UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Team 80 MazEscape

Electrical & Computer Engineering

Jayanto Mukherjee, Jatin Tahiliani, Will Knox

Spring 2025

Our objective

Problem

Issue at hand

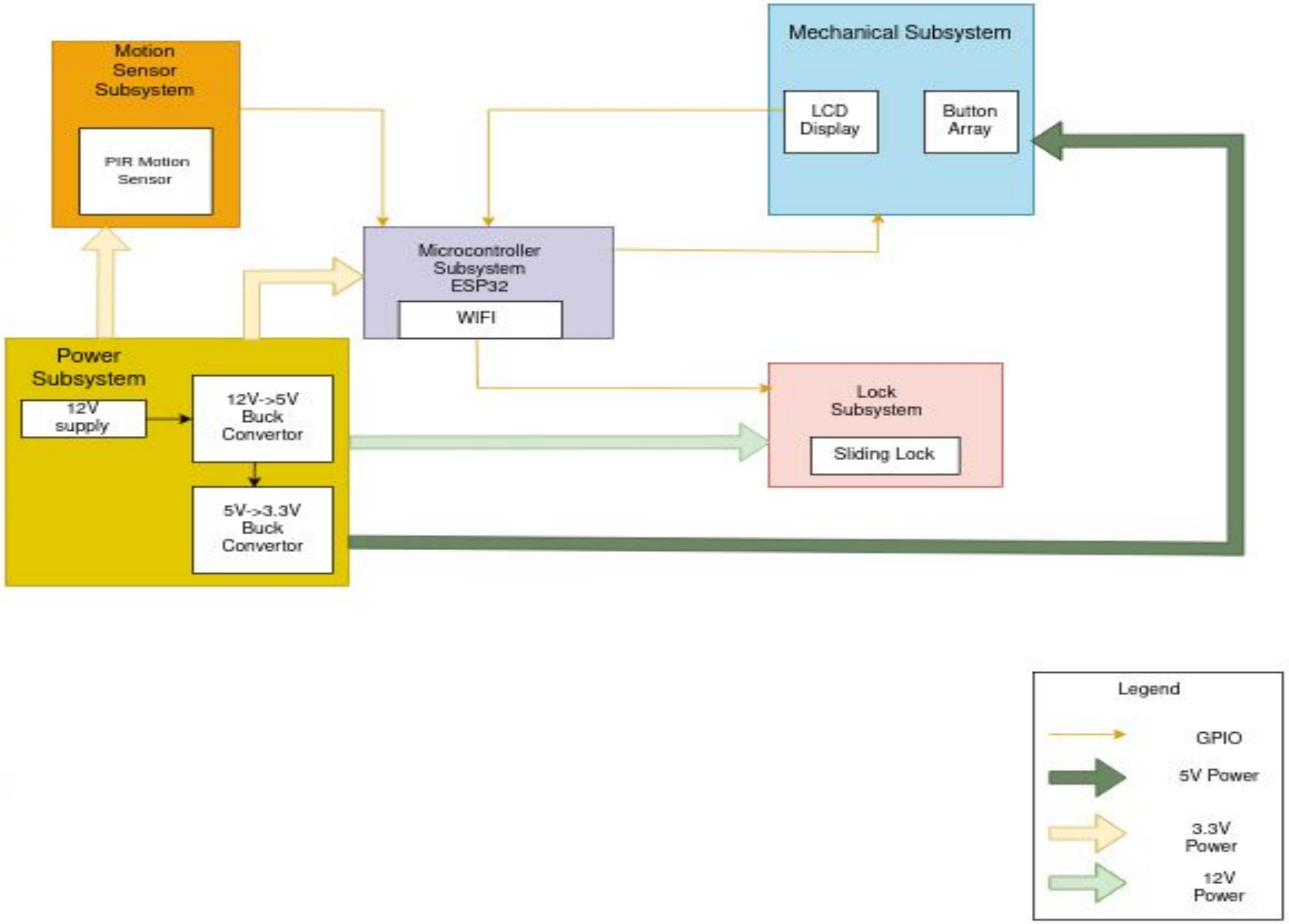
Modern-day theme park immersive games have become stale and predictable, so we wanted to make them more entertaining by seeing if it is possible to mix some of them. So, we devised a fun idea for a mix between a maze and an escape room where the participants will enter a labyrinth and answer questions to move onto the next level or to the next room and complete the game.

