ABSTRACT

LIVINGSTON, FREDERICK JERARD. Development of an Internet Addressable Pneumatically Controlled Instrument for Applying Strain to Cells In-Vitro. (Under the direction of Dr. Edward Grant.)

Mechanical stimulation of tissue cells is a popular technique used by tissue engineering researchers to stimulate cell growth. This research requires an instrument that applies in-vitro compression and tension to individual cells, through mechanical loading. The mechanical load in the research reported on here is generated using a vacuum system under computer control. The vacuum system consists of a pneumatic valve that is proportionally controlled from a single board computer, and a pressure transducer to monitor the waveform of the applied mechanical loading. Because the computer control is based on a single-board computer, the mechanical loading of cells can be carried remotely using a network environment and a dedicated IP address. The system is a good example of a smart mechatronic system. The research and development was carried out with support from Flexcell International, a North Carolina based biotechnology company.

Development of an Internet Addressable Pneumatically Controlled Instrument for Applying Strain to Cells In-Vitro

by

Frederick Jerard Livingston

A thesis submitted to the Graduate Faculty of

North Carolina State University

In partial fulfillment of the

Requirements for the degree of

Master of Science

Electrical Engineering

Raleigh, NC

2006

Approved by:

Edward Grant (Chair of Advisory Committee)

Ola Harrysson

John Muth

BIOGRAPHY

Frederick Livingston was born March 21, 1980 in Greensboro, North Carolina to Willie and Deborah Livingston. He received his Bachelor of Science in Computer Engineering from North Carolina State University, Raleigh, NC in 2003, where he graduated cum laude. He remained at North Carolina State University to study mechatronics, robotics, and control system with the Center of Robotics and Intelligent Machines under the direction of Dr. Edward Grant. As a graduate student, Frederick worked with Sandia National Laboratories in the advance manufacturing engineering science. His responsibilities included the development of meso-scale fixturing, assembly, and tooling. Frederick also worked closely with Flexcell International to develop a new biomedical instrument using feed-back pneumatic controls. Frederick plans to continue his educational experience and obtain a doctor of philosophy degree. Frederick would like to combine machine learning, machine vision, mechatronics, robotics, and control systems to develop cutting edge manufacturing processes.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

- A/D = Analog to Digital Converter
- CRIM = Center of Robotics and Intelligent Machines
- CSV = Comma Separated Value
- DHCP = Dynamic Host Configuration Protocol
- DPDT = Double-Pole, Double-Throw
- EEPROM = Electronically Erasable Programmable Read-Only Memory
- ES = Embedded Systems
- GUI = Graphical User Interface
- HTTP = Hyper Text Transfer Protocol
- IAPCI = Internet Addressable Pneumatically Controlled Instruments
- I/O = Input/Output
- IT = Internet Technology
- LCD = Liquid Crystal Display
- OS = Operating System
- PC = Personal Computer
- PDA = Personal Digital Assistant
- PLC = Programmable Logic Controller
- PnP = Plug-and-Play
- PSI = Pounds per square inches
- PWM = Pulse Width Modulation
- SBC = Single Board Computer
- USB = Universal Serial Bus
- WWW = World Wide Web

Chapter 1. Introduction

Tissue cells encounter different types of stress depending on their location within in the human body. Such stresses are important for developing and maintaining cell functions. Bone tissue changes its form, mass, and internal structure depending on the stress applied to it [3]. Although certain levels of stress are vital for bone tissue development, overstressing tissue is potentially harmful. Overstressing bone cartilage can lead to cell reduction and is the main cause for implants operations [5]. Tissue Engineers study the impact of different stresses on cell behavior. The majority of cell straining analysis is performed outside of the human body, and is based on mechanical stimulation. A variety of devices have been developed for mechanical cell stimulation [1, 2, 4, 6, 7, 10]. This project involves the development of an Internet Addressable Pneumatically Controlled Instrument (IAPCI) for cell stimulation. This new device functions similar to commercially available devices that apply stress to cell tissue, e.g., the Flexcell 4000. However, the technological features possessed by this new device make it better suited for remote and distributed operation, since this new system is based on embedded systems and state of the art information technology.

1.1 Current Methods for Cells Stimulation

In 2000, Thomas D. Brown published an article in the Journal of Biomechanics reviewing the current techniques for mechanical stimulation of cells. According to Brown the common methods for delivering stress includes compressive loading, longitudinal stretching, substrate bending, out-of-plane circular substrate distention, in-plane substrate distention, and fluid shear systems[2]. Electric motors, springs, pulley, pistons, and piezoelectric actuators are commonly found in cell culture systems. These devices control displacement that apply longitudinal cell loading. In Figure 1a mechanical components are attached to the ends of the substrate to elongate the tissue cells. In Figure 1b mechanical forces are applied at particular pressure points to flex the substrate. The flexing causes the attached tissue cells to elongate with curvature of the substrate.

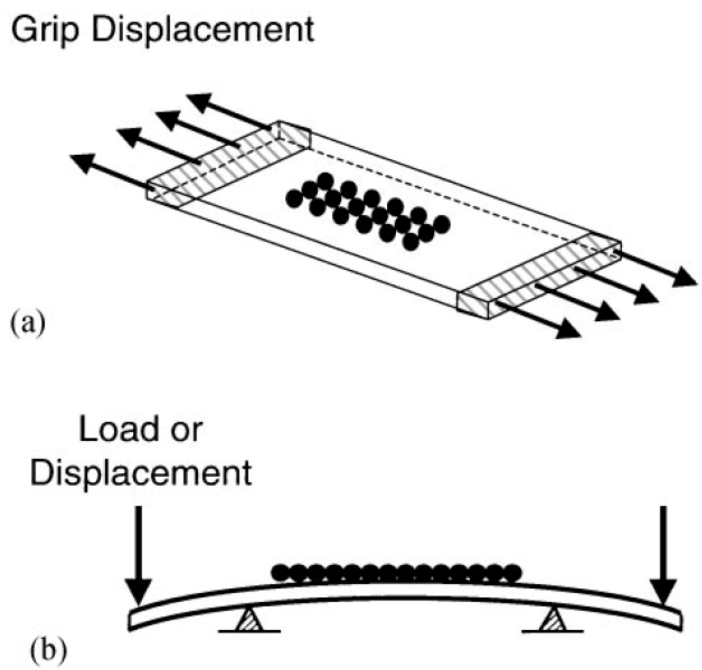


Figure 1. Longitudinal Stretch [2]

The amount and the resolution of the cell elongation depend on the material properties of the substrate and the loading mechanism. Devices such as piezoelectric actuators can deliver significant amounts of force with a high degree of resolution and accuracy. A disadvantage of using these devices for cell culturing systems is that many researchers prefer to perform multiple experiments simultaneously. With pneumatics, static or transient loads can be applied to multiple loading replicates simultaneously. A disadvantage of using pneumatic devices is that air is sensitive to temperature, sensitive to entrapped air in the system, and highly compressible which makes controlling pressure harder to achieve.

Commonly, pneumatics is used to apply compression cell loading. Compression loading involves stimulating cells by applying a force directly over the object. In Figure 2a compression loading was applied using hydrostatic pressure and a sealed chamber filled partially with a liquid medium and investigated cells. As the load is applied, the pressure in the system increases, stimulating the cells. In Figure 2b the cell experience compression loading directly from the load. In this configuration the cells are usually seeded in a matrix (scaffold) and a load is applied directly to the three-dimensional specimen.

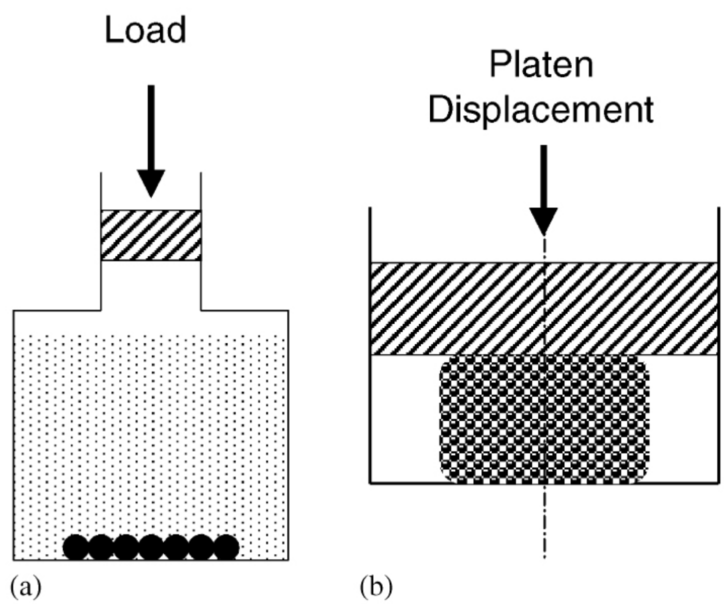


Figure 2. Compression Loading [2]

The Fx-4000 system developed by the Flexcell International Corporation in the USA applies strain to cells using pneumatics. Earlier Flexcell systems used vacuum pressure to apply out-of-plane cells distention. In these systems, tissue cells are attached to a flexible membrane. As a vacuum force is applied underneath the membrane, the cells are elongated. Using this particular technique it was discovered that the outer cells experience a larger percentage of elongation than the inner cells. Figure 3 display examples of out-of-plane cell loading.

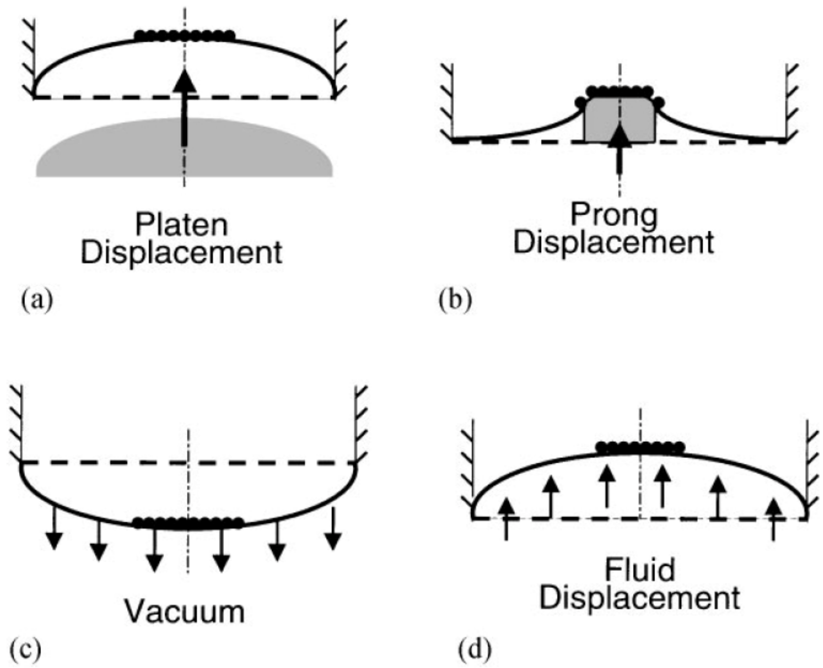


Figure 3. Out-of-Plane Loading [2]

A loading post, a low friction rigid structure, was added to more recent systems to apply a more uniform cell elongation. Figure 4 displays examples of uniform cell elongation configurations.

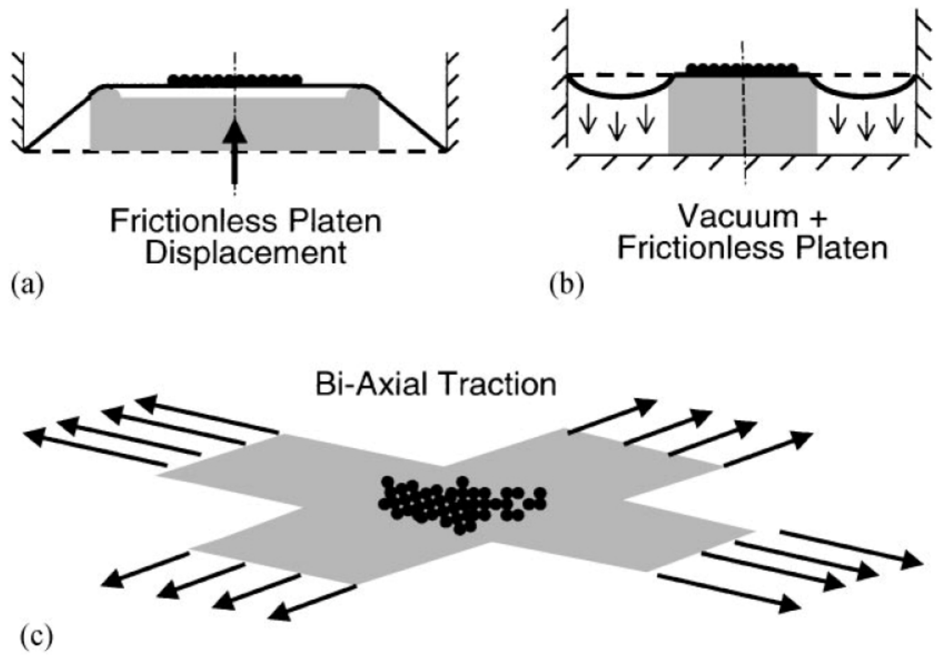


Figure 4. In-Plane Loading [2]

Today, the Flexcell Fx4000 system consists of the following four major components: an incubator, FlexCentral, FlexLinks, and Baseplates. The incubator is used to ensure that continuity of humidity and carbon dioxide levels, thereby creating the ideal environmental conditions for stressing the cells.

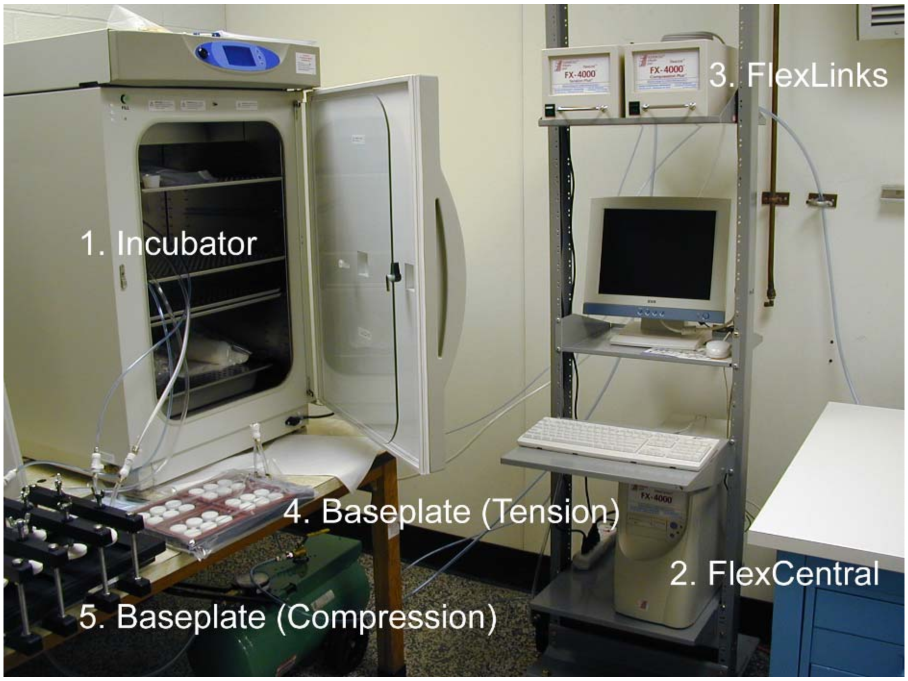


Figure 5. Flexcell Culture System

1.1.1 FlexCentral

The FlexCentral is a Microsoft Windows 98 based computer that controls and communicates between the components of the culture system (Fx-4000). The user interface for the entire system resides in the FlexCentral. The user interface has the following features:

- It produces multi-frequency waveforms and amplitudes for the test regimens.
- It is capable of producing the following waveforms: static, sinusoidal, heart (electronic and pressure), triangular, square wave, and custom.
- It is capable of varying the duty cycle of the triangular waveform and the square waveform.

- It is be capable of saving data, by capturing data from the screen every "x" minutes, where "x" is specified by the user.
- It converts all data curves from vacuum or pressure sensors into percent elongation or force. These are stored within a single text file so that they can be easily processed, or sent, by the customer and reviewed. This file also contains all of the customer's regimens and other configuration data necessary for the proper functioning of the unit.
- It allows any regimen created by a user to be simulated or run.
- It allows several FlexLinks to run at simultaneously. The user can display any graph by selecting a radio button on the main screen.
- The main screen displays the following information:
    - The number of FlexLinks connected at any given point in time, and which FlexLink is currently operating.
    - The waveform in terms of percent elongation, kPa, strain, millistrain, and microstrain.
    - A "smoothing" option that allows the user to apply a 3-point moving average to the data from the feedback sensors.
    - The X and Y coordinates of the displayed graph.
    - The current user, regimen, baseplate being used, and run-time.
    - The current regimen step, cycle, jump, minimum-level, maximum-level.
    - The plot of the current state of a simulated or actual regimen running.
    - Control buttons that allow the user to:

- Configure baseplates.
- Configure communications.
- Add users to the systems.
- Create and edit regimens.
- State an active regimen run.
- Pause a regimen run.
- Resume a paused regimen run.
- Stop a regimen run.
- Reset the controller.
- Collect and store data.
- Create waveform templates.
- Print the current waveform on the plot.

1.1.2 FlexLink

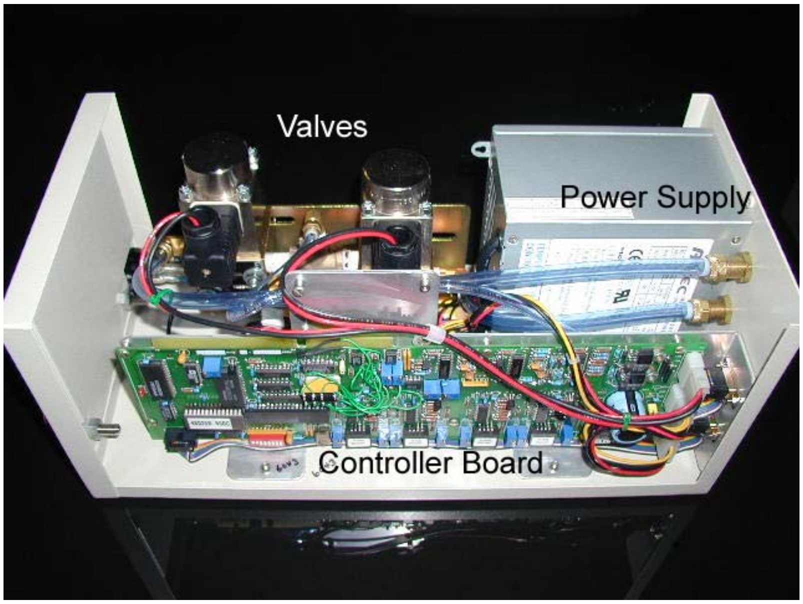


Figure 6. FlexLink

FlexLink's communicate with the FlexCentral via an RS-422/485 communication protocol. The main purpose of the FlexLink unit is to monitor and deliver vacuum, or pressure, to the baseplates. Theses units are linkable; they can be organized into multiple units that can supply pressure to multiple baseplates simultaneously. These devices consist of the following components: Controller board, electro-mechanical control valves, power supply, EEPROM, 8 bit A/D converter, 8 bit D/A converter, memory, 4 transducers, and signal conditioner circuits.

1.2 Research Goals

The research goal is to develop a new generation of instruments to stimulate cells in-vitro. This new instrument was specified to be compatible with pre-existing Flexcell International Corporation's compression and tension baseplates. The system was specified to perform the same task as the Fx4000, but with more enhanced, and user friendly features. The following is a primary list of features that were required in developing the new culture system: Information technology-based features, e.g., IP addressable units, EEPROM replaced with embedded microcontrollers/microprocessors, memory that is cheaper and faster, 8 bit A/D to 16 bit A/D, 8 bit D/A to 12 bit D/A, a proportional pneumatic valve that is computer controllable.

- In addition, the new cell straining instrument has external device controls; that enable the user to program control sequences for a vacuum pump or similar devices to turn on and off.
- The ability to control low vacuum and positive pressure levels to achieve low-levels and/or high-levels of strain and compression if desired.
- Auto calibration.
- Waveform frequency from 0.05 Hz upwards. The current limitation of the Fx4000 is 0.1 Hz.
- The ability for the controller to connect to any computer. The preferred communication is Ethernet.
- A single controller box that can control up to 4 individual valve boxes, either tension or compression.
- Multiple satellite or valve box systems. To have theses auto-numbered and named within the software and according to the order by which they are connected to the central processing box.
- An indicator light on each FlexLink to show whether or not it is the active FlexLink, this is displayed on the screen.
- The capability to use power of Internet Technology (IT0 to expand the functionality of the cell straining instrument to include auto-diagnose of the instrument over the web.

1.3 Outline of Thesis

The design and development of the IAPCI is described in this thesis. Chapter 2 presents an overview of the hardware and software components. This chapter provides an overview of modern day technical components use for controlling and monitoring pneumatic pressure. It also provides design guidelines and explanations for component selection.

Chapter 3 discusses the implementation of the user interface. It discusses methods for communicating data over a network. These methods consist of TCP sockets, html, php, and java applets and applications. The user interface uses theses communication methods to provide a touch screen interface and a web browser interface to the controller.

Chapter 4 involves the development of the control software and its interaction with the user interface. This chapter provides methods for creating real time control application using C++, multithreads, thread priorities, and task scheduling.

The control algorithms are discussed in Chapter 5. This chapter discusses four possible algorithms for controlling the IAPCI pressure according to information received from the feed-back sensors. This chapter compares the pros and cons of each algorithm and provided a detail explanation of the selected algorithm.

Chapter 6 provides experiments and results use to determine the characteristics of the new cell culture device. Lastly, Chapter 7 presents some ideas for further improvements of the pneumatic system using image processing and advance PID tuning algorithms.

Chapter 2. IAPCI Design

The design of the new system was driven by four primary goals. These goals were determined primarily by the need to develop a user friendly IAPCI system for a wide range of advanced cell straining experiments. This new device was designed to: (1) perform experiments with various input signals, (2) amplitudes, (3) frequencies, with a high-level of accuracy, and (4) to improve user-friendliness by adopting the power of the Internet.

In order to meet such a demanding set of requirements the system was required to possess the following specifications, including:

- Adaptable Controls.
- Robustness.
- Repeatable performance metrics.
- Provide a user-friendly control interface.

The IAPCI was designed to be a generic device for applying strain to cells. It was specified to be capable of performing experiments from low amplitudes and frequencies to high amplitudes and frequencies. The system was designed to control a variety of waveforms, from step inputs through sinusoidal and custom waveforms. Therefore, the systems required an on-board controller, a microcomputer that was capable of producing the complex waveforms required for the tests. The microcomputer must also be able to host the control algorithms that make up test regimes.

The system was designed to be as robust as possible; in that it is easily integrated with components of today's commercially available systems, e.g., Flexcercell's baseplates. It is designed to have sensor expandability if and when required. That is, a vision system can be added to monitor the deflection of the baseplate unit's if desired. Any new sensors to be added should be "Plug-and-Play", and it is for this reason that the USB architecture was adopted in the original design specification. Lastly, the system design specification allowed for easily upgradeable alterations to the control software.

The system should be capable of processing the large amounts of data generated during the cell strain tests. The processor must gather data from sensors that will allow for the making its display and control decisions that will ensure experimental consistency. The processor chosen, the pc-104 card, has the capability to process data in real time, in addition to performing any user interface functions that are required simultaneously. Plus, the system specification requires that this data must be stored for a period of time. Again, the PC 104 card provides this capability.

Finally, the IAPCI should be as user friendly as possible. The system must be Internet accessible, so that users have the capability of monitoring and controlling the system remotely. The system also has the capability of remote diagnostics if and when they are developed.

2.1 Single Board Computer

To meet the design specifications, the control devices market was explored. In today's market there are three main devices used for control. The main types of control devices are programmable logic controllers (PLC), embedded systems, and hybrid system.

PLC’s are highly reliable special-purpose computers used in industrial monitoring and control application. These devices are capable of producing the currents and voltages required for controlling industrial rated equipment. Such devices typically have proprietary programming, networking protocols, and special-purpose digital and analog input and output (I/O) ports. Commonly, these systems do not support a graphical user interface. Typically a second computer hosts the user interface and communicates with the PLC.

An Embedded System (ES) is similar to a personal computer (PC), but with the additional capability to perform digital and analog I/O operations directly. Unlike a general-purpose computer, an ES has specific requirements and performs pre-defined tasks. ES’s can exist solely as a microcontroller, or as an entire system; such as single board computer (SBC or PC-104 as an example). These devices are found today in many low power electronic devices such as cell phones, personal digital assistant (PDA), toys, camera, and etc. Although these devices are not capable of delivering high currents directly; in combination with other interface components, such as relays and power transistors, they are adaptable for use with industrial devices. Due to the pc-104 similarity to a PC’s, adding new interfacing devices can be as easy as plug and play.

There are many hybrid controllers, such as the National Instrument Field Point, which is technology that lies between a PLC and ES. The field point is design to operate in variety of environmental conditions, and is capable of performing control, digital, and analog I/O as well as acting as a host for the user interfaces. However, these devices typically have a limited amount of resources such as processing speed, and memory, and can are rather expensive to upgrade when this becomes necessary.

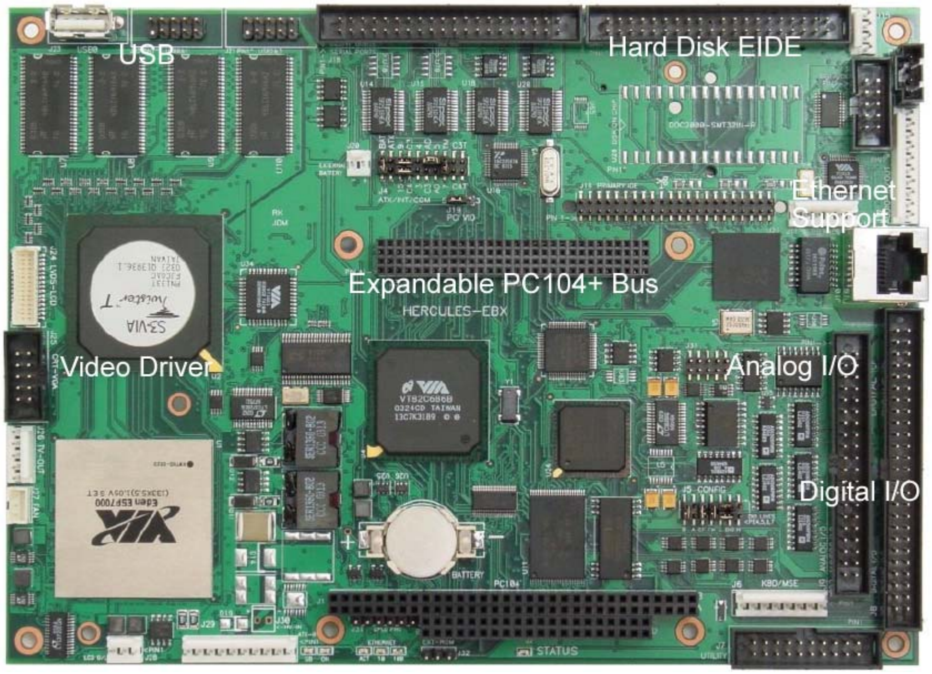


Figure 7. Hercules-EBX

For the IAPCI system the Diamond System's Hercules-EBX was chosen as the main controller. The Hercules-EBX is a single board computer (SBC) that supports standard features such as keyboard, mouse, hard disk, video, serial, and parallel interface, as well as advanced features, such as data acquisition (DAQ). To meet the performance needs of the pneumatic control system the SBC includes 750 MHz Pentium III class chipset and 256 MB of ram. The SBC is capable of 16-bit Analog to Digital (A/D) and 12-bit Digital to Analog (D/A) conversions. In additional the Hercules includes an expandable pc-104+ stack as well as USB for future add-on support. The SBC is also internet protocol (IP) addressable for remote access. Table 1 provides a list of the Hercules-EBX additional features provided by diamond system.

Table 1. Hercules-EBX Specs

Feature Details	Data Acquisition	Power Supply
VIA Eden Pentium III class 550-750 MHz processor	32 wide-rande analog inputs, +/- 10V down to 0-1.25V	5-28VDC input range standard - direct connect to vehicle power systems
128-256MB SDRAM soldered on board	16-bit A/D resolution	20-48VDC input range optional
VGA support up to 32 per pixel and max resolution of 1920x1440	Single-channel, multi-channel round robin, and multi-channel scan sampling	40 watts output power (12-16 watts used by CPU)
Integrated S3 ProSavage 4 AGP-4X graphics	250kHz max sample rate	Switched auxiliary power for IDE drives and accessories
Advance 3D rendering and 3D hardware acceleration	2048 A/D sample FIFO with programmable threshold	
LCD support any LVDS screen up to 1400x1050	Interrupt-based A/D data transfer with FIFO support	
AC97 audio with on-board codec and 2W per channel Hi-Fi stereo amplifier and DC volume control	Programmable input ranges	
10/100Mbps Ethernet with RJ-45 and pin header connections	4 wide-range analog outputs with simultaneous update capability	
TV out feature - composite and S-video	12-bit D/A resolution	
UDMA-100 IDE support	Multi-range autocalibraion for highest accuracy	
IDE flashdisk module socket	40 digital I/O lines with programmable direction	
Compact flash socket	16-bit and 24-bit counter/timers	
4 USB ports	4 programmable pulse-width modulation (PWM) outputs	
4 serial ports - 2 fixed RS-232 or RS-485, 2 selectable RS232/485	Field-upgradeable logic circuit for performance upgrades or custom designs	
PS/2 Keyboard and mouse		
Programmable watchdog timer, 0 to 2 seconds		
Low power fanless operation at 550 Mhz		
PC/104-plus ISA and PCI expansion		
-40 ro +85 Celsuis Operation		

2.2 Operating System

The IAPCI system's SBC is based on the PC architecture, and as such requires the use of an operating system (OS). The Hercules SBC supports many different operating system including Microsoft Windows, OS X, Linux, and other PC compatible OS's. Microsoft provides the operations system Windows CE .NET and Windows XP Embedded which are designed for the embedded system environments. Windows XP embedded allows for user configuration and selection of components that helps to develop the OS with only the required components. This is helpful in freeing up memory, processing time, and reducing overhead. Windows CE is similar to the XP embedded. The complete customization features of XP embedded are not available with the Windows CE operating system. Both of these operating systems are plug-and-play compatible with many electronic devices which make the system easily adaptable for future technology. Disadvantages with Microsoft products are the occasional system crashes occurrences and the expensive licenses. To avoid these issues the Linux operation system was investigated.

Linux is an open source, UNIX-like operation system that can be fully customized to meet specific needs. The core of the Linux operation system is the kernel. The kernel is loaded when the computer boots. The kernel provides policing and services to all of the other running programs. The kernel can be configured to meet real-time performance. The Linux operating system supports higher level languages and newer versions of the Linux kernel also interact with plug-and-play devices.

A customized version of Red Hat Fedora 3 is used as the operating system for the pneumatic control instrument. Libraries are included to support Java, C/C++, Perl, shell scripts, and data acquisition. Modifications to the startup process are implemented for faster startup and to lower system overhead. These modifications are found in the Appendix A.2.1 Modified Linux Startup Routine.

2.3 Pneumatic Valve

An important requirement of the IAPCI system is the manner in which air pressure or vacuum is supplied to cells in a controlled manner. Pressure is a function of resistance to flow in a fluid system, including wall resistance, bends, and the valves. A review of these instruments reveals that several devices have been used to control air flow. Earlier systems used on-off devices such as solenoid valves, and although these valves are usually fast acting, the air flow rate can not be controlled using them.

Pneumatic proportional valves have been developed in recent years to allow for variable flow rates according to the input signal. These valves usually contain a spool attach by some mean of actuation such as magnetic, piezoelectric actuators, or servo motors. These valves tend to have a slightly slower response time than solenoid valves. Proportional pneumatic valves can have hysteresis problems associated with them; caused by spool “stiction” and. high and low levels of air flow can effect the spool position. More advance proportional valves incorporate their own closed loop control system, to monitor and control the spool position.

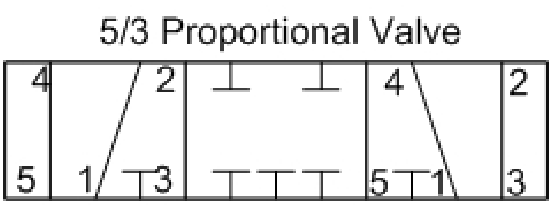


Figure 8. Proportional Valve

Table 2. Port Connections

Port	**Connection**
1	Out to Baseplates
2	Pressure Supply
3	No Connect (Plugged)
4	Free Exhaust
5	No Connect

The Norgren VP60 was selected to control the air flow in the new system. The VP60 is an electromagnetic proportional valve with a built in microprocessor to provide a linear response between input voltage and spool position. These valves contain 5 ports with 3 configurations. Port1 of the valve is connected to the baseplates chambers. Port2 of the valve is connected to the pressure source (vacuum pump or compressor). Port 4 is the system exhaust port. When the input voltage is 5 volts there is no flow in the system and the pressure is maintain at it current levels. When the input voltage is 0 volts; Port 4 is completely connected to Port 1 and pressure in the system is released. When the input voltage is 10 volts; Port 2 is completely connected to Port1 and system differential pressure increases.

Table 3. Port Configuration

Configuration	Input Voltage	Active Flow
1	0->5V	1-4 and 2->3
2	5V	No Flow
3	5->10V	1->2 and 4->5

Note: Using these valves Port2 and Port4 is not accessible to Port1 at same time, and Port 3 should be plugged to relieve pressure on the pressure source

2.4 Feedback Sensor

A transducer is needed to monitor the systems pressure. A pressure transducer is a device that converts pressure into an electrical signal. The most common pressure transducers are diaphragm gages.

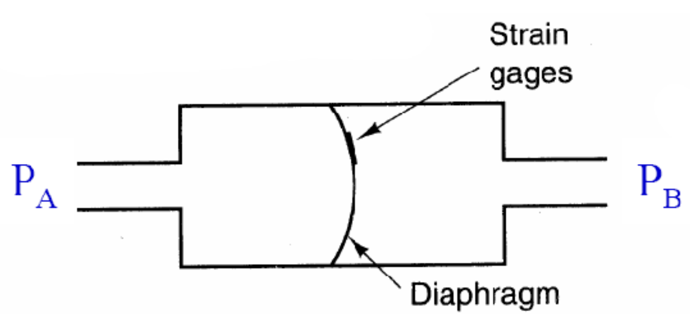


Figure 9. Pressure Transducer

These gages consist of thin diaphragms rigidly mounted around their perimeter that separate two chambers. A pressure difference between the two chambers causes the diaphragm to deflect. The degree of flexure, and hence strain, is sensed by strain gages and a Wheatstone Bridge. There is usually a linear relationship between pressure difference and the output signal. Gage pressure is sensed by opening Port B (see Figure 9) of the transducer to the atmosphere. Absolute pressure is sensed by vacuum sealing Port B. If both ports are available the transducer senses differential pressure.

Pressure transducers are available with several types of output, and several styles. Pressure transducer are generally available with three types of electrical outputs; milli-voltage, voltage, and current.

Cheaper transducer tends to use milli-volts for their output signal. This signal requires amplification in order to be processed by the microcontroller. These types of pressure transducers are highly sensitive to noise. A regulated power supply is needed to reduce noise. A voltage output transducer does not need amplification. Voltage transducers tend to output 0 to 5 or 0 to 10 volts. Voltage transducers are less influenced by noise but are more costly. Transducers that output current are least effected by noise. Current transducers are best used when the signal must be transmitted over long distances. Due to the wire resistance there is a voltage drop between the output of the transducer and the input into the microcontroller, this has to be compensated for. Just prior to the microcontroller a known resistance is attached to the circuit; this converts the current into a known voltage. Wire resistance has no effect on the current.

The Omega PX139-30A4V was chosen to monitor the system pressure. These transducers responded to absolute pressure, delivering an output signal from 0.25 to 4.25 volts. Differential transducers could have been used in place of an absolute pressure. The differential transducer signal senses the difference between atmospheric pressure and the baseplate vacuum pressure. In the future, if it is decided to use both compressed air and vacuum at the same time, these types of transducers would not have the capability of deciphering between too much positive pressure and too much negative pressure. Care must be taken here!

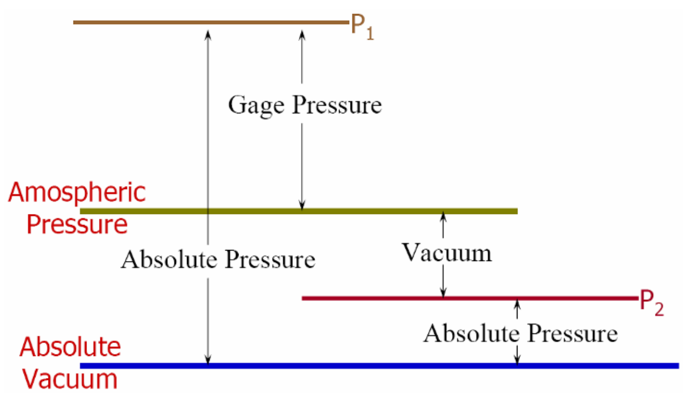


Figure 10. Types of Pressure Readings

In order to decipher the difference an absolute pressure is used and the differential pressure is computed. In order to compute differential pressure the system must take an atmospheric reading at the beginning of each experiment. The differential pressure is the absolute pressure minus the atmospheric pressure. This technique could cause potential problem for very long experiments where the atmospheric pressure varies.

2.5 Data Acquisition

Data Acquisition (DAQ) is responsible for collecting signals from the transducer and converting the data into a digital form. The Hercules SBC has a built in DAQ circuitry that will allow up to 32 single-ended, 16 bit analog to digital conversions. It is capable of performing 4 simultaneous, 12-bit resolution digital to analog conversion. The input range of the A/D is programmable for either a 10 or 5 volts range. The circuit has the ability to produce a stable output in 10 micro seconds.

The system developed by the Flexcell International Corporation (i.e., the Fx4000) used four pressure transducers for controlling the systems pressure. Two transducers are used to control a low range of pressures from 0 to 5 psi (34.474 kPa). The remaining transducers are used to control higher pressures up to 15 psi (103.421 kPa). The reason for four separate transducers was due to the hardware limitations of their interface board. The previous system used an 8 bit Analog to Digital converter only.

2.5.1 15psi Transducer with an 8 bit A/D

$$\mathrm{Re}\,solution = \frac{InputRange}{2^{\#ofbits}}$$

Equation 1. General Resolution

Using 15 pounds per square inch (psi) transducer with a 5 volt output range and an 8 bit analog converter the resolution of the transducer to in the system is calculated using Equation 1. Using this equation the system can only respond to changes in voltages greater than 0.02 volts.

$$\mathrm{Re}\,solution[Voltage] = \frac{InputRange}{2^{numberofbits}} = \frac{5}{2^8} = \frac{5}{256} \approx 0.01953125V$$

Equation 2. Resolution of 5 volt range using 8bit A/D

To calculate the transducer resolution in pressure the following assumption are made. There is a linear relationship between voltage and pressure. The transducer outputs 0 volts when the input pressure is 0 psi. The transducer outputs 5 volts when the input pressure is 15 psi. Using the proceeding assumptions the following relationships were derived to relate voltage and pressure.

$$Voltage[V] = \frac{1}{3}\left(\mathrm{Pr}\,essured_{gage}\left[psi\right]\right), \quad \mathrm{Pr}\,essure[psi] = 3\left[\left(Voltage[V]\right)\right]$$

Equation 3. 15psi Transducer Voltage vs. Pressure

Substituting the minimal detectable voltage change into Equation 2 gives the transducer resolution. The following calculations reveal that the system can detected a 0.404kPa change in pressure using the 15psi transducer in association with an 8 bit A/D converter.

Table 4. 15 psi Resolution Calculation

Step 1. Pressure at 0 volts	$\mathrm{Pr}\,essure(0v) = 3(0) = 0\,psi$
Step 2. Pressure at minimum voltage resolution	$\mathrm{Pr}\,essure(19.53125mv) = 3(19.53125m) = 0.05859375\,psi$
Step 3. Final Resolution Calculation	$\mathrm{Re}\,solution = \lvert \Delta P(1res) \rvert = \lvert \mathrm{Pr}\,ess(19.53125mV) - \mathrm{Pr}\,ess(0V) \rvert$ $= 0.05859375\,psi = 0.404kPa$

2.5.2 5 psi Transducer with an 8 bit A/D

The same procedures were used to calculate the system resolution of a 5 psi transducer in association with an 8 bit A/D. These calculations are shown below.

Table 5. 5 psi Transducer w/ 8 bit A/D Calculations

Step 1. Voltage resolution calculations	$\mathrm{Re}\,solution[Voltage] = \frac{InputRange}{2^{numberofbits}} = \frac{5}{2^8} = \frac{5}{256} \approx 0.01953125V$
Step 2. Voltage and Pressure relationship	$Voltage[V] = \left(\mathrm{Pr}\,essured_{gage}\left[psi\right]\right)$
Step 3. Pressure at 0 volts	$\mathrm{Pr}\,essure(0v) = 0\,psi$
Step 4. Pressure at minimum voltage resolution	$\mathrm{Pr}\,essure(19.53125mv) = 0.01953125\,psi$
Step 5. Final Resolution Calculation	$\mathrm{Re}\,solution = \lvert\Delta P(1res)\rvert = \lvert\mathrm{Pr}\,ess(19.53125mV) - \mathrm{Pr}\,ess(0V)\rvert$ $= 0.01953125\,psi = 0.135kPa$

Theses calculations reveal that the system can detected a 0.135 kPa change in pressure using a 5psi transducer along with the 8 bit A/D converter.

2.5.3 30psia Transducer with an 16 bit A/D

In today's market the 8 bit A/D have been replaced with 12 and 16 bit technology. The increase in number of bits allow for higher resolutions. Using Equation 1 for an 16 bits A/D and 5 volt range reveals that the systems is able to detect a change in voltage as little as 76.294 micro volts.

$$\mathrm{Re}\,solution[Voltage] = \frac{InputRange}{2^{numberofbits}} = \frac{5}{2^{16}} = \frac{5}{65536} \approx 76.294\mu V$$

Equation 4. Resolution of 5 volt range using 16bit A/D

The Omega px-139 transducer outputs 0.25 volts when the input pressure is at absolute vacuum (0 psia). The transducer outputs 4.25 volts when the system is at its maximum pressure of 30 psia. Figure 11 displays a graph of the relationship between absolute pressure and output voltage.

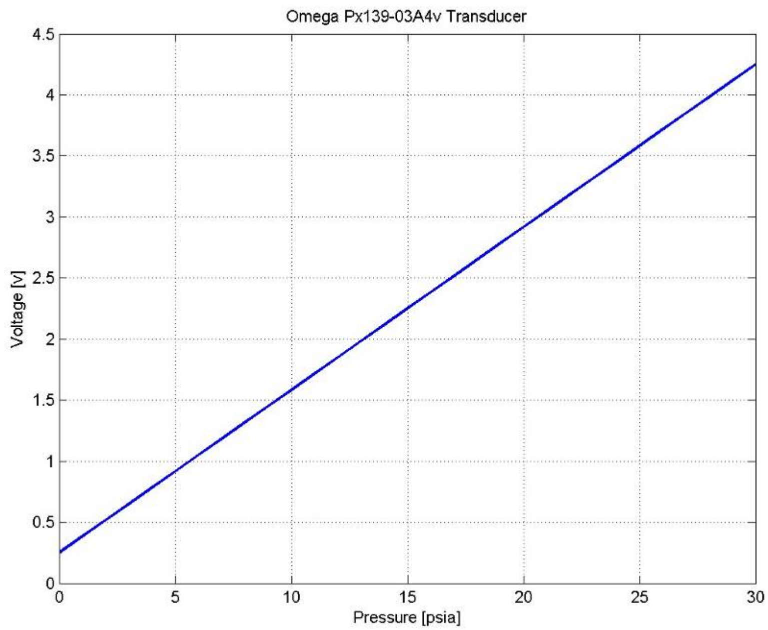


Figure 11. Omega Px-139 Pressure Transducer Function

Calculating the slope and y intercept the following equations where derive.

$Voltage[V] = \frac{2}{15}\left(\Pr essure_{Absolute}\left[psi_a\right]\right) + 0.25$	$\Pr essure_{Absolute}[psi_a] = \frac{15}{2}\left[\left(Voltage[V]\right) - 0.25\right]$
Equation 5. 30 psia Transducer Voltage vs. Pressure	

Repeating the resolution calculations reveals that the 30 psia transducer along with a 16 bit A/D provides a resolution of 0.0039452 kPa. These calculations are shown below.

Table 6. 30 psia Transducer w/ 16 bit A/D Calculations

Step 1.	$\Pr essure(0v) = \frac{15}{2}(-0.25) = -1.875kPa$
Step 2.	$\Pr essure(76.294\mu v) = \frac{15}{2}(76.294\mu - 0.25) = -1.87442779541 psi_a$
Step 3.	$\operatorname{Re} solution = \lvert \Delta P(1res) \rvert = \lvert \Pr ess(61.035\mu V) - \Pr ess(0V) \rvert = 0.00057220459\, psi_a$ $= 0.0039452kPa$

Table 7 is a summary of the results from the pressure transducer resolution calculations. These calculations reveal that a single 16 bit A/D will produce far more information than the Fx4000 systems sensors configuration.

Table 7. Transducer Resolution Summary

Resolution	5 psi Transducer with 8 bit A/D	15 psi Transducer with 8 bit A/D	30 psia Transducer with 16 bit A/D
Voltage	0.01953125 volts	0.01953125 volts	76.294 micro volts
Pressure	0.135 kPa	0.404 kPa	0.0039452 kPa

2.5.4 Digital to Analog Conversion for Valve Control

Digital to Analog (D/A) circuit is used to control the spool position of the valve. The Hercules SBC has 4, 12 bit D/A with programmable ranges. The D/A can output voltage from -10 volts to 10 volts. To control the valve the range is program to output voltages from 0 to 10 volts. Using Equation 1 the resolution of the D/A to the valve is 0.00244140655 volts.

$$\operatorname{Re} solution[Voltage] = \frac{InputRange}{2^{numberofbits}} = \frac{10}{2^{12}} = \frac{10}{4096} \approx 0.00244140625V$$

Equation 6. Resolution of 10 volt range using 16 bit A/D

The valves ports have a 5 volt range. Port 4 is completely open (0% closed) at 0 volts. Port 4 is completely closed (100% closed) at 5 volts (See Table 3 for more information). Using this information the following equation was derived.

$Voltage[V] = \frac{1}{25}\left(SpoolPosition[\%closed]\right)$	$SpoolPosition[\%closed] = 25(Voltage[V])$
Equation 7. Valve Voltage vs. Spool Position	

Using the calculations below, the DAQ can control the valve with a spool resolution of 0.061035 percent.

Table 8. Valve Spool Position Resolution Calculations using 16 bit A/D

Step 1.	$SpoolPosition(0v) = 25(0) = 0\%$
Step 2.	$SpoolPosition(0.00244140625v) = 25(0.00244140625) = 0.061035\%$
Step 3.	$SpoolPosition = \left\|\Delta S(1res)\right\| = \left\|S(0.0024V) - \Pr ess(0V)\right\| = 0.061035\%$

2.5.5 Software Interaction with DAQ

The Table 9 provides a list of all the devices that are controlled by the onboard DAQ. The Board class (board.cpp) was implemented to provide an interface with theses devices. The *initDAC* and *sample* methods are used to read signals from the A/D. *initDIO*, *setDIValue*, *getDIValue* is used to for digital I/O.

Table 9. DAQ Control Devices

Type	Port	Device/Pin#	Note
Digital Out	DIO A0	J8/1	External Power
Digital Out	DIO B0	J8/9	LCD Screen
Analog IN	Vin 0	J9/7	Transducer
Analog Out	Vout 3	J9/4	Valve

The *initDAC* method is used to set the system registers to allow unipolar output from 0 to 10 volts. The *sample* returns the average of 75 A/D samples from the port configure in *initDAC*. *InitDIO* method used to set the DAQ port for either input or output. The *setDIValue* and *getDIValue* methods perform digital read or write operations.

2.6 Power Supply

All of the system hardware requires the use of DC power. The pneumatic control system uses a 200 watt power supply to convert the AC voltage into DC. There is a need of variety of DC power levels. The proportional valves are powered by 24 volts. The feedback transducer uses 5 volts. The touch screen, PC-104, and storage device requires 12 volts. The Sola Hevi-Duty GLQ-05-200 well surpasses the system power needs. This power supply provides the following voltages: 5v (30A), 12V (8A), -12V (4A), 24V (4A). This power supply offers vacant wattage and easy access screw terminals for future additional add-ons.

One advantage of this particular power supply is that the input voltage can vary from 85 to 264 VAC at frequencies 47 to 63 Hz. The robustness of the power supply allows the system to operate in both the American and European standards. One disadvantage of the power supply that it requires a 10% minimal load on its 5 volts source. A power resistor is used in order to meet the minimal load requirement.

$V = I(R)$	$P = I(V) = \frac{V^2}{R}$

Equation 8. Ohm's Law and Power

Using Equation 8 the system would require a resistor less than 1.67 Ohms to meet the minimal loading of 15 watts. These calculations are shown below.

Step 1. Total Power Calculation of primary source	$P = I(V) = 5v(30A) = 150 watts$
Step 2. Minimum Load Calculation	10%(150 watts) = 15 watts
Step 3. Ideal Resistor Value	$R = \frac{V^2}{P} = \frac{5^2}{15} \approx 1.67\Omega$
Step 4. Power consumption using a 1.5 Ohms resistor	$P = \frac{V^2}{R} = \frac{5^2}{1.5} = 16.67 watts$

A 35 watts 1.5 Ohms resistor (DigiKey TCH35P1R50J-ND) consumes 16.67 watts which exceeds the minimal load requirements and stabilizes the power supply. This resistor requires the use of a heat sink (DigiKey HS110-220-ND) to keep the resistor's temperature levels within its operating range.

2.7 External Device Control

The pneumatic control unit provides two AC outlets for external device control. These outlets are design to control the vacuum pump or compressor. The outlet provides an AC voltage at the same amplitude and frequencies as the input supply voltage. The circuitry for these features consists of fuses, relays, and voltage regulators to supply an output signal up to 15 amps.

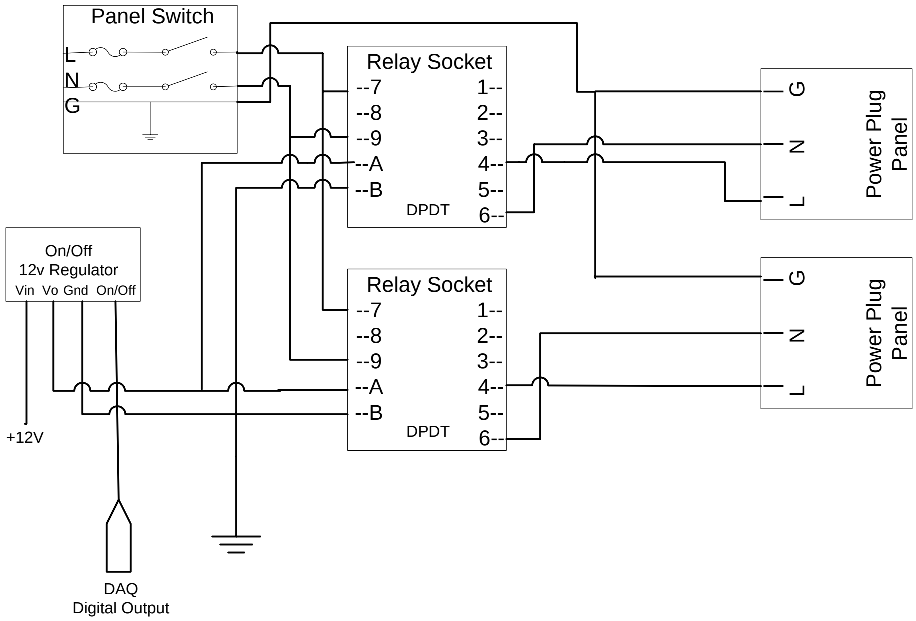


Figure 12. External Event Circuitry

To protect the system from power disturbances two 15 amps fast blow fuses (Mouser 504-GMA-15) are located within the panel switch. The controllable voltage regulator (Mouser 852-PQ12RD21JOOH) is enabled by the system DAQ. Once enabled the voltage regulator supplies 12 volts to activate the DPDT relay (Mouser 528-38815-6).

An external event controller was developed to allow for the following external events settings: Always On, Always Off, Auto On, and Timer. The executable *setPower* (compiled from setPower.c) interacts with the circuitry to activate or deactivate the external power. If the system is selected for “Auto On”, external power is provided when the experiments are started and turned off when the experiment is completed. The Timer mode uses the built-in Linux crond feature for external power control. The methods in scheduler.c interact with crond to allow users to schedule event at specific times.

2.8 Web Services

The popularity of the World Wide Web (WWW) in providing a vast array of information has drawn a large number of users in the past few years. The dramatic increase in of Internet users has led to the development of Internet assessable equipment such as network printers and other devices. Users are now able to monitor and control these devices from anywhere in the world when they have access to properly networked equipment. The IAPCI uses network technology to allow user to setup, monitor, and run experiments from a web browser such as Internet Explorer.

In order for the system to be accessed by a web browser a local web server is required on each instrument. There are several available web servers currently available on the market. The Apache HTTP server is one of the most powerful, stable, open source, web server available that is compatible with Linux OS. The Apache HTTP server can be freely downloaded at www.apache.com. The Apache server lacks a user interface, and one may find it difficult to configure. There are various more user friendly web servers available that are based on the Apache server. The Apache HTTP server is installed in to allow network execution of HTML files, Java Applets, and Java Scripts. For additional dynamic web features, open source PHP (www.php.net) plug-in for Apache is installed on the system to allow for general purpose HTML scripting.

The Apache server communicated with an onboard SQL server for data storage. SQL serves allow for fast convenient access to shared data over the network. The IAPCI uses MySQL (www.mysql.com) for data transactions. MySQL is reliable open source database software that is compatible with the systems software environment. MySQL offers free tools such as “MySQL Query Browser”, and “MySQL Administrator” that are useful for real time debugging. Figure 13 displays important global variables that are accesses by the SQL server.

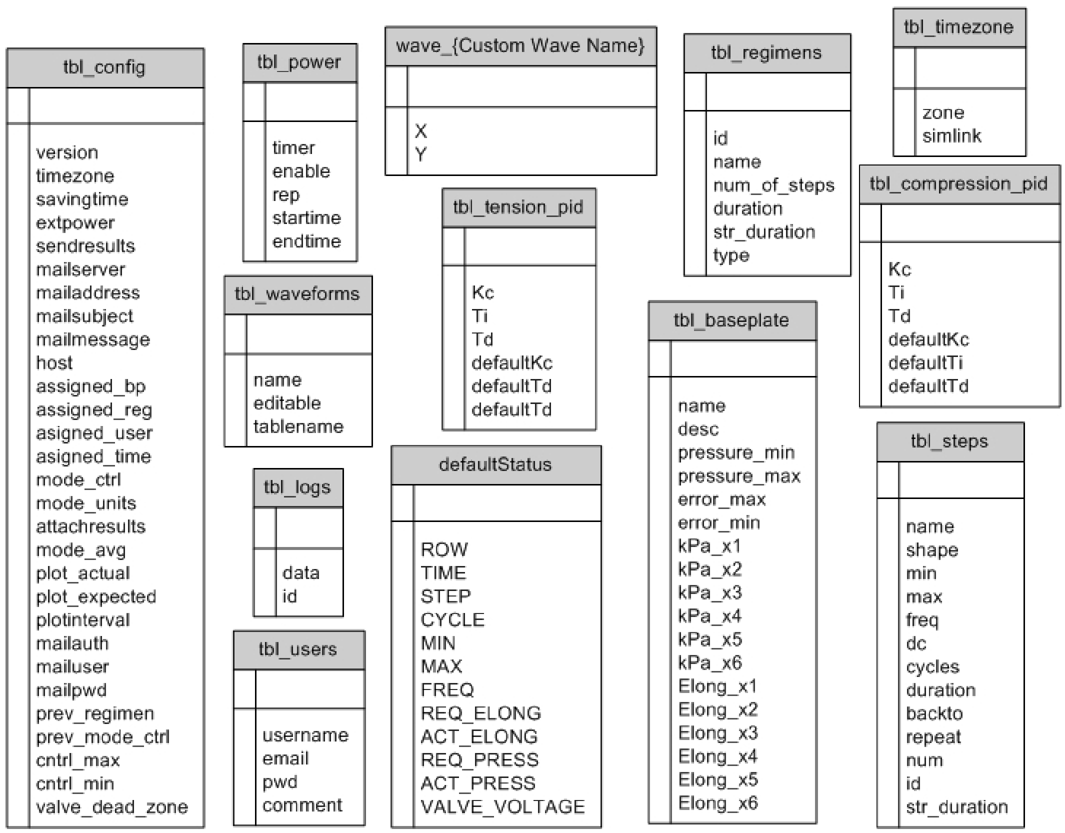


Figure 13. SQL Variables

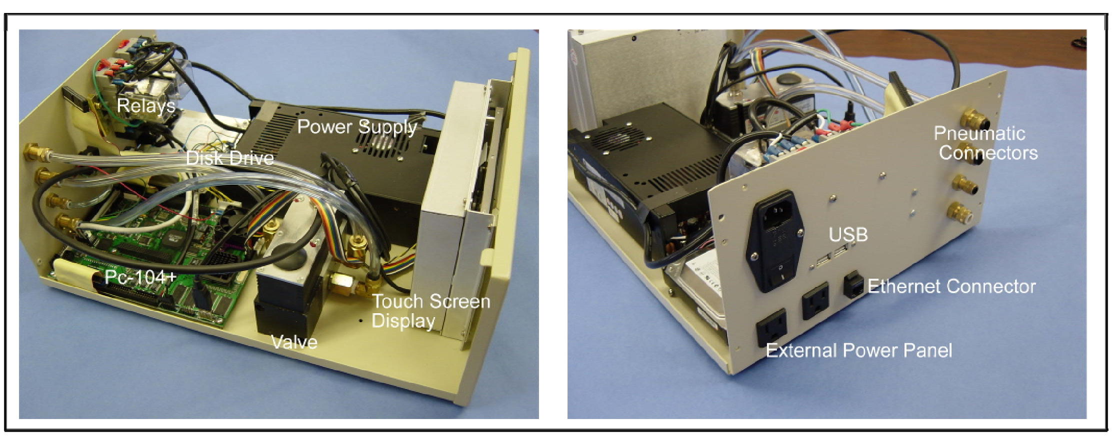


Figure 14. System Hardware

Chapter 3. User Interface

User interfaces were developed to interact with the system hardware components. They provide user with the capability to create, start, and monitor a simulation or a real experiment with the appropriate feedback, and of viewing the results as they become available. There are two type of user interface for the pneumatic control system. The first type of interface is the Control Client. This interface runs locally on each machine and is accessed using the touch screen display. The second type of interface allows remote operation of the system. This type of interface is discussed in Section 3.2 Remote Interface.

3.1 Control Client

The Control Client GUI is written in Java. This class is design to execute as an applet or an application. When Java applications are executed the main method is automatically invoked. Applets are started by the browser call to the init and start method. The appearance of the interface is affected depending upon whether the interface is executed as an application or applet. The Control Client application appears on the touch screen display requires larger components than an Applet. The enlarge components allows for easier selection using a touch screen. The IAPCI system lacks an external keyboard which is usefully for data entry. The application implements an on-screen keyboard which assist the user in data entry. Data entry for the application is kept to a minimal. The browser client allows for additional data entry and full control of the system. The following sections discuss individual aspects of the Control Client user interface.

3.1.1 Communication Utilities

Sun Java has developed a protocol, Java Database Connectivity (JDBC), to allow for interaction with various SQL database. The JDBC doesn't interact with the database directly, it communicates directly with drivers which then communicates with the database. There are different types of JDBC drivers. Many of these drivers require local installation on each client machine. This requirement reduces the robustness of operation the IAPCI system when run remotely from any web browser. To work around this issue the MySQL support is compiled directly into the code. The MySQL class was developed to interact with the IAPCI system MySQL database. This class contains methods such as getRegimen() and getLog() that is essential to the user interface.

3.1.2 On-Screen Keyboard

The system provides two on-screen display interfaces for data input. The AlphaNumInput class provides an interface to allow both alpha and numeric inputs. The Input class is used for only numerical inputs. The system GUI classes implements the java.awt.event.FocusListener interface. When an editable text field receives focus the focusGained method is called. This method saves the current text field information and either the AlphaNumInput or Input class. The AlphaNumInput GUI is shown in Figure 15.

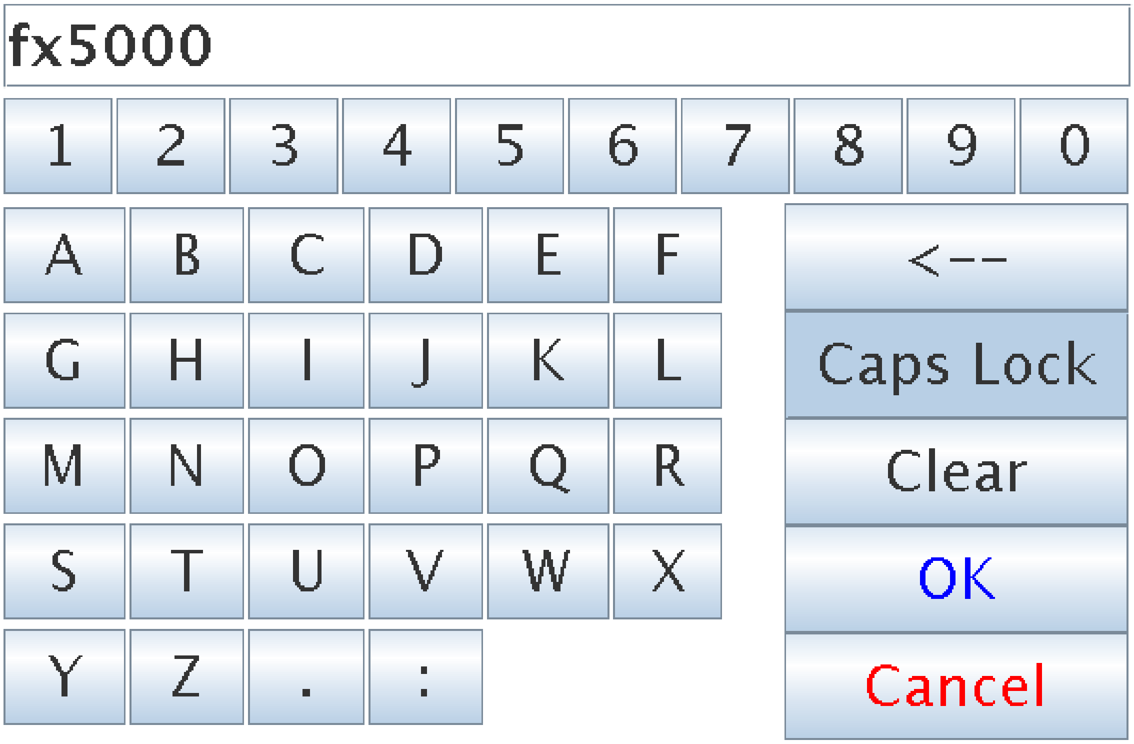


Figure 15. On-Screen Keyboard

3.1.3 Connection Class

To give a more "commercial" feel to the IAPCI system, and to provide a more realistic user interface, it was given the designation Fx5000, with the permission of the Flexcell International Corporation. This permission was sought because we were using the Fx4000 as the basis of our new design. The Fx5000 class was created to allow user friendly methods to login into the unit, remotely or locally. This class is invoke by the issuing following command; "java –jar Fx5000.jar fullscreen". The command line argument, "fullscreen" is an optional argument to resize the GUI to the screen maximum resolution.

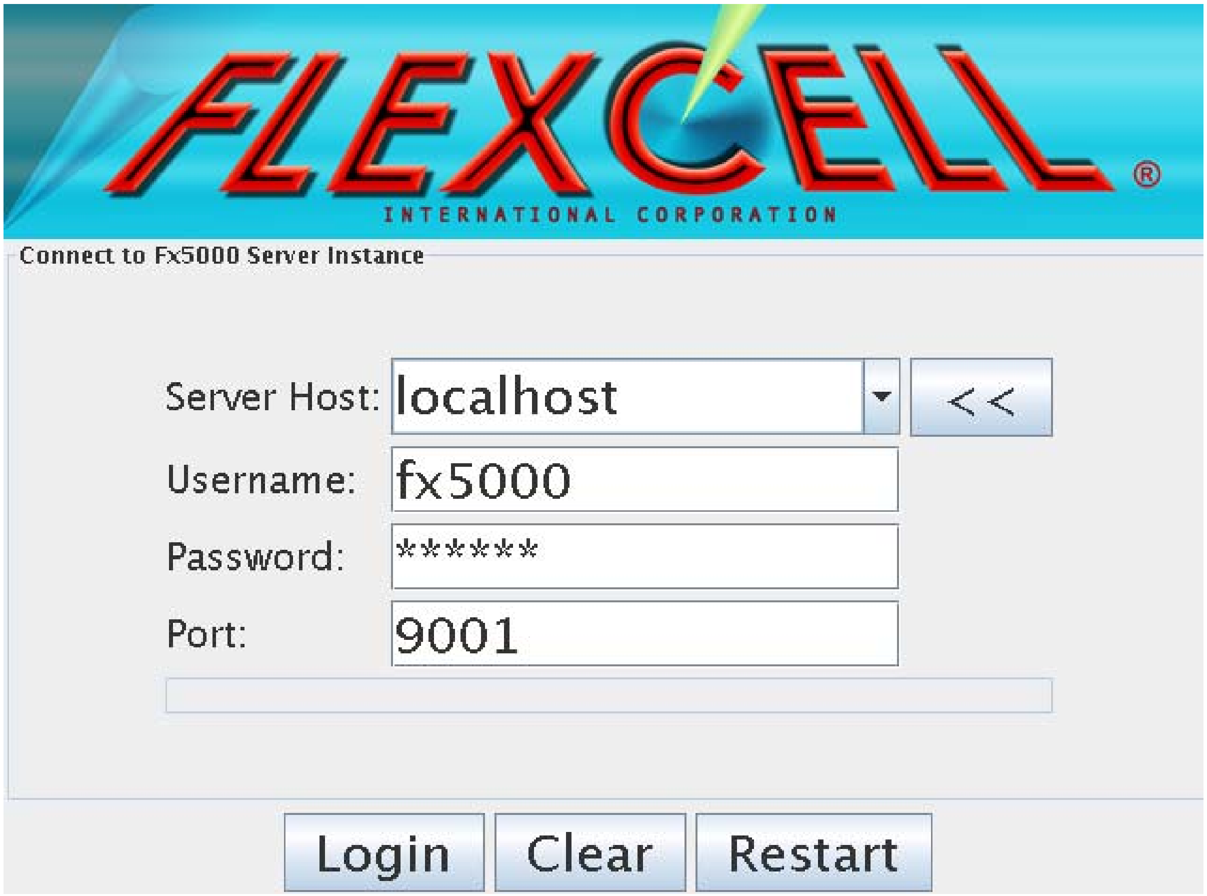


Figure 16. Connection Screen

This class creates a window frame (shown in Figure 16) that contains the Flexcell logo, a combo box with contains the host address, three labels with relating text fields for the username, password, and port, a text field that displays error messages, a login, clear, and cancel/restart buttons.

- **[Server Host]** This combo box, maintains a list of known host address. This list is imported from the ASCII text file *conn.dat*. The combo box is editable to allow for additional host names that are not listed in the file. The default host, 'localhost' or 127.0.0.1 allows users to login locally to the machine. The host address is editable to allow users to access secondary units, from the main unit. The secondary units may not contain touch screen displays.

- **[<<]** This button is used to remove hosts from the system. The hosts are removed from the combo box as well as from the *conn.dat* file.
- **[Username]** Case sensitive String containing a valid username.
- **[Password]** This text field allows the user to enter a case sensitive password. The password is not visible on the screen.
- **[Port]** This text field provides the port in which is used to connect to the control server. The control server default port is 9001.
- **[Login]** The login button is used to authenticate with the system. If the username and password are incorrect an error message will be display on the screen. If a successful authentication the host is added to the *conn.dat* file and a new instance of the *FxClient* class is invoked.
- **[Clear]** The clear button removes text from the username and password text fields.
- **[Cancel]** This button is used to exit the execution of the program. System.exit(1) is use to terminate the program.
- **[Restart]** If the application is started using the full screen argument the **[Cancel]** button is replaced with **[Restart].** This button is used to restart the system. The system is restart by execution the command "shutdown –r now".

3.1.4 Client Class

Once a user logs into in to the system, the window frame changes it current panel to the *FxClient* panel. The *FxClient* panel consists of the following tab panes: System Control, System Status, Plot Config, External Event Control, Network Config, and Log Off.

3.1.4.1 System Control Pane

The System Control Pane consists of two combo boxes and ten buttons to allow users to configure, start, and stop pneumatic pressures. Figure 17 contains a screen shot of the control pane.

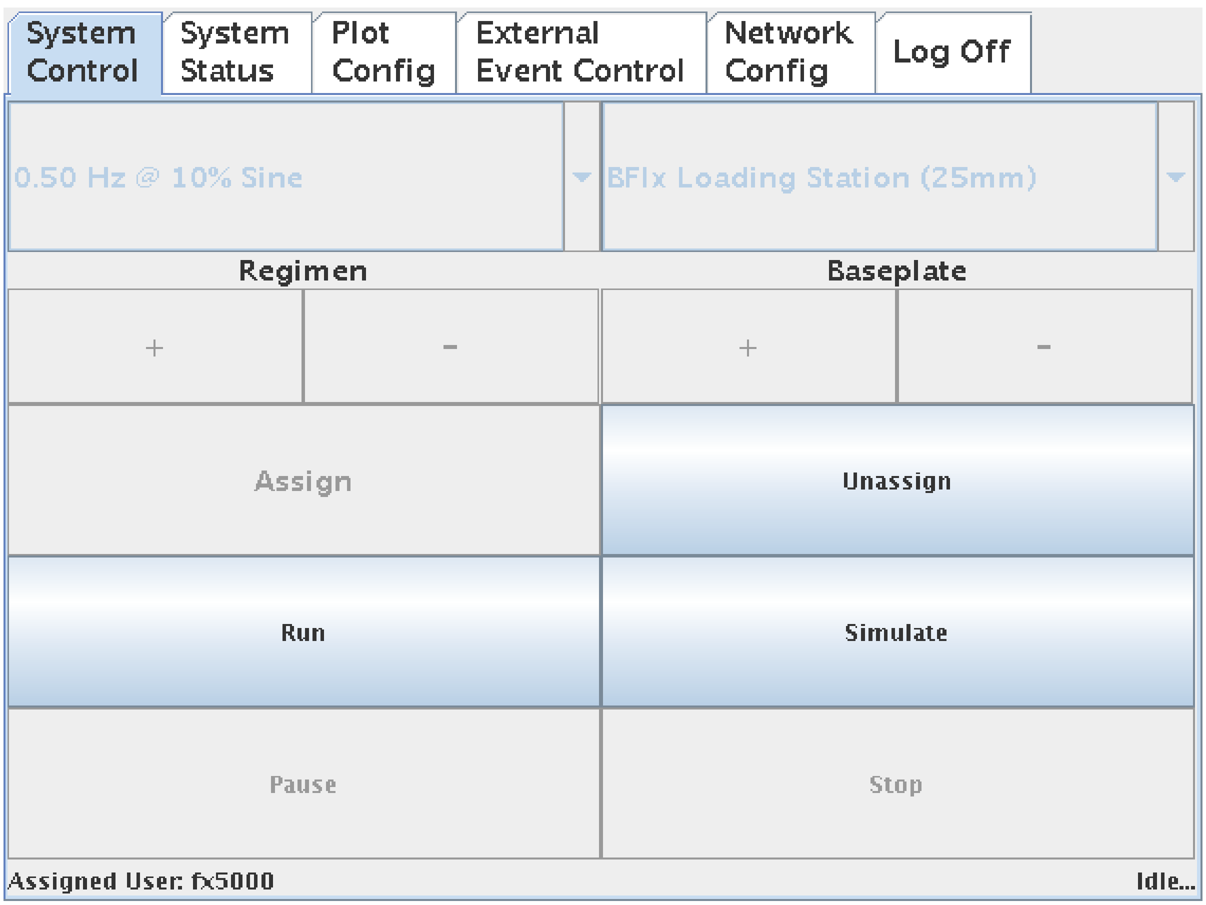


Figure 17. System Control Pane

- **[Combo box "Regimen Name"]** The first combo box contains a list of all the available regimens. The combo box contents are retrieved from the SQL database using the *MySQL.getRegimens()*. *MySQL.getRegimens()* returns a vector of Strings containing the regimen names.
- **[Combo box "Baseplate"]** This combo box contains a list of all the available baseplates. This list is created similar to the previous combo box using the function *MySQL.getBaseplates()*.
- **[Assign]** This button notifies the system that the user has chosen a regimen and a baseplate. These selected items are changed by using the corresponding **[+]** or **[–]** buttons or by using the combo box. The currently selected regimen and baseplate is used for the chosen assignment. Once the assignment is completed the user can no longer change the regimen and the baseplate until it is unassigned. The selected regimen, baseplate, user, and current time are updated to the SQL database and the current user name is displayed at the bottom left hand corner of the screen.
- **[Unassign]** The *Unassign* button stops the experiment, if running, and then reset the system to allow the user to select alternative regimen and/or baseplate. The selected regimen, baseplate, and user are cleared from the SQL database.
- **[Run]** This button is use to start the pneumatic control process. Once this button is pressed the graph in the System Status pane (See Figure 18) is cleared and the control server is notified to start execution.

- **[Simulate]** This button is use to simulate the pneumatic control process. This process is similar to the process started by run except for in simulation actual waveforms are not applied to the baseplates.
- **[Pause/Resume]** This button is used to pause and resume the experiment.
- **[Stop]** This button is used to stop running experiments.

3.1.4.2 System Status Pane

The System Status Pane as seen in Figure 18, it provides the user with information regarding the current status of a running experiment. The red curve is the graph of desired waveform. The blue curve displays the actual waveform obtained from by the system. These graphs are generated from querying the SQL database using the *fxclient.db.getStatus()* method. This method is called every second if the experiment is started and if *System Status Pane* is visible.

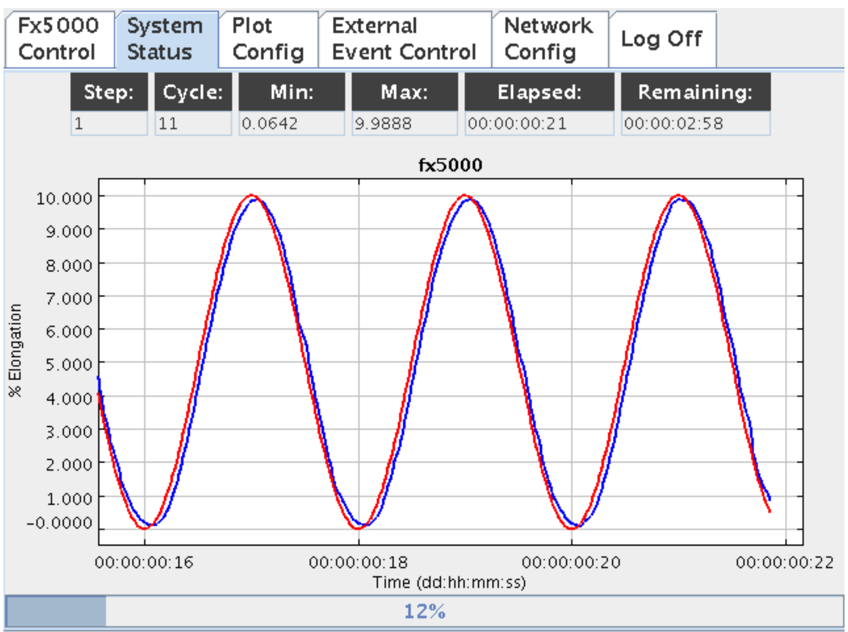


Figure 18. System Status Pane

The *fxclient.db.getStatus()* method provides the client with the data structure *Status*. The *Status* class contains data such as current step, cycle, min, max, elapsed time, and remaining time.

3.1.4.3 Plot Config Pane

The Plot Config Pane is design to control how the information is display on the *System Status Pane*. The Plot units are change by pressing the corresponding button on the Pane shown in Figure 19.

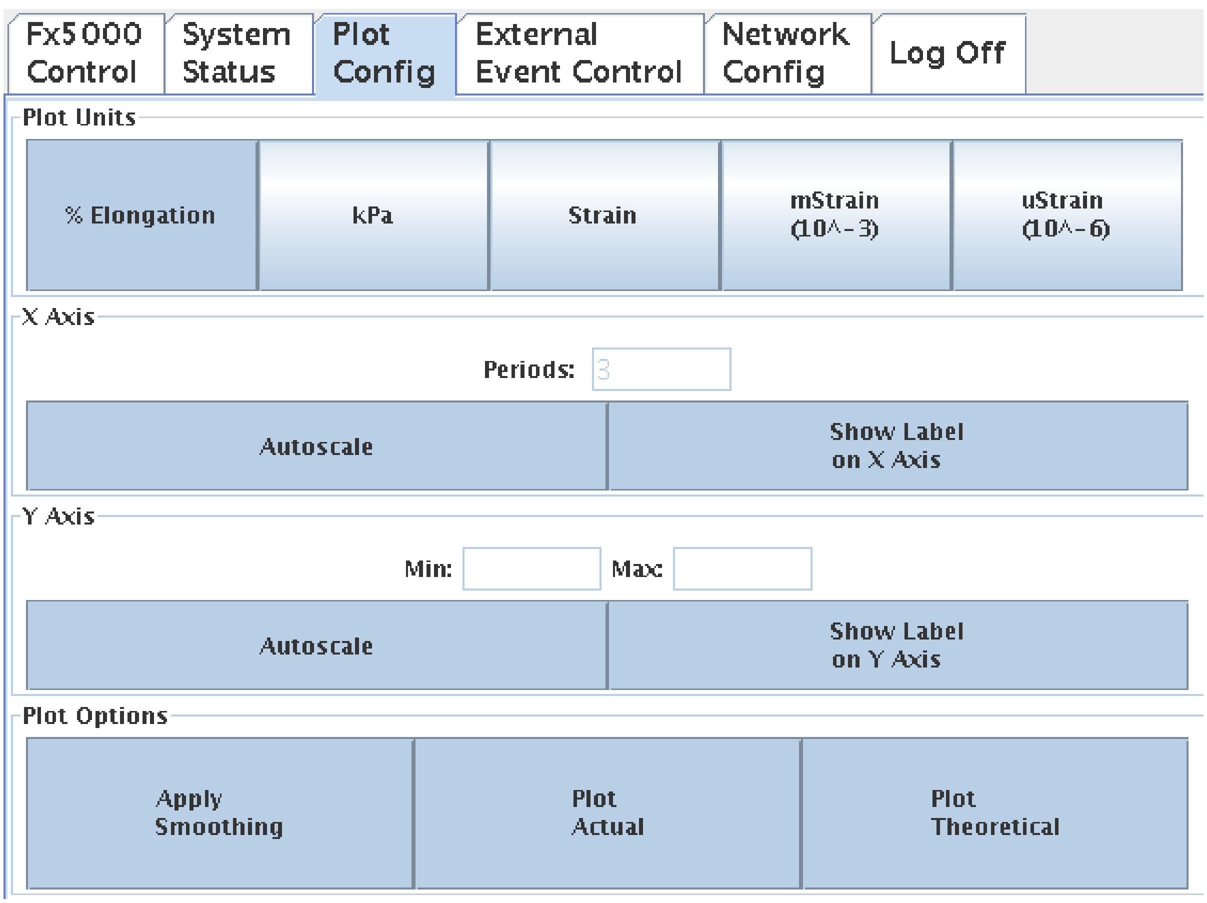


Figure 19. Plot Config Pane

Other plots setting are control using the buttons below.

- **[X Axis Autoscale]** This button allows the users to change the number of periods displayed on the graph. The default setting allows the users to view 3 periods of the waveform. Note: Changing the number of display periods only affects the current clients display.
- **[Show Label on X Axis]** This button is used to remove or add the time label on the graph.
- **[Y Axis Autoscale]** This button allows the users to change the y axis range of the waveform. If *Autoscale* is enabled and an active experiment of running the y axis is set to show the full range of the actual results. If simulating the y axis is set to show the full range of the desired waveform.
- **[Show Label on Y Axis]** This button is used to remove or add the unit label on the graph.
- **[Apply Smoothing]** This button is used to apply a 5 point moving average on the results.
- **[Plot Actual]** This button is used to enable or disable the plot of the actual waveform.
- **[Plot Theoretical]** This button is used to enable or disable the plot of the expected waveform.

3.1.4.4 External Event Control Pane

The External Event Control Pane, as seen in Figure 20, provides a user interface for control external power events. This interface contains a combo box where the user can select the power modes as list in Section 2.7 External Device Control. The selected power mode is stored in the SQL database using the *MySQL.setPowerMode()* method.

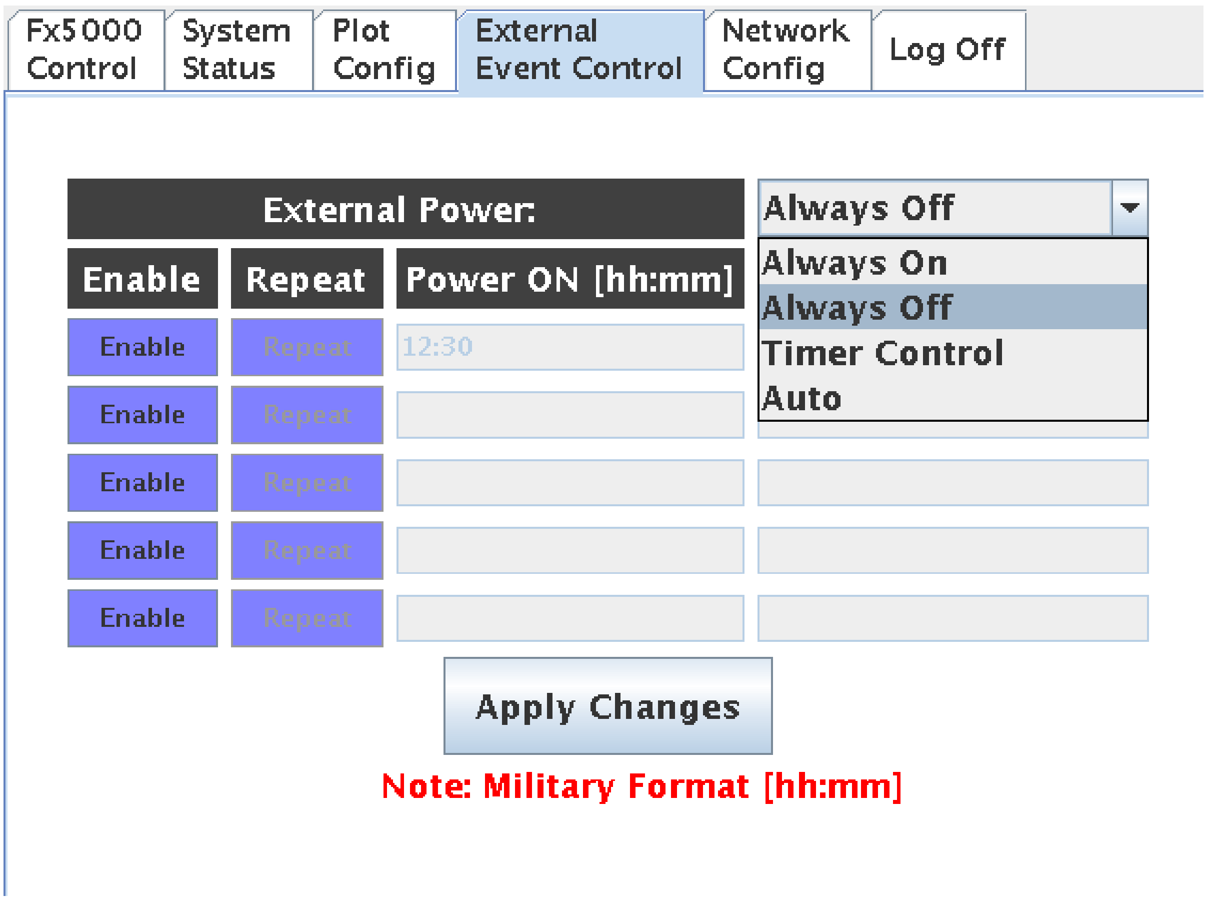


Figure 20. External Event Control Pane

The user interface provides buttons and text fields to allow up to five timed programming external events. Once enabled using the **[Enable]** button, the **[Power ON]** and

[Power OFF] text field are enabled and editable. The **[Repeat]** button allows the event to occur for multiple days until it is disabled. The Timer events are stored in the SQL database using the *MySQL. setPowerStatus()*.

3.1.4.5 Network Config Pane

The *Network Config Pane* is essential for the proper setup of the device. This Pane provides the user with the current IP address which is needed to fully control this device. The system default network status is displayed in the IP address, subnet mask, and default gateway text field.

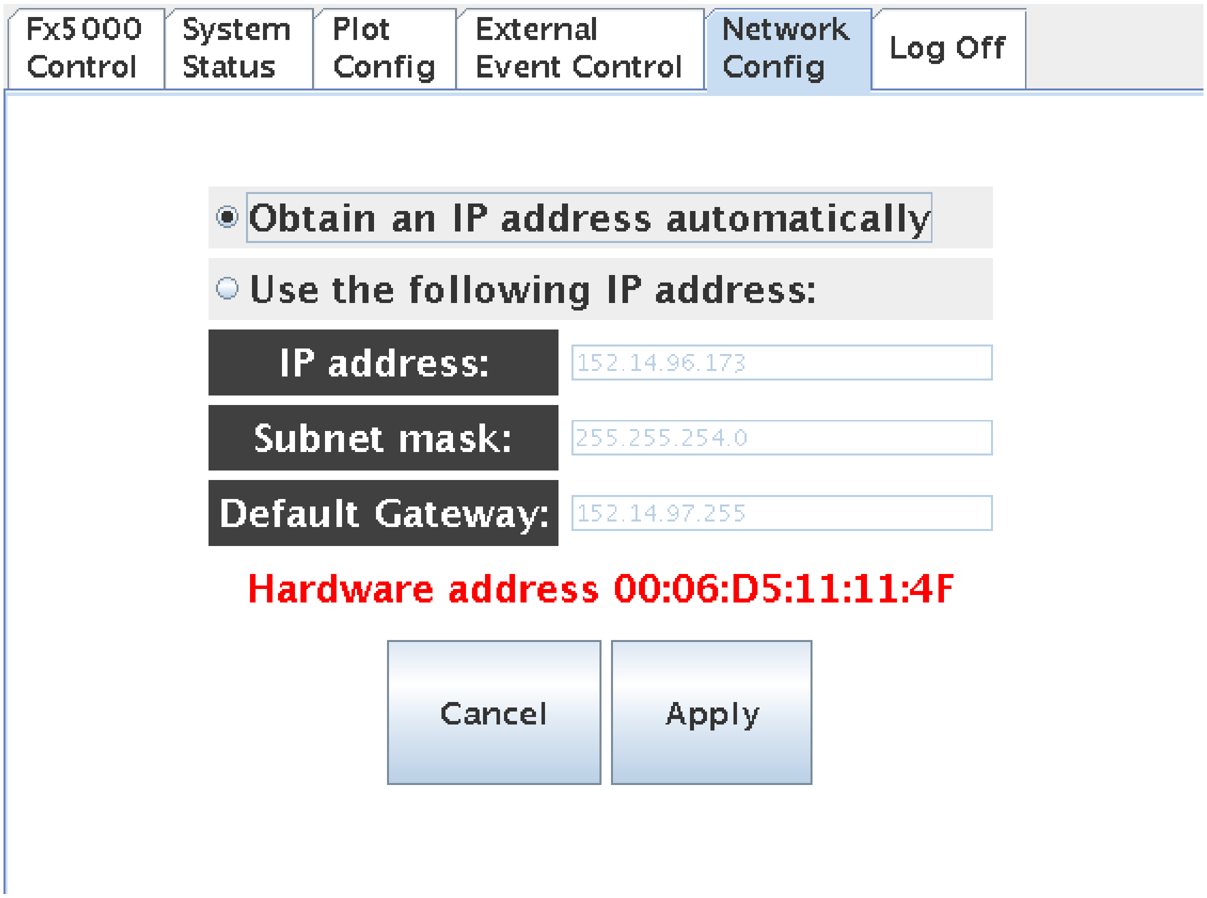


Figure 21. Network Config Pane

The default configuration allows the system to automatically connect to the network using Dynamic Host Configuration Protocol (DHCP). The user interface provides a radio button to allow for manual network configurations. If "Use the following IP address" radio button is selected the text fields are enabled and editable. Once enabled the Input class interface is displayed to allow user to input the network information. Network changes are not effective until the **[Apply]** button is pressed. Pressing the **[Apply]** button executes the command *setNetwork* (compiled from network.c) with the IP address, sub mask, and gateway as the input arguments. *setNetwork* is responsible changing the network configuration. These changes are effective at the end of execution and there is no need for a system restart.

3.1.4.6 Log Off Pane

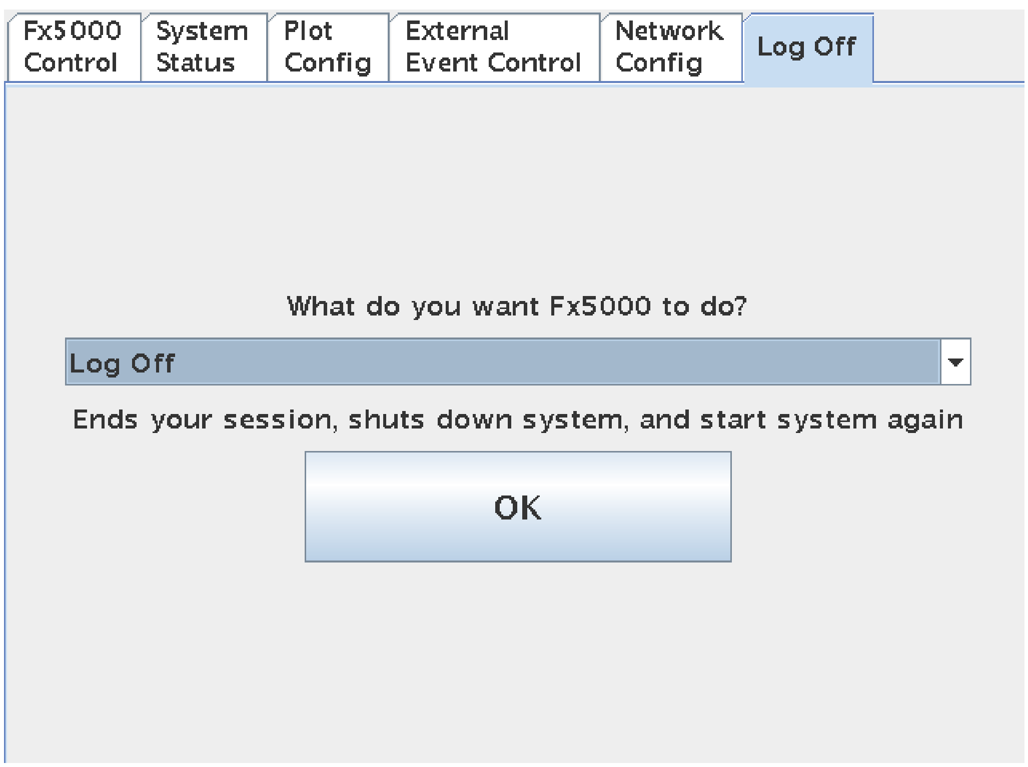


Figure 22. Log Off Pane

Figure 22, display a screen shot of the *Log Off* Pane. This Pane allows the user to either log-off or restart the system. If the user choose to the log-off the connection to the server are closed and the window pane is change to the Fx5000 Panel.

3.2 Remote Interface

The remote interface is designed to fully control the pneumatic culture system. The touch screen interface was primary designed to monitoring the system, but it does include control features such as stopping and starting the device(s). There a several methods used for remote interfacing. One method is too developed and distribute network applications. These applications are usually are installed on each client machine. An example of this type is the once famous “Kazaa”. In the early 21st century, Kazaa was a popular application using the Peer-To-Peer (p2p) internet protocol for file sharing. To interact with the file server each user would have to download and install the application on their personal machine. This installation procedure reduces the availability of accessing the machine remotely. Another disadvantage of network applications is the capability of the software between different operating systems such a Palm OS, Pocket PC, Windows, and Linux. The Remote Interface makes use of the web browser to communicate with the pneumatic culture system. Web browsers such as Internet Explorer are becoming standard applications and are usually included with the operation system. Advantages of using a web browser are that the average users are already familiar with the basic operations of web browsers, and web pages are typically easier to update for future development. The following sections provide an overview of the web pages used to interact with the pneumatic culture systems.

3.2.1 Control.php

Once the user is properly authenticated the screen shown in Figure 23 is visible by the browser. The *FxClient* applet is embedded within control.php. This applet is identical to the *Client* class mentioned in Section 3.1.4 Client Class.

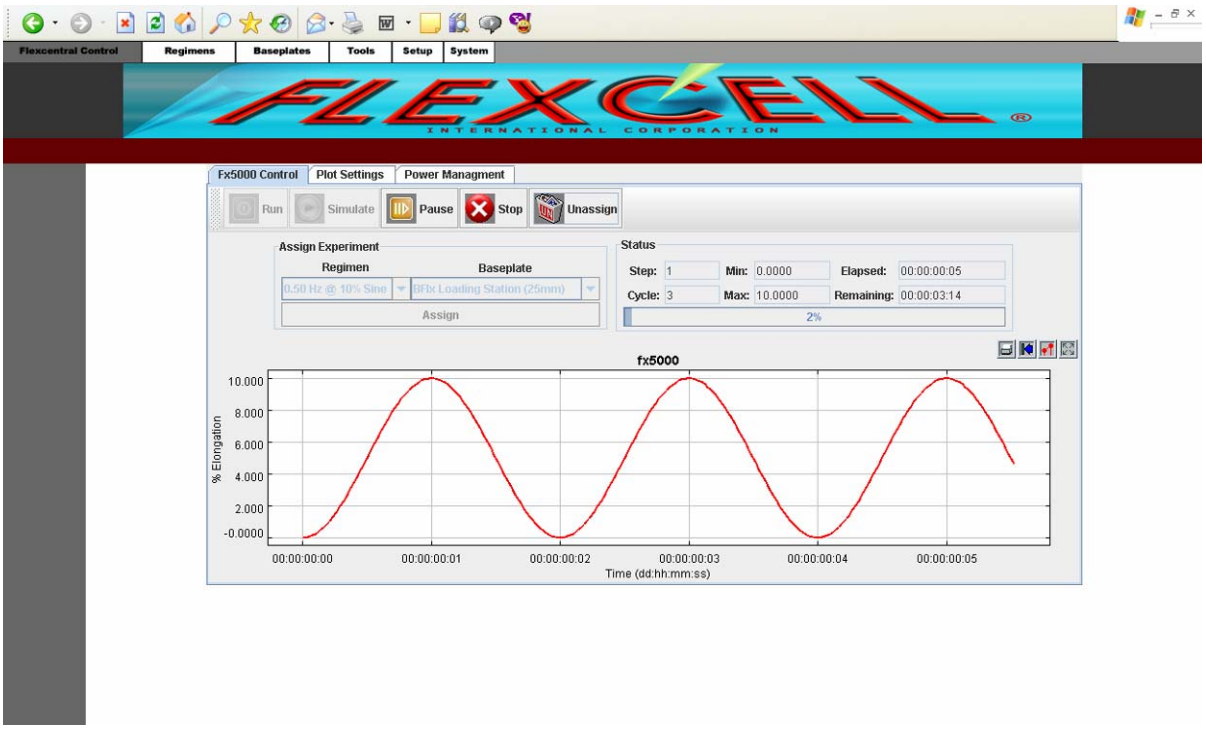


Figure 23. control.php

3.2.2 Regimen.php

The regimen.php provides a user interface for configuring experiments (Regimens). Here the user can create new experiments, duplicate experiments, and delete experiments.

The web page communicates with the pneumatic system SQL database to retrieve the experiment's name, type, number of steps and total duration from table tbl_regimens. Figure 24 is a screen shot of Internet Explorer interpretation of regimen.php.

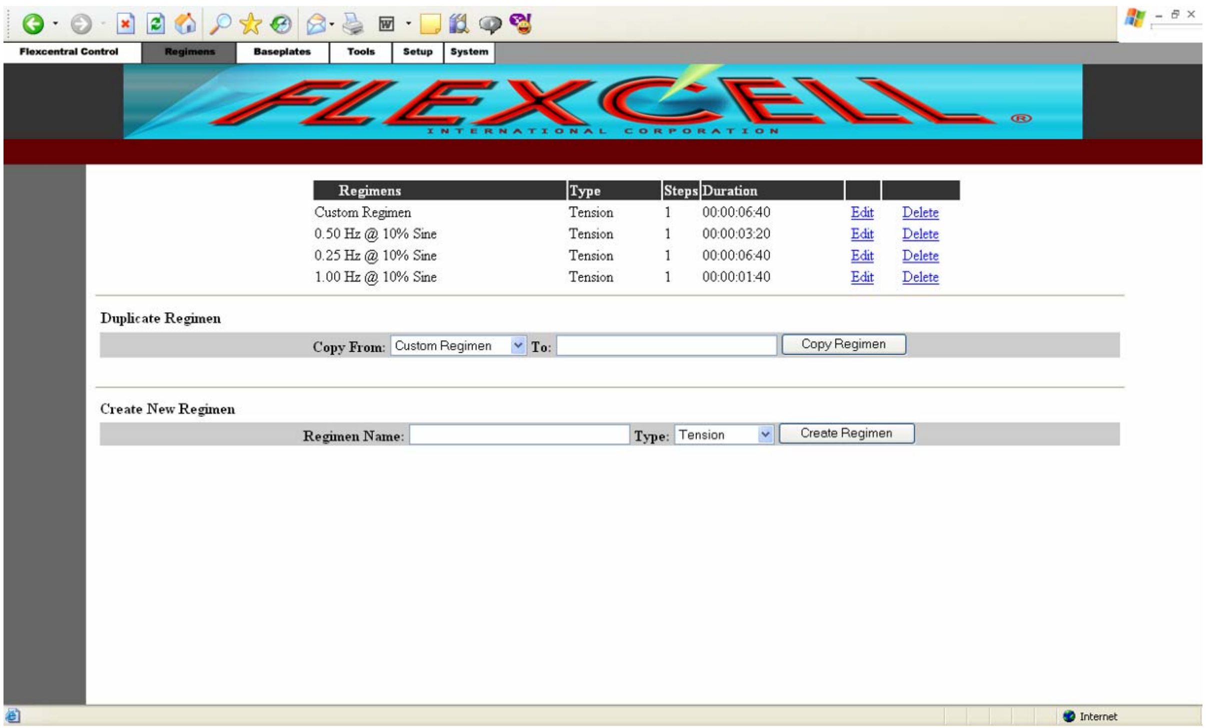


Figure 24. regimen.php

3.2.3 Regimen_step2.php

By selecting **[Create Regimen]** or **[Edit]** the user is redirected to regimen_step2.php (Figure 25). This webpage is use to configure waveforms. Here the user can select the waveform shape, frequency, minimum percent elongation, maximum percent elongation, and duty cycle. The combo box containing the waveform shapes are generated from an SQL query to the table tbl_waveforms. Depending on the selected shape, the other fields are enabled or disable. For example selecting a sinusoidal waveform disabled unnecessary fields such as duty cycle. If a desired waveform shape is not visible custom waveform shape can be generated using the custom waveform template (see Section 3.2.6 WaveTemplate.jar). The waveform is repeated continuously for the number of cycles indicated in the cycle text field. The waveform duration field is automatically computed using the frequency and the number of cycles. The **[Back To]** and **[Repeat]** fields are useful when an experiment contains multiple steps (waveforms). For example, in Figure 25, the **[Back To]** and **[Repeat]** fields was used to create an experiment that applies a 0.5 Hz sinusoidal waveform for one hour a day for 7 days.

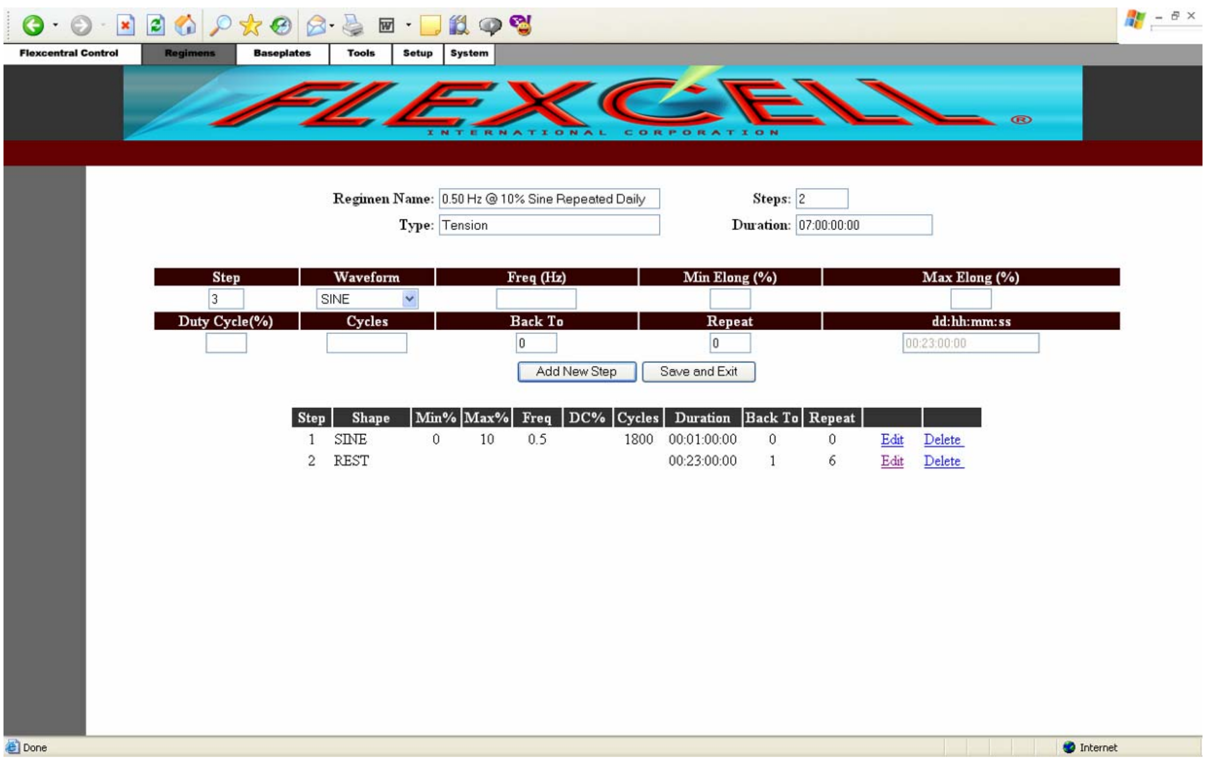


Figure 25. regimen_step2.php

3.2.4 Baseplate.php

The Baseplate.php provides methods for configuring the baseplates equations. This webpage, as shown in Figure 26, allows user to create, delete, and edited baseplates data. The baseplate equations represents and polynomial relationships between pressure and percent elongation. Section 4.4 Communication Thread provides more information about these relationships.

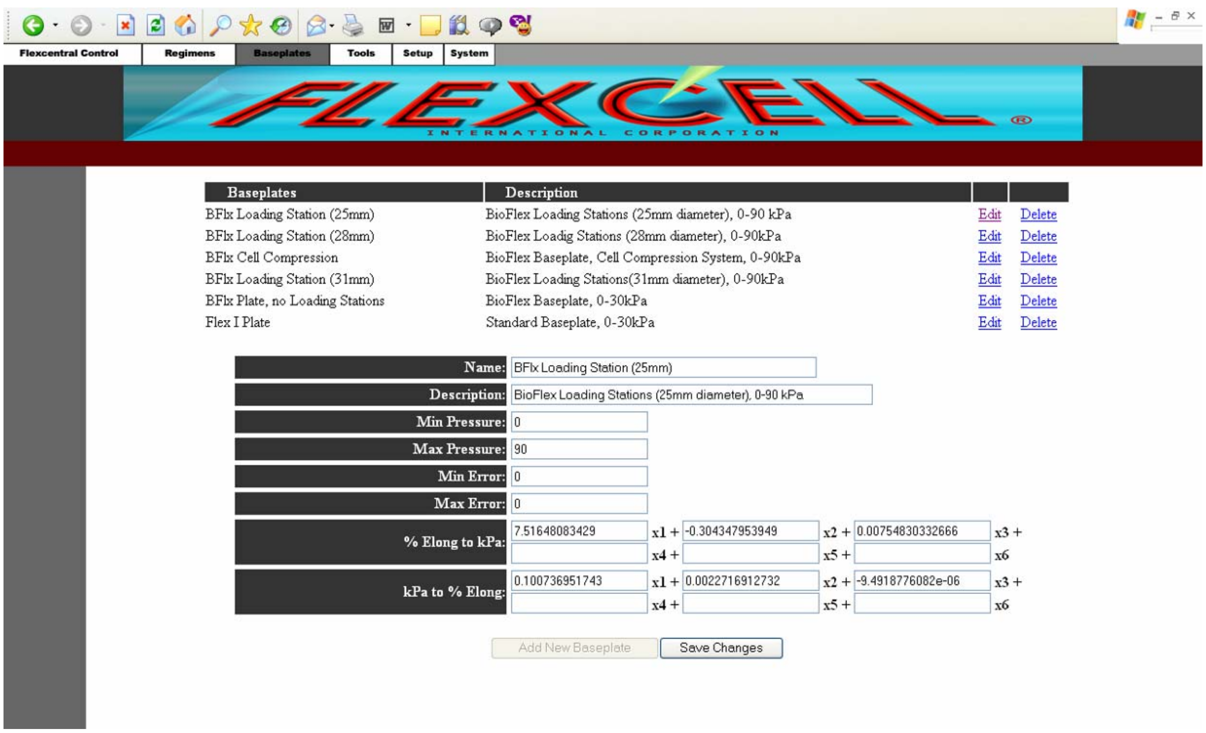


Figure 26. baseplate.php

3.2.5 Calc.jar

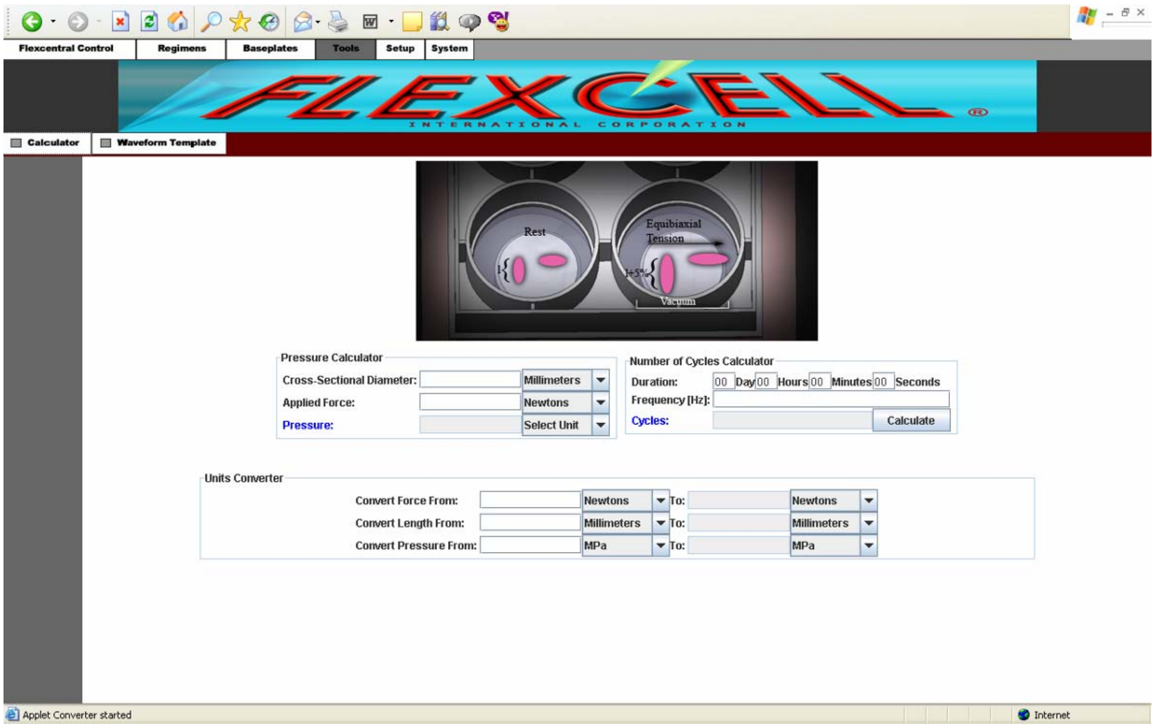


Figure 27. calc.jar

The Calculator applet (Figure 27) is design to aid users while creating new regimens. This tool contains features for calculating pressure, number of cycles, and unit conversions. The pressure calculator is use for calculating pressure from the cross-sectional diameter, and the applied force. The units for the cross-sectional diameter can be selected from the combo box to have a value of millimeters, centimeters, and inches. The pressure is calculated using the following equation.

$$\text{Pr}\,essure = \frac{Force}{Area}$$

Equation 9. Pressure Equation

The applied forces units can be in Newton of lbsforce. The output pressure can be calculated in MPa, kPa, Pa, Torr, ATM, Bar, or PSI.

The “Number of Cycles Calculator” tool is uses to computer the number of cycles given the duration and frequency. To make the tools more user-friendly the duration is enter in number of days, hours, minutes, and seconds. From this information the duration in seconds is computed. The numbers of cycles is computed by multiplying the frequency and the duration. Below is snippet of code that performs this operation.

```
public int calcCycles(){
    int retval = 0;
    int days, hours, mins, secs;
    double duration = 0.0;
    double freq = 0.0;
    try {
        days = Integer.valueOf(txt_day.getText());
        hours = Integer.valueOf(txt_hour.getText());
        mins = Integer.valueOf(txt_min.getText());
        secs = Integer.valueOf(txt_secs.getText());
        duration = days*86400.0 + hours*3600.0 + mins*60.0 + secs;

        freq = Double.valueOf(txt_freq.getText());
        double cycles = Math.ceil(freq*duration);
        retval = (int) cycles;
    }
    catch (Exception e) {
    }
    return retval;
}
```

3.2.6 WaveTemplate.jar

As mentioned in Section 3.2.3 Regimen_step2.php, the WaveTemplate applet is used to create custom waveform shapes. The WaveTemplate applet is visible in Figure 28. This applet allows the user to create custom shapes using either their mouse or importing data from a CSV documents. The CSV is a universal format for ASCII data that the user can be created, imported, and edited into most popular spreadsheets and databases, such as Excel.

Mouse clicks are used to create the red dots on the screen. The dots are connected together to form a continuous waveform. Using the GUI's radios buttons the points can be move, deleted, or added. Once the user save the data, the user can create regimens with the new custom shape as described in Section 3.2.3 Regimen_step2.php.

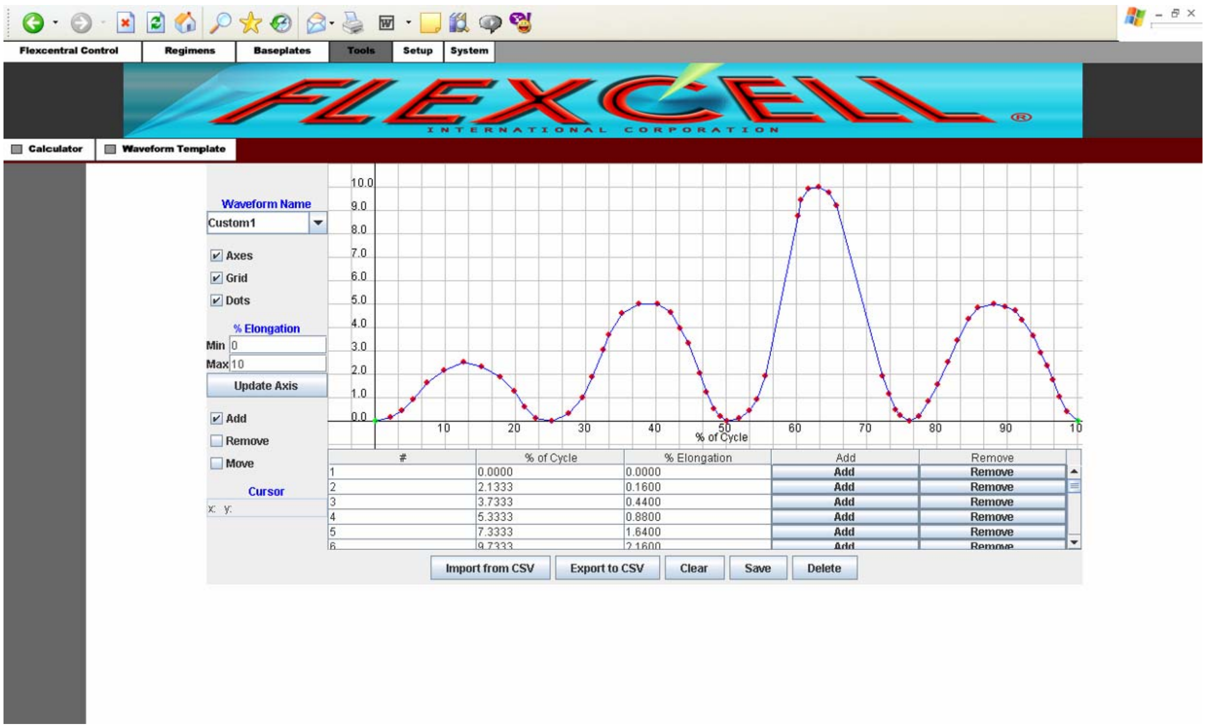


Figure 28. WaveTemplate.jar

3.2.7 Users.php

The Users.php webpage allows one to configure the system's user accounts. The username and password are used to login into the system and for experiment protection. The user information is stored in the SQL table tbl_users. If a particular user assigns an experiment only that user has permission to control the experiment. The other user account can only monitor the system, until the master user assigned user unassignes the experiment.

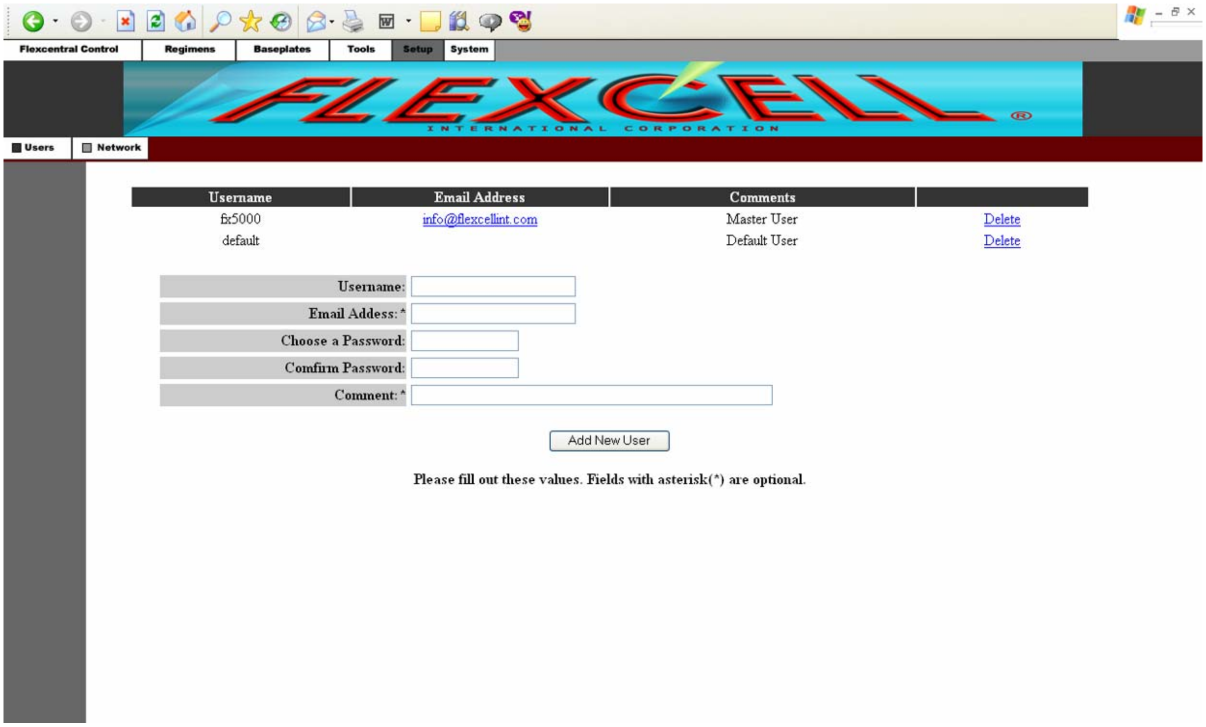


Figure 29. users.php

3.2.8 Network.php

The Network.php, shown in Section 3.2.8 Network.php allows user to configure the network connection remotely. This webpage operates similar to the Network Config Pane in Section 3.1.4.5 Network Config Pane.

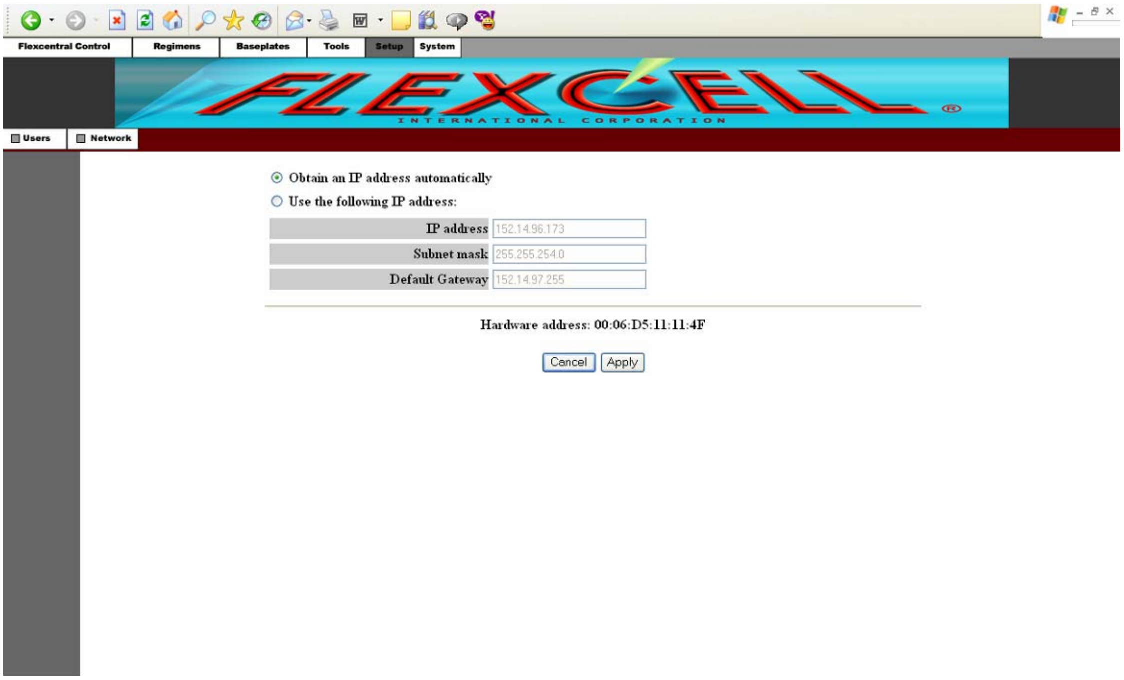


Figure 30. network.php

3.2.9 Setup.php

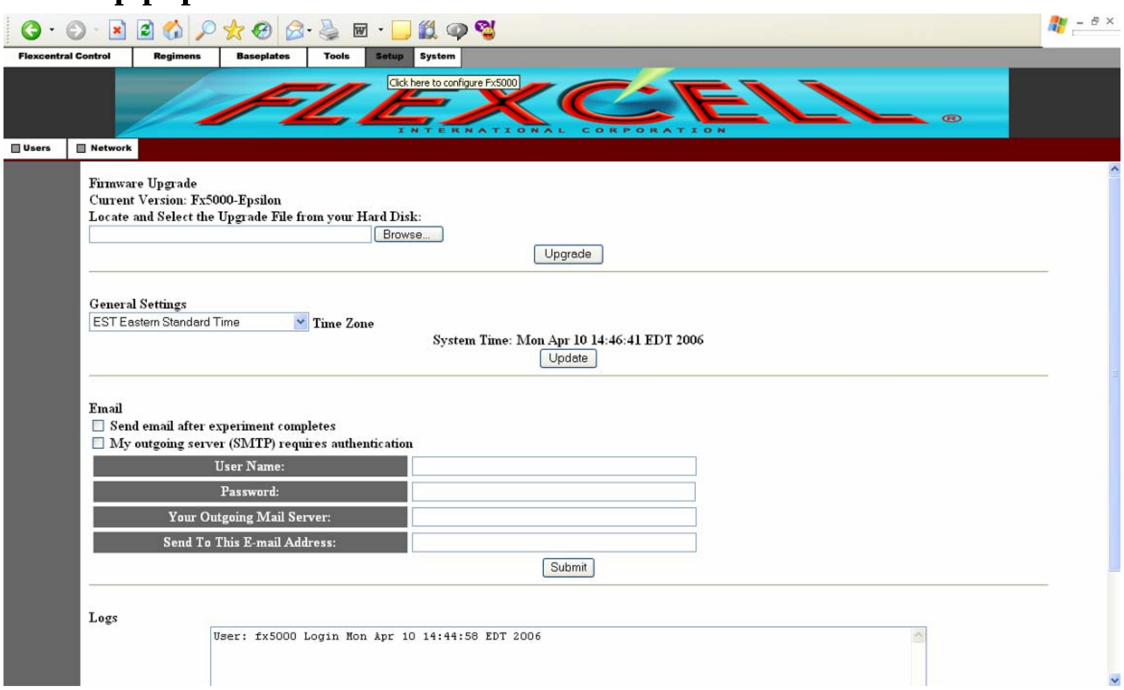


Figure 31. setup.php

The setup.php webpage allows for a verity of features such as upgrading the system software, time, configuring email, and displaying log files.

3.2.9.1 Software Upgrades

This webpage allows the users to install updates and features to the system. The **[browse]** button is use to select the upgrade file. The upgrade must be a compressed tar file (e.g. tar.gz). The upload file must contain a subfolder name 'firmware' and the files 'firmware/makefile' and 'firmware/version.txt'. If the upload file does not meet these requirements the file is automatically deleted. Once the file is successfully uploaded the file is decompressed and the Linux command *make* is performed on the makefile. 'version.txt' contains the current software version. Below is the procedure of making an upgrade files within Linux.

Step 1. Create firmware directory	mkdir firmware
Step 2. Create makefile	nano firmware/makefile
Example makefile (where setup.php and fx5000server are placed in the firmware folder)	#makefile #fx5000 Upgrade #modified setup.php and fx5000server all: update_web conf_server update_server update_web: setup.php\ ; mv setup.php /www conf_server: \ ; chmod 777 fx5000server #this will requires a restart to take effect update_server:\ ; mv –f fx5000server /usr/sbin/fx5000server
Step 3. Create version.txt	nano firmware/version.txt
Example version.txt	Fx5000-distro1

Step 4. Create tar file	tar cvfz fx5000-distro1.tar.gz firmware/*
Step 5. Distribute upgrade	Using scp upload tar file from development system and distribute file to other systems.

3.2.9.2 Configuring Time

This feature allows the user to set the system time. The combo box contains a list of the popular time zones. The **[update]** button is use to set the time. Once the **[update]** button is pressed the system time is synchronized with the 'time.nist.gov' time server. The system time is automatically adjusted for daylight saving times.

3.2.9.3 Email Configuration

The system has the capability of sending emails. This feature is configured using this webpage. If enabled the Email class is invoked at the end of the run to create and send an email to the configured address. The email configurations are stored in the SQL table tbl_config.

3.2.9.4 Logging

The system keeps a log file of system events such as a login history to the system, starting an experiment and stopping an experiment. These events appear in the text area located at the bottom of the users.php webpage.

Chapter 4. Software Control Threads

The user interfaces interact with the system components through system threads running on the pc-104. Although the pc104 performs other operations, such as providing the user interface; the main purpose for this device is to provide control pressure to the baseplates. It is important that other tasks do not interfere with the performance of the control process. In order to ensure that the control performance, the software makes use of the following:

- C/C++ programming language
- Multithreading
- Thread Priorities
- Operating System Scheduler

Although Java is used to implement the user interface it is not appropriate language for running real-time operations. Java applications and applets are executed using the Java Runtime Environment (JRE). The JRE provides additional overhead which increased processing time and memory. Real-time execution is mostly performed with low level assembly languages. Higher levels language such as C is also acceptable for a real-time programming. The IAPCI systems software is written in a combination of C and C++. C is use to perform valve control, sensors readings, and the PID algorithm. C++ is used to make the program object oriented, adaptable, and user friendly.

The program is divided into multiple threads to speed up execution. Using a general-purpose thread Pthreads library the program was split into the following threads: Main

Thread, Data Storage, Socket Parser (Communication), and Control. A block diagram of these threads is found in Figure 32. Linux operating system is reconfigured to use a priority round-robin scheme. This scheduler allows for the processor to effetely switch between the multiple processes (threads). Each thread is given a priority. Higher priority threads are given more CPU time than lower priority threads. The Control thread is given a priority of 90 out of the maximum priority of 99. If the priority is set too high the operation system is unable to perform other tasks including user interface functionality. A low priority could impact the sampling rate of the control system, which was introduces error in controlling pressure.

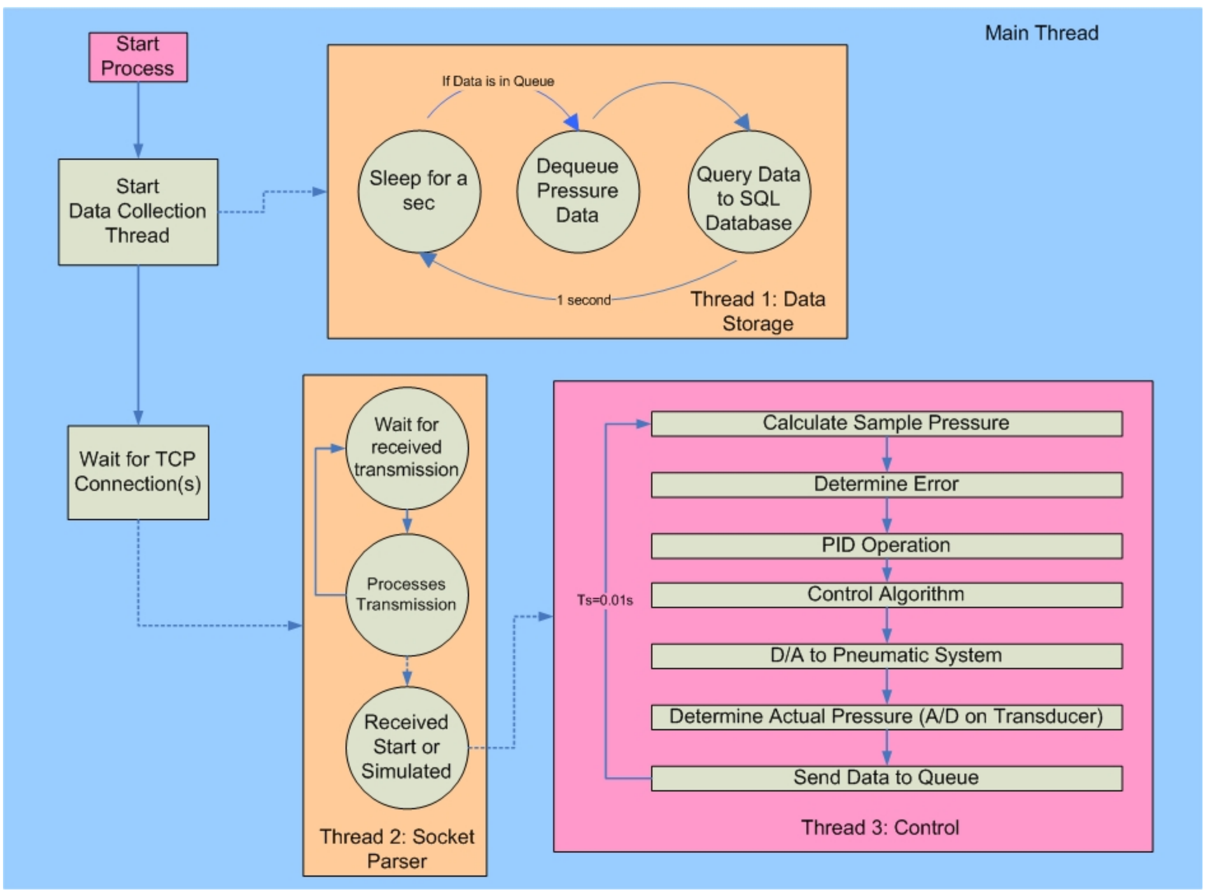


Figure 32. Fx5000Server Flow Diagram

4.1 Main Thread

The main thread is automatically started during boot up. The boot up process is explained in more detail in Appendix A.2.1 Modified Linux Startup Routine. This thread is use to initialize the DAQ drivers and socket communication. The primary task of this thread is to listen for incoming connections from the user interface clients.

4.2 Data Storage

The data storage thread is used to communicate with the SQL database. SQL queries are used to supply the user interface clients with data such as waveform plots and etc. This thread contains a link list of SQL queries. Queries are added to the queue every other sample (sample time is approximately 0.01 seconds) from the Control Thread. Every second the queries are dequeued and executed. While observing the system, it was notice that the SQL queries require intense amount of CPU time. These queries must be performed in a separate process to reduce the impact on the control algorithm.

4.3 Control Thread

The control thread is the most important thread. This thread executes an algorithm for feedback pressure control. This algorithm is discussed in Chapter 5. Control System

4.4 Communication Thread

This thread checks for received messages from connected user clients. Sockets are used for communication between clients and servers. Sockets allow data transfer across a network using TCP/IP and UDP protocols. The TCP protocol uses handshaking for reliable data transfer. When using handshaking an acknowledge packet is returned with each transfer, to confirm the transaction. This protocol will continue to transmit data until that transfer is complete or a time out event occurs. The UDP protocol does not perform handshaking. UDP protocol is mostly used for streaming data, in instances where lost packets are acceptable. The pneumatic system server uses TCP socket for its communication protocol. It is important to have a reliable transfer of data between the client and server. For example if the client tells the server to stop the experiment the server must receive this information. The server broadcasts it acknowledged packets to all of the connected clients. This is necessary for client synchronizing. Figure 33 display the synchronizing of the two clients and the server, where an experiment is started by one client and stopped by another client.

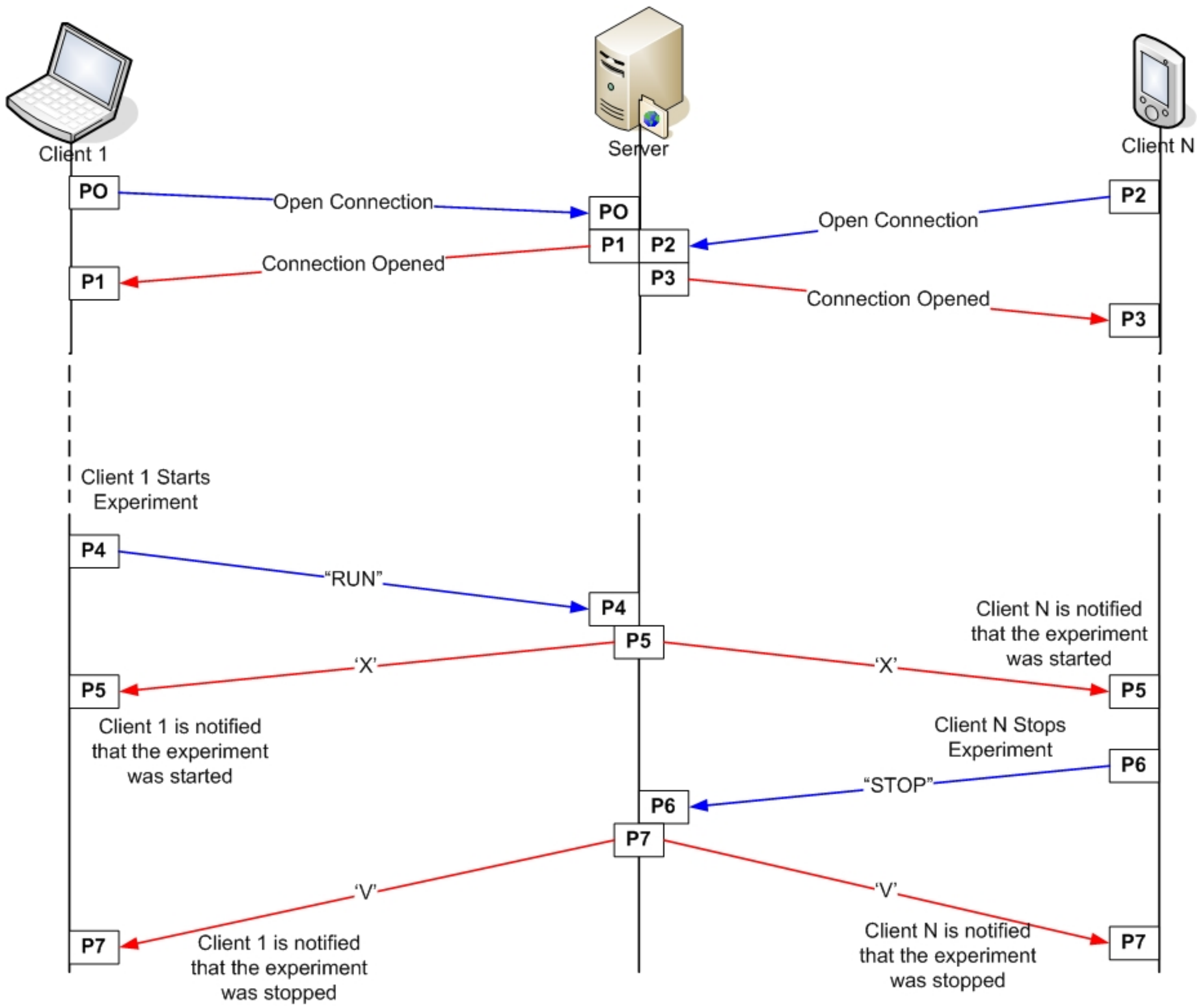


Figure 33. Network Protocol

As the clients connect to the server they are given a unique session id. These ids are stored by the system in a link list. If the client drops its connection with the server, the session id is removed from the list. If the client reconnects later the client would be given different session id. Servers receives packet from the client, where a packet is process. Acknowledgement of a packet received is sent to each client that is in the session id list. In Figure 33, **[P#]** represents the communication packet. The figure displays the packet traveling across the network and the server acknowledgement of the receipt of the packet. Valid communication packets from client to server are found in Section 4.5.1 Client to Server Packets. Valid communication packets from the server to client are found in Section 4.5.2 .

4.5.1 Client to Server Packets

- **RUN –** This packet is sent once a client presses the run button.
- **KEEPALIVE –** This packet is sent every 5 minutes to keep the connection alive.
- **PAUSE –** This packet is sent once a client presses the pause button.
- **RESUME –** This packet is sent once a client presses the resume button.
- **STOP –** This packet is sent once a client presses the stop button.
- **RESET –** This packet is sent once a client presses the reset button.
- **SIMULATE –** This packet is sent once a client presses the simulate button.
- **ASSIGN** [Regimen Name] [Baseplate Name]

  – This packet is sent once a client presses the assign button. It is followed by a packet containing the regimen name and a packet containing the baseplate name.
- **QUIT –** This packet is used to stop the server.
- **UNITS –** This packet is sent if the client changes plotting units. It is followed by one of the following packets.
  - ***ELONG*** **–** This packet is sent to notify a unit change request to percent elongation.
  - ***PRESS*** **–** This packet is sent to notify a unit change request to pressure.
  - ***STRAIN*** **–** This packet is sent to notify a unit change request to strain.

    - ***MSTRAIN* –** This packet is sent to notify a unit change request to milli strain.
    - ***USTRAIN* –** This packet is sent to notify a unit change request to micro strain.
- **SMOOTHING** – This packet is sent if the client change the state of plot smoothing. It is followed by one of the following packets.
    - ***ENABLED* –** This packet is sent to enable smoothing.
    - ***DISABLED* –** This packet is sent to disable smoothing.
- **GRAPHACTUAL -** This packet is sent if the clients enables or disable the plot of the actual waveform. It is followed by one of the following packets.
    - ***TRUE*** – This packet is sent to enable plot of actual waveform.
    - ***FALSE* –** This packet is sent to disable plot of actual waveform.
- **GRAPHEXPECTED -** This packet is sent if the client enables or disables the plot of the desired waveform. It is followed by one of the following packets.
    - ***TRUE* –** This packet is sent to enable plot of desired waveform.
    - ***FALSE* –** This packet is sent to disable plot of desired waveform.
- **EMAILNOATTACH –** This packet is sent to send email with an attachment.
- **EMAILWITHATTACH -** This packet is sent to send an email with attachments.
- **UPDATEPOWER –** This packet is sent to notify the system that the external power configurations where modified.

4.5.2 Server to Client Packets

- **'Q'** - This packet is sent to confirm the termination of the experiment.
- **'V'** - This packet is sent to disable user interface buttons.

- **‘A’** - This packet is sent to confirm the assignment of the experiment**.**
- **‘X’** - This packet is sent to confirm the start of the experiment.
- **‘S’** - This packet is sent to confirm the start of a simulation.
- **‘P’** - This packet is sent after the experiment or simulation was paused.
- **‘R’** - This packet is sent after the experiment or simulation was resumed.
- **‘Y’** - The packet is after a reset (un-assignment) occurred.
- **‘E’ –** This packet is sent to confirm as the units change to percent elongation.
- **‘B’** – This packet is sent to confirm as the units change to pressure.
- **‘C’ –** This packet is sent to confirm a unit change to strain.
- **‘D’ –** This packet is sent to confirm as the units change to milli strain.
- **‘F’** – This packet is sent to confirm as the units change to micro strain.
- **‘G’** - This packet is sent to confirm that plot smoothing is enabled.
- **‘H’** - This packet is sent to confirm that plot smoothing is disabled.
- **‘N’** - This packet is sent to confirm that the plotting of the actual waveform.
- **‘O’** - This packet is sent to confirm the removal of the actual waveform.
- **‘T’** - This packet is sent to confirm that the plotting of the desired waveform.
- **‘U’** - This packet is sent to confirm the removal of the desired waveform.
- **‘W’ –** This packet is sent to disable menu buttons.
- **‘Z’** - This packet is sent to acknowledge client’s keep alive packets.

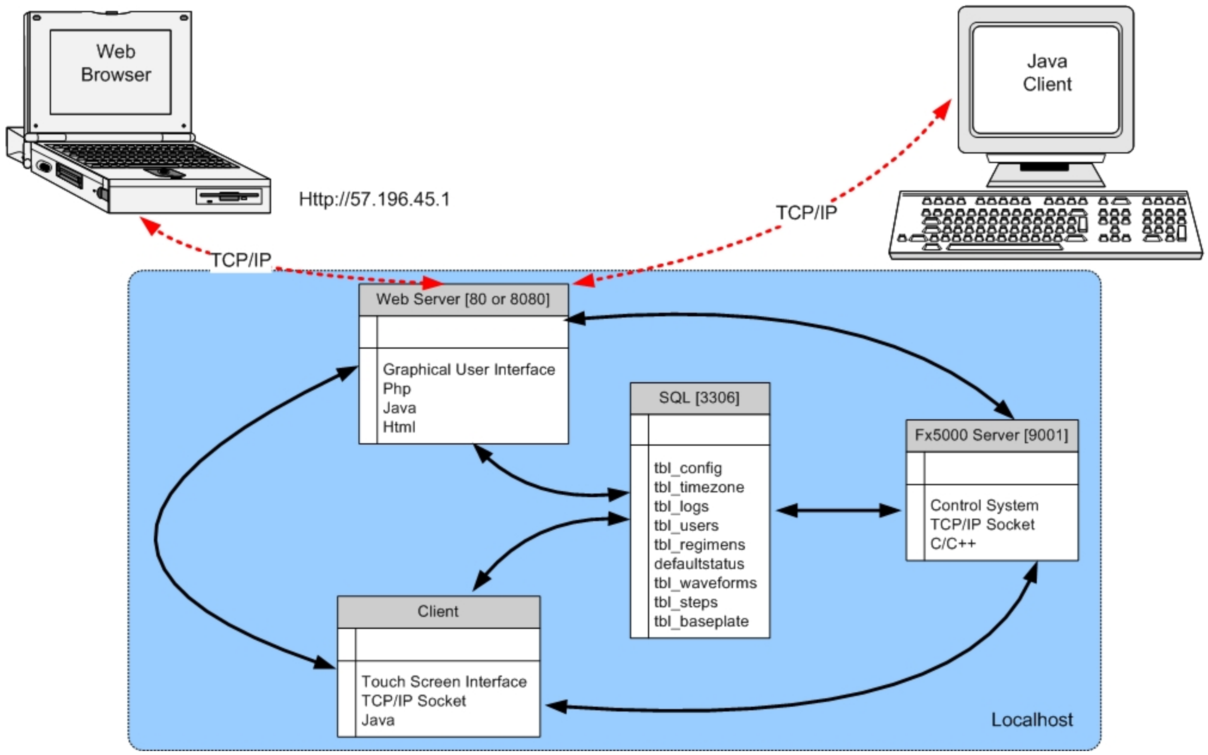


Figure 34. Communication Flow Diagram

Chapter 5. Control System

This chapter presents the approaches to designing a controller for a pneumatic system. Hardware devices from Chapter 2. IAPCI Design and software implementations from Chapter 4 and 5 are combined to create the feed back pressure control system shown in Figure 35.

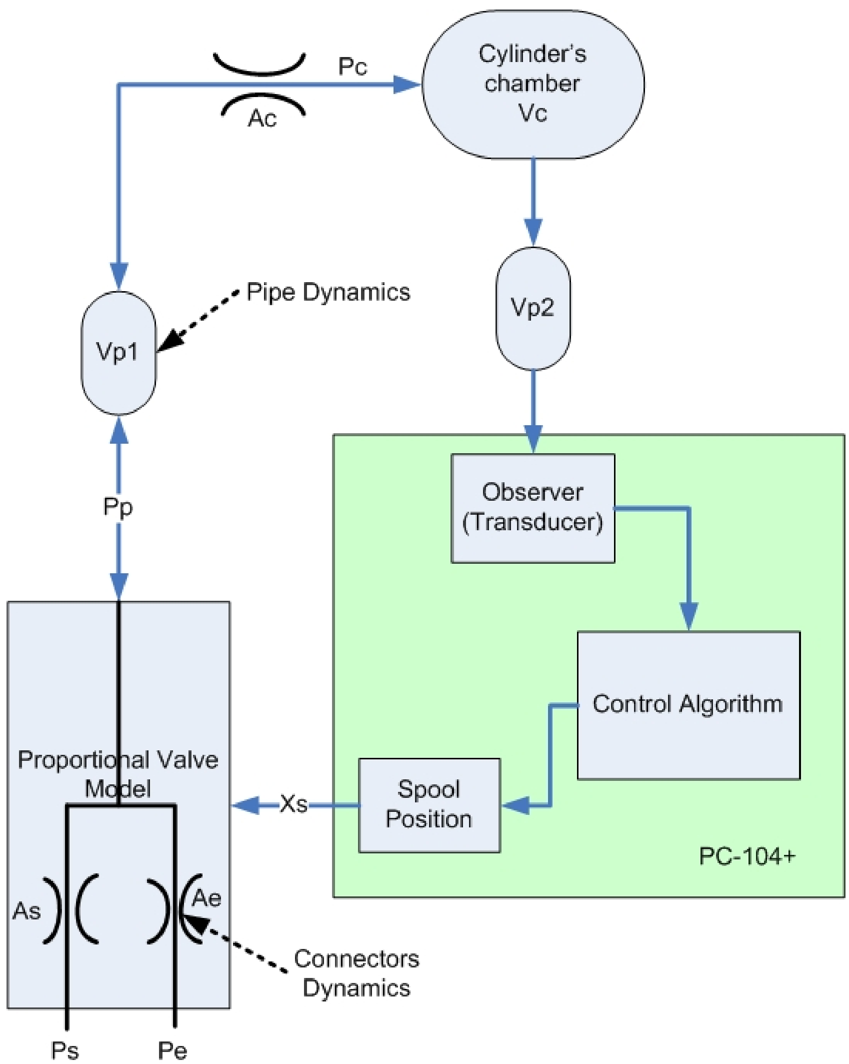


Figure 35. System Model

Ps and Pe are the supply pressure and the environment pressure. A vacuum pump is used as the supply source (Ps) to control pressure below the atmospheric pressure. An air compressor is used as the air supply source; to ensure the control pressure is above the atmospheric pressure. The environment pressure (Pe) is open to atmosphere (free exhaust). The pressure sources are connected to the proportional valve whose flow direction and orifice size is controllable through displacing the internal spool (Xs). Tubing is used to connect and deliver air from the valves to the chamber (baseplates). Tubing connecters, diameters, and length all affect the system response [8, 9]. In Figure 35, As, Ae, and Ac represents dynamic model of connectors. Vp1 and Vp2 represent the dynamic model of the tubing.

In a model similar to that of Figure 35, Pascal Bigras [8, 9], derived the equations that models the system behavior. Using the follow assumptions, Bigras developed relations found in Equation 10 [8, 9]:

- Gas is ideal.
- Gas density is uniform in the chamber and in the pipe.
- Gases in the chamber and in the pipe are isothermal processes.
- Flow in the valve and in the connection port is isentropic with negligible temperature variation.
- Flow leakages are negligible in the valves.

$$\dot{P}_c = -\frac{\dot{V}_c}{V_c} P_c + \frac{RT_c}{V_c} \dot{m}_c\left(P_p, P_c\right) \quad \dot{P}_p = -\frac{RT_p}{V_p}\left(\dot{m}_p\left(P_p, x_s\right) - \dot{m}_c\left(P_p, P_c\right)\right)$$

$$\dot{m}_c\left(P_p, P_c\right) = \begin{cases} \frac{C_c A_c P_p}{\sqrt{RT_p}} f_r\left(\frac{P_c}{P_p}\right) if P_p \geq P_c \\ -\frac{C_c A_c P_c}{\sqrt{RT_c}} f_r\left(\frac{P_p}{P_c}\right) if P_p < P_c \end{cases} \quad \dot{m}_p\left(P_p, x_s\right) = \begin{cases} \frac{C_s W_s x_s P_s}{\sqrt{RT_s}} f_r\left(\frac{P_p}{P_s}\right) if x_s \geq 0 \\ \frac{C_s W_s P_p}{\sqrt{RT_p}} f_r\left(\frac{P_e}{P_p}\right) if x_c < 0 \end{cases}$$

$$f_r(y) = \begin{cases} \sqrt{\frac{2\gamma}{\gamma - 1}} \sqrt{y^{2/\gamma} - y^{(\gamma+1)/\gamma}} \; if y \geq r_c \\ \sqrt{\gamma \left(\frac{2}{\gamma + 1}\right)^{(\gamma+1)/(\gamma-1)}} \; if y < r_c \end{cases} \quad r_c = \left(\frac{2}{\gamma + 1}\right)^{\gamma/(\gamma-1)}$$

P_c, V_c, T_c and $\dot{m}_c$ are respectively the pressure, the volume, the temperature and the mass flow in the chamber

P_p, V_p, T_p and $\dot{m}_p x_s$ are respectively the pressure, the volume, the temperature and the mass flow in the pipe

x_s is servo valve spool position

R is ideal-gas constant

A_c is the geometric orifice area of the connection port

$W_s X_s$ is the geometric orifice area of the servo valve

P_s and T_s are respectively the pressure and the temperature of the air supply

P_e and T_e are respectively the pressure and the temperature of the atmosphere

C_c and C_s are the orifice discharge coefficients

r_c is the critical pressure ratio

γ is the ratio of specific heats of gaseous medium

Equation 10. Fluid Dynamics Equations

One of the drawbacks for using pneumatics to control strain is the complex system behaviors. In observing Bigras' equations (Equation 10) it is noted that the system behavior is highly non-linear and contains uncontrollable parameters such as temperature, and supply pressure. The IAPCI system requires adaptable control, it must be capable of using a variety of vacuum pumps or compressors; each with their own specifications. For example, more expensive vacuum pumps can deliver lower pressures than their cheaper counterparts and pipe lengths and diameters can vary from system to system. These changes in parameters would have to be inserted into the equations in order to derive a system response. Solving these equations is complex. The equations can be linearized by ignoring certain parameters; but this technique increase error. Using control logic along with feed back control these equations can be eliminated. Section 5.1 Control Algorithm explains these algorithms.

5.1 Control Algorithm

There are four different algorithms investigated for controlling system pressure. These algorithms are as follows: On/Off Controller, PWM Controller, Bleed Off Controller, and Proportional Controller.

5.1.1 On/Off Controller Design

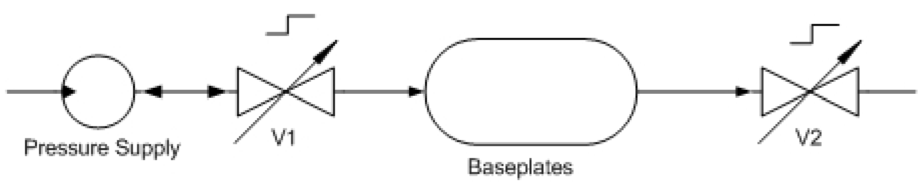


Figure 36. Pressure Model (on/off)

The first algorithm simply states if the pressure in the system is below the ideal pressure increase the supplied pressure; if the pressure is too high release pressure. By using this type of control, expensive proportion valves are replaced with cheaper fast switching solenoids. To adapted this logic to a pneumatic system, the actual pressure is computed using the transducer and then is compare to a desired pressure. The pressure error is calculated by subtracting the actual pressure from the desired pressure. If the system is using a compressor and the error is positive, there is too little pressure in the system and port 2 of the 5/3 valve (or V1 in Figure 36) is opened. If the system is using a vacuum source, a positive error indicates too much differential pressure. Figure 37 is a visual of the On/Off control algorithm response.

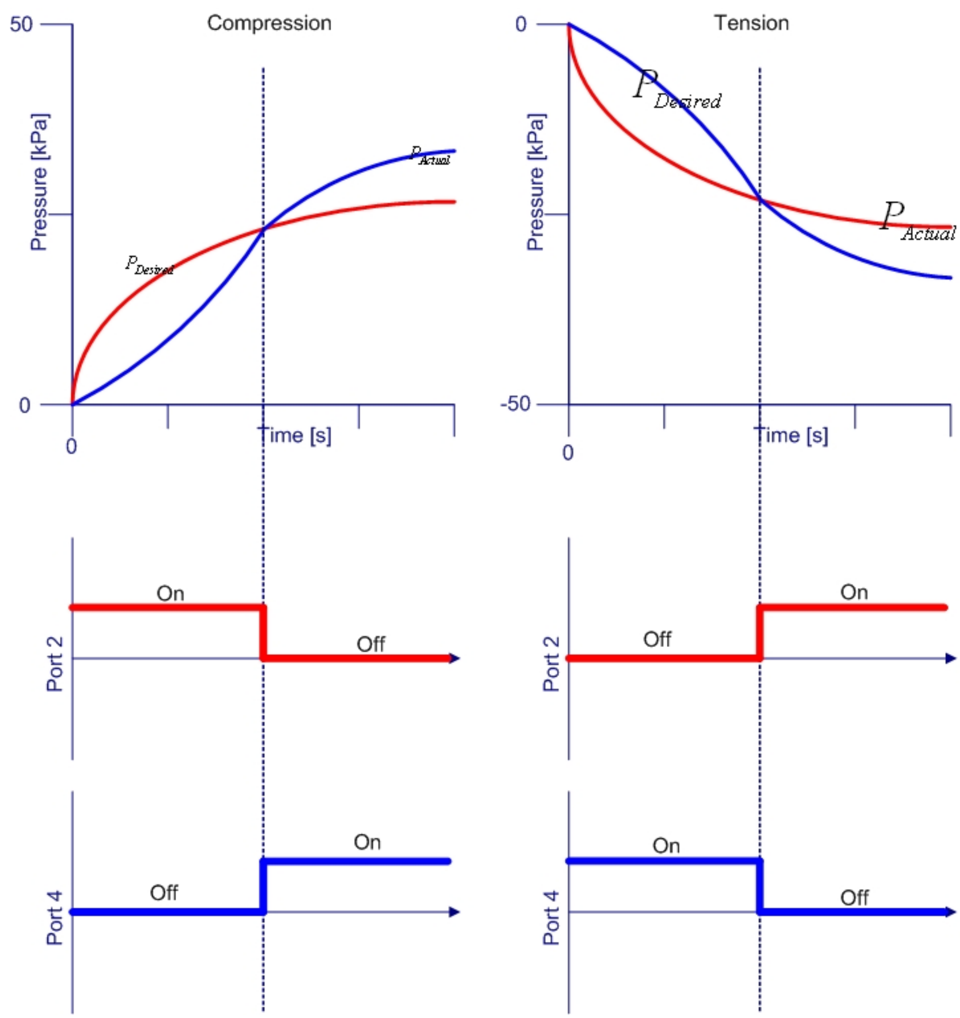


Figure 37. Valve Response (On/Off Control)

Due to disturbances, systems delays, and high resolution sensors, it is very rare that the desired pressure is reached. With on/off control the system will never reach a steady state value. The actual pressure will tend to oscillate around the desired pressure.

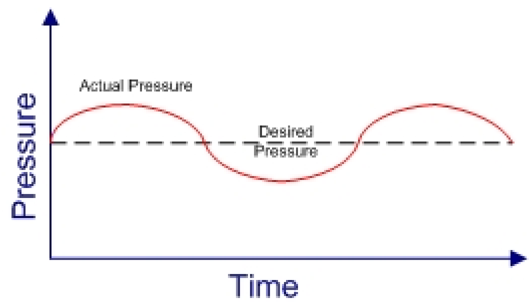


Figure 38. Steady State Error

5.1.2 PWM Controller Design

The amplitude of the oscillation may be reduced by pulsing the valves. This design still uses fast switching on/off solenoid valves, but with an input signal that is a square wave instead of logically high or low. A Pulse Width Modulation (PWM) signal is sent to the valves with a frequency much higher than the sampling rate (frequency >> 100 Hz). The duty cycle of the pulse is controlled by the error. A small error between desired and actual pressure will result in a low duty cycle (Almost Off). Large errors would increase the duty cycle until it signals becomes logically high. Additional research is needed to determine if the PWM signal has an effect on the lifetime of the valves.

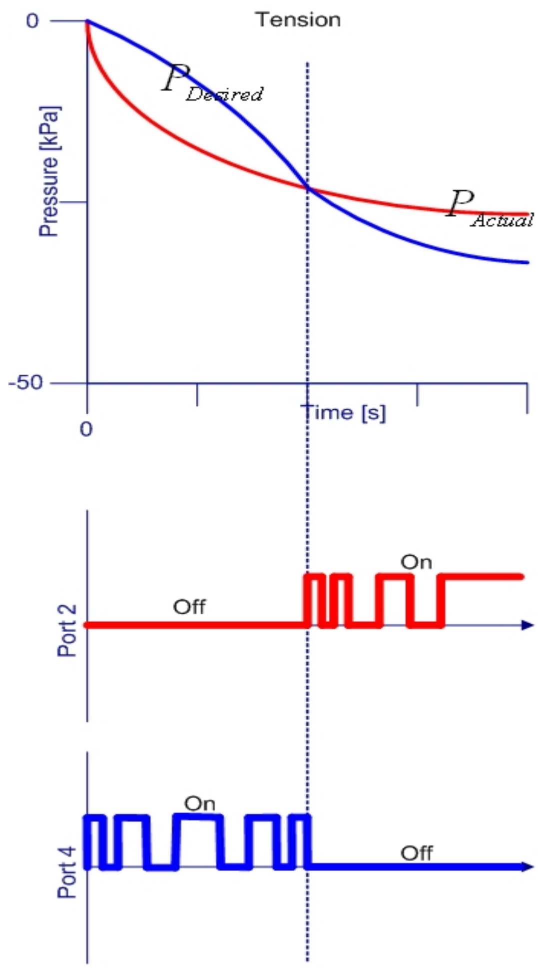


Figure 39. Valve Response (PWM Control)

5.1.3 Proportional Bleed Off Control

Another control algorithm investigated involves the use of two valves in a 'T' configuration, see Figure 40. This control algorithm uses proportionally control valves. The first valve (V1) is control by a step input. In most cases the first valve is always open. The second valve (V2) is use to control the pressure of the system. If the pressure in the baseplates is too large V2 is opened (See Figure 41) to reduce the pressure. If the actual pressure is lower than the desired pressure the valve is closed (See Figure 42) to increase the system pressure.

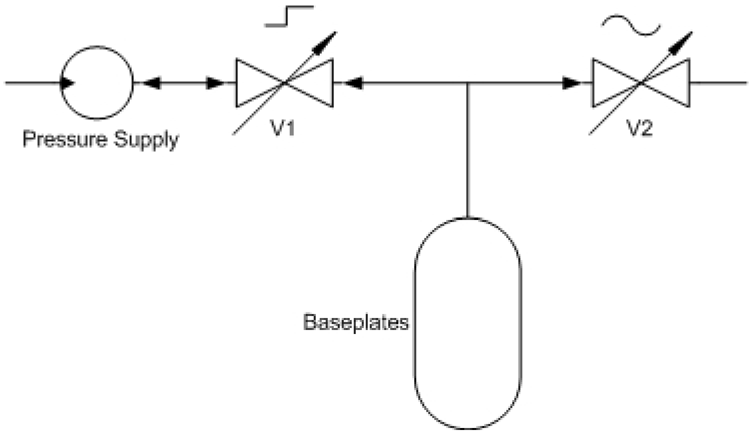


Figure 40. Bleed Off Control Valve Configuration

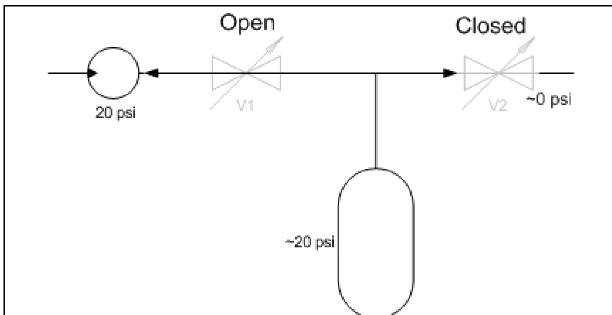
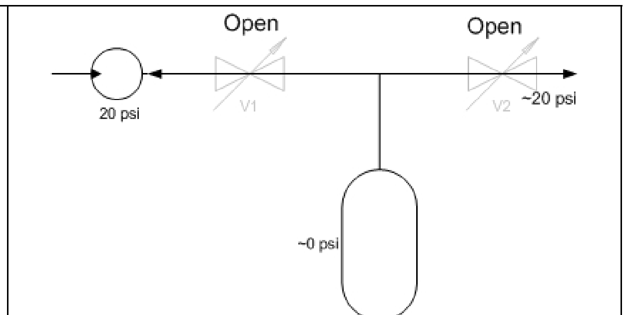
	

Figure 41. Bleed Off Control V2 Closed	**Figure 42. Bleed Off Control V2 Open**

This control algorithm has some disadvantage. Controlling very low pressure is difficult. When valve 2 is completely open, air still find their way to the baseplates. To completely exhaust the baseplates valve 1 would have to be shut off. Now the algorithm has to control both valve 1 and valve 2; which increases the complexity of the algorithm. Another potential problem is the drain on the supply source. This algorithm requires a constant supply of air flow into valve 1. The majority of the time Valve 2 is somewhat open and exhausting air. A large pressure reservoir is needed to keep the input pressure constant. Even with a large reservoir the pump is working harder than the previous algorithms.

5.1.4 Proportional On/Off Controller Design

The implemented control algorithm is similar to the PWM algorithm described in Section 5.1.2 PWM Controller Design. The on/off valves are replaces with a proportional valve (discuss in Section 2.3 Pneumatic Valve), and the PWM signal is replaces with an analog voltage. The input voltage is determined by the pressure error calculations. Figure 43 show the relationship between pressure error and valve input voltage and Figure 44 displays the error relationship between actual and desired pressures.

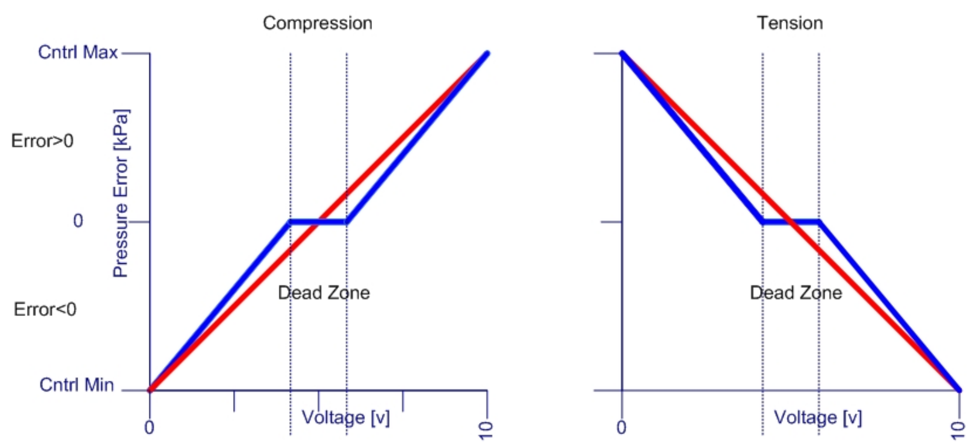


Figure 43. Error vs. Input Valve Voltage

In compression if the error is positive the baseplate pressure is too low. The system pressure is increased with an input voltage in the range of 0 volts to 5 volts. A low voltage would increase the pressure at a greater rate than 5 volts. If the pressure error is negative the system pressure is too high. The system pressure is reduced by applying an input voltage from 5 – 10 volts. An input voltage near 10 reduces pressure faster than a voltage near 5.

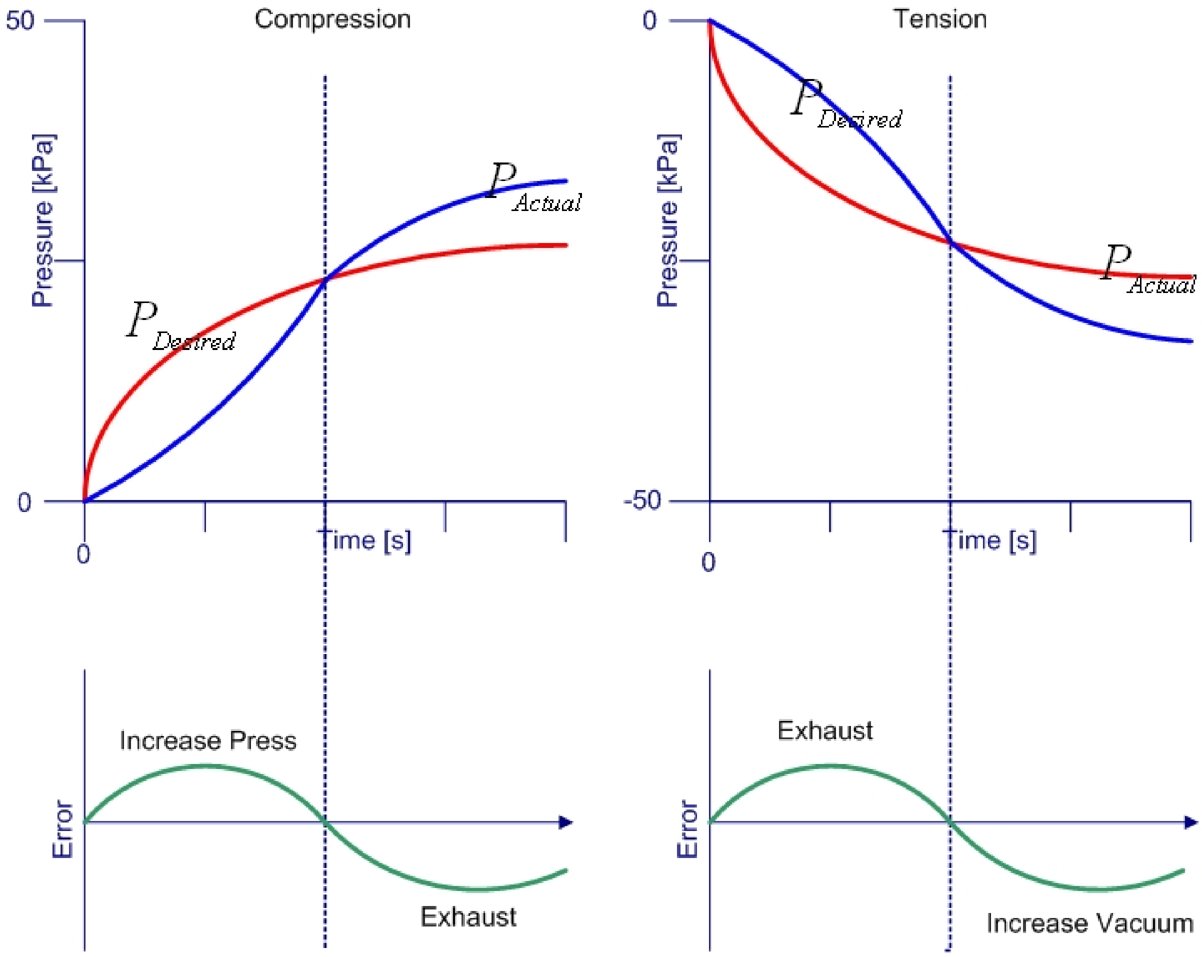


Figure 44. Pressure Error Relationship

The dead zone in Figure 43 has been exaggerated. The dead zone represents inputs (voltages) that there is little to no system response. This dead zone exist near the set point 5 where all ports are closed. This dead zone is determined experimentally to be around 4.948 to 5.00525. The relationship in Figure 43 is represented mathematically in Equation 11 where $X_d = 0.0525$, $CNTRL_{MIN} = \text{-}55.0$, and $CNTRL_{MAX} = 55.0$

$$Voltage(Input) = \begin{cases} \dfrac{(Input - CNTRL_{MAX})(10 - 2X_d)}{CNTRL_{MIN} - CNTRL_{MAX}}, Error > 0 \\ 5v, Error = 0 \\ \dfrac{(X_d - 5)Input}{-CNTRL_{MIN}} + (5 + X_d), Error < 0 \\ 5, Error = 0 \end{cases}$$

Equation 11. Tension Control Equations

5.1.4 Control Algorithm Summary

Table 10 compares the three algorithms inputs control signals using the Norgren valve. The simplest control technique but least effective methods are the on/off approach. The PWM control algorithm sound good on paper, but needs to future explored future. The pc-104 is capable of performing PWM but it is not capable of providing 5 volts offset needed to interact with the valves. An external offset circuit would have to be implemented. The preferred algorithm is the proportional on/off algorithm.

Table 10. Control Algorithm Summary

	On/Off Control		Pulse Width Modulation		Proportional On/Off Control	
	Port2	Port4	Port2	Port4	Port2	Port4
Below Desired Pressure, Increase Pressure	On (10V)	Off	0%<DC<100% (5V offset)	0% DC	On (5 - 10V)	Off
Reached Desired Pressure, Maintain Pressure	Off (5V)	Off (5V)	0% DC	0% DC	Off (5V)	Off (5V)
Above Desired Pressure, Decrease Pressure	Off	On (0V)	0% DC	0%<DC<100%	Off	On (0 - 5V)

5.2 Elongation and Pressure Relationship

The pressure seen by the transducer is not necessarily the same pressure as that in the chamber. In Bigras paper “Nonlinear Observer for Pneumatic System With Non Negligible Connection Port Restriction”, he compares the pressure variation in the chambers and in the pipe. According to his research there is a large degree of pressure difference from the pressure in the chamber and pressure in the pipe pressure connected to the transducer. To accurately control the chamber pressure a relations between the sensor and the chamber must be developed. In the journal article, “In vitro strain-induced endothelial cell dysfunction determined by DNA synthesis” and earlier work performed by Flexcell International calibration curves where derived by placing dots 1mm apart across the diameter of the membranes. Calibrated pressure was applied to the membranes and the elongation was measures using optical measuring devices. Equation 12 through Equation 21 is the resulting calibration equations provided by Flexcell International.

$$\mathrm{Pr}\,essure[kPa] = 7.51648(\%Elongation) - 0.30435\left(\%Elongation^{2}\right) + 0.00755\left(\%Elongation^{3}\right)$$

Equation 12. BFlx Loading Station (25mm): %Elongation to Pressure

$$Elongation[\%] = 70.10074(\mathrm{Pr}\,essure) + 0.00227\left(\mathrm{Pr}\,essure^{2}\right) - 9.49188\left(\mathrm{Pr}\,essure^{3}\right)$$

Equation 13. BFlx Loading Station (25mm): Pressure to %Elongation

$$\mathrm{Pr}\,essure[kPa] = 11.14286(\%Elongation) - 0.48083\left(\%Elongation^{2}\right) + 0.00865\left(\%Elongation^{3}\right)$$

Equation 14. BFlx Loading Station (28mm): %Elongation to Pressure

$$Elongation[\%] = 0.11091(\mathrm{Pr}\,essure) - 0.000734\left(\mathrm{Pr}\,essure^{2}\right) + 1.57935e^{-5}\left(\mathrm{Pr}\,essure^{3}\right)$$

Equation 15. BFlx Loading Station (28mm): Pressure to % Elongation

$$\text{Pressure}[kPa] = 35.05422(\%Elongation) - 7.13939(\%Elongation^2) + 0.711638(\%Elongation^3)$$

Equation 16. BFlx Loading Station (31mm): %Elongation to Pressure

$$Elongation[\%] = 0.013171(\text{Pressure}) + 0.000805(\text{Pressure}^2) - 3.153344e^{-6}(\text{Pressure}^3)$$

Equation 17. BFlx Loading Station (31mm): Pressure to %Elongation

$$\text{Pressure}[kPa] = 5.494505(\%Elongation)$$

Equation 18. BFlx Cell Compression: %Elongation to Pressure

$$Elongation[\%] = 0.182(\text{Pressure})$$

Equation 19. BFlx Loading Station (28mm): Pressure to %Elongation

$$\text{Pressure}[kPa] = 0.222919(\%Elongation) + 0.0111226(\%Elongation^2) + 0.000860932(\%Elongation^3) - 1.62523e^{-5}(\%Elongation^4)$$

Equation 20. FlexI Plate: %Elongation to Pressure

$$Elongation[\%] = 3.81161655(\text{Pressure}) - 0.434540391(\text{Pressure}^2) + 0.0325233436(\text{Pressure}^3) - 0.00118088852(\text{Pressure}^4) + 1.61727412e^{-5}(\text{Pressure}^5)$$

Equation 21. FlexI Plate: Pressure to %Elongation

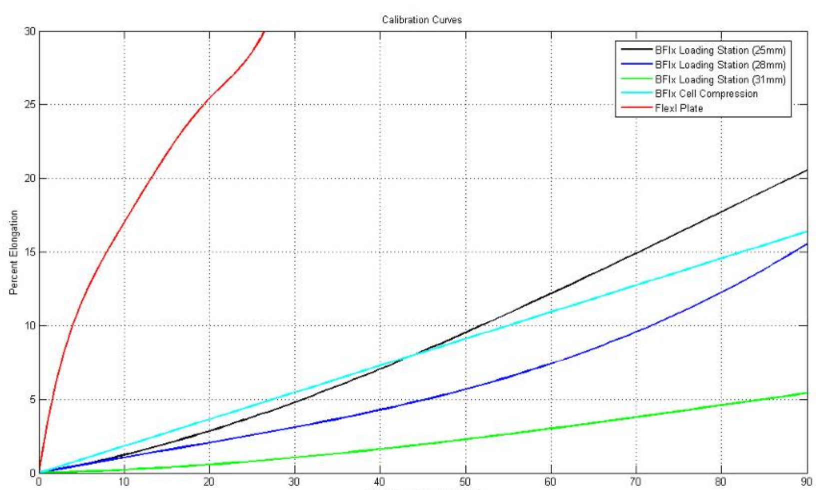


Figure 45. Baseplates Elongation vs. Pressure

Figure 45, displays the relationships between differential pressure, as seen by the transducer, and measured percent elongation. These curves are helpful in loading post size selection. Only the Flex1 Plates are capable of performing higher levels of percent elongation (greater than 20%). One should choose a loading post in the range of their experiment, i.e., with the lowest slope, since lower slopes are less sensitive to error. For example, if the instrument experiences an error of 5 kPa (note: very unrealistic, over prediction amount of error) using the 25mm loading post would result in less than a 1 percent error in percent elongation. With the Flex1 pates this same error will produce around 5 percent error.

5.3 Digital PID control System

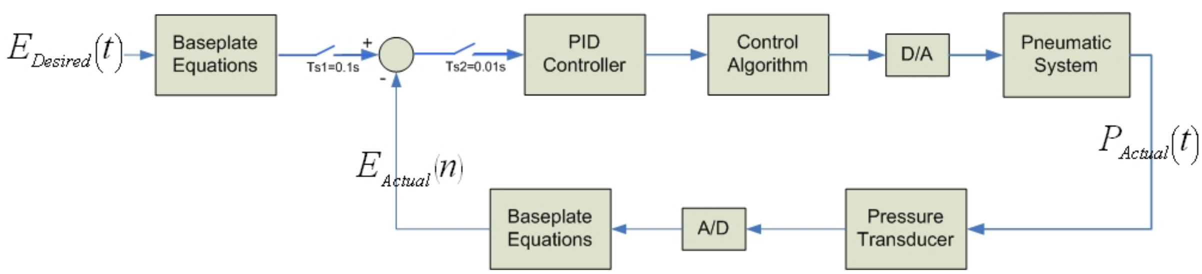


Figure 46. Elongation Control Block Diagram

The closed-loop cell elongation system developed for the pneumatic cultures device was implemented using the PID controllers running on the pc-104+. The block diagram for the control loop is found in Figure 46.

The first step in the PID control loop is to retrieve the desired waveform from the user interface and convert the waveform into a continuous function. For the sinusoidal waveforms the system can used built in functions from the mathematics library. For complex waveforms, such as waveforms generated by the waveform template, a continuous function is generated by connecting the data point's together using line equations. After the continuous equations are developed the algorithm can use the onboard clock to generate a set point for a given time. A new set point is generated every 0.1 seconds (Ts1) using the *getSampleInput()* method from fx5000.c

The set point is compared with the previous cell elongation to determine elongation error. This error could be plug directly into the control algorithm discussed in Section 5.1.4 Proportional On/Off Controller Design. Instead the error is sent to the PID controller in attempt to speed up the system response time. The function *ComputePID()* is uses to adjust the calculated elongation error using the ideal PID equation below where K_c, T_i, T_d are respectably the proportional gain, integration reset, and derivative reset constants and $e(t)$ is current error. Equation 22 is the PID formula in continuous time and Equation 23 is the PID formula in discrete time.

$$u(t) = K_c\left[e(t) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\,\frac{de(t)}{dt}\right]$$

Equation 22. Ideal PID Formula

$$u(n) = K_c\left[e(n) + \frac{1}{T_i}\sum\left(\frac{1}{2}(e(n) + e(n-1))T_{s2}\right) + \frac{e(n) - e(n-1)}{T_{s2}}\right] + u_0$$

Equation 23. Discrete PID Formula

The control algorithm uses the newly adjust error to calculated an analog voltage which is sent to the control valve. The *calcValveVoltage()* method is used to calculate the output voltage and the *pc104.setDACOutput()* is used to output the voltage from the DAQ.

After the signal is sent to the valves, the transducer acquires the system pressure and queues this data for the next iteration. This process is repeated every 0.01 seconds until the experiment completes or stopped by the user.

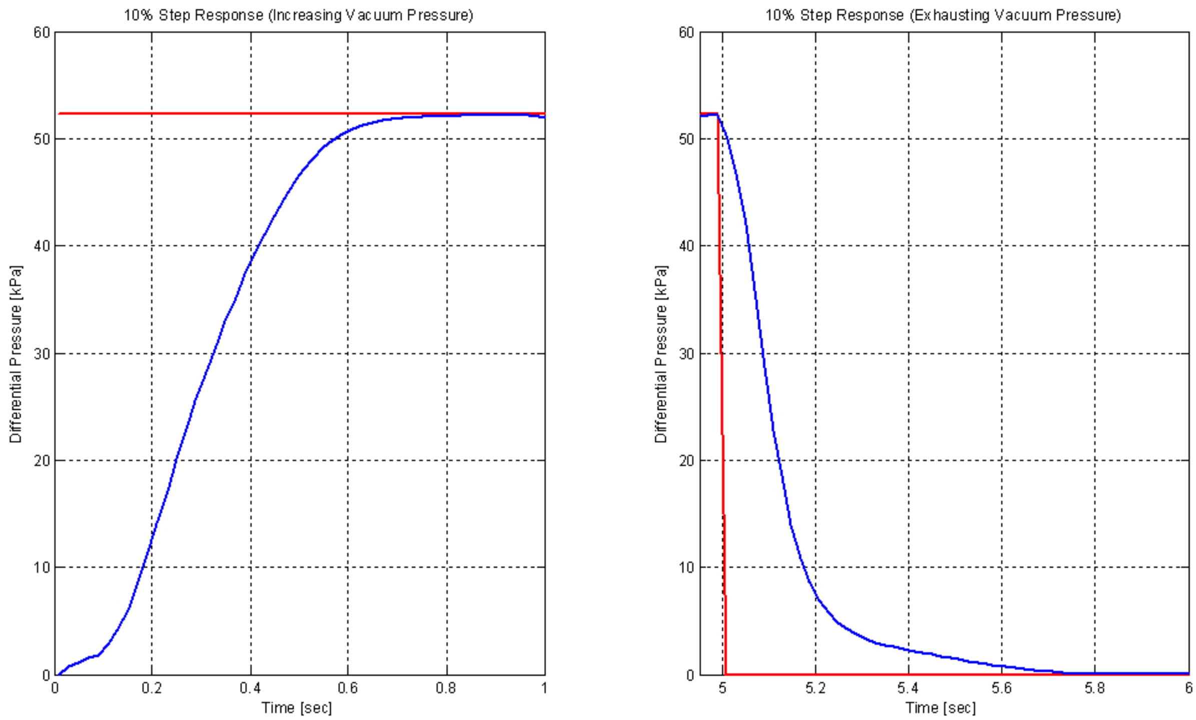


Figure 47. System Responses

Using the digital PID control loop with the proportional gain of 1, integration gain of 100 (where T_i = 0.01) and a no derivative gain the system has a 90 percent rise time of 0.5 seconds for a 10 percent elongation (52.278 kPa using 25mm loading post) step input. The system experience a 90 percent fall time of 0.25 seconds when the vacuum gage pressure change from 52.278 kPa to 0 kPa.

The sampling rates have a major impact on the system performance. The above control loop samples the desired set points every 0.1 seconds and samples the pressure sensor every 0.01 seconds. In general the faster the sensor sampling rate the better the performance. Increasing the sampling rate requires more CPU time. If the sensor sampling rate is too high, the operating system becomes bogged down and the performance decreases. Notice that there are two different sampling rates where the set points sampling rate is 10 times slower than the sensor sampling rate. This is due to the control algorithm. The control algorithm compares the actual pressure with the desired pressure. The amount of error determines the valve position. An error of zero indicates that the desired pressure is equal to the actual pressure in the system and the valves closed off to maintain current pressure. If the set point and the sensor sampling rate is equal (Ts1 = Ts2) the system may experience undesired results as the frequency increases. The pneumatic system experiences a lag, due to tubing length, valve response and other uncontrollable features. At low frequencies the lag isn't as significant because the set point values are not changing very fast. If Ts1 is equal to Ts2 the pressure error can have a value of zero without reaching the desired set point. The top graph in Figure 48 show a situation where the error is zero but maximum and minimal pressure was not attained. If the set points are samples slower than the sensor, the set point signal appears as miniature step inputs in which the control system attempts to reach the desired pressure.

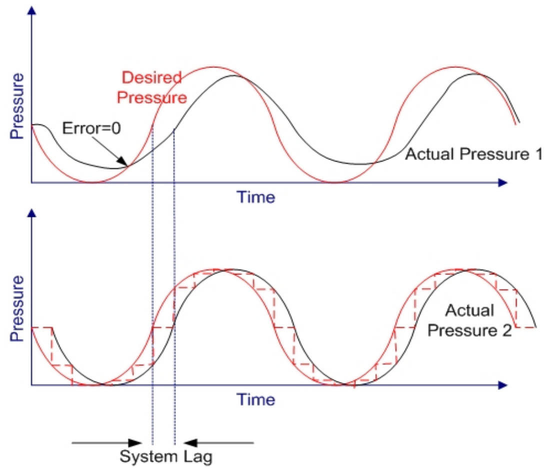


Figure 48. System Lag

Chapter 6. Experiments and Results

To characterize the systems ability to deliver control pressure both in tension and in compression tests were performed. Regimens where created to deliver a sinusoidal waveforms from 0.1 percent elongation up to 10 percent elongation with frequencies variation from 0.10 Hz to 1 Hz.

The first set experiment uses Flexcell's baseplates with 25mm loading post. With the waveforms amplitude set at 5 percent elongation, the frequencies are varied from 0.1 Hz to 1 Hz. A 5 percent elongation using the baseplate with 25mm loading post is equivalent to 30.9172 kPa or 4.4841 psi. Figure 49 display the system error in pressure from the experiments. The experiments reveal that as the frequency increases the amount of error increases. The figures reveal the instantaneous errors cause by the miniature step inputs discuss in Section 5.3 Digital PID control System. Each spike in the figure represents a change in the set point pressure.

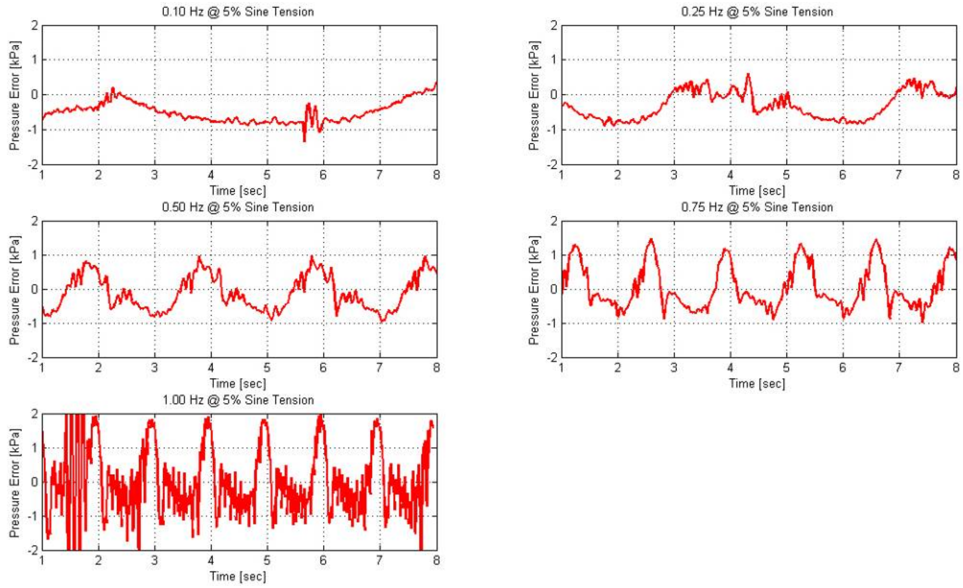


Figure 49. Frequencies Responses for a 5% Elongation waveform using 25 mm loading post

This experiment was repeated using the compression baseplate system. The results are similar to that of the tension setup. The pressure error is related to the frequency. In general the results show that the pressure error is slightly lower using the compression system than the pressure error using the baseplates with 25 mm loading post.

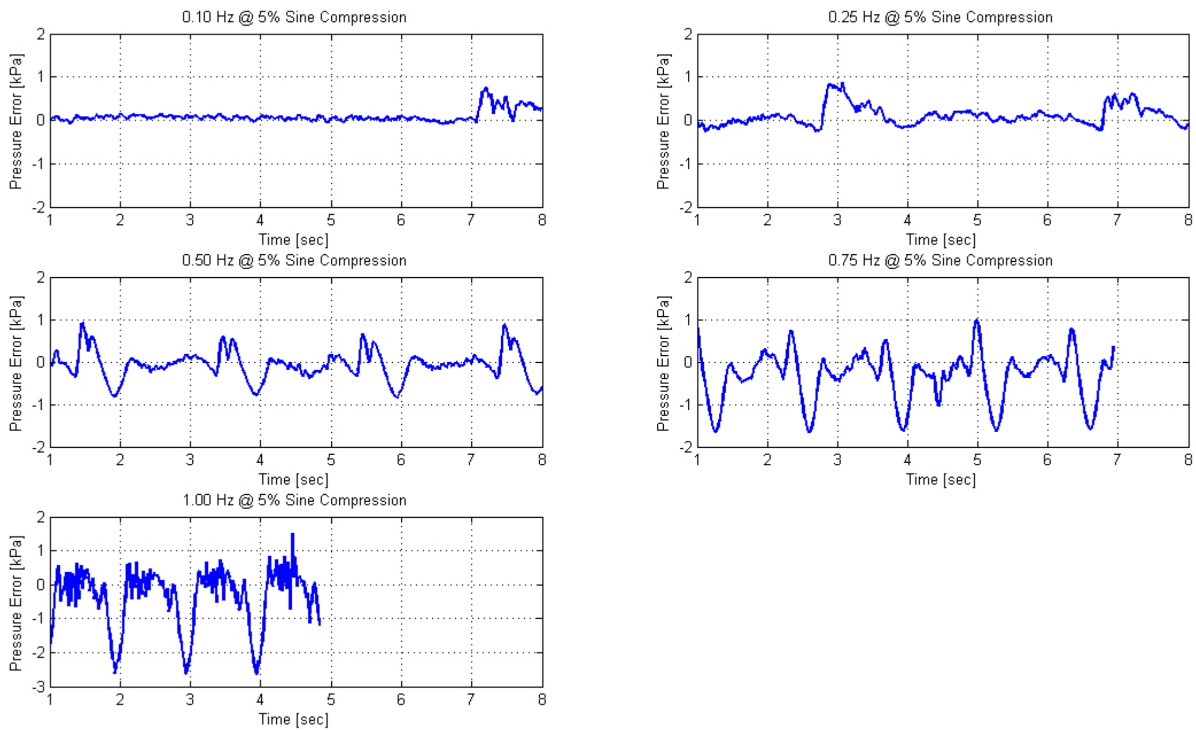


Figure 50. Frequencies Responses for a 5% Elongation waveform using compression system

For the next set of experiments the waveform amplitudes were decreased to 1 percent elongation, to see if the amplitude had an effect on the system error. A 1 percent elongation corresponds to 0.7486 kPa or 0.10857 psi. Figure 51 and Figure 52 are the result after repeating the previous experiments using a 1 percent instead of 5 percent elongation. For the tension system the decrease waveform elongation had little effect on error. With the compression system the system error was reduce by half.

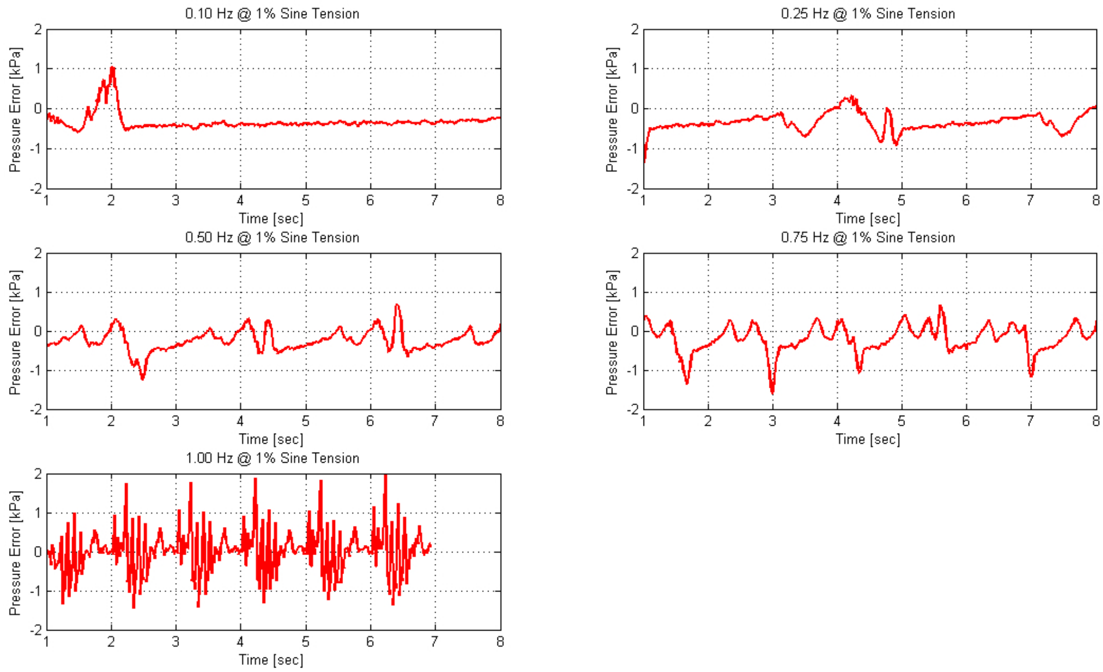


Figure 51. Frequencies Responses for a 1% Elongation waveform using 25 mm loading post

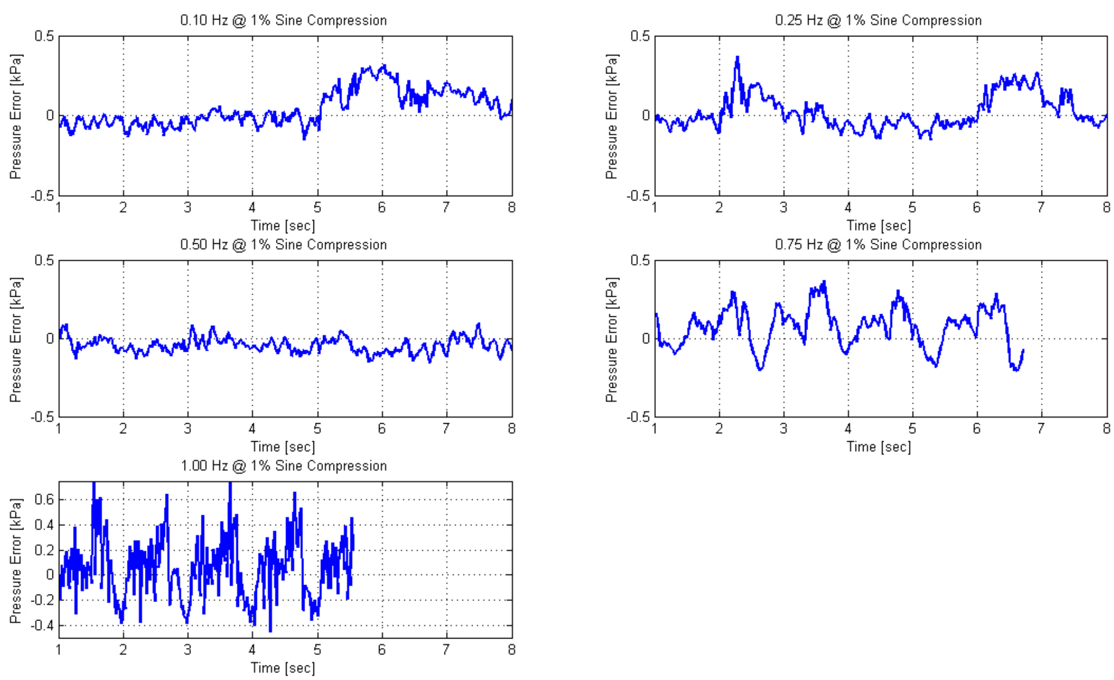


Figure 52. Frequencies Responses for a 1% Elongation waveform using compression system

Since the compression system performs well with 1 percent elongation waveforms. It was desired to see how the system performs with less than 1 percent elongation. Experiments where performed with waveform frequencies varying from 0.1 percent elongation to 1 percent elongation. The results from the experiments are shown in the figure below.

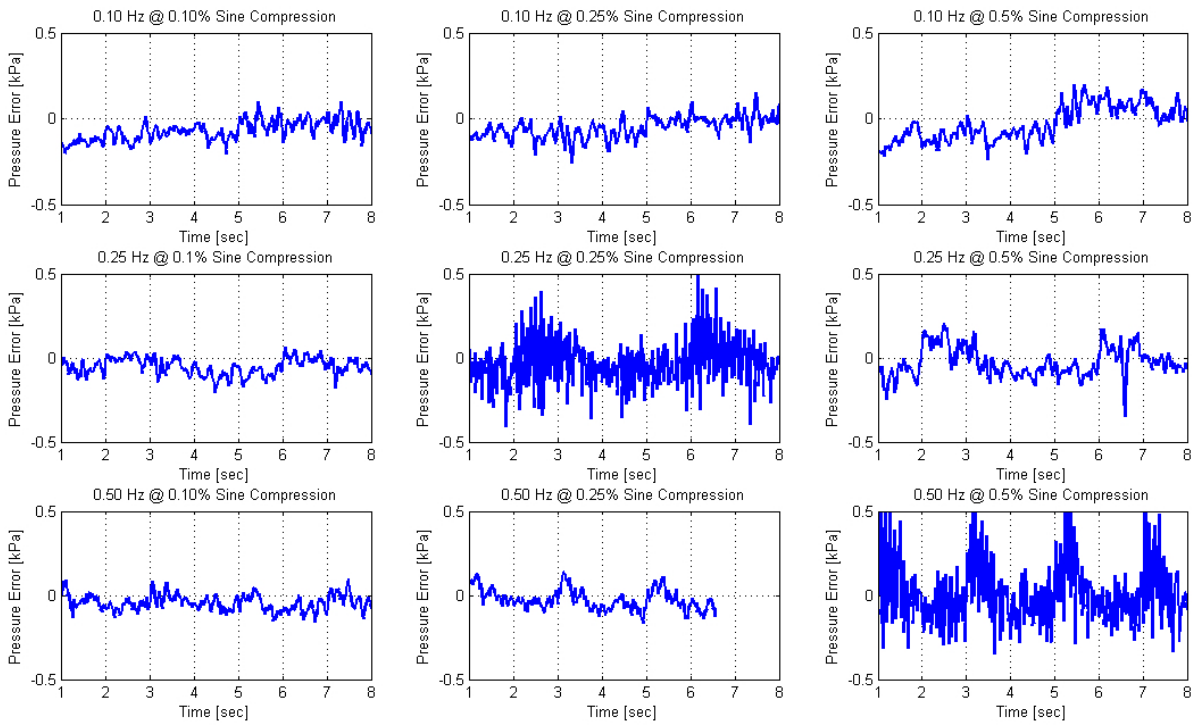


Figure 53. Low amplitude responses using compression system

Chapter 7. Conclusion and Future Research

7.1 Concluding Remarks

The objective of this research was to develop an internet addressable instrument for applying strain to cell in-vitro. The system was required to be adaptable, robust, accurate, and user-friendly to meet the current as well as future demands of Tissue Engineers.

The pneumatic culture straining instrument was developed using high resolution DAQ, proportional valves, embedded systems, and real time priority threads to accuracy control pressure down to 0.1 percent elongation with frequencies varying from 0.06 Hz to 1 Hz. To make the system user-friendly, the system is accessable using a standard web browser. From the web browser the user can perform operations such as create custom waveform, run and monitor experiments.

Although, the system specification and design was driven by a system for performing the same task, a system from the Flexcell International Corporation, this new system was radically different in design. From screw selection to control algorithm was design, developed and tested by the author under the guidance of advisors. This new system is an IT-age system

7.2 Future Research

One concern about the pneumatics system is the use of equations to determine the percent elongation with the respect to the pressure senses by the pressure transducer. These relationships are characterized by the calibration curves from Section

5.2 Elongation and Pressure Relationship. These equations represent the pipe model dynamics Vp2 from Figure 35. These equation where experimentally developed under certain environmental condition that may not be attainable by all users. Another concern is that these equations assume that the cell properties have similar properties of the baseplate membranes. What if the cells are not completely attach to the baseplates? Are the elasticity proprieties of hard tissues, such as bones, the same as soft tissue, such as tendons?

With the addition of high speed cameras and image processing, the author believes that the above problem can be solved. Figure 54 curtsey of Nikon Microscopy, is an image of monkey kidney cells capture with a 12-bit gray scale camera coupled to a microscope. Image processing could be used to determine elongation prosperities as the cells are stressed. This would eliminate the need for calibrations equations.

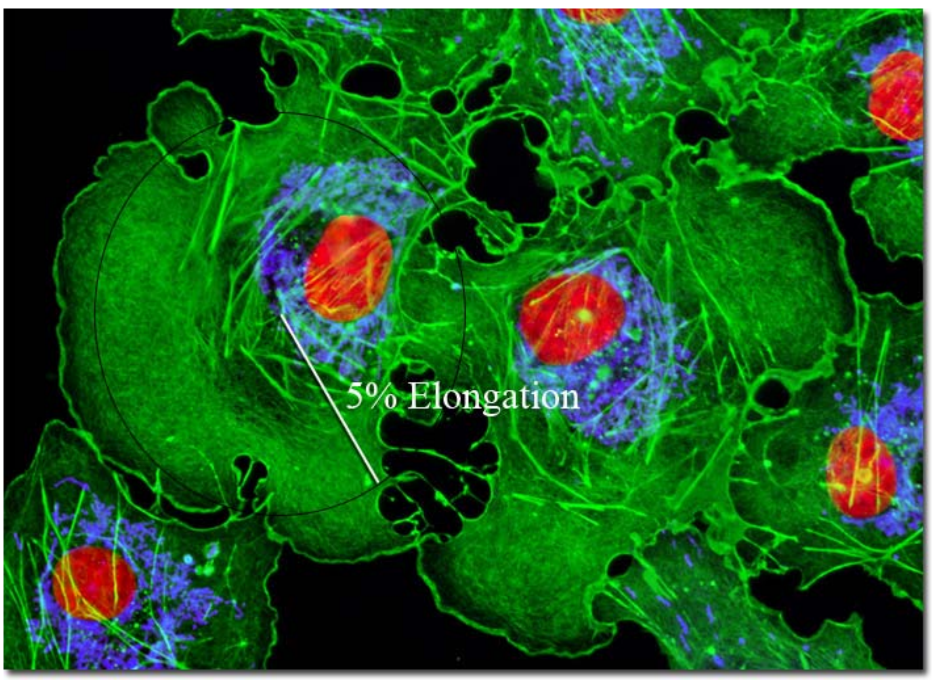


Figure 54. Transformed African Green Monkey Kidney Fibroblast Cells

Another area for future research involves automate tuning the PID gain values. Due to the robustness of the culturing device finding particular values of the PID is hard to attain. Certain PID values might work really well for high elongation experiment, but experience a lot of overshoot for low elongation experiment. Development of an algorithm that could auto tuning PID controller would allow the device to perform to optimally for the full range of experiments and when part start to wear. One proposed idea is to use machine learning algorithms to developed PID values. During the experiment the algorithm would monitor the performance and adjust levels accordingly for future experiments. This smart PID system could be applied on a verity of control devices.

Chapter 8. References

1. Banes, A.J., et al., *A new vacuum-operated stress-providing instrument that applies static or variable duration cyclic tension or compression to cells in vitro.* J Cell Sci, 1985. **75**(1): p. 35-42.
2. Brown, T.D., *Techniques for mechanical stimulation of cells in vitro: a review.* Journal of Biomechanics, 2000. **33**(1): p. 3-14.
3. Cehreli, M., S. Sahin, and K. Akca, *Role of mechanical environment and implant design on bone tissue differentiation: current knowledge and future contexts.* Journal of Dentistry, 2004. **32**(2): p. 123-132.
4. Derwin, K.A., et al., *A new optical system for the determination of deformations and strains: Calibration characteristics and experimental results.* Journal of Biomechanics, 1994. **27**(10): p. 1277-1285.
5. Devkota, A.C. and P.S. Weinhold, *A tissue explant system for assessing tendon overuse injury.* Medical Engineering & Physics, 2005. **27**(9): p. 803-808.
6. H. Hu, F.S., *Mechanically Activated Currents in Chick Heart Cells.* Membrane Biology, 1996: p. 205-216.
7. J E Woodell, M.L., E M Langan III, R H Hilderman, *In vitro strain-induced endothelial cell dysfunction determined by DNA synthesis.* Engineering in Medicine, 2003. **207**(H): p. 13-20.
8. Pascal Bigras, K.K. *Nonlinear Observer for Pneumatic System With Non Negligle Connection Port Restriction*. in *American Control Conference*. 2002. Anchorage, AK.
9. Pascal Bigras, K.K., Tony Wong. *Modified feedback linearization controller for Pneumatic System With Non Negligible Connection Port Restriction*. in *Systems, Man, and Cybernetics* 2002.
10. Shimko, D., et al., *A Device for Long Term, In Vitro Loading of Three-Dimensional Natural and Engineered Tissues.* Annals of Biomedical Engineering, 2003. **31**(11): p. 1347-1356.

Chapter 9. Appendices

Appendix 1. User Guide

A.1.1 Hardware Setup

The rear panel, shown in Figure 60, contains the following connectors: AC Power, Programmable External AC Power Out, Pneumatics and Network. The unit requires an AC voltage from 85 to 265 with frequencies between 47 to 63 Hz to operate correctly. The power plugs can provide external power to additional equipment up to 125 VAC at 15 amps. The Flex Out port delivers pressures to the base plates using 3/8” (9.5 mm) tubing. The Flex IN port is used for pressure feedback from the base plates. This connector uses ¼” (6.4mm) tubing. The System connector is used to connect the vacuum or compressed air line to the system using 3/8” (9.5 mm) tubing. The Vent port should be left open. The System is connected to the network using a CAT5e network cable.

A.1.2 Network Configuration

The system is initially configured to retrieve an IP address automatically using the DHCP protocol. The Network Config panel, located on the touch screen display, provides an interface for viewing and/or changing the network settings.

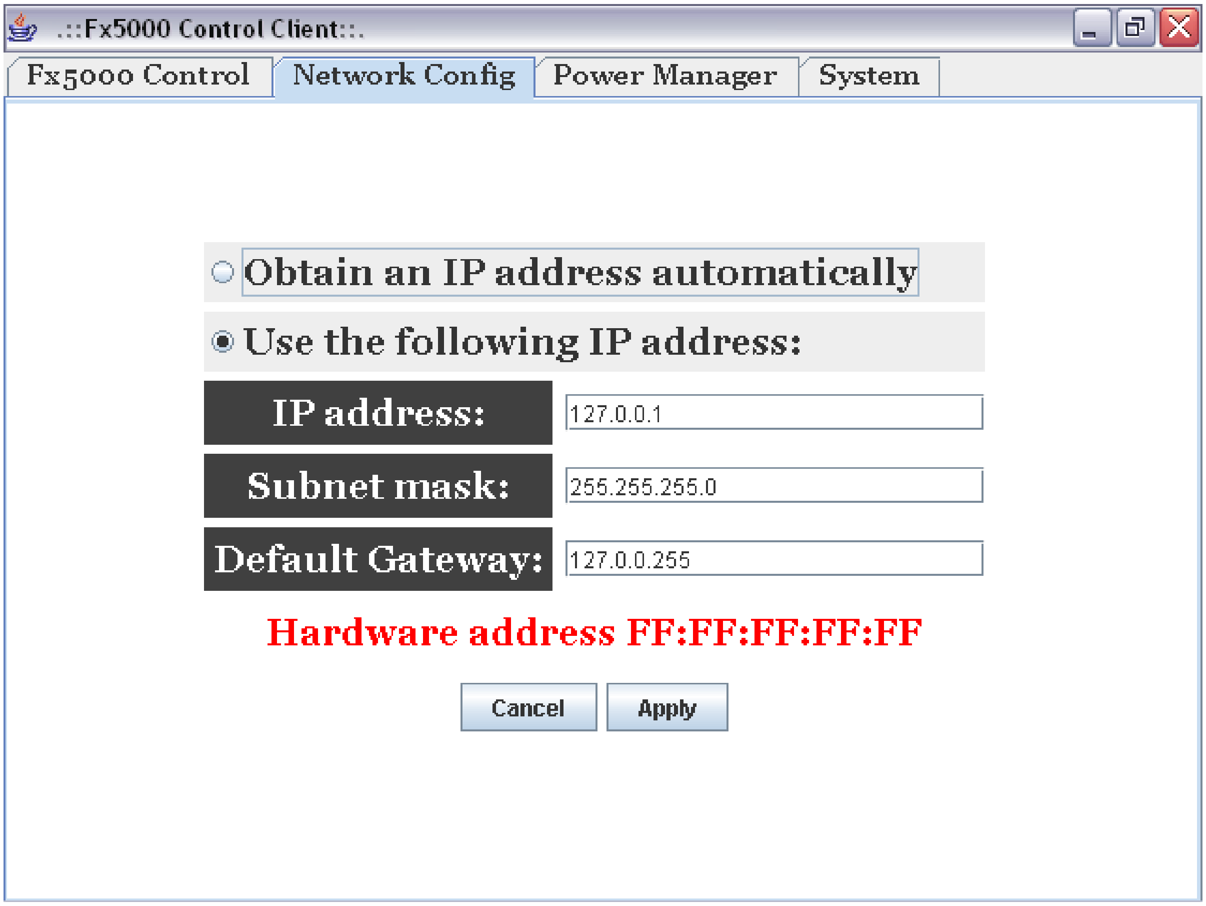


Figure 55. Network Configuration

It is desirable to establish a static network connection. In order to gain access to the unit from wide-area network (WAN), the system needs admittance to following ports. Please contact your network administrator for assistance.

Table 11. Network Ports

Port	Protocol	Description
80 and/or 8080	TCP/IP	Http Server
3306	TCP/IP	SQL Server
9001	TCP/IP	Fx5000 Controller

A.1.3 Procedures

Power up System

1. Turn on the unit using the rocker switch located on the back off the unit.

Logging In

1. Using a web browser, go to the unit website site. For example if the unit has the IP address 192.168.0.1 the address to the main web site is http://192.168.0.1

Note: The above example use the system local IP address, contact your network administrator for an extern (WAN) IP address.

2. Enter username and password. The default username and password is fx5000.

Creating/ Editing Regimen

1. Click on the Regimens tabbed, located at the top of the screen to display the following screen.

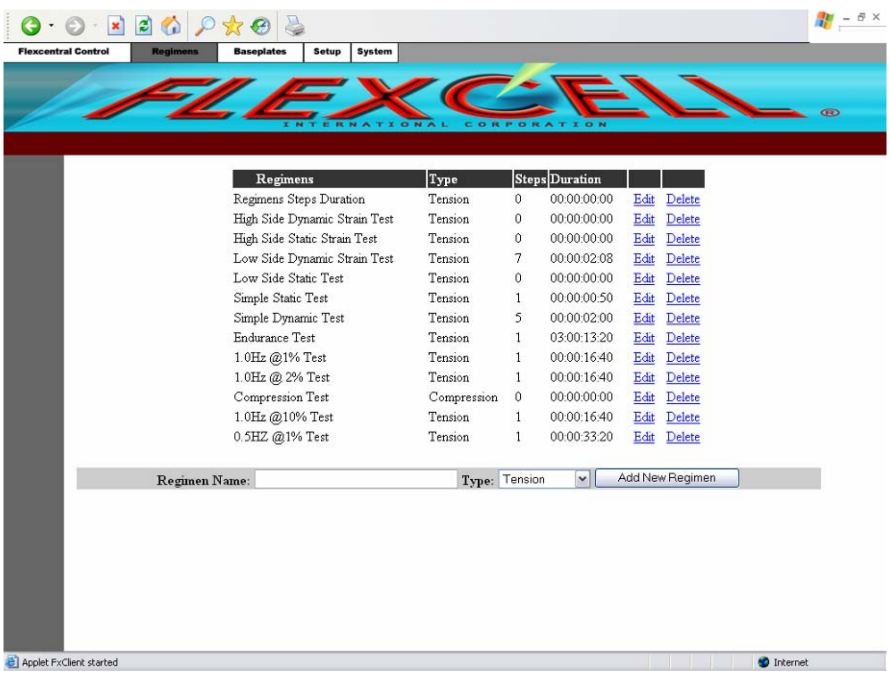


Figure 56. Creating a new regimen

2. Enter a name for the regimen and click the “Add New Regimen” Button.

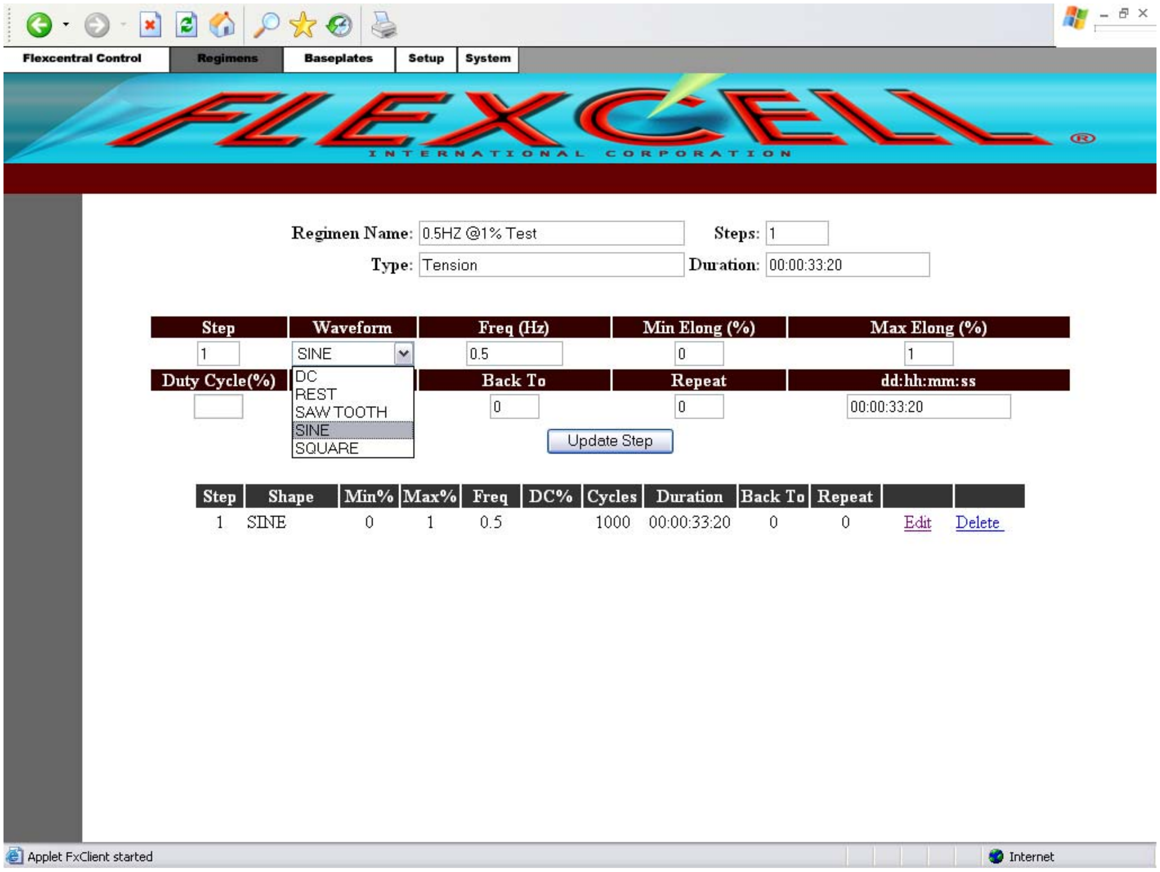


Figure 57. Configuring regimen

3. Enter in the desired parameters to create the regimen of your choice.

Running Regimen

1. Click on the “Flexcentral Control” tabbed, located at the top of the screen to bring up the control applet.

Note: The applet requires java run time environment version 1.5 or greater. If your system does not have the require software the browser will attempt to install the software.

2. Select the desired regimen and baseplate and pressed "Assign Regiment"
3. Press the Start Button to begin the experiment.

Appendix 2. Pneumatic Culture System Software

A.2.1 Modified Linux Startup Routine

Bios, Boot Sector Grub, Linux Kernel, then run time scripts

1. Runtime Level 3 Scripts (/etc/rc.d/rc3.d)

Script	Note
S01lcdoff	Turn LCD Screen Power On
S01pwroff	Turn off External Power
S64mysql	Start SQL Server
S85httpd	Start Web Server
S90crond	Schedule Automatic Task (External Power Timer)
S98dscud	Startup DAQ Services
S99clearlog	Remove old Log file
S99sched	Schedule Automatic Task (External Power Timer)
S99touchscreen	Load TouchScreen Drivers

2. /etc/inittab (Login Script)

```
#
# inittab       This file describes how the INIT process should set up
#               the system in a certain run-level.
#
# Author:       Miquel van Smoorenburg, <miquels@drinkel.nl.mugnet.org>
#               Modified for RHS Linux by Marc Ewing and Donnie Barnes
#

# Default runlevel. The runlevels used by RHS are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3, if you do not have networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
```

```
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now

# When our UPS tells us power has failed, assume we have a few minutes
# of power left.  Schedule a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"

# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"


# Run gettys in standard runlevels
1:2345:respawn:/sbin/agetty -n -l /usr/local/sbin/autologin 38400 tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6

# Run xdm in runlevel 5
x:5:respawn:/etc/X11/prefdm –nodaemon
```

autologin.c

```
//:autologin.c
//Frederick Livingston (fjliving@ncsu.edu) 4-6-2005

int main(){
   execlp("login", "login", "-f", "root", 0);
}
```

3. Start Fx5000Server Process (/root/.bash_profile)

```
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin
BASH_ENV=$HOME/.bashrc
USERNAME="root"

export USERNAME BASH_ENV PATH
```

```
export
CLASSPATH=$CLASSPATH:/usr/local/javamail/mail.jar:/usr/local/javamail/activation.jar/:www/Fx5000.jar:.


ps -a | grep fvwm2
if [ $? == 0 ]; then
        echo "Fx5000 Software Already Started"
else
        echo "Starting Fx5000 Software"
        /usr/local/sbin/fx5000Server 9001&
        startx
fi
```

4. User Interface Startup (root/.fwm2rc)

```
DeskTopSize 1x1

DestroyFunc InitFunction
AddToFunc   InitFunction
+ I Exec exec xhost +localhost
+ I Exec exec xscreensaver&
+ I Exec exec java -cp /usr/local/javamail/mail.jar:/usr/local/javamail/activation.jar:/www/Fx5000.jar Fx5000
fullscreen&
+ I Exec exec /usr/local/sbin/signApplet WaveTemplate.jar
+ I Exec exec /usr/local/sbin/pwrScreen on
```

A.2.2 Important Linux Executables

Directory: /usr/local/sbin

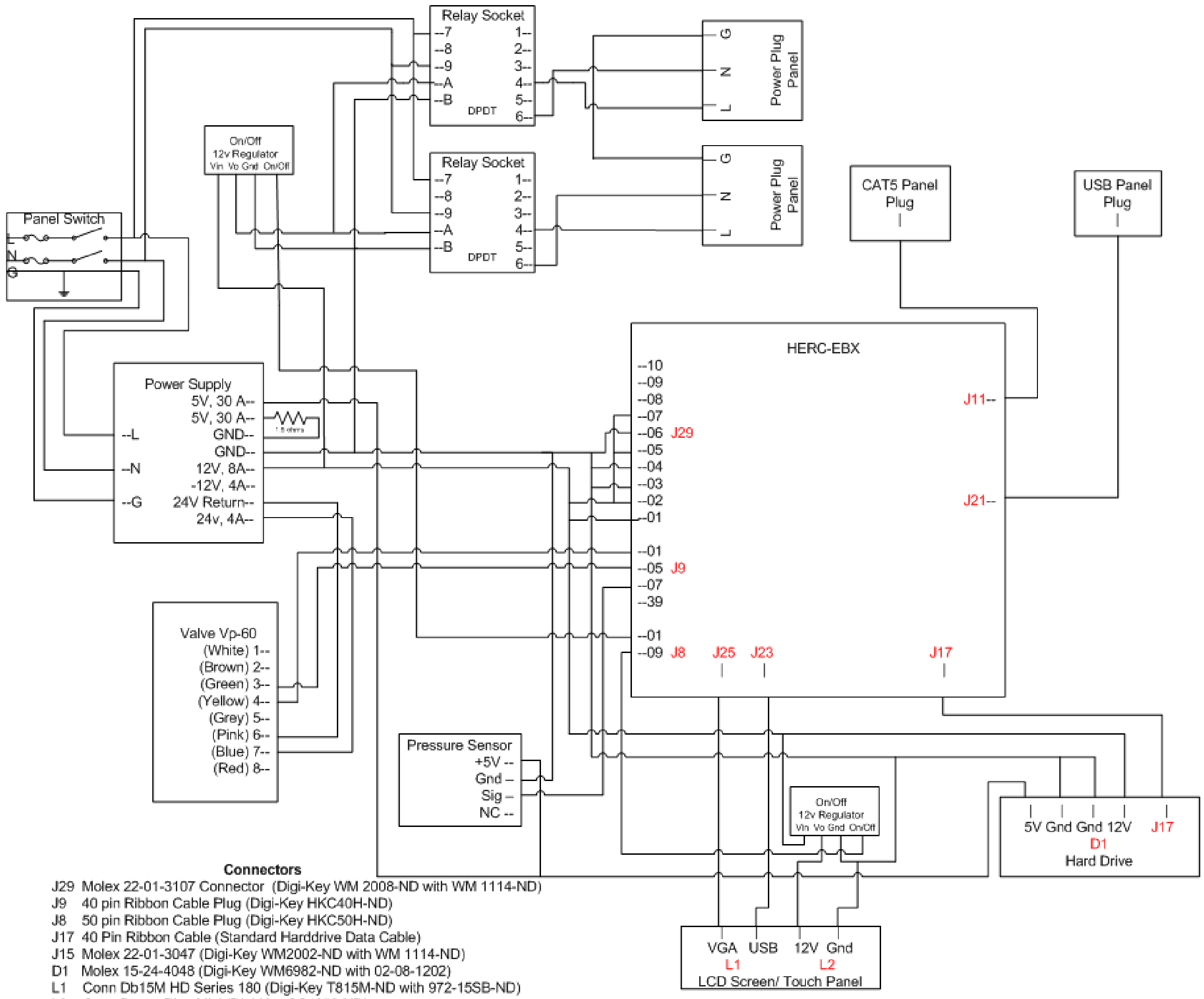
Script/Executable	Note
autologin	Automatically log user in as root during startup
clearLog	Clears system logs to free disk space
extPowerOff	Switches off the relays to disable voltage to the external power plugs
fx5000Server	Control System Excitable Note: Must run as root in order to use real time scheduler
fxEmailier	Sends out emails
fxQuery	Use to query the SQL database during system upgrade
fxUpgrage	Use to update the system software
genWaveform	Obsolete: Use to output waveforms to analog out
lcdOff	Switches off the LCD display
pumpOff	Same as setPower ON
pumpOn	Same as setPower OFF
pwrScreen	Turns on/off LCD Screen
readTransducer	Polls the pressure transducer and prints of transducer voltage and pressure, very useful for debugging
scheduler	Use to communicate with crond to schedule time events
setNetwork	Use to configure the system network connection
setPower	Turns on/off the external power
setTime	Use to configure the system time
setValve1	Use to manually apply voltage to Valve 1 (very useful for debugging)
setValve2	Use to manually apply voltage to the second Valve
signApplet	Use to sign apply for security reasons
sshd	Use to start openssh services

Appendix 3. Part List

PARTS LIST 05-02-2006

✓	PART	MANUFACTOR	VENDER	PART #	STOCK	QUANT.	CPU	COST
	Relay	Magnecraft Struthers-Dunn	www.mouser.com	528-38815-6	Y	2	$9.42	$20.48
	Relay-Socket	Magnecraft Struthers-Dunn	www.mouser.com	528-4631	Y	1	$3.14	$3.14
	Relay-DIM clip	Magnecraft Struthers-Dunn	www.mouser.com	528-16-1278	Y	1	$0.56	$0.56
	15 amp 5x20mm Fuse	Cooper/Bussmann	www.mouser.com	504-GMA-15	Y	2	$0.24	$0.48
	Voltage Regulator	Sharp Microelectronics	www.mouser.com	852-PQ12RD21J00H	Y	2	$2.24	$4.48
	35 watt 1.5 Ohm Resistor	Ohmite	www.digikey.com	TCH35P1R50JE-ND	Y	1	$5.90	$5.90
	Heat Sink	Aavid Thermalloy	www.digikey.com	HS110-220-ND	Y	1	$1.80	$1.80
	Control Valve w/ 1/4" NPT and 0-10v	Norgren VP60	www.norgren.com	VP6010LK161M0000	Y	1		
	Power Supply	Sola/HeviDuty	www.solaheviduty.com	GLQ-05-200		1	$299	$299
	LCD/TouchScreen Screen	Apollo Displays	www.apollodisplays.com	KI-1A-103B	Y	1	$603.22	$603.22
	Pressure Transducer	Omega	www.omega.com	PX139-030A4V	Y	1	$85	$85.00
	Power Switch	Qualtek Electronics	www.qualtekusa.com	764-00/002		1		
	Power Outlet	Qualtek Electronics	www.qualtekusa.com	738W-X2/03		2		
	CAT5 Panel Plug (RJ45 F/F)	Diamond Systems	www.diamonsystems.com	698002	Y	1	$10.00	$10.00
	Dual-port USB cable	Diamond Systems	www.diamonsystems.com	698012	Y	1	$8.00	$8.00
	Power Cable	Qualtek Electronics	www.qualtekusa.com	212004-01		1		
	Motherboard	Diamond Systems	www.diamonsystems.com	HRC750-5A256	Y	1	$995.00	$995
	Hard drive-40Gb+		http://www.outpost.com/					
	40 Pin Ribbon Cable (J17 HardDrive)		http://www.outpost.com/					
	Custom Box	U.S.A Dutch, INC	usadtc@netpath.net		N	1		
			Wiring					
	Molex 22-01-3107 (J29 Power Housing)	Digi-Key	www.digikey.com	WM2008-ND	Y	1	$0.81	$0.81
	Molex 08-50-0160 (J29 Power Contact)	Digi-Key	www.digikey.com	WM1114-ND	Y	11	$0.74	$8.14
	2x25 IDC 0.1" pitch plug (J8)	Digi-Key	www.digikey.com	HKC50H-ND	Y	1	$1.78	$1.78
	2x20 IDC 0.1" pitch plug (J9)	Digi-Key	www.digikey.com	HKC40H-ND	Y	1	$1.39	$1.39
	Molex 22-01-3047 (J15 Aux Power)	Digi-Key	www.digikey.com	WM2002-ND	Y	1	$0.39	$0.39
	Molex 15-24-4048 (D1 HardDrive PH)	Digi-Key	www.digikey.com	WM6982-ND	Y	1	$0.87	$0.87
	Molex 02-08-1202 (D1 Power Contact)	Digi-Key	www.digikey.com	WM6983-ND	Y	4	$0.29	$1.16
	10 Pin Ribbon Plug (J25 LCD VGA)	Digi-Key	www.digikey.com	ASA10K-ND	Y	1	$2.66	$2.66
	DB15m Series 180	Digi-Key	www.digikey.com	T815M-ND	Y	1	$2.13	$2.13
	Power Plug (L2 LCD Power)	Digi-Key	www.digikey.com	SC1052-ND	Y	1	$3.08	$3.08
			Assembly					
	Screw #3-48x1/4 (Pack of 100)	McMaster-Carr	www.mcmaster.com	91773A092	Y	10	$6.84	$6.84
	Screw #6-32x5/16 (Pack of 100)	McMaster-Carr	www.mcmaster.com	91773A145	Y	14	$3.01	$3.01
	Screw #6-32x5/8 (Pack of 100)	McMaster-Carr	www.mcmaster.com	91773A150	Y	2	$3.39	$3.39
	Screw #8-32x1/4 (Pack of 100)	McMaster-Carr	www.mcmaster.com	91773A190	Y	4	$3.65	$3.65
	Screw #8-32x2-1/4 (Pack of 100)	McMaster-Carr	www.mcmaster.com	91772A206	Y	2	$11.77	$11.77
	Flang Nut #3-48 (Pack of 100)	McMaster-Carr	www.mcmaster.com	91841A004	Y	4	$3.04	$3.04
	Flang Nut #6-32 (Pack of 50)	McMaster-Carr	www.mcmaster.com	93776A370	Y	6	$4.63	$4.63
	Nut #8-32 (Pack of 50)	McMaster-Carr	www.mcmaster.com	90205A309	Y	2	$4.41	$4.41
	Thread-lock (1-oz)	McMaster-Carr	www.mcmaster.com	75145A69	Y	1	$16.84	$16.84
	35mm DIN Rail (1 meter)	McMaster-Carr	www.mcmaster.com	8961K15	Y	1	$4.33	$4.33
	Adhesive-Backed Bumpers (Pack of 64)	McMaster-Carr	www.mcmaster.com	9723K28	Y	1	$8.18	$8.18
	***Note: 1 UNIT OF THESE ITEMS CAN BUILD SEVERAL FX5000 UNITS**							
			Fittings					
	3/8" OD Tubing	Flexcell Int	www.flexcellint.com					
	1/4"" OD Tubing	Flexcell Int	www.flexcellint.com					
	3/8" OD Brass Comp. Tube Fitting	Flexcell Int	www.flexcellint.com		Y	3		
	1/4" OD Brass Comp. Tube Fitting	Flexcell Int	www.flexcellint.com		Y	1		
	1/4NPT 3/8 OD"	McMaster-Carr	www.mcmaster.com	51875K66	Y	1	$2.36	$2.36
	1/4NPT 3/8 OD" 90 Elbow	McMaster-Carr	www.mcmaster.com	51875k86	Y	2	$3.56	$7.12
	Total							$2,139.04

Appendix 4. Wiring Diagram



Connectors

1. J29 Molex 22-01-3107 Connector (Digi-Key WM 2008-ND with WM 1114-ND)
2. J9 40 pin Ribbon Cable Plug (Digi-Key HKC40H-ND)
3. J8 50 pin Ribbon Cable Plug (Digi-Key HKC50H-ND)
3. J17 40 Pin Ribbon Cable (Standard Harddrive Data Cable)
4. J15 Molex 22-01-3047 (Digi-Key WM2002-ND with WM 1114-ND)
5. D1 Molex 15-24-4048 (Digi-Key WM6982-ND with 02-08-1202)
6. L1 Conn Db15M HD Series 180 (Digi-Key T815M-ND with 972-15SB-ND)
7. L2 Conn Power Plug Mini (Digi-Key SC1052-ND)

Appendix 5. CAD Drawings

A.5.1 Exterior Views

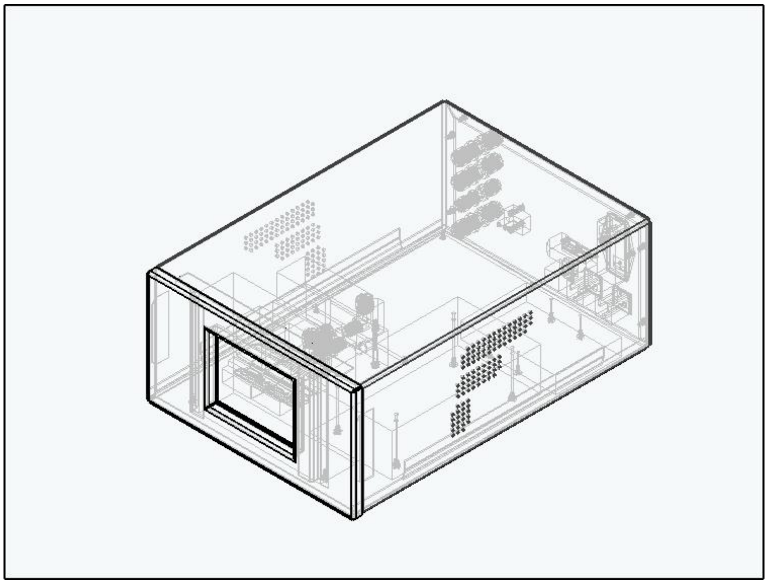


Figure 58. Pneumatic Culture System Isometric View

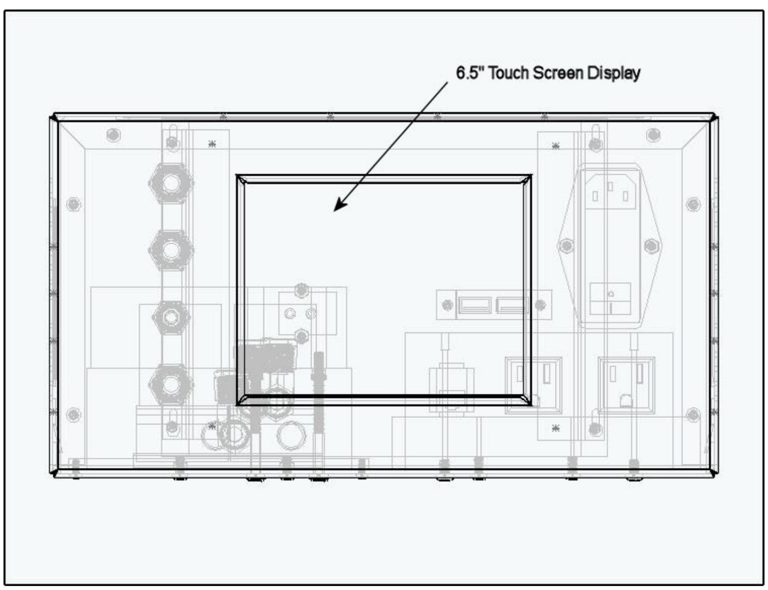


Figure 59. Pneumatic Culture System Front Panel

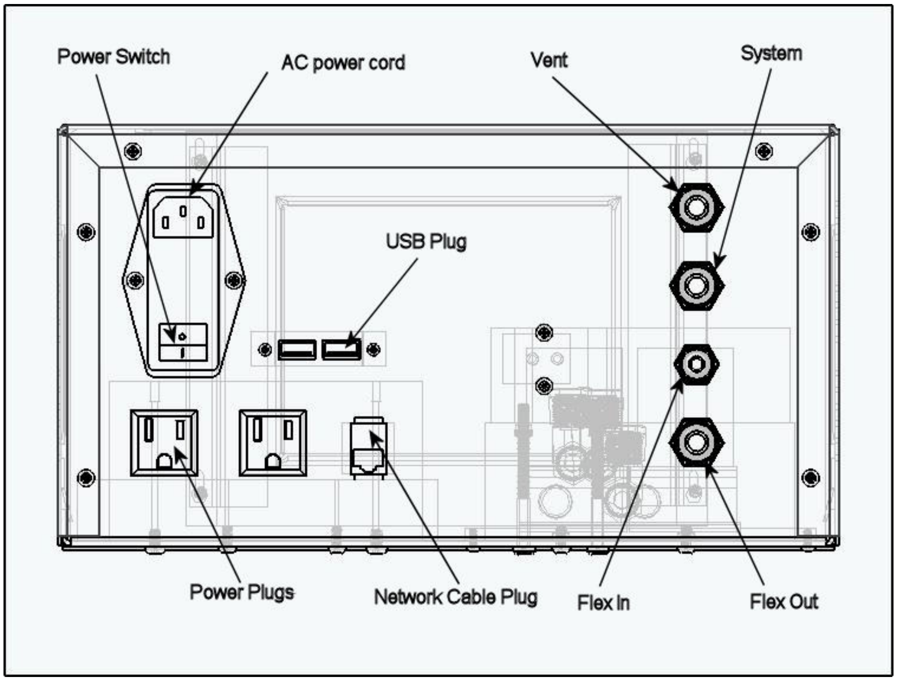


Figure 60. Pneumatic Culture System Rear Panel

A.5.2 Physical Dimensions

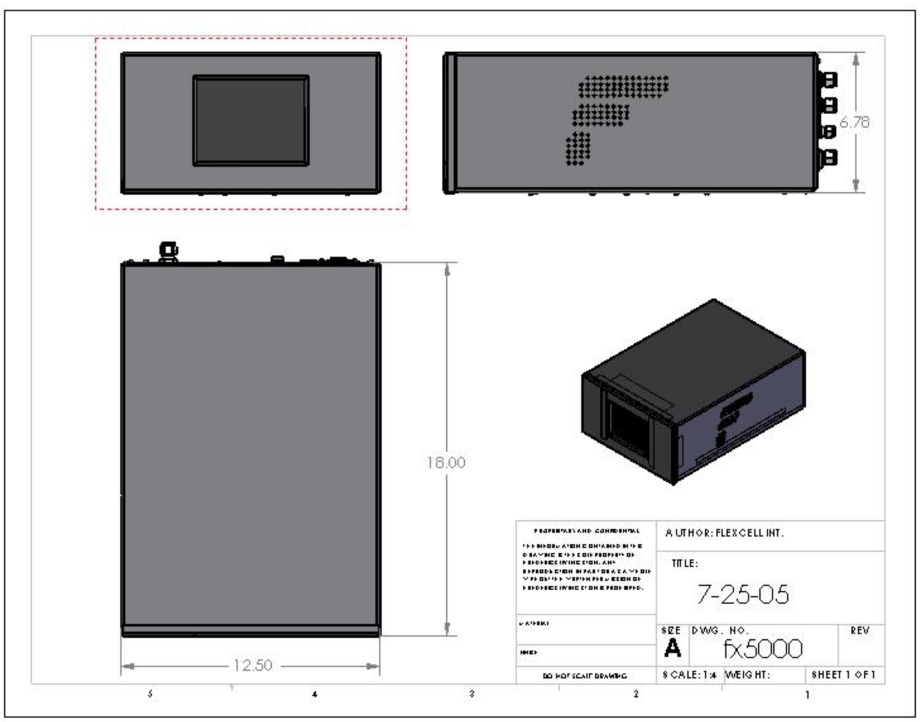
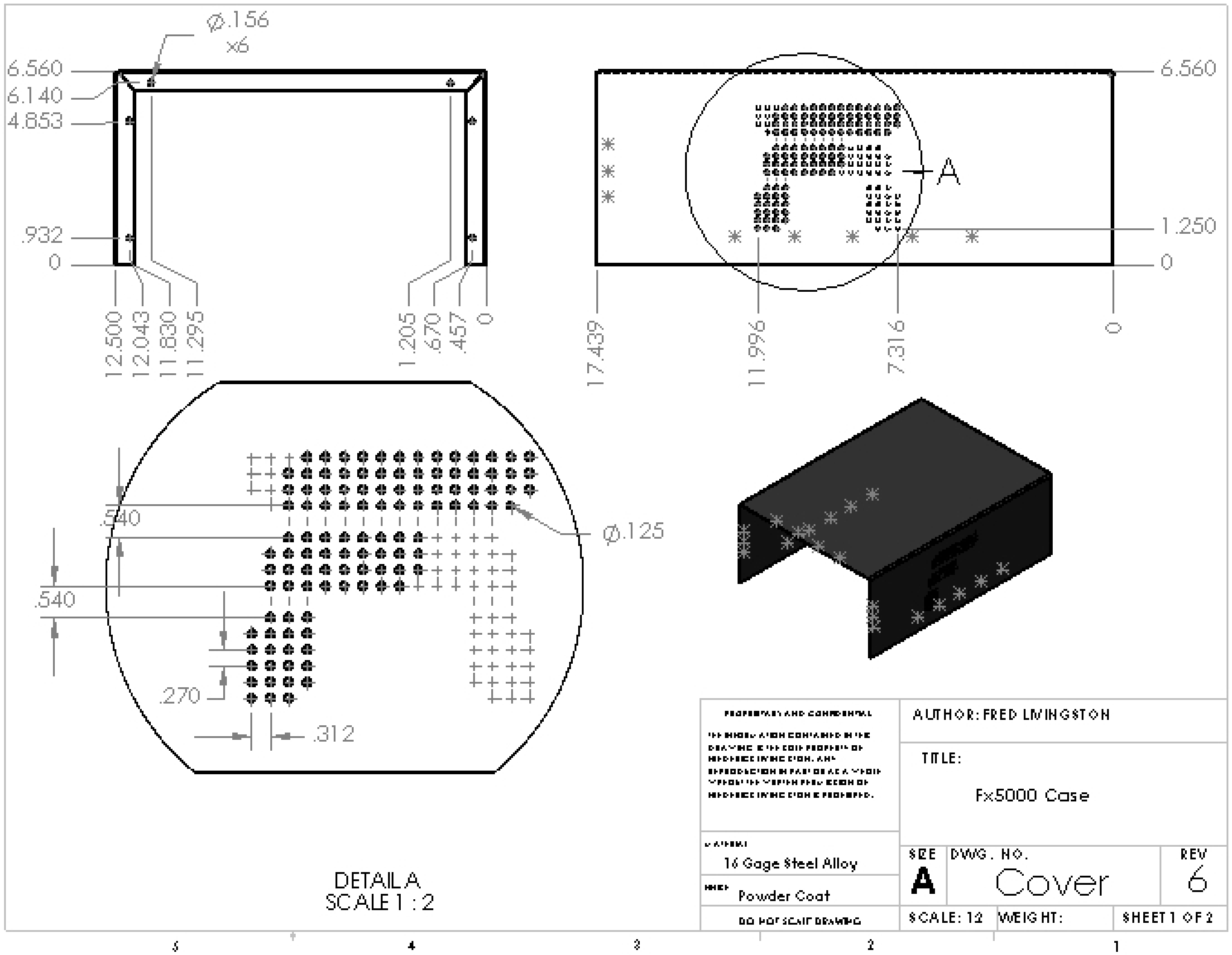
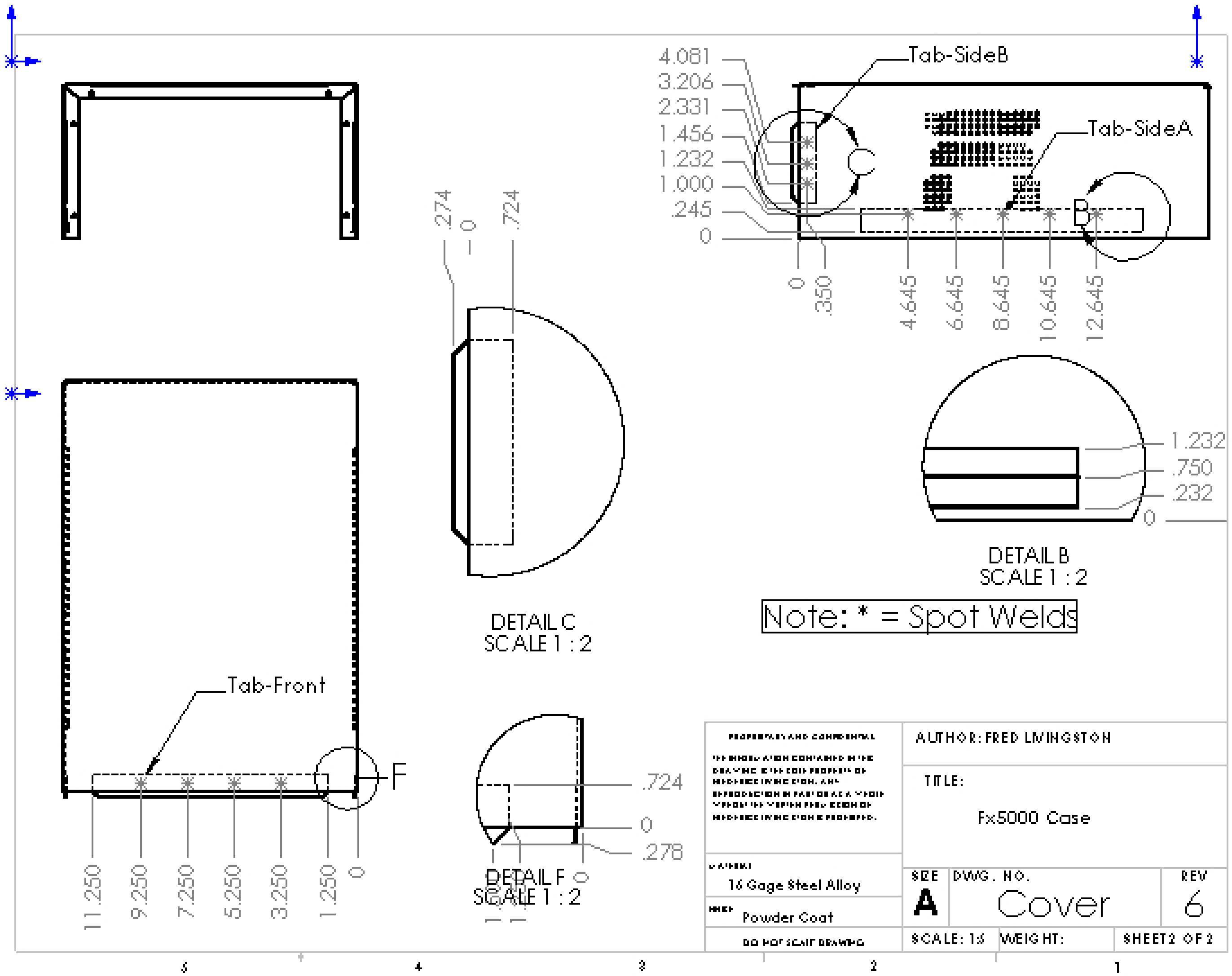


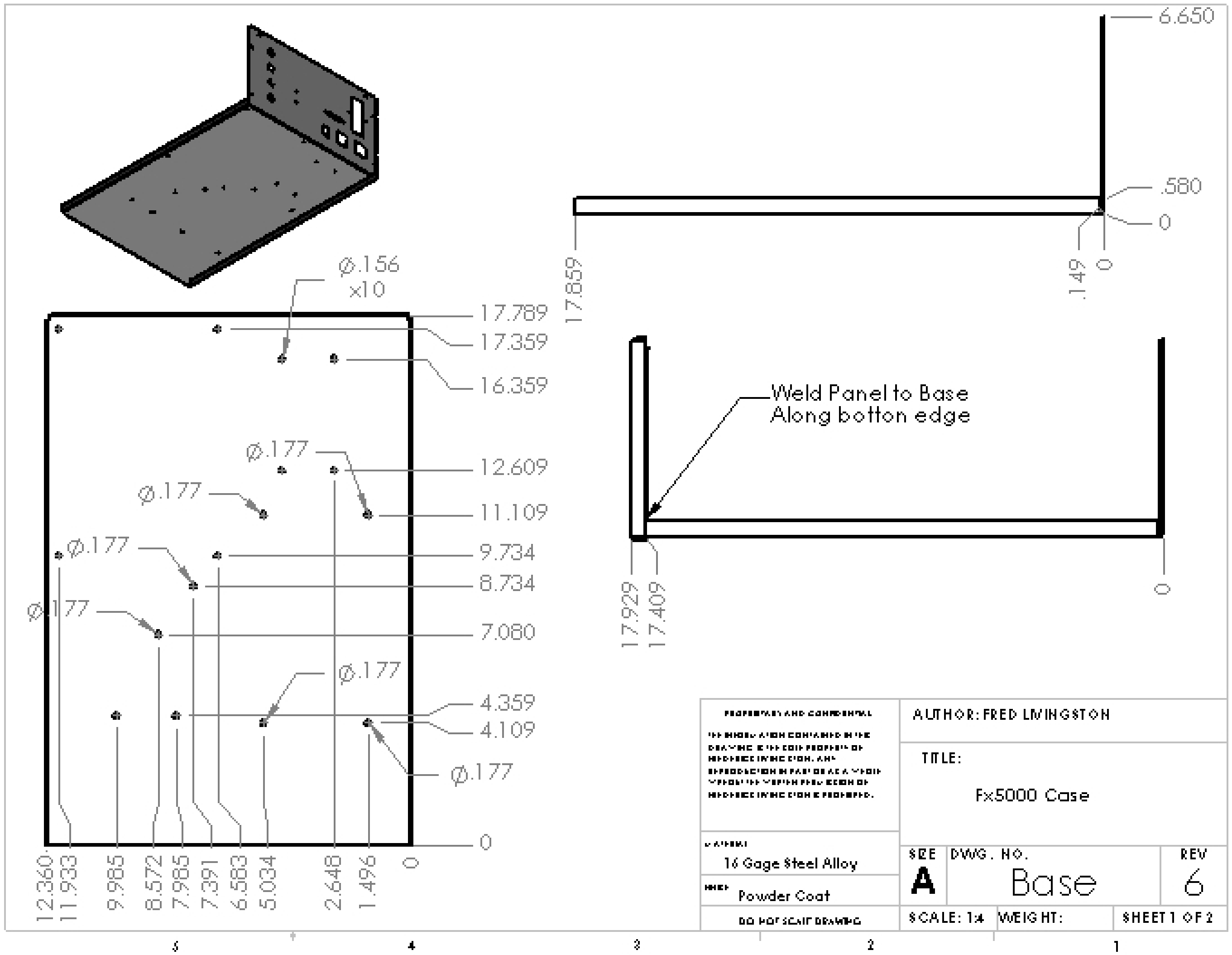
Figure 61. Pneumatic Culture System Dimensions

A.5.3 Cover

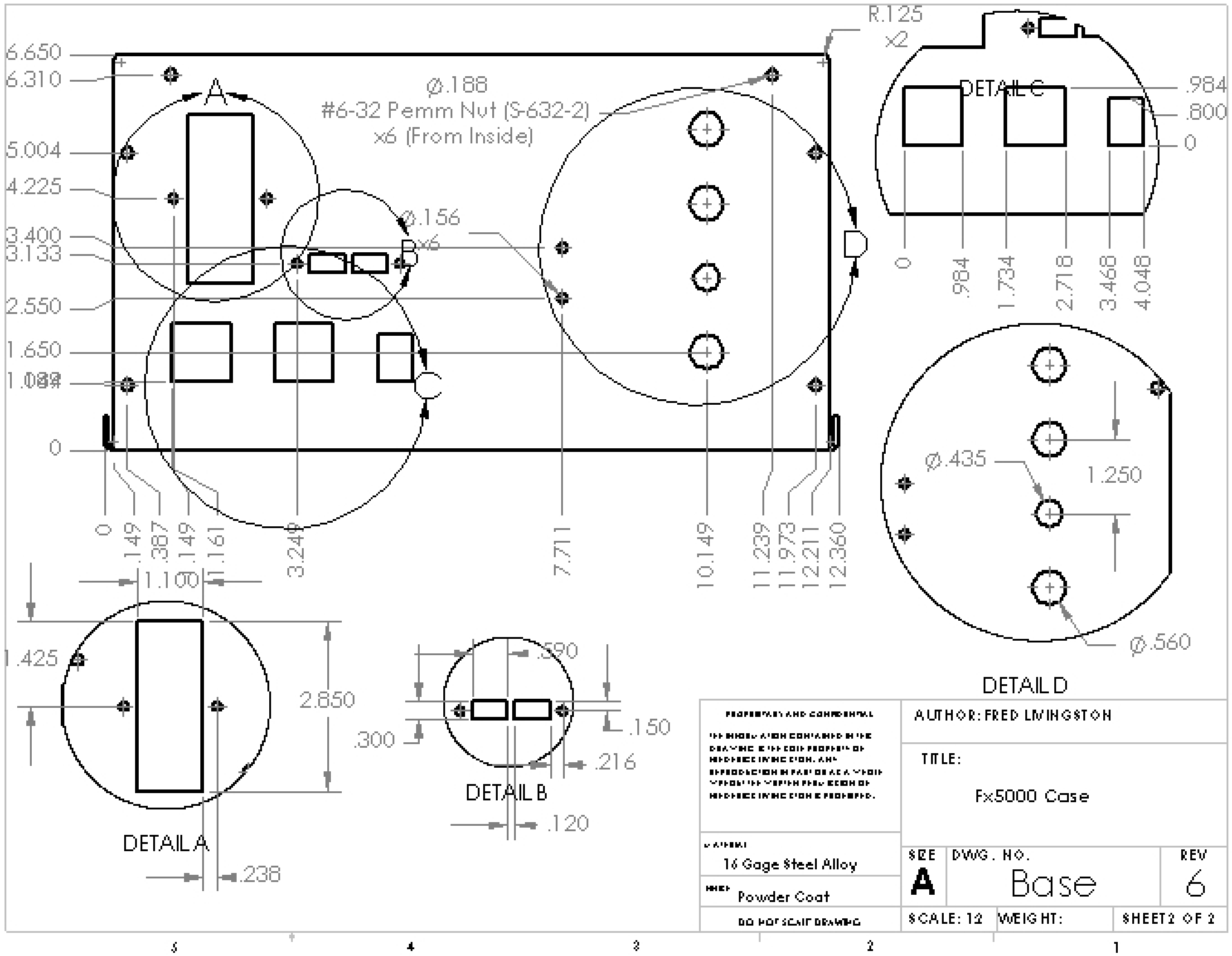


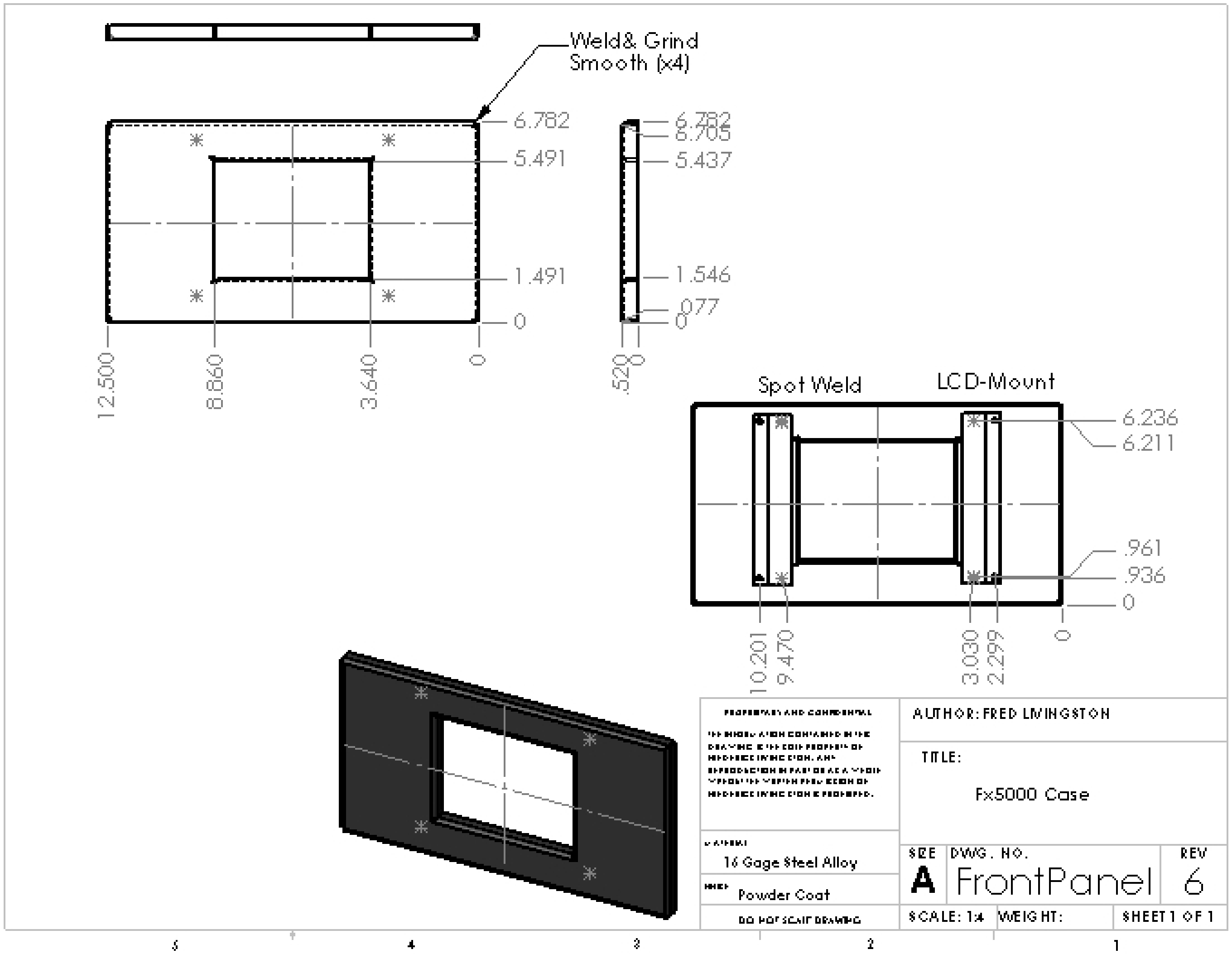
⌀.156
x6
6.560
6.140
4.853
.932
0
12.500
12.043
11.830
11.295
1.205
.670
.457
0
6.560
A
1.250
0
17.439
11.996
7.316
0
.540
.540
⌀.125
.270
.312
DETAIL A
SCALE 1 : 2
AUTHOR: FRED LIVINGSTON
TITLE:
Fx5000 Case
16 Gage Steel Alloy
Powder Coat
SIZE
A
DWG. NO.
Cover
REV
6
SCALE: 1:2
WEIGHT:
SHEET 1 OF 2


Tab-SideB
Tab-SideA
4.081
3.206
2.331
1.456
1.232
1.000
.245
0
.350
4.645
6.645
8.645
10.645
12.645
B
C
.274
0
.724
DETAIL C
SCALE 1 : 2
1.232
.750
.232
DETAIL B
SCALE 1 : 2
Note: * = Spot Welds
Tab-Front
F
11.250
9.250
7.250
5.250
3.250
1.250
.278
DETAIL F
SCALE 1 : 2
AUTHOR: FRED LIVINGSTON
TITLE:
Fx5000 Case
16 Gage Steel Alloy
Powder Coat
SIZE
A
DWG. NO.
Cover
REV
6
SCALE: 1:5
WEIGHT:
SHEET 2 OF 2

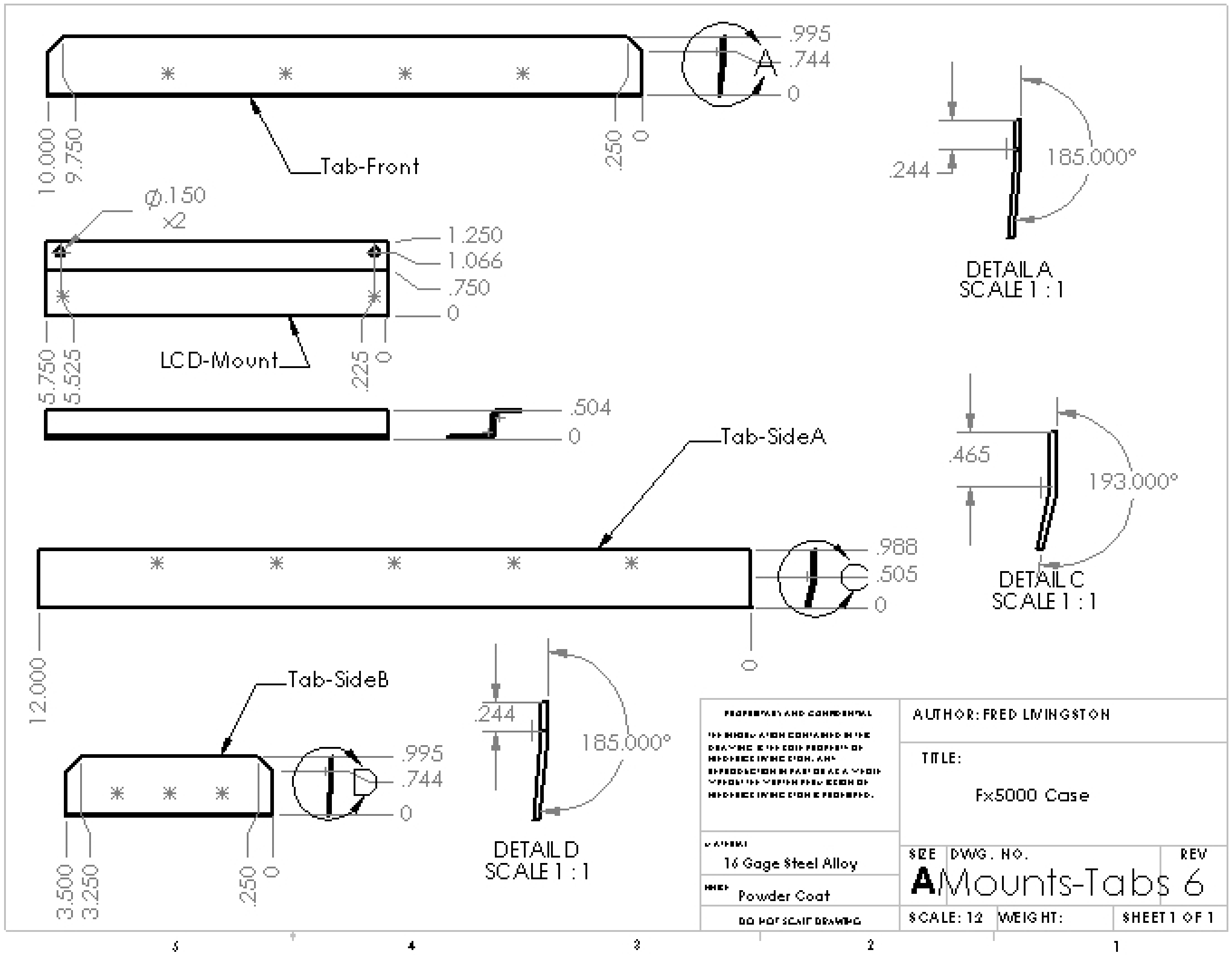


6.650
.580
0
17.859
.149
0
⌀.156
x10
17.789
17.359
16.359
⌀.177
12.609
⌀.177
11.109
⌀.177
9.734
8.734
⌀.177
7.080
⌀.177
4.359
4.109
⌀.177
0
12.360
11.933
9.985
8.572
7.985
7.391
6.583
5.034
2.648
1.496
0
Weld Panel to Base
Along botton edge
17.929
17.409
0
PROPRIETARY AND CONFIDENTIAL
AUTHOR: FRED LIVINGSTON
TITLE:
Fx5000 Case
16 Gage Steel Alloy
Powder Coat
DO NOT SCALE DRAWING
SIZE
A
DWG. NO.
Base
REV
6
SCALE: 1:4
WEIGHT:
SHEET 1 OF 2

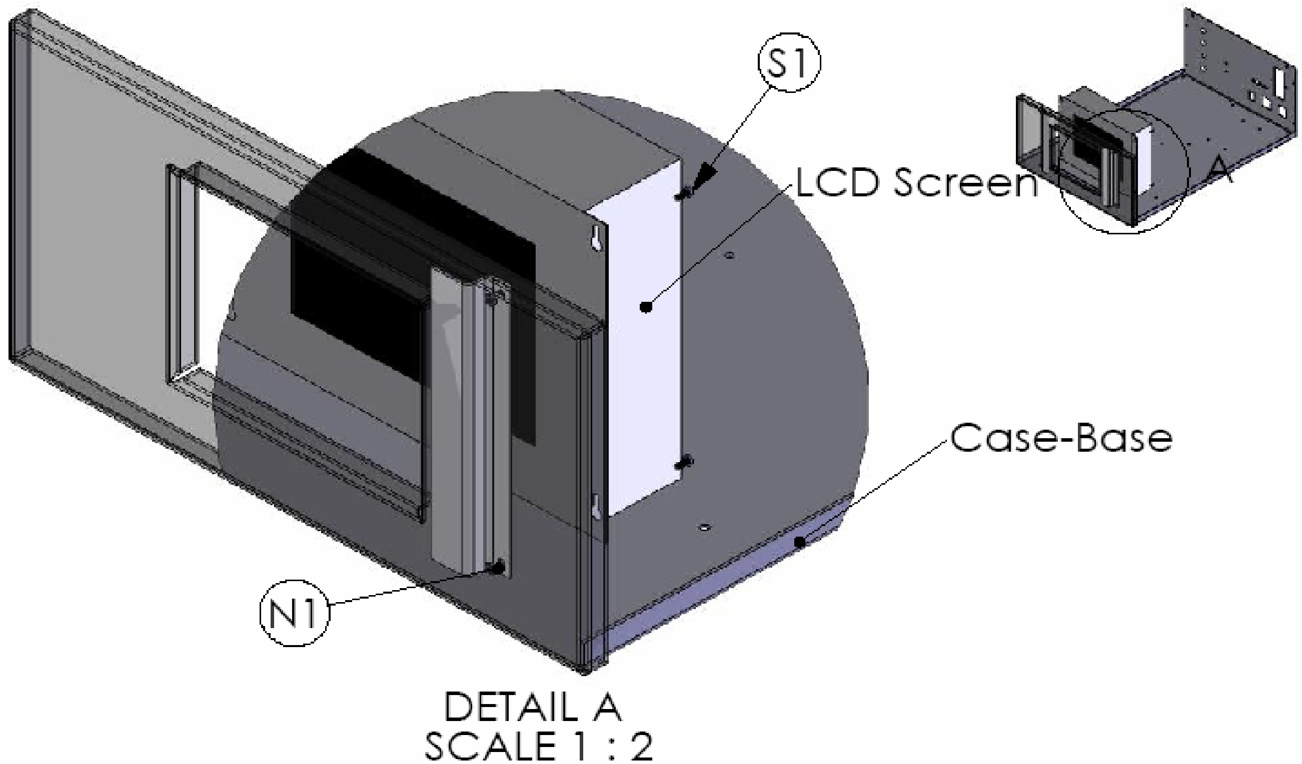

6.650
6.310
5.004
4.225
3.400
3.133
2.550
1.650
0
Ø.188
#6-32 Pemm Nut (S-632-2)
x6 (From Inside)
Ø.156
x6
A
B
R.125
x2
DETAIL C
.984
.800
0
0
.984
1.734
2.718
3.468
4.048
0
.149
.387
.149
.161
3.249
7.711
10.149
11.239
11.973
12.211
12.360
1.100
1.425
2.850
.238
DETAIL A
.590
.300
.150
.216
.120
DETAIL B
Ø.435
1.250
Ø.560
DETAIL D
PROPRIETARY AND CONFIDENTIAL
AUTHOR: FRED LIVINGSTON
TITLE:
Fx5000 Case
16 Gage Steel Alloy
Powder Coat
DO NOT SCALE DRAWING
SIZE
A
DWG. NO.
Base
REV
6
SCALE: 1:2
WEIGHT:
SHEET 2 OF 2
5
4
3
2
1



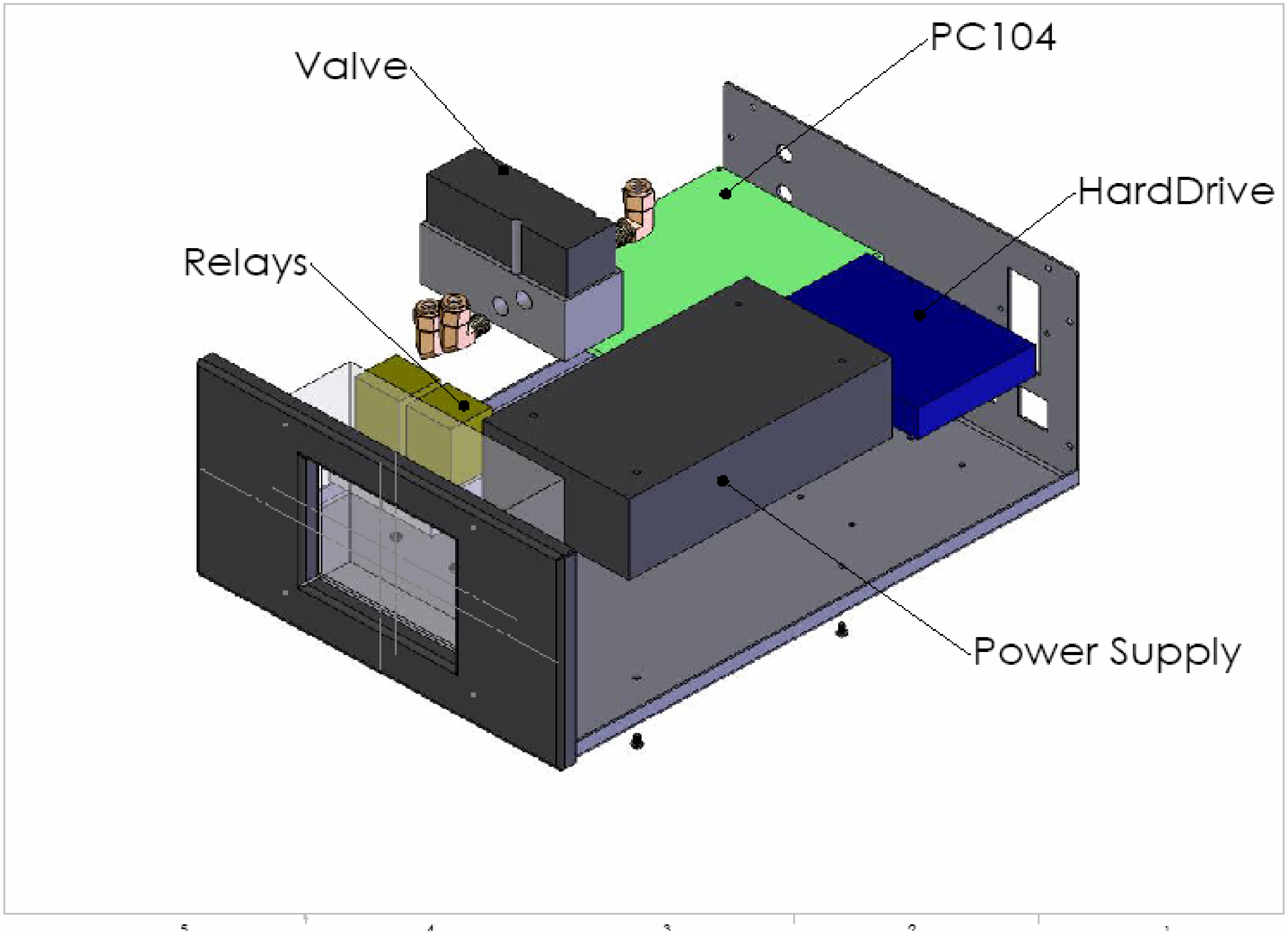
Weld& Grind Smooth (x4)
6.782
5.491
1.491
0
12.500
8.860
3.640
0
6.782
6.705
5.437
1.546
.077
0
.520
0
Spot Weld
LCD-Mount
6.236
6.211
.961
.936
0
10.201
9.470
3.030
2.299
0
AUTHOR: FRED LIVINGSTON
TITLE:
Fx5000 Case
16 Gage Steel Alloy
Powder Coat
SIZE
A
DWG. NO.
FrontPanel
REV
6
SCALE: 1:4
WEIGHT:
SHEET 1 OF 1

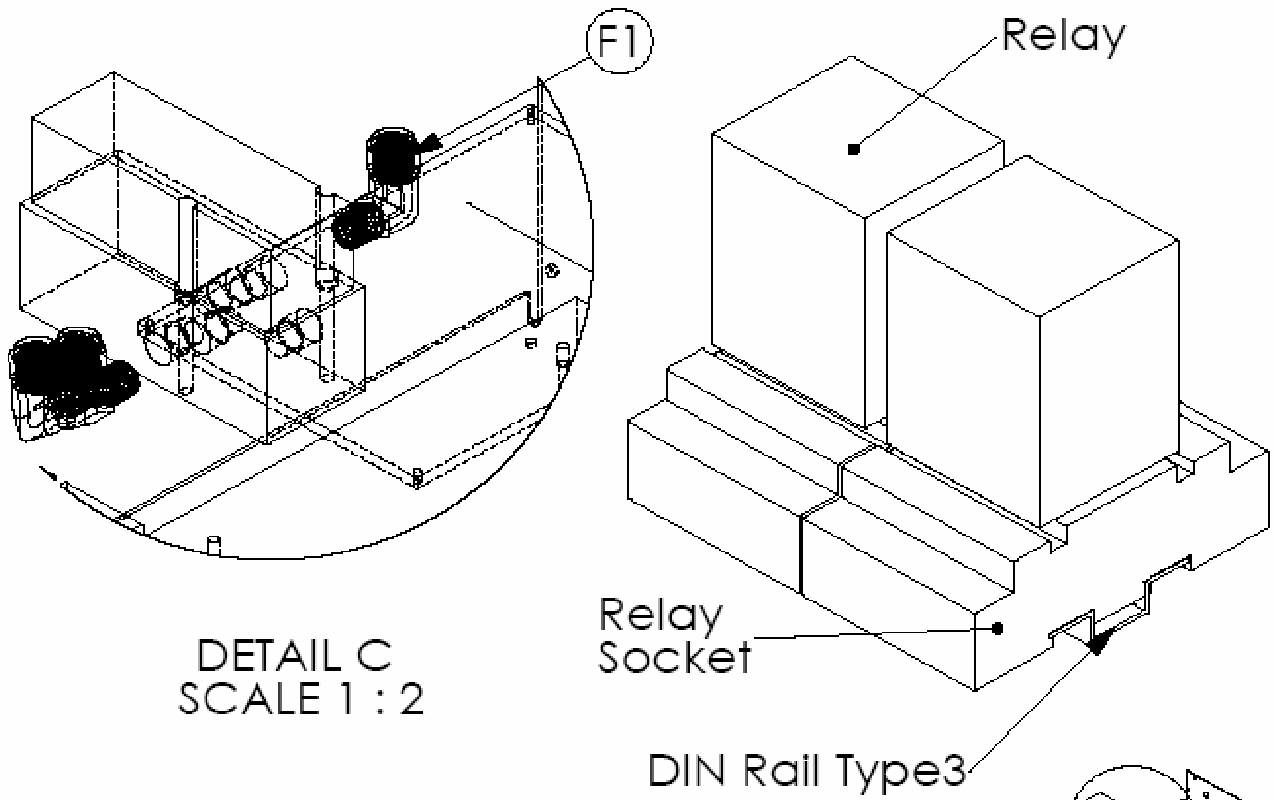

Tab-Front
10.000
9.750
.250
0
.995
.744
0
A
Ø.150
x2
1.250
1.066
.750
0
LCD-Mount
5.750
5.525
.225
0
.504
0
.244
185.000°
DETAIL A
SCALE 1 : 1
Tab-SideA
.465
193.000°
.988
.505
0
C
DETAIL C
SCALE 1 : 1
12.000
0
Tab-SideB
.995
.744
0
D
3.500
3.250
.250
0
.244
185.000°
DETAIL D
SCALE 1 : 1
AUTHOR: FRED LIVINGSTON
TITLE:
Fx5000 Case
16 Gage Steel Alloy
Powder Coat
SIZE
DWG. NO.
REV
A
Mounts-Tabs
6
SCALE: 12
WEIGHT:
SHEET 1 OF 1

Appendix 6. Assembly Plan

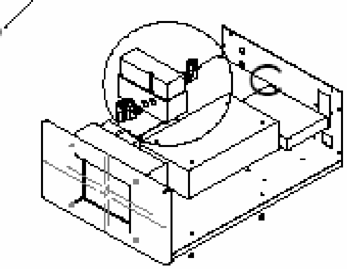


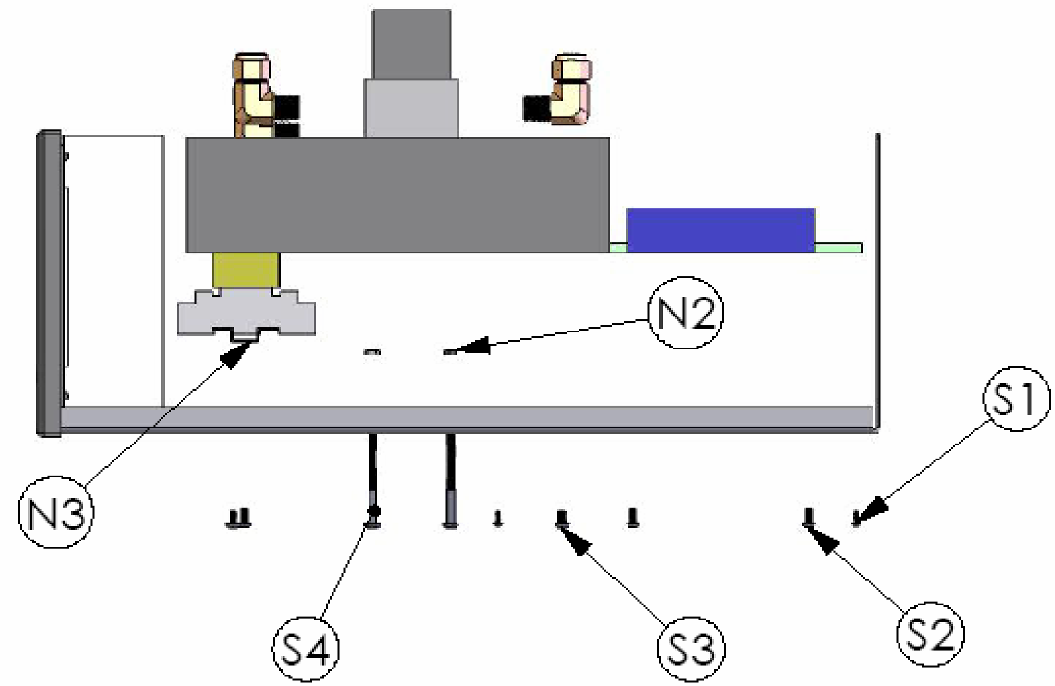
	Type	Thread	Length	Tool	Quantity	McMaster#
S1	Screw	#3-48	1/4"	#1 Phillips	10	91773A092
N1	Flang Nut	#3-48	N/A	3/16 Hex Wrench	4	91841A004


Valve
PC104
HardDrive
Relays
Power Supply

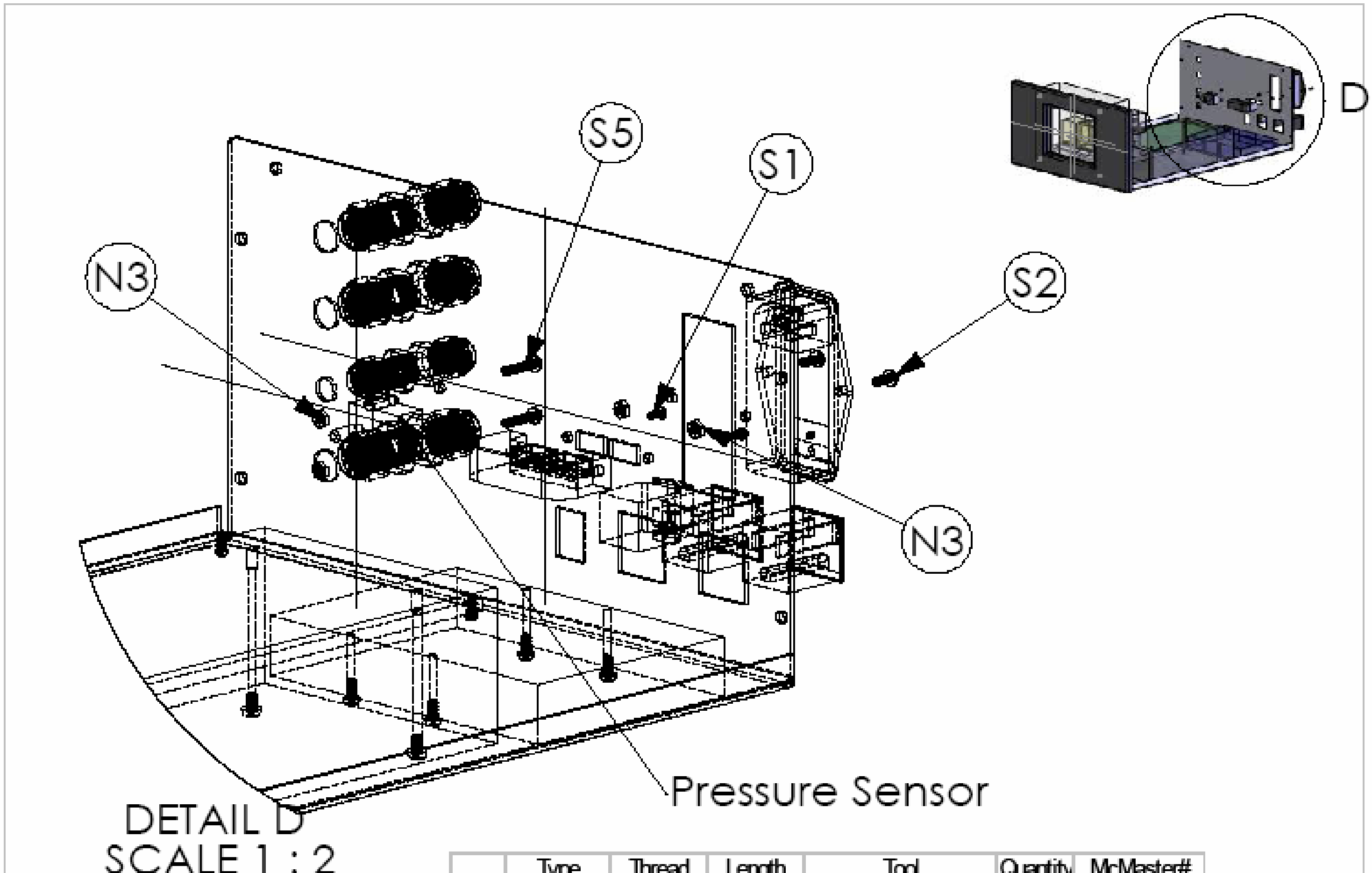


	Type	Thread	Material	McMaster#
F1	Compression Tube Fitting Elbow	1/4 NPT	Brass	5272K204

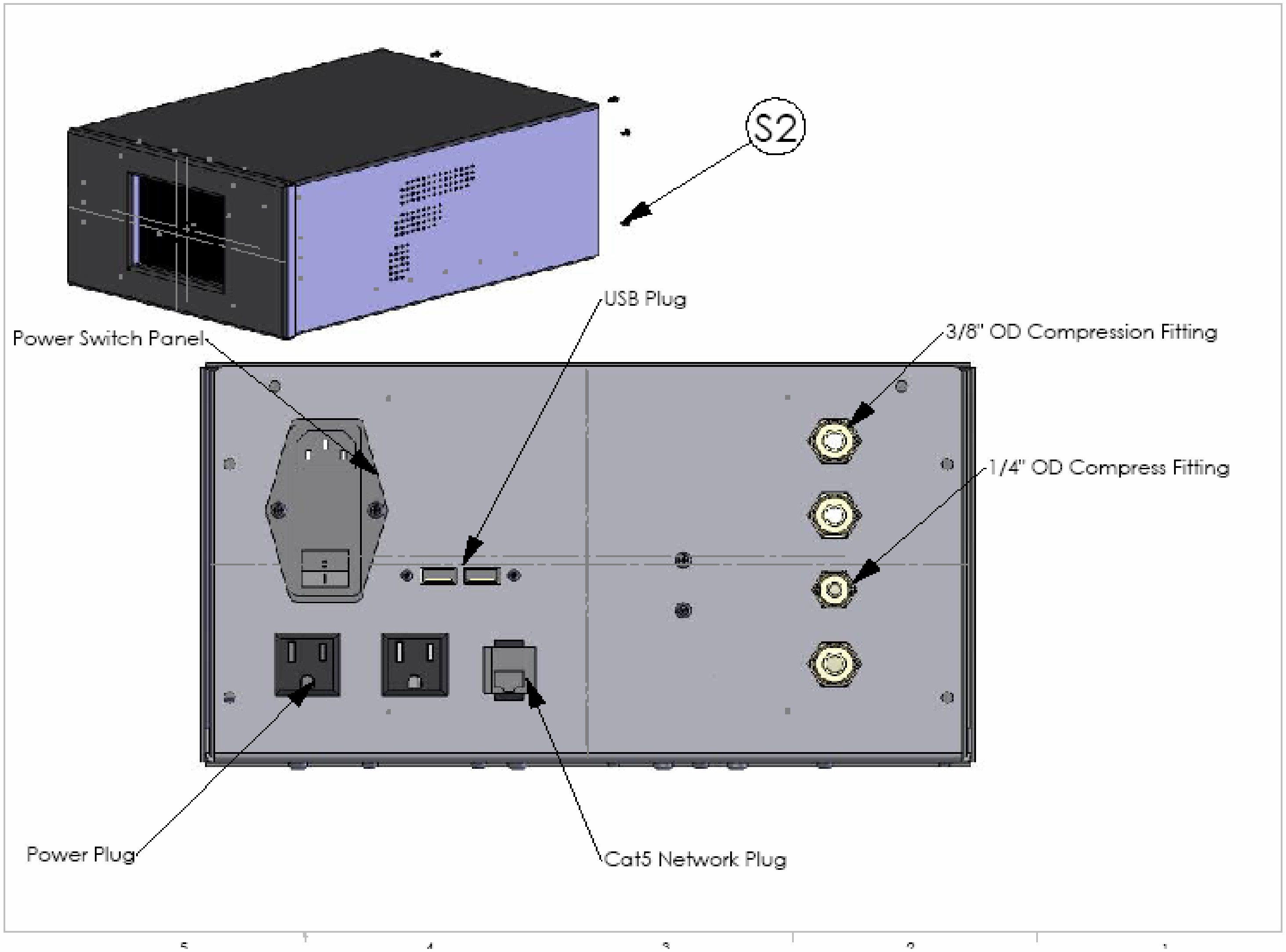




	Type	Thread	Length	Tool	Quantity	McMaster#
S1	Screw	#3-48	1/4"	#1 Phillips	10	91773A092
S2	Screw	#6-32	5/16"	#2 Phillips	14	91773A145
S3	Screw	#8-32	1/4"	#2 Phillips	4	91773A190
S4	Screw	#8-32	2-1/4"	#2 Phillips	2	91772A206
N2	Nut	#8-32	N/A	1/4" Hex Wrench	2	90205A309
N3	Flang Nut	#6-32	N/A	5/16" Hex Wrench	6	93776A370



	Type	Thread	Length	Tool	Quantity	McMaster#
S5	Screw	#6-32	5/8"	#2 Phillips	2	91773A150



S2
USB Plug
3/8" OD Compression Fitting
Power Switch Panel
1/4" OD Compress Fitting
Power Plug
Cat5 Network Plug

Appendix 7. Pneumatic Culture System Pictures

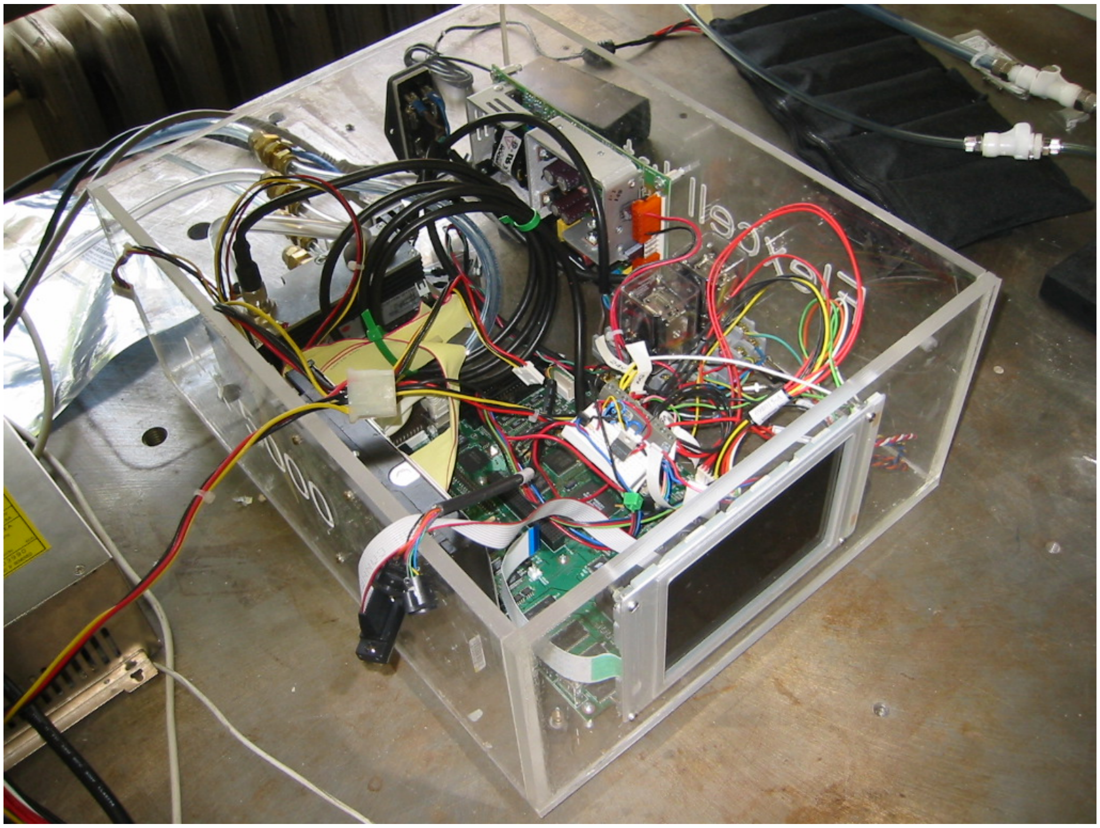


Figure 62. Initial Prototype

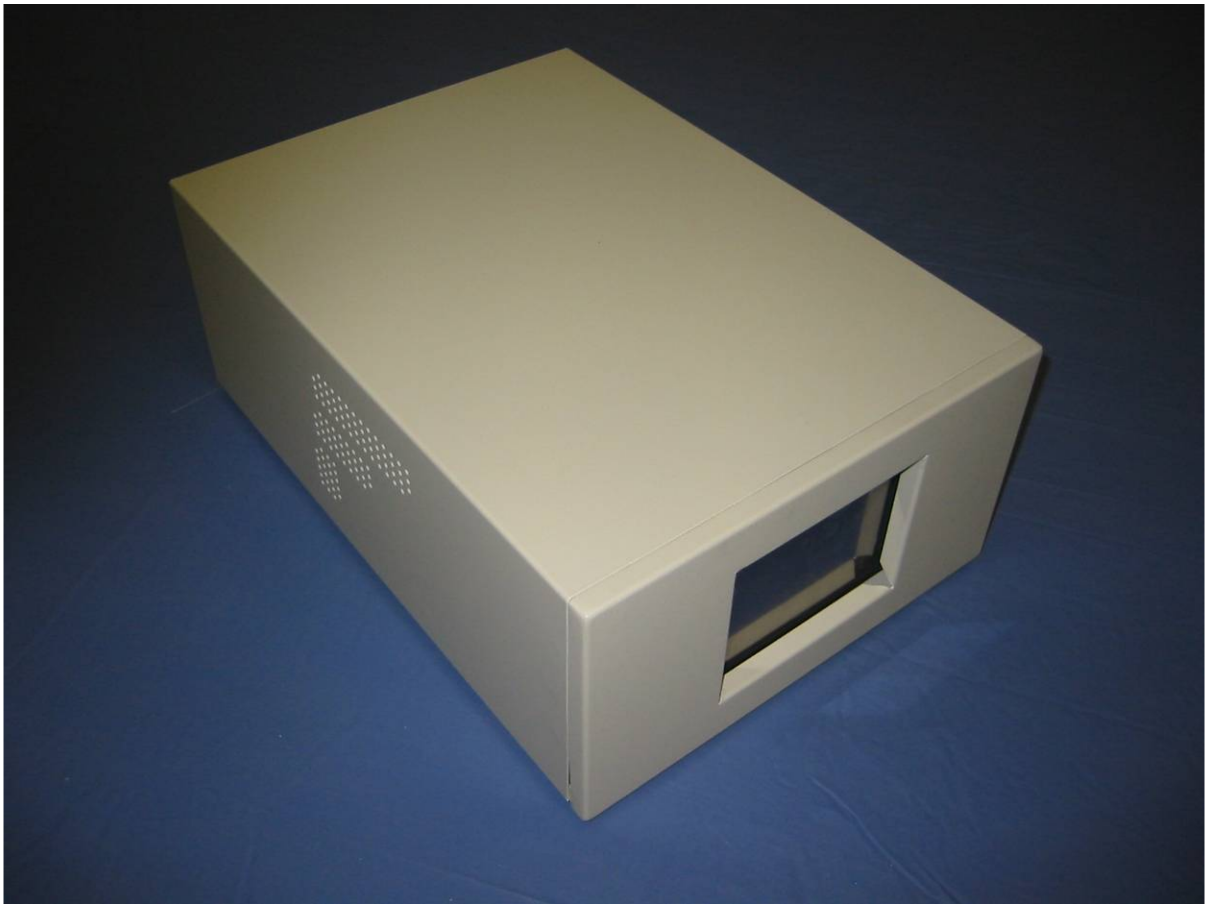


Figure 63. Finish System View1

Figure 64. Finish System View2

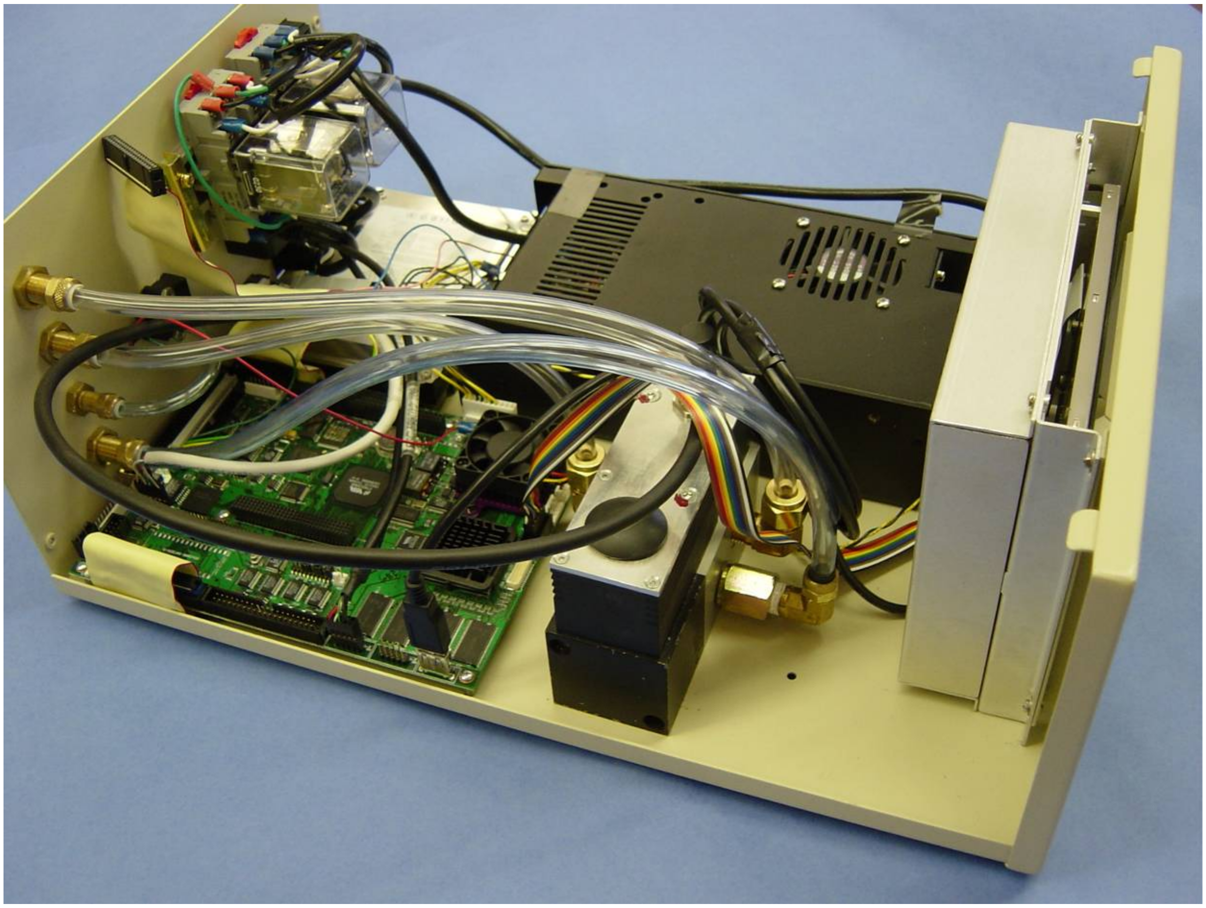


Figure 65. Finish System View3

Appendix 8. Pneumatic Culture System Movie

Fx5000-Movie.wmv