Item-Tracking Backpack

By

Abdullah Alfaraj

Raef Almuallem

Final Report for ECE 445, Senior Design, Spring 2024

Professor: Arne Fliflet

TA: Surya Vasanth

1 May 2024

Project No. 66

Abstract

This document outlines the motivation, design, development, results, and costs of the Item-Tracking Backpack project. This backpack enables users to track the items they plan on having in it at a given time through the use of RFID. An ESP32 microcontroller was used to interface with RFID readers and a mobile application to facilitate the tracking of items with RFID tags placed on them. This project was mostly successful, with each of the individual subsystems having accomplished the set goals. However, due to some issues occurring towards the latter end of the project timeline, the final product was somewhat flawed, with tracking not being as what was desired. The report therefore also details some of the improvements and further work that can be made onto the design.

Table of Contents

1. Introduction

1.1 Problem

A study conducted by Name Bubbles, surveying 400 parents, showed that around 80% of children lose their belongings at school [2]. 52% of the parents answered that the items lost included school supplies, while 36% stated that lunch boxes were among what was lost [2]. For most students, backpacks are their primary means of storing such items. With the ability of backpacks to store many items in various compartments, it is easy for students to lose track of what they have placed in them. It can also become inefficient and time-consuming for students to constantly check whether they have forgotten an item, especially when the number of items they have stored in their backpacks is typically relatively high. Moreover, the different compartments available for students to place their belongings in make it simple for them to become disorganized and lose track of a given item. This issue also results in students occasionally forgetting items at home, forcing parents to need to deliver them in cases where the items are important.

1.2 Solution

To solve this issue, the project we developed provides users with the ability to track the items in their backpacks. RFID is used to scan tagged items as they are placed and removed from the backpack. Students will be able to make lists of items they are planning on having in the backpack using a mobile application before their commute to school. Separate lists are available in the application for different compartments to assist users in keeping track of where the items are in the bag. The lists can be modified by adding and removing items, and the status of each item−present, missing, or misplaced−is displayed for the user to see in real time. The present status indicates that items are currently in the backpack and in the correct compartment; the missing status shows that an item is not currently in the bag; and the misplaced status means that an item is currently in the bag but in the incorrect compartment. In addition, the time at which a missing item was removed from the bag is indicated on the application to aid in locating it.

During school hours, many students will not have permission to access mobile phones, so LEDs are used to allow for quick, but less explicit real-time tracking. Each compartment has a red LED designated to it. These LEDs will turn on when the user has incorrectly placed an item in the compartment, or if an item is missing from it. The LEDs will turn off after the compartment zipper has been closed, or if they have been active continuously for 30 seconds. A green LED is used to indicate that all items are present in the backpack and in the correct compartment, indicating that nothing has been forgotten. This LED will similarly turn off after being continuously on for 30 seconds, or after all compartments in the backpack have been closed. The application can then be used after school by parents picking up their children, or by the students themselves, to ensure that nothing is missing or forgotten.

1.3 High-level Requirements

We identified three essential requirements that our project must meet for it to be considered successful. The following requirements were not only used to evaluate our design but also to guide us through the development and testing process:

1. The mobile application shall enable the user to add and remove items to three separate lists, each corresponding to a compartment in the backpack. The user shall be able to track a maximum of at least five items per compartment.

2. The user shall be able to monitor the status of each item—present, missing, misplaced— using the mobile application, and they will be updated at least once every ten seconds. The application shall display what time a missing item was removed from the backpack.

3. The LEDs shall activate within ten seconds of a requirement being met. To conserve power, each red LED shall turn off when its corresponding compartments are closed, and the green LED shall turn off when all compartments are closed. Otherwise, all LEDs shall turn off after being continuously on for 30 seconds.

2 Design

2.1 Physical Design

The RFID readers are attached to the interior walls of the backpack, towards the top. Tags placed on the items can be scanned in the process of placing and removing them from each of the compartments. For protection from items being placed and removed, the readers are placed in cases open from the side facing away from the wall, as shown in Figure 1. Each case also has an LED to facilitate quick tracking. Moreover, a light sensor is placed above each case to allow for the detection of when the zipper is closed. Finally, aluminum is placed on the back wall of each case to prevent interference between RFID readers. Each of the three compartments therefore has a separate reader, light sensor, and LED, with one of them—the front-most compartment—having a green LED. These are all connected to the PCB, which is placed at the bottom surface of the compartment at the center of the three compartments, as shown in Figure 2. To protect the PCB from the items in the backpack, it is also placed in a case for protection. Holes were made in the walls of the backpack compartments to allow for wires to be connected to the devices in the cases.

Figure 1: Final Design of Sensor and LED Case

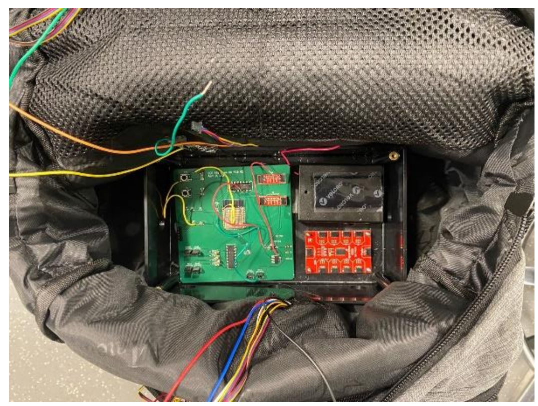

Figure 2: PCB and Power Case

2.2 Block Diagram

As shown in Figure 3, the overall system consists of five main modules: control, power, sensor, indication, and user-interface (phone application) subsystems. All subsystems, other than the user-interface subsystem, reside within the backpack. The power subsystem will supply 3.3 V to the entire subsystem, and the control subsystem, through the microcontroller, will interface with the sensors through I2C. The microcontroller will also control the turning on and off of the LEDs, and it will interact with the phone application through 2.4 GHz Bluetooth.

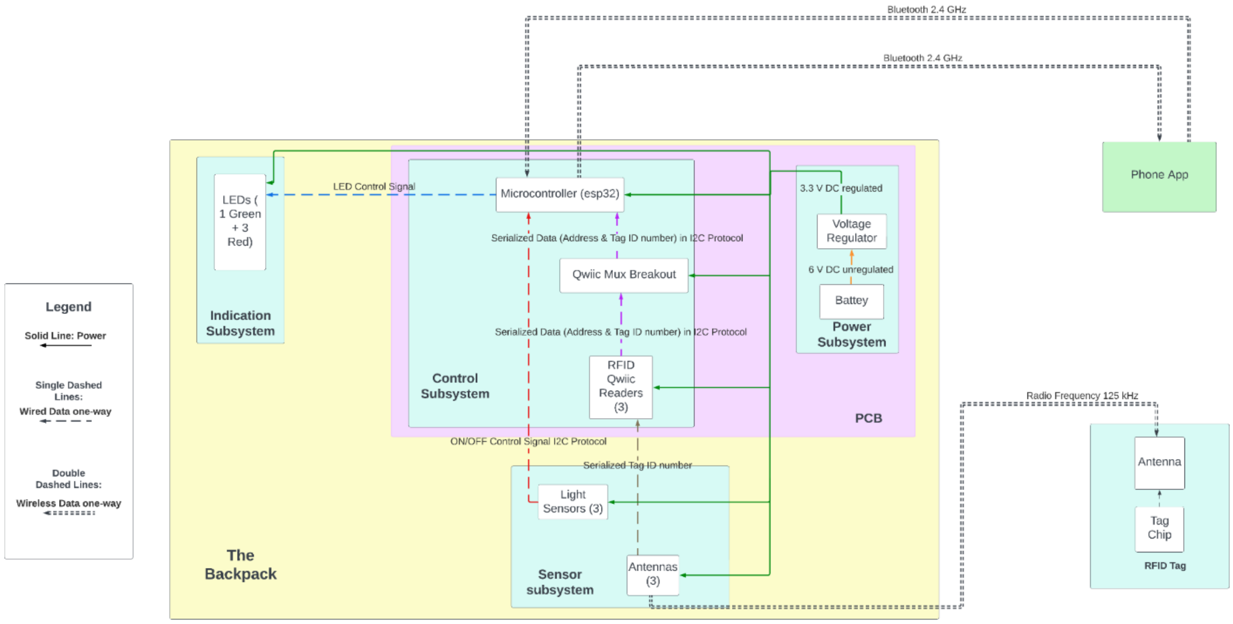

Figure 3: Design Block Diagram

2.3 Control Subsystem

The control subsystem consists of ESP32-S3. We chose it because it includes Bluetooth LE, which is important for communication with the user interface subsystem. The Bluetooth LE (Low Energy) module is incompatible with standard Bluetooth code and code, and it needs modifications. This subsystem will also consist of three RFID Qwiic readers, each connected to a separate RFID ID-12LA sensor. The Qwiic readers will track scanned tags by marking and placing them on a stack, and they will relay this information to the microcontroller through the I2C protocol. This subsystem will also include two Qwiic multiplexers (MUXs) because all the RFID readers share the same address, which is 0x7D. This prevents the microcontroller from communicating with them on the same bus. The MUX works by allowing a specific RFID reader to access the bus at a given point in time. The same situation applies to the light sensors and the second MUX as well.

