Cardiac Catheterization Testing Apparatus (CCTA)

Capstone Project Handoff Report

Boston Scientific Contacts:
Jiao Foss,
Rhodaba Ebady

By: Dabeer Abdul-Azeez (Engineering Physics),
Syed Saad Ali (Engineering Physics & BME),
Yousuf Araim (Mechanical & BME),
Owen Johnstone (Engineering Physics & BME),
Aly Pirbay (Mechanical)

Date: April 25th, 2025

Summary

Cardiac catheterization is a critical procedure for diagnosing and treating heart diseases [1]. However, current testing environments for catheter performance rely heavily on animal models or insufficient in-vitro setups [2]. Animal testing is not only time-consuming but also raises significant ethical concerns. Meanwhile, commercial testing systems are often prohibitively expensive or overly simplified, failing to meet the comprehensive requirements of proper catheter evaluation within a dynamic fluid environment. While laboratory-based solutions have been developed to address these issues at a lower cost and with greater anatomical accuracy, they lack precise control over pressure and flow characteristics, as well as a robust user interface for parameter monitoring, adjustment, and consistency [3]. This project aims to address these limitations by creating a cardiac catheterization testing apparatus that simulates physiological pressures and flow conditions in a controlled, reproducible environment.

The proposed system (the Cardiac Catheterization Testing Apparatus, CCTA) includes a modular flow loop capable of replicating dynamic pressure profiles within a range of (7 ± 2) to (300 ± 17) mmHg and flow rates (0.5 ± 0.1) to (4.7 ± 0.1) L/min [4]. Key components include embedded sensors for precise pressure and flow monitoring, a positive displacement pump to drive the fluid flow and a control box to tune the system to the desired flow and pressure parameters. To ensure durability and ease of use, the device features a leak-proof design for prolonged operation and a user-friendly interface for real-time configuration and monitoring. These innovations make the system adaptable for diverse testing scenarios while addressing the shortcomings of existing methods.

The impact of this design extends beyond technical improvements. For Boston Scientific, the apparatus provides a cost-effective solution to streamline the development and testing of devices like Farapulse [5]. Additionally, the system reduces reliance on animal testing by offering a reliable alternative, aligning with efforts to promote ethical testing practices. By setting a new standard for cardiac device testing, this innovation can drive broader industry improvements, benefiting healthcare systems globally.

Need Statement

Boston Scientific requires a cost-effective, reliable testing apparatus for their catheterized cardiac devices (e.g., Farapulse) that can accurately simulate a human circulatory environment, prioritizing stable and physiologically relevant pressure values and flow dynamics. This is because current in-house testing setups are inadequate, animal testing raises ethical concerns, and commercially available options are prohibitively expensive, time-consuming, or overly simplified, failing to meet the specific demands of their devices' performance evaluation.

Contents

1 System Design

1.1 System Design Overview

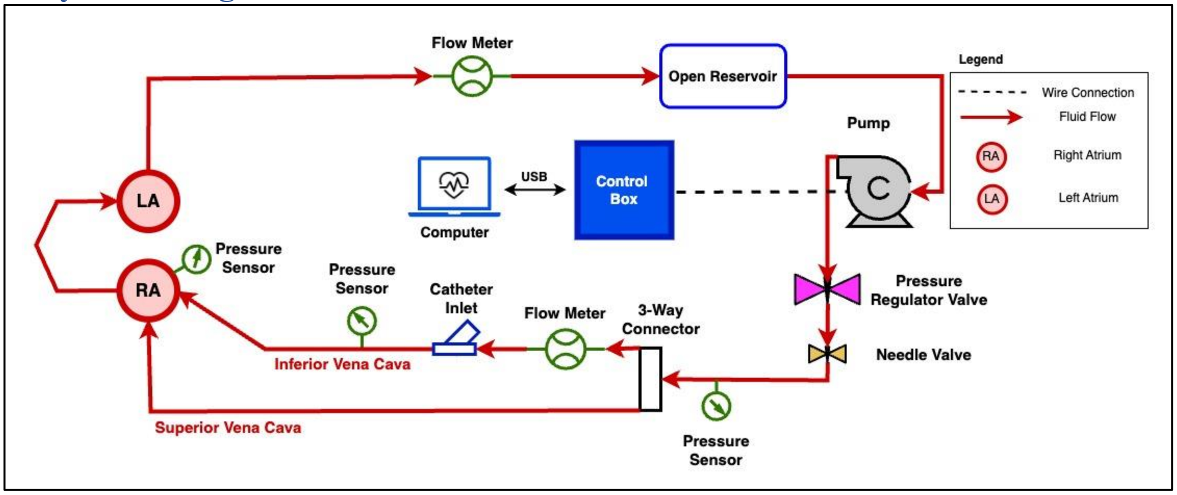

Figure 1: Full system schematic for the Cardiac Catheterization Testing Apparatus (CCTA)

Figure 1 shows the full system schematic for the Cardiac Catheterization Testing Apparatus (CCTA). The device is designed to simulate physiological flow and pressure conditions within the inferior vena cava (IVC) and heart atria, enabling safe, repeatable catheter testing outside of a clinical setting. At its core, the system consists of a flow loop powered by a pump, connected to tubing, sensors, and a 3D-printed heart model.

Fluid is pushed through the loop from an open reservoir, into the pump, and through a pressure regulator and needle valve to control pressure. The flow is then split and directed using various tubing and connectors past sensors and into the heart chambers before returning to the reservoir. The sensors measure pressure and flow at key locations, providing data to a control system housed in a custom electrical box that connects to a computer application. The heart model allows catheter insertion and pressure monitoring inside the simulated right and left atria.

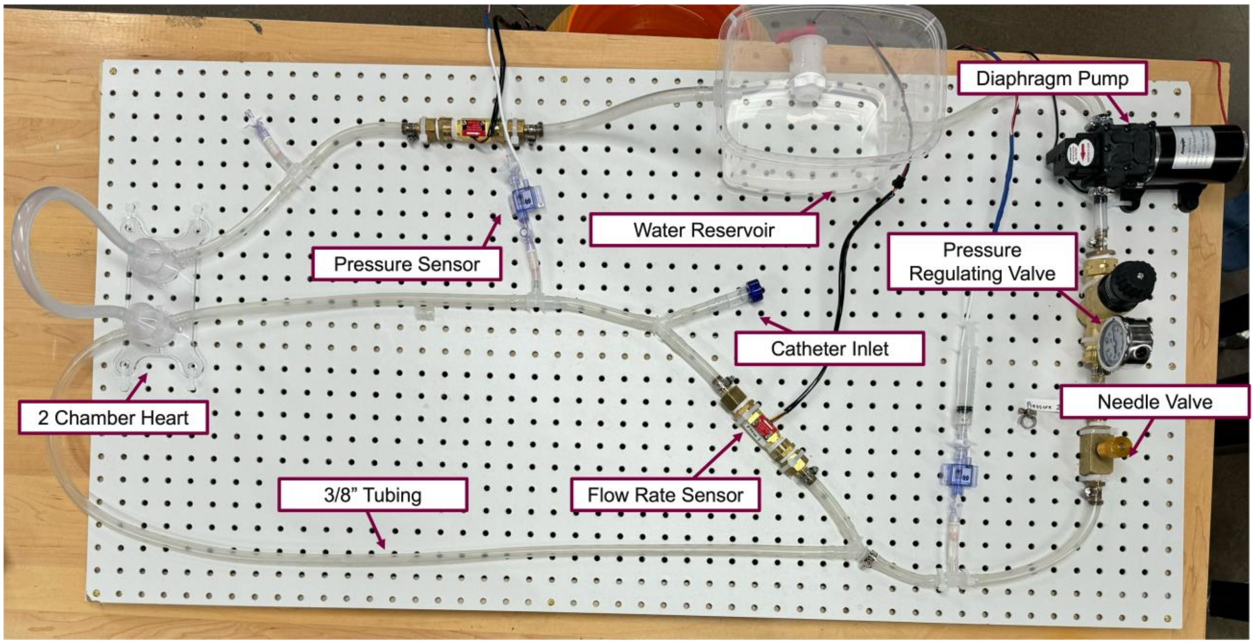

Figure 2: Labeled image of the physical prototype system mounted on a pegboard

Figure 2 shows the physical prototype of the system on a pegboard, with key components labeled and set up as outlined in Figure 1. The tubing layout and reservoir mimic venous return, while the modular connections make it easy to change test conditions or component locations.