How we plan to solve it

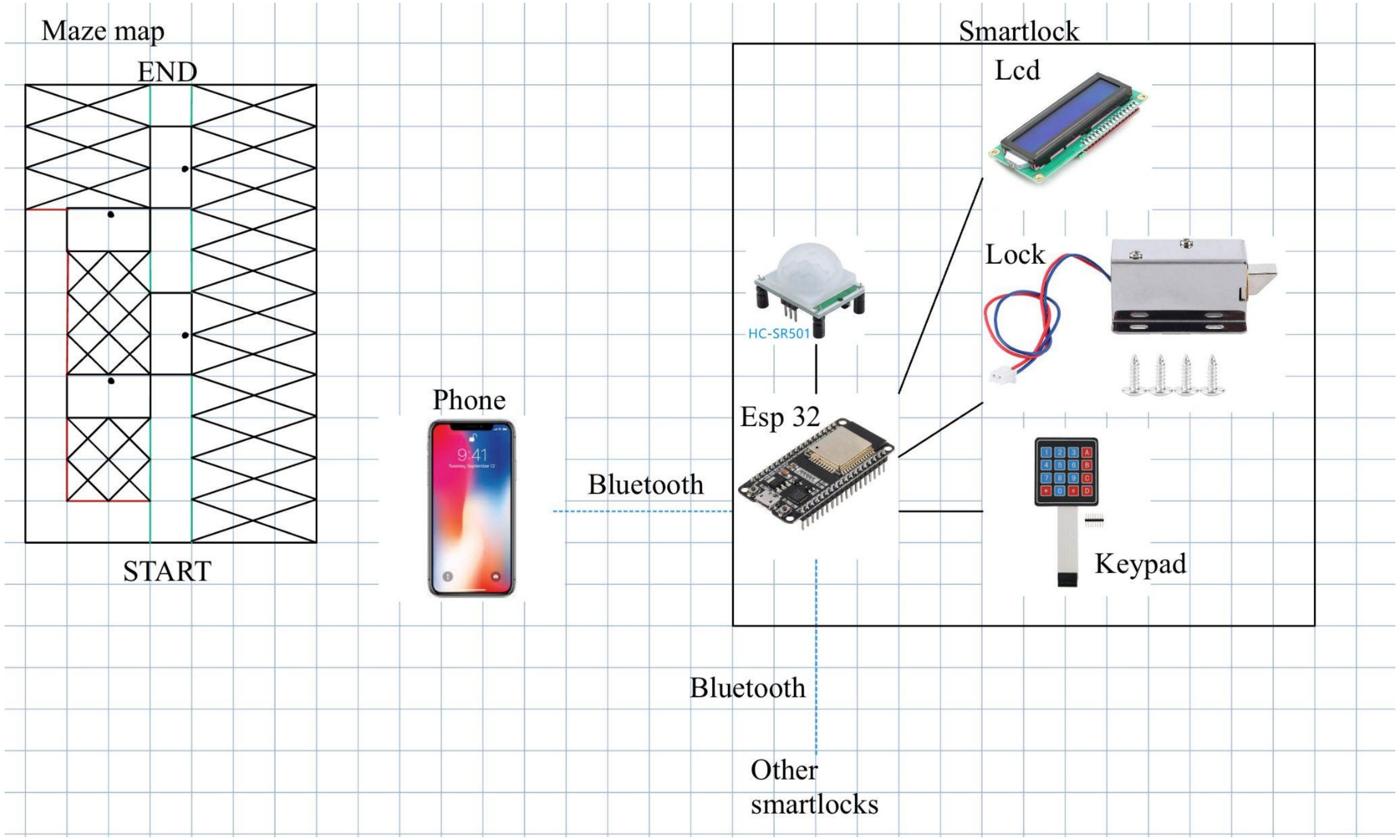
The maze uses LCD-equipped smart locks (one per level) that present a single-chance, three-option multiple-choice puzzle. Each level unlocks the gate to the next until the exit is reached with your prize. three consecutive wrong answers immediately trigger the escape locks to open and return you to the start; motion sensors wake each puzzle lock on approach, and the locks share data to avoid repeating questions.