The microcontroller will keep track of the registered items' location in the backpack (compartment 1, 2, 3, or outside the backpack) using a map from the tag RFID to a list containing the item's name, the assigned location, the last known location, and whether they are inside the backpack. The microcontroller will use timers to switch off the LEDs as well.

The microcontroller will then transmit the results to the user interface subsystem through Bluetooth at 2.4 GHz. Moreover, the microcontroller will use the information it receives from the Qwiic readers and light sensors to send the appropriate signals to switches for the activation or deactivation of the LEDs.

2.4 Power Subsystem

The power subsystem is responsible for delivering 3.3 V DC power to the rest of the subsystems. The power subsystem should supply up to 500 mA of current. This value is an overestimate to ensure the stability of the power subsystem. It consists of a battery and an LM317 voltage regulator. The voltage regulator used in this subsystem is a linear voltage regulator (low dropout regulator) with an output voltage of 3.3 V DC. A linear voltage regulator was used instead of a switching regulator because of its low noise output. This is important in this project because the noise might interfere with the sensors. Specifically, the output voltage ripple must be less than 3 mV peak-to-peak to allow the sensor subsystem to work properly. The batteries used in this project are 4 AA batteries, which have a total voltage of 6 V. We chose AA batteries due to their safety, availability, and cheap price. The value of the capacitors and resistor R1 below were chosen based on the manufacturer's suggestion. The resistor R2 value is chosen based on equation (1), where I_adj is approximately negligible. A simulation of the power subsystem was performed in LTspice and the result is shown in the appendix.

$$V_{out} = V_{ref}(1 + \frac{R_2}{R_1}) + I_{adj}R_2 \qquad \text{Eq(1)}$$

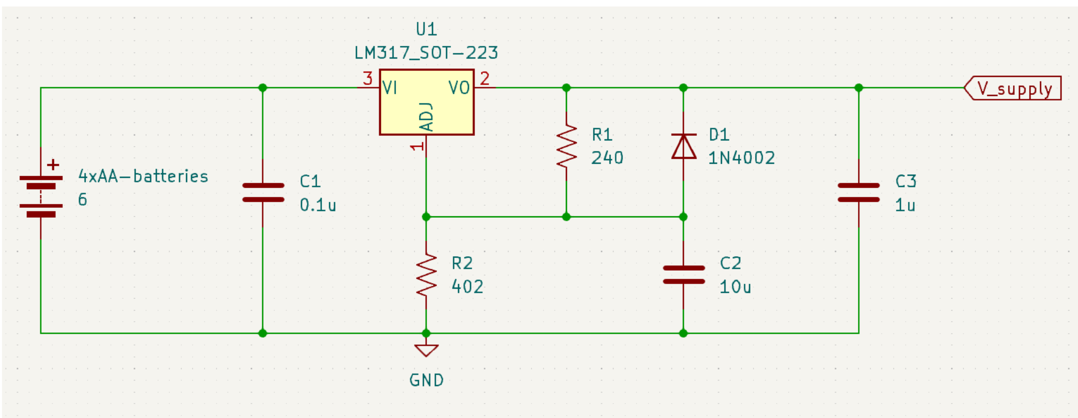

Figure 4: Power Subsystem Schematics

2.5 Sensor Subsystem

The sensor subsystem is tasked with keeping track of what items are in the backpack and the compartments they are located in at a given time. It will also detect changes in the ambient light in each compartment to identify when they have been closed. To carry out the item-tracking functionality, RFID ID-12LA readers, operating at 125 kHz, were used to read the RFID tags of items as they were placed in and removed from the different backpack compartments. These readers were connected directly to the RFID Qwiic readers of the control subsystem, through which the IDs of scanned items were temporarily stored and sent to the microcontroller. This specific reader was chosen over alternatives for various reasons. The relatively low frequency of the readers results in a relatively large read range. This would provide users with a better experience, as it would prevent them from having to bring items close to the readers when placing or removing them. Moreover, the RFID Qwiic readers, which the ID-12LA readers are attached to, provide several benefits, most importantly being their ability to store at most 20 reads at a given time. This would prevent any issues involving the deletion of read tag IDs before they have been sent to the microcontroller. The I2C functionality of the RFID Qwiic readers also offers a simple means for sending read IDs to the microcontroller. Finally, the ease at which the RFID Qwiic readers can be mounted makes them ideal for this project, where the positioning of the sensors is essential. Regarding design alternatives with the RFID readers, we initially considered having them turned off when the compartment they were located in was closed. However, we realize that this would be an unnecessary addition, as the readers would not consume much power, and only showed an increase in power consumption in the instant at which a tag was read.

To allow for the detection of changes in the ambient light in each compartment, Adafruit LTR-329 light sensors were used. These specific light sensors were chosen for their sensitivity and their ability to send exact illuminance readings to the microcontroller through I2C. This would offer us the ability to control the illuminance threshold between the closed and open compartment states. Finally, both ID-12LA and Adafruit LTR-329 were connected to the PCB through Qwiic cables, which provided a 3.3 voltage supply (which both sensors required) and I2C connections simultaneously. This would reduce clutter in the wiring, making for a more organized design compared to alternatives that lacked Qwiic connectivity. The RFID tag data and ambient light readings were both accessed by the microcontroller through Arduino I2C protocols, allowing the control subsystem to make use of this collected information to send to other subsystems.

2.6 Indication Subsystem

The indication subsystem will quickly inform the user about whether all their desired items are in the backpack. It will also quickly notify the user if the items in the compartments do not exactly match what was specified in the application. To these ends, a green 5 mm LED was used to indicate to the user that no items had been forgotten. The LED will activate once all the items listed on the application have been placed in the backpack and in the specified compartments. It will turn off once the compartment it resides in is closed, or once it has been continuously on for 30 seconds. This will be done to conserve power and maximize the life of the LED to prevent it from burning out. One red 5 mm LED will be assigned to each compartment. The red LEDs will turn on if the items currently in the compartment do not exactly match what is on the application. They will, for the same reason as the green LED, turn off when their corresponding compartment has been closed, or after they have been continuously on for 30 seconds. A DIP switch, controlled by signals from the microcontroller, was used to turn the LEDs on and off. To drive the LEDs, they will all be supplied with 3.3 V, with series resistors being used to step the voltage down to 2.2 V, within the permissible voltage range of the LEDs. The resistor values were then chosen to ensure that the current into the LEDs would not exceed the rated values. The rated currents were 25 mA and 30 mA for the green and red LEDs, respectively. Using Ohm's law, as shown below, the resistances to use were obtained.

$$R = \frac{V}{I} = \frac{1.1\,V}{0.025\,A} = 44\,\Omega \qquad \text{Eq(2)}$$

One alternative design that could have been chosen would have been to keep the LEDs active constantly while the backpack is open, rather than have them turn off after 30 seconds. This would prevent the user from misunderstanding the status of the compartments. However, it is also likely that the user would simply forget to close their backpack after using it. Unnecessary power consumption and accelerated burnout of the LEDs would then be an issue. As a result, the LEDs being on for 30 seconds at a time would be the better design choice, as it would give the user a sufficient amount of time to notice them being on. If the user is unsure about the status of the compartment, the timer could be easily reset by closing the compartment of a given LED.

2.7 User Interface Subsystem

The user interface subsystem consists of a mobile application to allow the user to track the items in the backpack in real time. The application also gives the user the ability to change the items they wish to track. Furthermore, it notifies the user when an item is placed in the incorrect compartment. To carry out these functions, the application enables the user to add and remove items from separate lists corresponding to each compartment. The user will have the option of choosing between adding a registered item (one they have previously tracked) or inserting a new one. The status of each item—present, missing, or misplaced—is displayed adjacent to each item and is updated in real time. Bluetooth (2.4 GHz) allows the mobile application to interface and communicate with the microcontroller.

2.8 PCB

We designed two versions of the PCB. The first version was ordered during the first ordering group but, it had some misconnections in the switches, which prevented the LEDs from getting the correct voltage. Also, some of the connectors were larger than the assigned footprint. For these reasons, a second version of the PCB was ordered. It has the power subsystem in the bottom left corner, the microcontroller in the center, the programming circuit on the left side, the indication subsystem in the bottom left corner, and the sensor subsystem connectors in the top right corner. It uses a two-pin connector for the batteries and the LEDs. It uses the Qwiic connectors for the multiplexers (MUXs). We initially designed the PCB to have 2 battery packs for increased capacity, but that was not needed in the final design.