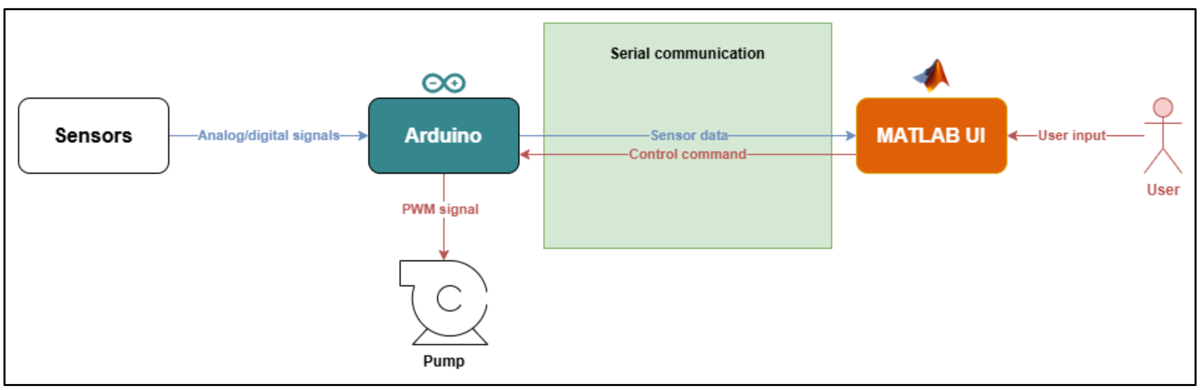

Figure 3: High-level data and control flow between user interface, Arduino, and sensors

Pump control is handled through an Arduino microcontroller and a MATLAB-based graphical user interface (GUI), as shown in Figure 3. The Arduino collects real-time system data (including flow and pressure values) and sends it to the GUI for data monitoring. In the UI, the user can view the data numerically and graphically and can perform actions such as calibrating the system, selecting different control modes, and setting pressure or flow targets. These actions all end up as control commands which the Arduino parses and converts into appropriate signals sent to the pump.

The rest of this section outlines each of the specific components in the setup in terms of their contents, function, mechanism, and rationale for inclusion into the system. Note: see the attached CCTA BOM.xlsx spreadsheet for a full list of all components used, along with pricing and other details.

1.2 Mechanical Components

This section details all physical and fluidic components of the system.

1.2.1 Bayite 12V Positive Displacement (PD) Pump

Contents: This is an off-the-shelf Bayite 12V Diaphragm Water Pump, commonly used for gardening, camping, and boating applications. It was sourced from Bayite.

Function: Drives fluid through the open-loop system and enables precise control of flow rate.

Mechanism: Operates using a voltage-controlled diaphragm that draws fluid in from one end and expels it from the other.

Rationale: A positive displacement pump was selected because it allows for accurate flow control independent of system pressure. This specific model is cost-effective (~$65), reliable, and commonly used in circulation simulation setups.

1.2.2 3/8" Tubing

Contents: The system primarily uses Continuous-Flex Soft Tygon PVC Tubing (3/8" ID), with rigid PVC tubing used in localized high-stress areas. Both types were sourced from McMaster-Carr.

Function: Serves as the primary fluid conduit for testing catheter flow and pressure response.

Mechanism: N/A

Rationale: 3/8" was chosen to support up to 16 Fr catheters (as requested) and to approximate adult femoral vein diameters [6]. Flexible tubing minimizes pressure losses and eases setup, while rigid tubing near the pump and regulators prevents deformation or failure under backpressure conditions. Flexible tubing also allows for low-pressure operation than rigid tubing. Lastly, the availability and compatibility of 3/8" tubing with standard connectors make it a practical choice.

1.2.3 Pressure Regulator Valve

Contents: A spring-loaded inline valve used to regulate and stabilize pressure in the system.

Function: Dampens pump outlet oscillations to ensure a more consistent system pressure.

Mechanism: Uses an internal spring to absorb fluid energy and reduce pressure spikes.

Rationale: Helps maintain steady physiological pressure conditions, protecting downstream sensors and components from instability. Also ensures consistent conditions across tests.

1.2.4 Needle Valve

Contents: 3/8" NPTF Female Needle Valve made of brass, sourced from McMaster-Carr.

Function: Provides precise pressure throttling to tune system pressure.

Mechanism: Adjusting the valve alters the flow cross-sectional area, regulating downstream pressure.

Rationale: Allows fine-grained pressure control when the pump overshoots target pressures, especially during high-flow conditions. Adjust fluid pressure via orifice restriction. Enables fine-tuned pressure control to match physiological setpoints when pump output exceeds requirements.

1.2.5 Connectors and Fittings

Contents: Includes barbed fittings, Y-connectors, BSPP-to-NPT adapters, bulkhead fittings, and stopcocks, primarily sourced from Home Depot, McMaster-Carr, and Qosina.

Function: Connect various tubing segments to sensors, the pump, and other system components.

Mechanism: N/A

Rationale: Standardized fittings allow for simplicity, modularity, secure connections, easy component swaps, and minimal leakage.

1.2.6 Heart Model (RA/LA)

Figure 4: Heart CAD Model.

Contents: A custom-designed 3D-printed resin chamber with barbed ports and a female Luer fitting (see CAD model in Figure 4). Manufactured in-house.

Function: Mimics internal heart chambers for pressure monitoring and catheter insertion during tests.

Mechanism: N/A

Rationale: Offers anatomical realism and internal pressure access. Resin printing ensures a leak-resistant and durable model compared to regular 3D printing, suitable for repeated testing.

1.2.7 Open Reservoir

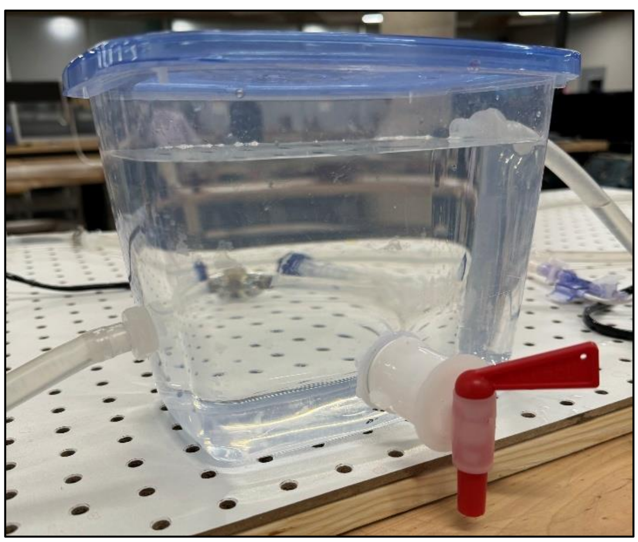

Figure 5: Open Fluid Reservoir with the various connectors.

Contents: Fabricated using a plastic container from Dollarama, with 3/8" barbed bulkhead fittings and a bottling bucket spigot (see Figure 5).

Function: Acts as a low-pressure, open-ended fluid collection and recirculation point.

Mechanism: N/A.

Rationale: Provides simple and accessible access to the fluid loop and allows for easy cleaning and fluid replacement. An open reservoir was chosen to facilitate system debubbling and prevent vacuum development. The barbed connectors were chosen for easy in-line integration and a spigot for easy draining and cleaning.

1.2.8 Pegboard

Contents: A wooden pegboard bought from Home Depot.

Function: Used as the base of the system. Components are zip-tied to the pegboard at various locations to build the circuit.

Mechanism: N/A

Rationale: A cheap and easy way to make the system modular. Lightweight as well, which helps with portability.

