# Until Take Home Project

Brian Check

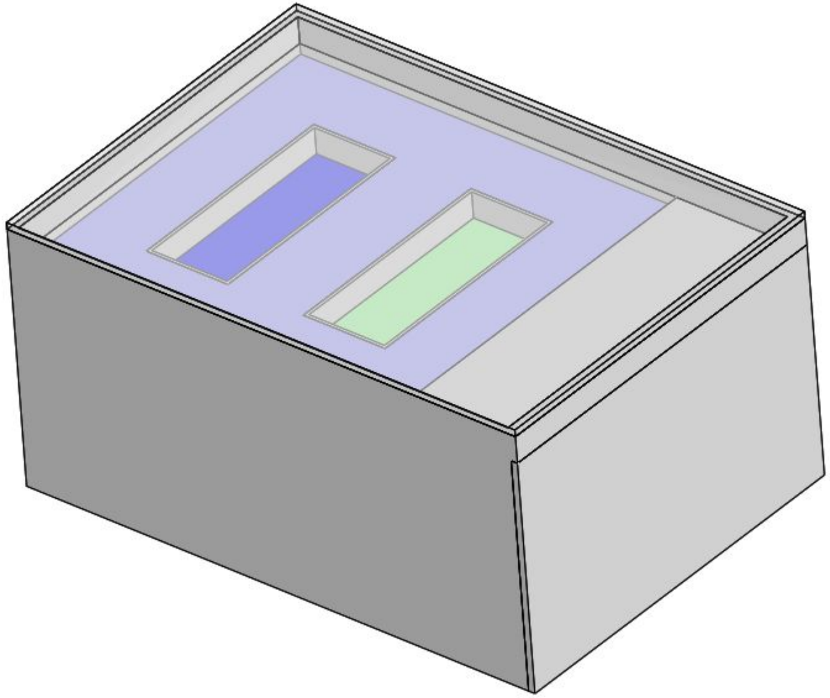
Files available on GitHub

# Presentation Overview

Task 1: Mobile CPA Loading System
1. Problem Overview
2. System Architecture
3. Controller Design and Simulation
4. Hardware Overview

Task 2: Cradle Design
1. Task Overview
2. Design Overview
3. BOM

# <u>Task 1</u>
# Mobile CPA Loading System



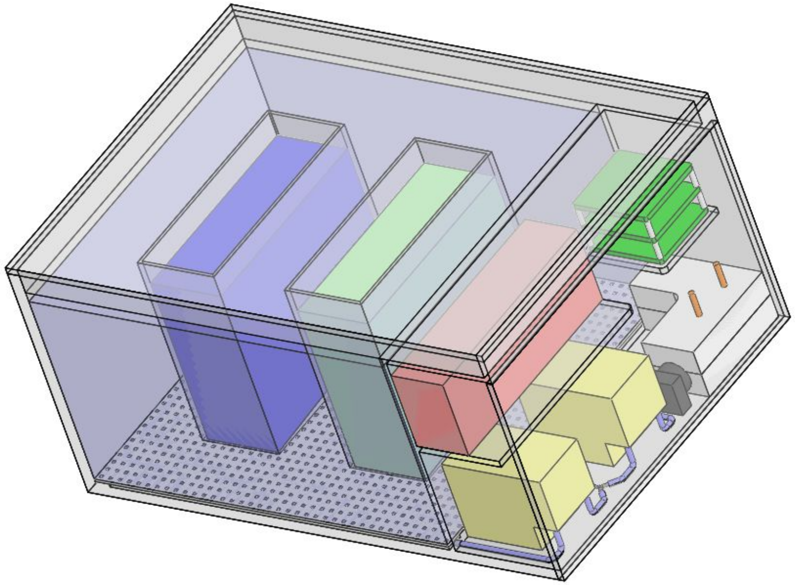
# Task Overview

**Challenge**

- Design mobile system to load cryoprotectants into porcine kidneys during transport

**Requirements**

- Temperature: 4°C ± 2°C
- Pressure: 30 mmHg ± 3 mmHg
- Variable flow: 3-30 mL/min
- Two-fluid mixing
    a. Carrier + CPA, ramping up CPA over time
- Single-pass perfusion
- ~2 hour runtime, battery powered



# Domain Research and Benchmarking

<u>LifePort Kidney Transporter</u> - gold standard for clinical kidney transport

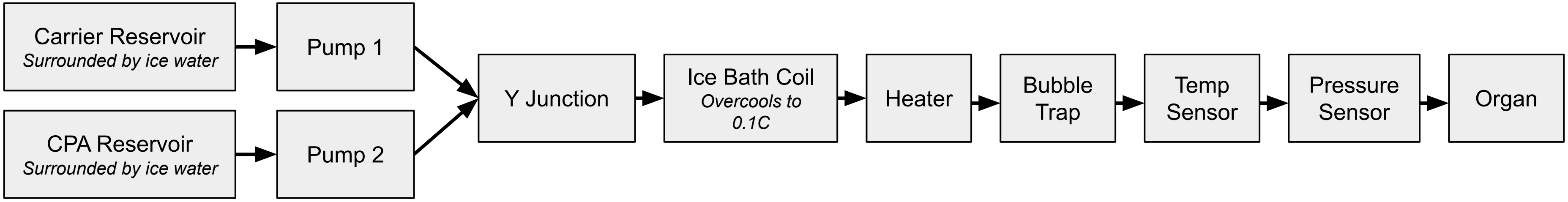
- Pulsatile flow, pressure control, recirculating
- Passive cooling with ice bath


LifePort keeps organs cold, but doesn't meet Until's requirements:

- Cooling: passively controlled, ~2-10°C (not 4±2°C precision)
- Recirculating flow (not single-pass)
- Limited to hypothermic preservation (no CPA loading capability)

**Key Takeaway**: Until's challenge requires *active temperature control* with *single-pass perfusion* during CPA concentration ramping - fundamentally different operating regime than existing clinical transporters

# System Architecture




## How I got here

**The Fundamental Challenge**

- To hold ±2°C, you need closed-loop control. Closed-loop control requires an actuator you can modulate
- *How do you actively control temperature in a fluid line?*
  - Peltier/TEC: Bulky, power-hungry, complex heat rejection
  - Recirculating chiller: Not mobile - requires benchtop equipment
  - Variable ice contact: No practical way to modulate heat transfer dynamically

**The realization**

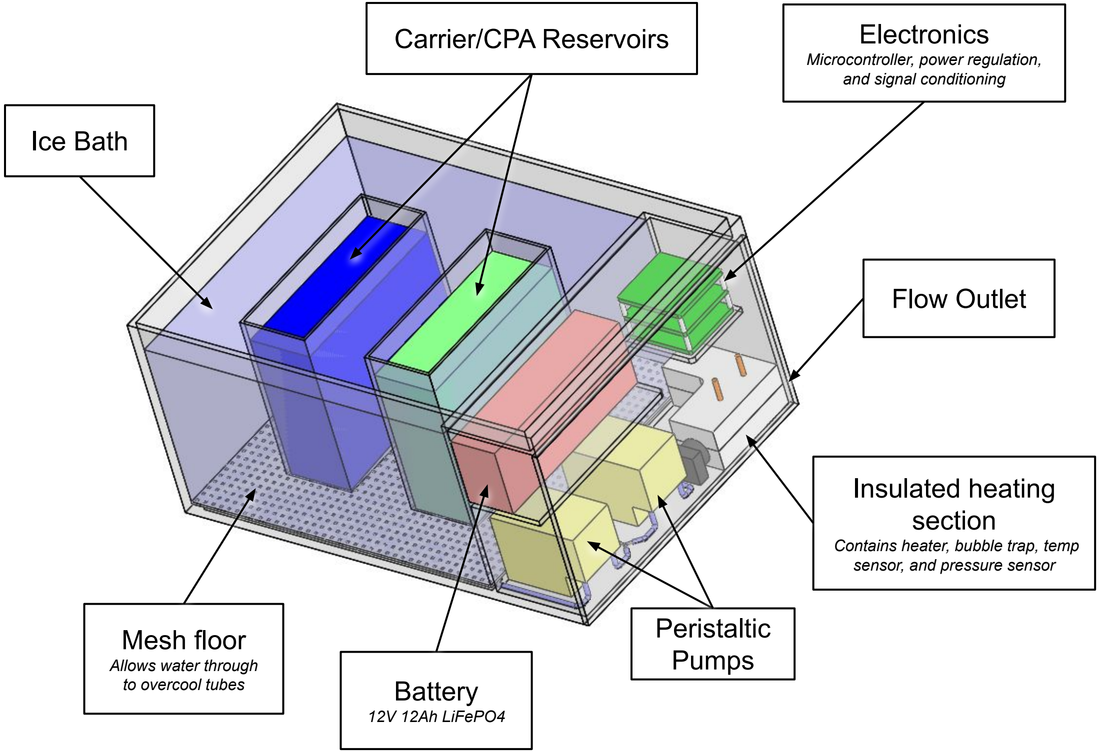
- Cooling is hard to modulate. But *heating* is trivial - a resistive heater with PWM gives you precise, instantaneous control.

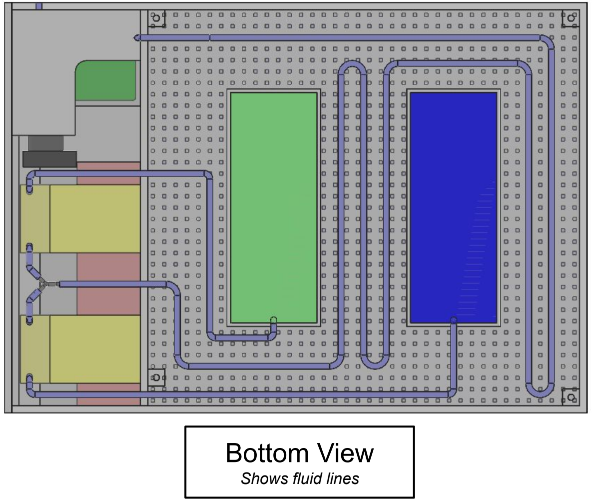
**The architecture flip**

- Instead of trying to cool *precisely* to 4°C, overcool *past* the setpoint, then heat back up.

# System Architecture - Layout [Overcooling]

*Ice water cooling reservoir and isolated electronics compartment with insulated top lid and removable side access panel*



# Control: Setup & Challenge

Two independent control loops:

- **Pressure → Pump speed** (sets flow rate)
- **Temperature → Heater power** (compensates for varying flow)

Challenge: Flow varies 10x during protocol - heater must maintain 4°C regardless despite temp being coupled with flow speed

# Simulation Setup [Pseudocode]

**Setpoints (targets):**
- P_setpoint, T_setpoint

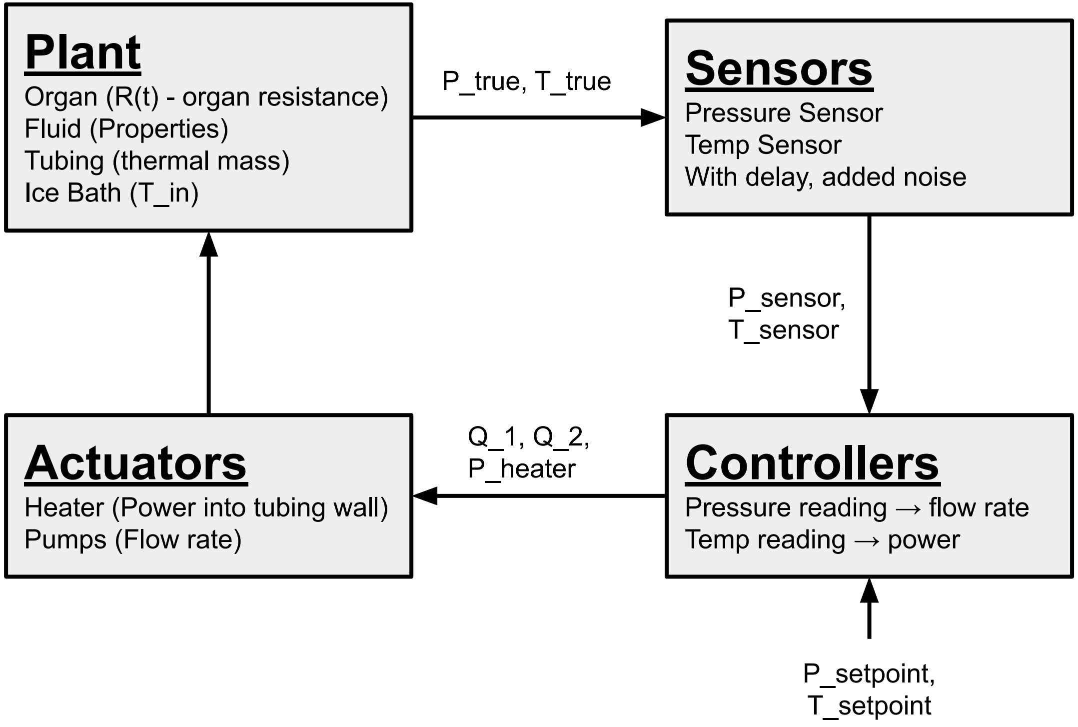
**Disturbances:**
- R(t) - organ resistance changing over time
- T_in - cold fluid from ice bath

**Measured (controller inputs):**
- P_sensor
- T_sensor

**Commanded (controller outputs):**
- Q_total → split to pumps
- P_heater → to heater

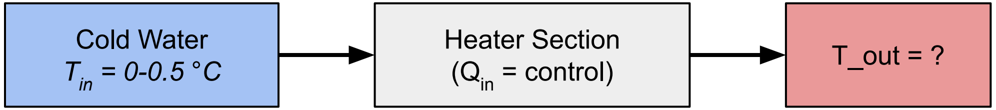
**States (internal dynamics):**
- T_wall — tubing wall temperature
- P_integral, T_integral — controller memory

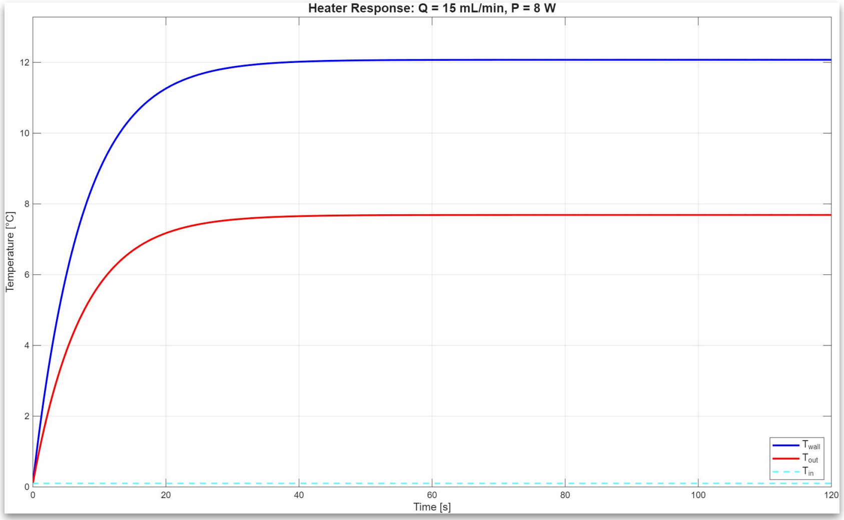
**Outputs (what you care about):**
- P_true — actual organ perfusion pressure
- T_out — actual fluid temperature to organ



# Heat Transfer: Transient Dynamics Model [More Detail]

*Before designing the controller, I modeled the heater dynamics. This isn't a steady-state problem. The wall has thermal mass, so there's a transient response. When I change heater power, the wall temperature evolves according to an energy balance ODE.*




Simulation: Temp vs Time for Fixed Flow Rate and Heater Power
(Code: heater_dynamics.m)

**Initial condition:** $T_{wall}(0) = T_{in}$

**Reynolds and Prandtl numbers:**

$$Re = \frac{\rho V D}{\mu} = \frac{4\dot{m}}{\pi D \mu} \qquad Pr = \frac{\mu c_p}{k}$$

**Nusselt number [1]:**

$$\overline{Nu}_D = 3.66 + \frac{0.0668(D/L)Re_D Pr}{1 + 0.04[(D/L)Re_D Pr]^{2/3}}$$

**Heat transfer coefficient:**

$$h = \frac{\overline{Nu}_D \cdot k}{D}$$

**Number of Transfer Units [2]:**

$$NTU = \frac{h \cdot \pi D L}{\dot{m} c_p}$$

**Effectiveness [2]:**

$$\varepsilon = 1 - e^{-NTU}$$

**Wall energy balance (first principles):**

$$C_{wall}\frac{dT_{wall}}{dt} = P_{heater} - \dot{m} c_p \varepsilon \left(T_{wall} - T_{in}\right)$$

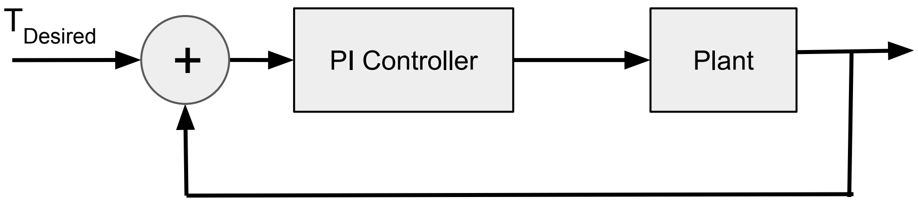
**Update wall temperature (Euler integration):**

$$T_{wall}^{n+1} = T_{wall}^{n} + \frac{dT_{wall}}{dt} \cdot \Delta t$$

**Outlet temperature (effectiveness definition):**

$$T_{out} = T_{in} + \varepsilon \left(T_{wall} - T_{in}\right)$$

# Control Attempt #1: Pure PI

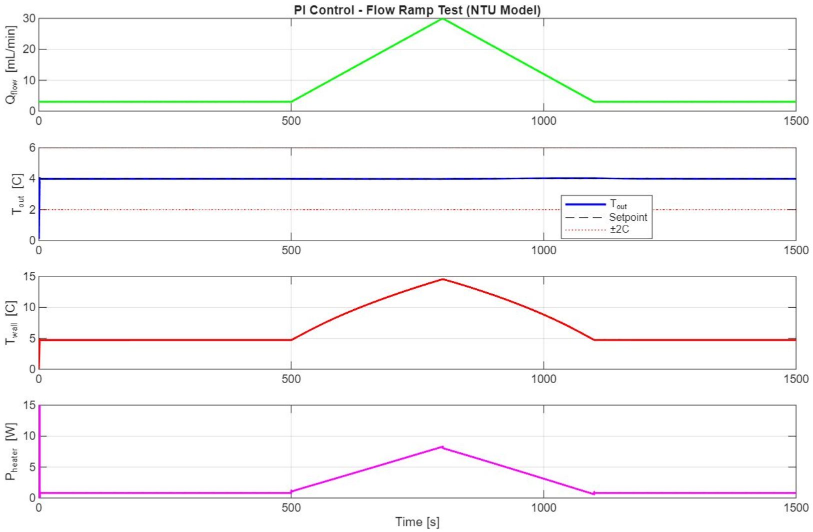



"Worked great" - but suspiciously great

- Cranking Kp to 100, 1000, arbitrarily high → still perfect tracking, no overshoot
- Red flag: Real systems don't behave this way

The problem: First-order lag =/= transport delay

- Wall dynamics are a lag
  a. Output starts changing immediately, just slowly
- Controller sees error right away, can start correcting
- First-order lag + PI is always stable, can't overshoot

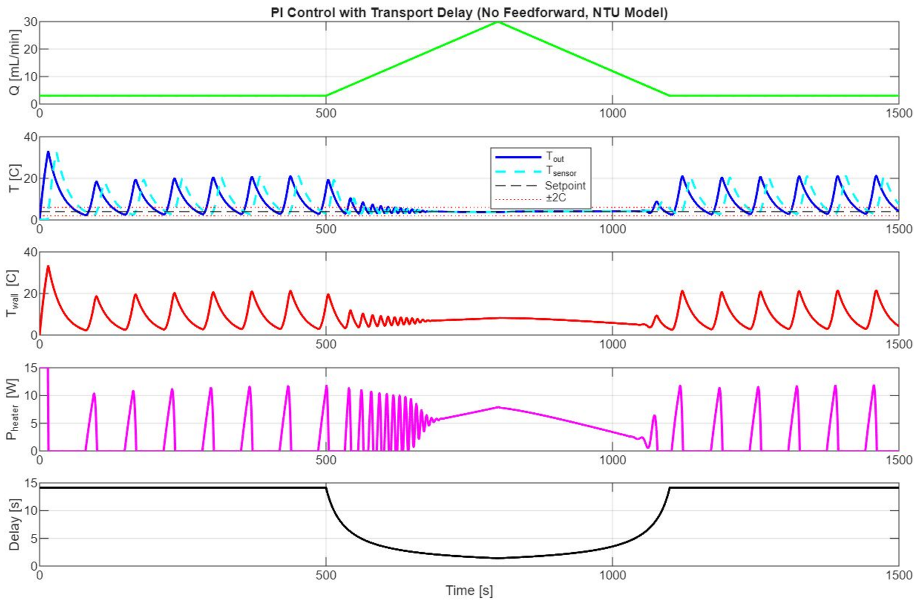



"Perfect" temp control with $K_p$ = 1000 in simulated pump velocity ramp up/down (Code: Pure_PI.m)

# Control Attempt #2: Pure PI with Added Transport Delay

A Realization

- Sensor located 8 cm downstream of heater
  - Transport delay = tube volume / flow rate
  - At 30 mL/min: **1.1 s delay**
  - At 3 mL/min: **11.3 s delay**
- PI now controlling based on stale information
- Variable delay makes tuning impossible - gains that work at high flow oscillate at low flow

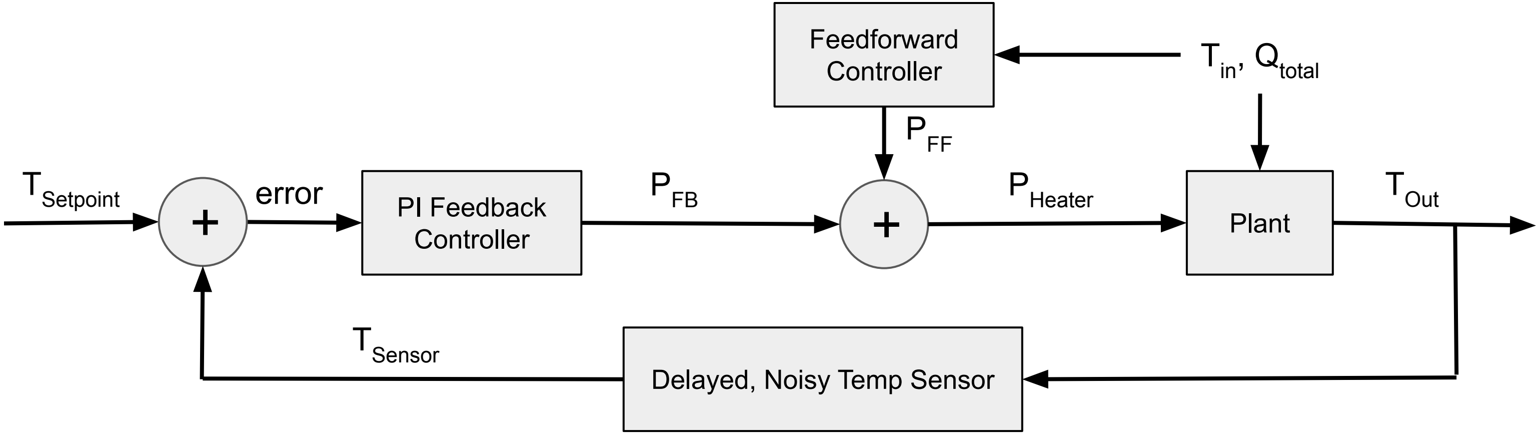



Adding delay to sensor input blew up controller
(Code: Pure_PI_Lag.m)

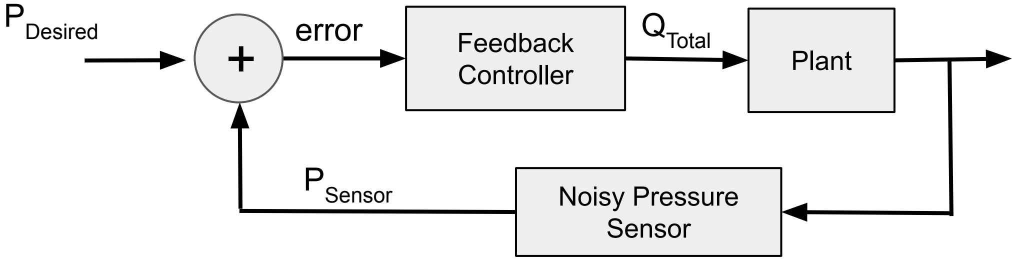
# Final Control Architecture: Feedforward PI Controller

## Temperature Loop

*Change in Q or $T_{in}$ modeled as Disturbance to the system. Compensated for by FF controller*




## Pressure Loop

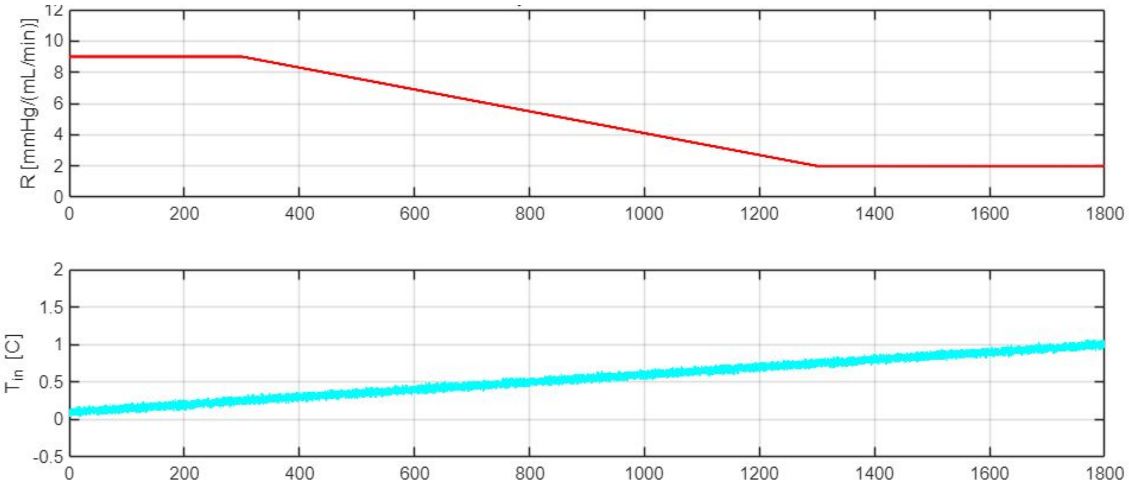



Note: Mixing ratio open-loop controlled by ratio of flow rates
(Code: main.m and run_simulation.m)

# Simulation Setup

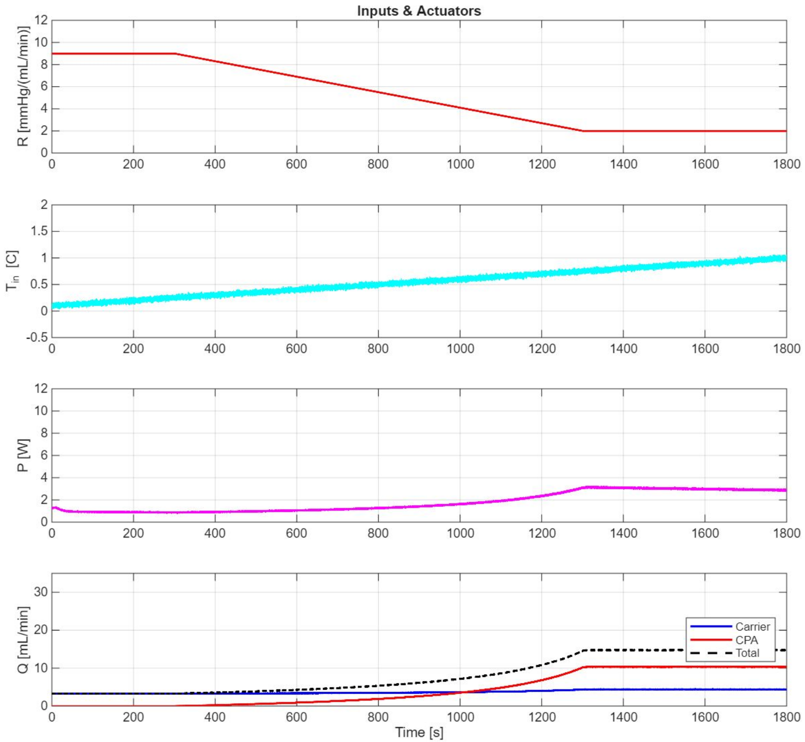
30 min protocol: equilibrate → ramp CPA 0→70%
Disturbances modeled:

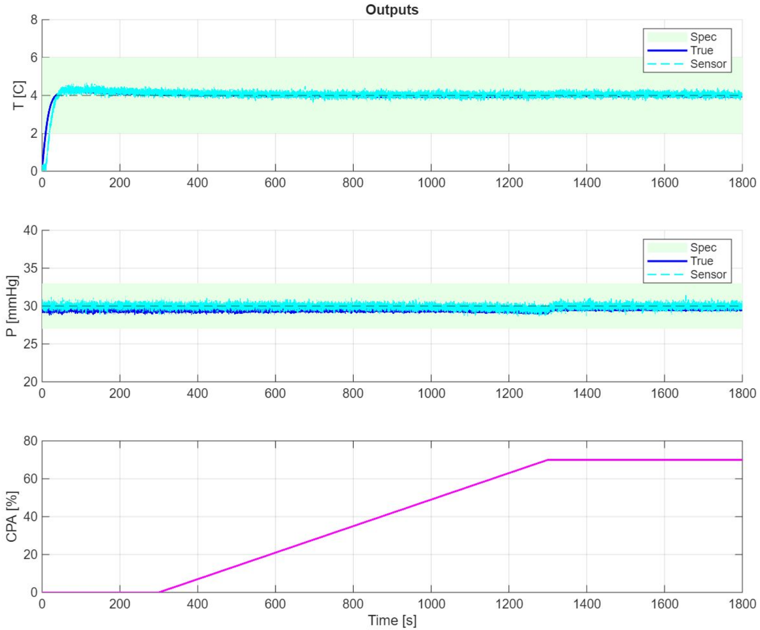
- Vascular resistance: 9 → 2 mmHg/(mL/min) (organ relaxes)
  a. P = Q x R - Pressure adjusts accordingly
- Ice bath drift: +1°C over 30 min
- Sensor noise/bias: Temp (σ=0.1°C, bias=0.1°C), pressure (σ=0.3 mmHg, bias=0.5 mmHg)




Environmental changes to ice bath and organ resistance over simulation

# Simulation Results - Plots



# Simulation - Results and Limitations

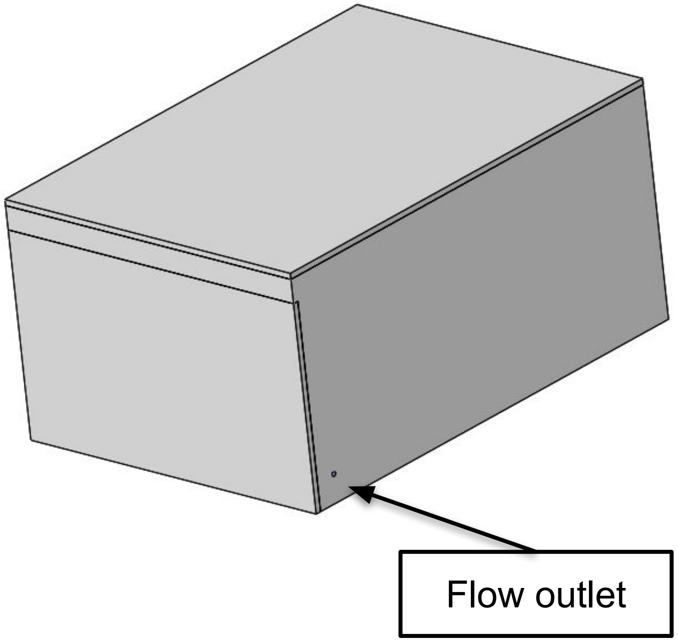
| Metric | Result | Spec |
|---|---|---|
| Temp max error | 0.2°C | ±2°C |
| Temp mean error | 0.06°C | — |
| Pressure max error | 1.1 mmHg | ±3 mmHg |
| Pressure mean error | 0.56 mmHg | — |
| In spec | 100% | — |
| Heater power | 3.2 W max, 1.8 W mean | |

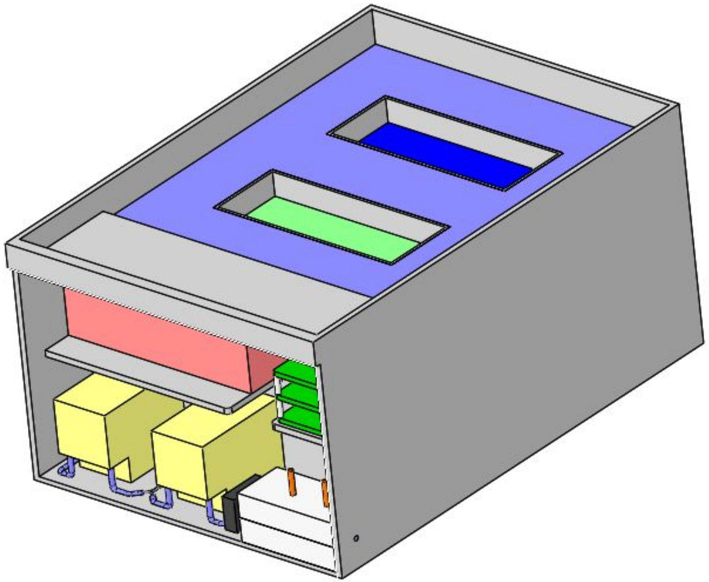
**Limitations**

- FF uses same thermal model as plant → results are optimistic.
    a. Real system will have model mismatch, would require tuning
    b. Cp varies with CPA%, heater efficiency
- PI gains need tuning on hardware
- Not modeled: bubbles, pump pulsation, startup transients
- TLDR; have to tune IRL

# Hardware Overview

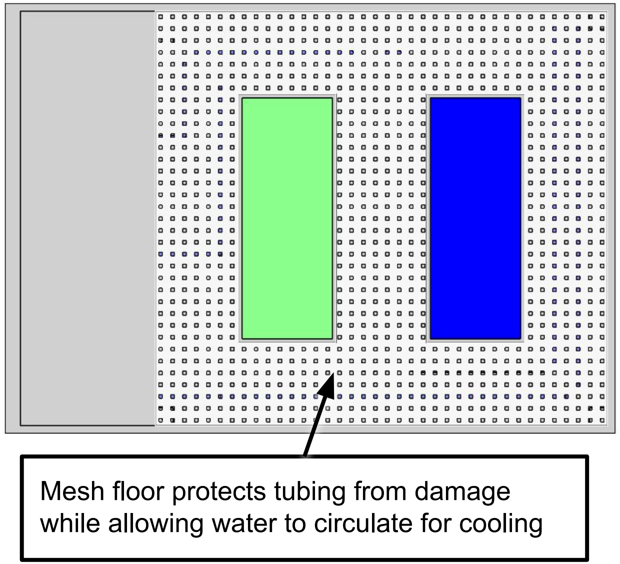
Insulated lid, removable electronics panel

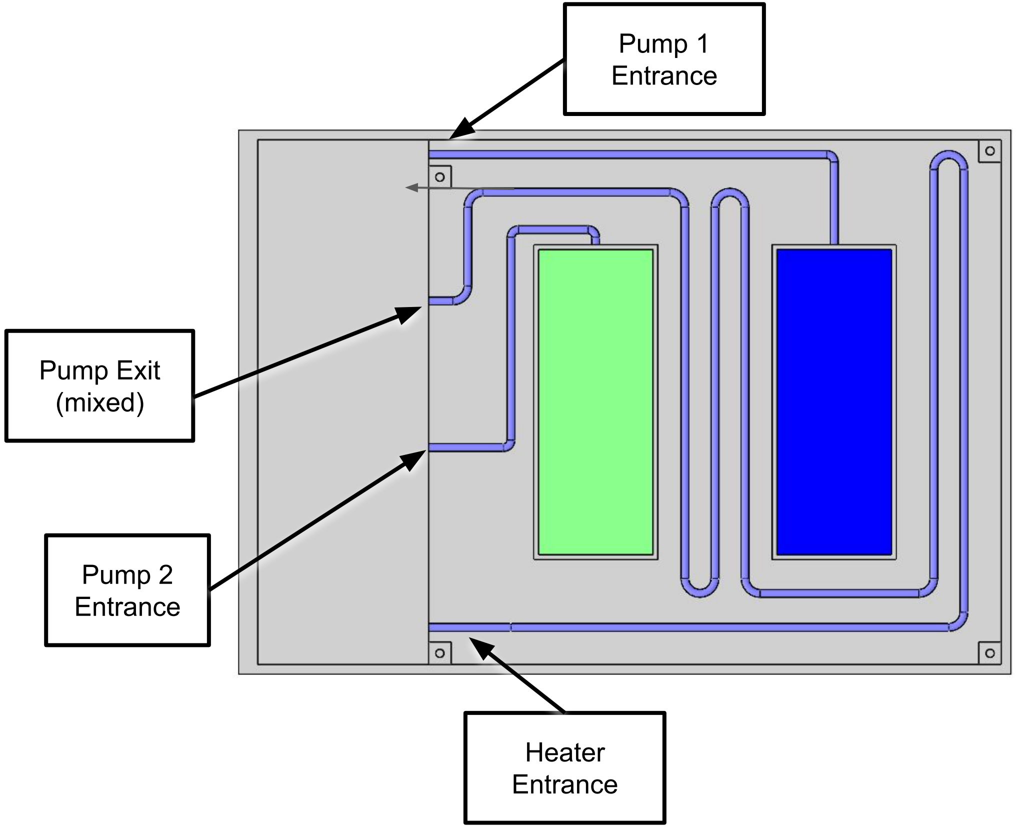


# Hardware Overview - Plumbing (Top view, no ice water)

3mm ID tubing



# Hardware Overview - Electronics

Components

- Temp Sensor
- Pressure Sensor
- Heater (wrapped around tube)
- Battery
- Bubble Trap
- Y fitting
- T100-WX10 OEM Peristaltic Pump
- Arduino/ESP32 Microcontroller

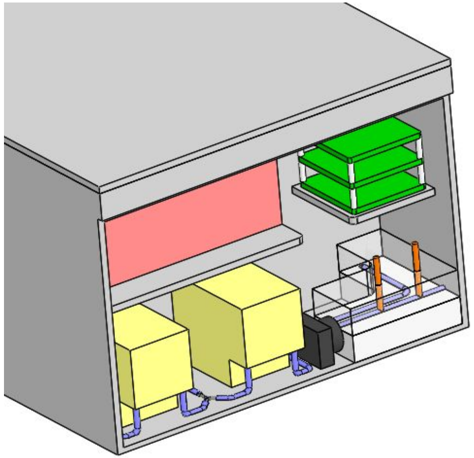
Power: 20W typical, 35W max

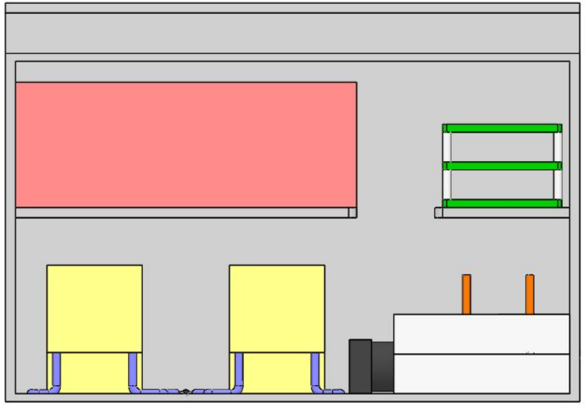
- Heater: 3W
- Pumps: 2x 12W max
- Controller/Sensors: ~5W

Size: 14.3" × 20.2" × 9.9"

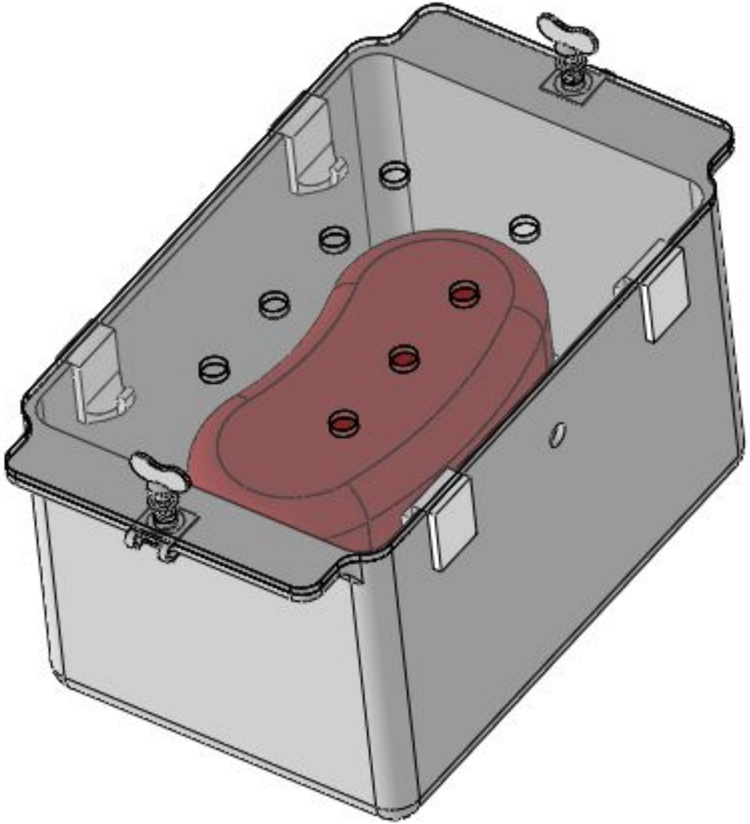
Next Steps

- Power distribution (12V → 5V/3.3V)
- Signal conditioning
- Battery charging
- Mechanical DFM/DFA





# Task 2
Cradle



# Task 2

Requirements

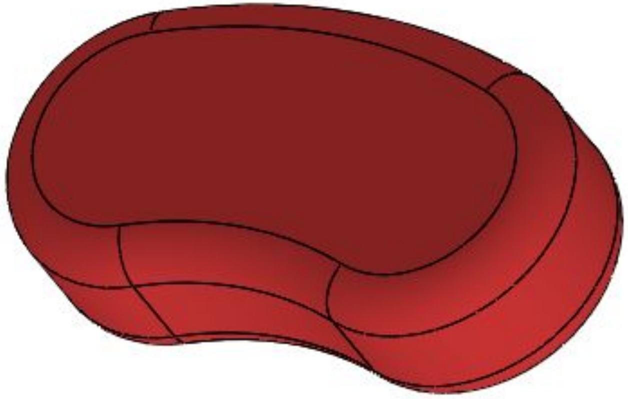
- Compliantly hold adult pig kidney in sealed bag
- Fit within 8" diameter × 12" length cylinder
- 9 sealable ports for fluid/probe access
- Arterial cannula at 90° to kidney

Kidney sizing (3σ range) [Source]

- Length: 92 - 148 mm
- Width: 37 - 77 mm
- Thickness: 14 - 41 mm

Design envelope

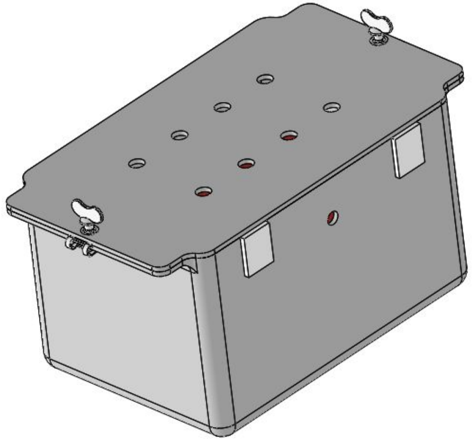
- Fits largest kidney + margin for bag + effluent pool

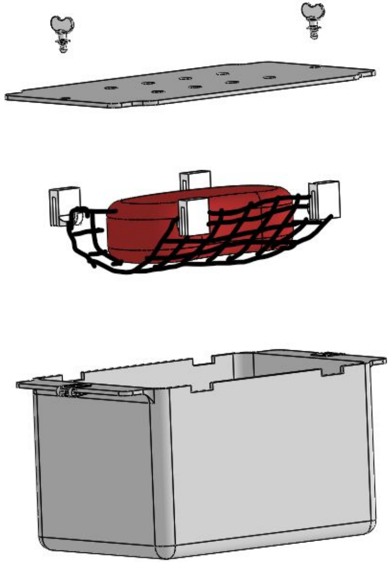


# Design Overview

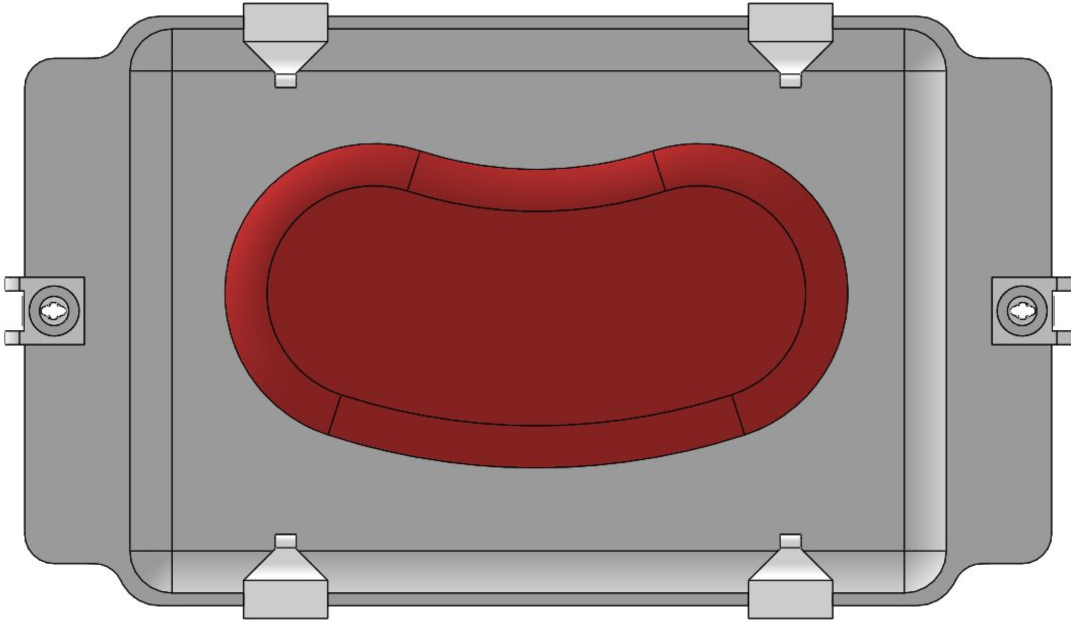
**Features**

- Rectangular box with flanged rim
- Removable lid with 8 luer ports
    a. 9th port on side wall
- Mesh sling suspended from hooks on rim
- Press fit hooks slide onto box
- Cannula holder on side wall (90° to kidney)
- Quarter-turn latches secure lid

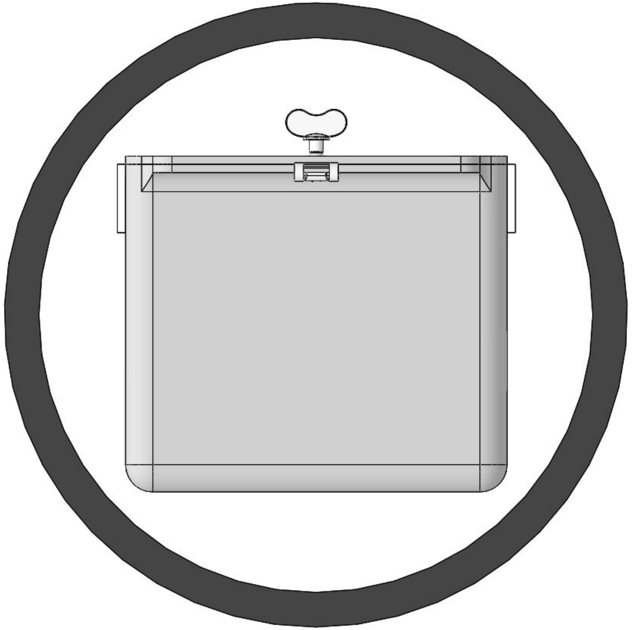




# It Fits! Dimensions: 5.75” x 10” x 5.5”



Largest (3σ) Kidney



# BOM, Materials, and Manufacturing

| Part | Qty | Prototype Mfg Method | Production Mfg Method |
|---|---|---|---|
| Box | 1 | SLS Tough, print upright | Injection molded PP |
| Lid | 1 | SLA Tough, print flat | Injection molded PP |
| Mesh hooks | 4 | SLA Tough, print on side | Injection molded PP |
| Mesh sling | 1 | Off-the-shelf | Off-the-shelf |
| Port fittings | 9 | Off-the-shelf luer | Same |
| Quarter turn fasteners | 2 | McMaster 91382A350 | Off-the-shelf |

# Appendix

# Controller Simulation Loop Pseudocode [Go Back]

**Inputs**

- Set R from protocol schedule
- Set cpa_ratio from protocol schedule → update fluid properties (ρ, Cp, k, μ)
- Update T_in from ice bath model

**Pressure control → (True pressure → Pressure sensor reading → Flow speeds)**

- P_true = Q_total(i-1) x R
- P_sensor = P_true + bias + noise
- P_error = P_setpoint - P_sensor
- Q_total = Q_nominal + Kp x P_error + Ki x P_integral
- Clamp Q_total to [Q_min, Q_max] with anti-windup
- Split: Q_cpa = Q_total x cpa_ratio, Q_carrier = Q_total x (1 - cpa_ratio)

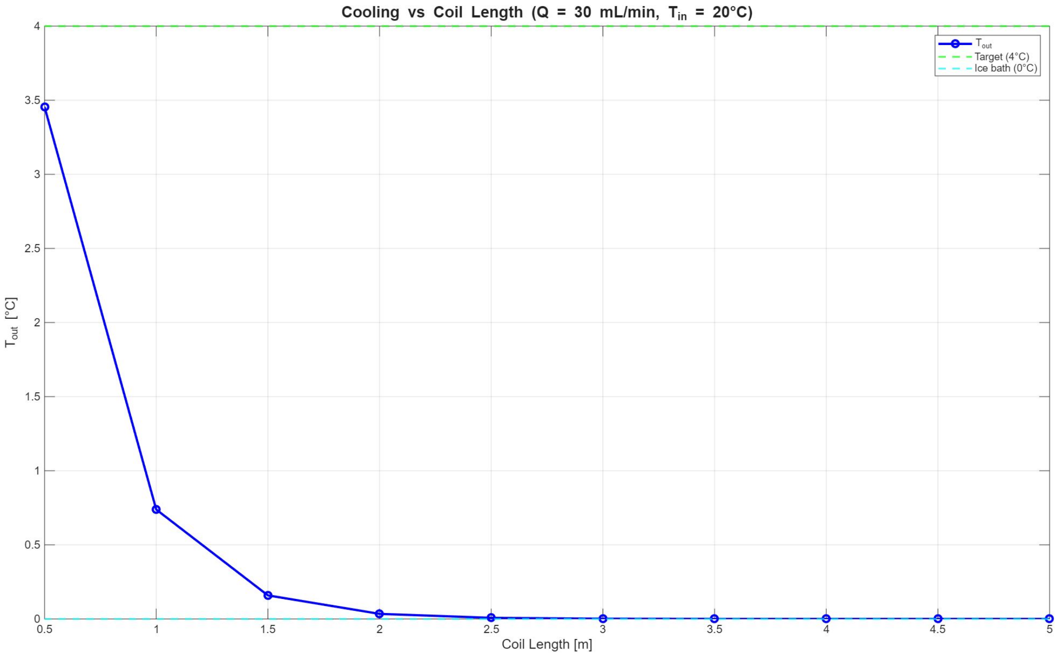
**Temperature control → (True temp → Temp sensor reading → Heater power)**

- T_out = heater_dynamics(T_in, T_wall, Q_total, P_heater, fluid, ...)
- delay_samples = V_sensor / Q / dt (quasi-steady approximation)
- T_sensor = T_out(delayed) + bias + noise
- T_error = T_setpoint - T_sensor
- P_ff = ṁ x Cp x (T_setpoint - T_in) (steady-state feedforward)
- P_fb = Kp x T_error + Ki x T_integral
- P_heater = P_ff + P_fb
- Clamp P_heater to [0, P_max] with anti-windup

**Plant update:**

- T_wall += dT_wall/dt × dt (from heater_dynamics)

# Overcooling [Go Back]



# 3D Printing Method Selection

| | SLS (PA12) | SLA (Resin) | FDM (PLA/ABS) |
|---|---|---|---|
| **Strength** | **High** | Low | Med |
| **Surface Finish** | Med | **High** | Low |
| **Watertight** | High | **High** | Low |
| **Large Parts** | High | Low | High |
| **Thin Walls** | **High** | Low | Low |
| **Biocompatibility** | Med | Med | Low |
| **Tolerance** | Med | **High** | Low |
| **Flexibility** | Med | Low | Med |

# Heat Transfer Calculations [Go Back]

Inputs:
- Geometry: D (tube diameter), L (heated length)
- Fluid properties: ρ, μ, cp, k (functions of CPA concentration)
- Operating conditions: mdot (mass flow rate), Tin (inlet temperature), Pheater (heater power)
- Thermal mass: Cwall (wall thermal capacitance)

Assumptions:
- Laminar flow
- Fully developed velocity profile, developing thermal profile
- Lumped wall thermal mass (uniform Twall over heater length)
- Negligible axial conduction in fluid
- Cr = 0 (wall thermal capacity ≫ fluid capacity in heater section)

This is a first-principles thermal model. The heater wall has thermal mass - it stores energy. When I apply power, the wall heats up, and heat transfers to the fluid based on the NTU-effectiveness relationship. The key thing is that the dynamics depend on flow rate: at high flow, the fluid pulls heat away faster, so the wall responds differently than at low flow. This is a nonlinear, flow-dependent system.

**Initial condition:** $T_{wall}(0) = T_{in}$

**Reynolds and Prandtl numbers:**

$$Re = \frac{\rho V D}{\mu} = \frac{4\dot{m}}{\pi D \mu} \qquad Pr = \frac{\mu c_p}{k}$$

**Nusselt number [1]:**

$$\overline{Nu}_D = 3.66 + \frac{0.0668(D/L)Re_D Pr}{1 + 0.04[(D/L)Re_D Pr]^{2/3}}$$

**Heat transfer coefficient:**

$$h = \frac{\overline{Nu}_D \cdot k}{D}$$

**Number of Transfer Units [2]:**

$$NTU = \frac{h \cdot \pi D L}{\dot{m} c_p}$$

**Effectiveness [2]:**

$$\varepsilon = 1 - e^{-NTU}$$

**Wall energy balance (first principles):**

$$C_{wall}\frac{dT_{wall}}{dt} = P_{heater} - \dot{m} c_p \varepsilon\,(T_{wall} - T_{in})$$

**Update wall temperature (Euler integration):**

$$T_{wall}^{n+1} = T_{wall}^{n} + \frac{dT_{wall}}{dt} \cdot \Delta t$$

**Outlet temperature (effectiveness definition):**

$$T_{out} = T_{in} + \varepsilon\,(T_{wall} - T_{in})$$

# Pig Kidney: Worst Case Dimensions (+-3σ) [Go Back]

| Dimension | Mean | SD | Smallest (−3σ) | Largest (+3σ) |
|---|---|---|---|---|
| **Length** | 120 | 9.2 | 92.4 | 147.6 |
| **Width (cranial pole)** | 56.9 | 6.7 | 36.8 | 77.0 |
| **Width (hilum)** | 52.6 | 6.1 | 34.3 | 70.9 |
| **Width (caudal pole)** | 52.4 | 6.7 | 32.3 | 72.5 |
| **Thickness** | 27.9 | 4.5 | 14.4 | 41.4 |

*All dimensions in mm*
*Max: Roughly a 148x77x41 bean*

# Sources


1. Incropera, F.P., DeWitt, D.P., Bergman, T.L., Lavine, A.S. *Fundamentals of Heat and Mass Transfer*, 7th ed., Wiley, 2011, Eq. 8.57, p. 517
    a. For heat transfer basics

2. Kays, W.M. \& London, A.L. *Compact Heat Exchangers*, 3rd ed., McGraw-Hill, 1984, Ch. 2, pp. 19-24
    a. For NTU effectiveness calculations to model changing wall temperature

3. Gómez FA, Ballesteros LE, Estupiñán HY. Anatomical study of the renal excretory system in pigs. A review of its characteristics as compared to its human counterpart. *Folia Morphologica*. 2017;76(2):262–268. doi:10.5603/FM.a2016.0065
    a. For pig kidney size