UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Kitchen Dry Ingredient Tracker

Electrical & Computer Engineering

Team 43

By: Anju Jain, Sanjana Kumar, Nynika Badam

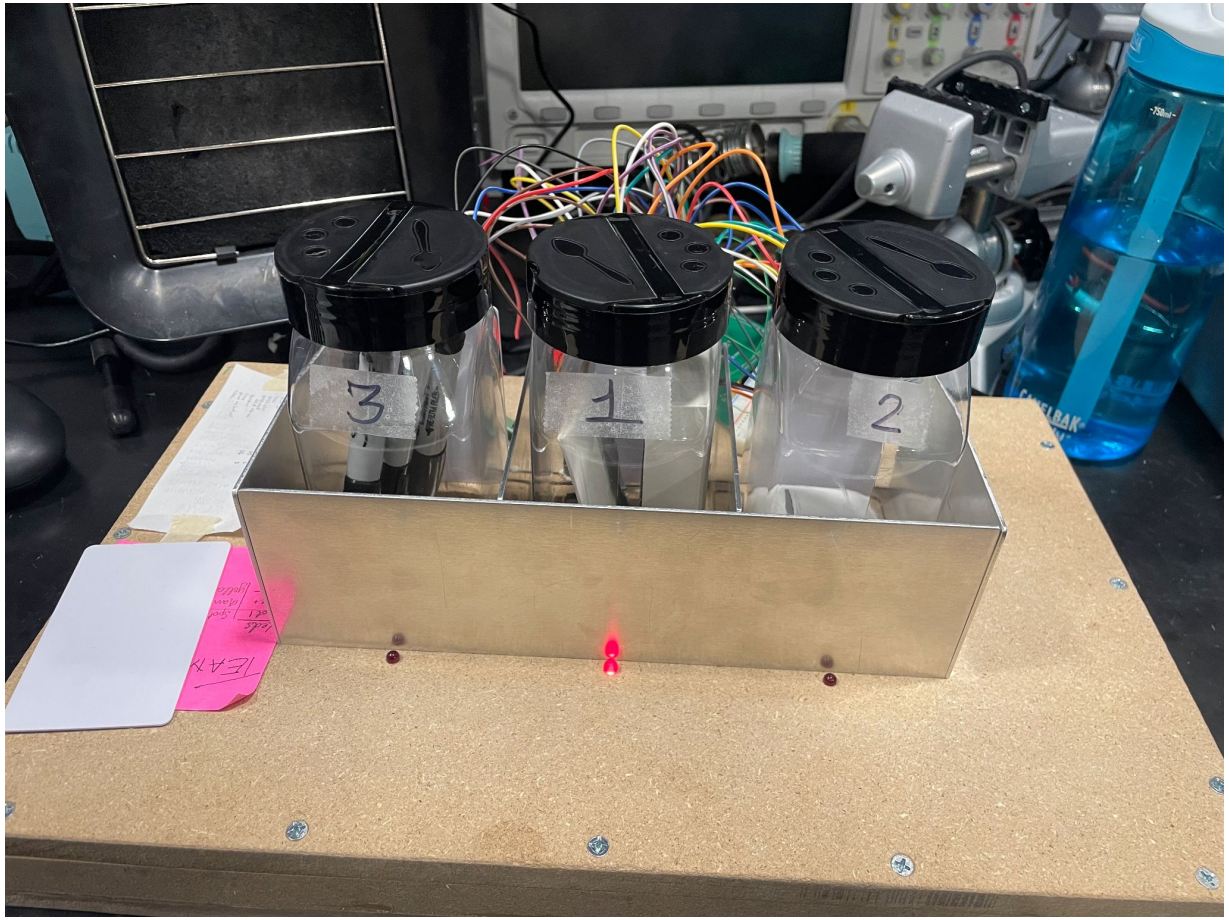
April 30th, 2024

Objective

"The primary grocery shopper in U.S. households made an average of 1.6 shopping trips per week in 2022" [4]

Project Build

Physical Design



The physical design contains 3 spots. Each spot is tied to a load sensor, a RFID reader, and LED.

High Level Requirements

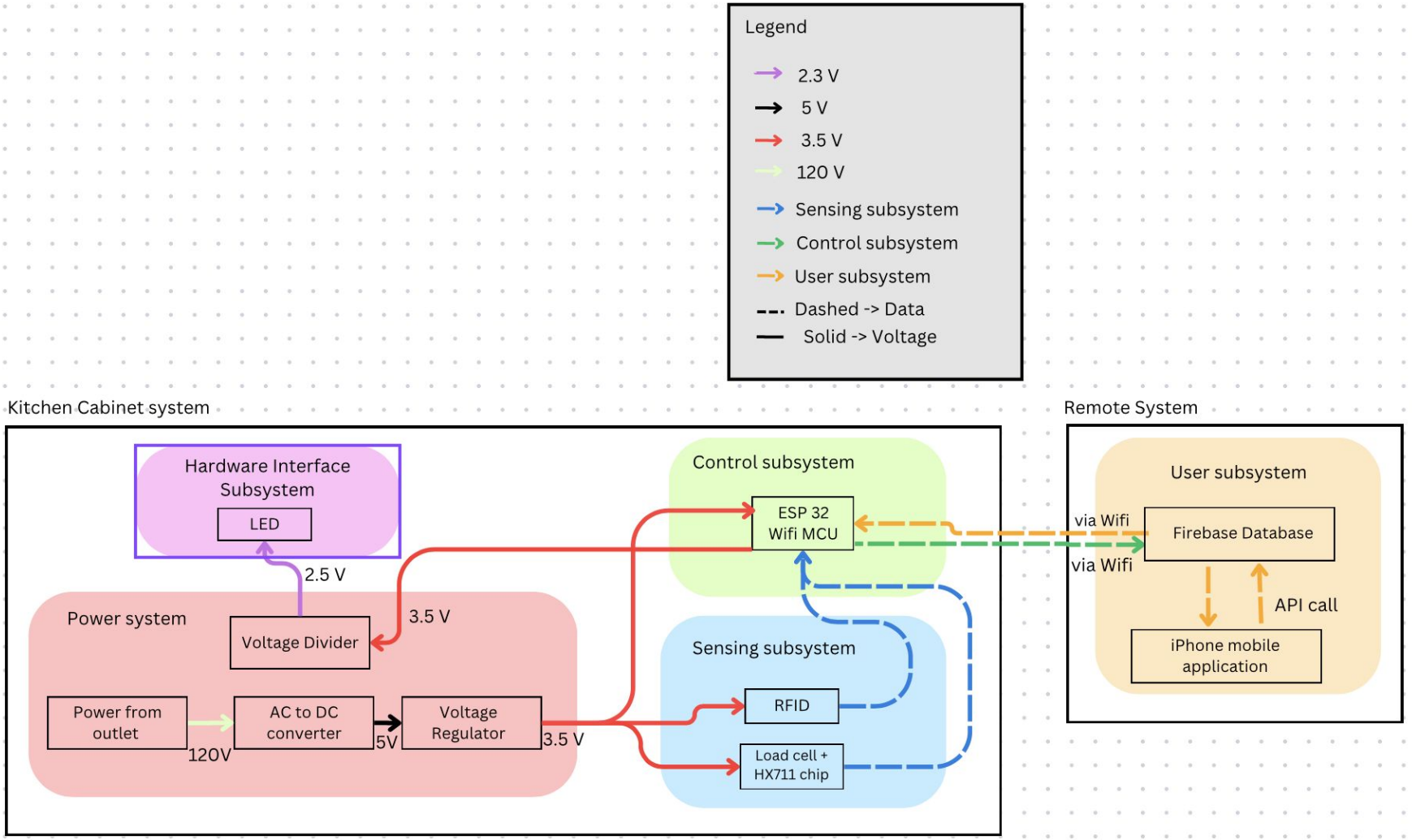
1. The MCU will pull weight data captured from the load cells every 30 seconds for the 3 spices tracked.