1.3 Electrical Components

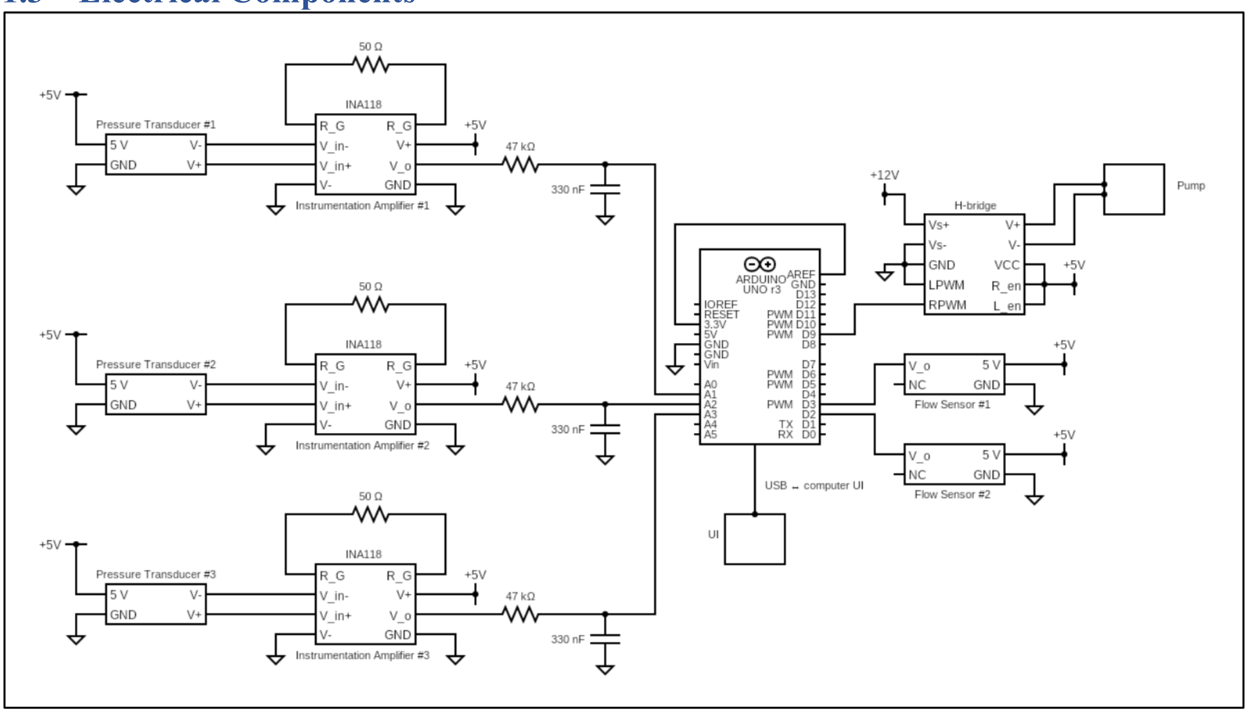

Figure 6: Full system circuit schematic showing sensor signal conditioning, Arduino interfacing, PWM pump control via H-bridge driver, and serial communication with the GUI.

This section includes all circuitry, signal processing, and power delivery elements. For reference, a full system circuit schematic is shown in Figure 6.

1.3.1 Pressure Sensors

Contents: Medical-grade pressure transducers sourced from Utah Medical, capable of measuring -50 to +300 mmHg.

Function: Monitor pressure values at key locations in the system for closed-loop control and data visualization.

Mechanism: Convert fluid pressure into an analog voltage signal readable by the Arduino (with the help of amplifier circuits, see Section 1.3.2). Pressure sensors were calibrated using a manometer (see Sections 2.7and 2.8).

Rationale: These sensors are industry-standard in medical applications and are compatible with catheter testing conditions.

1.3.2 Amplifier Circuits

Contents: INA118 instrumentation amplifier circuits, each configured with a 50 Ω gain resistor for a total gain of approximately 1000. These were built on solderable prototype boards (see Section 1.3.7). A low-pass RC filter was added at the amplifier outputs using a 47 kΩ resistor and 330 nF capacitor.

Function: Amplify low-voltage analog signals from the pressure sensors to match the Arduino's readable input range while filtering out high-frequency noise.

Mechanism: The INA118 supports differential input, offering high common-mode rejection and robust impedance matching. The added RC filter creates a cutoff frequency of approximately 10 Hz, enabling the amplifier to preserve physiological signals while suppressing noise.

Rationale: Early versions used standard op-amps, which caused signal saturation due to poor impedance matching. Transitioning to INA118 instrumentation amplifiers provided better signal integrity, precision gain control, and noise immunity. This design choice ensured accurate and stable pressure readings, essential for reliable feedback control. Amplification is necessary to improve resolution and accuracy of sensor readings within the Arduino's 10-bit ADC limits. Prototype boards were used for easy development and tuning during calibration.

1.3.3 Flow Meters

Contents: GREDIA 3/8” quick-connect turbine flow sensors sourced from Amazon. Plastic inline flow meters.

Function: Measure volumetric flow rates and feed this information to the control system.

Mechanism: Use a spinning rotor with embedded magnets to generate pulses proportional to flow rate. Sensors were calibrated by measuring a known volume over time (see Section 2.6).

Rationale: Compact, reliable, and sufficient for expected flow ranges (1–25 L/min). Economical and widely used in prototyping.

1.3.4 Control Box

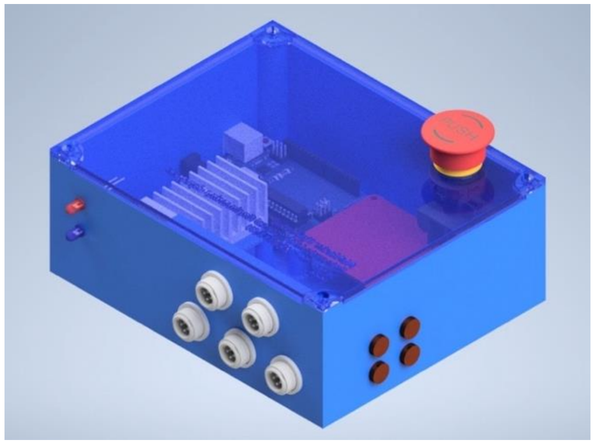

Figure 7: Rendered Control Box CAD Assembly.

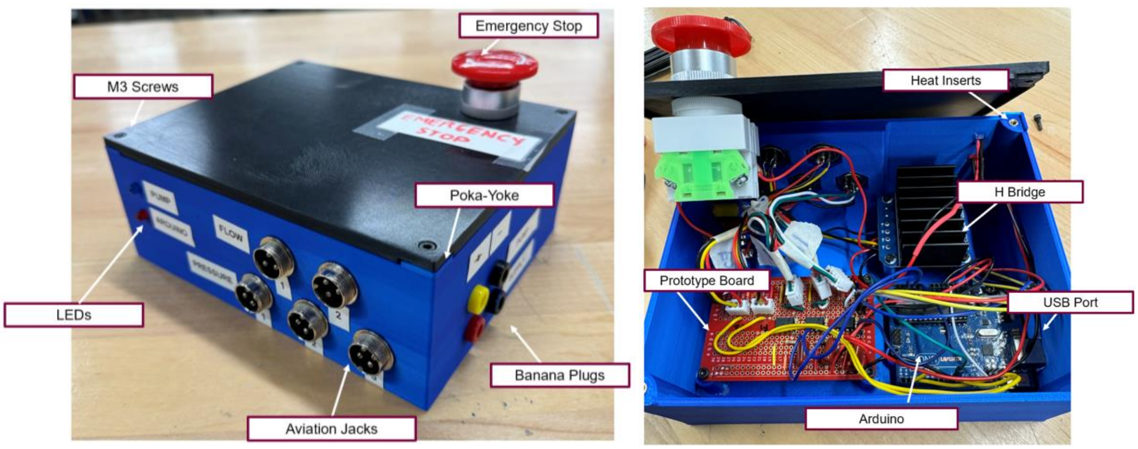

Figure 8: Control Box Physical Prototype.

