helped me through difficult situations and provided guidance when I needed it the most. I also acknowledge the assistance of the graduate staff in the Electrical and Computer Engineering department, and the financial support provided by the University of Waterloo and NSERC.

Thank you to my fellow colleges and collaborators, especially Gordon H. Hall and Tianchi Ma, for your recommendations, friendship, and making the working environment a great place to be. I would like to acknowledge the assistance provided by past co-op students, Daniel Pinto Ramos and Hirad Raisi, for providing insightful suggestions.

Dr. Howard Siu and Jenn Coggan, you both have helped me tremendously throughout my time as a graduate student. You lent a helpful ear, provided great ideas, and helped me with access to various equipment to test out theories. I am also grateful for the fundamental advice and guidance provided to me by Dr. Robert Donkers.

I greatly appreciate the friendship provided by Mike J. Coleman, who shared a similar masters research journey of his own. To Conor Nosal, who has provided a helpful ear since the early days of High School right through to the completion on my masters. I thank you. Lastly, a special thank you to my partner, Kate Brockman, who provided untiring encouragement.

And finally, I would like to thank my loving family: my mother, Deborah Shadbolt, and my brother, Vincent Shadbolt. The two of you have helped me more than you know. Vincent, thank you for your complex design ideas. Mom, thank you for your genuine and limitless support. I love you both.

Dedication

To my father, who taught me by example the meaning of professionalism, integrity, and excellence. A man whom I thank for what I am today. You shall never be forgotten.

Table of Contents

List of Tables

List of Figures

Abbreviations

Ω	Ohm
μTAS	Micro-Total-Analysis System
μTK	Microfluidic ToolKit
A	Amperes
AC	Alternating Current
ADC	Analog to Digital Converter
AES	Atomic Emission Spectroscopy
AML	Applied Miniaturisation Laboratory
AS	Atomic Spectroscopic
ASCII	American Standard Code for Information Interchange
ATX	Advanced Technology eXtended
CCD	Charge Coupled Device
CDA	Clean Dry Air
CE	Capillary Electrophoresis

CMOS	Complementary Metal-Oxide Semiconductor
CNC	Computer Numerical Control
CSV	Comma-Separated Values
DAC	Digital to Analog Converter
DC	Direct Current
EM	ElectroMagnetic
EMC	ElectroMagnetic Compatibility
ESD	ElectroStatic Discharge
ETV	ElectroThermal Vaporizer
FAES	Flame Atomic Emissions Spectroscopy
FES	Flame Emission Spectroscopy
FFT	Fast Fourier Transform
GF	Graphite Furnace
GUI	Graphical User Interface
HSCS	High-Side Current-Sense
IC	Integrated Circuit
ICP-AES	Inductively Coupled Plasma-Atomic Emission Spectroscopy
LED	Light-Emitting Diode

LOC	Lab on Chip
LOD	Limit of Detection
LOQ	Limit of Quantification
LVDS	Low Voltage Differential Signaling
MDF	Medium-Density Fibreboard
PCR	Polymerase Chain Reaction
PID	Proportional-Integral-Derivative
POC	Point-of-Care
PPM	Parts Per Million
PSI	Pounds per Square Inch
RFLP	Restriction Fragment Length Polymorphism
SCFM	Standard Cubic Feet per Minute
SPI	Serial Peripheral Interface
TCR	Temperature Coefficient of Resistance
TTK	Tricorder ToolKit
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
V	Volts
W	Watts

W-Coil Tungsten Coil

Chemical Abbreviations

$\mathbf{CaCl_2 \cdot 2H_2O}$	Calcium Chlorate Di-Hydrate
CaOH	Calcium Hydroxide
$\mathbf{CuSO_4 \cdot 5H_2O}$	Cupric Sulfate
KCl	Potassium Chloride
$\mathbf{LiOH{\cdot}H_2O}$	Lithium Hydroxide Monohydrate
NaCl	Sodium Chloride
PMMA	Poly(Methyl MethAcrylate)

Chapter 1

Introduction

1.1 Introduction

In his historical review of work in the field, Jarrell notes that Flame Emission Spectroscopy (FES) is one of the oldest chemical analysis techniques, with some of the earliest examples dating back to the 1800s. [1] Its operating principle stems from elements having unique spectral signatures. The flame, acting as the thermal source, excites the elements within the sample, thereby enabling them to emit characteristic wavelengths. Monitoring these wavelengths allows for the quantification of the metallic elemental composition of the sample. [2, 3] As sensitivities have increased, the technique has donned the name of Flame Atomic Emissions Spectroscopy (FAES). Despite serious limitations due to spectral interferences, such as the classical spectral interference between Na and Ca and the prominent matrix interference between Al and alkaline earth metals, this technique is considered mature and it remains a standard in trace elemental analysis. [4–9] In the last few decades it has become one of the most important techniques for trace metal determinations. [10] However, due to the present limitations of the technique, which mostly consists of interferences and limited temperatures, the use of FAES in an industrial setting is primarily limited to the detection of Na, Ca, K, Li, and Mg within simple samples [11–17]. As noted by [18], the use of non-oxygen assisted flames, such as propane and acetylene, limits the temperature to less than 2,400°C; thereby limiting elements to those that are readily excited. Such elements are listed above.

Trace elemental analysis, when applied to biological samples, allows for the determination of the concentration of essential elements in both normal and disease conditions. This, when coupled with the ability to detect potential toxic metals, allows for the diagnosis of trace-element deficiencies and related diseases. [19] This biological analysis is typically performed in centralized labs; and because of this centralization, turnaround time for biological samples can be quite long. As a result, there has been a strong push towards Point-of-Care (POC) testing. [20] This is being realized through the use of microfluidic Lab on Chip (LOC) devices.

A LOC POC device, or Micro-Total-Analysis System (μTAS), integrates one or more laboratory functions onto a single platform typically utilizing sub-microlitre volumes. [21] These microfluidic devices, when fully developed, may contain the ability to acquire, pretreat, separate, post-treat and detect a variety of samples. [22] This has great potential to revolutionize the health-care industry by providing fast, automated, and inexpensive POC diagnostics. [23, 24] Presently, to our knowledge, there has been no demonstration of FAES in conjunction with such a LOC POC device.

A LOC compatible FAES must be: a system that can be moved from location to location easily without the need for extensive setup; capable of analyzing sub-micro to

miro-litre samples with Limits of Detection (LODs) in the picomolar to nanomolar range; have analysis times of less than 30 seconds; and be inexpensive.

In this thesis, the work presented is an initial foray into coupling the capabilities and benefits of the traditional FAES with the revolutionary LOC μTAS devices. This is through the redesign of the FAES specifically for LOC applications. Keeping with the idea of POC, the traditional FAES is redesigned to utilize the latest in sample introduction techniques, require microlitre sized samples, and be extremely cost effective. In addition, a novel sample introduction technique is developed to solve the curtailing limitation of spectral interferences associated with complex samples. The developed hardware is designed to be used for both the FAES as well as a LOC μTAS, either separately or integrated.

The final system was not tested in application to LOC as there was insufficient time. A possible approach to integration and testing is proposed in the Conclusion and Future Directions Chapter.

1.2 Objectives

The initial motivation behind the LOC application of FAES was to create LOC-based testing for Copper (Cu) in volumes of 10 μL at concentrations of 200 μM. Early into the research, while using a propane flame source, it was realized that a hotter thermal source would be required to adequately detect Cu. It was decided to implement a hotter flame after fully developing the system in order to simplify its development. Unfortunately, time ran out before this was achieved. For the system development itself, asides from implementing a hotter flame, three main objectives were established.

The first objective of this research is to further enhance the field of FAES, and μTAS, by designing a LOC POC compatible FAES. In addition, the development of a μTAS designed to be used in conjunction with the FAES would be beneficial. As such the hardware designed is to be compatible with a stand-alone FAES, a stand-alone μTAS, and a joint FAES-μTAS. The μTAS development itself is outside the scope of this thesis.

The second objective of this research is to maximize the availability of FAES to the masses. In order to do this, the total cost of the system should be kept as low as possible while maximizing the instrument's performance characteristics.

The third, and final, objective of this research is to improve upon the FAES technique itself in addition to the development of a FAES instrument, capable of coupling to a LOC POC system. This improvement should come in the form of addressing the fundamental limitation of the technique: spectral interferences within complex samples.

1.3 Thesis Overview

As might be expected from a relatively mature field, a search of research databases such as Web of Science and IEEE Explore did not return any references that matched the topic criteria of: Review and Flame Atomic Emissions Spectroscopy or Review and FAES. As such, reviews in broader or related areas such as those of Evans et al. [25] and Butcher [26] (in Atomic Spectroscopic (AS) methods) and Hanna and Jones [27] (in ElectroThermal Vaporizer (ETV) use with AS) were used. Within each chapter, the appropriate literature is presented and discussed in lieu of the traditional literature review chapter.

Chapter 2, Mark-1: Nebulizer, discusses the initial development of a FAES with an off-the-shelf propane blowtorch thereby vetting the use of the torch as a flame source and the use of the spectrometer for multi-element detection. This version of the FAES utilizes a nebulizer based approach to atomize the sample of interest. Over the course of three generations of development, the final Mark-1 instrument required sample sizes of 100 mL, which is significantly larger than the targeted microliter volume sizes required for LOC applications; Although rivalling that of commercial instruments available on the market, the detection capabilities were limited to micromolar detection levels. The temperature of the flame and the sample introduction methodology is found to be the limiting aspects for improving detection. The chapter ends with a look at the next steps in improving performance: the Mark 2.

Chapter 3, Mark-2: ETV Development, discusses the design and development of the Tungsten Coil (W-Coil) ETV. In order to ensure the Mark 2 remains open source, a freeware version of Cadsoft Eagle is utilized and the software is written in Python 2.7. The development of a power delivery system capable of controlled sustainable power output of 120 Watts (W) is documented, and the control architecture is explained in detail. Most importantly, however, is that the use of the ETV allows for variable sample volumes and automated cleaning leading to rapid analysis. [4] In this thesis, volumes on the order of microlitres are used; enabling the coupling to LOC POC diagnostics.

Chapter 4, FAES and μTAS Hardware Development, discusses the hardware requirements for the dual-purpose modules capable of serving both a LOC compatible ETV-FAES as well as a LOC POC device. These circuits are designed using the freeware version of Cadsoft Eagle, and they are programmed using Python 2.7. The modules themselves are targeted to solve problems relating to power harmonics and unreliable communications in high electric fields. The modules are comprised of an Low Voltage Differential Signaling (LVDS), ETV Proportional-Integral-Derivative (PID) Heater Controller, Interface Board, Constant Current Source Board, High-Voltage Fluidics Board, Solenoid Board, Photodiode

Board, and Environmental Monitoring. Basic software control and is presented for each module.

Chapter 5, LabJack and Ocean Optics Timing, discusses the limitations of some commercial hardware utilized for the Mark 2 ETV-FAES. More specifically, the LabJack Serial Peripheral Interface (SPI) implementation, which is documented as a solid, functioning, feature shows sporadic timing. This results in errors on connected SPI devices, and is demonstrated and documented. A solution to the problem is investigated. In addition to the LabJack timing issues, the Ocean Optics USB4000 triggering options are investigated. It is discovered that the triggering does not function as documented by Ocean Optics, and a crude fix is developed and implemented. This triggering error of the Ocean Optics, at the time of this writing, is currently under review by Ocean Optics Inc.

Chapter 6, Mark-2: ETV Application, discusses the Mark 2 ETV-FAES capabilities. The instrument's electronic and software overview is presented in order to document how the instrument has been fabricated and how it is controlled. In addition, performance characteristics such as the LOD, Carry-Over, and flame stability are discussed. With a sample size of 10 microlitres, the instrument closely approaches the target of microlitre sized samples. More importantly, however, preliminary testing of the LOD of the system suggests LODs to be in the tens of nanomolars, demonstrating significant improvement over the Mark-1. To our knowledge, this is the highest sensitivity ever achieved for a LOC compatible ETV-FAES utilizing a propane flame source from an off-the-shelf blowtorch. The total cost of the instrument, minus the spectrometer and computer, is less than $500.

Chapter 7, Removing Interferences in Atomic Emission Spectroscopy, discusses the fundamental limitation of the ETV-FAES: interferences. More specifically, this chapter discusses the use of the ETV to measure the interfering components separately at successively higher temperatures, thereby converting the complex sample into a series of simple samples, thereby minimizing interferences. This approach circumvents the need to external sample preparation and is demonstrated by detecting a micromolar concentration of Ca in the presence of an isotonic concentration (154 millimolar) of Na. To our knowledge this is the first use of an ETV to solve this fundamental challenge to the use of a LOC relevant implementation of FAES.

Chapter 8, Conclusion and Future Directions, discusses the contributions relative to the state of the art, conclusions and possible future directions for this project; most notably, some possible methods for coupling the ETV-FAES with a LOC POC diagnostic instrument. This is proposed through the use of microfluidic pumping, supported by the Solenoid Board, to couple the output of the microfluidic chip with a stainless steel needle to deliver the sample to the ETV.

Chapter 2

Mark-1: Nebulizer

2.1 Introduction

In this chapter the development of the Mark 1: Nebulizer is discussed. Three generations of modifications to the Mark 1 result in having detection capabilities significantly better than commercial flame photometers on the market as of this writing. The major shortcomings of this architecture, however, are the use of a ultrasonic nebulizer which necessitates large volumes of sample, and cumbersome cleaning procedures. These limitations necessitate the revision of sample introduction; however, the Mark 1 Nebulizer serves the purpose of vetting the use of conventional tools and equipment available at a local hardware store for a fraction the price of a commercial system.

It is important to vet these tools and as such the purpose behind the Mark 1: Nebulizer is to answer a few key questions: Will the basic propane flame from an off the shelf blowtorch be a suitable flame source for a Flame Atomic Emissions Spectroscopy (FAES)? and How feasible is the use of the spectrometer chosen for performing simultaneous analysis of multi-element samples? Answering these questions requires a simple FAES implementation which could serve as a platform for modifications.

The setup of this Mark 1: Nebulizer system along with the separation of the three modules is discussed. These three modules, the flame, optical, and sample injection, comprise the entire system and as such are able to be individually modified without affecting the other two. The system is then characterized for sample flow rate, minimal sample volume, and the detection limits of five select elements. To conclude the chapter, a multi-element test is performed in order to determine the ability of the system to simultaneously detect multiple elements.

2.2 Setup

The Mark-1 Nebulizer can be segregated into three different modules: The Flame Module, The Optics Module, and the Sample Injection Module. Within this, there were three generations of improvements to the Mark 1 Nebulizer FAES.

The functional diagram behind the entire setup is presented in Figure 2.1. The methodology behind its operation in its most simplistic form is as follows:

1. Sample is aerosolized by means of a nebulizer.
2. Aerosolized sample is injected into the base of the flame.
3. Photonic emission from the sample is collected by the collection lens
4. Collection lens focuses the light onto a collimating filter

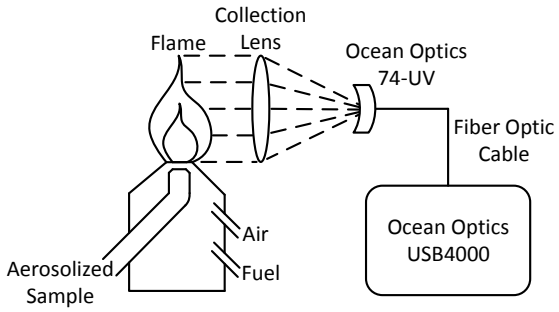


Figure 2.1: Functional diagram of the Mark 1 Nebulizer. The aerosolized sample is generated by a nebulizer; the air fuel mixture is provided by the BernzOmatic TS4000 Blowtorch; The glass collection lens is unknown but has a focal length of $5\frac{1}{2}$ inches and has a diameter of 1 inch; The collimating lens, which focusses the light into a Fiber Optic Cable, is a Ocean Optics 74-UV; The Fibre Optic Cable is an Ocean Optics P200-2 UV/VIS; The Ocean Optics USB4000 is the spectrometer used to collect the data.

5. Collimating filter collimates the light for transport to the photodetector
6. Photodetector and accompanying electronics provide digital readout of wavelength intensity.

Generation Two of the entire setup can be seen in Figure 2.2.

Within this section, a description of each module will be presented and discussed.

2.2.1 Flame Module

The flame module consists of a *BernzOmatic TS4000* blowtorch coupled with either a *Worthington Propane*, or *Worthington MAP//Pro* gas cylinder. As a result, this flame is able to produce an estimated 2200 K to 2330 K at the tip, respectively, allowing for elemental excitation. [28, 29] Temperatures of this level are typically used to measure easily excitable elements such as Ba, Ca, Li, K, and Na. [18] For Cu detection, temperatures as low as 2600 K, from an air-acetylene flame, have been used, although typically flame sources of 3200 K or hotter are utilized. [4, 18, 30, 31]

The energy levels of each element are unique as the atom can only exist with certain discrete values of internal energy. The lowest possible energy of the atom is called the ground level, and each electron at this energy level is said to be in the ground state. As the outer electrons, which are loosely bound to the atom, are excited by the flame they

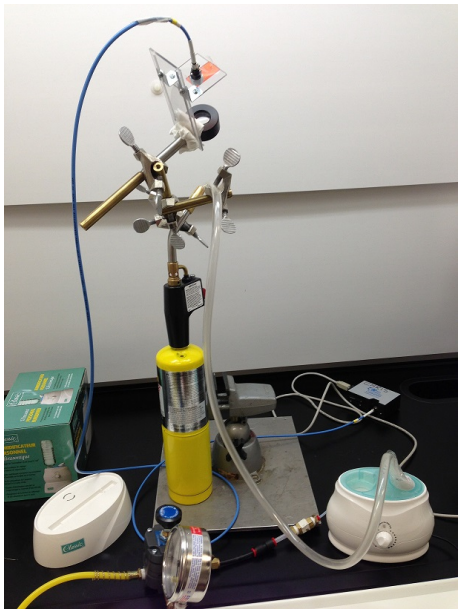


Figure 2.2: Image of the Flame Photometer of Mark-1 Nebulizer Generation Two. The aerosolized sample is generated by an ultrasonic nebulizer; The flame is generated by a BernzOmatic TS4000;The collection lens and collimating lens is a Ocean Optics 74-UV; The photodiode and respective electronics is a Ocean Optics USB4000. In order to ensure adequate sample injection, externally regulated Clean Dry Air (CDA) at 2 Pounds per Square Inch (PSI) is utilized to pressurize the ultrasonic nebulizer.

enter an excited state. The electron in the excited state sheds the excess energy in the form of a photon, in order to return to the ground state. [32]

In order to achieve consistency in the emission intensity from samples of identical composition and volume, the flame source must be consistent with little variability. These two characteristics are tested by coupling the flame module with the optical module discussed in Section 2.2.2, and monitoring the individual spectra of the fuel gasses themselves.

These spectra were collected on the first generation of flame photometer and are the average of 10 collections with the average of 10 backgrounds immediately prior to collection, subtracted. The integration time for each collection was 500 milliseconds. No samples were injected during the tests. Over the course of the 10 collections, the average intensity of the Propane spectrum was 2500 counts, and demonstrated a standard deviation of 48 counts, or 1.96%; the average intensity of the MAP//Pro spectrum was 7200 counts, and demonstrated a standard deviation of 78 counts, or 1.09%.

The collected Propane and MAP//Pro (Propylene) emission spectrum's are presented in Figure 2.3 on Page 11, with an offset between them for easy viewing. [33] As can be seen, the collected spectra are visually very similar to each other with peaks in similar locations. The relative peak intensities are as expected for the gases, while the multiple peaks are due to hydrocarbon fragments burning. These hydrocarbon fragments include C_2, CH, and CH_2. [34]

These tests were repeated for all three generations of the Mark 1 with similar results; demonstrating flame consistency and little variability.

2.2.2 Optical Module

The optics module was fabricated in the Student Machine Shop to support holding a one inch diameter optical lens with a focal length of 2.5 inches, as well as an *Ocean Optics 74-UV* Collimating lens on the end of the TS4000 blowtorch. A fiber optic cable then transmits the collected light to a *Ocean Optics USB4000* spectrometer connected to a computer through a Universal Serial Bus (USB) 2.0 interface. Although it has not changed much, there have been subtle revisions to each generation of Mark-1 Nebulizer FAES. The first generation module can be seen in Figure 2.4 on Page 11. Generation two can be seen in Figure 2.2 on Page 9. Generation three, although not imaged here, is identical to generation two, however, Poly(Methyl MethAcrylate) (PMMA) support structures were implemented to improve stability of the system.

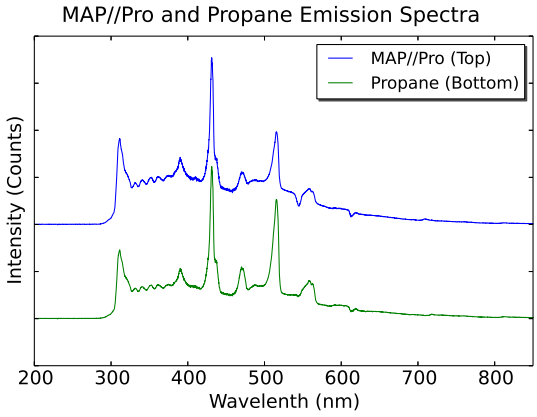


Figure 2.3: Average of 10 Captured Emission Spectra for MAP//Pro (Propylene) and Propane Flames on Mark-1 Nebulizer Generation 1 using Ocean Optics USB4000 Spectrometer with 500 millisecond integration time. Standard deviation of intensity is 1.96% and 1.09% for Propane and Propylene respectively. The temperature of the Propane flame is estimated to be 2200 K while the temperature of the Propylene is estimated to be 2330 K. [28, 29]

Figure 2.4: BernzOmatic Torch Optical Rig for Mark-1 Nebulizer Generation 1.

2.2.3 Sample Injection Module

The typical flame from the TS4000 when using Propane gas is a nice blue, identical in observational characteristics to the flame in Figure 2.4 on page 11, which is using the MAP//Pro gas. Generations one through three utilized a *Classic Ultrasonic Personal Humidifier* to generate a cool mist (essentially a ultrasonic nebulization) of the sample which is then transported into the flame of the TS4000 blowtorch. On generation one it was observed that replacing the air-intake on the TS4000 with the output of the ultrasonic nebulizer was sufficient to thoroughly mix the nebulized sample with the gas prior to ignition. When a solution of Sodium Chloride (NaCl) in regular tap-water was nebulized, and then introduced into the TS4000, a very uniform and bright yellow flame resulted, and can be seen in Figure 2.5a. The yellow is from the Na emission wavelength of approximately 589 nm. [35]

Unfortunately for generation one, it was found that mineral build-up was occurring within the TS4000 itself, and obtaining reliable and reproducible spectra was a significant challenge. The TS4000 needed to be cleaned thoroughly between tests by flushing the TS4000 with Milli-Q, which then necessitated sufficient time to dry the internal workings of the TS4000 in order to operate properly. As a result, a new methodology needed to be adapted: enter generation two of the nebulizer.

Generation two focused on injecting the sample to the base of the flame itself. In order to ensure continuous and constant flow of sample, a regulated CDA line was used to push the sample from the nebulizer itself to the flame. This required some modifications of the humidifier, more specifically, the interlock to determine whether or not there was fluid within the sample chamber needed to be removed; as well as sealing off the nebulizing chamber from the environment in order to maintain pressure. This updated nebuliser setup can be seen in Figure 2.2 on Page 9. It was found, however, that the long tubing required to transport the sample was ineffective resulting in significant sample loss and carry-over. In addition it was found that slight alterations in the injection location caused significant distortions in the acquired spectra.

Generation three aimed to improve upon these issues. This was done by shortening the tubing to the flame, and creating a customized tube holder rather than relying upon crude tape and caulking methods. In conjunction to shortening the travel distance of the nebulized sample, a rig to hold the tubing exhaust in the exact same position relative to the TS4000 was fabricated and implemented. This allowed for the entire disassembly and re-assembly of the generation three setup while maintaining reproducibility. Also a low-flow rate 2 PSI CDA line was utilized to transport the sample to the flame itself. It must be noted that the tubing was sufficiently close to the flame that under some circumstances the

flame would singe the tubing. The flame produced, however, is qualitatively of significantly higher quality than that observed previously. The flame is visually more stable and no modifications to the TS4000 needed to be done leading to a safer working environment. The flame can be seen in Figure 2.5b on page 13.

(a) (b)

Figure 2.5: Image of TS4000 Propane flames as an Ultrasonic Nebulizer introduces aerosolized NaCl solution into the Air/Gas mixture.

(a) Generation 1: Sample is introduced prior to flame ignition via the venturi effect at the air-intake on the TS4000. Sufficient turbulence thoroughly mixes the nebulized sample with the gas resulting in complete yellow flame.

(b) Generation 3: Sample is injected into the flame post ignition via continuous CDA carrier. This method resulted in a much smoother and hotter flame as well as significantly reducing any safety concerns by avoiding modification to the TS4000.

2.3 Characterizing Single Element Detection

With Mark-1 Nebulizer Generation Three, initial performance characteristics need to be determined. This will allow for the identification of potential applications in addition to

evaluating its ability to be used for Lab on Chip (LOC) applications. The initial elements of interest are Na, Li, Cu, K, and Ca. These elements, with the exception of Cu, are commonly analyzed with commercial flame photometers which utilize a low temperature flame such as propane-air as used in the Mark-1 Nebulizer. [18]

2.3.1 Preparing Samples

Each of the five elements were prepared individually using a *Mettler Toledo AG104* microbalance to weigh out appropriate masses of each chemical. 100 mL solutions of 1, 10, and 100 μmol L^{-1} (μM) NaCl, Calcium Chlorate Di-Hydrate ($CaCl_2 \cdot 2H_2O$), Potassium Chloride (KCl), Lithium Hydroxide Monohydrate ($LiOH{\cdot}H_2O$), and Cupric Sulfate ($CuSO_4 \cdot 5H_2O$) were made from Milli-Q water. Table A.1 in Appendix A on Page 144 lists the characteristics for these elements.

100 mL volumes were used as that was the minimum required volume for the ultrasonic nebulizer. Significantly lower volumes resulted in failure of the ultrasonic membrane thereby stopping the aerosolization of the sample.

2.3.2 Collecting Data

The data was collected using *Ocean Optics Spectra Suite v2.0.162* using a *Ocean Optics USB4000 firmware 0.99.1* spectrometer. The optics module, as described in Section 2.2.2, collects the light into a *Ocean Optics P200-2-UV/VIS* fiber-optic cable which transfers it to the spectrometer. The software was set to acquire 20 spectra with an integration time of two seconds, and save each spectra to a separate delimited American Standard Code for Information Interchange (ASCII) file.

2.3.3 Processing Data

The collected spectra were amalgamated into a single ASCII Comma-Separated Values (CSV) file using the python script in Appendix B.2. This script requires that the parent folder contains only the script along with the delimited ASCII **.txt* files from the Ocean Optics software and will output a single *Compiled_Data.csv* file with a simple data structure. An example of this data structure can be seen in Table 2.1 on Page 15.

Once the data has been compiled into a ASCII CSV file for each concentration for a particular element, a MATLAB script is run. This script is designed to generate a complete

Table 2.1: Example Data Structure in *Compiled_Data.csv* as built by Python Script in Appendix B.2

Wavelength (nm)	Test 0 (Counts)	Test 1 (Counts)	Test 2 (Counts)	⋯	Test n (Counts)
178.32	1277.93	1296.36	1288.17	⋯	1288.17
178.53	1277.93	1296.36	1288.17	⋯	1288.17
178.75	1277.93	1296.36	1288.17	⋯	1288.17
⋮	⋮	⋮	⋮		⋮
886.57	2353.12	2416.6	2562.01	⋯	2498.52

report outlining the code, figures, and results from the collected data. This code reads the raw Ocean Optics ASCII CSV data and processes it to determine the Limit of Detection (LOD), Limit of Quantification (LOQ), and their associated Parts Per Million (PPM) values. During this process there is no background subtraction performed.

The LOQ is important as it calculates the minimum concentration with which the values can be confidently determined. The LOD determines the minimum concentration which can be confidently detected.

The MATLAB codes utilized can be seen in Appendix C. More specifically, the code used to read the ASCII CSV file into an array in which MATLAB may process the data is in Appendix C.3. The code used to process this rebuilt array with Linear Regression Analysis and then plot the results is in Appendix C.4.

The data used for analysis is extracted from the compiled data. This is performed by stating which wavelength to monitor from Test 0 to Test n, as well as stating which concentration of the element corresponds to each compiled data file. For example, if the four concentrations used are 0 μM, 1 μM, 10 μM, and 100 μM and the desired wavelength is 589 nm (Na), a four by n array comprising of the data at wavelength 589 nm is generated. This specific array is still the raw data, but localized for the element of interest.

The raw data is then analyzed by following a Linear Regression Analysis. This is performed by curve-fitting a One-Dimensional polynomial to the data (Linear Regression) through the use of a Least Squares method as presented in Appendix C.4. A detailed discussion on Linear Regression Analysis is available in any good statistical analysis textbook such as *Statistics for Experimenters: Design, Innovation, and Discovery* and its MATLAB equivalent in such texts as *Applied Numerical Methods with MATLAB for Engineers and Scientists* [36, 37]

The data is assumed to be normally distributed via a T-Distribution with a 95% confi-

dence t-value of 12.7062 ($\alpha = 0.05$, degrees of freedom $(n-2) = 1$). This is used as the sample size is small and the standard deviation of the data is unknown. [36]. Once the model has been calculated, the LOD and LOQ are calculated following Equations 2.1 and 2.2 respectively.

$$LOD = 3 \times \left(\sqrt{\frac{(Residual\ Sum\ of\ Squares)}{(Slope\ of\ Model)}} \right) \tag{2.1}$$

$$LOQ = \frac{10}{3} \times LOD \tag{2.2}$$

The uncertainties of the calculated LODs and LOQs are determined based upon the 95% confidence interval of the slope. The minimum and maximum values of the slope of the model are then utilized to provide an error range for the calculated values.

2.3.4 Characterization of the Elements

Analyzing the three concentrations (1, 10, and 100 μM) of Na, Li, Ca, K, and Cu following the methodology discussed in Sections 2.3.1-2.3.3 results in Figure 2.6. These data values were not background corrected. Contained within this figure are the results of the linear regression analysis along with 95% T-confidence intervals (CI) for each of the 60 data points for each concentration for the elements. In addition, the LODs and LOQs are presented. Error bars are included for each data point. Larger variability in the Ca data is possibly due to the monitoring of the Calcium Hydroxide (CaOH) peak at 622 nm; rather than the Ca peak at 422 nm which is masked by the flame emission of the propane gas. Furthermore, it is well documented that CaOH dissociates at higher temperatures to form CaO. [38]

The data is processed without background removal leading to higher than zero y-intercepts. This is done due to the methodology implemented to perform the Linear Regression Analysis as discussed in Section 2.3.3. The data does follow a linear trend, as expected.

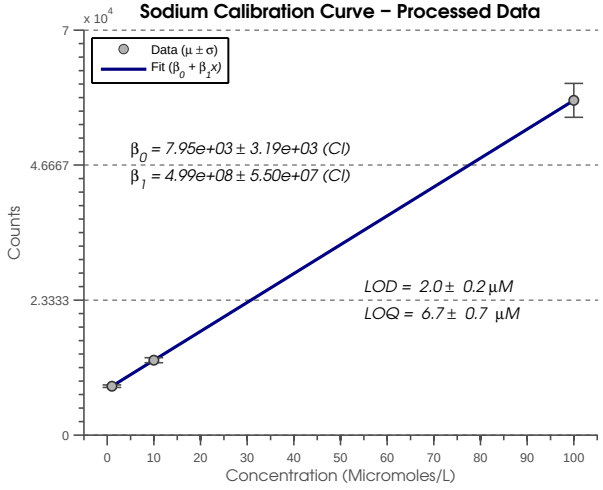
Processing the spectra reduced the noise to around 200 counts from over 1000. This allows for a preliminary estimate of the LOD. Based upon the slopes of the elements in Figure 2.6, the LOD is estimated by dividing three times the noise by the slope (β_i); For Na this is 1.2 μM, Li: 8.9 μM, K: 1.0 μM, Ca: 64.1 μM, and Cu: 379.74 μM. The calculated LODs are similar for all but Ca and Cu. This may be due to lower noise levels

at the wavelengths of interest (available in Appendix A). A detailed investigation was not performed.

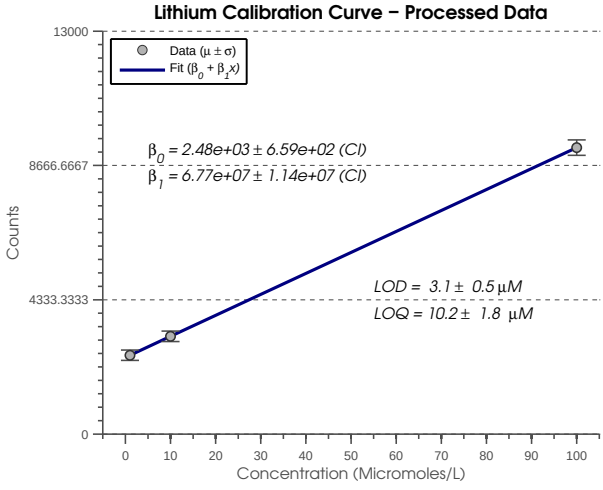
For each element, tests were performed in the order listed in Table 2.2. These test runs were repeated in triplicate over the course of three days for each element (one day per test run, three runs per element) with thorough cleaning of the entire apparatus between samples in order to eliminate carry-over. As discussed in Section 2.3.2, 20 data points were collected at each concentration for each test run. Only the data points between 1→100 μM were plotted in Figure 2.6 as the remaining data points were below the calculated LOD and lost in the noise.

Table 2.2: Order of Concentrations Tested for Each Element

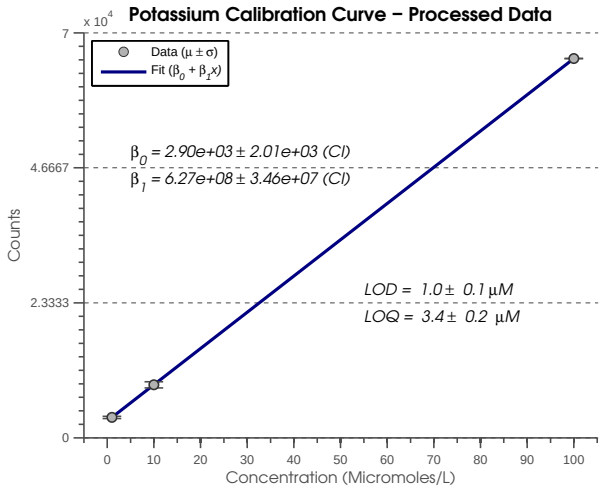
Test	Concentration
1	100 mM
2	10 nM
3	10 μM
4	1 μM
5	10 mM
6	1 nM
7	1 mM
8	100 nM
9	100 μM


Sodium Calibration Curve − Processed Data
x 10^4
Data (μ ± σ)
Fit (β0 + β1x)
β0 = 7.95e+03 ± 3.19e+03 (CI)
β1 = 4.99e+08 ± 5.50e+07 (CI)
LOD = 2.0 ± 0.2 μM
LOQ = 6.7 ± 0.7 μM
Counts
Concentration (Micromoles/L)

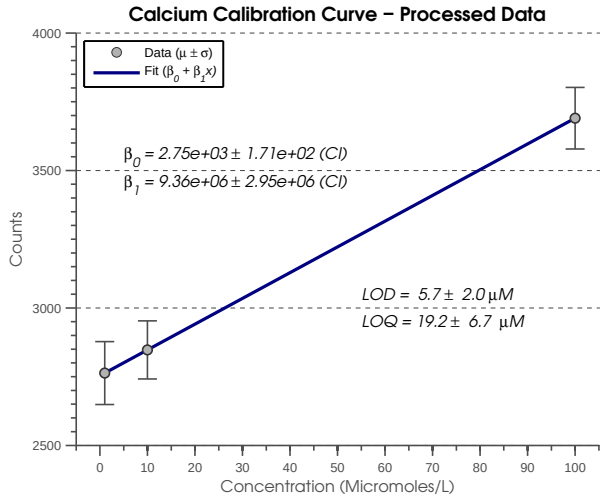
(a)


Lithium Calibration Curve − Processed Data
Data (μ ± σ)
Fit (β0 + β1x)
β0 = 2.48e+03 ± 6.59e+02 (CI)
β1 = 6.77e+07 ± 1.14e+07 (CI)
LOD = 3.1 ± 0.5 μM
LOQ = 10.2 ± 1.8 μM
Counts
Concentration (Micromoles/L)

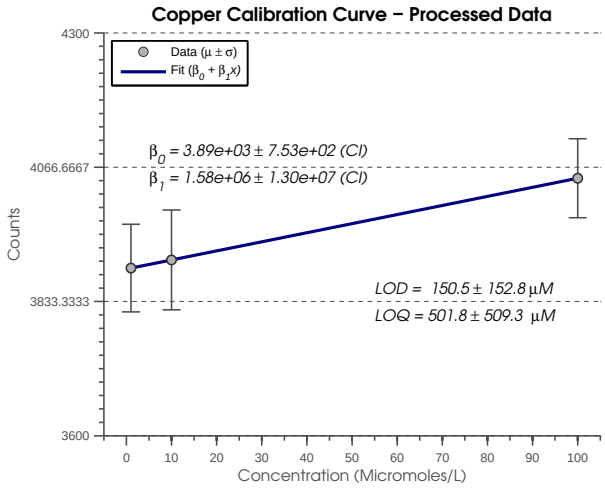
(b)


Potassium Calibration Curve − Processed Data
x 10^4
Data (μ ± σ)
Fit (β0 + β1x)
β0 = 2.90e+03 ± 2.01e+03 (CI)
β1 = 6.27e+08 ± 3.46e+07 (CI)
LOD = 1.0 ± 0.1 μM
LOQ = 3.4 ± 0.2 μM
Counts
Concentration (Micromoles/L)

(c)


Calcium Calibration Curve − Processed Data
Data (μ ± σ)
Fit (β0 + β1x)
β0 = 2.75e+03 ± 1.71e+02 (CI)
β1 = 9.36e+06 ± 2.95e+06 (CI)
LOD = 5.7 ± 2.0 μM
LOQ = 19.2 ± 6.7 μM
Counts
Concentration (Micromoles/L)

(d)



(e)

Figure 2.6: Mark-1 Generation Three Calibration Curves as Processed with Linear Regression Analysis including LODs, LOQs, and 95% T-Confidence Intervals (CI) for the Five Elements of Interest over the range of 1 to 100 micromoles. All 60 data points for each concentration of each element are processed without background subtraction leading to higher than zero y-intercepts.

(a) Na, demonstrating a LOD and LOQ of 2.0 ± 0.2 μM and 6.7 ± 0.7 μM respectively. Measured at 589 nm.

(b) Li, demonstrating a LOD and LOQ of 3.1 ± 0.5 μM and 10.2 ± 1.8 μM respectively. Measured at 671 nm.

(c) K, demonstrating a LOD and LOQ of 1.0 ± 0.1 μM and 3.4 ± 0.2 μM respectively. Measured at 766 nm.

(d) Ca, demonstrating a LOD and LOQ of 5.7 ± 2.0 μM and 19.2 ± 6.7 μM respectively. Measured at 622 nm.

(e) Cu, demonstrating a LOD and LOQ of 150.5 ± 152.8 μM and 501.8 ± 509/3 μM respectively. Measured at 327.8 nm. These errors indicate that Cu is not reliably detected.

Cu was chosen as a preliminary element of interest as it is biologically significant to our collaborative lab in the Biology Department at the University of Waterloo. More specifically, Cu plays a role in the life cycle of yeast and is currently one of the elements under investigation for the role it plays. [39] In order to increase sensitivity towards Cu detection, a significantly higher flame temperature is required; others have used an oxygen-acetylene flame, at 3475K, as a result. [4, 18, 40] This oxygen assisted flame requires

welding equipment and infrastructure, which would rapidly increase the complexity of the system. As a result, it was decided to continue to use the relatively simple propane flame source, and TS4000, for system development and implement a hotter flame later.

A comparison of the results for these elements to what is commercially available from Sherwood Scientific Ltd. is presented in Table 2.3 on Page 20. [41] The Model 360 instrument was chosen as a representative model to what is available commercially from multiple companies, is of a similar type with no oxygen assist, and is used in many industries solely for the detection of Barium (Ba), Ca, Li, K, and Na. The respective LODs for these elements are provided by Sherwood Scientific Ltd.. [41] As can be seen, the Mark 1: Generation 3 FAES demonstrates a minimum of 100% improvement over the Model 360 Flame Photometer for the elements tested on both instruments. This improvement is likely due to the use of an ultrasonic nebulizer, in comparison to the traditional nebulizer on the Model 360, which increases the sample introduction efficiency.

Table 2.3: Comparison of the preliminary LODs for the Mark 1: Generation 3 FAES and the LODs for the Sherwood Scientific Ltd. Model 360 Flame Photometer. [41] A minimum of 100% improvement, over what is commercially available, is realized. This is likely due to the use of an ultrasonic nebulizer which increases the sample introduction efficiency.

	LOD (μM)	
Element	Model 360 Flame Photometer	Mark 1: Generation 3
Na	4.3498	2.0144
K	2.5577	1.0105
Li	14.4071	3.0693
Ca	49.9027	5.7496
Cu		150.5444

The reliability of the detection on the Mark 1 is highly dependent upon post-processing of the data. Within each concentration for each element there are multiple samples, which are averaged to reduce noise levels. This averaging of the data reduces noise by a factor of about 10, thereby improving the LODs by a factor of about 10. In addition, the results obtained are for idealised samples. As such the LODs calculated and presented are only preliminary.

2.4 Multiple Simultaneous Elemental Detection

In order to predict the simultaneous detection of multiple elements, the individually collected spectra for Na, Li, Cu, K, and Ca were combined. This was accomplished by overlaying the spectra and effectively tracing the uppermost edge in MATLAB. The spectra themselves were treated prior to merging by the subtraction of an averaged background. A concentration of 10 μM for Na, Li, K was used while a concentration of 100 μM Ca and 500 μM Cu were selected to maximize peak heights. A single test comprised of all elements was not performed on the Mark 1.

An averaged background needs to be established for all elements. In order to accomplish this, it is important to identify any anomalies between the backgrounds which could indicated an error in the spectrum collection, and is accomplished by overlaying all background information as presented in Figure 2.7. As can be seen, the backgrounds are virtually identical for the different elements, with the Na background off by a little over 500 counts. This indicates proper standard operating procedure for the cleaning and resetting of the Mark-1 Nebulizer between the elements and their respective concentrations tested. This collection of backgrounds is then averaged as seen in Figure 2.8 to generate a common background suitable for the tested elements. The reduced background of Na will lead to a slight downshift of the background subtraction of the other elements potentially resulting in increased noise.

With the average background subtracted from each elemental spectra, the resulting data is merged into a single plot. These spectra are overlayed as can be seen in Figure 2.9, and have a 200-400 count difference. This difference is more than likely due to the background averaging, however, the data was collected over the course of several weeks and it is possible slight environmental changes resulted in emission intensity variations.

The collection of spectra will be added by taking the highest count rating associated with each wavelength and can be seen in Figure 2.10. This merged data clearly shows the well defined peaks associated with Na at 588.71 nm; Ca at 624.68 nm; Li at 671.55 nm; and K at 768.03 nm. Unfortunately the Cu spectra at 327.8 nm, when combined with other elements, is lost within the noise.

It can therefore be concluded that the individual elemental peaks are specific and significant once we remove the background and add the additional noise when merging the data. The exception is copper which has a very weak signal. It is therefore anticipated that combined mixtures of the elements would produce the same combined result as presented in Figure 2.10. It also stands that the LODs would not be significantly affected within a mixture due to the isolation of each peak.

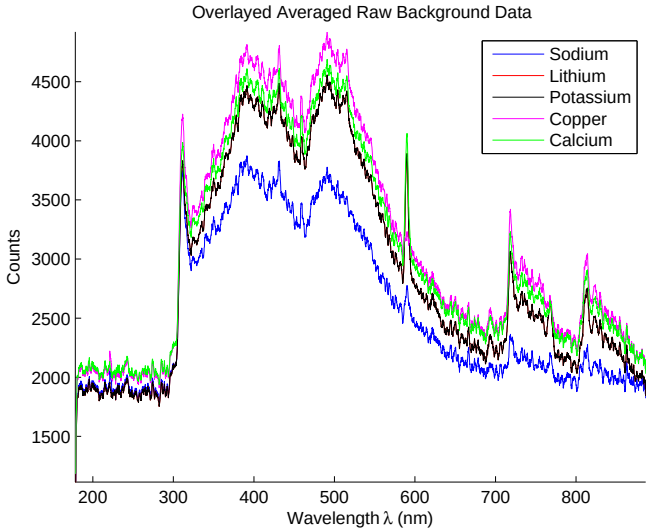


Figure 2.7: Overlayed elemental backgrounds for Mark-1 Nebulizer Multiple Simultaneous Elemental Detection. Na, in blue, demonstrates a lower background due to slightly different collection parameters.

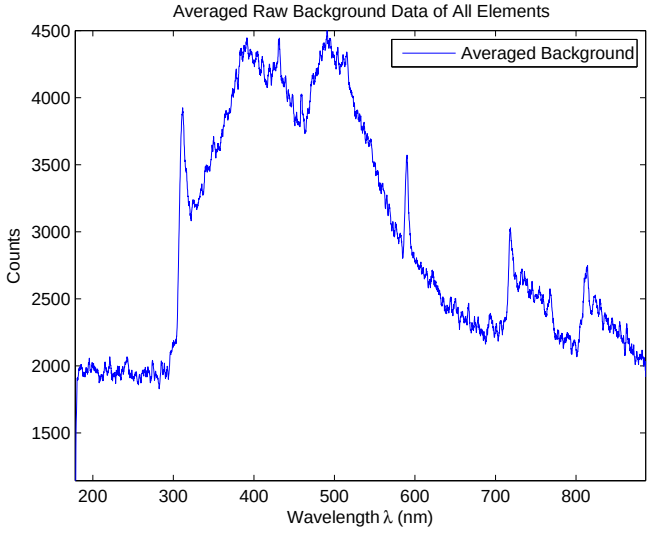


Figure 2.8: The averaged elemental background for Mark-1 Nebulizer Multiple Simultaneous Elemental Detection. This background is captured between each concentration tested for each element and is comprised of only the background light of the laboratory. The data presented here takes into account the Na data as well as it will be used for all elements.

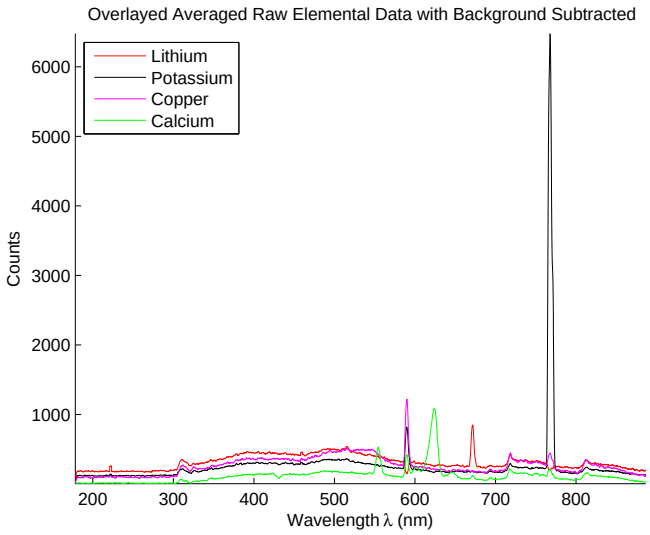


Figure 2.9: Overlayed Averaged Raw Elemental Data with Average Background of Figure 2.8 subtracted for Mark-1 Nebulizer Multiple Simultaneous Elemental Detection. Here it can be seen that Cu is not sufficiently strong enough at 500 μM to have a visible peak.

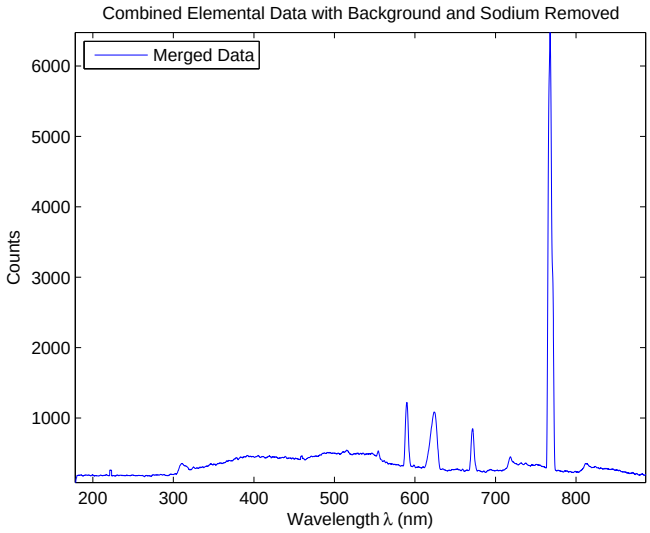


Figure 2.10: Merged Mark-1 Nebulizer elemental data to predict Multiple Simultaneous Elemental Detection capabilities. Each peak associated with Na (588.71 nm), Ca (624.68 nm), Li (671.55 nm), and K (768.03 nm) can easily be seen.

2.5 Next Steps

Two key questions were answered during this initial investigation: With the sensitivities reached it can be concluded that the basic flame from an off the shelf blowtorch is a suitable flame source; and The Ocean Optics USB4000 spectrometer is very capable and should easily be able to perform simultaneous analysis of multi-element samples. That being said there were some significant limitations with the Mark-1 Nebulizer Generation Three. More specifically, the device is limited by the nebulizer itself: it has a minimum sample volume required (100 mL), as well as a large sample burn rate of 1 mL/min. Another limitation is that the sample being injected into the flame is wet, and energy from the flame needs to dry, then heat, the sample prior to characteristic spectral emission. At 1 mL/min, and assuming a 25°C starting point, the flame must supply a minimum of 313.8 J/min of energy to heat the sample to 100°C, and another 2257 J/min to evaporate the water. The center of the propane flame only generates 14233 J/min. [42] This corresponds to a 18% decrease in flame temperature, or about 400 K, to an effective temperature of 1800 K. This reduces the energy available to excite the atoms and reduces emission intensity. Finally, the low temperature of the propane flame, even prior to the additional cooling by the water, is a significant limiting factor with regards to the detection and quantification of several less volatile elements such as Cu. [4, 18, 40]

Despite these limitations, the Mark 1 was able to achieve preliminary LODs on the order of micromolars which is significantly better than commercial offerings by such companies as Sherwood Scientific Ltd.. Also, it is anticipated that simultaneous multi-elemental detection is possible with the chosen hardware.

The next step in producing a LOC compatible FAES is going to require a significant architecture change to reduce the sample volumes required from 100 mL, and increase the sample injection efficiency. In addition, the sample injection used should further enhance sensitivity while ensuring reproducibility. It will also be beneficial to automate the process which will require writing software in a readily available cross-platform language such as Python, an open-source programming language.

Traditionally, nebulizers have been used with the Flame Emission Spectroscopy (FES), however, in 2011, Hanna et al. introduced the use of a Tungsten Coil (W-Coil) ElectroThermal Vaporizer (ETV). The idea is that the sample of interest is dried and vaporized prior to introduction into the thermal source. In addition, because the sample is vaporized, both liquid and solid samples may be used and the limitation on sample volume is only restricted by the surface tension of the sample itself to remain on the heating filament. This would indubitably allow for LOC sample volumes to be used, while increasing sample introduction efficiency.

This improved sample introduction methodology was deemed a higher priority than implementing a hotter flame to achieve Cu detection. This is due to the potential benefits of reducing sample volumes, increasing sample introduction efficiency, and allowing for the possibility to remove interferences. Once the system is fully developed and characterized the flame source can be changed.

Chapter 3

Mark-2: ETV Development

3.1 Introduction

In this chapter the development of the Tungsten Coil (W-Coil) ElectroThermal Vaporizer (ETV) is discussed. This includes the design of the housing and the power delivery as well as the control architecture. Furthermore, the methodology behind processing small liquid samples is discussed along with why an ETV is the optimal sample delivery method for the final end goal of a Lab on Chip (LOC) compatible Flame Atomic Emissions Spectroscopy (FAES) instrument. This instrument could be combined with other LOC instruments such as a Micro-Total-Analysis System (μTAS) for enhanced diagnostic ability.

3.2 What is an ElectroThermal Vaporizer?

ETVs, such as a Graphite Furnace (GF), are an established analytical method for the atomization of elements in analytical samples and has been employed for more than four decades. [40] It has been used to enable the multi-analysis of solid samples without any sample preparation. [43, 44] More importantly, however, the use of an ETV reduces the required sample volumes, reduces the possibility of sample contamination, and improves sensitivity. [44–46] The use of an ETV for Atomic Emission Spectroscopy (AES) does have some disadvantages including difficulties in calibration, lower precision, and, as described by Donati et al., the possibility for severe matrix interferences due to drastic temperature fluctuations in the gas phase. [10, 44, 47].

The ETV itself is a device which is heated by resistance according to select temperature programs. [27, 44] They are typically comprised of graphite (tubes or furnaces) [43, 45, 48], molybdenum (tubes) [49], tungsten (tubes or coils) [50–55], or tantalum. [56, 57] The sample is loaded into or onto the heating element for testing and can be either a liquid or solid. [44]

There are four stages involved in the use of an ETV: [44]

1. Sample Drying.
2. Sample Thermal Decomposition, which leads to pyrolysis of the sample matrix to potentially eliminate interferences. [58, 59]
3. Sample Evaporation and the transfer of the products of evaporation by a carrier gas into the thermal source.
4. Cleaning the ETV of any sample residues in order to prepare it for another sample.

The cleaning process is accomplished by heating the ETV to extremely high temperatures to ensure the elimination of memory effects and carry-over. [40] The carrier medium

for the vaporized sample is a gaseous blend of Argon (Ar) and Hydrogen (H), which inhibits the oxidation of the ETV while having no negative impact on the emission spectrum of the elements. [27, 40]

This thesis focuses primarily on the use of a W-Coil ETV for three major reasons over other materials: the ability for mass production (it is used in common lamp filaments), ease of use, and low power requirements. [25, 53] As with other materials, it has a high melting point, high purity, fast heating rate, chemical and physically stable, relative inertness at high temperature, and is durable. [53] Some have even used a W-Coil ETV as a W-Coil-AES without a separate thermal source. [26, 40, 60] A very important consideration is that W-Coil ETVs do not require refrigeration or water circulation cooling devices. [40]

3.3 Why an ElectroThermal Vaporizer is Needed

When considering the next iteration of sample injection, final application arrangements need to be taken into account. More specifically, it is desirable to have the FAES coupled to existing microfluidic separation techniques such as Capillary Electrophoresis (CE). CE operates through the use of applied high-voltage electrical fields across a separation channel. In addition, extremely low sample volumes on the order of microlitres is desired. An ETV is able to achieve such sample sizes. [26] Finally, with the evaporation of the sample taking place externally to the flame, the temperature of the flame is no longer cooled by the water in the sample, leading to hotter temperatures and higher sensitivies.

As discussed earlier, Electrothermal Vaporization has been used in the past with many different AES techniques resulting in higher sensitivities and lower Limit of Detection (LOD). [44, 46, 61, 62] Most recently, a group in 2011 coupled a W-Coil ETV to a FAES, also demonstrating sensitivity improvements over traditional nebulization [4]. They demonstrated that the ETV provides improvements to the traditional nebulizer FAES by a factor of 4 to 25. [4, 61] Another important consideration is that the typical sample volume when utilizing an ETV is between $2 \rightarrow 40$ μL; Reaching microlitre sized samples is critical for coupling to LOC devices. [26]

Referencing the results of Chapter 2, data demonstrating the potential sensitivity gains by using an ETV is tabulated in Table 3.1. This table compares the results of Hanna et al. along with those of the Mark 1 Generation 3 FAES, from Chapter 2, and a representative commercial counterpart, the Sherwood Scientific Ltd. Model 360. [4, 41, 61].

Although the LOD of the FAES from Hanna et al. is up to 40 times more sensitive than the Mark 1 Generation 3 FAES, the important consideration is the improvement realized

by changing the sample introduction module from a nebulizer to an ETV. Reportedly the flame temperature is a key factor in determining this LOD. [4, 18, 40] The FAES used by Hanna et al. utilizes a Air-Acetylene flame with a temperature of 2450 K in comparison to the propane flame used here at 2200 K. [4, 28] Furthermore the torch utilized by Hanna et al. had separate cylinders to provide direct control over the air-fuel mixture allowing for further tailoring of the temperature and flame profile versus the TS4000 used here, which has no control over gas and air flow rates.

Table 3.1: Limits of Detection Comparison between the State of the Art, the Mark 1 Generation 3 (utilizing the results of Chapter 2), and the Sherwood Scientific Ltd. Model 360 instruments.

Element	Wavelength (nm)	ETV-FAES [4] LOD (μM)	FAES LOD [4] (μM)	M1: Gen 3 FAES LOD (μM)	Flame Photometer [41] LOD (μM)
Li	670.8	0.0058	0.0256	3.0693	14.4071
K	766.5	0.0205	0.4322	1.0105	2.5577

3.4 Processing Small Liquid Samples

The two primary reasons behind opting for the ETV as the sample injection module is to reduce sample volume to LOC levels, and to increase sensitivity. As demonstrated by Hanna et al., increasing sensitivity is easily achieved. [4] In addition, it is possible to utilize volumes outside of the reported working range of $2 \rightarrow 40$ μL by taking advantage of the processing methodology. [26]

The W-Coil ETV used a *Osram Xenophot HLX 64633* halogen bulb (10-turn filament; 150 Watts (W), 15 Volts (V)) and was carefully extracted from its glass housing. [4, 61, 63] The tightly coiled filament allows for the surface tension of the sample to keep itself on the filament. As a result, there is no minimal sample volume required but rather a minimum number of elemental atoms within the sample. Maximum volumes, however, are limited but well over the working volumes of traditional LOC devices at or above 40 μL. [4, 64] This maximal volume limitation can be overcome through the use of repeated drying steps.

In order to vaporize the sample properly, the ETV undergoes a multi-step procedure beginning with a drying step. This serves the purpose of removing any water which would otherwise limit the temperature of the sample to 100 Celsius. Following this, the ETV is allowed to cool before being heated to extreme temperatures over 3000 K. [4]

On LOC devices, the total working volume is typically on the order of microlitres. In addition most micro-pumping solutions require solution in order to operate properly. [65] Therefore, in order to test the entire contents of the LOC device, it would be necessary to dilute the sample to transport it to the ETV. The drying step may be repeated as many times as necessary in order to allow for the entire volume of the LOC to be deposited on the filament allowing for the removal of the transport medium.

3.5 Development

There are three submodules which comprise a complete ETV sample injection module. These are the ETV itself, the power delivery system, and the control architecture. The ETV consists of the W-Coil along with the housing and is where the sample is vaporized. The power delivery system provides the variable power to the W-Coil in order to properly process the sample of interest. Finally, the control architecture is what will control the ETV as well as the optics module in order to facilitate higher levels of reproducibility.

3.5.1 Design and Fabrication of the ElectroThermal Vaporizer

The ETV consists of a W-Coil filament within an air-tight enclosure. Within this enclosure, a forming gas comprising of a mixture of H (10%) and Ar (90%) is passed over the filament. The sample of interest is deposited onto the tungsten filament in liquid form and a small current of approximately 3 Amperes (A) is passed through the filament until the sample is dry. Once dry, the current is set to zero, the filament is allowed to cool for 10 seconds, then a high current of 9 A is applied to vaporize the sample off the filament over the course of 10 seconds. The forming gas prevents oxidation of the tungsten while carrying the vaporized, and dry, sample into the flame to be analyzed. The filament is then allowed to cool before the next sample is loaded. A rough schematic representation of the ETV device as designed by Hanna et al. in 2011 can be seen in Figure 3.1 on page 31. [4]

A modified version of the ETV device has been designed and produced in-house and preliminary testing suggests it functions in a similar if not identical way as that built by Hanna et al. in 2011. [4] The primary modification is the addition of a heated housing to improve consistency between subsequent tests. This modified version was fabricated using all components which can be purchased in Home Depot, and the original design is presented in Figure 3.3 on Page 33. The lower half of the ETV can be unscrewed from the top for easy replacement of the W-Coil and cleaning, and is presented in Figure 3.3a. The fabricated ETV is presented in Figure 3.4.

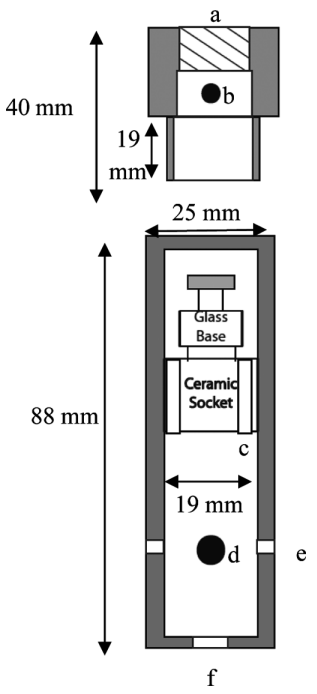
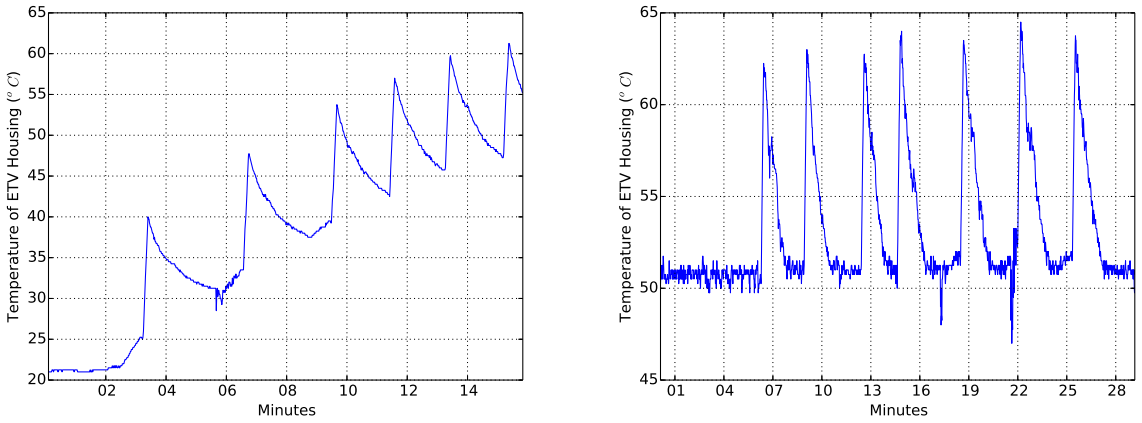


Figure 3.1: Illustration of the ETV device as presented by Hanna et al. 2011 [4]; (a) 1/2 NPT; (b) 8-32 tapped hole centered and in the line of sigh of coil; (c) opening in ceramic base for purge gas throughput; (d) two 1/16 NPT gas fittings (one front and one rear of the rod); (e) electrical line feedthrough; (f) 1/4-20 tapped hole for mounting.

