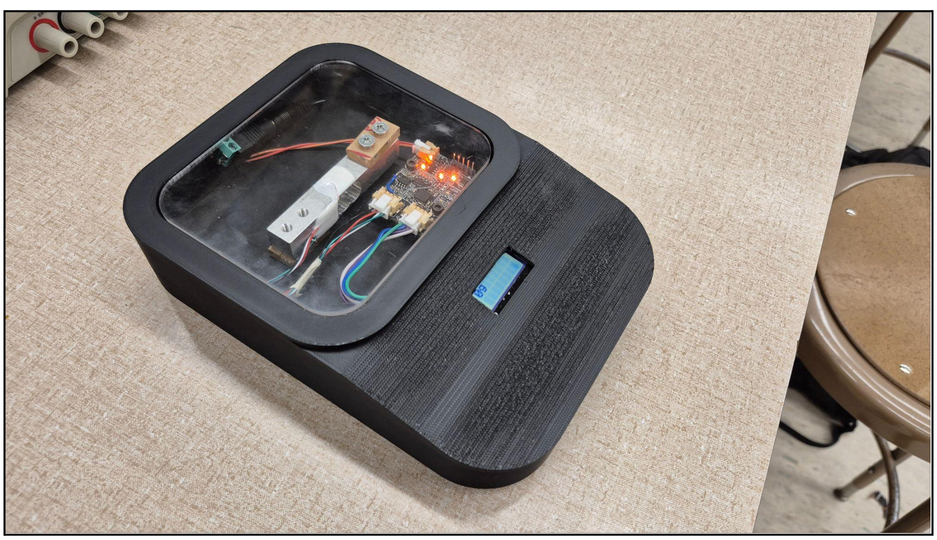
Junior Design Project Final Report

Manufacturing a Weigh Scale

TA: Hannah Veith

Josh Johnson, John Struyk, Alex Tong, Maichee Yang

May 15th, 2025



Design

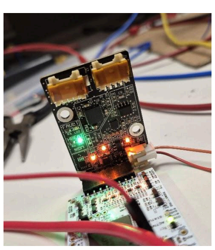
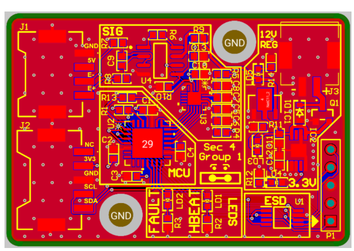
Overview

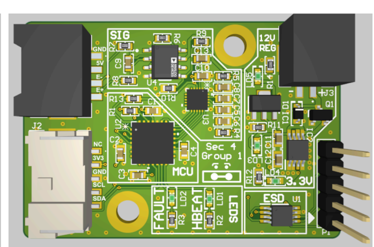
For our junior project design, we built a digital weigh scale that can measure up to 1,000 grams. We used a strain gauge load cell in a Wheatstone bridge configuration to detect weight and amplified the signal using a custom analog circuit. The signal was then filtered to reduce noise and sent directly to the internal ADC of the PIC24 microcontroller, which converted it to digital data for processing. The weight is displayed on an LCD screen, and the whole system is powered by a 12V battery and enclosed in a 3D-printed case.

Hardware

Our PCB is 4 layers designed in Altium Designer, with the outer 2 layers being signal layers and the inner layers being GND and power planes. The components are placed on only the top layer and we used only surface mount components, using the reflow soldering method to connect them all on. We used silkscreen to separate circuits into different sections that indicated their functionally, including a SIG section, MCU section, power section, LEDs section, and ESD section. This helped make the PCB easy to layout and understand, and easy to debug when things went wrong.

Some issues we encountered is when we changed our power for the load cell from 5V to 12V, we forgot to change the net of a pin on our connector, so it was left floating. We fixed this by soldering a lone wire from a nearby 12V pin to our load cell connector pin. We also had an issue with ordering the 5 pin connector we designed the board for, and instead had to order a 4 pin connector and solder it in place of the 5 pin. Finally, the other issue that we had was that we originally intended on using an external ADC for converting the output voltage of our load cell to a digital signal, but then ended up having issues communicating with it. We instead went with the backup option of connecting the output of the amplifier straight to a PIC24 ADC input. This worked, but caused some inaccuracy in our measurement as this was a 10 bit ADC rather than 16 bit.