2. The app will have an “Ingredient Dashboard” and “Grocery List” interface.

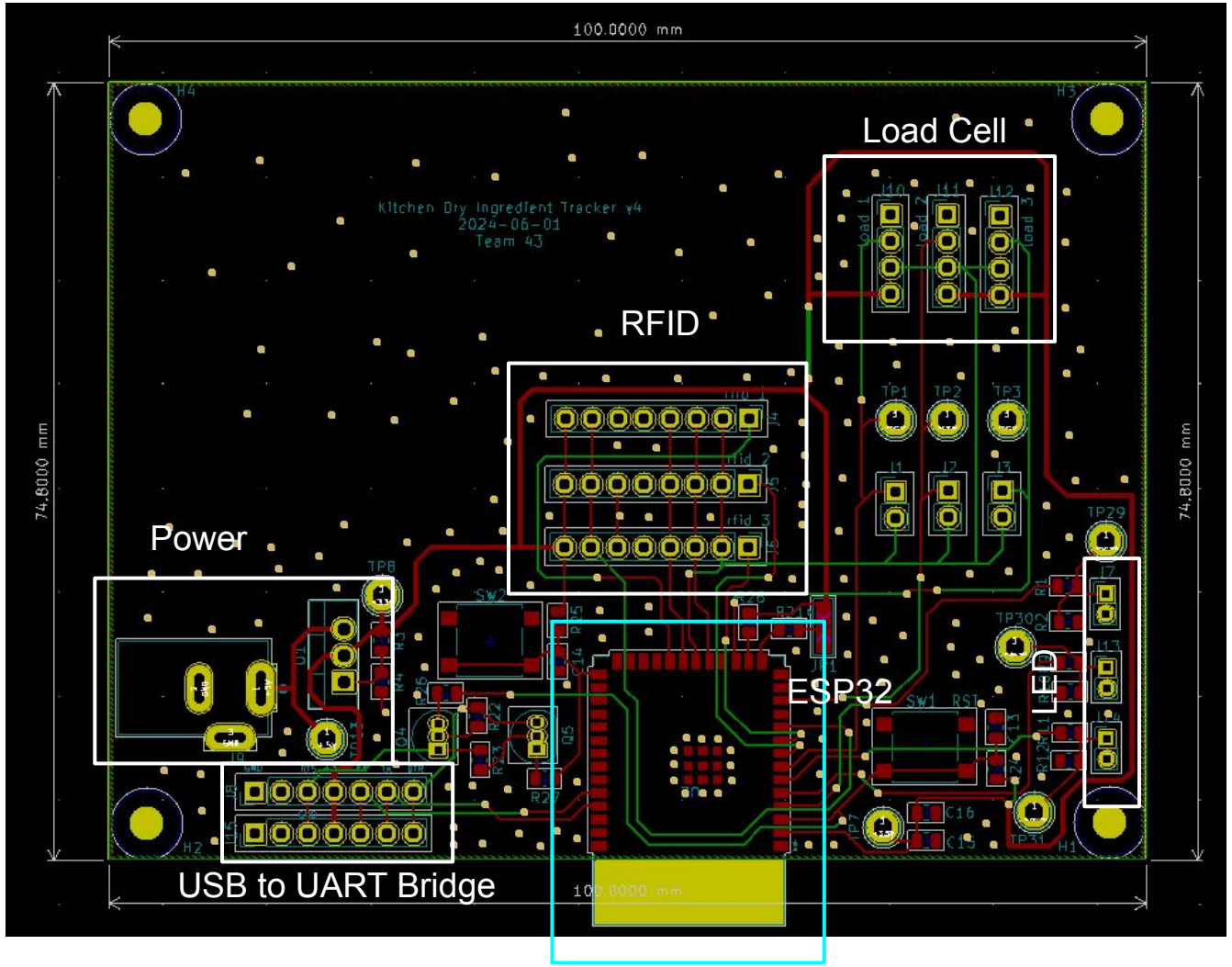
3. Users can place spice containers in any of the 3 spots. The container’s RFID will keep track of the container's position, allowing for the right comparison between the spice’s weight and its lower threshold weight.