Contents: A plastic 3D enclosure (see Figure 7 and Figure 8) housing the Arduino Uno, motor driver, custom prototype board, and connectors (aviation, banana plugs, USB). Includes LED indicators and aviation connectors for sensor ports.

Function: Serves as the central electrical hub for signal routing, power management, and controller integration.

Mechanism: N/A

Rationale: Designed in-house for compactness, durability, and ease of maintenance. Waterproof connectors minimize electrical risks and enable clean wiring. Encloses Arduino, H-bridge motor driver, and circuitry effectively. Facilitates power delivery, sensor interfacing, and safe operation. Waterproof aviation connectors, LED indicators, and secure assembly features are included for durability and usability.

1.3.5 Arduino Uno Microcontroller

Contents: Off-the-shelf Arduino Uno board sourced from Arduino.

Function: Performs real-time data acquisition, sensor reading, and control logic processing.

Mechanism: Reads analog and digital signals from sensors, executes control routines, and outputs PWM to motor driver.

Rationale: Widely supported and easily programmable platform suitable for rapid prototyping and flexible development. Handles real-time data acquisition and control logic. Parses mode commands and executes appropriate control strategies.

1.3.6 Motor Driver (H-Bridge)

Contents: Packaged H-bridge motor driver sourced from Reland Sun (Amazon).

Function: Interfaces between the Arduino and pump, allowing variable-speed motor control.

Mechanism: Converts PWM signals into regulated output voltage and current to drive the pump motor.

Rationale: Essential for precise speed control in low-voltage DC motor systems. Selected for electrical compatibility (since the Bayite pump has only two wires to control it, and no built-in control system), the ability to control it easily through software (since the Arduino can natively output PWM signals), and cost-effectiveness.

1.3.7 Custom Prototype Board

Contents: A hand-wired prototype breadboard, similar to a prototype board but with built-in power rails and rows of interconnected pins, more akin to a breadboard.

Function: Used to process analog sensor signals before sending them to the Arduino. Also acts as a pass-through for flow signals before they are re-routed to the Arduino.

Mechanism: N/A

Rationale: Chosen instead of a custom PCB to allow for rapid modifications and easier troubleshooting during development. This flexible setup simplifies future iterations and repairs.

Optimization: A prototype breadboard was chosen over a regular prototype board to facilitate power transfer through the system via built-in rails and rows, though further development revealed it makes the circuitry bulkier. If further development is required, it would necessitate a second prototype breadboard, or that the current circuitry be compacted onto a regular prototype board.

1.3.8 Power Supply

Contents: DC Power Supply, Variable 30V 10A, sourced from Sky TopPower.

Function: Provides regulated 12V DC power to the system.

Mechanism: Converts AC wall power into adjustable DC output suitable for system electronics.

Rationale: Delivers sufficient current for all components and allows future scalability or test condition changes.

1.4 Software & Control System Components

This section describes the software architecture and control logic.

1.4.1 MATLAB GUI

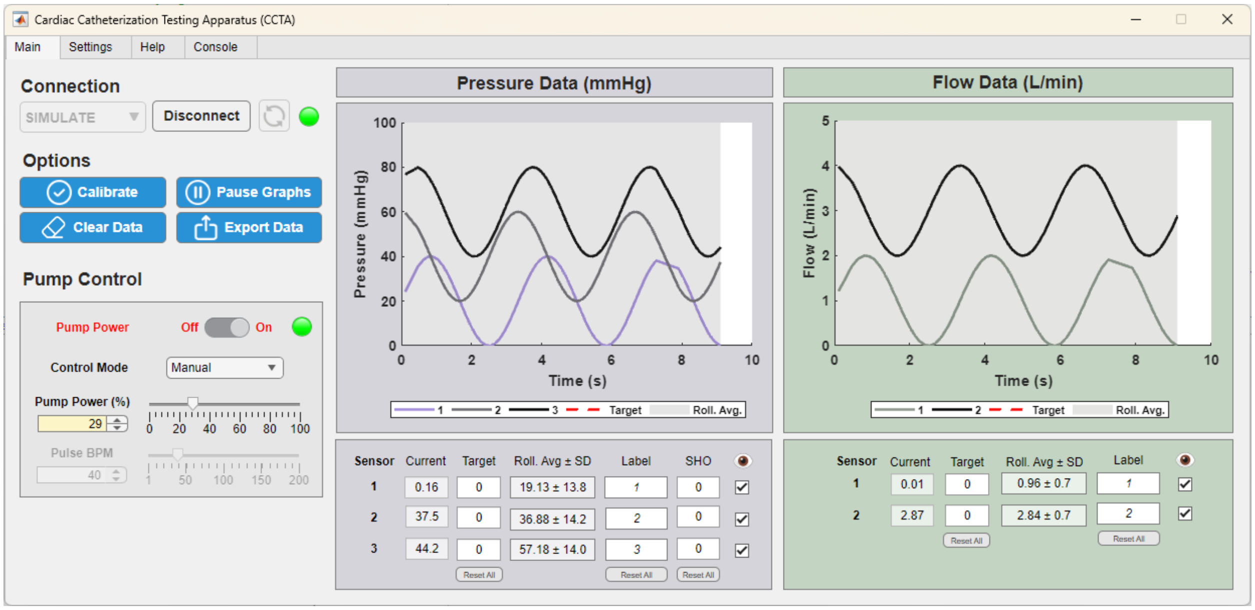

Figure 9: Screenshot of CCTA MATLAB GUI displaying simulated sinusoidal values under manual pump control.

Contents: Custom-developed user interface built using MATLAB App Designer.

Function: Allows users to configure system parameters, select operating modes, and monitor sensor data in real time. The GUI supports calibration of the pressure sensors to account for static head offset (SHO), displays both raw and rolling average data numerically and graphically, and computes real-time measurement uncertainty. Displays serial messages on a built-in console. Also, users can export collected data to a spreadsheet for post-processing and documentation. The interface enables full pump control across three modes (see Section 1.4.3).

Mechanism: Runs a timed loop (`runMainLoop()`) that communicates with the Arduino via serial commands and plots sensor values. There are also specific functions tied to various buttons and inputs across the GUI (see the code files for more specific documentation).

Rationale: MATLAB offers built-in plotting tools, numerical processing, serial communication architecture, and an App Designer which makes it an ideal platform for rapid GUI development.

1.4.2 Arduino Firmware

Contents: Embedded code running on the Arduino Uno.

Function: Executes core control algorithms, controls the pump through a pulse-width modulated (PWM) signal, and handles communication with the GUI.

Mechanism: Uses the `loop()` function to continuously read sensor inputs, compute outputs, and send back data.

Rationale: Integrates seamlessly with the Arduino Uno microcontroller, and contains built-in libraries for serial communication, digital/analog signal reading/writing. Extensive documentation online.

Optimization: The PID control loop and pulsatile flow control were implemented on the Arduino for speed -- originally the control logic was in the GUI as well, but generally the GUI has been shifted to focusing on data visualization to free up the Arduino to take on more control logic.

1.4.3 Operating Modes

- **Manual Mode**: Direct voltage control of pump speed via GUI.
- **Automatic Mode**: PID controller adjusts pump output to maintain target pressure or flow.
- **Pulsatile Mode**: Generates a sinusoidal output to simulate cardiac cycles.

1.4.4 PID Control

Contents: A proportional-integral-derivative (PID) control algorithm implemented in the Arduino firmware, with user-accessible parameters via the MATLAB GUI.

Function: Maintains target pressure or flow levels in the system by dynamically adjusting pump output based on feedback from sensors.

Mechanism: The user sets a desired setpoint in the GUI (either pressure or flow), and the Arduino receives this target along with the current measured value and PID coefficients (Kp, Ki, Kd). It computes the error between the setpoint and measured value and calculates a correction value, which is sent as a PWM signal to the pump.