Software

Software Functionality

Our firmware uses a timer-driven sampling loop to collect weight data from the PIC24's internal ADC at a steady 250ns interval. Each raw conversion is scaled with a zero offset and gain that was computed from known masses to tweak errors in the hardware's processing of the signal. To smooth out noise and mechanical bounce, we store the most recent 16 readings in a circular buffer. The displayed weight is the average of these entries, balancing stability against responsiveness.

All peripheral setup and I/O routines are organized into dedicated modules for ease of understanding and use. The ADC module handles channel selection and sampling timing, the LCD module drives the SSD1803A over I²C for neat character updates, and the main application loop simply averages buffered values, applies calibration, and issues display commands. In the main loop, the display refreshes every 100ms to provide smooth, flicker-free readouts.

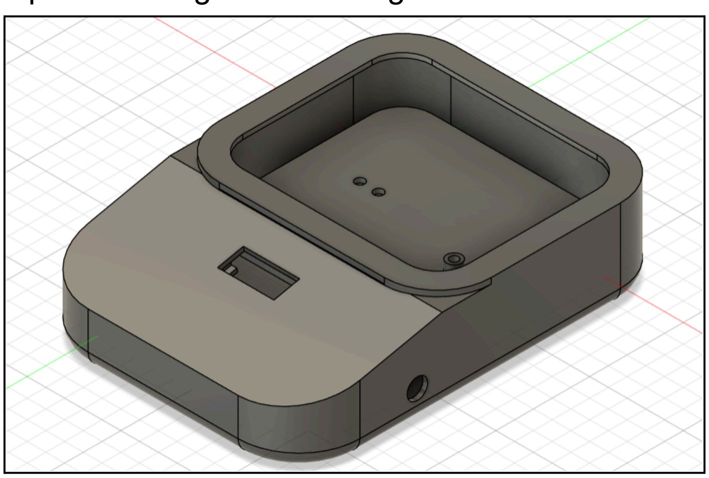
Challenges & Troubleshooting

- **External ADC via SPI:** We initially integrated the MCP3461 for its 16-bit resolution. Despite correctly toggling CS and issuing command bytes, its SDO pin remained high-impedance. Status-byte reads showed no communication, pointing to a SPI mode or timing misconfiguration. We couldn't fully isolate the issue under the schedule pressure. As a result, we reverted to the internal ADC and shifted to software calibration.

- **ADC Range & Amplifier Variance**: The internal ADC's 10-bit range and analog front-end gain didn't perfectly align with the hardware output, requiring us to tune using a calibration slope to achieve 0.1 g resolution.

Schematics

The layout of our 3D printed design for our weigh scale is shown below:



For our weigh scale, we created a custom 3D-printed box to hold everything together. We used Fusion 360 to design and model the enclosure. Inside the box, we split up space for a few key parts:

Battery Pack:
The 12V battery pack sits in the back of the enclosure. We placed it there to help balance the scale and keep it away from sensitive analog circuitry to avoid adding extra noise.

LCD Placement:
The LCD screen is mounted at the front and angled slightly upward so it's easy to read from above when you're using the scale. It can be slid into a slot that holds it firmly in place, even if the enclosure moves a bit during weighing.

Circuit Board Area:
Our custom PCB sits in the middle of the enclosure, mounted on standoffs to keep it from touching the enclosure floor. This helps protect it from vibrations and lets heat escape through ventilation slots below.

The slab on which weights are placed is able to be screwed into our 3D printed box, so that everything stays in place. A side hole was then added so that you could charge the battery pack without having to take everything out of the enclosure.

Conclusion

Overall, our design met the core specifications by successfully measuring weights from 0 to 1,000 grams with a resolution of 1 gram and displaying the values on an LCD. The scale was fully functional, portable, and accurate within our design goals. However, we did face some challenges throughout the process. One major difficulty was dealing with the very small output signal from the strain gauge, which required precise amplification and filtering to avoid noise and ensure reliable readings. We also ran into a problem when dealing with the external ADC and the SPI communication, which ultimately led us to using the internal ADC, but overall, we were happy with our final design.