Our PCB had some issues which will be discussed in the design verification section.

3. Design Verification

All the R&V tables are shown in Appendix A.

3.1 Control Subsystem Verification & Results

The control subsystem has four requirements. The first one is the ability to transmit and receive Bluetooth LE (Low Energy) 2.4 GHz signals. This was verified using the nRF Bluetooth testing application. The nRF application successfully received Bluetooth advertisements and notifications. It also was able to transmit data in string (UTF-8 TEXT) format.

The second requirement is the ability to receive and transmit serial signals in the I2C protocol. This was verified by connecting one of the RFID sensors and one of the ambient light sensors to the same I2C bus. Then, the esp32 microcontroller sent serial signals containing the scanned tag RFID number and the light intensity to the computer's serial monitor. The results were successful. Then, all three RFID readers and all three ambient light sensors were connected to the same bus using two multiplexers (MUXs), and the same testing procedure was performed. The results were also successful.

The third requirement is the ability to track all the items' names, assigned locations, last known locations, and whether they are inside the backpack. This was verified using the nRF testing app and the computer's serial monitor. The nRF testing application was used to register 5 items, and the serial monitor was used to view the status (present, missing, or misplaced) of each registered item. Then, multiple tags were scanned using the RFID sensors. The status of all the registered items was successfully tracked.

The fourth requirement is that the microcontroller must supply the correct voltage to switch the LEDs on or off. Since the design uses active low switches, the microcontroller should output 0 V to the switches to turn the LEDs on and 3.3 V to the switches to turn the LEDs off. This requirement was verified by a digital multimeter to measure the voltage of all four LED switch signals. The microcontroller was initially set up to have the green LED on the rest of the LEDs off. This was successfully measured. Then, the microcontroller was set up to have the green LED off and the rest of the LEDs on. This test was also successful.

3.2 Power Subsystem Verification & Results

There are two requirements for the power subsystem. The first requirement for the power subsystem is that it must supply 3.3 ± 0.1 V DC to a load that requires a maximum of 500 mA. The actual power consumption of the full design is 200 mA, but the testing was performed for higher power output to ensure the safety and stability of the power subsystem. The second requirement is that the output voltage must be less than 3 mV peak-to-peak for the sensors to work properly.

Both of these requirements were tested using a digital multimeter and an electronic load. The electronic load was configured in constant current mode and it was connected to the power subsystem. The result was that the output voltage was 3.336 V and the ripple was less than 1 mV peak-to-peak.

3.3 Sensor Subsystem Verification & Results

To be considered successful, the ID-12LA RFID readers needed to read tags relatively quickly, within around two seconds, and it was necessary that the readers did not interfere with one another. To verify that these readers performed reads within two seconds of the tag being within range, we simply brought the RFID tags into contact with the readers. This was done to ensure that the tag to be scanned was within the range of the reader. The RFID Qwiic readers that the ID-12LA readers are connected to contain a buzzer, which sounds when a read is made. We measured the time for the buzzer to sound once the tag was brought into contact with the ID-12LA readers, which was found to be almost instantaneous, much less than the two seconds we required. Testing of the lack of interference between readers was done by measuring the read range of each sensor by slowly bringing a tag close to the reader and waiting for the RFID Qwiic reader buzzer to sound. A ruler was then used to measure the distance between the reader and the tag at the edge of the read range, which was found to be approximately 5 cm. This read range was much smaller than the spacing of the readers in the backpack which was found to be approximately 12 cm. While this confirmed that no interference would occur, we decided to place a block of aluminum on the back wall of the cases that held the sensors and LEDs, therefore blocking any radio frequency signals from adjacent RFID readers.

Regarding the light sensors, we required that they would be sufficiently sensitive to changes in the ambient light so that the system could differentiate between an open and closed backpack compartment. To verify this, we ran a simple Arduino program, which retrieved the illuminance of the ambient light the sensor detects, which was specifically visible and infrared light. We compared the readings of a slightly open backpack compartment to those of a closed compartment and found that there was a noticeable difference in the readings. The readings showed that a closed compartment would have an illuminance of less than 10 lux, which we set to be our threshold between an open and closed compartment. This confirmed to us that the light sensors were sufficiently sensitive to minor changes in light. However, we noticed minor issues with the performance of these sensors. Although they could detect small changes in light when the compartment was closed directly above them, they were unable to determine that a partially closed compartment, where one side was closed to an extent to where it covered the sensor, but the other was left open. We did not consider this to be an issue, as if the compartment was closed in this manner, it likely meant that the user was done using it. Another issue we found was due to the flexibility of the backpack. In some instances, this would result in a portion of the “ceiling” of the compartment covering the light sensor while the compartment was still open, which would turn off the LEDs while the compartment was still open. Nonetheless, this was not much of a problem, as the user would simply need to hold the compartment wall still when accessing it to prevent this from occurring.

3.4 Indication Subsystem Verification & Results

Testing of the indication subsystem was done by connecting the LEDs, resistors, and switches to an ESP32 development board through a breadboard. A simple program that sends a high and low signal to open or close the switches was performed. With 3.3 V being supplied from a power supply, we noticed that the LEDs turned on and off as the switches were closed and opened. To ensure that the LEDs would turn off automatically after 30 seconds, we simply ran another program on the ESP32 to open the switches after 30 seconds. By using a timer and noting when the ESP32 sent the signals to the switches using a serial monitor, we were able to confirm that the LEDs remained on for 30 seconds at a time. We also verified that the LEDs took less than 10 seconds to activate or deactivate after the ESP32 had sent the desired signals to the DIP switches.

3.5 User Interface Subsystem Verification & Results

The mobile application for the user interface subsystem was implemented using the MIT App Inventor software. The application needed to give users the ability to create three separate lists for three different compartments in the backpack. The option for adding unregistered items, adding registered items, removing registered items, and updating the names of registered items needed to be provided. In addition, the users needed to be able to connect to the microcontroller, update the status of items in the list, and be able to clearly see the status of each item. Verification for all these features, besides Bluetooth connectivity, was done independently of the ESP32 microcontroller. The need to scan a tag before registering a new item was later added, but for testing purposes, this feature was temporarily disabled. Instead, testing involved registering five items to each of the compartment lists. Adding, removing, and updating the names of registered items were done afterward. After some testing and debugging, we verified that these features were functional, so we then proceeded to add Bluetooth connectivity to the application.

The Bluetooth functionality was tested after the code for Bluetooth connectivity in the microcontroller was completed. Due to the precision of constant communication between the application and microcontroller, specifically with there being various signals being sent to and from each device, debugging and polishing both codes was somewhat challenging. However, we were able to successfully transfer data consisting of the names of registered items, their status, and their locations between the application and microcontroller. It was necessary for the application to provide the names of newly registered items or items that saw a name update. Upon adding items to the application, the necessary signals needed to be sent to the microcontroller, detailing what compartment the item was added in. This was also a requirement for when an item was removed from a compartment list. Finally, the application needed to be able to send a signal to the microcontroller when the user refreshes the status of the items in a list so that the statuses are updated. To prevent issues with connectivity from occurring on the side of the application, all features were disabled until it had connected to the microcontroller. If the user attempted to use them regardless of this, they would be notified with an error. These features were the last to be verified in the project, as the microcontroller needed to be able to interface with the RFID readers for these tests to be performed. After verification, we found that the application performed what was required of it, while also being user-friendly. A requirement we set was that we needed the status of the items to be updated on the application within ten seconds of a user prompt. We found that the application was able to perform these updates much faster than this, as can be seen in Table 1.

Table 1: Status Update Time for Different Number of Items in Application Lists

Number of Items	1	2	3	4	5
Update Time (s)	< 1	1.12	1.57	2.30	2.83

3.6 PCB Verification & Results

The first requirement is that all the components are correctly connected to match the schematics. We planned on verifying this by using a digital multimeter continuity test. However, this test failed because we had a few ungrounded components such as the microcontroller and the capacitor around the microcontroller’s voltage supply input. We also had misconnections such as the switching not being connected to the 3.3 V supply inputs. We fixed these issues by soldering some wires on the PCB. This was not elegant, and it made the PCB more fragile and harder to work with.

The second PCB requirement was that all the components had enough space to be soldered on the PCB. This was verified by placing the parts on the PCB to check their sizes and then the parts were soldered on the PCB. This verification was successful.

The third requirement is that all the components receive the required voltage under full load. This was verified using a digital multimeter. However, the RFID readers did not receive enough voltage under full load. This was because they were connected to a switch to turn them on or off. However, the switch's resistance was too large, which stepped down the voltage below the minimum level to make the RFID work. We fixed this by soldering a wire to supply the RFID readers directly from the voltage regulator instead of using a switch. This allows them to work, but it prevents them from being switched on or off. We do not think this was necessary because the RFIDs did not consume much power.

4. Costs

4.1 Parts

The list of parts is shown in Appendix F, in Table 8.

4.2 Labor

The average hourly wage of an electrical engineer graduate from the University of Illinois Urbana-Champaign is approximately $50 [1]. The project took 8 weeks to complete, and each member spent an average of 15 hours per week working on it. The labor cost per person can be therefore calculated to be 2.5 x $50/hr x 120 hr = $15000. There are two members on our team, resulting in the total labor cost of the team being around $30000. Summing this with the cost of the necessary parts results in a total cost of the project being approximately $30209.74.

5. Conclusion

5.1 Accomplishments

By the end of the project, we were able to meet the requirements of each subsystem individually. The RFID readers had a satisfactory read range, which allowed for relatively easy scanning of items as they were placed and removed from each of the compartments. The read range was also small enough to prevent any interference between readers. The light sensors were sufficiently sensitive and offered us the flexibility of controlling the illuminance threshold of the ambient light between an open and closed backpack. The timing of the activation and deactivation of the LEDs was as desired, and the placement of the sensors and LEDs in the backpack was successful.

We also saw success with our control subsystem. The microcontroller was able to send and retrieve the necessary data and signals from the other subsystems sufficiently quickly. The microcontroller's ability to interface with the mobile application through Bluetooth, which we found to be the most difficult part of the project, was also successful. The mobile application was also user-friendly and the sending of data between it and the microcontroller was faster than we had required. The speed at which the status of items listed in the application, in particular, was much greater than what we had decided upon in the initial design. Moreover, the power subsystem was able to operate as desired, and it supplied the subsystems with the required voltages. Finally, despite some connection issues, the PCB was mostly able to carry out the functions required of it.

5.2 Uncertainties

Throughout the duration of this project, we faced several issues and certainties in certain aspects of the design. We were able to overcome most of these problems, but some of them appeared towards the latter stages of the project, preventing us from fully satisfying the high-level requirements we had set. One uncertainty that caused us many difficulties was the presence of misconnections in the PCB we had designed. Initially, these issues prevented us from being able to program our microcontroller. We then found additional misconnections that prevented parts, such as the RFID readers, from being supplied with power. We were able to overcome these uncertainties by thoroughly debugging the PCB and identifying the exact locations of the misconnections. Through the use of jumper wires, we were able to fully connect the PCB and make it functional. However, the soldered jumper wires were somewhat loose, and this caused us trouble with debugging the system after connecting the PCB.

An uncertainty we were unable to solve before the project ended was with the packaging of the backpack. Due to the loose PCB connections, we found it difficult to properly enclose the PCB in the case we had designed. In addition, since the sensor and indication subsystems were isolated from the PCB and power subsystem, we needed long wires for connecting these components. We were able to find the necessary wires, but we were unable to enclose them in material, such as tubes, which left them exposed, making the final project less safe and consumer-ready than we had planned initially. Finally, the most significant uncertainty we faced, involved a misconnection in the power supply we had made late into the project. This resulted in some of our components, primarily the breakout muxes, to overheat and stop functioning properly. While the muxes remained operational, they were significantly damaged which made the connections between the PCB, RFID readers, and light sensors highly inconsistent. This was ultimately the main factor preventing us from fully fulfilling our high-level requirements by the project deadline.

5.3 Ethical considerations

In designing and implementing our project, it was crucial that we adhered to the proper safety and ethical practices. To that end, the IEEE Code of Ethics adopted in June 2020 provided us with important guidelines to be followed. Several of the guidelines mentioned in this code are applicable to our project, addressing aspects of safety and ethics. We attempted to follow these guidelines to the best of our abilities when designing and constructing our project.

1. Ensuring the safety of the users of this project [3]

It is likely that most of the users of this project will not be familiar with the technology implemented. The power subsystem and PCB, in particular, may cause harm to those who unknowingly tamper with them. The power subsystem in particular may cause harm to those who unknowingly tamper with it. Therefore, it was important that we isolate the PCB and power subsystem, as well as the other subsystems, from the users. With the aid of the machine shop, we were able to attach four cases to the backpack to prevent tampering. However, we believe that isolating the wires connecting the sensors and LEDs to the PCB was also necessary to improve the safety of the design.

2. Safeguarding the privacy of the users of this project [3]

One of the major components of this project is the mobile application, which would allow users to control which items they want to track. The application will therefore contain important information about the user, which can possibly be used against them maliciously. To safeguard the data of the users of the application, we ensured that all information was stored locally, either in the microcontroller or the application. None of the data pertaining to the items stored in the backpack was stored externally using a server.

5.4 Future work

While our project was mostly successful, there were certain aspects of the design that could have been improved. One such aspect is the read range of the RFID readers. Although we found it to be satisfactory, the read range of the readers could have been improved by 2-3 cm to allow for better scanning of items placed further away from the readers in the compartment. This increase in range could be achieved by supplying the readers with more voltage. Another improvement that could be made to the project would be the addition of better connectors between the PCB and power supply to prevent safety issues with connections. A possible connector that could be used instead of a simple two-pin header would be a barrel connector. Modifications could also be made to the application to allow for a better user experience. Specifically, the location of items displayed as misplaced was not shown, leaving users needing to search the other two compartments to locate them. Therefore, adding the location of the misplaced item would help make organizing the backpack much easier. Furthermore, the data sent between the application and the microcontroller could be encrypted using encryption software to prevent potential attacks directly on the backpack. Finally, firmer backpack material could be used or a means of stabilizing the walls of the backpack containing the RFID sensors could be implemented. This is due to these walls bending more than desired, which caused the process of scanning items as they are placed and removed from the backpack somewhat difficult, especially when the backpack was fully open.

References

[1] “Graduate Electrical Engineer Salary in Illinois,” ziprecruiter.com, https://www.ziprecruiter.com/Salaries/Graduate-Electrical-Engineer-Salary--in-Illinois#:~:text=How%20much%20does%20a%20Graduate,be%20approximately%20%2449.36%20an%20hour.

[2] “*Parents Spend $26.7 Billion in Back-to-School; 80% of Children Will Lose Pricey Supplies, Lunch Boxes and Clothing,” prnewswire.com, https://www.prnewswire.com/news-releases/parents-spend-267-billion-in-back-to-school-80-of-children-will-lose-pricey-supplies-lunch-boxes-and-clothing-219434601.html (accessed Feb. 15, 2024).

[3] “IEEE Code of Ethics,” ieee.org, https://www.ieee.org/about/corporate/governance/p7-8.html.

[4] S. Kim, “An approximate approach to determining the permittivity and permeability near λ=2 resonances in transmission/reflection measurements,” *Progress In Electromagnetics Research*, pp. 95–109, Jan. 2014.

[5] A. R. Jabur, “Effect of polyaniline on the electrical conductivity and activation energy of electrospun nylon films,” *ScienceDirect*, vol. 43, no. 1, pp. 530–536, Jan. 2018.

[6] “Electrical Conductivity and Resistivity for Aluminum and Aluminum Alloys,” nde-ed.org, https://www.nde-ed.org/NDETechniques/EddyCurrent/ET_Tables/ET_matlprop_Aluminum.xhtml.

Appendix A: Requirement and Verification Tables

Table 2: Control Subsystem Requirements and Verification

Requirements	Verification	Verification Status (Y or N)
• The microcontroller should be able to read all three RFID readers' signals at the same time.	• Place a tag in one compartment. • Check if the microcontroller can read that tag ID. • Place three more tags, one in each compartment. • Check if the microcontroller can read all three tag IDs.	Y
• RFID Qwiic readers must be able to store the data of at least five different RFID tags at a time.	• Place five 125 kHz RFID tags in the same compartment of the backpack. • Check if the first reader was able to read each tag by using an oscilloscope to measure the waveform of the SDA (the address & the data) and SCL (the clock) pins in the RFID Qwiic reader. • Confirm that all five RFID tags are present on the Qwiic reader stack using the necessary I2C commands.	Y
• The microcontroller should store up to 15 RFID tag IDs along with their names and locations in the backpack	• Register 15 items in the microcontroller • Check if the microcontroller still has the 15 items' information stored in its flash memory.	Y
• The microcontroller should compare the read RFID tag ID to the list of registered RFID tags IDs to determine if any item is missing. • The microcontroller should turn on the LEDs in the indication subsystem	• Register 15 items in the microcontroller. • Place 14 out of the 15 registered items in the backpack one by one. • The red LED signal for the compartment with an item missing should be HIGH. The other two red LED signals and the green LED signal should all be LOW. • Place the 15th tag in the backpack. • The red LED signal should all be LOW, and the green LED signal should be HIGH.	Y
• The microcontroller should transmit 2.4 GHz Bluetooth signals to the user interface subsystem.	• Register one item in the microcontroller • Place the registered item in the backpack • Use a phone (or any other Bluetooth device) to read the Bluetooth signal. • The signal should specify the name and	Y

	location of the tag.	
• The signal to activate the green LED must be sent only when all the desired items are present and are in the correct compartments (as specified in the application). The green LED must not be active if there are items present in the backpack that have not been added to any list in the application, or if the backpack is closed. The green LED must take a maximum of ten seconds to activate and deactivate after the necessary conditions have been met.	• Ensure that the backpack is empty and that the mobile phone used to access the application is no more than two meters away from the backpack. • Add a single item to any list on the application. • Place the item listed on the application in the corresponding compartment in the backpack and ensure that its RFID tag was successfully scanned. • Immediately start a timer to measure the time taken for the green LED to light up. Confirm that this time is no more than ten seconds. • Remove the item from the list on the application and immediately start a time to measure the time taken for the green LED to light up. Confirm that this time is no more than ten seconds.	Y
• A red LED must turn off only if all the items specified in the application are present in the corresponding compartment. If an item not specified in the application is placed in the compartment, the red will remain on, even if all the items listed in the application are present. • The red LEDs must take a maximum of ten seconds to activate and deactivate after the necessary conditions have been met.	• Ensure that the backpack is empty and that the mobile phone used to access the application is no more than two meters away from the backpack. • Place an RFID in an arbitrary compartment and ensure that it was successfully scanned. • Immediately start a timer and measure the time taken for the red LED corresponding to that compartment to light up. Confirm that this value is no more than ten seconds. • Add the item placed in the backpack to the list corresponding to the compartment in which it was placed in. • Immediately start a timer and measure the time taken for the red LED to turn off. Confirm that this time is no more than five seconds.	Y

Table 3: Power Subsystem Requirements and Verification

Requirements	Verification	Verification Status (Y or N)
• The power subsystem should output a constant voltage of 3.3V with less than 3 mV peak-to-peak noise at a maximum current of 500 mA.	• Connect the power subsystem to a power supply set to 100 mA on constant current mode. • Use a digital multimeter to measure the voltage of the power subsystem. • Verify that the DC component is 3.3 ± 0.1 V. • Verify that the voltage ripple is less than 3 mV peak-to-peak. • Increase the current of the power supply by 100 mA increments. • Verify that the DC component is 3.3 ± 0.1 V. • Verify that the voltage ripple is less than 3 mV peak-to-peak.	Y

Table 4: Sensor Subsystem Requirements and Verification

Requirements	Verification	Verification Status (Y or N)
• RFID readers must only scan the items of the compartment they are allocated to. The RFID readers must also not interfere with one another.	• Place two RFID ID-12LA readers, contained in the containers described in the Physical Description section, within 2 ± 0.1 cm of each other. • Connect a multimeter to measure the voltage across the "Tag in Range" pin of each reader. • Ensure that the voltage across this pin is 0.0 ± 0.1 V for both readers. • Place a 125 kHz RFID tag within range of one of the RFID readers. • Check if that reader was able to read the tag by confirming that the voltage across the "Tag in Range" pin is 3.3 ± 0.1 V. • Check if the other reader was not able to read the tag by confirming that the voltage across the "Tag in Range" pin is 0.0 ± 0.1 V.	Y
• The light sensor must be able to determine when the backpack is open or closed by differentiating between the light	• Hold the light sensor in place near the top of an open backpack compartment. • Confirm that the light sensor was able to identify that the backpack is open by using an oscilloscope to measure its SDA and	Y

intensity of these two states.	SCL waveforms. • Close the backpack compartment while keeping the light sensor near its top. Use a cloth to cover open areas if necessary. • Confirm that the light sensor was able to identify that the backpack is closed by using an oscilloscope to measure its SDA and SCL waveforms.	
• The RFID readers must be able to scan a tag within two seconds of the tag entering its range.	• Connect a multimeter to measure the voltage across the "Tag in Range" pin of an RFID ID-12LA reader. • Ensure that the voltage across this pin is 0.0 ± 0.1 V. • Place a tag within range of the reader and immediately start a timer. • Confirm that the time needed for the voltage across the pin to reach 3.3 ± 0.1 V is no more than two seconds.	Y

Table 5: Indication Subsystem Requirements and Verification

Requirements	Verification	Verification Status (Y or N)
• The LEDs must be able to remain active continuously for 30 ± 5 seconds.	• Supply 3.3 ± 0.1 V to one of the LEDs. • Ensure that the LED turns on and immediately start a timer. • Confirm that the LED can remain continuously on for 30 ± 5 seconds. • Repeat these three steps for the remaining three LEDs.	Y

Table 6: User Interface Subsystem Requirements and Verification

Requirements	Verification	Verification Status (Y or N)
• Each item's status must be updated at least once every ten seconds if the user is at a maximum of at least 2 ±0.1 meters away from the backpack. • The time at which missing items were	• Ensure that the mobile phone used to access the application is 2 ±0.1 meters away from the backpack. • Add a single item to any list on the application. Confirm that the item is shown as *missing* on the application. • Scan the RFID tag of the item listed on the application with the RFID reader of the corresponding compartment in the	Y

removed from the backpack should be displayed once their status changes from *present* to *missing*.	backpack. • Immediately start a timer and measure the time required for the *missing* status to change to *present* on the application. Confirm that this time is no more than ten seconds. • Scan the RFID tag with the same reader and note the hour and minute when the scan was complete. • Immediately start a timer to measure the time taken for the status of the item on the application to be changed from *present* to *missing*. Confirm that this time is no more than ten seconds. • Also confirm that the hour and minute appear once the status has changed to *missing*. Ensure that they are the same as what was noted earlier. • Immediately start a timer to measure the time taken for the status of the item on the application to be changed from *present* to *misplaced*.	
• Users must be able to add and remove items from at least three different backpack compartments. Users must be able to add a maximum of at least five items to a given list.	• Ensure that the mobile phone used to access the application is no more than two meters away from the backpack. • Add five unique items to each of the three lists on the application. • Scan the RFID tag of each item with the RFID reader corresponding to the compartment it is assigned to in the application. • Confirm that the application shows that all 15 items are present. • Scan the RFID tag of one item from each compartment with the RFID reader corresponding to that compartment. Confirm that the application shows that they are missing, while the remaining items should remain displayed as present. Also confirm that the hour and minute displayed are the same as when the scan was complete. • Scan each of these items with a reader corresponding to a compartment that they were not assigned to. • Confirm that the application shows that those specific items are misplaced, while the remaining items remain present.	Y
• Users must be able to add items and track items that have not been	• Ensure that the mobile phone used to access the application is no more than two meters away from the backpack.	Y

previously tracked.	• Add a new item to any given list on the application. Confirm that it is shown as *missing* in the application. • Scan the RFID tag of that item with the RFID reader corresponding to the compartment it is assigned to. Confirm that the item is shown as *present* in the application. Also confirm that the hour and minute displayed are the same as when the scan was complete. • Scan the RFID tag of the item with the same RFID reader. Confirm that the item is shown as *missing* in the application. • Scan the RFID tag of the item with an RFID reader corresponding to a compartment that it was not assigned to. Confirm that the item is shown as *misplaced* on the application.	

Appendix B: PCB Subsystem Schematics and PCB Layout

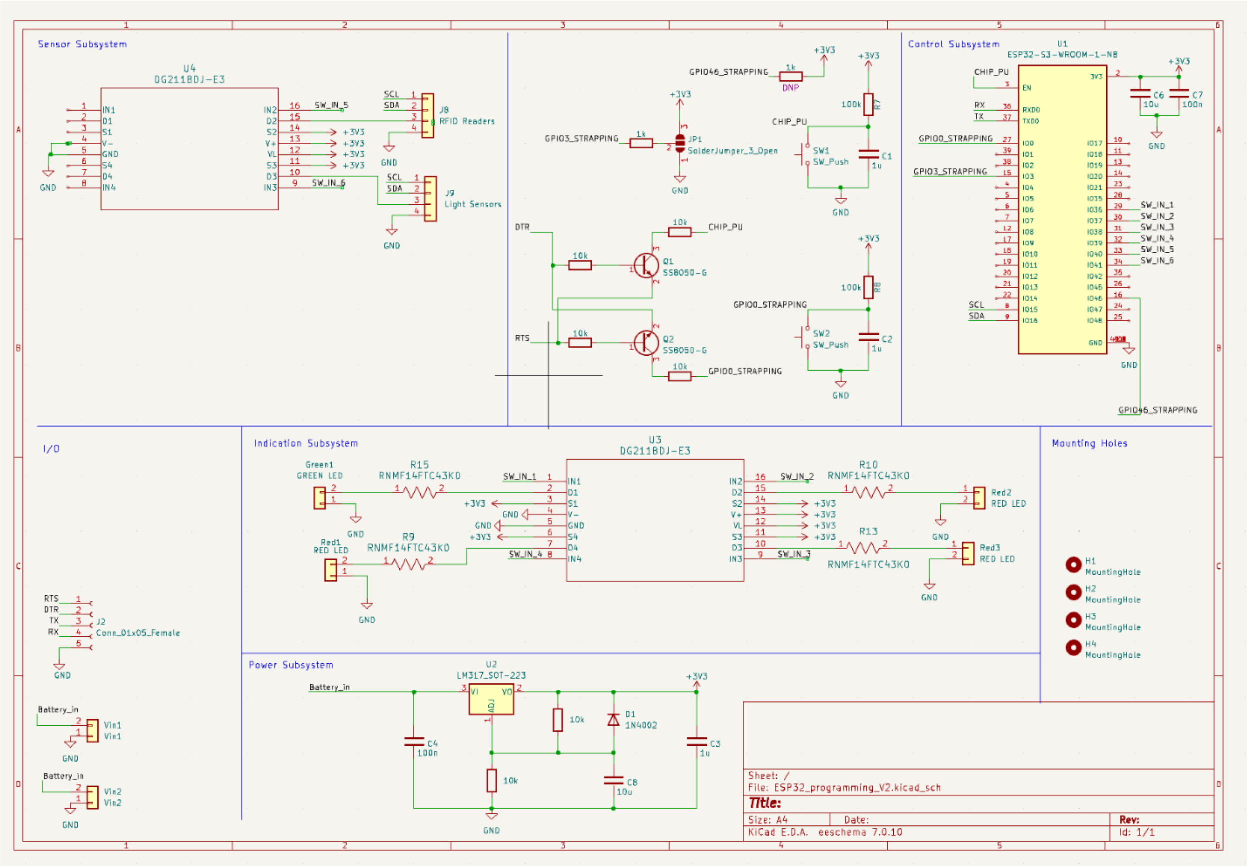

Figure 5: Final PCB Schematic

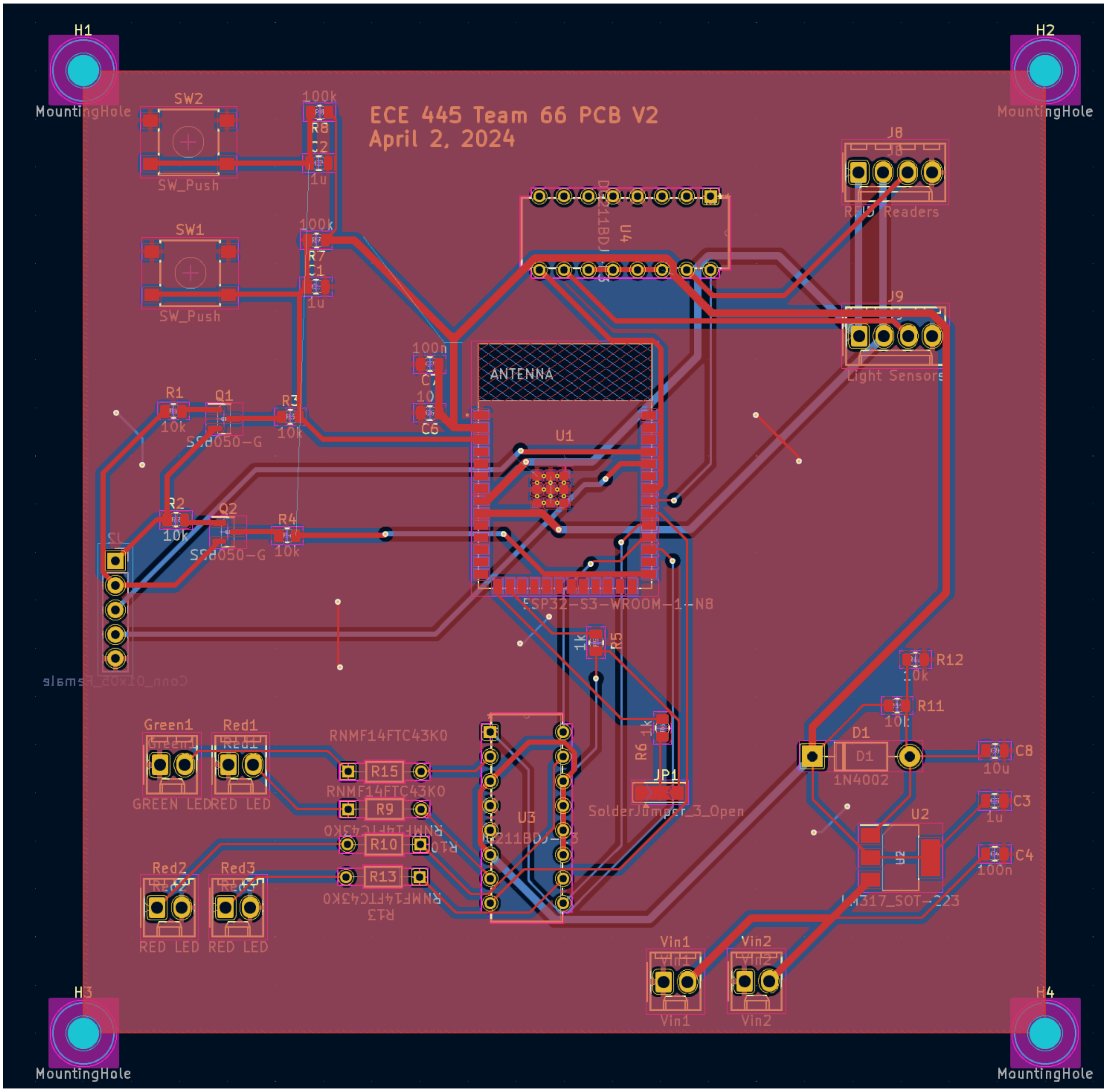

Figure 6: Final PCB Layout

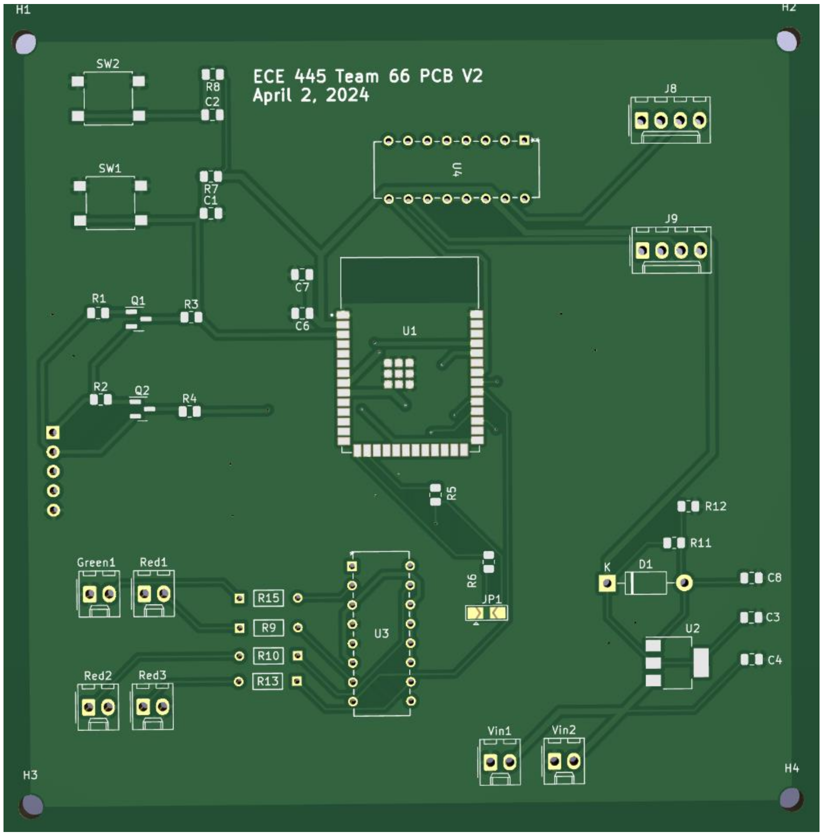

Figure 7: Final PCB Design

Appendix C: Project Visual Aid

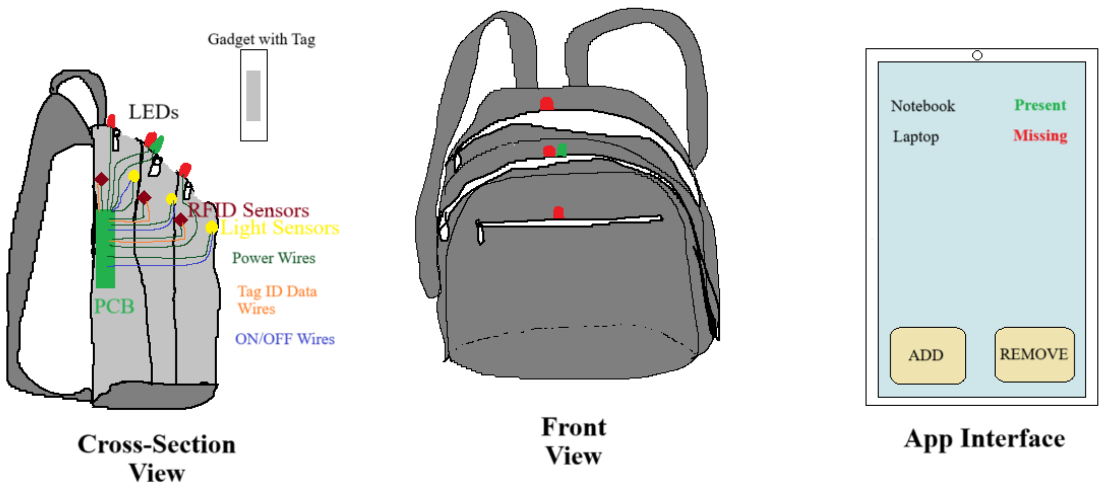

Figure 8: Project Visual Aid

Appendix D: Project Schedule

Table 7: Project Schedule/Timeline

Week	Task	Person
2/19 - 2/25	Finish design document	Everyone
	Make modifications to proposal and resubmit	Everyone
	Order parts	Abdullah
2/26 - 3/3	Finish first PCB design	Everyone
	Complete Design Review	Everyone
	Protype Microcontroller	Everyone
	Provide Machine Shop with necessary parts	Abdullah
	Protype Power Subsystem	Raef
3/4 - 3/10	Pass audit and order PCB	Everyone
	Protype RFID readers	Abdullah
	Start programming RFID Qwiic readers	
	Get Machine Shop updates	
	Start programming microcontroller	Raef
3/11 - 3/17	Assemble PCB and start testing	Everyone
	Make modifications to PCB design	
	Program RFID Qwiic Readers	Abdullah
	Test RFID readers	
	Order new parts if necessary	
	Finish programming microcontroller	Raef
3/18 - 3/24	Order second PCB	Everyone
	Start code for mobile application	Abdullah
	Get Machine Shop updates	
	Program and test light sensors	Raef
3/25 - 3/31	Assemble second PCB and test	Everyone
	Make final changes to PCB design	
	Continue working on mobile application	Abdullah
	Get Machine Shop Updates	
	Test indication subsystem	Raef
4/1 - 4/7	Make final PCB order	Everyone
	Finish mobile application design and code	Abdullah
	Test mobile application	Everyone
4/8 - 4/14	Assembly of final PCB	Everyone
	Assembly of entire system	
4/15 - 4/21	Debugging and final fixes	Everyone
4/22 - 4/28	Final demo	Everyone
4/29 - 5/1	Final Presentation and Final Paper	Everyone

Appendix E: Microcontroller Flow Chart and Code

The entire code is in Github:

https://github.com/Raefmm/ECE445.git

Here are a few snippets of the main loop in the code:

```
void loop() {

  for(int i = 0; i < 4; ++i){
    leds[i] = false;
  }

  bool valid;
  uint16_t visible_plus_ir[NUMBER_OF_SENSORS], infrared[NUMBER_OF_SENSORS];

  for(int i = 0; i < NUMBER_OF_SENSORS; ++i){


    myMuxLight.setPort(i);
    myMuxRFID.setPort(i);

    if (ltr[i].newDataAvailable()) {
      valid = ltr[i].readBothChannels(visible_plus_ir[i], infrared[NUMBER_OF_SENSORS]);
      if (valid) {
        Serial.print("CH0 Visible + IR: ");
        Serial.print(visible_plus_ir[i]);
        Serial.print("\t\tCH1 Infrared: ");
        Serial.println(infrared[NUMBER_OF_SENSORS]);
      }
    }
    //RFID stuff
    delay(300);
    scan(i+1);
  }




  //Indication Subsystem (LEDs) turn on
  CheckLEDs();
  //Indication Subsystem (LEDs) turn on
  CheckLEDs();

  for(int i = 0; i < NUMBER_OF_SENSORS; ++i){
    if(visible_plus_ir[i] > LightThreshold){


      if(timerReadSeconds(Timer0_Cfg[i]) > 30) {
        continue;
      }

      timerStart(Timer0_Cfg[i]);

      if(leds[i]) digitalWrite(LED_ADDR[i], HIGH);

      if(i == NUMBER_OF_SENSORS-1){
        if(leds[3]) digitalWrite(LED_ADDR[3], HIGH);
      }
    }
  }

  //Bluetooth Stuff
  Blueread();

  save_values();
  showStatus();


  // Indication subsystem turn off
  delay(500);
  // Indication subsystem turn off
  delay(500);

  for(int i = 0; i < 4; ++i){
    digitalWrite(LED_ADDR[i], LOW);
  }


  for(int i = 0; i < NUMBER_OF_SENSORS; ++i){
    if(visible_plus_ir[i] <= LightThreshold){

      timerWrite(Timer0_Cfg[i], 0);
      timerStop(Timer0_Cfg[i]);

      digitalWrite(LED_ADDR[i], LOW);
      if(i == NUMBER_OF_SENSORS - 1)  digitalWrite(LED_ADDR[3], LOW);

    }
  }

  Serial.println();
}
```

Figure 9: ESP32 code snippet

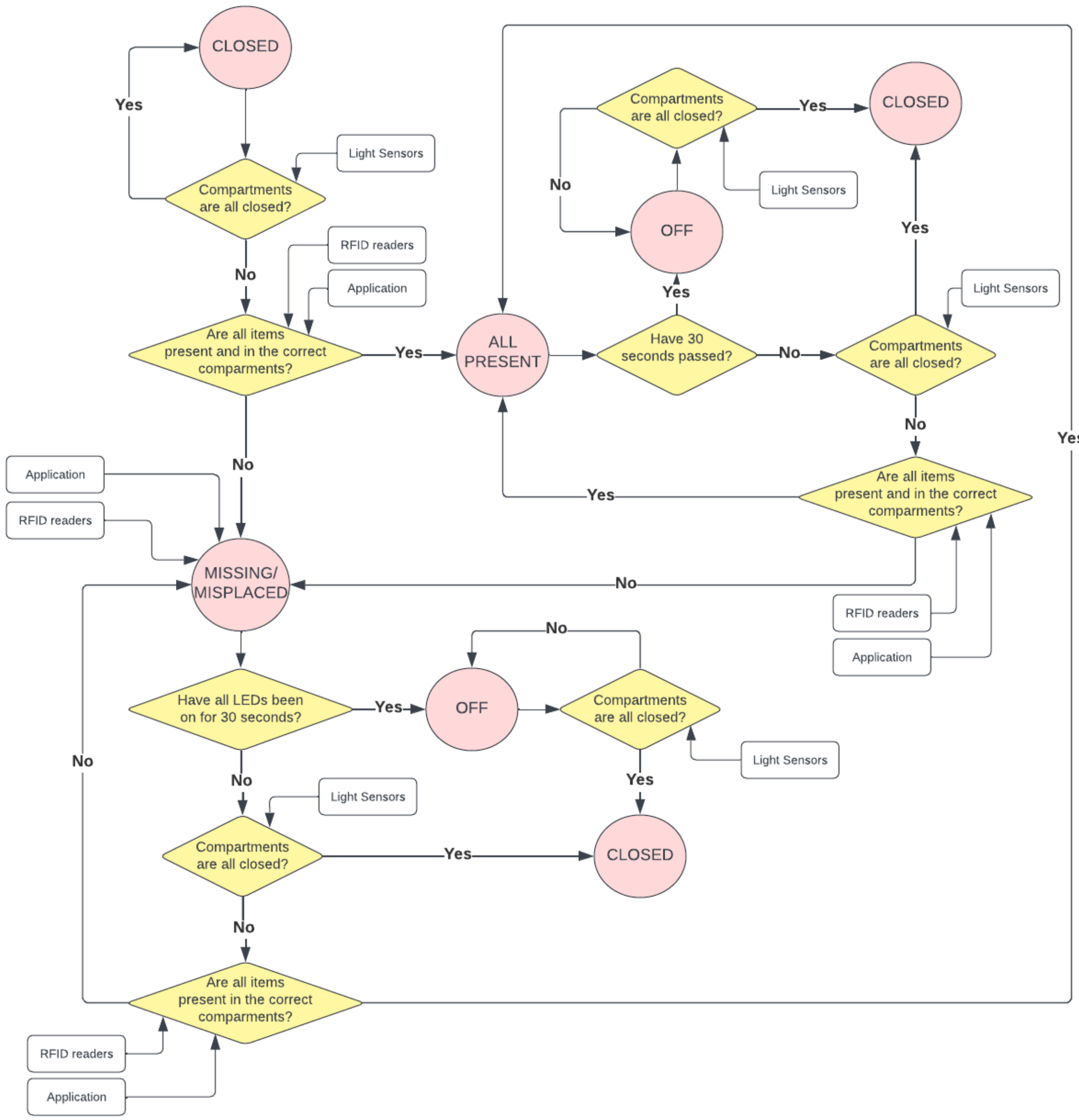

Figure 10: Real-time tracking flow chart

Appendix F: Cost of Parts

Table 8: List of Components and Prices

RFID Reader ID-12LA (125 kHz)	SEN-11827	SparkFun Electronics	3	97.50	Link
SparkFun RFID Qwiic Reader	SEN-15191	SparkFun Electronics	3	64.50	Link
SparkFun Qwiic Mux Breakout	BOB-16784	SparkFun Electronics	1	12.95	Link
ISO RFID Tag (125 kHz)	RFID 125-ISO	Olimex LTD	15	8.25	Link
5 MM Green LED	SSI-LXH600GD-150	Lumex Opto/Components Inc.	1	1.32	Link
5 MM RED LED	SSI-LXH600ID-150	Lumex Opto/Components Inc.	3	3.75	Link
ESP32-S3 microcontroller	ESP32-S3-WROOM-1-N8	Espressif Systems	1	3.20	Link
LM317 Voltage Regulator	LM317DCYR	Texas Instruments	1	0.77	Link
AA Battery	815 BULKJ2	Energizer Battery Company	6	1.80	Link
Ambient Light Sensor	5591	Adafruit Industries LLC	3	13.50	Link
Total				209.74	

Appendix G: Images of Project and Subsystems

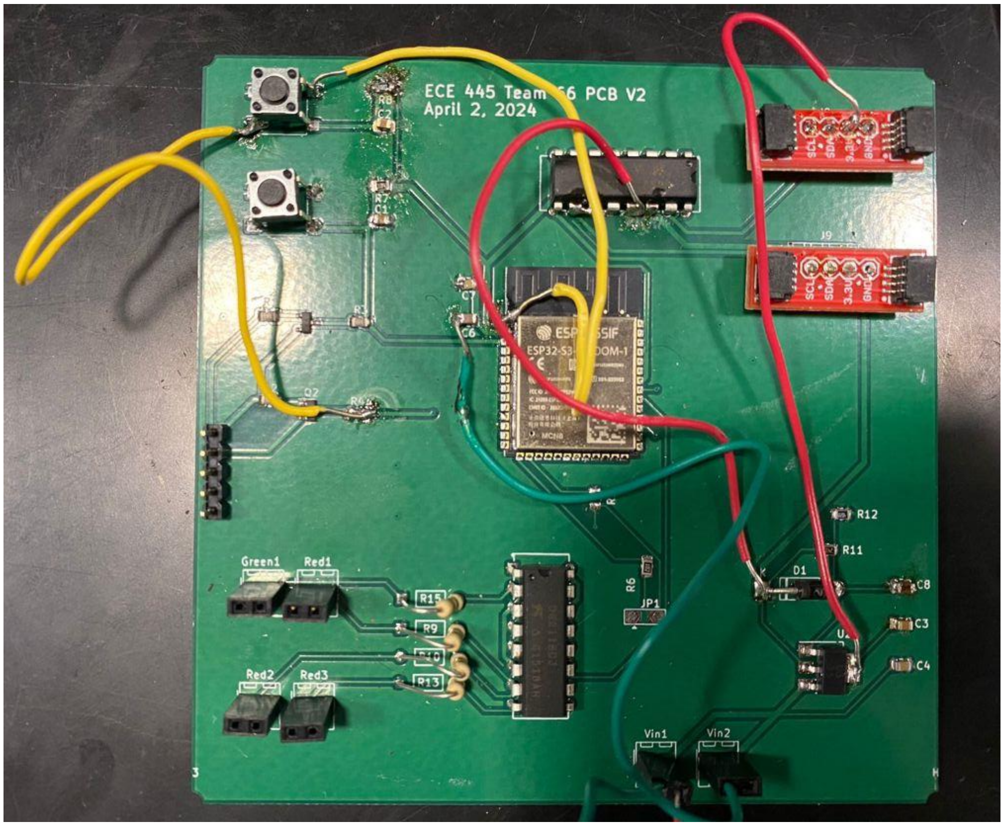

Figure 11: Final PCB With Soldered Components

Figure 12: Final Assembled Project

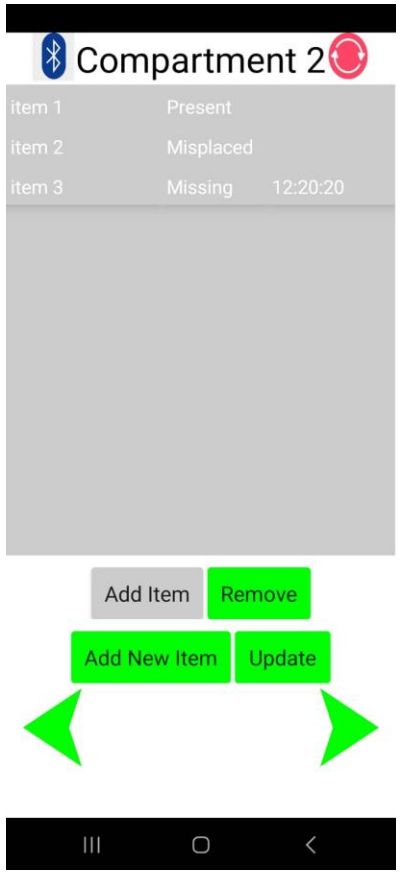

Figure 13: Application Page Layout

Appendix H: Simulations

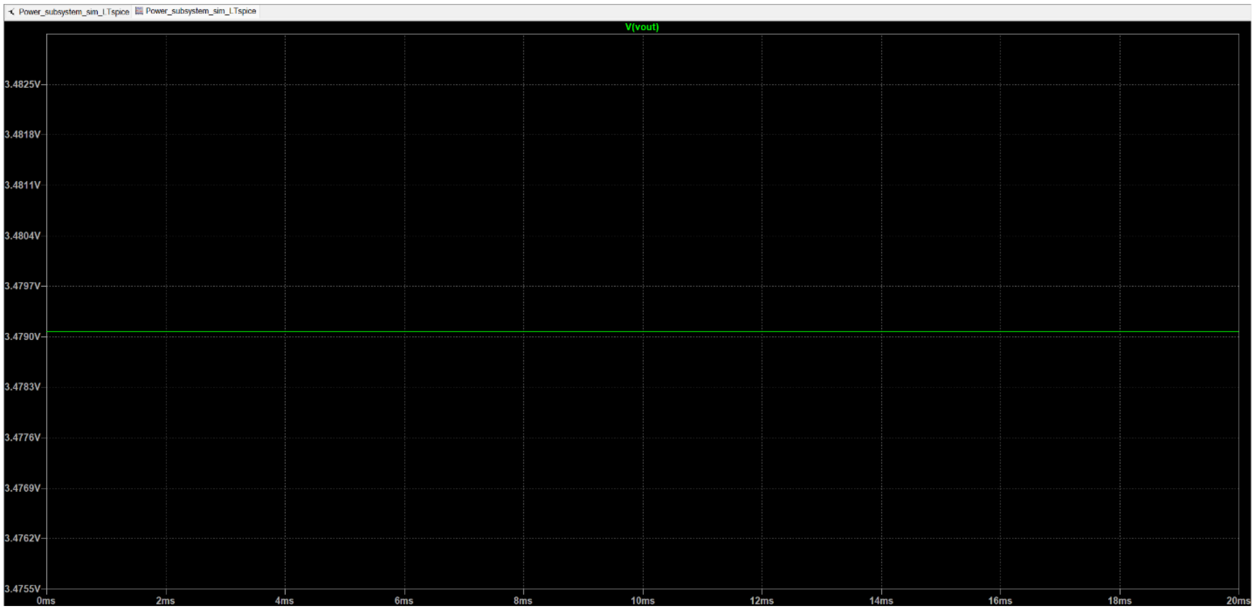

Figure 14: Simulation of the power subsystem