Rationale: PID control is widely used in process control systems for its ability to respond proportionally to error, account for accumulated discrepancies over time, and anticipate future trends. It is well-suited for the catheter testing apparatus because it offers smooth and precise adjustments that avoid sudden changes in flow or pressure—critical for simulating physiological conditions. Including adjustable PID coefficients in the GUI gives users the flexibility to tune the response for different catheter test setups.

2 System Verification

Please see the CCTA Verification Tests folder for the full set of test reports and results which elaborate on the test procedures for each test and other details. Below are summaries of the tests and results.

2.1 System Capabilities Test (Mechanical)

Please refer to the "System Capabilities Test" folder under "CCTA Verification Tests".

2.1.1 Objective

The objective of this test was to define the system limits in terms of both pressure and flow. Additionally, this test was completed to verify whether we can achieve the desired system outputs defined by Boston Scientific. Table 1 below shows the target pressure and flow values our team was trying to achieve.

Table 1: Desired System Outputs set by Boston Scientific.

Location	Flow Rate (L/min)	Pressure (mmHg)
Inferior Vena Cava	1.02	5
Common Femoral Vein	0.91	10
Right Atrium - IVC	3.75	3
Right Atrium - SVC	2.42	8
Left Atrium	Complex	8

2.1.2 Results Summary

The system's minimum and maximum pressures and flow rates are summarized in Table 2 below. The system can generate pressures as low as 3.02 mmHg (± 0.3 mmHg accuracy) and as high as 123.66 mmHg (± 0.5 mmHg accuracy), and flow rates from 0.02 L/min up to 4.67 L/min. Additionally, the system meets all targets highlighted in green in Table 1 above. Those highlighted in orange were difficult to verify. Regarding pressures, only a single sensor is connected to the right atrial wall, so only the average atrial pressure could be measured. Regarding flow, the pump's maximum flow rate is 4.7 L/min; given the current setup, the RA–IVC and RA–SVC target flows cannot be achieved simultaneously because the fluid splits into two lines before combining in the atrium (and 3.75 L/min + 2.42 L/min ≠ 4.7 L/min). However, since these target values remain within the system limits shown in Table X below, they are achievable, just **not simultaneously**.

Table 2: CCTA System Limits.

	Flow Rate (L/min) (±SD)	Pressure (mmHg) (±SD)	Settings
Minimum	0.02 (± 0.0)	3.02 (± 0.3)	• Needle Valve **Red Line**. • Pressure Regulator Valve at **0 PSI**. • Pump Power at **30%.**
Maximum	4.67 (± 0.0)	123.66 (± 0.5)	• Needle Valve **Fully Open**. • Pressure Regulator Valve at **0 PSI**. • Pump Power at **100%.**

2.2 Flow Profile Test (Mechanical)

Please refer to the "Flow Profile Test" folder under "CCTA Verification Tests".

2.2.1 Objective

The objective of this test was to determine whether the CCTA can achieve laminar flow in the IVC and Turbulent flow in the Right Atrium. The requirements set by BSC are as follows:

1. Laminar in the IVC.
2. Turbulent in the RA.

2.2.2 Results Summary

Overall, the test was successful in characterizing the flow behavior within the CCTA. Results indicate that the flow within the right atrium (RA) is clearly turbulent, as expected due to the complex geometry and the direct inflow from both the inferior vena cava (IVC) and superior vena cava (SVC). In contrast, the flow in the IVC did not exhibit characteristics that are definitively turbulent or laminar. Instead, it appeared transitional, with initial laminar behavior near the plastic fitting and the onset of turbulence at the junction with the flexible tubing.

To achieve fully developed laminar flow in the IVC, a fluid with higher dynamic viscosity should be considered. Glycerol-water mixtures, commonly used in cardiovascular flow modeling, provide an effective solution due to their significantly higher viscosity compared to water or saline. For instance, a 60% glycerol solution yields a dynamic viscosity of approximately 0.013 Pa·s at 25 ℃ [7]. Using this fluid in 3/8" tubing at a flow rate of 1.02 L/min would result in a Reynolds number of approximately 178, which is well below the critical threshold ($Re < 2000$), confirming laminar flow conditions.

Using a fluid with higher viscosity not only enables better control of the flow regime but also enhances the physiological relevance of the test setup, especially when modeling venous return where laminar flow is typically expected. Future testing should incorporate glycerol-based fluids to ensure consistent and predictable flow profiles within the IVC and throughout the loop.

2.3 Durability Test (Mechanical)

This test does not have a separate test folder. The results will be discussed entirely in this section.

2.3.1 Objective

The objective of this test was to evaluate the CCTA's long-term mechanical robustness by running the system continuously for one hour under variable operating conditions. Our first goal is to confirm leak-free operation and monitor the diaphragm pump for any signs of overheating. We then assess the control-box electronics for electrical safety, checking for shorts, loose connections, or thermal issues. Finally, we inspect critical hardware - flexible tubing and brass fittings - for early wear or fatigue. Together, these tests establish whether the apparatus can sustain reliable performance over extended use.

2.3.2 Results Summary

Standard tube fittings and connectors were installed to achieve reliable seals between all components. Maintaining these seals is critical: any leakage alters the system's pressure and flow readings, rendering test results invalid.

The greatest risk for fluid leakage lies in the tubing between the pump and the pressure-regulator valve. When the needle valve is throttled, back-pressure in this segment can climb to 50 PSI (Figure 10). Although a more rigid, high-pressure–rated tubing was selected for this section, complete closure of the needle valve can still induce catastrophic failure and a substantial leak. Operators should therefore avoid fully closing the needle valve and remain vigilant for pressure spikes.

Figure 10: Image showing the location with the highest risk of failure and potential leakage.

Under both the previously defined maximum and minimum pressure conditions, all other tubing connections remained leak-free, confirming that the selected clamps and fittings are both appropriate and effective.

Periodic inspections were performed every few weeks during the testing phase to evaluate long-term wear on system components. Pressure and flow sensors were recalibrated multiple times to maintain accuracy. No significant degradation was observed in the prototype's main components; however, two issues were noted: the soft tubing may shear if clamped too tightly around the barbed fittings, and some discoloration appeared on the brass fittings (Figure 11).

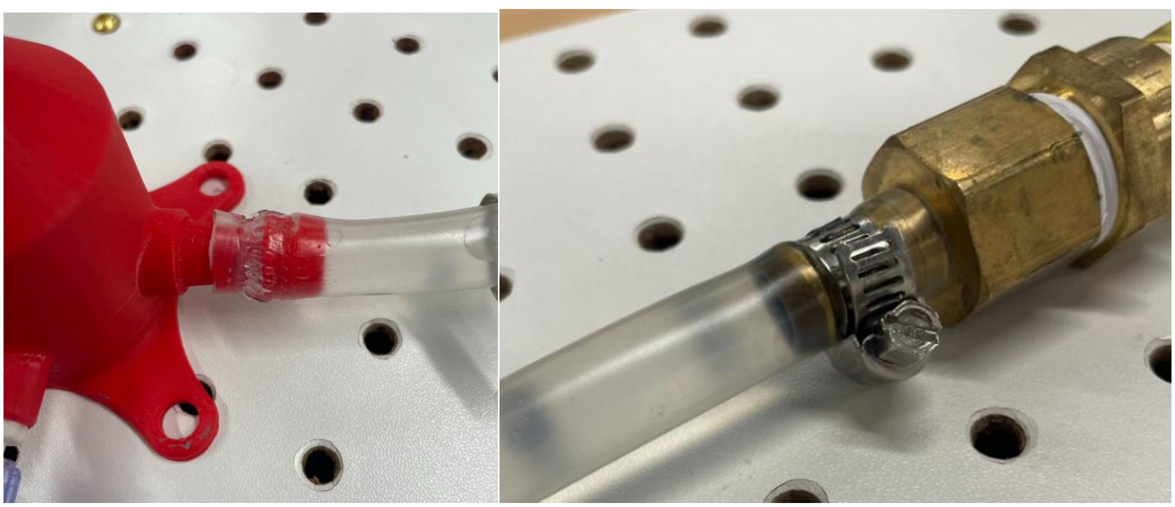

Figure 11: Wear & Tear in Soft Tubing (Left). Discolouration in Brass Barbs (Right).

2.4 Electrical Safety Test (Mechanical)

This test does not have a separate test folder. The results will be discussed entirely in this section.

2.4.1 Objective

The objective of this test was to verify the control-box enclosure's electrical safety and resilience by:

- Validating its ability to prevent water ingress
- Demonstrating reliable operation under simulated voltage surges and unexpected power loss
- Confirming emergency-stop functionality
- Ensuring durable, secure wire connections under light mechanical stress.

2.4.2 Results Summary

A water splash test was conducted, and no electrical failures occurred (Figure 12). However, the control box still contains gaps - specifically at the top-lid seam and the USB cutout - that could allow water ingress. Future prototypes should feature a redesigned enclosure to achieve full waterproofing. Further recommended improvements are discussed later in this report.

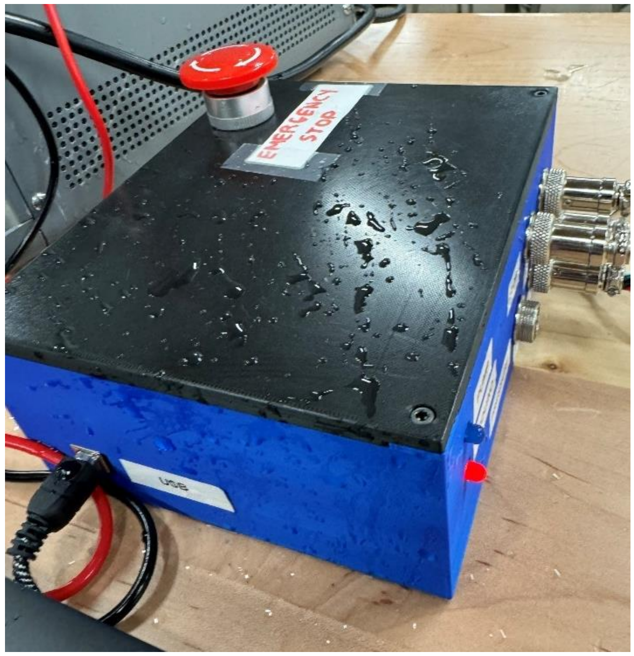

Figure 12: Water Splash Test.

Power surges were simulated using the available power supply; however, a supply with a broader voltage range (up to 30 V) is recommended for future testing. An emergency-stop button was added as a safety feature and successfully cut power to the pump when pressed. To simulate unexpected power loss, the system was unplugged during operation and safely recovered with no issues. Light tugging tests on various wire connections confirmed that the connectors and solder joints remained robust. Although not part of the prototype, GFCI outlets and surge protectors were recommended for the stakeholder's lab setup.

2.5 Thermal Safety Test

This test does not have a separate test folder. The results will be discussed entirely in this section.

2.5.1 Objective

The objective of this test was to ensure that the diaphragm pump and adjacent components never exceed safe surface-temperature limits during continuous operation. A calibrated infrared (IR) sensor was mounted on a fixed stand and aimed at the pump housing. Surface temperatures were logged after two hours of continuous operation under variable testing conditions. Measured values were compared against the maximum allowable temperatures specified in IEC 60601-1 for medical equipment [8].

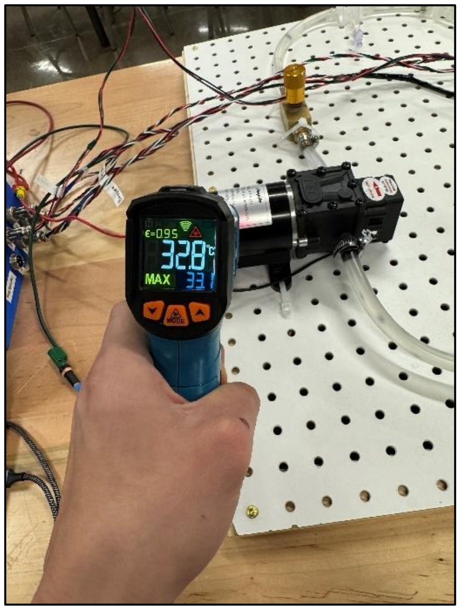

Figure 13: Thermal Safet Test Setup. Image showing the IR sensor used to measure the pump temperature.

2.5.2 Results Summary

Peak pump housing temperature: 47 °C (measured after 2 hours), comfortably below the 56 °C IEC 60601-1 threshold for metal surfaces and a contact duration of <10 s.

User safety: No surface exceeded burn-hazard levels, and the system exhibited zero performance drift.

These findings confirm compliance with IEC 60601-1 thermal-safety requirements and demonstrate that the pump can operate continuously for one hour without additional cooling, insulation, or thermal-management features.

2.6 Flow Sensor Calibration (Electrical)

2.6.1 Objective

The objective of this test was to calibrate the flow sensor by establishing the relationship between its raw output and the true flow rate of the system. By measuring actual flow rates at different pump duty cycles and comparing them to sensor readings, a calibration curve can be generated, ensuring reliable data acquisition in future system operation. The calibration curve should be generated with a y-offset of 0, to ensure that no flow rate is measured when the system is off.

2.6.2 Results Summary

The calibration process revealed a linear relationship between the flow sensor readings (x) and the true volumetric flow rates (y), measured by timing the filling of a 2-liter jug at various pump duty cycles. A calibration curve was generated (approximately $y = 1.37x$) and this linear equation was integrated into the Arduino code (see function `readFlowSensor()`) which enables precise monitoring and thus control of flow rates in the system.

2.7 Pressure Sensor Calibration Pre-Amplification (Electrical)

2.7.1 Objective

This objective of this test is to validate the raw output of the pressure transducer by comparing it to a reference manometer. The focus was to ensure the unamplified sensor accurately reflected applied pressures, providing a reliable baseline for subsequent signal amplification and calibration stages.

2.7.2 Results Summary

The raw voltage outputs from the pressure transducer closely matched the theoretical values expected for each applied pressure, from 25 mmHg up to 150 mmHg. Across all six pressure steps, deviations between measured and theoretical outputs remained minimal, within a margin of approximately 1-2%. These small discrepancies could be due to factors like minor air leaks or inherent sensor drift but were not significant enough to impact overall calibration reliability.

Figure 14: Pressure signal verification setup

2.8 Amplified Pressure Sensor Calibration (Electrical)

2.8.1 Objective

The objective of this test was to assess the performance of the INA118P instrumentation amplifier in boosting the pressure sensor signal. The focus was on confirming that the amplified output remained within the Arduino's readable range (0–3.3V) while achieving the desired gain, ensuring proper integration with the control system.

2.8.2 Results Summary

The INA118P instrumentation amplifier successfully increased the pressure transducer's low-level signals to the desired voltage range, maintaining close alignment with expected outputs across the majority of the pressure range. The system exhibited reliable, linear amplification from 25 mmHg to approximately 125 mmHg, with measured outputs nearly identical to theoretical predictions. However, as pressure approached the amplifier's upper limit (150 mmHg), the output began to saturate around 3.2V, just below the Arduino's maximum input of 3.3V. This indicates the amplifier's gain setting is effective but slightly constrained at higher pressures due to its single-supply design.

2.9 UI Verification (Software)

2.9.1 Objective

The objective of this test was to verify the functionality and reliability of the GUI for the CCTA. The test focused on ensuring that the UI correctly displayed system data, allowed accurate and responsive control of the pump system, and enabled real-time interaction with embedded sensors and hardware.

2.9.2 Results Summary

All tested UI functionalities passed. The interface successfully displayed live data and accepted user input for pump control, mode selection, and data export. The following key UI components were verified:

- ✓ Connection management (Connect/Disconnect buttons and status indicators)
- ✓ Data handling features including, Pause Graphs, Clear Data, and Export Data
- ✓ Real-time graphing of pressure and flow with target and rolling average overlays
- ✓ ON/OFF toggle for activating and deactivating pump output
- ✓ Manual control of pump power from 0–100% via slider
- ✓ Automatic control using targets, with UI highlighting setpoints accurately (full control system verification is seen in Section 2.10)
- ✓ Pulsatile control of pump by setting BPM and Pump Power inputs
- ✓ Numerical/graphical display of current, target, and rolling average values for all sensors
- ✓ Static Head Offset (SHO) calibration measures and integrates appropriate SHO values for each sensor after a 30 seconds calibration period

Various other small parts of the UI all worked as expected; full details are available in the testing folder for this test

These results confirm that the UI operates as intended and supports all core functions required for catheterization system operation and testing.

2.10 Control Systems Verification (Software)

2.10.1 Objective

The objective of this test was to validate the PID control system implemented in MATLAB, ensuring it can accurately track and respond to flow and pressure setpoints. The test evaluated key performance metrics like overshoot and settling time across different operating conditions to confirm system stability and responsiveness.

2.10.2 Results Summary

The PID control system demonstrated effective regulation of both flow and pressure across a variety of setpoint changes, showcasing its ability to maintain system stability. For flow control, overshoot was minimal, with the highest recorded at ~0.7 L/min during a sharp downward transition, and all settling times remained consistent around 30-60 seconds. This indicates a well-tuned response capable of smoothly adapting to changes. In contrast, pressure control exhibited faster and more controlled behavior, particularly during large step changes. The most significant overshoot occurred when transitioning from 20 mmHg to 100 mmHg, reaching a peak overshoot of 65 mmHg before settling. Smaller pressure changes were handled more gracefully, with overshoots limited to 10–15 mmHg.

3 Future Work

The following future work items are meant to guide future improvements and major design overhauls in upcoming iterations of the project. They have been split into Interdisciplinary, Mechanical, Electrical, and Software Engineering-focused design aspects.

3.1 Interdisciplinary

3.1.1 Implement a motorized pressure relief valve

The current system uses a manually adjusted pressure relief valve, which limits the ability to precisely tune pressure conditions during active tests. A future enhancement would be to replace this component with a motorized valve that can be electronically actuated and integrated into the existing GUI. This would allow for software-controlled pressure modulation in real time and enable the user to program dynamic pressure targets or implement safety cutoffs remotely, similar to other examples in literature [14].

From a mechanical standpoint, the team would first select a suitable motor-driven relief valve and design the mounts, couplings, and enclosure modifications needed to install it safely in the flow loop. On the electrical side, engineers would specify the motor driver, power requirements, and signal conditioning, then route the new control and feedback lines into the existing control box so the valve can be addressed by the Arduino. Finally, the software team would extend the firmware and MATLAB App Designer GUI to issue valve commands, display its state, and incorporate the new actuator into the PID logic—allowing the system to coordinate pump speed and valve position to hit dynamic pressure-flow targets and enforce automated safety cut-offs.

3.2 Mechanical Engineering

3.2.1 Optimize pump selection

The pump drives fluid through the tubing and is therefore critical to system performance. Upgrading to a medical-grade pump could improve pressure stability; two strong candidates are the MasterFlex pump [9] and the ViVitro Super Pump [10], both of which show data sheets suited to physiological flow applications.

However, swapping in a different pump would disrupt the existing CCTA control logic and circuitry, since the control box and pressure-regulator valve were tuned specifically for the current diaphragm pump. Adding the pressure regulator has already stabilized output pressures, enabling control within ±2 mmHg - well within acceptable limits.

If higher flow rates are needed, a larger-diaphragm model from the same pump family could instead be installed without compromising control. For seamless integration, any replacement must support a simple two-lead electrical connection. Further bench testing will be required to adjust control parameters and validate performance with the new pump.

3.2.2 Optimize control box design

The control box should be reconfigured as a single, self-contained enclosure that houses both the electronics and the power supply. Right now, separating the power supply forces multiple external cables and adds unnecessary bulk, which not only clutters the test bench but also creates potential trip hazards and connection points for electrical noise.

To improve waterproofing and electrical safety, mating seams and cable entries need tighter tolerances. Integrating O-rings or silicone gaskets at all joints will create reliable seals against accidental splashes or leaks.

Finally, outfitting the enclosure with non-marking rubber feet will boost stability and prevent the box from sliding during operation. These simple changes will streamline the setup, reduce wiring complexity, and enhance overall safety.

3.2.3 Perform vibration analysis and reduce base separation

During performance testing of the CCTA across various pump duty cycles, the system exhibited noticeable resonance at specific pump power values. This resonance was identified through waveform analysis, where the pressure signal trace displayed periodic sinusoidal oscillations indicative of vibrational coupling between the pump and the base structure.

Such resonance can interfere with pressure accuracy and flow stability, compromising the reliability of test results. To mitigate these effects, vibration isolation measures are necessary. One effective solution is to install gasket-type dampeners or compliant mounting pads beneath the base of the apparatus. These components absorb and dissipate vibrational energy, minimizing mechanical feedback and ensuring that pressure readings remain consistent and artifact-free during operation.

3.2.4 Increase anatomical realism

To more accurately simulate the human cardiovascular system, a future version of the CCTA should consider a comprehensive redesign of anatomical elements. This would include resizing tubing to better reflect human vessel diameters and incorporating compliant, elastic materials to replicate the mechanical properties of veins and heart tissue. Such a redesign would require re-evaluating many components - like the flow sensors - which are typically constrained to specific tubing sizes.

One promising direction would be to integrate a commercial heart model such as the Heartroid system, which the stakeholder initially intended to use but was unable to procure. Unlike the current prototype—which simulates flow primarily in the inferior and superior vena cava—models like Heartroid are designed for full heart anatomy, including chambers and valves, and would better simulate procedures like transseptal puncture or interventions beyond the venous system [11].

This goal is consistent with the improvements outlined by Johnson et al. in their cardiovascular simulator study [3]. Their discussion emphasized limitations like the absence of heart valves, non-physiological pump placement, and limited flow pattern variability, which are areas that are desired to be improved upon. For instance, embedding the pump within or directly connected to the heart model—as opposed to being external—could better simulate the apex-to-arterial flow dynamic, which is important for catheter trajectory realism.

Furthermore, future systems could support various pathological flow patterns such as tachycardia or heart block, which commercial products such as the "Harvard Pump," "SuperPump" and "CardioFlow" simulators can emulate [12]. This would involve expanding the control logic and possibly using more advanced flow generation methods like peristaltic pumps. Adding multiple flow sensors in different anatomical regions would allow for region-specific hemodynamic tracking and calibration. Finally, using a fluid with viscosity similar to blood (e.g., water-glycerol mixtures) could improve the fidelity of catheter response under physiological conditions.

In short, increasing anatomical realism would not only align the system more closely with real patient anatomy but would also make it a more valuable tool for both clinical training and catheter prototype development.

3.3 Electrical Engineering

3.3.1 Characterize the new flow sensor

The YF-B3 flow sensor is what was used for most of this project up until a week or so before handoff. It is rated for flows of 1-20 L/min and therefore loses significant accuracy at rates around 0.5-5 L/min (especially on the lower end) which is approximately what the CCTA can currently achieve.

The GREDIA 3/8" quick-connect sensor has been very recently added to the system and shows promising results – it is rated for flow rates from 0.3 to 10 L/min, which is much closer to the performance range of the CCTA. Preliminary tests show that with a sampling time of 1 second, this flow sensor allows PID control of flow to settle within about 10-20 seconds as opposed to 30-60 seconds with the YF-B3 sensor. Only a preliminary calibration was done, so a few more steps are required before this sensor can be fully integrated:

1. Perform a better calibration (see Section 2.6)
   a. For reference, for the GREDIA sensor, only a simple two-point calibration was performed before project handoff due to limited time. Gathering more points and measuring flow values more precisely is recommended.
2. Quantify the flow measurement error across a range of flow values
3. Precisely quantify the rise time, settling time, and overshoot characteristics of the PID control system using this new sensor

Some other sensors such as the YF-S402 (same brand as the current sensor) and others from the GREDIA family of 3/8" and ¼" food-grade flow meter sensors may be better candidates if even faster or more precise flow values are to be measured – this is because those sensors have a higher flow-rate-to-pulse-frequency conversion factor, meaning that they generate more pulses per volume of water pushed through, which allows for shorter sampling periods at the same accuracy level.

3.3.2 Use an ESP-32 microcontroller (or similar)

The ESP-32 offers several advantages over the Arduino Uno, making it a superior choice for more complex and connected projects. Unlike the Uno, the ESP-32 comes with built-in WiFi and Bluetooth, enabling wireless communication out of the box without extra modules. It has significantly more flash memory and RAM, which allows for larger, more capable programs. A key benefit is its ability to handle floating-point values directly over serial communication, which simplifies the code architecture and increases precision when sending or receiving values—ideal for applications like real-time sensor data or PID control. Additionally, the ESP-32 features a faster processor (dual-core 240 MHz vs. Uno's 16 MHz), more I/O pins, and support for multitasking, making it far more versatile for demanding or connected embedded systems.

3.4 Software Engineering

3.4.1 Implement PID autotuning

The current control loop uses fixed, manually tuned PID parameters, which perform well under steady-state conditions but may require recalibration when the system configuration or fluid resistance changes. A proposed future improvement is to implement an automatic PID tuning algorithm. While machine learning-based methods (e.g., Q-learning or DDPG) have shown success in complex systems, such approaches are not well-suited for microcontrollers with limited computational resources. For this system, it would be more practical to develop a lightweight, manually programmed autotuning algorithm.

One common method is the Ziegler-Nichols tuning approach, which involves setting the integral and derivative gains to zero, gradually increasing the proportional gain until the system output exhibits sustained oscillations, and using that critical gain and oscillation period to calculate suitable PID gains. Alternatively, a step response-based method can be used: the system applies a small step input (e.g., 30–50% power), records the response (rise time, overshoot), and adjusts PID gains based on predefined heuristics. This type of rule-based autotuner could be executed on startup or when prompted by the user, offering a good balance between adaptability and computational simplicity.

3.4.2 Allow upload of custom pressure and flow waveforms

The current system is already capable of generating pulsatile flow at specified BPMs and amplitudes, but only in the form of a continuous sinusoidal waveform. While this is sufficient for basic testing, future improvements should focus on implementing more physiologically realistic waveforms, such as those mimicking atrial or ventricular pressure profiles. A practical next step would be to program custom waveforms directly onto the Arduino as arrays of time-varying data points. These could be triggered at a defined BPM and amplitude as with the system currently.

In the longer term, it would be beneficial to enable uploading of custom pressure profiles derived from clinical recordings or device-specific testing requirements. Implementing this functionality would require extending the serial communication protocol to accept and parse full waveform datasets, as the current system only handles short, simple command strings. Accurate simulation of time-dependent pressure dynamics is especially important for testing devices like balloon catheters and ablation tools, which interact closely with the cardiac cycle [13].

3.4.3 Various code changes

Various suggested changes to the code are already commented within the codebase (look for comments with the key word TODO). Some of the major changes that should be implemented as of now are:

- Optimizing the serial communication protocol to reduce data losses and crashes
    - Sometimes the serial communication will crash after a few minutes, and sometimes only after a few hours. The source of these crashes should be investigated, and the GUI should be updated so that it does not completely crash if values have not been sent right away.
- Testing UI edge cases
    - Certain combinations of button presses and interactions with the UI cause unexpected errors in the system. These edge cases should be tested and documented, added into a full UI verification test plan to expand on Section 2.9

4 Conclusion

Cardiac catheterization testing currently depends on time-consuming animal models or costly, simplified in vitro systems, which pose ethical concerns and lack precise control over physiological parameters [1], [2].

Our capstone prototype overcomes these limitations with a modular flow loop that accurately reproduces static pressures (7 ± 2 to 300 ± 17 mmHg) and flow rates (0.5 ± 0.1 to 4.7 ± 0.1 L/min). It integrates a diaphragm pump, calibrated pressure and flow sensors, leak-proof tubing, and a MATLAB-based user interface for real-time monitoring and adjustment, ensuring reproducibility and ease of use [4].

By delivering stable PID-regulated control at a single test location and a user-friendly GUI, the system fulfills all stakeholder requirements, reduces reliance on animal testing, and offers Boston Scientific a cost-effective platform to accelerate device development—lowering R&D costs and expediting patient access to treatments like Farapulse [3], [5]

Future enhancements—such as advanced peristaltic pumps, motorized pressure regulation, anatomically realistic heart models, and pulsatile flow simulation—will further boost accuracy, safety, and clinical fidelity, positioning this apparatus as a new industry standard for ethical, efficient cardiac device testing [3].

5 References

[1] R. S. Wijesurendra and B. Casadei, “Mechanisms of atrial fibrillation,” *Heart*, vol. 105, no. 24, pp. 1860–1867, Dec. 2019, doi: 10.1136/heartjnl-2018-314267.

[2] M. Peirlinck *et al.*, “Precision medicine in human heart modeling,” *Biomech Model Mechanobiol*, vol. 20, no. 3, pp. 803–831, Jun. 2021, doi: 10.1007/s10237-021-01421-z.

[3] A. Johnson *et al.*, “An Inexpensive Cardiovascular Flow Simulator for Cardiac Catheterization Procedure Using a Pulmonary Artery Catheter,” *Front Med Technol*, vol. 3, p. 764007, Oct. 2021, doi: 10.3389/fmedt.2021.764007.

[4] J. Foss and Ebady, Rhodaba, “Stakeholder Discussion with Boston Scientific Representative,” May 10, 2024.

[5] Boston Scientific, “FARAPULSE™ Pulsed Field Ablation (PFA) System,” Boston Scientific. Accessed: Sep. 17, 2024. [Online]. Available: https://www.bostonscientific.com/en-US/products/catheters--ablation/farapulse.html

[6] “Femoral Vein: Anatomy & Function,” Cleveland Clinic. Accessed: Dec. 04, 2024. [Online]. Available: https://my.clevelandclinic.org/health/body/23041-femoral-vein

[7] “Calculate density and viscosity of glycerol/water mixtures.” Accessed: Apr. 22, 2025. [Online]. Available: https://www.met.reading.ac.uk/~sws04cdw/viscosity_calc.html

[8] “an_maximum_allowable_temperature.pdf.” Accessed: Apr. 23, 2025. [Online]. Available: https://www.advancedenergy.com/getmedia/8544158d-181a-4083-814e-90ffb75cf298/an_maximum_allowable_temperature.pdf

[9] “Masterflex Selection Guide for Masterflex I/P Pump Heads.” Accessed: Apr. 11, 2025. [Online]. Available: https://neurophysics.ucsd.edu/Manuals/Cole%20Parmer/ColeP_Masterflexguide_ML.pdf

[10] “SuperPump Pulsatile Pump,” ViVitro Labs. Accessed: Apr. 23, 2025. [Online]. Available: https://vivitrolabs.com/product/superpump-pulsatile-blood-pump/

[11] “HEARTROID | MEDICAL TRAINING SYSTEM,” HEARTROID | MEDICAL TRAINING SYSTEM. Accessed: Apr. 11, 2025. [Online]. Available: https://www.heartroid.com

[12] Rahul Raj Mechoor and Tyler Schmidt, “A Real-Time Programmable Pulsatile Flow Pump for In Vitro Cardiovascular Experimentation,” *Journal of Biomedical Engineering*, vol. 138, no. 11, Oct. 2024, doi: 10.1115/1.4034561.

[13] S. Drost, B. J. de Kruif, and D. Newport, “Arduino control of a pulsatile flow rig,” *Medical Engineering & Physics*, vol. 51, pp. 67–71, Jan. 2018, doi: 10.1016/j.medengphy.2017.10.006.

[14] Bosco Giada, “Multi-physics simulation for a pressure relief valve,” Politecnico di Torino, 2024. Accessed: Apr. 11, 2025. [Online]. Available: https://webthesis.biblio.polito.it/31236/1/tesi.pdf