Subsystems

Block Diagram

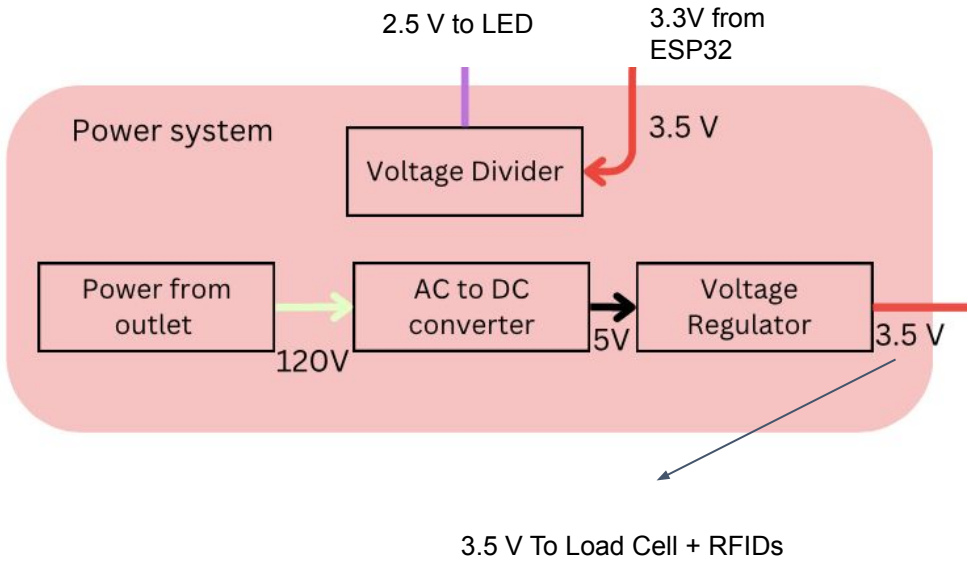


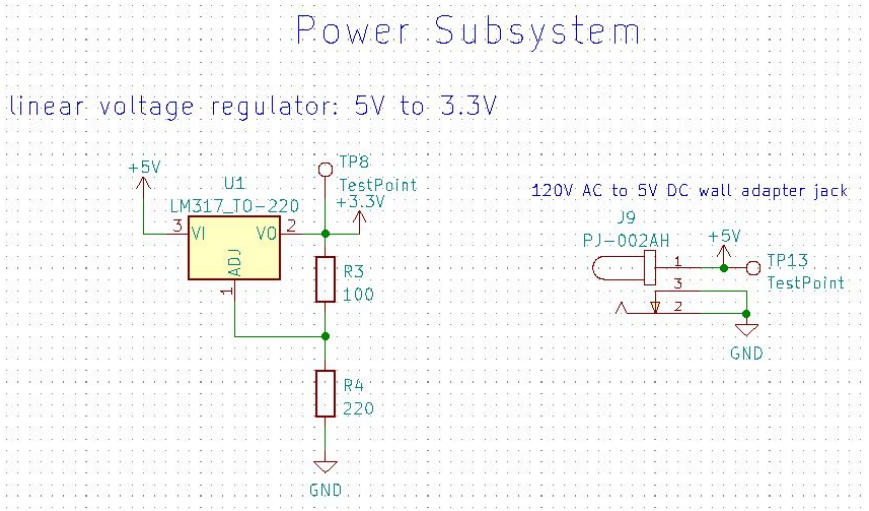
PCB


100.0000 mm
74.8000 mm
Kitchen Dry Ingredient Tracker v4
2024-06-01
Team 43
Load Cell
RFID
Power
ESP32
LED
USB to UART Bridge

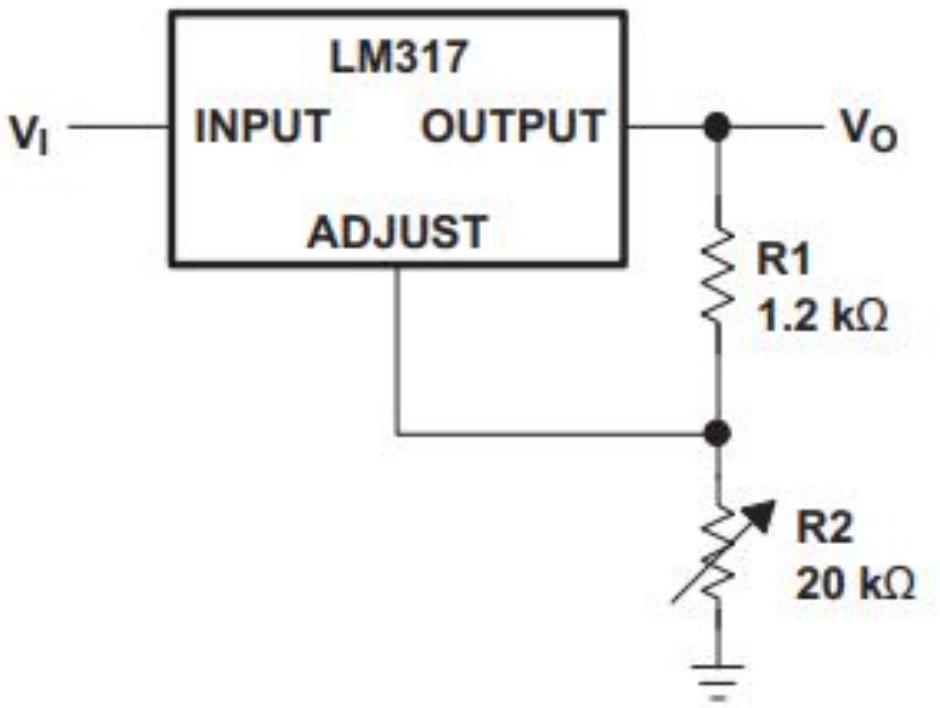
Overview

- Wall power adapter converts 120V AC → 5V DC
- Linear voltage regulator drops the 5V → 3.5V for HX711 chips, RFIDs, and ESP32
- Voltage divider steps down the 3.3V → 2.5V for the LEDs.





Incorrect Voltage Drop from Wall Power Adapter to Sensors and ESP32



LM317 Datasheet [3]

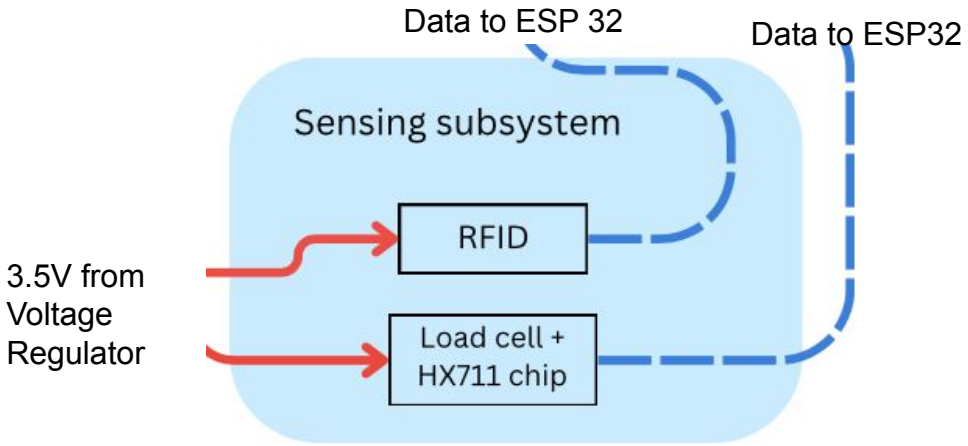
Design	R1	R2	Vout
Original	390 ohm	220 ohm	1.75V
Revised	100 ohm	220 ohm	3.5V

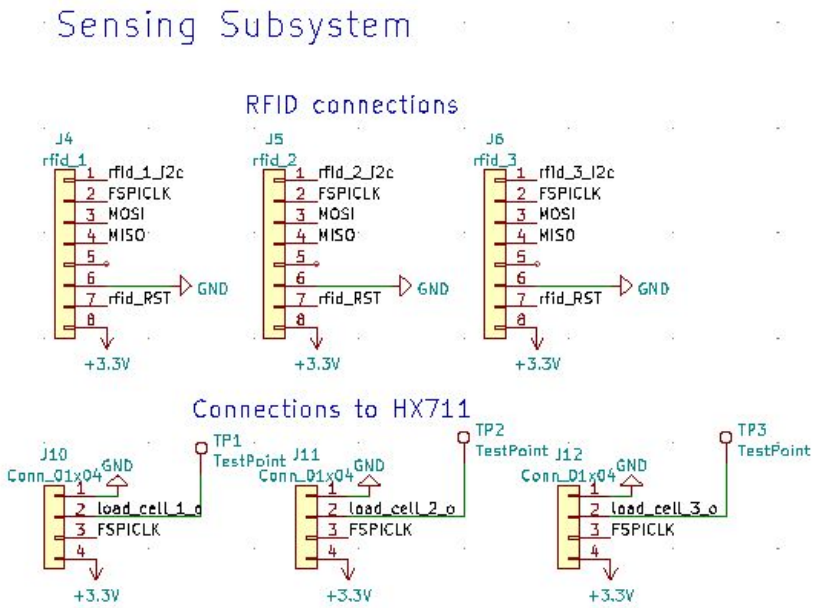
Power Subsystem

Wall adapter outputs 5V ± 0.25V.	1. Connect (+) probe of the multimeter to pin 1 of the barrel jack connector and (-) probe to the pin 2 of the barrel jack connector. 2. The multimeter should read 5V ± 0.25V.
LM317 outputs 3.3V ± 0.3V.	1. Connect (+) probe of the multimeter to Vout of the LM317 and (-) probe to the GND. 2. The multimeter should read 3.3V ± 0.3V.

Overview

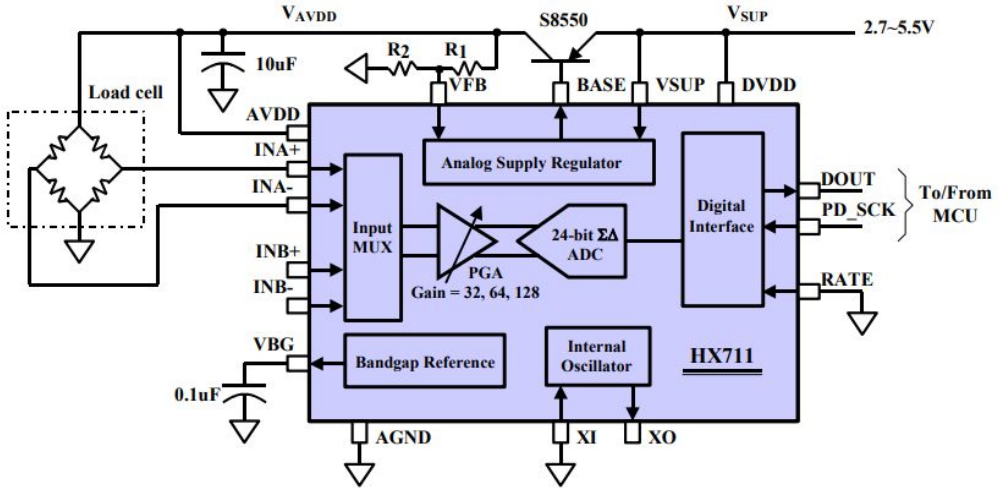
- RFIDs allow users to place spice containers in any slot
- Load cells measure weight of the spices and send the information to the ESP32 via the HX711.





Incorrect Operating Voltage

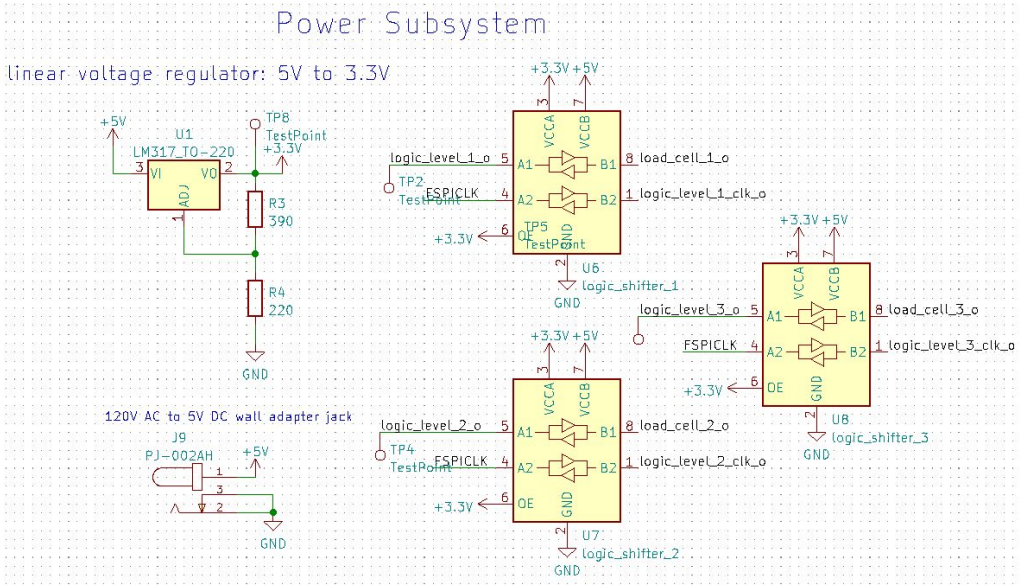
“ Channel A can be programmed with a gain of 128 or 64, corresponding to a full-scale differential input voltage of ±20mV or ±40mV respectively, when a 5V supply is connected to AVDD analog power supply pin.” [1]



HX711 Datasheet [1]

Incorrect Operating Voltage

- The HX711 depends on the ESP32's CLK signal to output the weight from the load cells bit-by-bit.
- HX711 and ESP32 operate at the same voltage
  - logic level shifters were unnecessary.



PCB v1 Schematic

Load Cell

One Load Cell + Firebase Connection	Load Cell: 1. Measure out 50 grams of spice using a scale (baseline weight). 2. Place the 50 grams of spice on the load cell to see if the value outputted is the same as the baseline weight with a resolve of 10 grams. Load Cell + Firebase Connection 1. Send data from load cell to ESP32 via logic level chip shifters. 2. Send the collected data from ESP32 to the Firebase database. 3. Check to see if collected data is present in the database.
Reading Multiple Load Cells Consecutively	Put three different weights in each spot. Move them around to ensure that each load cell measures the appropriate weight.
Vcc = 3.3V± 0.3V	Take a multimeter and attach the positive probe to Vcc of the HX711 and the negative probe to GND. The multimeter needs to read 3.3V± 0.3V.

RFID

Only the 3 valid RFID tags will be read and recognized by the device	To ensure only valid tags are read, a fourth tag will be used as an edge case. 1. Place an invalid card at a spot, the serial monitor prints out the tag number, the spot the card is in and that it is an invalid card.
Reading tags despite one or multiple spots not having tags.	1. Place 1 tag at spot 1 and keep spot 2 and 3 without any tags. 2. The serial monitor prints out the tag number, the spot the card is in and whether tag is valid/invalid.
Vcc = 3.3V± 0.3V	1. Take a multimeter and attach the positive probe to Vcc and the negative probe to GND. 2. The multimeter needs to read 3.3V± 0.3V.

Challenges for RFID + Load Cell

- RFID & load cell using same CLK + RFID reset using GPIO17 caused problems with SPI protocol
  - load cell to read -234 and 668 no matter the item places on the load cells

Solutions

1. Separated the load cell readings and the RFID readings.
2. Set RFID CLK pin separate from load cell CLK pin.
3. Through trial and error, found changing the RFIDs reset pins to GPIO 38.

RFID + Load Cell + Firebase

Load cell reading is displayed on serial monitor for valid RFID tag at particular spot.	1. Place a valid RFID tag and weight at any spot. The serial monitor prints out the tag number, the spot the card is in, confirms the tag's validity, and weight of the item in grams. 2. Place an invalid RFID tag and weight at any spot. The serial monitor prints out the tag number, the spot the card is in, and that the tag is invalid.
Load cell readings should be reflected in Firebase database	1. Place a valid RFID tag and container of spice at any spot. Each container is tied to a specific RFID tag. 2. The weight of the container should be reflected in both the app's Ingredient Dashboard page and Firebase database.

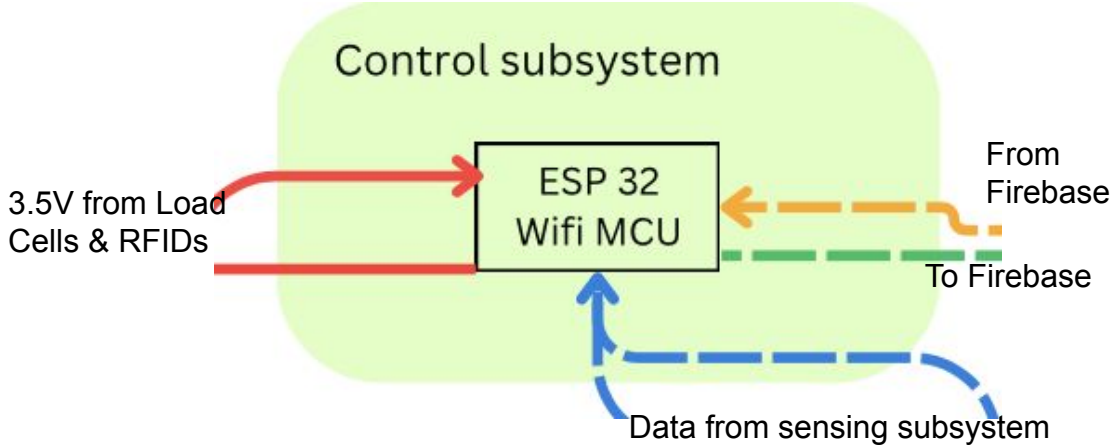
Sensing Subsystem

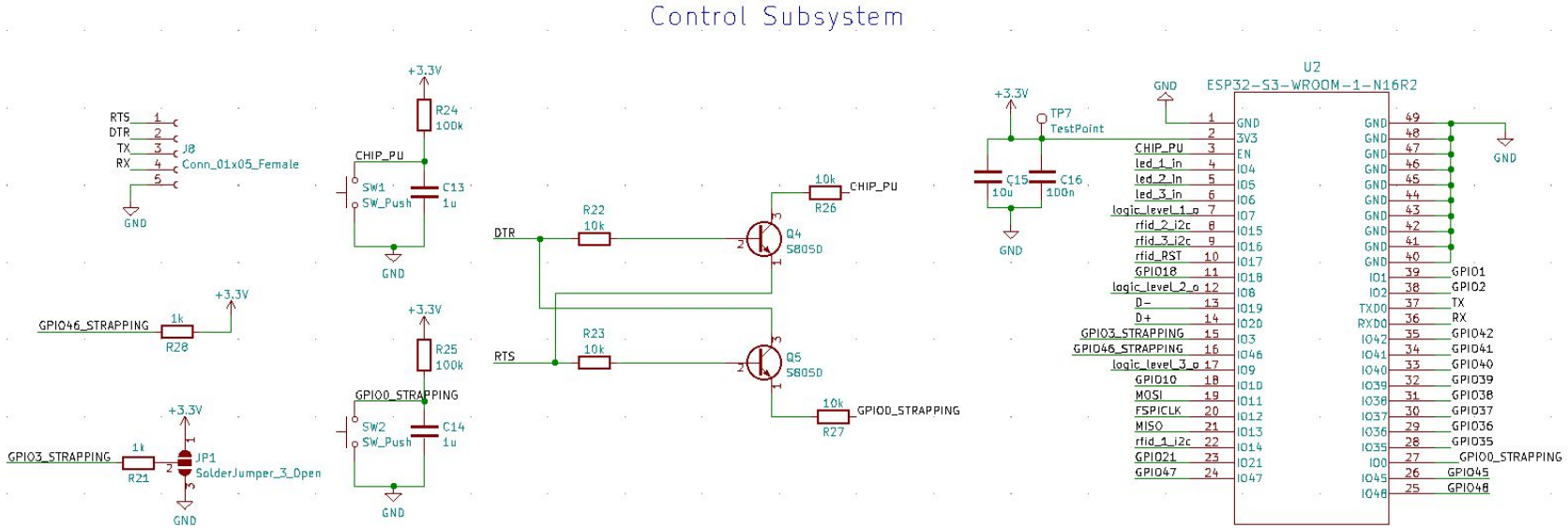
RFID + Load Cell + Firebase

```
Spot 2 Card UID: D5 8F FE 29 Valid tag at spot: 2
loadcell reading 214.00
true:Spot 3 Card UID: 96 8E EA AD InValid tag at spot: 3
```

Overview

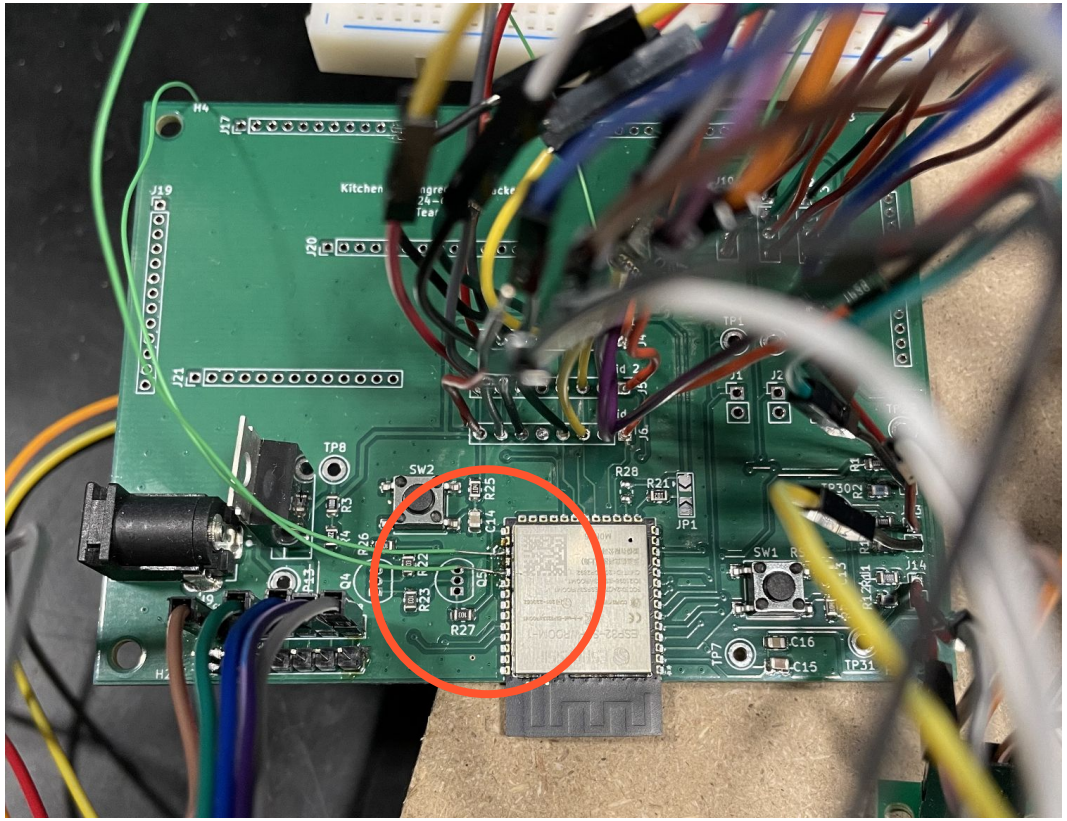
- Connects the user interface subsystem to the sensing and hardware interface subsystems
- Uses ESP32 Wifi protocol





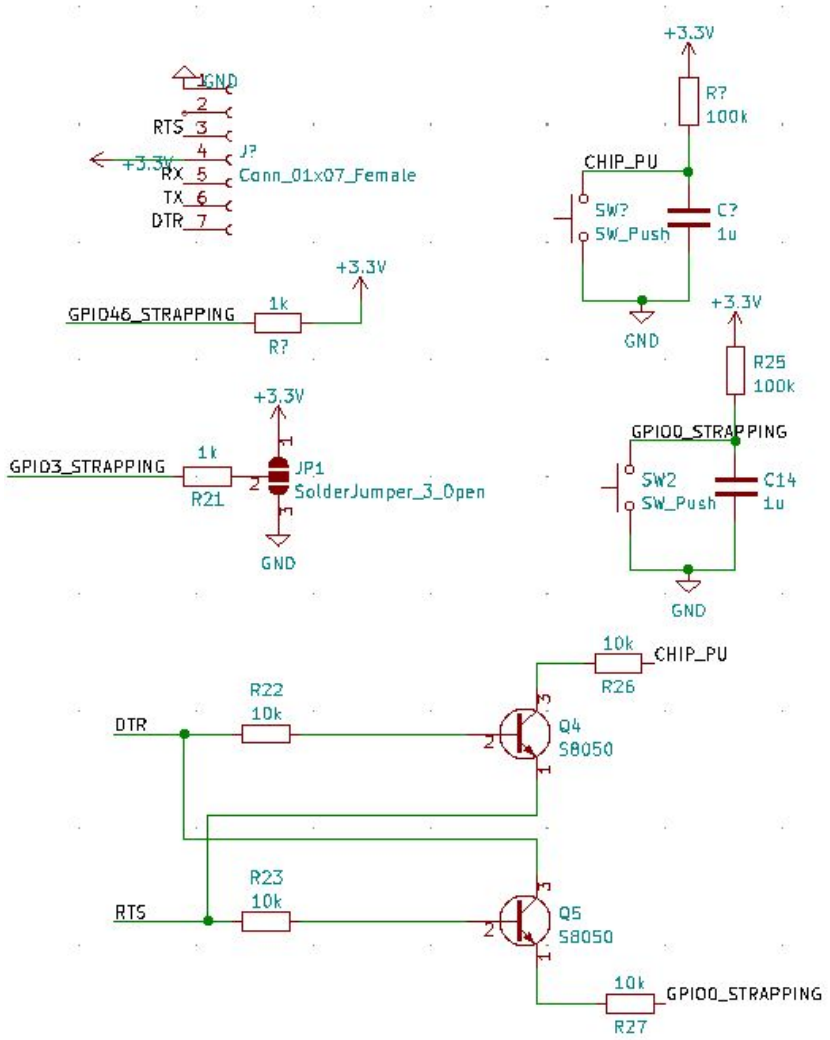
Challenges for ESP32

1. Initial ESP32 checked out dev board did not work.
2. New dev board worked for 2 weeks and then broke
    a. Final round of testing on PCB



Incorrect Footprint for S8050 BJT in PCB

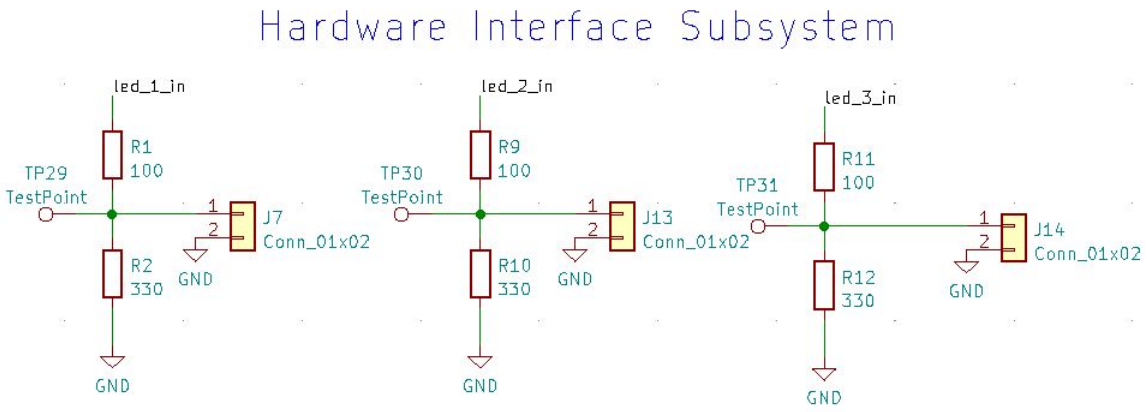
- ESP32 schematic uses Bare Minimum schematic from course website
- Used through hole instead of SMD footprint for BJTs

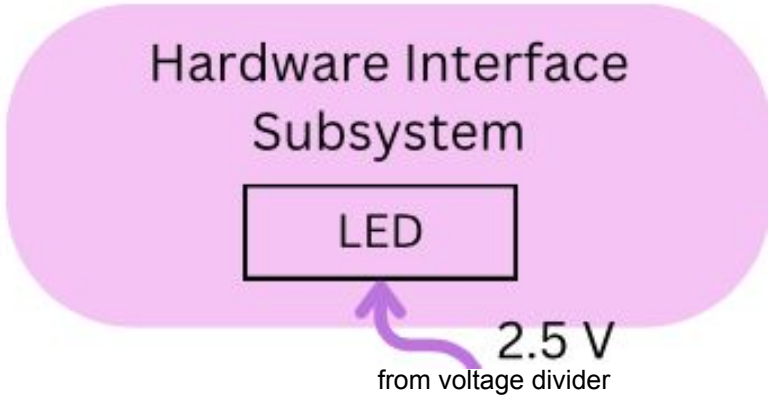


ESP32 Bare Minimum Schematic

Overview

- Operating voltage: 2.3V±0.3V at each spot.
- Provides visual representation for spices below the user specified lower threshold.





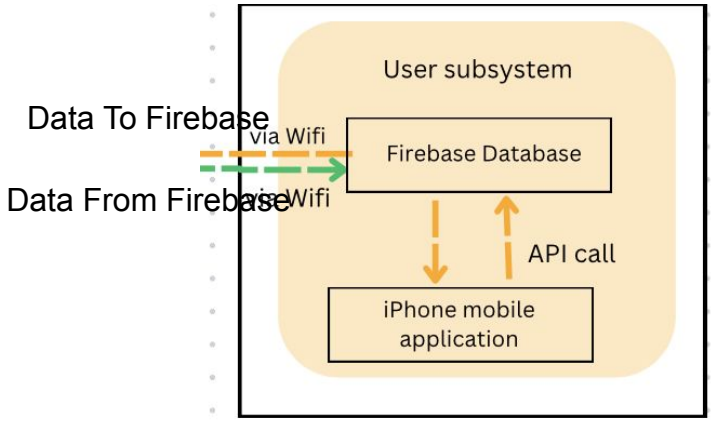
LEDs

The LED switches on at a spot when the spice's weight is below its lower threshold.	Place a valid RFID tag and container of spice at any spot. The container of spice should weigh below its lower threshold. The Arduino code pulls the spice's lower threshold from Firebase database. If the lower threshold is greater than the load cell reading at that spot, the LED at the spot is switched on. The code also updates the runningLow flag of the spice to True in Firebase. Add spice to the container such that its weight is above its lower threshold. The Arduino code pulls the spice's lower threshold from Firebase database. Since the lower threshold is lower than the load cell reading at that spot, the LED at the spot is switched off. The code also updates the runningLow flag of the spice to False in Firebase. Replace the valid tag with an invalid tag. If the spot's LED was switched on, the LED switches off.
Vcc = 2.3V± 0.3V	Take a multimeter and attach the positive probe to positive lead of LED and the negative probe to GND. The multimeter needs to read 2.3V± 0.3V.

User Subsystem

Overview

- Uses Firebase database for data storage

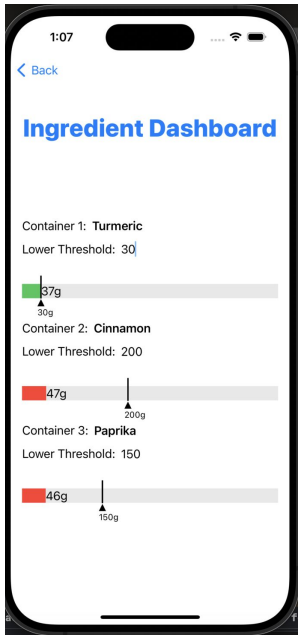


App Features:

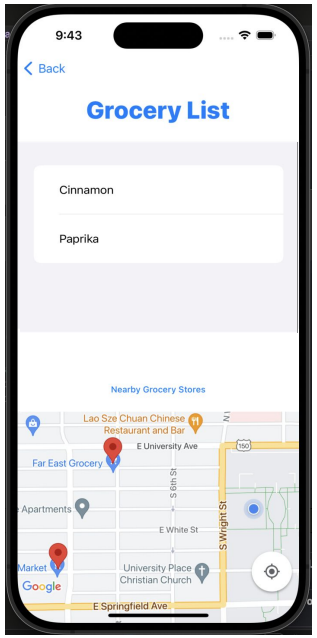
- Track weights of each spice
- view their digital grocery list
- notifications when ingredients run low
- view nearby grocery stores.

App Screens

- Ingredient Dashboard
- Grocery List



Screen 1

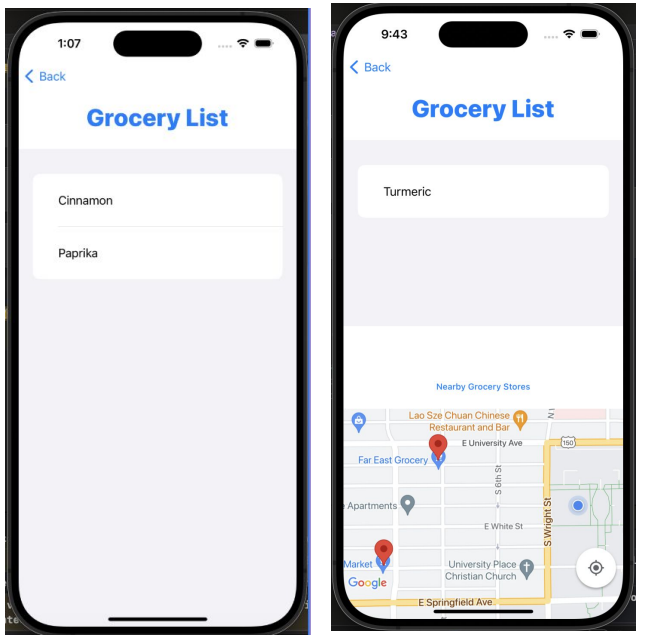


Screen 2

Screen 3

Location Services

- Simulating an active journey within the iOS simulator was beyond its capabilities - required an Apple Developer account and APN certification.
- Alternative: created a map view within the grocery list screen that show grocery stores nearby to a user's location



Before After

Notifications

- Necessary to create an Apple Developer account which requires a paid subscription and APN (Apple Push Notification) certificate for many methods of enabling notifications
- Strictly use the iOS simulator feature in Xcode - local notification generation

Ethical Considerations

The app's location services feature requires the consideration of ethics involving location tracking.

- Current implementation doesn't tuse active location of a user
- Ideal product: follow Google's Privacy and Terms
  - Employ robust security measures to protect all user data
  - Option for users to opt out of location tracking - keeping the power in the user's hands [2].

iOS App

Progress bar turns from green to red in Ingredient Dashboard View if the measured weight is below the user set threshold.	For ingredient 3, set the lower threshold as 50 grams in app view. Change measured weight to 70 grams inside Firebase json file. The preview of the app should show a green progress bar. Now change the measured weight to 20 grams inside Firebase json file. The preview of the app should show a red progress bar, for ingredient 3.
Ingredient Dashboard View displays an error message for invalid numbers entered into the Lower Threshold subfield. Lower Threshold must be a whole number in the range of 0 - 500 grams.	For ingredient 1, set the lower threshold as 100 grams. The Ingredient Dashboard View should not show an error message and the progress bar should have a marker at 100 grams. Now, set the lower threshold for ingredient 1 as 2.5 grams. The Ingredient Dashboard View shows an error message and its progress bar turns gray.
A notification is generated and the ingredient name and google maps view will appear in the dynamic grocery list view when a measured weight falls below the user set threshold.	For ingredient 2, set the lower threshold to a value lower than the current tracked weight in the app view. A notification should appear in the notification center of the phone immediately. The name of ingredient and google maps view of nearby grocery stores should appear in the grocery list

Conclusion

What we Learned

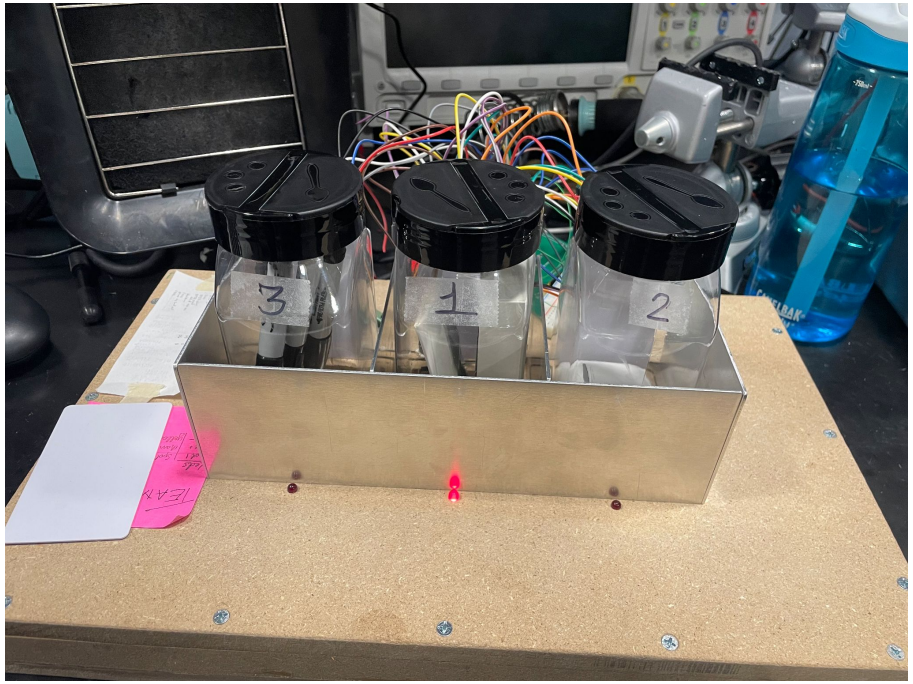
Technical

- Design and troubleshoot PCBs.
- Program various sensors and the ESP32 through Arduino.
- Build a functional iOS app using SwiftUI.
- Integrate Firebase database with ESP32 + app

Overall

- How to find a problem to tackle
- Create achievable timeline to reach expected goals

Redesign of Project



- Adjust timeline
- Use correct footprints in PCB
- Remove metal casing from physical design

Recommendations for Further Work

- Reduce fire hazards
  - Add thermal sensors
  - Cover exposed wires
  - Place PCB within an enclosure
- Create HX711 module from scratch
- Design app to be more user friendly and responsive

Citations

[1] AVIA Semiconductor. "24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales." HX711 datasheet.

[2] “Privacy and Terms.” Google. Accessed: April 24th, 2024. [Online]. Available: https://policies.google.com/privacy?hl=en-US

[3] Texas Instruments. "LM317 3-Terminal Adjustable Regulator." LM317 datasheet. Sept. 1997 [Revised Apr. 2020].

[4] T. Ozbun. "Grocery shopping: U.S. consumers' weekly trips per household 2006-2022." statista.com. Accessed: Feb. 14, 2024. [Online.] Available: https://www.statista.com/statistics/251728/weekly-number-of-us-grocery-shopping-tripsper-household/

Questions

The Grainger College of Engineering

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN