UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

SpoilSense

Team 25

Electrical & Computer Engineering

Azim Shad (azimms2)
Sarthak Shah (spshah8)
Vikram Harish (vharish3)

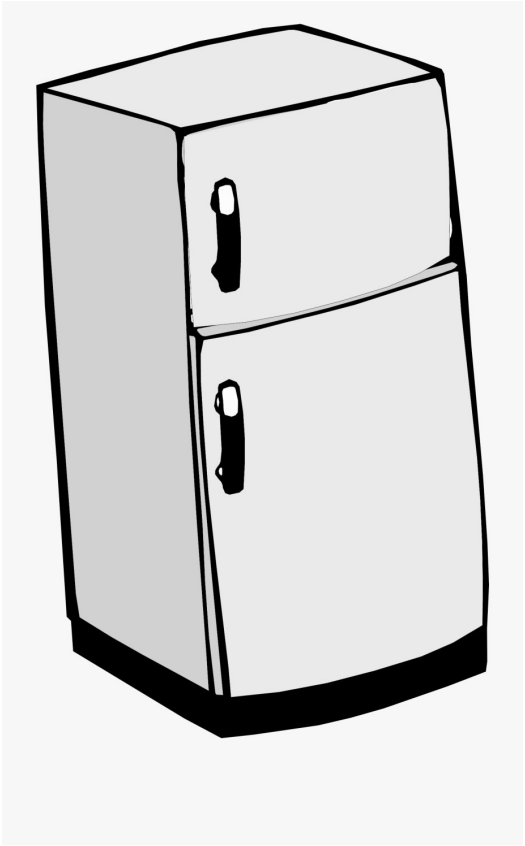
12/10/2024

Overview

1. Motivation
2. Solution
3. Implementation
4. Results
5. Looking Back
6. Looking Ahead
7. Questions

Overview

1. **Motivation**
2. Solution
3. Implementation
4. Results
5. Looking Back
6. Looking Ahead
7. Questions



Problem

Food spoilage has significant global monetary, environmental, health, and social implications

Understanding the Problem

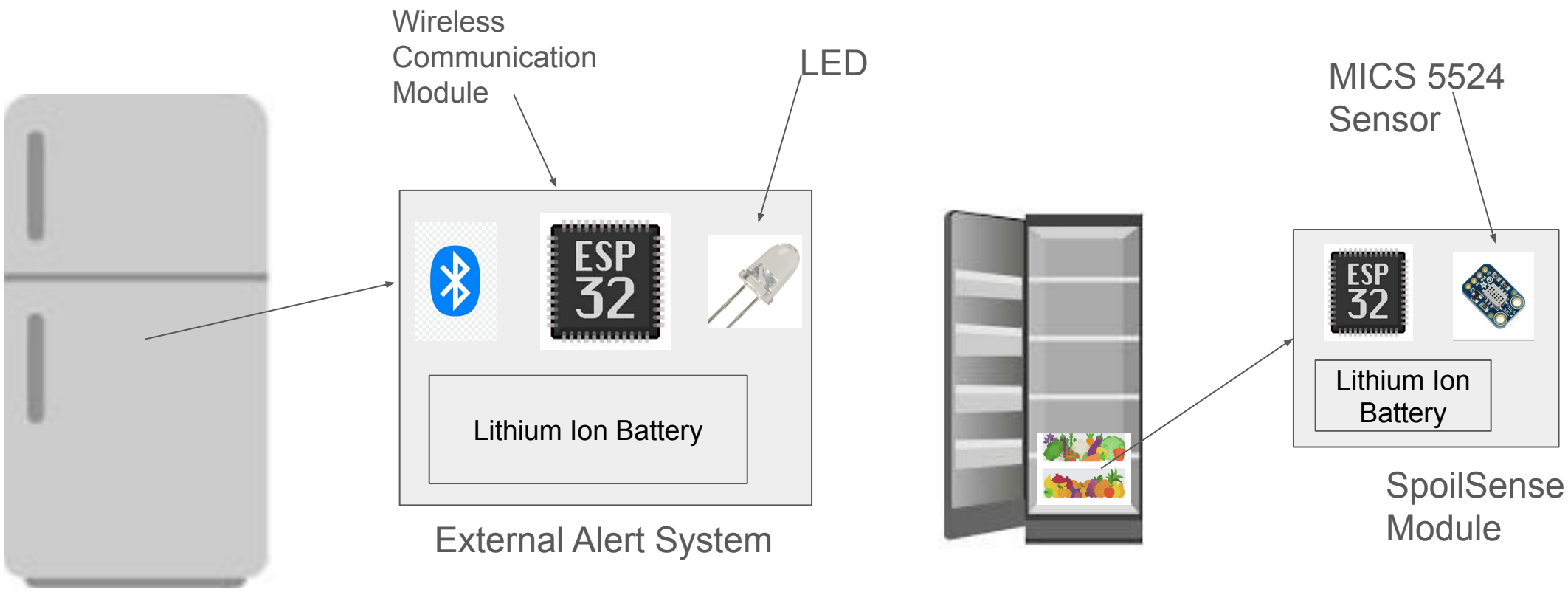
- Food spoilage is driven by the natural enzymatic processes within the food itself
- The break down of complex molecules into simpler compounds produces gases as byproducts
- Gases Include:
  - Ethylene
  - Ammonia
  - Methane
  - Carbon Dioxide

- The gases released by one spoiling food causes surrounding produce to spoil as well

Overview

1. Motivation
2. **Solution**
3. Implementation
4. Results
5. Looking Back
6. Looking Ahead
7. Questions

Spoilsense


Wireless
Communication
Module
LED
MICS 5524
Sensor
ESP
32
Lithium Ion Battery
External Alert System
ESP
32
Lithium Ion
Battery
SpoilSense
Module

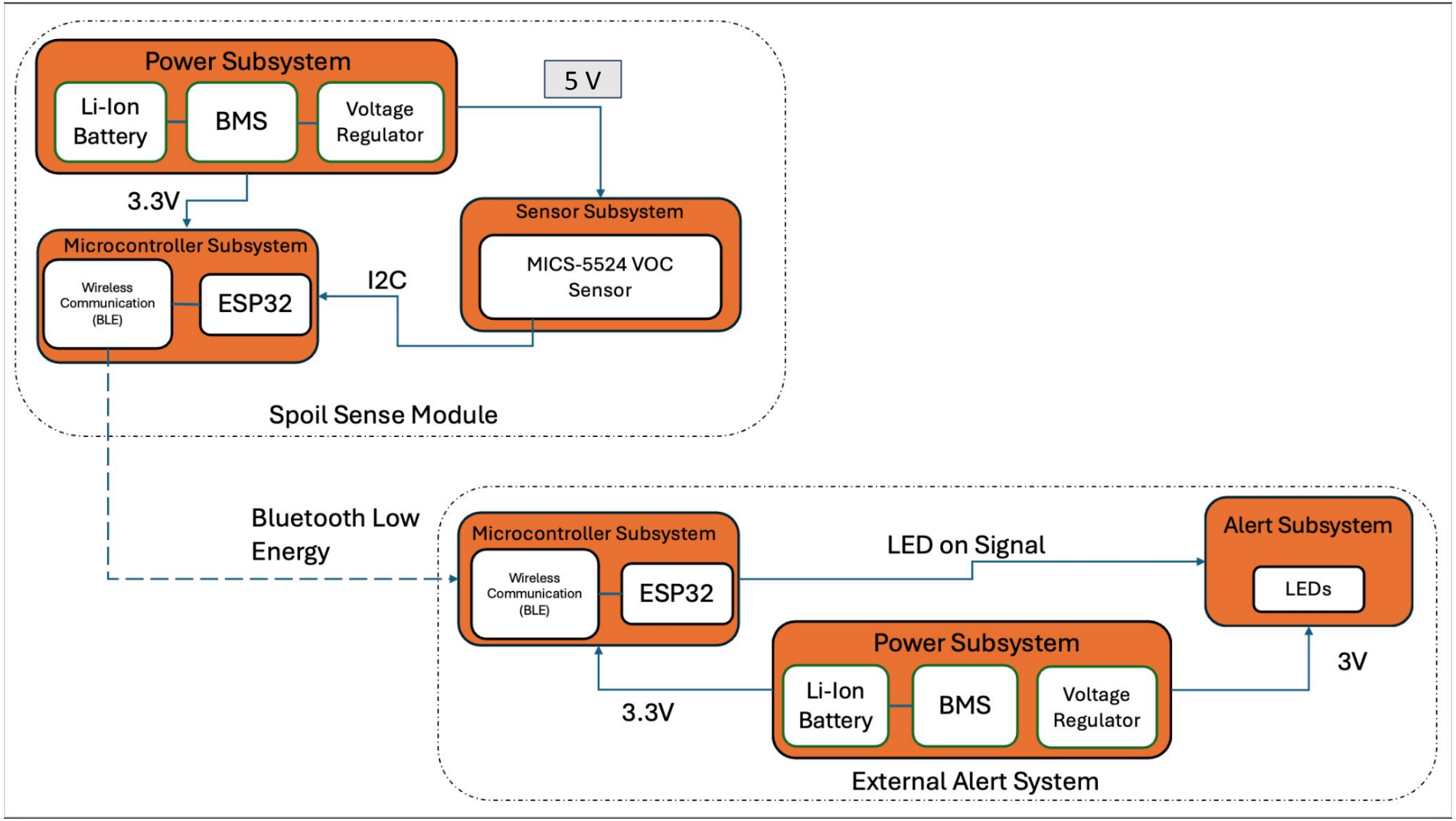
Why SpoilSense

- Not all fridges are smart
- SpoilSense gives the ability to make fridges “smarter” by allowing them to detect detect spoiled food and alert users

Overview

1. Motivation
2. Solution
3. **Implementation**
4. Results
5. Looking Back
6. Looking Ahead
7. Questions

Block Diagram



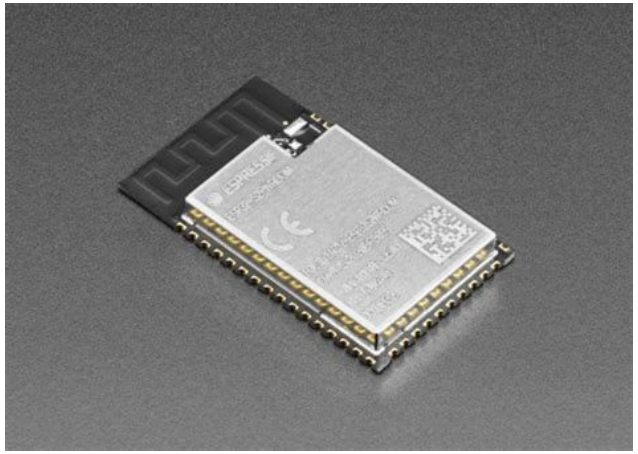
Microcontroller Subsystem

Requirement

- The microcontroller subsystem of the SpoilSense module is the central processing unit that manages sensor data and coordinates wireless communication.

Changes

- This subsystem did not experience significant changes throughout the development of our project.
- We proceeded with our original plan to use the ESP32



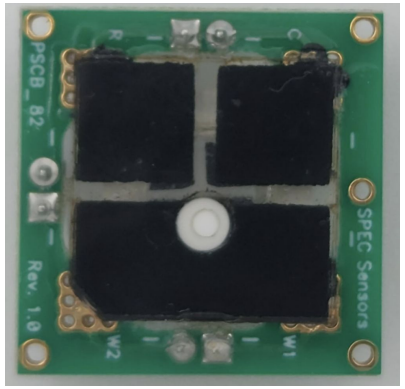
Sensor Subsystem: SpoilSense Module

Requirement

- The sensor subsystem is responsible for monitoring environmental conditions in the produce drawer of a refrigerator.
- In particular, it is responsible for detecting the presence of various volatile organic compounds in order to accurately recognize potential spoilage.

Changes

- We first tried to use an ethylene sensor, but ran into logistical issues.
- Then we tried to use a BME680 to detect CO2, but ran into issues with getting accurate readings.
- Finally, we settled on the MICS-5524 which can detect various VOCs



Power Subsystem

Requirement

- The power subsystem is designed to supply energy to all components, ensuring long-term operation with minimal maintenance.
- It includes a battery that powers the sensors and microcontroller. A linear voltage regulator ensures the correct voltage is supplied to the various components, providing 3.3V for the microcontroller and 5V for the sensors.

Changes

- Original design had a 9V battery for longer battery life
- Final design includes a 6V battery such that the voltage regulators temperature does not get too hot

Alert Subsystem

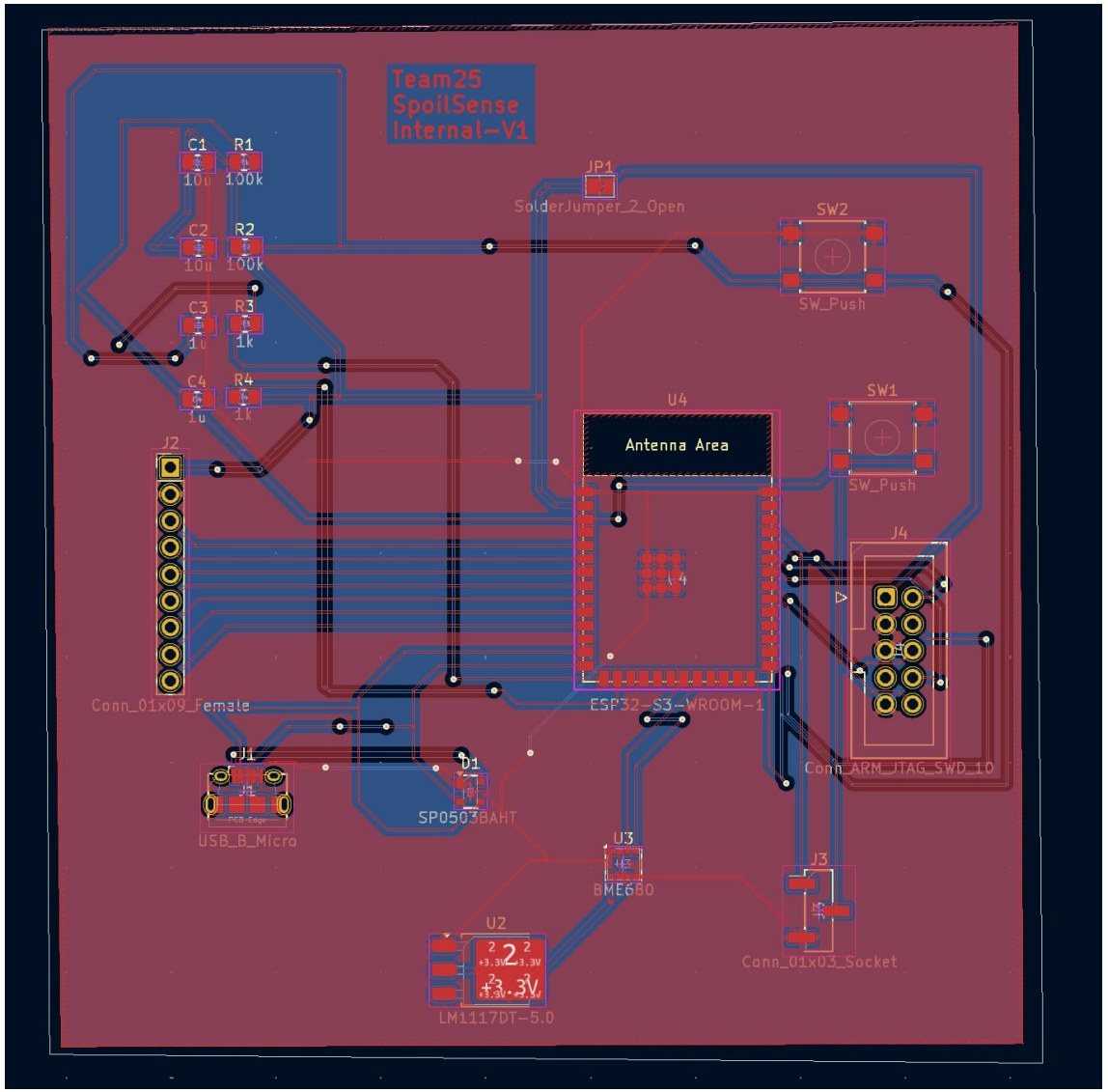
Requirement

- The alert component consists of 1 LED indicator which provide real-time visual alerts to the user, turning on based on spoilage or no spoilage, detected by the internal sensor
- The LED is controlled by the microcontroller inside the external system which is wirelessly connected to the microcontroller of the SpoilSense module, receiving signals via Bluetooth Low Energy (BLE) when spoilage thresholds are met

Changes

- Original design included 3 LEDs which turned on based on ranges of spoilage detection, was hard to implement because fluctuations in readings could cause LEDs to toggle on/off unpredictably, reducing reliability
- Final design reduced down to 1 LED which turns on once threshold is reached. This is simpler to implement and test/verify

PCB Design - SpoilSense Module (Internal)



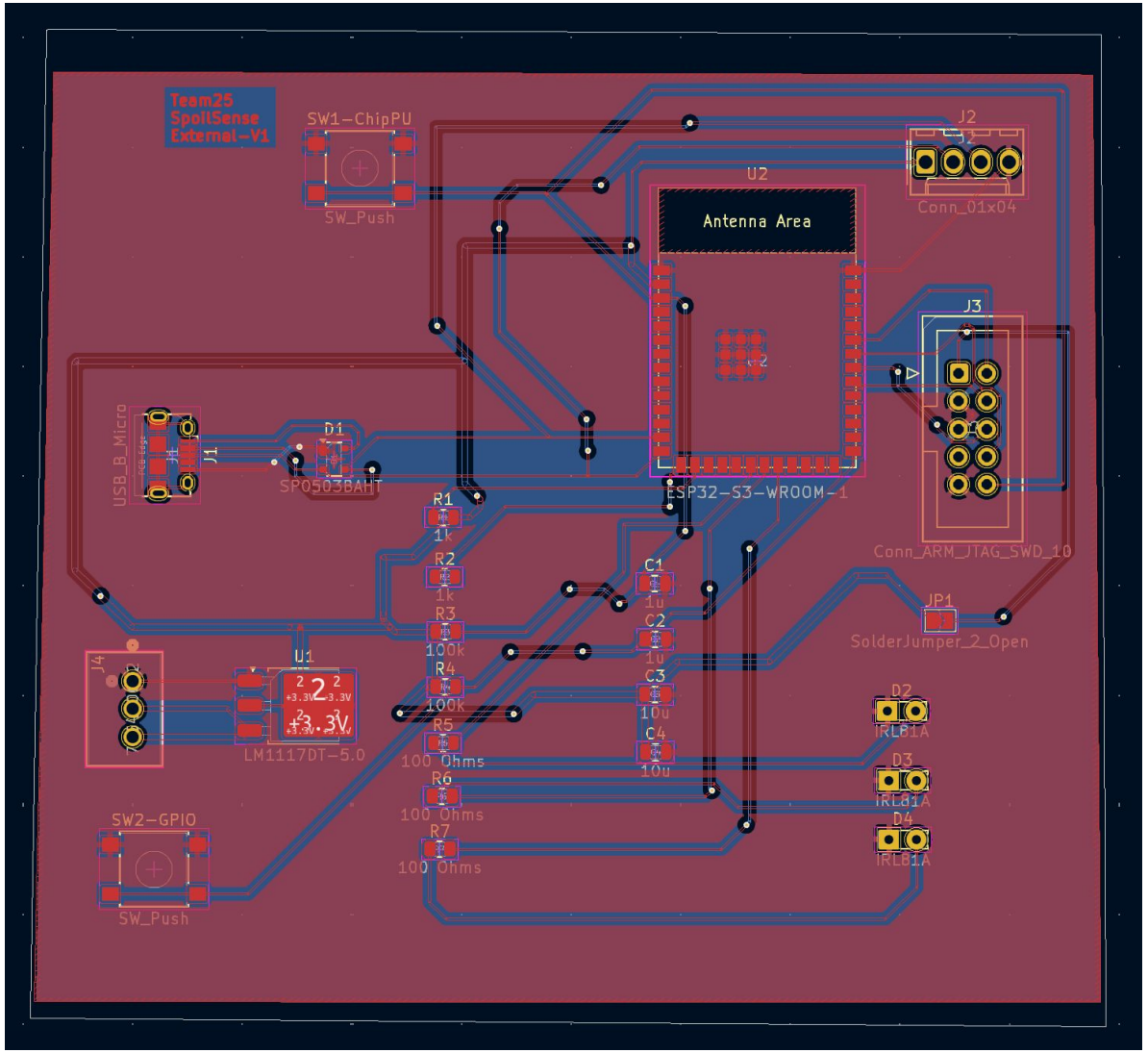
Major Components:

- ESP32 Microcontroller
- MICS-5524 VOC Sensor
- Power Distribution System

Updates + Changes:

- Inadequate power supply
- Updating Sensors

PCB Design - Alert Module (External)



Major Components:

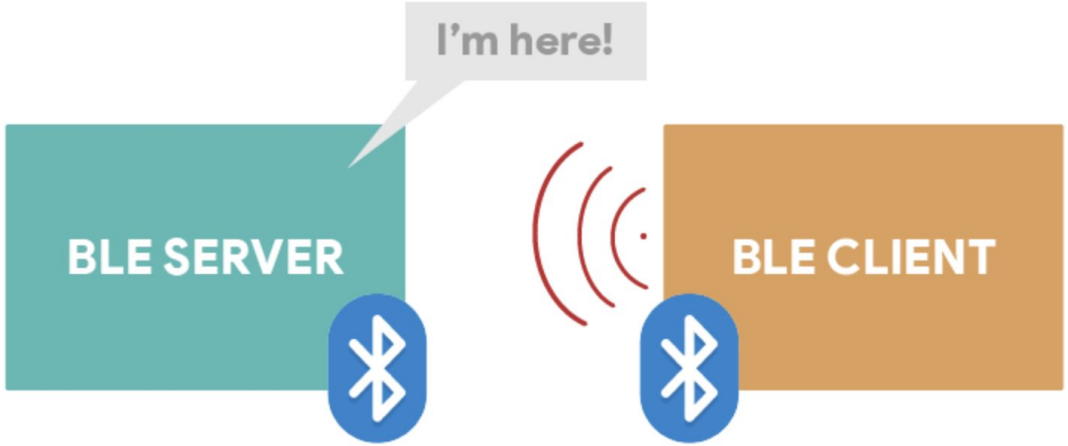
- ESP32 Microcontroller
- LED
- Power Distribution System

Updates + Changes:

- Inadequate power supply
- Reducing number of LEDs

Code Implementation

- Our code for the SpoilSense and External Alert modules follow a client-server architecture
  - The Internal SpoilSense Module is the Server
  - The External Alert Module is the Client



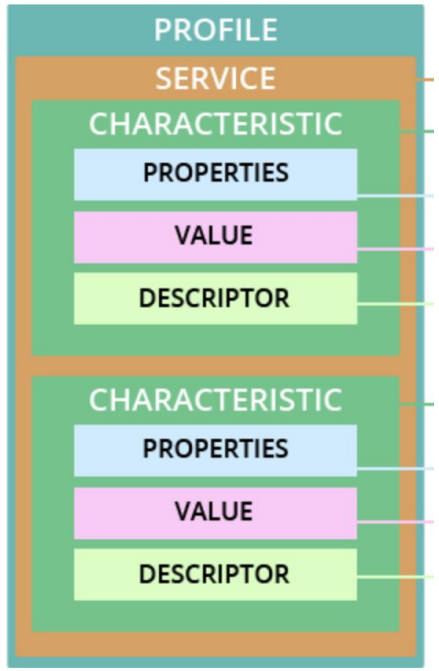
Code Implementation: Server

MICS Sensor Initialization (`initMICS`):

- Waits 3 minutes for the sensor to complete its calibration/warm-up process.

BLE Server Setup:

- Initializes the BLE server with a custom name (`SpoilSenseINT`).
- Creates a BLE service (`SERVICE_UUID`).
- Adds Alert characteristics:
  - **Alert Characteristic (`ALERT_UUID`)**:
    - Notifies the client when gas levels exceed thresholds.
    - Descriptor indicates the purpose of the notification.



Connection Status Callbacks:

- Tracks when a client connects or disconnects to/from the BLE server.
- Updates `deviceConnected` accordingly and logs connection status

Main Loop:

- If a BLE client is connected:
  - Reads gas concentration data (NH3 and CH4) from the MICS sensor.
  - Checks if the gas data exceeds thresholds:
    - If thresholds are exceeded, sends a notification to the client via the Alert characteristic.
  - Waits for 10 seconds before the next reading.
- If no client is connected:
  - Restarts BLE advertising and logs a status message.

Code Implementation: Client

BLE Client Initialization:

- Sets up a BLE client to communicate with a specific BLE server (SpoilSenseINT).

BLE Server Connection:

- Scans for the server and establishes a connection if the server is found.
- Retrieves specific BLE characteristics (ALERT_UUID, ACK_UUID) from the server.

Alert Notification Handling:

- Listens for notifications from the ALERT_UUID characteristic.
- Triggers an RGB LED (red) to visually indicate an alert.

Overview

1. Motivation
2. Solution
3. Implementation
4. **Results**
5. Looking Back
6. Looking Ahead
7. Questions

High Level Requirements

1. The module should be accurate in detecting the presence of volatile organic compounds (VOCs) up to a threshold of 3000 PPM for Methane (CH4) and 50 PPM for Ammonia (NH3) in order to trigger an alert.
2. The design should be capable of withstanding standard refrigerator temperatures (3-5 ° C) and should maintain a form factor that is at most dimensions of 5 in x 4 in x 2 in.
3. The system must respond to detected gas concentration changes and trigger an alert within 30 seconds of the threshold being met to ensure timely responses.



SpoilSense Microcontroller R&V

Requirement	Verification	Result

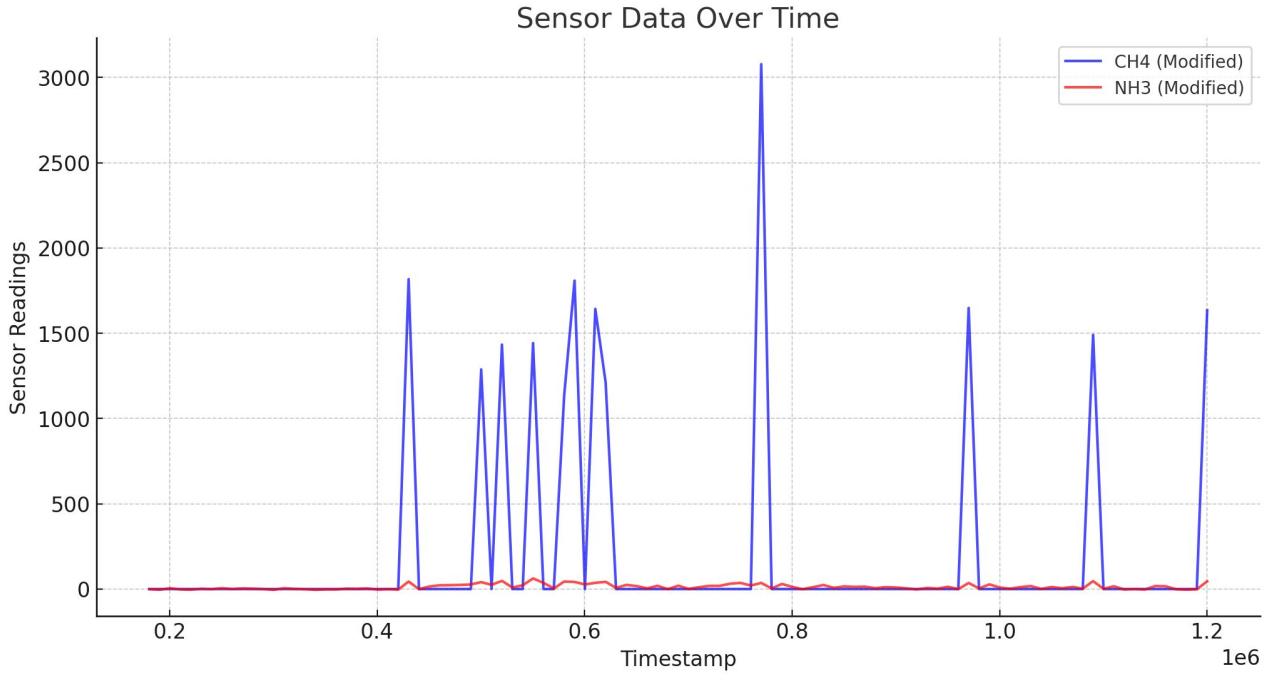
Must process sensor data and trigger alerts within 30 seconds of spoilage detection.	- Created a controlled test environment where spoilage conditions are simulated and detected by the sensors.
	
	- Initiated the spoilage event and utilized alert sent/received timestamps on the respective server and client side in order to ensure timely communication between modules.
		Server:
		`12:36:21.712 -> NH3: 76.6 PPM, 185094`
		`12:36:21.712 -> CH4: 4073.6 PPM, 185094`
		`12:36:21.712 -> Alert sent to client!`
		
		Client:
		`12:36:26.722 -> Alert received: CH4/NH3 > threshold`

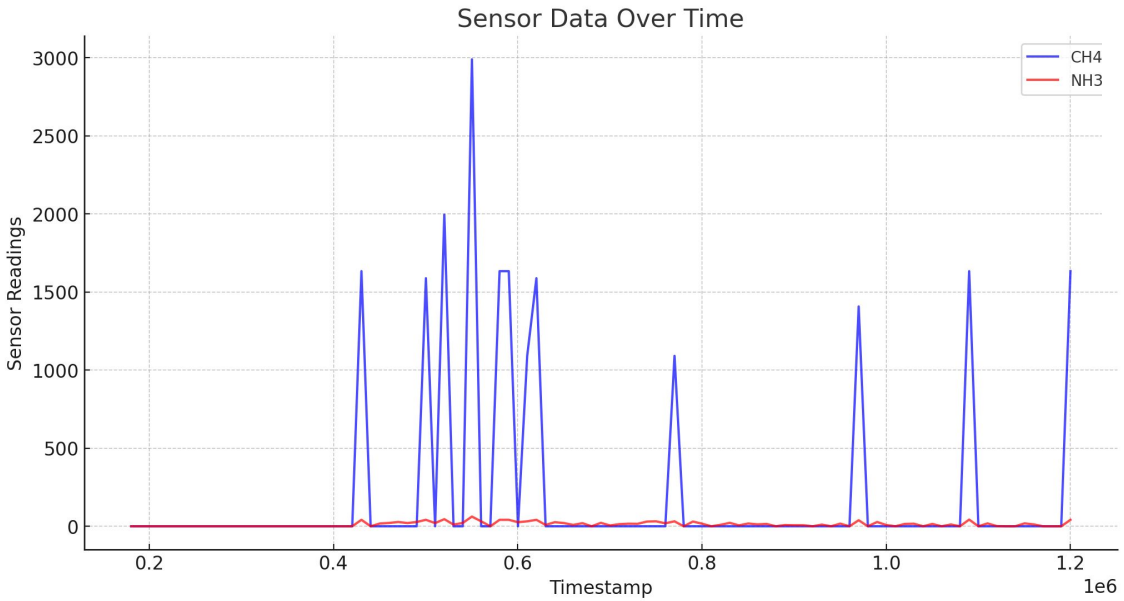
Must maintain a wireless communication through the door of the fridge with the external alert system, using BLE	- Set up an environment where the device and the alert system are placed at varying positions on either side of the metal barrier. - At each position, conduct communication tests to assess successful pairing, data transmission reliability, signal strength, and latency, while recording any instances of dropped connections or errors.	

SpoilSense Sensor Results

Requirement	Verification	Result
The VOC sensor should be able to accurately detect the presence of volatile compounds up to thresholds	- To verify the VOC sensor accuracy up to necessary thresholds, we set up a controlled test environment with some sort of spoiled food. - We conduct tests with various fruits, recording the sensor readings to a serial console. Once we found a common value that indicates spoilage, that value will be our threshold for the microcontroller.	On next slide

SpoilSense Sensor Results





SpoilSense Power R&V

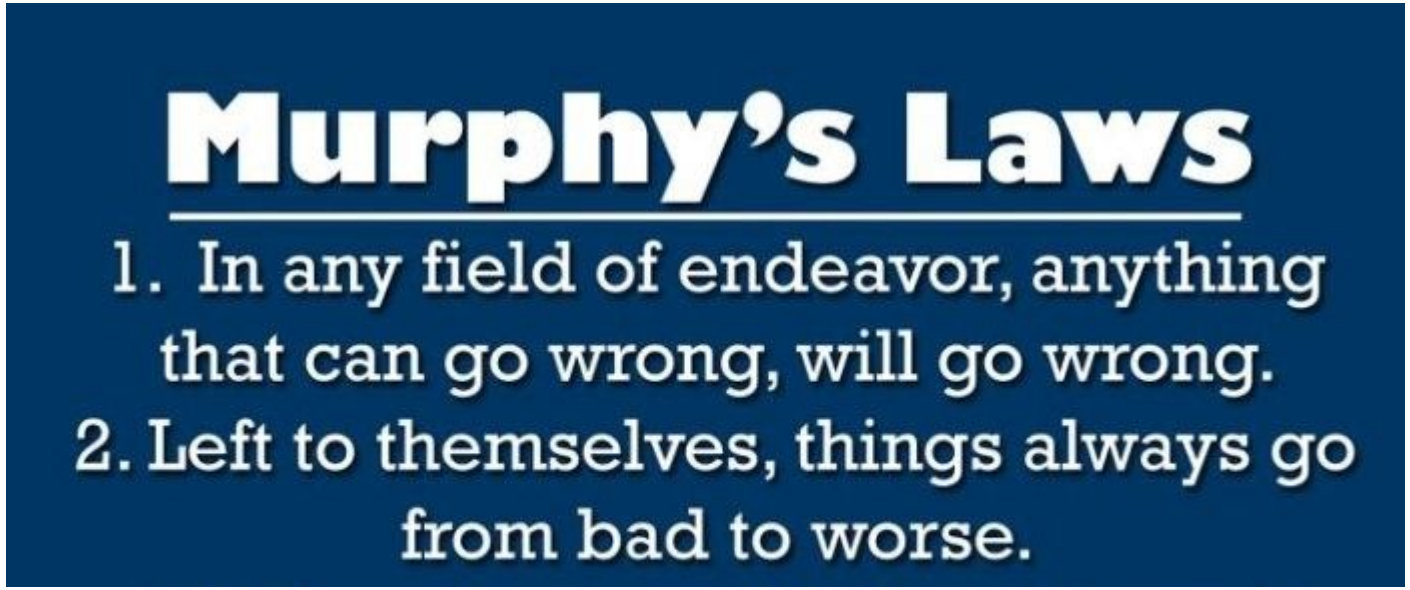
Requirements	Verification	Results
Must supply voltage to microcontroller within operating range of 3V - 3.3V	- Measured voltage reading across terminals via oscilloscope, to verify it is within range	Not able to verify on PCB, but our prototype can confirm
Supply voltage to VOC sensor should range between 4.9 V - 5.1 V	- Operating bias voltage range is 4.9 V - 5.1 V, so measured voltage readings across terminals via oscilloscope	Not able to verify on PCB, but our prototype can confirm

Overview

1. Motivation
2. Solution
3. Implementation
4. Results
5. **Looking Back**
6. Looking Ahead
7. Questions

Looking Back

- What did we learn?
  - Murphy's Law!
    - PCB logistics, sensors, scheduling
- What would we do different?
  - More research into sensors and collect more data for thresholds
  - Apply a moving average for more accurate data collection



Overview

1. Motivation
2. Solution
3. Implementation
4. Results
5. Looking Back
6. **Looking Ahead**
7. Questions

Looking Ahead

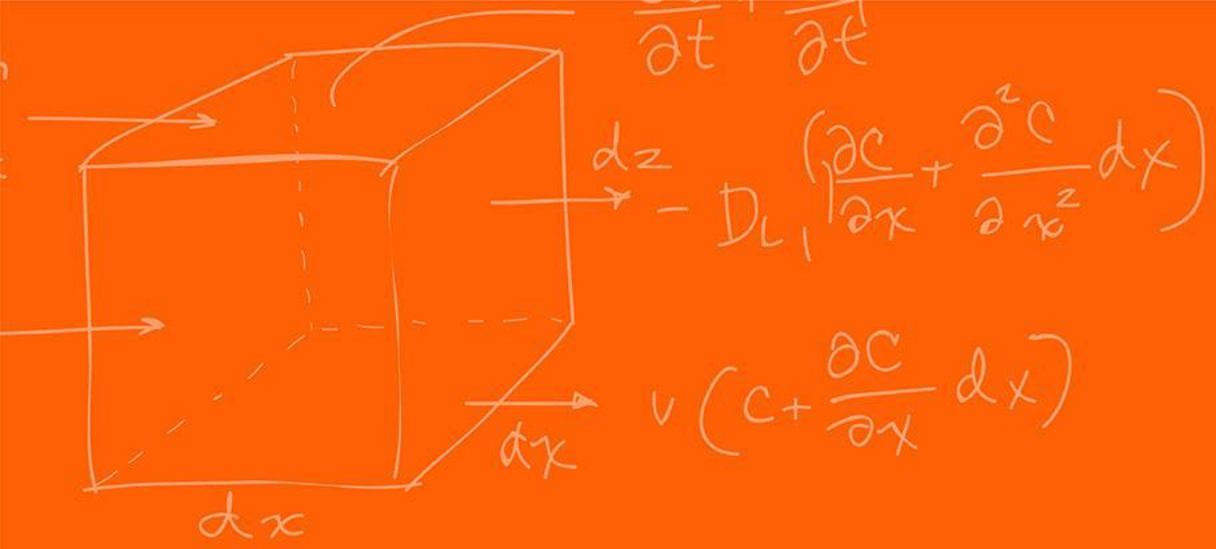
- Where can we go in the future?
  - SpoilSense modules for grocery stores + delivery trucks
  - Apply different thresholds for specific produce
    - Alert with different LED colors

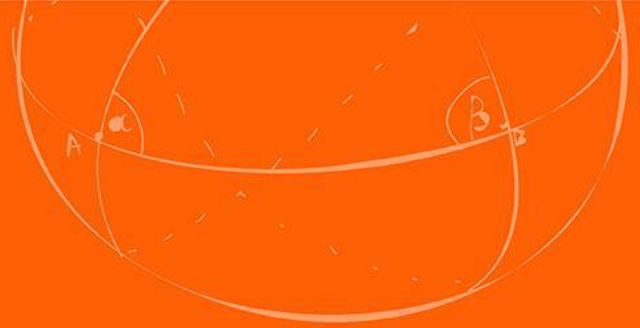
Thank You

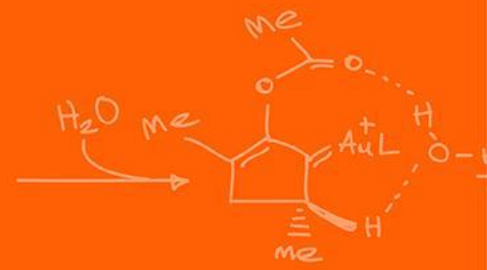
Questions?

Citations

- https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.istockphoto.com%2Fphotos%2Fmoney-stack-cartoon&psig=AOvVaw1CEv27eMAuJfzGWXfYLaIA&ust=1733865571339000&source=images&cd=vfe&opi=89978449&ved=0CBQQjRxqFwoTCNjhw97Om4oDFQAAAAAdAAAAABAE
- https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.shutterstock.com%2Fsearch%2Ftraining-health-care-cartoon&psig=AOvVaw1hY3mNWbUMqfCVPS8XpESa&ust=1733865777840000&source=images&cd=vfe&opi=89978449&ved=0CBQQjRxqFwoTCMD-kMDPm4oDFQAAAAAdAAAAABAE
- https://www.google.com/url?sa=i&url=https%3A%2F%2Fstock.adobe.com%2Fimages%2Frotten-food-product-set-vector-illustration-cartoon-isolated-spoiled-and-damaged-fruit-and-vegetables-with-rot-danger-mold-and-poisons-moldy-expired-pieces-slices-and-whole-fruit-collection%2F522208219&psig=AOvVaw1Qu1cExa3QWHE_VNm-CDuJ&ust=1733865906233000&source=images&cd=vfe&opi=89978449&ved=0CBQQjRxqFwoTCND92JLQm4oDFQAAAAAdAAAAABAE
- https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.linkedin.com%2Fpulse%2Fwhat-murphys-how-law-works-siddharth-human&psig=AOvVaw2epogm-AEkCFLdXx2ZidOK&ust=1733881471075000&source=images&cd=vfe&opi=89978449&ved=0CBQQjRxqFwoTCPis5vuJnIoDFQAAAAAdAAAAABAE
- https://www.google.com/imgres?q=groceries&imgurl=https%3A%2F%2Fwww.bankrate.com%2F2020%2F08%2F01170557%2FHow-to-save-money-on-groceries.jpeg%3Fauto%3Dwebp%26optimize%3Dhigh%26crop%3D16%3A9&imgrefurl=https%3A%2F%2Fwww.bankrate.com%2Fbanking%2Fsavings%2Fways-to-save-money-on-groceries%2F&docid=j5mUWtcgWDuV2M&tbnid=h89OsD9ZTVS6cM&vet=12ahUKEwjB7vauipyKAxXnv4kEHeIvHwQQM3oECFQQAA..i&w=1280&h=720&hcb=2&ved=2ahUKEwjB7vauipyKAxXnv4kEHeIvHwQQM3oECFQQAA
- https://www.google.com/url?sa=i&url=https%3A%2F%2Fstock.adobe.com%2Fimages%2Ftruck-car-full-of-food-products-shop-and-farm-delivering-service-delivery-and-selling-grocery-products-concept-meat-milk-bread-vegetables-cargo-and-logistic-cartoon-flat-vector-illustration%2F381087006&psig=AOvVaw1JK1tpHKgfylNwjkegZnwE&ust=1733881683926000&source=images&cd=vfe&opi=89978449&ved=0CBQQjRxqFwoTCLDIgOKKnIoDFQAAAAAdAAAAABA0







The Grainger College of Engineering

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN