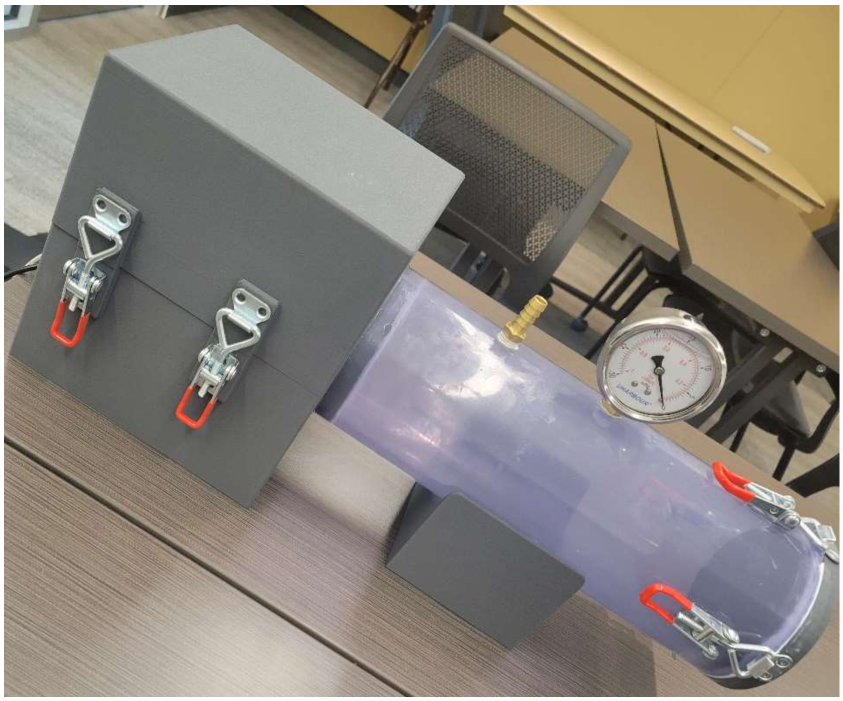


Vacuum Vibes

David Kimball, John Lear and Julia Swarbrick

Southern New Hampshire University

4/23/2024

Abstract

This project's objective is to create a versatile and reliable testing environment capable of simulating a wide range of altitudes to meet the stringent requirements of aerospace research and development.

The altitude chamber design encompasses a robust structural framework, precise control systems, and advanced safety features. The chamber is equipped with instrumentation to monitor and regulate temperature, and pressure within tight tolerances from altitudes of sea level to 5000 meters (about 3.11 mi). The design prioritizes flexibility to accommodate various testing scenarios, allowing for the simulation of conditions that aerospace components may encounter during their operational life.

In addition to its core functionality, the altitude chamber integrates technologies to enhance user interface and data acquisition. A user-friendly control interface enables researchers to easily configure and monitor test parameters, ensuring a seamless and efficient testing experience while simulated altitude conditions.

The collaborative effort of the engineering design group emphasized a multidisciplinary approach, drawing on expertise in mechanical, electrical, and software engineering. The resulting altitude chamber stands as a testament to planning, and iterative design refinement. This project contributes to aerospace research capabilities by providing a reliable and adaptable platform for testing and validating critical components that operate in diverse environmental conditions.

Contents

- Abstract 2
- Project Description 5
  - Problem statement 5
  - Summary of Concept of Operations 5
- System Requirements and Verification 6
  - Environmental 6
  - Safety Requirement 6
  - Functional Requirement 7
- Resource and Functional Requirements 7
  - Performance Requirement 7
- Verification 8
- Conceptual Design 9
- Detailed Design 13
  - Overall system description 13
  - Mechanical Subsystem 16
  - Electrical/Electronic Subsystem 20
  - Software/Programming Subsystem 22
  - Design Features 24

Fabrication and Integration 25

Comprehensive Performance Test 27

- Schedule 28
- Project Budget 29

Conclusion and Lessons Learned 30

- Appendices 32
  - Appendix A. Verification Report 32
  - Appendix B. Acceptance Data Package List 32

List of Tables

Table 1: Environmental Requirements 6
Table 2: Safety Requirement 6
Table 3: Functional Requirement 7
Table 4: Resource and Functional Requirements 7
Table 5: Performance Requirements 7
Table 6: Microcontroller Matrix 11
Table 7: PVC Pipe Matrix 11
Table 8: Software Flow Chart 23

List of Figures

Figure 1: Verification Matrix 8
Figure 2: ALTI-2 Competitors Design 9
Figure 3: AvionTEq Competitors Design 9
Figure 4: Altimeter Schematic 10
Figure 5: Photo of Complete System 13
Figure 6: Photo of Complete Electrical Components Box. 14
Figure 7: Block Diagram of System and Components 15
Figure 8: Mechanical Subsystem Design 16
Figure 9: Back Cap 3D Print. 17
Figure 10: Front Cap 3D Print. 17
Figure 11: Control System Box Top 18
Figure 12: Control System Box Bottom 18
Figure 13: PVC Pipe Stand 19
Figure 14: Analog Guage 19
Figure 15: Control System Block Diagram 20
Figure 16: Schematic Diagram 21
Figure 17: Program Manager Schedule 28

Project Description

Problem statement

Create an accurate, affordable, and user-friendly altimeter test chamber that can function in a range of settings with minimal user training.

Summary of Concept of Operations

System:

The system will consist of a control unit and a vacuum chamber. The control unit will be PC based and controlled by Arduino and LabView, will read environmental conditions and chamber pressure, and will control a vacuum pump and a pressure relief valve to regulate chamber pressure. The chamber will be cylindrical in shape, large enough to fit a standard altimeter, and will be able to sit in a stable manner on a work bench.

Operation:

The operation of the chamber will be powered by an electric vacuum pump. Using this there will be an input made into a code to test a specific elevation. The automatic control system will adjust the chamber pressure to match the pressure expected at the requested altitude. Then the altimeter will read the given pressure level.

Use:

The elevation chamber system will be used to replicate environments to test an airplane altimeter. The user must be able to perform full operation of the system in less than five minutes. In manufacturing environments, the system will be used daily, whereas in a maintenance or hobbyist setting use will be sporadic.

Support:

The altimeter chamber based off of use needs to be calibrated every 6 months to a year. For an environment within a test/assembly, the altimeter chamber would be used more frequently requiring an interval closer to 6 months, whereas in a maintenance team setting it would be used more commonly for troubleshooting, which means the calibration cycle would be every year. The calibration items needed would be:

- Known good altimeter.
- A parameter of levels needing to be read based on a set of barometric pressures (a process).
- A program to pull the data into a collection site.

Impact Considerations:

A less expensive alternative for altimeter testing. It also allows for rapid on-site verification of hardware with minimal training, expanding the ease with which equipment can be verified.

System Requirements and Verification

Initial requirements were set to define system scope. These requirements included operation range and capability specifications, operational performance targets, general physical specifications, environmental resilience expectations, and required safety features.

Except for the specification noted below, all requirements were either met or de-scoped. Regarding the de-scoped requirements, the pressure relief valve requirement was removed due to determination that it was overly redundant, as the front door can be easily removed whenever necessary, and the warning indicators were de-scoped after it was found that LabVIEW functionality was not feasible, as these indicators were to be built in to that portion of the project and were also deemed to be redundant features that were not critical to safe operation.

Table 1: Environmental Requirements

Environmental	Description	Notes/Status
Ambient Temperature	The System shall operate at ambient temperature ranging from -10°F to 110°F.	(Not tested, environmental condition simulator not available)
Stable	The System shall rest in a stable manner on a work bench.	Met, test
Dry	The System shall only operate in dry environments of ≤ 60% humidity.	(Not tested, environmental condition simulator not available)
Low dust	The System shall only operate in low-dust environments [possible ppm value TB.	(Not tested, environmental condition simulator not available

Table 2: Safety Requirement

Safety Requirement	Description	Notes/Status
Under-pressure	The System shall warn users of under-pressure conditions [possible value].	De-scope
Over-pressure	System shall warn users of over-pressure condition [possible value].	De-scope
Relief valve	The System shall incorporate relief valve as emergency pressure vent.	De-scope
Operation	Main chamber shall operate without needing operation within 5 ft.	Met, demonstration
Hazardous Material	The System shall not have exposed hazardous materials.	Met, inspection

Table 3: Functional Requirement

Functional Requirement	Description	Notes/Status
Autonomous	The System shall allow for simulated elevation from 0-5000m automatically.	Met, test
Pressure	The system shall incorporate an electric pressure sensor.	Met, inspection
Display	The system shall display chamber pressure in kPa.	Met, demonstration
User Commands	The System shall respond to user commands via user interface.	Met, demonstration
Temperature Changes	The System shall account for temperature changes due to varying altitudes.	Met, demonstration and analysis

Table 4: Resource and Functional Requirements

Resource and Functional Requirements	Description	Notes/Status
Transportation and Storage	The system shall be packaged into one unit for transportation and storage	Met, observation
Altimeter Accommodation	The system shall accommodate a standard 3.5” pattern altimeter	Not Met (see note 1)
Weight	The system shall weigh less than 15 lbs.	Met, test

Table 5: Performance Requirements

Performance Requirement	Description	Notes/Status
Test Cycles	Complete test cycles shall take no longer than 5 minutes.	Met, test
Accuracy	System shall have altitude accuracy of ± 3 %.	(Not tested, school’s manometer broken, gauge used to verify has lower accuracy than ± 3 %)
Vacuum Pump	The system shall utilize electric vacuum pump to achieve altitude simulation.	Met, inspection
Current Draw	System shall draw no more than 10 amps under operation.	Met, analysis

Verification

Verification Matrix

Requirement Type→	Performance				Functional			Design			Interface		Resource		Verification Reference
Requirement Title→	Initial Start Up/Shutdown	Pressure Sealing	Control	Operation	Pressure Control	Front Cover Cap Latching	Electrical Box	Materials	3D Print	Wiring/Electrical Placement	Mechanical	Electrical	Mass	Power	This column identifies the specific verification summary related to the activity on each line. Multiple summaries may apply to a single line item.
Level of Assembly															
System															Section 3.3.X System Refrences
Altitude Simulator	T				DM						DI	DI	M	M	Section 3.3.1 - 3.3.6 *See note 2
Subsystem/Assembly															Section 3.2.X Subsystem Refences
Mechanical/Housings		T			T	T	MV	V	TA	MV	T				Section 3.2.1-3.2.11 *See note 3
Electrical/Control	DTM		DM	DM	DT		I			DI	DI	DI	AV	AV	
Component															Section 3.1.X Component Refences
Arduino Uno R4	D		D		DT		I			DI	DI	D	AV	AV	3.1.1 *See note 5
Vac. Pump	D		DT	DM	DT		I			DI	DI	D	AV	AV	3.1.2 *See note 5
Vent Valve	D		DT	T	DT		I			DI	DI	D	AV	AV	3.1.3 *See note 5
Pressure Sensor	TM		DT	T	DT		I			DI	DI	D	AV	AV	3.1.4 *See note 5
Relay Block	DM		DM	T			I			DI	DI	D	AV	AV	3.1.5 *See note 5
Power Harness	TM		TD				I			DI	DI	D	AV	AV	3.1.6 *See note 5
Communication Cable	TD		D				I			DI	DI	D	AV	AV	3.1.7 *See note 5
Software			D		D							D			3.1.8*See note 6
Front Cap		T			T	T		DI	TM		DI				3.1.9 *See note 5
Back Cap		T			T			DI	TV		DI				3.1.10 *See note 5
PVC Pipe		T			T			DI	AD		DI				3.1.11 *See note 5
PVC Stand								DI	DI		DI				3.1.12 *See note 5
ELectrical Components Box							T	DI	DI	DI	DI				3.1.13 *See note 5
Hardware						TV		DI	DI		DI				3.1.14 *See note 5

Verification Method:

A - Analysis

D - Demonstration

I - Inspection

M - Measurement

T - Test

V - Validation of Records

Notes:

1. Color Scheme: Activities completed highlighted in GREEN; Activities to be completed highlighted in GOLD

2. System References: Initial Startup/Shutdown (3.3.1), Pressure Control (3.3.2) Mechanical (3.3.3), Electrical (3.3.4), Mass (3.3.5), Power (3.3.6)

3. Subsystem References: Pressure Sealing (3.2.1), Pressure Control (3.2.2), Front Cover Cap Latching (3.2.3), Electrical Box (3.2.4), Materials (3.2.5), 3D print (3.2.6), Wiring /Electrical Placement (3.2.7), Mechanical (3.2.8), Electrical (3.2.9), Mass (3.2.10), Power (3.2.11).

4. Component-Level Refrences: Arduino Uno R4 (3.1.1), Vac. Pump (3.1.2), Vent Valve (3.1.3), Pressure Sensor (3.1.4), Relay Block (3.1.5), Power Harness (3.1.6), Communication Cable (3.1.7), Software (3.1.8), Front cap (3.1.9), Back Cap (3.1.10), PVC Pipe (3.1.11), PVC Stand (3.1.12), Electrical Components Box (3.1.13). Hardware (3.1.14).

5. 3.1.1-3.1.14 Component verfications specified in verification breakdown.

Figure 1: Verification Matrix

Conceptual Design

Initial design and reasoning

Our design is going to be something simpler than those of the competitors. The vacuum vibes have been designed for something that can be used on a technician's bench, calibration, testing and troubleshooting. Most competitors' designs have shown themselves to be a little less compact. Possibly competitors could take up more room on a manufacturing floor plane and bench space.

Below you can see one that is used for aircraft side troubleshooting made by ALTI-2. This is a little more in-depth gauge reader and chamber style.

Figure 2: ALTI-2 Competitors Design

Another style would be something more for a test bench that a technician would use daily. One of the challenges with this one is the down time for the calibration. This can be down to whether it can be done on sight or would have to go to the vender. This one is manufactured by AvionTEq:

Figure 3: AvionTEq Competitors Design

One of the challenges we will have to overcome is how we can properly seal the chamber. This is tough because of the number of access-points the design incorporates. Another challenge is the sourcing of an altimeter within budget for the project. The chamber assembly is going to have a budget that fits within all companies' ranges. Whereas our competitors are going to be above price margin for many small name companies. The last biggest challenge is how to keep the down time to a minimum, so there is more up time within the use of the product. Down time can include scheduled maintenance, chamber breaks, or even calibration.

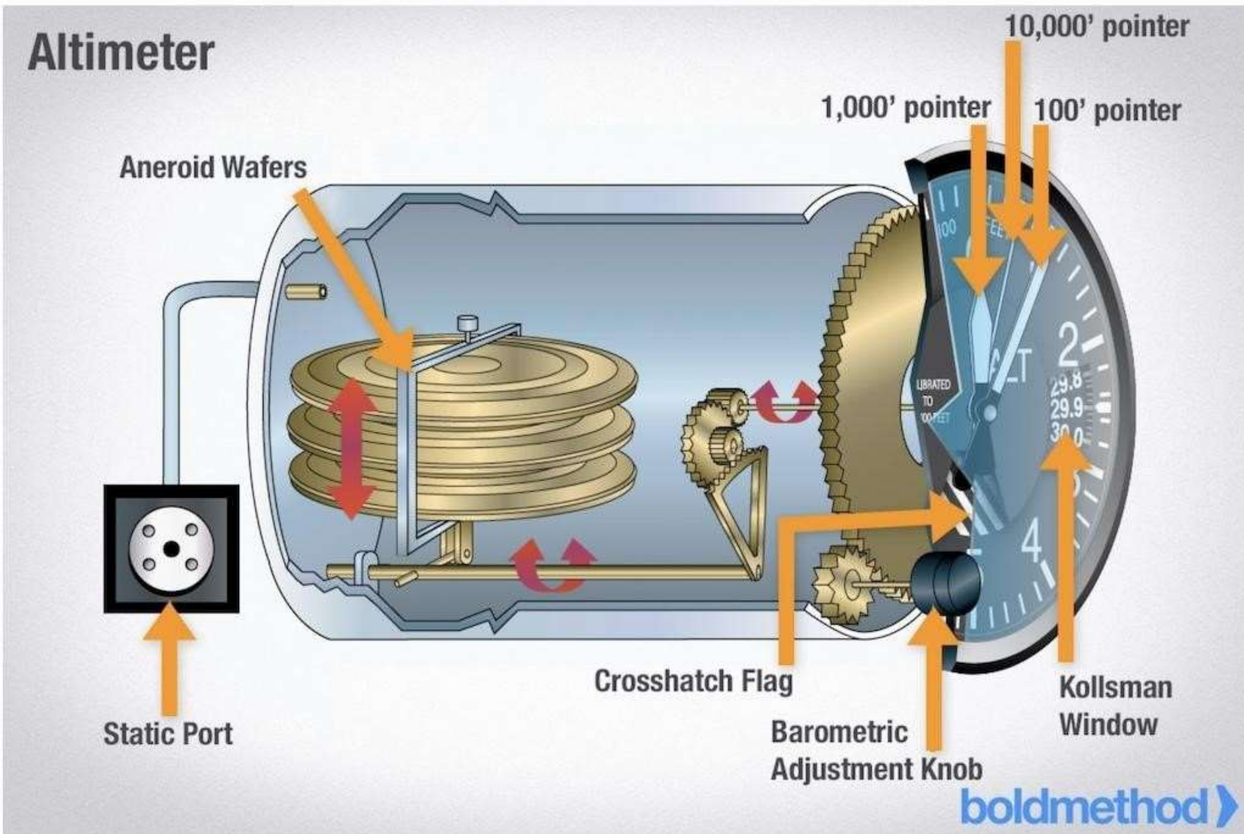


Figure 4: Altimeter Schematic

The altimeter chamber's goal is to simulate a flight scenario so that the altimeter being tested can be within the right range. Below shows a description of the different parts and how they work:

1. This is changing the mercury input. 10,000 ft and below every 100 ft it is 1/10 in of mercury. In higher altitudes, a 1/10 in of mercury is 300 ft.
2. Travels through the **aneroid wafers** which expand and contract as atmospheric pressure from the static port changes.
3. The mechanical linkage translates these to the pointers. These pointers read like a clock, so a small hand indicates thousands of ft and a large hand indicates hundreds of ft.
4. A **crosshatch flag** appears at any level below 10,000 ft.
5. The **Kollsman window** is controlled by the **barometric adjustment knob**. This number is given by air traffic control to pilots. A lower number lowers the indicated altitude. And raising this number raises the indicated altitude.

We chose this system because it gave our group a good mix of challenges. The system has aeronautics, coding, 3D printing and machining involved. Which meant everyone in the group got to see how different processes for each happened. To start the initial calculations for the pressure allowed to the chamber we needed to use what we learned in statics and find the max hoop stress of system. Our electrical system required some splicing and soldering. It is also a good scope for the short amount of time we had with the course only being one semester.

Trade Off Studies

Table 6: Microcontroller Matrix

Trade Study Matrix (TS-004)								
WBS:		**Study: Microcontroller**						**Trigger:**
Trade Configuration		**Impact Assessment**						**Status**
Option	**Description**	**dimensions**	**Microcontroller**	**Software compatibility**	**Flash memory**	**Cost**	**Risk**	
1	Arduino Uno	53.4 x 68.6 mm	ATmega328P	Yes	32 KB	$22.00	Insufficient current to input pins	Selected
2	Rasperry Pi	85 x 56 mm	RP2040	Yes*	2 MB	$60.00	Compatibility issues	Rejected
3	Elegoo MEGA 2560	101.5 x 53.3 mm	ATmega2560	Yes	256 KB	$19.99	Design changes to accommodate larger dimension	Rejected
4	NodeMCU	49 x 26 mm	ESP-8266	Yes	4 MB	$6.99	Insufficient digital pin terminals. Difficult to source microcontroller	Rejected

Table 7: PVC Pipe Matrix

Trade Study Matrix (TS-001)								
WBS:		**Study: PVC pipe**						Trigger:
Trade Configuration		**Impact Assessment**						Status
Option	**Description**	**max presure**	**weight**	**diameter ID**	**diameter OD**	**Cost**	**Risk**	
1	Clear PVC	110 psi	2 lb	4 in	4.5 in	$38	N/A	Selected
2	white PVC	100-150 psi	2.8 lb	4 in	4.5 in	$17	no visibility	Rejected

Feasibility study and prototyping

As other versions of this equipment have a long history of use in industrial applications, our feasibility study consisted of identifying any disqualifying budgetary or material strength considerations. First, a first-pass stress analysis was performed with very general material property and dimensional specifications. It was determined that, due to the relatively low delta pressure this system would create, there would be no practical limitations regarding material or shape selection related to potential stress failure. Second, initial research was done to determine rough component costs. While the initial plan was to have an actual altimeter for demonstration purposes, it was determined that even used altimeters cost more than our entire budget, and we were also unable to locate anyone who was willing to loan us one. However, it was found that all other components could be sourced with a solid budget margin, so the decision was made to proceed with this project.

The prototyping phase consisted primarily of electronics testing and initial calibration. First, the pump and vent valve were powered on and off manually, and operation was verified. Next, the pressure sensor was connected to the Arduino R4 and a rough calibration was performed with a hand pump to convert the analog signal to a kPa value. A breadboard was then built with a series of buttons to simulate various pressure conditions and two LEDs to simulate the pump and vent valve. The basic Arduino code was tested and adjusted with these components to ensure that coding issues would not result in component damage. The next step was to replace the buttons with the pressure sensor and again test and adjust the Arduino code using a vacuum pump to simulate changes in chamber pressure. The relay block was then added, again being used to control LEDs instead of the pump and vent valve. Finally, 12v was applied to the relay block and the vent valve and pump were connected to it, and full operation was simulated to verify that the relay block could properly control the pump and vent. After successful control of this test, no further testing was deemed practical until fabrication of the vacuum chamber was complete, and the prototyping phase was concluded.

Detailed Design

Overall system description

The full vacuum chamber system consists of the main chamber, the control housing, an analog pressure gauge, an auxiliary/calibration port, and a chamber door. The door, control housing shells, chamber support, and chamber back plate are 3d printed in ABS, and the chamber body is machined from clear PVC tubing. All components are held to the assembly either by direct threading or high-strength adhesive. Sealing was achieved using thread tape on all pipe fittings, silicone caulking on the back plate, and an O-ring seal on the chamber door. The completed system is shown in the figures below.

Figure 5: Photo of Complete System

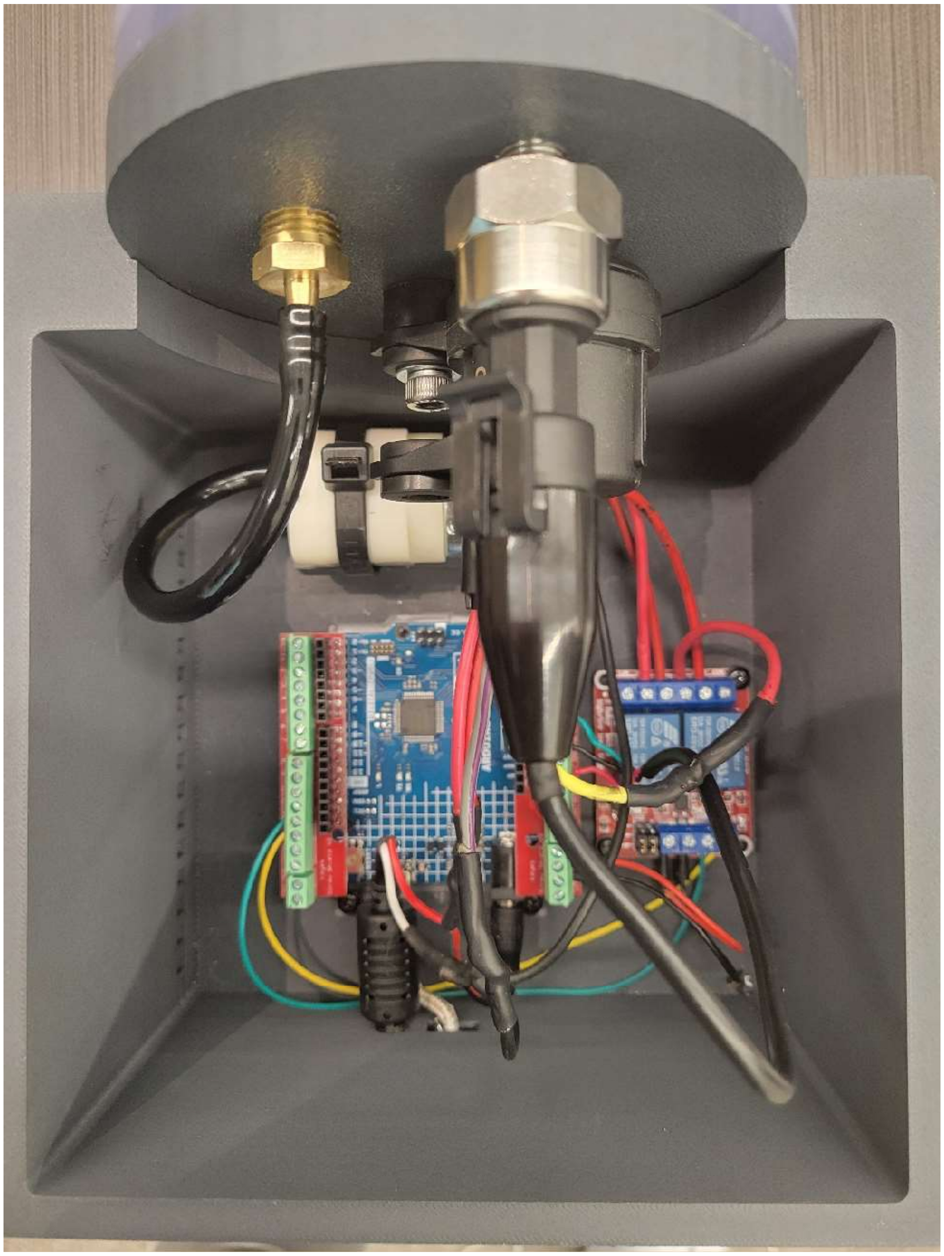


Figure 6: Photo of Complete Electrical Components Box.

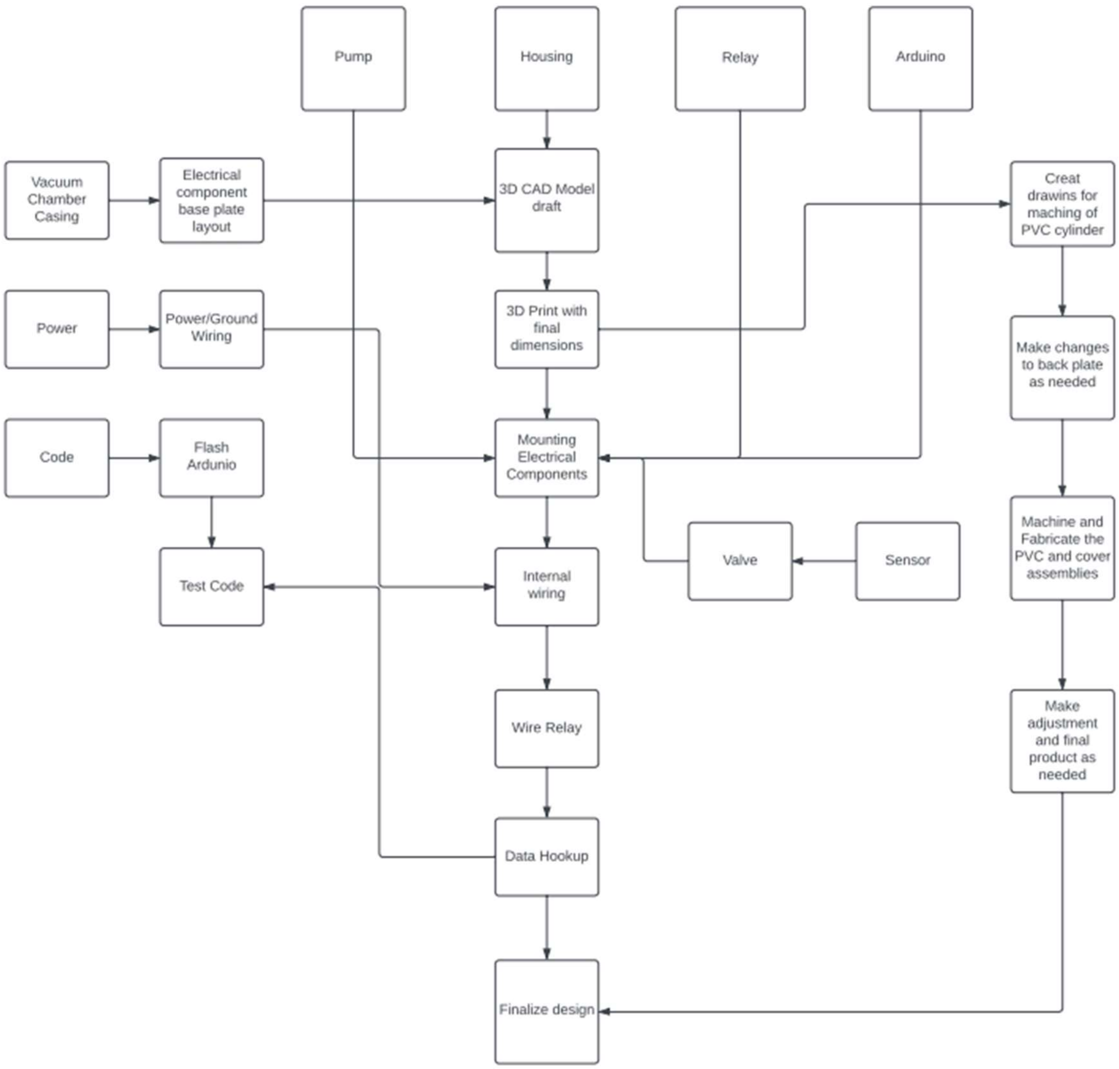


Figure 7: Block Diagram of System and Components

Mechanical Subsystem

The Mechanical Subsystem consists of the following:

1. PVC Pipe machining.
2. Closure Caps and latching clamps.
3. Electrical Components box.
4. Analog pressure gauge.
5. PVC pipe stand.

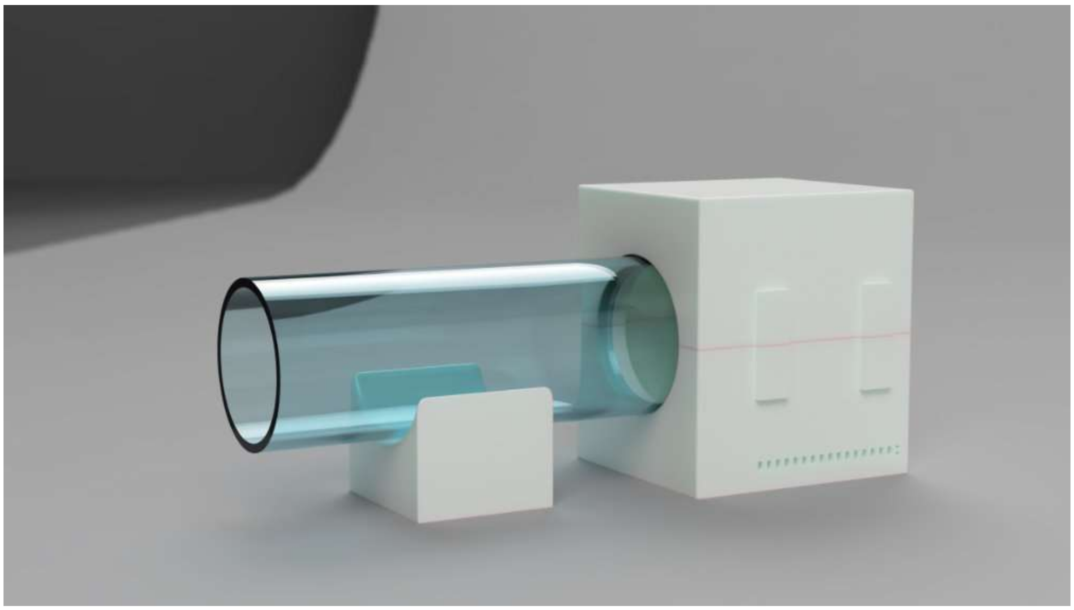


Figure 8: Mechanical Subsystem Design

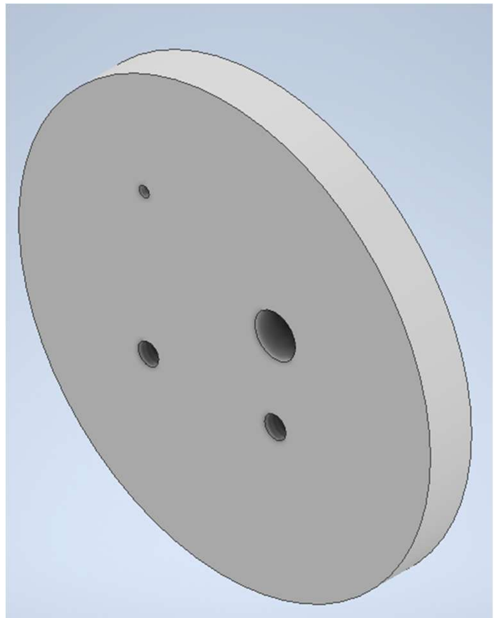


Figure 9: Back Cap 3D Print.

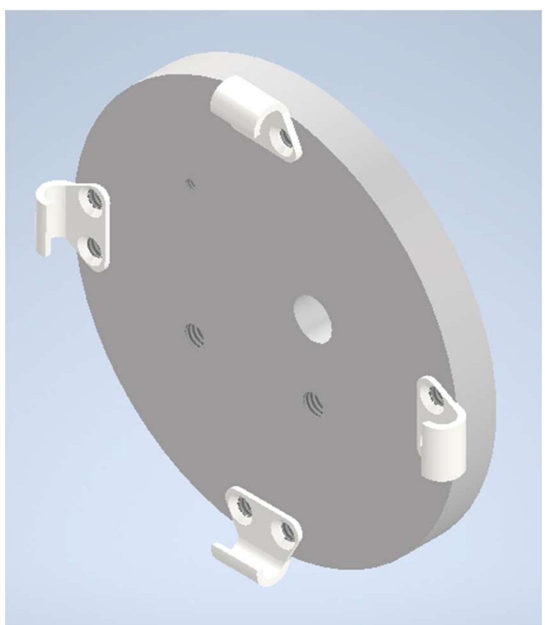


Figure 10: Front Cap 3D Print.

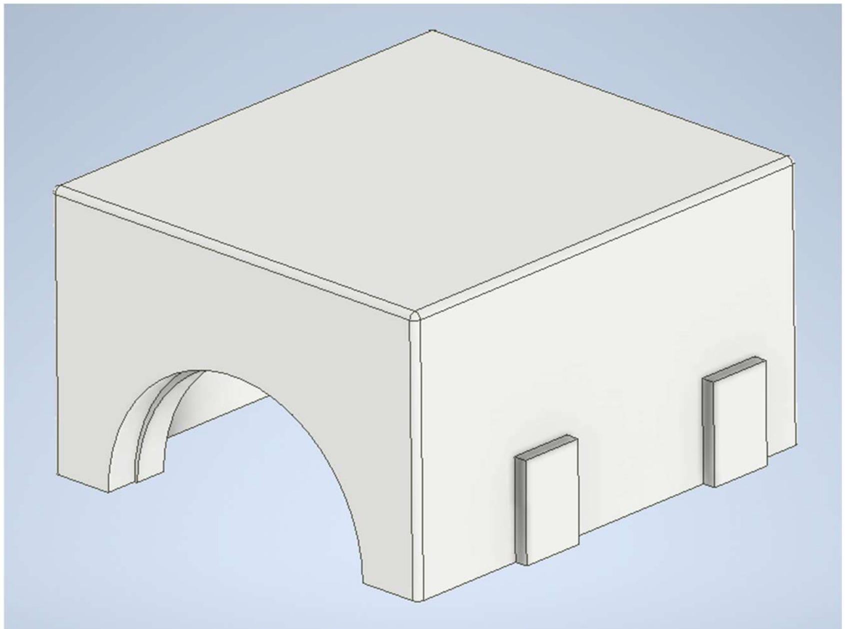


Figure 11: Control System Box Top

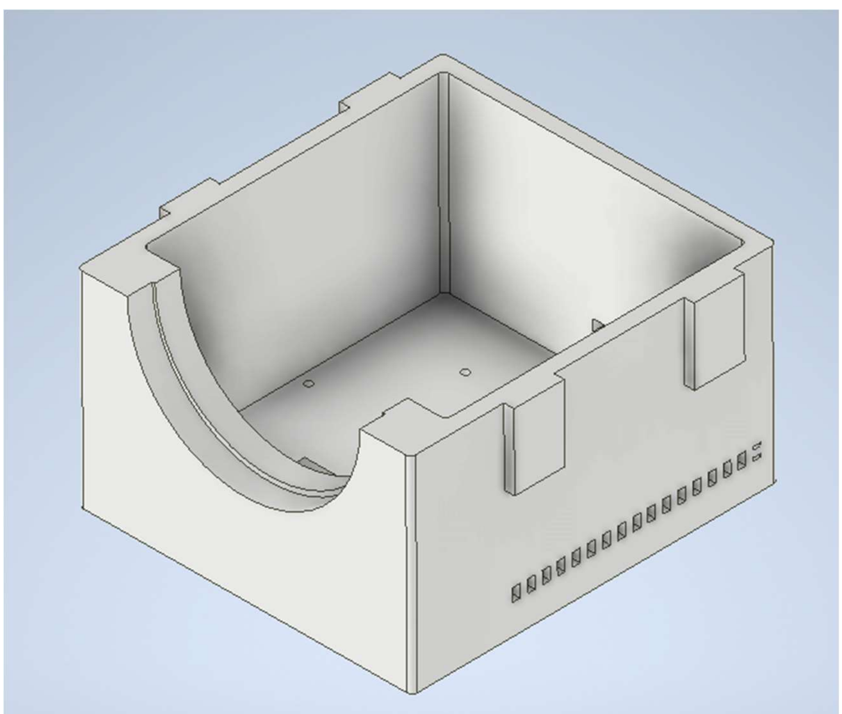


Figure 12: Control System Box Bottom

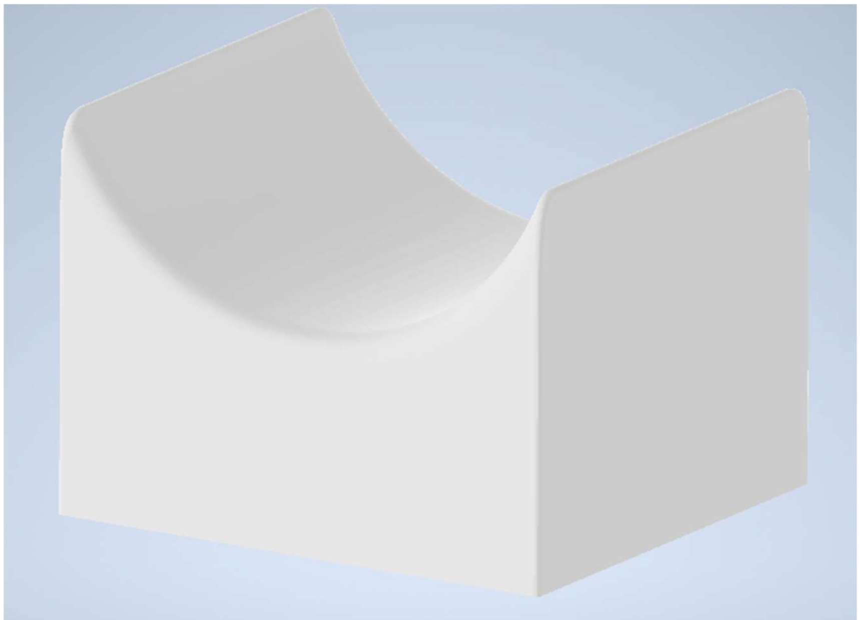


Figure 13: PVC Pipe Stand.

Figure 14: Analog Guage

Electrical/Electronic Subsystem

The electrical subsystem was further broken down into four additional subsystems: I/O, Control, Power Distribution, and Communication. The I/O subsystem consisted of the pressure sensor (which is the system's only non-user input), the vacuum pump, and the vent valve. These three components worked together to perform the physical operations required for operation. Two components made up the Control subsystem: an Arduino Uno R4 microprocessor and a two-relay 24 volt switching block. The Arduino Uno was responsible for monitoring inputs, commanding outputs, and performing all necessary calculations, display operations, and serial communication functions. The relay block allowed control of high amp, 12v components (the pump and vent valve) using low current 5-volt signals from the Arduino. The Power Distribution subsystem consisted of a 12v power harness, including a 120v-to-12v power inverter, and a 5v power loop from the Arduino Uno to the pressure sensor. The Communication subsystem consisted of two switched 5v signal lines from the Arduino to the relay block, a signal line from the pressure sensor to the Arduino, and a USB-C cable to handle serial communications between the Arduino and controlling computer. The interface between these additional subsystems is shown below in Figures 12 and 13.

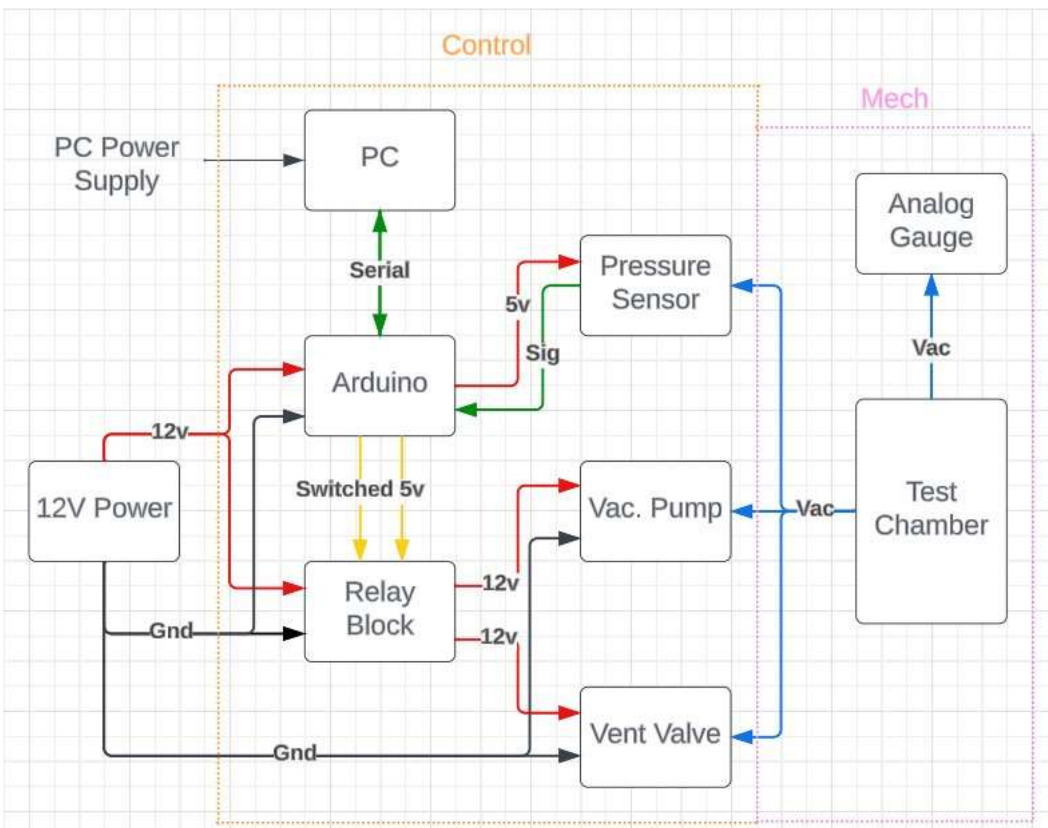


Figure 15: Control System Block Diagram

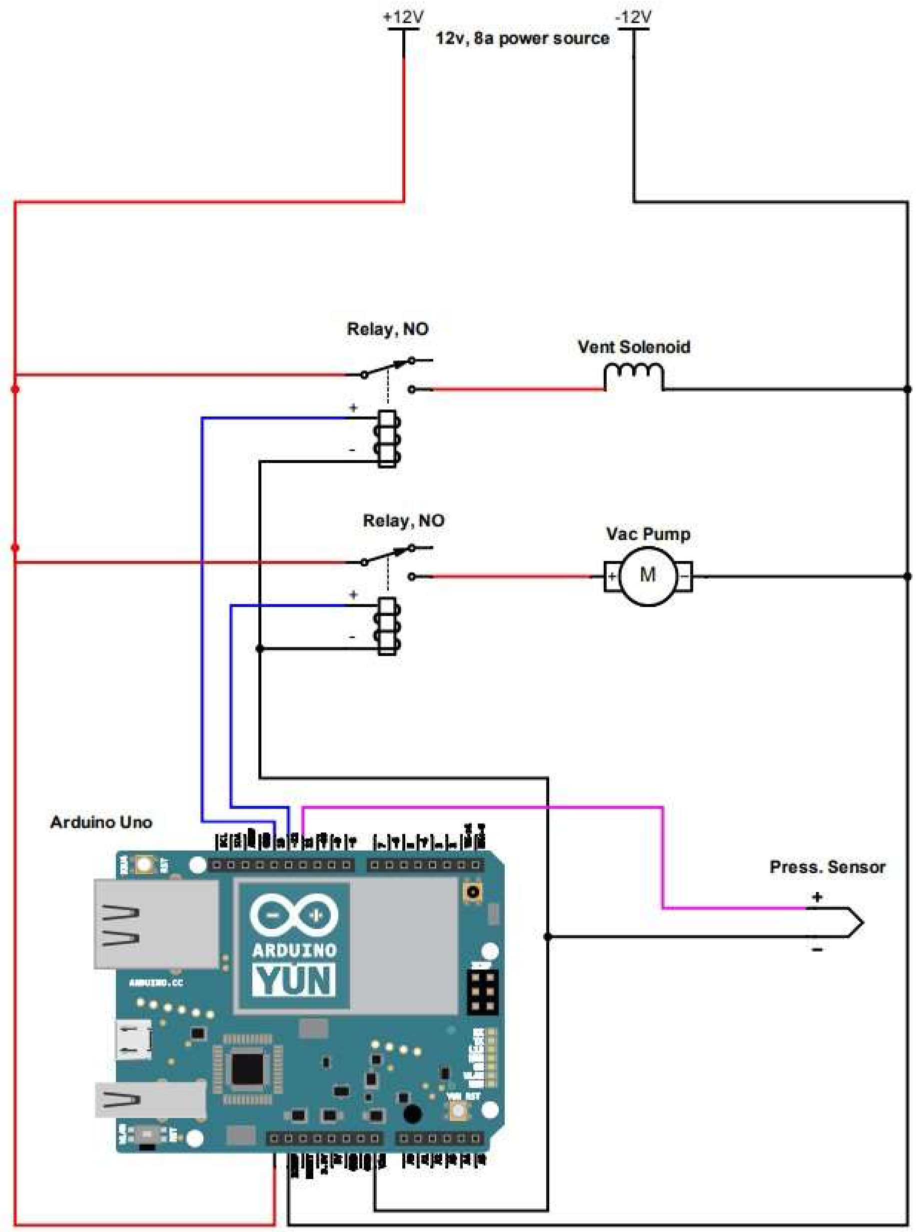
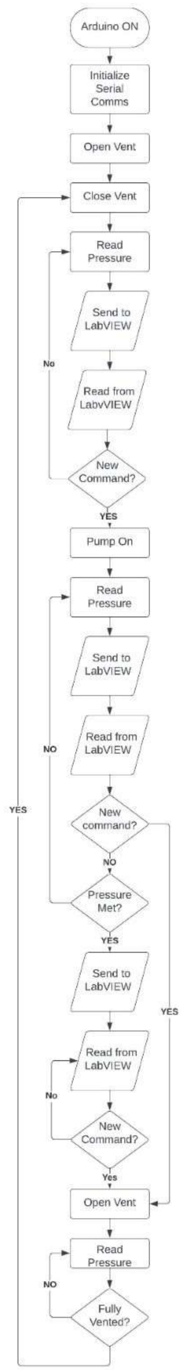


Figure 16: Schematic Diagram

Software/Programming Subsystem

The original intention was to utilize an Arduino Uno R4 to perform all I/O and calculation operations, and LabVIEW as a user interface to accept altitude simulation requests and display real-time chamber pressure. The functional block diagrams shown in table 7 show the intended interface between these code systems. During software development, it became apparent that a compatibility issue was causing very unstable communication between LabVIEW and the Arduino Uno R4. After researching the concern to resolve the issue, it was discovered that the Arduino Uno R4 uses slightly different firmware than the Arduino R3 and, because the Uno R4 is such a new microcontroller, LabVIEW does not currently offer an add-on to allow for stable interfacing with the R4. Therefore, the decision was made to de-scope the LabVIEW portion of the project and instead focus on optimizing an Arduino-only operation method that utilized the Arduino IDE’s built-in Serial Monitor for two-way communication. The code used for this operation can be found in Appendix B of this document.



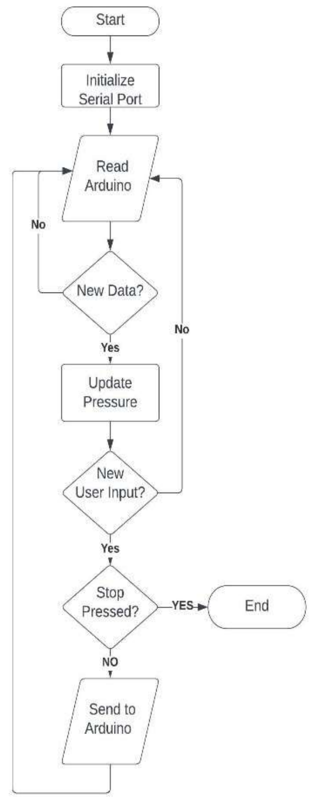


Table 8: Software Flow Chart

Design Features

Public health, Safety, and Welfare Features

Due to the low-pressure delta this system will be operating under, there are very few health and safety considerations to be accounted for in this design. The small risks posed by having a pressurized cylinder will be mitigated through in-depth stress analysis using initial hand calculation approximations and, later in the design, FEA analysis techniques.

Global, Societal, Environmental, and Economic Factors

The system will be designed to use as many off-the-shelf components as possible to minimize cost, both for initial purchase and for maintenance. This, coupled with the ease of use of the user interface will make the product more accessible to end users who may not previously have had access to this type of equipment

Fabrication and Integration

The fabrication and integration portion of the project consisted of four main processes: 3d printing, machining, electrical subsystem assembly, and subsystem integration. After the completion of the subsystem integration phase, preliminary testing was completed, and corrections were made as described below.

Five separate components were fabricated from ABS plastic using 3d printing. The upper and lower control housing clamshells, as well as the chamber support block, were printed using standard (unspecified) infill due to the lack of need for structural integrity of those parts. The end plates were both printed with 100% infill for two reasons. First, because these components are structural parts of the vacuum chamber, they needed to be both strong and airtight, which are two attributes best achieved by increasing density. Second, the back plate was tapped with one 1/8”-18 NPT hole and one ¼" -18 NPT hole, making a high density print necessary to resist the lateral stresses created by the tapered threads.

The main tube body was the only component to be machined. A top flat was milled onto the tube to allow for front plate clip mounting as well as to ease drilling and tapping a ¼"-18 NPT hole for the analog gauge and test port. The original intention was to use one hole for both components, having the gauge installed during normal operation and the test port installed when interfacing with a manometer for calibration. However, a communication breakdown with the machine shop led to the hole being tapped far too deep, which prevented either component from being sealed using thread tape. Due to time constraints, another hole was added to allow installation of the test port, and the analog gauge was permanently installed using silicone caulking.

Before electrical fabrication and subsystem assembly could be performed, the control housing clamshells and back plate needed to be printed. Once these parts were complete, the back plate was tapped, a thread insert was installed for the vent valve mount, and the back plate was glued into the lower clamshell with 5,000 PSI epoxy. Individual electrical components were then mounted into the lower clamshell, and necessary wiring was run through the box wall. Individual components were connected, using a combination of screw-down connectors, and soldering as necessary. Measurements were then taken for power and ground wires, and a main power harness was soldered, and heat shrunk together. The power harness was connected, and a hose was added to connect the pump to the chamber, completing the electrical subsystem.

The subsystem integration and final assembly portion of the fabrication stage was straight forward. First, the chamber body was attached to the back plate with 5,000 PSI epoxy for strength and silicone caulking for an airtight seal. The support block was also attached using epoxy at this time. Next, clamps were installed using epoxy to both the control housing clamshells and the front plate. After allowing time for adhesives to dry, the front plate O-ring was installed, all latches were adjusted to create proper tension, and the system was tested for its ability to hold vacuum using a hand pump. At this time, it was found that the seal between the front plate and chamber was not acceptable, the system as-is was incapable of achieving vacuum conditions. During the design phase, machining of the main body to include a 15-degree inward facing chamfer across the full thickness of the tube wall to allow for proper O-ring sealing was considered. However, due to the size of the machine shop's lathe, it would have been very difficult to safely hold the tube for this operation to be completed. Additionally, it was thought that the O-ring would be pliable enough to fully conform to surface imperfections from the factory finish, and so the decision was made not to attempt this machining operation. As the tube was already permanently mounted to the control housing, corrective machining was no longer possible, so the O-ring was removed, and the front plate was permanently mounted and sealed with silicone caulking to allow for all other systems to be tested. In hindsight, the rear plate should have been mounted separately and the sealing ability should have been tested prior to system integration to allow for further R&D into the sealing issues we encountered.

Comprehensive Performance Test

The comprehensive Performance Test was performed to demonstrate complete functionality, as well as to verify that no lingering issues were present. The first step of testing was to run the system through a series of test cycles, starting with a simulated altitude of 1000 meters and increasing in 1000-meter increments to a final altitude of 5000 meters. During each test cycle, the system was carefully monitored for any unsafe conditions, including but not limited to electrical system overheating, excessive pump run time, and abnormal noises from or visible distortion of the vacuum chamber. Once the simulated altitude was achieved, the pressure recorded by the system electronically was checked against the analog gauge to verify relative accuracy.

After this initial testing was completed, random altitudes between 500 and 5000 meters were input with minimal downtime between test cycles. This was repeated approximately 20 times. The purpose of this phase of testing was to verify the altitude calculation algorithm, as well as to test for system performance with a high frequency of test cycles. Again, the system was monitored for relative accuracy, using the analog gauge as a reference, and for overheating conditions. The system was then asked to simulate 5000 meters one more time, and repeatability of the system was verified using data points from the first phase of testing as a reference.

Most of our key performance and operation requirements were verified during testing. The goal of these tests was to verify the following requirements:

- System ability to simulate altitudes from 0-5000m.
- System ability to display chamber pressure during testing.
- System ability to respond to user commands; System ability to automatically calculate correct simulator pressure.
- System ability to complete test cycle in <5 mins; System accuracy +/- 3%.

Of these requirements, all passed except the system accuracy requirement, which was not fully analyzed. The method of verification of this requirement was to involve hooking the machine up to the school's wind tunnel manometer. However, it was discovered that the manometer had been damaged and required service to be usable, and this service had not been completed at the time of the comprehensive performance test.

Another performance issue that does not directly correspond to any stated requirement but is a major concern with regards to system usefulness was discovered during testing. After the vacuum pump is turned off, the system very slowly loses vacuum through the motor. Check valves were purchased in anticipation of this problem, but airflow was severely restricted during pumping operation with the check valves installed, which would likely result in premature pump failure. Because this problem was identified so late in the process, no corrective actions were taken, and further research and development would be required to find a check valve solution that would allow the system to maintain vacuum without creating unnecessary stress on the pump.

Schedule

The schedule for this project was driven by deliverable due dates. Sub-deliverable tasks were scheduled to account for cascading dependencies between various operations, and additional time was built into each deliverable block to account for unforeseen issues. This was particularly critical during the prototype and testing phases, where some issues did occur and had to be overcome. The schedule was planned and monitored using Microsoft Projects. A screenshot of this document can be found below in Figure 17.

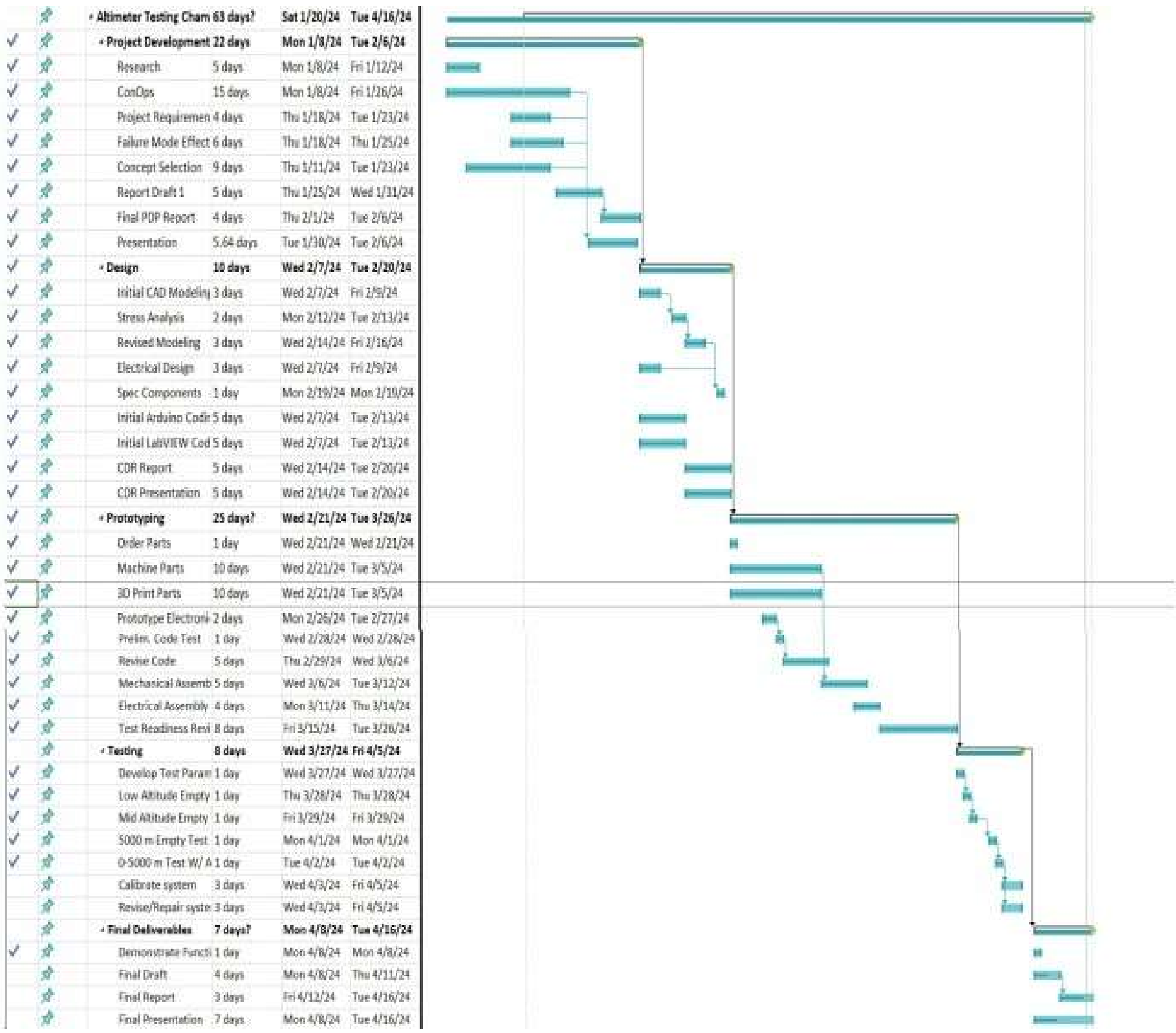


Figure 17: Program Manager Schedule

Project Budget

To start the budget for this project was $400 and our team did not exceed that. Not included in that is the price for an altimeter as we could not source one within the $400 range. Below is the BOM for the vacuum chamber:

Table 3: Vacuum Chamber BOM

QTY	Description	Status	Part No.	Use	Cost	$Total
1	HYUDYO DC Vacuum Pump (12V ,1.8A)	Received	B0B6VKDJK4	Pressure Regulation		$9.54
1	Check Valves - 5mm (10 Pack)	Received	Marhynchus8gn7yedswz-13	Anti-Pressure Backflow		$9.06
1	Canister Purge valve	Received	RB-D547	Vent pressure after cycle		$13.99
1	5V relay module	Received	WJ-00010	Electronically Switch Components on/off		$5.90
1	Arduino Uno R4	Received	ABX00080	Control System		$22.00
1	Power Supply - (12V, 8A)	Received	PW12-8A	Power System		$17.99
1	Screw Shield Prototype shield	Received	GR-US-168	Secure Wiring		$9.99
2	PVC Clear Rounds	Received	Not used in Production	End Caps	$16.00	$32.00
1	4 Inch Diameter Clear PVC	Received	PVC SCH40/SCH80	Chamber Body		$38.00
1	Adhesive	Received	4200130	End Cap Sealing		$10.18
1	Toggle clamps with 1-inch screws (6 pack)	Received	4001-6PK	Removable end caps		$6.99
1	Vacuum Gauge (1/4" NPT)	Received	GDD001	Reference Gauge		$12.96
1	3/8" barb x 1/4" NPT Barb Fitting	Received	3/8" Barb x 1/4" NPT Male Pipe	Manometer Adapter		$5.99
1	4" ID O-Ring (5 Pack)	Received	Not used in Production	End Cap Sealing		$12.64
1	5mm barb x 1/4" NPT Barb Fitting	Received	B0C3V3F21S	Adapter for Vac Pump		$7.99
					Total:	$215.22

Conclusion and Lessons Learned

Overall, this project had mixed success. While we were able to create a prototype that could accept user inputs and simulating altitude pressure conditions, there are outstanding issues related to both hardware and software that are the result of design and component selection choices made early in the development process.

Three issues were encountered that we were unable to correct during this project. The first involved communication instability between the Arduino Uno R4 and LabVIEW, which was to be used as a user interface. The Uno R4 is an updated version of the Uno R3, which has been proven to operate with only minor complications with LabVIEW. The R4 was chosen due to its updated communication port, which allowed for a cleaner and more stable connection to an external computer. However, minor changes in the R4’s firmware are not yet supported by LabVIEW, resulting in unstable, if any, communication. After complications arose, this incompatibility was discovered while searching for solutions on a LabVIEW forum. LabVIEW has very little official information regarding interfacing with Arduino products, as Arduino is considered a “hobbyist” tool and therefore receives minimal support from LabVIEW. This resulted in the compatibility issue not being identified during controller selection; while an attempt was made to verify compatibility, more time should have been spent confirming feasibility, which would have prevented this issue.

The second issue was related to the sealing surface between the chamber housing tube and the front plate. As discussed in the Fabrication section, a decision to not machine the mating surface was made to simplify manufacturing under the assumption that the O-ring would be pliable enough to conform to the surface irregularities in the factory surface and seal the chamber. This was not the case. A second mistake was made by not testing this seal before attaching the chamber tube to the control housing, which again was done to simplify assembly. The issue was not discovered until after the full system was permanently assembled, which made any corrective machining impossible. As a result, the front plate had to be permanently affixed to the chamber tube to allow for completion of testing of the rest of the system. In hindsight, a testing method should have been identified and implemented to verify that the system would be properly sealed without permanently attaching any other components to the tube, therefore allowing further research and development processes, including iterative corrective machining operations to be performed on the tube. Subsystem testing at this level was not considered due to the perceived simplicity of the system, which resulted in this oversight.

The final issue did not prevent successful testing of the system as it was intended to operate and would be easily remedied with enough time and resources to perform additional testing and is therefore the least serious problem with the project. When the vacuum pump is turned off, the system slowly bleeds up through the pump. This problem was anticipated during the initial round of purchasing, and check valves were selected to be installed in line with the pump to prevent the issue. During testing, however, it was found that these valves severely restrict airflow and create an unacceptable load on the vacuum pump. With some experimentation, a better quality, or possibly larger diameter, valve could easily be identified, and a tubing solution could be fabricated to allow its installation, thereby remedying the problem fully. However, because of the short amount of time available after the concern was identified, this problem was not able to be corrected.

Another issue that arose and had to be handled during this project was not related to performance or design. A team member dropped the course very abruptly without communication with the rest of the team or the professor. This resulted in several weeks of uncertainty, followed by a need for the team to quickly adapt to the situation and adjust role definitions to complete the project. This, along with the technical issues noted above, resulted in several important lessons being learned by the team members. Namely, we learned the importance of not making assumptions, testing sub-assemblies regardless of perceived simplicity, having several options on hand for inexpensive components if performance is unknown, and running projects in a decompartmentalized manner, where no one person is the only individual aware of an aspect of the project.

This project also introduced the team members to several new skills. Through research, we all gained a better understanding of the relationship between pressure and altitude, and how to incorporate temperature changes into these calculations. The 3d printed end caps gave us all a new appreciation for the abilities of modern 3d printed parts with respect to their ability to accept tapping and hold threads under torque. The required drawings also gave us experience with proper dimensioning and tolerancing methods. On a more individual level, lessons regarding machining practices and coding syntax were learned by the individuals involved in these aspects of the project.

From a management standpoint, the team worked well together, especially given the unanticipated difficulty of having a team member leave the project suddenly. Team member contributions were as follows:

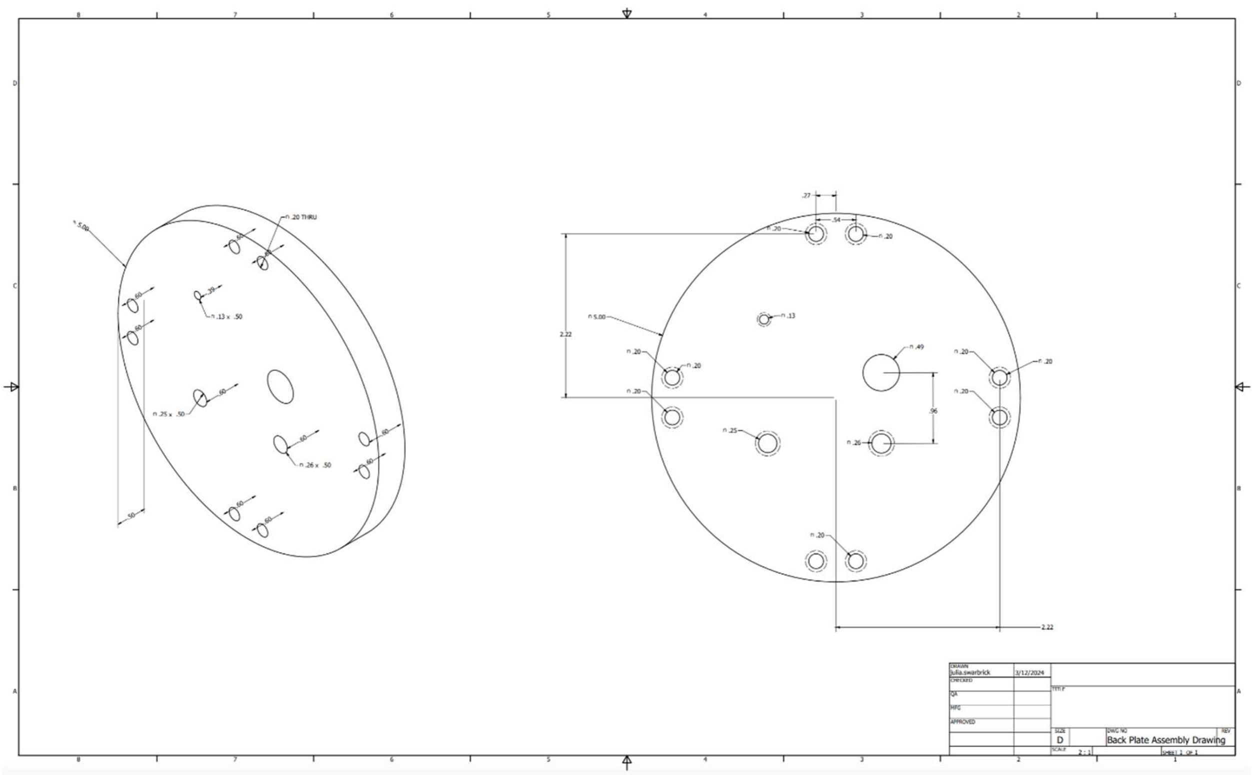
- D. Kimball- Team lead for first four weeks. Lead for electrical design, electrical assembly, mathematical analysis, and coding. Assistant for system integration assembly. Involved in all group decisions regarding mechanical components, assisted, when necessary, with mechanical aspects. Co-wrote reports and presentations, presented PDP and TRR. Shared team lead role during last four weeks.
- J. Swarbrick- Team lead after John for 4 weeks. Co-lead the mechanical design with John, helped with all adjustments as needed throughout the process. Helped Dave when he needed it after a loss of a teammate. Lead for reports and presentations, presented PDP and TRR. Shared team lead role during last four weeks. Assisted with system integration when it came time for assembly.
- J. Lear- Team lead from weeks 4 through 8. Co-lead the mechanical design along with Julia, manufacturing (chambers analogue pressure gauge inlet holes), CAD (control box, support leg), assisted in presentation construction as well as reports. Presented CDR.

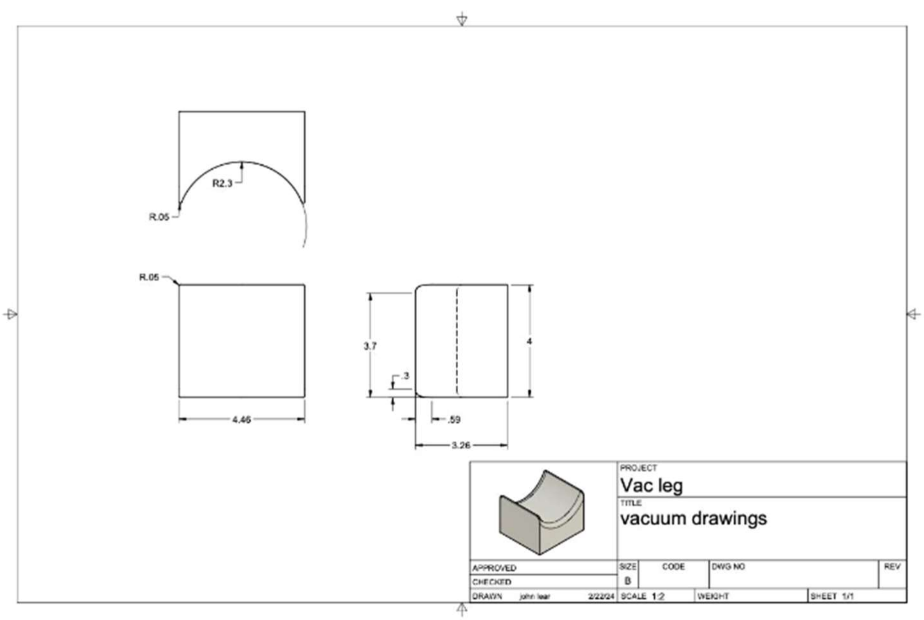
Appendices

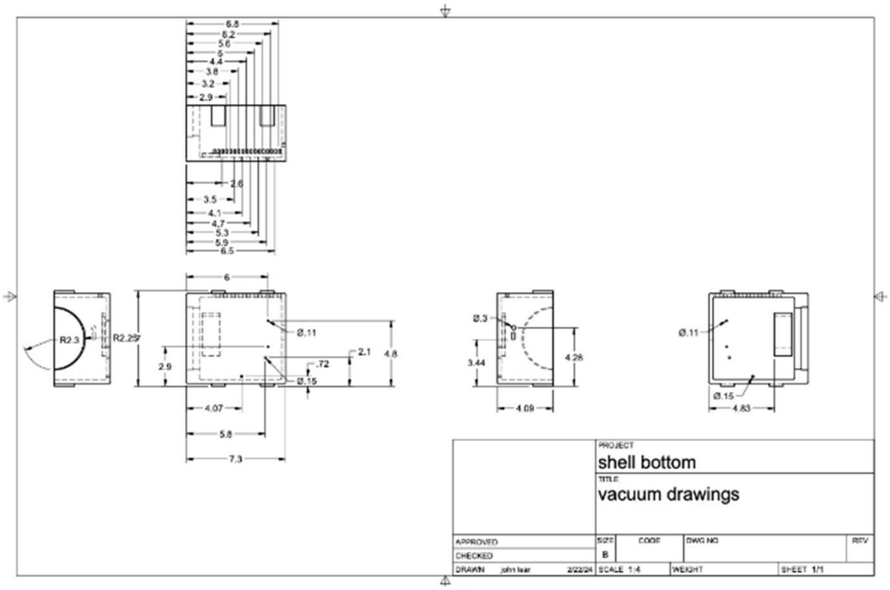
Appendix A. Verification Report

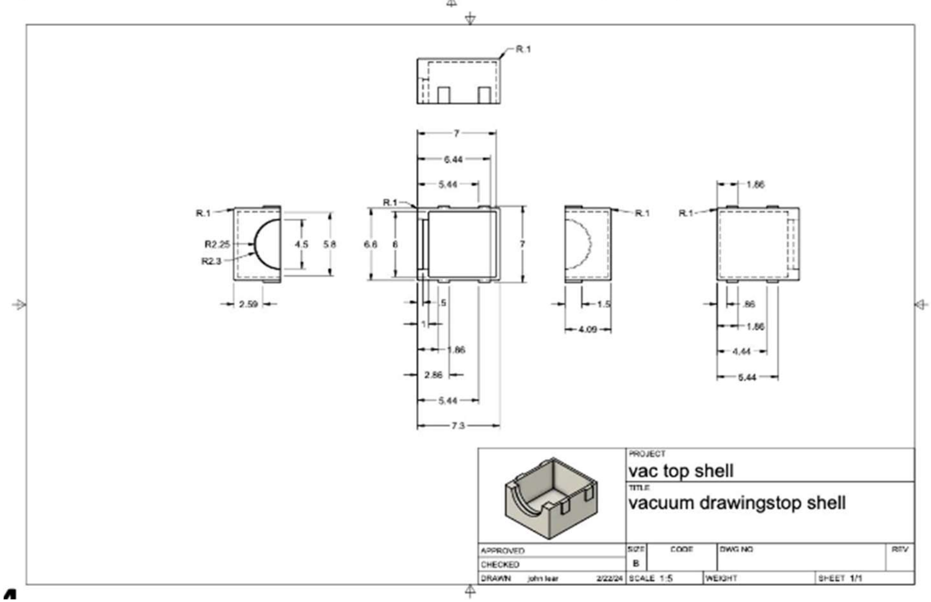
Appendix B. Acceptance Data Package List

1. Mechanical drawings, including assembly drawings with balloon callouts and part files in Inventor and STL files for 3D printed parts

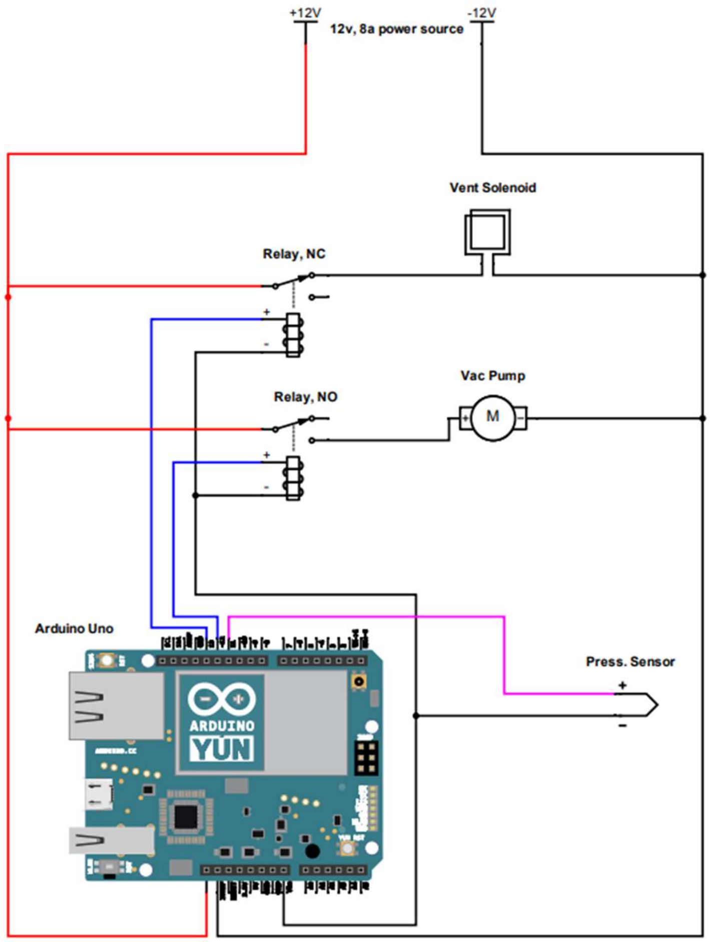








2. Electrical circuit diagrams and schematics



3. Software code listing with flow chart of operation

Arduino-only operation code listing:

```
/*                    Jr. Design Vacuum Chamber Arduino Code:
  Enter desired altitude. Arduino will convert to desired pressure, and operate
pump and vent valve
      as requested to achieve and maintain requested altitude specific
pressure.The system will then
      pause for 10 seconds before resetting and awaiting new desired altitude
input.

  Arduino only functionality: Revision by D. Kimball 4/15 for Team Vacuum Vibes

*/        //Variables/Pins:
  #define pumpCmd 12
```

```
  #define ventCmd 13
  #define pressVal A0
  String dsdAltStr;
  float dsdAlt = 0;
  int i = 0;
  float dsdPress = 0;
  float pressRaw = 0;
  float pressAct = 0;
  float initPress = 0;
  float tDsd = 0;
  float tAct = 295;
  float g = 9.81;
  float m = 0.02896;
  float r = 8.31432;
  float exponent = 0;
  float a = 0.3579;        //press sensor to kPa correction, y=ax+b
  float b = -39.075;       //press sensor to kPa correction, y=ax+b

int firstSensor = 0;       // first analog sensor
int secondSensor = 0;      // second analog sensor
int thirdSensor = 0;       // digital sensor
int inByte = 0;            // incoming serial byte
      //Subroutine to check chamber pressure and convert analog val to kPa
void ActPress(){
  pressRaw = analogRead(pressVal);                //read pressure sensor
  pressAct = a * pressRaw + b;                    //convert sensor reading to kPa
  //pressAct = 101;                               //FOR TESTING ONLY!!! COMMENT
OUT!!!
}
      //Subroutine to calculate pressure necessary for altitude simulation
void DsdPress(){
  tDsd = tAct - (6.5 * dsdAlt) / 1000;          //calculate temp at desired
altitude
  exponent = (-g* m * dsdAlt) / (r * tDsd);     //calculate exponential component
of press calc
  dsdPress = pressAct * exp(exponent);          //calculate pressure at desired
altitude
  //dsdPress = 55;                              //FOR TESTING ONLY!!! COMMENT
OUT!!!
}
      //subroutine to look for serial input
void establishContact() {
  while (Serial.available() <= 0) {
    Serial.println("Please enter desired altitude in Meters...");
```

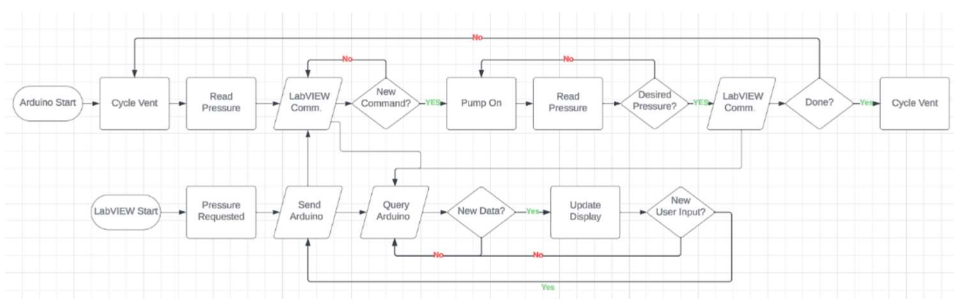
```
    delay(300);
  }
}

void setup() {
  // start serial port at 9600 bps:
  Serial.begin(9600);
  while (!Serial) {
    ;  // wait for serial port to connect. Needed for native USB port only
  }
  pinMode(pressVal, INPUT);
  pinMode(pumpCmd, OUTPUT);
  pinMode(ventCmd, OUTPUT);
}

void loop() {
  establishContact();                                //Check for attempted contact
              //Altitude Requested
  if (Serial.available() > 0) {
    dsdAltStr = Serial.readString();                 //read input string
    dsdAlt = dsdAltStr.toFloat();                    //convert string to float
    //dsdAlt = 50;                                   //FOR TESTING ONLY!!! COMMENT
OUT!!!
    ActPress();                                      //call subroutine to determine
pressure at current altitude
    DsdPress();                                      //call subroutine to determine
desired pressure
    Serial.print("Sea Level Pressure: ");            //display current and desired
pressures
    Serial.print("\t");
    Serial.println(pressAct);
    Serial.print("Desired Pressure: ");
    Serial.print("\t");
    Serial.println(dsdPress);
    delay(1000);                                     //1 sec delay
              //run if pressure above desired - pump on
    while(pressAct > 1.005 * dsdPress){
      digitalWrite(ventCmd, HIGH);                   //close vent valve
      digitalWrite(pumpCmd, HIGH);                   //turn on pump
      ActPress();                                    //call subroutine to determine
current pressure in chamber
      Serial.println(pressAct);                      //display current pressure
      Serial.print("Pump On Vent Closed");           //display system condition
    }
```

```
           //run if pressure below desired - vent open
    while(pressAct < 0.995 * dsdPress ){
      digitalWrite(ventCmd, LOW);                //open vent valve
      digitalWrite(pumpCmd, LOW);                //turn off pump
      ActPress();                                //call subroutine to determine
current pressure in chamber
      Serial.print("Pump Off Vent Open");        //display system condition
    }
           //run when pressure within desired range - conditions met
    while(1.005* dsdPress < pressAct < 0.995 * dsdPress){
      digitalWrite(ventCmd, HIGH);               //close vent valve
      digitalWrite(pumpCmd, LOW);                //turn off pump
      ActPress();                                //call subroutine to determine
current pressure in chamber
                                                 //Display current pressure and
system condition
      Serial.print("Pressure for Desired Altitude Met. Chamber pressure:");
      Serial.print("\t");
      Serial.println(pressAct);
      delay(10000);                              //10 sec delay
      digitalWrite(ventCmd, LOW);                //open vent valve
      delay(3000);                               //1 sec delay
      dsdPress = 0;                              //reset desired pressure
variable
      Serial.flush();                            //clear serial to restart
    }
  }
}
```

Code Flowchart (Arduino/LabVIEW Functionality, this communication was not achieved):



4. Component data sheets

Component data sheets for most components were not available, due to the components being inexpensively produced items purchased from a bulk online retailer. Arduino has published a 46-page specification document for the Arduino Uno R4, which was used for reference during this project. This document can be found at the following link:

https://docs.arduino.cc/resources/datasheets/ABX00087-datasheet.pdf

Note that the document above refers specifically to the Wireless version of the R4. The Uno R4 used in this project did not have wireless functionality, but all other information in this datasheet is relevant.

5. System performance calculations and analysis

Linear Interpolation for Pressure Sensor Analog-to-kPa Conversion:

Gauge Reading:	Conv. To kPa:	Raw Reading:
993.4	99.34	389
5	82.41	337.75
10	65.48	290
15	48.55	245
20	31.62	199