Project build and results

Block Diagram



Visual Aid



Maze map
END
START
Phone
Bluetooth
Smartlock
Lcd
Lock
HC-SR501
Esp 32
Keypad
Bluetooth
Other
smartlocks

Final End Product

High level requirements

- The system shall unlock within 2 seconds of receiving correct input from the participant to ensure a smooth and engaging user experience.
- The motion sensor must detect an approaching participant within 3-5 meters and trigger the display of the quiz question within 1 second to facilitate prompt interaction.
- The Wifi module integrated within the smart lock systems shall reliably exchange data—specifically transmitting quiz questions and lock/unlock commands.

Mechanical Subsystem

Requirements

- The LCD display must update content reliably at ≥60 Hz.
- The keypad must accurately capture and transmit inputs within 0.5 seconds

Verifications

- Reference datasheet for refresh rate
- Access an external tool that can timestamp both the physical press and the system’s input reception.

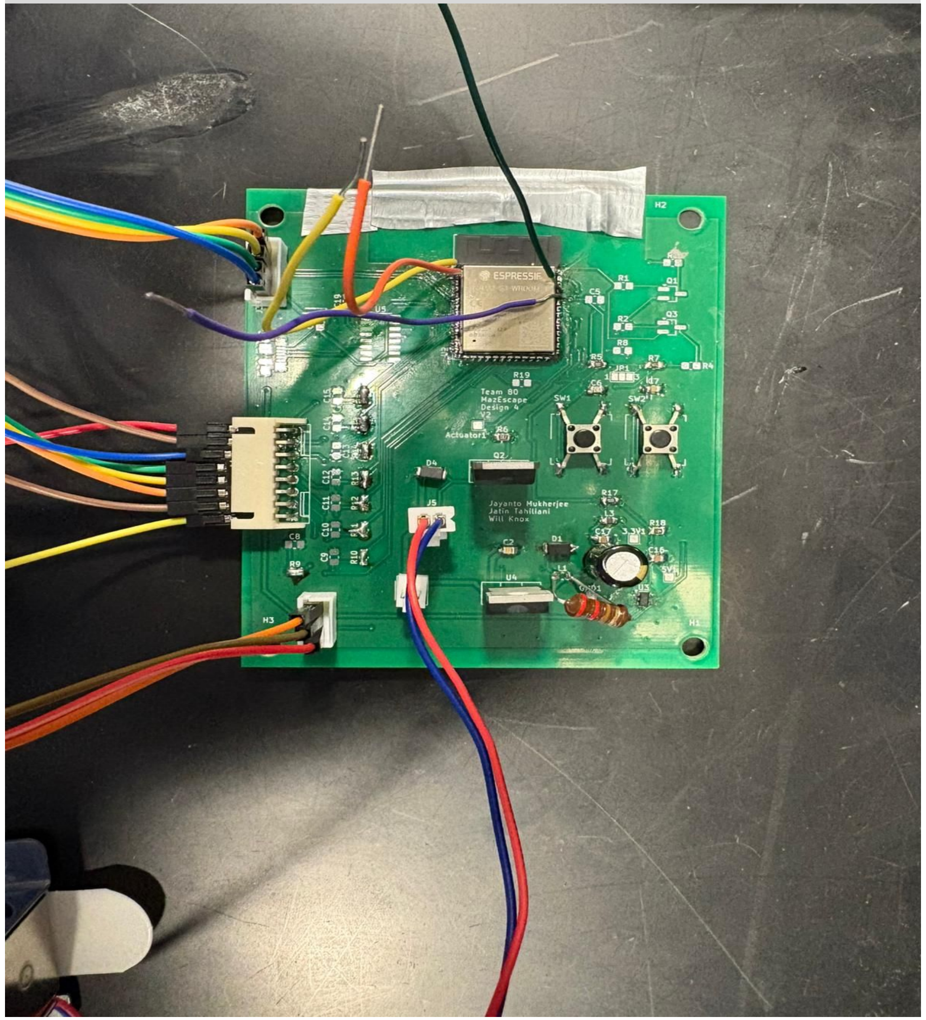
Microcontroller Subsystem

Requirements

- The microcontroller, ESP32, must be able to interpret the status of the user input
- 3.3±0.5V must supply the microcontroller

Verifications

- Record the behavior of the DC solenoid lock in response to each command using an oscilloscope.
- Use a multimeter to measure the voltage of the microcontroller
- Ensure voltage readings are within 3.3±0.5V for each microcontroller.



