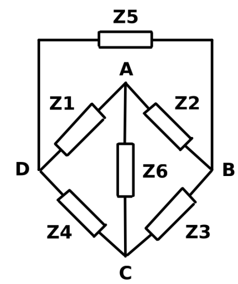
ECE 202 Project Proposal: Simple Electrical Impedance Tomography

Team Members: Jonah Spector, Connor Cassidy, Christian Rayner
EIR Mentor: Chuck Duey

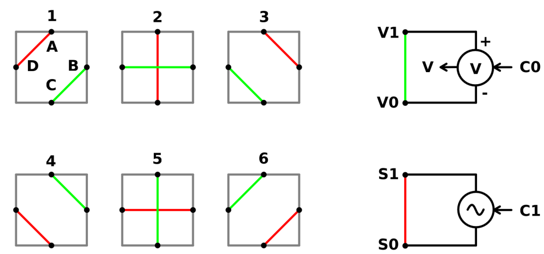
Electrical impedance tomography is a method for medical imaging which visualizes the internal resistivity of the human body, namely for real-time monitoring of the lungs. It measures the voltage on n-electrodes and sends an alternating current through 2 other electrodes. This results in a non-linear, ill-posed boundary value problem which, when solved, gives a 2D tomogram of resistivities. We have decided to make a 4-electrode electrical impedance tomography device for our project using a first-order approximation of the resistivity in the medium. Our visualization will show four pixels, each representing the four quadrants outlined on the sides by the electrodes. Each pixel will show if it is likely that there is an object of lower/higher resistivity contained in that quadrant. This visualization, we hope, will update at a reasonable framerate of 5-60 fps. If we succeed at this, we may expand our project to more electrodes, better temporal resolution, or improve it in other ways.

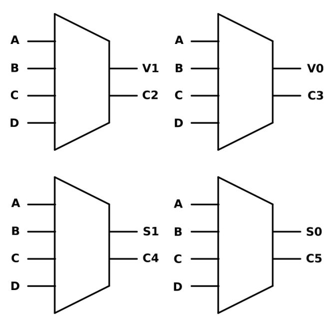
Project Idea



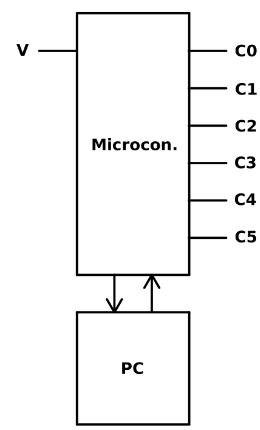
In order to calculate the location of the object of lower resistivity (OLR), we plan to use a first-order circuit as a model for the actual medium of known concentration saline. By sending an AC signal at a frequency high enough to avoid electrolysis of the saline, and low enough to be captured by the digital multimeter (~5-10 kHz), we can reconstruct the impedances of the first-order circuit.

We plan to send the signal across 2 electrodes, and measure the voltage across the remaining 2 electrodes. With 4 electrodes, this gives us 6 configurations (or 6 unique voltage measurements). We will use four, 1-to-4 MUXs to switch between the 6 configurations.





A microcontroller will control the 4 synchronous MUXs at 30-360 Hz (configurations per second). Each time the configurations switch, the microcontroller starts a new multimeter measurement and saves the data. Every frame, all 6 configurations' data are packaged into 6 values sent to the computer. The computer takes the 6 measurements and computes the visualization (5-60 fps).



Overall, the microcontroller will need to switch configuration, tell the multimeter to measure the voltage for 2-30 ms, repeat this 6 times, package the 6 measurements, and send them to the computer for processing. The computer will take the 6 values, compute the 4 pixel image, and repeat for a new set of values 5-60 times per second.

The data will be processed into a 4 pixel image showing which resistivity (Z1, Z2, Z3, or Z4) is the lowest (where the OLR is). The OLR will be made of plastic, and a test rig will be constructed to hold the electrodes and saline.

Material List: 4 x 1-to-4 MUX, 4 x Electrodes, AC Power Supply, Digital Multimeter, Microcontroller, 2 x 3D Printed Test Rig (Container and OLR), Known Concentration Saline, Computer.

Goals	
Must	Test rig, 4-electrode network, independent AC power supply, digital voltmeter, MUX to switch configurations (30 Hz), Python visualizer (4 pixel display), microcontroller system integration and ADC.
Want	Higher speed configuration switching (~360 Hz), faster Python/C visualizer, custom AC power supply, high pixel display (if possible).
Would be nice	More electrodes (5), bigger test rig, full BVP solver.
Miracle	Even more electrodes, more MUXs, organic tests (arteries, veins, trachea). Full EIT with lung test (impossible).

Deadlines	
Week 2	Advanced proof of concept, test rig, device requirements.
Week 3-4	Hard-/software requirements, order MUXs, pseudo-code manual algorithm, AC power supply, digital multimeter.
Week 5	Manual MUX, manual algorithm (PyVISA).
Week 6 (Mid-report)	Microcontroller program, real-time algorithm.
Week 7 (Spring Break)	Catch-up week.
Week 8	Integrate shipped MUXs, continue programming.
Week 9	Complete system integration, debugging (device complete).
Week 10	Improve speed of microcontroller, MUX, visualizer.
Week 11	Finalize demo presentation and project report.
Week 12 (Demo)	Catch-up week.