Title
A vascular access system (VAS) for preclinical models

Author
Berry-Pusey, Brittany Nan

Publication Date
2012

Peer reviewed|Thesis/dissertation

University of California
Los Angeles

A Vascular Access System (VAS) for Preclinical Models

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Biomedical Physics

by

Brittany Nan Berry-Pusey

2012

Abstract of the Dissertation

A Vascular Access System (VAS) for Preclinical Models

by

Brittany Nan Berry-Pusey

Doctor of Philosophy in Biomedical Physics

University of California, Los Angeles, 2012

Professor Arion-Xenofon Hadjioannou, Chair

Preclinical Molecular Imaging technologies have an increasingly broader application base while they at the same time are becoming more user-friendly. Tail vein injections are a routine but critical step in most imaging applications, with poor injections greatly affecting the experimental results. The high skills and experience required to perform successful tail vein injections leave many preclinical imaging scientists ill-suited to perform this task. In a recent study, we found that trained routine injectors left on average 14% of the injected probe in the tail tissue. Improvements in injection accuracy, injection consistency, safety, and a reduction in time required to perform the task are needed in preclinical molecular imaging. To achieve these goals, we have devised a semi-automated vascular access system (VAS) to facilitate injections and eventually blood sampling from the mouse tail. We have eliminated much of the human error involved in the manual approach by using a computer-controlled mechanically moving micro-needle.

To make use of the VAS, one places an anesthetized mouse onto the temperature controlled mouse bed and secures the tail on a heated tail holder. The VAS uses NIR light, cross-polarizers, and a basic CCD camera to image the tail. The reflection image is processed and the vein is located using edge detection methods. The vein location is plotted and overlaid onto the live video feed of the mouse tail. Using a custom designed user interface, the user properly aligns the needle to the tail vein by employing computer-controlled motors.

Once the needle is properly aligned, it begins to penetrate the tail tissue and enter the vein. A pressure transducer attached to the needle detects when the needle has entered the vein, and automatically stops further progression of the needle. With the needle inside the vein, probes can be injected manually, with a liquid handling system, or via a syringe pump. The VAS was first validated using a mouse tail phantom. The phantom was a PDMS chip with channels equivalent to the dimensions of a mouse tail vein (300um). The channels were filled with water and pressurized to variable pressures. With the phantom, the ability for the VAS to align a needle according to an image, insert a needle into a desired location, and stop the progression of the needle based on a pressure signal were tested and verified. Mouse studies were also performed with the VAS. These studies showed that the accuracy of the device, as measured by the percentage of injected probe left in the tail, is 3.4% (+/- 4.5). The VAS reduces the operator skill requirements and training, has the potential to improve injection accuracy, reduces the time required to perform a tail vein injection, and is potentially safer for users and mice in comparison to current manual methods.

The dissertation of Brittany Nan Berry-Pusey is approved.

Chang-Jin Kim

Robert Michael Van Dam

Caius Gabriel Radu

Arion-Xenofon Hadjioannou, Committee Chair

University of California, Los Angeles

2012

To my husband . . .

who joined and supported me

on this crazy adventure

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank my collaborators from the Mechatronoics and Controlls lab for their contributions to this project. Specifically Stephen W. Prince for his assistance in the design and fabrication of the final VAS model, Kevin Chu for his contributions to the controlls system and arduino programming, and Yen-Chi Chang for his contributions to the controlls system and labview programming. Also, their advisor T.C. Tsao for his words of wisdom. Their contributions helped to take this project to the next level.

Many helpful discussions and much advice came from the Crump Imaging Center, specifically from Waldemar Ladno, Darin Williams, and Dr. David Stout. Waldemar provided me with much information on injections and things to be aware of. He was my gold standard of what the VAS should be.

Thank you to Dr. Alex Garcia for his assistance in performing blood smears.

None of this would have been possible without my advisor, Dr. Arion Hadjioannou, and the support of my labmates. Thank you to Dr. Hadjioannou for trusting me to move this project forward and believing along with me that is was possible. Thank you to Bob Silverman for his design of the pressure transducer amplifier circuit and for sharing his electrical engineering insights with me. I am always amazed at the brilliance of Dr. Richard Taschereau. I had many helpful discussions about image processing and PET analysis with him. Dr. Nam Vu was a great sounding board for me and I am of appreciative for the example he set for me. It was a privilege and honor to work alongside and learn from Dr. Hui Zhang, Dr. Ritva Lofstedt, Dr. Hongkai Wang, Dr. Elena Heckathorne, Dr. David Prout, Dr. Ali Douraghy, Dr. Qinan Bao, Dr. Jennifer Cho, Yanisley Valenciaga, Zheng Gu, John David, and Alex Dooraghi.

Thank you for the administrative support of Terry Moore, Reth Thach, Karen Lum, and Erika Corrin.

It was a pleasure coming into the BMP program with classmates Megan Brown, Neil Wilson, Wei Sha, Di Zhang, and Michele Zhang. I am also grateful for the support and

good times with other BMP and Pharmacology colleagues including Graham Cole, Robin Saxon, Nate Martin, Mirwais Wardak, Adam Kesner, Nicole Detorie, Jimmy Ly, Marissa Briones, Sherly Mosessian, Shannon Sirk, Katelyn McCabe, Scott Knowles, and Freddie Daver. Thank you to my "science ladies" Sarah Gibson and Katie Bulgrin.

Working with the Business of Science Center during my graduate career has been a life changing experience for me. I will always be grateful for the mentorship and experience gained by working with Roy Doumani, Cheryl Matter, and Nicole Matter. Thank you for taking a chance on me.

I came to graduate school because of Dr. Brian Saam. Working in his lab for four years gave me the foundation to succeed as a scientist. He is an outstanding mentor and I thank him for his patience with me. Dr. Steve Morgan, Dr. Jared Teter, Dr. Ben Anger, and Dr. Tricia Hauschild also contributed to my desire to continue with my studies.

Most importantly, I would like to acknowledge my family. Because of them, I know I can achieve anything.

Vita

1998–2002	Park City High School, Park City, UT.
2002	University of Utah ACCESS Program and Honors at Entrance Scholarship
2003–2006	Research Assistant, Saam Hyperpolarized Nobel Gas Lab, University of Utah, SLC, UT.
2003-2004	Intellectual Exploration Scholarship
2005	Sigma Pi Sigma Physics Honor Society
2005–2006	University of Utah Physics Department Scholarship
2006	University of Utah College of Science Research Scholar Award
2006	B.S. (Physics) and Minor (French), University of Utah.
2007	UCLA Eugene Cota Robles Fellowship
2008	UCLA Molecular and Medical Pharmacology Student Representative
2008-2012	Graduate Student Researcher, Department of Molecular and Medical Pharmacology (Biomedical Physics), The David Geffen School of Medicine at UCLA, Los Angeles, CA.
2009	Best Poster Presentation Award, Biomedical Physics Colloquium
2009-2012	UCLA Business of Science Center Student Chair, Los Angeles, CA.

2010	M.S. (Biomedical Physics), David Geffen School of Medicine at UCLA, Los Angeles, California.
2011	Perkins Coie Innovative Minds and UCLA Business of Science Center Venture Team Winner
2012	World Molecular Imaging Congress Young Investigator Award Semifinalist

CHAPTER 1

Introduction

1.1 Motivation

The NIH has identified translational research as a top priority, meaning there will be a larger emphasis on taking research from the lab bench to the clinical bedside [14]. To achieve this, in-vivo experiments with animal models are often used due to the similarities of their biochemical interactions and pathways to those of humans. *In-vivo* imaging of animals has emerged as an important tool in biomedical translational research [25]. Suitable preclinical imaging technologies range from optical (bioluminescence and fluorescence), radionuclides (PET and SPECT) as well as ultrasound, x-ray CT and MRI. All of these different imaging modalities have experienced an explosive growth in both academic research centers and the biotechnology and pharmaceutical industries. The choice of imaging modality is dependent on the specifics of the experimental goal, but most imaging experiments require vascular access of an anesthetized mouse for the injection of a molecular probe or a contrast agent. Furthermore, blood samples are required for longitudinal analysis of treatment effects, and to measure physiological changes as well as metabolites from drugs and imaging probes. Accessing the mouse vasculature is a process that requires technical skill and experience and therefore has a constrained success rate, highly dependent on the dexterity of the individual researcher [22], [26]. It is not uncommon that poor probe injections compromised the results of complicated and expensive experimental protocols. Furthermore, withdrawal of blood samples from mice is even more difficult experimentally and in most cases requires terminal experiments. These types of endpoint experimental protocols are in general incompatible with the advantages of noninvasive in-vivo imaging.

Over the past 11 years, at the Crump Institute, we have performed more than 80 thousand injections in mice for molecular imaging [42]. A repeated concern and request among all the investigators and their research staff has been assistance with tail vein injections for the experimental protocol. Furthermore, a number of terminal experiments have been performed, to provide measurements of the blood concentration of molecular imaging probes [27]. In this work, we aim to automate the injection and blood sampling process, and eliminate it as a potential source of experimental failure, which will certainly impact and improve the results from our, and hopefully many other, preclinical imaging centers.

1.2 Why Mice?

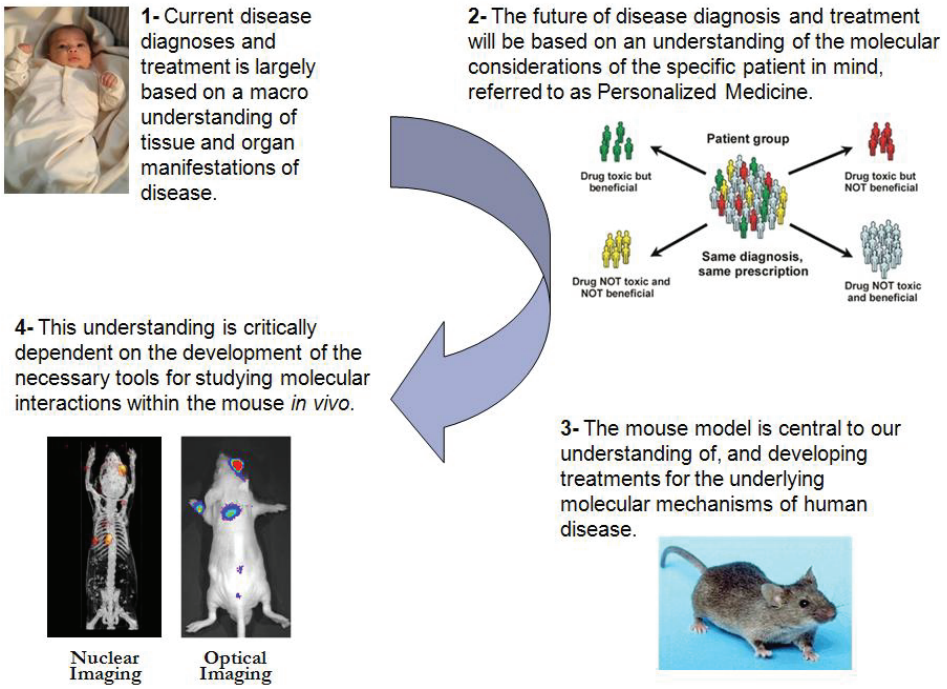


Figure 1.1: The role of murine models in the study and treatment of human disease. Fig. from ref. [18]

Many diseases are studied in non-human models, including primates, dogs, cats, swine, rabbits, and rodents, but 90% of all mammalian disease studies use the mouse (murine) model [31]. The murine model is a top choice for several reasons, including the high genetic

homology with humans, the ease of genetic manipulation, the ability to transplant human cells into the model, the short gestation period, and the low cost of housing and maintaining murine models [12]. With the creation of genetically manipulated mice, the genotypic basis of diseased and normal biological processes can be readily studied [35]. Using and studying murine models is a crucial and necessary element of the scientific process of studying fundamental biological processes and developing treatment therapies, especially as human treatments become more personalized.

1.3 Manual Injection and Blood Sampling Techniques

Intravenous (IV) or intraperitoneal (IP) mouse injections play an important role in basic science, drug development, imaging, and toxicology studies. Due to its easy access, the most common site for IV injections in murine models is the tail vein. Tail vein injections are a non-terminal procedure that can be repeatedly performed with or without anesthesia, enabling longitudinal studies. To perform a manual tail vein injection, the mouse is anesthetized or restrained. The tail is gently heated or massaged to promote the dilation of the blood vessels. The tail is stretched and held between the thumb and forefinger (fig. 1.2). A 25-30 gauge needle is aligned parallel to the tail vein and inserted approximately 15-40 mm from the base of the tail, bevel up, into the vessel. Once the needle is in the vein, which can be validated by the detection of a flashback of blood into the syringe, the desired compound can be administered to the subject by depressing the syringe plunger. The quality of the injection can be qualitatively assessed by noting the ease of the plunger movement, the lack of blanching around the injection site, and the observation of bleeding at the injection site after the needle is removed [8]. These qualitative observations are not accurate indications of a good quality injection, defined as an injection with less than 10% of the injected volume remaining in the tail. In a study performed by Groman et al. three trained technologists each injected ten different mice. Out of the thirty injections, one injection was qualitatively observed to be a poor injection. A quantifiable analysis was executed on the tail injections and is was discovered that twelve of the thirty injections were misadministered [22]. This

high percentage of tail vein injection failure highlights the difficulty of a user to accurately assess the quality of a manual tail vein injection and also the difficulty in performing a skillful injection.

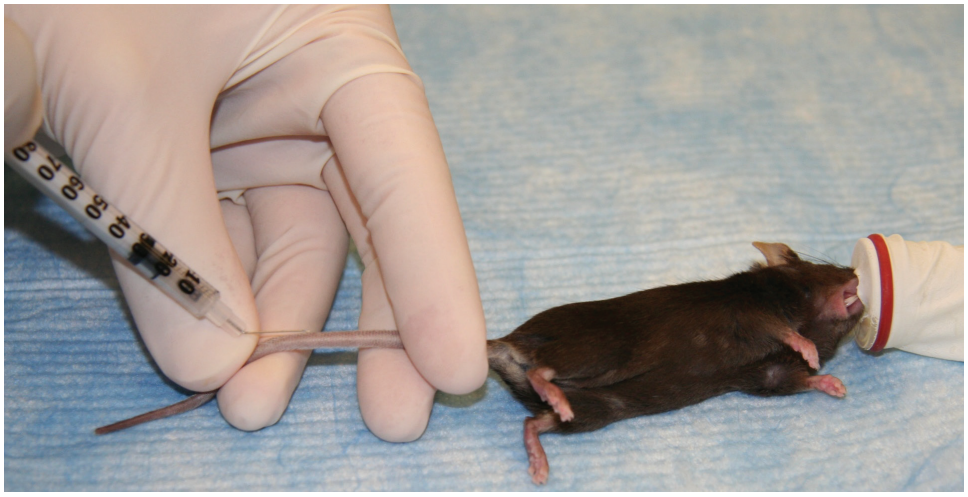


Figure 1.2: An image of the setup for a manual tail vein injection.

Another reason scientists need access to the vasculature of murine animals is for the collection of blood samples. There are several methods and locations to collect blood from the animal. Without anesthetizing the mouse, blood is commonly secured from the saphenous vein and dorsal pedal vein. Anesthesia is required when blood is collected from the tail tip, orbital sinus, or jugular vein. The subject may be sacrificed when blood is drawn from the posterior vena cava, orbital sinus, axillary vessels, or by cardiac puncture. Depending on the location and method used, the blood sample may include venous blood, arterial blood, or a mixture of both [26]. Many articles describe how to perform the mentioned procedures [26], [24]. To perform such procedures, *The Guide for the Care and Use of Laboratory Animals* dictates that personnel must be properly trained [15].

If blood is collected for a survival study, there is a limit on how much blood can be sampled. The average mouse has 6-8 mL of blood per 100 g of body weight, or the blood volume is approximately 6-8% of the total weight of the mouse. Thinner animals as well as younger animals have larger blood volumes relative to the body mass compared to average or older mice [5]. Guidelines dictate that the maximum safe volume of blood to withdraw

at one time is 10-15% of the total blood volume. Because blood volume is restored in 24 hours, but normal levels of erythrocytes and reticulocytes take two weeks to be achieved, it is recommended that this high level of blood sampling should only occur once every two weeks. For daily blood sampling, removing up to 1% of the total blood volume is permissible [26].

Access to the vascular system in murine models is often required for both injections and blood sampling. This task is difficult and requires training. Many guidelines have been produced to optimize this manual process [2], [3], [1].

1.4 Mouse Tail Anatomy

Surprisingly, there is little published information on the anatomy of the mouse tail. To obtain a better understanding of the tail, we gathered information by collecting several different types of data. By visually inspecting different tails of various strains of mice, one can observe that the tail has scales, hair, and pigmentation on it's surface. By measuring

Table 1.1: Mouse Tail Dimensions

	mm	S.D.
Average Tail Length	80.6	8.7
Diameter at Base	3.21	.27
Diameter at Midpoint	2.42	.24
Diameter 1 cm from Tip	1.21	.17

the tails of ten different mice of two different strains that ranged in age from 14-18 weeks old, it was found that the average length of the tail was 8.06 cm with a SD of .87 cm. The outer diameters of these tails were measured at three different locations, the base of the tail (section closest to the body), the midpoint of the tail, and 1 cm from the tip of the tail. The mean diameters were 3.21 mm, 2.42 mm, and 1.21 mm respectively with standard deviations of 0.27 mm, 0.24 mm, and 0.17 mm. There are four main vessels in the mouse tail: two lateral tail veins, a dorsal vein, and a ventral artery. The two lateral veins are

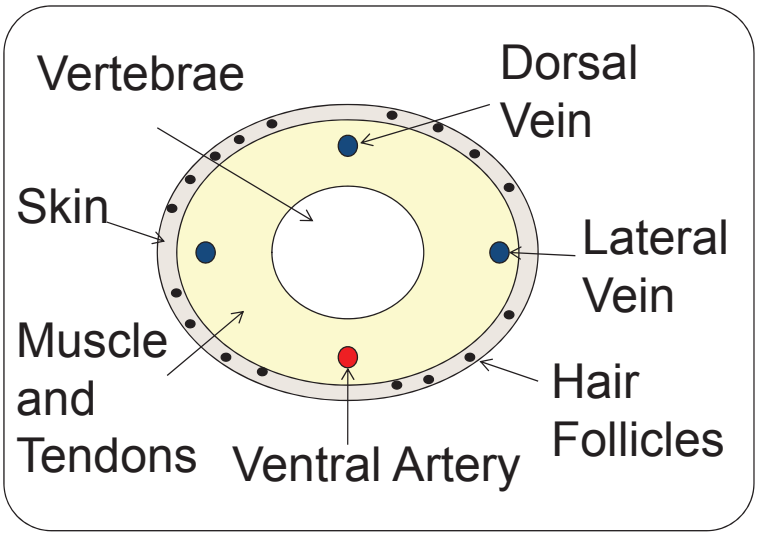


Figure 1.3: Drawing of the cross-section of the mouse tail [23].

used for tail vein injections. To obtain a better understanding of the vessel sizes in the tail, we worked with Dr. Steven Staelen's group at Ghent University in Belgium. They have a casting technique where they perfuse a casting agent through the arterial system or the venous system [10]. From the venous cast seen in Figure 1.4, the diameter of the lateral veins were measured to be about 600 μm. This size is assumed to be an overestimate as veins tend to expand from the casting agent. An artery in the tail was measured to be approximately 300 μm from an arterial cast. This diameter is more accurate than the vein measurement as the arteries have muscles surrounding them that give them extra support and reduce the ballooning effect from the casting agent. Cross-sectional histological slices of the mouse tail were also looked at. These slices were decalcified using formic acid and a hematoxylin-eosin stain was used. The diameters of the tail veins and arteries from these slices were measured to be about 150 μm. There is a shrinkage that occurs in histological slices that is estimated to be about 50%. We therefore estimate the true diameter of the vessels in the tail to be about 300 μm. We also estimated from the histological slices that the depth of the center of the vessel is about 300-600 μm from the surface. It is worth noting that the diameter of the mouse tail vein is about six times smaller than the diameter of the average human median cubital vein, which is commonly used for vascular access in humans [41]. Knowing

the general anatomy of the mouse tail and approximate vessel diameters, we can design and optimize a system for automated injections and blood sampling.

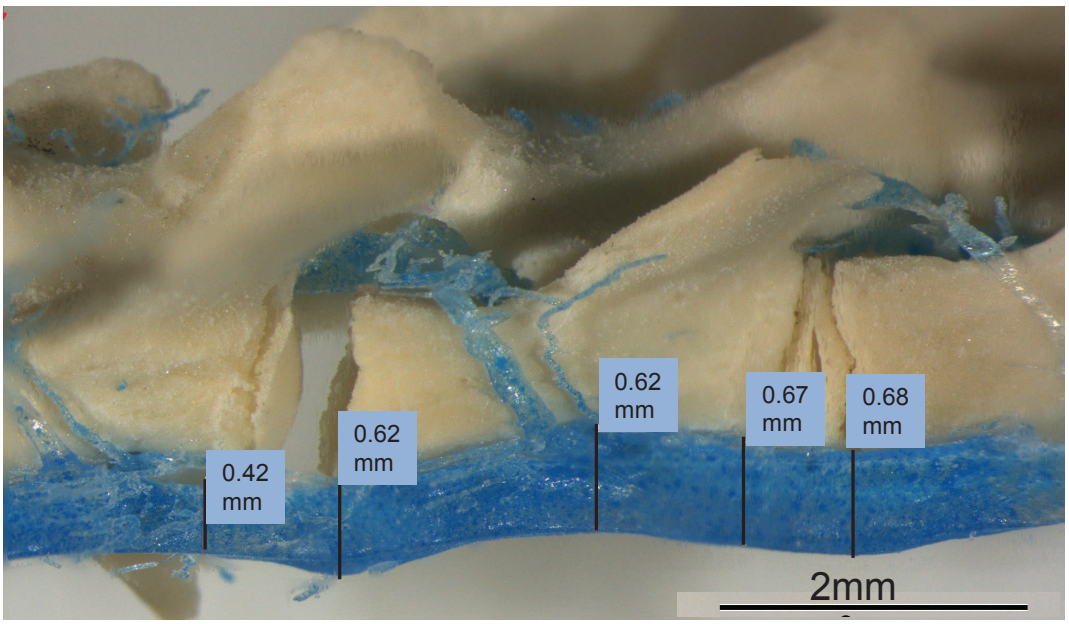


Figure 1.4: Tail vein cast

1.5 Positron Emission Tomography (PET)

Positron Emission Tomography (PET) is a non-invasive imaging modality that detects the temporal and spatial biodistribution of molecules labeled with positron emitting isotopes. Trace amounts of these molecules are injected into the vasculature of a subject to monitor biological processes such as blood perfusion, enzyme activity, and metabolism, and to detect cancer, monitor treatment, track drugs, and more [33] [39]. The basis of PET is the coincidence detection of two 511 keV annihilation photons. These annihilation photons, or γ-rays, are the result of an emission of a positron from the nucleus of an injected radioisotope. The positron is emitted, travels a short distance, loses its kinetic energy and annihilates with an electron resulting in the simultaneous emission of two γ-rays in opposite directions. A ring of detectors surrounding the subject detects the two γ-rays. The detectors commonly consist of scintillating crystals and photomultiplier tubes (PMT). The scintillator stops the gamma rays and converts their energy to visible light, which is then detected by the PMTs.

When opposing PMTs both detect an event in a defined timing window, these events are attributed to the same positron and can be traced back to the location of the positron annihilation. Millions of the these annihilation events form the raw PET data, which can then be mathematically reconstructed to generate an image of the 3D spatial distribution of the labeled molecules. PET imaging is used both clinically and pre-clinically. Because PET requires radioisotopes, which are most commonly intravenously introduced, our device focuses mainly on, but is not limited to, the needs of preclinical PET imaging. Since PET is quantitative, we will also use PET to validate the performance of the VAS.

1.6 The Need for the VAS in PET and Other Applications

There are many reasons that a vascular access system (VAS) are beneficial to preclinical scientist including:

- Reducing the skill set required for scientist to perform tail vein injections
- Increase the repeatability, robustness, and consistency of injections, which can impact experimental outcomes
- Increase injection safety for the subject and the user
- Enable high-throughput animal handling for optical (fluorescence and bioluminescence), magnetic resonance (MR), computer tomography (CT), and PET imaging

Training and experience is required to perform good tail vein injections. This is explored further in Section 4.2 of this dissertation. A semi-automated system, like the VAS, will allow all scientist to perform their own tail vein injections without depending on the relatively few trained and experienced injectors.

Not having repeatable and consistent injections can effect experimental outcomes. For example, for preclinical PET imaging, image quality and analysis can be impacted based on the quality of the probe injection. Probe injections for murine models are commonly administered through the tail vein. Poor tail vein injections can affect image quality by

lowering the uptake value due to the small fraction of dose that is actually delivered to the subject, the correct input function is unknown, and standard uptake value (SUV) or calculated kinetic parameters can be inaccurate because the dose delivered to the subject during the time of the scan is unknown. A paravasculature injection causes slow diffusion, which can alter the time activity curve of PET probes and even the biodistribution.

Many preclinical imaging modalities are becoming more user-friendly to enable high-throughput imaging studies. As an example, our lab has recently developed PETBox: a low-cost, benchtop, user friendly PET scanner for the imaging of mice [49]. Because of the low start-up cost and reduction of required skills to use PETBox, this scanner is designed to increase the number of the biological scientists who capitalize on the advantages of using PET imaging for research. The VAS will assist new user friendly and high-throughput imaging

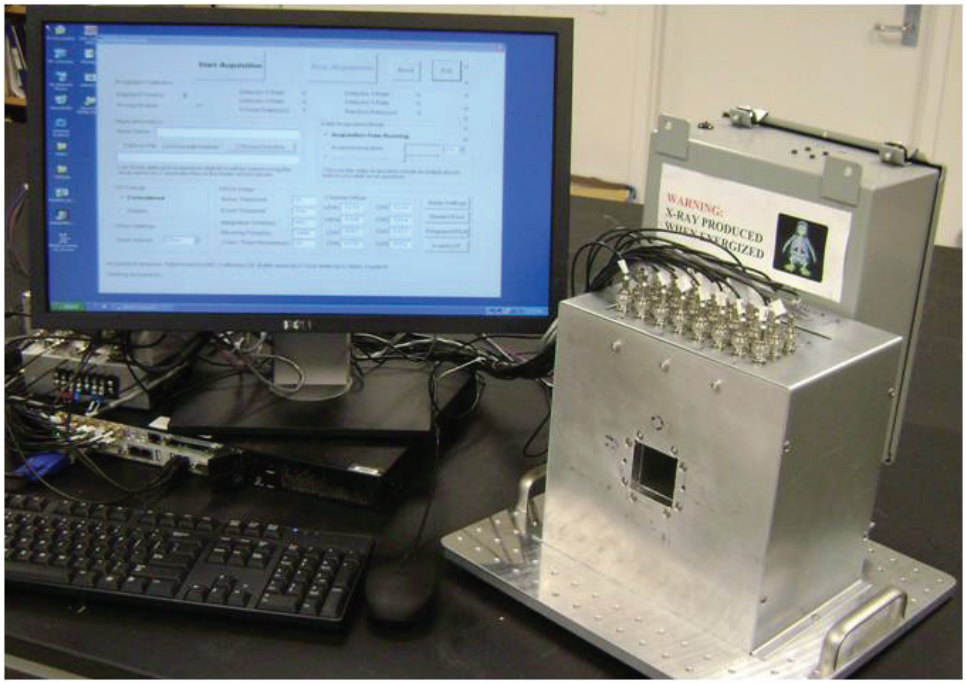


Figure 1.5: Prototype of PetBox4, a low-cost, bench top preclinical PET scanner

devices with some of the animal handling requirements required to perform an imaging study by allowing scientists to inject murine models with a push of a button.

1.7 Specific Aims

The Vascular Access System (VAS) is a dedicated device to aid with injections and blood sampling from the mouse tail vein.

The specific aims of this work are to:

1. Use optical imaging to identify and locate subsurface blood vessels in a mouse tail,
2. Use computer controlled mechanical means to insert a needle into the identified vessel,
3. After the needle is properly placed in the vessel, inject a probe into a mouse,
4. Integrate the above components into a user friendly benchtop device (VAS)that can be sterilized,provide reproducible results, and increase the safety for researchers injecting hazardous materials, and
5. Validate the VAS for reliability and consistency using PET imaging.

CHAPTER 2

System Concept

2.1 VAS Concept Overview

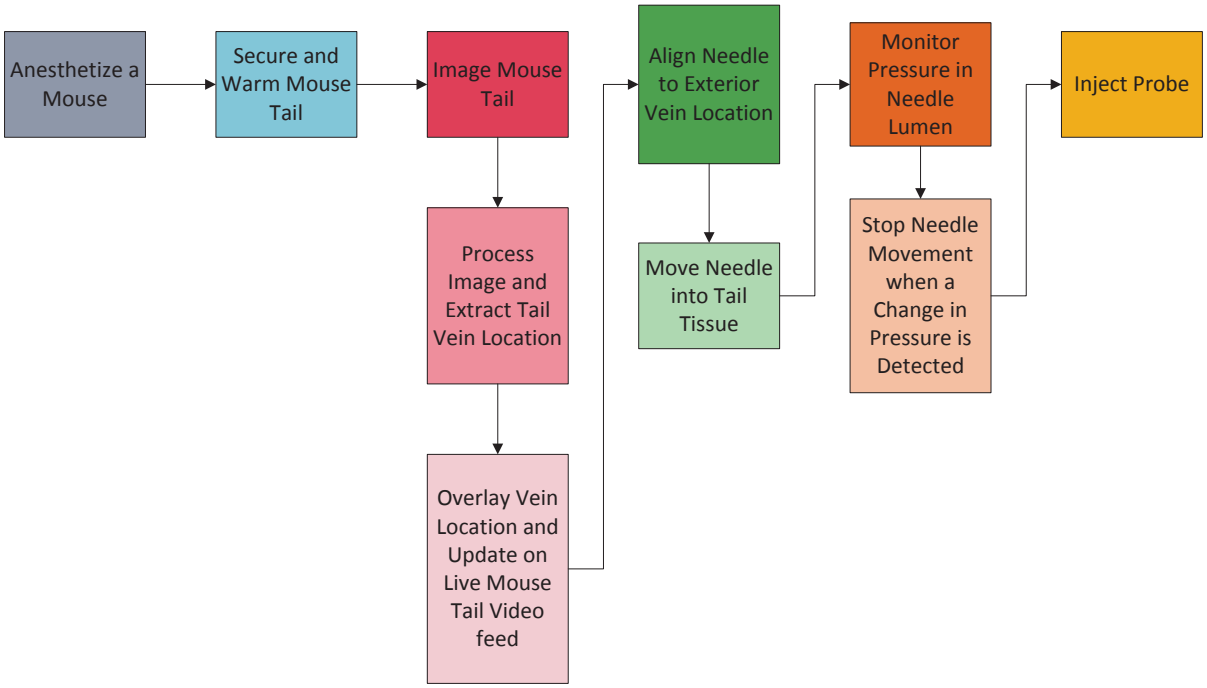


Figure 2.1: High-level VAS methodology

The main concept of the VAS is to semi-automatically insert a needle into the tail vein of an anesthetized mouse. Figure 2.1 is a flowchart of the basic principles of the VAS. The VAS works by first securing the tail to prevent it from moving during needle insertion while still allowing physical and visual access to the lateral vein. The vein is located using a CCD camera and near infra-red (NIR) light. Once the vein location is identified, a needle is

moved to that location by computer-controlled motors. Because the vein is imaged only in two dimensional space, the depth of the vein is unknown. A pressure feedback system is used to signal to the motor when the correct depth of the needle is reached. A pressure transducer is attached to the needle. This pressure transducer measures the pressure inside the lumen of the needle. The motors move the needle inside the tail tissue. When the needle enters the vein lumen, a change in pressure in the needle occurs because of the blood pressure of the mouse. This change in pressure indicates that the needle has entered the tail vein and signals the motors to stop moving the needle. Once the needle is inside the vein, the desired probe can be injected through the needle into the mouse vascular system. Thus, the VAS uses a combination of a camera, motors, and a pressure feedback system to insert a needle into the mouse vascular system.

2.2 Warming the Mouse and Tail

Without an external heat source the core body temperature of a mouse, under anesthesia, can drop dangerously low [20]. During sedation, a mouse is not moving and creating heat. A mouse body is usually stretched out while sedated, increasing the surface area to volume ratio and, therefore, increases heat loss. For these reasons it is important to actively keep a mouse warm while sedated. It is also noteworthy that the mouse thermo-regulates itself through the tail. As the temperature of the mouse rises, vasodilation occurs to decrease the mouse temperature [40], [21]. Vasodilation is beneficial for the VAS as it increases the diameter and the blood flow through the vein making the vein easier to image and a larger target into which the needle can be inserted. Heating elements were added to the VAS to prevent mice from entering a hypothermic state while under anesthesia, causing vasoconstriction in the tail. In turn with the active heating, the tail vein vasodilates and is easier to access.

The VAS uses two different heating strips with independent controllers to heat the tail and the body of a mouse. More information about the heaters and controllers can be found in section 3.2.4 of this dissertation. The mouse body heater is used to maintain the core body temperature of the mouse while anesthetized. The normal core body temperature

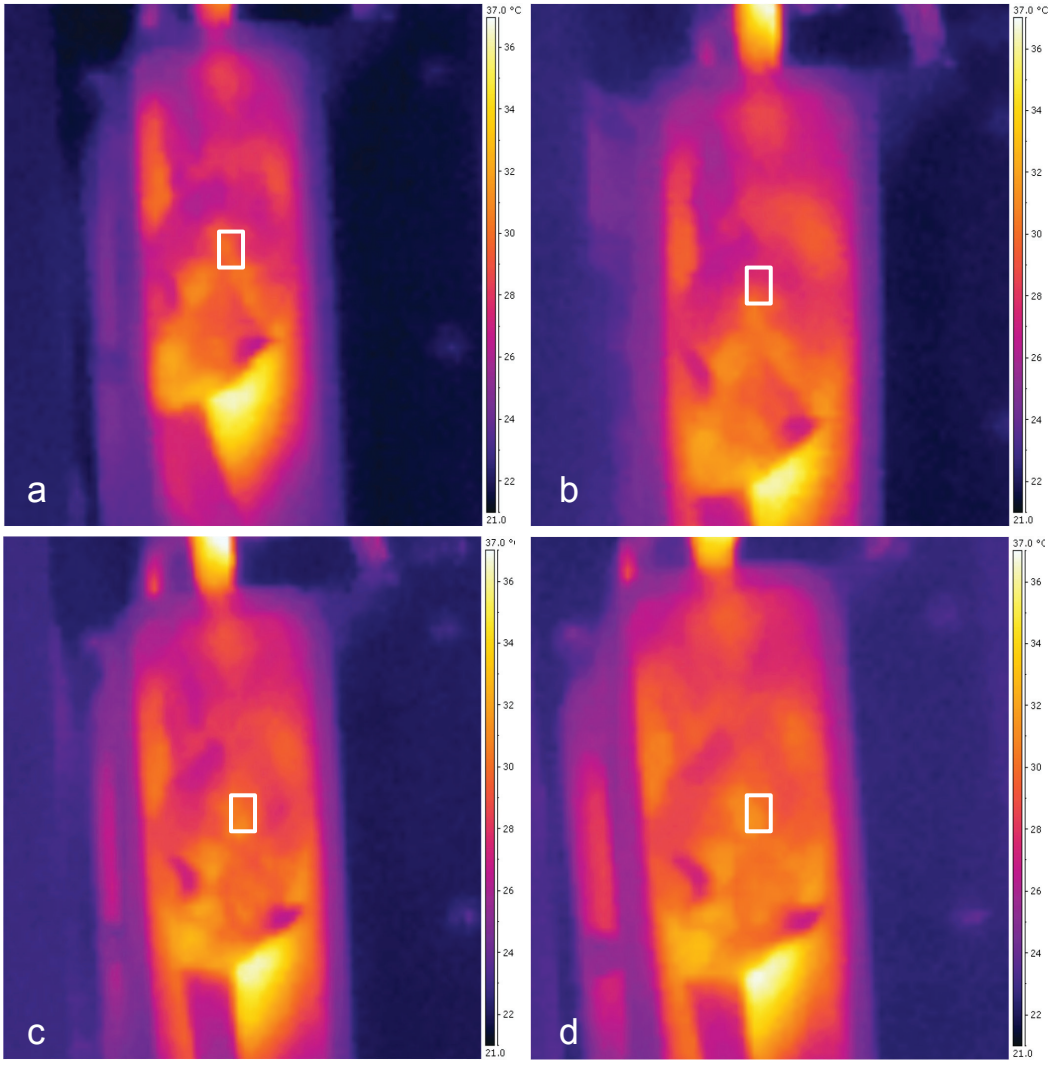


Figure 2.2: Thermal images of a mouse on the heated VAS mouse holder. The average temperature was calculated for the areas within the white box. Image *a*, one minute after placement on heater, average temperature 29.5°C. Image *b*, four minutes after placement on heater, average temperature 29.6°C. Image *c*, eight minutes after placement on heater, average temperature 30.0°C. Image *d*, 10 minutes after placement on heater, average temperature 30°C.

of a mouse is 36-38°C [21], [44]. The heater for the mouse body is a heated plastic plate that the mouse rests on. The Crump Institute Preclinical Imaging Center at UCLA has historically used heating plates to maintain the core body temperatures of mice. It is has been found that heating the plates to a set point between 37-38°C adequately maintains the

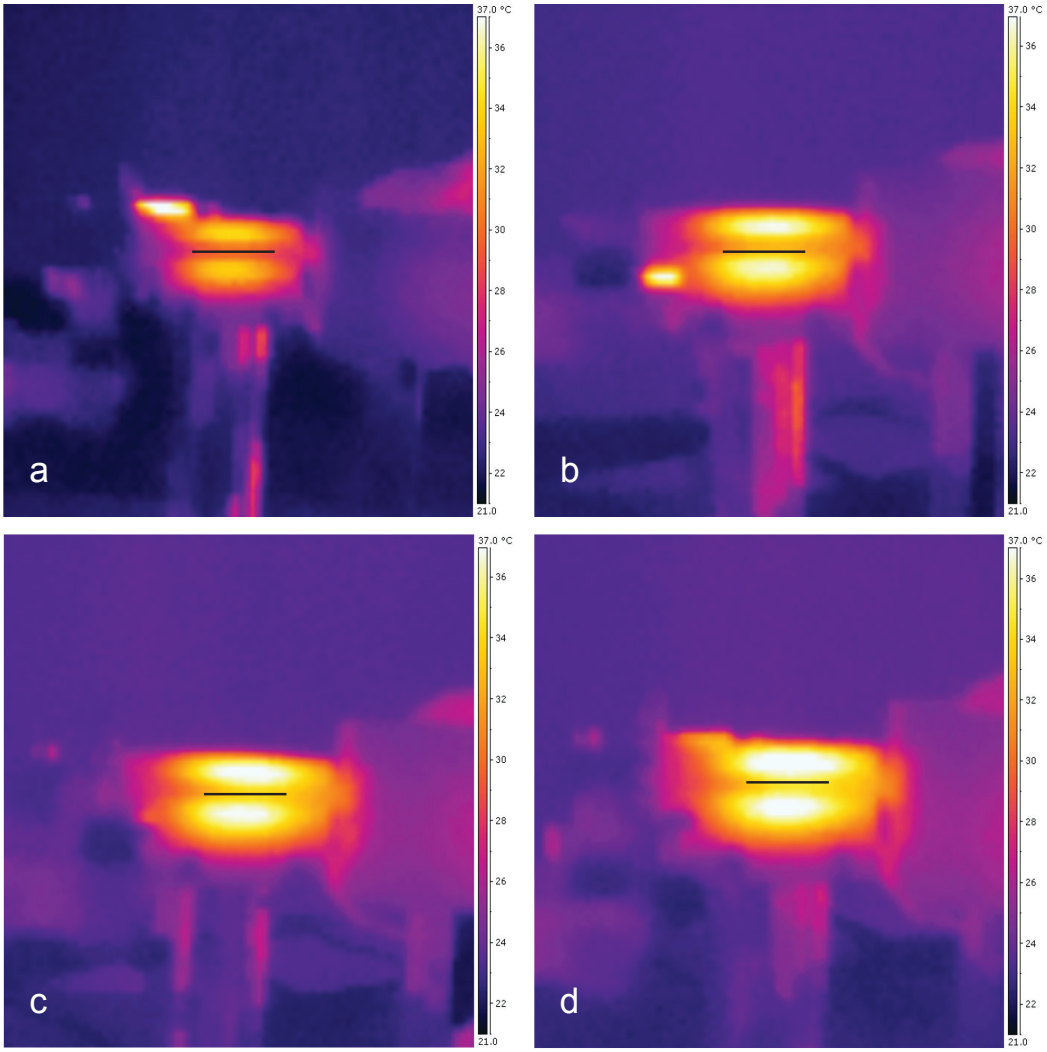


Figure 2.3: Thermal images of a mouse tail on the heated VAS tail holder. The entire tail holder is heated. The tail is located near the middle of the heated region. The average temperature of the tail was calculated along the black line. Image *a*, one minute after placement on heater, average temperature 28.8°C. Image *b*, four minutes after placement on heater, average temperature 31.9°C. Image *c*, eight minutes after placement on heater, average temperature 33.1°C. Image *d*, 10 minutes after placement on heater, average temperature 34.2°C.

body temperature of a mouse [43]. The heating plate that holds the mouse for the VAS is heated to 37.5°C by a heating strip (Minco HK5329R24.8L12B) and a heaterstat sensorless DC controller (Minco CT198-1007R24.8L1). The heated tail holder for the VAS is made out

of plastic delrin and heated with a similar Minco heating strip and controller as the mouse body heater. To encourage vasodilation in the tail, the temperature set point for the tail heater was set at 41.5°C. This is below the critical thermal maximum for mice that could cause the death of a mouse if the mouse core temperature rose above it [21], and similar to the temperature used by manual injectors to warm the tail before an injection.

The temperature of a mouse on the VAS mouse holder heating bed and heated tail holder was tested using an infrared thermal imaging camera (Flir i5). For these experiments, four minutes after the heating strips were turned on, an anesthetized mouse was placed on the VAS heating plates. Surface thermal images were taken at different time intervals and analyzed. These images are displayed in Figure 2.2 and Figure 2.3.

2.3 Imaging of the Blood Vessel

To image a blood vessel in a mouse tail, an Imaging Source monochrome CCD camera DMK 31BU03 was used with a variable focal lens (Computar H3Z1014CS). The camera has a field of view of 1024x768 pixels and an 8 bit dynamic range. The camera is operated with an exposure time of 0.0625 seconds and an aperture of f/1.4.The spectral sensitivity of the camera peaks at a wavelength of about 500nm, as seen in Figure 2.4. In the NIR range, the camera has a relative response of approximately 40% at 750nm and drops to 5% at 1000nm. One of the most important features of this camera is that it does not have an IR cut filter, which allows us to image in the desired NIR range. We added an optical cast plastic IR longpass filter (EO NT43-948) between the lens and CCD to block visible light and only pass IR light. The spectral response of this filter is seen in Figure 2.5. This allows us to image in the range of interest without requiring strong environmental controls, i.e., our setup can be used on a benchtop without a black box. This is beneficial for our system as it gives us more flexibility in the design of the system and makes the design more practical. After the wavelength of light was chosen for the VAS, a narrower NIR bandpass filter was chosen. The current filter has a central wavelength of 850 nm and a full width half maximum of 100 nm (Ocean Thin Films NIR Bandpass filter), as seen in Figure 2.6.

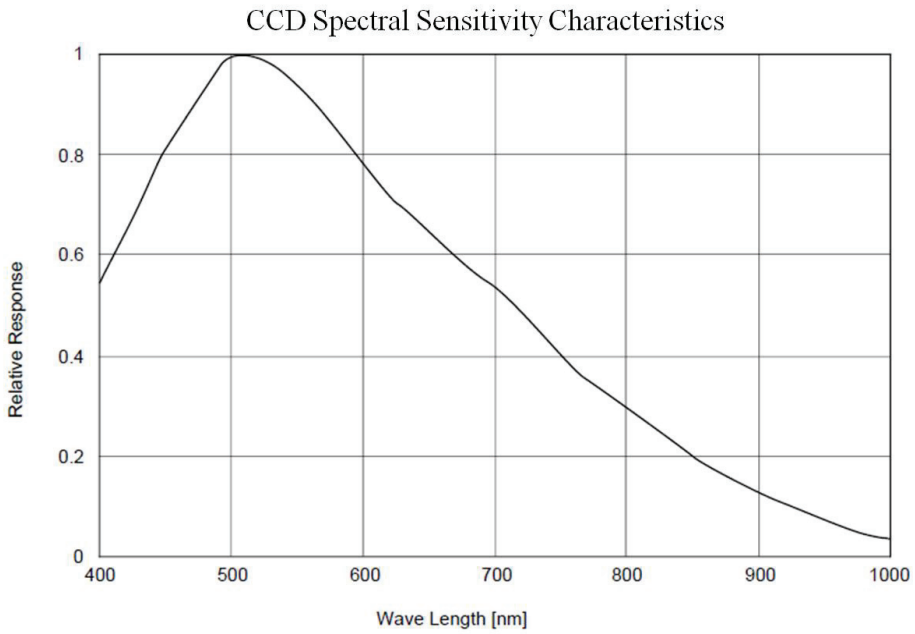


Figure 2.4: Spectral Sensitivity of VAS CCD camera

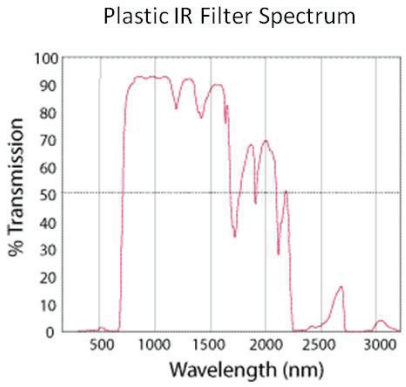


Figure 2.5: IR longpass filter spectrum

2.3.1 Near Infrared Light and Crosspolarizers

The surface of a mouse tail may have scales, hair, and skin pigmentation in the visible wavelength, which can vary from mouse to mouse and between strains. For this reason, we use near infrared light (NIR) to image the tail vessels. T. Vo-Dinh *et al.* previously characterized the propagation of light in tissue as a function of wavelength [47]. NIR light is less sensitive than visible light to skin pigmentation and penetrates deeper into the skin

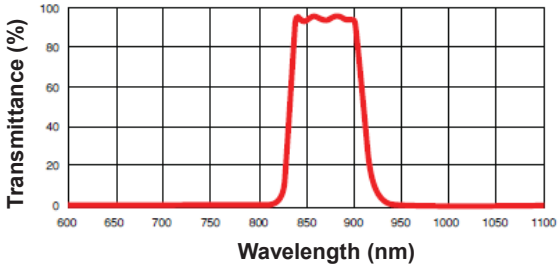


Figure 2.6: NIR bandpass filter spectrum

because of the decrease in tissue absorption, allowing for deeper vessels to be imaged [37]. NIR light may also be imaged with standard CCD cameras. Although the absorption of light

Figure 2.7: Image of a mouse tail

in tissue has been well characterized, it is still important to experimentally test different wavelengths as surrounding tissue, fat, etc. will have a large impact. We tested several different wavelengths to image the tail vein of a mouse. We used LEDs with peak emissions at 700, 750, 880, 900 and 970 nm to capture an image of the tail. Regions of interest (ROI's) were drawn on the unprocessed image in the tissue area of the tail and also the vein section. The ROI's were analyzed to find the mean pixel values and the standard deviation. Using the ROI values, the contrast to noise ratios (CNR) were calculated using the following equation:

$$CNR = \frac{\mu_V - \mu_T}{\sigma_T}, \tag{2.1}$$

where μ_V is the mean image intensity value of the vein, μ_T is the mean image intensity value of the background, which in this case is the tail tissue, and σ_T is the standard deviation of intensity in the tissue [48]. The CNR for different wavelengths is found in Table 2.1 The

Table 2.1: CNR of the tail for different wavelengths

Wavelength	Vein Mean	Vein STD	Tissue Mean	Tissue STD	CNR
700	60.2	5.2	79.2	7.8	-2.4
750	90.4	14.0	109.6	16.9	-1.1
880	24.4	4.7	35.7	4.2	-2.7
900	29.8	4.2	31.9	4.2	-0.5
970	51.9	4.3	59.6	4.8	-1.6

largest CNR was found with an LED with a peak emission at 880nm. We built an LED light ring consisting of 13 LEDs emitting at 880nm with an intensity of 94mW/Sr (Fairchild QEC122), as seen in Figure 2.8. The LEDs were wired in parallel to each other, each LED

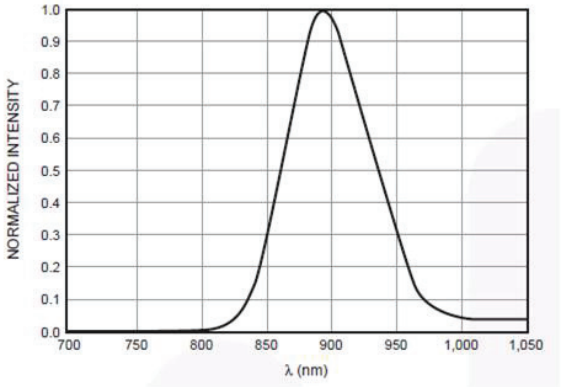


Figure 2.8: QEC122 LED intensity vs. wavelength

having a 150Ω resistor in series.

The VAS imaging setup also includes NIR linear cross-polarizers. It has been shown that cross-polarizers can be used to separate the reflectance of the surface from the light that is backscattered in tissue [4], [16], [29]. There is a change in the refractive index at the air-tissue interface. At this interface, approximately 4% to 7% of the incident light is reflected causing glare at the surface. The remaining portion of incident light enters the tissue and is scattered and/or absorbed. Scattering of the light causes depolarization, yet regular reflectance preserves the plane of polarization [4]. This provides an avenue to minimize the surface reflectance in the mouse tail image. The incident light can be linearly

polarized. An analyzing polarizer is used at the CCD. The analyzing polarizer, which is set orthogonal to the polarization of the light, rejects the surface reflected light yet allows some of the scattered light (with random polarization) from the tissue to pass to the CCD. This method of using cross-polarizers enhances the view of the vasculature of the tail. We use the NIR linear polarizing film (Edmund Optics NT54-112) to polarize the LED light and to analyze the light coming into the CCD camera. Figure 2.9 is a photo depicting the VAS CCD camera with a LED ring and polarizer.

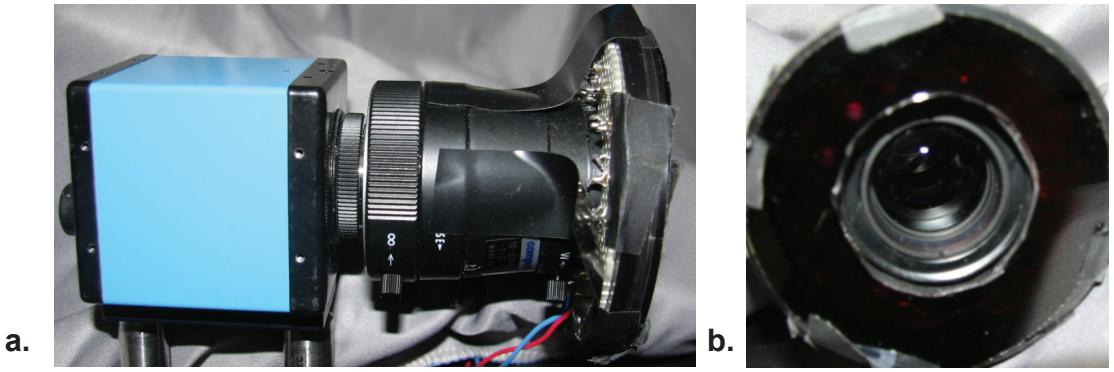


Figure 2.9: *a.* CCD camera with LED ring and polarizer, *b.* LED ring covered by a linear polarizer

2.3.2 Image Processing and Vein Extraction

Once the tail image is obtained, it must be processed to extract the location of the vessel. First, the tail is selected from the image. Then, a heavily smoothing 2D Gaussian filter with a sigma of 25 is convolved with the tail image creating a blurred image. A 2D Gaussian is defined in Equation 2.2, where x and y are the distances from the origin for the respective axis and σ is the standard deviation of the Gaussian distribution (the width of the Gaussian) [34]. This blurred image is subtracted from the original image. The result is a high pass filtered image. A high pass filtered image appears sharper than an unfiltered image and emphasizes the fine details in an image. It also amplifies noise in an image. To reduce noise, a second Gaussian filter, with a smaller sigma of 7.7, is convolved with the high pass filtered

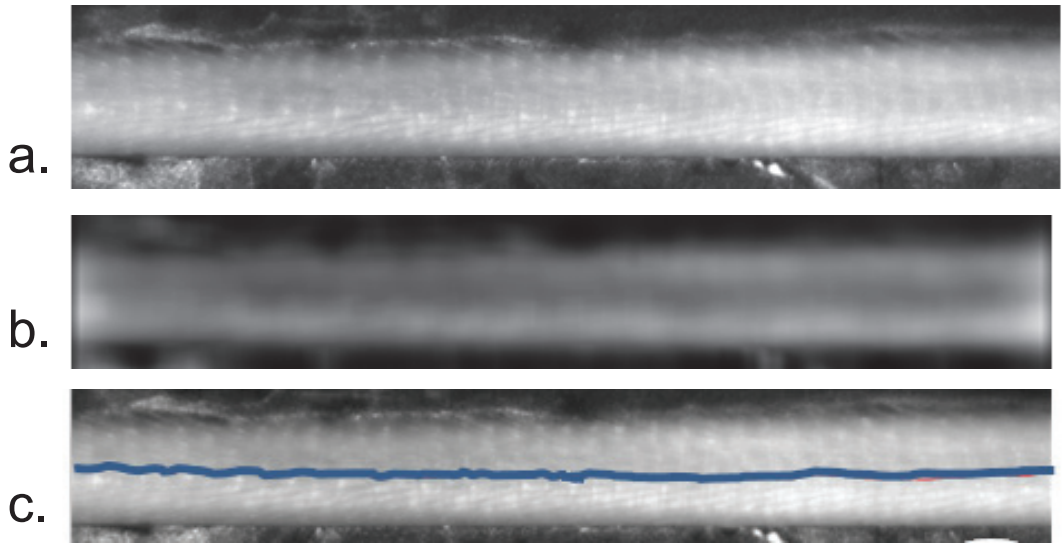


Figure 2.10: Processed tail images with the vein location identified. *a.* is the NIR image of the tail, *b.* is the bandpass filtered image, and *c.* is the overlay of the vein location onto the NIR image

image. This low pass filter, also known as a "blurring" or "smoothing" filter, averages out rapid changes in intensity, which suppresses noise. Low pass filters make gradual changes more noticeable. The effect of applying both a high pass and low pass filter to the tail image is a bandpass filtered image.

$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{2.2}$$

Once the tail image is processed, the vein location can be extracted using edge detection methods to locate the edges of the tail and vessel. For this, we use the zero-crossing edge detection method, which is a second order derivative. By applying a second derivative to a cross section of the tail image, we can detect the edges of the tail and vein at each zero crossing point [34]. When the vein edges are identified, the center point between the two vessel edges becomes our targeted insertion point. The location of the center of the vessel is overlaid onto the tail image.

We have measured the average vessel size in our images to be approximately 600 μm.

This is slightly larger than what we expect, but we believe this increase in apparent vessel size is from the diffusion of light. We have also validated that the vessel on our image is a true vessel by injecting saline into the tail vein. We observed a change in contrast in our image from the saline.

2.4 Needles for the VAS

Choosing the optimal size and material composition of a needle is important for the VAS. Microneedles are currently fabricated out of polymers [38], silicon [28], and stainless steel [36]. The VAS requires material that is robust and that will not fracture during insertion. At the Crump Institute Preclinical Imaging Center, a stainless steel 28 G needle with an O.D. of 356 μm is used for manual tail vein injections. This is larger than the estimated diameter of the tail vein (300μm). Decreasing the size of the needle used by the VAS will aid in reducing dead space for injections or blood collection, minimize vessel damage from the insertion of a needle, and allow for more tolerance in the placement of the needle in to the vessel.

One limit on the reduction of needle size is blood cell size. Red blood cells range from 6-8 μm and white blood cells typically range from 6-30 μm [45]. It is important not to damage the blood cells during collection as blood time activity curves (TAC) used for tracer kinetic modeling for PET ideally looks at free tracers in the plasma. Tracer in the plasma has the opportunity to leave the intravascular space directly and go into tissue, unlike tracer in the blood cell, which first must exit the cell and enter the plasma to leave the vascular space [39]. When red blood cells are punctured, they can spill into the plasma. This can lead to inaccuracies with analysis.

Beveled stainless steel 35 G and 34 G microneedles manufactured by World Precision Instruments were tested for the lower limits on needle size for the VAS. The 35 G needle has a 55 μm I.D. and a 135 μm O.D. with a 25° bevel while the 34 G has an 85 μm I.D. and a 185 μm O.D. with a 25° bevel. Relying on the capillary effect, blood from a rat tail was collected with both needles. The blood samples were stained with a Giemsa stain and

analyzed under the microscope. As seen in Figure 2.11, no damage was done to the red blood cells when using the 34 G needle, but the red blood cells were damaged by the 35 G needle. This indicates that for the VAS, a 34 G or larger needle should be used.

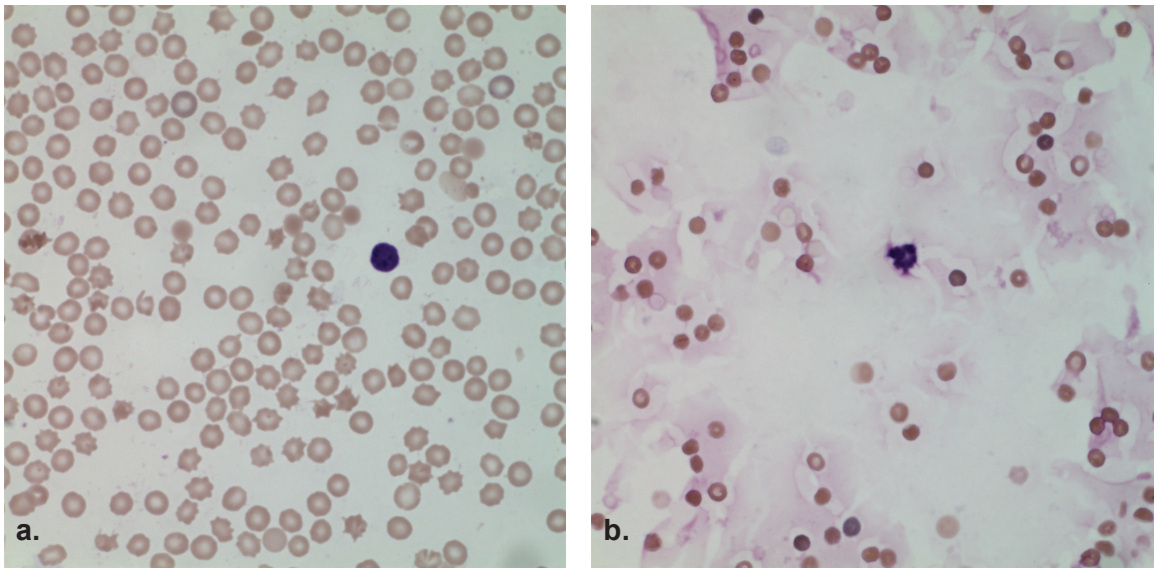


Figure 2.11: Blood smear using a Giemsa stain. Erythrocytes (RBC) are pink and leukocytes nuclear chromatin are magenta. In image *a*, blood was collected with a 35 G needle using the capillary effect. The star shaped RBC are damaged. In image *b*, blood was collected with a 34 G needle using the capillary effect. No damaged RBC are observed.

The insertion of a needle by the VAS must be monitored by a pressure transducer attached to the needle. It was verified that the pressure inside the lumen of a 34 G needle could be monitored during a needle insertion into a mouse, as seen in Figure 2.12. Unfortunately, it was later confirmed that the pressure monitoring with a 34 G needle was unreliable for unknown reasons. This is further discussed in Section 4.1.3. In addition to the pressure reading issues with a 34 G needle, these needles are costly. The cost of $45 per needle makes it impractical to dispose of the 34 G needle after a single use, which creates the need to safely and easily sterilize the needles. To overcome this issue, and more importantly, to have a reliable pressure reading from the needle during insertion, a 30 G needle was chosen to be used with the VAS. The 30 G needle, with an I.D of 140μm and O.D. of 305μm, is smaller than the standard 28 G needle used in the Crump Institute Preclinical Imaging Center. The

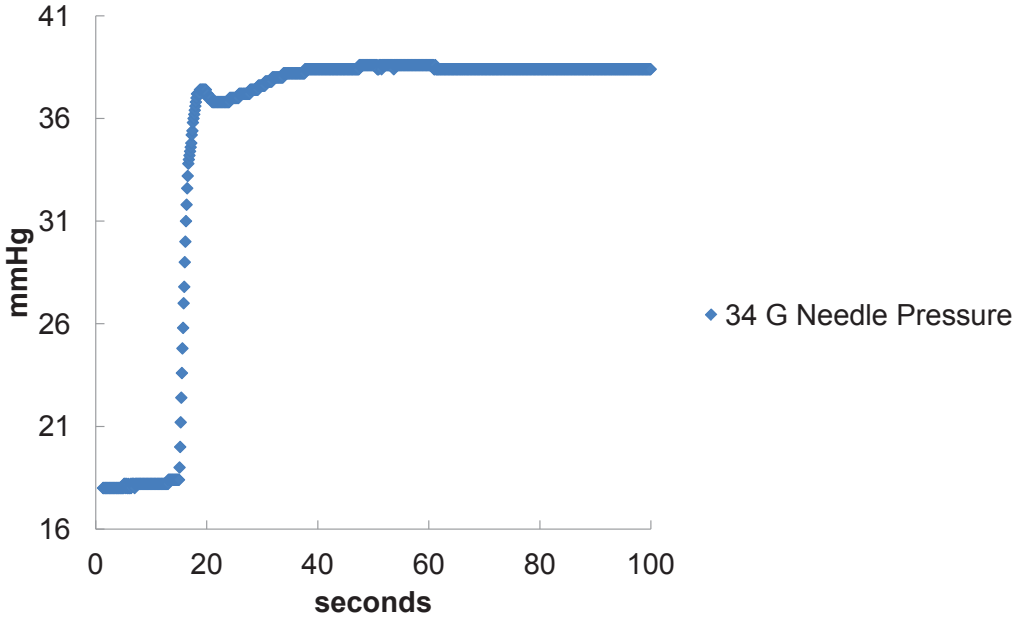


Figure 2.12: Pressure reading from a 34G needle manually inserted into a mouse tail vein

beveled 30 G needle is disposable and costs about $0.08.

2.5 Mechanical Alignment and Movement of the Needle

To insert the needle into the tail vein, the needle movement has four degrees of freedom: x, y, z, and θ, as pictured in Figure 2.13 and Figure 2.14. The accuracy of needle placement in the y direction is the most important as it aligns the needle to the correct height, relative to the vein. Since the VAS uses a needle with a diameter similar to that of the vein, it is crucial to align the center of the needle with the center of the vein, in the y direction. Z moves the needle towards or away from the plane containing the vein. When the tail of the mouse is bent and the target entry point of the needle into the tail is along the bend of the tail, the z movement places the needle at the proper location along the curve. If the needle is placed tangent (or close to tangent) to the curved surface of the tail, the needle will never reach the vein as the needle will be too shallow. If needed, θ changes the angle of insertion of the needle into the tail. Once the needle is properly aligned to the tail blood vessel, the needle moves in the x direction (axial relative the tail) into the tail tissue and into the blood

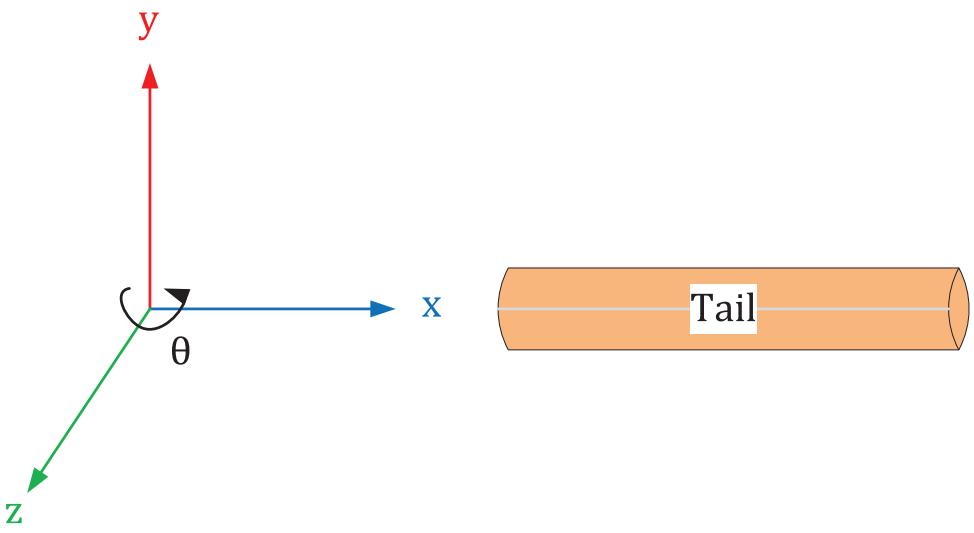


Figure 2.13: Degrees of freedom for the movement of a needle with the VAS

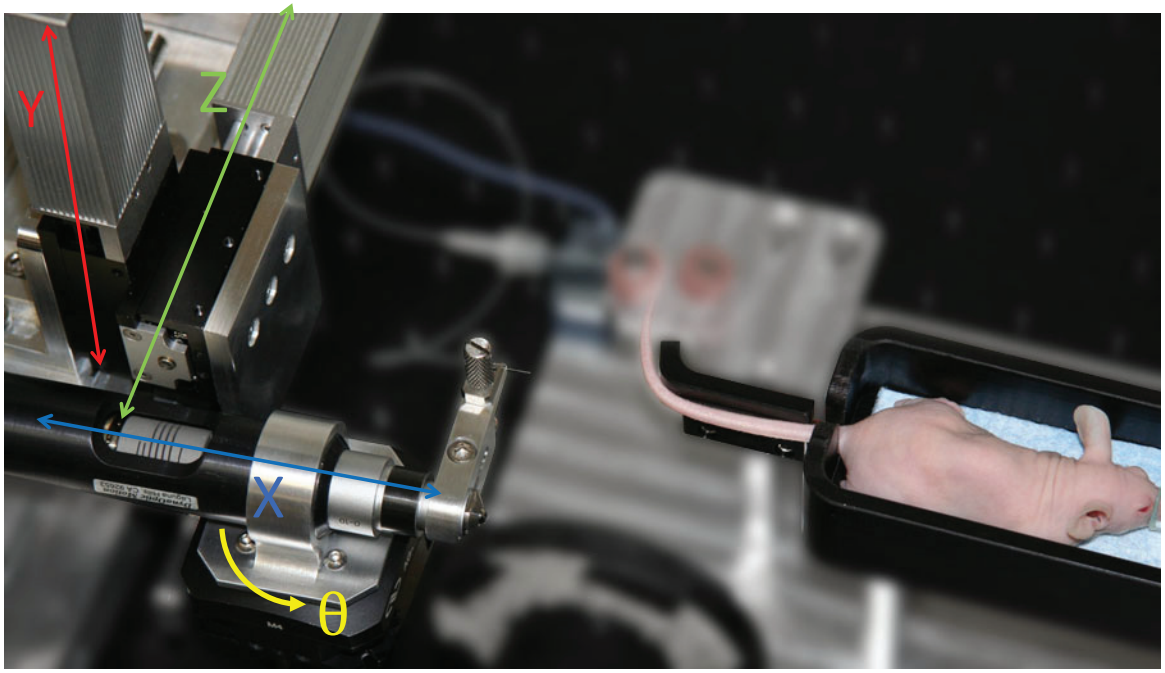


Figure 2.14: Photograph showing the different degrees of freedom for the movement of the VAS needle relative to the tail

vessel. The distance the needle needs to travel in the x direction is unknown and controlled by a pressure-based feedback system.

To move the VAS needle semi-automatically with four degrees of freedom, four different motors are used, one for each degree of freedom. The motors used to move the needle in y and z are Siskiyou encoded motors. The Drive-Master CTC-165-1 motor moves the needle in the x direction. All three of these motors have specialized controls designed and built for the VAS. The control system is discussed further in Section 3.2.2. The motor used for θ is an off-the-shelf rotational table with its own controller (Newport CONEX-AG-PR100P).

2.6 Pressure-Based Feedback System

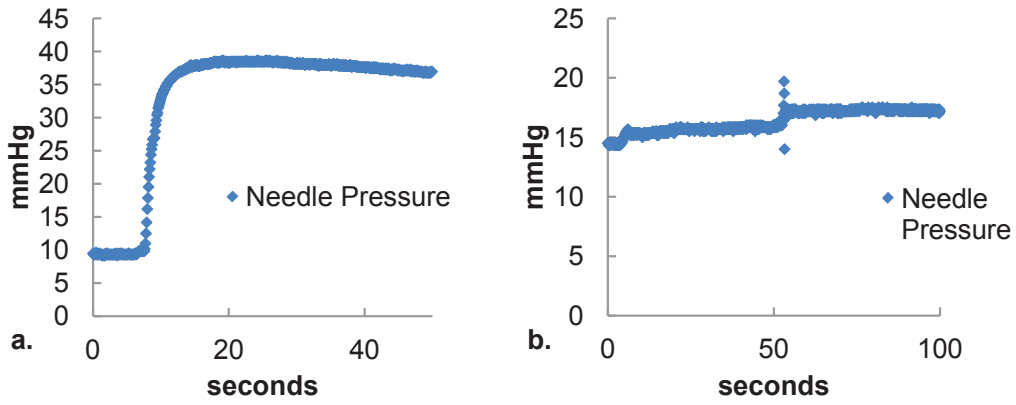


Figure 2.15: Good and bad pressure readings. *a.* is an example of a pressure reading from a successful needle insertion into a mouse tail. *b.* is an example of a pressure reading from an unsuccessful attempt a inserting a needle into the mouse tail vein

Because the exact depth of the vein is unknown, a feedback system is required to verify that the VAS needle has entered the blood vessel. Since the needle will be traveling through tissue and then entering a blood vessel that is pressurized, it was decided to create a feedback system based on pressure. To do this, a VAS needle is attached to a pressure transducer via tubing. The pressure sensor has piezoresistive strain gauges diffused onto a micro-machined silicon diaphragm. The piezoresistive material changes electrical resistance when it is compressed or strained. The resistors are connected as a Wheatstone Bridge, which allows the unknown resistance to be measured with high precision and, in turn, allows us to

monitor a change in pressure.

It is hazardous to inject air into an animal [32]. For this reason the needle, tubing, and pressure transducer are filled with saline and sealed. The only opening on the sealed system is the needle lumen. The base pressure of the pressure feedback system is the vapor pressure at room temperature of saline. As the needle enters the tail tissue, no change of pressure is recorded. When the needle enters the blood vessel, a rapid change in pressure is observed. This rapid change in pressure signals the motor to stop moving as the needle has reached the desired location. The observed change in pressure is due to the blood pressure of the mouse. Because many environmental factors can affect the blood pressure, it is not desired to set an absolute threshold to trigger the motor stopping. Instead, a relative change in pressure is monitored so that when the pressure system measures a predefined change in pressure in a set time period, the feedback system stops the motor. Because the pressure system is monitoring the pressure inside the needle lumen, the pressure reading should remain elevated and level off after the motor stops. If the needle is no longer inside the vein when the needle stops, the pressure reading will decrease. This is seen in Figure 2.15. The pressure feedback system not only signals when to stop the motor but can also be used to predict whether the VAS will perform a good injection or a poor injection. When the pressure is indicating that the needle is no longer inside the blood vessel and that the injection will be poor, the injection can be canceled or the needle can be repositioned.

2.7 Injection

After the needle is properly inserted into the blood vessel, the last step is to inject the desired probe through the needle and into the mouse. It is important not to inject the probe too fast which could damage the vessel. It is also advantageous for the injected volume to be small to further reduce the likelihood of damaging the vessel, but this requires a small dead volume between the needle and probe source.

The injection can be performed in many different ways. The probe can be injected manually with a syringe, a syringe pump can be employed, a microfluidic chip or a liquid

Figure 2.16: Syringe pump

handling system could be used. The only requirement is that the injection device needs to be connected to the non-beveled end of the needle with minimal dead volume and without hindering the insertion of the needle into the tail. For the current design of the VAS, a syringe pump pictured in Figure 2.16 (Syringepump.com NE-1000) can be used for the injection. Different injection rates and volumes can be programmed for the injection.

CHAPTER 3

System Design

3.1 Mechanical Design

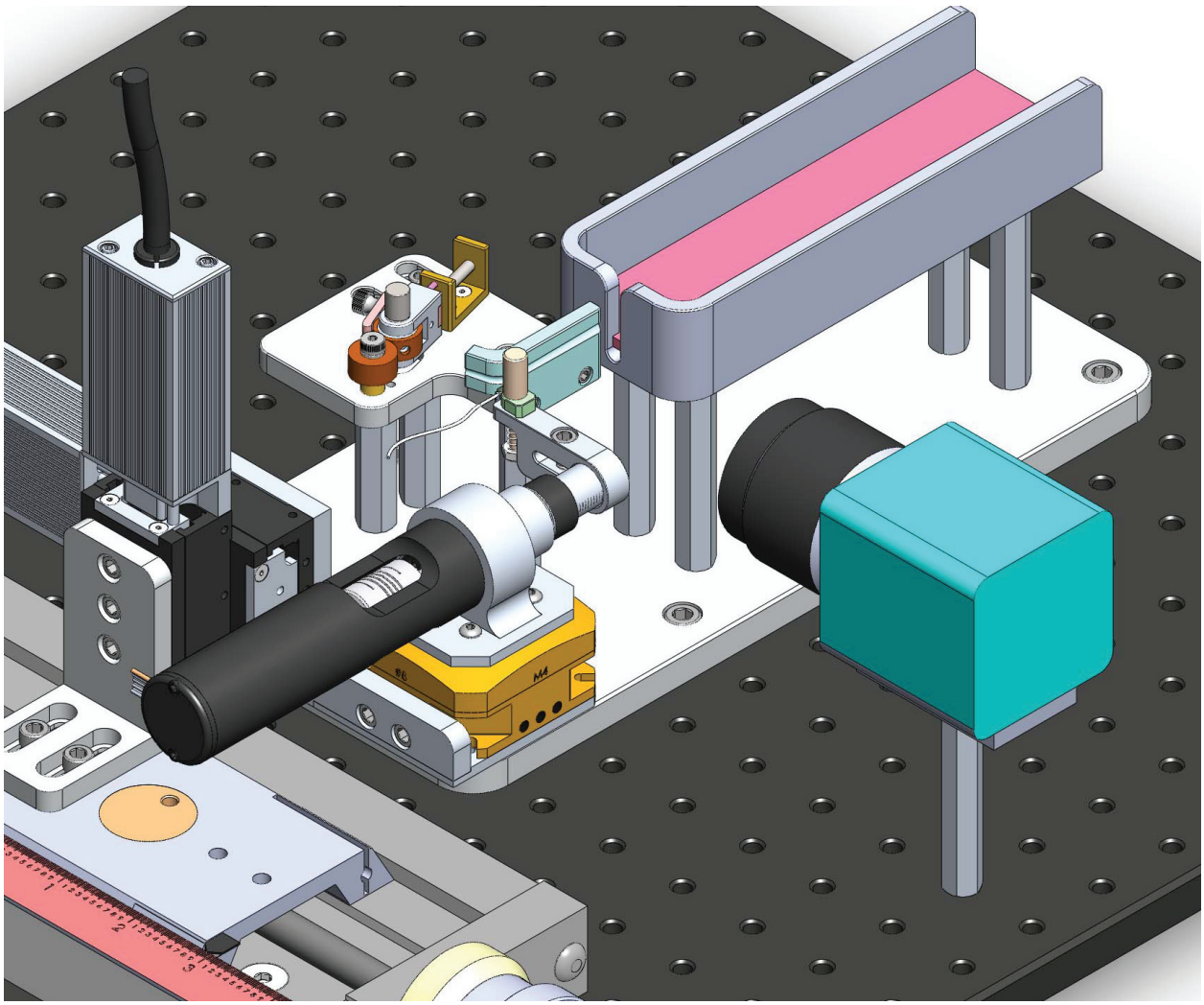


Figure 3.1: VAS mechanical model

Many components make up the complete VAS including cameras and light sources to

image the tail vein, holders to secure the mouse and tail, needles, needle holders to attach the needle to motors that move the needle, a pressure transducer that monitors the insertion of the needle, a syringe pump that controls the injection of probes into the tail vein, and a computer to control the electronic components. In this section, we will discuss some of the custom made mechanical components for the VAS including: the mouse holder, tail support, tail clamp, and the needle holder as seen in Figure 3.1. Below are the high level design requirements for these different mechanical components that were designed and built specifically for the VAS. The final version of the VAS was built with particular objectives, design requirements, and knowledge from previous trials and errors guiding the process. These requirements drove the design of the final VAS prototype for this dissertation.

3.1.1 Mouse Holder

The main objective of the mouse holder is to hold an anesthetized mouse. Other considerations that went into the design include:

- One must be able to routinely sterilize the holder by cleaning it with a solvent.
- The holder must be heated to maintain the temperature of a mouse at 38°C.
- It must be wide enough and long enough to accommodate a mouse and a nose cone.
- The holder should have sides to prevent the mouse from falling off of the platform.
- One end must have a "rump stop" on it. This is used to pull the mouse body against it to allow us to pull the tail taut. This stop needs to have an opening for the tail, but it should not be too wide to allow part of the body of the mouse through. The diameter of the tail near the mouse body is about 3.5mm so the opening can be about 4.5mm.
- The height of the mouse holder should be dependent on the height of the needle when it is attached to the different stages.

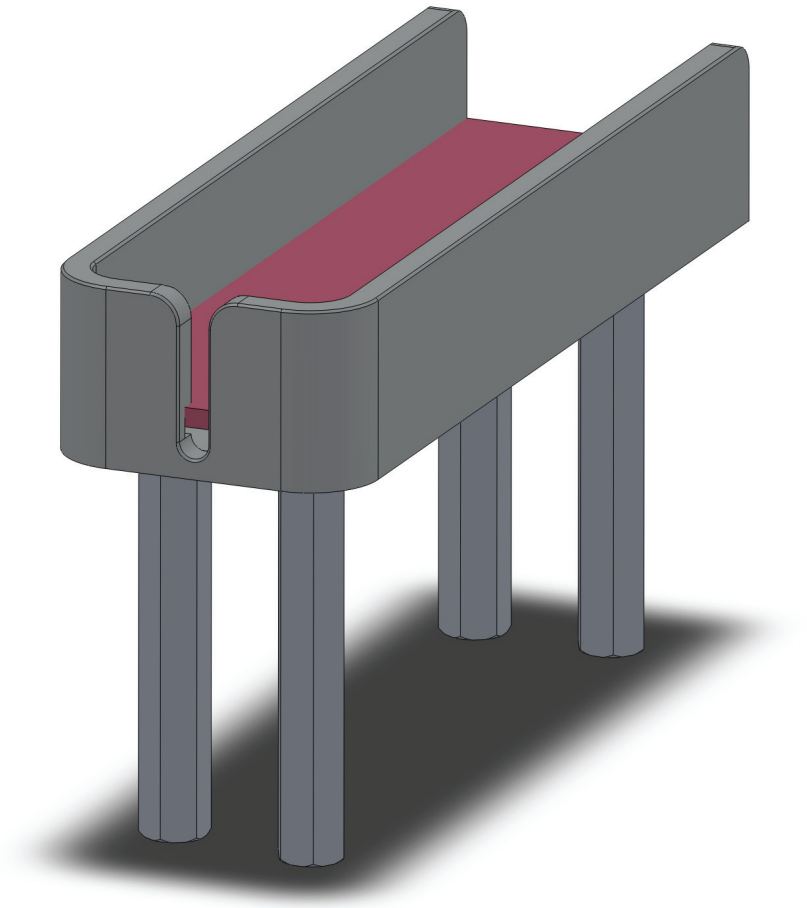


Figure 3.2: Mouse holder model. The pink area is the mouse heating plate

The mouse holder was made out of black delrin. A plastic was desired to insulate the heater on the mouse holder from the aluminum stands and base plate connected to the holder. Delrin was chosen since it is easy to machine. A black CPVC (chlorinated polyvinyl chloride, another plastic) insert was added to the mouse holder to position the mouse at the correct height. The thickness of the insert was too thin to easily machine from delrin. The insert is the actual piece that the mouse rests on. The insert was needed to reduce the thickness of plastic that needed to be heated, but still let us use off-the-shelf stands for the support of the mouse holder and its connection to the base plate. The insert is colored pink in Figure 3.2. A heating strip was placed on the bottom face of the insert to warm the mouse.

3.1.2 Tail Support

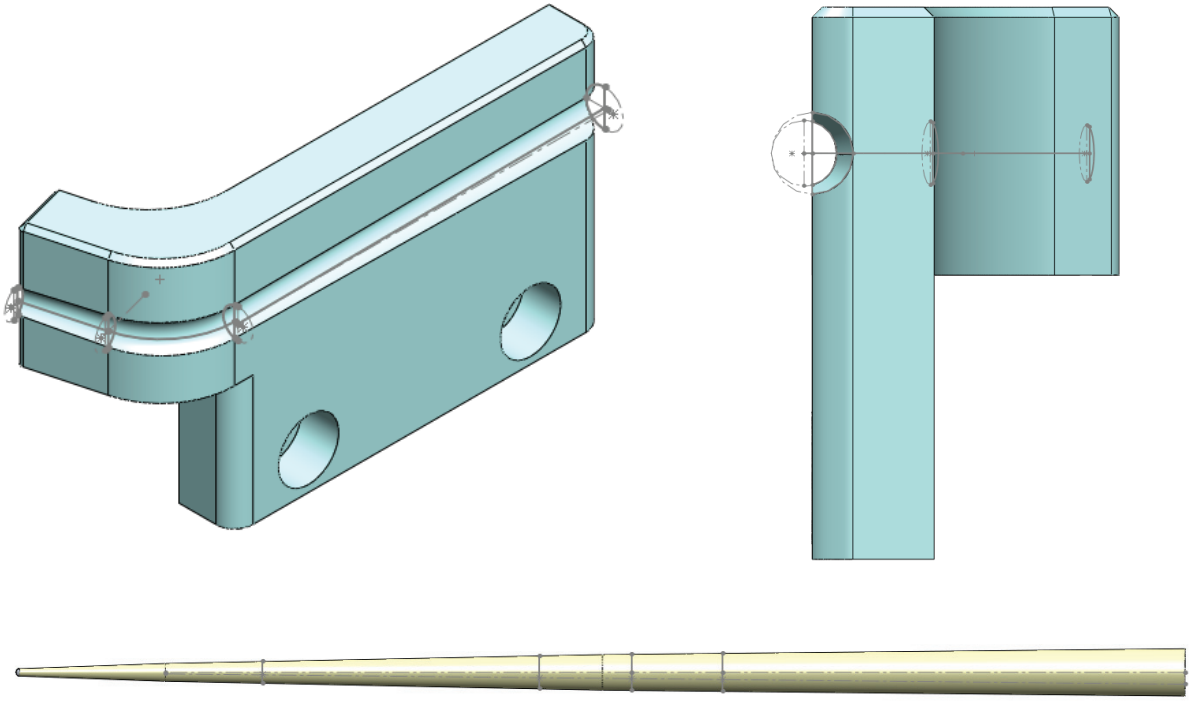


Figure 3.3: Tail support model. The top two images are different views of the tail support. The left image shows how the tail support changes in size to match the diameter of an average mouse tail. The right image shows how the tail cut is angled in to keep the vein colinear with the needle. The tail profile is the bottom image.

The main purpose of the tail support is to support the tail so that when a needle is being inserted, the tail remains rigid and tail movement is as minimal as possible. The tail support also bends the tail so that the needle can be inserted at the bend location, collinear to the vein. Other considerations that went into the design include:

- It needs to provide access to one tail vein.
- It should support at least the first 3-4 cm of the tail (before the bend). It might also

be beneficial to support the tail after the bend up to the tail clamp.

- The tail support needs to heat the tail at 38°C.
- One must be able to sterilize the support with solvent.
- The support must not damage the tail or restrict blood flow.
- The support must not restrict the visualization of one side of the tail (especially at the bend location).
- The support must not interfere with the needle movement.
- Preliminary estimates for the curvature of the tail at the bend location is a radius of 0.25 inches.
- The tail support should be black to reduce glare from light.
- Tail placement should be easy and require very little time.
- The tail support should accommodate different sized tails.
- The support should keep the tail level. The tail diameter shrinks concentrically from the body, but the support should keep the tail vein as parallel to the mouse body support as possible.

As the tail diameter decreases, we assume that the distance from the surface of the tail to the vein remains constant. Therefore, the depth of the cut out for the tail on the tail support is greatest at the base of the mouse and the cut becomes more shallow as the tail dimension decreases. This can be seen in Figure 3.3. For the cut on the tail support, we built and tested a "V-groove" and "semi-circle" design. We found that the semi-circle reduced the movement of the tail that occurs during needle insertion more than the V-groove design. We believe that this is because the tail is nonrigid body, which allow it to deform to match the shape of the semi-circle groove and increase the surface contact area between the holder and the tail more so than with the V-groove.

The tail support was machined out of black delrin to reduce the transfer of heat from the tail support heater to that of the rest of the system. The tail cut out section was roughened with sandpaper to increase the friction between the tail support and the tail.

3.1.3 Tail Clamp

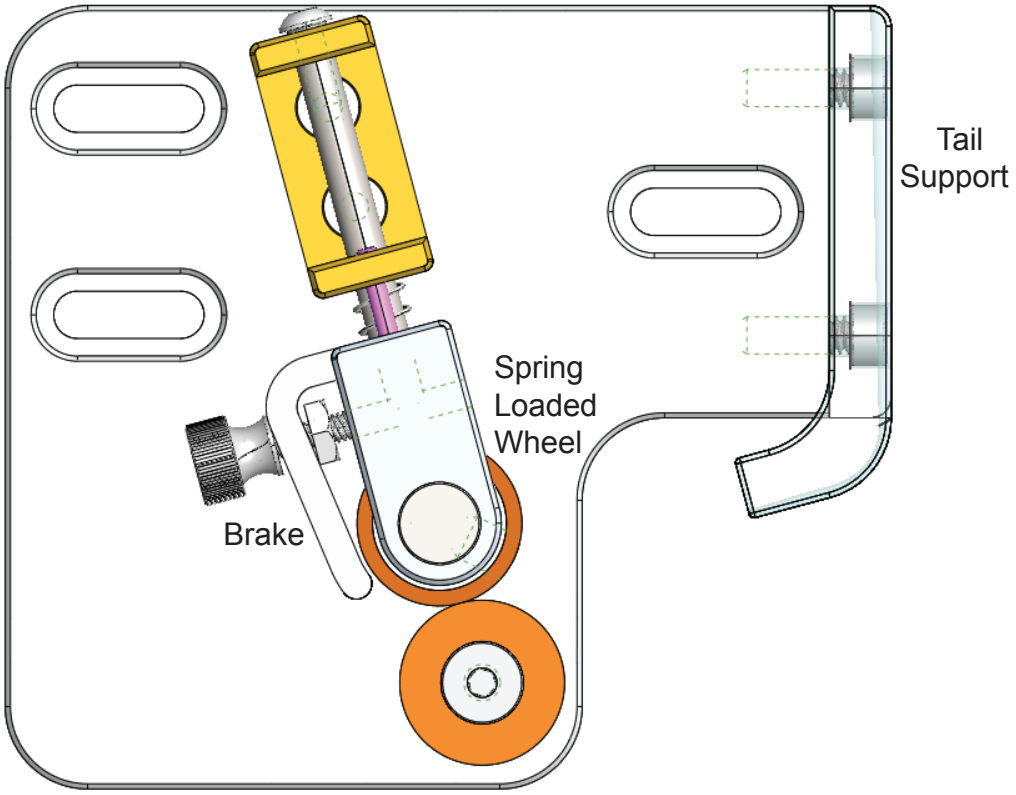


Figure 3.4: Top view of the tail clamp model

The accuracy of the VAS decreases if the tail moves during the insertion of the needle. Movement is caused from the force of the needle insertion. To prevent tail motion, it is important to keep the tail taut. This is done with the VAS tail clamp. The main objective of the tail clamp is to pull and maintain tension in the tail and reduce tail movement. Other design considerations include:

- One must be able to sterilize the tail clamp.
- The tail clamp must not damage the tail or restrict blood flow.
- The tail clamp should secure the tail towards the tip of the tail.
- The tail clamp must not obstruct the path of the needle.
- The tail clamp should keep the tail level.
- The tail clamp should use a spring loaded wheel system with at least one wheel having a thumb screw to easily pull tension on the tail.
- The tail clamp should be easy to use and require minimal time to secure the tail.

As seen in Figure 3.4, the tail clamp consists of two wheels. One wheel is free to rotate around its axis, but is otherwise fixed. The second wheel rotates around its axis but is also spring loaded to linearly translate along an axis. The spring loaded wheel design was modeled after spring loaded gate casters. The spring loaded wheel has a thumb screw, which allows the tail to be easily pulled into the spring loaded clamp. A brake is attached to the spring loaded wheel to prevent movement once the desired tension in the tail is achieved.

3.1.4 Needle Holder

The main objective of the needle holder is to secure a needle onto a linear actuator so that is can be inserted in the mouse tail vein. Other design considerations include:

- It must be user friendly and safe to use.
- It must be easy to change out the needle.
- It must accommodate a 30G needle.
- It needs to hold the needle parallel to the base plate and co-linear to the vein.
- One must be able to add the needle with the correct bevel orientation.

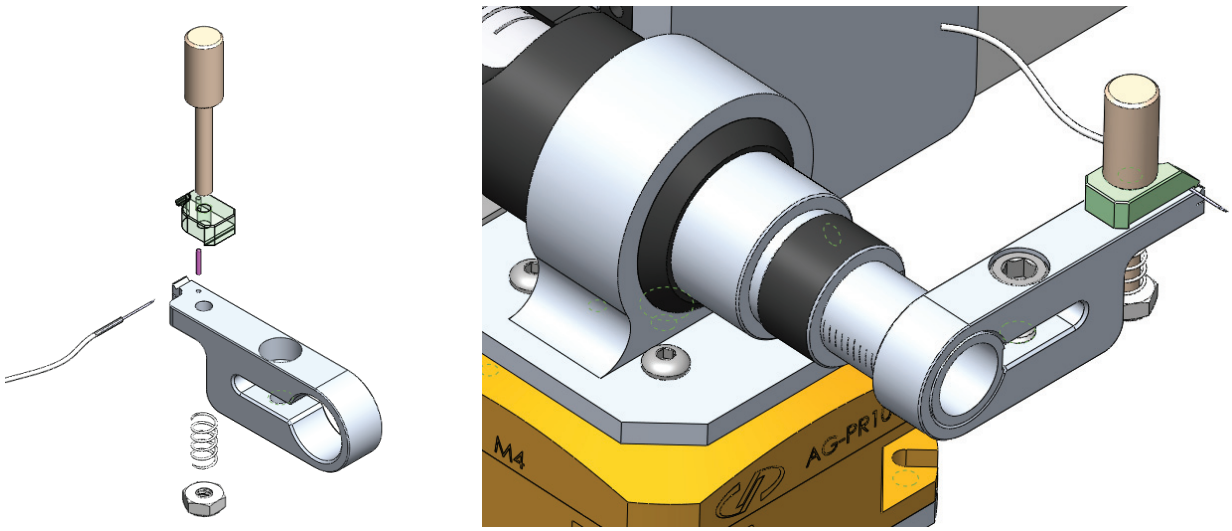


Figure 3.5: Needle holder model. The left image displays the different parts of the needle holder, including the spring loaded handle for the v-chuck. The right image shows the needle holder attached to the linear actuator.

- The bulk around the needle must be minimized so that the holder will not touch the tail during the insertion.
- The needle needs to move with each motor.
- The needle needs to be attached to the pressure transducer and the radioactive dose. The needle holder must accomodate this.
- The needle must be secure and not move unintentionally as it is inserted in the tail vein.

A v-groove is notched out of the needle holder and is used to properly align the needle. A v-chuck, in green in Figure 3.5, clamps down on the needle. A dowel, in pink in Figure 3.5, properly aligns the v-chuck clamp. The clamp is secured with a spring loaded handle or a screw. The needle holder is attached to the linear actuator via a slip fit.

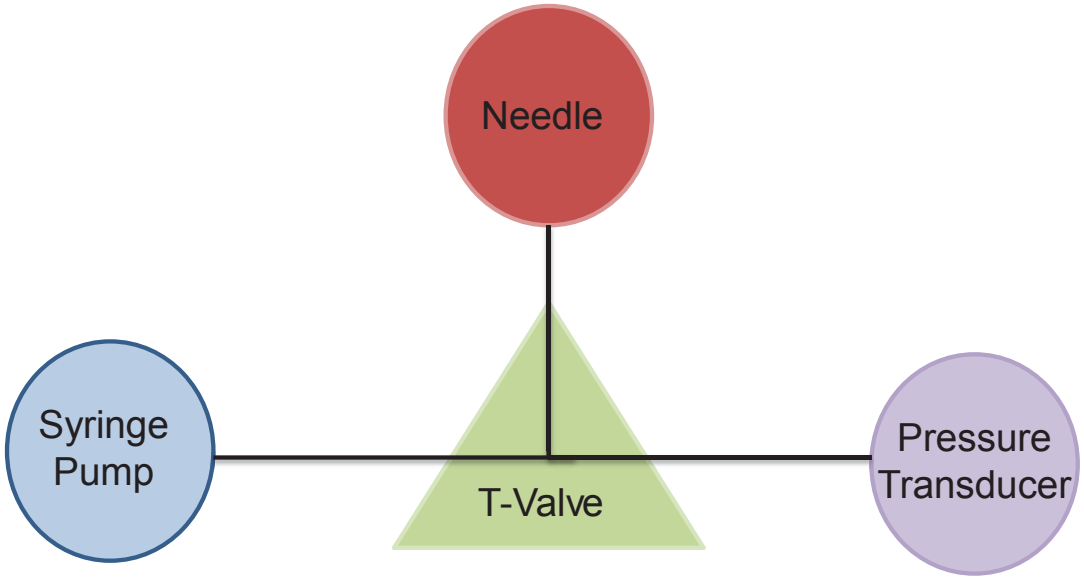


Figure 3.6: T-Valve connection schematic

3.1.5 Needle connection to the pressure transducer and injection probe

The needle needs to be connected to a pressure transducer to provide feedback on when the needle reaches the vein. The needle also needs to be connected to a syringe pump that will inject the desired probe through the mechanically placed needle inside the tail vein. This needs to be done with as minimal dead-volume as possible.

The hubless 30 G beveled needle that we use for our final VAS design is connected to PTFE 30 AWG thin wall tubing (Cole Palmer 06417) by simply slipping the tubing over the non-bevel end of the needle. This connection is not air tight, so this junction is sealed with super glue. The other end of the tubing is connected to a 4-port T manual switching valve (IDEX V-101T). This valve has a dead volume of 9.7μL between 3 ports. Other ports are connected to the pressure transducer and the syringe containing the injection probe. By using this manual valve, we connect the needle to a pressure transducer and the injection probe while minimizing the dead volume. By using the T valve, we can monitor the pressure while we inject a probe.

3.2 Electronics Design

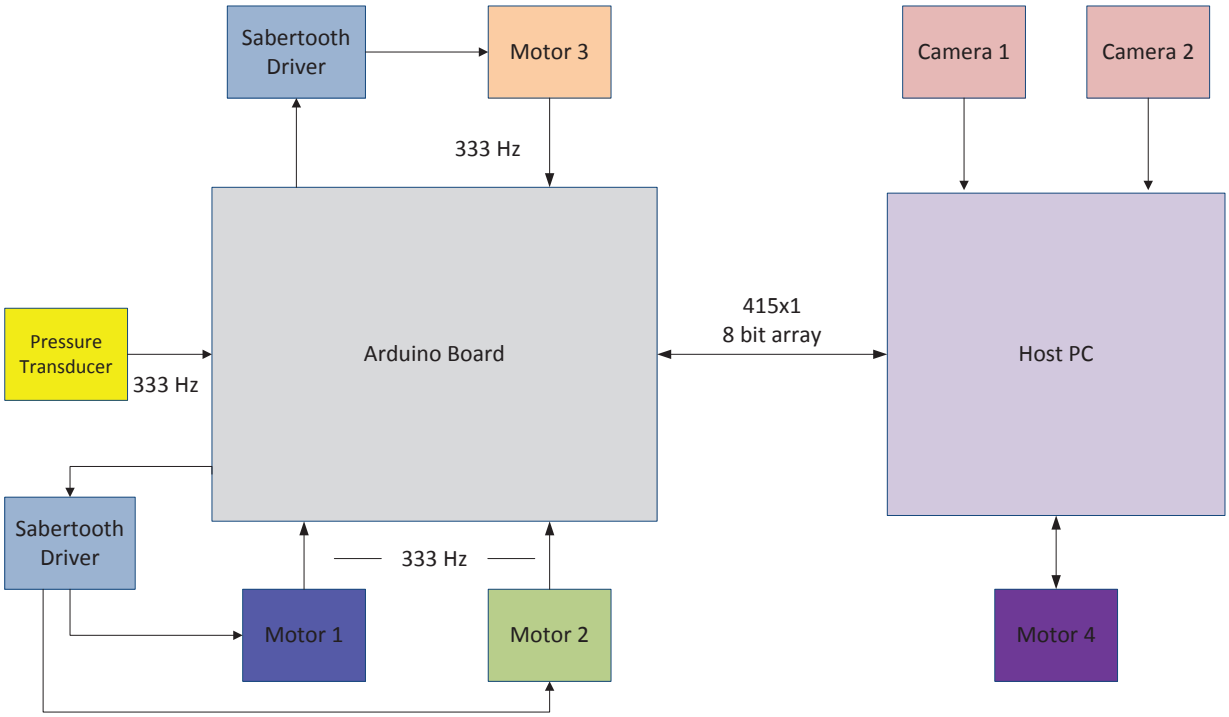


Figure 3.7: VAS main electronics schematic

There are many different electronic components that contribute to the workings of the VAS. The top-level VAS electronics schematic is outlined in Figure 3.7. The VAS is controlled by a Labview based interface on the host computer. Images are processed and the tail vein location is identified directly on the host computer. The host computer also communicates with the Arduino board, which is a microcontroller that interfaces with the pressure transducer and 3 of the 4 motors. This section will go into more details pertaining to the different electronic components of the VAS including the Arduino Mega 2560, motor controllers, the pressure transducer electronics, mouse body and tail heaters, and the VAS electronics box that houses the majority of the VAS electronics.

3.2.1 Arduino Mega 2560

The Arduino Mega 2560 is a microcontroller board that uses the Atmel AVR 8-bit microcontroller. It has 54 I/O pins, of which 15 provide PWM outputs. It has 16 analog input pins and a 3.3V supply pin, which powers the LED ring for the VAS. The Arduino can be programmed in C by using the Arduino software. The Arduino integrates the VAS motors with the pressure transducer signal. To minimize delays, the positions of the motors and the pressure transducer signal are monitored on the Arduino board, not the host computer. The trigger to stop the motor based on a change in the pressure signal also occurs on the Arduino board. The analog signal from the pressure transducer circuitry is read into and digitized on the Arduino. To avoid stopping the motors based on a random change in the pressure signal and to make sure that the motor only stops when there is a sudden increase in pressure, a difference of two running averages of the pressure signal is used to trigger the stop of the motor. The Arduino stores 512 pressure transducer data points in memory and collects new data points at a rate of 333 Hz. With each new data point, the average of the 16 oldest pressure points is subtracted from the average of the 16 newest pressure points. If this difference exceeds a preselected value, a trigger occurs where the Arduino sends a signal to stop the x motor and backs up to the location where the motor was located when the trigger occurred. The code for the Arduino board can be found in Appendix B.

3.2.2 Motor Electronics

The VAS has four motors to allow the needle to move with 4 degrees of freedom. Motors 1, 2,and 3 are DC brush servo motors. Motor 1 and 2 move the needle along the 'y' and 'z' axes (refer to Figure 2.13) and are both Siskiyou encoded motors, while motor 3 moves the needle in the 'x' direction and is a Drive-Master CTC-165-1 motor. Motor 4 is a rotational table that moves the needle in 'θ' and is a piezo motor driven rotation stage (Newport CONEX-AG-PR100P). Motor 4 has an off-the-shelf controller and can be directly connected to the host computer. The control systems for motor 1,2, and 3 were created specifically for the VAS. The two main components of the motor control system for the VAS include

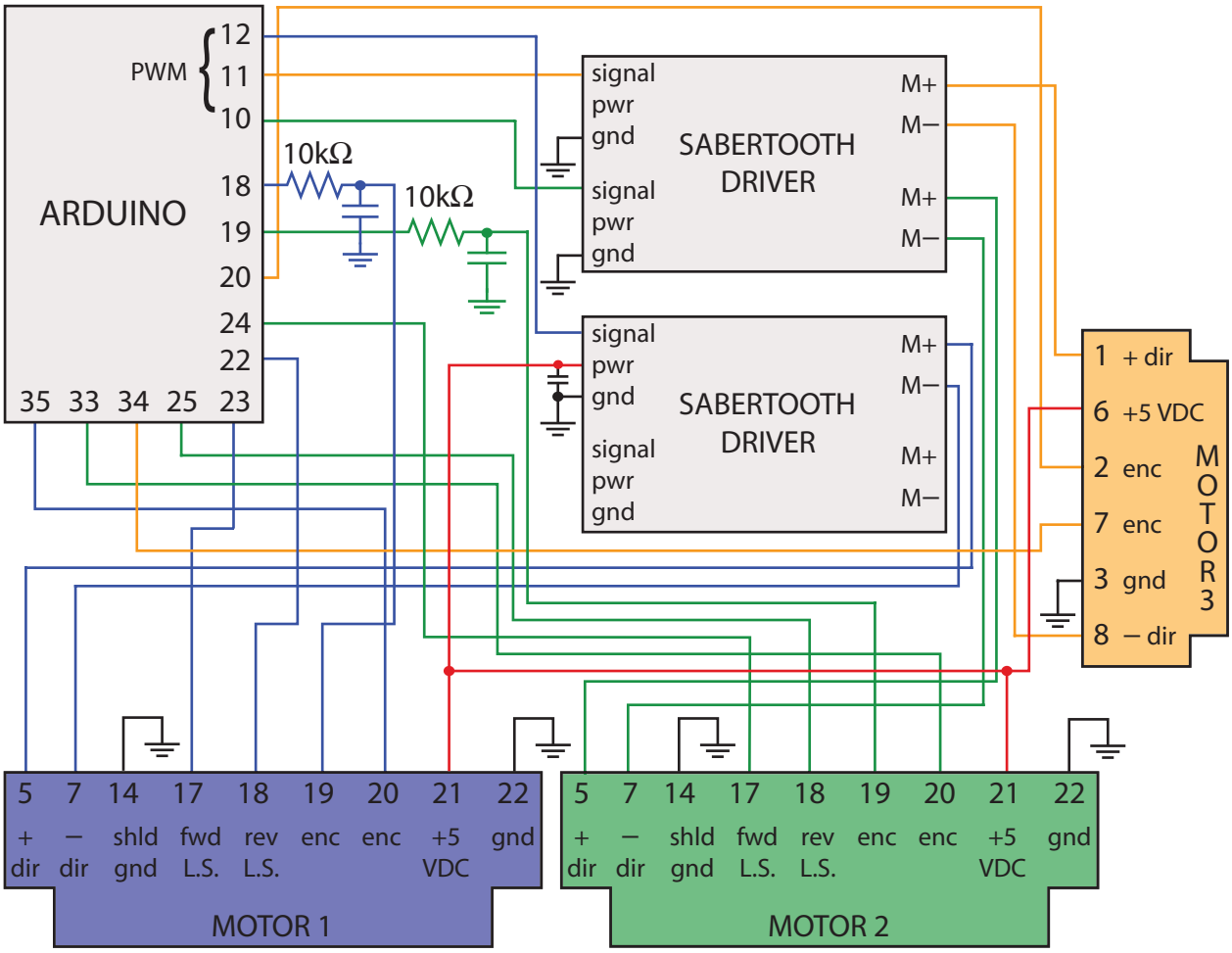


Figure 3.8: Wiring diagram for the VAS motors 1,2,and 3

an Arduino Mega 2560 board and two Dimension Engineering Sabertooth dual 12 A motor drivers. The Sabertooth drivers are the power amplifiers for the motors. They receive a pulse width modulation (PWM) signal from the Arduino board, amplify that signal, potentially changes the direction of the current to alter the direction of the motor movement, and sends the signal to the motor. PWM signals are a standard way to control servo motors, where the width of the pulse is related to the desired motor position [11]. Besides providing the PWM signal based on higher level commands from the host computer, the Arduino board monitors the motor's limit switches and encoder counts.

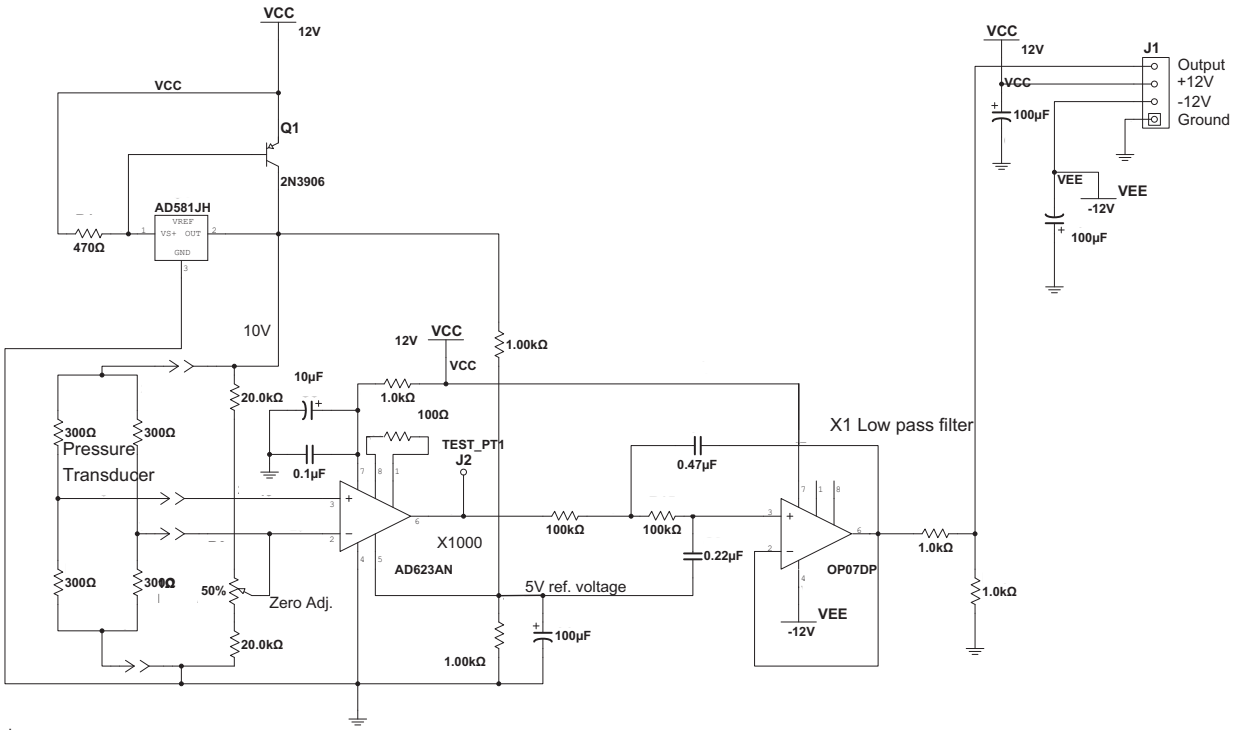


Figure 3.9: Pressure transducer amplification circuit

3.2.3 Pressure Transducer Electronics

The VAS uses the Deseret Disposable Pressure Transducer (Deseret Medical Inc. 8000) to monitor the pressure inside the lumen of a needle. This pressure sensor has piezoresistive strain gauges, so a pressure change induces a strain which causes the resistive value of the strain gauges to change. The resistors are connected as a Wheatstone Bridge. The voltage difference from the bridge is proportional to pressure [7]. Robert W. Silverman designed a circuit for the VAS to amplify the signal from the pressure transducer. It is a two stage circuit. The first stage amplifies the signal by 1000. The second stage is a low pass filter used to attenuate any high frequencies from the input signal. The pressure transducer circuit output was calibrated by using regulated nitrogen gas. A known pressure of nitrogen gas was introduced into the pressure transducer chamber and the output from the pressure transducer

circuit was recorded. This was repeated for several different pressures. This data is charted in Figure 3.10. A linear fit was applied to the data with the slope representing the change in voltage per mmHg. This was found to be 23mV/mmHg.

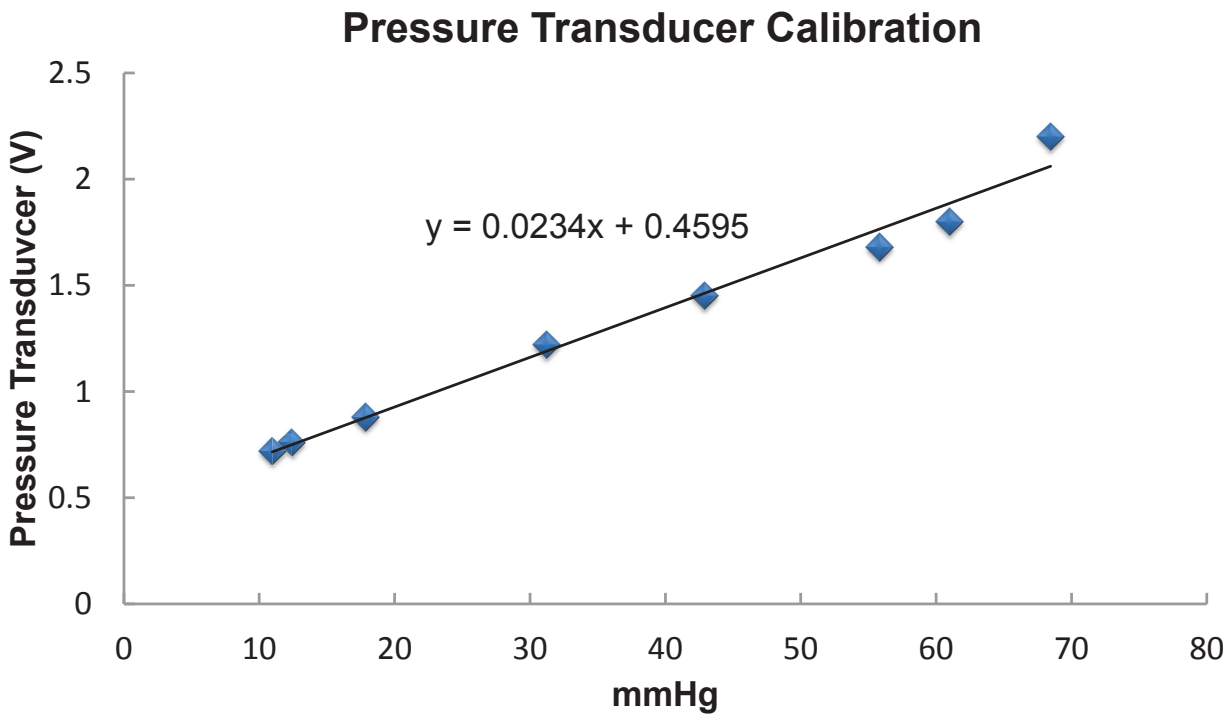


Figure 3.10: Pressure transducer calibration

3.2.4 Heating Electronics

It is important to regulate the body temperature of mice. The average core body temperature for mice is approximately 38°C [44]. The VAS includes two heaters, one for the mouse body and one for the mouse tail. Each heater is monitored and controlled independently from the other. Minco heaters and sensorless controllers were used. The flexible thermofoil heaters consist of etched-foil resistive heating elements between kapton tape. The heaterstat sensorless temperature controller is an adjustable set point solid state on-off controller. The controller checks the resistance of the heater every second. If the heater is below the setpoint value, the controller remains powered on and continually checks the resistive value of the

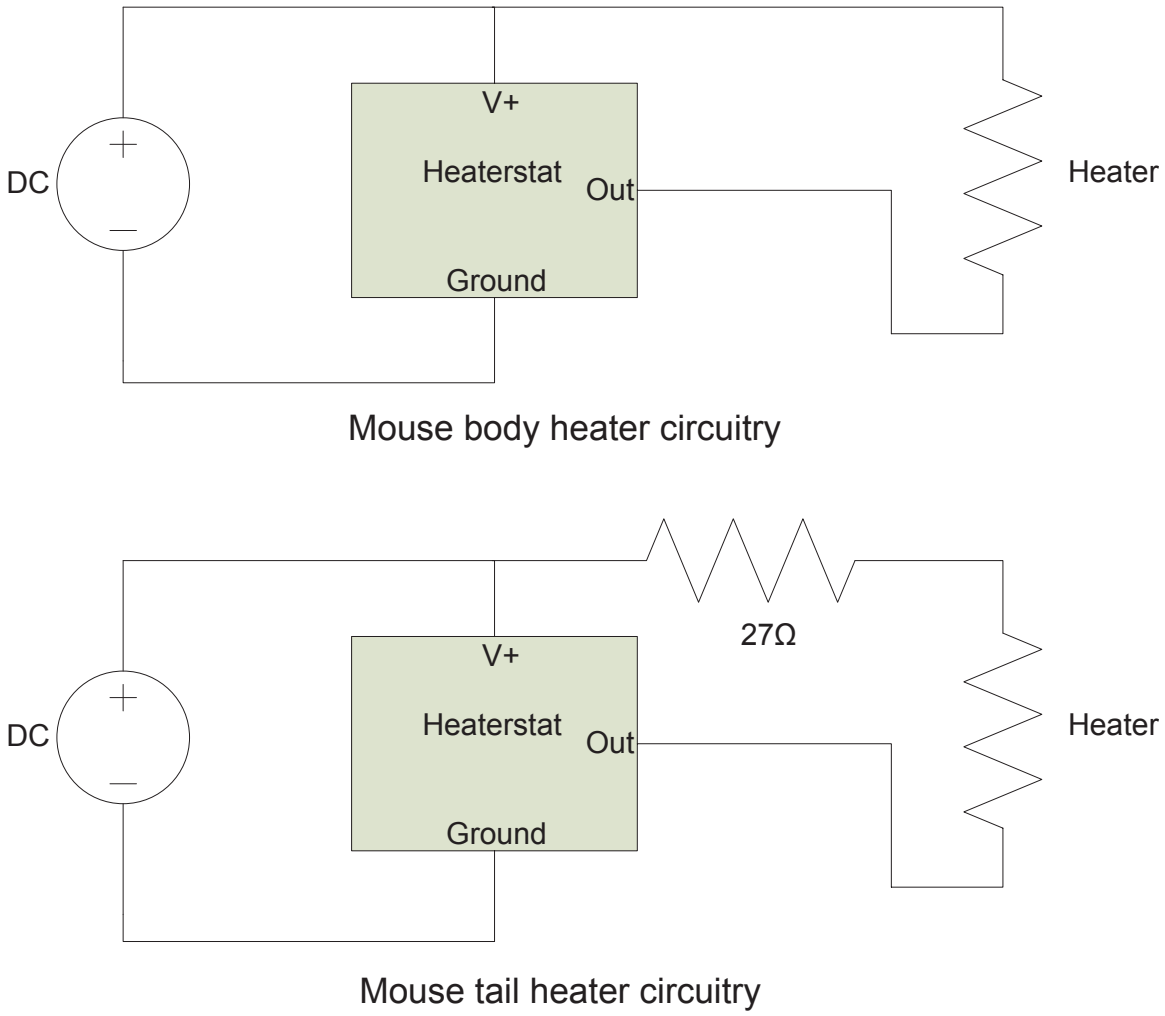


Figure 3.11: VAS heater circuitry

heater until the desired setpoint is reached. If the heater resistive value is above the setpoint, power is turned off. The mouse holder uses a 1" x 3.82" heater with a resistive value of 24.8Ω (Minco HK5329R24.8L12B) and a heaterstat sensorless DC controller (Minco CT198-1007R24.8L1). An appropriate sized heater was not available for the tail holder. A Minco heater (HK5329R24.8L12B) was cut to the appropriate size. This caused a break in the resistive element. The kapton tape was peeled back and new leads were soldered on at the break points of the resistive element. The resistive value of the modified heater was 4.2Ω. Since this heater was connected to the Minco controller CT198-1008R30.00L1 calibrated for

a 30.0Ω heater, a 27Ω value resistor was added in series with the heating element. Because the added resistor has a fixed value, this lowered the sensitivity of the heating controller. The modified heater was tested and the desired setpoint was reached.

3.2.5 VAS Electronics Box

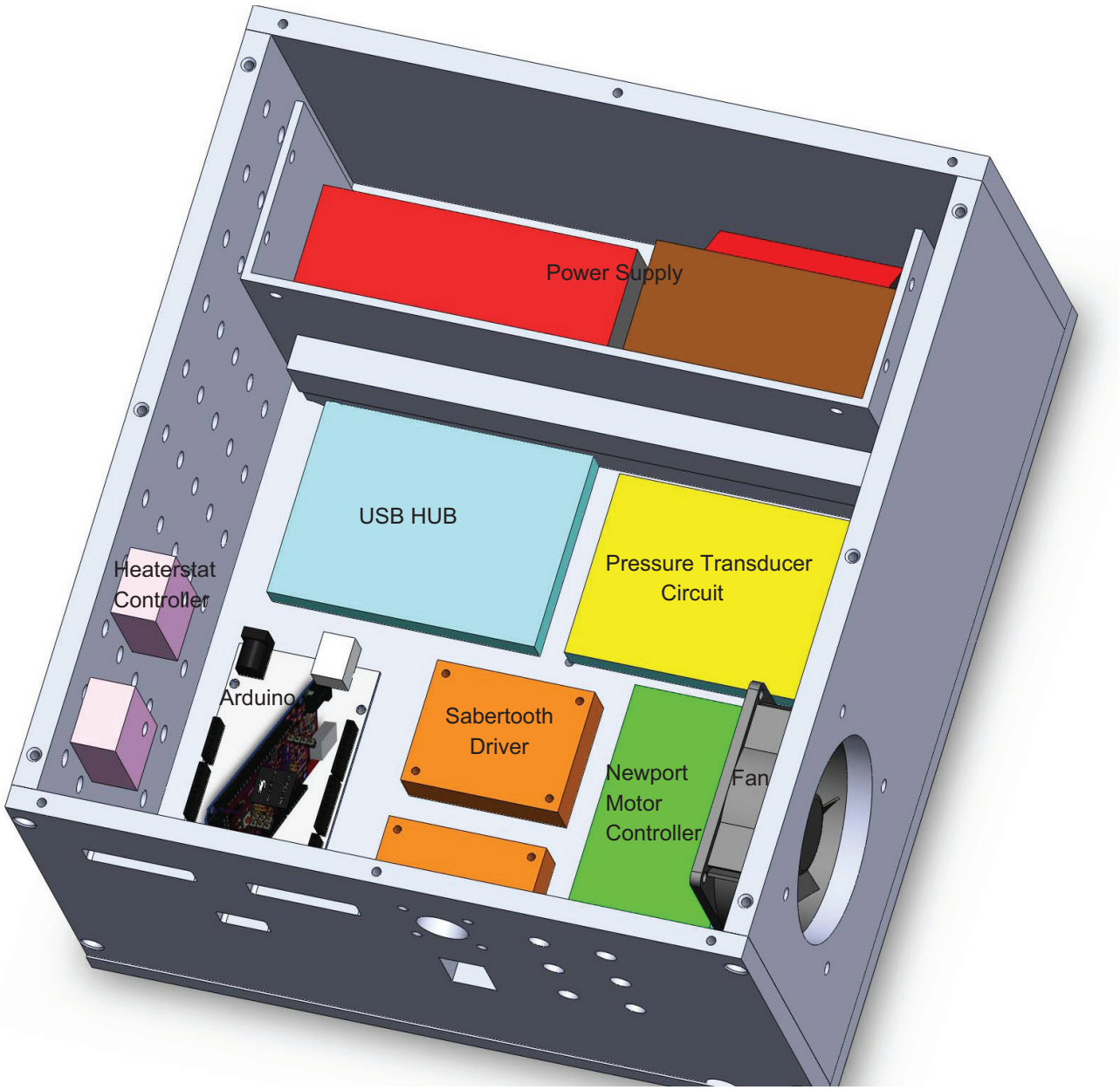


Figure 3.12: VAS electronics box

The VAS electronics box was designed and built to house the VAS electronics and reduce the number of cables going directly to the VAS laptop. Connections into the VAS electronics box include:

- Cable connectors for 3 motors
- LED light ring
- Pressure Transducer
- 2 Heating strips
- Power Cord

Connections within the box include:

- Newport CONEX motor controller to USB hub
- Sabertooth amplifier boards and motors to Arduino Mega 2560 and power
- Arduino Mega 2560 to USB hub
- Heaters to temperature controllers and power
- LED ring to power
- Pressure transducer to pressure transducer amplifier board and Arduino Mega 2560 (ADC)
- Fans to power

The power supply within the VAS electronics box is a Power-One 12V-15V linear power supply (HCC15-3-A). The only cable leaving the VAS electronics box is a USB cable that connects to the computer. The two USB cameras used with the VAS are connected directly to the laptop.

3.3 Graphical User Interface Design

Figure 3.13 is a Labview based graphical user interface (GUI) was created for the user to control the VAS. The VAS GUI displays real time images of the mouse tail with an overlay of the vein location. The GUI also plots the output of the pressure transducer and allows the user to issue commands to move the motors.

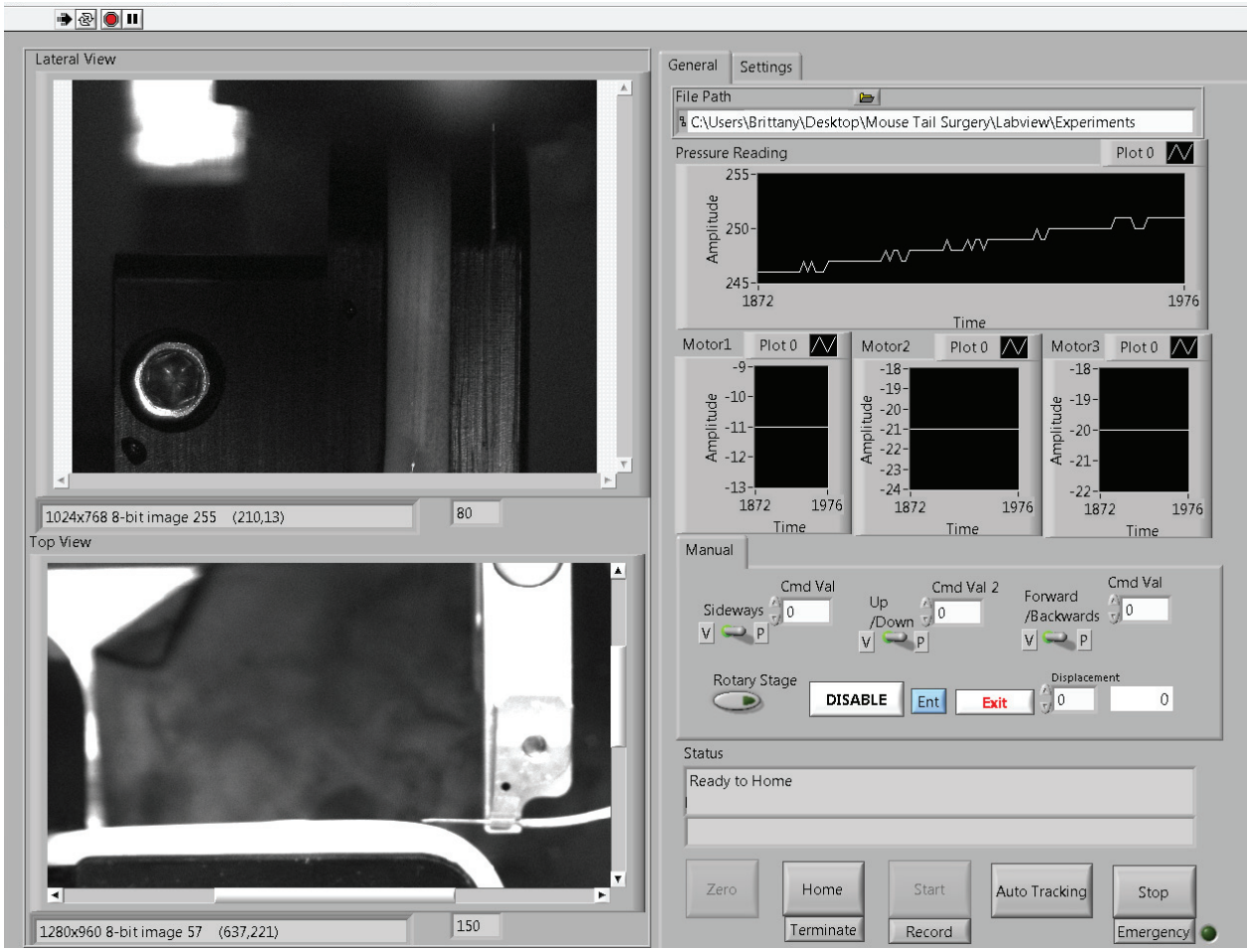


Figure 3.13: VAS GUI

CHAPTER 4

System Performance

4.1 Phantom Experiments

In order to test and control different experimental variables, a mouse tail model or phantom should be built. Two different types of mouse tail phantoms were used for the VAS studies. One mouse tail vessel equivalent phantom was created using Polydimethylsiloxane (PDMS). The purpose of this model was to mimic vessel tail sizes encased in a tissue-like polymer while having viewing access to the vessel and control over vessel pressure settings. The other mouse tail phantom used was rubber tubing with an inner diameter (ID) of 1/16" and an outer diameter (OD) of 1/8". The rubber tubing is larger than the size of a mouse tail vein, but it mimics the elasticity of tissue and the vein. The pressure inside the tubing can also be controlled.

4.1.1 Building a PDMS Mouse Tail Equivalent Phantom

The first step to building a PDMS phantom is creating a mold. A mold resembling the mouse tail vessels was created by gluing nine 28G needles to a glass slide. The 28G needles have an O.D. of 356 μm, which is similar to the tail vessel diameter estimation of 300 μm. Each needle represents a different vessel. Once the mold was created, a $\approx$5mm thick layer of PDMS (Dow Corning Sylgard 184 Silicone Elastomer) was poured over the mold. The thickness of this layer is not crucial, but it needs to be thick enough to support tubing accessing the inlet and outlet ports at the end of each artificial vessel. A separate flat layer of PDMS of 800μm was poured. The thickness of the this layer is more crucial as it represents the vessel depth from the surface of the tail. After the PDMS layers were poured, they were degassed by pulling a

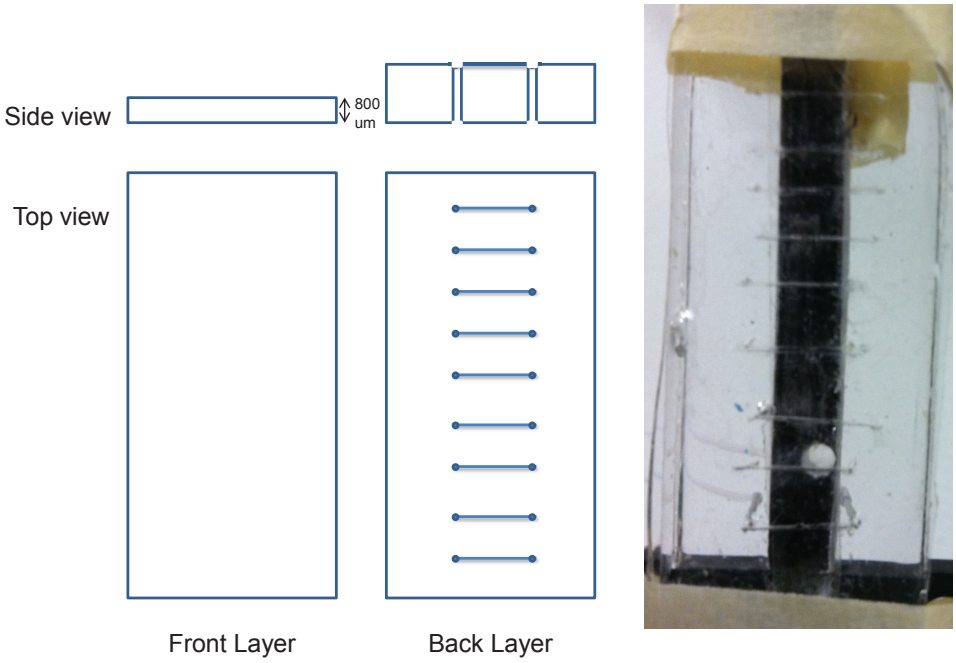


Figure 4.1: PDMS mouse tail vessel equivalent phantom schematic and phantom photo mounted on a black stand

vacuum on the layers housed in a desiccator. After the layers were degassed, they were cured in an 80°C oven overnight. Then they were peeled off of their molds. Holes were punched through both ends of each artificial vessel to serve as inlet and outlet ports. The holes were created with a 0.69mm rod and went through the entire thickness of the PDMS vessel layer. The two layers were bonded together using a corona discharge treatment (Enercon DyneA-mite HP) employing an output voltage of 15V and an exposure time of 30 seconds [30], [19]. The PDMS chip was then baked for an additional three hours at 80°C to recover the hydrophobicity after the corona treatment. The final PDMS phantom consisted of nine different vessels that were located 800μm below the surface. Figure 4.1 shows a picture of the mounted PDMS phantom.

4.1.2 General Phantom Experiment Setup

For phantom experiments, the pressure inside the artificial veins needs to be controlled. This is done by using the setup depicted in Figure 4.2: Compressed nitrogen gas is regulated

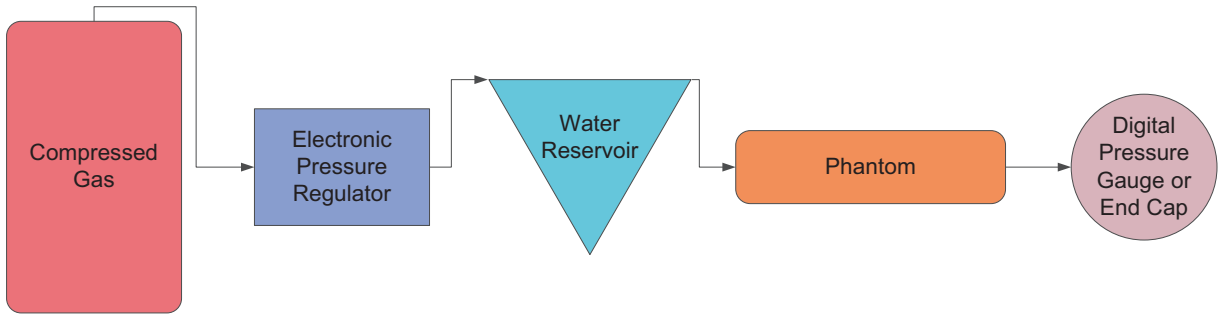


Figure 4.2: Phantom setup diagram

using a low-flow electronic precision regulator with an onboard adjustable potentiometer (Kelly Pneumatics Inc. LFR-05TK-0005-R4). The pressure setting of the regulator can be monitored by reading the voltage across the potentiometer. This signal can be read out by a voltmeter or digitized and recorded by an Arduino Mega board. The regulated nitrogen gas is fed into a capped test tube that has an inlet and outlet port drilled in the cap. The test tube is filled with dyed water. The outlet port of the test tube is connected to the inlet port of a vessel on the phantom. When the test tube is pressurized, water flows out of the test tube outlet and into the phantom vessel. The outlet port of the phantom vessel is connected to a digital pressure gauge (Dwyer DPGW-04) to verify the pressure inside the phantom vessel or simply capped after water has flown through the system.

4.1.3 Phantom Experiment Design and Results for Testing Different Needle Sizes

The PDMS phantom was used to test the pressure transducer with different sized needles. The PDMS phantom was set up as described in 4.1.2 and secured to a stand to prevent movement of the phantom while still allowing access to the vessels by the needles. Needles were attached to a linear actuator by one of two different needle holder designs. Needle holder NHV2 was used to attach a 34G needle to the linear actuator. The needle holder NHV3 was used to attach 30G needles with a luer fitting to the linear actuator. Because the best-sized needle was unknown at the time of these experiments, the final version of the VAS

needle holder as seen in Figure 3.5 was not designed or constructed before these experiments. For these experiments, a linear actuator (Zaber T-NA08A50-S) was attached to a manual

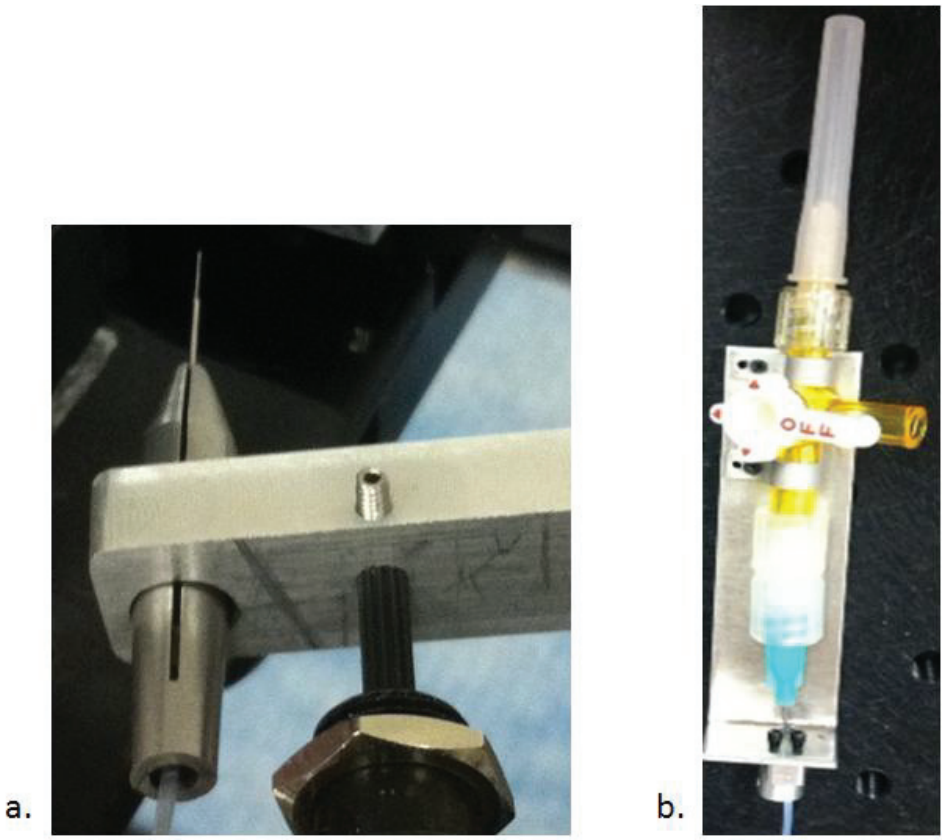


Figure 4.3: a. Needle holder NHV2 b. Needle holder NHV3

rotation mount (Thorlabs PR01/M) that was attached to a manual vertical translation stage. The vertical stage allowed the needle to be properly aligned to the correct height relative to the artificial vessel. The height of the needle relative to the vessel was monitored with a CCD camera (Imaging Source DMK 31BU03). The rotation mount altered the angle of insertion of the needle relative to the vessel. The linear actuator controlled the movement into and out of the vessel. The linear actuator was controlled by a Labview program coded to stop the linear actuator when the pressure transducer readout reached a predefined value. The pressure transducer that was used was a disposable transducer designed for clinical use (Deseret Medical Inc 8000) with custom electronics (described in Section 3.2.3) to amplify the signal. It was connected to the needle through tubing and filled with deionized water. This setup allows for the monitoring of a change in pressure in the needle when it reaches

the lumen of the vessel. The needle pressure can then be compared to the pressure inside of the artificial vessel. The pressure transducer signal and the phantom pressure regulator were both digitized with the Arduino board. By reading both the vessel pressure and the pressure inside the needle, the effectiveness of the pressure transducer with different needle sizes was monitored. The following is the Arduino code used for these phantom experiments:

```
Arduino 2 Pressure Signals Code
int analogPin = 0;      // analog pin with connection
int analogPin1= 1;

void setup()
{
  Serial.begin(9600);        //  setup serial
}

void loop()
{
  int sum=0;                //variable to store sum
  int val = 0;             // variable to store the value read
  float voltage=0;    //variable for voltage
  float mmHgTransducer=0; //variable for mmHg Transducer readout
  for(int i=0;i<5;i++){
    sum = sum + analogRead(analogPin); //avg signal for slower
     //output and to reduce noise
  }
  val=sum /5;
  voltage= val/204.6;         // Dividing by 204.6 gives the
   //output in voltage instead of 0-1024
  mmHgTransducer=20*voltage;  //Conversion factor for transducer.
   //Still needs to be baseline corrected
```

```
  Serial.print(mmHgTransducer);          // print out mmHg
  Serial.print("\t");           // but in 2 different columns

  delay(10);  //added delay for labview
  // second reading
  int sum1=0;             //variable to store sum1
  int val1 = 0;            // variable to store the value read
  float voltage1=0;         //variable for voltage1
  float mmHgReg=0; //variable for mmHg Regulator readout
  for(int i=0;i<5;i++){
    sum1 = sum1 + analogRead(analogPin1); //avg signal for slower
     //output and to reduce noise
  }
  val1=sum1 /5;
  voltage1=val1/204.6;  // Dividing by 204.6 gives the output in
   //voltage instead of 0-1024
  mmHgReg=51.71*voltage1; //Conversion factor for regulator
  Serial.println(mmHgReg);

  delay(10);
}
```

The readout from the arduino board was in mmHg for both pressure readings. The pressure readings from the needle pressure transducer have an arbitrary reference point so the readout had to be baseline corrected to match the water vapor pressure for the temperature of the water inside the pressure transducer and needle.

With the PDMS phantom, the pressure transducer readout from different needle sizes was explored. From Figure 4.4, it is apparent that the needle pressure transducer was unsuccessful in acquiring data that matched the pressure inside the artificial vessel. After

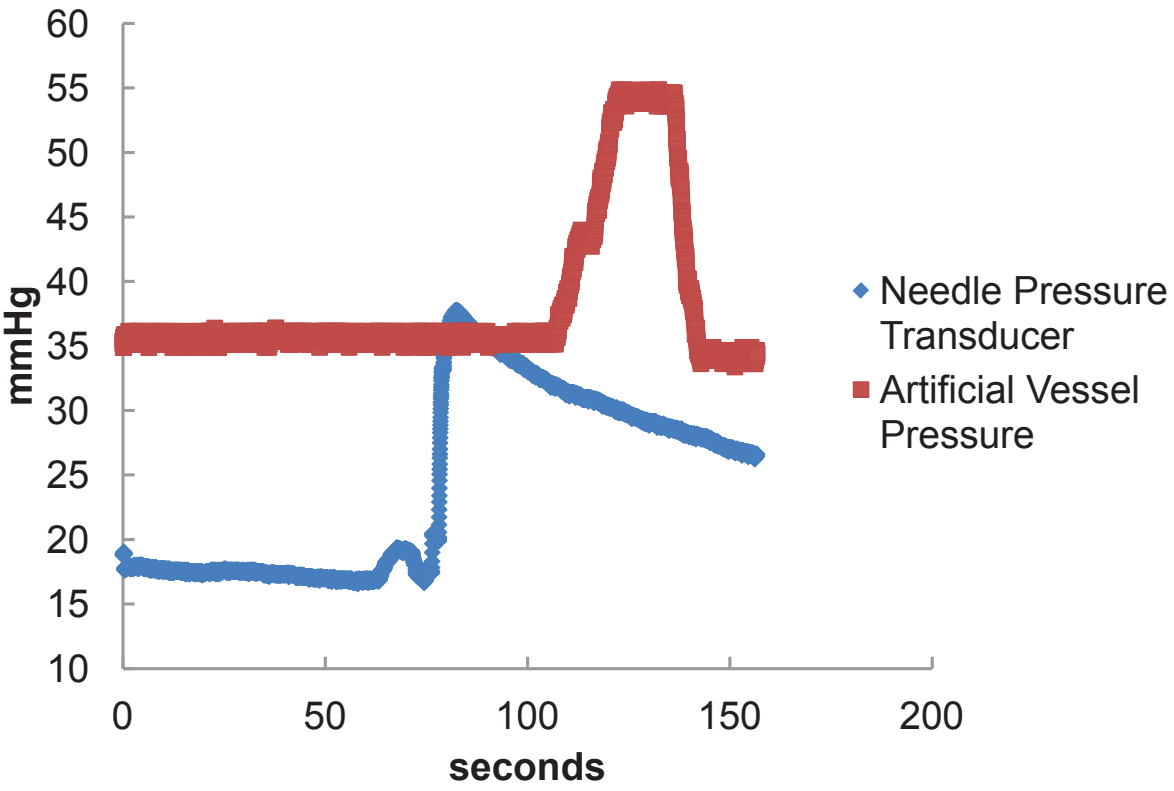


Figure 4.4: Phantom experiment results using a 34G needle

numerous experimental attempts, the cause of this discrepancy is still unknown. Rarely, a 34G needle does register the pressure inside of the vessel, but this does not happen often. We hypothesize that a small air plug may reside at the tip of the 34G needle. Due to the small size of the lumen of the needle and the potential large surface area of the air bubble, the air bubble does not move and acts as a dampener preventing the pressure inside the vein to be propagated down to the pressure transducer attached at the end of the needle. This theory has not been confirmed.

Due to the unsuccessful results using a 34 G needle, a larger 30G (O.D. 305μm, I.D. 140μm) needle was tested. With the 30G needle, the pressure transducer attached to the needle measured the pressure inside the artificial vein as is evident in Figure 4.5. In Figure 4.5a it can be seen that after the needle is inserted into the vessel, if the pressure in the vessel is changed, the needle pressure transducer detects that change. For each exper-

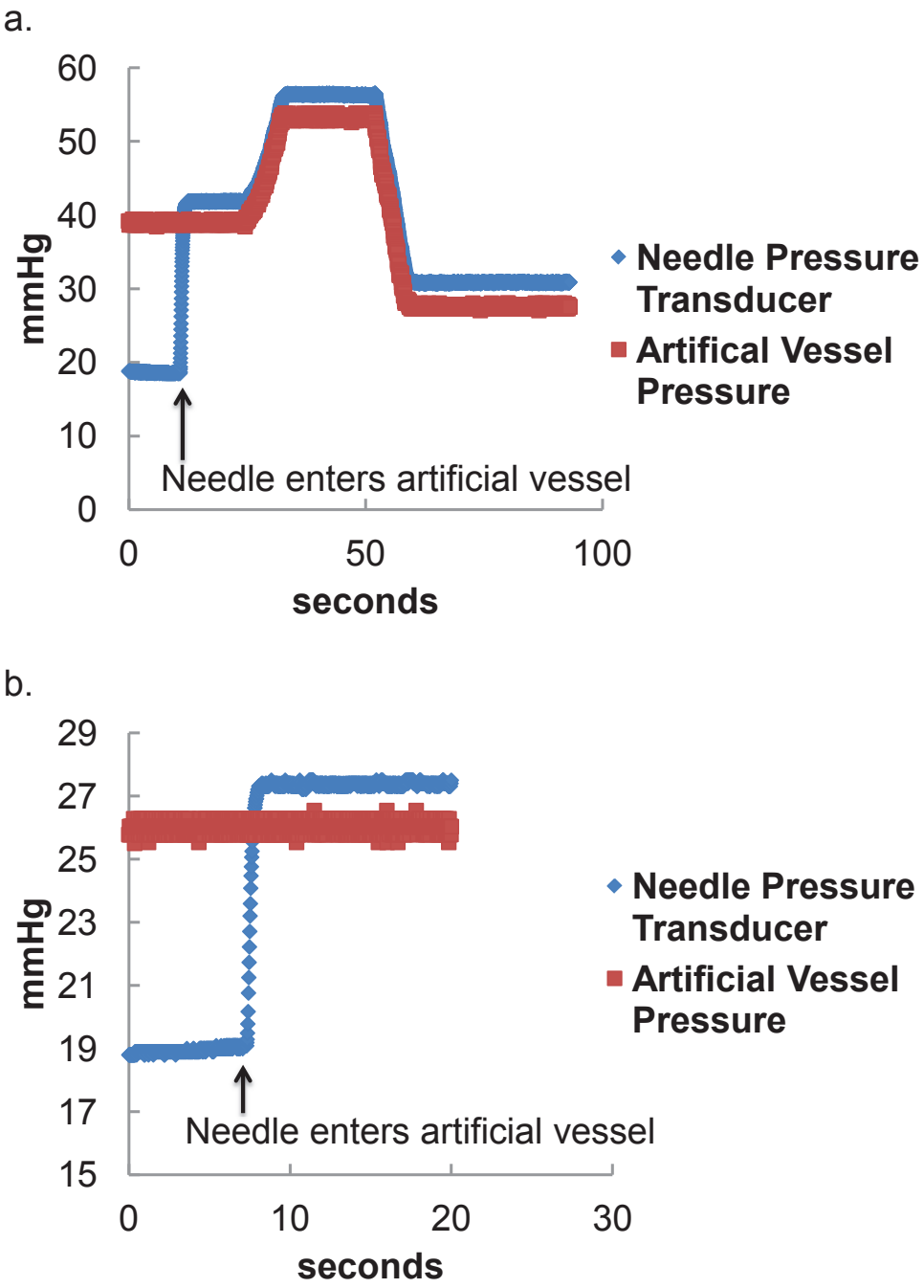


Figure 4.5: Phantom experiment results using a 30G needle. a.) The needle pressure changes as the vessel pressure changes. b.)At low pressures, a change in the pressure in the needle can be detected.

iment, the insertion of the needle, controlled by the linear actuator, was halted when the needle pressure transducer reached a predefined value. The predefined 'stop' values for the experiments in Figure 4.5 were 26 and 23 mmHg, respectively. Since the blood pressure in a mouse can drastically drop depending on environmental factors, experiments were also conducted with lower starting pressures in the vessel. This reduced the difference between the needle pressure transducer reading before entering the vessel and the pressure inside the vessel. The needle pressure transducer can detect a change in pressure and stop the linear actuator when the difference between the starting needle pressure and the pressure inside the vessel is as small as 1 mmHg.

The mouse phantom results confirm that a pressure transducer attached to a 30G needle can detect when a needle enters a pressurized vessel and signal to a linear actuator to stop further movement of the needle.

4.1.4 Phantom Experiment Design and results for Testing Motor Control Based on the Needle Pressure Signal

Due to the lack of vein depth information, the distance the needle needs to travel in the tail tissue to reach the vein is unknown. For this reason, the stopping of the linear actuator in the x direction is controlled by the pressure transducer connected to the needle. When the needle reaches the vein, a change in pressure is detected due to the blood pressure inside of the vein. The pressure feedback system is programmed to stop the motor when a running differential of the pressure points reaches a predefined value. To reduce inertial backlash, the motor does not instantaneously stop when the threshold value is reached. The motor is programmed to slow down and then back up to the location where the motor was when the threshold was first met. Appendix B contains the code for the Arduino Mega 2560 that controls the motors and pressure feedback system.

For the VAS, the distance the needle travels after the pressure threshold is met and the location that the needle backs up to, needed to be verified. To do this the rubber tube phantom was set up as described in Section 4.1.2. Using the VAS and motors described in

Chapter 3, a needle was aligned with the rubber phantom and inserted. The needle was automatically stopped based on the pressure transducer feedback system.

To confirm the needle movement and pressure feedback system, the position encoder counts from the motor, the pressure transducer output, and the time were recorded for these phantom experiments. This data was analyzed and plotted (Figure 4.6). As seen in

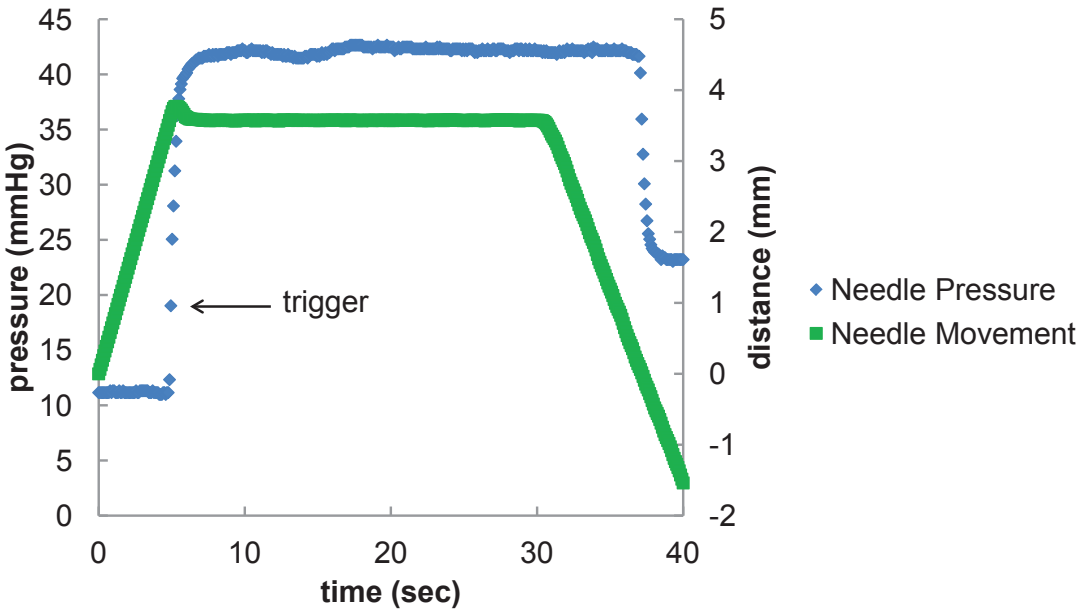


Figure 4.6: A plot of the needle movement and pressure output during a phantom experiment. The trigger point that signals to the motor to stop moving is indicated. After the trigger, the motor slows down and backs up to the location the motor was at when it was triggered.

Figure 4.6, for a needle that is traveling 800 μm/sec, the motor travels an additional 139 μm before backing up to the location where the pressure trigger occurred. After the trigger, it takes the motor 1.07 seconds to reach its resting position. This is dependent on the speed of travel of the needle. A slower moving needle will stop more quickly and will not have to backup as much as a faster moving needle.

4.1.5 Phantom Experiment Design and Results for Monitoring Pressure During an Injection

The needle on the VAS is connected to the pressure transducer and an injection probe with an IDEX 4 way T-valve (IDEX V-101T). For our purposes it was necessary to test the dead volume of the T-valve, confirm that pressure readings can be made when the pressure transducer is connected to the needle via the T-valve, and understand what the pressure readings look like during an injection. For these phantom experiments, the T-valve is set up as it would be for mouse studies. Three ports on the valve are opened and the fourth port is plugged. The nonbeveled end of the VAS needle has a 0.3048mm ID and 0.762 mm OD polytetrafluoroethylene (PTFE) tubing over the end. This needle/tubing junction is sealed with super glue. The other end of the tubing is connected to the T-valve with a flange and nut (IDEX P-248x and LT-115x). The tubing length is approximately 27 mm. The second port of the T-valve is connected to the pressure transducer using the same size tubing and type of connector as the VAS needle connection. At the pressure transducer end, a 30G needle with a luer hub is attached to the pressure transducer. The tubing from the T-valve is slid over this needle and sealed with super glue. The third port of the T-valve is connected to tubing with an ID of 0.38mm and OD of 1.09mm. The length of this tubing is approximately 8.5 mm. The tubing is connected to the T-valve with a 1/16in sleeve, flange, and nut (IDEX F252x, P240, P218x). The other end of the tubing is connected to a 28G needle and syringe. This syringe contains the desired probe to be injected. The syringe is on a syringe pump (Syringepump.com NE-1000) that controls the injection rate and volume. The VAS needle, all connecting tubing, the T-valve, and the pressure transducer are filled with DI water. A rubber phantom is pressurized and set up as recorded in section 4.1.2.

To test the dead volume between the injection syringe and the VAS needle, the injection syringe was filled with water that had been dyed red. The syringe pump injected the colored water until the first drop of colored water exited the VAS needle. It took 36μL of water for this to happen, indicating the dead volume between the syringe and VAS needle is 36μL. If the length of tubing connecting the ports is altered, this number could also change. It

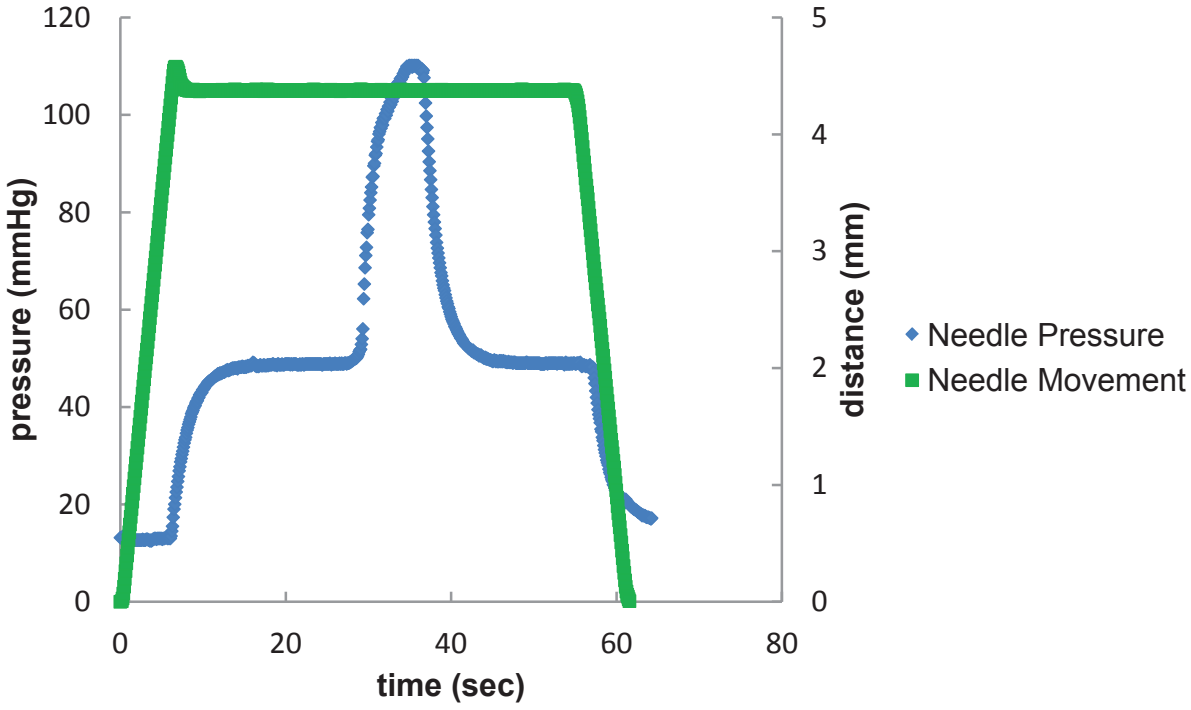


Figure 4.7: Plot of the needle movement and pressure during the needle insertion and injection in a phantom

is also worthwhile to note that no red water was observed in the line going to the pressure transducer, indicating that the injected volume does not mix much, if at all, with the water going to the pressure transducer. The next experiment used the same setup, except that the VAS inserted its needle into the pressurized phantom. After insertion, the syringe pump was used to inject 50μL at a rate of 300μL/min. The data from this experiment is plotted in Figure 4.7. The figure shows that the motors stop when the needle reaches the pressurized phantom and does not move during injection. The pressure reading on the needle levels off at the pressure inside the phantom. During the injection, the needle pressure quickly rises and after the injection the pressure declines back to same value as before the injection began. The decline in pressure is due to the relief valve on the pressure regulator controlling the pressure of the compressed gas. We do not expect to see such rapid pressure reduction after an injection in a mouse.

4.2 Mouse Tail Vein Injections

4.2.1 Categorization of Tail Vein Injectors

Tail vein injections are often performed by veterinarians, technicians, graduate students, or post docs. Training is required to perform such a procedure. At UCLA, training to perform tail vein injections consists of online modules as well as hands on 'wet lab' training. During this training session, the Department of Laboratory Animal Medicine (DLAM) personnel demonstrates and assists trainees with tail vein injections. Trainees are then encouraged to practice on their own. This training satisfies UCLA's Animal Research Committee requirements [17].

Even after training, performing a successful tail vein injection can be a challenge. Experience seems to be a key component of consistently performing a successful tail vein injection. For this reason, manual tail vein injections have been split into two different categories: expert injectors and trained routine injectors. Expert injectors are those who perform tail vein injections on an almost daily occurrence. For example, imaging center staff who assist scientists with their imaging studies by performing tail vein injections for the researchers. By contrast, trained routine injectors are trained to perform tail vein injections, but they do not perform this protocol daily. These are scientists, technicians, and students who feel comfortable performing tail vein injections, but only do them for their own studies. They are therefore not as experienced as experts. The majority of scientists using preclinical PET imaging systems will be categorized as trained routine injectors.

4.2.2 Analysis of Manual Tail Vein Injections Performed by Experts

The Crump Institute at UCLA has a long history in preclinical PET imaging. At the Crump Institute Preclinical Imaging center, Waldemar Ladno is a staff member who is considered an expert tail vein injector. He has his doctorate in veterinary medicine specializing in laboratory animals and more than 20 years of experience. He has performed thousands of tail vein injections. Mr. Ladno performed three tail vein injections for this analysis. These

studies were imaged using the Siemens Inveon preclinical PET scanner and analyzed in the same manner as studies performed by the VAS, as outlined in section 4.2.4. Four studies were also collected and analyzed from the preclinical imaging center at Washington University in St. Louis. These studies were also performed by an expert. From these seven studies, the amount of radioactivity left in the tail after a tail vein injection performed by an expert was found to be 1.8% $\pm$1.0%. This number is consistent with the amount of radioactivity found in the tail after an intraperitoneal injection due to the circulation of the probe in the blood, according to the paper by *Vines et al* [46].

4.2.3 Analysis of Manual Tail Vein Injections Performed at the Crump Institute at UCLA by Trained Routine Injectors

From the Crump Institute database, PET/CT FDG mouse studies that were injected by trained personnel other than the veterinarian on staff (Waldemar Ladno) were randomly selected. For these studies, data was acquired for about 1 hour after the injection on a microPET Focus system, which has an axial FOV of 7.6 cm [6]. Because of this limited axial FOV, the tail is often not included in the image. Therefore, the amount of unaccounted for radioactivity cannot be directly measured from the images, but must be calculated. This is done by measuring the radioactivity present in the image and subtracting it from the decay corrected amount injected into the mouse. The difference between the amount of radioactivity that was injected and the amount in the image is the amount of unaccounted for radioactivity. The majority of the unaccounted for radioactivity is assumed to be left in the tail as the tail is the only part of the mouse body not present in the image FOV. Errors in this method can come from incorrect recordings of the injected dose, lack of residual radioactivity measurements after an injection, errors in PET calibration constants, and errors related to the dose calibrator that is used to measure the injected dose.

To calculate the amount of unaccounted for radioactivity, the randomly selected filtered back projection (FBP) reconstructed studies from the database were imported into the software program AMIDE. The mouse bed chambers were removed from the CT image using

AMIDE or an in-house developed bed chamber stripping software program. AMIDE was then used to draw a mouse body 3D isocontour region of interest (ROI) based on the CT image. That ROI was then applied to the PET image. ROI analysis was performed for each study using the AMIDE software and the ROI mean values and sizes (mm^3) were recorded.

To calculate the unaccounted for radioactivity is a three step process. We begin by calculating the injected activity at the starting scan time (*A(t)*) by using the following decay correction equation:

$$A(t) = A_0 e^{-\lambda t}, \tag{4.1}$$

where *A(t)* is the decay corrected injected activity at the scan time in μCi. A_0 is the drawn dose, as recorded in the database records. λ is the decay constant, which is .006315 min^{-1} for F^{18}. t is the time between the dose being drawn and the scan start time, in minutes [13]. If there is residual activity after the injection, this activity is also decay corrected and subtracted from *A(t)*. *A(t)* is the amount of activity that we expect to be in our PET image as it is the injected dose.

The second step is to calculate the amount of radioactivity present in the PET image FOV by using the following equation:

$$B = 1E10^{-6}\frac{G_{AC}\mu V}{C} \tag{4.2}$$

B is the amount of radioactivity (μCi) in the mouse body ROI. G_{AC} is a general attenuation correction constant that is applied to the non-attenuation corrected reconstructed images retrieved from the database. This general attenuation correction value was calculated by comparing the mean value of the ROI analysis from a study that was reconstructed with attenuation correction versus the same study reconstructed without attenuation correction. This was done for 6 different studies. The general attenuation correction value was calculated to be 1.3 with a standard deviation of 0.01. The small standard deviation along with the similarities in mouse size and chamber design used for all studies, implies that applying a general attenuation correction value is reasonable at least for these FDG type studies. μ is the mean value of the ROI. V is the size, in mm^3, recorded from the ROI analysis. C is a PET system specific attenuation corrected calibration constant. This constant is found

by imaging a cylinder with a known volume and radioactivity in a PET scanner. The PET units are then converted to nCi/mL. The constant applied to the Focus studies from the Crump database was 1.2710^{-7} per unit· mL/nCi.

By subtracting the activity present in the PET image from the decay corrected injected activity, the missing amount of radioactivity is calculated. Because the tail is the only part of the mouse body missing from the PET image, it is assumed that the difference between the injected dose and the actual dose in the image is mostly the amount of radioactivity left in the tail. To compare different studies, the percentage of unaccounted for radioactivity compared to the recorded injected dose is calculated using Equation 4.3.

$$\%T = \frac{A(t) - B}{A(t)} \tag{4.3}$$

47 different PET studies injected by 17 different trained injectors were analyzed using the method discussed above. The average percentage of radioactivity left in the tail by each injector is plotted in Figure 4.8.

The average percentage of unaccounted for radioactivity from these 17 trained injectors was found to be 16%. This number represents the accuracy of injections performed by trained tail vein injectors. To calculate the consistency of injections performed by trained tail vein injectors, one-way analysis of variance (ANOVA) was performed by using equations 4.4 and 4.5 [9].

$$SD_{BetweenInjectors} = \sqrt{\frac{\sum n_i(\bar{Y_i} - \bar{Y})^2}{I - 1}} \tag{4.4}$$

$$SD_{WithinInjectors} = \sqrt{\frac{\sum (n_i - 1)s_i^2}{n - I}} \tag{4.5}$$

$\bar{Y_i}$ is the mean value of radioactivity left in the tail by the i injector. $\bar{Y}$ is the overall mean value of radioactivity left in the tail. I is the total number of injectors. n_i is number of studies performed by the i injector. s_i^2 is the variance of the i injector and n is the total number of studies.

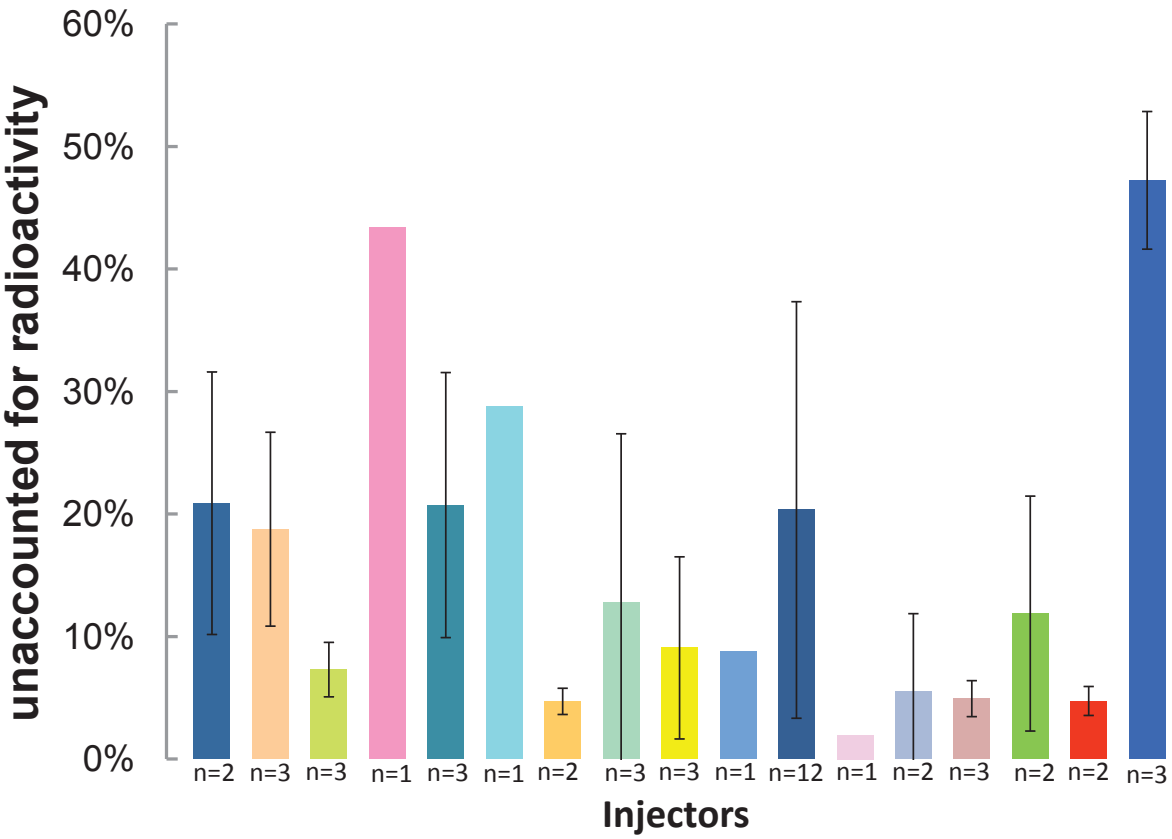


Figure 4.8: The percentage of radioactivity left in a mouse tail after injections performed by trained routine injectors

Equation 4.4 is used to determine the standard deviation between the different injectors. Using this equation, the standard deviation between the 17 different injectors was found to be 18.8%. Besides the variability between injectors, there is also a large fluctuation in the injection accuracy of one injector. Equation 4.5 was used to calculate the standard deviation within an injector, based on these 47 studies. The standard deviation within the injectors was calculated to be 12.1%.

To validate our assumption that the majority of unaccounted for radioactivity really is in the tail, three additional trained injectors were recruited to perform tail vein injections. They injected FDG into a total of 7 mice. Approximately 60 minutes after the injection, the mice were scanned in the Siemens Inveon scanner, with the tail in the FOV. These images

were attenuation corrected and analyzed just like studies performed by the VAS, as detailed in Section 4.2.4. From these studies, the average percentage of radioactivity remaining in the tail was 13.9% with a standard deviation between injectors of 12.3. These numbers are lower than what was found when calculating the percentage of unaccounted for radioactivity, but the general trend continues of larger than desired amounts of radioactivity remaining in the tail and large variations between injectors.

4.2.4 Analysis of tail vein injections performed by the VAS

The following protocol was used to perform tail vein injections with the VAS:

1. Turn VAS on.
2. Anesthetize mouse approximately 30 minutes before injection.
3. Inject 200 μL of saline subcutaneously to prevent the subject from being dehydrated at the time of injection.
4. Remove mouse from a sedated state.
5. Secure a 30G beveled needle to the needle holder of the VAS. The needle has approximately 6 inches of 0.012in ID tubing attached to the back end of the needle. The tubing/needle junction is sealed with super glue. The other end of the tubing has a 30 G needle with a leur fitting hub. This junction is also sealed with super glue.
6. Verify that the needle bevel is properly positioned with the bevel point rotated to be closest to the tail support.
7. Attach the injection needle to the pressure transducer via the needle with the fitting.
8. Fill the pressure transducer, tube, and needle with room temperature distilled water or saline.
9. Close the filling port on the pressure transducer.
10. Re-anesthetize mouse.

11. Place mouse on heated mouse chamber and cover mouse with a blanket.

12. Secure mouse tail on tail support using the tail clamp.

13. Verify that the tail is secure and taut. If not, resecure the tail.

14. Using the labview GUI and a theta of zero, properly align the needle relative to the vein in the y direction.

15. Using the manual stage, move the needle in the z direction so that the needle is touching the surface of the tail.

16. Back the needle up in the x direction.

17. Move the needle with the computer controlled motor in the z direction 800-1300μm towards the tail.

18. Add additional heat to the tail by placing gauze that has been heated with warm water over the tail for 10-20 seconds.

19. Move the needle in the x direction at a speed of 200μm/sec.

20. When the needle automatically stops due to a change in pressure, verify that the increase in pressure is sustained.

    If the needle does not stop during the insertion into the tail, remove the needle at a speed of 200μm/sec. If the needle automatically stops, refer to step 20. If the needle does not stop during the removal process, the attempted needle insertion has failed and start over again at step 12.

21. If the needle pressure increase is sustained, cut the needle-pressure transducer tubing near the needle.

22. Insert syringe needle, containing about 100μCi of FDG in 50μL of saline, into the tubing.

23. Push syringe plunger down and infuse the FDG into the mouse.

24. Flush line with 20μL of saline.

25. Remove VAS needle from the mouse tail.

26. Remove the mouse from the VAS system, but keep it sedated in an induction chamber.

27. Measure and record and residual activity from the injection.

28. After one hour of uptake, perform a 10 minute PET scan of the mouse on the Siemens Inveon scanner, with the mouse tail wrapped in the FOV.

29. Take a CT the mouse in the same position.

30. Perform attenuation correction on the PET image and analyze the image.

The accuracy and consistency of the VAS is measured by analyzing the radioactivity in the mouse tail after an injection is performed by the VAS. Mice are injected with FDG by the VAS. The mice remain under anesthesia during uptake, which is usually around one hour. One hour uptake was chosen because it is the standard uptake time for most PET studies performed at the Crump Institute. Since the VAS is being compared to other studies performed at the Crump, the experimental conditions are designed to be similar. Keeping the mouse under anesthesia during uptake will prevent the mouse from urinating and excreting some of the injected radioactivity. For imaging, the mouse tail was wrapped in a manner so that the entire mouse, including the tail were in the PET FOV.

Mouse injection studies by VAS were imaged with the preclinical Siemens Inveon system. A ten minute PET scan was followed by a CT scan. The CT scan is used for the attenuation correction and ROI drawing. PET attenuation corrected PET scans were reconstructed using filtered back projection [39]. The reconstructed PET images and the CT image were imported into AMIDE. Because the entire mouse is in the FOV, the conversion from PET counts to radioactivity is not necessary as well as the injected amount of radioactivity. Two ROIs can simply be compared, a whole body ROI and a ROI excluding the tail. The ROIs are drawn in AMIDE and are based on the CT scan. After the chamber has been stripped from the CT scan, a 3D isocontour is drawn on the CT image by setting the minimum CT

value to include in the ROI at 450. This ROI is then applied to the PET image. Depending on the injection, the ROI might need to be manually enlarged around the injection site to include spillover. This is done by using the ROI draw tool in AMIDE. The statistics of the whole body ROI are analyzed. For the tail data, the tail of the mouse is erased in the CT and PET image. Another 3D isocontour is drawn on the non-tail data using the same thresholds as were applied for the whole body ROI. The percentage of radioactivity in the tail compared to the entire body is calculated by using the following equation:

$$\%T = \frac{\mu_W * V_W - \mu_{NT} * V_{NT}}{\mu_W * V_W} * 100, \tag{4.6}$$

where μ_{NT} is the mean value in the no tail ROI and V_{NT} is the size (volume) of the no tail ROI. μ_W is the mean value of the whole body ROI and V_W is the size of the whole body ROI.

Following the above mentioned protocol, the VAS was used on 15 mice. Only 5 of the 15 mice were injected with radioactivity, due to the unsuccessful insertion of the needle. While it is undesirable not to inject 10 of the 15 mice, it is beneficial that using the VAS, one knows when not to proceed with injection. This is because the pressure reading provides insight into whether or not the needle was properly placed inside the vein. Of the 5 mice that were injected, the average amount of radioactivity left in the tail was 3.4% with a standard deviation of 4.5. More information about these 5 studies is available in Table 4.1. The PET images of the 5 studies are in Figure 4.9 and pressure and needle movement plots are in Figure 4.10.

4.2.5 Comparison Between Manual Injections and Injections Performed by the VAS

Seven PET studies performed by experts, seven PET studies performed by trained injectors, and five PET studies performed by the VAS, all containing the entire mouse body, including the tail, in the FOV were analyzed. As seen in Figure 4.11, the VAS performs better than the average trained injector and it is comparable to an expert injector in terms of the amount of radioactivity left in the tail after an injection. With further optimization, the VAS could

be a useful tool to perform accurate, reliable, and safe tail vein injections in mice.

Table 4.1: Mouse studies performed by the VAS

Study #	Mouse Strain	Mouse Age (wks)	Total Injected Volume (μL)	Uptake Time Before Scan (min)	% of Radioactivity in the Tail
1	C57 Bl/6	71	70	72	2.1
2	C57 Bl/6	115	70	61	1.9
3	Bl6	15	70	60	1.1
4	Bl6	15	70	61	11.4
5	Bl6	16	70	157	0.7

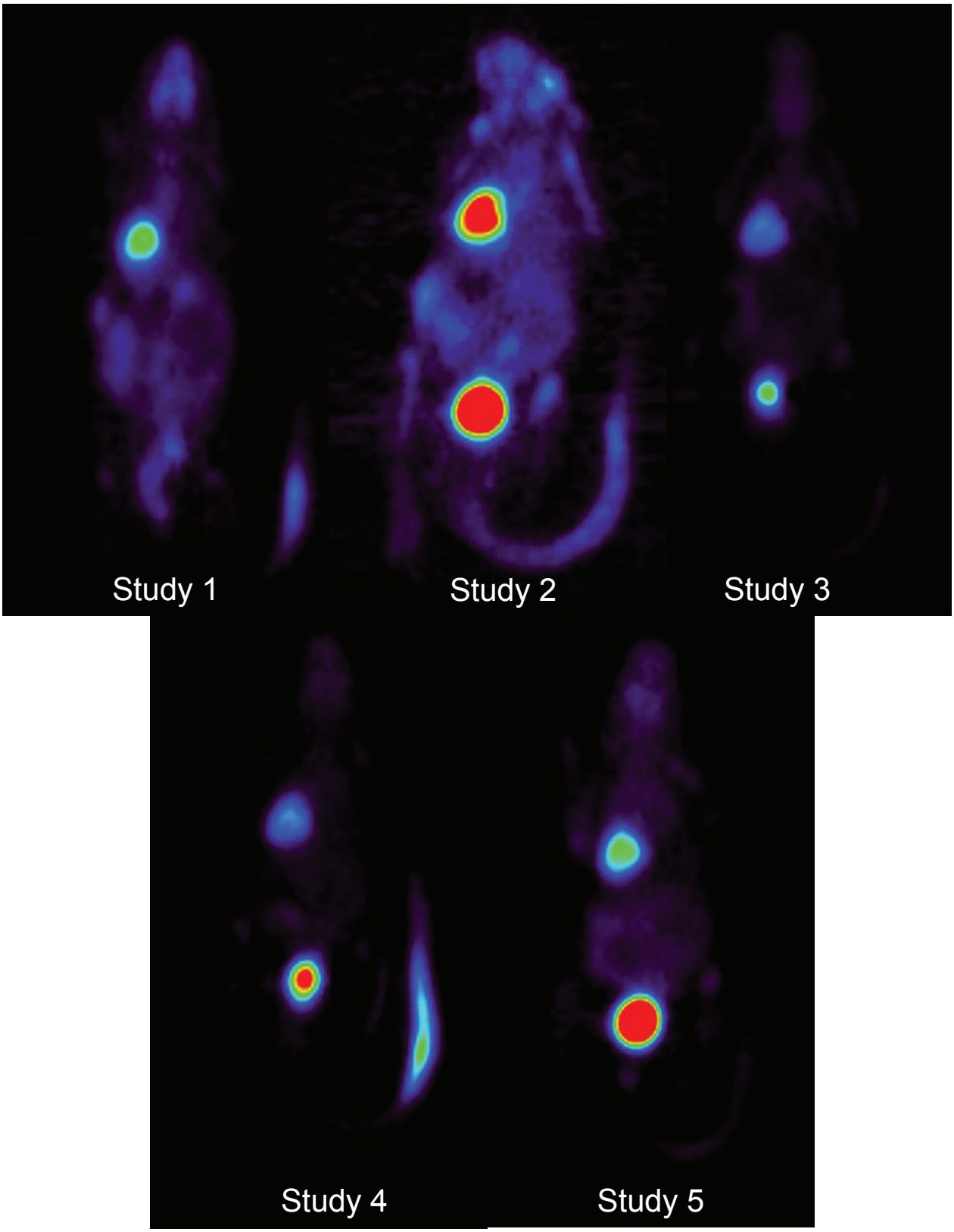


Figure 4.9: VAS injected mouse PET images with the tail in the FOV

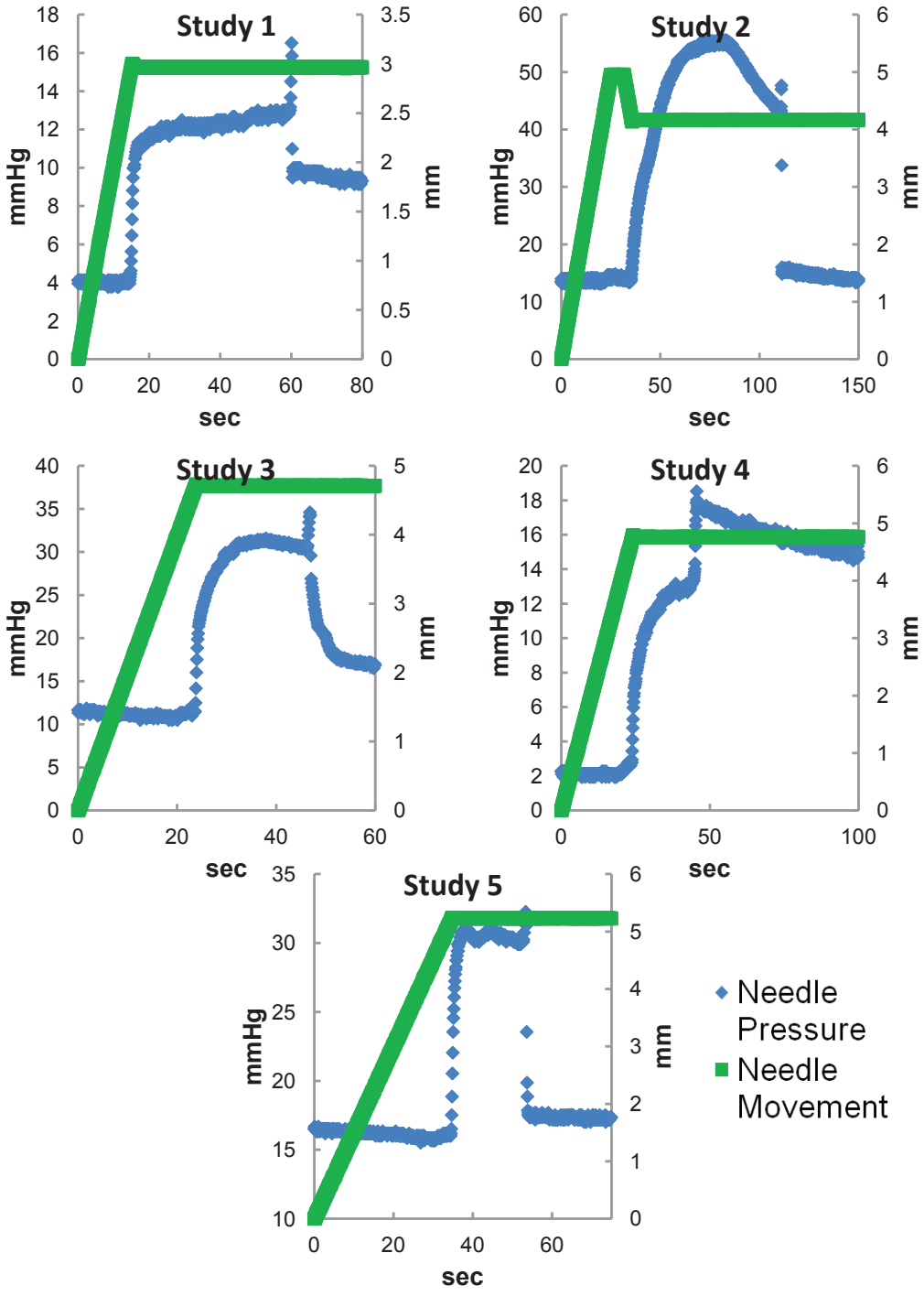


Figure 4.10: Pressure readings and needle movements in the x direction for the needle insertion in to the mouse tail vein for 5 different studies. The sudden drop in pressure is due to cutting the connection of the needle to the pressure transducer.

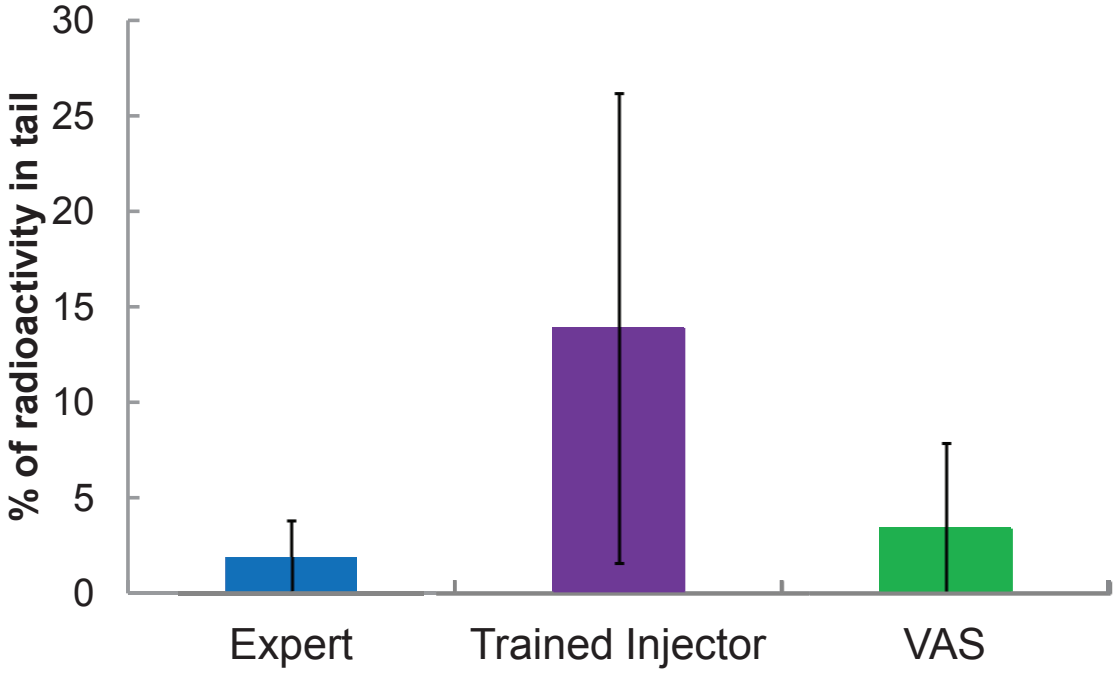


Figure 4.11: Comparison of injections performed by manual injectors and the VAS

CHAPTER 5

Conclusions

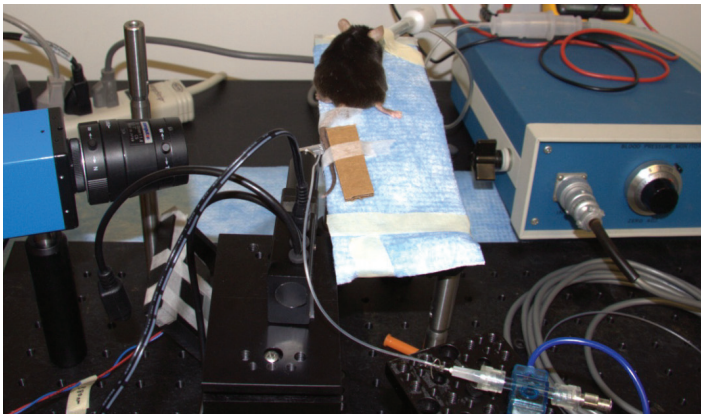
5.1 Overview of the VAS

For this dissertation, we have created a vascular access system that semi-automatically inserts a needle in to the mouse tail vein for probe injections. When properly used, our current design hits the desired target 33% of the time. While this number is lower than desired, our device has the benefit of knowing when the vein was not reached and therefore prevents a poor injection from occurring. When the vein is reached and an injection happens, the VAS performs great injections by only leaving on average 3.4% of the injected probe in the tail.

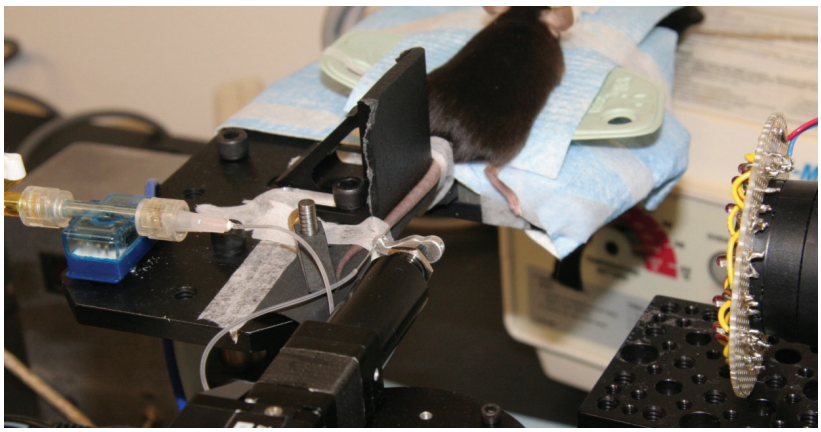
There have been many different VAS prototypes before this final design presented in this dissertation, as seen in Figures 5.1 and 5.2. Through the different iterations of the VAS, the basic concepts have remained the same: Use NIR light and image processing to identify the tail vein, use mechanical means to move the needle into the tail tissue, and stop the movement of the needle and confirm that the needle is inside the vein by using a pressure feedback system. All of this is done by using low cost components to enable the VAS to be feasibly used by others.

For this dissertation, a working proof of concept prototype was created. Further improvements will be made to make the VAS a reliable and useful tool to scientist using murine models for their research.

Prototype 1



Prototype 2



Prototype 3

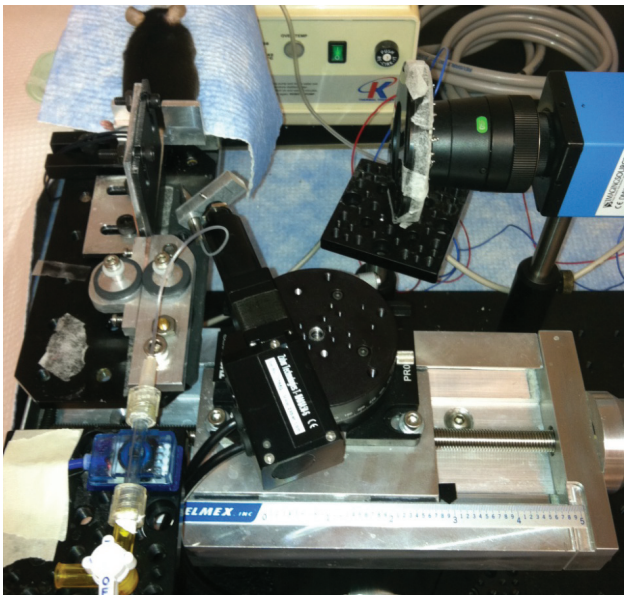
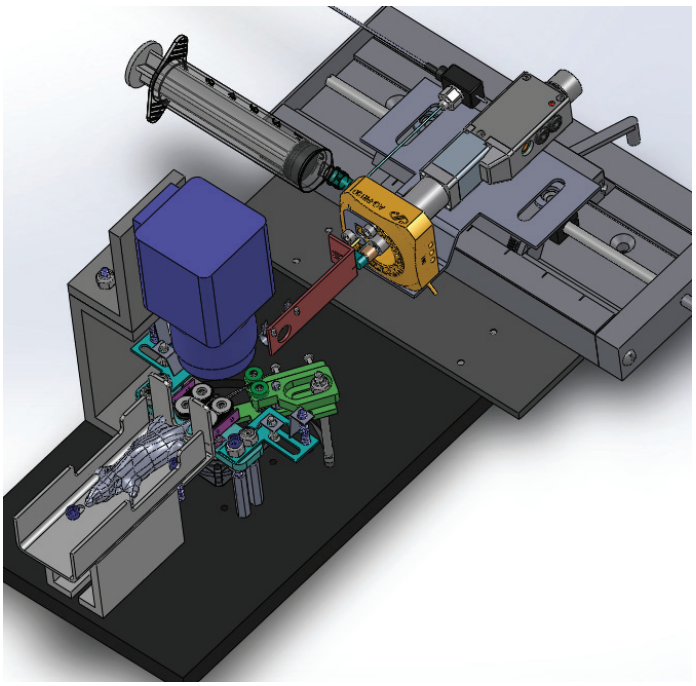


Figure 5.1: Different iterations of the VAS prototype

5.2 Future Work

The VAS will continue to be tested and optimized. It needs to be tested with different strains and ages of mice to better understand the limits of the system. It will continue to be

Prototype 4



Prototype 5
(final design)

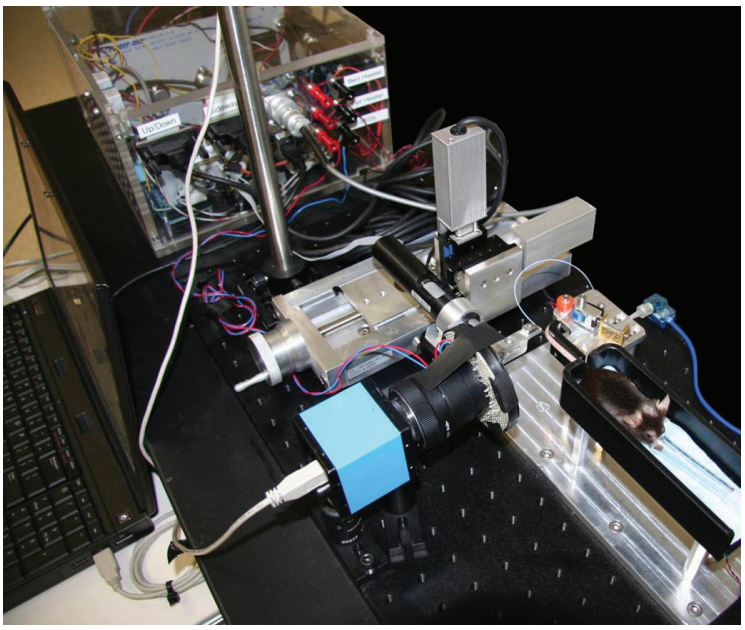


Figure 5.2: More iterations of the VAS prototype

optimized to increase the number of successful needle insertions for each attempt. Besides further testing, the protocol and design will be optimized to increase ease of use and utility for other types of studies, besides static mouse PET scans.

5.2.1 Increasing ease of use

One way of increasing the ease of use of the VAS is to take it from a semi-automatic system to a fully automatic system. Currently, the VAS is semi-automatic as it requires user inputs for the location to move the needle for the proper alignment. The VAS software identifies the vein and the user inputs computer commands to move the needle to that location. To make the system fully automatic, needle tracking is required so that the VAS software knows where the needle is located and moves the needle to the identified vein location without requiring user inputs. Needle tracking software is already in the development stage. During testing of the semi-automatic VAS, data was collected to help develop a needle tracking program. This program will be tested in the near future with the aim of making the VAS more user friendly and robust.

Another area for improvement to increase ease of use is the needle holder design. Currently, much work must go into preparing a needle to be used with the VAS including removing it from the luer hub, attaching it to tubing, and sealing the tubing-needle junction with glue. After the needle is prepped, it must be secured on the VAS needle holder which can be cumbersome and a potential safety issue. In the future, needles with very little prep work requirements should be used that can easily and safely be attached and detached from the system. Perhaps a "butterfly" like needle can be used in the future, where the wings are used for securing the needle to the VAS making it easier to handle the needle. A needle shielding mechanism should also be employed to increase the safety of using the VAS. By adding this additional safety measure, the VAS could potentially be safer for scientists to use than performing a traditional manual injection.

Lastly, additional heating mechanisms should be explored so that the additional step of heating the tail with warm gauze can be removed. While the tail support is currently heated, it does not seem to be enough. Potential solutions include redesigning the tail support so that there is more contact with the tail, increasing the heat transfer from the tail support to the tail. The tail support should also be made out of a material with better heat conduction properties. While the current version of the tail was purposely made out of delrin to reduce

heat transfer to other components of the VAS, in the future the tail support can be made out of a better heat conducting material, such as aluminum, and isolated from other VAS components by using nylon buffers. Other mechanisms such as increasing the contact area of the tail support and tail, adding a heating sleeve over the tail, or using external lights can also be explored to increase the temperature of the tail. Any additional heating mechanism will be constrained by the fact that it cannot interfere with the viewing of the tail vein and it cannot heat the needle or pressure transducer as this will negatively effect pressure readings.

5.2.2 Adapting design for other needs

The current VAS was created for anesthetized mice to be injected on the benchtop. To increase the utility of the device, it should also be able to accommodate different situations, depending on the experimental protocol. This includes injecting non-anesthetized mice, catheter insertion for longitudinal studies, MRI environments, dynamic PET scans, blood sampling, and injection of rats.

A possible modification of the VAS for the injection of non-anesthetized mice is to replace the mouse bed with a mouse restraining cylinder. These restraining cylinders are used for manual injections of non-anesthetized mice. They prevent the mouse from moving much by having the body of the mouse in the cylinder, yet the tail outside of the cylinder. Additional tail clamps may need to be added to the VAS to further restrain the tail of an non-anesthetized mouse.

It would be beneficial for the VAS to accommodate catheter insertion in the mouse tail vein. For this, a tubing is slid over a needle once it is in the vein. The tubing remains in the vein as the needle is extracted. By redesigning the needle holder, the VAS should be able to perform catheter insertions too. Besides the benefits of a catheter for longitudinal studies, a catheter could be used for dynamic PET studies and MRI studies. The catheter would be placed in the subject on the benchtop. After the catheter was inserted, the subject would be moved to the imaging system. Once inside the imaging system, the desired probe or contrast could be infused through the VAS placed catheter. Besides the catheter option, it would

be beneficial for the VAS footprint to shrink so that it could be used in or just outside of a scanner for dynamic studies. The mouse imaging chamber would need to be altered to accommodate the miniaturized VAS, but it could be a useful modification for many imaging scientists.

Using the same setup as for tail vein injections, the VAS should be tested for the use of blood sampling. Because it has yet to be tested, it is unclear if blood sampling from the tail vein or the ventral tail artery is possible with the VAS. If possible, blood sampling would be an added benefit to using the VAS.

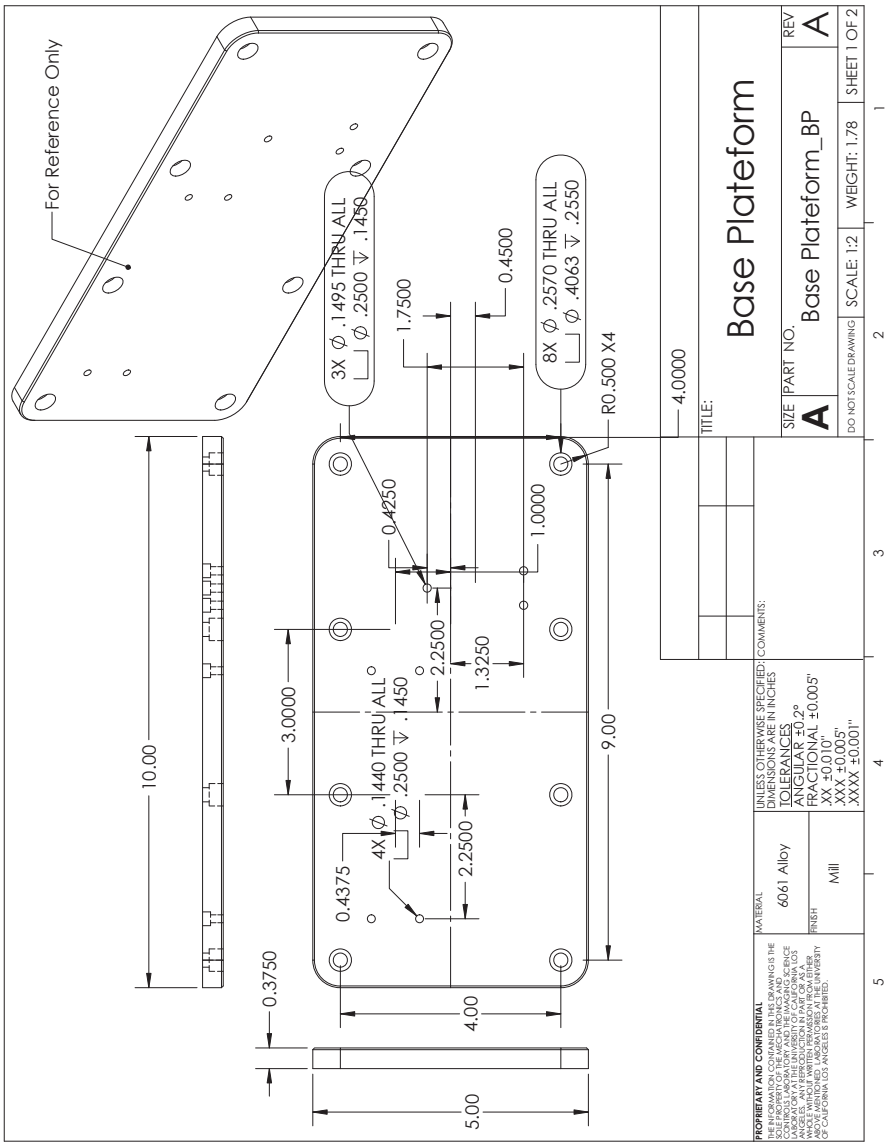
Finally, the VAS was designed to be used on mice. By using the same general concepts, but larger tail supports and tail clamps, the VAS could possibly be used on rats too. Rats are not only larger, but their tails are thicker with scales. This could cause a problem for the VAS, but would need to be tested. The imaging processing would also need to be re-optimized for rats. There is also the possibility of scaling the VAS even larger to be used on humans. The same basic principles the VAS used for mice, could translate to vein access of humans. With some modifications, the VAS has the potential to be adapted to meet the needs of a wide range of experimental protocols, including injections for non-anesthetized mice, catheter-based studies, dynamic PET scans, blood sampling, and injections of rats.

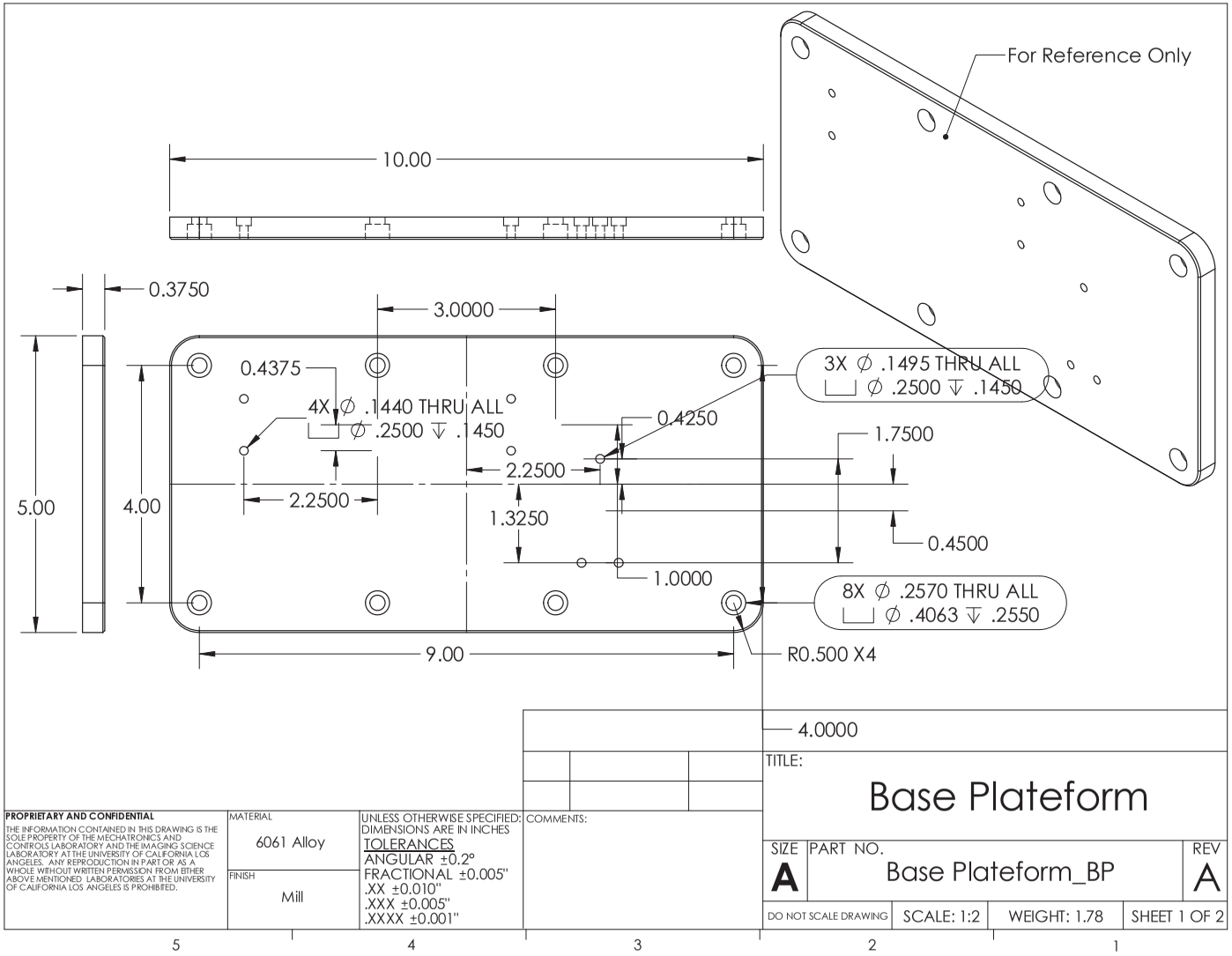
5.3 Conclusion

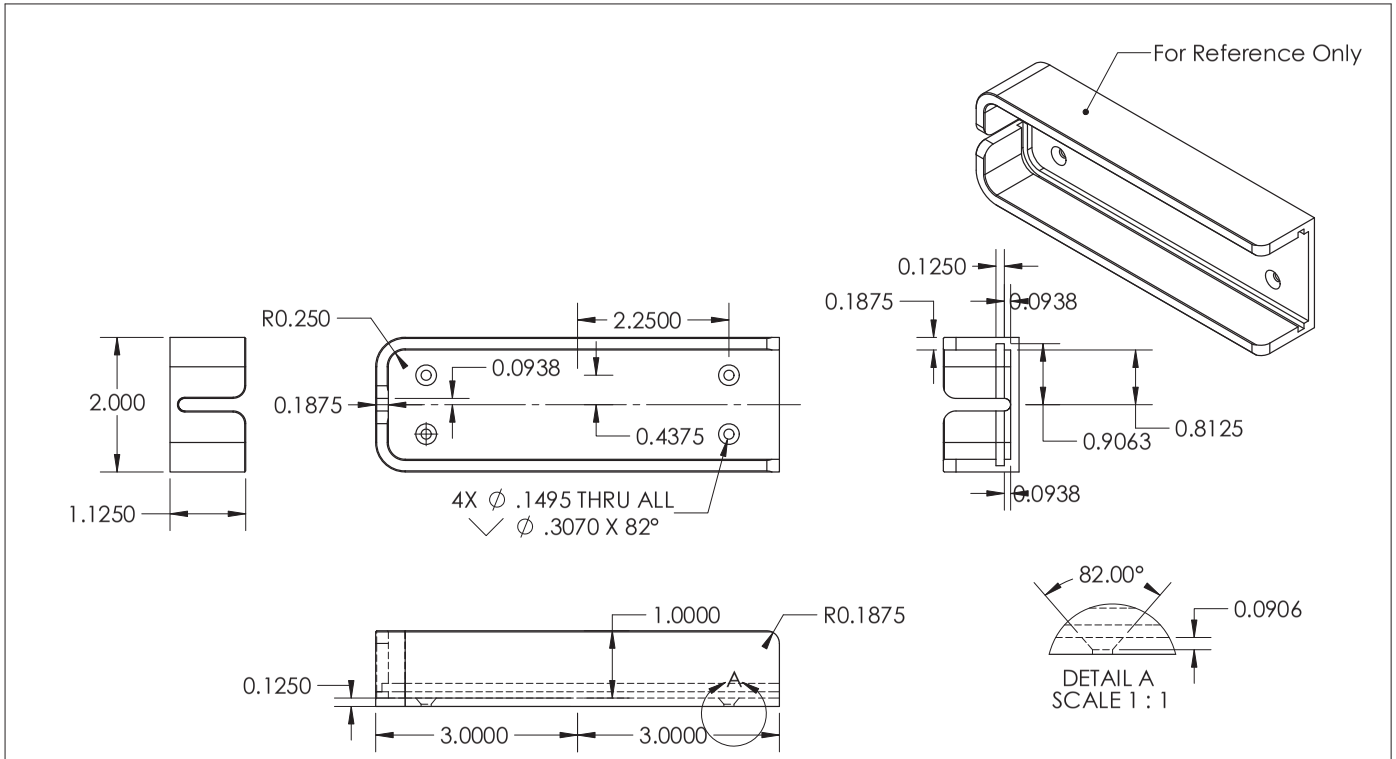
In conclusion, a semi-automated vascular access system (VAS) has been created for this dissertation. It uses NIR light to image and locate a mouse tail vein, computer controlled motors move a needle to the vein location and insert the needle into the vein, and a pressure feedback system that stops the motor and confirms that the needle is inside of the blood vessel. This device will assist scientists with mouse studies that require an injection of a substance into the mouse vascular system. It is especially useful to preclinical PET users and will aid in the further development of high throughput, user friendly imaging studies.

APPENDIX A

VAS Parts Drawings

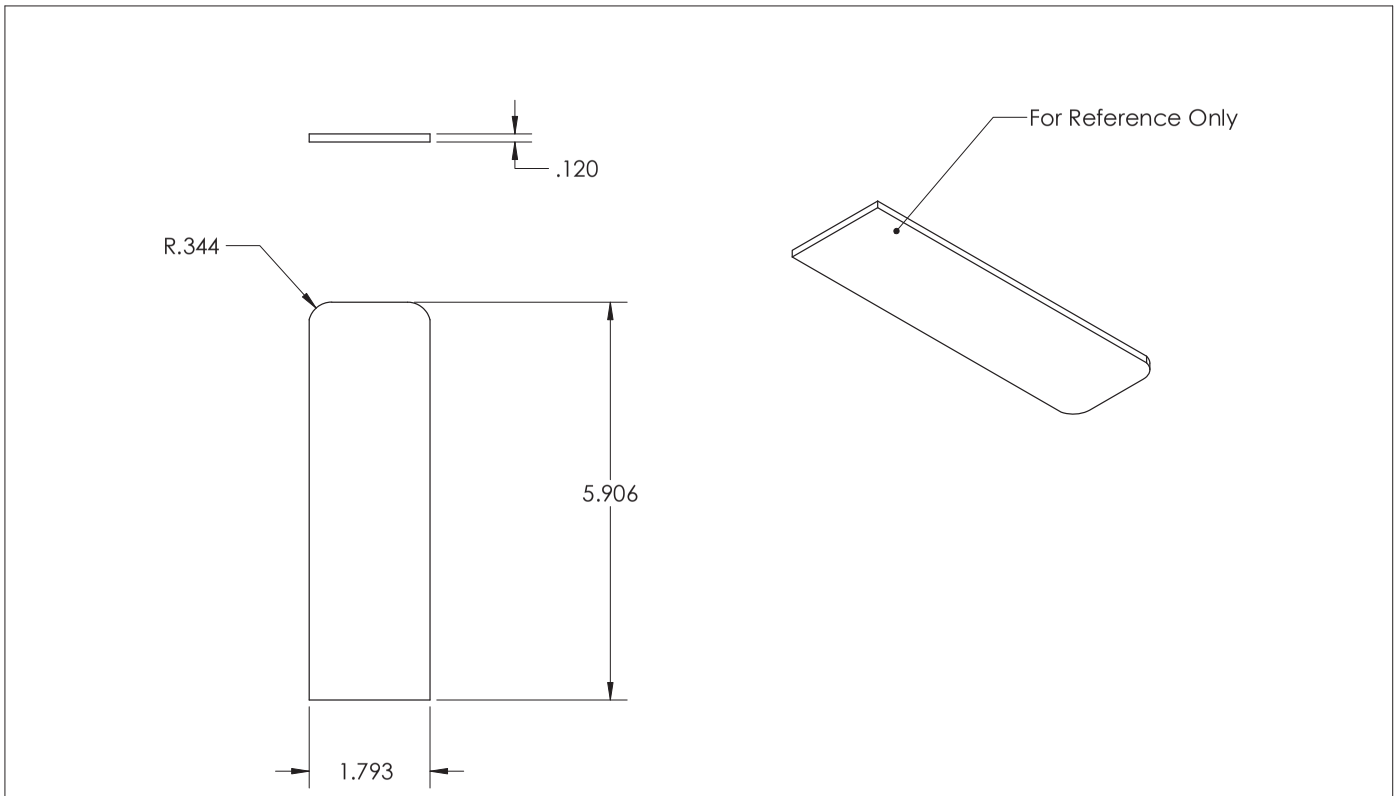



For Reference Only
10.00
0.3750
3.0000
0.4375
4X ∅ .1440 THRU ALL
⌴ ∅ .2500 ↧ .1450
3X ∅ .1495 THRU ALL
⌴ ∅ .2500 ↧ .1450
0.4250
1.7500
2.2500
2.2500
1.3250
0.4500
1.0000
5.00
4.00
8X ∅ .2570 THRU ALL
⌴ ∅ .4063 ↧ .2550
9.00
R0.500 X4
4.0000
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER ABOVE MENTIONED LABORATORIES AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES IS PROHIBITED.
MATERIAL
6061 Alloy
FINISH
Mill
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES
ANGULAR ±0.2°
FRACTIONAL ±0.005"
.XX ±0.010"
.XXX ±0.005"
.XXXX ±0.001"
COMMENTS:
TITLE:
Base Plateform
SIZE
A
PART NO.
Base Plateform_BP
REV
A
DO NOT SCALE DRAWING
SCALE: 1:2
WEIGHT: 1.78
SHEET 1 OF 2
5
4
3
2
1



PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND IMAGING SCIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER OF THE ABOVE MENTIONED LABORATORIES AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES IS PROHIBITED.	MATERIAL Delrin 2700 NC010, Low Viscosity Acetal Copolymer (SS) FINISH Mill	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	COMMENTS:	TITLE: MOUSE HOLDER SIZE A | PART NO. Mouse Holder | REV A DO NOT SCALE DRAWING | SCALE: 1:2 | WEIGHT: 0.26 | SHEET 1 OF 1

5 4 3 2 1



		VAS
		TITLE: Heating Plate for Mouse Holder
		SIZE A | PART NO. Mouse Caddy Heating Plate | REV A
		DO NOT SCALE DRAWING | SCALE: 1:2 | WEIGHT: 0.06 | SHEET 1 OF 1

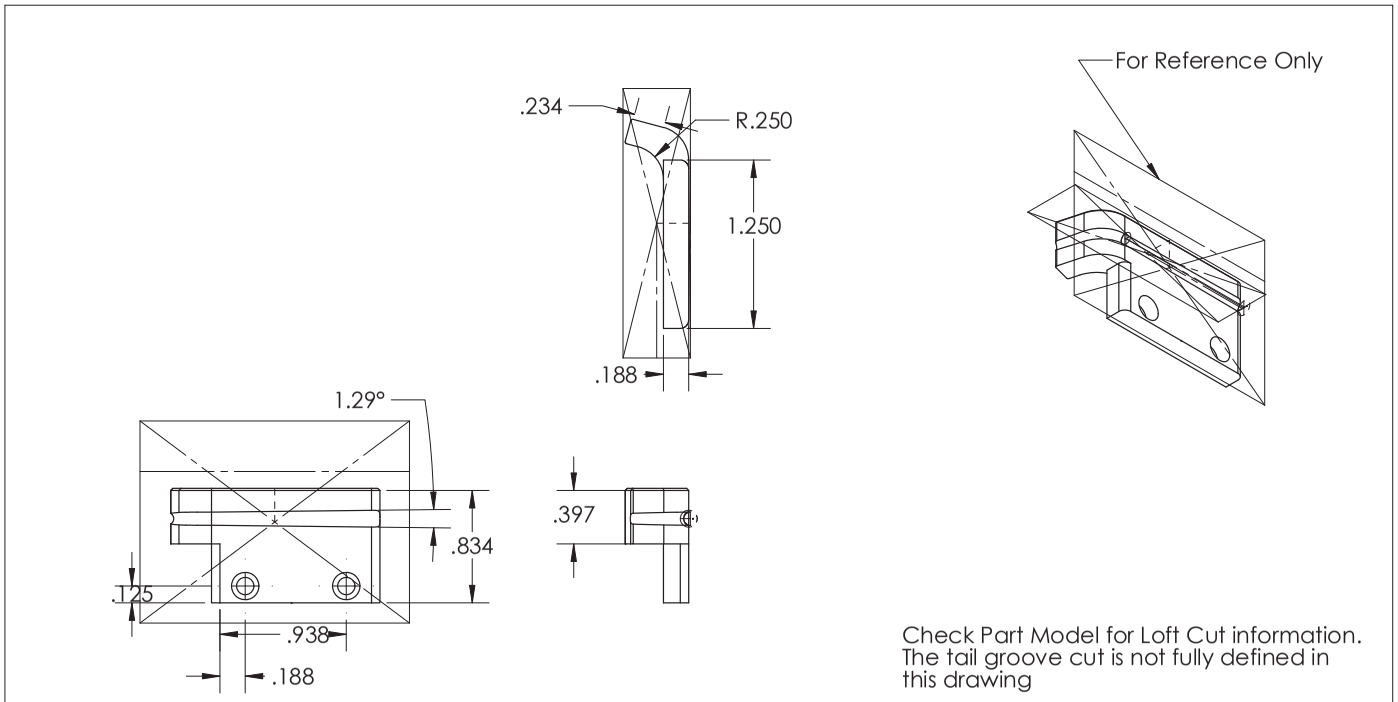
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER LAB(LOS

MATERIAL
Delrin 2700 NC010, Low Viscosity Acetal Copolymer (SS)

FINISH

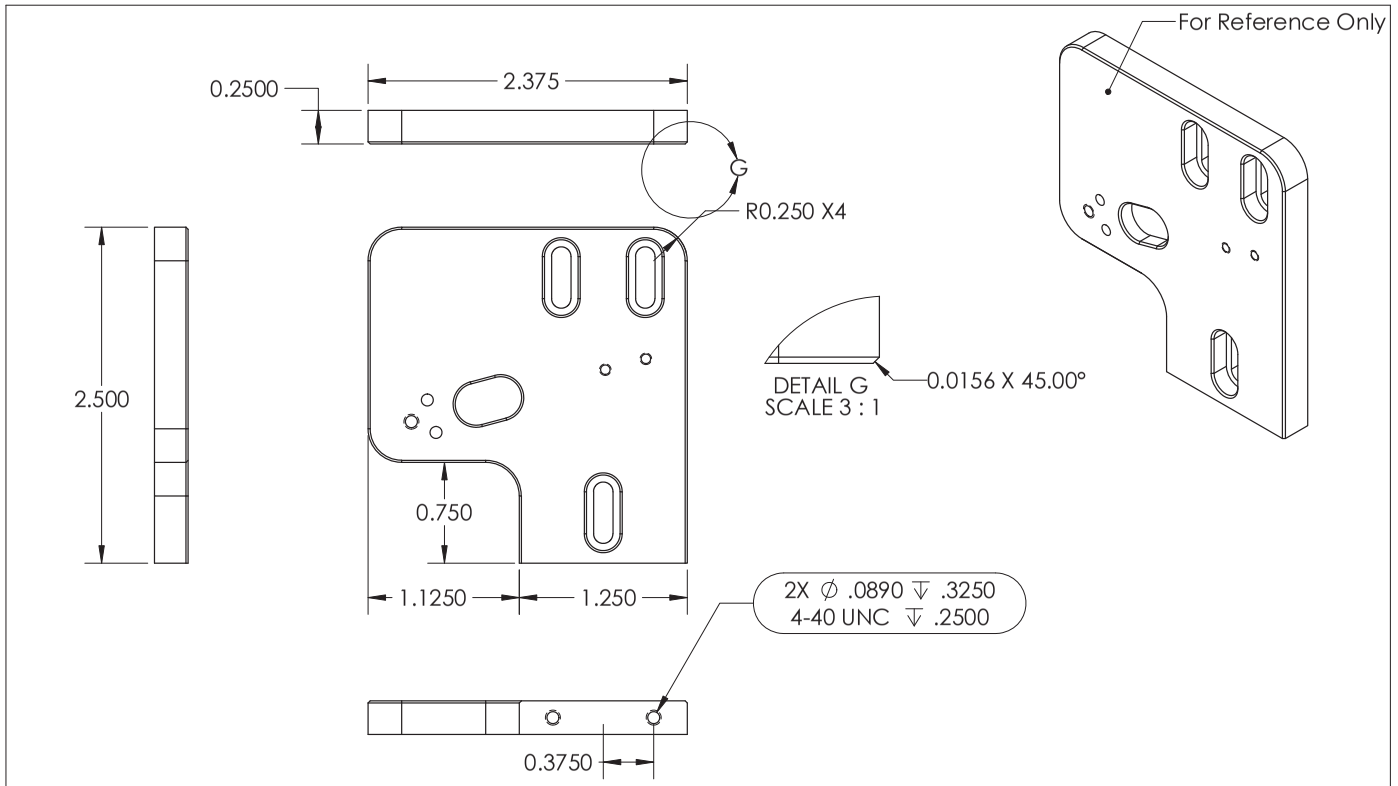
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES
ANGULAR ±0.2°
FRACTIONAL ±0.005"
.XX ±0.010"
.XXX ±0.005"
.XXXX ±0.001"

4 3 2 1



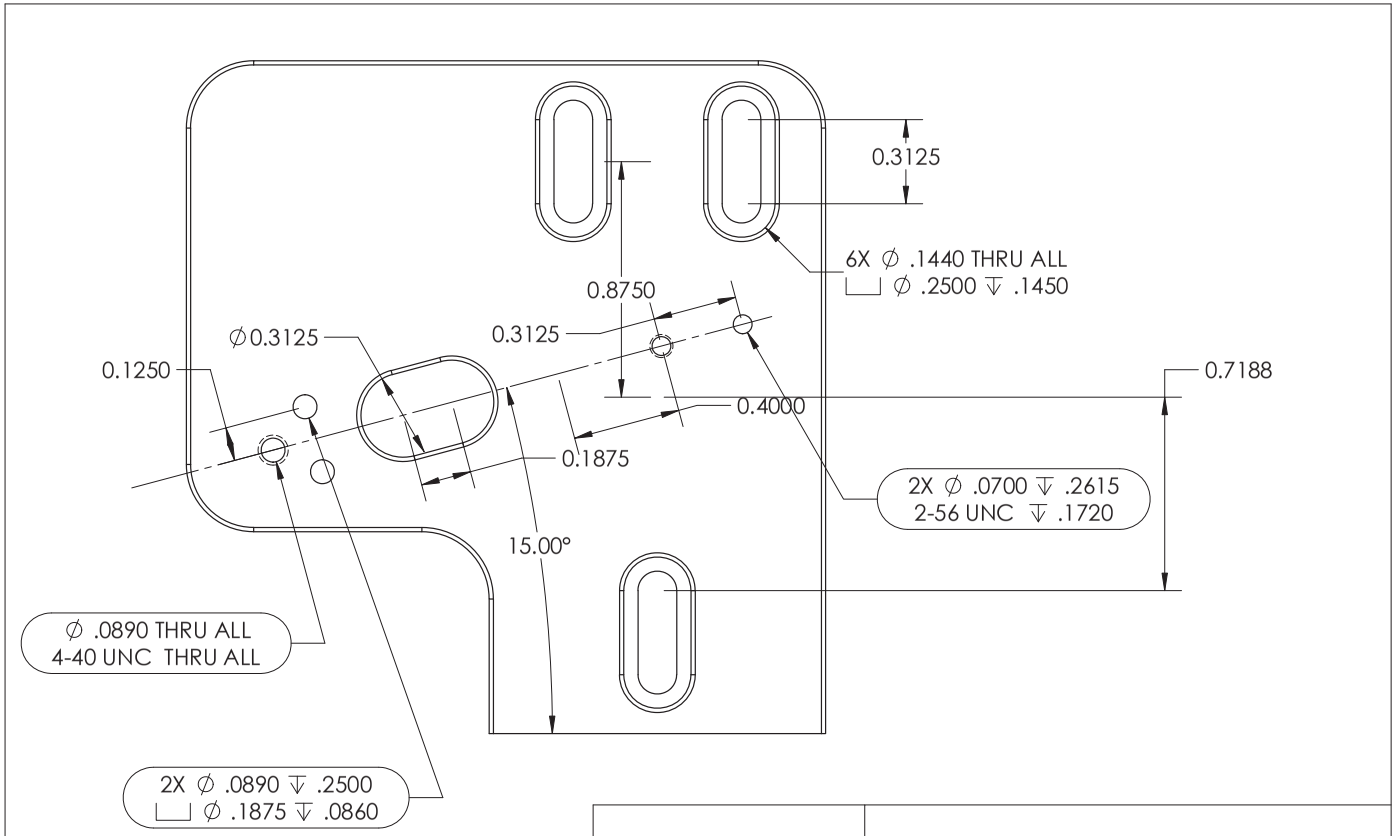
PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER LAB(LOS	MATERIAL Delrin FINISH Mill	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	VAS TITLE: Tail Support SIZE A | PART NO. Tail Support Guide | REV A DO NOT SCALE DRAWING | SCALE: 1:1 | WEIGHT: 0.01 | SHEET 1 OF 1

1 4 3 2 1



PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND IMAGING SCIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER OF THE ABOVE MENTIONED LABORATORIES AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES IS PROHIBITED.	MATERIAL 6061 Alloy FINISH Mill	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	COMMENTS:	TITLE: Tail Support Mount SIZE A | PART NO. Tail Support Mount | REV A DO NOT SCALE DRAWING | SCALE: 1:1 | WEIGHT: 0.11 | SHEET 1 OF 2

5 4 3 2 1



MATERIAL: 6061 Alloy

FINISH: Mill

UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES
ANGULAR ±0.2°
FRACTIONAL ±0.005"
.XX ±0.010"
.XXX ±0.005"
.XXXX ±0.001"

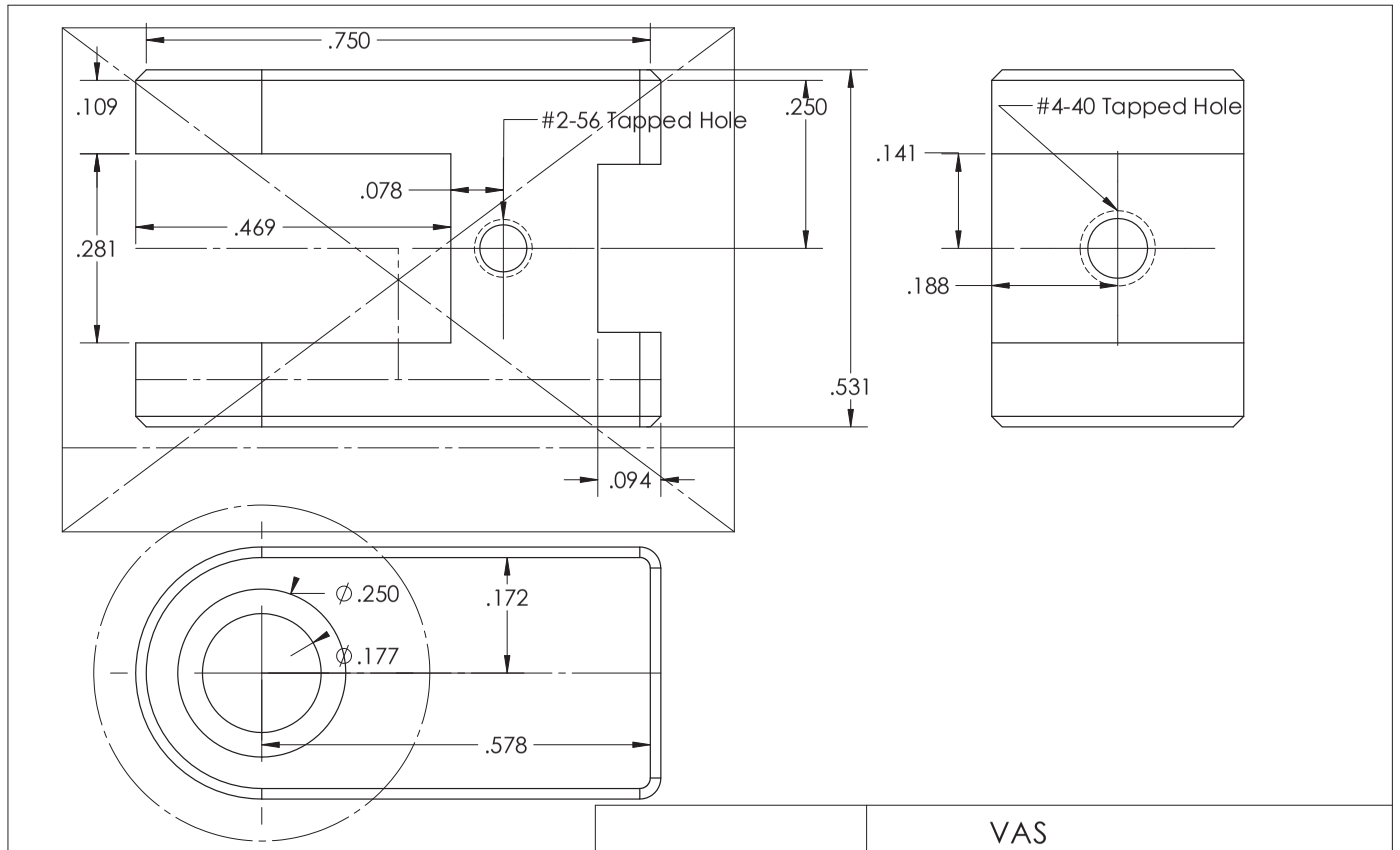
COMMENTS:

TITLE:
Tail Support Mount

SIZE	PART NO.	REV
A	Tail Support Mount	A

DO NOT SCALE DRAWING | SCALE: 2:1 | WEIGHT: 0.11 | SHEET 2 OF 2

5 4 3 2 1



			VAS
			TITLE: Tail Clamp Wheel Housing
			SIZE A | PART NO. Frictin Wheel Housing | REV A
			DO NOT SCALE DRAWING | SCALE: 5:1 | WEIGHT: 0.00 | SHEET 1 OF 1

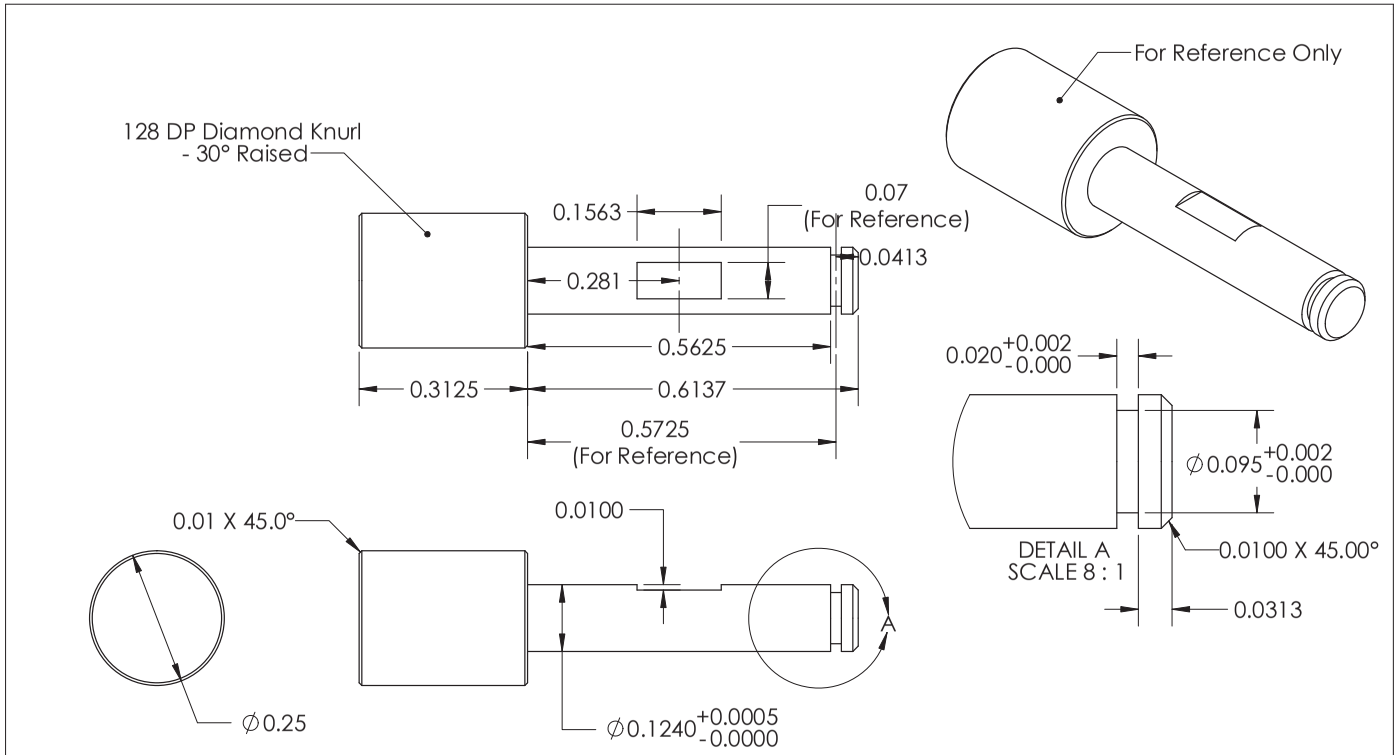
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER LAB(LOS

MATERIAL

FINISH

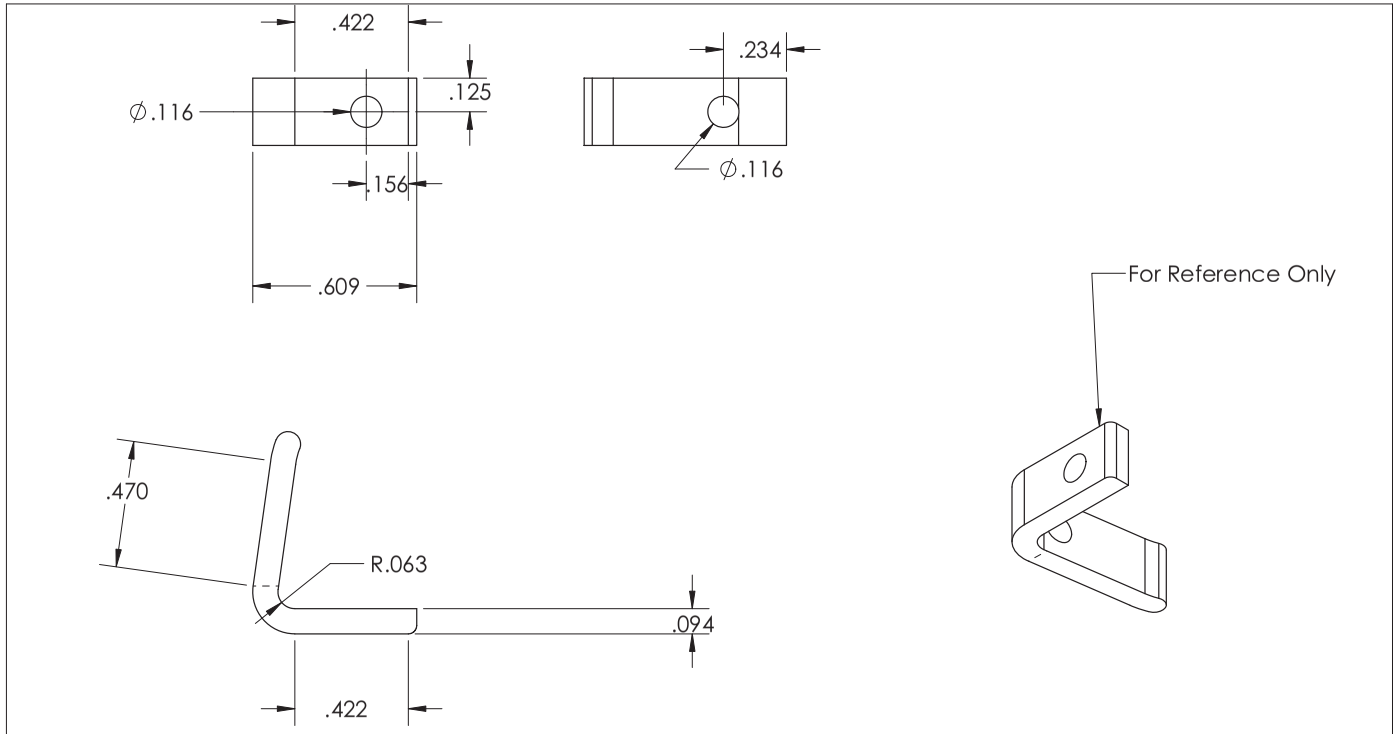
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES
ANGULAR ±0.2°
FRACTIONAL ±0.005"
.XX ±0.010"
.XXX ±0.005"
.XXXX ±0.001"

4 3 2 1



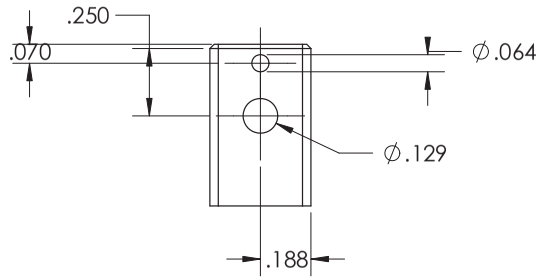
PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND IMAGING SCIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER OF THE ABOVE MENTIONED LABORATORIES AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES IS PROHIBITED.	MATERIAL AISI 304 FINISH Lathe, 8-17µin	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	COMMENTS:	TITLE: E-Clip Shaft (0.1240" OD) SIZE A | PART NO. E-Clip Shaft_0.1240OD_91746A545 | REV A DO NOT SCALE DRAWING | SCALE: 4:1 | WEIGHT: 0.01 | SHEET 1 OF 1

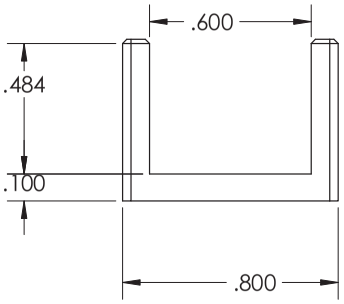
5 4 3 2 1

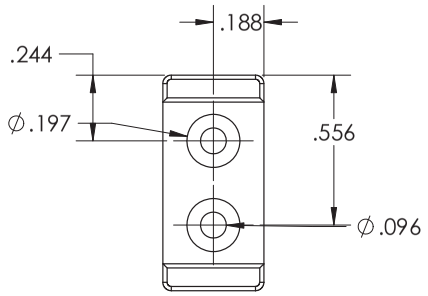


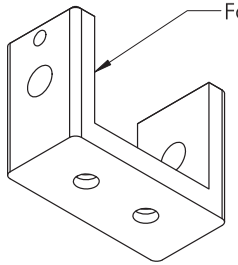
			VAS
			TITLE: Wheel Brake
PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER LABO LOS	MATERIAL: Plastic FINISH: Mill	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	SIZE: A PART NO.: Frictin Wheel Housing Brake REV: A DO NOT SCALE DRAWING | SCALE: 2:1 | WEIGHT: 0.00 | SHEET 1 OF 1

1 4 3 2 1

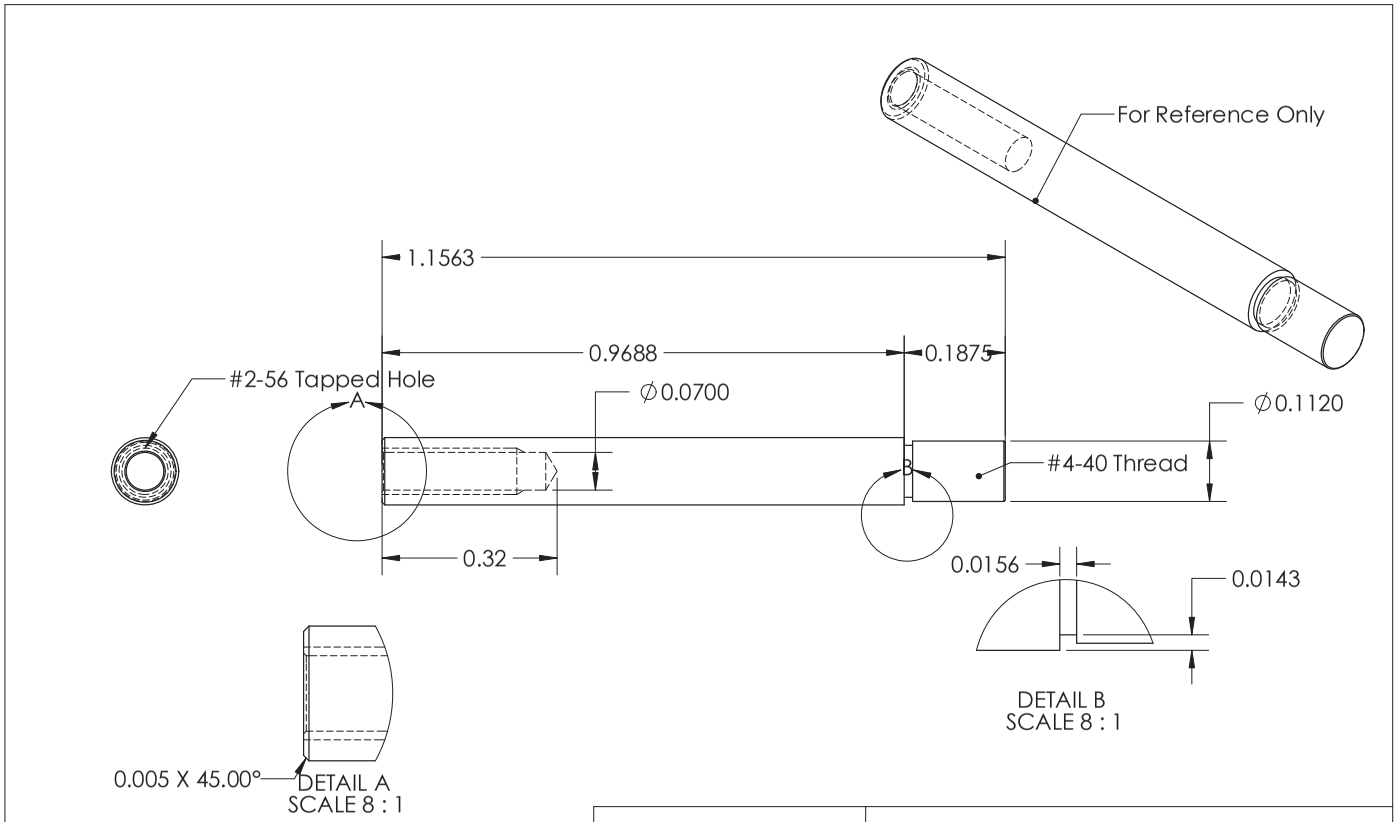






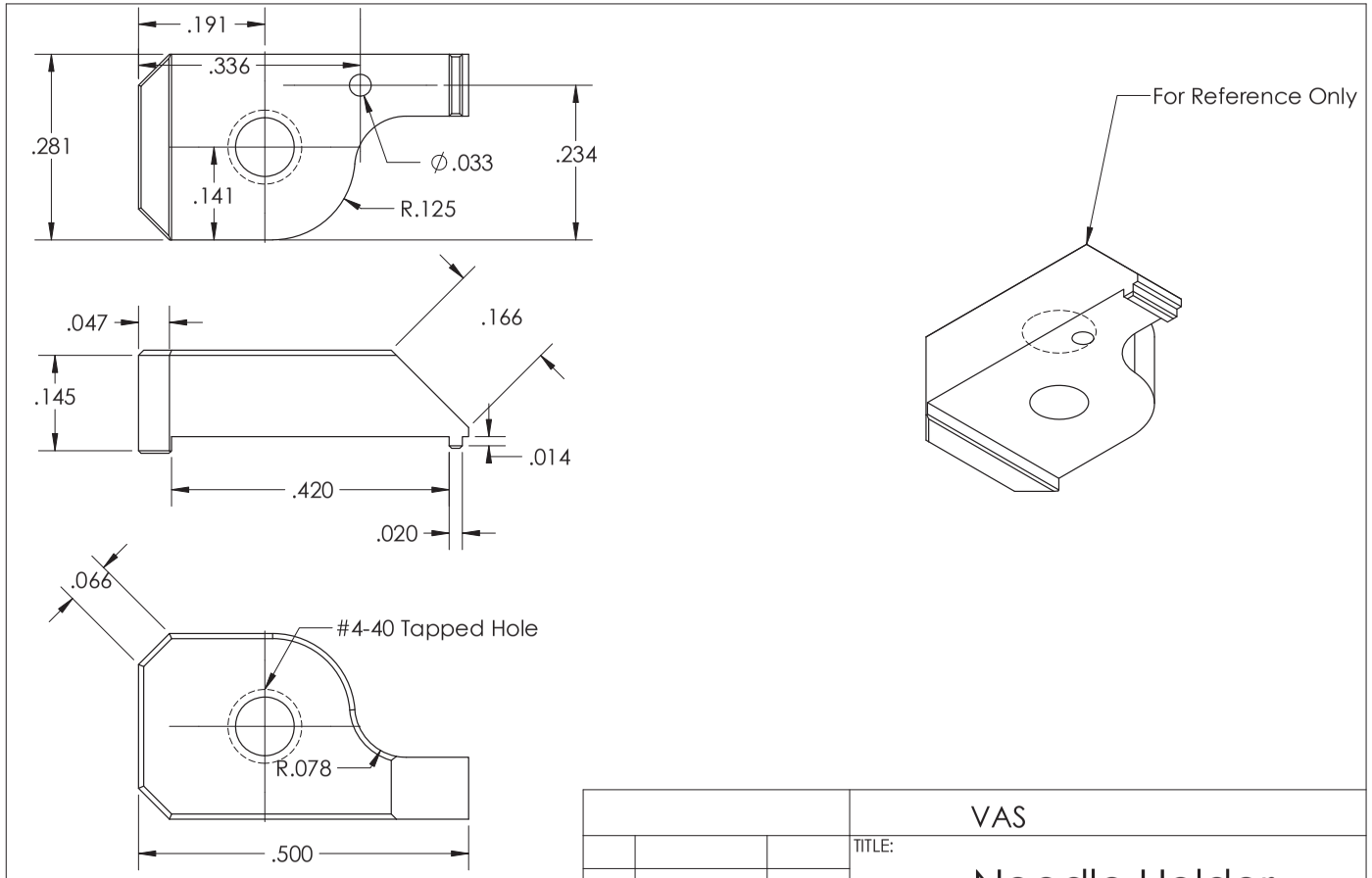


			VAS
			TITLE: Tail Clamp Spring Housing
PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER LABO LOS	MATERIAL Brass FINISH	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	SIZE A | PART NO. Friction Spring Mandrel Housing | REV A DO NOT SCALE DRAWING | SCALE: 2:1 | WEIGHT: 0.02 | SHEET 1 OF 1
4	3	2	1



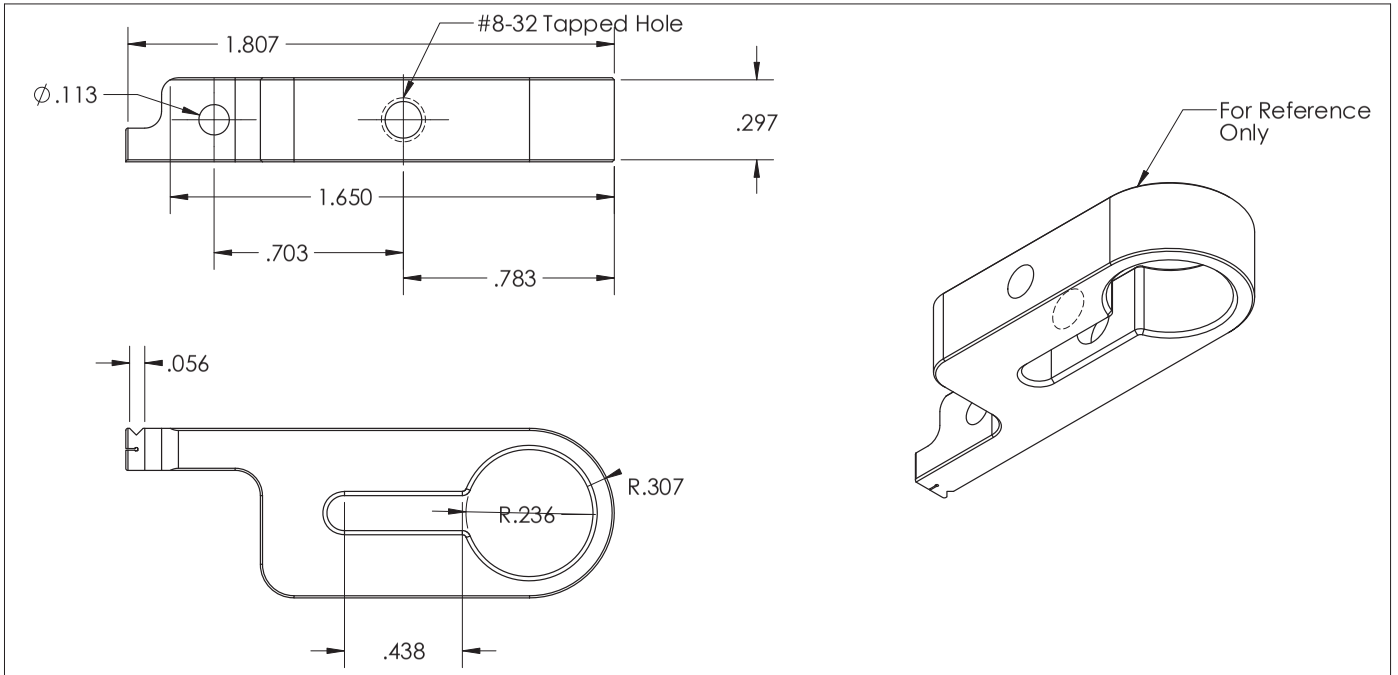
PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND IMAGING SCIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER OF THE ABOVE MENTIONED LABORATORIES AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES IS PROHIBITED.	MATERIAL AISI 304 FINISH Lathe, 8-17µin	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	COMMENTS:	TITLE: Friction Spring Madrel SIZE A | PART NO. Friction Spring Mandrel | REV A DO NOT SCALE DRAWING | SCALE: 4:1 | WEIGHT: 0.00 | SHEET 1 OF 1

5 4 3 2 1



PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER LABO LOS	MATERIAL 6061 Alloy FINISH Mill	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	VAS TITLE: Needle Holder SIZE A | PART NO. Needle Holder | REV A DO NOT SCALE DRAWING | SCALE: 5:1 | WEIGHT: 0.00 | SHEET 1 OF 1

4 3 2 1



PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER LABO LOS	MATERIAL 6061 Alloy FINISH Mill	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"

VAS

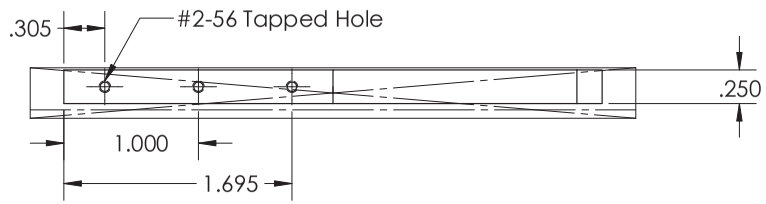
TITLE:

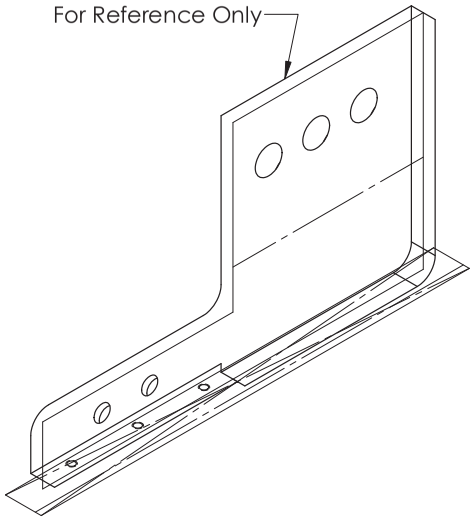
X Motor Needle Adaptor

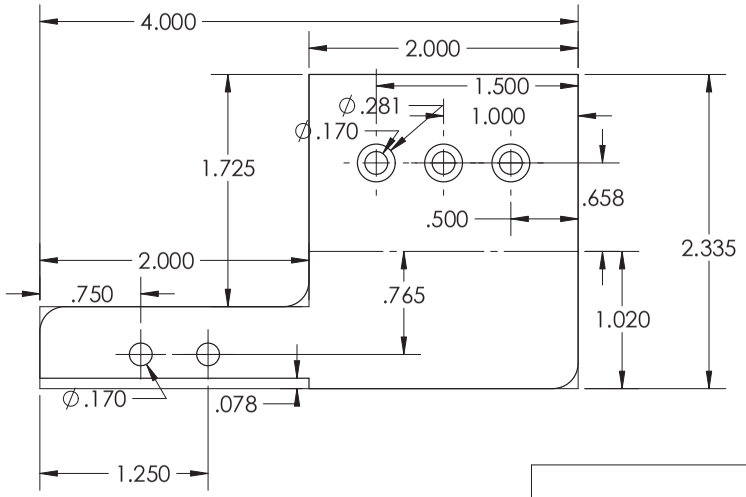
SIZE	PART NO.	REV
A	DynaOptic Adaptor_Needle Holder	A

DO NOT SCALE DRAWING	SCALE: 2:1	WEIGHT: 0.01	SHEET 1 OF 1

1 4 3 2 1







VAS

TITLE:

Motor Z & Theta Adaptor

SIZE	PART NO.	REV
A	Siskiyou-Newport Adaptor Plate	A

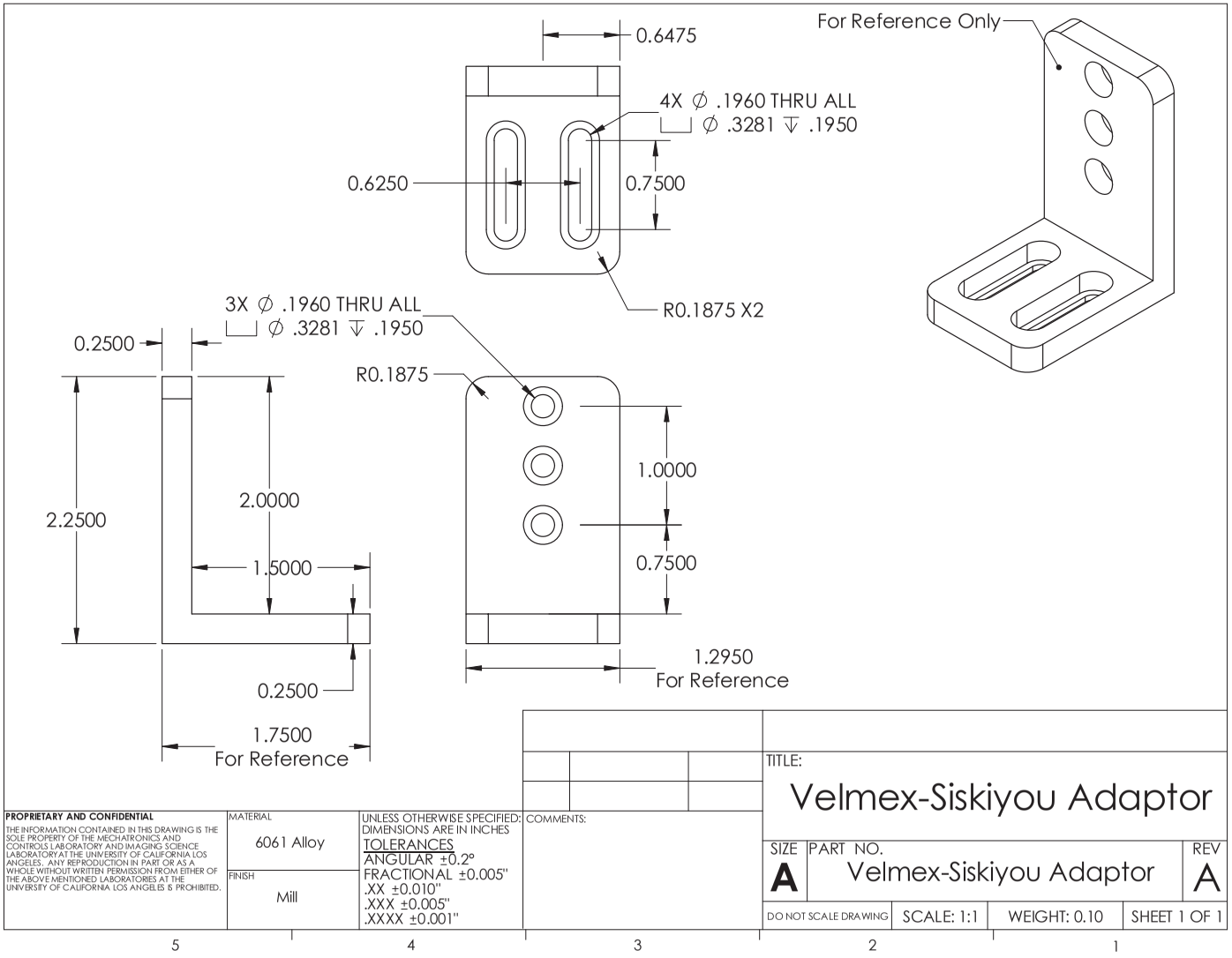
DO NOT SCALE DRAWING	SCALE: 1:1	WEIGHT: 0.05	SHEET 1 OF 1

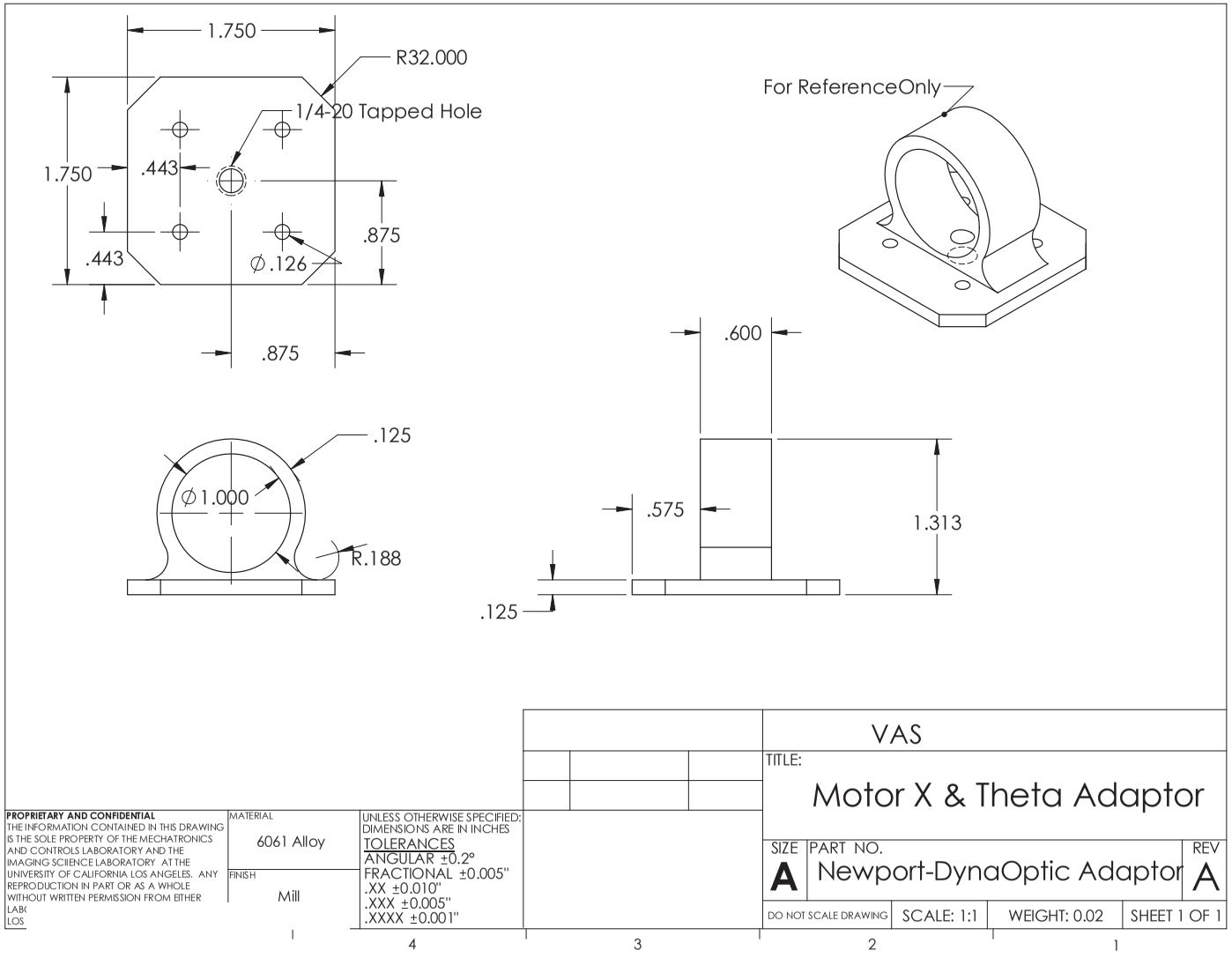
MATERIAL: 6061 Alloy

FINISH: Mill

UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES
ANGULAR ±0.2°
FRACTIONAL ±0.005"
.XX ±0.010"
.XXX ±0.005"
.XXXX ±0.001"

4 3 2 1


For Reference Only
0.6475
4X ⌀ .1960 THRU ALL
⌴ ⌀ .3281 ↧ .1950
0.6250
0.7500
R0.1875 X2
3X ⌀ .1960 THRU ALL
⌴ ⌀ .3281 ↧ .1950
0.2500
R0.1875
2.2500
2.0000
1.5000
1.0000
0.7500
0.2500
1.7500
For Reference
1.2950
For Reference
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND IMAGING SCIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER OF THE ABOVE MENTIONED LABORATORIES AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES IS PROHIBITED.
MATERIAL
6061 Alloy
FINISH
Mill
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES
ANGULAR ±0.2°
FRACTIONAL ±0.005"
.XX ±0.010"
.XXX ±0.005"
.XXXX ±0.001"
COMMENTS:
TITLE:
Velmex-Siskiyou Adaptor
SIZE
A
PART NO.
Velmex-Siskiyou Adaptor
REV
A
DO NOT SCALE DRAWING
SCALE: 1:1
WEIGHT: 0.10
SHEET 1 OF 1
5
4
3
2
1



			VAS
			TITLE: Motor X & Theta Adaptor
PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF THE MECHATRONICS AND CONTROLS LABORATORY AND THE IMAGING SCIIENCE LABORATORY AT THE UNIVERSITY OF CALIFORNIA LOS ANGELES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT WRITTEN PERMISSION FROM EITHER LAB(LOS	MATERIAL: 6061 Alloy FINISH: Mill	UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES TOLERANCES ANGULAR ±0.2° FRACTIONAL ±0.005" .XX ±0.010" .XXX ±0.005" .XXXX ±0.001"	SIZE: A PART NO.: Newport-DynaOptic Adaptor REV: A DO NOT SCALE DRAWING | SCALE: 1:1 | WEIGHT: 0.02 | SHEET 1 OF 1

1 4 3 2 1

APPENDIX B

VAS Arduino Code

```
/*****************************************************************************
 **
 **  LVFA_Firmware - Provides Basic Arduino Sketch
 **For Interfacing With LabVIEW.
 **
 **  Written By:    Sam Kristoff - National Instruments
 **  Written On:    November 2010
 **  Last Updated:  Dec 2011 - Kevin Fort - National Instruments
 **
 **  This File May Be Modified And Re-Distributed Freely.
 ** Original File Content
 **  Written By Sam Kristoff And Available At www.ni.com/arduino.
 **
** Modified by Kevin Chu for the VAS Project
 *****************************************************************************/


/*****************************************************************************
 **
 ** Includes.
 **
 *****************************************************************************/
```

```
// Standard includes.  These should always be included.
#include <Wire.h>
#include <SPI.h>
#include <Servo.h>
#include "TimerOne.h"
#include "LabVIEWInterface.h"
#include "kevMotor.h"
#include "kevFilter.h"

// motor initializtion
kevMotor kMot1;
kevMotor kMot2;
kevMotor kMot3;

// filter setup
kevFilter kFilt1;


// servo initializations
Servo myServo1;
Servo myServo2;
Servo myServo3;

// motor 1 definitions
#define mot1EncAPin 18
#define mot1EncBPin 32
#define mot1OuterPin 22
#define mot1InnerPin 24
#define mot1CmdPin 2
```

```
#define mot1CmdPin_check 52

// motor 2 definitions
#define mot2EncAPin 19
#define mot2EncBPin 34
#define mot2OuterPin 26
#define mot2InnerPin 28
#define mot2CmdPin 3
#define mot2CmdPin_check 9

// motor 3 definitions
#define mot3EncAPin 20
#define mot3EncBPin 36
#define mot3OuterPin 23
#define mot3InnerPin 25
#define mot3CmdPin 4
#define mot3CmdPin_check 10

// Pressure Reading definitions

#define pressurePin 0

//volatile long kevEnc=0;
//volatile int miniKevEnc=0;
//int pos = 0;
//int temp = 0;
//int bias = 90;
//int ref1 = 0;
//int myMax = 30;
```

```
//int kinv = 4;
//long errorSum = 0;
//long kevinHugs=0;
//int innerCnt1 = 0;
//int outerCnt1 = 0;
//long pastError = 0;

/*********************************************************
 **  setup()
 **
 **  Initialize the Arduino and setup serial communication.
 **
 **  Input:  None
 **  Output: None
 *********************************************************/
void setup()
{
  // Initialize Serial Port With The Default Baud Rate
  syncLV();
  // motor 1 setup
  pinMode(mot1EncAPin,INPUT);
  pinMode(mot1EncBPin,INPUT);
  pinMode(mot1OuterPin,INPUT);
  pinMode(mot1InnerPin,INPUT);
  pinMode(mot1CmdPin_check,OUTPUT);
  digitalWrite(mot1EncAPin,HIGH);
  digitalWrite(mot1EncBPin,HIGH);
  digitalWrite(mot1OuterPin,HIGH);
  digitalWrite(mot1InnerPin,HIGH);
```

```
myServo1.attach(mot1CmdPin);
//  myServo1.attach(mot1CmdPin);
attachInterrupt(5, cookie1, RISING);
kMot1.setPID(20,10,0);

pinMode(51,OUTPUT);
pinMode(53,OUTPUT);

// motor 2 setup
pinMode(mot2EncAPin,INPUT);
pinMode(mot2EncBPin,INPUT);
pinMode(mot2OuterPin,INPUT);
pinMode(mot2InnerPin,INPUT);
digitalWrite(mot2EncAPin,HIGH);
digitalWrite(mot2EncBPin,HIGH);
digitalWrite(mot2OuterPin,HIGH);
digitalWrite(mot2InnerPin,HIGH);
attachInterrupt(4, cookie2, RISING);
myServo2.attach(mot2CmdPin);
kMot2.setPID(20,10,0);

// motor 3 setup
pinMode(mot3EncAPin,INPUT);
pinMode(mot3EncBPin,INPUT);
pinMode(mot3OuterPin,INPUT);
pinMode(mot3InnerPin,INPUT);
digitalWrite(mot3EncAPin,HIGH);
digitalWrite(mot3EncBPin,HIGH);
digitalWrite(mot3OuterPin,HIGH);
```

```
  digitalWrite(mot3InnerPin,HIGH);
  myServo3.attach(mot3CmdPin);
  attachInterrupt(3, cookie3, RISING);
  kMot3.setPID(20,10,0);


  // my debug pin
  pinMode(37,OUTPUT);


  //  attachInterrupt(1, cookie, CHANGE);


  // Place your custom setup code here

  // magic timer
  //  Timer3.initialize(1000);
  //  Timer3.attachInterrupt(cheese);
  Timer1.initialize(3000);
  Timer1.attachInterrupt(cheese);

  // Filter Setup
  kFilt1.initialize(89);

}


/*****************************************************************************
 **  loop()
```

```
 **
 **  The main loop.  This loop runs continuously on the Arduino.  It
 **  receives and processes serial commands from LabVIEW.
 **
 **  Input:  None
 **  Output: None
 ************************************************************************/
void loop()
{
  // Check for commands from LabVIEW and process them.
  checkForCommand();
  //  digitalWrite(51,digitalRead(mot1EncAPin));
  //    digitalWrite(53,digitalRead(mot1EncBPin));
}



void cookie1(){
  //  int foo;
  //  for (int ii = 0; ii<5;ii++) {foo++;};
  kMot1.updateEnc(digitalRead(mot1EncBPin));
  //  kMot1.updateEnc(LOW);
}



void cookie2(){
  kMot2.updateEnc(digitalRead(mot2EncBPin));
}
```

```
void cookie3(){
  kMot3.updateEnc(digitalRead(mot3EncBPin));
}


void cheese(){

  //myServo1.write(kMot1.runLoop(digitalRead(mot1InnerPin),
   //digitalRead(mot1OuterPin)));
  //  myServo2.writeMicroseconds(2000);
  //  myServo1.writeMicroseconds(1500);
  //  myServo3.writeMicroseconds(1500);

  //  myServo1.write(90);
  //  myServo2.write(90);
  //  myServo3.write(90);

  //  myServo1.writeMicroseconds(kMot1.reference);
  //  myServo2.writeMicroseconds(kMot2.reference);
  //  myServo3.writeMicroseconds(kMot3.reference);

  //   myServo2.writeMicroseconds(kMot2.runLoop(digitalRead(mot2InnerPin),
   //digitalRead(mot2OuterPin)));
  if(kFilt1.insertAndTrig(analogRead(pressurePin)))
  {
    kMot3.stopNow();
  }

  // if (filter_check || filter_check2)
```

```
  // {
  //   filter_check==true;
  //   kMot3.stopNow();
  // }

  myServo3.write(kMot3.runLoop(0,0));
  myServo2.write(kMot2.runLoop(digitalRead(mot2InnerPin),
   digitalRead(mot2OuterPin)));
  myServo1.write(kMot1.runLoop(digitalRead(mot1InnerPin),
   digitalRead(mot1OuterPin)));


  digitalWrite(mot1CmdPin_check, mot1CmdPin);
  // }

  //  // if it triggers stop yay?


  //myservo.write(kevinHugs+bias);
  //Serial1.write(kevinHugs);
}
```

B.1 timerOne.h

```
****Timer1.h
 *  Interrupt and PWM utilities for 16 bit Timer1 on ATmega168/328
 *  Original code by Jesse Tane for http://labs.ideo.com August 2008
 *  Modified March 2009 by Jrme Despatis and Jesse Tane for
```

```
**ATmega328 support
*  Modified June 2009 by Michael Polli and Jesse Tane to fix a bug
** in setPeriod() which caused the timer to stop
*  Modified June 2011 by Lex Talionis to add a function to
**read the timer
*  Modified Oct 2011 by Andrew Richards to avoid certain problems:
*  - Add (long) assignments and casts to TimerOne::read() to ensure
** calculations involving tmp, ICR1 and TCNT1 aren't truncated
*  - Ensure 16 bit registers accesses are atomic - run with interrupts
**disabled when accessing
*  - Remove global enable of interrupts (sei())- could be running within
** an interrupt routine)
*  - Disable interrupts whilst TCTN1 == 0.  Datasheet vague on this,
 but experiment shows that overflow interrupt
*    flag gets set whilst TCNT1 == 0, resulting in a phantom interrupt.
** Could just set to 1, but gets inaccurate
*    at very short durations
*  - startBottom() added to start counter at 0 and handle all interrupt
** enabling.
*  - start() amended to enable interrupts
*  - restart() amended to point at startBottom()
* Modiied 7:26 PM Sunday, October 09, 2011 by Lex Talionis
*  - renamed start() to resume() to reflect it's actual role
*  - renamed startBottom() to start(). This breaks some old code
**that expects start to continue counting where it left off
*
*  This program is free software: you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation, either version 3 of the License, or
```

```
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program.  If not, see <http://www.gnu.org/licenses/>.
 *
 *  See Google Code project http://code.google.com/p/arduino-timerone/
 ** for latest
 */
#ifndef TIMERONE_h
#define TIMERONE_h

#include <avr/io.h>
#include <avr/interrupt.h>

#define RESOLUTION 65536    // Timer1 is 16 bit

class TimerOne
{
  public:

    // properties
    unsigned int pwmPeriod;
    unsigned char clockSelectBits;
char oldSREG; // To hold Status Register while ints disabled
```

```
    // methods
    void initialize(long microseconds=1000000);
    void start();
    void stop();
    void restart();
void resume();
unsigned long read();
    void pwm(char pin, int duty, long microseconds=-1);
    void disablePwm(char pin);
    void attachInterrupt(void (*isr)(), long microseconds=-1);
    void detachInterrupt();
    void setPeriod(long microseconds);
    void setPwmDuty(char pin, int duty);
    void (*isrCallback)();
};

extern TimerOne Timer1;
#endif
```

B.2 LabVIEWInterface.h

```
/*********************************************************************
 **
 **
 **  LVFA_Firmware - Provides Functions For Interfacing With
 ** The Arduino Uno
 **
```

```
 **  Written By:    Sam Kristoff - National Instruments
 **  Written On:    November 2010
 **  Last Updated:  Dec 2011 - Kevin Fort - National Instruments
 **
 **  This File May Be Modified And Re-Distributed Freely.
 ** Original File Content
 **  Written By Sam Kristoff And Available At www.ni.com/arduino.
 **Modified by Kevin Chu for the VAS project
 ******************************************************************************/


/******************************************************************************
**  Define Constants
**
**  Define directives providing meaningful names for constant values.
******************************************************************************/

#define FIRMWARE_MAJOR 02
#define FIRMWARE_MINOR 00
#if defined(__AVR_ATmega1280__) || defined(__AVR_ATmega2560__)
#define DEFAULTBAUDRATE 9600    // Defines The Default Serial Baud Rate
//(This must match the baud rate specifid in LabVIEW)
#else
#define DEFAULTBAUDRATE 9600
#endif
#define MODE_DEFAULT 0     // Defines Arduino Modes (Currently Not Used)
#define COMMANDLENGTH 15    // Defines The Number Of Bytes In A Single
// LabVIEW Command (This must match the packet size specifid in LabVIEW)
// #define STEPPER_SUPPORT 0         // Defines Whether The
```

```
//Stepper Library Is Included -Comment This Line To Exclude Stepper Support


// Declare Variables
unsigned char currentCommand[COMMANDLENGTH];    // The Current
//Command For The Arduino To Process
//Globals for continuous aquisition
unsigned char acqMode;
unsigned char contAcqPin;
float contAcqSpeed;
float acquisitionPeriod;
float iterationsFlt;
int iterations;
float delayTime;


/*****************************************************************
**  syncLV
**
**  Synchronizes with LabVIEW and sends info about
**the board and firmware (Unimplemented)
**
**  Input:  None
**  Output: None
*****************************************************************/
void syncLV();

/*****************************************************************
**  setMode
```

```
**
**  Sets the mode of the Arduino (Reserved For Future Use)
**
**  Input:  Int - Mode
**  Output: None
*****************************************************************/
void setMode(int mode);

/*****************************************************************
**  checkForCommand
**
**  Checks for new commands from LabVIEW and processes
**them if any exists.
**
**  Input:  None
**  Output: 1 - Command received and processed
**          0 - No new command
*****************************************************************/
int checkForCommand(void);

/*****************************************************************
**  processCommand
**
**  Processes a given command
**
**  Input:  command of COMMANDLENGTH bytes
**  Output: 1 - Command received and processed
**          0 - No new command
*****************************************************************/
```

```
void processCommand(unsigned char command[]);

/*****************************************************************
**  writeDigitalPort
**
**  Write values to DIO pins 0 - 13.
**Pins must first be configured as outputs.
**
**  Input:  Command containing digital port data
**  Output: None
*****************************************************************/
void writeDigitalPort(unsigned char command[]);

/*****************************************************************
**  analogReadPort
**
** Reads all 6 analog input ports, builds 8 byte packet, send via RS232.
**
**  Input:  None
**  Output: None
*****************************************************************/
void analogReadPort();

/*****************************************************************
**  sevenSegment_Config
**
**  Configure digital I/O pins to use for seven segment display.
**Pins are stored in sevenSegmentPins array.
**
```

```
**  Input:  Pins to use for seven segment LED [A, B, C, D, E, F, G, DP]
**  Output: None
*************************************************************************/
void sevenSegment_Config(unsigned char command[]);

/*************************************************************************
**  sevenSegment_Write
**
**  Write values to sevenSegment display.
** Must first use sevenSegment_Configure
**
**  Input:  Eight values to write to seven segment display
**  Output: None
*************************************************************************/
void sevenSegment_Write(unsigned char command[]);

/*************************************************************
**  spi_setClockDivider
**
**  Set the SPI Clock Divisor
**
**  Input:  SPI Clock Divider 2, 4, 8, 16, 32, 64, 128
**  Output: None
*************************************************************/
void spi_setClockDivider(unsigned char divider);

/*************************************************************
**  spi_sendReceive
**
```

```
**  Sens / Receive SPI Data
**
**  Input:  Command Packet
**  Output: None (This command sends one serail byte back to LV
**for each data byte.
*****************************************************************/
void spi_sendReceive(unsigned char command[]);

/*****************************************************************
**  checksum_Compute
**
**  Compute Packet Checksum
**
**  Input:  Command Packet
**  Output: Char Checksum Value
*****************************************************************/
unsigned char checksum_Compute(unsigned char command[]);

/*****************************************************************
**  checksum_Test
**
**  Compute Packet Checksum And Test Against Included Checksum
**
**  Input:  Command Packet
**  Output: 0 If Checksums Are Equal, Else 1
*****************************************************************/
int checksum_Test(unsigned char command[]);

/*****************************************************************
```

```
**  AccelStepper_Write
**
**  Parse command packet and write speed, direction, and number
** steps to travel
**
**  Input:  Command Packet
**  Output: None
*************************************************************/
void AccelStepper_Write(unsigned char command[]);
/*************************************************************
**  SampleContinuosly
**
**  Returns several analog input points at once.
**
**  Input:  void
**  Output: void
*************************************************************/
void sampleContinously(void);

/*************************************************************
**  finiteAcquisition
**
**  Returns the number of samples specified at the rate specified.
**
**  Input:  pin to sampe on, speed to sample at, number of samples
**  Output: void
*************************************************************/
void finiteAcquisition(int analogPin, float acquisitionSpeed,
int numberOfSamples );
```

```
/*************************************************************
**  lcd_print
**
**  Prints Data to the LCD With The Given Base
**
**  Input:  Command Packet
**  Output: None
*************************************************************/
void lcd_print(unsigned char command[]);


/*********************************************************
**  kevWriteLong
**
**  sends a long in byte using Serial.write
**
**  Input:  long integer
**  Output: None
*********************************************************/
void kevWriteLong(long number);
long kevReadLong(byte byte0, byte byte1, byte byte2,byte byte3);
```

B.3 kevMotor.h

```
//Written by Kevin Chu for the VAS Project
//class TimerThree
//{
//  public:
```

```
//
//    // properties
//    unsigned int pwmPeriod;
//    unsigned char clockSelectBits;
//
//    // methods
//    void initialize(long microseconds=1000000);
//    void start();
//    void stop();
//    void restart();
//    void pwm(char pin, int duty, long microseconds=-1);
//    void disablePwm(char pin);
//    void attachInterrupt(void (*isr)(), long microseconds=-1);
//    void detachInterrupt();
//    void setPeriod(long microseconds);
//    void setPwmDuty(char pin, int duty);
//    void (*isrCallback)();
//};


#define MAX_CMD 50
#define BIAS 90


class kevMotor
{
  public:

    // constructor
```

```
kevMotor();
// methods
void initialize();
void setPID(int nKp,int nKd,int nKi);
void setReference(long newReference);
void setVelocity(long newVelocity);
void stopNow();
void jumpBy(long jumpVal);
// running methods
void updateEnc(char quadBPinVal);
long runLoop(char innerVal, char outerVal);



// data
long reference;
long newReference;
long reference_check;
long reference_stop;
long curPosition;
long reference_resume;
long velocity_resume;
long velocity;
long newVelocity;
long velocity_check;
long velocity_stop;
long pastError;
long pastError2;
long intError;
```

```
    long ki;
    long kp;
    long kd;
    char innerCnt;
    char outerCnt;
    long command;
    boolean check_filt;
    boolean check_filt2;
};

kevMotor::kevMotor()
{
}

void kevMotor::initialize()
{
}

void kevMotor::setPID(int nKp,int nKd,int nKi)
{
  kp = nKp;
  ki = nKi;
  kd = nKd;
}

void kevMotor::setReference(long newReference)
{
  if (check_filt){// && (abs(newReference - reference_resume)==0)){
  reference=reference_stop;
```

```
  check_filt2=true;
      if(newReference != reference_resume){
        check_filt=false;
        check_filt2=false;
      }
  }else{// if (reference_resume==newReference){
  reference = newReference;
  reference_resume = newReference;
  }
  velocity = 0;
}

void kevMotor::stopNow()
{
  if(!check_filt2){
  reference_stop = curPosition*16;
  reference_resume = reference;
  velocity_stop = 0;
  check_filt = true;
  check_filt2 = true;
  //long newVelocity = 0;
  }
}

void kevMotor::jumpBy(long jumpVal)
{
  reference = curPosition+jumpVal;
  velocity = 0;
}
```

```
void kevMotor::setVelocity(long newVelocity)
{
//  reference = curPosition;
  if (check_filt){
  velocity=0;
  check_filt2=true;
  reference=reference_stop;
      if(newVelocity != velocity_resume){
      check_filt=false;
      check_filt2=false;
      }
  }else{
  velocity = newVelocity;
  velocity_resume = newVelocity;
  }
}

void kevMotor::updateEnc(char quadBPinVal)
{

  if (quadBPinVal) {
    curPosition++;
  } else {
    curPosition--;
  }
}
```

```
long kevMotor::runLoop(char innerVal, char outerVal)
{
  reference += velocity/4;
  innerCnt += innerVal*2-1;
  outerCnt += outerVal*2-1;
  innerCnt = constrain(innerCnt,0,20);
  outerCnt = constrain(outerCnt,0,20);
  long error = (reference/16-curPosition);
  intError += error;
  intError = constrain(intError,-1000,1000);

  command = (error*kp+(error-pastError2)*kd*128+intError*ki/16);

  command = constrain(command,-MAX_CMD,MAX_CMD);
//  if (outerCnt>=10) {
//    command = min(command,0);
//  }
//  if (innerCnt>=10) {
//    command = max(command, 0);
//  }

  pastError2 = pastError;
  pastError = error;
  return command+BIAS;
}
```

B.4 kevFilter.h (Pressure Trigger)

```
//Written by Kevin Chu for the VAS Project
//#define KMASK 0x01FF
//#define KMEM_SIZE 512
#define KMASK 0x01FF
#define KMEM_SIZE 512
#define KAVG_LEN 16



class kevFilter
{
  public:
    // constructor
    kevFilter();
    // methods
    void initialize(int newValue);
    int getAverage();
    boolean insertAndTrig(int newValue);
    int getDiff();
    boolean getTrig();
    void setTrigVal(int newValue);

//    // running methods
//    void updateEnc(char quadBPinVal);
//    char runLoop(char innerVal, char outerVal);
//
```

```
    // data
    int kevMem[KMEM_SIZE];
    int runningAvg;
    int runningDiff;
    int triggerVal;
    int kevIndex;
//    long reference;
//    long curPosition;
//    long velocity;
//    long pastError;
//    long pastError2;
//    long intError;
//    long ki;
//    long kp;
//    long kd;
//    char innerCnt;
//    char outerCnt;
//    long command;
};

kevFilter::kevFilter()
{
}

void kevFilter::initialize(int newValue)
{
  for (int ii = 0; ii<KMEM_SIZE; ii++)
  {
    kevMem[ii]=newValue;
```

```
  }
  runningAvg = newValue*8;
//  runningAvg = newValue*KAVG_LEN;
  runningDiff = 0;
  triggerVal = 150;  //100;//45 //Change this value to change Trig
  kevIndex=0;
}

int kevFilter::getAverage()
{
  return runningAvg/KAVG_LEN;
}

int kevFilter::getDiff()
{
  return runningDiff/KAVG_LEN;
}

boolean kevFilter::getTrig()
{
  return runningDiff>triggerVal;
}

void kevFilter::setTrigVal(int newValue)
{
  triggerVal = newValue;
}
```

```
//#define KMASK 0x01FF
//#define KMEM_SIZE 512
//#define KAVG_LEN = 8;


boolean kevFilter::insertAndTrig(int newValue)
{
 runningDiff += (newValue + kevMem[kevIndex] -
  kevMem[KMASK&(kevIndex-KAVG_LEN)] - kevMem[KMASK&(kevIndex+KAVG_LEN)]);
  runningAvg += newValue - kevMem[KMASK&(kevIndex-KAVG_LEN)];
  kevMem[kevIndex] = newValue;
  kevIndex = (++kevIndex)&KMASK;
 // if (runningAvg>=3200)
  {  return (runningDiff>triggerVal);}
//  else{
//    return false;
//  }
}



//
//void kevMotor::setPID(int nKp,int nKd,int nKi)
//{
//  kp = nKp;
//  ki = nKi;
//  kd = nKd;
//}
//
//void kevMotor::setReference(long newReference)
```

```
//{
//  reference = newReference;
//  velocity = 0;
//}
//
//
//void kevMotor::setVelocity(long newVelocity)
//{
////  reference = curPosition;
//  velocity = newVelocity;
//}
//
//void kevMotor::updateEnc(char quadBPinVal)
//{
//  if (quadBPinVal) {
//    curPosition++;
//  } else {
//    curPosition--;
//  }
//}
//
//
//char kevMotor::runLoop(char innerVal, char outerVal)
//{
//  reference += velocity;
//  innerCnt += innerVal*2-1;
//  outerCnt += outerVal*2-1;
//  innerCnt = constrain(innerCnt,0,20);
//  outerCnt = constrain(outerCnt,0,20);
```

```
//  long error = (reference/16-curPosition);
//  intError += error;
//  intError = constrain(intError,-1000,1000);
//
//  command = (error*kp+(error-pastError2)*kd*128+intError*ki)/128;
//
//  command = constrain(command,-MAX_CMD,MAX_CMD);
//  if (outerCnt>=10) {
//    command = min(command,0);
//  }
//  if (innerCnt>=10) {
//    command = max(command, 0);
//  }
//
//  pastError2 = pastError;
//  pastError = error;
//  return char(command+BIAS);
//}
```

References

[1] http://iacuc.ucsf.edu/policies/awsptailveinjection.asp.

[2] https://www.aalaslearninglibrary.org.

[3] http://www.bu.edu/orccommittees/iacuc/policies-and-guidelines/administration-of-drugs-and-experimental-compounds-in-mice-and-rats/.

[4] R. R. Anderson. Polarized light examination and photography of the skin. *Arch Dermatol.*, 127(July):1000–1005, 1991.

[5] R. M. Bannerman. The mouse in biomedical research. *Hematology*, 3:294, 1983.

[6] Q. Bao, D. Newport, M. Chen, D. B. Stout, and A. F. Chatziioannou. Performance evaluation of the inveon dedicated pet preclinical tomograph based on the nema nu-4 standards. *Journal of Nuclear Medicine*, 50(3):401–408, 2009.

[7] T. Bartelt. Wheatstone bridge.

[8] J. Bazare, M. L. Leamons, and J. F. Young. Sampling methods for pharmacokinetic studies in the mouse. *Journal of Pharmacological Methods*, 5(2):99–120, 1981. doi: 10.1016/0160-5402(81)90002-4.

[9] G. v. Belle, L. D. Fisher, P. J. Heagerty, and T. Lumley. *Biostatistics A Methodology for the Health Sciences*. John Wiley and Sons, Hoboken, 2nd edition, 2004.

[10] B. L. Callewaert, B. L. Loeys, C. Casteleyn, A. Willaert, P. Dewint, J. De Backer, R. Sedlmeier, P. Simoens, A. M. De Paepe, and P. J. Coucke. Absence of arterial phenotype in mice with homozygous slc2a10 missense substitutions. *genesis*, 46(8):385–389, 2008.

[11] S. Cetinkunt. *Mechatronics*. John Wiley and Sons, Hoboken, 2007.

[12] S. R. Cherry and S. S. Gambhir. Use of positron emission tomography in animal research. *Institute for Laboratory Animal Research*, 42(3), 2001.

[13] S. R. Cherry, J. A. Sorenson, and M. E. Phelps. *Physics in Nuclear Medicine*. Elsevier, Philadelphia, 3rd edition, 2003.

[14] F. S. Collins. Opportunities for research and nih. *Science*, 327(5961):36–37, 2010.

[15] N. R. Council. National research council, 2010.

[16] S. G. Demos and R. R. Alfano. Optical polarization imaging. *Appl. Opt.*, 36(1):150–155, 1997.

[17] U. DLAM. Wet lab for mice and the laboratory ray ucla dlam pamphlet, 2008.

[18] A. Douraghy. *Design and Development of an Optical and Positron Emission Tomograph*. PhD thesis, 2009.

[19] M. A. Eddings, M. A. Johnson, and B. K. Gale. Determining the optimal pdmspdms bonding technique for microfluidic devices. *Journal of Micromechanics and Microengineering*, 18(6):067001, 2008.

[20] B. J. Fueger, J. Czernin, I. Hildebrandt, C. Tran, B. S. Halpern, D. Stout, M. E. Phelps, and W. A. Weber. Impact of animal handling on the results of 18f-fdg pet studies in mice. *Journal of Nuclear Medicine*, 47(6):999–1006, 2006.

[21] C. J. Gordon. Influence of heating rate on control of heat loss from the tail in mice. *American Journal of Physiology - Regulatory, Integrative and Comparative Physiology*, 244(6):R778–R784, 1983.

[22] E. V. Groman and C. P. Reinhardt. Method to quantify tail vein injection technique in small animals. *Journal of the American Association for Laboratory Animal Science*, 43(1):35–38, 2004.

[23] H. Hedrich. *The Laboratory Mouse*. Elsevier Limited, Amsterdam, 2004.

[24] A. Hem, A. J. Smith, and P. Solberg. Saphenous vein puncture for blood sampling of the mouse, rat, hamster, gerbil, guineapig, ferret and mink. *Laboratory Animals*, 32(4):364–368, 1998.

[25] H. R. Herschman. Molecular imaging: Looking at problems, seeing solutions. *Science*, 302(5645):605–608, 2003.

[26] J. Hoff. Methods of blood collection in the mouse. *Lab Animal*, 29(10):47–53, 2000.

[27] S.-C. Huang, H.-M. Wu, K. Shoghi-Jadid, D. B. Stout, A. Chatziioannou, H. R. Schelbert, and J. R. Barrio. Investigation of a new input function validation approach for dynamic mouse micropet studies. *Molecular Imaging and Biology*, 6(1):34–46, 2004. doi: 10.1016/j.mibio.2003.12.002.

[28] H. Izumi, T. Yajima, S. Aoyagi, N. Tagawa, Y. Arai, M. Hirata, and S. Yorifuji. Combined harpoonlike jagged microneedles imitating mosquito's proboscis and its insertion experiment with vibration. *IEEJ Transactions on Electrical and Electronic Engineering*, 3(4):425–431, 2008. 10.1002/tee.20295.

[29] S. L. Jacques, J. C. Ramella-Roman, and K. Lee. Imaging skin pathology with polarized light. *Journal of Biomedical Optics*, 7(3):329–340, 2002.

[30] K. Liu, Y.-C. Chen, H.-R. Tseng, C. Shen, and R. van Dam. Microfluidic device for robust generation of two-component liquid-in-air slugs with individually controlled composition. *Microfluidics and Nanofluidics*, 9(4):933–943, 2010.

[31] D. Malakoff. The rise of the mouse, biomedicine's model mammal. *Science*, 288(5464):248–253, 2000.

[32] R. Moore and C. Braselton. Injections of air and of carbon dioxide into a pulmonary vein. *Annals of Surgery*, 112(2), 1940.

[33] R. Myers. The biological application of small animal pet imaging. *Nuclear Medicine and Biology*, 28(5):585–593, 2001. doi: 10.1016/S0969-8051(01)00213-X.

[34] M. Nixon and A. Aguado. *Feature extraction and image processing*. Elsevier, Oxford, 2nd edition, 2008.

[35] P. M. Nolan and e. al. A systematic, genome-wide, phenotype-driven mutagenesis programme for gene function studies in the mouse. *Nature Genetics*, 25:440–443, 2000.

[36] A. Oki, H. Ogawa, M. Nagai, S. Shinbashi, M. Takai, A. Yokogawa, and Y. Horiike. Development of healthcare chips checking life-style-related diseases. *Materials Science and Engineering: C*, 24(6-8):837–843, 2004. doi: DOI: 10.1016/j.msec.2004.08.032.

[37] V. C. Paquit, T. L. Ferrell, F. Meriaudeau, J. R. Price, R. Seulin, K. W. Tobin Jr, and R. H. Farahi. Near-infrared imaging and structured light ranging for automatic catheter insertion". In K. R. Cleary and J. Robert L Galloway, editors, *Medical Imaging 2006: Visualization, Image-Guided Procedures, and Display*, volume 6141, San Diego, Ca, 2006. SPIE. [ORNL] [Le2i, University of Burgundy].

[38] J.-H. Park, M. G. Allen, and M. R. Prausnitz. Biodegradable polymer microneedles: Fabrication, mechanics and transdermal drug delivery. *Journal of Controlled Release*, 104(1):51–66, 2005. doi: DOI: 10.1016/j.jconrel.2005.02.002.

[39] M. E. Phelps. *PET Molecular imaging and its biological applications*. Springer, New York, 2004.

[40] R. P. Rand, A. C. Burton, and T. Ing. The tail of the rat, in temperature regulation and acclimatization. *Canadian Journal of Physiology and Pharmacology*, 43(2):257–267, 1965. doi: 10.1139/y65-025.

[41] H. Shima, K. Ohno, K.-i. Michi, K. Egawa, and R. Takiguchi. An anatomical study on the forearm vascular system. *Journal of Cranio-Maxillofacial Surgery*, 24(5):293–299, 1996. doi: 10.1016/S1010-5182(96)80062-X.

[42] D. Stout, A. Chatziioannou, T. Lawson, R. Silverman, S. Gambhir, and M. Phelps. Small animal imaging center design: The facility at the ucla crump institute for molecular imaging. *Molecular Imaging and Biology*, 7(6):393–402, 2005.

[43] C. Suckow, C. Kuntner, P. Chow, R. Silverman, A. Chatziioannou, and D. Stout. Multimodality rodent imaging chambers for use under barrier conditions with gas anesthesia. *Molecular Imaging and Biology*, 11(2):100–106, 2009.

[44] M. Talan. Body temperature of c57bl/6j mice with age. *Experimental Gerontology*, 19(1):25–29, 1984. doi: 10.1016/0531-5565(84)90028-7.

[45] M. L. Turgeon. *Clinical Hematology: Theory and Procedures*. Lippincott Williams and Wilkins, 2004.

[46] D. C. Vines, D. E. Green, G. Kudo, and H. Keller. Evaluation of mouse tail-vein injections both qualitatively and quantitatively on small-animal pet tail scans. *Journal of Nuclear Medicine Technology*, 2011.

[47] T. Vo-Dinh. *Biomedical Photonics Handbook*. CRC, Oak Ridge, 2003.

[48] M. S. Wang, D. R. Haynor, G. J. Wilson, T. Leiner, and J. H. Maki. Maximizing contrast-to-noise ratio in ultra-high resolution peripheral mr angiography using a blood pool agent and parallel imaging. *Journal of Magnetic Resonance Imaging*, 26(3):580–588, 2007.

[49] H. Zhang. Performance evaluation of petbox: A low cost bench top pet scanner dedicated to high throughput preclinical imaging, 2009.