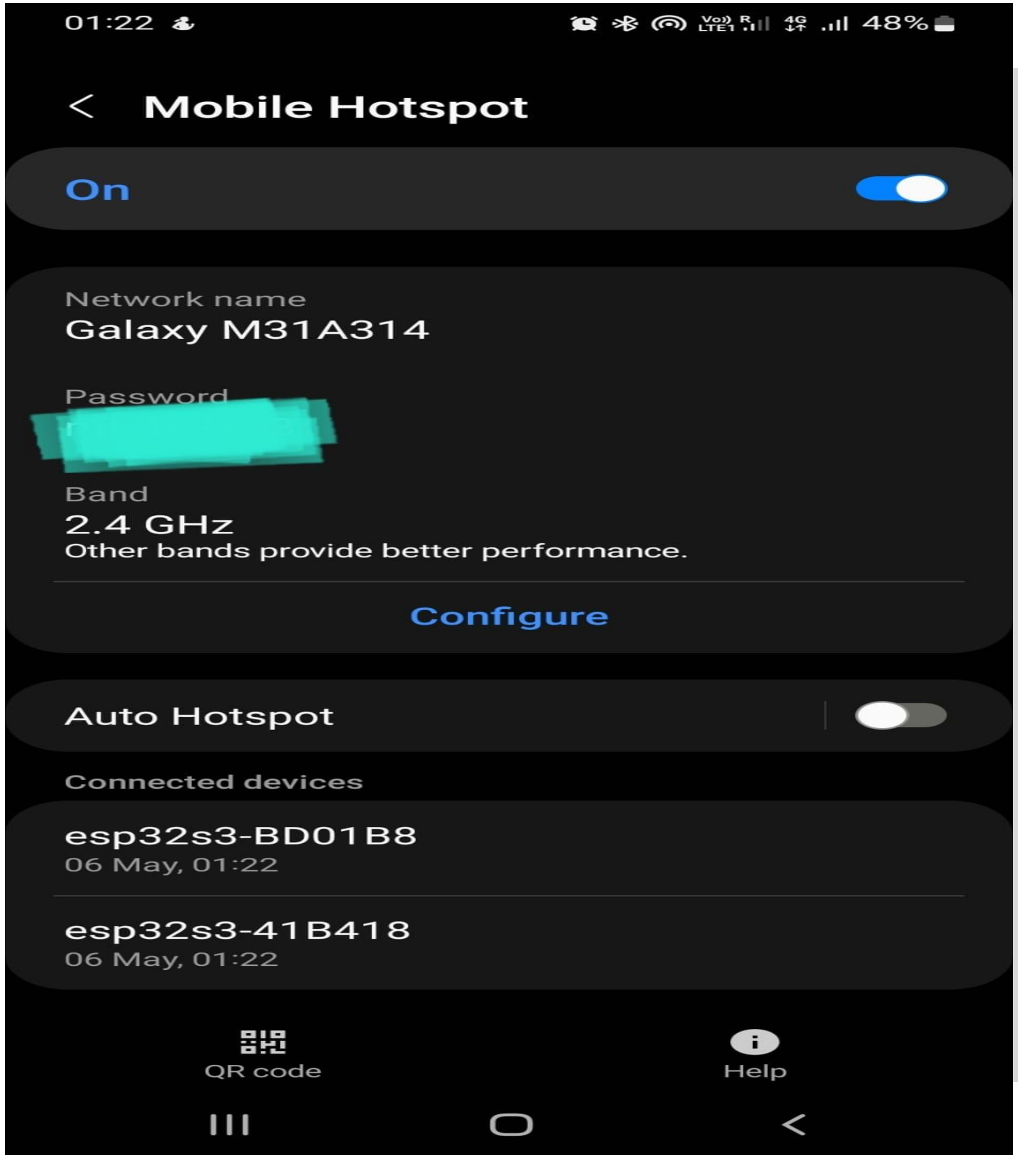
Wifi Subsystem

Requirements

- Provide stable and responsive (<100ms input lag) Wifi connection
- It must interface seamlessly at 3.3V logic levels with the ESP32 microcontroller to relay commands accurately.

Verifications

- For stability, tests should be performed in a crowded environment such as the ECEB
- Test the amount of time it takes to receive the updates.
- Ensure voltage readings are within 3.3±0.5V for each microcontroller.



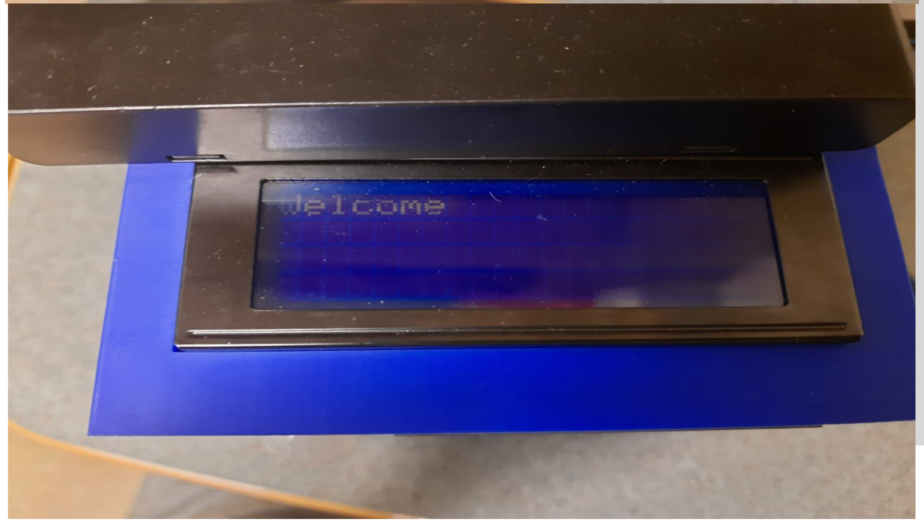
Motion Sensor Subsystem

Requirements

- When there is no human present, the PIR sensor module should output Logic Low (0 V)
- The PIR sensor module should output 3.3±0.5V digital pulse when a human is detected

Verifications

- Ensure voltage readings are within 3.3±0.5V for each sensor
- Have a person walk before the sensor and observe. Repeat this procedure multiple times and confirm the expected behavior occurs.



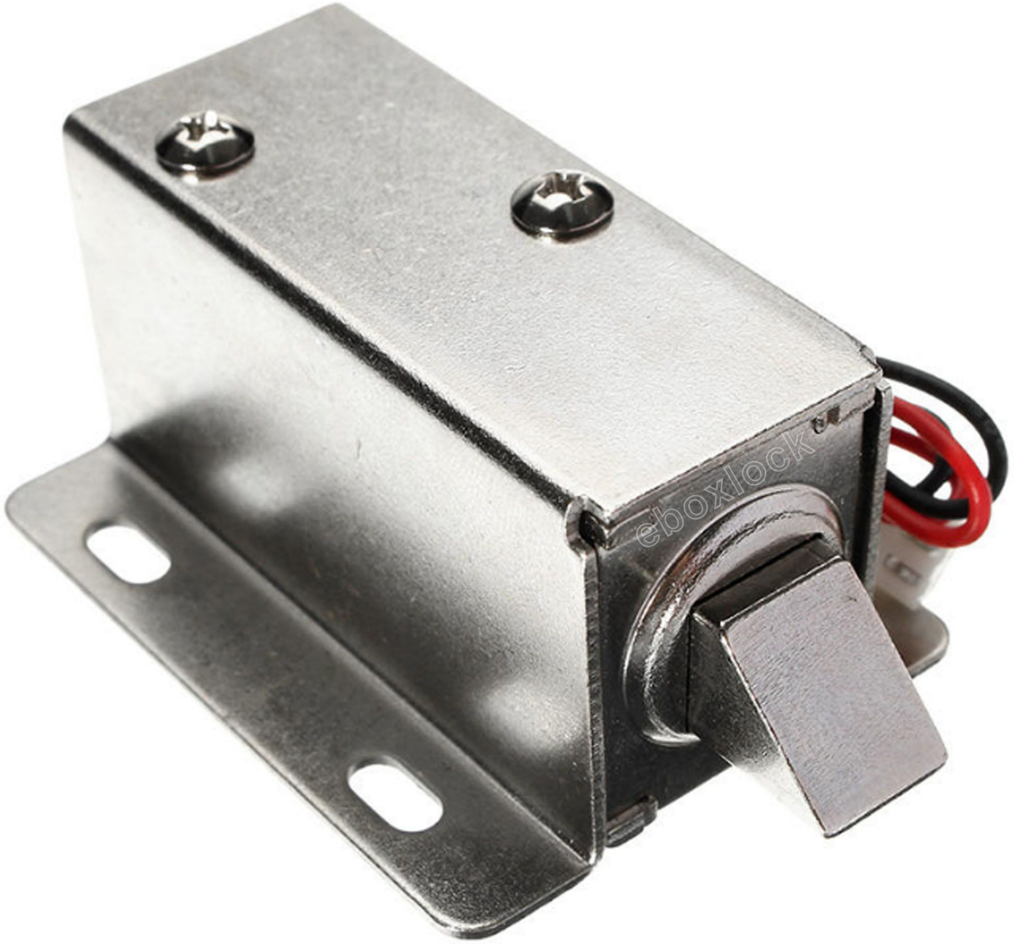
Lock Subsystem

Requirements

- The digital control interface must reliably receive 3.3V logic-level commands from the microcontroller.

Verifications

- Use a multimeter to measure the voltage of the microcontroller
- Ensure voltage readings are within 3.3±0.5V for each microcontroller.



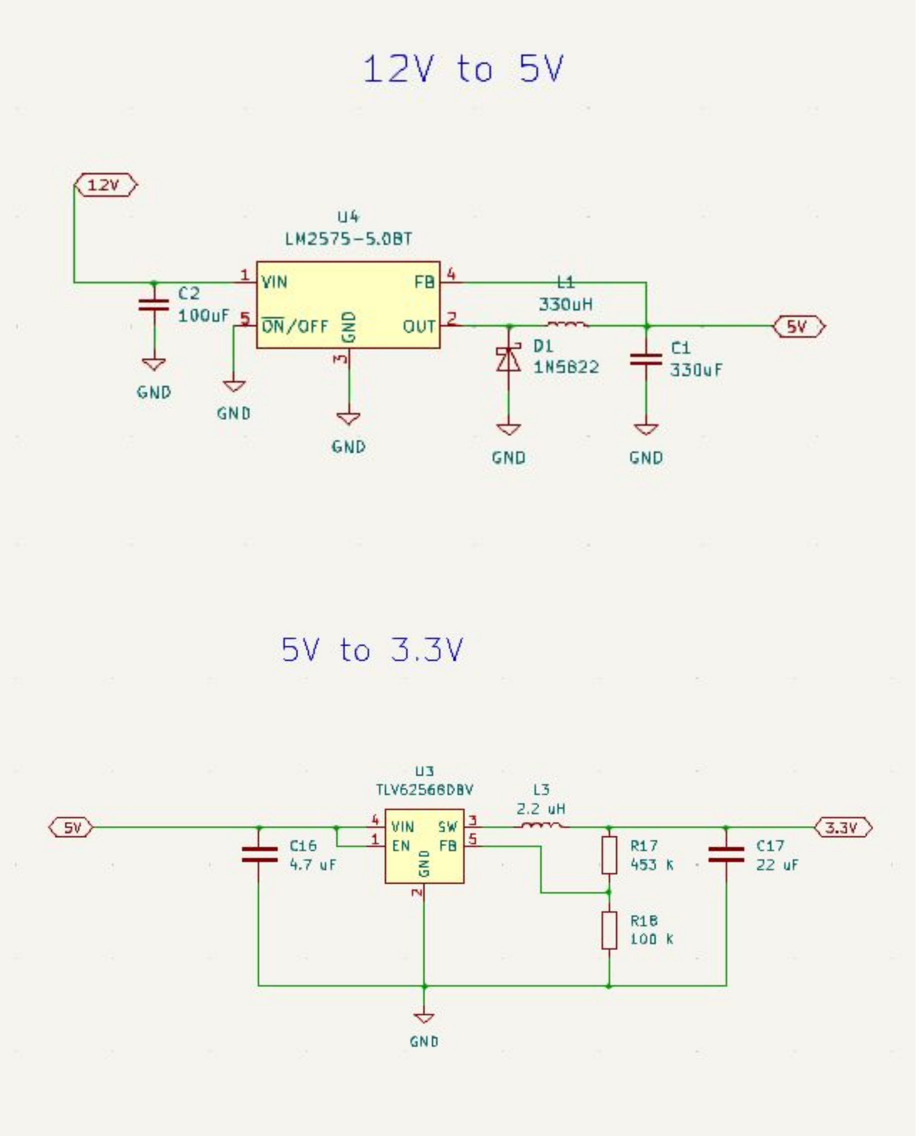
Power Subsystem

Requirements

- The 12V to 5V buck converter must provide a regulated 5V±0.5V output
- The 5V to 3.3V buck converter must provide a regulated 3.3V±0.5V output

Verifications

- Connect the Buck converter to a 12 V battery and measure its 5 V output while varying the load. The output must remain between 5.5 V and 4.5 V
- Connect the Buck converter to a 5 V battery and measure its 3.3V output while varying the load. The output must remain between 3.8 V and 2.8 V



Video of project working

Our successes and challenges

Our successes

- Powered every component accordingly (safely and through buck converters)
- Got the microcontrollers communicating with each other via WiFi
- LCD output and lock functionality worked as intended
- Soldered 2 PCBs efficiently
- Motion sensor worked as intended

Challenges

- Had to reorder PCBs because we did not account for USB to UART connection
- Used wrong USB footprint, wires had to be soldered directly to ESP32
- Originally used bluetooth, did not work as intended
- Initial troubles in finding the second ESP32’s IP address
- Hard to get the buck converters to operate smoothly at all times
- LCDs giving issues in terms of display and brightness.

Future applications and conclusion

Future Applications

- Expand the project to 4/6 PCBs instead of 2.
- The main PCB design can be made more concise.
- Can make the escape PCB more concise because keypad, motion sensor, and LCD is not needed
- Attach to doors to make the product ready for customers and the market

Questions?

The Grainger College of Engineering

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN