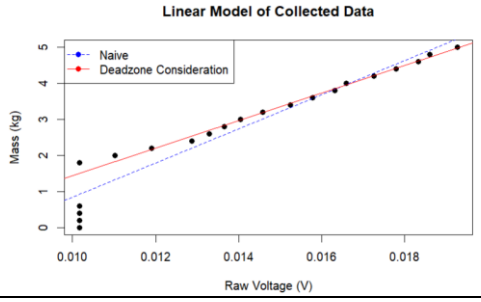
CC3501 weekly report example

Group number: 2
Team members: Hunter Kruger-Ilingworth, Thomas Mehes, Quentin Bouet
Week number: 9

Progress this week:

Task	Who did it?	What were the outcomes?	Who did the peer review?	What did you learn?
Work on Schematic	Quentin Thomas	Connect pins to RP2040 (debug, DAC, flash, SDI-12/tranceiver, dew point generator (DPG)). Add layout notes. Add decoupling capacitors. Fix errors. Clean up schematic. Add JLCPCB part numbers and verify some components. Move RP2040 power supply from USB to barrel jack.	Hunter	Layout adjustments were required, such as adding missing decoupling capacitors for the USB flash. Power tracks were thickened. Redundant capacitors were removed. Zener diodes were added to protect the RP2040 analogue inputs.
Submit board	Quentin	Uploaded the output files and BOM to Altium and verified the IC orientations	Thomas, Terence	No errors were present
Implement SD card library	Hunter	Followed the documentation provided by this repository to get reading and writing to a micro sd card to work on the microSD card breakout board (using the Fatfs api). It was wired up using the SPI protocol.	Quentin Laurance	Gave insight as to the plan for the software whilst the board is being fabricated (get a rudimentary i2c driver for the DAC working, even if we do not yet have something to prototype with)
Analyse the characteristics of the scale	Hunter	Discovered a significant deadzone in the client's scale apparatus when cross-referencing objects' mass using a scale. Accounting for this deadzone (red line) improved the accuracy of output voltage predictions, as confirmed by preliminary plotting. 	Hunter Laurance	Persistent vibrations in the apparatus due to its non-rigid construction. A software solution to average past data points was suggested to cancel out oscillations, rather than a hardware or digital low pass filter

Overall project tracking:

Week number	Milestones
4	Confirm project topic
5	Begin Overview and planning
6	Hardware design: Microcontroller, DAC, SD card, flash and usb interface
7	Hardware design: Voltage regulators, loadcell circuit layout and testing, SDI-12 testing and interfacing and Informal check with Laurance
8	Hardware design: write working SDI-12 code, start PCB layout Finalise draft schematic for Laurance to review.
9	Finish PCB layout and review to make sure all design rules pass. Implement fixes to the PCB. Final PCB design submitted on Friday to Terence
LR	Software: Begin development that doesn't require hardware testing Report: begin report writing
10	Hardware: Solder components to PCB and begin interfacing Software: Coding to receive data from I^2C DAC and optimise more SDI-12 sensor code
11	Software: data logging applications including averaging, variable sampling periods and clean exported data.
12	Verify all hardware functionality, perform testing of existing software on the physical board. Polish the software.
13	Implement final bug fixes. Write the report. Demo day during Friday lab.