The addition of the two *KHLV-0502* Omega Heaters to the ETV housing serves two important purposes:

1. Provides consistent starting conditions across multiple environmental conditions.
2. Reduces required cool-down time in order to expedite subsequent testing.

This is demonstrated by monitoring a thermocouple attached to the ETV housing over multiple thermal cycles. The housing temperature both before and after the addition of the Omega Heaters is demonstrated in Figure 3.2. Without the heaters, the housing temperature begins at room temperature and rapidly increases during sample vaporization by almost 20°C, cooling by less than 10°C over the course of three minutes. Subsequent tests continue to increase the housing temperature. It is observed in Figure 3.2a, that as the housing temperature increases, cooling rates also increase. Therefore, as presented in Figure 3.2b, with the housing heated to a little over 50°C, time to cool between tests is reduced to less than two minutes, and each test has identical starting conditions regardless of the testing order provided a minimum of two minutes between tests.



(a) (b)

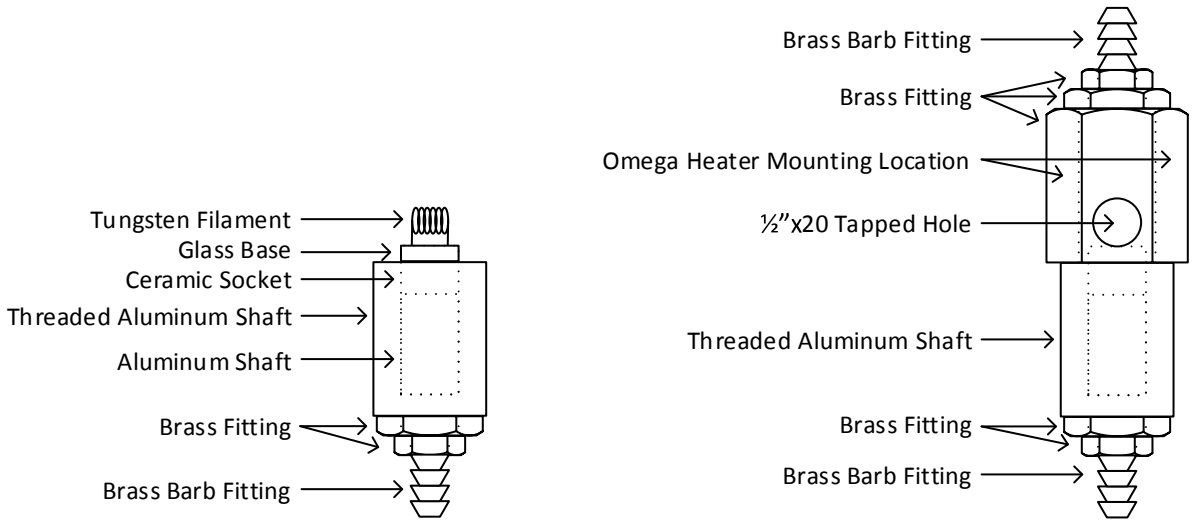
Figure 3.2: ETV Housing Temperature.

(a) Without Proportional-Integral-Derivative (PID) Heater: Each vaporization results in drastic temperature change of ETV housing leading to variable starting conditions for each sample test.

(b) With PID Heater: Heater effectively controls the ETV housing temperature such that start conditions for each test are identical. In addition the required time between subsequent tests is drastically reduced.

A *Praxair AR HY10P-K* forming gas (90%Ar-10%H) is used to shield the W-Coil from

oxidation while providing a transport medium for the vaporised sample. The chemical methodology behind how the forming gas protects the W-Coil is discussed by Hanna and Jones. [27]

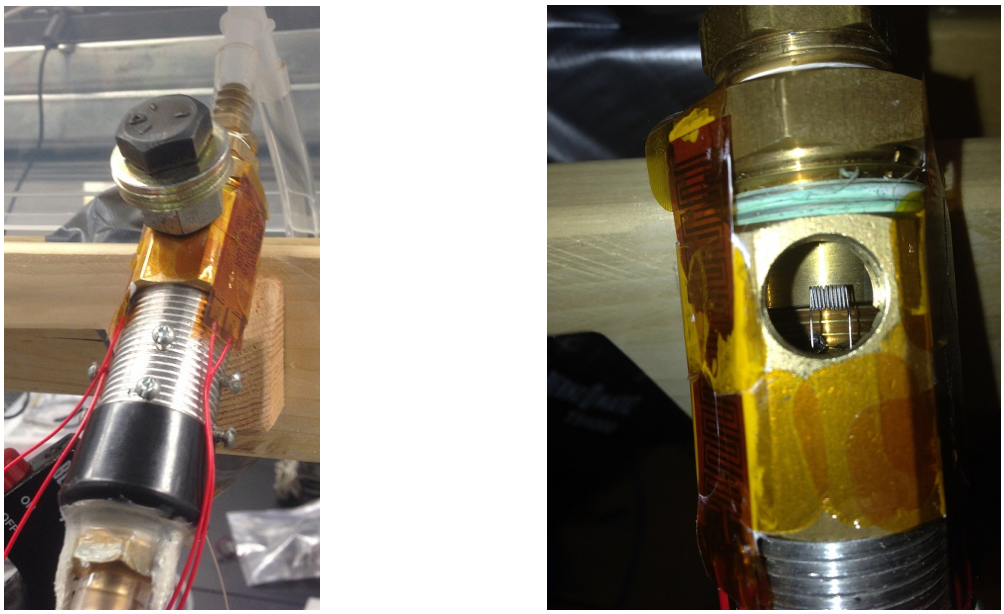


(a) (b)

Figure 3.3: Modified ETV Housing design using Home Depot components.
(a) Lower half of the housing which can be unscrewed from the top for easy access to the W-Coil. The interior aluminum shaft provides support for the ceramic socket which holes the W-Coil.
(b) Complete ETV housing. The forming gas flows from the bottom barb fitting through to the top. The tapped hole is plugged using a bolt and allows for easy loading of the W-Coil with a micropipette. The Omega Heaters are affixed to the ETV housing with thermal paste and Kapton tape.

3.5.2 Design of the Power Delivery System

In order to appropriately power the ETV, a computer controlled power supply capable of supplying up to 120 W is required. This requirement is based upon the vaporization parameters presented by Hanna. [63] In order to achieve this, a customized power supply was developed by modifying an original power supply design by Jones of EEVBlog.com. [66]



(a) (b)

Figure 3.4: Images of the Modified ETV Housing fabricated in the Student Machine Shop from Home Depot components.

(a) Complete ETV with Omega heaters attached. The screws in the threaded Al shaft support the inner Al shaft as presented in Figure 3.3. Kapton tape secures the heaters to the brass enclosure.

(b) Exposed W-Coil when the 1/2"-20 bolt is removed. The sample is loaded directly onto the coils by micropipette. The individual pieces of Kaptop tape used to secure the heaters are visible. White thermal grease under the heaters improves thermal contact.

This customized power supply is designed to be controlled via Serial Peripheral Interface (SPI) interface by either a LabJack, Arduino, or other form of microcontroller. It is capable of outputting a voltage between 0 and 12 V at a user-defined current of 0 to 10 A; corresponding to a maximum power output of 120 W. A block diagram depicting how the computer, LabJack and Power Supply could be connected and controlled is presented in Figure 3.5.

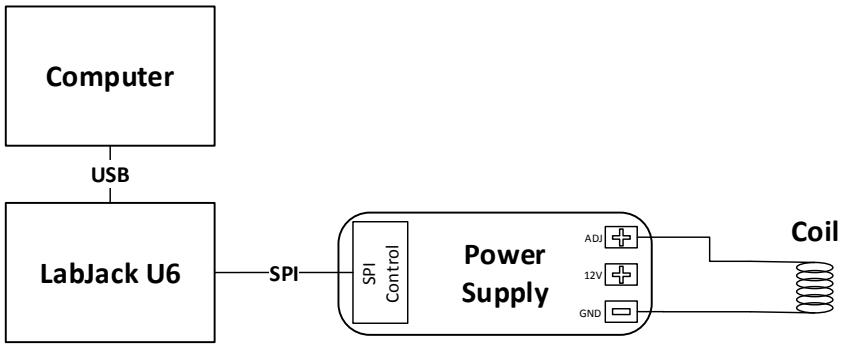


Figure 3.5: Block Diagram of the ETV Power Supply driving the ETV; Demonstrating LabJack U6 and Computer connectivity by means of Universal Serial Bus (USB) and SPI.

The computer interacts with the LabJack U6 through Python 2.7 over a USB cable. The LabJack then communicates with the power supply over SPI communication, which requires four wires: clock, slave data in, slave data out, and chip select. The power supply will send the LabJack the real-time current draw, and the LabJack will tell the power supply what voltage to maintain. There is no microprocessor on the power supply leaving the control entirely up to the LabJack. As the computer has full control over the LabJack, the LabJack effectively only relays the information and performs the necessary data conversion (SPI to USB). This maximizes the potential capabilities and uses of the power supply for not only driving an ETV, but also powering other applications requiring similar power capabilities.

This highly-capable power supply is fed an input voltage of 17 V at 26 A by two *Enermax EG365P-VE* Computer Advanced Technology eXtended (ATX) power supplies connected in series. These ATX supplies were modified by disconnecting the ground-plane of the circuit from the enclosure while maintaining an earth-ground connection to the enclosures themselves. This results in a safer working environment by drastically reducing the risk of electrical shock while ensuring that the ATX supplies are floating. That being

said, these power supplies are extremely high power and caution should always be taken when using them. The 17 V is produced by connecting the ground of ATX supply two to the 12 V line of ATX supply one. The customized power supply designed for the ETV is then connected to the +5 V of ATX supply two, and to the ground connection of ATX supply one.

In order to ensure that these switching ATX supplies produce smooth power, a load of approximately five Ohm (Ω) is placed on the +5 V rail of each. This was accomplished by connecting two 5 W 11 Ω resistors in parallel within the ATX enclosures themselves and powering them with the +5 V rail, prior to the rail exiting the enclosure. Preliminary testing of these modifications indicate power stability down to 1 mV prior to connecting them in series or to the customized ETV supply.

The ETV supply, as mentioned previously, is a modified supply of Jones design. The modified supply schematic can be seen in Figure 3.6 on page 37. This supply works by taking the input power from the ATX supplies in series and passing it through the equivalent of a 0.1 Ω resistor capable of handling 20 W of power dissipation. The voltage drop across these resistors corresponds to 1 V at a current of 10 A, and is monitored by the *MAX4080F High-Side Current Monitor*. This Integrated Circuit (IC) outputs the voltage drop with a gain of five (5 V corresponds to 1 V drop across the 0.1 Ω resistor, which in turn corresponds to a current of 10 A). This 0-5 V output voltage is passed onto both a comparator, which compares the current passing through the resistors to the desired output current which is set via a *MCP4922* Digital to Analog Converter (DAC) IC, and a *MCP3202* Analog to Digital Converter (ADC). If the current is too high it will activate a current-limit Light-Emitting Diode (LED) as well as begin to lower the output voltage of the ETV supply by dropping the voltage on the Set pins on the linear voltage regulators (*LT3083*). The output voltage itself is set through the use of an OpAmp. The OpAmp outputs the desired voltage (0 to 12 V) to the Set pins on the LT3083, which in turn will set the output voltage to match the set pins. Each LT3083 can handle a maximum of 3 A, thus paralleling four of the regulators allows up to 12 A to pass through at the desired set voltage.

The power supply was designed using CadSoft EAGLE v6 freeware version thereby limiting the board design to 4 x 3.2 inches. In order to facilitate this, the power supply was separated into two modules. The first module is the complete power supply but with only a single LT3083. The second module is a board containing the supporting hardware for an additional LT3083. Therefore the complete power supply is made up of a single control module and multiple secondary modules to reach the desired current handling capability. The two modules as designed in EAGLE can be seen in Appendix F.2. All of the boards designed in the Freeware version of EAGLE are two layers.

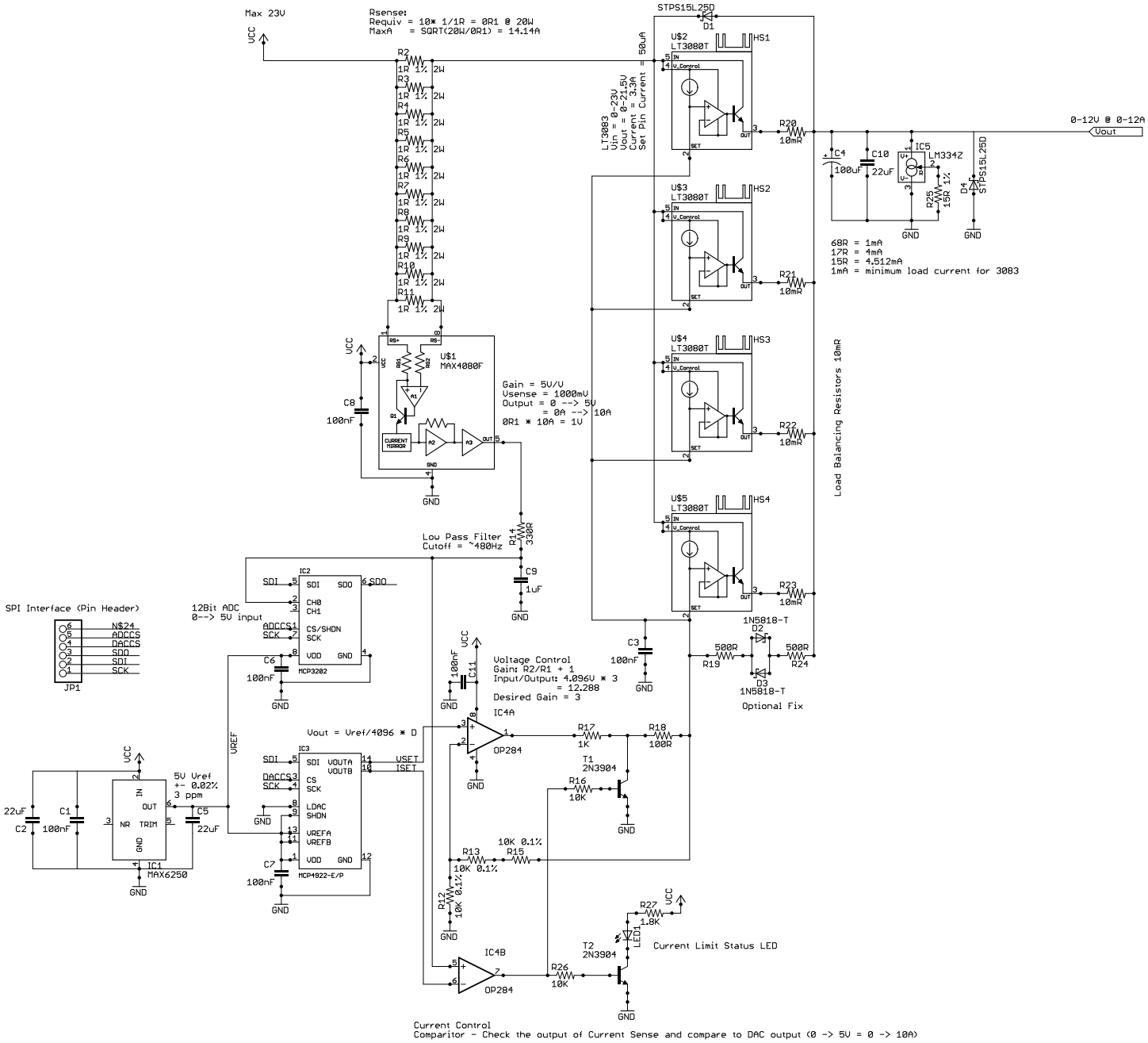
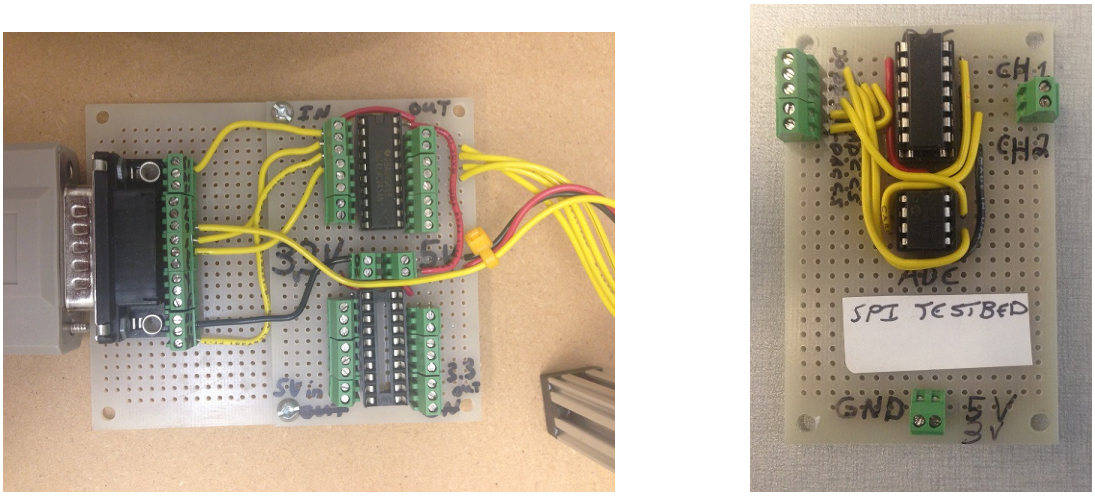


Figure 3.6: Power Supply Schematic, Modified from Original Version designed by David L. Jones on EEVBlog, prior to Splitting into Two Modules to meet Freeware EAGLE requirements. [66]

3.5.3 Control Architecture

The ETV power supply is controlled over the SPI communication protocol. More specifically, the power supply was designed to operate over 5 V logic in order to be Arduino compatible. Initially, however, the ETV was to be controlled through the use of a *LabJack U6*. In order to prototype this SPI interface with the LabJack, an SPI development board was designed (Figure 3.7b) in addition to a voltage-level translator board (Figure 3.7a), which allows for the voltage-level transition of 3.3 V to 5 V logic.



(a) (b)

Figure 3.7: SPI Test Circuits for LabJack development of ETV control architecture.

(a) Voltage level translator using a 74LVC245 to convert 3.3 V to 5 V logic. DB15 connector connects to the LabJack U6 over a DB15 cable. The end board provides the 5 V power source to eliminate the potential for power spikes. The voltage level translator board also provides a common ground connection eliminating ground loops.

(b) SPI Testbed comprised of a MCP4922 DAC and a MCP3202 ADC. The DAC channels 0 and 1 are connected to the ADC channels 0 and 1 as well as providing a terminal to connect a multimeter. The board may be powered by a 5 V or 3.3 V supply in order to test the voltage level translator separately from the SPI communication on the LabJack U6.

The initial test code for the SPI interface to the SPI Testbed for the LabJack U6 was written in Python and is presented in Appendix B.3. This code was simplistic in that it asks the user for the supply voltage (3.3 V or 5 V digital logic) and the voltage levels to set the DAC channels to. It then reads the ADC and displays the voltage. This, when

combined with an external multimeter, allows for the testing of the SPI protocol during both the write and read stages independently. Once fully established, the voltage level translator as imaged in Figure 3.7a was inserted into the communications path to establish any impact it has.

Once the SPI communication protocol on the LabJack U6 was fully established it became necessary to modify the SPI testing code to allow for the full control of the Power Supply as described earlier. This modified code requires the ability to voltage map the 0-5 V of the DAC to the 0-15 V output of the power supply in addition to being able to monitor, and set, the current in real time. In order to accomplish this a Graphical User Interface (GUI) was developed to provide a front-end the user may be able to interact with, along with shifting the functions into threads. This modified, threaded, version of the SPI test code is presented in Appendix B.4.

Threading is used to provide parallelism to the program. In this case, constantly reading and displaying the value from the ADC. This allows the user to utilize the GUI and enter values without being interrupted by this background thread. Caution was taken to ensure a single thread interacts with the LabJack at any given time by implementation of a semaphore. A great discussion, along with background information on the use of threads in Python, can be found in such texts as *Core Python Applications Programming.* [67]

Chapter 4

FAES and μTAS Hardware Development

4.1 Introduction

An infrastructure was developed, based upon a series of modules, for Lab on Chip (LOC) operation that would facilitate both general LOC prototyping and the interface of LOC-Flame Atomic Emissions Spectroscopy (FAES). This infrastructure was developed with the idea that the LOC application would consist of performing elemental analysis on the output of electrophoretic protein tests performed on a Micro-Total-Analysis System (μTAS). Therefore the designed boards were to serve a dual purpose: Enable the design and fabrication of the FAES, and enable the control and integration of a μTAS capable of performing diagnostics such as Restriction Fragment Length Polymorphism (RFLP).

The design and implementation of these electronics are necessary in order to meet the three objectives set out at the beginning of this research as there is virtually zero open-source infrastructure readily available to perform the tasks at hand. More importantly, the ability to automate the entire process is essential for reliability and reproducibility. In order to reduce complexity, a series of modules were designed. These modules comprised of a Fluidics Board, which performs high-voltage switching to enable Capillary Electrophoresis (CE); LT3092 Board, which allows for fine current control of LEDs; Photodiode Board, which provides optical detection of extremely weak signals; Solenoid Board, which performs the toggling of pneumatic solenoids for air and vacuum control; ElectroThermal Vaporizer (ETV) Power Supply Board One, which provides the control interface and connections to additional power supply modules for the FAES; ETV Power Supply Board Two, which increases the current handling ability of ETV Power Supply Board One; An Interface Board, which allows for the integration and control of all boards simultaneously from a single LabJack U6; And a Proportional-Integral-Derivative (PID) Heater Controller board, which provides the heater control for the ETV housing.

The goal of this chapter is to discuss the different modular circuits in detail along with the design characteristics.

4.2 Design Considerations and Preliminary Testing

4.2.1 Power Harmonics

One of the key problems identified by Gordon H. Hall, in June of 2013, when performing optical detection of low strength signals was power harmonics in the captured signal. These harmonics were easily removed through the use of Fast Fourier Transform (FFT)

processing. However, due to the gain required to accurately see the desired signal, these power harmonics were clipping as they hit the rails on the Analog to Digital Converter (ADC); and thereby resulted in false signals remaining after FFT processing. These false signals were of the same magnitude as the desired signals. It therefore became paramount to identify the source of power harmonics and eliminate them.

In order to identify the source of the power harmonics, a basic setup was put together: A laptop was connected through Universal Serial Bus (USB) to a LabJack U6; The LabJack U6 pulls power from the computer's USB port and subsequently distributed this power to the photodiode board. The photodiode board was located in an enclosed, dark, container in order to isolate the electrical noise from any light emissive sources. A block diagram of this setup can be seen in Figure 4.1.

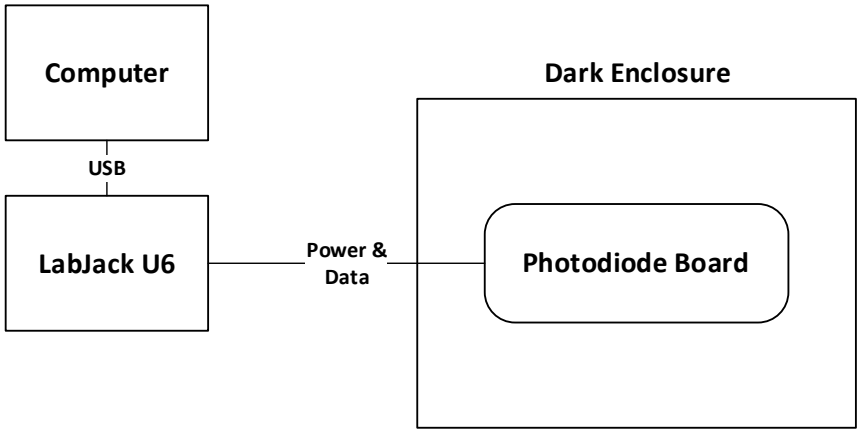


Figure 4.1: Block Diagram of Power Harmonics Test Setup.

The laptop being used was disconnected from the Alternating Current (AC) wall outlet, causing all of the electronics to run off the Direct Current (DC) battery; unfortunately this caused the noise to increase over having the laptop powered by a wall adapter. Following this, a USB cable was spliced in order to allow for a Laboratory Power Supply to be used to fully power the photodiode board; this also increased the noise over using the power supplied by the laptop's USB port. At this point, all steps available at the time had been taken. The use of a separate external battery was not considered to be sustainable due to the power draw of the LabJack.

It then became necessary to engineer a solution to this problem. Gordon H. Hall needed a power supply able to eliminate power harmonics and deliver a smooth and consistent DC

power to sensitive electronics responsible for signal collection. A tightly integrated low-noise power supply was needed as all other options led to substantial noise.

The requirements for this power supply were that it be able to:

- Provide 0-15 Volts (V) at up to 500 mAmperes (A).
- Have zero AC component (no power harmonics).
- Be low cost and easy to mass produce.

Due to different AC/DC Converters and power supplies delivering different and unknown qualities of power, a rigorous design needed to be put in place to ensure complete AC noise rejection. There were three potential options:

1. Buy a power supply able to meet these requirements
2. Use a linear regulator in order smooth out the power delivered
3. Use a combination of switching and linear regulators to smooth out the power delivered

The first option of purchasing a power supply is not realistic in scaling or creating multiples; The cost would be prohibitive and detracts from the field deployability of the final system.

The second option could work, however, the power requirements of the TLV272 for the full-swing of the op-amp requires up to 15 V. This means that the initial AC/DC Converter used would need to be 20-24 V and then down regulated. Upon initial testing, a singular linear regulator was not sufficiently able to remove all power harmonics. Furthermore, the high voltage of the AC/DC Converter would preclude its use in other applications.

The third, and final, option was to utilize the combination of a switching and linear regulator. Because power can fluctuate, a boost switching regulator circuit was developed. This circuit operates in the MHz range and boosts the input voltage to an output of 20 V using a *LT3467* boost regulator. This 20 V output is independent of the input voltage (2.4-16 V) and completely removes the initial power harmonics from the original AC/DC Converter due to its own switching speed. This 20 V signal with 1.3 MHz harmonics is then linearly regulated down to 15 V using a *LM2940*. The noise rejection from the linear regulator, according to the data sheet, is upwards of 90 dB on MHz frequencies.

As a result of implementing option 3, the power delivered to the TLV272 was regulated in such a way that no detectable power harmonics remained. A stable 15 V was continuously supplied, able to deliver up to 800 mA. This was measured using a known load and multimeter. Furthermore, because this power supply was designed using discreet components it is readily reproduced.

The final power circuit is separated onto two different boards: the Photodiode Board, and the Interface Board. Both of these are discussed in detail in Sections 4.3.3 and 4.3.5 respectively.

4.2.2 Unreliable Communication and Noise Pickup on Digital Lines

It is well known that electronics are susceptible to electronic noise which is why Electro-Magnetic Compatibility (EMC) is so important. There are many texts on the subject such as *Introduction to Electromagnetic Compatibility*, which talks about this subject as well as discusses the EMC regulations towards commercial and residential products. [68]

With this in mind, one of the major problems identified by Dr. Chris J. Backhouse on earlier μTAS produced by his research group, such as the Tricorder ToolKit (TTK) and Micralyne Inc. Microfluidic ToolKit (μTK) systems, was significant noise being injected into the Serial Peripheral Interface (SPI) rails when the high-voltage modules were activated. [69] As a result, the stability and signal integrity were always questionable. Therefore, as one of the common practices in a μTAS is high-voltage switching, it was highly suspect that the new system would fall prey to the same issue of the older systems: the high voltage would inject too much noise on the SPI rail for it to be reliable.

In order to resolve this potential curtailing issue, an engineering solution was required. The requirements for this new communications methodology were that it:

- Be backwards compatible with the current electrical modules and systems.
- Be robust.
- Able to perform voltage level conversions (3$\leftrightarrow$5 V).
- Be low cost and easy to mass produce.

As a result, the following options were derived:

1. Presume the issue would not be present on the system.
2. Utilize Low Voltage Differential Signaling (LVDS)
3. Utilize signal buffers on the SPI rails
4. Utilize a combination of options 2 and 3.

The first option was not viable. It was highly suspect that the large electromagnetic noise generated within the system would cause signal degradation leading to significant errors on the SPI rail (as per TTK and μTK).

The second option on its own would not be able to adequately perform the voltage level translations. The *65LVDS1* LVDS Drivers are able to convert 5 V signals down to 3

V, however, the *65LVDT2* LVDS Receivers are unable to boost the incoming signal back to 5 V. As a result any boards requiring 5 V logic would not work. However, the noise rejection ability of the LVDS is substantial. There are many technical reports and books on the subject of how differential signalling works to reject noise and improve EMC, such as [70–72]. In general, LVDS works by splitting each digital signal into a differential pair; this is performed by the driver. The differential pair is then merged back into a single signal by the receiver. This differential pair uses voltages differences between the two lines; as noise is injected, it is injected onto both lines equally. The receiver takes the difference between the differential pair, eliminating the injected noise, and determines if the original signal was a 1 or 0. This is important for EMC compatibility. [72]

The third option of using *74LVC2G34* Signal Buffers on the SPI rails proved to be the best initial choice. The buffers are able to perform the logic level conversions and operate at such a frequency that the 20 MHz maximum SPI clock frequency would remain unaffected. More importantly, these buffers are easily added and implemented onto circuit boards and are backwards compatible. They work well at noise rejection in that they do not propagate noise, but they are not as capable at noise rejection as the LVDS. These devices read a weak digital signal and determine if the original signal was a 1 or a 0; they then output the original signal as if it were the originating device thereby boosting, or buffering, the signal. They are unidirectional and typically placed at the signal source as they usually are much stronger sources or sinks to help eliminate noise.

The fourth option would incorporate the buffers to perform the voltage level translations and the LVDS to transport the signal between the electrical boards. The received signal would then be buffered by a second buffer to bring the voltage level up to the appropriate level for the board.

As a result of implementing option 3, the communication within the system has never had any issues. The SPI signal is reliable and electromagnetic noise has had no effect thus far. If it does prove to be an issue, the LVDS can be added to further improve the noise rejection. Hirad Raisi, a co-op student, wrote his work report on the new communication protocol and working together it was fully characterized. The SPI test circuit presented in Figure 3.7b was used to perform this characterization. A script would, using SPI, continuously change the output value of the Digital to Analog Converter (DAC) channels and confirm proper adjustment by reading the values on the two ADC channels until stopped by the user. The script would increment a counter to indicate the number of faults detected. Over the course of testing, no faults were detected. Option 4 would result in a maximum SPI rate of 16.97 MHz over a 1 meter cable. This was calculated following Equation 4.1, provided by Texas Instruments, which provides a means of estimating the maximum cable frequency given the cable length and component delay times. [71] Specified

maximum delay times for the Buffer, LVDS Driver, and LVDS Receiver used in the LVDS Test Circuits presented in Appendix F.9 were used for the calculation to provide a worst case (lowest speed) scenario. [73, 74]

$$\text{Max Frequency (Hz)} = \left((4 \times B_D) + (2 \times D_D) + (2 \times R_D) + \frac{2 \times x}{70\% \times 3 \times 10^8} \right)^{-1} \quad (4.1)$$

Where B_D is the Buffer Delay in seconds; D_D is the Driver Delay in seconds; R_D is the Receiver Delay in seconds; and x is the distance of the cable in meters.

The LVDS boards along with the schematic diagram are presented in Appendix F.9. These boards are strictly logic based and do not require any external programming or shielding. The signals were tested using a function generation along with an analog Tektronix oscilloscope. ElectroMagnetic (EM) noise was generated around the circuit through the use of Cell Phones making and receiving calls and text messages. Furthermore, an additional design for over-voltage protection was also put together but has not been tested. This design is primarily for the protection against ElectroStatic Discharge (ESD) on connections made; ensuring that the power rails are not artificially elevated; and to enable continuous 12 V short to any pin without causing damage. The 12 V condition is to allow for a short to the Interface Board power supply while eliminating the risk of damage to the board.

Unfortunately, the ETV Power Supply was designed and developed prior to investigating methods to improve noise rejection. As such these improvements were not implemented. As will be discussed in Chapter 5, this proves to be a problem which could have been avoided.

4.2.3 Unable to Run All Necessary Boards off One LabJack U6 Simultaneously

An issue identified by the research group was that there were more than one electronic board which needed to be run by the LabJack U6 at any given time. Unfortunately, the boards all use a DB15 connector, and the LabJack U6 only has one DB15 output. A block diagram presentation of this limitation is presented in Figure 4.2. It is conceivable that the DB37 connector on a single LabJack could also be used, however, there are a grand total of 20 digital I/O pins available between both the DB15 and DB37 connectors. [75] This would also require complex customized cables in order to map these digital pins to

the daughter boards. Unfortunately, all of the daughter boards require a total of 25 digital I/O pins for proper functionality; and this leaves no room for future expansion. Therefore the LabJack on its own is unable to perform the task at hand.

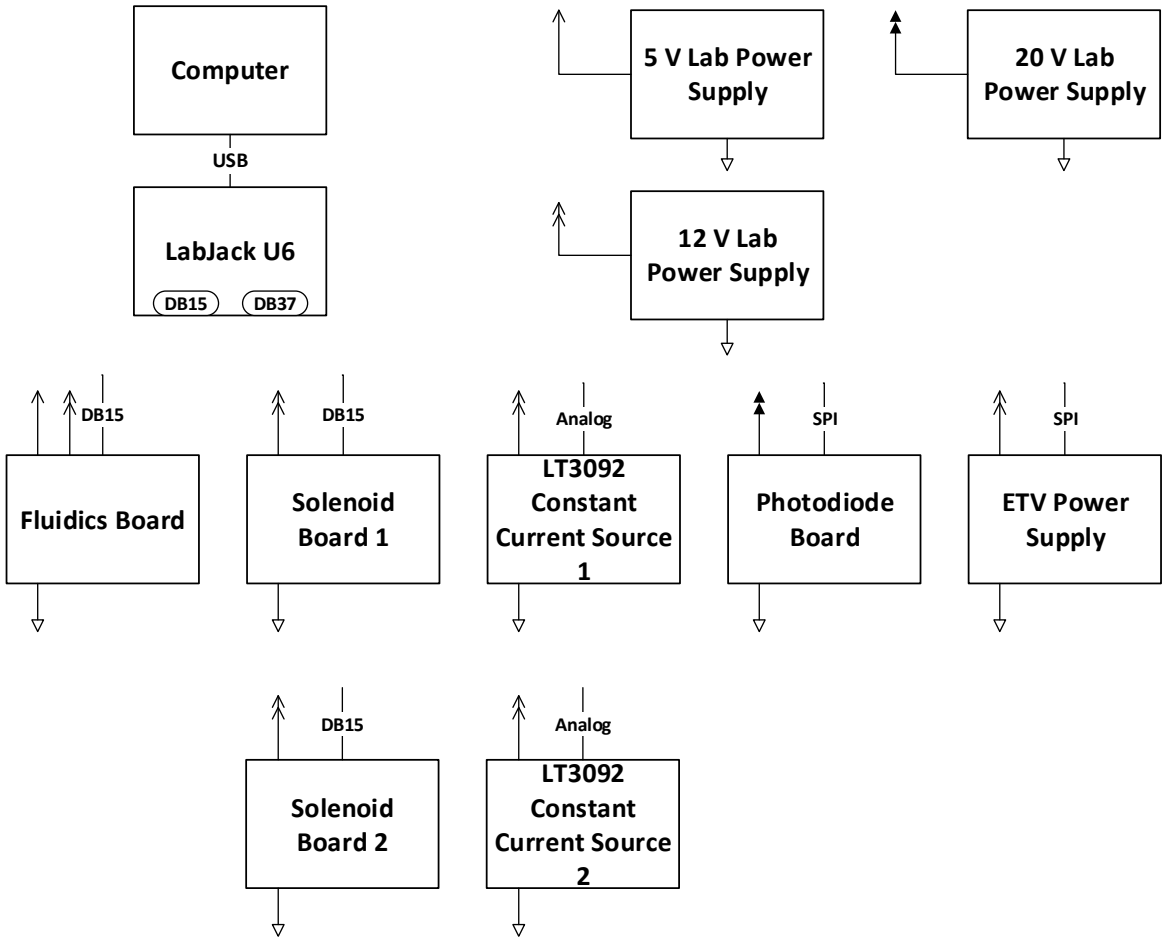


Figure 4.2: Block Diagram of Connection Limitation with One LabJack U6 and Multiple Power Supplies.

In order to resolve this issue, it was attempted to use multiple LabJacks on the same computer simultaneously. Unfortunately, the LabJack Python implementation, at the time of this writing, does not support this capability. An attempt to synchronize multiple computers in order to run multiple LabJacks proved too difficult and would not be easy

to implement when more than two boards (computers and LabJacks) are required in the system.

It became necessary to engineer a solution which would enable all of the developed boards thus far to be used simultaneously with one LabJack. This meant the designing of an interface board. A block diagram of the desired interface board and capability is presented in Figure 4.3.

There were two other options:

1. Attempt to re-design all of the boards onto a single PCB with the free-ware version of CadSoft EAGLE
2. Find a way to sync multiple computers and multiple LabJacks

These other options were not possible due to the complexity of synchronizing multiple computers and due to the limitations of the free-ware software. Furthermore, additional functionality would require a complete re-design of the PCBs with option 1.

Moving forward the most logical solution was the design and fabrication of an interface board. The requirements for this board were as follows:

- Powered from single power supply (ie AC/DC Converter or Advanced Technology eXtended (ATX))
- Utilize the DB37 of the LabJack leaving the DB15 port available for future use
- Able to provide 3.3V output at 500 mA
- Able to provide 5V output at 500 mA
- Able to provide 12V output at 1 A
- Able to provide 20V output at 500 mA (see Section 4.2.1)
- Able to connect all boards at the same time (High Voltage, Solenoid Board A, Solenoid Board B, LT3092 Breakout A, LT3092 Breakout B, Photodiode, and ETV Power Supply)
- Possible to expand outputs easily

In order to simplify the design, the schematic was segregated into modules and an extension board was designed. A series of *74HC595*s were used in order to increase the digital outputs to toggle the relays and solenoids. Extra outputs were routed to a 40 pin IDE header for ease of expansion.

A series of buffers were implemented, as per Section 4.2.2, to clean up digital communication and increase the reliability of the SPI signals.

A DAC circuit was implemented with a jumper in order to toggle the output to be from either 0→5 V or 0→1 V (both using the full 4096 steps). The output of 0→1 V is necessary to adequately provide digital control of the LT3092 breakout boards.

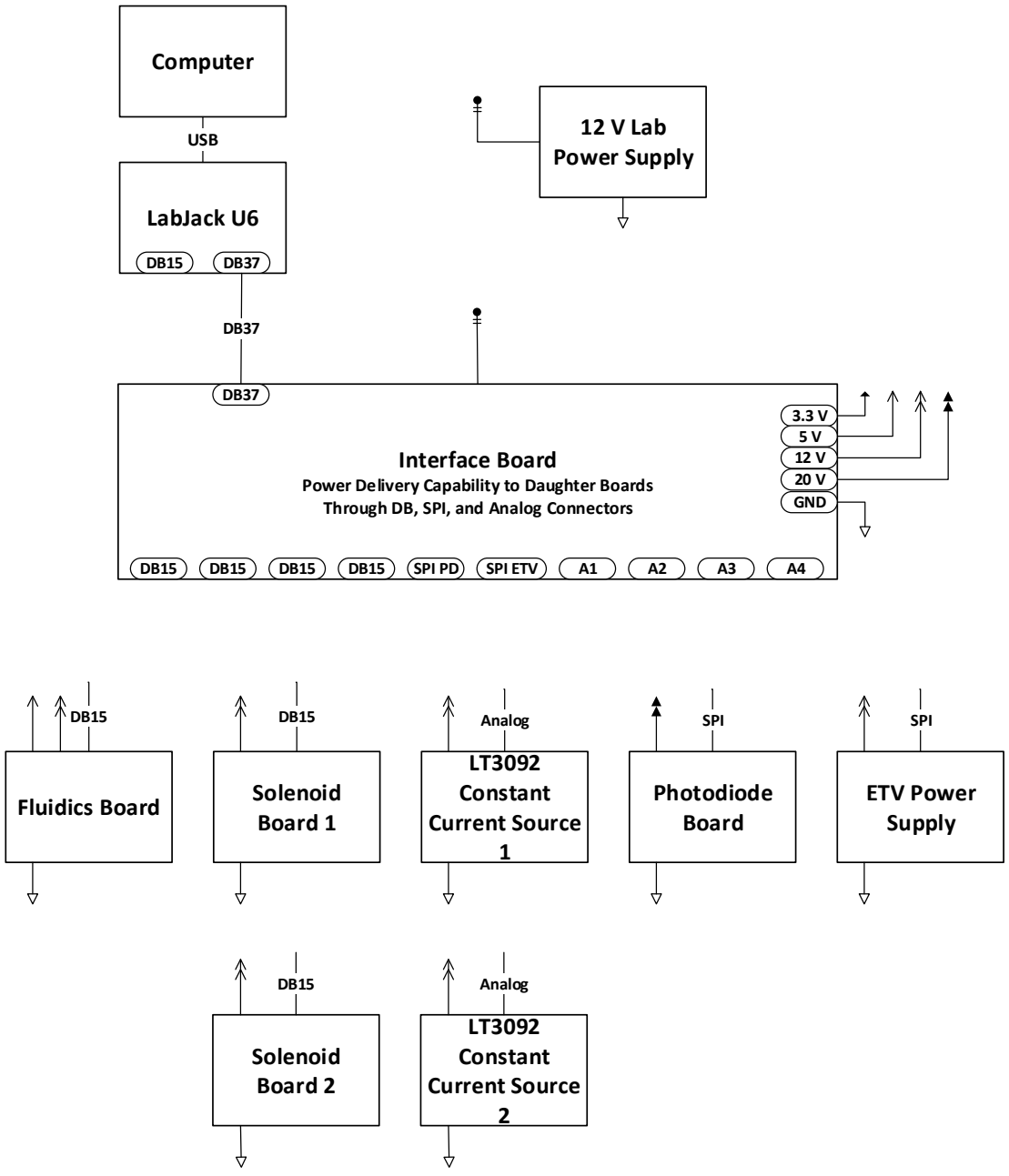


Figure 4.3: Block Diagram of Interface Board Connection Capabilities and On-Board Power Supplies.

A power regulating circuit and power distribution was developed in order to allow the interface board to directly power all daughter boards connected to it. The output current of each power rail is ≥500 mA and linearly regulated (with the exception of the 20 V rail). This enabled the use of a single AC/DC Converter to power all of the electronics connected to the interface board.

As a result, the interface board has become the heart of all systems. It is powered from a single 12 V AC/DC Converter, uses only the DB37 port of the LabJack U6, provides 4 power rails to connected daughter boards, and is able to connect the majority of all boards at the same time. An SPI bit-bang issue from the LabJack precludes the connection of the ETV boards through the interface board. This is explored in Chapter 5.

4.2.4 CMOS Compatible Photodiode Board

The original photodiode board was extremely sensitive to noise pickup, especially on the power lines as discussed in Section 4.2.1. Gordon H. Hall identified several key issues; primarily that the base noise was significantly larger than the signals being detected. The board was not able to utilize the gain required to see the desired concentration levels. Ideally this photodiode board would be used as the optical detection circuitry in the FAES with optical filters to select the wavelength of interest.

There were several options in order to resolve this:

1. Salvage optical detection from μTK
2. Salvage optical detection from TTK
3. Design a new photodiode board

Options 1 and 2 were not viable as at the time, fully dismantling these systems was not appropriate. Furthermore, it is highly desirable to have the final solution Complementary Metal-Oxide Semiconductor (CMOS) compatible and unfortunately the optical detection on the μTK and TTK utilize traditional confocal optics and lasers. This need for CMOS compatibility is due to the desire to eventually move the RFLP system to an integrated CMOS-LOC device. Therefore option 3, a new photodiode board, was to be engineered.

Therefore the requirements on the photodiode board were as follows:

- High level of power noise rejection
- Local collection of signal
- Ultra low noise to enable $1 \rightarrow 4 \times 10^{10}$ gain
- Able to connect to the interface board
- High speed

In order to reach this $1 \rightarrow 4 \times 10^{10}$ level of gain reliably, the power regulation as per Section 4.2.1 needed to be implemented. The interface board (Section 4.2.3) has the 20 V supply voltage readily available. In addition, collection of the signal needed to be performed locally in order to reduce noise pickup on the analog signal.

The photodiode board schematic was then designed modularly to easily facilitate the modifications and improvements which may be required in future revisions. This is presented in Appendix F.3.

Due to the need of collecting the signal locally, an *MCP3202* ADC was implemented directly on the photodiode board. This ADC has two channels which allows for monitoring of the first and second stage of the OpAmp amplification. This monitoring of both stages enables the tuning of gain on the second stage to reach the desired baseline signal levels (mostly important when the photodiode is swamped).

As the TLV272 Rail-to-Rail OpAmp swings between 0→15 V, a passive resistive divider network was needed in order to convert the 0→15 V to 0→3.3 V (the ADC operates on 3.3 V). Furthermore, a *MAX6043* high-end voltage reference was added in order to power the ADC directly on the photodiode board in order to eliminate any residual power noise. This voltage reference was powered by the *LM2940* 15 V linear regulator in order to further reject any power harmonics.

Buffers were added to boost signal integrity as per Section 4.2.2. The board layout was performed in such a way to as to isolate the gain stages eliminating stray capacitance. Multi-meter testing probe locations were added to allow for an oscilloscope or multimeter to be used to easily monitor the power rails.

As a result of the engineering design: the photodiode board has a high level of noise rejection; the signal is collected locally; achievable gain is around 3.2×10^{10}; it is able to connect to the interface board; and high speed communication through SPI is now standard.

Gordon H. Hall performed the noise calculations; and the board is approaching the theoretical noise floor of the design. These calculations do not account for stray capacitance, power supply noise, or any other non-ideality. A discussion on this Photodiode Board, as well as its noise characteristics, is provided in Section 4.3.3.

4.3 Working Modules

With the exception of the ETV Power Supply Boards One and Two, which are discussed in Section 3.5.2, each module is described in detail here. Schematics and board layouts

may be seen in Appendix F.

4.3.1 Fluidics Board

The Fluidics Board provides switchable high-voltage for the use of CE on the microfluidic chips used within the RFLP system. In addition, because of its design, the high-voltage module may be removed and relays changed in order to toggle wall-voltage for use in extremely high power applications such as in the FAES.

The Matsusada TS-0.3P High Voltage DC-DC converter module provides 0→300 V at up to 0.5 mA on the output from a DAC input of 1.2→10 V following Equation 4.2. This equation was developed by mapping the output voltage of the TS-0.1P with a multimeter and applying linear regression analysis in Excel. In order to provide the 1.2→10 V, a TLV272 OpAmp was utilized to double the maximum 5 V input voltage from the LabJack U6 DAC (the Interface Board maps the V_{in} to DAC0).

$$V_o = 32.978 \times (V_{in} - 1.2) + 16.549 \tag{4.2}$$

Two of the four *Coto 9104-12-11* 12 V relays (V_{OUT} 3 and V_{OUT} 4) toggle between high-voltage and floating while the other two relays (GND_{OUT} 1 and GND_{OUT} 2) toggle between ground and floating. This allows for the appropriate control over CE on microfluidic chips. These relays are controlled through four *MIC4422* MOSFET drivers, which in turn are controlled from digital IO on either the LabJack, or the Interface Board. The current draw on these digital IO lines is typically less than 1 mA while the MOSFET driver itself can toggle the output at up to 9 A. In addition, the digital signals for the MOSFET drivers can be either 3.3 V or 5 V without any hardware changes.

As the board is designed to be used either standalone on the LabJack or with the other modules via the Interface Board, protection diodes were added. These *1N5817* diodes provide up to 20 V protection and will allow up to 1 A forward current. This serves the purpose of also acting as a fuse on the board by burning out if more than 1 A of current is being drawn through either the Matsusadas or the MICs. In addition, if a different power level is desired on the board, the diodes prevent back-driving the LabJack or Interface Board.

All of the outputs support indefinite short-circuit conditions without any damage to the components. However, a negative supply voltage will cause significant damage as no polarization diode is present on the board. Furthermore, the power utilized by the board is dependent upon the state of the relays, which can draw up to 100 mA each.

In the FAES, the Fluidics Board is modified to utilize three *JTN1AS-PA-F-DC5V* high-power relays to toggle the electrical connection between: the ETV Power Supply and the ETV; the 12 V supply of the second ATX power supply and the ETV; and a 5 V rail and the BoArduino controlling the Ocean Optics trigger pulses. In this case, the voltages being toggled are directly connected to the relays rather than though the FR4 board as is typically the case with the RFLP version of the Fluidics Board as described above.

The board supports direct connection to a LabJack in addition to the Interface Board through the DB15 connector. When used in conjunction with an Interface Board, no other external connections are required as power (12 V) and the DAC signal (V_{in}) is delivered over the DB15 cable. When used with the LabJack, CIO0→CIO3 are used to control the MOSFET drivers and external power and analog signals are required. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.7 and F.8 respectively in Appendix F. Figure 4.4 demonstrates a populated Fluidics Board as used by the RFLP systems. The dimension of the Fluidic Board measures 2.75 inches wide by 2.25 inches tall; the mounting holes are spaced 2.4 and 1.95 inches apart.

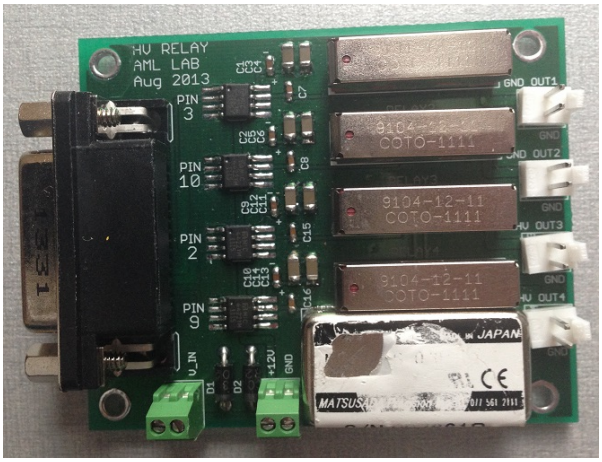


Figure 4.4: Top side of populated Fluidics Board showing the MIC4422 MOSFET drivers; Coto 9104 relays; and Matsusada TS-0.3P high voltage supply. The keyed two-pin headers provide connections to the output of the relays while the terminal blocks allow for external power and DAC connections. The OpAmp to double the DAC voltage to make use of the full range of the TS-0.3P is on the bottom of the board. The DB15 connector allows for connection directly to either the Interface Board (power and DAC provided through the cable) or the LabJack U6 (external power and DAC required).

4.3.2 LT3092 Constant Current Source Board

The LT3092 Constant Current Source Board is a breakout board for the 200 mA *LT3092* Programmable Current Source. Two terminal blocks provide the input and output to the LT3092; when the R_{SET} resistor is unpopulated the LT3092 may be controlled via an external DAC output of 0.2→1 V. The DAC control follows Equation 4.3 while the control with R_{SET} follows Equation 4.4. The LT3092 can work with supply voltages of 1.2→40 V and sports a <0.001%/V line regulation. This makes this breakout board ideal for controlling LEDs and other sensitive low-current applications. It is able to control the current between 20 mA and 200 mA in a static fashion, and between 20 mA and 100 mA dynamically through the external analog voltage. For larger currents it is possible to parallel many boards.

$$I = (99.5\ mA \times V_{in}) + 0.5\ mA \tag{4.3}$$

$$I = \frac{R_{SET}}{R_{OUT}} \times 10\ \mu A \tag{4.4}$$

Ideally, this board is used to power the illuminating LEDs for the RFLP system, while the Photodiode Board monitors the fluorescent output of the dyes used in the microfluidic chips. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.9 and F.10 respectively in Appendix F. The board measures 0.95 inches wide by 0.85 inches tall, and has two mounting holes located in opposite corners. These mounting holes are designed for 4-40 screws and are located 0.15 inches from the edges.

All of the outputs support indefinite short-circuit conditions without any damage to the components. However, there are no protection diodes and a negative supply voltage can cause damage as a result. The power utilized by the board is dependent upon the desired current flow and the attached device, which can draw up to 200 mA.

4.3.3 Photodiode Board

The Photodiode (PD) Board is used to read the fluorescent signal from a microfluidic separation with high sensitivity over SPI in order to drastically improve detection capability and noise rejection. The two-stage transimpedance amplification through the *TLV272* OpAmp provides extremely high gains which are also susceptible to noise on the power

rails. As such a linear regulator (*LM2940*) and voltage reference (*MAX6043*) has been added to the board directly to reduce the potential for noise injection. In accordance with Section 4.2.2, *SN74LVC2G34* buffers have also been added to the SPI communication rails to further improve noise rejection and dropped bits.

The *MCP3202* ADC has been incorporated to allow for reading of the outputs of both stages of the amplification at the source with 12-bit accuracy, and is powered by a *MAX6043* 3.3 V voltage reference. The values can then be transmitted digitally thereby removing any potential analog noise injection. This ADC addition does not need to be used as analog outputs remain through the use of terminal blocks to support its use directly with the LabJack. As the LabJack ADC supports a maximum input of 15 V, the board supply voltage is regulated to 15 V to make maximum use of this ADC range. In order to effectively use the 3.3 V MCP3202 ADC, a passive voltage divider maps the maximum 15 V of the OpAmp output to 3.3 V. This eliminates any possible damage to the ADCs on both the LabJack and the PD board itself.

Furthermore, in order to eliminate the risk of damage to the LabJack ADC and the MCP3202 ADC, the OpAmp is supplied from a single supply. Should a negative supply be incorporated, additional components would be required to ensure the safety of the ADCs which could inject additional noise. The first stage is set to a constant gain of 100×10^6, selected along with an appropriate capacitor, to facilitate a 1 Hz bandwidth and reduce the required second stage gain. The second stage gain is then tuned according to the application.

The outputs of the PD board, as well as the gains, can be tuned and set according to both the jumper settings in Table F.2 and Equations 4.5a and 4.5b.

$$Gain_{Stage\ 1} = F_R 1 \tag{4.5a}$$

$$Gain_{Stage\ 2} = 1 + \frac{F\_R2}{F\_RIN2} \tag{4.5b}$$

Note that the current from the photodiode is converted into a voltage and amplified by the transimpedance amplifier (Stage 1). The stability of the first stage is controlled through capacitor F_C. The second stage is a typical feedback amplifier. A large ground plane ensures consistent ground levels, however, isolation was performed around the photodiode itself and the amplification stages to eliminate stray capacitance.

The SPI communication to the board can only be used in conjunction with an Interface Board unless modifications are made. Power is delivered over the DB9 cable, however,

external power may be supplied. Protection diodes (*1N5817*) provide reverse polarity protection and prevents back-driving of the Interface Board. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.3 and F.4 respectively in Appendix F.

The PD board requires a voltage supply of 17→26 V. If a lower voltage is supplied the circuit will breakdown, however, no component will be damaged. If the power supplied exceeds 26 V components will suffer irreversible damage. Ideally 20 V is supplied.

The ADC utilizes SPI communication though the DB9 connector. The LabJack does not directly have a DB9 connector, however, the interface board or other adapter can be used to facilitate this communication.

The PD board measures 2.7 inches wide by 1.45 inches tall, and has four mounting holes, one in each corner. These mounting holes are spaced 2.4 and 1.15 inches apart, and can be seen in Figure 4.5. Note that the photodiode location has been centered and is set far enough from the edge to support the use of a terminal block.

As previously stated, the Photodiode board operates the transimpedance amplifier from a single 0→15 V supply rather than dual supply of say -8 V → +8 V. Despite the OpAmp being rated for rail-to-rail, there is always a risk that an offset voltage will prevent the OpAmp from swinging all the way down to 0 V. However, in this application, with the gain levels used, there is always a substantial baseline present of about 0.82 V. As such, any offset limiting the ability of the OpAmp to reach 0 V does not negatively impact performance.

This baseline was measured, along with the noise, in a similar manner to the measurements performed for measuring the power harmonics. The block diagram presented in Figure 4.1 shows the test setup. In this way the raw noise of the circuit is measured and the baseline of 0.82 V determined. It should be noted that if the optical signal entering the photodiode is of sufficient strength, given the gain levels employed, the OpAmp will saturate.

Gordon H. Hall performed a noise calculation on the board, using methodologies provided by Graeme, and compared it to the measured noise. [76] The measured noise on the raw output data from the Photodiode board is between 1 mV and 6 mV, while the theoretical noise calculation places the design at 6.94 mV. The gain used for measuring and calculations was 3.17×10^{10}. Previous customized photodiode boards by others within the laboratory, which used a lower gain of 10^9, had noise on the order of 30 mV. More importantly, the vetted optical detection system on the TTK, for which this photodiode board is a direct competitor, utilizes significantly more expensive components, and has noise levels on the order of 2-3 mV with gains of 10^9. [69] If the gain on the TTK were increased

to 3.17×10^{10}, noise levels would be on the order of 63-95 mV. This supports the need for this new photodiode board as well as emphasizing its significantly better performance characteristics over alternatives.

These substantial noise improvements stem from three main factors: The power harmonics are virtually non-existent; the ADC is located as close to the OpAmp as possible and is powered by a reference class regulator; and the use of a transimpedance amplifier has a low bandwidth on the order of hertz. These combined result in an exceptional photodiode board.

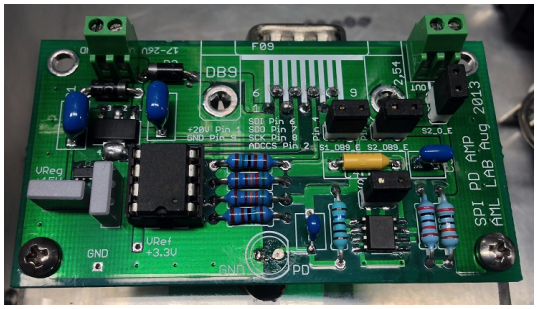


Figure 4.5: Top side of populated Photodiode Board showing the power regulation in the top left; ADC and passive voltage conversion in the bottom left; TLV272 OpAmp and Transimpedance amplifier for the photodiode in the bottom right. The terminal blocks allow for external power supply (left) and external ADC (right). The jumpers allow for the enabling or disabling of the second stage amplification as well as passing the output of each stage through the DB9 connector and terminal block on right. The DB9 connector must be connected to an Interface Board and provides SPI communication as well as power to the board.

All of the outputs support indefinite short-circuit conditions without any damage to the components. The power utilized by the board is dependent upon the state of the photodiode and ADC, however, it should remain below 100 mA.

4.3.4 Solenoid Board

The Solenoid board is used to control up to six solenoid valves which in turn can supply air or vacuum to microfluidic pumps. The exact output voltage on the outputs is the same as what is supplied to the Solenoid board itself. When used in conjunction with an Interface Board Extension, no external power is required as the power is provided through the DB15

connector itself. A protection diode (*1N5817*) prevents back-driving the Interface Board Extension or the LabJack pins if an external power source is supplied.

In order to control the output states, the board utilizes six MOSFET drivers (*MIC4422*) at 5 V in the same way as the Fluidics Board. The drivers themselves are controlled using six digital signals over a DB15 connector (EIO0→EIO5 on the LabJack). The current draw on these digital IO lines is typically less than 1 mA while the MOSFET driver itself can toggle the output at up to 9 A. In addition, the digital signals for the MOSFET drivers can be either 3.3 V or 5 V without any hardware changes.

The Solenoid board measures 1.7 inches wide by 2.7 inches tall, and has four mounting holes, one in each corner. These mounting holes are spaced 1.4 and 2.4 inches apart. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.5 and F.6 respectively in Appendix F. Figure 4.6 demonstrates a populated Solenoid Board.

All of the outputs support indefinite short-circuit conditions without any damage to the components. One must be careful, however, to not supply negative voltages to the board as it will cause damages. The power utilized by the board is dependent upon the number of attached solenoid valves, which can draw up to 100 mA each.

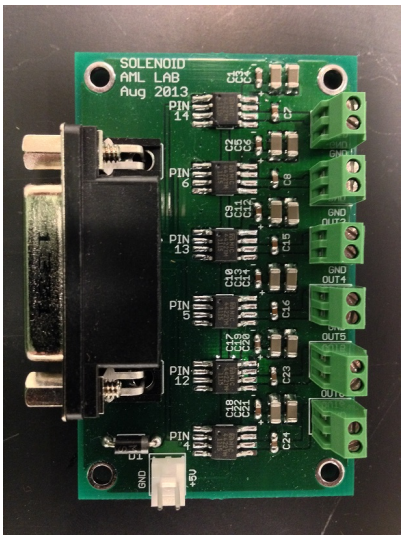


Figure 4.6: Top side of populated Solenoid Board showing the MIC4422s; The keyed two-pin header allows for external power to be supplied when the board is not connected to the Interface Board Extension; The terminal blocks connect to the MIC4422s for control of solenoids or other relays.

4.3.5 Interface Board and Extension

The Interface Board is the answer to the LabJacks inability to connect all of the modules required for the RFLP and FAES systems, which ideally will be combined into a single instrument. The board connects to the DB37 port on the LabJack U6 which contains 11 digital IO pins, two DAC outputs, 14 ADC inputs, and 5 ground connections. All of the digital pins, ground pins, and two of the analog pins are used as defined in Table F.20.

The Interface Board is powered by a 12 V DC, 2.5 A, power supply with a 2.1 mm jack. The input jack is fully-rectified using six *SR202* 20 V 2 A diodes. This protects the board against reverse polarity connections and ensures proper power is distributed to onboard components. This 12 V rail is then down regulated with: a *LT1963* 3.3 V linear regulator to generate a 1.5 A 3.3 V power rail; and a *LM2940* 5 V linear regulator to generate a 1 A 5 V rail. In addition, the 12 V rail is boost-regulated by a *LT3467* to 20 V at 800 mA. These four power rails provide all of the power necessary to the Interface Board and all connected modules.

To expand the number of digital outputs on the LabJack, three HC595 shift registers were utilized. The first shift register, *IC8*, generates all of the SPI Chip Selects for the Integrated Circuits (ICs) on the interface board and connected modules. The remaining two shift registers, *IC6* and *IC7* are daisy-chained to provide the 3.3 V signals to control the *MIC4422*s on the attached Solenoid and Fluidic boards (total of 16 outputs). Additional shift registers may be added to expand upon the number of digital outputs available. *IC8* was not included in the daisy-chain of *IC6* and *IC7* in order to increase the speed with which a SPI chip select could be toggled. Therefore the two HC595 communication rails, being considered SPI by the LabJack, are bit-banged by the LabJackPython software through the DB37 connector. [75] Unfortunately, the SPI on the LabJack is only supported in software and is not a hardware function. [75] In order to ensure that the digital communications to the HC595s are correct, the signals from the LabJack are processed through *SN74LVC2* Buffers operating at 3.3 V.

The devices which require 5 V logic are enabled through the use of a *TXB0108* voltage level translator, a bi-directional IC. These 5 V devices include the *MCP4922* DAC on the Interface Board for controlling the LT3092 Constant Current Source boards; the *MCP4922* DAC on the Extension Board; and the ETV Power Supply Board.

The control for two ETV Power Supply boards was introduced in order to enable a second ETV power supply board to be used as a power supply for Polymerase Chain Reaction (PCR) heaters on future LOC applications.

The Interface Board supports a single Fluidics Board (DB15) and a single Photodiode

Board (DB9) along with two DAC outputs. The Extension board connects through a keyed 40-pin IDE cable which provides the SPI and Digital Outputs necessary to control two Solenoid Boards and two ETV Power Supplies. In addition, the 3.3, 5, 12, and 20 V rails are supplied to the extension board. A DAC pass-through from the LabJack is provided to the Fluidics Board to control the Mastsusada HV module. In addition, an ADC pass-through is provided to the Photodiode Board to allow for streaming mode of the desired stage output.

The Extension was designed due to the limitation of the freeware version of CadSoft Eagle. The Interface Board measures 3.99 inches wide by 3.2 inches tall, and has four mounting holes. These mounting holes are spaced 2.4 and 1.95 inches apart, and can be seen in Figure F.15 centered around the bottom left hand corner of the board. Both the Interface Board and the Interface Board Extension have four mounting holes aligned with one another. These mounting holes are grounded to further eliminated ground loops and improve ground connection between the two boards when metallic standoffs are used. The Extension board measures 3.99 inches wide by 2.25 inches tall.

The DB15 Pin Assignments (Fluidics Board) for the Interface Board are listed in Table F.21. The DB9 Pin Assignment (Photodiode Board) for the Interface Board are listed in Table F.22. The 40-Pin IDE Pin Assignments (Interface Board Extension) are listed in Table F.23. The Extension Board X1 and X2 connectors (Solenoids and ETVs) pin assignments are listed in Tables F.24 and F.25 respectively.

4.3.6 ETV PID Heater Controller

The ETV PID Heater Controller Board is controlled via a BoArduino. The board allows for the control of up to 20 Watts (W), 36 V, or 4 A (whichever comes first). In the FAES, the board controls 12 V to the dual Omega Heaters on the ETV housing as located in Figure 3.3b. The board works by having the BoArduino control a *MCP4822* DAC which sends out a 0→4.094 V signal corresponding to the current (A) sunk through the *D44H8* transistor.

The two power rails are isolated from each other but maintain a common ground. The 5 V rail is used for the BoArduino, DAC, and is down-regulated with a *LT1963* to 3.3 V for the *MAX31855* thermocouple monitor. The 12 V rail is used to power the *LT1006CN* OpAmp and the *KHLV-0502* Omega Heaters. Protection diodes (*1N5817* and *SR206*) prevent reverse polarity.

To monitor the current through the Omega heaters, a set of 10 parallel resistors equivalent to a 1 Ohm (Ω) 20 W resistor is used. Therefore, one A corresponds to a voltage

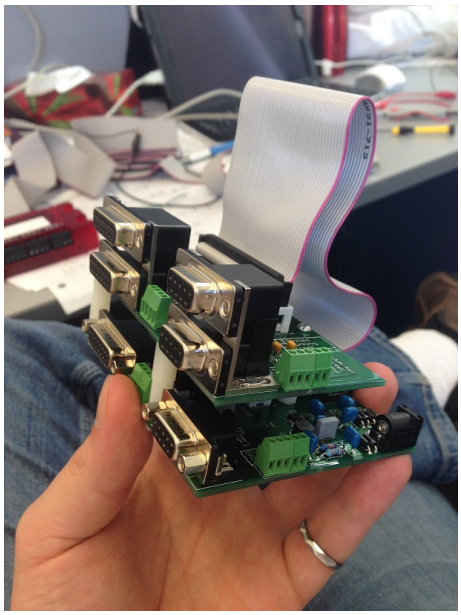


Figure 4.7: The Interface Board (Bottom) and Extension Module (Top) connected with an IDE cable to provide additional capabilities. The top left connector allows for a Solenoid Board to be connected; The middle left connector allows for a second Solenoid Board to be connected; The bottom left connector allows for a fluidics board to be connected; The top right connector allows for an ETV power supply to be connected; The middle right connector allows for a second ETV power supply to be connected; the bottom right connector allows for a Photodiode Board to be connected. The terminal jacks between the DB connectors allow for up to four LT3092 Boards to be connected. The terminal blocks along the right edge allow for the power rails to be accessed by additional boards and modules (3.3; 5; 12; and 20 V). The DB37 port along the back of the Interface Board (not imaged) connects to the LabJack U6 DB37 port through a DB37 cable.

drop of one V. The LT1006CN compares this voltage drop to the desired drop set by the DAC and adjusts the gate voltage on the D44H8 transistor accordingly to match.

The K-Type thermocouple is monitored by the *MAX31855* Cold-Junction Compensated Thermocouple-to-Digital Converter. This IC operates over SPI communications. To simplify the voltage level translations, the chip select and clock signals are tied high at 3.3 V and a *PMEG3015* diode protects the board from the 5 V signals from the BoArduino. However, when the BoArduino drives those pins low, the diode will conduct and the pins go low effectively translating the 0→5 V of the BoArduino to 0→3.3 V. The data out at 3.3 V can be read directly by the BoArduino without any conversion.

The schematic and board layouts were developed in CadSoft Eagle v6 Freeware and are presented in Figures F.17 and F.18 respectively in Appendix F. Figure 4.8 demonstrates the prototype version utilized for the control of the *KHLV-0502* Omega heaters.

The operating code on the BoArduino is presented in Appendix D.

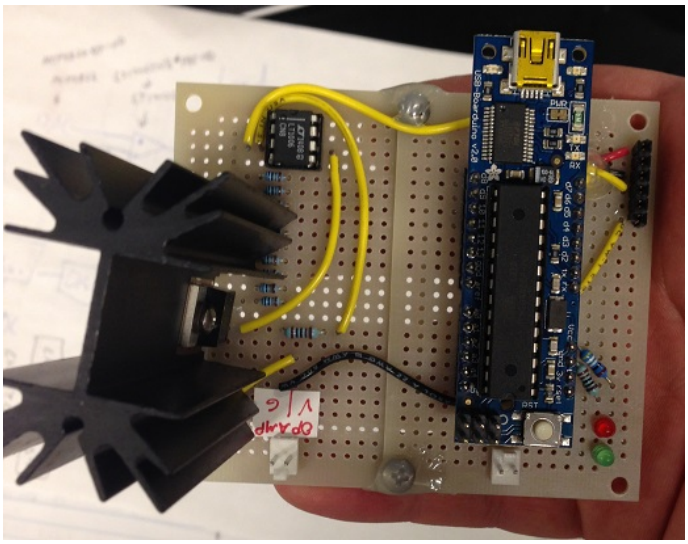


Figure 4.8: Top side of populated PID Heater Board showing the BoArduino; connection to the MAX31855 K-Type Thermocouple Breakout on the top right; indicator LEDs on the bottom right; Keyed two-pin headers for external power (left for heater, right for BoArduino); D44H8 Transistor with heatsink on the left; and the OpAmp on the top left. Under the BoArduino is the DAC, and under the heatsink are the parallel resistors.

4.4 Software

In order to control the interface board and all daughter boards, as per Sections 4.2.3 and 4.2.4, software needed to be developed. As most modules are able to interact directly with the LabJack through the DB15 port, simple toggling of the GPIO is sufficient for proper control. Here the focus is on the Interface Board.

4.4.1 LabJack Initialization and Basic Control

In order to interface with the LabJack, a Python 2.7 program was written to utilize the LabJackPython package. In order to initialize the LabJack and enable control the following Python code is required.

```
import u6                 #Load the LabJackPython Library for the U6

d = u6.U6()               #Defines d as the u6 device
d.configU6()              #Write the Local ID and read some hardware info
d.getCalibrationData()    #Get the slope and offset for AIN and DACs, and
                          #Calibration data
d.configTimerClock()      #Writes and reads the timer clok configuration
```

SPI Control

The SPI communication from the LabJack is software driven by the LabJackPython module for Python 2.7. [75] As a result, all SPI communication is bit-banged and can therefore operate on any digital pin. [75] The following code describes how to utilize the SPI communication protocol through Python. A working example is presented in Appendix B.3. Note that the pin numbers assigned to each pin are added to a value of 6000 to correspond to an appropriate register. Table 4.1 maps the registers to the pin names and location.

```
d.spi(SPIBytes,                 #The data to send
      AutoCS=True,              #Automatically drive the CS line
      DisableDirConfig = False, #Enable two-way communication
      SPIMode = 'A',            #Can be A, B, C, or D
      SPIClockFactor = 0,       #Sets the frequency of the SPI clock
      CSPINNum = 0,             #Chip select pin
      CLKPinNum = 1,            #Clock Pin
      MISOPinNum = 2,           #Slave Data Out pin
      MOSIPinNum = 3)           #Slave Data In pin

#This above command can be shortenned into the following:
d.spi(SPIBytes, True, False, "A", 0, 0, 1, 2, 3)
```

GPIO, DAC, and ADC Control

The GPIO is very easy to utilize on the LabJack. The following code example explains its use. A map of the Registers and the corresponding pin location and name is provided in Table 4.1.

```
#Write a value to the GPIO or DAC
d.writeRegister(Register,Value) #Value can be 0 or 1 for digital
                                #             0-5 for DAC

#Read a value from the GPIO or AIN
d.readRegister(Register)
```

UART Control

The typical Rx Tx pins for Universal Asynchronous Receiver Transmitter (UART) communication begin on FIO0 and FIO1. To shift the communication to the DB15 port, a pin offset must be used. In the following code the UART is enabled and the communication is shifted by 8 pins to EIO0 and EIO1 (pins 4 and 12 on the DB15 connector). A full working example of the UART on the DB15 connector is presented in Appendix B.5.

Table 4.1: LabJack U6 GPIO Names, Registers, and Pins

Name	Register	Connector	Pin
FIO0	6000	DB37	6
FIO1	6001	DB37	24
FIO2	6002	DB37	5
FIO3	6003	DB37	23
FIO4	6004	DB37	4
FIO5	6005	DB37	22
FIO6	6006	DB37	3
FIO7	6007	DB37	21
CIO0	6016	DB37, DB15	25, 9
CIO1	6017	DB37, DB15	7, 2
CIO2	6018	DB37, DB15	26, 10
CIO3	6019	DB15	3
EIO0	6008	DB15	4
EIO1	6009	DB15	12
EIO2	6010	DB15	5
EIO3	6011	DB15	13
EIO4	6012	DB15	6
EIO5	6013	DB15	14
EIO6	6014	DB15	7
EIO7	6015	DB15	15
DAC0	5000	DB37	11
DAC1	5002	DB37	29
AIN0	0	DB37	37
AIN1	1	DB37	18
AIN2	2	DB37	36

```
d.asynchConfig(DesiredBaud=9600)         #Configures the U6 UART for
                                         #asychronous communication

#Write and reads the current IO configuration
d.configIO(NumberTimersEnabled = 0,      #Number of timers to enable
           TimerCounterPinOffset = 8,    #Where should the UART start?
           EnableUART = True)            #Enable UART

#Send the message (30 character maximum)
d.asynchTX(txSend)

#Read the input buffer (30 character maximum)
rxResponse = d.asynchRX()['AsynchBytes']
```

4.4.2 Interface Board Software

The complete software control for the Interface Board is several thousands of lines of code long, however, using the building blocks presented in Section 4.4.1 the code can be redeveloped.

The code is so large as each individual circuit board needs to have its own operating code. A series of flags are used by the main control program to determine what each module is doing in order to prevent SPI overlap between the modules as the LabJack can only perform a single SPI communication at time. [75] The addition of a Graphical User Interface (GUI), which changes depending upon the use of system, also adds considerable length to the code. This amalgamation of code is necessary to ensure that all of the daughter boards operate as desired. The use of threading, as mentioned in Section 3.5.3, adds to ease of use to the end user.

The *HC595*s on the board are communicated through the SPI protocol. In order to shift out the data, however, an additional toggling of the Chip Select pin is required. This changes the chip select line of the shift registers to zero thereby enabling the data to be recognized (the shift register works on the rising edge of the data, not the falling edge).

As the shift registers are what provide the chip selects for the connected SPI boards, such as the Photodiode Board, the SPI command must be preceded and followed by the command to toggle the shift registers. Therefore the command structure would resemble the following.

```
# Enable the chip select (low) to begin SPI
Toggle2 = [0b11110111]
d.spi(Toggle2, SPIMode = "D", CSPINNum = 3, CLKPinNum = 4, MOSIPinNum = 5)
d.writeRegister(6003,0)

# Sending the required SPI message to read from PD ADC channel 1
ADCB = d.spi([0b00000001, 0b11100000, 0b11111111],
             True, False, "A", 0,
             CSPINNum = 99, CLKPinNum = 17, MOSIPinNum = 18, MISOPinNum = 16)

# Deactivating the chip select (high) to complete SPI
Toggle2 = [0b11111111]
d.spi(Toggle2, SPIMode = "D", CSPINNum = 3, CLKPinNum = 4, MOSIPinNum = 5)
d.writeRegister(6003,0)
```

For the Solenoid and Fluidics board, only the shift registers need to be toggled. For the DAC output required for the HV Module on the Fluidics Board, the simple writeRegister command may be used.

Upon further development, it was discovered that the bit-banging of the SPI communication protocol was resulting in erroneous data being transmitted. This had the negative effect of causing dropped or added bits being registered and was cause for significant error in the connected devices. This will be explored more in depth in Chapter 5.

4.5 Stand-Alone Environmental Monitoring

A large component of debugging and identifying problem areas of an analytical instrument is knowing the environmental conditions. Within the Applied Miniaturisation Laboratory (AML), there have been multiple cases of heating, cooling, and humidity control failures resulting in damage to multiple devices within the laboratory, including an ATX Power Supply. Furthermore, it is unknown what effect different humidities may have on the sensitivity of the ETV-FAES.

In order to monitor, in real-time, the environmental conditions, a *Weather Station* was developed. This weather station monitors the temperature, humidity, and pressure of the surrounding environment and uploads this information every minute to an online server.

The heart of the Weather Station is a *Beaglebone Black* running with the *Angstrom* Operating System. A python script running on the Beaglebone was developed and is presented in Appendix E.3. This code works by sending a 3.3 V trigger pulse to a BoArduino

through a *TXB0108* Voltage Level Translator. Upon receipt of the trigger the BoArduino sends the weather information to the Beaglebone over UART communication.

The received message resembles the following:
*Humidity: 38.70 % Temperature: 25.30 *C Pressure (kPa): 96.7988 kPa.*

The BoArduino code is presented in Appendix D.6. Every three seconds the code probes the attached sensors (*DHT22* Humidity and Temperature Sensor, and a *MPL115A2* Barometric Pressure Sensor) for the latest environmental information. When a Trigger signal is received, the BoArduino sends out the most recent environmental information. Therefore the largest delay in reading the information to when it is published online is going to be approximately three seconds. In this case, a Xively server was utilized for its ease of use and incorporation with Python 2.7.

The Weather Station was built on protoboard and is presented in Figure 4.9. It is setup to automatically begin the Python script presented in Appendix E.3 upon power-up. Therefore if any error occurs, or if there is a power failure, re-connecting the power will fully restore the weather station. The Weather Station is connected to the internet through a Ethernet cable, and is powered by a 5 V Wall-Wart.

The humidity, according to the datasheet for the sensor, is accurate to ±5%; and the temperature is accurate to ±2 °C. [77] Barometric pressure, according to the datasheet for the sensor is accurate to ±1 kPa. [78] This was checked against a calibrated hygrometer from Fisher Scientific, and it was confirmed that the temperature is indeed accurate, however, the humidity was consistently 5→10 % higher than the hygrometer. At either extreme (0% and 100%), the humidity values converged to within a few percent. The barometric pressure was not checked against another device.

Within the first week of use, the Weather Station reported drastic environmental condition fluctuations. These recordings are presented in Figure 4.10. This data, specifically, was used to report a safety issue to the Safety Office at the University of Waterloo due to condensing humidity causing indoor fog, short circuits, and slippery floors.

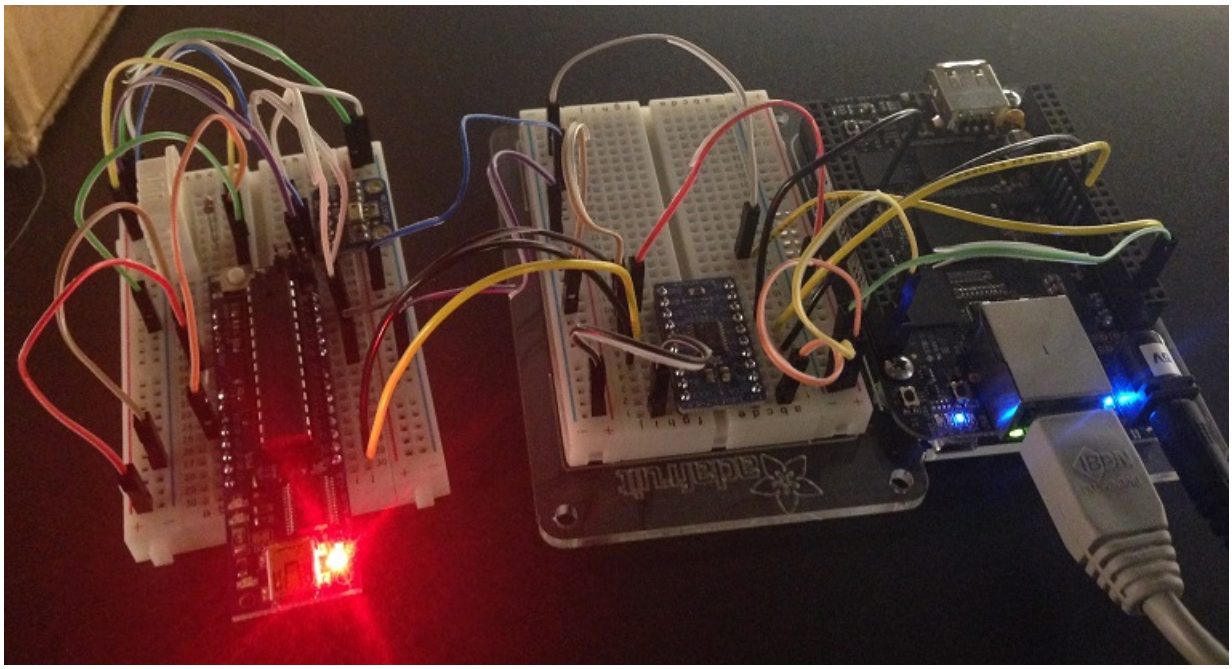
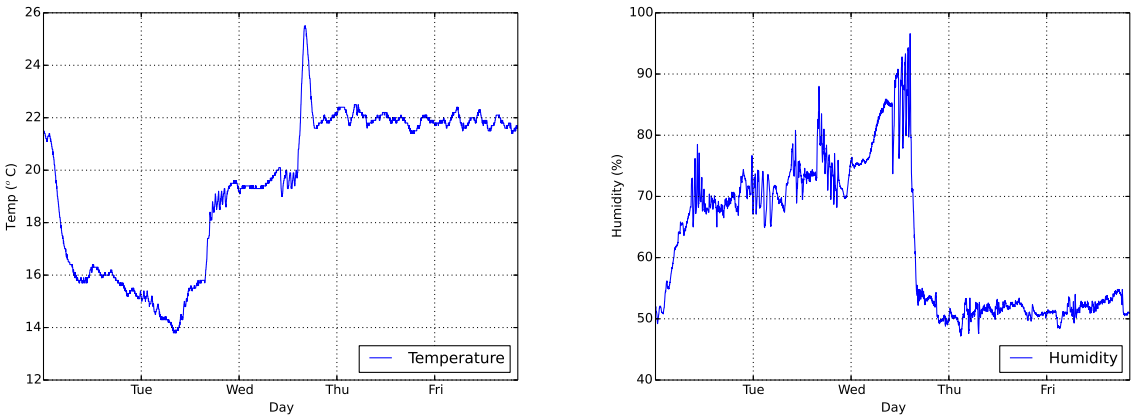


Figure 4.9: The Beaglebone Black and BoArduino Weather Station utilizing a *DHT22* Humidity and Temperature sensor and a *MPL115A2* Barometric Pressure sensor. A *TXB0108* provides the voltage level conversions between the Beaglebone and BoArduino. Communication between the two microcontrollers is GPIO and UART.



(a) (b)

Figure 4.10: AML Environmental Fluctuations during the week of August 18, 2014 (QNC 3508). Data is directly from the Stand-Alone Environmental Monitoring module presented in Figure 4.9, and is presented in its raw form. The humidity is accurate to ±5%; and the temperature is accurate to ±2 °C. [77]

(a) Temperature fluctuations showing the laboratory temperature ranging between 14°C and 25.8°C.

(b) Humidity fluctuations showing the laboratory humidity ranging between 47 % and 97 %. This high humidity caused an indoor fog and condensation on all surfaces which led to equipment failure.

Chapter 5

LabJack and Ocean Optics Timing

5.1 Introduction

Thus far, for the Mark-2 Flame Atomic Emissions Spectroscopy (FAES), the ElectroThermal Vaporizer (ETV) has been fabricated, the electronics have been designed and fabricated, and software has been written for their control. There are two fundamental necessities for the fine control of the ETV and the optical collection of the flame emission: there can be no miscommunication or errors.

Here the timing issues of the Serial Peripheral Interface (SPI) bit-banging of the LabJack, which is documented as a solid, functioning, feature, is investigated along with the solution. [75] In addition, the external triggering control of the Ocean Optics is discussed in order to time the collection of the flame emission with the vaporization of the sample. Both of these are non-trivial in their implementation.

It must be noted that the improvements to the communication, namely the use of signal buffers and LVDS as discussed in Section 4.2.2, were not implemented on the ETV control circuitry, meaning that the digital lines may also be susceptible to noise. These improvements were not included as they were developed after the original ETV Power Supply.

5.2 LabJack SPI Timings

Initial testing of the SPI communication on the test boards presented in Figure 3.7 did not demonstrate any errors for the few trials performed. The devices operated as intended. However, once the process was implemented to control the ETV Power Supply, the continuous SPI communication was able to unearth significant errors in the protocol as implemented in the LabJack. SPI communication is described in the documentation as a solid, functioning, feature. [75] Under closer investigation, the communication protocol is implemented through means of a bit-banging process, and this is the root of the problem. More specifically, the timing of the clock signals were not consistent and there is significant noise on the SPI signals. As a result the *MCP4922* Digital to Analog Converter (DAC) was having difficulty setting the set voltage correctly, as demonstrated in Figure 5.2. This has the effect of drastically altering the power transmitted to the tungsten filament of the ETV, which in turn changes the temperature of the filament. For example, in the case of Figure 5.2, the 2.5 Volts (V) increase in DAC output corresponds to a voltage increase at the filament in the ETV of nearly 8 V, which in turn causes up to an additional 80 Watts (W) of power to flow through the filament.

5.2.1 Investigating the DAC Error

The error was investigated by replacing the exposed Tungsten Coil (W-Coil) with a fully-intact *Osram Xenophot HLX 64633* halogen bulb in order to eliminate the possibility of the W-Coil interacting with the environment causing dynamic load changes to the Power Supply. A *Tektronix MDO3104* Oscilloscope was used to monitor the SPI rails in parallel with the output voltage of the DAC; this was performed by utilizing the DIP probes of the oscilloscope on the DAC itself. The LabJack Python program sends commands to the DAC via SPI approximately every 250 ms and was set to hold a current of 3 Amperes (A) through the halogen bulb. The Oscilloscope was set to trigger on the rising edge of the DAC output voltage, when the voltage increases above a 1.86 V threshold. A single-shot was performed.

The LabJack was connected directly to the ETV Power Supply with no other modules in order to eliminate as many variables as possible. A block diagram of this setup is presented in Section 3.5.2, but is repeated here in Figure 5.1 for ease of reading. A continuously running script read the current through the ETV and set the output voltage to maintain a current of 3 A; no threads were used. For details on the electrical connections please see Section 3.5.2. Note that SPI mode utilized is 0,0; a supported mode of the DAC. [79]

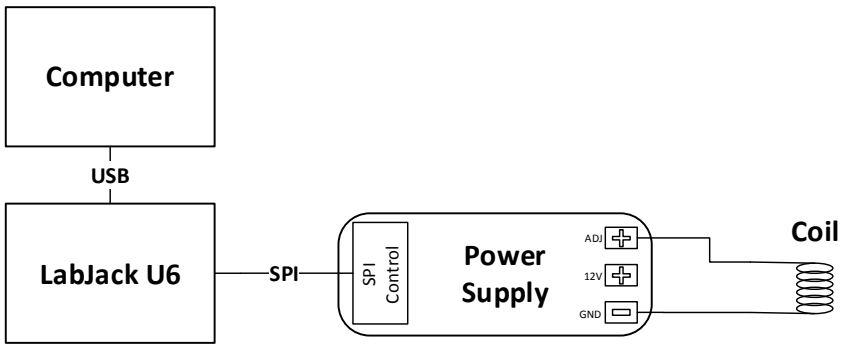


Figure 5.1: Block Diagram of the ETV Power Supply driving the ETV; Demonstrating LabJack U6 and Computer connectivity by means of Universal Serial Bus (USB) and SPI.

Figure 5.2 demonstrates the captured error in the DAC output as well as the fully decoded SPI command being sent to the DAC. The DAC output (pin 14) is displayed in Yellow at 1 V/Div. SDI (pin 5), SCK (pin 4), and CS (pin 3) are displayed in Blue, Purple, and Green respectively at 5 V/Div. It was expected that the DAC output would remain at approximately 0.36 V over the command, but the output has increased to over 2.8 V.

The noise on the output of the DAC at high voltage levels is suspected to be due to the noise on the power rails being fed into the ETV Power Supply when high loads are applied. Although the power is regulated down to 5 V by a *MAX6250*, the Integrated Circuit (IC) is susceptible to noise on its input power as specified in its datasheet. [80] In retrospect, a dual-stage power regulation circuit as implemented for the Photodiode Board discussed in Chapter 4 could work well to resolve this.

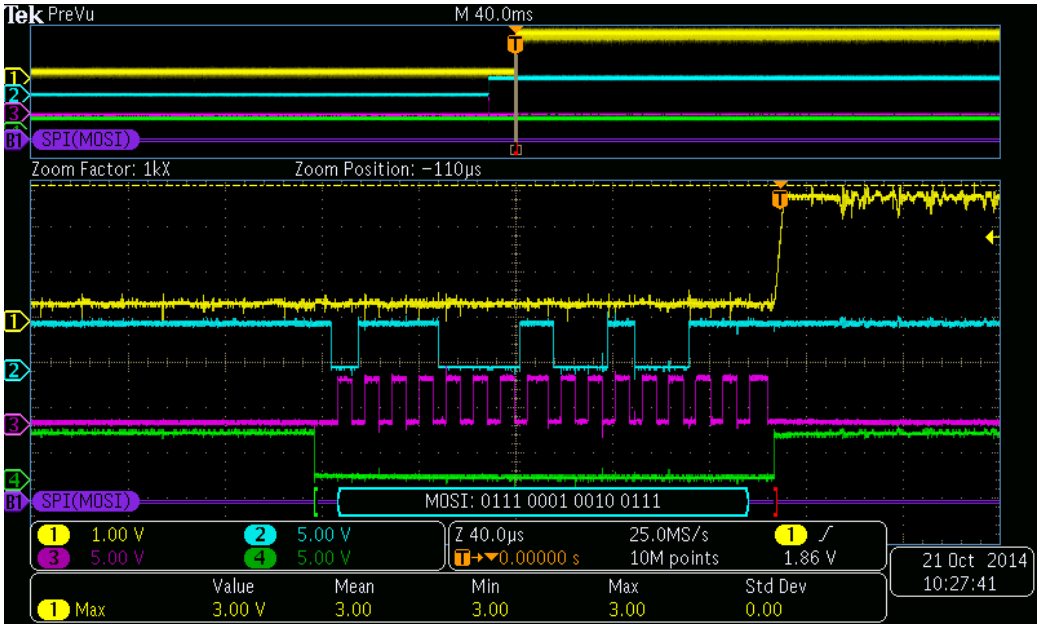


Figure 5.2: Fully decoded SPI signal to the MCP4922 DAC from the LabJack demonstrating variable SPI clock pulse timings and DAC voltage output error. Expected voltage is 0.36 V, but the measured value is over 2.8 V. Yellow is the output voltage of the DAC at 1 V/Div, Blue is the SDI, Purple is the SCK, Green is the CS. SDI, SCK, and CS rails are displayed at 5 V/Div.

The error on the clock pulses is difficult to see within Figure 5.2. However, when zoomed in, it is apparent that the LabJack does not strictly control the SPI timings. Figure 5.3a and Figure 5.3b demonstrate a 6 μs clock pulse as well as a 8.8 μs clock pulse from the LabJack within the same 16-bit word. Ideally this should not happen. However as the SPI is bit-banged by the LabJack it is possible for the error to occur. Furthermore, there appears to be noise on the SPI rails which could also be contributing to the DAC output error. It is suspected that these two factors combined are what is causing the miscommunication to the DAC. It would be advisable to implement the Low Voltage Differential Signaling

(LVDS) or Signal Buffers as discussed in Section 4.2.2 to eliminate the noise.

Fortunately, even with this variable clock timing and noise on the SPI rails, the MDO3104 is able to successfully decode the SPI command being sent to the DAC. This command signal is 0111 0001 0010 0111. The decoding for the command signal was set based upon the SPI Electrical Characteristics of the MCP4922 (Appendix G, Page 246). [79] A high is considered $0.7 \times VDD$ and a low is considered $0.2 \times VDD$. As VDD is 5 V these levels are 3.7 V and 1 V respectively.

Looking at the write command registers for the MCP4922 in Appendix G on Page 245, this command signal can be decoded as presented in Table 5.1. [79] A/B denotes which output the DAC is to update. In this case the 0 corresponds to DAC_A. BUF, GA, and SHDN are all given a 1, which means to buffer the V_{REF}, use a unity gain, and activate the DAC. Bits 11→0 corresponds to the output voltage (D/4096).

Table 5.1: Decoded MCP4922 SPI Command Registers as sent by the LabJack U6

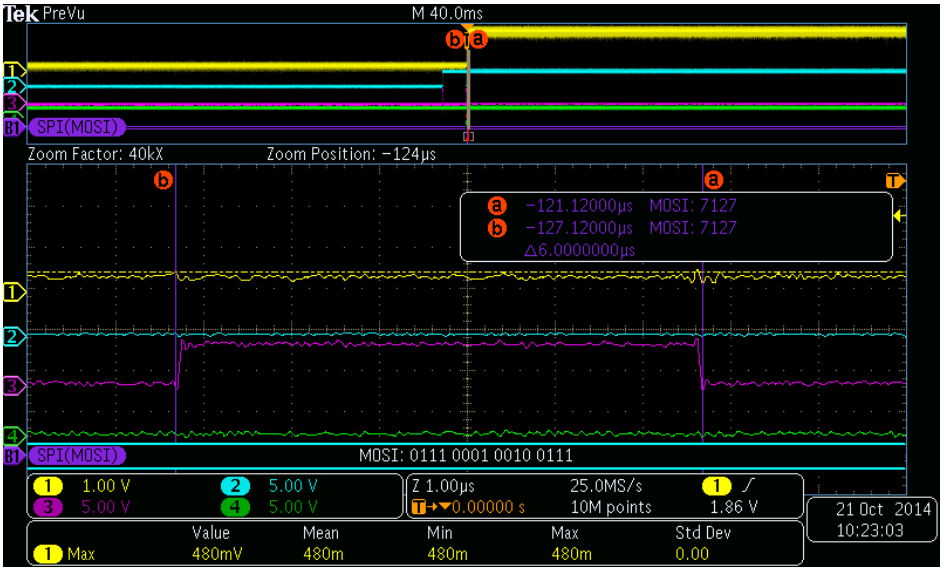
Register	A/B	BUF	GA	SHDN	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Command	0	1	1	1	0	0	0	1	0	0	1	0	0	1	1	1

The DAC has received a command to set the output voltage level to 5 V × 295/4096 = 0.360 V, however, the DAC has instead output a voltage of ∼3 V.

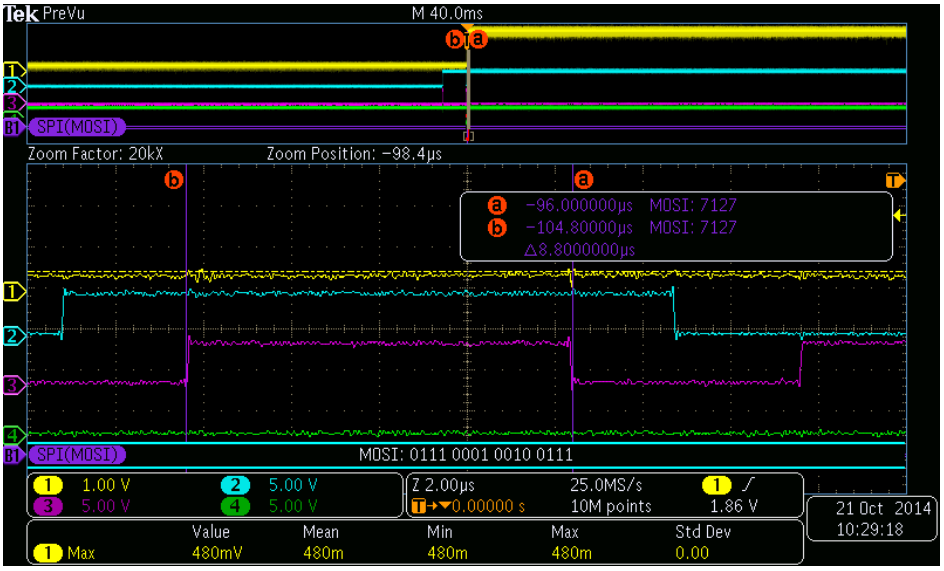
The electrical characteristics in Appendix G on Page 246 demonstrate that there is no minimum clock speed and that as long as the SPI rails are longer than 100 ns the MCP4922 is able to read the signal. [79] As the LabJack is running a variable clock speed, measured to be between 6→8.8 μs (166→113 KHz), the timings are well within the specifications of the DAC. It is possible a combination of variable timings and noise on the SPI rails is to blame for the output error.

5.2.2 Recovery of DAC Error

The reason for the error from the DAC is unknown. A new MCP4922 was inserted in its place and the same error occurred. In both instances, however, it is important to understand that the start of the error was due to a command being sent to the DAC, and the error was resolved upon receipt of a second command. These two DAC commands surrounding the unexpected high DAC output are demonstrated in Figures 5.4 and 5.5 respectively. This isolates the problem to the SPI command itself; either due to noise on the line or the variable timing from the LabJack.



(a)



(b)

Figure 5.3: Measured clock pulses on the single SPI communication between the LabJack U6 and the MCP4922 DAC presented in Figure 5.2, demonstrating timing inconsistencies.
(a) 6 μs measured Clock Pulse.
(b) 8.8 μs measured Clock Pulse.

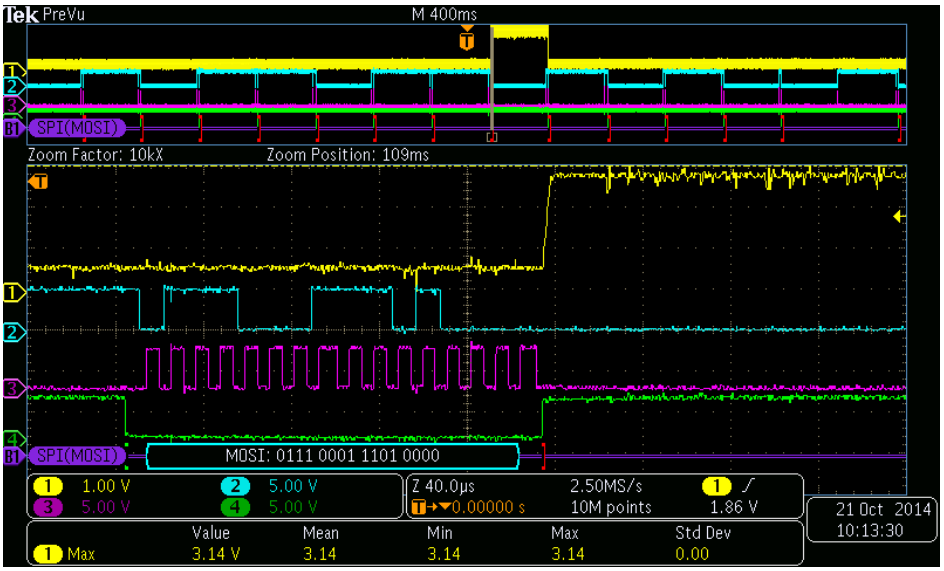


Figure 5.4: Start of MCP4922 DAC Output Error on Receipt of Command. The output voltage (yellow) raises when it should stay the same; signal is presented at 1 V/Div. Noise on the SDI rail (blue), specifically on the second rising edge of the clock pulse (purple), if it affected the received signal, would switch the second received bit (BUF) to 0 thereby disabling the DAC output.

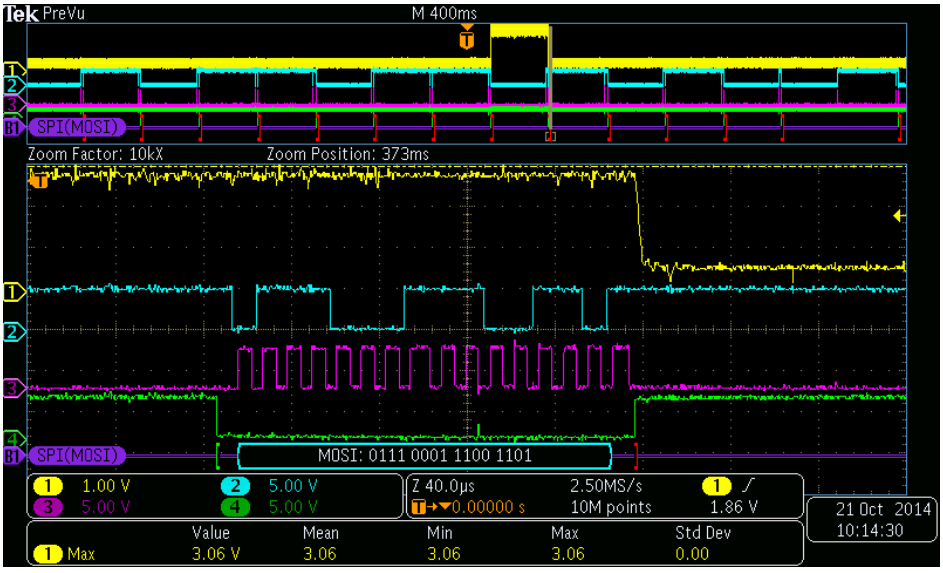


Figure 5.5: Recovery of MCP4922 DAC Output Error on Receipt of New Command. The output voltage (yellow) returns to the expected value; signal is presented at 1 V/Div.

To confirm that the error is sporadic, upon scanning a captured oscilloscope capture, working DAC command signals as well as properly functioning DAC output is evident. The error in the DAC output is indeed sporadic in nature.

Based upon the captured signals on the MDO3104 and the datasheet for the MCP4922 there is no reason for the DAC to be outputting drastically different voltages than what it is being told to output. Furthermore, the majority of the commands sent to the DAC do not cause this error to occur and replacing the DAC has not resolved the issue. This leads to a high probability of noise being the issue.

Fortunately this error is resolved upon receipt of a new command. It is therefore advisable to rapidly increase the speed of commands sent to the DAC in order to reduce the amount of time the error occurs for. Currently, with a 250 ms update rate, the DAC error continues for 1/4 second which is sufficient time to cause significant errors in the vaporization of the sample and cause erroneous measurements. Unfortunately the speed of the LabJack and the python program limits this update rate. As such an external microcontroller such as an Arduino must be implemented to provide an adequate update rate for the DAC which will effectively eliminate this sporadic DAC error. In addition, the use of a hardware controlled SPI may reduce noise on the SPI rails leading to higher communications reliability.

5.2.3 Solution

At the time of these issues, the SPI buffering solutions presented in Chapter 4 were not available. Instead, the use of an external microcontroller, which will hold fixed SPI timings through hardware rather than a software bit-banged communication, was used. Therefore, to keep complexity to a minimum, a BoArduino will be used as the microcontroller has already proven itself with the Proportional-Integral-Derivative (PID) Heater Controller Board. In addition, the use of a hardware SPI may reduce the noise on the SPI rails.

The BoArduino is an Arduino and supports Hardware SPI and Hardware Universal Asynchronous Receiver Transmitter (UART). Therefore in order to ensure proper control of the ETV, a bidirectional digital communication between the LabJack and the BoArduino is required. As already established, this communication must be timing tolerant to allow for the variable timings of the LabJack which more than likely extends to all bit-banged communications.

As the LabJack DB37 port has been used up to this point for the SPI communications, and these SPI rails are shared with other devices, the DB15 port must be used. Furthermore, for simplicity of the communication, UART is to be used. This serial interface is well documented online for the Arduino, and has already been investigated for the LabJack in Chapter 4.

Therefore the fix to the LabJack timing issues will revolve around using the DB15 port to communicate over serial to a BoArduino which will in turn parse the UART data and control the ETV over hardware SPI. This is demonstrated in Figure 5.6. As the microcontroller operates at 16 MHz, the speed of the SPI will be much higher so if an error does occur, a followup command will be sent sufficiently quickly to prevent failure of the sample vaporization.

BoArduino UART

The BoArduino is programmed through two hardware UART pins which are connected to an FTDI chip to allow Serial over USB. If the LabJack is connected to these hardware pins, then the BoArduino will not be able to be programmed or communicate properly with a computer as the Serial communication is intended for device↔device communications. The third device (LabJack in this case) will prevent the Tx lines from being properly toggled by the FTDI chip. Fortunately, a software UART library (*SoftwareSerial.h*) is available for the Arduino. In addition to being a software bit-banged serial communication, this software implementation is much more timing tolerant, thereby improving the communications between itself and the LabJack.

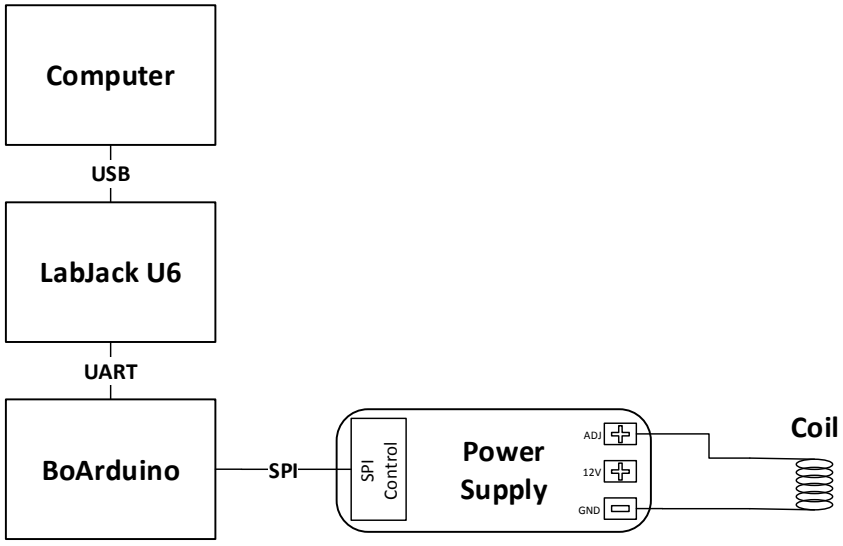


Figure 5.6: Block diagram of Computer, LabJack, BoArduino, and ETV Power Supply connectivity. Here the BoArduino performs the SPI communication with the ETV Power Supply, and is controlled over UART by the LabJack.

By utilizing the software serial, the hardware serial between the BoArduino and computer is unaffected, allowing for real-time monitoring as well as re-programming the device without the need to change any connections between boards. The BoArduino is, however, a 5 V device, but the UART on the Arduino is designed to be able to accept 3.3 V signals; and the LabJack is able to accept 5 V signals. Therefore the implementation of the BoArduino requires connecting Pins 4 and 12 of the DB15 connector (EIO0 and EIO1) to available digital pins on the BoArduino.

The Arduino code DAC and ADC Control presented in Appendix D utilizes pins d7→d2 for the Chip Selects of connected SPI devices. In addition, the hardware SPI pins are located on d11 (SDI), d12 (SDO), and d13 (CLK). This leaves d8→d10 available for the software serial, and therefore pins d8 and d9 will be used as RX and TX respectively. Figure 5.7 presents a populated BoArduino without the installed pin-headers showing the location of all digital pins. [81]

Figure 5.7: Populated BoArduino as used in the FAES system. [81] Digital pins d7→d2 are located in the top left pin header; digital pins d8→d13 are located in the bottom left pin header; the bottom right pin header is analog inputs; and the top right pin header is power.

The following main loop code on the BoArduino was used to test the UART communication between the LabJack and BoArduino. This code works by reading the serial data sent out by the LabJack and returning it to the LabJack. This coupled with the LabJack U6 UART Communication code presented in Appendix B.5 allowed for the testing and debugging of the serial interface between the two devices.

```
void loop(){

  if (mySerial.available()){            //Check if serial data is waiting
    index = 0;
    while(mySerial.available()){
      data[index] = mySerial.read();    //Read in the data one bit at a time
      index++;
      data[index] = '\0';               //Increase the size of the data array
      delay(5);
    }
    mySerial.println(data);             //Send the received data back
  }

  delay(25);
}
```

Software was written for the BoArduino in order to parse the UART communications from the LabJack, and is presented in Appendix D.5. This function is called on every iteration of the main loop and will call the appropriate command and pass along the relevant conditions to maintain. For instance, if a command to dry the sample is received, the script will call the *Dry()* function. Upon completion, the BoArduino will send the LabJack the string "Dry Done". At all times, if a command to 'Kill' is received, the BoArduino will exit all functions and set the DAC output for the current and voltage control to zero.

The SPI communications and timings were again measured using the MDO3104 oscilloscope once the BoArduino was implemented. Figure 5.8 demonstrates consistent behavior with a clock frequency of 2 MHz. The anomalies which were found, notably noise and variable timing, were conveniently removed by the use of an Arduino; rather than requiring the implementation of the LVDS or Signal Buffers as proposed in Section 4.2.2.

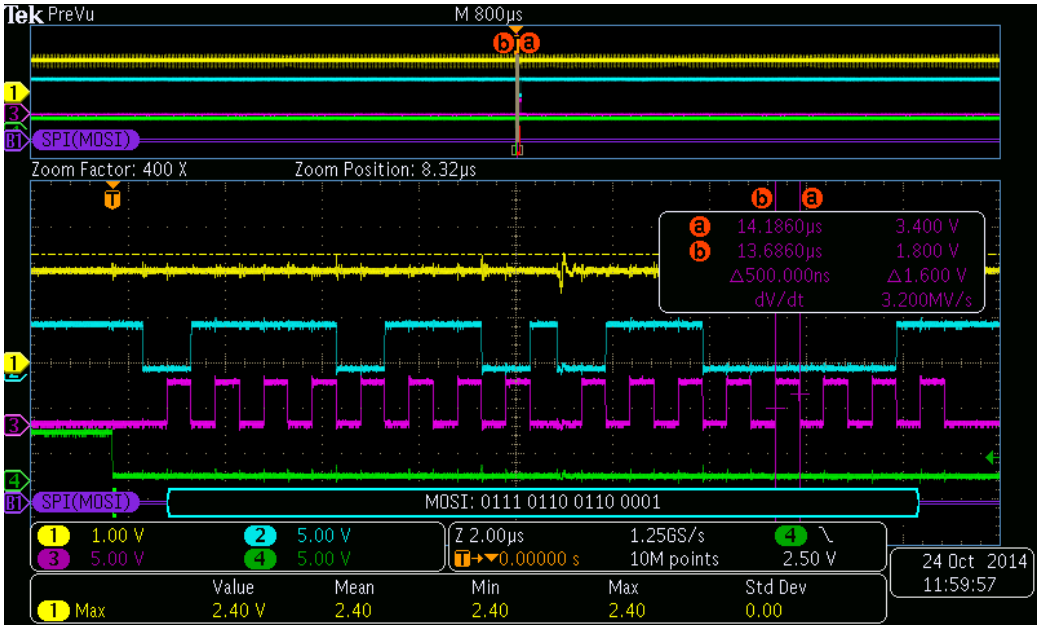


Figure 5.8: BoArduino hardware SPI communication as measured at the MCP4922 DAC on the ETV Power Supply. Tight clock frequency of 2 MHz is present as indicated by the 500 ns clock pulse. Note the drastically reduced noise on both the DAC voltage output and SPI rails compared to when the LabJack was used. The sporadic error is now non-existent.

5.3 Ocean Optics Trigger

The *Ocean Optics USB4000 FW 1.03.3* Spectrometer used in the FAES has an external triggering option. The spectrometer itself has a *Samtec IPT1-111-01-S-D-RA* connector. Pin 7 provides the external trigger input, and is a TTL input signal requiring 5 V logic; Pin 6 provides the signal ground. The software operating the device then allows for configuration of the external trigger. According to the triggering manuals, there are three trigger modes available: [82, 83]

1. External Hardware Level Trigger
2. External Hardware Edge Trigger
3. External Synchronous Trigger

5.3.1 External Hardware Level Trigger

In this trigger mode the integration time is set via software. The spectrometer then waits for a digital high (5 V) on the trigger signal, and then acquires continuous spectra until a digital low (0 V) is received. For example: if the integration time is set to 1 second and the trigger pulse is set high for 10.25 seconds, the device will capture 11 one second integrations. More importantly, however, is that if the integration time is 50 ms and the triggering event occurs 48 ms into the integration cycle, the 48 ms prior to the trigger will be acquired in addition to the 2 ms following the trigger. [82, 83]

With integration times of one second or longer, this trigger mode leads to a potential loss of data of up to one second. This is a substantial amount of data and would ultimately result in insufficient timing accuracy to catch the resulting spectra from a vaporized sample, which is important in the data analysis. As such this triggering mode cannot be not used for the FAES.

5.3.2 External Hardware Edge Trigger

In this trigger mode the integration time is set via software. The spectrometer waits for a digital high (5 V) on the trigger signal, and acquires one spectrum. This trigger acquires one spectrum each time that there is a sharp rising edge (digital high), provided an acquisition is not already in progress. This solves the problem of the External Hardware Level Trigger in that tight timing tolerances are possible with regard to the start of the acquisition. However, it does not allow for continuous data acquisition without sufficient delay between trigger pulses. For example, if the integration time is set to one second, and the trigger pulses are set to rise every second for 10 seconds, every other trigger would be detected resulting in five captured spectra. The timing diagram for the Edge Trigger mode works well to explain this and is presented in Figure 5.9. [82, 83]

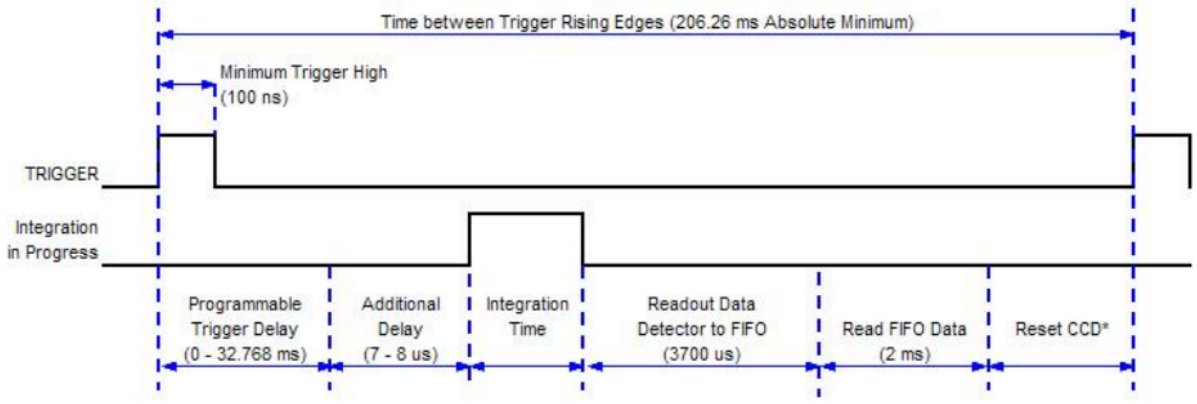


Figure 5.9: Ocean Optics Hardware Edge Triggering timing diagram. [82]

As can be seen, the absolute minimum time between Trigger Rising Edges is the sum of the Programmable Trigger Delay (0 ms), Additional Delay (7-8 μs), Integration Time, Readout Data Detector to FIFO (3700 us), Read FIFO Data (2 ms), and Reset CCD. The Reset CCD time is given by Equation 5.1 for integration times between 200→2097 ms. [82] Table 5.2 uses this timing diagram and Equation 5.1 to calculate the absolute minimum time between trigger rising edges for integration times of one and two seconds.

$$\text{Reset Time } (\mu\text{s}) = \left(\frac{\text{Integration Time } (\mu\text{s})}{32} \times 10\right) - 2000\mu\text{s} \tag{5.1}$$

Table 5.2: Absolute minimum time between Trigger Rising Edges for External Hardware Edge Trigger on a USB4000 Spectrometer.

Trigger Delay (ms)	Additional Delay (μs)	Integration Time (s)	Readout Data Detector to FIFO (μs)	Read FIFO Data (ms)	Reset CCD Time (s)	Minimum Time between Trigger Rising Edges (s)
0	8	1	3700	2	0.3105	1.316208
0	8	2	3700	2	0.6230	2.628708

In this case, an absolute minimum loss of 0.316208 seconds of data between subsequent spectrum acquisitions for a one second integration time is too substantial. A different triggering mode must be utilized for the FAES.

5.3.3 External Synchronous Trigger

In this trigger mode the spectrometer acquires data from an external trigger event (digital high of 5 V on the trigger pin) until the next time the trigger is activated, at which time the spectrometer ceases spectral acquisition and begins a new one. Integration time is entirely dependent upon the trigger signal and the trigger can fire at random intervals. [82, 83]

According to the documentation for the external triggering, the trigger specifications are as follows: [82, 83]

- Minimum Trigger High = 100 ns
- Minimum Time Between Trigger High = 7.61 ms
- Trigger Low (0→0.8 V)
- Trigger High (2.2→5.5 V)

As the LabJack seems to lend itself to larger amounts of noise, as per Section 5.2, a BoArduino is to be used to send the trigger pulses to the Ocean Optics USB4000. Therefore, an Arduino script was created and loaded onto a BoArduino in order to keep tight timing tolerances on the integration times. The initial trial was set to one second integrations (half second trigger pulses). The code is presented below, and as can be seen is completely stand alone from other devices or input. When the board is powered on it will indefinitely supply square-wave pulses at a frequency of 1 Hz.

The test system for the triggering is very simple, and is presented in Figure 5.10. The computer used was isolated from any other external connections and running a fresh installation of Windows 7 Professional 64-Bit and Ocean View 1.4.0 to record the spectra. The trigger signal is very clean and to tight timing tolerances, as seen in Figure 5.11.

```
const int ooPin = 13;
int ooState = LOW;
unsigned long ul_PreviousMillis = 0UL;
unsigned long ul_Interval = 500UL; //half a second

void setup() {
  pinMode(ooPin, OUTPUT);
  ul_PreviousMillis = millis();
}

void loop() {
  unsigned long ul_CurrentMillis = millis();
  if( ul_CurrentMillis - ul_PreviousMillis > ul_Interval)
    {
      ul_PreviousMillis = ul_CurrentMillis;
      if( ooState == LOW){
        digitalWrite(ooPin, HIGH);
        ooState = HIGH;
      }
      else{
        digitalWrite(ooPin, LOW);
        ooState = LOW;
      }
    }
}
```

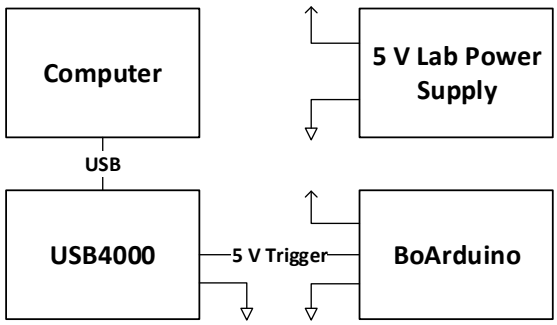


Figure 5.10: Block diagram of Computer, USB4000, and BoArduino Test System for Ocean Optics Triggering. Here the BoArduino provides the trigger pulses as a stand-alone unit, and the computer is set to save all recorded spectra.

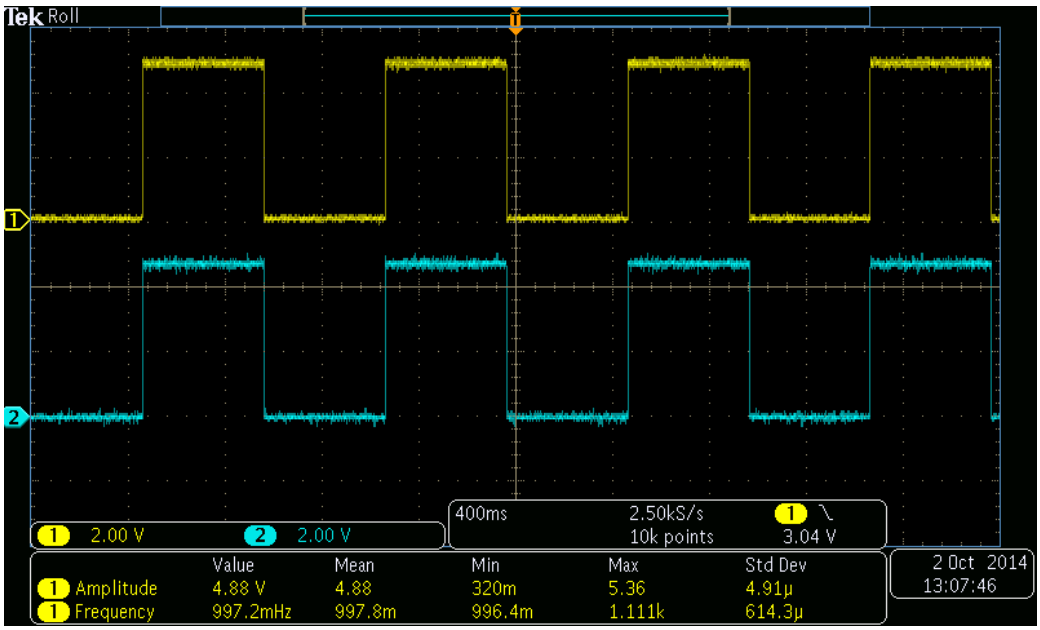


Figure 5.11: BoArduino Trigger Pulse as measured at the BoArduino and USB4000 Spectrometer with a MDO3104, demonstrating a 1 Hz trigger pulse at an amplitude of 4.88 V with extremely low standard deviations. The blue signal (2 V/Div) is measured at the BoArduino, while the yellow signal (2 V/Div) is measured at the USB4000.

This trigger signal is within the specifications and, according to the documentation, should produce continuous one second integration captures. Unfortunately this does not occur. The USB4000, tested under both firmware 1.03.3 and firmware 3.01.00, misses some trigger pulses. The setup was checked for ground loops, and other sources of noise. A personal communication with Russ Gill of Gamble Technologies Inc., who provides support to the Applied Miniaturisation Laboratory (AML) on Ocean Optics equipment, confirms that Ocean Optics Inc. is presently, at the time of writing, looking into this issue as they do not know why this is occurring. Multiple computers, software revisions, trigger source signals, and spectrometers were tested in order to resolve the issue to no avail following the same system configuration as presented in Figure 5.10. Unfortunately, the spectrometer would randomly miss a trigger and integrate for two seconds rather than one resulting in the loss of two seconds of data!

Over 10 trials of exactly 100 trigger pulses at 1 Hz, an average of 6.82% of trigger pulses were missed. This was calculated by comparing the total number of files generated to the number of files expected, giving the number of missed trigger pulses. This high percentage of missed triggers results in an average of one lost acquisition for every ten, however, because the pulse was missed a total of two spectra acquisitions is lost. As an example to illustrate this, consider five trigger pulses being sent to the USB4000; if the third trigger pulse is missed and the fourth trigger pulse is detected, the second acquired spectrum would have a two second integration time. This is demonstrated in Table 5.3. The fifth trigger pulse does not generate a file as it denotes the end of the third acquisition in this case, and the beginning of the fourth; this due to the missed trigger pulse. This is as documented in the Ocean Optics Triggering Manuals. [82, 83]

Table 5.3: Example integration times when a trigger pulse is missed.

Trigger Number	Pulse Detected?	Integration Time (s)	Actual Integration time (s)
1	Yes	1	1
2	Yes	1	2
3	No	1	N/A
4	Yes	1	1
5	Yes	1	N/A

A “fix” was generated using the BoArduino and the trigger pulses themselves. A series of short triggers would be sent every time a trigger pulse is desired. Therefore instead of sending a singular pulse every second, a series of 4 pulses would be sent. In this case a 75% failure mode on the detection of the trigger pulse is tolerated. According to the MDO3104

Oscilloscope, the BoArduino series of trigger pulses being sent to the USB4000 have the following specifications, and can be seen in Figure 5.12:

- Rise Time: 15 ns
- Fall Time: 15 ns
- Trigger High: 1 ms
- Trigger Low: 10 ms
- Time Between Trigger High = 11 ms
- Trigger Low Voltage = 0→0.38 V
- Trigger High Voltage = 4.88→5.04 V
- Number of Pulses = 4

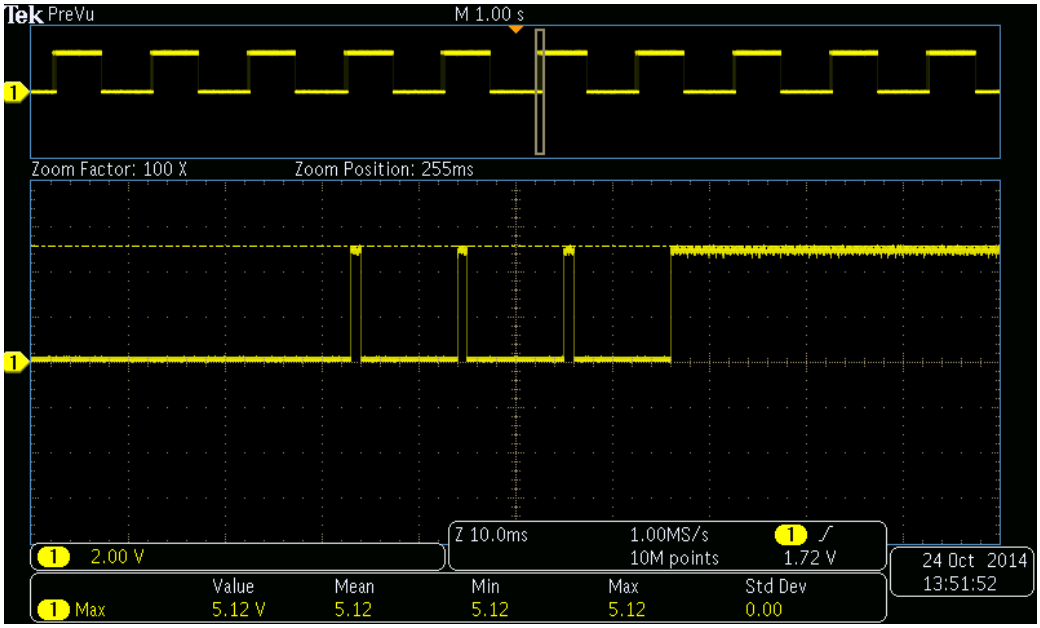


Figure 5.12: Modified BoArduino Trigger Pulses as measured at the USB4000 Spectrometer with a MDO3104, demonstrating the four trigger pulse highs each second to resolve the missed trigger pulses.

These multiple trigger pulses results in a maximum of three short integration times followed by a long integration of desired length, in this case one second. This "Band-Aid" solution results in a maximum loss of 64 ms of data each second, which is significantly less than the other trigger modes. In addition it allows for the capture of spectra to be timed precisely to an external event as desired. The microcontroller is able to ensure the

timing of the pulses remains accurate. Unfortunately, this approach can result in up to 40 captured spectra over the course of 10 seconds with a one second trigger (four pulses per trigger). As such a script to differentiate between the different integration times will need to be developed in order to reject the files generated by the shorter integration times from the multiple short pulses. This script, and its development, is discussed in Chapter 6, Section 6.5.

The BoArduino, which controls the trigger signal, has two inputs to inform the microcontroller when to enable the trigger. One input is connected to the Fluidics Board which will toggle when the ETV is to be cleaned. The second is connected to the BoArduino which is controlling the ETV Power Supply. In both cases the signal to initiate the trigger is digital and as long as the signal remains high the modified trigger signal will be generated. This connectivity is described in more detail in Chapter 6, Section 6.2.

5.4 Conclusion

To conclude, the LabJack SPI function, which is supported via bit-banging, does not support rigorous timings and has significant noise on its digital outputs. This results in miscommunication to connected devices. This communication error has been eliminated through the use of UART communication between the LabJack and a timing-tolerant BoArduino, which then provides the necessary hardware SPI to properly control the connected device. Furthermore, the use of a BoArduino is required in order to brute-force a "Band-Aid" solution for the Ocean Optics External Synchronous Trigger mode, which has the tendency to miss trigger pulses. Ocean Optics Inc., at the time of this writing, is investigating the issue. The associated hardware modifications and scripts were presented and discussed for both solutions.

Chapter 6

Mark-2: ETV Application

6.1 Introduction

In this chapter the amalgamation of circuits, software, and samples are discussed as a working Flame Atomic Emissions Spectroscopy (FAES). The instrument will be characterized and limitations identified.

6.2 Electronic Overview

The electronic control overview is presented in Figure 6.1. As can be seen the system is fairly complex. If tight timing tolerances and repeatability were not important, one could use a basic variac to control the flow of power to the ElectroThermal Vaporizer (ETV). However, the benefits surrounding automation far outweigh the complexity of the system.

The first computer connects to the USB4000 Spectrometer and sets the acquisition parameters to External Synchronous Trigger, which is described in Section 5.3.3. This computer will then save the resulting collected spectra into a single folder as a delimited file type for further processing. The second computer runs the main Python 2.7 program which interfaces with the LabJack to control the entire FAES instrument.

The LabJack provides the DB37 and DB15 ports to communicate with the Interface Board and the BoArduino Power Supply Control respectively. The Universal Asynchronous Receiver Transmitter (UART) communications are based at 9600 baud to relax communication speed requirements of the LabJack. The control of the Interface Board and connected daughter boards is discussed in both Chapter 4 and Appendix F.

The power for the Interface Board, BoArduino Power Supply Control, and BoArduino ETV Housing Proportional-Integral-Derivative (PID) Heater is provided by Advanced Technology eXtended (ATX) Power Supply 1. This allows for a common ground and similar 5 and 12 Volts (V) rails among the connected circuits. The Interface Board provides all the necessary power to connected daughter boards. Not listed in this diagram are the addition of the Power Supply Extension Modules which boost the ETV Power Supply current handling abilities to 10 Amperes (A). These fans are powered by the 12 V rail on ATX Power Supply 1 as well. The second ATX Power Supply is used solely to perform the cleaning of the Tungsten Coil (W-Coil) ETV.

Originally the two ATX power supplies were connected in series to provide a 17 V supply to the ETV Power Supply. This, however, proved to be unstable when large currents were required. As such the supplies were split. In addition, when performing the cleaning on the ETV with ATX Supply 2, the current draw is considered a short. The ATX Power

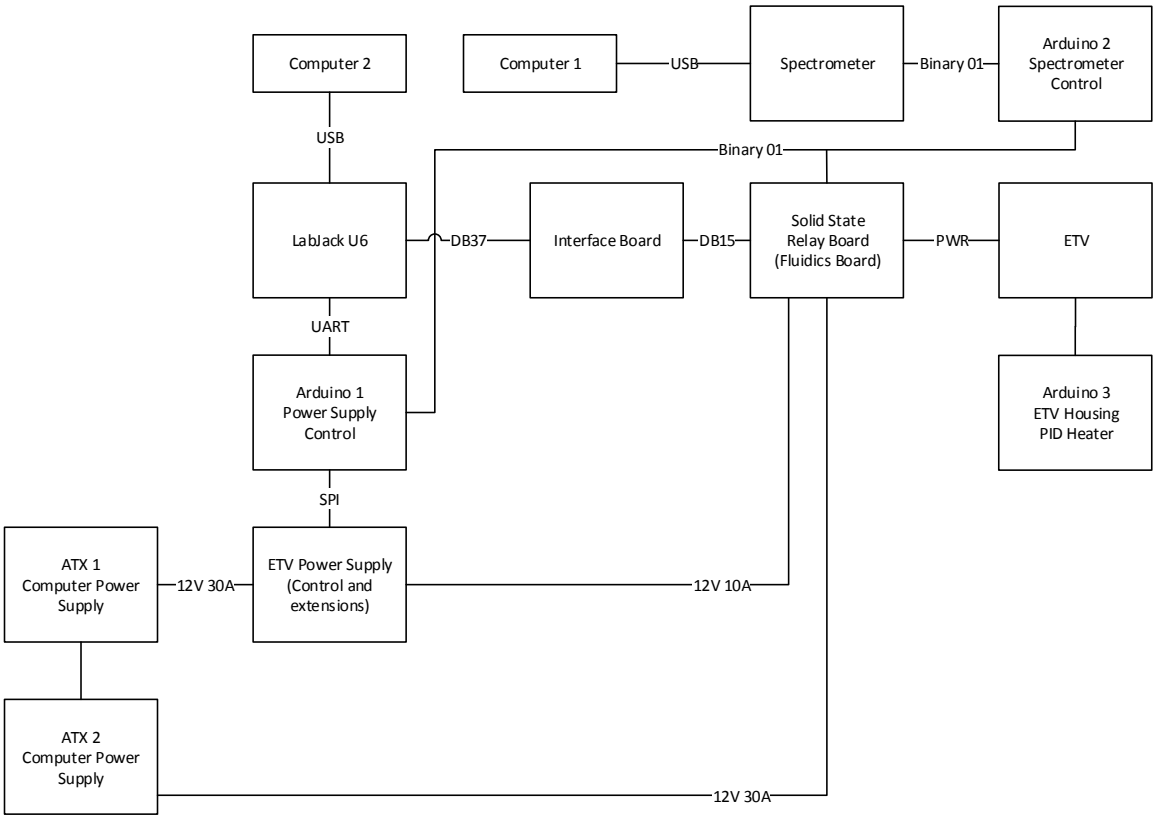


Figure 6.1: FAES Instrument Electronic Connections Overview.

Supply would shut down. This is resolved by turning the power supply on when it is connected to the filament directly, as the start-up current draw is expected and settles as the W-Coil heats. Therefore the Fluidics Board (or as labeled in Figure 6.1, Solid State Relay Board) toggles the activation of the ATX Power Supply 2 as well as the connecting and disconnecting of the power rails to the ETV.

A condition was programmed into the LabJack which prevents the Fluidics Board from enabling both solid state relays at the same time, which would cause damage to the ETV Power Supply. Therefore only one relay is active at any given time, and if an error is detected, the LabJack program will automatically disable power to the ETV.

The Spectrometer Control is provided via two digital input pins on the BoArduino Spectrometer Control. When the binary signal is high, the modified trigger pulse is sent as described in Section 5.3.3. As these two binary lines are independent to one another, either condition may be met in order to simplify the code. When the ETV is to be cleaned, the Fluidics Board will signal the Spectrometer Control to begin the modified trigger 0.25 seconds before the actual cleaning process begins. When the sample is being vaporized for the test, the trigger command signal is sent slightly before the power is supplied. Both binary signals must be low in order to disable the modified trigger pulse.

The BoArduino ETV Housing PID Heater is stand-alone and will continuously operate when power is supplied. The heaters themselves may be activated or deactivated with the supply of a 12 V rail to the board. However, when 5 V is supplied the BoArduino will activate. No damage will arise to the board or heaters if either the 12 or 5 V rails are connected at any given time.

6.3 Software Control Overview

The control software takes advantage of the threading capabilities of Python. In this case a Graphical User Interface (GUI) was designed for ease of use. Each button, once pressed, will disable command input until the thread has completed with the exception of the 'Kill' command to stop all power. This modular approach to programming the control algorithm allows for simple debugging as required. In addition, since the majority of the control is handled by a BoArduino through UART, a very logical set of function are possible to reduce program size. The GUI is presented in Figure 6.2.

A few key elements in the software were duplicated from the work of Donati et al. in 2008. [84] Specifically the New Filament Cleaning Procedure, and the New Filament Conditioning Procedure. These two procedures improve the coil performance and remove

residual oxidative components when used in W-Coil-Atomic Emission Spectroscopy (AES), as described in the literature. [84] As the FAES here also uses a W-Coil, these procedures should have the same positive affect on the coil.

The *'Dry'* button send the command "Dry" to the BoArduino. The BoArduino then hold a steady current of 3 A through the filament. While doing so it monitors the required voltage to maintain the current levels and once the voltage begins to rapidly increase, indicating a rapid increase in temperature, the filament is considered dry.

The *'Bake →'* button will read the Bake Time in seconds and the Bake Temp in Celsius to construct an appropriate message to send to the BoArduino. For example, if the Bake Time is 45 seconds and the Bake Temp is 1150 Celsius the message will read: "Bake 45 1150". The BoArduino will then parse this UART message and hold the set temperature of 1150 Celsius for a total of 45 seconds.

The *'Auto Dry/Bake'* button combines the above two actions and will dry the sample prior to baking (vaporizing) the sample. Following this the ETV will be cleaned for 10 seconds by enabling a power conduit between the W-Coil and ATX Power Supply 2 and turning on the power. The thread which performs this task is presented on Page 98.

The *'New Clean Filament'* button sends several bake commands to the BoArduino following the new coil cleaning procedure as described by Donati et al. in 2008. [84] Once completed, the last step follows the cleaning procedure which runs for 35 seconds rather than the traditional 10.

The *'Condition Filament'* button sends the UART command "Condition" to the BoArduino. The BoArduino then follows the new coil conditioning procedure as described by Donati et al. in 2008. [84]

The *'Auto Dry/Clean Filament'* button combines the Dry and Clean procedures into one. It is designed to rapidly remove any oxide layers on the W-Coil. A small sample of Milli-Q water is typically used for this.

The *'Clean Filament'* button initiates the cleaning procedure for the ETV. It works by enabling a power conduit between the W-Coil and ATX Power Supply 2 and turning on the power for 10 seconds.

The *'Kill Power'* button is always available to the user. It has the power to instantly disconnect the power conduits to the ETV in case of an error. In addition it sends the "Kill" command to the BoArduino which will set the voltage and current controls to 0 on the ETV power supply. Once pressed the program will lock the user out for a few seconds while it kills all of the power and resets all of the boards.

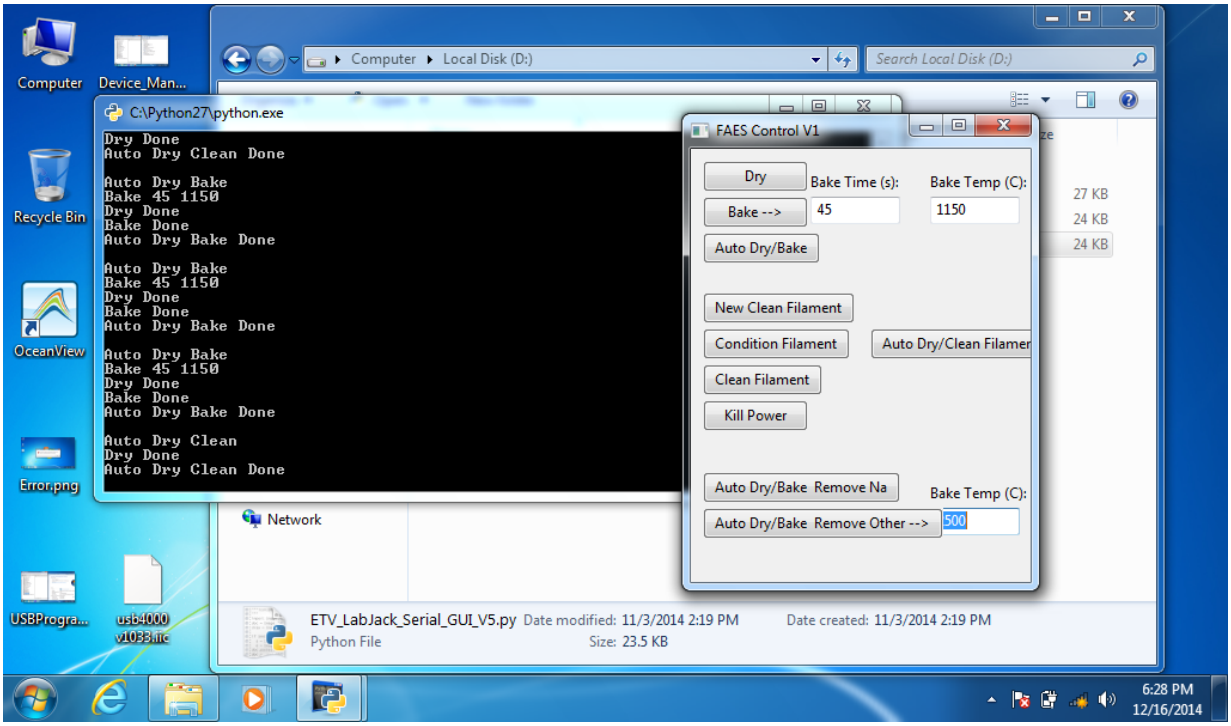


Figure 6.2: FAES Instrument Control GUI.

The *'Auto Dry/Bake Remove Na'* button is designed to provide a secondary bake function. This is discussed further in Chapter 7, but it is used to vaporize off interfering elements. The Na aspect means that a defined temperature has been pre-set.

The *'Auto Dry/Bake Remove Other →'* button is identical to the button above, however, the user may select the intermediate bake temperature. In this case all three text fields are utilized to build the test profile desired.

The use of a GUI, with the buttons, to provide full control over the FAES provides an elegant and easy to use user-interface. In addition, a full integration of automated control with minimal user input is possible.

The thread used to automatically dry and bake (vaporize) the sample of interest is presented below:

```
1 #Thread to autmoatically dry and then bake (vaporize) the sample of interest
2 def Auto_Dry_Bake_sample(self):
3     print " "
```

```
    print "Auto Dry Bake"
    #Enable ETV relay. Power now available to W-Coil
    turn_ETV_relay_on()

    #Prepare the UART message for the BoArduino
    message = 'Bake ' + str(Bake_Time.GetValue()) + ' ' + str(Bake_Temp.GetValue())
    print message

    #Send the Dry command and wait for a responce
    if (Send_UART_Message('Dry','Dry Done') == 1):
        #Sample is now dry, disable power to W-Coil
        turn_relays_off()
        print "Dry Done"
        #Wait a few seconds between steps
        time.sleep(time_btn_steps)
        #Enable ETV relay. Power now available to W-Coil
        turn_ETV_relay_on()
        #Send UART message to BoArduino and wait for a responce
        if (Send_UART_Message(message,'Bake Done') == 1):
            #Sample has now been vaporized, disable power to W-Coil
            turn_relays_off()
            print "Bake Done"
            time.sleep(time_btn_bake_clean-2)
            #Clean the filament, first activate Ocean Optics trigger
            turn_OO_relay_on
            time.sleep(2)
            #Turn on ATX Supply 2 for 10 seconds
            turn_ATX_relay_on()
            time.sleep(10);
            #Kill all power
            turn_relays_off()
            Enable_Buttons()

    #Exit the thread cleanly
    turn_relays_off()
    print "Auto Dry Bake Done"
    thread.exit()
```

6.4 Fabricated FAES Instrument

The electronics and fans were mounted on a half-inch thick piece of Medium-Density Fibreboard (MDF). A series of 3.5 inch long spacers were made from 3/4 inch MDF cut into

one inch wide strips in order to produce a raised platform to support the TS4000 Blow Torch, ETV, USB4000 Spectrometer, and collection optics. This upper platform was made using 1/4 inch MDF. Larger supports were fabricated using 2x2 pieces of lumber cut to appropriate heights.

The mounted electronics can be seen in Figure 6.3. Here the majority of the electronics are shown, however, the BoArduino fixes have not yet been installed. Figure 6.4 shows the complete instrument along with Computer One connected to the USB4000, and Computer Two which is running the control software for the instrument. In Figure 6.5, the 'Y' connection which allows for the mixing of the vaporized sample (in forming gas) and the propane fuel is displayed.

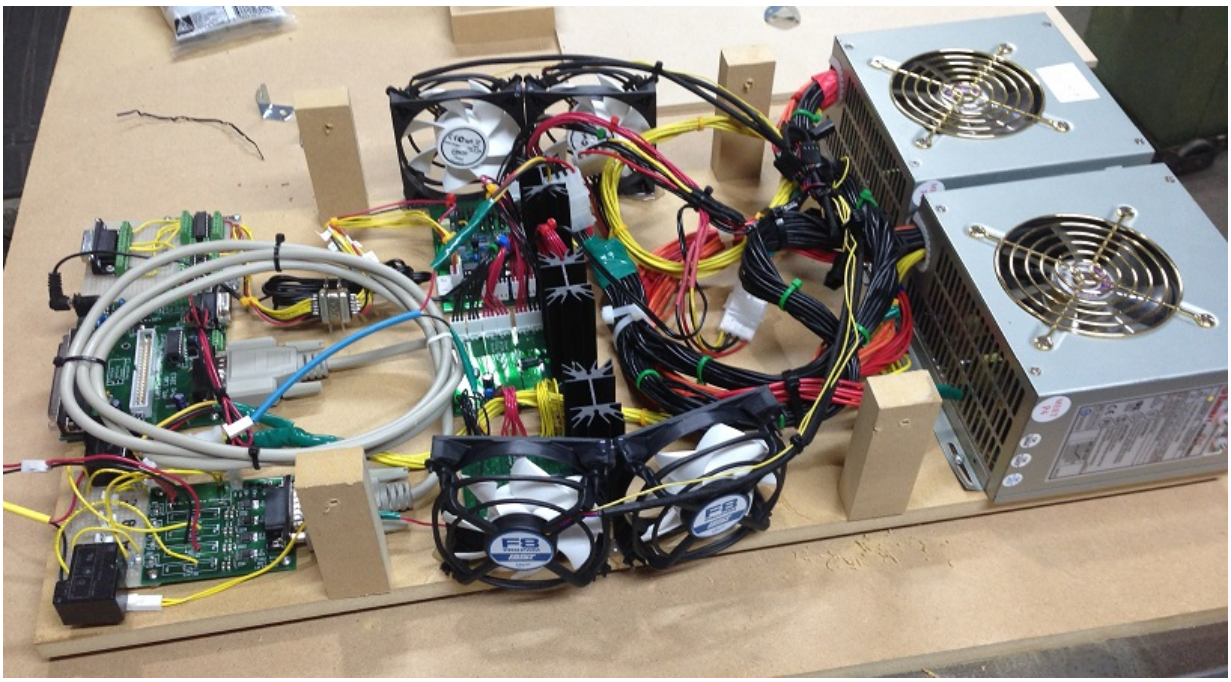


Figure 6.3: FAES Electronics mounted on 1/2" MDF. Here the BoArduino fixes for the LabJack and Ocean Optics timings have not yet been implemented. The ATX power supplies are on the right; ETV Power Supply and Extension Modules are in the middle with the four 80 mm fans providing active cooling; Interface Board, modified Fluidics Board, and the Voltage-Level Translator board are on the left.

As modeling the flame temperature as a function of the environmental conditions is outside the scope of this thesis, the flame temperature is assumed to be as reported, at 2200 K. [28] Therefore, as well documented by others, and as discussed in Section 2.3.4,

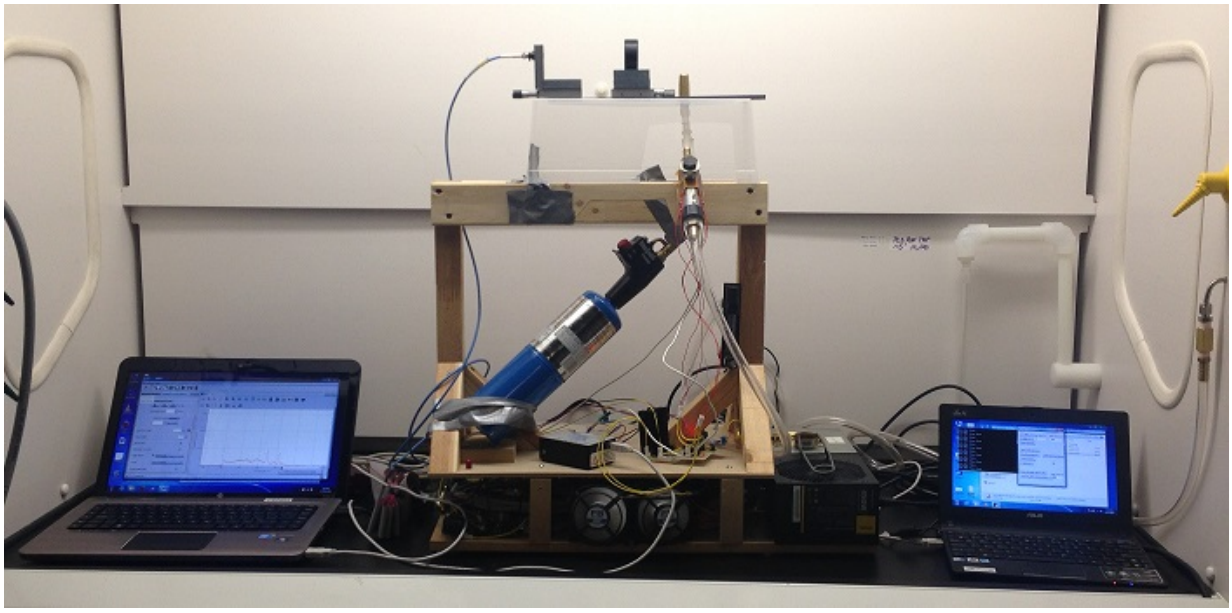


Figure 6.4: Fully Assembled FAES with Both Computers. The larger 2"x2" lumber can be seen mounted to the 1/4" MDF; the PID Heater controller, USB4000, and TS4000 are resting on the 1/4" MDF; The ETV is mounted to the top of the wood frame with two Omega heaters; Optics at the top are resting on a clear plastic bin to allow for a 'Y' connection. Computer One is on the left connected to the spectrometer; Computer Two is on the right running the control algorithms. Forming gas is provided through the connection on the right.

this temperature flame is limited to easily excitable elements such as Ba, Ca, Li, K, and Na. [4, 18, 40]

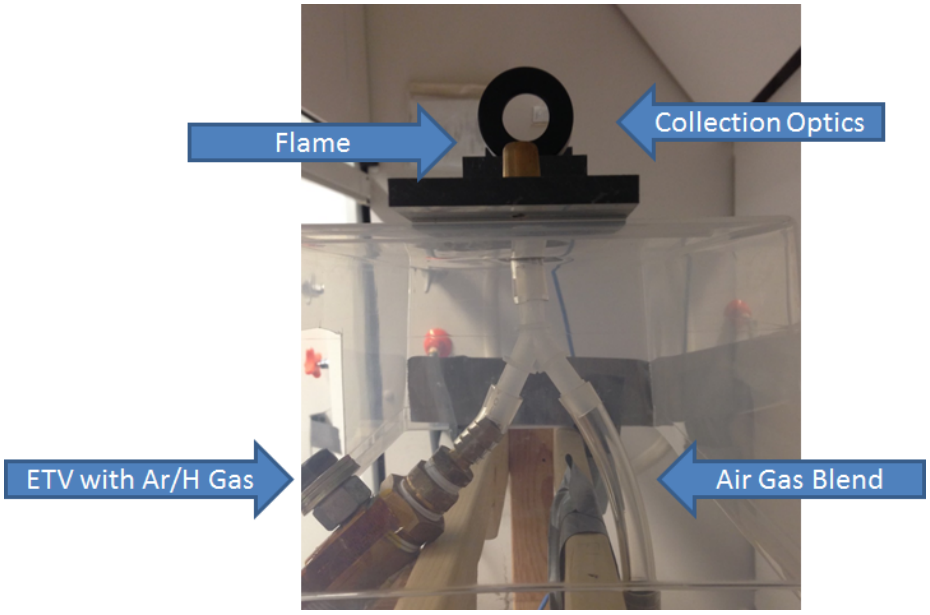


Figure 6.5: Closeup of FAES 'Y' Connection for Mixing Gas and Vaporized Sample; Pre-Ignition. The ETV is on the left supplying vaporized sample in forming gas, Propane-Air fuel mixture from the TS4000 is on the right, Flame and Optics Collection are at the top. The total inner volume of the tubing with the pre-mixed gas is less than 0.5 cubic inches.

6.5 Data Processing

The data processing for the ETV-FAES begins with identifying the proper acquisition files which correspond to each sample, and which within those correspond to the one second integrations. This is due to the multiple trigger pulses sent to the Ocean Optics USB4000 Spectrometer each second, as discussed in Section 5.3.3. The second step is to perform a baseline correction to the data. The third step is to present the data as a single acquisition.

The methodology behind calibrations curves is very similar, however, instead of plotting the data directly the peak intensity for the element of interest is plotted over multiple concentrations. Using Linear Regression Analysis, a series of single-blind, or unknown, samples are analyzed and the measured intensities are converted into starting concentrations to validate the system.

All of these stages are discussed in the following subsections.

6.5.1 Identifying Proper Acquisition Files

As discussed in Chapter 5, Section 5.3.3, the modified trigger pulse being sent to the Ocean Optics USB4000 from the BoArduino can result in up to 40 captured spectra over the course of 10 seconds. This is because the modified trigger sends out four trigger pulses each second to account for trigger loss by the Ocean Optics. In addition, because of the way the Ocean Optics software saves the individual acquisition files, each file is saved within one folder. This means that for 10 consecutive tests, there could be a total of 400 acquisition files in a single folder. As such, a script was developed in Python 2.7 which would be able to distinguish between files which were integrated for the desired time.

This script operates using two properties of the acquisition files:

1. The files generated contain the time, to the millisecond, that the acquisition was completed.
2. The first five pixels (wavelengths) within the files are covered by an opaque material and are designed to drift downwards with larger integration times. These are negative values which continue to decrease with longer integration times. [85]

The folder containing all of the generated spectra files, which can be in the hundreds after several tests, is sorted. This is done based upon the timing difference between files. As previously discussed in Section 5.3.3, the maximum time lost is 64 ms on a one second integration; therefore the maximum time between subsequent one-second acquisitions is 1.064 seconds. The time between subsequent testing of additional samples is significantly larger than this time, and as such these time gaps correspond to new samples. The files are then sorted into folders corresponding to the sample number. The script written to perform this sorting is presented in Appendix B.6. Once this sorting is completed, a modified version of the script presented in Appendix B.2 is run. The modified version cycles through all of the *Test_OO_N* folders, where N corresponds to the sample number, and generates a Comma-Separated Values (CSV) file for each test. An example CSV file for any given test may resemble that presented in Table 6.1.

In this table, the larger integrations of around -500 counts in intensity on Test 1 and Test n correspond to one second integrations. During post-processing of this data, the first column (Test 0) is removed as this may be an integration of up to four seconds in length. [82, 83] Following this, the average intensity count across the first row is calculated and would, in the case of Table 6.1, result in an absolute intensity of around 290 counts. Any column which has an absolute intensity count greater than this average value for the first three wavelengths is considered to have been integrated for the full desired integration time of around one second. The number of successful one-second integrations is counted, and if

Table 6.1: Example Data Structure in *Test_OO_N.csv* as built by Python Script from Spectra generated with Modified Trigger Signal. The data is sorted with the script presented in Appendix B.6 and compiled with the base script is presented in Appendix B.2.

Wavelength (nm)	Test 0 (Counts)	Test 1 (Counts)	Test 2 (Counts)	$\cdots$	Test n (Counts)
178.32	-97.67	-528.69	-49.31	$\cdots$	-495.77
178.53	-97.67	-528.69	-49.31	$\cdots$	-495.77
178.75	-97.67	-528.69	-49.31	$\cdots$	-495.77
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$
886.57	62.07	298.69	102.24	$\cdots$	280.41

that number matches the number of seconds the test was run (typically 10), the test data is considered complete. If there is data missing, the test run is considered lost.

The negative intensities stem from a baseline thermal calibration of the USB4000 as described in the datasheet. [85] These values are typically set between 90→140 counts at the time of manufacture. This baseline number will slowly drift downwards a few counts depending upon the external environment. This drift, however, does not affect the differentiation as an averaging process to determine a threshold intensity is used, as previously discussed. During integration, these negative values, which are factory set, are also integrated leading to higher absolute values with larger integration times. This provides a consistent basis to differentiate between different integration times.

6.5.2 Baseline Correction

The idea behind the baseline correction is to account for any environmental and flame instabilities between sample runs. It is highly unlikely that the environmental and flame conditions are identical between each sample test. This is because minute changes in temperature will affect the sensitivity of the spectrometer, and as the propane cylinder loses pressure, the flame will slowly change. [85] The affect of the propane cylinder on the flame is investigated in Section 6.6. As such, a background and baseline is determined for each test individually.

After the successful acquisitions have been identified for the sample (Figure 6.6a), the total area under the entire acquisition curve is calculated, through integration, for each second of acquisition. The acquisition with the smallest area under the curve is considered to be the background. This is because the acquisition of the emission spectrum occurs prior to the vaporization of the sample and extends beyond the completed test. The additional

emission by an element causes an increase in the area under the curve in relation to the background flame. Therefore, the raw background data, determined based upon the area under the curve, from the same acquisition set is then subtracted from the entire data set (Figure 6.6b). As there are slight fluctuations of up to 64 ms in each spectra, the baseline hovers around zero after the subtraction. This is corrected by detecting the baseline and shifting the spectrum accordingly (Figure 6.6c). These three steps are demonstrated on test data from a trial run of the instrument in Figure 6.6. The sample was comprised of Na and K.

Due to the methodology behind the ETV, the entire sample is vaporized rapidly under test conditions. In real-world timing, this takes just over one second to accomplish. As such, there are two acquisitions which correspond to the actual data. Through experience, these two acquisitions are the third and fourth one-second integrations as captured by the USB4000. This is taken into account when the baseline correction is being performed, in that these two integrations are never used as the baseline.

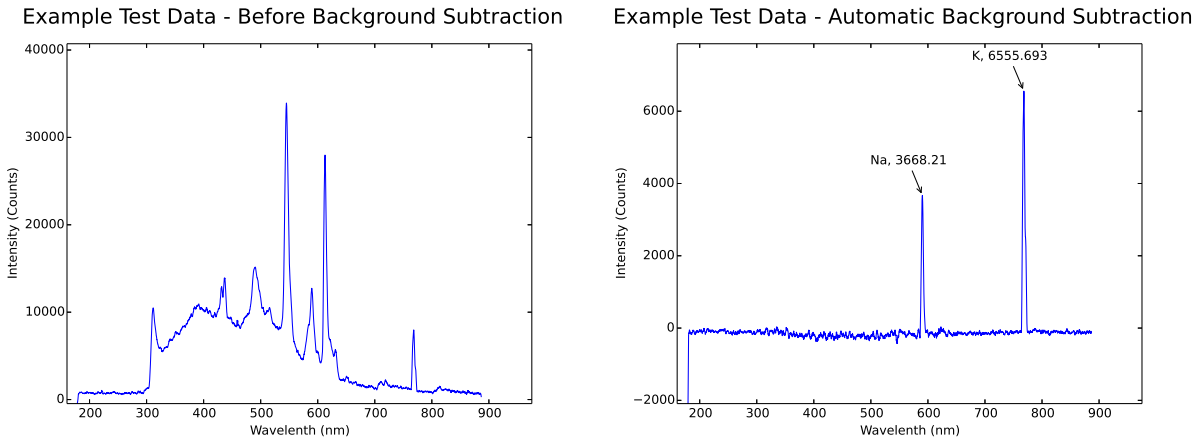
After the background is subtracted, under some situations, propane flame emission spectrum is not completely removed. In these cases, a second background subtraction is performed.

This second background subtraction needs to be performed after the first background has been subtracted. The second background is determined by calculating the average intensity for each acquisition spectra between 340 and 420 nm. This region corresponds to the beginning of the propane flame emission and does not overlap with any of the samples of interest investigated within this thesis. Once the average intensity for each acquisition is determined, the largest average intensity, which does not correspond to the two largest integrations, is selected. This selected acquisition is then subtracted from the entire data-set.

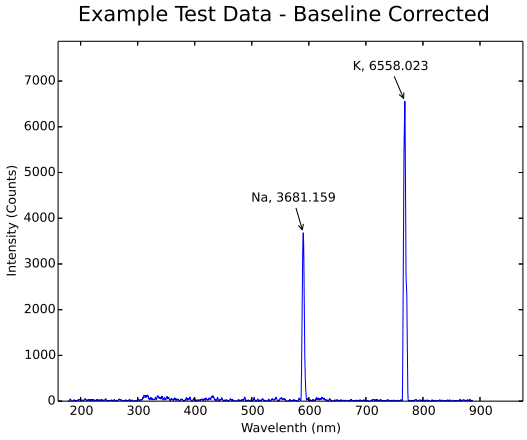
6.5.3 Generate Single Plot

After the background and baseline correction, the next step is to generate a single acquisition curve corresponding to the data of interest. This is performed by adding the two acquisition files, with the largest integrations, together. A moving average over 10 data points is then performed to smooth the data.

Within this new single data set, the top peaks are identified and labelled according to the element. This is performed by comparing the wavelength location of the peaks with Table A.1 in Appendix A. In addition to labelling the element, the peak intensity is also labelled for each peak.



(a) (b)



(c)

Figure 6.6: Example data processing on acquired data, demonstrating the three analysis steps. The 10 μM sample tested contained 1 μM Na and 1 μM K.

(a) The proper acquisitions are identified and amalgamated into a single acquisition.

(b) The background is automatically detected and subtracted from the merged acquisition. Na and K are now easily identified at 589 nm and 766.5 nm.

(c) The baseline is detected and corrected to zero. Peak intensities for Na and K are changed by less than 1 %.

This information is then plotted for easy analysis.

6.5.4 Calibration Curves

A *N_LOD_Test_Summary.csv* file is manually generated, where N is the element of interest, and contains a table listing the date, sample number, element, concentration, volume, procedure, and whether or not all the data was present. A script is run to perform complete data processing for each test listed in the CSV file. However, instead of plotting the results for each sample, the peak data corresponding to the element of interest is saved to a new CSV file: *Rebuilt_N_Peak_Data.csv*.

Once the peak data is amalgamated, the standard deviation and average intensity of the peak for each concentration is calculated. This averaged data is then plotted for each concentration to form the calibration curve. The Limit of Detection (LOD) is determined by dividing three times the standard deviation of the blank by the slope of the calibration curve [61].

The sample standard deviation is calculated for the peak of interest following Equation 6.1. Where s is the sample standard deviation; y is the peak intensity; $\bar{y}$ is the average peak intensity; and n is the number of samples of the same concentration tested. [36]

$$s = \sqrt{\frac{\sum (y - \bar{y})^2}{n - 1}} \tag{6.1}$$

6.5.5 Calculating Concentrations of Single-Blind Samples

In order to validate the instrument, a series of single-blind trials were performed. Dr. Chris Backhouse generated a series of 'mystery samples' to be analyzed using the FAES. These samples contained various mixtures of multiple elements at different concentrations.

In order to analyze the data from the single-blind trials, the same methodology is used as in Section 2.3.3 in Chapter 2: Linear Regression Analysis. The peak intensity values corresponding to the element of interest are then matched to the calibration curve to determine the approximate concentration.

6.6 Characterizing the Instrument

6.6.1 Preparing Samples

Stock solutions were prepared in the same manner as those prepared for the Mark 1 Nebulizer. They were prepared individually using a *Mettler Toledo AG104* microbalance to weigh out appropriate masses of each chemical. 100 mL solutions of 0.1 mol L^{-1} (μM) Mg, Manganese (Mn), Na, Ca, Cu, Li, and K were made from Milli-Q water. Table A.1 in Appendix A on Page 144 lists the characteristics for these elements.

The chemicals utilized to generate the stock solutions for each element are listed in Table 6.2.

Table 6.2: List of Elements and Chemicals used in testing the Mark 2: ETV-FAES

Element	Abbreviation	Chemical	Chemical Formula
Magnesium	Mg	Magnesium Chloride	$MgCl_2$
Manganese	Mn	Manganese(II) Chloride	$MnCl_2$
Sodium	Na	Sodium Chloride	NaCl
Calcium	Ca	Calcium Chlorate Di-Hydrate	$CaCl_2 \cdot 2H_2O$
Lithium	Li	Lithium Hydroxide Monohydrate	$LiOH{\cdot}H_2O$
Potassium	K	Potassium Chloride	KCl
Copper	Cu	Cupric Sulfate	$CuSO_4 \cdot 5H_2O$

From each stock solution, a series of 1:10 serial dilutions were performed using Milli-Q water, 1.5 mL centrifuge vials, a *Gilson Pipetman P200* and a *Gilson Pipetman P1000*. The following steps were performed for each stock solution:

1. Pipette 1 mL using the P1000 into vial 1.
2. Add 900 μL Milli-Q to vials 2→9 using the P1000
3. For n=1 to 8; using a new pipette tip, add 100 μL from vial n to vial n+1 and mix thoroughly.

These steps result in concentrations ranging from 100 mM to 1 nM allowing for calibration curves to be developed.

6.6.2 Checking for Carry-Over and Flame Stability

As discussed in Chapter 2, two major contributors to instability and accuracy of the FAES are the flame source and the effect of carry-over.

Flame Stability

One of the primary causes for concern for reproducibility of the ETV-FAES Mark 2 is the potential for propane variability. When the flame is starved for oxygen, as in Figure 6.7a, there are multiple unexplained peaks with intensities as high as 2000 counts. These erroneous peaks even overlap with the emission lines for Na and K. As a result, when this flame is used for analysis, artificial peaks are detected.

Here the affect oxygen deprivation on the flame instability is discussed. Figure 6.7 looks at the propane variability under air starvation and not for one second integrations.

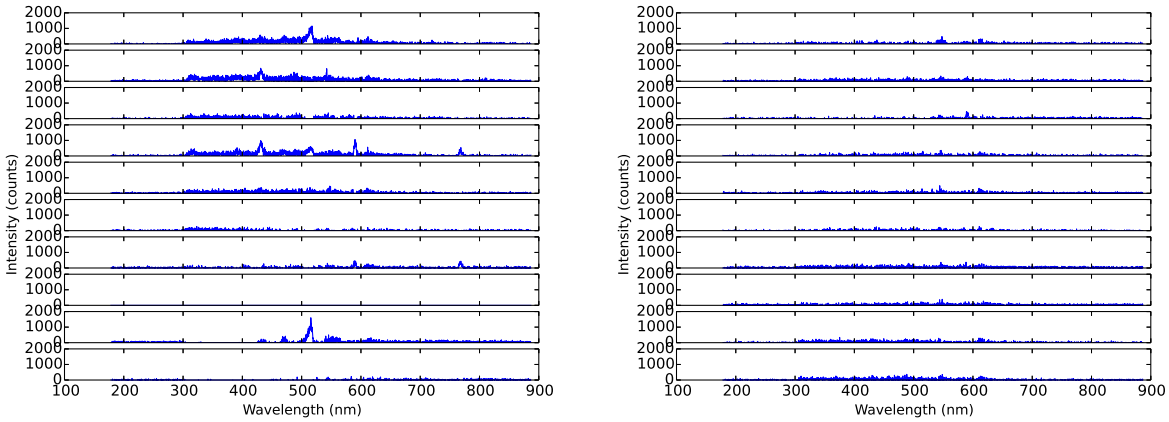
As can be seen, the total variability on one-second integrations of the propane flame from the TS4000 is extremely stable with non-restrictive air-flow, at less than ∼300 counts variability and no distinguishable peaks. The variability when air-flow is restricted, on the other hand, is extremely high with various repeating and non-repeating peaks.

Beyond the effect of air on the flame, the amount of propane in the gas cylinder may also have an effect. Unfortunately, only observable qualitative information can be used to analyze this. It was observed throughout the use of many Propane Cylinders, that the only flame differences that can be seen (and heard) are within the last five minutes before the cylinder becomes empty (approximately the last 3% of the fuel cylinder, given 2.82 hours of burn time). [42] This is most likely because the average gas consumption of 2.36 g/min is regulated by the pressure of the gas itself within the cylinder by the TS4000; with the consumption of gas, the pressure drops below the regulation pressure, and the flame rapidly degrades. Unfortunately, this regulation pressure is not published. Figure 6.8 depicts the flame throughout the four main stages, as observed, as the propane cylinder empties.

Therefore, looking at Figure 6.7 and utilizing the qualitative information as presented in Figure 6.8, it can be concluded that the TS4000 propane flame is sufficiently stable and unchanging provided the fuel is not empty and there is sufficient air supply.

Carry-Over

Carry-Over occurs when the sample tested is not completely removed from the filament during the cleaning phase. In this case, residual sample will be vaporized on subsequent

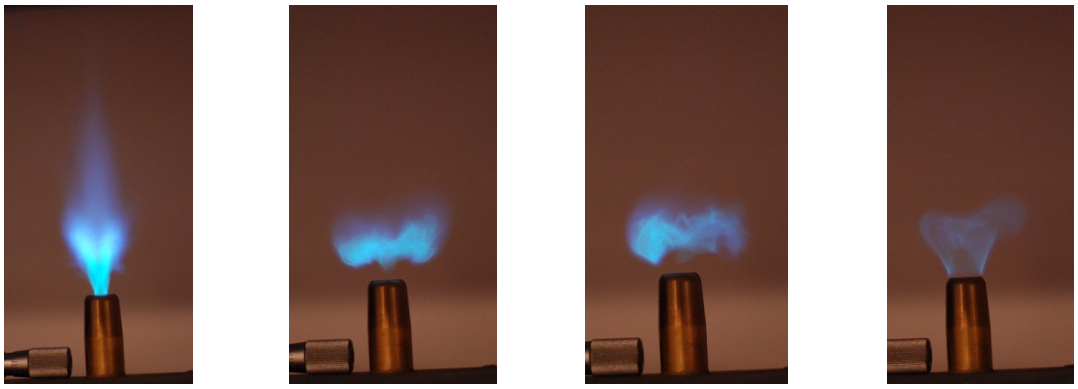


(a) (b)

Figure 6.7: ETV-FAES Mark 2 TS4000 Blowtorch Propane Flame Stability for 10 consecutive one-second integrations with the USB4000, under two different air supplies. The background has been subtracted and baseline correction has been performed. In order for a flame to be stable, no unexplainable peaks should be present.

(a) With limited air supply: Total variability of up to 2000 counts with various repeating and non-repeating peaks. The flame is unstable.

(b) With full air supply: Total variability less than 300 counts. No blips or unexplainable peaks present. The flame is a factor of 6.67 times more stable than that of the limited air supply.



(a) (b) (c) (d)

Figure 6.8: ETV-FAES Mark 2 TS4000 Blowtorch Propane Flame as Propane Cylinder Empties.

(a) Stage 1: Full Cylinder; Flame is crisp. Remains this way until cylinder is empty.

(b) Stage 2: Cylinder almost Empty; Flame rapidly jumps from Stage 1 to Stage 2 in appearance. Remains in this stage for about 2 minutes.

(c) Stage 3: Cylinder almost Empty; Flame shifts from stability of Stage 2 to rough Stage 3. Remains in this stage for about 1 minute.

(d) Stage 4: Cylinder Empty; A slow decline from Stage 3 to Stage 4 as the cylinder empties. Will remain in this stage for approximately 30 seconds before completely empty.

tests resulting in fouling of the data. Another potential for carry-over occurs from the housing itself. If vaporized sample condenses on the housing and is loosened on subsequent tests, those tests would contain fouled data.

In order to test for this, high concentrations of Sodium were used as this element is extremely prevalent in biological specimens. In the human body, blood Na concentration is matched by a saline solution consisting of 154 mM (0.9%) Sodium Chloride (NaCl). [86] Therefore a 154 mM concentration of Na is prepared in Milli-Q. Milli-Q water is also used as the blank sample to check for residual Na peak. 10 μL sample volumes are used.

The samples were processed by utilizing the *Auto Dry/Bake* function as described in Section 6.3. A bake temperature of 1150 °C was used for both samples.

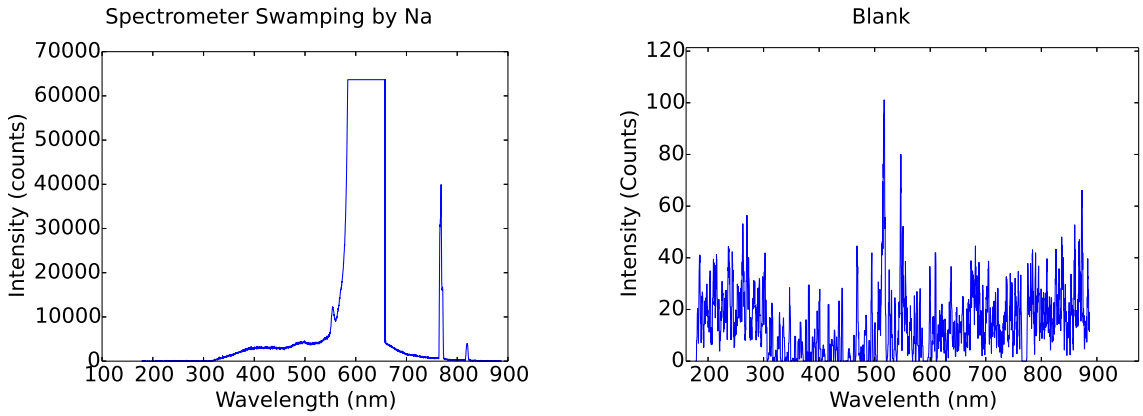
The first test is to ensure that the high concentration of Na (154 mM) is sufficient to swamp the spectrometer. This allows for the confirmation of complete Na removal from subsequent runs and that carry-over no longer occurs. This swamping of the spectrometer can be seen in Figure 6.9a. The second test is the blank Milli-Q sample run after the automated cleaning procedure to confirm whether carry-over is a present within the system. The blank, as can be seen in Figure 6.9b, shows zero Na emission thereby proving Carry-Over is not an issue with the Mark 2 ETV-FAES.

6.6.3 Generating Calibration Curves

The ETV-FAES is powered on in the following manner:

1. Check environmental conditions
2. Begin Forming Gas flow at 6 Standard Cubic Feet per Minute (SCFM)
3. Turn on Computers One and Two
4. Connect the USB4000 to Computer One
5. Connect the LabJack U6 to Computer Two
6. Turn on ATX Power Supply One
7. Begin ETV Control Program on Computer Two
8. Turn on ATX Power Supply Two
9. Wait for PID Heater Controller to indicate proper temperature (50°C) has been reached

The data presented in Chapter 4 Figure 4.10 clearly identifies the need to check the environmental conditions in order to ensure similar operating conditions. As such the limitations on the Temperature and Humidity were set; The Temperature must be between 18→25°C, and the Humidity must be between 25→65 %.



(a) (b)

Figure 6.9: Study of Carry-Over on the Mark 2 ETV-FAES. Tests were performed in triplicate and representative data is presented.

(a) Test 1: 10 μL 154 mM Na Sample, which is sufficiently high concentration to swamp the USB4000 Spectrometer.

(b) Test 2: 10 μL blank Milli-Q sample run immediately after Test 1 showing zero carry-over.

As discussed in Chapter 3, the PID Heater Controller is used in order to maintain the housing temperature. This is important in order to ensure that each test occurs with the same starting conditions. The results of implementing this heater are presented in Figure 3.2.

As previously discussed, there are a total of nine samples for each element being tested. In order to test for carry-over effects, a Milli-Q sample is tested between each sample. Any residual element, after the cleaning step, will be vaporized and be indicated on the resulting spectrum as a peak.

A sample volume of 10 μL has been selected for all tests as this constitutes the total working volume among other Applied Miniaturisation Laboratory (AML) Lab on Chip (LOC) designs. [64, 65] This closely approaches the target of microlitre sized samples in the interim and represents a decrease in volume of 4 orders of magnitude over the 100 mL sample volumes required by the Mark 1.

Therefore, for each of the seven elements, nine samples of each were run with a Milli-Q sample between each to search for carry-over. This results in 126 tests, which were each performed in triplicate resulting in 378 total runs. These runs were performed in a random order to reduce the impact of systematic error and ensure reproducibility. Unfortunately, three of the elements (Cu, Mn, and Mg) could not be easily identified or quantified.

Initial calibration curves were created from the data generated from testing concentrations of 0, 1, and 10 μM in order to perform a direct comparison to the Mark 1 Nebulizer system. Concentrations $\geq$100 μM resulted in a sufficiently strong emission spectrum to swamp the spectrometer. These concentrations could be tested if a lower working volume was used. The LOD presented is calculated by dividing three times the standard deviation of the blank by the linear slope of the data. [63] The error is the average standard deviation multiplied by the LOD. The calibration curves are presented in Figure 6.10.

A table listing the Peaks and their Standard Deviations for the LOD plots of Figure 6.10, corresponding to each concentration of Ca, K, Li, and Na, is presented in Table 6.3. The standard deviations were calculated following Equation 6.1 as described in Section 6.5.4. The non-zero values for each of the elements at 0 μM could be attributed to flame variability or room light; Carry-over is unlikely due to the automated cleaning step between each sample. It is most likely, however, that the sample was contaminated with minute amounts of each of the elements prior to testing.

As can be seen in Figure 6.10, non-linear plots are apparent. The non-linear trend of Na can be explained by the swamping of the spectrometer; the peak is located within the region of the flame thereby significantly limiting the maximum peak intensity. Ca on the other hand was not reaching sufficiently high peak intensities to swamp the spectrometer.

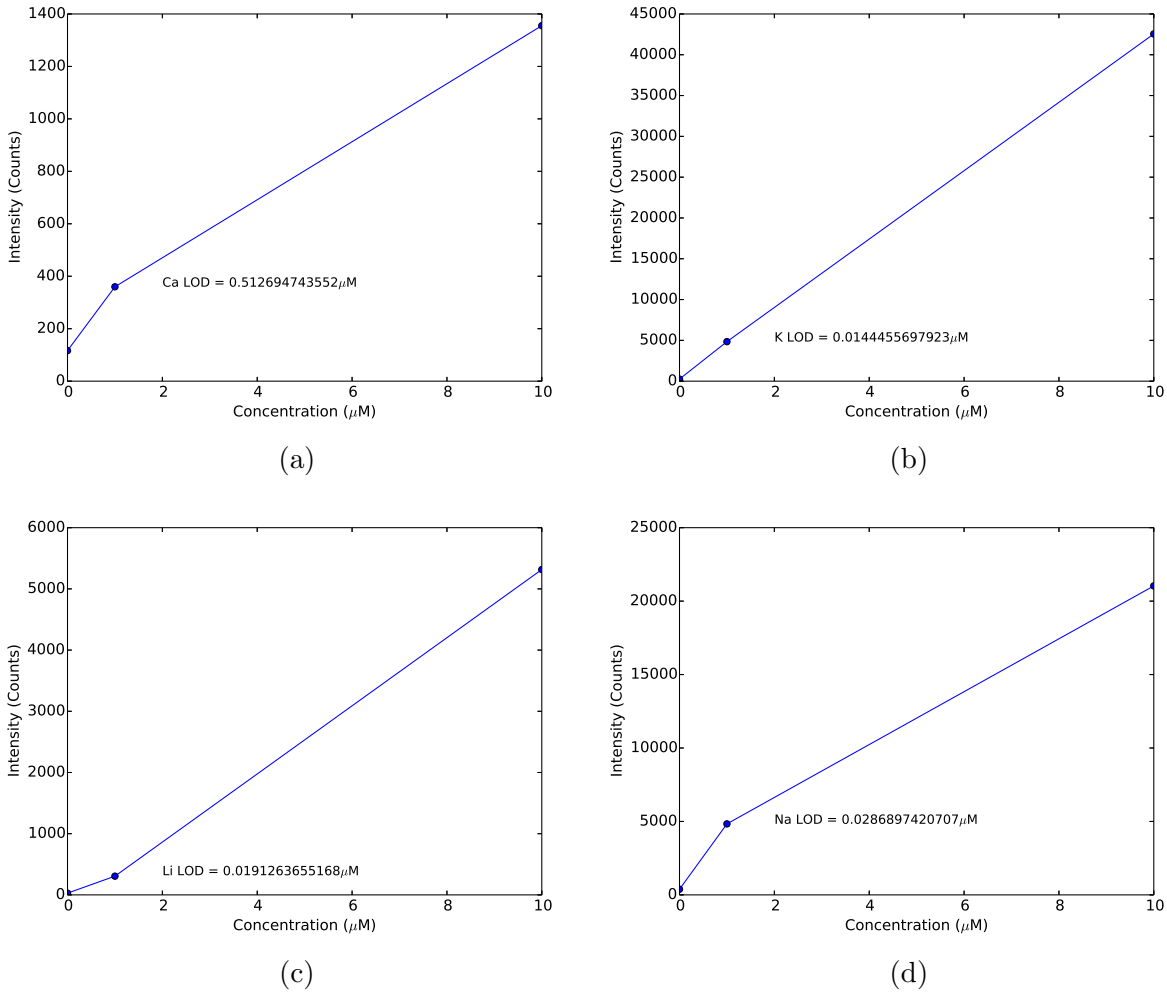


Figure 6.10: Mark 2: ETV-FAES LOD Calculations and Calibration Curves. This is a preliminary assessment at the calculations presume the data to be linear.

(a) Ca: LOD is 512.7 nM ± 16.626 nM; non-linear trend is suspected to be due to Self-Absorption.

(b) K: LOD is 14.4 nM ± 1.798 nM

(c) Li: LOD is 19.1 nM ± 3.403 nM; non-linear trend is suspected to be due to ionization interference.

(d) Na: LOD is 28.7 nM ± 4.600 nM; non-linear trend is suspected to be due to spectrometer swamping.

Table 6.3: List of LOD Peaks and Standard Deviations for Ca, K, Li, and Na

	0 μM	1 μM	10 μM	
Ca	116.455	359.578	1355.268	Peak
	25.27284	78.03493	294.1177	St. Dev.
K	254.0623	4841.427	42550.85	Peak
	161.2394	520.1997	292.2455	St. Dev.
Li	28.32533	305.972	5315.989	Peak
	24.75465	153.3921	83.384	St. Dev.
Na	384.4603	4833.196	21035.89	Peak
	304.8091	626.2475	826.1575	St. Dev.

In order to investigate, a full panel of concentrations were tested in triplicate for Ca. The randomized test order as well as the concentrations tested and peak heights are presented in Table 6.4. The resulting figure is presented in Figure 6.11. It is suspected, based off these preliminary results, that the non-linear trend of Ca is due to Self-Absorption, which causes a decrease in emission intensities as concentrations increase. [87] The non-linear trend of Li is peculiar; it is most likely due to ionization interference. [88] This could be confirmed by adding Cs or K to the sample; [88] however, these experiments were not performed.

Table 6.4: Tests, Concentrations, and Results for the Extended Ca Calibration Curve Testing

Test	Concentration (μM)	Average Peak (Intensity)
1	10	1355.268
2	1	359.578
3	50	3559.806
4	5	798.444
5	15	1653.956
6	20	1902.44
7	75	4224.055
8	0	116.455

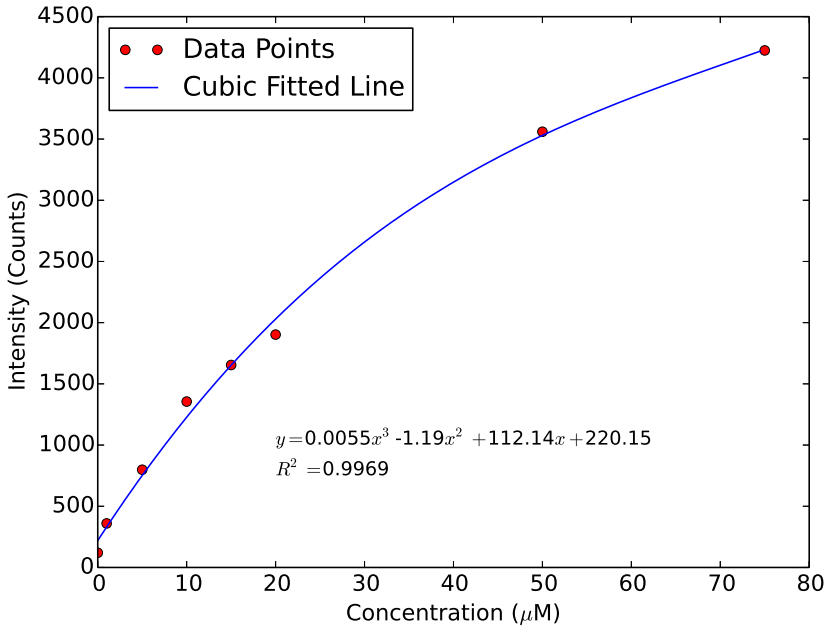


Figure 6.11: Extended calibration curve for Ca showing a cubic trend with an R^2 value of 0.9969. The equation of the cubic line is $y = 0.0055x^3 - 1.19x^2 + 112.14x + 220.15$.

The Cubic trend of Figure 6.11 is not what was expected. On first principle, the data should be linear within the working range of the spectrometer. Larger concentrations of an element should result in linearly proportioned increases in atomic emission. It is hypothesized that the non-linear trend has to do with Self-Absorption. [87]

The remaining three elements for which a calibration curve could not be generated, upon testing, demonstrated extremely broad peaks; if any peaks were present at all. As discussed in Section 2.5, it can be concluded that the temperature of the propane flame is not sufficient to properly excite these elements. If a nitrous-oxide acetylene flame is used it is highly likely these elements would be easily detected. [4, 18, 40]

A comparison between the ETV-FAES by Hanna et al., who uses a 2700 K flame source; [4] the Mark 2 ETV-FAES, and the Mark 1: Generation 3 FAES which both utilize a 2200 K flame source; and a representative commercial counterpart, the Sherwood Scientific Ltd. Model 360 Flame Photometer, which uses a 2200 K flame source and nebulizer, is presented in Table 6.5 for two elements which have been tested on all systems. Each of these systems introduce the sample to be tested at the base of the flame, if not prior to emission. As

can be seen, the detection limits are similar to, if not better than, the results obtained by Hanna et al. in 2011. [4] The results also show drastically improved detection capabilities over the current representative commercial counterpart offering from Sherwood Scientific Ltd., at the time of this writing. [41]

Table 6.5: Limits of Detection Comparison between the State of the Art ETV-FAES, the Mark 2: ETV-FAES, the Mark 1: Generation 3, and a representative commercial counterpart, the Model 360 Sherwood Scientific Ltd. instruments. The two elements selected were tested on all systems in order to provide a direct comparison.

Element	Wavelength (nm)	ETV-FAES [4] LOD (μM)	M2: ETV-FAES LOD (μM)	M1: Gen 3 FAES LOD (μM)	Flame Photometer [41] LOD (μM)
Li	670.8	0.0058	0.0191	3.0693	14.4071
K	766.5	0.0205	0.0144	1.0105	2.5577

6.6.4 Limitations

As previously discussed, a major limitation in the instrument is the temperature of the flame. The propane flame, at 2200 K, is significantly cooler than a nitrous-oxide acetylene flame at 3200 K. [18, 28] The state of the art instrumentation tend to utilize 2700 K or hotter flame sources to maximize sensitivity. [4, 28, 40] As a result, by using the cooler propane flame, the instrument is only capable of measuring elements with a low vaporization point such as Na, K, Ca, and Li. [18] As discussed in Section 2.3.4, the propane flame was selected as it was most convenient despite this limitation.

Another limitation, which became apparent during the testing for carry-over, is the swamping of the spectrometer. When this occurs the width of the peak becomes significantly large and effectively blocks the reading of other elements within close proximity. An example of this is Na (Peak at 589 nm) blocking Ca (Peak at 622 nm). This interference is common in all atomic emission spectroscopy, and must be addressed.

6.7 Simultaneous Multiple-Element Detection

One of the benefits of AES is that multiple element detection becomes possible. In order to verify this functionality with the current ETV-FAES, a multi-element sample was analyzed. This sample is comprised of minute amounts of Na, Ca, Li, and K, and the resulting spectra

can be seen in Figure 6.12. This figure confirms the capability of multi-element samples. It also clearly demonstrates an effective background and baseline correction for multiple elements.

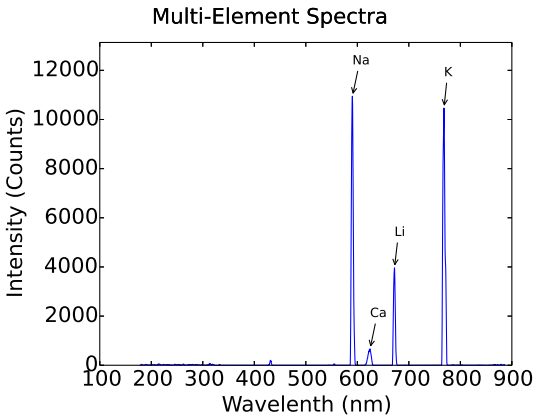


Figure 6.12: Multi-Element Spectra demonstrating simultaneous detection of Na, Ca, Li, and K.

In order to further analyze the capability and reproducibility of the instrument, a series of single-blind samples were generated by Dr. Chris Backhouse. These samples were run in triplicate and the corresponding intensity count was compared to the calibration curve in order to determine the concentrations initially used. A summary of the mystery samples, the actual concentrations, and the calculated concentrations are presented in Tables 6.6, 6.7, and 6.8. Note that the actual concentrations were given after the calculations were performed to determine the starting concentrations.

The concentrations were calculated as discussed in Section 6.5.5. The peak values were compared against a Linear Regression Analysis of the calibration curve to determine the starting concentration. Wavelengths used are as given in Appendix A.

Mystery samples $X1$ and $X2$ contained elements which were unable to be detected using the Mark 2 ETV-FAES. This is due to the temperature of the flame source not being sufficiently high. One of these additional elements is Mg. The addition of Mg, however, forms an interference in the expression of Ca; an investigation by Thompson and Ramsey in 1985 investigates the matrix interferences of Ca and Mg, which causes a suppression in their emission intensities of up to 30%. [89] As the calculations to determine the concentration are based upon the peak intensities, a 30% shift causes a significant concentration shift. The average peak intensity for sample $X1$ Ca is 3065.597 counts. A 30% increase in this

value corresponds to a peak intensity of 4379.939. This peak value in turn corresponds to a starting concentration of 80.718 μM. This matches very closely with what was expected. Unfortunately the inclusion of Mg in sample $X2$ drops the intensity of the Ca peak to below detectable levels. The saturation of the Na curves is likely due to slight Na contamination in the stock solutions used to generate the mystery samples.

Table 6.6: First Set of Mystery Samples, listing the elements, actual concentrations, and calculated concentrations after testing with the Mark 2: ETV-FAES

Sample: X1			Sample: X2		
	Concentration (μM)			Concentration (μM)	
Element	Nominal	Calculated	Element	Nominal	Calculated
Na	10	Saturated	Na	1	Saturated
Ca	70	37.79	Ca	10	0
Li	10	6.23	Li	10	0.717
K	10	10.445	K	10	0.365
Cu	10	Not Detected	Cu	0	Not Detected
Mg	10	Not Detected	Mg	10	Not Detected

The second set of mystery samples, $Y1$ and $Y2$, were created without the inclusion of either Cu or Mg to avoid the lack of detection and matrix effects. For both $Y1$ and $Y2$, the calculated concentrations for Li and K are within 1$\rightarrow$20% of the expected value. The slight variations could be due to errors in the pipette during either the sample generation stage, or sample testing. Unfortunately, the Ca concentration is again different than what is expected. Again, the saturation of the Na curves is likely due to slight Na contamination in the stock solutions used to generate the mystery samples. It could also be due to contamination of the ultrapure water source used for the mystery samples.

The third set of mystery samples, $Z1$, $Z2$, $Z3$, and $Z4$ are used to determine whether the water source used for the mystery samples is contaminated. Samples $Z1$ and $Z2$ were made using the same Milli-Q water source used for the stock solutions. Samples $Z3$ and $Z4$ were made using the same ultrapure water source used for mystery samples $X1$, $X2$, $Y1$, and $Y2$. As can be seen, the accuracy of the Ca concentration using the stock Milli-Q water source is significantly more accurate that those samples made with the secondary ultrapure source of Dr. Chris Backhouse. The detection of minute amounts of Na and K in $Z1$ and $Z2$ confirms minute contamination of either the stock solutions or sample holders with these two elements. Fortunately samples $Z3$ and $Z4$ again show a 5$\rightarrow$20 % variation in detected concentration levels from the expected values for K and Li.

Table 6.7: Second Set of Mystery Samples, listing the elements, actual concentrations, and calculated concentrations after testing with the Mark 2: ETV-FAES

Sample: Y1

Element	Concentration (μM)	
	Nominal	Calculated
Na	2	Saturated
Ca	5	1.067
Li	5	3.668
K	5	3.96

Sample: Y2

Element	Concentration (μM)	
	Nominal	Calculated
Na	0	Saturated
Ca	4	2.468
Li	5	3.686
K	2	1.971

Table 6.8: Third Set of Mystery Samples, listing the elements, actual concentrations, and calculated concentrations after testing with the Mark 2: ETV-FAES

Sample: Z1

Element	Concentration (μM)	
	Nominal	Calculated
Na	0	3.679
Ca	8.3	6.848
Li	0	0
K	0	1.492

Sample: Z2

Element	Concentration (μM)	
	Nominal	Calculated
Na	0	5.524
Ca	4.1	4.998
Li	0	0
K	0	1.824

Sample: Z3

Element	Concentration (μM)	
	Nominal	Calculated
Na	0.7	Saturated
Ca	4.1	8.4976
Li	0.7	0.877
K	0.7	0.8837

Sample: Z4

Element	Concentration (μM)	
	Nominal	Calculated
Na	0	Saturated
Ca	4.1	2.148
Li	40	38.022
K	40	Saturated

This system shows promise, and, although suffering some reliability issues, has demonstrated good sensitivity in a way that could be integrated with LOC. The reliability problems are very significant and likely arise from some combination of baseline variations and contamination. The consistently saturated values for Na for all samples derived from the 'ultrapure water' suggests that the 'ultrapure water' was contaminated. It was also observed that sporadic flashes of signal can arise spontaneously, possibly due to silicate particulates from the ceiling fire retardant coating. In hindsight, it is clear that there should have been ongoing validation testing of the various systems built in this work with reference samples that were also assessed by another reference system.

A future redesign of the system should assess baseline stability with and without filtered air and monitor for contaminants, likely with the active and ongoing use of a reference system such as Inductively Coupled Plasma-Atomic Emission Spectroscopy (ICP-AES). Although preliminary in nature, this demonstration did show that the ETV is capable of managing small samples in a way that will enable LOC integration.

6.8 Conclusion

The ETV-FAES designed and fabricated herein has a total cost, minus the spectrometer and computer, of less than $500. The system also demonstrates reproducibility and the capability to analyse multiple elements simultaneously. The addition of the ETV to the FAES significantly improves the sample introduction efficiency over the ultrasonic nebulizer previously used, which in turn improves upon the detection capabilities of the system.

As previously discussed, a major limitation in the instrument is the temperature of the flame. The propane flame, at 2200 K, is significantly cooler than a nitrous-oxide acetylene flame at 3200 K. [18, 28] As a result the instrument is only capable of measuring elements with a low vaporization point such as Na, K, Ca, and Li. [18]. Others have also noted that a hotter flame is required for increased sensitivity for elements such as Cu. [4, 18, 40] Unfortunately, a different flame source would require a new generation of the ETV-FAES due to the requirement to change the flame nozzle to support the higher temperature flames. The transition to other torch nozzles was deemed too large a financial burden for the initial development of a LOC compatible ETV-FAES. The Propane gas was utilized for its convenience and simplicity in this preliminary system.

Another limitation, which became apparent during the testing for carry-over, is the swamping of the spectrometer. When this occurs the width of the peak becomes significantly large and effectively blocks the reading of other elements within close proximity. An

example of this is Na (Peak at 589 nm) blocking Ca (Peak at 622 nm). This interference is common in all AES, and must be addressed.

Chapter 7

Removing Interferences in Atomic Emission Spectroscopy

7.1 Introduction

Flame Atomic Emissions Spectroscopy (FAES) is a powerful spectroscopic technique that has been used to detect trace levels of over 60 different elements [90–94]. These detections, however, have needed to be individually optimized and isolated to achieve sufficient detection levels and avoid interferences, spectral or otherwise [48, 92]. Interferences, if present, hinders accurate analysis and can be detrimental in sensitive applications [7–9, 48, 87, 93, 95–100]. Therefore, in order to circumvent these issues, the interferant(s) needed to be removed by techniques such as solvent extraction or ion exchange [93, 101]. This is not only elaborate and time consuming, but, depending upon the sample of interest, not always possible. As a result, despite being the one of the oldest chemical analysis techniques [1, 87], the use of FAES in an industrial setting is primarily limited to the detection of Na, Ca, K, Li, and Mg within simple samples [11–17].

The benefit of FAES is that it is relatively simple and low-cost instrumentation with the ability for field-deployment for on-site analysis in numerous fields [4, 9]. Its emissive nature is also conducive to simultaneous multi-element detection, apart from the issue of interferences, in contrast to absorptive techniques [92]. With the introduction of the Tungsten Coil (W-Coil) ElectroThermal Vaporizer (ETV) in 2011 as an effective means of drying and delivering the sample, overall sensitivity was increased 100 fold [4]. However, we demonstrate that the ability to do this in a temperature selective manner can be used to overcome a central challenge to the use of FAES by analysing the various components of the sample separately.

We note a similar idea has been followed by the Inductively Coupled Plasma-Atomic Emission Spectroscopy (ICP-AES) work of Hou et al. [102]. Hou et al. resolved a previously highly interfered sample by using a temperature ramp to partially separate the components. Although the method of ramping the temperature worked to resolve the components, there was a significant loss of sensitivity. The Limits of Detection (LODs) were 10x worse in analyzing the complex sample as compared with simple samples. In the present work we present a method to selectively remove a specific component in a manner, that to first order, preserves our LOD.

7.1.1 Interferences

The challenge of interferences in FAES is ongoing [6–9, 48, 90, 93, 96, 97, 100]. As described by one of the standard texts in the field, Spectrochemical Analysis [87], they can arise from one of many sources. Spectral interferences occur when any interferant emits

within the line width of another element. Whether this is observed or not is directly dependent upon the composition of the sample and the excitation technique [87]; the higher the excitation energy the more likely a spectral interference will occur. Generally, spectrometers with higher spectral resolution will, in principle, reduce these effects [87]. Some elements, however, have large effective line widths due to one or more of the four sources of line broadening: the uncertainty effect, the Doppler effect, pressure effects, and/or electric and magnetic field effects [95]. Thus higher resolution spectrometers will not always resolve a spectral interference. Matrix interferences, however, are often called specific or chemical interferences. These interferences are the result of compounds within the samples themselves that either suppress or enhance the emission of the analyte [87]. Similar to spectral interferences, whether or not this is observed is dependent upon the sample itself.

Some examples of interferences include the classical spectral interference between Na and Ca [6] and the prominent matrix interference between Al and alkaline earth metals [7–9, 93, 96, 97].

7.1.2 FAES Background

As described by [27, 103, 104], FAES typically uses a nebulizer which aerosolizes the liquid sample of interest. This usually results in only 3-12% of the sample actually reaching the excitation source. As such, other methodologies have been utilized for aerosolizing the sample including electrothermal devices. These devices have been used for over four decades and can be made out of materials such as graphite, molybdenum, or tungsten [40]. They can be used in one of two arrangements: as both the atomizer and excitation source; or as just the atomizer, vaporizing the sample. When used as the latter, it is considered an ETV. The vaporized sample is then swept into the excitation source by an inert gas [40].

The inert gas serves two main purposes: to reduce the oxidation of the ETV, and to transport the vaporized sample to the flame [4, 40]. Therefore the selection of the gas is important as it can have an effect on the resulting LOD. Generally, Ar provides the best protection but the inclusion of H improves the reproducibility of the measurements [40, 105]. Therefore a mixture of 90%Ar-10%H has been used with positive results [27, 40, 54].

As described above, the addition of an ETV to FAES has been used to improve the LOD of the technique. However, to our knowledge, the methodology discussed thus far in literature vaporizes the sample as a single plug to be injected into the atomizer [4, 27]. This serves the purpose of replacing the traditional nebulizer, but does not take full advantage of the ability to selectively release a component into the flame by controlling the temperature of the ETV.

7.1.3 Present Work

In the present work we combine the use of a W-Coil ETV with a FAES, both developed in-house, to successively deliver components of a complex sample that are evolved at progressively higher filament temperatures. This allows for the conversion of a single complex sample into a series of easily analyzed simple samples. This approach circumvents the need for external sample preparation as a means to avoiding interference.

We will demonstrate this by monitoring a weak signal from Calcium Hydroxide (CaOH) in a strong signal of Na, an interferant. These elements, Ca and Na, have a 600 K difference in boiling points and been reported to interfere with each other [6, 91]. By maintaining the temperature of the ETV above the boiling point of Na and below the boiling point of Ca we should be able to separate the elements and eliminate the spectral interference with minimal loss of sample. To our knowledge this is the first use of an ETV to solve this fundamental challenge to the use of a Lab on Chip (LOC) relevant implementation of FAES.

7.2 Method and Materials

Table 7.1: Stepped Analysis Testing Procedure

	Step	Current (A)	Temperature[a] (K)	Time (s)
1:	Drying	3	<373[b]	Variable
2:	Cooling	0	273	10
3:	Interferant Evolution at T_A-100K	4.3	1650	45
4:	Cooling	0	273	10
5:	Analysis	8	2797	10
6:	Cooling	0	273	10
7:	Cleaning	>10[c]	>3400	10
8:	Cooling	0	273	60

Notes:
[a] Temperature estimated by Equation 7.1.
[b] The W-Coil is wet, therefore the temperature must be below 373K. [84]
[c] 12 Volts (V) is held rather than a current due to the negative Temperature Coefficient of Resistance (TCR) of the W-Coil.

Typically when an ETV is used in conjunction with FAES, a basic dry-vaporize methodology is used. This serves the purpose of vaporizing the entire sample as a single plug to be introduced to the excitation source [4]. While this works well for low concentrations of elements and for those elements which do not have any confounding effects, as soon as an interferant is present within the sample the methodology begins to fail. This direct analysis approach is described in detail by Hanna et al., and can be summarized as: a basic drying step at 3.4 Amperes (A); cooling for 10 seconds; then vaporizing and analyzing the spectrum of the sample at 8 A for 10 seconds with a 60 second cooling in between tests [4].

The method proposed herein resolves around the vapor pressures of the elements within the complex sample of interest. For simplicity the boiling points of the elements are used as a good approximation for the temperature of evolution. Once the sample is taken to be dry the interferant(s) are removed by setting an interferant evolution temperature 100K below the boiling point of the target element (T_A). The target is then evolved and analysed through a subsequent vaporization, as described previously. For the examples mentioned previously: using Ca and Na with boiling points (T_b) of 1757K and 1156K respectively, the evolution temperature ($T_A - 100K$) is set to 1650 K; and for Al ($T_b = 2740K$) and Ca, an Alkaline Earth Metal, the evolution temperature is set to 2640 K such as to evolve the Ca and not the Al interferant. The spectrums during each evolution step are recorded for analysis.

The temperature of the W-Coil ETV is estimated through the use of Levine et al.'s model (Equation 7.1) which is valid for currents between 4.5 A (1715 K) and 8 A (2797 K) [106]. It is assumed that the linear trend of the model continues down to 0 A, and this assumption has been used successfully by others [107]. Using this model, the protocol listed in Table 7.1 was developed.

$$T(K) = 309I + 325 \tag{7.1}$$

Much as described by [4, 84], the liquid sample on the W-Coil causes a reduction in resistance; as the sample is dried the resistance increases causing a sharp rise in voltage. This sharp voltage rise is detected and at this point the filament is taken to be dry. Typically a final voltage of 1.2 $\pm$ 0.15 V is observed across the filament. The software encodes this first step and, once dry, performs all remaining steps of Table 7.1. Although the model in Equation 7.1 dictates a temperature of 1252 K, the liquid keeps the filament below 373 K as it is drying and the software detects and stops the rapid temperature increase of the filament once dry. This drying step takes a variable amount of time to complete.

Once the sample is dried, the filament is cooled for a period of 10 seconds by the flowing of forming gas (90%Ar-10%H) at 273K. This cooling step is repeated every other step in order to allow for the filament to return to room temperature, which improves the accuracy and reproducibility of the set temperatures at each step due to the negative Temperature Coefficient of Resistance (TCR) of the W-Coil [108].

The third step of the protocol, the interferant evolution or thermal separation, involves the selective removal of a portion of the sample. The temperature for this separation procedure is set to 100K below the analyte of interest's boiling point ($T_A - 100K$). In the case of Ca, the selected temperature to remove any interferant is 1650 K, corresponding to a current of 4.3 A. This temperature is held for a period of 45 seconds to ensure the majority of any interferant with a boiling point below this temperature is removed.

Conversely, for the Al and Ca example, this thermal separation step may be used to evolve the Ca for analysis without the Al interferant. However, for the demonstration the interferant is being evolved.

The sample analysis occurs after a cooling step as the remaining sample is vaporized as a single sample plug at 8 A (2797 K) for a period of 10 seconds in accordance with traditional testing [4]. However, additional separation procedures may be performed prior to this step, as needed.

The cleaning step is performed by exposing the ETV to a continuous 12 V at up to 300 Watts (W) for a period of 10 seconds. This cleaning procedure proved to be more effective at eliminating carry-over effects than maintaining a constant current, again due to the negative TCR of the W-Coil [108].

7.2.1 Apparatus and Software

The apparatus used in this work consists of an ETV fabricated similarly to that described by Hanna et al. in 2011 [4]. The ETV is powered by a laboratory designed power supply capable of delivering up to 150 W (15 V at 10 A) Direct Current (DC) in a controlled manner, and over 300 W at 12 V DC. This power supply is controlled by a customized python program running through a LabJack U6. A pair of high-power relays are used to toggle the ETV between the controlled 150 W and the full power 300 W rail, which is used to clean the filament between trials. This electronic system is summarized in Figure 7.1. A forming gas consisting of 90%Ar-10%H was used to carry the vaporized samples to a standard air-propane flame (Bernzomatic TS4000). The distance between the ETV and flame is less than 1 cm. The resulting spectrum is collected by a series of optics focusing

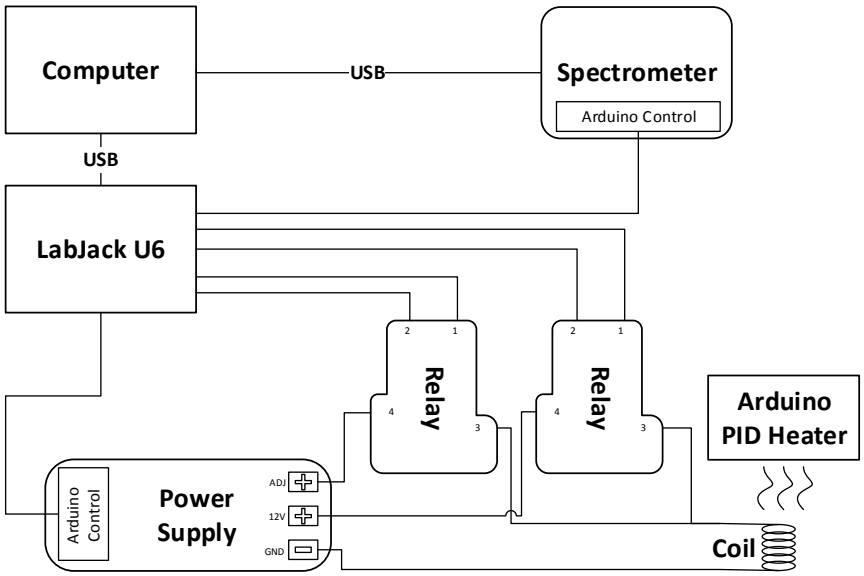


Figure 7.1: ETV-FAES Control Electronics.

the light onto a fiber optic cable and into a Charge Coupled Device (CCD) spectrometer (Ocean Optics USB4000). This collection is exemplified in Figure 7.2.

7.2.2 Power Supply

For this application, a high power stability is desired in order to minimize temperature fluctuations on the filament. This becomes extremely important when vaporizing individual components of a sample. Ideally the fluctuations would be nonexistent, however, a fluctuation of only 10 mA on the 10 A rail corresponds to a temperature change of 3K. Therefore a power supply design capable of high resolution control was developed. The present design was based on one for a 1 A laboratory power supply [66], but was modified to use higher current regulators (Linear Technology LT3083) in parallel to attain the needed current range of 0-10 A. A 12-bit Digital to Analog Converter (DAC) (MCP4922) is used to control this current in 2.5 mA steps. Furthermore, the voltage output from this power supply can be controlled between 0-12.288 V. Again a 12-bit DAC is used to allow 3 mV steps. Unlike the original laboratory supply, an Arduino was not used for control and instead the LabJack U6 was selected for python integration.

Table 7.2: Summary of Samples used to Validate the Stepped Analysis Testing Procedure. Each sample has a total volume of 10 μL. The K component is due to contamination of the samples; its concentration is estimated based upon the peak intensities and using the calibration curves of Chapter 6.

	Component A		Component B		Component C	
Sample	Element	Concentration	Element	Concentration	Element	Concentration
S1	Ca	10 μM	Na	154 mM	K	9.3 μM
S2	Ca	10 μM	Na	22.3 μM	K	7.2 μM

A High-Side Current-Sense (HSCS) (MAX4080F) is used to monitor the voltage drop across a 0.1 Ohm (Ω) 20 W equivalent resistor. A voltage drop across the resistor of 0-1 V corresponds to a 0-10 A draw by the load. This voltage drop is converted into a 0-5 V signal by the HSCS chip which is measured by a dual-channel 12-bit Analog to Digital Converter (ADC) (MCP3202). This is performed on the power supply to provide feedback to the python program as to the real-time current-draw of the load. How the HSCS functions is well documented in the data sheet. [109]

7.2.3 ElectroThermal Vaporizer

Much as described by Hanna et al. [4], the ETV was fabricated using off-the-shelf components. An aluminum tube cut to the appropriate size was threaded such that brass fittings could be used to fully enclose the filament, and provide a connection to the forming gas tubing. The ceramic mount for the W-Coil was centered using four set-screws and holes were drilled through the aluminum for the power cables. The power cables are terminated in a connector which allows for easy assembly and disassembly of the thermal system and power supply. A large diameter bolt has been added in order to expose the filament for manual sample application with a micropipette.

The W-Coil (10-turn filament; 150 W, 15 V Osram Xenophot HLX 64633) is carefully extracted from its glass housing and placed into its ceramic mount. This filament has the capability to reach over 3000 K which in turn causes significant heating of the ETV housing if left unchecked. This is readily seen in Section 3.5.1 as the temperature of the housing is monitored as the filament is heated.

Others have also used a coil for supporting the sample to be vaporized, however, they utilized a Re wire and fabricated the assembly in-house rather than utilizing a commercially available coil. This has the benefit of being able to modify the coil volume capacity. [110]

7.2.4 Optics

The optical components (collection lens and optical fiber lens) were mounted on micro-positioners such that the focal point could be shifted for optimal light collection. These micro-positioners are mounted to a PVC frame custom fabricated on a Computer Numerical Control (CNC) Milling Machine. Also on this frame is a mount for the blow-torch, to maintain a consistent flame position, and a mount for the ETV. The ETV is held at a 45 degree angle to the flame as depicted in Figure 7.2. As the vaporized components of the sample enter the flame, the color change is captured by the optics and focused into an optical fiber which relays the light to the spectrometer.

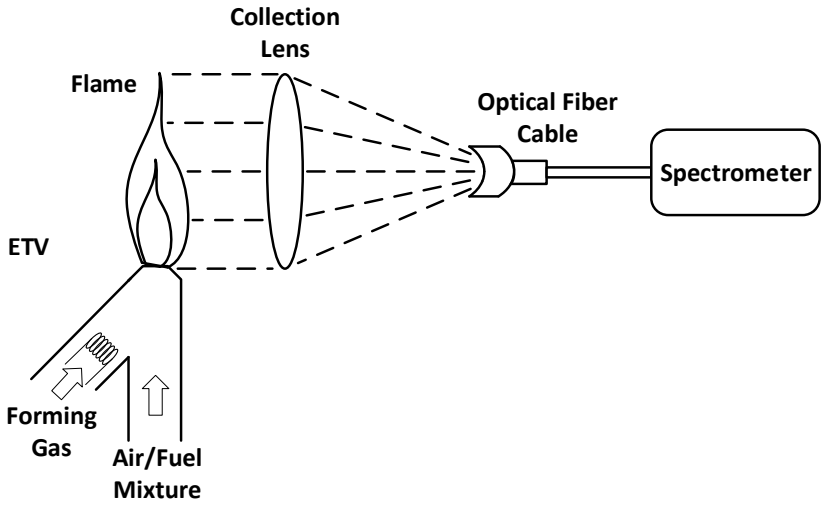


Figure 7.2: ETV-FAES Physical Setup and Optical Collection

7.2.5 Python Software

The Python software, which controls the drying and subsequent baking of the sample, fully automates the testing procedure in order to ensure repeatable testing. This program also triggers the spectrometer to collect the emission spectra for subsequent analysis. A Graphical User Interface (GUI) for the program was created for simplicity. Utilizing the temperature model (Equation 7.1) of the W-Coil, approximate temperatures can be maintained on the filament [106, 107, 111]. This program makes use of the real-time feedback from the power supply as to the load provided by the W-Coil ETV to adjust the output

power. It also uses this information to estimate the temperature of the filament. For safety, the program ensures only one relay is active at a time, and at any time the user may kill power to the entire setup. During cooling steps, and when the filament is not being actively heated, both relays are off such that zero power flows across the ETV.

7.2.6 Coil Preparation

In order to prepare a freshly exposed W-Coil, a new coil conditioning and cleaning step was added to the program as documented by Donati et al. in 2008 [84]. These procedures serve to "break in" the new coil by both adding and removing an oxidation layer by operating the coil in open air and forming gas respectively. In contrast to Donati et al., rather than maintaining a current of 10 A during the oxidation removal, 12 V at up to 300 W was supplied to the W-Coil. This step proved to be more effective at eliminating any residual oxidation.

7.2.7 Samples

Stock solutions of 0.1 mol L^{-1} (M) Sodium Chloride (NaCl) and Calcium Chlorate Di-Hydrate ($CaCl_2 \cdot 2H_2O$) were made from Milli-Q water and 1:10 serial dilutions were performed to reduce the concentrations to workable levels. The final sample used for testing the spectral interference between Na and Ca (CaOH) consisted of 10 μL of both solutions at concentrations of 0.1 M and 1 mM respectively. Within these samples some K and Ba contamination existed and, therefore, will be included in the analysis. This contamination is not due to carry-over but the stock solutions themselves as investigated in Section 6.6.2. Table A.1 in Appendix A lists the characteristics for the elements of interest, and Table 7.2 lists the sample mixtures used. The high concentrations were used in order to fully exemplify spectral interference between CaOH and Na and the utility of the proposed testing methodology.

7.2.8 Reliability Testing

The preliminary system used for the demonstration is still being characterized, however, it was tested for the repeatability of both direct and stepped analysis procedures for the demonstration. This was carried out by performing statistical analysis on the Ca peak strength of the S2 emissions spectra over the course of 24 tests randomly performed with either the direct or stepped analysis procedures. In between tests, residual carry-over was

tested for by running a blank and monitoring the resulting spectrum. In addition, further testing of the carry-over effect was performed by the direct analysis and subsequent cleaning of a 1 M Na sample. Again, the resulting spectrum was analyzed after running a blank.

7.3 Results and Discussion

The preliminary system used herein for the demonstration was tested for analytical error for both the direct and stepped analysis procedures encoded in the software. It was found that the signal intensity is accurate to 14.62% for direct analysis and 17.56% for the stepped analysis. Although less accurate, the proposed methodology may circumvent the need for external sample preparation for FAES while preserving, to first order, the LOD. This higher error is most likely due to the simplified model of the W-Coil which assumes a consistent temperature across the entire filament. However, as the W-Coil heats, the center will be at a much higher temperature than the edges resulting in a thermal gradient. Furthermore, the cool gas flowing over the filament will increase this temperature gradient. In addition, as the coil heats, the sample will dry at the center of the coil before drying at the edges, leading to larger elemental deposits at either extremes. The software encoding this variable process enables the accuracy presented.

The blanks performed between tests to obtain the repeatability of the instrument demonstrated no significant signal between 200-800nm indicating zero carry-over for the lower concentrations. The more rigorous testing, however, revealed a spectrum indicating residual Na, however, following a second cleaning procedure the spectrum indicated zero carry-over. Although some residual Na remained after the first cleaning, the extremely high Na concentration is not indicative of samples which would typically be tested with a FAES. Prolonged cleaning at 12 V does resolve this at the expense of substantial heating of the ETV housing. This temperature is monitored as discussed in Section 3.5.1 in order to allow for consistent start temperatures for repetitive runs. The effects of the increased housing temperature and alternative cleaning procedures are under investigation along with the LOD. We believe the LOD of the instrument is comparable to others, however, the LOD has no impact on the stepped analysis procedure used in the present work.

The samples listed in Table 7.2 were tested by following both direct and stepped analysis procedures. Representative captured raw spectra (background removed) corresponding to these tests can be seen in Figures 7.3 and 7.4. The initial Ca signal, evident in Figure 7.3, is completely confounded by the addition of high concentration Na to the sample (Fig 7.4a). In addition, the signal intensity from the Na is sufficient enough to saturate the CCD causing erroneous readings. By performing the stepped analysis, the confounding

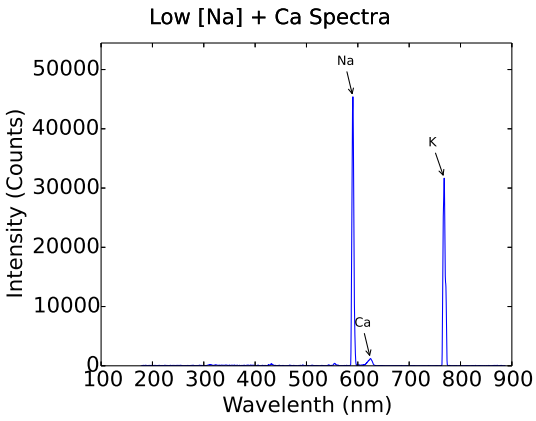
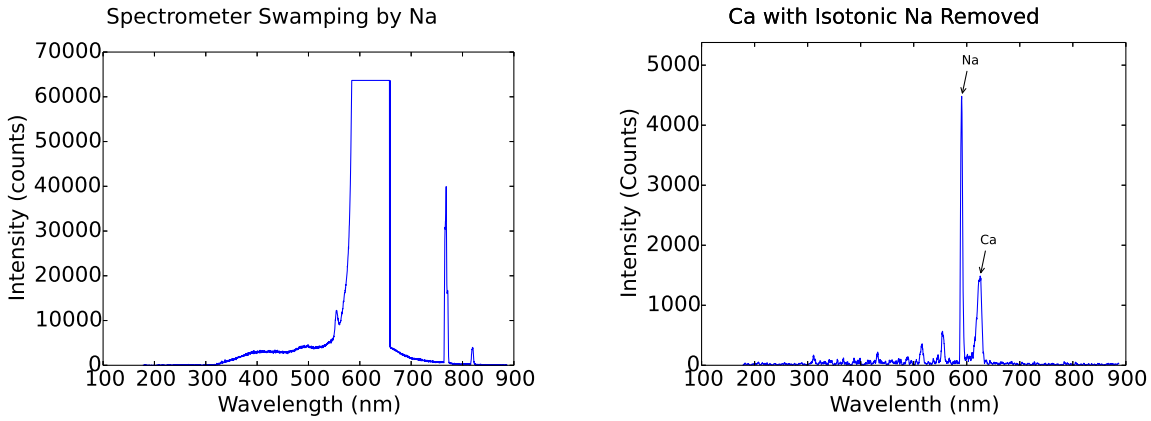


Figure 7.3: The direct analysis of S2, without the interferant, revealing Ca (1223 counts), K (31666 counts), and Na (45399 counts) peaks. The low concentration of Na does not confound the measurement of Ca. S2 is comprised of 10 μM Ca, 22.3 μM Na, and 7.2 μM K.

concentration of Na is significantly reduced such that it no longer plays a significant role in the detection of Ca (Fig 7.4b). In addition to the reduction of Na, the K peak is also removed.

Following the stepped analysis procedure, the remaining Ca concentration in S1, after removing the Na from the sample, differed by an average of 7.53% from the Ca concentration in S2 over multiple trials. This change in Ca signal intensity between the two methods is comparable to the error on the measurement itself ($\pm$3.84%) and is significantly less than the 10 fold reduction experienced by Hou et al. with ETV-ICP-AES [102]. Furthermore, the stepped analysis was able to recover the Ca signal in the presence of the interferant which would otherwise not be possible without external sample preparation.

Although demonstrated with elements that have a 600 K difference in boiling points, the successful reduction of Na without significant change in Ca signal strength is significant. Out of select biologically significant elements of interest in Table A.1, one might expect to perform the same enhancement with the: separation of Zinc (Zn) and Mg from Mn and Silver (Ag); and separation of Mn and Ag from Cu and Iron (Fe). In addition, the selective testing of the majority of the Alkaline earth metals may proceed unhindered, even in the presence of Al, by carefully choosing the temperature of vaporization. Following this, the concentration of Al in these samples could be analyzed. Unfortunately, a higher flame temperature is required in order to test these elements.



(a) (b)

Figure 7.4: Attempted quantification of Ca within a sample containing an isotonic concentration of Na, an interferant. The numbers in brackets are the peak intensity counts. S1 is comprised of 10 μM Ca, 154 mM Na, and 9.3 μM K.

(a) Shows the direct analysis of S1, with the isotonic Na (>60000) interferant, showing that the Ca signal has been completely obscured. K is present.

(b) Shows the stepped analysis of S1, with the isotonic Na (4481) interferant, showing a Ca (1438) signal that is comparable to that of the sample without the interferant (a). Here the K has also been removed.

It should be noted that this technique also has the advantage of facilitating a two dimensional analysis by monitoring both the temperature of the ETV as well as the emission peak(s) simultaneously. This could aid in the identification of an otherwise confounding contaminant. If the speed of capture was increased, a three dimensional analysis could be performed such as that done by Weagant et al. in 2011 with a microplasma. [112]

7.4 Conclusion

The methodology described in this work demonstrates the feasibility of reducing, if not eliminating, the interferences limiting the use of FAES without external sample preparation and with insignificant loss of sample. By selectively vaporizing the element(s) based upon their boiling points, an otherwise confounded concentration of an element of interest can be effectively analyzed. This is possible by the conversion of a complex sample into a series of simple samples successively delivered to the excitation source for analysis. To our knowledge this is the first use of the ETV to solve this fundamental challenge to the use of FAES which presently limits its use for a variety of applications.

Demonstrated for the classical spectral interference between Na and Ca, the technique might also lend itself to the resolution of the prominent Al interference for the majority of the Alkaline earth metals with simple modifications to the thermal separation step(s). Furthermore, the modified testing procedure presented herein makes use of the advantages provided by W-Coil ETV devices for sample introduction with only software modification.

The procedure, which is based upon the use of the W-Coil ETV, is not exclusive to FAES. The sample introduction methodology has been applied to other spectroscopic techniques which utilize an external excitation source. Such instrumentation includes ICP-AES [102, 110, 113, 114], and microplasmas. [112, 115]

Chapter 8

Conclusion and Future Directions

8.1 Contributions Relative to State of the Art

This research focussed on developing a low-cost, portable, ElectroThermal Vaporizer (ETV)-Flame Atomic Emissions Spectroscopy (FAES) for Lab on Chip (LOC) applications. The research is similar in nature to the work performed by Hanna et al. in 2011, but differs in execution. [4] The current work utilizes a set of Omega heaters on the ETV enclosure to decrease required cool-down times between samples, resulting in a higher throughput. In addition, we utilize a blowtorch that is over one order of magnitude less in cost than that used by Hanna et al., while obtaining comparable Limits of Detection (LODs). [4]

The use of an ETV is not novel with regard to sample introduction and separation of interfering elements. Previously, Hou et al. resolved an interfering sample by using a temperature ramp to partially separate the components. Although the method of ramping the temperature worked to resolve the components, there was a significant loss of sensitivity. The LODs were 10x worse in analyzing the complex sample as compared with simple samples. [102] In the present work, we introduce a method to selectively remove a specific component in a manner that preserves the LOD.

The entire vaporization and thermal source setup could likely be integrated onto a single microfluidic chip, with a small resistive Tungsten wire embedded in the chip itself; similar to [116], but with higher temperatures. This wire could be mounted in a similar fashion as the Osram Xenophot HLX 64633 used here. The flame, however, would more than likely need to be replaced with a substitute technology. Substitution with the controlled microscale explosives presented by Sailor et al. in US Patent 7,942,989 may be possible. [117] This patent specifically discusses the use of such technology in the analysis of elemental samples in a similar manner to conventional Atomic Emission Spectroscopy (AES).

A partial integration may, however, be more suitable. It could be possible to utilize a similar method to that used by Nagy et al. in 2014 to couple the chip to a flame. [118] Although this used a nebulizer, which by definition has an active suction, a similar concept would be possible. The sample of interest could be vaporized on the chip as previously discussed and transported in its gaseous form to the flame.

8.2 Conclusion

In this work an ETV-FAES, which uses LOC-like sample volumes of 10 microlitres, has been developed for LOC applications. Although there are some reliability issues as discussed in Chapters 5 and 6, the instrument demonstrated detection capabilities comparable to the

state of the art with an inexpensive propane flame source, and was used to characterize a variety of single-blind samples. In addition, a new sample test protocol has been developed for the instrument which reduces, if not eliminates, interferences within samples without external sample preparation. This protocol is validated by detecting an extremely weak Ca signal in the presence of isotonic Na, a situation which is known to be problematic. [6]

The hardware required to fabricate the instrument was segregated into multiple independent modules, which, when combined, forms an infrastructure which can be used for an ETV-FAES, a LOC Micro-Total-Analysis System (μTAS), or a joint LOC ETV-FAES-μTAS. These hardware modules have been developed utilizing freeware, and programmed using Python 2.7. Although this thesis focuses on the design, development, and characterization of the ETV-FAES for LOC applications, these hardware modules are designed to work together simultaneously allowing for a LOC ETV-FAES-μTAS. This enables the Point-of-Care (POC) diagnosis of trace-element deficiencies and related diseases for the medical industry.

The preliminary demonstration of the ETV-FAES shows comparable, or better, performance to the commercial state of the art offerings. In addition, the present ETV-FAES system is able to address some of the challenges in the field; Notably the small volumes needed for LOC, and the interferences of complex samples. The preliminary LODs are calculated to be sub micromolar. With the total cost of the instrumentation kept below $500; excluding the spectrometer and computer, this represents a disruptive technology, at a significantly lower price point, in comparison to comparable commercial offerings on the market today.

The three objectives set out at the beginning of this research have been met: The field of FAES has been enhanced by designing a LOC POC compatible FAES; The cost of the instrument has been kept low to maximize its availability to the masses; And, the FAES technique has been improved by addressing and solving a fundamental limitation of the technique: spectral interferences within complex samples.

To conclude, the system presented herein is one that has reached the stage of preliminary testing and has shown significant promise, and although suffering from some reliability issues, has demonstrated good sensitivity in a way that could be integrated with LOC. In addition, the ability to selectively vaporize elements in order to avoid spectral interferences could be quite powerful. The Propane gas was utilized for its convenience and availability in order to perform this feasibility investigation; and future redesigns of the system would do well to utilize a hotter thermal source.

8.3 Improvements

The ETV-FAES has one major limitation which hinders its capabilities. This is the temperature of the flame source. It is well documented that an increase in temperature corresponds to higher sensitivity as well as an increased range of elements which can be analyzed. A future version of the instrument would be well suited to utilizing a higher temperature flame such as Oxygen-Acetylene or Nitrous Oxide-Acetylene. This temperature increase would, unfortunately, come at an increased cost of the instrument and a re-design of the FAES.

8.4 Suggested Next Steps

Moving forward with the ETV-FAES, it is suggested that a further investigation into the Ultrapure Water versus Milli-Q in multi-elemental samples is investigated. Furthermore, the sample introduction technique should be investigated to see if it can be used to remove highly complex interferences such as the matrix effects between Ca and Mg. There was not enough time to begin and complete these investigations throughout this research.

An investigation into the suspected Self-Absorption and Ionizing Interference experienced for Ca and Li is highly recommended. It is important to understand this phenomenon in order to fully characterize the system.

Another suggestion is to exchange the LabJack for a Beaglebone or Raspberry Pi. These Linux microcontrollers have the GPIO as well as the communication protocols such as Universal Asynchronous Receiver Transmitter (UART) and Serial Peripheral Interface (SPI) to provide true hardware-based functions to the connected hardware. Furthermore, the code may be written in Python and could eliminate the need for a computer. This will significantly reduce the entire cost of the system while improving performance.

8.5 Future Directions

Merging the ETV-FAES with the μTAS does have some complications to consider. Although the hardware has been designed from the beginning to work in unison, the fluidic requirements need to be investigated. Microfluidic pumping is a capability, and the Solenoid Board module is designed specifically for it. The question is how to effectively pump microlitre samples onto the Tungsten Coil (W-Coil) filament of the ETV, thereby

coupling to a LOC device. Another consideration is how quickly the samples could be dried; presently on the Mark-2 it takes just over a minute to fully dry a 10 μL sample.

The pumping of the sample could be achieved by adding a significantly sized buffer well which can be used to dilute the sample on the microfluidic chip itself after the sample has been tested on the μTAS. This would drastically increase the volume allowing the fluid to traverse what could be a relatively long channel to get to the filament. The pumping of this fluid is taken care of by the Solenoid module. A stainless steel needle tip could be used to couple the chip to the W-Coil, with the sample being dripped onto the filament. The tight filament coils, along with the surface tension of the fluid, would be capable of catching the droplets. Several repeated drying steps may be used to completely remove the transport medium in order to preserve the concentrations of the original sample being tested.

As the ETV vaporizes the sample, it would significantly heat the stainless steel. The forming gas should, however, be sufficient to keep the temperature of the needle below its melting point. It is possible that the high temperature may be enough to eliminate any carry-over effects from the needle between samples. As the ETV is kept at a distance away from the flame source, there would be no risk of melting the μTAS with the flame. In addition, the enclosure of the ETV never exceeded a temperature of 65°C.

It goes without saying that the μTAS would need to be fully vetted as an independent system prior to coupling to the ETV-FAES. This will reduce the complexity of the system and allow for any changes to occur. The complete analysis and development of the μTAS is outside the scope of the work presented herein.

After integration with, or in lieu of, the μTAS, as discussed in Section 8.1, the vaporization source could be integrated onto the microfluidic chip. It could also be possible to integrate a series of microexplosives to provide the thermal source, reducing the footprint and infrastructure needed to perform elemental analysis. This could prove to be a strong revision to a combined open source ETV-FAES-μTAS.

Appendices

Appendix A

Reference Element Information

Contained in Table A.1 is all of the information pertinent to elements which have been detected through the use of Flame Emission Spectroscopy (FES) by Christian and Feldman in 1971. In addition, elemental melting and boiling points have been added from the CRC Handbook. [119] This information is used throughout all analysis to correlate spectral peaks with the appropriate element. It is also used in the determination of effective pre-bake temperatures for the ETV in order to remove spectrally interfering elements.

Table A.1: Elemental Data including Wavelengths, Melting Points, and Boiling Points for Select Elements in which FES has been used. [91, 119]

Name	Element	Wavelength (nm)	Molar Mass (g)	Melt (K)	Boil (K)
Arsenic	As	193.7	74.9216	273.15	876.15
Selenium	Se	196	78.96	493.15	958.15
Zinc	Zn	213.9	65.41	692.68	1183.15
Beryllium	Be	234.9	9.01218	1551.15	2742.15
Silicon	Si	251.6	28.0855	1687.15	3538.15
Antimony	Sb	252.9	121.76	903.94	1860.15
Mercury	Hg	253.6	200.59	234.32	629.88
Germanium	Ge	265.1	72.64	1211.15	3103.15
Platinum	Pt	265.9	195.08	2041.15	4093.15
Gold	Au	267.6	196.9665	1337.33	3123.15
Tin	Sn	284	118.69	505.078	2893.15

Continued on next page...

Table A.1 – continued from previous page

Name	Element	Wavelength (nm)	Molar Mass (g)	Melt (K)	Boil (K)
Magnesium	Mg	285.2	24.305	923.15	1363.15
Bismuth	Bi	306.8	208.9804	544.55	1837.15
Cadmium	Cd	326.1	112.41	594.25	1038.15
Copper	Cu	327.8	63.546	1357.77	2833.15
Silver	Ag	328.1	107.868	1235.1	2428.15
Titanium	Ti	334.9	47.87	1941.15	3560.15
Rhodium	Rh	343.5	102.9055	2236.15	3968.15
Cobalt	Co	345.4	58.9332	1768.15	3203.15
Rhenium	Re	346	186.207	3459.15	5869.15
Nickle	Ni	352.4	58.693	1728.15	3263.15
Zirconium	Zr	360.1	91.22	2123.15	4673.15
Palladium	Pd	363.5	106.4	1828.15	3233.15
Thulium	Tm	371.8	168.9342	1823.15	2223.15
Iron	Fe	372	55.847	1798.25	3023.15
Ruthenium	Ru	372.8	101.07	2603.15	4423.15
Molybdenum	Mo	390.3	95.94	2896.15	4913.15
Aluminum	Al	396.1	26.98154	933.47	2740
Ytterbium	Yb	398.8	173.04	1097.15	1473.15
Erbium	Er	400.8	167.26	1803.15	3133.15
Tungsten	W	400.9	183.84	3695.15	5823.15
Scandium	Sc	402.4	44.9559	1813.15	3103.15
Manganese	Mn	403.3	54.938	1523.15	2333.15
Dysprosium	Dy	404.6	162.5	1683.15	2833.15
Lead	Pb	405.8	207.2	600.61	2023.15
Niobium	Nb	405.9	92.9064	2750.15	5017.15
Holmium	Ho	410.4	164.9303	1743.15	2973.15
Gallium	Ga	417.2	69.723	302.91	2473.15
Chromium	Cr	425.4	51.996	2180.15	2943.15
Vanadium	V	437.9	50.9415	2193.15	3673.15
Lanthanum	La	441.8	138.9055	1813.15	3103.15
Osmium	Os	442	190.2	3303.15	5285.15
Indium	In	451.1	114.82	429.75	2343.15
Lutetium	Lu	451.9	174.97	1933.15	3663.15
Cesium	Cs	455.6	132.9055	301.55	943.15

Continued on next page...

Table A.1 – continued from previous page

Name	Element	Wavelength (nm)	Molar Mass (g)	Melt (K)	Boil (K)
Europium	Eu	459.4	151.96	1095.15	1873.15
Strontium	Sr	460.7	87.62	1050.15	1653.15
Tantalum	Ta	474	180.947	3293.15	5833.15
Samarium	Sm	476	150.4	1443.15	2063.15
Tellurium	Te	486.6	127.6	723.15	1263.15
Thorium	Th	492	232.0381	2023.15	5063.15
Neodymium	Nd	492.5	144.24	1289.15	3343.15
Praseodymium	Pr	495.1	140.9077	1204.15	3783.15
Boron	B	518	10.81	2348.15	4000.15
Hafnium	Hf	531.2	178.49	2503.15	4873.15
Terbium	Tb	534	158.9254	1633.15	3493.15
Thallium	Tl	535	204.383	577.15	1746.15
Uranium	U	544.8	238.0289	1408.15	4403.15
Iridium	Ir	550	192.22	2720.15	4703.15
Barium	Ba	553.6	137.33	998.15	2170.15
Cerium	Ce	570	140.12	1071.15	3716.15
Sodium	Na	589	22.98977	370.87	1156.15
Yttrium	Y	597.2	88.9059	1798.15	3613.15
Calcium[1]	Ca	622	40.078	1115.15	1757.15
Gadolinium	Gd	622.1	157.25	1587.15	3533.15
Lithium	Li	670.8	6.941	453.69	1620.15
Potassium	K	766.5	39.0983	336.55	1038.75
Rubidium	Rb	780	85.468	312.45	963.15

Notes:
[1] Ca atomic emission occurs at 422.7 nm. [91] However the Calcium Hydroxide (CaOH) radical has a stronger emission at 622 nm and is the peak of interest. [15, 18]

Although metals such as Ca, Na, Li, and K are the typical targets of analysis of FAES, as discussed by [11–17], The metals Cu and Pb have also been reported; although this typically requires thermal sources of 3475 K or hotter. [4, 18] All of the elements in Table A.1 were tested by Christian and Feldman in 1971 with a Nitrous Oxide-Acetylene flame of temperature 3200 K. [18, 91]

Appendix B

Python Codes

B.1 Introduction

Contained here are the Python 2.7 scripts designed to be operated on a Windows Operating System.

B.2 Combine Ocean Optics Spectra Files into Single Comma-Separated Values File

```
#Victor Shadbolt - Jan 7 2013

#This python script will combine all of the .txt
#files generated by the OceanOptics capture software
#when it is saved as a delimited file type within a
#single folder (ie each new sample to new folder); and
#save the resulting file as Compiled_Data.csv

#Import required packages native to Python 2.7
import csv
import glob
import os

#setup initial list and tracking variables
counter=-1
a = []
a.append([])
first_run = True
#cycle through all of the .txt files in the folder which
#contains this script
for files in glob.glob("*.txt"):
    a.append([])
    counter = counter + 1
    if first_run == True:
        a[counter].append('wavelength')
        a[counter+1].append('test ' + str(counter))
    if first_run == False:
        a[counter+1].append('test ' + str(counter))
    #inform user which text file is currently being read
    print counter
    SpectrumReader = csv.reader(open(files, 'r'),
                                delimiter='\t', quotechar='|')
    Start_Row = False
```

```
    #Extract the information from the ASCII files
    for row in SpectrumReader:
        if row[0] == '178.32': #Starting wavelength
            Start_Row = True
        if row[0] == '886.57': #Ending wavelength
            if first_run == True:
                a[counter].append(row[0])
            a[counter+1].append(row[1])
            Start_Row = False
        if Start_Row == True:
            if first_run == True:
                a[counter].append(row[0])
            a[counter+1].append(row[1])
    first_run = False

#write the combined csv files into Compiled_Data.csv
item_length = len(a[0])
with open('Compiled_Data.csv', 'wb') as test_file:
  file_writer = csv.writer(test_file)
  for i in range(item_length):
    file_writer.writerow([x[i] for x in a])

print 'done'
```

B.3 LabJack U6 Serial Peripheral Interface Control TestBed File

```
import u6

DAC0 = 5000; DB37pin11 = DAC0
DAC1 = 5002; DB37pin29 = DAC1

d = u6.U6()

d.writeRegister(6112,0)

## Asking for user inputs of Analog Voltage Outputs for each Channel
#  and the Supply Voltage value
#
VDD = float(raw_input("What is the Supply Voltage?"))
VinA = float(raw_input("What Voltage would you like at Channel 0?"))
VinB = float(raw_input("What Voltage would you like at Channel 1?"))
```

```
#

d.writeRegister(DAC0,VDD)

d.writeRegister(DAC1,VDD)

## Rules in order to prevent errors coming from chosen value being too high or too low
#
if VinA > VDD:
    print "DAC A WAS SET TOO HIGH"
    VinA = 0
if VinA < 0:
    print "DAC A CANNOT BE NEGATIVE"
    VinA = 0
if VinB > VDD:
    print "DAC B WAS SET TOO HIGH"
    VinB = 0
if VinB < 0:
    print "DAC B CANNOT BE NEGATIVE"
    VinB = 0
#

## Converting the analog value into binary through the equation found in datasheet
#
if VinA == VDD:
    VintA = bin(4095)
else:
    VintA = bin(int(round(4096*VinA/VDD)))
if VinB == VDD:
    VintB = bin(4095)
else:
    VintB = bin(int(round(4096*VinB/VDD)))
#

## Setting the first four bits for DACA (Channel 0) - xA,  and DACB (Channel 1) - xB
##  First bit determines which channel: 0 = DACA (Channel 0), 1 = DACB (Channel 1)
##  Second bit determines whether to have the selected channel buffered or not:
##     0 = not buffered, 1 = buffered
##  Third bit determines the GAIN Select (TO BE TESTED): 0 = 2x, 1 = 1x
##  Fourth bit determines the state of the channel: 0 = Off/Shutdown, 1 = On/Active
#

xA = "0b111"
xB = "0b1111"
#
```

```
61
62  ## Determining the length of the bit to then find out the number of zeroes infront
63  ## are missing from the binary code
64  #
65  bytelengthA = len(VintA)-2
66  bytelengthB = len(VintB)-2
67
68  startA = 12-bytelengthA
69  startB = 12-bytelengthB
70  #
71
72  ## Setting the rules to create two bytes (8 bit groups) that will transfer data
73  #
74  if startA ==3:
75      xA = xA + startA*"0" + VintA[2]
76  elif startA>3:
77      xA = xA + 4*"0"
78  else:
79      xA = xA + startA*"0" + VintA[2:6-startA]
80
81  if bytelengthA < 8:
82      yA = "0b" + (8-bytelengthA)*"0" + VintA[2:]
83  else:
84      yA = "0b" + VintA[len(VintA)-8:]
85
86  if startB ==3:
87      xB = xB + startB*"0" + VintB[2]
88  elif startB>3:
89      xB = xB + 4*"0"
90  else:
91      xB = xB + startB*"0" + VintB[2:6-startB]
92
93  if bytelengthB < 8:
94      yB = "0b" + (8-bytelengthB)*"0" + VintB[2:]
95  else:
96      yB = "0b" + VintB[len(VintB)-8:]
97  #
98
99  ## Converting the binary values for the MCP4922 SPI control bytes to their integer
100 ## counterparts (not necessary but for simplicity)
101 #
102 SPIBytesDA = [int(xA,2),int(yA,2)]
103 SPIBytesDB = [int(xB,2),int(yB,2)]
104 #
105
```

```
## Bytes to begin SPI communication with the ADC to read voltages
SPIBytesAA = [0b00000001, 0b10100000, 0b11111111]
SPIBytesAB = [0b00000001, 0b11100000, 0b11111111]
#

##DAC VoutA
d.writeRegister(DAC1,0)
d.spi(SPIBytesDA, True, False, "A", 0, 99, 0, 2, 1)
d.writeRegister(DAC1,VDD)
##DAC VoutB
d.writeRegister(DAC1,0)
d.spi(SPIBytesDB, True, False, "A", 0, 99, 0, 2, 1)
d.writeRegister(DAC1,VDD)

##ADC CH0 VOUTA
ADCA =  d.spi(SPIBytesAA, True, False, "A", 0, 3, 0, 2, 1)
##ADC CH1 VOUT B
ADCB =  d.spi(SPIBytesAB, True, False, "A", 0, 3, 0, 2, 1)

##Selecting the received bytes that contain actual information
xA = bin(ADCA["SPIBytes"][1])
yA = bin(ADCA["SPIBytes"][2])

xB = bin(ADCB["SPIBytes"][1])
yB = bin(ADCB["SPIBytes"][2])
#

print ADCA,ADCB

print xA,yA
print xB,yB

##Decoding the received two bytes into a leading 4 bit value and a second 8 bit value
if len(xA) < 6:
    byte4A = "0b" + (6-len(xA))*"0" + xA[2:]
elif len(xA) == 3:
    byte4A = "0b" + (6-len(xA))*"0" + xA[2]
elif len(xA) < 3:
    byte4A = "0b0"
else:
    byte4A = "0b" + xA[len(xA)-4:]

if len(yA) <= 10:
    byte8A = (10-len(yA))*"0" + yA[2:]
elif len(yA) == 3:
```

```
    byte8A = 7*"0" + "1"

if len(xB) < 6:
    byte4B = "0b" + (6-len(xB))*"0" + xB[2:]
elif len(xB) == 3:
    byte4B = "0b" + (6-len(xB))*"0" + xB[2]
elif len(xB) < 3:
    byte4B = "0b0"
else:
    byte4B = "0b" + xB[len(xB)-4:]

if len(yB) <= 10:
    byte8B = (10-len(yB))*"0" + yB[2:]
elif len(yB) == 3:
    byte8B = 7*"0" + "1"
#

## Putting together the 4 bits from the first byte and the second byte to make the
## 12 bit value
byteA = byte4A + byte8A
byteB = byte4B + byte8B
#

print byteA,byteB

## Converting the 12 bit value into a Voltage value using the equation found in
## MCP4922 datasheet
VoutA = (int(byteA,2)/4096.)*VDD
VoutB = (int(byteB,2)/4096.)*VDD
#

print VoutA
print VoutB
```

B.4 LabJack U6 Serial Peripheral Interface Control of ElectroThermal Vaporizer

```
#This code is used to test SPI signals with the LabJack to and from the DAC/ADC
#Test Board. Set a voltage on channel 0 or channel 1, and the ADC will read that
#voltage and feed it back to the LabJack and display it.

#Connections:
```

```
#CS ADC = FIO3 = 3
#CS DAC = DAC1 = 99
#SCK = FIO0 = 0
#SDI = FIO1 = 1
#SDO = FIO2 = 2

#d.SPI works with (CS, SCK, SDO, SDI)
#so for the dac: (99, 0, 2, 1) and the ADC: (3, 0, 2, 1)

import wx
import u6
import time
import thread
import os

# Setting up a list of all the associations to simplify changing of variables
# The pin number, InputOutput label, or Labjack python number may be used

global FIO0;global FIO1;global FIO2;global FIO3;global FIO4;global FIO5;
global FIO6;global FIO7
global DB37pin6; global DB37pin24; global DB37pin5; global DB37pin23;
global DB37pin4; global DB37pin22; global DB37pin3; global DB37pin21
global EIO0; global EIO1; global EIO2; global EIO3; global EIO4; global EIO5;
global EIO6; global EIO7
global DB15pin4; global DB15pin12; global DB15pin5; global DB15pin113;
global DB15pin6; global DB15pin14; global DB15pin7; global DB15pin15
global CIO0; global CIO1; global CIO2; global CIO3
global DB15pin9; global DB15pin2; global DB15pin10; global DB15pin3
global DAC0; global DAC1
global DB37pin11; global DB37pin29
global AIN0
global DB37pin37

FIO0 = 6000; DB37pin6 = FIO0
FIO1 = 6001; DB37pin24 = FIO1
FIO2 = 6002; DB37pin5 = FIO2
FIO3 = 6003; DB37pin23 = FIO3
FIO4 = 6004; DB37pin4 = FIO4
FIO5 = 6005; DB37pin22 = FIO5
FIO6 = 6006; DB37pin3 = FIO6
FIO7 = 6007; DB37pin21 = FIO7
CIO0 = 6016; DB37pin25 = CIO0

EIO0 = 6008; DB15pin4 = EIO0
EIO1 = 6009; DB15pin12 = EIO1
```

```
EIO2 = 6010; DB15pin5 = EIO2
EIO3 = 6011; DB15pin13 = EIO3
EIO4 = 6012; DB15pin6 = EIO4
EIO5 = 6013; DB15pin14 = EIO5
EIO6 = 6014; DB15pin7 = EIO6
EIO7 = 6015; DB15pin15 = EIO7

CIO0 = 6016; DB15pin9 = CIO0
CIO1 = 6017; DB15pin2 = CIO1
CIO2 = 6018; DB15pin10 = CIO2
CIO3 = 6019; DB15pin3 = CIO3

# These are only for streaming mode
# If using as normal terminals they are in increments of two
# (eg. AIN0 = 0, AIN1 = 2)
AIN0 = 0; DB37pin37 = AIN0
AIN1 = 1; DB37pin18 = AIN1
AIN2 = 2; DB37pin36 = AIN2

# Each DAC also takes up two registers and therefore skip 5001
DAC0 = 5000; DB37pin11 = DAC0
DAC1 = 5002; DB37pin29 = DAC1

global VDD
VDD = 3.3
global CH0pre
CH0pre = 0
global CH1pre
CH1pre = 0
global Max_CH1
Max_CH1 = 3.3
global Max_CH0
Max_CH0 = 3.3
global High
High = VDD
global Low
Low = 0

d = u6.U6()

#Set the power for the DAC/ADC board to 3.3V
d.writeRegister(DAC0,VDD)

d.writeRegister(DAC1,VDD)

```

```
#Set the DAC channels 0 and 1 to 0V
d.writeRegister(DAC1,Low)
d.spi([0b11110000, 0b1011], True, False, "A", 0, 99, 0, 2, 1) # vs_edit
d.writeRegister(DAC1,High)
# Channel 0
d.writeRegister(DAC1,Low)
d.spi([0b1110000,0b1011], True, False, "A", 0, 99, 0, 2, 1) # vs_edit
d.writeRegister(DAC1,High)

#d.spi([0b11110000, 0b11011], True, False, "A", 0, 8, 10, 12, 11)
#d.spi([0b1110000,0b11011], True, False, "A", 0, 8, 10, 12, 11)

d.writeRegister(6112,0)

def num2byte(xA,CH0pre,Max):
    if CH0pre == Max:
        VintA = bin(4095)
    else:
        VintA = bin(int(round(4096*CH0pre/(Max))))

    bytelengthA = len(VintA)-2
    startA = 12-bytelengthA

    if startA ==3:
        xA = xA + startA*"0" + VintA[2]
    elif startA>3:
        xA = xA + 4*"0"
    else:
        xA = xA + startA*"0" + VintA[2:6-startA]

    if bytelengthA < 8:
        yA = "0b" + (8-bytelengthA)*"0" + VintA[2:]
    else:
        yA = "0b" + VintA[len(VintA)-8:]

    SPIBytes = [int(xA,2),int(yA,2)]

    return SPIBytes

def byte2num(ADCA):
    # Taking the two (of four) important bytes that are received,
    # containing the measurement taken by the ADC.
    # The second 4 bits of the second byte and the 8 bits of the
    # third byte combine to produce a 12 bit number which when divided
    # by 4096 results in the percentage of VREF that was measured at
```

```
141         # the ADC.
142         # For example, if the 12 bits were all ones then converting that
143         # binary number to an integer would give 4095. This divided by 4095
144         # times 100% gives 100% and if VREF was 5V then what the ADC read was
145         # 5 volts
146         # xA is the last 4 bits of the second byte
147         # yA is the 8 bits of the third byte
148         xA = bin(ADCA["SPIBytes"][1])
149         yA = bin(ADCA["SPIBytes"][2])
150 
151         # This process converts the 4 and 8 bit binary codes into one
152         # twelve bit message
153         if len(xA) < 6:
154             byte4A = "0b" + (6-len(xA))*"0" + xA[2:]
155         elif len(xA) == 3:
156             byte4A = "0b" + (6-len(xA))*"0" + xA[2]
157         elif len(xA) < 3:
158             byte4A = "0b0"
159         else:
160             byte4A = "0b" + xA[len(xA)-4:]
161 
162         if len(yA) <= 10:
163             byte8A = (10-len(yA))*"0" + yA[2:]
164         elif len(yA) == 3:
165             byte8A = 7*"0" + "1"
166 
167         byte = byte4A + byte8A
168 
169         # Vout1A is the result of converting the 12 bit binary code to
170         # an integer and then finding its overall percentage of VREF
171         # which in this case is 5, but due to amplification the board
172         # is actually at twice that voltage and therefore 10 is used.
173         Vout1A = (int(byte,2)/4096.)*VDD #1 = 1 amp max (5V = 1A) # vs_edit
174 
175         return Vout1A
176 
177     class Example(wx.Frame):
178 
179         def __init__(self,parent,id):
180             wx.Frame.__init__(self,parent,id, "DAC ADC Test", size = (300,150))
181             global panel
182             panel = wx.Panel(self)
183 
184             global busy
185             busy = False
```

```
186         global Updating_On
187         Updating_On = False
188 #
189         global CH0
190         CH0 = wx.TextCtrl(panel, -1, "0", pos = (10,10), size = (125,-1),
191                           style = wx.TE_PROCESS_ENTER)
192         CH0.Bind(wx.EVT_TEXT_ENTER, self.CH0input)
193         global CH0text
194         global CH1text
195         CH0text = wx.StaticText(panel, -1, "Channel 0 = 0", pos = (10,35))
196         CH1text = wx.StaticText(panel, -1, "Channel 1 = 0", pos = (10,50))
197 
198         global CH0set
199         CH0set = wx.Button(panel, label = "Channel 0", pos = (150,10))
200         CH0set.Bind(wx.EVT_BUTTON, self.CH0input)
201 #
202         global CH1
203         CH1 = wx.TextCtrl(panel, -1, "0",pos = (10,70), size = (125,-1),
204                           style = wx.TE_PROCESS_ENTER)
205         CH1.Bind(wx.EVT_TEXT_ENTER, self.CH1input)
206 
207         global CH1set
208         CH1set = wx.Button(panel, label = "Channel 1", pos = (150,70))
209         CH1set.Bind(wx.EVT_BUTTON, self.CH1input)
210 
211         #button = wx.Button(panel, label = "Auto Update", pos = (150,120))
212         #button.Bind(wx.EVT_BUTTON, self.start)
213 
214 
215         ## This function creates a new thread where the AutoUpdate function takes
216         # place and the following brackets contain the variable arguments that
217         # will be inputted into the AutoUpdate function
218         Updating_On = True
219         thread.start_new_thread(self.AutoUpdate,())
220 
221         thread.start_new_thread(self.Keep_Alive,())
222 
223     def start(self, event):
224         global Updating_On
225 
226         if Updating_On == False:
227             Updating_On = True
228             thread.start_new_thread(self.AutoUpdate,())
229 
230     def Keep_Alive(self):
```

```
231         global last_run_time
232 
233         try:
234             last_run_time
235         except NameError:
236             last_run_time = 0
237 
238         while 1:
239             if time.time() - last_run_time > 2:
240                 thread.start_new_thread(self.SetPWRLevel,())
241                 print ("starting it up!")
242             time.sleep(1)
243 
244     def SetPWRLevel(self):
245         global CH0pre
246         global CH1pre
247         global busy
248         global Set_CH0_Level
249         global Set_CH1_Level
250         global last_run_time
251 
252         try:
253             CH0pre
254         except NameError:
255             CH0pre = 0
256         try:
257             CH1pre
258         except NameError:
259             CH1pre = 0
260         try:
261             Set_CH1_Level
262         except NameError:
263             Set_CH1_Level = 0
264         try:
265             Set_CH0_Level
266         except NameError:
267             Set_CH0_Level = 0
268 
269         print ("started")
270 
271         while 1:
272             last_run_time = time.time()
273             if round(CH1pre,2) != round(Set_CH1_Level,2):
274                 while (round(CH1pre,2) != round(Set_CH1_Level,2)):
275                     if busy == False:
```

```
                        busy = True
                        #if Set_CH1_Level-CH1pre < 0:
                        #    CH1pre -= 0.01
                        #else:
                        #    CH1pre = Set_CH1_Level
                        CH1pre = Set_CH1_Level
                        SPIBytes = num2byte("0b1111", CH1pre, Max_CH1)
                        d.writeRegister(DAC1,Low)
                        d.spi(SPIBytes, True, False, "A", 0, 99, 0, 2, 1)
                        d.writeRegister(DAC1,High)
                        busy = False
                        last_run_time = time.time()
                        time.sleep(.025)
            if round(CH0pre,2) != round(Set_CH0_Level,2):
                while round(CH0pre,2) != round(Set_CH0_Level,2):
                    if busy == False:
                        busy = True
                        #if Set_CH0_Level-CH0pre < 0:
                        #    CH0pre -= 0.01
                        #else:
                        #    CH0pre += 0.01
                        CH0pre = Set_CH0_Level
                        SPIBytes = num2byte("0b111", CH0pre, Max_CH0)
                        d.writeRegister(DAC1,Low)
                        d.spi(SPIBytes, True, False, "A", 0, 99, 0, 2, 1)
                        d.writeRegister(DAC1,High)
                        busy = False
                        last_run_time = time.time()
                        time.sleep(.025)


    def LabelUpdate(self):
        global LabelUpdate_On
        global Set_CH0_Level
        global Set_CH1_Level

        while 1:
            if LabelUpdate_On == False:
                thread.exit()
            wx.CallAfter(self.UpdateLabels)
            time.sleep(0.25)

    def UpdateLabelsCH1(self):
        CH1.SetLabel(str(Set_CH1_Level))

```

```
321     def UpdateLabelsCH0(self):
322         CH0.SetLabel(str(Set_CH0_Level))
323 
324     def AutoUpdate(self):
325         ## This function periodically reads the Electrothermal Vaporizer ADCs
326         # (only channel 0) and displays them. Channel 0 contains the voltages
327         # so no current is read
328 
329         # Setting the Vouts as global so that they can be read elsewhere if
330         # necessary
331         global VoutCH0
332         global VoutCH0_str
333         global VoutCH1
334         global VoutCH1_str
335         global busy # vs_edit
336         global Updating_On
337         global last_update
338 
339         # A While loop that goes on until the program is closed which reads both
340         # ETV ADCs (channel 0)
341 
342         while 1:
343             if Updating_On == False:
344                 thread.exit()
345 
346             # to prevent overlap of SPI
347             if busy == False:
348                 busy = True
349 
350                 # The SPI byte necessary to read from channel 0 of an ADC
351                 ETVBytesCH0 = [0b00000001, 0b10100000, 0b11111111]
352                 ETVBytesCH1 = [0b00000001, 0b11100000, 0b11111111]
353                 # Sending the message to read from channel 0
354                 VoutCH0 = byte2num(d.spi(ETVBytesCH0, True, False, "A", 0, 3, 0, 2, 1))
355                 VoutCH1 = byte2num(d.spi(ETVBytesCH1, True, False, "A", 0, 3, 0, 2, 1))
356                 VoutCH0 = round(VoutCH0,5)
357                 VoutCH0_str = str(VoutCH0)
358                 if len(VoutCH0_str)<7:
359                     for i in range(0, 7-len(VoutCH0_str)):
360                         VoutCH0_str = VoutCH0_str + "0"
361 
362                 VoutCH1 = round(VoutCH1,5)
363                 VoutCH1_str = str(VoutCH1)
364                 if len(VoutCH1_str)<7:
365                     for i in range(0, 7-len(VoutCH1_str)):
```

```
366                         VoutCH1_str = VoutCH1_str + "0"
367
368                 # This function allows smooth communication between a thread and
369                 # a main window when using a GUI and calls the function Update
370                 # to take the readings and display them
371                 wx.CallAfter(self.Update)
372                 last_update = time.time()
373
374                 busy = False
375                 # The increment between readings
376                 time.sleep(0.25)
377
378     def Update(self):
379         ## This function takes the readings from AutoUpdate and displays them
380         # in the main window
381         CH0text.SetLabel("Channel 0 = " + VoutCH0_str + " V")
382         CH1text.SetLabel("Channel 1 = " + VoutCH1_str + " V")
383
384     def CH0input(self,event):
385         global CH0pre
386         global Set_CH0_Level
387
388         try:
389             VinA = float(CH0.GetValue())
390         except ValueError:
391             CH1.SetLabel("Voltage must be between 0 and " + str(Max_CH0) + " Volts")
392             return
393
394         if VinA > Max_CH0:
395             CH0text.SetLabel("Max = " + str(Max_CH0) + " V")
396             return
397         if VinA < 0:
398             CH0text.SetLabel("Voltage CANNOT BE NEGATIVE")
399             return
400
401         Set_CH0_Level = VinA
402
403     def CH1input(self,event):
404         global CH1pre
405         global Set_CH1_Level
406
407         try:
408             Cur = float(CH1.GetValue())
409         except ValueError:
410             Constant_Current = False
```

```
411             CH1.SetLabel("Current must be between 0 and " + str(Max_CH1) + " Amps")
412             return
413         Cur = float(CH1.GetValue())
414         if Cur > Max_CH1:
415             CH1.SetLabel("Max = " + str(Max_CH1) + " V")
416             return
417         if Cur < 0:
418             CH1.SetLabel("Voltage CANNOT BE NEGATIVE")
419             return
420 
421         Set_CH1_Level = Cur
422 
423 if __name__ == "__main__":
424 
425     app = wx.PySimpleApp()
426     frame = Example(parent = None,id = -1)
427     frame.Show()
428     app.MainLoop()
```

B.5 LabJack U6 UART Communication

```
1  import u6
2  
3  d = u6.U6()
4  d.configU6()
5  d.getCalibrationData()
6  d.configTimerClock()
7  
8  d.asynchConfig(DesiredBaud=9600)
9  
10 d.configIO(NumberTimersEnabled = 0, TimerCounterPinOffset = 8, EnableUART = True)
11 
12 #What do I want to send?
13 To_Send = 'This is my sample text&' #23 char long
14 
15 #Convert this string into bytes for the LabJack
16 txBytes = []
17 for x in range(0, len(To_Send)):
18     txBytes.append(ord(To_Send[x]))
19 
20 #Send the message
21 d.asynchTX(txBytes)
22 
```

```
#Read the response; Every time you call asynchRX 30 bytes are returned (0 if empty)
rxResponseT = ''
while '&' not in rxResponseT: #& is the terminator in the message
    rxResponse = d.asynchRX()['AsynchBytes']
    if rxResponse[0] != 0:
        for x in range(0, len(rxResponse)):
            if rxResponse[x] != 0:
                rxResponseT = rxResponseT + (chr(rxResponse[x]))

    print rxResponseT
```

B.6 Sort Ocean Optics Spectra Files into Multiple Folders based on Time

```
#Victor Shadbolt - October 30 2014

#This python script will sort all of the generated .txt
#files generated by the OceanOptics capture softare when
#it is saved as a delimited file type within a single folder.
#The script will look at the timing information in each file
#and separate each test into its own folder 'Test_00_N' where
#N is the test number

#Import required packages
from StringIO import StringIO
import csv
from datetime import datetime
from datetime import timedelta
from dateutil import tz
import os
import glob
import time
import shutil

#Set the time limit. A gap in file times more than this
#corresponds to a new test
limit = timedelta(seconds = 1, milliseconds = 64)

#Cycle through all of the .txt files in the folder which
#contrains this script
file_names = []
for files in glob.glob('*.txt'):
```

```
29      file_names.append(files) #store file names
30
31  #Extract the time and date information from each file
32  file_times = []
33  seconds = []
34  miliseconds = []
35  minutes = []
36  hours = []
37  day = []
38  month = []
39  year = []
40  d = []
41  for x in range(0, len(file_names)):
42      day.append(file_names[x].split('_')[1])     #extract day
43      month.append(file_names[x].split('_')[0])   #extract month
44      year.append(file_names[x].split('_')[2])    #extract year
45
46      d5 = datetime.strptime(str(day[x]).zfill(2), "%d")  #day
47      d6 = datetime.strptime(str(month[x]), "%b")         #month
48      d7 = datetime.strptime(str(year[x]).zfill(4), "%Y") #year
49
50      #extract time stamp from file name
51      file_times.append(file_names[x].split('_')[4].split('.')[0])
52      seconds.append(int(file_times[x].split('-')[2]))        #extract seconds
53      miliseconds.append(int(file_times[x].split('-')[3]))    #extract miliseconds
54      minutes.append(int(file_times[x].split('-')[1]))        #extract minutes
55      hours.append(int(file_times[x].split('-')[0]))          #extract hours
56      d1 = datetime.strptime(str(hours[x]).zfill(2), "%H")    #hour
57      d2 = datetime.strptime(str(minutes[x]).zfill(2), "%M")  #minute
58      d3 = datetime.strptime(str(seconds[x]).zfill(2), "%S")  #second
59      d4 = datetime.strptime(str(miliseconds[x]*1000).zfill(6), "%f") #microsecond
60
61      #Rebuild time information
62      d.append(datetime.strptime(str(d7.strftime("%Y") +
63                                     ' ' + d6.strftime("%b") +
64                                     ' ' + d5.strftime("%d") +
65                                     ' ' + d1.strftime("%H") +
66                                     ' ' + d2.strftime("%M") +
67                                     ' ' + d3.strftime("%S") +
68                                     ' ' + d4.strftime("%f")), "%Y %b %d %H %M %S %f"))
69
70  #Setup the counter to monitor the current test number
71  counter = 1
72  #Sort through the files and move them to an appropriate folder
73  for x in range(1, len(file_names)):
```

```
74      flagged = True
75      if ((d[x] - d[x-1]) < limit):
76          flagged = False #File times within the limit. Same test
77
78      if flagged == True:
79          counter += 1    #File times outside limit. New test
80
81      newpath = r'./Test_00_' + str(counter)
82      if not os.path.exists(newpath): os.makedirs(newpath)
83      try:
84          shutil.move(file_names[x], newpath)
85          if x == 1:
86              shutil.move(file_names[x-1], newpath)
87      except WindowsError:
88          print files + " error"
89          pass
90
91  print counter
```

Appendix C

MATLAB Codes

C.1 Introduction

Contained here are the MATLAB scripts designed to be operated on a Windows Operating System.

C.2 Reset MATLAB

```
%% Reset Matlab and Clean Up
close all    % Close all previous figures
clc          % Clear the command window
clear        % Clear all variables

True = 1;    % Define boolean operators
False = 0;
```

C.3 Read Comma-Separated Values File into Array

```
%% Read the CSV Data
% Read the CSV files and rebuild all of the data into the variable: Rebuilt
%
% Note that the _Rebuilt_ variable will be a three dimentional matrix which
% grows in size depending upon the amount of data collected.

%Read all of the files one by one & rebuild CSVs into Rebuilt matrix
for Conc = [1 : Total_Conc+1]
    M = importdata(CSVFile{Conc}, ',', 1);              %Read the CSV file
    Total_Rows = length(M.data(:,1));                   %Define Num Rows
    Total_Columns = length(M.data(1,:));                %Define Num Cols
    for row = [1 : Total_Rows]
        Wavelength = M.data(row, 1);                    %Extract wavelength
        Values = M.data(row,2:length(M.data(row,:)));   %Extract data
        Rebuilt(Conc, row, 1) = Wavelength;
        for k = [1 : Total_Columns-1]
           Rebuilt(Conc, row, k+1) = Values(k);         %Build Rebuilt
        end
    end
end

%Clean up after running this code segment
clear CSVFile
```

```
clear M
clear Values
clear Conc
clear row
clear k
clear Wavelength
```

C.4 Process Data with Linear Regression and Plot

```
%% Processed Data Calibration Curve with Linear Regression Analysis
% This calibration curve plot shows the processed data in red and forgoes
% the addition of error bars. Note that Processed means that all of the
% data at each concentration was averaged to a single value.
%
% In addition, a Linear Regression Analysis is performed by curve-fitting a
% One-Dimentional polynomal to the data (Linear Regression) through the use
% of a Least Squares method. This model is then plotted overtop of the
% data.
%
% The LOD and LOQ is determined on the processed data using the
% following formula:
%
% ££LOD = 3* \left( \sqrt{\frac{(Residual \hphantom SSum \hphantom oof
% \hphantom SSquares)}{(Slope \hphantom oof \hphantom MModel)}} \hphantom i
% \right) ££
%
% ££LOQ = LOD * \frac{10}{3}££


for l = [1 : Total_Conc]
    for i = [1 : length(Rows_of_Interest)]
        for j = [2 : length(Rebuilt(l, Rows_of_Interest(i), :))]
            temp_x(j-1) = Concentration(l);
            temp_y(j-1) = Rebuilt(l, Rows_of_Interest(i), j);
        end
        temp_x2(i) = mean(temp_x);
        temp_y2(i) = mean(temp_y);
    end
    x(l) = mean(temp_x2);   %Average the concentration
    y(l) = mean(temp_y2);   %Average all of the collected data
    margin_of_error(l) = std(temp_y2); %standard deviation
end

```

```
%Perform Linear Regression Analysis & Plot results on Raw Data
%Assume data is normally distributed via a T-Distribution

t = 12.7062; %alpha = 0.05, df = 1
[BetaHat,SSE,Ssquared,ULerror,yfit2] = FitLinearModel (x',y', t);
yfit2 = yfit2';

yresid2 = y - yfit2;                %Compute the residual values as a
                                    %vector signed numbers
SSresid2 = sum(yresid2.^2);         %Square the residuals and total them
                                    %obtain the residual sum of squares:
SStotal2 = (length(y)-1) * var(y);  %Compute the total sum of squares of y
                                    %by multiplying the variance of y by
                                    %the number of samples minus 1:
rsq = 1 - SSresid2/SStotal2;        %Compute R^2


%Note that sqrt(SSresid) provides the standard error of the predicted value
LOD2 = 3 *(sqrt(SSresid2)/BetaHat(2)); %Calculate LOD with 99% confidence
LOQ2 = 10*(sqrt(SSresid2)/BetaHat(2)); %Calculate the LOQ

LOD_ppm2 = ((LOD2 * Molar_Mass)/1000)*1E6;
LOQ_ppm2 = ((LOQ2 * Molar_Mass)/1000)*1E6;

disp(['Processed Data Calculations'])
disp(['LOD2 = ',num2str(LOD2*1e6),' umol/L']);
disp(['LOD2 = ',num2str(LOD_ppm2),' PPM']);
disp([' ']);
disp(['LOQ2 = ',num2str(LOQ2*1e6),' umol/L']);
disp(['LOQ2 = ',num2str(LOQ_ppm2),' PPM']);

figure(4)
hold on
E = margin_of_error;          %Confidence interval on the mean
hFit = line(x*1e6, yfit2);    %Plot the Linear Regression model in blue
hE = errorbar(x*1e6,yfit2,E); %Plot error bars on the data

%Set plot title, xlabel and ylabel
hTitle  = title(plot_title2);
hXLabel = xlabel('Concentration (Micromoles/L)');
hYLabel = ylabel('Counts');

%Display on the plot the values and confidence intervals for the linear
%regression parameters of the fitted model
loc1 = (((yaxismax-yaxismin)/3)*2 + yaxismin) + ((yaxismax-yaxismin)/3)/10;
```

```
loc2 = (((yaxismax-yaxismin)/3)*2 + yaxismin) - ((yaxismax-yaxismin)/3)/10;
loc3 = (((yaxismax-yaxismin)/3)*1 + yaxismin) + ((yaxismax-yaxismin)/3)/10;
loc4 = (((yaxismax-yaxismin)/3)*1 + yaxismin) - ((yaxismax-yaxismin)/3)/10;
hText1   = text(5, loc1, ...
  sprintf('\\it{\\beta_0 = %0.4e \\pm %0.4e (CI) }', ...
  BetaHat(1), ULerror(1)));
hText2   = text(5, loc2, ...
  sprintf('\\it{\\beta_1 = %0.4e \\pm %0.4e (CI) }', ...
  BetaHat(2), ULerror(2)));
hText3   = text(55, loc3, ...
  sprintf('\\it{LOD = % 4.1f \\muM/L}', ...
  LOD2*1e6));
hText4   = text(55, loc4, ...
  sprintf('\\it{LOQ = %4.1f \\muM/L}', ...
  LOQ2*1e6));

%Add a legend to the plot, and adjust location to top left corner
hLegend = legend( ...
  [hE, hFit], ...
  'Data (\mu \pm \sigma)' , ...
  'Fit (\it{\beta_0 + \beta_1x})', ...
  'location', 'NorthWest' );

axis([xaxismin xaxismax yaxismin yaxismax]) %Set Axis

%Save the figure as an EPS in the folder
set(gcf, 'PaperPositionMode', 'auto');
print('-depsc',save_plot_2_a)
fixPSlinestyle(save_plot_2_a, save_plot_2_b);

%Clean up after running this code segment
clear l
clear j
clear i
clear temp_x
clear temp_x2
clear temp_y
clear temp_y2
clear hleg1
clear x
clear y
clear yfit2
clear con2_loc
clear con3_loc
```

Appendix D

Arduino Codes

D.1 Introduction

Contained here are the Arduino scripts designed to be operated on the BoArduino.

D.2 DAC and ADC Control

The following code tab was written to work with the MCP3202 Analog to Digital Converter (ADC), and the MCP4822 and MCP4922 Digital to Analog Converters (DACs). The differential read mode on the ADC is performed by reading channel 0 followed by channel 1 and subtracting the two.

```
// This tab contains all of the functions related to setting the DAC, and reading
// from the ADCs.

////////////////////////////////////////////////////////////////////////////

int set_dac(byte cs, byte channel, byte gain, byte shdn, unsigned int v_set){
  commandbits = 0x0000;            // reset the command-bits. 16 binary large
  commandbits ^= (shdn<<12);       // Toggle shutdown?
  commandbits ^= (gain<<13);       // Set the gain to be 1x or 2x?
  commandbits ^= (channel<<15);    // Which Channel?
  commandbits ^= (1<<14);          // Buffered
  commandbits |= (v_set & 0x0FFF); // Set the desired voltage
                                   // from (0-4095)*G*2.048
  digitalWrite(cs, LOW);           // Select the DAC
  SPI.transfer(commandbits>>8);    // Send the first 8 bits
                                   // (4 command bits and the first 4 data bits)
  SPI.transfer(commandbits&0xFF);  // Send the last 8 data bits
  digitalWrite(cs, HIGH);          // DeSelect the DAC
}

////////////////////////////////////////////////////////////////////////////

float read_adc0(byte cs){
  // channel 0
  digitalWrite(cs, LOW);       // Select adc
  dummy = SPI.transfer(0x01);
  adc_h4 = SPI.transfer(0xA0); // unipolar, channel=0, msb first=0, returns
                               // B11-B8 in lower nibble
  adc_l8 = SPI.transfer(0x00); // returns B7-B0
  digitalWrite(cs, HIGH);      // DeSelect adc
```

```
  // Convert the received bits to binary
  adc_h4 = (adc_h4 & 0x0F);
  adc_value0 = (adc_h4 << 8) + adc_l8;
  //return the voltage equivalent
  return 5.0 * (float)adc_value0 / 4096.0;
}

////////////////////////////////////////////////////////////////////////////

float read_adc1(byte cs){
  // channel 1
  digitalWrite(cs, LOW);       // Select adc
  dummy = SPI.transfer(0x01);
  adc_h4 = SPI.transfer(0xE0); // unipolar, channel=1, msb first=0, returns
                               // B11-B8 in lower nibble
  adc_l8 = SPI.transfer(0x00); // returns B7-B0
  digitalWrite(cs, HIGH);      // DeSelect adc
  // Convert the received bits to binary
  adc_h4 = (adc_h4 & 0x0F);
  adc_value1 = (adc_h4 << 8) + adc_l8;
  //return the voltage equivalent
  return 5.0 * (float)adc_value1 / 4096.0;
}

////////////////////////////////////////////////////////////////////////////

float read_diff(byte cs){
  // This function will read both CH0 and CH1 of the MCP3202 within 56ms +- 4ms
  // Differential measurement
  commandbits1 = 0xFC;           // binary equivalent = B11111100
  commandbits2 = 0xFC;
  commandbits2^=(1<<cs);         // toggle bit corresponding to CS
  //Read ADC Channel 1
  //Directly control the digital port on the arduino. Note that this is ~10x faster
  //than using DigitalWrite, but it also means you are directly controlling pins.
  //NEVER change the last two bits corresponding to D0 and D1 (or Rx, TX) or it will
  //kill the Serial communication ability of the Arduino.
  PORTD = commandbits2;          // Select ADC
  //talk to ADC via hardware SPI
  dummy1 = SPI.transfer(0x01);
  adc_h41 = SPI.transfer(0xE0);  // unipolar, channel=1, msb first=0, returns
                                 // B11-B8 in lower nibble
  adc_l81 = SPI.transfer(0x00);  // returns B7-B0
  PORTD = commandbits1;          // DeSelect ADC
  //Read ADC Channel 0
```

```
  PORTD = commandbits2;          // Select ADC
  dummy2 = SPI.transfer(0x01);
  adc_h42 = SPI.transfer(0xA0);  // unipolar, channel=0, msb first=0, returns
                                 // B11-B8 in lower nibble
  adc_l82 = SPI.transfer(0x00);  // returns B7-B0
  PORTD = commandbits1;          // DeSelect ADC

  // Convert the collected bits to a voltage value
  adc_h41 = (adc_h41 & 0x0F);
  adc_value1 = (adc_h41 << 8) + adc_l81;
  adc_h42 = (adc_h42 & 0x0F);
  adc_value0 = (adc_h42 << 8) + adc_l82;
  // Return the differential voltage
  return (5.0 * (float)adc_value0 / 4096.0)-(5.0 * (float)adc_value1 / 4096.0);
}
```

D.3 MAX31855 Thermocouple Control

```
// This tab contains all of the functions related to reading the MAX31855 TC.

/***************************************************
 This is the Adafruit MAX31885 Arduino Library

 Tested and works great with the Adafruit Thermocouple Breakout w/MAX31885K
     ------> http://www.adafruit.com/products/269

 These modules use SPI to communicate, 3 pins are required to
 interface

 Adafruit invests time and resources providing this open source code,
 please support Adafruit and open-source hardware by purchasing
 products from Adafruit!

 Written by Limor Fried/Ladyada  for Adafruit Industries.
 BSD license, check license.txt for more information
 All text above must be included in any redistribution
 ****************************************************/

 //The provided code in the TC library written by Limor Fried/Ladyada was
 //modified to suit my purpose here.

////////////////////////////////////////////////////////////////////////////

```

```
26 float read_MAX_internal(byte cs){
27   int i;
28   union bytes_to_uint32 {
29     uint8_t bytes[4];
30     uint32_t integer;
31   } buffer;
32   int32_t v;
33   // channel 0
34   digitalWrite(cs, LOW);       // Select TC
35 
36   for (i=3;i>=0;i--) {
37     buffer.bytes[i] = SPI.transfer(0x00);
38   }
39   digitalWrite(cs, HIGH);      // DeSelect TC
40 
41   v = buffer.integer;
42 
43   v >>= 4;
44 
45   // pull the bottom 11 bits off
46   float internal = v & 0x7FF;
47   // check sign bit!
48   if (v & 0x800) {
49     // Convert to negative value by extending sign and casting to signed type.
50     int16_t tmp = 0xF800 | (v & 0x7FF);
51     internal = tmp;
52   }
53   internal *= 0.0625; // LSB = 0.0625 degrees
54   return internal;
55 }
56 
57 ////////////////////////////////////////////////////////////////////////////////
58 
59 float read_MAX_external(byte cs){
60   int i;
61   union bytes_to_uint32 {
62     uint8_t bytes[4];
63     uint32_t integer;
64   } buffer;
65   int32_t v;
66   // channel 0
67   digitalWrite(cs, LOW);       // Select TC
68 
69   for (i=3;i>=0;i--) {
70     buffer.bytes[i] = SPI.transfer(0x00);
```

```
  }
  digitalWrite(cs, HIGH);      // DeSelect TC

  v = buffer.integer;

  if (v & 0x7) {
    // uh oh, a serious problem!
    return NAN;
  }

  if (v & 0x80000000) {
    // Negative value, drop the lower 18 bits and explicitly extend sign bits.
    v = 0xFFFFC000 | ((v >> 18) & 0x00003FFFF);
  }
  else {
    // Positive value, just drop the lower 18 bits.
    v >>= 18;
  }
  //Serial.println(v, HEX);

  double centigrade = v;

  // LSB = 0.25 degrees C
  centigrade *= 0.25;
  return centigrade;
}
```

D.4 PID Controller

```
// The PID Controller was writen by Gordon H. Hall and modified by Victor G. Shadbolt

// D11 = SDI
// D12 = SDO
// D13 = CLK

#include <SPI.h>
#include <Math.h>

#define CS_DAC1 7 //setup the CS for the Arduino
#define CS_TC 6
#define OK_LED 4
#define BAD_LED 5

```

```
15 //Setup variables
16 unsigned long adc_value0, adc_value1;    // 12bit ADC values
17 unsigned char dummy, adc_h4, adc_l8;     // 4 high and 8 low bits
18 unsigned char dummy1, adc_h41, adc_l81;  // 4 high and 8 low bits
19 unsigned char dummy2, adc_h42, adc_l82;  // 4 high and 8 low bits
20 unsigned int commandbits;                // DAC command variable
21 byte commandbits1, commandbits2;         // Toggles which chips are on/off
22 float alpha, beta, omega, zeta;          // Store result
23
24 double temp,setvoltage,error,errorsum,differror,errorold,imax,imin,settemp,settempold;
25 float kI,kP,kD,adcread,tempsum;
26 int dacset1,dacset2,counter;
27
28 int OK_LED_State = LOW;
29 int BAD_LED_State = HIGH;
30
31 unsigned long tstart,time;
32 unsigned long ul_PreviousMillis = 0UL;
33 unsigned long ul_Interval = 1000UL; //a second
34
35 ////////////////////////////////////////////////////////////////////////////
36
37 void setup() {
38
39   pinMode(CS_DAC1, OUTPUT);
40   pinMode(CS_TC, OUTPUT);
41   pinMode(OK_LED, OUTPUT);
42   pinMode(BAD_LED, OUTPUT);
43   digitalWrite(CS_DAC1, HIGH);
44   digitalWrite(CS_TC, HIGH);
45   digitalWrite(OK_LED, OK_LED_State);
46   digitalWrite(BAD_LED, BAD_LED_State);
47
48   SPI.begin();
49   SPI.setBitOrder(MSBFIRST);
50   SPI.setDataMode(SPI_MODE0);
51   SPI.setClockDivider(SPI_CLOCK_DIV4); //4MHz clock. Use SPI_CLOCK_DIV8 for 2MHz.
52
53   /////////Init Variables/////////
54   setvoltage=0;
55   errorold=100;
56   errorsum=0;
57   //kI=0.0005;
58   kI = 0.0;
59   kP = 0.5;
```

```
  //kP=0.0391;
  kD = 0.0;
  //kD=0.1;
  settemp=52; //hold 50C
  settempold=0;
  counter=0;
  tempsum=0;
  ul_PreviousMillis = millis();

  Serial.begin(9600);
  //Serial.println("MAX31855 test");
  // wait for MAX chip to stabilize
  delay(500);
}

void loop(){
  // read the temp
  temp=read_MAX_external(CS_TC);//Reads the thermocouple
  if (isnan(temp)) {
    temp = 0.0;
  }

  //Update status LEDs
  if(temp<(settemp-2)){
    BAD_LED_State = HIGH;
    OK_LED_State = LOW;
  }
  if(temp>(settemp-2)){
    BAD_LED_State = LOW;
    OK_LED_State = HIGH;
  }
  if(temp>(settemp+2)){
    BAD_LED_State = HIGH;
    OK_LED_State = HIGH;
  }
  digitalWrite(OK_LED, OK_LED_State);
  digitalWrite(BAD_LED, BAD_LED_State);

  settempold=settemp;//archives old set temperature

  if(isnan(settempold)){
    settempold=0;//Corrects for the first step inaccuracy
  }

  if(settemp>95){
```

```
105       settemp=95;//Clamps set temperature to below 95 degrees
106     }
107     else if (settemp<0){
108       settemp = 0;//Clamps set temperature to above 0
109     }
110
111     error=settemp-temp; // Calculates the error for P controller
112     differror=error-errorold;//Calculates the differential error for D controller
113
114     errorsum=errorsum+error;//Creates an error sum for I controller
115
116
117     if(settemp>settempold&&errorold>10){
118       errorsum=0;//stops massive error accumulation (keeps I controller off until
119                  //within 10 degrees of setpoint
120     }
121     if(settemp<settempold&&errorold<-(settempold-settemp)/(settemp)){
122     errorsum=6.6361*settemp; // sets the I controller to a level ~ close to the
123                              // steady state for cooldown to prevent overshoot
124     }
125
126
127     setvoltage=kI*errorsum+kP*error+kD*differror;//Sets the voltage to the current
128                                                  //controller based on PID values
129     if(setvoltage>0.7){
130       setvoltage=0.7;//Clamps current at below 700 mA
131     }
132     else if (setvoltage<0){
133       setvoltage = 0; // Clamps current at above 0 mA
134     }
135     else if(settemp<settempold&&error<-(settemp)/(4*(settempold-settemp))){
136       setvoltage=0; // Keeps current off during cooldown until close to the set
137                     // temp to increase cooling rates
138     }
139     if(temp==0){
140       setvoltage=0;//Turns off heater if the thermocouple stops reading
141     }
142     errorold=error;//saves old error
143
144     dacset1=setvoltage/5*4096;
145     set_dac(CS_DAC1, 0, 0, 1, dacset1); // sets current with a voltage from a DAC
146
147   //Consol reporting (note: not saved)
148     unsigned long ul_CurrentMillis = millis();
149     time=millis()-tstart;//timer
```

```
  if( ul_CurrentMillis - ul_PreviousMillis > ul_Interval)
  {
    ul_PreviousMillis = ul_CurrentMillis;
    Serial.print("Set Temp = \t");
    Serial.print(settemp);
    Serial.print("\t Current Temp = \t");
    Serial.print(temp);
    Serial.print("\t Set Voltage = \t");
    Serial.println(setvoltage);
  }

}
```

D.5 UART Parsing

```
String Parse_Data(char data[256]){
  String what_to_do;
  const int MAX_SUBSTRINGS = 3;
  static char* substrings[MAX_SUBSTRINGS];

  Serial.print("Data: \t");
  Serial.println(data);


  for (int i = 0; i < MAX_SUBSTRINGS; i++)
  {
      substrings[i] = 0;
  }
  char *command = strtok(data, " ");
  int i = 0;
  while (command != 0 && i < MAX_SUBSTRINGS) {
    substrings[i++] = command;
    command = strtok(0," ");
  }

  if (strcmp(substrings[0], "Dry")  == 0)
    {
      if (Dry() == 1){
        what_to_do = String("Dry Done");
        Serial.println("Dry Done");
      }
    }
  else if (strcmp(substrings[0], "Bake") == 0)
```

```
    {
      if (Bake(atof(substrings[1]), atof(substrings[2])) == 1){
        what_to_do = String("Bake Done");
        Serial.println("Bake Done");
      }

    }
  else if (strcmp(substrings[0], "BakeA") == 0)
    {
      if (BakeA(atof(substrings[1]), atof(substrings[2])) == 1){
        what_to_do = String("BakeA Done");
        Serial.println("BakeA Done");
      }

    }
  else if (strcmp(substrings[0], "Condition") == 0)
    {
      if (Condition() == 1){
        what_to_do = String("Condition Done");
        Serial.println("Condition Done");
      }

    }
  else if (strcmp(substrings[0], "Kill") == 0)
    {
      what_to_do = String("Kill Done");
      Serial.println("Kill Done");
    }

  return what_to_do;
}
```

D.6 Weather Station

```
//Load the appropriate libraries
#include <AltSoftSerial.h>
#include <Wire.h>
#include <Adafruit_MPL115A2.h>
#include "DHT.h"

//Setup the software serial
AltSoftSerial mySerial;

```

```
10  //Setup the Humidity and Temperature Sensor
11  #define DHTPIN 2
12  #define DHTTYPE DHT22   // DHT 22  (AM2302)
13  //Define the Input Pin (Trigger)
14  #define PIN1 A0
15
16  DHT dht(DHTPIN, DHTTYPE);
17
18  //Setup the Pressure Sensor
19  Adafruit_MPL115A2 mpl115a2;
20
21  char data[256];
22  byte index = 0;
23  float Current_T = 0.0;
24  float Current_H = 0.0;
25  float Current_P = 0.0;
26  unsigned long current_time = 0;
27  unsigned long last_time = 0;
28  unsigned long last_time0 = 0;
29  unsigned long last_time1 = 0;
30  long lastDebounceTime = 0;
31  long debounceDelay = 50;
32  int lastTriggerState = LOW;
33  int TriggerState;
34
35  //Script to run on receipt of Trigger Signal
36  void pin1func() {
37
38    //Check for a message from the Beaglebone over UART
39    if (mySerial.available()){
40      index = 0;
41      while(mySerial.available()){
42        data[index] = mySerial.read();
43        index++;
44        data[index] = '\0';
45        delay(5);
46      }
47      //Relay message over USB to computer
48      Serial.println(data);
49    }
50    //Send data to the Beaglebone
51    else
52    {
53      //Reset weather variables
54      char tempH[10];
```

```
55      char tempT[10];
56      char tempP[10];
57      //Read current Temp, Humidity, and Pressure data
58      dtostrf(Current_T, 6, 2, tempT);
59      dtostrf(Current_H, 6, 2, tempH);
60      dtostrf(Current_P, 6, 4, tempP);
61      //Setup the string to send
62      String data_to_send = "Humidity: ";
63      data_to_send += tempH;
64      data_to_send += " %\t Temperature: ";
65      data_to_send += tempT;
66      data_to_send += " *C\t Pressure (kPa): ";
67      data_to_send += tempP; //%4
68      data_to_send += " kPa  ";
69      //Send the data to the Beaglebone
70      mySerial.println(data_to_send);
71    }
72  }
73
74  //Update the current Temp, Pressure, and Humidity from the Sensors
75  void update_info(){
76    float humid = dht.readHumidity();
77    float temp = dht.readTemperature();
78    float pressureKPA = mpl115a2.getPressure();
79
80    Current_T = temp;
81    Current_H = humid;
82    Current_P = pressureKPA;
83  }
84
85  void setup() {
86    //Setup the Hardware UART
87    Serial.begin(9600);
88    Serial.println("----------------------------------------");
89
90    //Initialize the sensors
91    dht.begin();
92    mpl115a2.begin();
93
94    while (!Serial) {
95      ; // wait for serial port to connect. Needed for Leonardo only
96    }
97    // set the data rate for the SoftwareSerial port
98    mySerial.begin(9600);
99    mySerial.println("Hello World");
```

```
}

void loop() {
  current_time = millis();

  //Check the trigger pin
  int reading = digitalRead(PIN1);
  //Make sure the trigger pin is stable
  if (reading != lastTriggerState) {
    lastDebounceTime = millis();
  }
  if ((current_time - lastDebounceTime) > debounceDelay) {
    if (reading != TriggerState) {
      TriggerState = reading;
      //If the trigger is high, indicate on the LED on pin 13 and run Pin1Function
      if (TriggerState == HIGH) {
        digitalWrite(13, HIGH);
        pin1func();
      }
    }
  }
  //Update the values from the sensors every 3 seconds
  if ((current_time - last_time) >= 3000){
    update_info();
    last_time = current_time;
  }
  //Make sure the LED goes low after 1 second
  if ((current_time - last_time1) >= 1000){
    digitalWrite(13, LOW);
    Serial.print(".");
    last_time1 = current_time;
  }
  //If the maximum size of the variables is reached, they will start over from 0.
  //Reset all incremental variables if this happens
  if (current_time < last_time1){
    current_time = 0;
    last_time = 0;
    last_time1 = 0;
  }
}
```

Appendix E

Beaglebone Black Codes

E.1 Introduction

Contained here are the Python 2.7 scripts designed to be operated on the Beaglebone Black.

E.2 DAC and ADC Control

The following code tab was written to work with the MCP3202 ADC, and the MCP4822 and MCP4922 DACs. The differential read mode on the ADC is performed by reading channel 0 followed by channel 1 and subtracting the two.

```
import Adafruit_BBIO.GPIO as GPIO
from Adafruit_BBIO.SPI import SPI
import time

#Define CS
CS_ADC1 = "P8_10";  # Setup the CS for the BBB
CS_ADC2 = "P8_12";
CS_DAC1 = "P8_14";

#Define the CS as output pins
GPIO.setup(CS_ADC1, GPIO.OUT);
GPIO.setup(CS_ADC2, GPIO.OUT);
GPIO.setup(CS_DAC1, GPIO.OUT);

#Turn off all devices on the SPI line.
GPIO.output(CS_ADC1, GPIO.HIGH);
GPIO.output(CS_ADC2, GPIO.HIGH);
GPIO.output(CS_DAC1, GPIO.HIGH);

# Setup the SPI on the BBB
spi = SPI(0,0);          # initialize /dev/spidev1.0
spi.msh = 1500000;       # SPI clock frequency in Hz (1.5 MHz)
spi.bpw = 8;             # Set bits per word. Input a value from 8-16; normally 8
spi.mode = 0;            # Set SPI Mode. Input a value from 0-3; normally 0
set_lsb_first = False;   # Send MSB first
spi.open(0,0);           # Open the device

spi.close()              # Close the device (call at end of script)

#Available clock frequencies:
```

```
##CLKD  Divide by       SPI_CLK [Hz]
##0     1               48.000.000
##1     2               24.000.000
##2     4               12.000.000
##3     8               6.000.000
##4     16              3.000.000
##5     32              1.500.000
##6     64              750.000
##7     128             375.000
##8     256             187.500
##9     512             93.750
##10    1024            46.875
##11    2048            23.438
##12    4096            11.719
##13    8192            5.859
##14    16384           2.930
##15    32768           1.465

def set_dac(cs, channel, gain, shdn, v_set):
    commandbits = 0x0000;               # reset the command-bits. This is 16 binary large
    commandbits ^= (shdn<<12);          # Toggle shutdown?
    commandbits ^= (gain<<13);          # Set the gain to be 1x or 2x?
    commandbits ^= (channel<<15);       # Which Channel?
    commandbits ^= (1<<14);             # Buffered
    commandbits |= (v_set & 0x0FFF);    # Set the desired voltage from (0-4095)*G*2.048
    GPIO.output(cs, GPIO.LOW);          # Select the DAC
    spi.writebytes([commandbits>>8])    # Send the first 8 bits
                                        # (4 command bits and the first 4 data bits)
    spi.writebytes([commandbits&0xFF]) # Send the last 8 data bits
    GPIO.output(cs, GPIO.HIGH);         # DeSelect the DAC
    return;

def read_adc0(cs):
    #channel 0
    GPIO.output(cs, GPIO.LOW);      # Select ADC
    spi.writebytes([0x01]);
    adc_h4 = spi.xfer2([0xA0]);     # unipolar, channel=0, msb first=0, returns B11-B8
                                    # in lower nibble
    adc_l8 = spi.xfer2([0x00]);     # returns B7-B0
    GPIO.output(cs, GPIO.HIGH);     # DeSelect ADC
    #Convert the received bits to binary
    adc_value0 = ((adc_h4[0] & 0x0F) << 8) + adc_l8[0];
    #return the voltage equivalent
    return 3.3 * adc_value0 / 4096;

```

```
def read_adc1(cs):
    #channel 1
    GPIO.output(cs, GPIO.LOW);      # Select ADC
    spi.writebytes([0x01]);
    adc_h4 = spi.xfer2([0xE0]);     # unipolar, channel=0, msb first=0, returns B11-B8
                                    # in lower nibble
    adc_l8 = spi.xfer2([0x00]);     # returns B7-B0
    GPIO.output(cs, GPIO.HIGH);     # DeSelect ADC
    #Convert the received bits to binary
    adc_value1 = ((adc_h4[0] & 0x0F) << 8) + adc_l8[0];
    #return the voltage equivalent
    return 3.3 * adc_value1 / 4096;

def read_diff(cs):
    # This function will read both CH0 and CH1 of the MCP3202 within ?ms +- ?ms
    # Differential measurement
    #Read ADC Channel 1
    GPIO.output(cs, GPIO.LOW);     # Select ADC
    #talk to ADC via hardware SPI
    spi.writebytes([0x01]);
    adc_h41 = spi.xfer2([0xE0]);   # unipolar, channel=1, msb first=0, returns B11-B8
                                   # in lower nibble
    adc_l81 = spi.xfer2([0x00]);   # returns B7-B0
    GPIO.output(cs, GPIO.HIGH);    # DeSelect ADC
    #Read ADC Channel 0
    GPIO.output(cs, GPIO.LOW);     # Select ADC
    spi.writebytes([0x01]);
    adc_h42 = spi.xfer2([0xA0]);   # unipolar, channel=0, msb first=0, returns B11-B8
                                   # in lower nibble
    adc_l82 = spi.xfer2([0x00]);   # returns B7-B0
    GPIO.output(cs, GPIO.HIGH);    # DeSelect ADC
    # Convert the collected bits to a voltage value
    adc_value1 = ((adc_h41[0] & 0x0F) << 8) + adc_l81[0];
    adc_value0 = ((adc_h42[0] & 0x0F) << 8) + adc_l82[0];
    # Return the differential voltage
    return (3.3 * adc_value0 / 4096)-(3.3 * adc_value1 / 4096);

print "Exiting Script";
```

E.3 Xively Weather Station

```
import Adafruit_BBIO.GPIO as GPIO
from Adafruit_I2C import Adafruit_I2C
```

```
3  from Adafruit_BBIO.SPI import SPI
4  import Adafruit_BBIO.UART as UART
5  import serial
6  import time
7  import datetime
8  import xively
9  from socket import error as SocketError
10 from httplib import BadStatusLine
11 import errno
12 from requests import HTTPError
13 from time import sleep
14 from threading import Timer
15 import thread
16 import smtplib
17
18 #**
19 #Use UART 4 or 5 to avoid conflicts with i2c and SPI
20 #i2c:
21 #    P9_19: I2C2, SCL
22 #    P9_20: I2C2, SDA
23 #SPI:
24 # PORT  CS0     D0      DI      SCLK
25 # SPI0  P9_17   P9_21   P9_18   P9_22
26 # SPI1  P9_28   P9_29   P9_30   P9_31 <-- Disabled due to HDMI output
27
28 #Set API information for Xively
29 api = xively.XivelyAPIClient("Q3VjBPqlv5y05HkOneqX3VnILaCMexWgAcEgJgkwbsnTa0IJ");
30 feed = api.feeds.get(841509688);
31
32 #Setup UART
33 # UART          RX      TX      CTS     RTS     Device
34 # UART1         P9_26   P9_24   P9_20   P9_19   /dev/ttyO1
35 # UART2         P9_22   P9_21                   /dev/ttyO2
36 # UART3         P9_42   P8_36   P8_34           /dev/ttyO3
37 # UART4         P9_11   P9_13   P8_35   P8_33   /dev/ttyO4
38 # UART5         P8_38   P8_37   P8_31   P8_32   /dev/ttyO5
39
40 UART.setup("UART1");
41 ser = serial.Serial(port = "/dev/ttyO1", baudrate=9600, timeout=1);
42 ser.close();
43 ser.open();
44
45 #Setup Pins of Interest
46 CS_ADC1 = "P8_12";  # Setup the CS for the BBB
47 CS_ADC2 = "P8_14";
```

```
48  CS_DAC1 = "P8_16";
49  CS_Arduino = "P8_18";
50
51  #Setup timing scripts
52  last_time = time.time();
53  probe = time.time();
54  start = time.time();
55  probed = False;
56  data = '';
57
58  #Setup Weather Variables
59  Temperature = 0;
60  Humidity = 0;
61  Pressure = 0;
62
63  # Watchdog Class
64  class Watchdog(object):
65          def __init__(self, time=1.0):
66                  ''' Class constructor. The "time" argument has the units of seconds. '''
67                  self._time = time
68                  return
69
70          def StartWatchdog(self):
71                  ''' Starts the watchdog timer. '''
72                  self._timer = Timer(self._time, self._WatchdogEvent)
73                  self._timer.daemon = True
74                  self._timer.start()
75                  return
76
77          def PetWatchdog(self):
78                  ''' Reset watchdog timer. '''
79                  self.StopWatchdog()
80                  self.StartWatchdog()
81                  return
82
83          def _WatchdogEvent(self):
84                  '''
85                  This internal method gets called when the timer triggers. A keyboard
86                  interrupt is generated on the main thread. The watchdog timer is stopped
87                  when a previous event is tripped.
88                  '''
89                  print 'Watchdog event...'
90                  self.StopWatchdog()
91                  thread.interrupt_main()
92                  return
```

```
93
94          def StopWatchdog(self):
95                  ''' Stops the watchdog timer. '''
96                  self._timer.cancel()
97
98  def delay(ms):
99          time.sleep(ms/1000.0);
100
101 def setup():
102
103         #Define the CS as output pins
104         GPIO.setup(CS_ADC1, GPIO.OUT);
105         GPIO.setup(CS_ADC2, GPIO.OUT);
106         GPIO.setup(CS_DAC1, GPIO.OUT);
107         GPIO.setup(CS_Arduino, GPIO.OUT);
108
109         #Turn off all devices.
110         GPIO.output(CS_ADC1, GPIO.HIGH);
111         GPIO.output(CS_ADC2, GPIO.HIGH);
112         GPIO.output(CS_DAC1, GPIO.HIGH);
113
114         GPIO.output(CS_Arduino, GPIO.LOW);
115         delay(200);
116
117         print "Starting Up!"
118
119         GPIO.output(CS_Arduino, GPIO.HIGH);
120         delay(200);
121         GPIO.output(CS_Arduino, GPIO.LOW);
122
123 def main():
124         global data, probed, start, probe, last_time
125         global Temperature, Humidity, Pressure
126
127         w = Watchdog(10.0) #10 second watchdog
128         w.StartWatchdog()
129
130         try:
131                 if (int(time.time() - probe)>=(60)):
132                         #Probe Arduino for Data
133                         GPIO.output(CS_Arduino, GPIO.HIGH);
134                         probe = time.time();
135                         probed = True;
136
137                 #Check if Data is waiting to be parsed
```

```
138                 if (ser.inWaiting()):
139                         data = '';
140                         #Rebuild data
141                         while(ser.inWaiting()):
142                                 last_time = time.time();
143                                 data += ser.read(1);
144                                 delay(5);
145                                 GPIO.output(CS_Arduino, GPIO.LOW);
146                                 #data_in += str(data);
147                                 probed = False;
148                         #Extract Temp, Humidity, and Pressure
149                         Temperature = data.split(" ")[6];
150                         Humidity = data.split(" ")[2];
151                         Pressure = data.split(" ")[10];
152
153                         #Get the date and time
154                         now = datetime.datetime.utcnow();
155                         #Setup the Xively data stream
156                         feed.datastreams = [
157                                 xively.Datastream(id='Temperature', current_value=
158                                                 Temperature, at=now),
159                                 xively.Datastream(id='Humidity', current_value=
160                                                 Humidity, at=now),
161                                 xively.Datastream(id='Pressure', current_value=
162                                                 Pressure, at=now)
163                                 ];
164
165                         #Send the data to Xively
166                         try:
167                                 feed.update();
168                                 #print "Values Pushed to Xively";
169                         except HTTPError as e:
170                                 print "Error connecting to Xively: " + str(e);
171                         except SocketError as e:
172                                 print "Socket Error: " + str(e);
173                         except BadStatusLine as e:
174                                 print "Bad Status Line Error";
175                         except Exception as e:
176                                 print "some other error: " + str(e);
177
178                 #Reset the trigger for the Arduino
179                 if ((abs(int(probe) - int(time.time()))>10) and (probed == True)):
180                         print "Resetting GPIO pin!";
181                         probed = False;
182                         GPIO.output(CS_Arduino, GPIO.LOW);
```

```
183
184                 delay(200);
185         except:
186                 print 'MAIN THREAD KNOWS ABOUT WATCHDOG'
187
188         w.StopWatchdog() # Not always necessary
189         return
190
191 print "Starting"
192 setup();
193 while 1:
194         if __name__ == '__main__':
195                 main()
```

Appendix F

Circuits

F.1 Introduction

Contained here are the board layouts for the circuits designed using CadSoft EAGLE v6 freeware. Each circuit is separated into its appropriate section. Note that each circuit contains grounded mounting holes of at least one matching dimension to one other to support the stacking of each and every module.

F.2 Power Supply

This power supply is used to power the ETV and is controlled via SPI from either a LabJack, Arduino, or other microcontroller. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware. In Figures F.1 and F.2, the top layer of the board is in red, while the bottom layer is in blue. Silkscreen is presented in white.

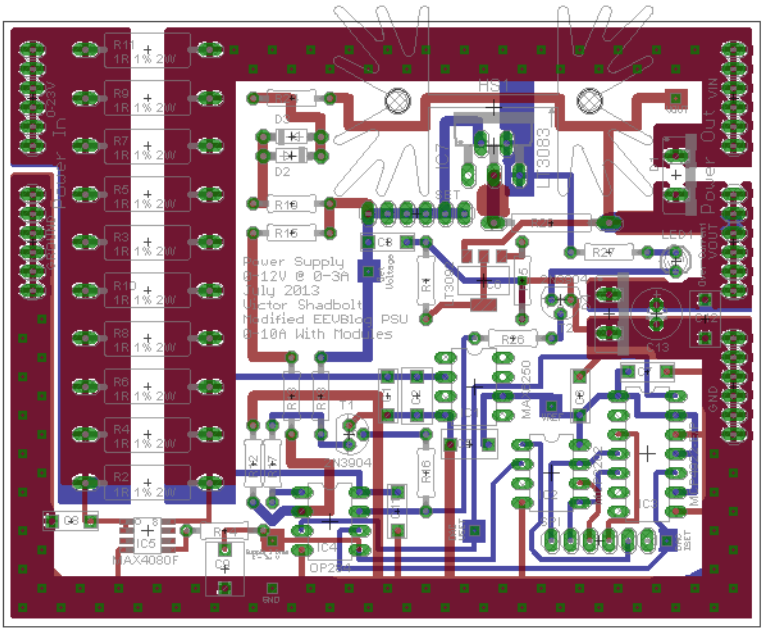


Figure F.1: Power Supply Module 1 with Control Circuitry. Image is actual size.

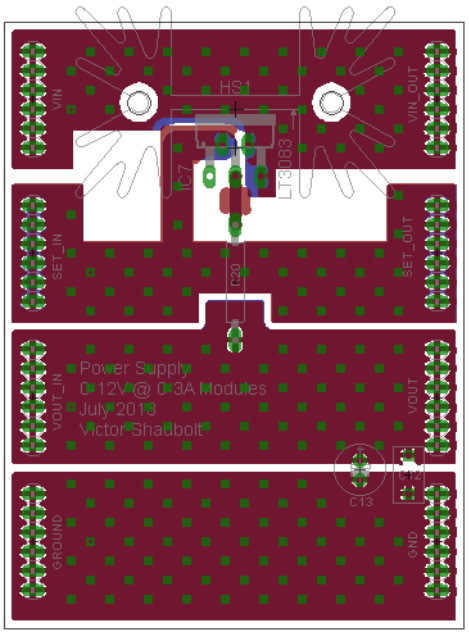


Figure F.2: Power Supply Module 2 which can be daisy chained to increase current handling capabilities. Image is actual size.

F.3 Photodiode Board

This Photodiode Board is used to monitor a photodiode over SPI in order to drastically improve detection capability and noise rejection. It can only be used in conjunction with an Interface Board unless modifications are made. Power is delivered over the DB9 cable, however, external power may be supplied. Protection diodes provide reverse polarity protection and prevents back-driving of the Interface Board. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.3 and F.4 respectively.

F.3.1 Schematic

The schematic is separated into 11 sections: Buffer Power, Buffers, Board Power, Protection circuit, Voltage divider, ADC, Reference voltage supply, Power regulator, Mounting holes, SPI connections, and Photodiode circuit.

The Photodiode board is sensitive to power fluctuations and noise on the power rails due to the level of gain required for detecting fluorescent signals. In order to remove any harmonics the board power (16.75→26 Volts (V)) is down-regulated to 15 V by a linear regulator with a high (64 dB minimum) ripple rejection. To further the quality of the photodiode readings a 3.3 V voltage reference is used for the ADC, which can measure the output of both stages of the photodiode output.

The protection diode between the DB9 connector and board power is used to protect from back-driving the connection when the board is powered externally. To add to this, jumpers have been added to the schematic to enable or disable the transmission of the output of the photodiode circuit through the DB9 connection (used for streaming mode on the LabJack).

The voltage divider maps the full 15 V range down to 3.3 V maximum for compatibility with the ADC, which is controlled via SPI over the DB9 connection. These SPI rails are buffered to further remove any digital noise.

The mounting holes (4-40 sized) are all grounded, and have been positioned such that stacking of multiple boards is possible while sharing a common ground to reduce potential ground loops.

The photodiode circuit itself is functional in a passive sense in that no external control is required to gain an output. Terminal blocks have been added to increase the flexibility of the circuit.

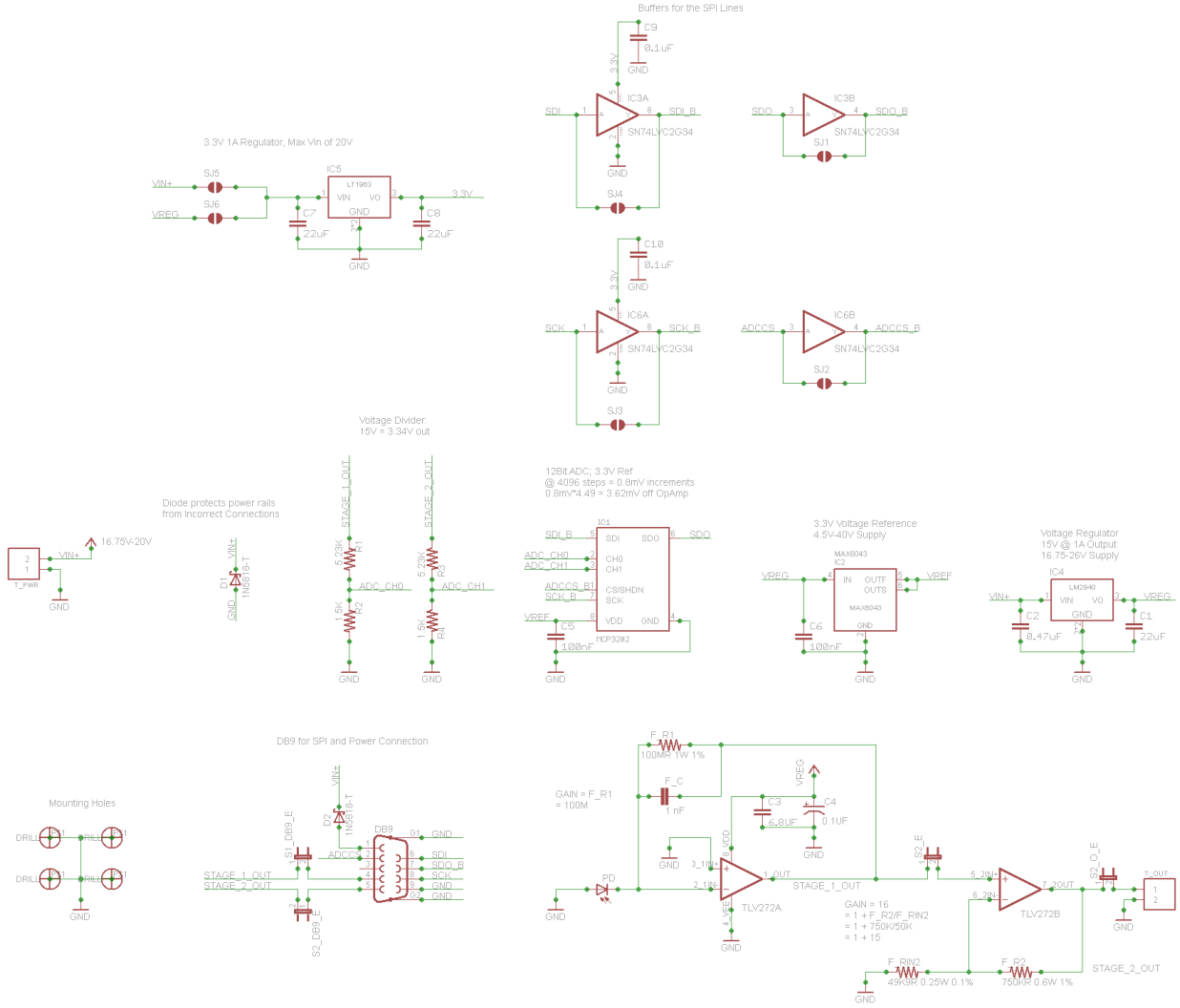


Figure F.3: Schematic diagram for the Photodiode Board.

F.3.2 Board Layout

In general the power input and the amplified photodiode output is on the top of the board, the photodiode itself is in the bottom center of the board, and the DB9 connector for digital communication is located in the top center of the board. As can be seen in Figure 4.5, a silkscreen layer is used to identify the position of components as well as the specifications for the inputs.

The diodes protects the DB9 connector from being back-driven if the board is to be powered externally, as well as protects the board from being supplied a negative voltage.

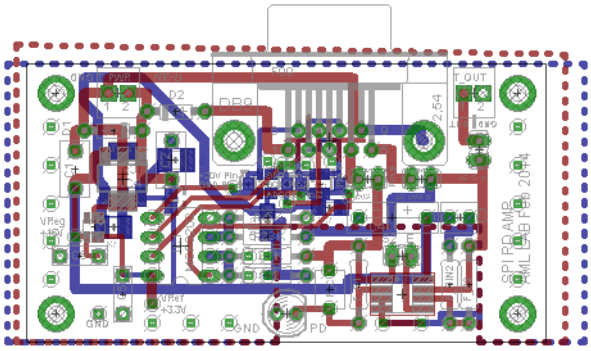


Figure F.4: Board layout for the Photodiode Board circuit. The top layer of the board is in red, while the bottom layer is in blue. Silkscreen is presented in white. Image is actual size.

F.3.3 Populating

Populating the PD board is a simple matter. It is best to start with the surface mount components, specifically the Buffers, OpAmp, voltage reference, and regulators. Then populating components in order of shortest to tallest allows for the best solder joints to be made. Figure 4.5 demonstrates the look of a populated PD board. Note that the tantalum capacitor is polarized and its conical end is positive. Table F.1 contains the parts list.

Table F.1: Photodiode Board Parts List

Description	Qty	Parts	DIGIKEY
Terminal Block 2 Position	2	T_{OUT}, T_{PWR}	A98333-ND
SUB-D	1	DB9	609-4009-ND
JUMPER	4	S1_DB9_E, S2_DB9_E, S2_E, S2_O_E	609-3469-ND
15V @ 1A Linear Low Dropout Regulator	1	IC4	LM2940IMP-15/NOPBCT-ND
3.3V Linear Regulator	1	IC5	LT1963AEST-3.3#PBF-ND
5mm Photodiode	1	PD	751-1001-ND
TLV272D Operational Rail-to-Rail OpAmp	1	TLV272	296-26806-1-ND
Cap Tant. 0.1uF	1	C4	718-1450-1-ND
CAP CER 0.1UF 25V 5% 1206	2	C9, C10	587-1147-1-ND
FK22 0.47uF Ceramic Cap	1	C2	445-2649-ND
Cap Cer 1nF FK28	1	F_C	445-5246-ND
Resistor 1.5K 1/4W 207/10	2	R2, R4	1.50KXBK-ND
Resistor 100M 1/4W 207/10	1	F_R1	MOX200F-100ME-ND
0.1uF 63V 5% Film Capacitor MKT370	2	C5, C6	3013PH-ND
1.0A SCHOTTKY BARRIER RECTIFIER	2	D1, D2	1N5817DICT-ND
CAP CER 22uF 25V 10% X7R 1210	2	C7, C8	1276-3392-1-ND
FK22 22uF Ceramic Cap	1	C1	445-8480-ND
Resistor 49.9K 1/4W 207/10	1	F_RIN2	49.9KXBK-ND
Resistor 5.23K 1/4W 207/10	2	R1, R3	5.23KXBK-ND
Cap Cer 6.8uF FK16	1	C3	445-8355-ND
Resistor 750K 1/4W 207/10	1	F_R2	750KXBK-ND
Precision high Voltage Reference	1	IC2	MAX6043BAUT33#TG16CT-ND
12 Bit ADC MCP3202	1	IC1	MCP3202-CI/P-ND
Single Bus Dual Buffer Gate	2	IC3, IC6	296-13496-1-ND

F.3.4 Usage

The PD board typically utilizes a 20 V, 200 mAmperes (A) supply.

Every jumper is labelled clearly as to its function and is summarized in Table F.2.

Tables F.3 and F.4 list the inputs and outputs.

Table F.2: Photodiode Board Jumpers

Jumper	Description
S2_E	Stage 2 Enable: Enables the second stage of the photodiode amplifier circuit.
S1_DB9_E	Stage 1 DB9 Enable: Transmits the output of the first stage of the amplification over the DB9 connector. Note that this could introduce noise on the output which will then be amplified in Stage 2.
S2_DB9_E	Stage 2 DB9 Enable: Transmits the output of the second stage of the amplification over the DB9 connector.
S2_O_E	Stage 2 Output Enable: Connects the output of the second stage to the output terminal block. It is possible to connect S2_E to S2_O_E to output the results of the first stage.

Table F.3: Photodiode Board Inputs and Power Supply

Input	Part(s)	Description
Power	DB9 Pin 1 or T_{PWR} Pin 2	17→26 V board supply power
GND	DB9 Pin 9 or T_{PWR} Pin 1	Circuit ground
ADC	DB9 Pins 2, 6, 7, and 8 (IC8 and CIO1→CIO2)	Output of both stages in digital form over SPI communication

Table F.4: Photodiode Board Outputs

Output	Part(s)	Description
PD Stage 2	T_{OUT} and DB9 Pin 5	Output of the second stage of the PD amplifier
PD Stage 1	DB9 Pin 4	Output of the first stage of the PD amplifier
ADC	DB9 Pins 2, 6, 7, and 8 (IC8 and CIO1→CIO2)	Output of both stages in digital form over SPI communication

F.4 Solenoid Board

The Solenoid Board allows for direct connection to the LabJack or to the Interface Board Extension. When connected to the Interface Board Extension, no external power is required as the power is provided through the DB15 connector itself. A protection diode prevents back-driving the Interface Board Extension or the LabJack pins if an external power source is supplied. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.5 and F.6 respectively.

F.4.1 Schematic

The schematic is separated into two sections: Mounting Holes and Board Power, and Solenoid control.

The mounting holes are all grounded, and have been positioned such that stacking of multiple boards is possible while sharing a common ground to reduce potential ground loops.

Six digital signals control the six MOSFET drivers, which in turn control the state of up to six connected solenoid valves. The capacitors surrounding the MOSFET drivers are for smoothing the switching transition as well as for supplying the surge current required when performing the switch between the OFF and ON states.

The six outputs are toggled between board supply voltage and ground by the digital control signals sent to the MOSFET drivers.

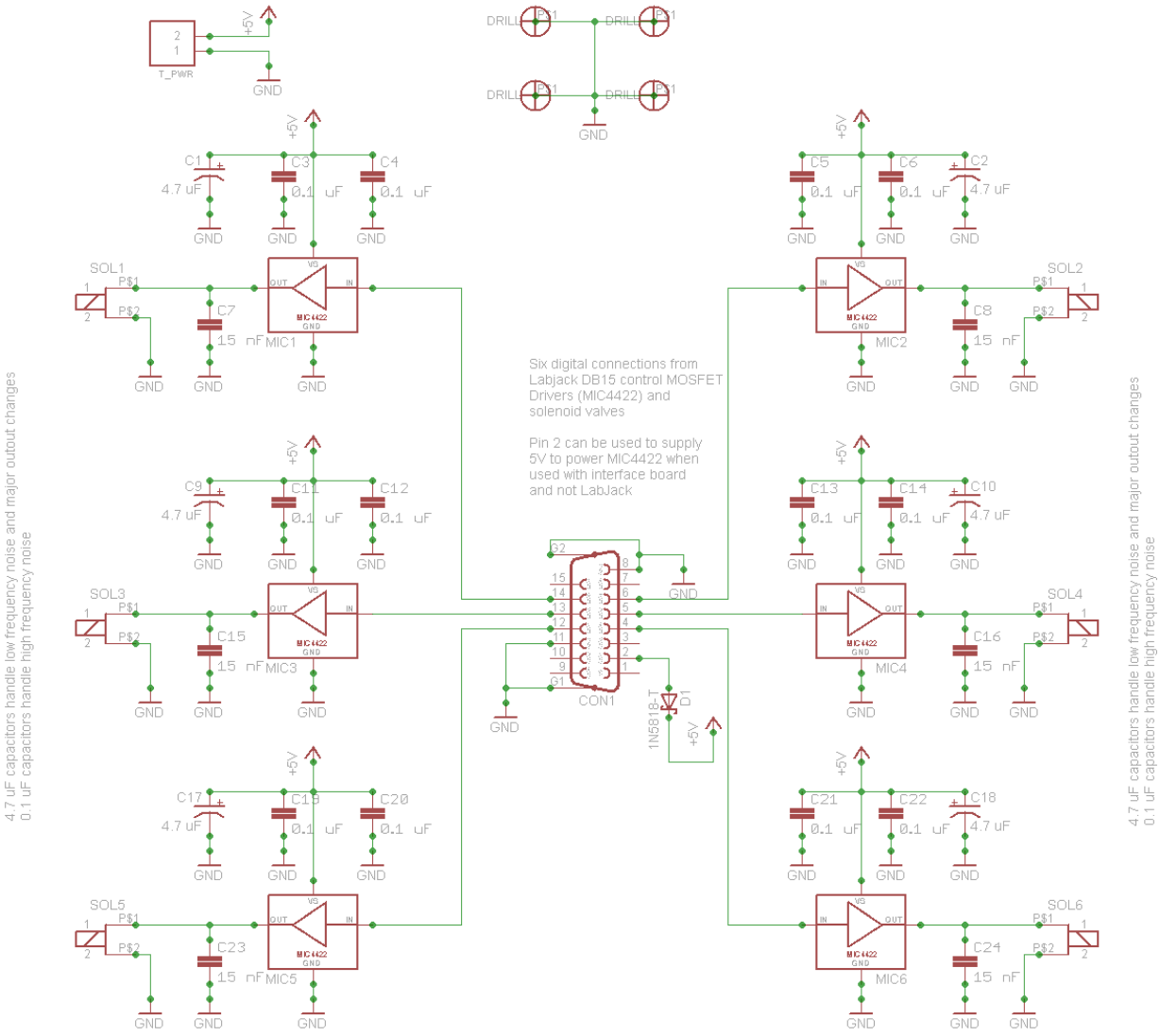


Figure F.5: Schematic diagram for the Solenoid Board.

F.4.2 Board Layout

In general the digital control is on the left side of the board, the power input is on the bottom of the board, the MOSFET drivers in the center of the board, and outputs on the right side of the board. As can be seen in Figure F.6, a silkscreen layer is used to identify the position of components as well as the specifications for the inputs. The symmetric layout allows for quick debugging.

The diode protects the DB15 connector from being back-driven if the board is to be powered externally (such as is the case when the board is being used with a LabJack).

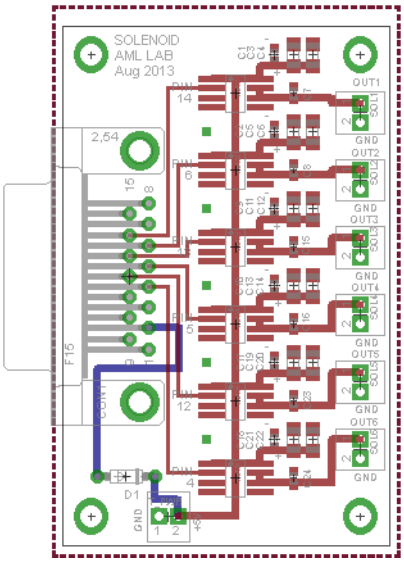


Figure F.6: Board layout for the Solenoid Board circuit. The top layer of the board is in red, while the bottom layer is in blue. Silkscreen is presented in white. Image is actual size.

F.4.3 Populating

Populating the Solenoid board is a simple matter. It is best to start with the surface mount components, specifically the MOSFET drivers followed by the capacitors. Then populating components in order of shortest to tallest allows for the best solder joints to be made. Figure 4.6 demonstrates the look of a populated Solenoid board. Table F.5 contains the parts list.

Table F.5: Solenoid Board Parts List

Description	Qty	Parts	DIGIKEY
Terminal Block 2 Position	7	SOL_1, SOL_2, SOL_3, SOL_4, SOL_5, SOL_6, T_{PWR}	A98333-ND
DB15 Right-Angle Connector	1	CON1	182-15FE-ND
MIC4422 IC DRIVER MOSFET 9A LOSIDE 8SOIC	6	MIC1, MIC2, MIC3, MIC4, MIC5, MIC6	576-1194-1-ND
CAP CER 0.1UF 25V 5% 1206	12	C3, C4, C5, C6, C11, C12, C13, C14, C19, C20, C21, C22	587-1147-1-ND
CAP CER 0.015UF 25V 10% X7R 0603	6	C7, C8, C15, C16, C23, C24	445-5102-1-ND
1.0A SCHOTTKY BARRIER RECTIFIER	1	D1	1N5817DICT-ND
CAP CER 4.7UF 25V 10% JB 0603	6	C1, C2, C9, C10, C17, C18	445-11253-1-ND

F.4.4 Usage

The Solenoid board requires a 5 V, 600 mA supply. Tables F.6 and F.7 list the inputs and outputs.

Table F.6: Solenoid Board Inputs and Power Supply

Input	Part(s)	Description
Power	DB15 Pin 2 or T_{PWR} Pin 2	5 V board power
GND	DB15 Pins 8 and 11 or T_{PWR} Pin 1	Circuit ground
MOSFET Control	DB15 Pins 4, 5, 6, 12, 13, and 14 (EIO0→EIO5)	MOSFET Driver Control

Table F.7: Solenoid Board Outputs

Output	Part(s)	Description
OUT_1	SOL_1	Output of MIC 1, Power or Ground
OUT_2	SOL_2	Output of MIC 2, Power or Ground
OUT_3	SOL_3	Output of MIC 3, Power or Ground
OUT_4	SOL_4	Output of MIC 4, Power or Ground
OUT_5	SOL_5	Output of MIC 5, Power or Ground
OUT_6	SOL_6	Output of MIC 6, Power or Ground

F.5 Fluidics Board

The Fluidics Board provides high-voltage for the use of Capillary Electrophoresis (CE) on the microfluidic chips. In addition, because of its design, the high-voltage module may be removed and relays changed in order to toggle wall-voltage for use in extremely high power applications such as in the FAES. The board supports direct connection to a LabJack in addition to the Interface Board. When used in conjunction with an Interface Board, no other external connections are required as power is delivered over the DB15 cable. Diodes prevent back-driving the LabJack or Interface Board if different power levels are desired. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.7 and F.8 respectively.

F.5.1 Schematic

The schematic is separated into three sections: Mounting Holes and Board Power, Relay control, and Outputs.

The mounting holes are all grounded, and have been positioned such that stacking of multiple boards is possible while sharing a common ground to reduce potential ground loops.

Four digital signals control the four MOSFET drivers, which in turn control the state of four relays. The capacitors surrounding the MOSFET drivers are for smoothing the switching transition as well as for supplying the surge current required when performing the switch between the OFF and ON states.

The high voltage module is isolated from the rest of the circuit and supplies Relay 3 and Relay 4 with the high voltages needed for separations. Relay 1 and Relay 2 control the ground output.

F.5.2 Board Layout

In general the digital control and power inputs are on the left side of the board, the MOSFET drivers and high-voltage generation in the center of the board, and outputs on the right side of the board. A silkscreen layer is used to identify the position of components as well as the specifications for the inputs. The symmetric layout allows for quick debugging.

The diodes protect the DB15 connector from being back-driven if the board is to be powered externally (such as is the case when the board is being used with a LabJack).

F.5.3 Populating

Populating the Fluidics board is a simple matter. It is best to start with the surface mount components, specifically the MOSFET drivers followed by the capacitors. Then populating components in order of shortest to tallest allows for the best solder joints to be made. Figure 4.4 demonstrates the look of a populated Fluidics Board. Table F.8 contains the parts list.

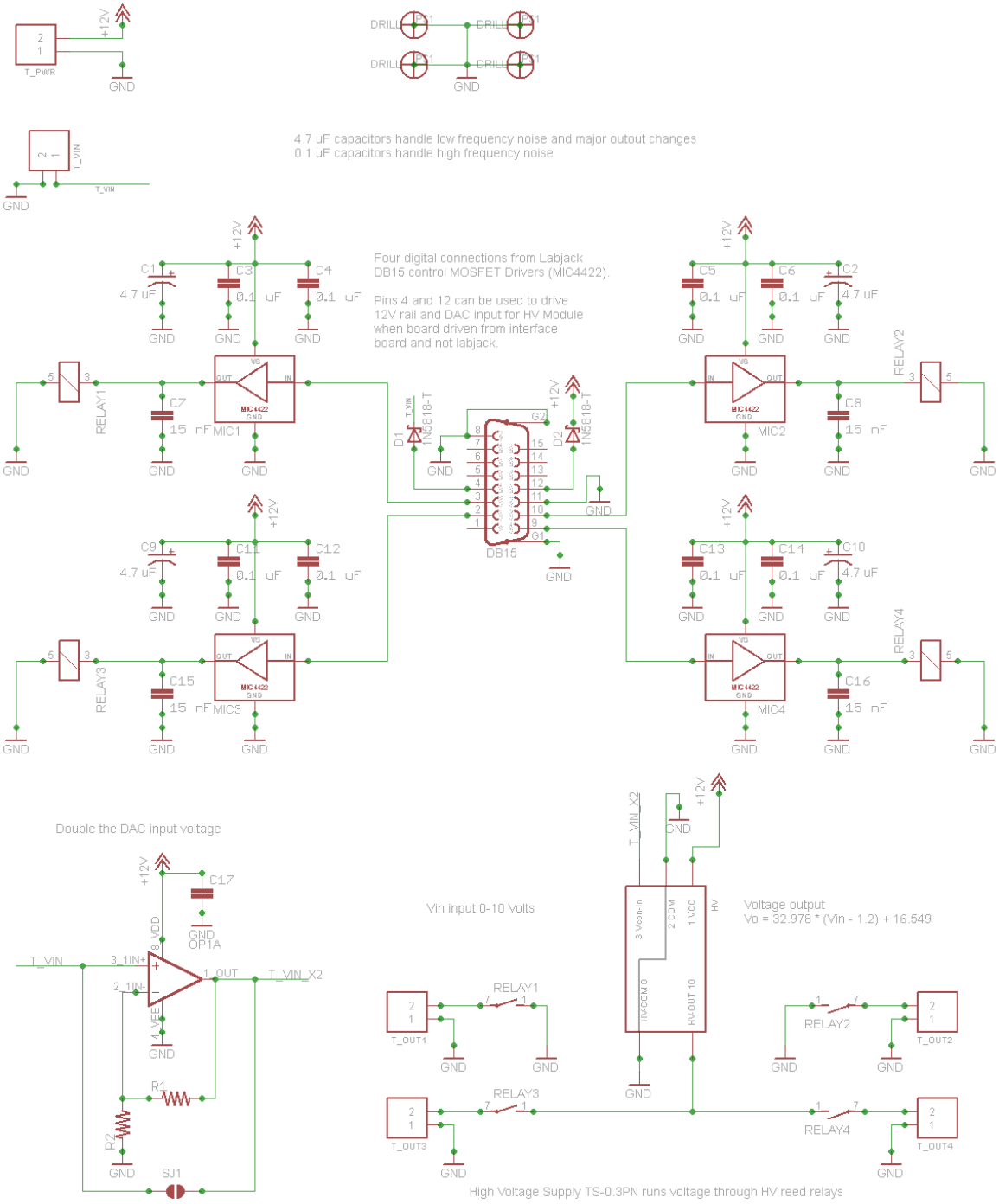


Figure F.7: Schematic diagram for the Fluidics Board.

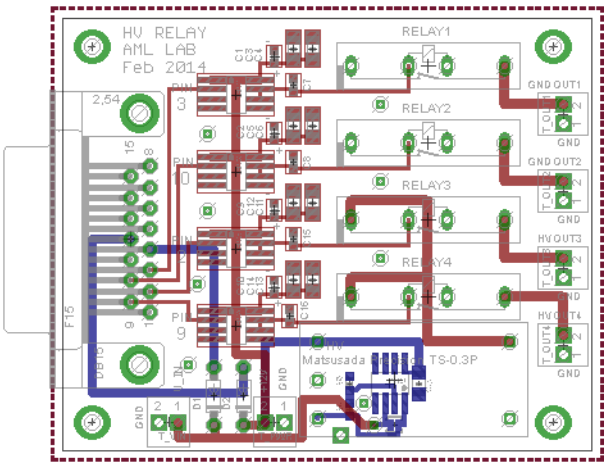


Figure F.8: Board layout for the Fluidics Board circuit. The top layer of the board is in red, while the bottom layer is in blue. Silkscreen is presented in white. Image is actual size.

Table F.8: Fluidics Board Parts List

Description	Qty	Parts	DIGIKEY
Terminal Block 2 Position	6	T_{OUT_1}, T_{OUT_2}, T_{OUT_3}, T_{OUT_4}, T_{PWR}, T_{VIN}	A98333-ND
CAP CER 0.1UF 25V 5% 1206	1	C17	587-1147-1-ND
12V Relay	4	RELAY1, RELAY2, RELAY3, RELAY4	306-1196-ND
DB15 Right-Angle Connector	1	DB15	182-15FE-ND
MIC4422 IC DRIVER MOSFET 9A LOSIDE 8SOIC	4	MIC1, MIC2, MIC3, MIC4	576-1194-1-ND
Resistor 10K 1/10W 5% 0603	2	R1, R2	311-10KGRCT-ND
TLV272D Operational Rail-to-Rail OpAmp	1	OP1	296-26806-1-ND
Matsusada TS-0.3P	1	HV	
CAP CER 0.1UF 25V 5% 1206	8	C3, C4, C5, C6, C11, C12, C13, C14	587-1147-1-ND
CAP CER 0.015UF 25V 10% X7R 0603	4	C7, C8, C15, C16	445-5102-1-ND
1.0A SCHOTTKY BARRIER RECTIFIER	2	D1, D2	1N5817DICT-ND
CAP CER 4.7UF 25V 10% JB 0603	4	C1, C2, C9, C10	445-11253-1-ND

F.5.4 Usage

The Fluidics Board requires a 12 V, 400 mA supply. The analog input requires 1.2→5 V to control the high-voltage outputs following Equation 4.2. Tables F.9 and F.10 list the inputs and outputs.

Table F.9: Hight Voltage Relay Board Inputs and Power Supply

Input	Part(s)	Description
V_{IN}	DB15 Pin 4 or V_{IN} Pin 1	0→5 V analog control of DC-DC converter
V_{PWR}	DB15 Pin 12 or V_{PWR} Pin 2	12 V board power
GND	DB15 Pins 8 and 11 or V_{IN} Pin 2 or V_{PWR} Pin 1	Circuit ground
MOSFET Control	DB15 Pins 2, 3, 9, and 10 (CIO0→CIO3)	MOSFET Driver Control

Table F.10: Hight Voltage Relay Board Outputs

Output	Part(s)	Description
GND_{OUT_1}	T_{OUT_1}	Output of Relay 1, GND or Floating
GND_{OUT_2}	T_{OUT_2}	Output of Relay 2, GND or Floating
HV_{OUT_3}	T_{OUT_3}	Output of Relay 3, HV or Floating
HV_{OUT_4}	T_{OUT_4}	Output of Relay 4, HV or Floating

F.6 LT3092 Board (Constant Current Source)

The LT3092 Constant Current Source Board is a breakout board for the LT3092. Two terminal blocks provide the input and output to the board. when the R_SET resistor is unpopulated the LT3092 may be controlled via an external DAC output of 0→1 V. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.9 and F.10 respectively.

F.6.1 Schematic

The schematic is very simple. The mounting holes are all grounded, and have been positioned such that stacking of multiple boards is possible while sharing a common ground to reduce potential ground loops.

The input of the constant current source is connected to the ground of the output terminal in order to allow the LT3092 to control the total current flowing through the connected device. The set and output resistors control the current in a static fashion following Equation 4.4. DAC control is performed by controlling the voltage level at the set pin of the LT3092.

Voltage supplied to the board via T_{IN} is passed through to LOAD.

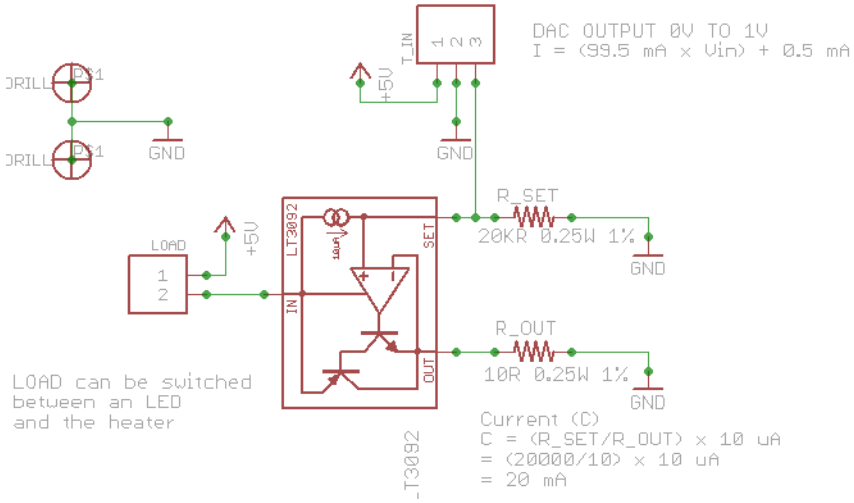


Figure F.9: Schematic diagram for the LT3092 Constant Current Source Board.

F.6.2 Board Layout

In general the inputs to the board (power, and DAC control if it is to be used) are located on the right side of the board, while the output is located on the left. As can be seen in Figure F.10, a silkscreen layer is used to identify the position of components as well as the specifications for the power input. The simple layout allows for quick debugging.

Note that DAC control takes an analog voltage of 0.2→1 V to control the current output from 20 mA to 100 mA following Equation 4.3. If resistor set current is to be used, follow Equation 4.4.

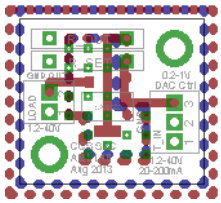


Figure F.10: Board layout for the LT3092 Constant Current Source Board circuit. The top layer of the board is in red, while the bottom layer is in blue. Silkscreen is presented in white. Image is actual size.

F.6.3 Populating

Populating the LT3092 board is a simple matter. It is best to start with the surface mount constant current source (*LT3092*) followed by the resistors. The remaining components can then be added to the board. Table F.11 contains the parts list.

Table F.11: LT3092 Parts List

Description	Qty	Parts	DIGIKEY
Terminal Block 2 Position	1	LOAD	A98333-ND
Terminal Block 3 Pos	1	T_{IN}	A98334-ND
LT3092 200mA IC CURRENT SOURCE 1% SOT223-3	1	LT3092	LT3092EST#PBF-ND
Resistor: User Selectable	2	R_{OUT}, R_{SET}	

F.6.4 Usage

The LT3092 board requires a minimum of 1.2 V at 200 mA to be supplied for operation. The actual power required for the device being controlled should be supplied, up to a maximum of 40 V. The analog input of 0.2→1 V from a DAC controls the current output of the board between 20→100 mA. Tables F.12 and F.13 list the inputs and outputs.

Table F.12: LT3092 Board Inputs and Power Supply

Input	Part(s)	Description
DAC Control	T_{IN} Pin 3	0.2→1 V analog control of constant current source
Board Ground	T_{IN} Pin 2	Board ground
Supply Voltage	T_{IN} Pin 1	1.2→40 V supply

Table F.13: LT3092 Board Outputs

Output	Part(s)	Description
GND_{OUT}	LOAD Pin 2	Controlled ground output between 20→200 mA when set using resistors, 20→100 mA when controlled via DAC
Supply Voltage	LOAD Pin 1	Voltage supplied to the board via T_{IN} is passed through LOAD Pin 1

F.7 Interface Board

The Interface Board and its extension, the Interface Board Extension, were both developed in CadSoft EAGLE v6 Freeware. Their purpose is to expand the connectivity of the LabJack U6 in order to support the simultaneous use of all modules required for the FAES and the Restriction Fragment Length Polymorphism (RFLP) systems when used either individually or when combined as a single instrument. The interface board connects to the LabJack through a DB37 cable, and connects to the extension board through an IDE cable. The base Interface Board is able to support dual DAC outputs, a single Fluidics Board, and a single Photodiode Board. The extension supports the addition of dual DAC outputs, dual Solenoid Boards, and dual ETV Power Supply Control Boards.

The schematic for the Interface Board is separated into Figures F.11 and F.12 while the board layout is presented in Figure F.14. The Interface Board Extension schematic and board layout are presented in Figures F.13 and F.16 respectively.

F.7.1 Schematic

The schematic is separated into ten sections: LabJack DB37 Input, PhotoDiode Board Connection, Fluidics Board Connection, 4-40 Mounting Holes, Power Rails, DAC Output, 3.3↔5 V Logic Level Converter, Inter-Board Connection, Shift Registers, and Buffers.

Each section is an individual module which allows for easy copy-paste for further board design. Furthermore the schematic and board design was completed within the freeware version of CadSoft Eagle v6.

The mounting holes, sized for 4-40 screws, are all grounded, and have been positioned such that stacking of multiple boards is possible while sharing a common ground to reduce potential ground loops.

The power rails section is split into multiple sections itself. The power-input from the 2.1 mm Direct Current (DC) power jack is rectified such that regardless of how the 12 V power is supplied, it is impossible to provide a negative power supply. This leads to guaranteeing proper power is supplied to the connected boards. More specifically the power rails section generates a 3.3 V rail, a 5 V rail, passes through the 12 V rail, and through a DC-DC boost regulator generates a 20 V rail.

The logic level converter is required as the LabJack operates off 3.3 V logic. As a number of devices (ie DAC's) currently operate off 5 V logic a conversion was required.

The shift registers are utilized in order to boost the number of digital outputs. the DB37 connection on the LabJack is limited in the number of connections thus a SPI communication was utilized in order to increase the number of digital outputs to 24. One shift register is used as a chip-select generator for SPI communications with many onboard components. The two daisy-chained shift-registers are used to control the digital signals sent to the Solenoid boards, and Fluidics board.

The inter-board connection utilizes a 40-pin IDE cable to connect to an expansion board. The expansion board has the capacity to connect to two solenoid boards as well as two ETV boards.

The DAC output section is used to increase the number of DAC outputs on the LabJack. More specifically the LT3092 board utilizes a DAC control for dynamic current setting. The DAC section has a jumper to switch the output between 0→5 V and 0→1 V passively.

In addition, terminal blocks are added to enable access to the power rails as well as the DAC outputs on the board.

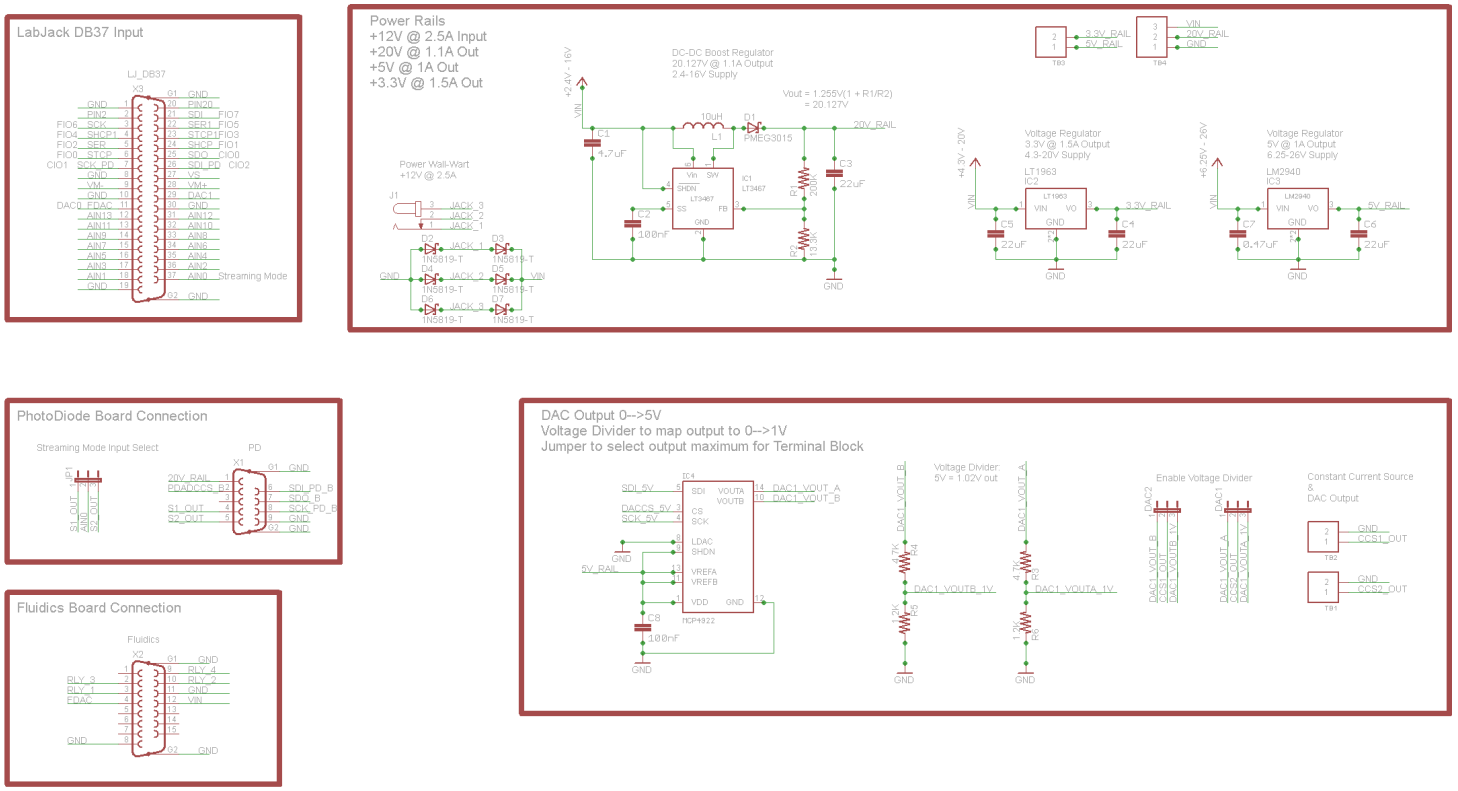


Figure F.11: Schematic diagram part A for the Interface Board.

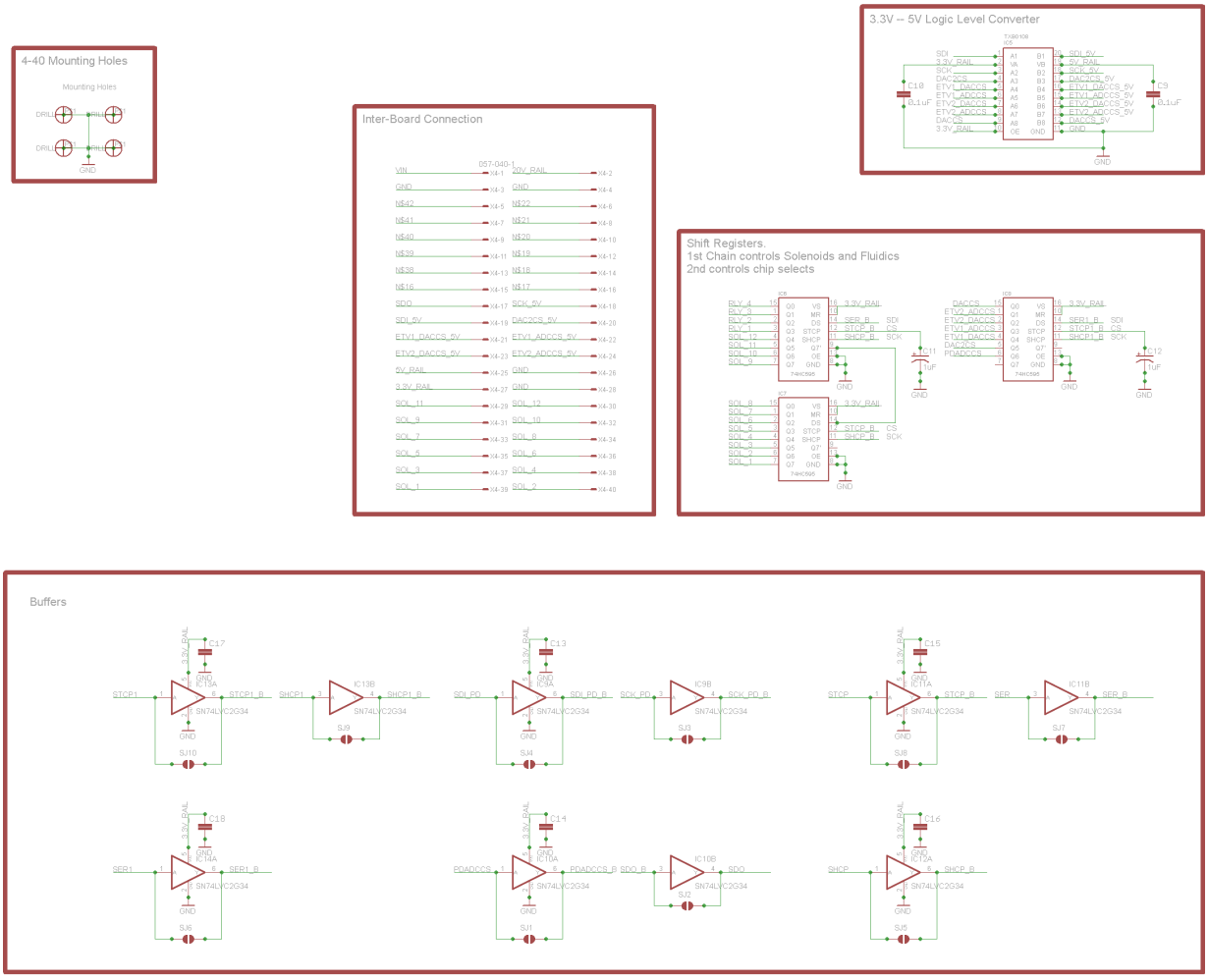


Figure F.12: Schematic diagram part B for the Interface Board.

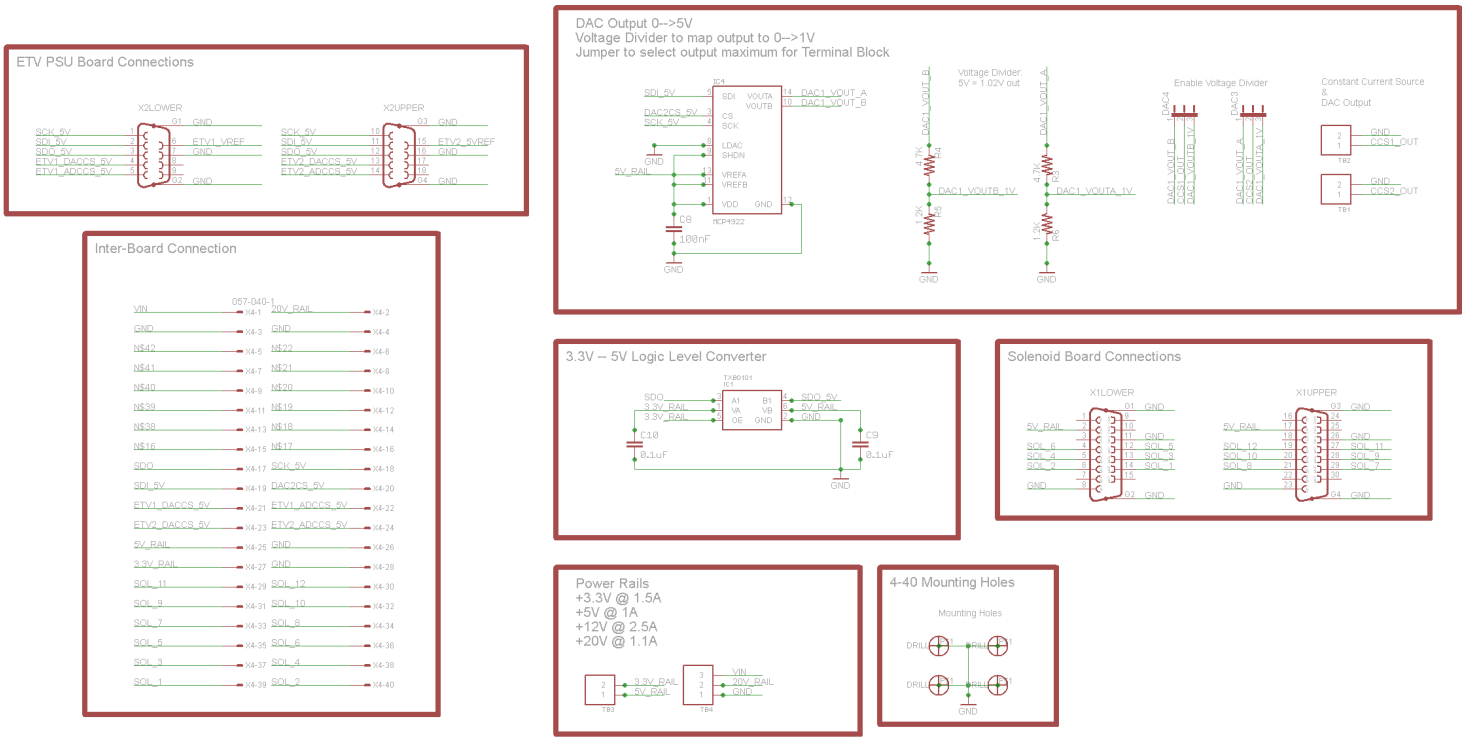


Figure F.13: Schematic diagram for the Interface Board Extension.

F.7.2 Board Layout

In general the digital control and power inputs are on the top side of the board, the power-rail outputs are on the right-side of the board. The board connections are located along the bottom along with the DAC outputs. As can be seen in Figure F.15, a silkscreen layer is used to identify the position of components as well as the specifications for the inputs.

As each module in the schematic is separated from other aspects the debugging process can occur in stages. Each individual "module" can be checked for consistency and function. It is ideal to check the power-rails first. Following this it is ideal to check the DAC output from the onboard DAC. This serves the purpose of checking the logic-level converter as well as the shift-registers used for the SPI communication. The next module to check should be the Fluidics board connection which will confirm the remaining shift registers. Lastly the photodiode connection should be checked.

The interface board primarily increases the number of digital outputs and capability of a single LabJack in order to be able to control multiple different boards simultaneously. Jumpers are used to make local configuration changes to the DAC as well as the streaming-mode input from the PD board.

F.7.3 Populating

Populating the Interface board can be tricky. It is best to start with the surface mount Integrated Circuits (ICs), specifically the shift registers, the buffers, the logic-level converter, boost regulator, and linear voltage regulators. Following this, populating the board with components in order of shortest to tallest allows for the best solder joints to be made. The Sub-D connectors should be added last. Table F.14 contains the parts list for the Interface Board, and Table F.15 contains the part list for the Interface Board Extension. Figure F.15 demonstrates the look of a populated Interface board.

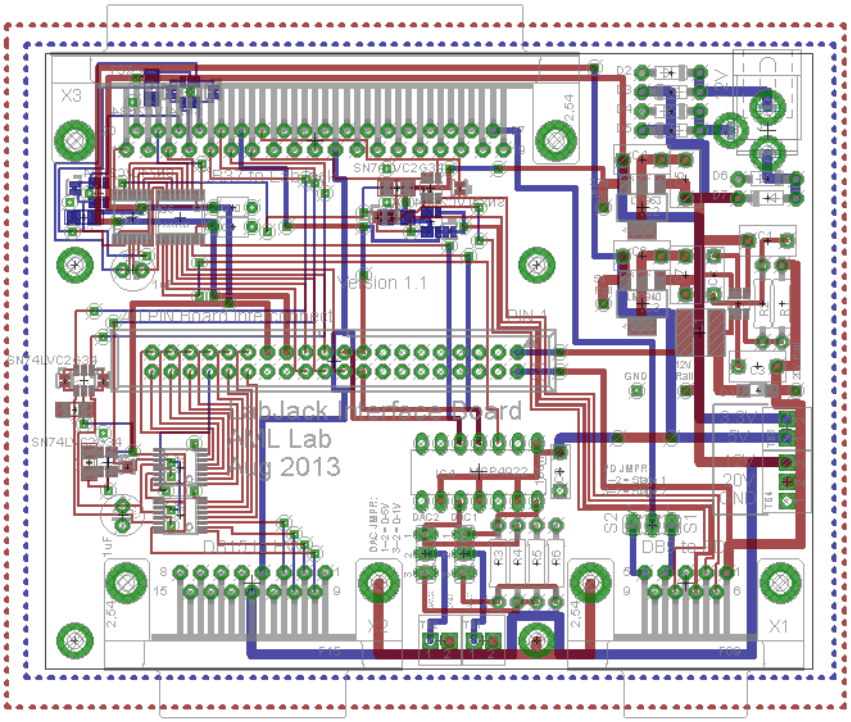


Figure F.14: Board layout for the Interface Board circuit. The top layer of the board is in red, while the bottom layer is in blue. Silkscreen is presented in white. Image is actual size.

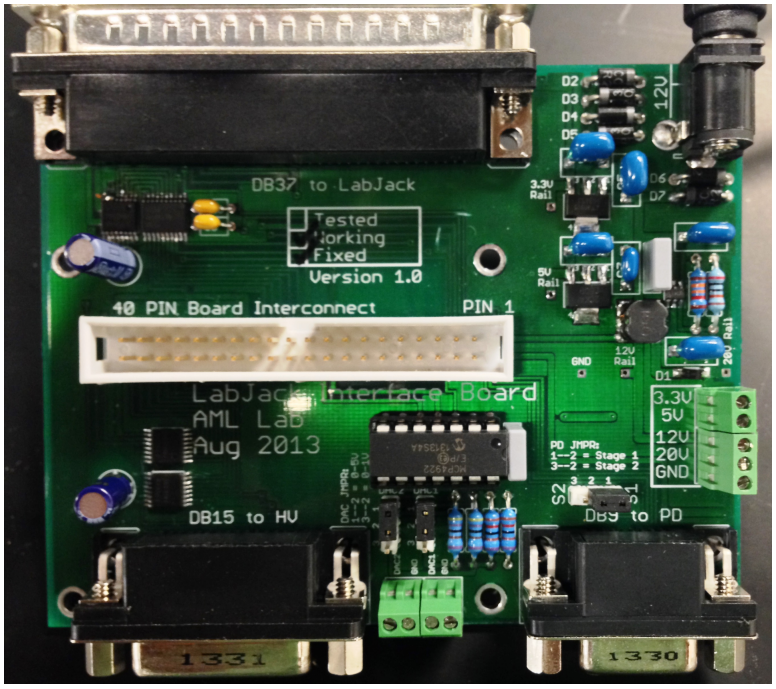


Figure F.15: Populated Interface Board circuit.

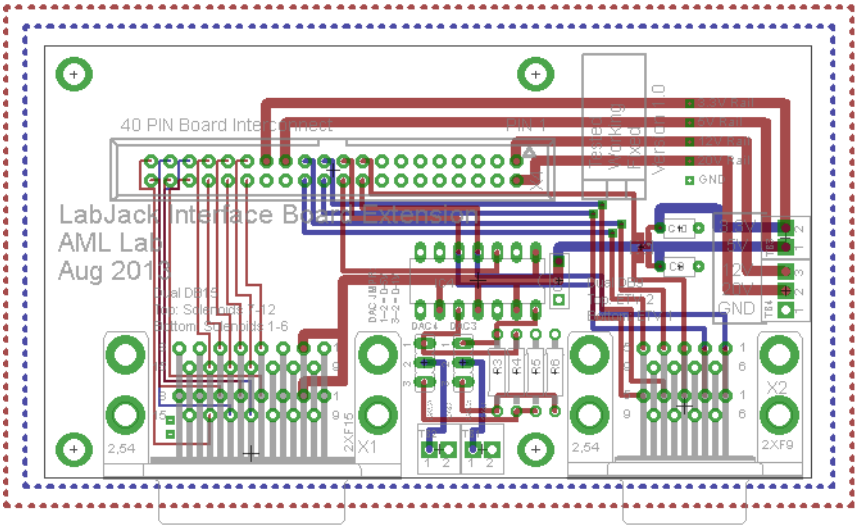


Figure F.16: Board layout for the Interface Board Extension circuit. The top layer of the board is in red, while the bottom layer is in blue. Silkscreen is presented in white. Image is actual size.

Table F.14: Interface Board Parts List

Description	Qty	Parts	DIGIKEY
Terminal Block 2 Position	3	TB1, TB2, TB3	A98333-ND
Terminal Block 3 Pos	1	TB4	A98334-ND
74HC595 Shift Register	3	IC6, IC7, IC8	MM74HC595MTCXCT-ND
CAP CER 0.1UF 25V 5% 1206	6	C13, C14, C15, C16, C17, C18	587-1147-1-ND
SUB-D	1	X1	609-4009-ND
SUB-D	1	X2	609-4011-ND
SUB-D	1	X3	182-837FE-ND
DC POWER JACK	1	J1	CP-202A-ND
JUMPER	3	DAC1, DAC2, JP1	609-3461-ND
0.1uF Ceramic Capacitor	2	C9, C10	BC1101CT-ND
FK22 0.47uF Ceramic Cap	1	C7	445-2649-ND
CONNECTOR	1	X4	
Resistor 1/4W 1.2K 207/10	2	R5, R6	1.2KADCT-ND
0.1uF 63V 5% Film Capacitor MKT370	2	C2, C8	3013PH-ND
SRN6045 Inductor	1	L1	SRN6045-100MCT-ND
Resistor 13.3K 1/4W 207/10	1	R2	13.3KXBK-ND
1.0A SCHOTTKY BARRIER RECTIFIER	6	D2, D3, D4, D5, D6, D7	SR202-TPCT-ND
Cap Pol 1uF 50V 20% 2.54mm	2	C11, C12	P5174-ND
Resistor 200K 1/4W 207/10	1	R1	200KXBK-ND
FK22 22uF Ceramic Cap	4	C3, C4, C5, C6	445-8480-ND
Resistor 1/4W 4.7k 207/10	2	R3, R4	4.7KADCT-ND
FK22 4.7uF Ceramic Cap	1	C1	445-8476-ND
5V @ 1A Linear Low Dropout Regulator	1	IC3	LM2940IMP-5.0/NOPBCT-ND
3.3V Linear Regulator	1	IC2	LT1963AEST-3.3#PBF-ND
1.1A Step-Up DC/DC Converter	1	IC1	LT3467IS6#TRMPBFCT-ND
12Bit SPI DAC MCP4902	1	IC4	MCP4922-E/P-ND
Schottky Diode, 30V @ 1.5A	1	D1	568-4127-1-ND
Single Bus Dual Buffer Gate	6	IC9, IC10, IC11, IC12, IC13, IC14	296-13496-1-ND
8-Pin Bi-Directional Logic-Level Converter	1	IC5	296-21527-1-ND

Table F.15: Interface Board Extension Parts List

Description	Qty	Parts	DIGIKEY
Terminal Block 2 Position	3	TB1, TB2, TB3	A98333-ND
Terminal Block 3 Pos	1	TB4	A98334-ND
DigiKey: 189-215FFE-ND	1	X1	189-215FFE-ND
JUMPER	2	DAC3, DAC4	609-3461-ND
0.1uF Ceramic Capacitor	2	C9, C10	BC1101CT-ND
CONNECTOR	1	X4	
Resistor 1/4W 1.2K 207/10	2	R5, R6	1.2KADCT-ND
0.1uF 63V 5% Film Capacitor MKT370	1	C8	3013PH-ND
DigiKey: 189-209FFE-ND	1	X2	189-209FFE-ND
Resistor 1/4W 4.7k 207/10	2	R3, R4	4.7KADCT-ND
12Bit SPI DAC MCP4902	1	IC4	MCP4922-E/P-ND
1-Pin Bi-Directional Logic-Level Converter	1	IC1	296-21664-1-ND

F.7.4 Usage

The Interface board requires a 12 V, 2 A supply. The board will then utilize this power to generate a 3.3 V, 5 V, and 20 V rails used to power the logic-level converters, Solenoid boards, and PD board.

Due to the nature of the board, a program specific to its use has been created. The individual inputs and outputs specific to each board it connects to are described within the respective section. The local inputs and outputs specific to the Interface board and extension are described in Tables F.16, F.17, F.18, and F.19.

Detailed pin assignments are presented in Tables F.20, F.21, F.22, F.23, F.24, and F.25.

Table F.16: Interface Board Inputs and Power Supply

Input	Part(s)	Description
Power	J_1	12 V at 2 A is supplied
LabJack	DB37	LabJack expansion port is utilized to connect to the interface board and provides all of the control
Photodiode Board	DB9	Connects to the Photodiode Board
Interface Board Extension	IDE	Connects to the Interface Board Extension

Table F.17: Interface Board Outputs

Output	Part(s)	Description
DAC_1	TB_2 Pin 1	Output of DAC 1, 0→5 V or 0→1 V depending upon jumper position
DAC_2	TB_1 Pin 1	Output of DAC 2, 0→5 V or 0→1 V depending upon jumper position
3.3V Rail	TB_3 Pin 2	3.3 V power rail
5V Rail	TB_3 Pin 1	5 V power rail
12V Rail	TB_4 Pin 3	12 V supplied power rail
20V Rail	TB_4 Pin 2	20 V power rail
GND	TB_4 Pin 1, TB_1 Pin 2, TB_2 Pin 2	Circuit ground
Fluidics Board	DB15	Connects to the Fluidics Board
Photodiode Board	DB9	Connects to the Photodiode Board
Interface Board Extension	IDE	Connects to the Interface Board Extension

Table F.18: Interface Board Extension Inputs

Input	Part(s)	Description
Interface Board	IDE	Connects to the Interface Board
ETV Power Supply Board A	DB9	Connects to the first ETV Power Supply control board
ETV Power Supply Board B	DB9	Connects to the second ETV Power Supply control board

Table F.19: Interface Board Extension Outputs

Output	Part(s)	Description
DAC_1	TB_2 Pin 1	Output of DAC 1, 0→5 V or 0→1 V depending upon jumper position
DAC_2	TB_1 Pin 1	Output of DAC 2, 0→5 V or 0→1 V depending upon jumper position
3.3V Rail	TB_3 Pin 2	3.3 V power rail
5V Rail	TB_3 Pin 1	5 V power rail
12V Rail	TB_4 Pin 3	12 V supplied power rail
20V Rail	TB_4 Pin 2	20 V power rail
GND	TB_4 Pin 1, TB_1 Pin 2, TB_2 Pin 2	Circuit ground
Solenoid Board A	DB15	Connects to first Solenoid Board
Solenoid Board B	DB15	Connects to second Solenoid Board
ETV Power Supply Board A	DB9	Connects to the first ETV Power Supply control board
ETV Power Supply Board B	DB9	Connects to the second ETV Power Supply control board

Table F.20: Interface Board DB37 Pin Assignments

Pin	LabJack	Interface Board	Use
1	GND	GND	
2	Pin 2		
3	FIO6	SCK	SPI Clock for IC4 MCP4922 and Extension Board (5 V)
4	FIO4	SHCP1	Shift register Clock Pin for IC8 HC595
5	FIO2	SER	Serial Data Input for IC6 HC595
6	FIO0	STCP	Storage register Clock Pin for IC6 and IC7 HC595
7	CIO1	SCK_PD	SPI Clock for Photodiode Board (3.3 V)
8	GND	GND	
9	VM-		
10	GND	GND	
11	DAC0	FDAC	DAC Control of HV Module in Fluidics Board
12	AIN13		
13	AIN11		
14	AIN9		
15	AIN7		
16	AIN5		
17	AIN3		
18	AIN1		
19	GND	GND	
20	Pin 20		
21	FIO7	SDI	Serial Data Input for IC4 MCP4922 and Extension Board (5 V)
22	FIO5	SER1	Serial Data Input for IC8 HC595
23	FIO3	STCP1	Storage register Clock Pin for IC8 HC595
24	FIO1	SHCP	Shift register Clock Pin for IC6 and IC7 HC595
25	CIO0	SDO	Slave Data Out for Photodiode Board (3.3 V) and Extension Board
26	CIO2	SDI_PD	Slave Data In for Photodiode Board (3.3 V)
27	VS		
28	VM+		
29	DAC1		
30	GND	GND	
31	AIN12		
32	AIN10		
33	AIN8		
34	AIN6		
35	AIN4		
36	AIN2		
37	AIN0	PD_Streaming	LabJack Streaming Mode to Monitor Stage 1 or Stage 2 of Photodiode Board (15 V)

Table F.21: Interface Board DB15 Pin Assignments

Pin	Interface Board	Fluidics Board	Use
1			
2	Relay 3	Relay 3	3.3 V MOSFET Driver Control (IC6 HC595)
3	Relay 1	Relay 1	3.3 V MOSFET Driver Control (IC6 HC595)
4	FDAC	V_{IN}	0→5 V DAC Control of HV Module in Fluidics Board
5			
6			
7			
8	GND	GND	
9	Relay 4	Relay 4	3.3 V MOSFET Driver Control (IC6 HC595)
10	Relay 2	Relay 2	3.3 V MOSFET Driver Control (IC6 HC595)
11	GND	GND	
12	V_{IN}	V_{PWR}	12 V Board Power
13			
14			
15			

Table F.22: Interface Board DB9 Pin Assignments

Pin	Interface Board	Photodiode Board	Use
1	20 V Rail	V_{IN}	20 V Board Power
2	PD ADC CS	ADC CS	3.3 V Chip Select for the ADC (IC8 HC595)
3			
4	S1 Out	Stage 1 Out	Stage 1 Amplifier Output (15 V) for Streaming Mode
5	S2 Out	Stage 2 Out	Stage 2 Amplifier Output (15 V) for Streaming Mode
6	SDI	SDI	Slave Data In (3.3 V)
7	SDO	SDO	Slave Data Out (3.3 V)
8	SCK	SCK	SPI Clock (3.3 V)
9	GND	GND	

Table F.23: Interface Board and Interface Board Extension 40-Pin IDE Pin Assignments

Pin	Interface Board	Interface Board Extension	Use
1	V_{IN}	V_{IN}	12 V Rail
2	20 V Rail	20 V Rail	20 V Rail
3	GND	GND	Circuit Ground
4	GND	GND	Circuit Ground
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17	SDO	SDO	ETV1 and ETV2 Slave Data Out (5 V)
18	SCK	SCK	Software Clock for MCP4922 DAC, ETV1, and ETV2 (5 V)
19	SDI	SDI	Slave Data In for MCP4922 DAC, ETV1, and ETV2 (5 V)
20	DAC2 CS	DAC2 CS	Chip Select for MCP4922 DAC (IC8 HC595)
21	ETV1 DAC CS	ETV1 DAC CS	Chip Select for ETV1 DAC (IC8 HC595)
22	ETV1 ADC CS	ETV1 ADC CS	Chip Select for ETV1 ADC (IC8 HC595)
23	ETV2 DAC CS	ETV2 DAC CS	Chip Select for ETV2 DAC (IC8 HC595)
24	ETV2 ADC CS	ETV2 ADC CS	Chip Select for ETV2 ADC (IC8 HC595)
25	5 V Rail	5 V Rail	5 V Rail
26	GND	GND	Circuit Ground
27	3.3 V Rail	3.3 V Rail	3.3 V Rail
28	GND	GND	Circuit Ground
29	Sol 11	Sol 11	MOSFET Driver Control (IC6 HC595)
30	Sol 12	Sol 12	MOSFET Driver Control (IC6 HC595)
31	Sol 9	Sol 9	MOSFET Driver Control (IC6 HC595)
32	Sol 10	Sol 10	MOSFET Driver Control (IC6 HC595)
33	Sol 7	Sol 7	MOSFET Driver Control (IC7 HC595)
34	Sol 8	Sol 8	MOSFET Driver Control (IC7 HC595)
35	Sol 5	Sol 5	MOSFET Driver Control (IC7 HC595)
36	Sol 6	Sol 6	MOSFET Driver Control (IC7 HC595)
37	Sol 3	Sol 3	MOSFET Driver Control (IC7 HC595)
38	Sol 4	Sol 4	MOSFET Driver Control (IC7 HC595)
39	Sol 1	Sol 1	MOSFET Driver Control (IC7 HC595)
40	Sol 2	Sol 2	MOSFET Driver Control (IC7 HC595)

Table F.24: Interface Board Extension X1 Pin Assignments

Pin	Extension Board	Solenoid Board(s)	Use
1			
2	5 V Rail	Power	5 V Board Power
3			
4	Sol 6	Sol 6	MOSFET Driver Control (IC7 HC595)
5	Sol 4	Sol 4	MOSFET Driver Control (IC7 HC595)
6	Sol 2	Sol 2	MOSFET Driver Control (IC7 HC595)
7			
8	GND	GND	Circuit Ground
9			
10			
11	GND	GND	Circuit Ground
12	Sol 5	Sol 5	MOSFET Driver Control (IC7 HC595)
13	Sol 3	Sol 3	MOSFET Driver Control (IC7 HC595)
14	Sol 1	Sol 1	MOSFET Driver Control (IC7 HC595)
15			
16			
17	5 V Rail	Power	5 V Board Power
18			
19	Sol 12	Sol 6	MOSFET Driver Control (IC6 HC595)
20	Sol 10	Sol 4	MOSFET Driver Control (IC6 HC595)
21	Sol 8	Sol 2	MOSFET Driver Control (IC7 HC595)
22			
23	GND	GND	Circuit Ground
24			
25			
26	GND	GND	Circuit Ground
27	Sol 11	Sol 5	MOSFET Driver Control (IC6 HC595)
28	Sol 9	Sol 3	MOSFET Driver Control (IC6 HC595)
29	Sol 7	Sol 1	MOSFET Driver Control (IC7 HC595)
30			

Table F.25: Interface Board Extension X2 Pin Assignments

Pin	Extension Board	ETV Board(s)	Use
1	SCK	SCK	SPI Clock (5 V)
2	SDI	SDI	Slave Data In (5 V)
3	SDO	SDO	Slave Data Out (5 V)
4	ETV1 DAC CS	DAC CS	Chip Select for DAC
5	ETV1 ADC CS	ADC CS	Chip Select for ADC
6	ETV1 VREF	VRef	5 V Board Reference Power
7	GND	GND	Circuit Ground
8			
9			
10	SCK	SCK	SPI Clock (5 V)
11	SDI	SDI	Slave Data In (5 V)
12	SDO	SDO	Slave Data Out (5 V)
13	ETV2 DAC CS	DAC CS	Chip Select for DAC
14	ETV2 ADC CS	ADC CS	Chip Select for ADC
15	ETV2 VREF	VRef	5 V Board Reference Power
16	GND	GND	Circuit Ground
17			
18			

F.8 ETV PID Heater Controller Board

The ETV Proportional-Integral-Derivative (PID) Heater Controller Board is controlled via a BoArduino. The board allows for the control of up to 20 Watts (W), 36 V, or 4 A (whichever comes first). In the FAES, the board controls 12 V to the heaters. The board works by having the BoArduino control a DAC which sends out a 0→4.094 V signal corresponding to the A sunk through the D44H8 transistor. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.17 and F.18 respectively.

F.8.1 Schematic

The schematic is separated into five sections: Power input, voltage regulation, thermocouple monitoring, BoArduino and monitoring LEDs, and the actual Heater Control.

The BoArduino is controlled over a USB-Serial connection from the included Mini-B Universal Serial Bus (USB) port. Once the program is uploaded to the microcontroller it will run independently. The LEDs are used to inform the user if the temperature is below (red), above (red and green) or at (green) the set temperature.

Terminal blocks or keyed pin headers may be used to form the connections to external components.

F.8.2 Board Layout

In general the power and heater control inputs/outputs are at the bottom of the board and the thermocouple connection is at the top of the board. The BoArduino is in the center of the board overtop the DAC and OpAmp to conserve board space. The resistors are paralleled and spaced apart sufficiently to allow for cooling and larger wattage. The large heatsink for the transistor is present to allow for sustained operation at high currents. Protection diodes on the power input protect from reverse polarity.

F.8.3 Populating

Populating the PID Heater Board is a simple matter. It is best to start with the surface mount Thermocouple-to-Digital converter (*MAX31855*) followed by the remaining surface

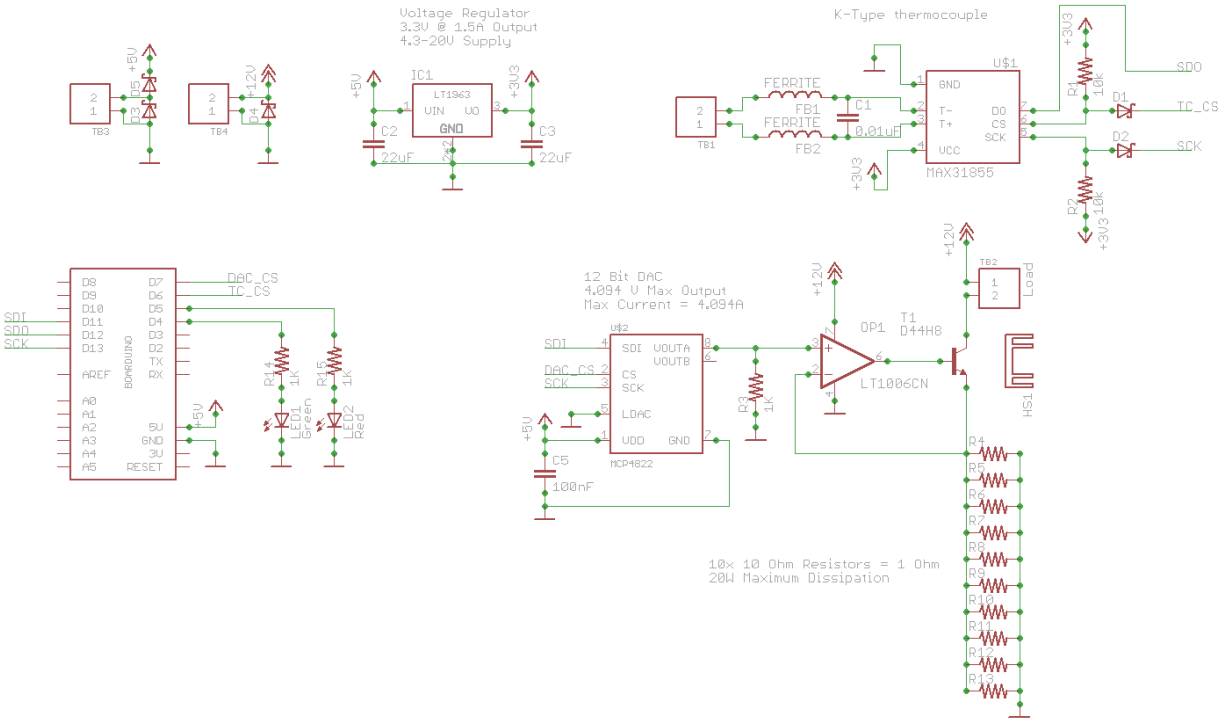


Figure F.17: Schematic diagram for the Arduino Controlled ETV PID heater controller circuit.

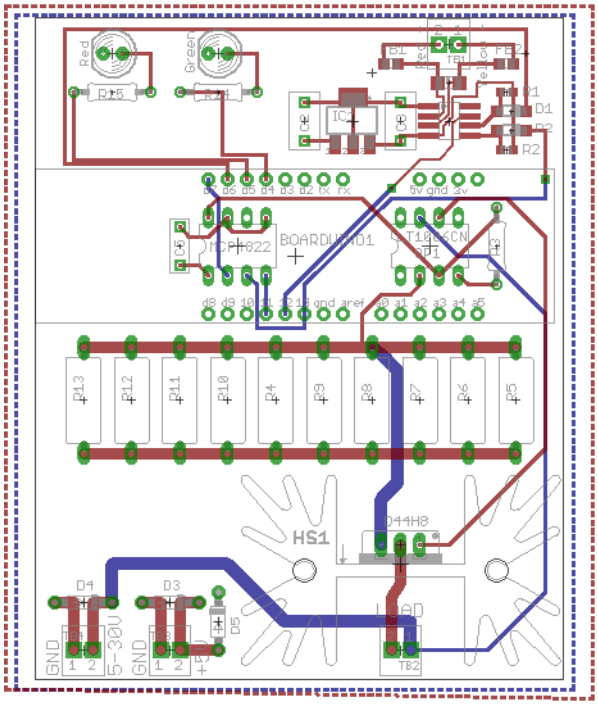


Figure F.18: Board layout for the Arduino Controlled ETV PID heater controller circuit. Image is actual size.

mount components and resistors. The remaining components can then be added to the board from shortest to tallest. Table F.26 contains the parts list.

Table F.26: PID Heater Board Parts List

Description	Qty	Parts	DIGIKEY
1.0A SCHOTTKY BARRIER RECTIFIER	2	D3, D5	1N5817DICT-ND
Terminal Block 2 Position	4	TB1, TB2, TB3, TB4	A98333-ND
Adafruit USB Boarduino	1	BOARDUINO1	
Diode: 60V 2A DO41	1	D4	SR206-TPCT-ND
Aavid Thermalloy Heatsink 2.6C/W	1	HS1	HS380-ND
3.3V Linear Regulator	1	IC1	LT1963AEST-3.3#PBF-ND
Schottky Diode, 30V @ 1.5A	2	D1, D2	568-4127-1-ND
Resistor 10 2W 1% 4.6x11.5mm	10	R4, R5, R6, R7, R8, R9, R10, R11, R12, R13	10ABCT-ND
CAP CER 0.1UF 25V 5% 1206	1	C1	587-1147-1-ND
0.1uF 63V 5% Film Capacitor MKT370	1	C5	3013PH-ND
Resistor 10K 1/10W 5% 0603	2	R1, R2	311-10KGRCT-ND
Resistor 1/4W 1K 207/10	3	R3, R14, R15	1KADCT-ND
FK22 22uF Ceramic Cap	2	C2, C3	445-8480-ND
NPN Transistor 60V 8A TO-220	1	T1	D44H8-ND
Ferrite Bead 300Ohm @ 100MHz	2	FB1, FB2	240-2391-1-ND
LED Green Indicator	1	LED1	754-1735-ND
LT1006CN GP (Single Supply) OpAmp	1	OP1	LT1006CN8#PBF-ND
Thermocouple-to-Digital Converter	1	U$1	MAX31855KASA+-ND
12Bit SPI DAC MCP4822	1	U$2	MCP4822-E/P-ND
LED Red Indicator	1	LED2	754-1275-ND

F.8.4 Usage

The PID Heater Board requires a 5 V, 500 mA, supply. In addition, the Heater required a 12 V, 4.096 A, supply. Otherwise the board operates stand-alone. External monitoring may be implemented through a USB Mini-B jack on the BoArduino itself. The inputs and outputs are as defined in Tables F.27 and F.28.

Table F.27: PID Heater Board Inputs

Input	Part(s)	Description
Mini-B USB Port	BoArduino	Used for monitoring and uploading new control algorithms
5 V Rail	TB3	5 V Rail
12 V Rail	TB4	5 V Rail
Thermocouple	TB1	K-Type Thermocouple

Table F.28: PID Heater Board Outputs

Output	Part	Description
Mini-B USB Port	BoArduino	Used for monitoring and uploading new control algorithms
Load	TB2	Used for connecting the load (Heaters)
Lights	LED1, LED2	Used to visually monitor temperature of thermocouple

F.9 Low Voltage Differential Signaling Boards

There are three boards which comprise each end-point: A buffer board to clean the incoming and outgoing signal; a recevier/driver board to convert the signal to/from Low Voltage Differential Signaling (LVDS); and a connector board used to connect the start and end points together. The three boards on each end of the signal are connected through the use of braided multi-core copper wire and pin-identified headers. The entire setup is powered by a 3.3 V supply for testing, however, the buffers may be powered up to 5 V depending upon the application. The schematic and board layouts were developed in CadSoft EAGLE v6 Freeware and are presented in Figures F.19 and F.20 respectively.

The parts list for the LVDS test circuits is presented in Table F.29.

An additional design for over-voltage protection was also put together but has not been tested. This design is primarily for the protection against ElectroStatic Discharge (ESD) on connections made; ensuring that the power rails are not artificially elevated; and to enable continuous 12 V short to any pin without causing damage. This is presented in Figure F.21.

Table F.29: LVDS Test Circuit Parts List

Description	Qty	Parts	DIGIKEY
Terminal Block 2 Position	2	TB1, TB2	A98333-ND
Terminal Block 3 Position	1	TB3	A98334-ND
JUMPER	5	JP1, JP2, JP3, JP4, JP5	609-3461-ND
0.1 uF Ceramic Capacitor	5	C1, C2, C4, C5, C7	BC1101CT-ND
IC Differential Line Driver	2	IC1, IC2	296-6919-1-ND
Terminated IC Differential Line Receiver	2	IC4, IC5	296-6896-1-ND
IC Bus Transceiver 8-Bit (Buffer)	1	IC7	296-4315-5-ND
10X2 Pin Header Socket	2	H1, H2	3M11932-ND

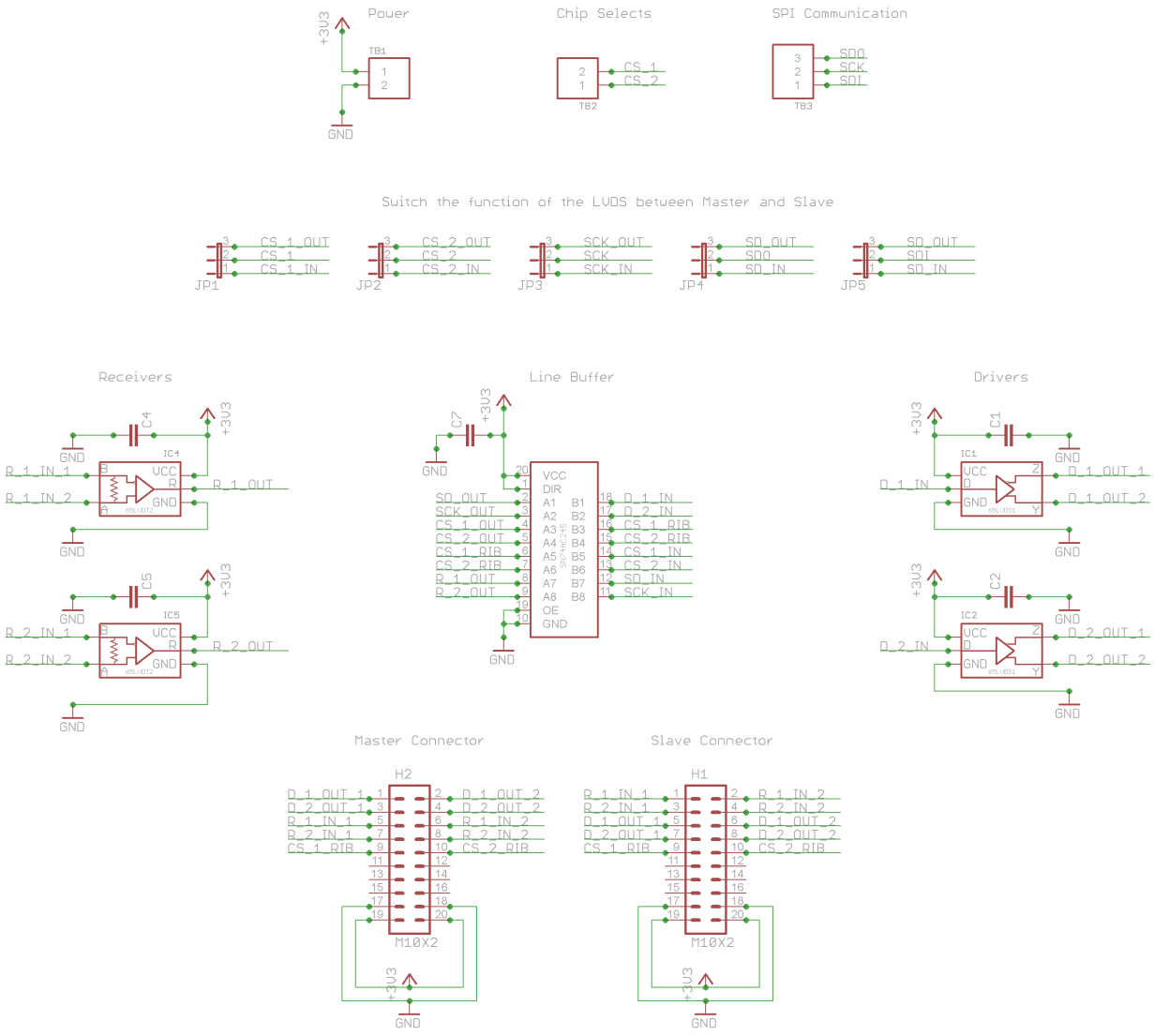


Figure F.19: LVDS Test Circuit Schematic.

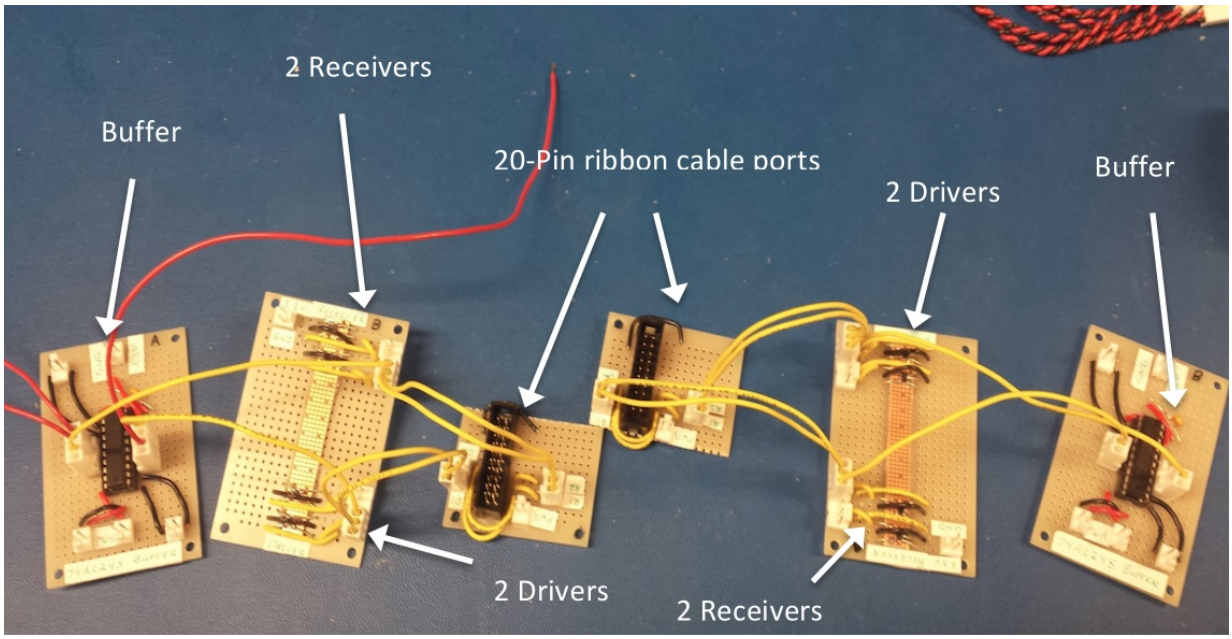


Figure F.20: LVDS Test Circuits fabricated on Protoboard. Buffer is a *74AC245*; Driver is a *65LVDS1*; Receiver is a *65LVDT2*. A 20-Pin ribbon cable connects the boards together.

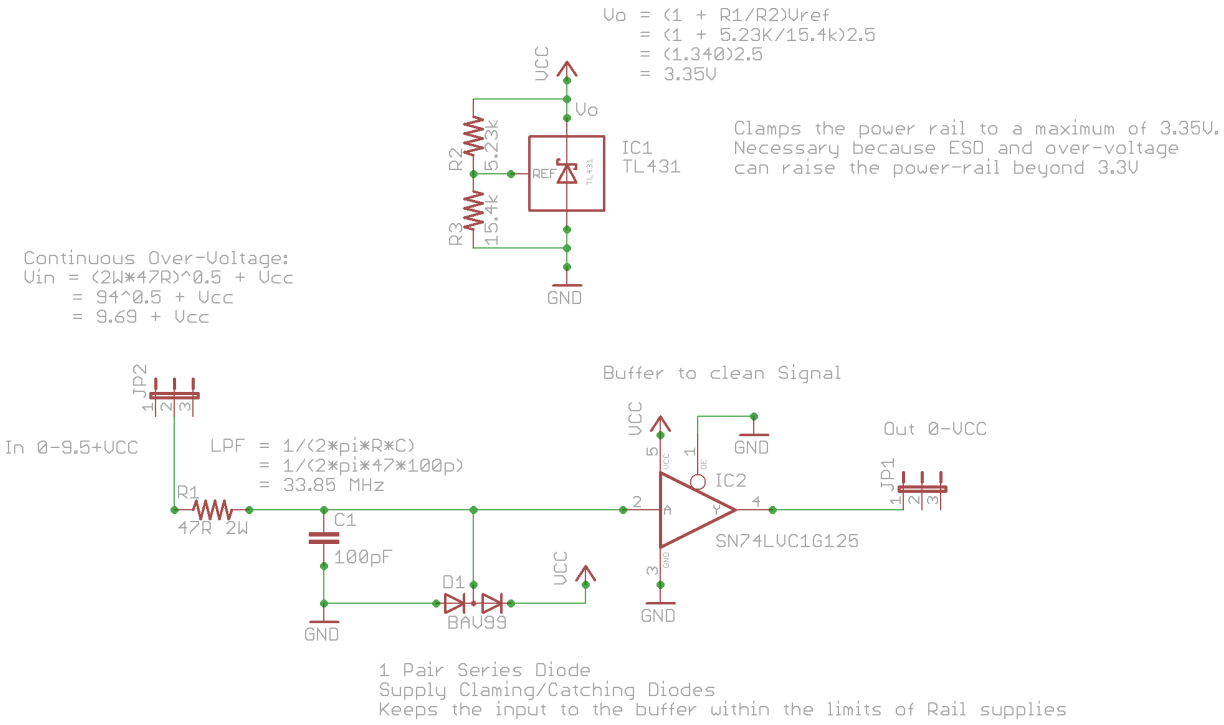


Figure F.21: Over-Voltage Protection Circuit before Buffer to alow for ESD on connections. This circuit allows for a continuous 12.5 V signal input without damaging any components. In addition a low pass filter of 33.85 MHz eliminates the introduction of high-frequency noise on the communication line without impeding the desired signal. A secondary power-regulator circuit clamps the 3.3 V rail at 3.3 V in case of ESD on the power connection.

Appendix G

Select Excerpts from Datasheets

G.1 MCP4922 DAC

G.1.1 SPI Command Registers for MCP4922

REGISTER 5-1: WRITE COMMAND REGISTER FOR MCP4922 (12-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
$\overline{A}$/B	BUF	$\overline{GA}$	$\overline{SHDN}$	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
bit 15															bit 0

REGISTER 5-2: WRITE COMMAND REGISTER FOR MCP4912 (10-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
$\overline{A}$/B	BUF	$\overline{GA}$	$\overline{SHDN}$	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	x	x
bit 15															bit 0

REGISTER 5-3: WRITE COMMAND REGISTER FOR MCP4902 (8-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
$\overline{A}$/B	BUF	$\overline{GA}$	$\overline{SHDN}$	D7	D6	D5	D4	D3	D2	D1	D0	x	x	x	x
bit 15															bit 0

Where:

bit 15 **$\overline{A}$/B:** DAC_A or DAC_B Selection bit
- 1 = Write to DAC_B
- 0 = Write to DAC_A

bit 14 **BUF:** V_{REF} Input Buffer Control bit
- 1 = Buffered
- 0 = Unbuffered

bit 13 **$\overline{GA}$:** Output Gain Selection bit
- 1 = 1x ($V_{OUT} = V_{REF} * D/4096$)
- 0 = 2x ($V_{OUT} = 2 * V_{REF} * D/4096$)

bit 12 **$\overline{SHDN}$:** Output Shutdown Control bit
- 1 = Active mode operation. VOUT is available.
- 0 = Shutdown the selected DAC channel. Analog output is not available at the channel that was shut down. VOUT pin is connected to 500 kΩ (typical).

bit 11-0 **D11:D0:** DAC Input Data bits. Bit x is ignored.

Legend			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	1 = bit is set	0 = bit is cleared	x = bit is unknown

G.1.2 SPI Electrical Characteristics for MCP4922

AC CHARACTERISTICS (SPI TIMING SPECIFICATIONS)

Electrical Specifications: Unless otherwise indicated, V_{DD}= 2.7V – 5.5V, T_A= -40 to +125°C. Typical values are at +25°C.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Schmitt Trigger High-Level Input Voltage (All digital input pins)	V_{IH}	0.7 V_{DD}	—	—	V	
Schmitt Trigger Low-Level Input Voltage (All digital input pins)	V_{IL}	—	—	0.2 V_{DD}	V	
Hysteresis of Schmitt Trigger Inputs	V_{HYS}	—	0.05 V_{DD}	—	V	
Input Leakage Current	$I_{LEAKAGE}$	-1	—	1	µA	$\overline{SHDN}$ = $\overline{LDAC}$ = $\overline{CS}$ = SDI = SCK + V_{REF} = V_{DD} or V_{SS}
Digital Pin Capacitance (All inputs/outputs)	C_{IN}, C_{OUT}	—	10	—	pF	V_{DD} = 5.0V, T_A = +25°C, f_{CLK} = 1 MHz **(Note 1)**
Clock Frequency	F_{CLK}	—	—	20	MHz	T_A = +25°C **(Note 1)**
Clock High Time	t_{HI}	15	—	—	ns	**Note 1**
Clock Low Time	t_{LO}	15	—	—	ns	**Note 1**
$\overline{CS}$ Fall to First Rising CLK Edge	t_{CSSR}	40	—	—	ns	Applies only when $\overline{CS}$ falls with CLK high. **(Note 1)**
Data Input Setup Time	t_{SU}	15	—	—	ns	**Note 1**
Data Input Hold Time	t_{HD}	10	—	—	ns	**Note 1**
SCK Rise to $\overline{CS}$ Rise Hold Time	t_{CHS}	15	—	—	ns	**Note 1**
$\overline{CS}$ High Time	t_{CSH}	15	—	—	ns	**Note 1**
$\overline{LDAC}$ Pulse Width	t_{LD}	100	—	—	ns	**Note 1**
$\overline{LDAC}$ Setup Time	t_{LS}	40	—	—	ns	**Note 1**
SCK Idle Time before $\overline{CS}$ Fall	t_{IDLE}	40	—	—	ns	**Note 1**

Note 1: This parameter is ensured by design and not 100% tested.

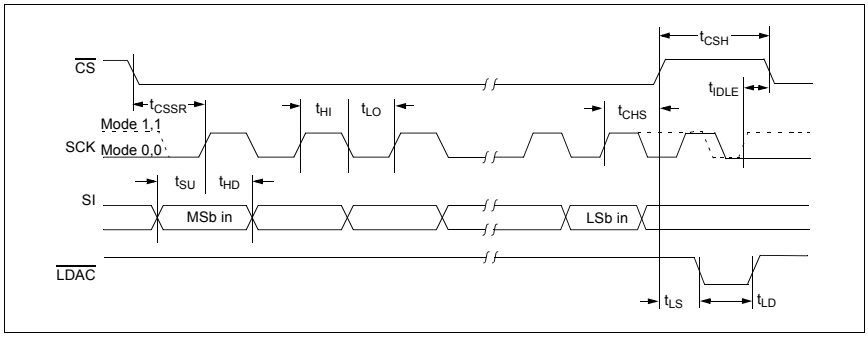


FIGURE 1-1: *SPI Input Timing Data.*

References

[1] Richard Jarrell. A Brief History of Atomic Emission Spectrochemical Analysis, 1666-1950. *Journal of Chemical Education*, 77(5):573–576, 2000. URL http://pubs.acs.org/doi/pdf/10.1021/ed077p573.1.

[2] JV Sokolnikova, IE Vasilyeva, and VI Menshikov. Determination of trace alkaline metals in quartz by flame atomic emission and atomic absorption spectrometry. *Atomic Spectroscopy Part B*, 58:387–391, 2003. URL http://www.sciencedirect.com/science/article/pii/S0584854702001532.

[3] N.J. Miller-Ihli. Atomic Absorption and Atomic Emission Spectrometry for the Determination of the Trace Element Content of Selected Fruits Consumed in the United States. *Journal of Food Composition and Analysis*, 9(4):301–311, December 1996. ISSN 08891575. doi: 10.1006/jfca.1996.9997. URL http://linkinghub.elsevier.com/retrieve/pii/S0889157596999974.

[4] Summer N. Hanna, Joseph Keene, Clifton P. Calloway, and Bradley T. Jones. Design of a Portable Electrothermal Vaporization Flame Atomic Emission Spectrometry Device for Field Analysis. *Instrumentation Science & Technology*, 39(4):345–356, July 2011. ISSN 1073-9149. doi: 10.1080/10739149.2011.585197. URL http://www.tandfonline.com/doi/abs/10.1080/10739149.2011.585197.

[5] Paula Michele Abentroth Klaic, Adriane Medeiros Nunes, Angelita Da Silveira Moreira, Claire Tondo Vendruscolo, and Anderson Schwingel Ribeiro. Determination of Na, K, Ca and Mg in xanthan gum: Sample treatment by acid digestion. *Carbohydrate Polymers*, 83(4):1895–1900, February 2011. ISSN 01448617. doi: 10.1016/j.carbpol.2010.10.059. URL http://linkinghub.elsevier.com/retrieve/pii/S0144861710008726.

[6] Wellington da Silva Lyra, Luciano Farias De Almeida, Francisco Antônio da Silva Cunha, Paulo Henrique Gonçalves Dias Diniz, Valdomiro Lacerda Martins, and

Mario Cesar Ugulino de Araujo. Determination of sodium and calcium in powder milk using digital image-based flame emission spectrometry. *Analytical Methods*, 6(4):1044, 2014. ISSN 1759-9660. doi: 10.1039/c3ay41005f. URL http://xlink.rsc.org/?DOI=c3ay41005f.

[7] Sumaira Khan, Tasneem G Kazi, Jameel A Baig, Nida F Kolachi, Hassan I Afridi, Abdul Q Shah, Ghulam a Kandhro, and Sham Kumar. Separation and preconcentration of trace amounts of aluminum ions in surface water samples using different analytical techniques. *Talanta*, 80(1):158–62, November 2009. ISSN 1873-3573. doi: 10.1016/j.talanta.2009.06.055. URL http://www.ncbi.nlm.nih.gov/pubmed/19782206.

[8] Bernhard Welz and Werner Luecke. Interference of aluminum chloride and nitrate on alkaline earth elements in an air-acetylene flame. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 48(14):1703–1711, December 1993. ISSN 05848547. doi: 10.1016/0584-8547(93)80157-P. URL http://linkinghub.elsevier.com/retrieve/pii/058485479380157P.

[9] Gustav N. Havre. The flame photometric determination of sodium, potassium and calcium in plant extracts with special reference to interference effects. *Analytica Chimica Acta*, 25(1):557–566, January 1961. ISSN 00032670. doi: 10.1016/S0003-2670(01)81614-7. URL http://linkinghub.elsevier.com/retrieve/pii/S0003267001816147.

[10] G. L. Donati, R. B. Wildman, and B. T. Jones. A new atomization cell for trace metal determinations by tungsten coil atomic spectrometry. *Analytica chimica acta*, 688(1):36–42, February 2011. ISSN 1873-4324. doi: 10.1016/j.aca.2010.12.036. URL http://www.ncbi.nlm.nih.gov/pubmed/21296202.

[11] EAG Zagatto, FJ Krug, F Bergamin, SS Jorgensen, and BF Reis. Merging zones in flow injection analysis: Part 2. Determination of calcium, magnesium and potassium in plant material by continuous flow injection atomic absorption. *Analytical Chemistry*, 104:279–284, 1979. URL http://www.sciencedirect.com/science/article/pii/S0003267001840095.

[12] JLFC Lima, AOSS Rangel, and MRS Souto. Simultaneous determination of potassium and sodium in vegetables by flame emission spectrometry using a flow-injection system with two dialysis units. *Analytical Sciences*, 12(February):81–85, 1996. URL http://cat.inist.fr/?aModele=afficheN&cpsidt=3029174.

[13] Jaroon Junsomboon and Jaroon Jakmunee. Determination of potassium, sodium, and total alkalies in portland cement, fly ash, admixtures, and water of concrete by a simple flow injection flame photometric system. *Journal of Automated Methods & Management in Chemistry*, 2011:742656, January 2011. ISSN 1464-5068. doi: 10.1155/2011/742656. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3124857&tool=pmcentrez&rendertype=abstract.

[14] Silvia MV Fernandes and Antonio OSS Rangel. Flow injection determination of sodium, potassium, calcium, and magnesium in beer by flame emission and atomic absorption spectrometry. *Journal of Agricultural and Food Chemistry*, 45(4):1269–1272, 1997. URL http://pubs.acs.org/doi/abs/10.1021/jf9604347.

[15] Zhaolun Fang, Joel M. Harris, Jaromir Ruzicka, and Elo H. Hansen. Simultaneous flame photometric determination of lithium, sodium, potassium, and calcium by flow injection analysis with gradient scanning standard addition. *Analytical Chemistry*, 57(7):1457–1461, June 1985. ISSN 0003-2700. doi: 10.1021/ac00284a062. URL http://pubs.acs.org/doi/abs/10.1021/ac00284a062.

[16] G N Doku and V P Gadzekpo. Simultaneous determination of lithium, sodium and potassium in blood serum by flame photometric flow-injection analysis. *Talanta*, 43 (5):735–9, May 1996. ISSN 0039-9140. doi: 10.1016/0039-9140(95)01808-5. URL http://www.ncbi.nlm.nih.gov/pubmed/18966541.

[17] M. Inês G. S. Almeida, Marcela A. Segundo, José L. F. C. Lima, and António O. S. S. Rangel. Interfacing multisyringe flow injection analysis to flame atomic emission spectrometry: an intelligent system for automatic sample dilution and determination of potassium. *Journal of Analytical Atomic Spectrometry*, 24(3):340, 2009. ISSN 0267-9477. doi: 10.1039/b813150c. URL http://xlink.rsc.org/?DOI=b813150c.

[18] Sherwood Scientific Ltd. A Guide to Single Channel Flame Photometer Analysis. URL http://www.sherwood-scientific.com/flame/flameanalysis.html.

[19] B J Bolann, R Rahil-Khazen, H Henriksen, R Isrenn, and R J Ulvik. Evaluation of methods for trace-element determination with emphasis on their usability in the clinical routine laboratory. *Scandinavian journal of clinical and laboratory investigation*, 67(4):353–66, January 2007. ISSN 0036-5513. doi: 10.1080/00365510601095281. URL http://informahealthcare.com.proxy.lib.uwaterloo.ca/doi/abs/10.1080/00365510601095281.

[20] Robert C Hawkins. Laboratory turnaround time. *The Clinical biochemist. Reviews / Australian Association of Clinical Biochemists*, 28(4):179–94, November 2007. ISSN 0159-8090. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2282400&tool=pmcentrez&rendertype=abstract.

[21] United States Environmental Protection Agency. *Potential Nano-Enabled Environmenatal Applications for Radionuclides*. 2009. URL http://www.epa.gov/radiation/cleanup/nanotechnology.html.

[22] S J Lee and S Y Lee. Micro total analysis system (micro-TAS) in biotechnology. *Applied microbiology and biotechnology*, 64(3):289–99, April 2004. ISSN 0175-7598. doi: 10.1007/s00253-003-1515-0. URL http://www.ncbi.nlm.nih.gov/pubmed/14714150.

[23] Geert Jannes and Daniel De Vos. A review of current and future molecular diagnostic tests for use in the microbiology laboratory. *Methods in molecular biology (Clifton, N.J.)*, 345:1–21, January 2006. ISSN 1064-3745. doi: 10.1385/1-59745-143-6:1. URL http://www.ncbi.nlm.nih.gov/pubmed/16957343.

[24] Arjan Floris, Steven Staal, Stefan Lenk, Erik Staijen, Dietrich Kohlheyer, Jan Eijkel, and Albert van den Berg. A prefilled, ready-to-use electrophoresis based lab-on-a-chip device for monitoring lithium in blood. *Lab on a chip*, 10(14):1799–1806, 2010. ISSN 1473-0197. doi: 10.1039/c005361a.

[25] E. Hywel Evans, Matthew Horstwood, Jorge Pisonero, and Clare M. M. Smith. Atomic spectrometry update: review of advances in atomic spectrometry and related techniques. *Journal of Analytical Atomic Spectrometry*, 28(6):779, 2013. ISSN 0267-9477. doi: 10.1039/c3ja90029k. URL http://xlink.rsc.org/?DOI=c3ja90029k.

[26] David J. DJ Butcher. Review: Recent Advances in Optical Analytical Atomic Spectrometry. *Applied Spectroscopy Reviews*, 48(4):261–328, May 2013. ISSN 0570-4928. doi: 10.1080/05704928.2012.717570. URL http://www.tandfonline.com/doi/abs/10.1080/05704928.2012.717570.

[27] Summer N. Hanna and Bradley T. Jones. A Review of Tungsten Coil Electrothermal Vaporization as a Sample Introduction Technique in Atomic Spectrometry. *Applied Spectroscopy Reviews*, 46(8):624–635, November 2011. ISSN 0570-4928. doi: 10.1080/05704928.2011.582659. URL http://www.tandfonline.com/doi/abs/10.1080/05704928.2011.582659.

[28] MA Karabegov. Flame photometers. Basic parameters and metrological support. *Measurement Techniques*, 54(6):735–742, 2011. URL http://www.springerlink.com/index/M646U74501250033.pdf.

[29] BernzOmatic. MG9 - 14.1 oz. MAP-Pro Hand Torch Cylinder, [Online]. Available: http://www.bernzomatic.com/item.html?id=48 [2015, January 19]. BernzOmatic A Worthington Industries Company, Ohio USA., 2012.

[30] Malcolm S. Cresser. *Flame Spectrometry in Environmental Chemical Analysis: A Practical Guide*. Royal Society of Chemistry, 1 edition, 1994. ISBN 978-0851867342.

[31] Gary D. Christian and Fredric J. Feldman. A Comparison Study of Detection Limits Using Flame-Emission Spectroscopy with the Nitrous OxideAcetylene Flame and Atomic-Absorption Spectroscopy. *Applied Spectroscopy*, 25(6):660–663, November 1971. ISSN 00037028. doi: 10.1366/000370271779951093. URL http://openurl.ingenta.com/content/xref?genre=article&issn=0003-7028&volume=25&issue=6&spage=660.

[32] Robert D Cowan. *The Theory of Atomic Structure and Spectra*. Univ of California Press, 3 edition, 1981.

[33] Worthington Cylinder Corporation. MAP-Pro Premium Hand Torch Fuel MSDS. Technical Report 909050; SDS WC001, Worthington Cylinder Corporation, Chilton, Wisconsin, 2014. URL http://www.worthingtonindustries.com/getmedia/ff77868e-b526-44ae-9064-e51aaa5ea02f/wc001-map-pro-premium-hand-torch-fuel.

[34] S. D. Gallimore. *A Study of Plasma Ignition Enhancement for Aeroramp Injectors in Supersonic Combustion Applications*. PhD thesis, 2001.

[35] A. Kramida, Yu. Ralchenko, J. Reader, and NIST ASD Team. NIST Atomic Spectra Database (ver. 5.2), [Online]. Available: http://physics.nist.gov/asd [2013, January 24]. National Institute of Standards and Technology, Gaithersburg, MD., 2012.

[36] George E.P. Box, J. Stuart Hunter, and William G. Hunter. *Statistics for Experimenters: Design, Innovation, and Discovery*. John Wiley & Sons, 2 edition, 2005. ISBN 978-0471718130.

[37] Steven C. Chapra. *Applied Numerical Methods with MATLAB for Engineers and Scientists*. McGraw-Hill, 2 edition, 2008. ISBN 978-0073132907.

[38] P. E. Halstead and A. E. Moore. The thermal dissociation of calcium hydroxide. *Journal of the Chemical Society (Resumed)*, 32(8):3873, 1957. ISSN 0368-1769. doi: 10.1039/jr9570003873. URL http://xlink.rsc.org/?DOI=jr9570003873.

[39] Jessica Garber Morales, Gregory P Holmes-Hampton, Ren Miao, Yisong Guo, Eckard Munck, and Paul A Lindahl. Biophysical characterization of iron in mitochondria isolated from respiring and fermenting yeast. *Biochemistry*, 49(26):5436–44, July 2010. ISSN 1520-4995. doi: 10.1021/bi100558z. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2898134&tool=pmcentrez&rendertype=abstract.

[40] Jennifer A. Rust, George L. Donati, Mirela T. Afonso, Joaquim A. Nóbrega, and Bradley T. Jones. An overview of electrothermal excitation sources for atomic emission spectrometry. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 64(3): 191–198, March 2009. ISSN 05848547. doi: 10.1016/j.sab.2009.02.003. URL http://linkinghub.elsevier.com/retrieve/pii/S0584854709000330.

[41] Sherwood Scientific Ltd. Model 360 Flame Photometer, 2014. URL http://www.sherwood-scientific.com/flame/360.html.

[42] BernzOmatic. Bernzomatic Product Catalogue, [Online]. Available: http://www.bernzomatic.sk/navody/katalog.pdf [2015, August 12]. BernzOmatic A Worthington Industries Company, Ohio USA., 2005.

[43] Henryk Matusiewicz. Electrothermal vaporisation sample introduction into plasma sources for analytical emission spectrometry. *Advances in Atomic Spectroscopy*, 2: 63–138, 1995.

[44] Mariusz Ślachcinski. Recent Achievements in Sample Introduction Systems for Use in Chemical Vapor Generation Plasma Optical Emission and Mass Spectrometry: From Macro- to Microanalytics. *Applied Spectroscopy Reviews*, 49(4):271–321, May 2014. ISSN 0570-4928. doi: 10.1080/05704928.2013.823547. URL http://www.tandfonline.com/doi/abs/10.1080/05704928.2013.823547.

[45] M. A. Belarra, M. Resano, F. Vanhaecke, and L. Moens. Direct solid sampling with electrothermal vaporization/atomization: What for and how? *TrAC - Trends in Analytical Chemistry*, 21(12):828–839, 2002. ISSN 01659936. doi: 10.1016/S0165-9936(02)01206-2.

[46] Peng Wu, Xiaodong Wen, Liang He, Yihua He, Minzhu Chen, and Xiandeng Hou. Evaluation of tungsten coil electrothermal vaporization-Ar/H2 flame atomic fluorescence spectrometry for determination of eight traditional hydride-forming elements and cadmium without chemical vapor generation. *Talanta*, 74(4):505–11, January 2008. ISSN 1873-3573. doi: 10.1016/j.talanta.2007.06.013. URL http://www.ncbi.nlm.nih.gov/pubmed/18371668.

[47] Yen-Jia Tseng, Yu-Duan Tsai, and Shiuh-Jen Jiang. Electrothermal vaporization dynamic reaction cell inductively coupled plasma mass spectrometry for the determination of Fe, Co, Ni, Cu, and Zn in biological samples. *Analytical and bioanalytical chemistry*, 387(8):2849–2855, 2007. ISSN 1618-2642. doi: 10.1007/s00216-007-1143-0.

[48] Kenneth W. Jackson and Guoru Chen. Atomic Absorption, Atomic Emission, and Flame Emission Spectrometry. *Analytical Chemistry*, 66(12):252R–279R, June 1994. ISSN 0003-2700. URL http://www.ncbi.nlm.nih.gov/pubmed/18967736.

[49] Masami Suzuki, Kiyohisa. Ohta, and Tatsuya. Yamakita. Atomic emission spectrometry of barium with a metal electrothermal atomizer. *Analytical Chemistry*, 53(12):1796–1798, 1981. ISSN 0003-2700. doi: 10.1021/ac00235a018. URL http://pubs.acs.org/doi/abs/10.1021/ac00235a018.

[50] Peter Barth and Viliam Krivan. Electrothermal vaporization inductively coupled plasma atomic emission spectrometric technique using a tungsten coil furnace and slurry sampling. *Journal of Analytical Atomic Spectrometry*, 9(7):773, 1994. ISSN 0267-9477. doi: 10.1039/ja9940900773.

[51] Tetsuya Sato, Misa Kato, Katsunori Kimijima, Masaki Okuyama, and Kiyoshi Chiba. Use of a coil atomizer in metastable transfer emission spectroscopy. *Spectrochimica acta, Part B: Atomic spectroscopy*, 55(4):383–387, 2000. ISSN 05848547. doi: 10.1016/S0584-8547(00)00151-8.

[52] Xiandeng Hou, Keith E. Levine, Arthur Salido, Bradley T. Jones, Muhsin Ezer, Seth Elwood, and Josef B. Simeonsson. Tungsten Coil Devices in Atomic Spectrometry: Absorption, Fluorescence, and Emission. *Analytical sciences : the international journal of the Japan Society for Analytical Chemistry*, 17(1):175–80, January 2001. ISSN 0910-6340. URL http://www.ncbi.nlm.nih.gov/pubmed/11993659.

[53] Xiandeng Hou and Bradley T. Jones. Tungsten devices in analytical atomic spectrometry. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 57(4):659–688, April 2002.

ISSN 05848547. doi: 10.1016/S0584-8547(02)00014-9. URL http://linkinghub.elsevier.com/retrieve/pii/S0584854702000149.

[54] Jennifer A. Rust, Joaquim A. Nóbrega, Clifton P. Calloway, and Bradley T. Jones. Advances with tungsten coil atomizers: Continuum source atomic absorption and emission spectrometry. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 60(5): 589–598, June 2005. ISSN 05848547. doi: 10.1016/j.sab.2005.02.006. URL http://linkinghub.elsevier.com/retrieve/pii/S0584854705000212.

[55] Jennifer a. Rust, Joaquim a. Nóbrega, Clifton P. Calloway, and Bradley T. Jones. Tungsten coil atomic emission spectrometry. *Spectrochimica Acta - Part B Atomic Spectroscopy*, 61(2):225–229, 2006. ISSN 05848547. doi: 10.1016/j.sab.2005.12.009.

[56] David E Nixon, Velmer a Fassel, and Richard N Kniseley. Inductively coupled plasma-optical emission analytical spectroscopy. Tantalum filament vaporization of microliter samples. *Anal. Chem*, 46(2):210–213, 1974. ISSN 00032700. doi: 10.1021/ac60338a018.

[57] E. Hywel Evans, Joseph A. Caruso, and R. Duane Satzger. Evaluation of a tantalum-tip electrothermal vaporization sample introduction device for microwave-induced plasma mass spectrometry and atomic emission spectrometry. *Appl. Spectrosc.*, 45(9):1478–1484, Sep 1991. URL http://as.osa.org/abstract.cfm?URI=as-45-9-1478.

[58] Richard F. J. Dams, Jan Goossens, and Luc Moens. Spectral and Non-Spectral Interferences in Inductively Coupled Plasma Mass-Spectrometry. *Mikrochimica Acta*, 119(3-4):277–286, September 1995. ISSN 0026-3672. doi: 10.1007/BF01244007. URL http://link.springer.com/10.1007/BF01244007.

[59] Marco Grotti, Cinzia Gnecco, and Fabio Bonfiglioli. Multivariate quantification of spectroscopic interferences caused by sodium, calcium, chlorine and sulfur in inductively coupled plasma mass spectrometry. *Journal of Analytical Atomic Spectrometry*, 14(8):1171–1175, 1999. ISSN 02679477. doi: 10.1039/a902625h. URL http://xlink.rsc.org/?DOI=a902625h.

[60] Jiyan Gu, Silvana R. Oliveira, George L. Donati, José Anchieta Gomes Neto, and Bradley T. Jones. Rugged, portable tungsten coil atomic emission spectrometer. *Analytical Chemistry*, 83(7):2526–2531, 2011. ISSN 00032700. doi: 10.1021/ac1027897.

[61] Summer N. Hanna, Clifton P. Calloway, Jason D. Sanders, Ronald A. Neslon, Jamaal Cox, and Bradley T. Jones. Design of a compact, aluminum, tungsten-coil electrothermal vaporization device for inductively coupled plasma-optical emission spectrometry. *Microchemical Journal*, 99(2):165–169, November 2011. ISSN 0026265X. doi: 10.1016/j.microc.2011.04.009. URL http://linkinghub.elsevier.com/retrieve/pii/S0026265X11000750.

[62] Raquel Sánchez, José Luis Todolí, Charles-Philippe Lienemann, and Jean-Michel Mermet. Determination of trace elements in petroleum products by inductively coupled plasma techniques: A critical review. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 88:104–126, October 2013. ISSN 05848547. doi: 10.1016/j.sab.2013.06.005. URL http://linkinghub.elsevier.com/retrieve/pii/S0584854713001602.

[63] SN Hanna. *Tungsten Coil Electrothermal Vaporization for Atomic Spectroscopy*. PhD thesis, Wake Forest University Graduate School of Arts and Sciences, 2011. URL http://wakespace.lib.wfu.edu/jspui/handle/10339/33451.

[64] Tianchi Ma, Victoria Northrup, Andrew O. Fung, D. Moira Glerum, and Christopher J. Backhouse. Polymeric rapid prototyping for inexpensive and portable medical diagnostics. 8412(519):84120B–84120B–8, October 2012. doi: 10.1117/12.2001470. URL http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1387270.

[65] Tianchi Ma. An Integrated Polymer Based Polymerase Chain Reaction And Capillary Electrophoresis System For Genetic Diagnosis. Master's thesis, University of Waterloo, 2014.

[66] David L. Jones. uSupply, 2012. URL http://www.eevblog.com/projects/usupply/.

[67] Wesley Chun. *Core Python Applications Programming*. Prentice Hall, 3 edition, 2012. ISBN 978-0132678209.

[68] Clayton R. Paul. *Introduction to Electromagnetic Compatibility*. Wiley-Interscience, 2 edition, 2006. ISBN 978-0471755005.

[69] Allison Christel Elizabeth Bidulock. Scalable, Modular, Integrated Genetic Analysis Systems. Master's thesis, University of Alberta, 2011.

[70] Chris Sterzik. Interfacing Different Logic With LVDS Receivers. Technical Report September, Texas Instruments, 2001.

[71] Chris Sterzik. Extending SPI and McBSP With Differential Interface Products. Technical Report May, Texas Instruments, 2003.

[72] LVDS Group - National Semiconductor. *LVDS Owner's Manual: Low-Voltage Differential Signaling*. National Semiconductor, 3 edition, 2004.

[73] Texas Instruments. SN54AC245, SN74AC245 Octal Bus Transceivers with 3-State Outputs. Technical Report SCAS461F, Texas Instruments, Dallas, Texas, 2003. URL http://www.ti.com/lit/ds/symlink/sn74ac245.pdf.

[74] Texas Instruments. SN65LVDxx High-Speed Differential Line Drivers and Receivers. Technical Report SLLS373L, Texas Instruments, Dallas, Texas, 2014. URL http://www.ti.com/lit/ds/symlink/sn65lvds1.pdf.

[75] LabJack Corporation. U6 User's Guide, 2012. URL http://labjack.com/support/u6/users-guide.

[76] Jerald Graeme. *Photodiode Amplifiers: OP AMP Solutions*. McGraw-Hill Professional, 1 edition, 1995. ISBN 978-0070242470.

[77] Aosong (Guangzhou) Electronics Co., Ltd. Digital-output relative humidity & temperature sensor/module. Technical Report AM2303, Aosong (Guangzhou) Electronics Co., Ltd, China. URL http://www.adafruit.com/datasheets/DHT22.pdf.

[78] Freescale Semiconductor. Miniature I2C Digital Barometer. Technical Report MPL115A2, Freescale Semiconductor, Austin, Texas, 2012. URL http://www.adafruit.com/datasheets/MPL115A2.pdf.

[79] Microchip Technology Inc. MCP4902/4912/4922 8/10/12-Bit Dual Voltage Output Digital-to-Analog Converter with SPI Interface. Technical Report DS22250A, Microchip Technology Inc., Chandler, Arizona, 2010. URL http://ww1.microchip.com/downloads/en/DeviceDoc/22250A.pdf.

[80] Maxim Integrated. MAX6225/MAX6241/MAX6250 Low-Noise, Precision, +2.5V/+4.096V/+5V Voltage References. Technical Report 19-1139, Maxim Integrated, San Jose, California, 2014. URL http://datasheets.maximintegrated.com/en/ds/MAX6225-MAX6250.pdf.

[81] Limor Fried. USB Boarduino Assembly, 2013. URL https://learn.adafruit.com/boarduino-kits/usb-boarduino-assembly.

[82] Ocean Optics Inc. New External Triggering Options Instructions for Spectrometer with Firmware Version 3.0 and Above. Technical report, 2014. URL http://oceanoptics.com/support/technical-documents/external-triggering-options_firmware3-0andabove/.

[83] Ocean Optics Inc. External Trigering Options Instructions. Technical report, 2014. URL http://oceanoptics.com/support/technical-documents/external-triggering-options/.

[84] George L. Donati, Jiyan Gu, Joaquim a. Nobrega, Clifton P. Calloway, Jr, and Bradley T. Jones. Simultaneous determination of the Lanthanides by tungsten coil atomic emission spectrometry. *Journal of Analytical Atomic Spectrometry*, 23(3):361, 2008. ISSN 0267-9477. doi: 10.1039/b710600a. URL http://xlink.rsc.org/?DOI=b710600a.

[85] Ocean Optics Inc. USB4000 Data Sheet. Technical Report 211-00000-000-05-201210, Ocean Optics Inc., Dunedin, Florida, 2012. URL http://oceanoptics.com/wp-content/uploads/USB4000-OEM-Data-Sheet.pdf.

[86] Sherif Awad, Simon P. Allison, and Dileep N. Lobo. The history of 0.9% saline. *Clinical Nutrition*, 27(2):179–188, 2008. ISSN 02615614. doi: 10.1016/j.clnu.2008.01.008.

[87] SR Crouch and JD Ingle. *Spectrochemical analysis*. Prentice-Hall, 1988. ISBN 0-13-826876-2.

[88] G.R. Kornblum and L. De Galan. Ionization interference in the acetylene-nitrous oxide flame, 1973. ISSN 05848547.

[89] Michael Thompson and Michael H. Ramsey. Matrix effects due to calcium in inductively coupled plasma atomic-emission spectrometry: their nature, source and remedy. *The Analyst*, 110(12):1413, 1985. ISSN 0003-2654. doi: 10.1039/an9851001413.

[90] Pickett and S Koirtyohann. Emission flame photometry-a new look at an old method. *Analytical Chemistry*, 4(14):28–42, 1969. URL http://pubs.acs.org/doi/abs/10.1021/ac50159a724.

[91] Gary D. Christian and Fredric J. Feldman. A Comparison Study of Detection Limits Using Flame-Emission Spectroscopy with the Nitrous OxideAcetylene Flame and Atomic-Absorption Spectroscopy. *Applied Spectroscopy*, 25(6):660–663, November 1971. ISSN 00037028. doi: 10.1366/

000370271779951093. URL http://openurl.ingenta.com/content/xref?genre=article&issn=0003-7028&volume=25&issue=6&spage=660.

[92] P Boumans and FJ De Boer. Studies of flame and plasma torch emission for simultaneous multi-element analysisI: Preliminary investigations. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 27B:391–414, 1972. URL http://www.sciencedirect.com/science/article/pii/0584854772800387.

[93] J. Quiñonero, C. Mongay, and M. de la Guardia. Determination of aluminum at the parts per billion level by solvent extraction and flame atomic emission spectrometry. *Microchemical Journal*, 43(3):213–221, June 1991. ISSN 0026265X. doi: 10.1016/S0026-265X(10)80008-6. URL http://linkinghub.elsevier.com/retrieve/pii/S0026265X10800086.

[94] J Posta, A Gáspár, R Tóth, and L Ombódi. Cr(III) and Cr(VI) on-line preconcentration and determination with high performance flow flame emission spectrometry in natural samples. *Analytical and Bioanalytical Chemistry*, 355(5-6):719–20, June 1996. ISSN 1618-2650. doi: 10.1007/s0021663550719. URL http://www.ncbi.nlm.nih.gov/pubmed/15045352.

[95] Douglas A. Skoog, F. James Holler, and Stanly R. Crouch. *Principles of instrumental analysis*. Thomson Brooks/Cole, 6 edition, 2007. ISBN 978-0-495-01201-6.

[96] I Rubeška and B Moldan. The mechanisms of interference effects and their elimination in the determination of alkaline earth metals by flame photometry. *Analytical Chemistry*, 37:421–428, 1967. URL http://www.sciencedirect.com/science/article/pii/S0003267001807029.

[97] TC Rains, HE Zittel, and M Ferguson. Elimination of anionic interferences in the flame spectrophotometric determination of calcium: Use of glycerol as a releasing agent. *Talanta*, 10:367–374, 1963. URL http://www.sciencedirect.com/science/article/pii/0039914063800338.

[98] Pawel Pohl, Helena Stecka, Iwona Sergiel, and Piotr Jamroz. Different Aspects of the Elemental Analysis of Honey by Flame Atomic Absorption and Emission Spectrometry: A Review. *Food Analytical Methods*, 5(4):737–751, September 2011. ISSN 1936-9751. doi: 10.1007/s12161-011-9309-y. URL http://link.springer.com/10.1007/s12161-011-9309-y.

[99] JA Dean and JH Lady. Application of Organic Solvent Extraction to Flame Specrophotometry. *Analytical Chemistry*, 27(10):1533–1536, 1955. URL http://pubs.acs.org/doi/abs/10.1021/ac60106a009.

[100] Min-Jane Chen, Ya-Ting Hsieh, Yih-Ming Weng, and Robin Y.-Y. Chiou. Flame photometric determination of salinity in processed foods. *Food Chemistry*, 91(4): 765–770, August 2005. ISSN 03088146. doi: 10.1016/j.foodchem.2004.10.002. URL http://linkinghub.elsevier.com/retrieve/pii/S0308814604007149.

[101] J Bassett, RC Denney, GH Jeffery, and J Mendham. Vogel's Textbook of Quantitative Chemical Analysis. *Longman: London 1989*, 519, 1989.

[102] X Hou, K E Levine, A Salido, B T Jones, M Ezer, S Elwood, and J B Simeonsson. Tungsten coil devices in atomic spectrometry: absorption, fluorescence, and emission. *Analytical Sciences : The International Journal of the Japan Society for Analytical Chemistry*, 17(1):175–80, January 2001. ISSN 0910-6340. URL http://www.ncbi.nlm.nih.gov/pubmed/11993659.

[103] Y Okamoto. Direct determination of lead in biological samples by electrothermal vaporization inductively coupled plasma mass spectrometry (ETV-ICP-MS) after furnace-fusion in the sample cuvette-tungsten boat furnace. *Analytical Chemistry*, 367(3):300–5, June 2000. ISSN 0937-0633. URL http://www.ncbi.nlm.nih.gov/pubmed/11227463.

[104] BL Sharp. Pneumatic nebulisers and spray chambers for inductively coupled plasma spectrometry. A review. Part 2. Spray chambers. *Journal of Analytical Atomic Spectrometry*, 3(October), 1988. URL http://pubs.rsc.org/en/content/articlepdf/1988/ja/ja9880300939.

[105] Masami Suzuki and Kiyohisa Ohta. Atomic Emission Spectrometry with Metal Microtube Atomization. *Analytical Chemistry*, pages 26–29, 1985. URL http://pubs.acs.org/doi/abs/10.1021/ac00279a011.

[106] Keith Levine, KA Wagner, and BT Jones. Low-cost, modular electrothermal vaporization system for inductively coupled plasma atomic emission spectrometry. *Applied Spectroscopy*, 52(9):1165–1171, 1998. URL http://www.opticsinfobase.org/abstract.cfm?uri=as-52-9-1165.

[107] George L. Donati and Bradley T. Jones. Development of a novel spectrometric-based temperature probe and the investigation of atomic cloud generation in a tungsten coil

atomizer. *Journal of Analytical Atomic Spectrometry*, 26(4):838, 2011. ISSN 0267-9477. doi: 10.1039/c0ja00172d. URL http://xlink.rsc.org/?DOI=c0ja00172d.

[108] WE Forsythe and AG Worthing. The properties of tungsten and the characteristics of tungsten lamps. *The Astrophysical Journal*, 61:146–185, 1925. URL http://adsabs.harvard.edu/full/1925ApJ....61..146F.

[109] Maxim Integrated. MAX4080/MAX4081 76V, High-Side, Current-Sense Amplifiers with voltage Output. Technical Report 19-2562, Maxim Integrated, San Jose, California, 2015. URL http://datasheets.maximintegrated.com/en/ds/MAX4080-MAX4081.pdf.

[110] Hamid R. Badiei, Bryant Lai, and Vassili Karanassios. Micro- and nano-volume samples by electrothermal, near-torch vaporization sample introduction using removable, interchangeable and portable rhenium coiled-filament assemblies and axially-viewed inductively coupled plasma-atomic emission spectrometry. *Spectrochimica Acta - Part B Atomic Spectroscopy*, 77:19–30, 2012. ISSN 05848547. doi: 10.1016/j.sab.2012.07.025. URL http://dx.doi.org/10.1016/j.sab.2012.07.025.

[111] A Salido and B T Jones. Simultaneous determination of Cu, Cd and Pb in drinking-water using W-Coil AAS. *Talanta*, 50(3):649–59, October 1999. ISSN 0039-9140. URL http://www.ncbi.nlm.nih.gov/pubmed/18967756.

[112] Scott Weagant, Vivian Chen, and Vassili Karanassios. Battery-operated, argon-hydrogen microplasma on hybrid, postage stamp-sized plastic-quartz chips for elemental analysis of liquid microsamples using a portable optical emission spectrometer. *Analytical and Bioanalytical Chemistry*, 401(9):2865–2880, 2011. ISSN 16182642. doi: 10.1007/s00216-011-5372-x.

[113] Hamid R. Badiei, Jennifer McEnaney, and Vassili Karanassios. Bringing part of the lab to the field: On-site chromium speciation in seawater by electrodeposition of Cr(III)/Cr(VI) on portable coiled-filament assemblies and measurement in the lab by electrothermal, near-torch vaporization sample introduction and indu. *Spectrochimica Acta - Part B Atomic Spectroscopy*, 78:42–49, 2012. ISSN 05848547. doi: 10.1016/j.sab.2012.10.002. URL http://dx.doi.org/10.1016/j.sab.2012.10.002.

[114] Hamid R. Badiei, Chuan Liu, and Vassili Karanassios. Taking part of the lab to the sample: On-site electrodeposition of Pb followed by measurement in a lab using electrothermal, near-torch vaporization sample introduction and inductively coupled plasma-atomic emission spectrometry. *Microchemical Journal*, 108:131–136, 2013.

ISSN 0026265X. doi: 10.1016/j.microc.2012.10.013. URL http://dx.doi.org/10.1016/j.microc.2012.10.013.

[115] Scott Weagant, Gurjit Dulai, Lu Li, and Vassili Karanassios. Characterization of rapidly-prototyped, battery-operated, argon-hydrogen microplasma on a hybrid chip for elemental analysis of microsamples by portable optical emission spectrometry. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 106:75–80, 2015. ISSN 05848547. doi: 10.1016/j.sab.2015.01.009. URL http://linkinghub.elsevier.com/retrieve/pii/S0584854715000208.

[116] Adil Masood, Marianna Pantouvaki, Danny Goossens, Guy Lepage, Peter Verheyen, Joris Van Campenhout, Philippe Absil, Dries Van Thourhout, and Wim Bogaerts. Fabrication and characterization of CMOS-compatible integrated tungsten heaters for thermo-optic tuning in silicon photonics devices. *Optical Materials Express*, 4 (7):1383, 2014. ISSN 2159-3930. doi: 10.1364/OME.4.001383. URL http://www.opticsinfobase.org/abstract.cfm?URI=ome-4-7-1383.

[117] M.J. Sailor, F.V. Mikulec, and J.D. Kirtland. Porous silicon-based explosive, May 17 2011. URL http://www.google.com/patents/US7942989. US Patent 7,942,989.

[118] Andrea Nagy, Edina Baranyai, and Attila Gaspar. Interfacing microfluidic chip-based chromatography with flame atomic absorption spectrometry for the determination of chromium(VI). *Microchemical Journal*, 114:216–222, 2014. ISSN 0026265X. doi: 10.1016/j.microc.2014.01.008. URL http://dx.doi.org/10.1016/j.microc.2014.01.008.

[119] David R. Lide. *CRC Handbook of Chemistry and Physics, 94th Edition, 2013-2014*, volume 53. 2013. ISBN 9781466571143. doi: 10.1136/oem.53.7.504.