UCF Senior Design 2: Study Buddy

Final Document

Dr. Lei Wei

Dr. Samuel Richie

Department of Electrical Engineering and Computer Science

Group 33

Cliff Porter	Computer Engineering
Kaiden Duvall	Computer Engineering
Denzel Stockhausen	Electrical Engineering
Idris Al-Shidhani	Electrical Engineering

Table of Contents

List of Figures

List of Tables

1.0 Executive Summary

Among the many problems that students face, one of the most obvious and personally impactful to each is the presence of distractions that can pull attention away from their studies. This issue has been exacerbated in the modern age, with many students having easy access to handheld devices, such as smartphones, that serve as a tempting distraction from their work. While many may consider this a problem relating purely to individual discipline, it seems certain that isolation from easy distractions should coincide with an increase in productivity.

Currently, there are very few mainstream products that seek to actually help solve this general issue. Some companies sell lock boxes made specifically for phones, but these products are often not designed around security, usually being made of thin plastic and having easily exploitable locking mechanisms. These properties allow these lock boxes to be used as effective self discipline tools, for individuals who simply need a reminder to not get distracted while working. However, for parents seeking to improve their child’s studying habits without constant supervision, such a device is insufficient, as it is easily bypassed.

We have designed an easy to use device that can securely store distracting objects away in a locked container with multiple compartments. The device is unlockable either by setting a timer when locking that will automatically unlock a compartment, or through several access methods such as a keypad, a fingerprint scanner, and an RFID scanner, all built into the device and configurable via a display. This “Study Buddy”, as we are calling it, also contains lid sensors that can detect if the container has been opened forcefully or prematurely and log the time of access to display to the owner of the device, as well as USB ports to allow devices to charge while being inaccessible to the owner. We have also designed and constructed a remote that can be used to monitor the device’s functionality from a distance.

In the following document, we will describe our goals for the Study Buddy’s design and the specific requirements we intended for it to meet when completed. We will then delve into the technical investigation our team did to learn about the technologies we used and the parts we used to construct the device. Then, after overviewing the general standards and constraints that apply to the device we have created, we will detail the design process of our hardware and software, in addition to including the schematics and block diagrams used in the design process. Finally, after detailing our testing of components and construction of the final product, we will include the financing and milestone information of the project as a whole.

2.0 Project Description

This section will give a high level overview of our project, its motivation, goals, requirements and some basic information on its design.

2.1 Project Motivation:

Procrastination is a well known and extremely common issue for a large number of people. Many don't necessarily feel motivated to avoid work and other responsibilities, but end up doing so regardless due to the large number of distractions present in everyday life. In the modern era in particular, the ubiquity of smartphones in many people's lives has given many easy access to social media and entertainment at any time, which acts as a strong temptation when one is meant to be working or studying.

Many current commercial products that approach this issue exclusively target smartphones. We have created a more general solution in the form of a locker that can hold other potential distractions, such as laptops and game controllers. In doing so, we created a product with several possible uses, including self discipline, parental control, and storage. The larger size of the Study Buddy in comparison to available phone lock boxes has allowed us to create a more flexible device that includes many methods of access that can be configured by the user.

The intended customer base for this product is parents, who may use the Study Buddy as a parental control tool to help ensure their child is studying, or to enforce device limitations on phones and gaming systems. An additional customer base is highschool and college aged students who could use the Study Buddy as a self-discipline tool to keep their phones and other devices out of reach and promote a more efficient studying environment. This audience extends into the general workforce, as older individuals may encounter similar problems with distractions while working and may find value in the Study Buddy's design.

2.2 Project Goals/Objectives:

Our primary goal in designing the Study Buddy was to create a secure container that can be used flexibly by the owner to improve their habits, enforce rules, or safely store objects. To meet this goal, we aimed for the following objectives:

- Create a container that cannot be easily opened nondestructively without having access to the required authentication factors
- Include three separate storage compartments that can be individually configured
- Include several authentication factors that a user can select from to make the Study Buddy as secure as they would like
    - Keypad entry (something you know)
    - An RFID reader (something you have)
    - ADVANCED GOAL: A fingerprint sensor (something you are)

- Include a timer that will unlock the device after a set period of time set by the owner
- Include both a wall charging method and internal battery in the main device to allow flexibility and avoid problems stemming from power loss
- STRETCH GOAL: Include an emergency opening method based on messages or calls received

In addition, we designed a remote that can be used to wirelessly view the compartment timers in a short range, as well as informing the user when the main container (the “locker”) is locked or unlocked. This makes parental control easier, and also allows for the locker to be in a different room while the user studies. The remote is battery operated to allow it to be easily taken with the device’s owner.

2.3 Requirement Specifications:

Table 1 shows the requirement specifications of the Study Buddy and its remote.

Table 1: Project Requirements - demonstrable requirements in gray

Feature	Feature Description	Units
Basic Requirements		
Empty weight	Be easy to carry	≤ 10 lbs
Carry capacity	Hold a decent amount of weight without damage	≥ 15 lbs
Large storage locker size	Store a standard 15in laptop	≥ (16in x 12in x 2in)
Smaller storage lockers size	Store phones or gaming controllers.	≥ (10in x 4in x 6in)
Locker battery life	Run without external power for the specified time	≥ 2 hours
Remote battery life	Run for the specified time	≥ 7 days
DC Power regulation	Running from standard wall AC power (120V AC 60Hz) and backup power (battery pack)	≥ 80% conversion efficiency
Wireless range	Distance the remote and locker can remain connected	15 feet
Tamper detection	Detect unauthorized opening of compartments and provide a means for the owner to be notified	90% reliability
Wireless latency	Updates between the remote and locker	≤ 400 ms
Security	Users can access device configuration on the locker via keypad PIN number and RFID scanner.	90% - 100% reliability
Advanced Features		
Wireless range	Distance the remote and locker can remain connected	40 feet
Power for phone chargers	Include 3 USB ports for phone charging	Three 5V 2A USB wires
Advanced security	Users can access device configuration on the locker via fingerprint access.	80% reliability
Stretch Goal Requirements		
Emergency override	Detect when a user might need emergency access to their phone and unlock	65% reliability

2.4 House of Quality:

Table 2 is the house of quality for the Study Buddy. This table lists several of the factors that we considered in the design process of the device, split into engineering requirements and marketing requirements. Arrows indicate the direction and magnitude of correlation between two requirements.

Table 2: Study Buddy House of Quality

			1	2	3	4	5	6	7
1		↓↓	↑	↑	↓	↓	↓
2							
3				↑↑	↓	↓	
4					↓		
5						↓↓	↓↓
6							↓

Row #	Direction of Improvement	Column #	1	2	3	4	5	6	7
		Direction of Improvement	▼	▲	▼	▼	▲	▲	▲
		Customer Requirements / Engineering Requirements	Cost	Sensor reliability	Dimensions	Weight	Remote battery life	Power efficiency	Wireless range
1	▼	Cost	↑↑	↓	↑	↑	↓	↓	↓
2	▲	Ease of use		↑↑	↓	↓	↑		↑↑
3	▲	Security	↑↑	↑					
4	▲	Portability			↑↑	↑↑			↑↑
5	▲	Storage space	↓		↑↑	↓			
6	▲	Battery life	↓			↑	↑↑	↑↑	↓
7	▲	Remote functionality	↓	↑↑			↑↑		↑↑
		Target for Engineering Requirements	≈250$	> 80%	> (16in x 12in x 2in)	< 10 lbs	≈ 7 days	> 80%	≈ 20 ft

Our house of quality highlights some key considerations we made while designing the Study Buddy to ensure it was practical to create while also satisfying the needs of a potential market. The most obvious and directly correlating factor is cost, as the higher the cost of materials for us, the higher the cost to a final consumer would be. This simply serves as a reminder that we must be mindful of the overall cost of parts and materials for our project, as making it too expensive would be inconvenient for both us and someone who would want to buy a finished product.

Additionally, it is important to note that the device's ease of use is significantly improved if the sensors are more responsive and reliable. We ensured not only that our sensors and methods of entry function properly, but also that they are intuitive for the user and do not noticeably impede usage of the device. To quantify this, we measured the response times of our entry methods, door sensors, and locking mechanisms to ensure we are creating a product that is easy to use.

2.5 Prototype Mock-up

Figure 1 below shows a rough drawing of the device as we initially intended it to look. It shows the positions and rough sizes of the different compartments, as well as the locations of the display, the RFID reader, the keypad, and the fingerprint sensor for our advanced goal. Note that the relative sizes of the compartments and the positions of the unlocking methods are not exactly accurate to those in the final design, and that the thickness of the compartment walls was not considered in this mock-up.

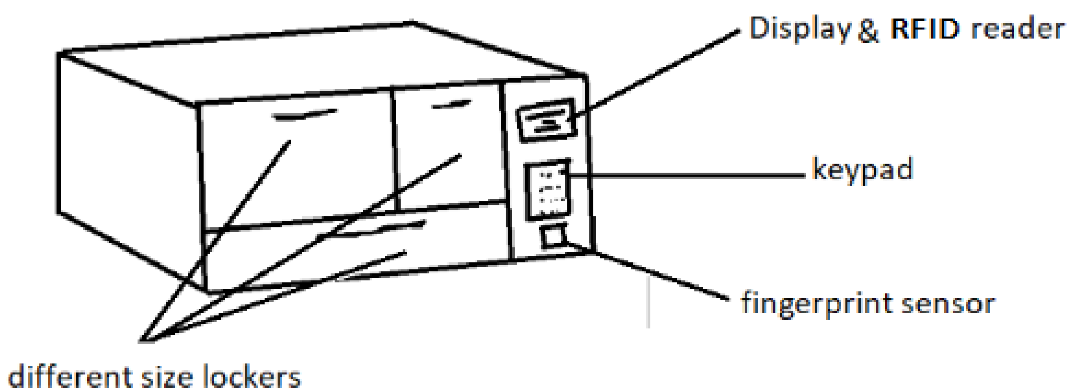

Figure 1: Initial Mockup of the Locker / Study Buddy

2.6 Design Block diagrams

The following figures are the hardware and software visualization of how our system operates based on our goals and specifications. These are diagrams based on the functional final version of the design, as opposed to the initial block diagrams used in the design phase.

2.6.1 Locker Hardware Design

Figure 2 below shows the hardware block diagram for the Study Buddy's locker. The power system and components are on the left, with the MCU in the center and the input/output peripherals, such as the unlocking devices and locking mechanism, branching off of it and the connected GPIO expander. The connections between the control unit and the locking mechanism, display, RFID sensor, and fingerprint sensor are both power and data, as a signal is required to toggle the state and actuate the device controlling the lock and the other components require power.

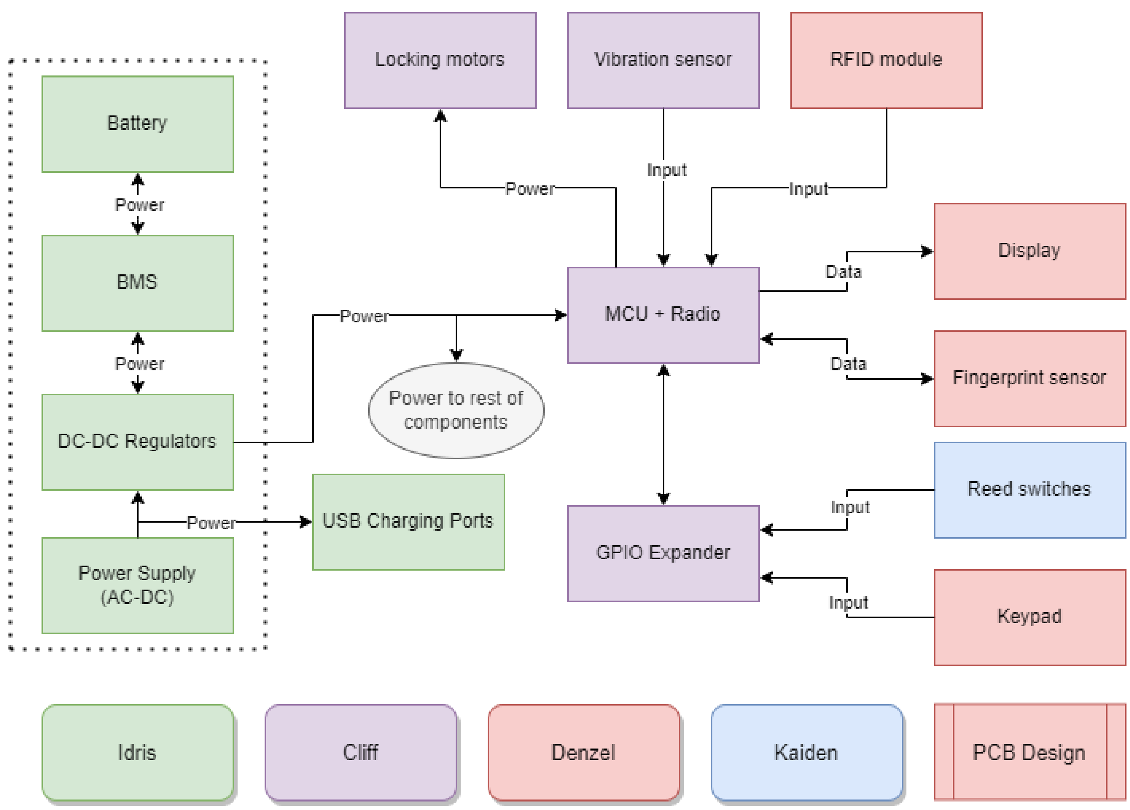

Figure 2: Locker / Study Buddy Hardware Block Diagram

2.6.2 Remote Hardware Design

Figure 3 below shows the hardware block diagram for the Study Buddy's remote. The diagram shows the relatively simple general design of the Study Buddy remote. The control unit, powered by a battery, communicates with a display, as well as being connected to a button interface for user input. The control unit contains a wireless radio that allows communication between the remote and the locker, with information such as the current lock states and remaining time being shown to the user on the small display. In addition to data, the connection between the control unit and both the display and radio also indicates power transfer, as both will require power to function.

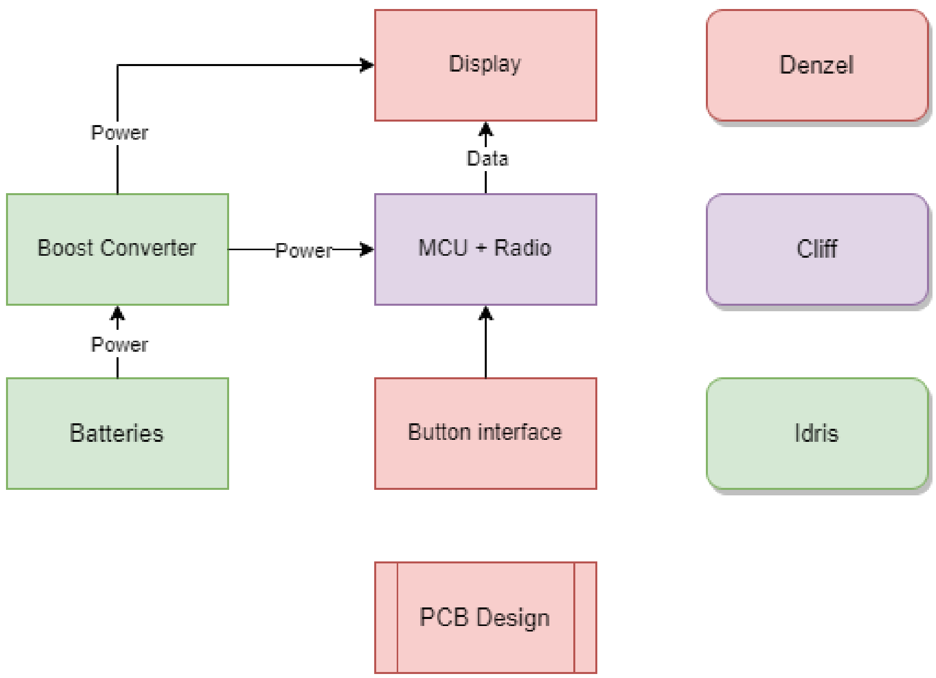

Figure 3: Study Buddy Remote Hardware Block Diagram

Because the Study Buddy's remote requires a method of communicating with the locker, a button interface, and a display, we designed a separate, smaller PCB for the remote using the same MCU as the locker to allow for more effective software design and ensure compatibility. The designs of both the main locker PCB and the smaller remote PCB will be elaborated on in the hardware design section of this document.

2.6.3 Software Design

Figure 4 below shows the software block diagram for the Study Buddy's locker. The software functions as a state machine, with the menu being displayed at any time and the functions of that menu being the primary content of each state. The configuration options on the left are all optional, but at least one must be selected before the compartment can be locked. The unlock function can be called through the manual unlock process, via the call vibration system, by a timer reaching 0, or though an override command from the remote.

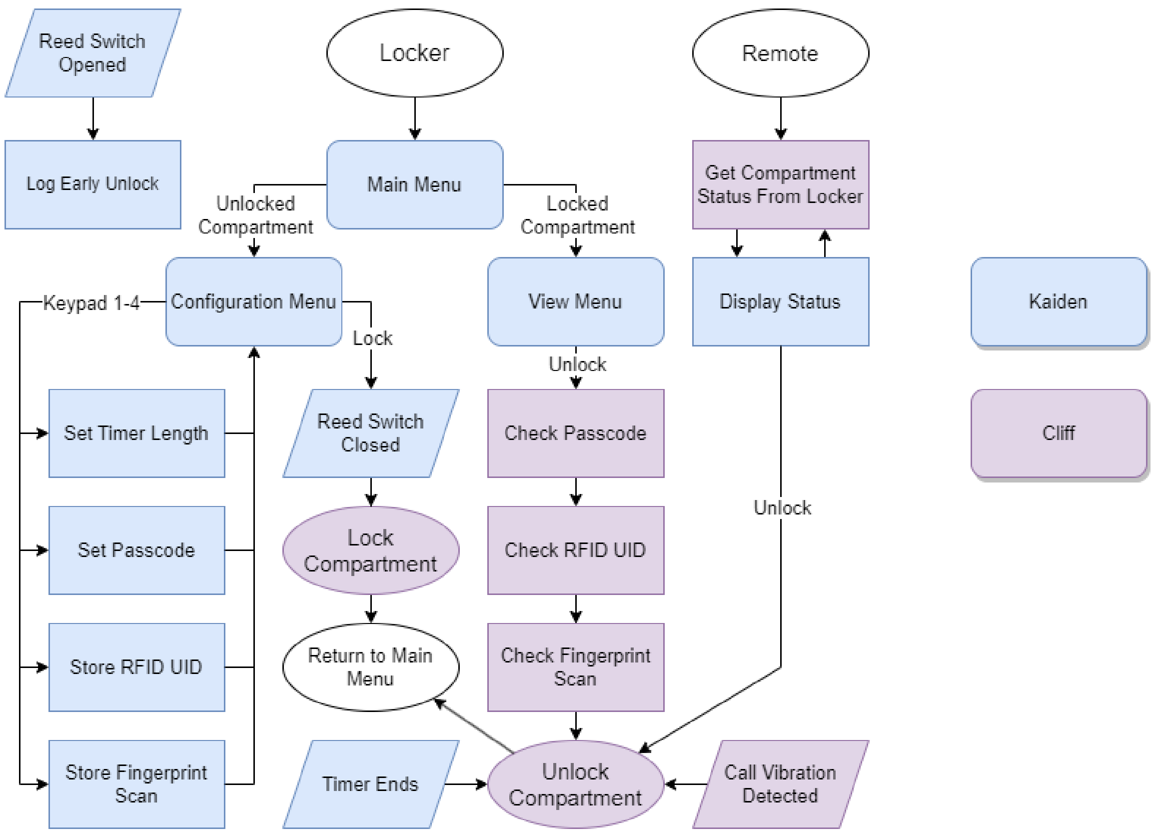

Figure 4: Study Buddy Software Block Diagram

3.0 Technology Comparison and Part Selection

The following section describes the research we did after our initial design concept was finished. Getting a workable prototype depended on each component that was chosen. Each component had to first undergo extensive investigation, and those that were chosen had to adhere to the same standards as the others. The design process would have been prolonged, and the project's cost would have increased if pieces were chosen that didn't function together. The three most crucial steps were, as with most projects, quality, time, and budget. The three requirements in the project's overall design were met by carefully studying and choosing the appropriate components.

3.1 Comparison to Existing Projects

<u>Smart Lock: UCF Senior Design Project Summer 2019</u>
One recent senior design project carried out by UCF students involved designing a lock used for household security. The locking and unlocking mechanisms used were very similar to the ones we used in our project including the use of a RFID sensor, a fingerprint sensor, and keypad input. The key difference in the design of these projects is that theirs focused much more on security as it was meant to guard something of more importance. Our project focuses more on affordability as it will be used on locking objects for short periods of time. We still learned from the troubles the group had in designing these systems and employed more efficient practices.

Figure 5: Smart Lock Senior Design Project

<u>Concentraining System: UCF Senior Design Project Spring 2021</u>
Another recent senior design project carried out by UCF students involved locking a user's phone and making use of eye tracking software to motivate them to stay focused on their task. The project is similar to our own in that it involves a lock box designed to store an object that may distract the user in their attempts to concentrate, but differs in that it focuses much more heavily on having the user stay concentrated rather than keeping the object away from them. Our project is of more use for parents who wish to keep objects away from their children rather than personal use, though that is still a potential use.

Figure 6: ConcenTraining System Senior Design Project

<u>Market Phone Lockboxes</u>
There are several kinds of Phone Lockboxes available on the market which allow the user to place their smartphone into a small box whose lid is secured by a latch set to release after the timer expires. These products were actually part of our inspiration for this project as we wanted to improve on the idea to create a more versatile and configurable version to be more useful in a variety of specific situations, as opposed to having the configurability of a kitchen timer. We also wanted to make our device support more kinds of distracting devices such as laptops and game controllers. In general, we were disappointed in the limited options that a small, phone shaped box with an egg timer inside gave us.

Figure 7: ySky branded ‘Portable Smart Phone Timer Lock Box’ on Amazon

3.2 Wireless Standard

Since our project involves wirelessly communicating between two devices, we needed to select a wireless standard to use. This section will compare what we discovered when researching different wireless standards and explain our choice to use in our project.

After researching popular wireless standards, we identified five that we could use for our project. WiFi, Bluetooth, Bluetooth Low Energy (BLE), Zigbee, and a proprietary protocol using the Nordic Semiconductor nRF24L series of radio modules. When choosing between these we needed them to meet our design requirements for range and low power draw, but also researched their available platforms, development environments and the amount of documentation, resources, and community support each has. These secondary considerations were very influential in our decision as we found that using some of these standards would require using hardware and software platforms which we had no experience in.

Since we wanted to spend the least amount of time and resources learning about how to use each chip and standard we chose, and instead work on developing our design, we favored the standards that were supported by some of the technologies we were already familiar with. This led us to favoring standards that were supported in the MSP430 family of microprocessors, letting us use Code Composer Studio or, were supported in the Arduino ecosphere and would let us use the Arduino IDE.

The following sections detail a summary of our findings on each wireless standard followed by a summary explaining our selection.

3.2.1 Wi-Fi

Wi-Fi is probably the most widely used wireless standard currently. It is most used to connect portable devices to local networks and the internet. There are many different protocols used to send data over Wi-Fi, however, these all work much higher on the TCP/IP stack and are implemented in the application layer. This means that, fundamentally, all communication protocols that use Wi-Fi are also TCP/IP based.
For our project, we needed to be able to send information between our two microcontrollers, which would either require them to both be on the same Wi-Fi network, or for one microcontroller to be connected to a network being generated by the other. Since we only wanted the communication between the two devices, we would have had to use the ladder option, which could limit our range due to the low power transmitter of available microcontrollers. Overall Wi-Fi is a much heavier protocol that requires a sizable amount of overhead.

<u>Range:</u>
Wi-Fi has a diverse spectrum of ranges. Most applications involve the use of an Access Point (AP) to act as the transmitter. In general APs have high ranges due to their high-power radios and can get ranges of up to 300 ft outdoors. Due to its extensive use, Wi-Fi devices can easily achieve ranges of the required 20-50ft.

<u>Platform support:</u>
Wi-Fi also has an extremely diverse range of supported devices. Almost all internet capable devices have Wi-Fi capabilities as well. As a result, there were plenty of options to research for use in our project. The two most common results were single board computers such as the Raspberry Pi or Nvidia Jetson Nano and embedded processors such as the ESP32/8266.

While single board computers provide plenty of power and flexibility, they have large power requirements, run a full operating system and are expensive. Since our project does not require practically any intense computing resources, we decided we would be much better suited using an embedded processor.

In terms of embedded processors with Wi-Fi capabilities, the clear choice is an ESP32 or ESP8266 module which include their own antenna in the PCB. This platform is the most widely used for Wi-Fi based DIY projects online and has plenty of support for just about anything. It also supports using the Arduino IDE, which was a plus for our familiarity with it.

<u>The Client/Server Model and Wi-Fi:</u>
As mentioned before, Wi-Fi’s primary use is to allow devices to connect to the internet over TCP/IP. As a result, the majority of Wi-Fi based communication follows a client/server model.

This posed a bit of trouble for our application since we wanted to be able to send messages directly with C code while the majority of Wi-Fi based applications send data between client and server programs over the application layer of the OSI model. Doing so required the device to support TCP/IP, usually at the operating system or driver level. Most online resources regarding the ESP32 and ESP8266 involve creating client and server applications on top of some form of base code or operating system to handle the TCP connection.

This level of involvement is much more than we needed for our project and unnecessarily increased the complexity of developing our code.

Power usage:
Wi-Fi is also poorly fitted for low power applications as it requires so much overhead to maintain the signal. Maintain the Wi-Fi signal as the client can draw up to 50mA continuously and disallows putting the processor to sleep. Trying to use sleep modes to save on power would have interfered with our ability to communicate between our devices, and the high power usage would have offset any amount we could save by sleeping.

Summary:
Overall, Wi-Fi is a very robust and versatile wireless standard. It has great range depending on how it is implemented, has plenty of support on the internet and is supported in the Arduino Ecosphere. However, due to its high-power usage, high overhead, and confusing implementation, we chose not to use Wi-Fi to communicate between our devices for this project.

3.2.2 Bluetooth

Bluetooth has quickly become the go-to wireless standard for communication between devices. It is most popular with devices that want to communicate wirelessly and run-on batteries such as smartphones, wireless headphones, computer mice and even car tire pressure sensors. It has become so popular due to its low relative power consumption and simplicity, allowing for wireless devices with usable battery lives.

Range:
Bluetooth has had many revisions over the years with each improving range, so depending on the hardware used and the versions of bluetooth it supports, one could get ranges between 25 ft and a few hundred feet. As a lower powered standard, it is also highly susceptible to power losses through walls, where range drops to the 10-15 ft range. Line of sight range is usually closer to 30-50 ft in most cases.

Introduced in the Bluetooth 5.0 specification is a long range mode which can stay connected from over 1000 ft line of sight. If we could have easily implemented this mode, then it would have been more than sufficient for our project.

Platform Support:
As a widely used standard, Bluetooth is supported by many different microprocessor platforms. There are several Bluetooth modules such as the HC05, which connect to a microcontroller over USART and can be controlled with a library. They have plenty of support in the Arduino Ecosphere as well.

Some microcontrollers support bluetooth natively as well, such as the ESP32/8266 or the nordic nRF52 series of microcontrollers. Both of which support using the Arduino IDE and have libraries for sending data over Bluetooth.

Power usage:
Due to its lower overhead than Wi-Fi, Bluetooth allows using sleep modes to save power when we aren't sending data. Even during continuous transmission, the average power draw is in the 10-20mA range, so by employing heavy usage of sleep states and infrequent transmission,onee should be able to reach the ideal ~1-3mA average range to achieve a long battery life.

Summary:
Bluetooth has plenty of support in the platforms we are familiar with, and would let us save power for our battery powered devices and made for a good candidate for use in our project. Our concern for it was meeting our advanced range goal, since we could not find a clear answer about how easy it would be to achieve a 40-50ft range or if the long range mode is supported by readily available libraries.

3.2.3 Bluetooth Low Energy (BLE)

Bluetooth Low Energy (or BLE) is a more recent addition to Bluetooth, introduced in the Bluetooth 4.0 specification. Its goal is to improve power savings for devices with smaller batteries. Recently, it has greatly increased in popularity for use in small tracking devices and other sensors due to its very low power usage, allowing for these devices to be only slightly larger than the coin cell batteries that power them. It has also caught the eyes of people looking to add wireless communication to their low power DIY projects.

Range:
We could not find much information on the specific range capabilities of BLE due to a lack of information on the specifics of BLE over using Bluetooth. We did find that there are not supposed to be significant differences in range capabilities between the two however, and one should be able to implement BLE to get the power saving benefits while still meeting our range goals.

Platform Support:
BLE is also supported by the ESP32 and nRF52 platforms in the Arduino IDE. However, as it is newer, online support is more sparse and hard to find information on. There is plenty of information regarding using and implementing BLE communication, but information regarding specific power consumption and range was lacking before testing ourselves. Fortunately, both of these platforms support both Bluetooth and BLE, so selecting one would not prevent us from testing both and comparing their ranges.

Power usage:
Compared to Bluetooth proper, BLE is optimized for small bursts of data transmission instead of continuous transmission. Depending on the implementation, BLE can use half to 1% as much power as Bluetooth.

Summary:
BLE seemed like a great standard to use for our project, as it uses less power than Bluetooth, and should be able to achieve the same range. We were able to use a programming environment we were familiar with, there is lots of support online, and every platform we found that supports it also has plenty of support for Bluetooth, so if we could not meet our range goal, we could switch to using Bluetooth fairly easily.

3.2.4 Zigbee

Zigbee is one of the less widely used standards we considered for use in our project. Its main idea is to create a mesh network of devices to increase range while still using minimal power. Recently it has become a popular standard amongst home automation devices due to these properties. It uses a message system and each device uses nearby, always powered router devices to relay its message to the network coordinator. It can be used to communicate on a two device network so long as one of them is a Zigbee coordinator.

Range:
Zigbee has an advertised range of about 30-300 ft indoors, and can easily support ranges around our required 50 ft. A device's range depends on how much it has been set to conserve power, but it should have worked well for our needs. In normal applications, a device usually only needs to send a signal to the nearest repeater, but our application would not be able to utilize this feature of the standard.

Platform Support:
In our research, we only found one platform that supported Zigbee and that we could find online support for, the nRF52 series of processors. These chips by

Nordic Semiconductor support multiple 2.4GHz wireless standards like Bluetooth, BLE, Zigbee and Thread. Nordic has an nRF5280 Development Kit available for testing and development and companies like Adafruit also make boards using the nRF52840, but Zigbee development is only available through Nordic's NRF Connect SDK application stack. Using this SDK was not preferable as we would not be able to utilize all of the online resources for all the devices in the Arduino ecosphere which we planned on using.

Power usage:
Zigbee is excellent for power savings. Included in Nordic's Zigbee development application stack is a tailored real time operating system and additional automatic power saving mechanisms. If we had developed our project using their software, we likely would have had no problems with power usage.

Summary:
Due to the lack of support in the Arduino echosphere, we did not use Zigbee in our project. Despite its excellent power savings and acceptable range, we could not find a way to incorporate Zigbee into our project without the use of the nRF52 platform, which does not support Zigbee without the use of the NRF Connect SDK.

3.2.5 Nordic ESB Protocol Using nRF24L01 Radio Modules

The Nordic Enhanced ShockBurst Protocol is the wireless standard associated with the popular nRF24L01 SPI radio module. This module is, or was, widely used in DIY embedded projects to wirelessly send small amounts of information. It is a cheap and simple way of adding wireless communication to projects and also has a few variants that add antennas and special amplifiers to improve range.

Range:
This module can achieve up to 200 ft of range with its basic, onboard antenna version. Variations with an external antenna improve this range and the version with additional amplification circuits can achieve over 2000 ft in open conditions. Online resources for this module report ranges above 30 ft, meaning it would not have been difficult to hit our 40 ft range requirement.

Platform Support:
Since this module uses SPI to send and receive data from the microprocessor, it supports all of the platforms we considered when researching. Along with having plenty of online resources, tutorials, and troubleshooting information, it also has pre-made Arduino libraries available to ease development.

Power usage:

The nRF24L01 has a quite considerable power draw. Using about 10mA when sending or receiving, it is slightly higher than other low power standards. It does support low power sleep modes which can be used between transmissions to save power.

Summary:
Due to the large amount of support we could easily find for using the nRF24L01 module in the Arduino IDE, and its sufficient range and low power consumption, It would have made for a great choice to use as our wireless standard. It also uses SPI to communicate with our MCU of choice, which would have given us more freedom in MCU selection.

3.2.6 Comparison and Standard Selection

In researching all of the wireless standards we identified, we learned that there are a lot of different ways we could implement the wireless communication we need for this project, even without comparing between different standards. We also discovered that many of the platforms that support these different wireless standards include the radio inside the chip, which means that choosing a wireless standard would limit the MCU platforms available to us.

The only standard we thought would not work for our project was Wi-Fi, which had good support and range but too much overhead, and thus, power consumption to work for our project since our remote needs to have a week long battery life. Wi-Fi was also the standard that we found was supported on the least amount of platforms (only really available on the ESP32 and ESP8266 microcontrollers). So, we chose not to consider Wi-Fi for our project.

Zigbee was an appealing choice for us due to its good range and very low power consumption. However we had a very hard time researching it due to nearly all the information on using it in microcontroller projects being from the manufacturer of the more popular chip that supports it. Much of our research difficulties might be attributed to the fact that the chip's main advertised feature is support of BLE instead of Zigbee and that it is really popular among hobby microcontroller board manufacturers as a BLE capable board supported in the Arduino IDE. Apart from the lack of non-manufacturer resources, the fact that we could only use the Zigbee functionality of the chip using the official non-Arduino SDK was the main reason we chose not to use Zigbee since we wanted to conserve as many resources we could when learning how to use each component in our project.

Bluetooth seemed like a decent choice for our project since the HC05 Bluetooth modules had plenty of support in the Arduino IDE. We decided against choosing bluetooth directly since these modules consumed a lot of power and all of the platforms that supported it also supported BLE.

This leaves us with BLE and using nRF24L01 modules. These were our two top choices for wireless standards as they both have good range, low power usage, support sleep modes to achieve extremely low power usage, can be used in the Arduino IDE and we could find plenty of online resources for using them. We were leaning slightly more towards the nRF24 due to it using SPI and letting us choose any MCU for our project. Most of our research found that the nRF52840 was the best MCU for on chip wireless and if we chose to use Bluetooth or BLE, but due to the possibility that we may have to buy a blank chip directly from a chip supplier, we would likely have to figure out how to load the correct firmware in order to program our device using the Arduino IDE. If we chose the nRF24, we would have been able to use officially supported chips and avoid the information cross pollution involving Nordic's nRF52840 information and the Arduino affiliated manufacturers information on their own Arduino IDE compatible nRF52840 based boards.

Table 3 shows a summary of our findings and justifications to standard selection. Overall, our top two choices for wireless standard are BLE and the nRF24 modules for the above reasons. The choice of which one we ultimately use depended on the amount of resources we could find on using blank nRF52840 modules, how difficult they would be to flash/program in the Arduino IDE and other hardware based considerations such as supported sleep and low power modes. If we had found a better MCU than the nRF52840, we would have chosen to use the nRF24 modules instead of BLE. But, if we did choose to use the nRF52840, we would have had the option to compare their performance in testing and choose the standard that gave us the best results and was easiest to implement.

Table 3: Wireless Standard Comparison

	WiFi	Bluetooth	BLE	Zigbee	nRF24L01
Practical range	50 to >100 ft	30 to 50 ft	30 to 50 ft	30 to 50 ft	50 to >100 ft
Transmission power	~100mA	~40mA	~10-20mA	~1-2mA	~15mA
Supports low power mode?	No	No	Yes	Yes	Yes
Can be used in the Arduino IDE?	Yes	Yes	Yes	No	Yes
Online resources?	Many	Many	Some	Few	Many

3.3 MCU

Our microcontroller unit is one of the most important components in our project. Just about every other component interacts with it, and most are controlled by it. Depending on the microcontroller platform we chose, we would have had to develop our project very differently. As such, there were many things to consider when we were selecting our MCU.

Our main priority for selection was how difficult it would be to understand how to program and write code for it. Mainly, we wanted to choose an MCU that is well supported in the Arduino IDE, as it is one of the largest resources for premade microcontroller and peripheral libraries. Having access to these libraries, and the wealth of knowledge built up around the Arduino IDE would help us greatly in getting started with our project quickly so we would not have to worry about meeting our requirements and have time to work on our advanced and stretch goals.

We also wanted to select a platform that has plenty of online resources (both official and from the community). Having access to this information would be important when we started working on our design and needed to answer a question, troubleshoot, or figure out how to accomplish something. If we could not find an online resource due to selecting an MCU platform with poor support, we ran the risk of wasting time getting stuck on the problem.

Our selected MCU also needed to support the wireless standard we selected above. This is far less of a concern as one of our selections uses the common SPI bus, but we had to weigh its compatibility against any possible difficulty we might have faced dealing with the platform. This point only affects the nRF52 series, and we will discuss it later.

With that said, in our research, we found four different, well supported platforms to consider; the MSP430 Family from Texas Instruments, the RP2040 from The Raspberry Pi Foundation, the ATmega family from Atmel, and the nRF52840 from Nordic Semiconductor.

3.3.1 MSP430 Family - Texas Instruments

The MSP430 series is a family of microprocessors made by Texas Instruments. It consists of many different microprocessors with different package types as well. The main microprocessor we researched was the MSP430FR6989, which we had experience working with from our previous coursework. It is very well documented and we already had experience looking though its datasheets to configure its peripherals. However, our experience with the MSP430FR6989 was very tedious and we had to write lots of code to configure basic things like setting a timer, or putting the processor to sleep for a certain time. As stated above, our goal was to reduce the amount of time we spent ‘reinventing the wheel’ so we could focus on the functionality of our project. We didn’t believe that the MSP430 family was a great choice for this project.

Online resources
When researching we tried to figure out how we could do certain things related to our project such as interface with our peripherals using the MSP430FR6869. We could not find much information directly about what we were hoping to do due to how diluted the MSP430 family is across all of its different variations. The best we could find is someone's GitHub page of a library for using the nRF24L01with the MSP430. We could not tell if it would be supported on the chip we were familiar with.
Overall it seemed that there was not much specific information, guides or libraries for more or less exactly what we would be doing.

Is it supported in the Arduino IDE?
As part of the Texas Instruments ecosystem, the MSP is programmed using the Code Composer Studio IDE, and is not supported by the Arduino IDE. While we were familiar with Code Composer Studio, we knew how to do low level functions such as configuring timers by manually setting its memory variables. Without well supported, pre-made libraries, we would not choose to use the MSP430 Family in our project.

Sleep modes
The MSP430FR6968 does support processor sleep modes and would allow us to enter the deep sleep mode required for a long battery life in our remote.

3.3.2 RP2040 - The Raspberry Pi Foundation

Known for their Raspberry Pi single board computer series, the Raspberry Pi Foundation has recently developed a cheap yet powerful new microprocessor, the RP2040. It is the center of their microcontroller board, the Raspberry Pi Pico, but since then, other popular microcontroller board manufacturers have developed boards featuring the RP2040. Released in early 2021, it mainly supported python but also had a C SDK, albeit with less user friendly resources and plenty of documentation. Support for it was quickly added into the Arduino IDE but it still remains a new and un-matured technology, at least compared to the amount of beginner friendly support that exists for Arduino boards. It is a capable chip that is very affordable and currently readily available, though certain features still require lower level programming to configure.

Online resources
Many of the online resources for the RP2040, namely the Raspberry Pi Pico, were very outdated. Since the platform is only two years old, a lot of the

resources online are from only a few months after its release, so many of the resources result in note that a feature is still being worked on.

Is it supported in the Arduino IDE?
Yes, currently RP2040 based boards are supported in the Arduino IDE. Some of the advanced features are still half-baked though. Due to its young age, some of the features we might have needed for our project did not have easy to use function calls or were not yet available entirely.

Sleep modes
In its earlier implementation of the Arduino IDE integration, it appears that sleep modes were not yet supported. More current information still points back to it not yet being supported. While it is possible to set sleep modes using the official C SDK, it does not appear to be supported yet in the Arduino IDE.

3.3.3 ATmega - Atmel

Core of the Arduino lineup of microcontroller boards, the ATmega series is likely the most popular DIY embedded platform. The chip we researched was the ATmega328, which is the 8-bit processor inside the Arduino Uno and Nano boards. The Arduino Uno is one of the older embedded learning and development boards and due to its own, and its accompanying IDE's popularity, many libraries, modules, peripherals and example programs were made for it. Today, the Arduino lineup and its IDE sit close to the center of the DIY embedded community. Due to this popularity, and Arduino's tendency to create new boards around other chips, non Atmel embedded platforms have been added to the 'Arduino Echosphere' so to speak. Devices officially (and unofficially) supported by the platform are easy to develop and work with, especially the mature ATmega lineup.

Online resources
The ATmega328 probably has the most online resources of any embedded device. Most tutorials about how to use embedded related devices use the Arduino Uno as its example microcontroller. Along with countless articles and video tutorials, the Arduino forums are a great resource since so many people use it, we would be likely to find a solution to any problem we ran into.

Is it supported in the Arduino IDE?
Yes, as the flagship microprocessor, the ATmega328 is truely, officially supported in the Arduino IDE.

Sleep modes
The ATmega328 does have low power and sleep modes we could easily use to achieve our remote's battery life requirement.

3.3.4 nRF52840 - Nordic Semiconductor

The nRF52840 is part of Nordic Semiconductor's nRF52 lineup of 2.4GHz microcontrollers. It features an on chip radio, so modules with included antennas are ‘out of the box’ capable of using wireless standards like Bluetooth, BLE, Thread, Zigbee and NFC. It has gained notoriety due to its use by popular microcontroller board manufacturers to create BLE capable boards. These boards, namely made by Adafruit and Arduino, both have their full schematics and board files available.

Online resources
When searching for the nRF52840 specifically, there were few to no resources available online other than those from Nordic. These resources are often nonspecific and unhelpful. However, due to the popular boards that use the nRF5280, there are resources from their respective manufacturers and the communities that use them asking questions and messing around with the hardware. These boards also have access to the general information regarding libraries and examples made for the Arduino IDE.

Is it supported in the Arduino IDE?
Officially it is supported by the Nordic NRF Connect SDK, which is required to implement wireless protocols like Zigbee or Thread but using supported boards (for firmware) it can be used in the Arduino IDE to develop using BLE. Most notably, the Arduino Nano 33 BLE uses the nRF52840, and officially supports BLE in the Arduino IDE. Either by referencing the

Sleep modes
This chip does support sleep modes, but due to the on chip power saving hardware, it already has a really low power draw. And more power can be saved by disabling a few of the peripherals.

3.3.5 Comparison and Selection

After researching our selected MCU platforms, we found that one of the largest factors to us for choosing which platform to choose was the amount of resources we could find online. We learned in our embedded systems class that having to reference a chips datasheet to accomplish something simple like activating sleep, a timer or using UART is tedious and an unproductive use of time. This is why we focused so much on if we could use the Arduino IDE and all of its supporting libraries. Some of us knew from experience that having to troubleshoot an issue without adequate online help can be time consuming and even demoralizing.

We also checked whether the boards we were looking at had their schematics, board file, and other supporting documentation available. Fortunately, all of them did, which will give us something to reference when designing our PCBs.

From past experience, we decided to rule out the MSP430FR6989 due to being locked to Code Composer Studio and lack of libraries and online information.

The RP2040 was a pretty compelling option to us for all of its benefits on paper such as its low cost, good performance, support in the Arduino IDE, online support and its availability. However, we found that its support was still half baked due to how recently it was released. Had the RP2040 been as supported as the other Arduino microprocessors, it would have been a very good option for our project paired with the nRF24L01 radio modules.

The two real options we found were the ATmega328 and the nRF52840. The ATmega328 is a great, general purpose microprocessor, cheap and at the center of the online support for embedded projects. It would be very simple to develop for and support most or all of our peripherals. The nRF52840 has significantly less online support, but is used in an official Arduino product, which we could use to base most of the peripheral library work. We would have to learn how to use BLE, though there are still some good resources we found to help us.

Table 4 summarizes our research. We decided to select the nRF52840 for two main reasons. The ATmega328 is completely unavailable currently, and the nRF52840 is used in the Arduino Nano 33 BLE, giving us official support for our peripherals and BLE. Also, because of the Arduino support, we would be able to test the nRF24L01 against BLE to see which was better suited for our project.

Table 4: MCU comparison and selection

	MSP430FR6989	RP2040	ATmega328	nRF52840
Cost	\$12	\$1	\$1-2	\$10
Arduino IDE	No	Mostly	Yes	Mostly
Active current	~1-5mA	~9-10mA	1-10mA	~0.5-2mA
Wireless support	External radio modules only			BLE on-chip
Development board schematics	Yes	Yes	Yes	Yes
Currently available?	No	Yes	No	Yes (low availability)

3.4 Screen Technology

Our research into screen technology began with looking into the different types of displays available. The first type of display considered were LCDs or Liquid-Crystal Displays. The basic structure of LCDs make use of multiple layers of material to create an image on a display by warping light. Two of these layers are polarized filters which only allow light of certain wavelengths to pass through them. One of these filters is rotated by 90 degrees of the other which blocks out light from the first filter. Next, a layer of Liquid crystals is placed between the two polarized filters and their structure changes depending on whether there is an electrical current flowing through them or not. When there is no current flowing through them, they are oriented in such a way that light from the first filter is warped to match that of the second filter. When a current is applied, the warped light from the crystals no longer matches the second filter so the amount of light that passes through can be influenced by the amount of current flowing through the crystals. Two additional layers are a cathode and an anode that allow current to be applied to the crystals. Additional layers can be added for more advanced displays such a backlight to allow better visibility to the display and RGB subpixels to display color. Some advantages of LCDs are that they have relatively low manufacturing costs as well as many different models are available for use. Some disadvantages of LCDs is that most of them require the use of a backlight in order for the contents of the display to be seen properly. This increases the size of the display as well as increases the amount of power needed to run the LCD.

The next type of display considered was OLEDs or Organic Light Emitting Diodes. OLEDs are composed of multiple thin layers of organic material that when current flows through them, light is emitted. The basic structure of an OLED includes a cathode and anode with emissive and conductive layers in between the two. When electricity flows through, holes are created in the anode and pass through the conducting layer while electrons move through the cathode into the emissive layer. The holes and electrons meet in the emissive layer creating photons of light. The intensity of the light can be influenced by the amount of current applied to the OLED. One layer of the OLED is made transparent so that light can be emitted. Another few layers of organic molecules can be added to add color to the display. One advantage of OLEDs is that they do not require the use of backlights which helps in reducing power usage as well as size of the display. Another advantage is that they can be made up of much more flexible and lighter material, making them easier to implement into designs. One disadvantage of OLEDs is that they are much more expensive to manufacture at the current moment making them likely more expensive than other display options. Another disadvantage of them is that water can be much more damaging to them compared to other types of displays.

After researching the different display technologies available, we needed to know what function our displays would serve in the overall design and decide what factors were most important to the display we chose. For the display of the locker, it was used to display to the user their options for which locker they would like to use and configure the unlocking conditions of each one. The display needed to show multiple elements at a time so a larger display was more preferable for readability of the contents of the screen as well as a display that can be seen in different light conditions that the locker may be placed in. The locker may need to run off of battery power for a period of time so decreased power consumption was ideal for the display. With this in mind, either type of display could be used so long as it was power efficient and could fit within our price budget. For the display on the remote, it was used to display to the user the timer of the locker as well as unlocking conditions and finally the status of the lockers. Due to the ideal size of the remote, a smaller screen was more desirable as it allowed for less power loss on the remote's battery and made it more portable. With this in mind, an OLED display was better for the remote display as a LCD display would likely come with a backlight, increasing power usage and size. Some final considerations for the displays was that they needed to be compatible with our microcontroller to make it easy to configure them. Color was also not needed if these displays allowed for enough customization. Finally some displays allowed for touch screen support which was unneeded in our design.

Table 5 displays the different displays that were up for consideration for the display on the locker. When comparing the 2 displays, both had similar sizes and were suitable for use on the lockbox. Both required similar voltages to power the devices and were similar in price to each other with display 2 being slightly more expensive. Both displays utilized SPI configuration to interface them and display 2 also came with a library for Arduino that made implementation considerably easier. Given these factors, display 2 was the more optimal choice.

Table 5: Locker screen comparison and selection

	Display 1	Display 2
Price	19.85	23.99
Power	3.3 V	3.3 V
Size (Diagonal)	2.4”	2.42”
Arduino library	no	yes

Table 6 displays the different screens that were up for consideration for the remote display. When comparing the 3 displays, all three had relatively similar power consumption to each when the screen was in use so other factors were considered for selection instead. As for compatibility, displays 1 and 2 were compatible with most microcontrollers and interfacing using I2C. Display 2 also had several available libraries that were used when designing ours. Display 3 had little information on compatibility with different microcontrollers other than being able to interface using SPI. All 3 displays were at acceptable sizes for the remote but display 2 was easier to implement due to its shape. As for availability, display 1 was available in limited quantity while displays 2 and 3 were abundantly available. It was risky to choose display 1 for if we needed to order more due to damage to the one we purchased, we would not be able to immediately order more. Given the parameters highlighted, display 2 was the most optimal display to use.

Table 6: Remote screen comparison and selection

	Display 1	Display 2	Display 3
Price	5.84	8.39	5.3
Power	3.3V - 6V	3.3V - 5V	3-4.6 V
Size (diagonal)	0.96"	0.91”	1.44"

3.5 Lid Sensor

In order to allow the system to determine whether a locker is open or closed, each compartment must be equipped with a sensor that can definitively recognize the state of the compartment door. This could be done in several ways, such as detecting a magnetic field between the door and the frame, sensing the light level of the compartment, or placing a physical switch that is pressed by the door in the closed position.

The simplest method of detecting the state of a compartment door would be to mount a physical switch, such as a standard button or limit switch, on the interior of the door frame. Thus, when a compartment’s door is closed, a circuit would be completed, and the MCU would receive a signal it would interpret as a closed door. This solution, however, comes with a few clear disadvantages. Depending on the switch selected, the fact that the door will likely be closed for much of the Study buddy’s lifetime could lead to deterioration faster than expected in the mechanical components of the switch. In addition, a physical switch visible in the compartment could easily be exploited by manually pushing it in by hand, leading to the locking mechanism triggering with the door open. This method of tricking the device could be easily learned by anyone, and thus could make the device less appealing to parents that would use it for parental control of small children.

An alternative method of detecting a door’s state would be to use a photoresistor or other light-sensing component to detect the brightness of a compartment’s door frame. With this solution, the MCU would receive a signal recognizing a door as closed when there is insufficient light detected by the sensor. Using a light sensor would avoid the possible issue of mechanical failure or deterioration, but could still leave the locker susceptible to intentional exploits, such as covering the sensor to convince the system that a door is closed. This method could also cause other unintentional problems, such as a lock triggering when the device is placed in a dark environment.

The most consistent method of sensing when a door is opened or closed is with a magnetic switch, such as a reed switch or a hall effect sensor. A reed switch is a passive component that actuates when in the presence of a magnetic field. It consists of a glass tube housing two metal prongs that can form a strong parallel connection. Reed switches can be NO (normally open), meaning they do not allow current to flow through until a magnetic field is encountered, or NC (normally closed), meaning that their connection breaks when a magnetic field is applied.

A Hall effect sensor, similarly to a reed switch, can be used to detect the presence of a magnetic field and send a signal. A Hall sensor differs in that it is a powered component, consisting of a metal strip with a current applied to it. A voltage difference is created across the strip when a magnetic field is introduced perpendicular to the current, and the value of this difference can be used to determine magnetic field strength. This means that a Hall sensor has uses beyond the binary detection of a magnetic field, and thus may be more robust than is necessary for our specific use case.

Both types of magnetic sensors are susceptible to exploitation by placing a magnet near the components to send the door closed signal to the MCU, but this method is difficult to accidentally encounter and requires a magnet to perform, making them less exploitable than the other options. In addition, placing a magnet on the door that lines up with the sensor while closed is sufficient to send a correct door closed signal consistently, while a physical switch or photoresistor would require fine tuning to ensure that the door will always be detected as closed.

Table 7 compares the above lid sensing options and their particulars and downsides in our application.

Table 7: Lid Sensor Component Comparison

	Limit Switch	Light Sensor	Reed Switch	Hall Sensor
Detection Method	Physical	Light	Magnetic	Magnetic
Deterioration	High	Low	Low	Low
Exploitability	Easy	Easy	Difficult	Difficult
Est. Price (3)	~$3	~$1	~$2	~$5

Given the general pricing of these components and the binary door detecting usage in our project, we selected a reed switch as the lid sensing component of the Study Buddy. In the Study Buddy's locker, a reed switch is placed in the frame of each compartment's door, with a magnet placed on the inner side of each of the doors. Using normally open switches, the MCU receives a signal from a door when it is closed. Thus, the MCU knows that a door is open when it is no longer receiving a signal through that door's switch. This is used to determine when the locking mechanism will lock the doors, and can be used to detect tampering or forceful access to the storage compartments.

3.6 Locking Mechanism

For the Study Buddy to effectively function as a tool to keep electronics or other distracting items away from an individual, the door to each compartment must be able to lock, preventing easy access of the interior to anyone without access to the correct authentication factors. Thus, we intended to create an electronically controlled locking mechanism that completely prevents normal access to the interior of locked compartments, while still providing unobstructed access to the storage compartments when their respective doors are open.

The most practical and secure way to lock a small compartment like those in the Study Buddy is for a sliding bolt to be extended from the door itself into the frame of the locker. This will allow for the lock to be positioned such that it does not obstruct the compartment's opening, while also not requiring extra physical structures on the locker, such as a lip surrounding the edge of the compartment or significant excess space between compartments.

In order to control the locks electronically, the locker contains 3 servo motors, one attached to each door, that each rotate to slide a bolt into the wall of their respective compartment. With this setup, the locking bolts are the only components connecting the locker doors to the locker body. Thus, the physical security of the locker relies on a component explicitly designed for security, and a locked compartment is not accessible outside of methods physically destructive to the locker's structure. In addition, attaching all of the locking components to the doors themselves prevents damage to electrical components in a scenario where the door is forced open.

To actuate a bolt, the corresponding servo motor's horn is connected via a stiff wire to the bolt itself, such that the rotational motion of the motor is converted to linear movement of the bolt. The servo only needs to be able to rotate within a range of 180 degrees or less, as rotating the motor 180 degrees from the fully closed state leaves the bolt at its maximum possible open state.

Table 8 compares relevant parameters of three potential servo motors for use in the Study Buddy's locking mechanisms. Because the servo motors are mounted on the compartment doors, a smaller overall size and weight is preferred.

Table 8: Locking Servo Motor Comparison

	DFRobot SER0039	Adafruit 1142	Adafruit 2442
Voltage Rating	5V	5V	5V
Rotation Type	Positional	Positional	Continuous
Weight	0.027lb	0.138lb	0.02lb
Dimensions	0.88”x0.46”x1.04”	1.58”x0.79”x1.69”	1.3”x1.2”x0.5”
Price	$8.73	$19.95	$7.50

Because we are using the motor to flip a bolt between two positions, we needed a “positional” motor that can have particular angles selected to stop at, unlike the continuous rotation of the Adafruit 2442. Considering the above servo motors, we selected the DFRobot SER0039 as our locking motor of choice in the Study Buddy. This is because it has the smallest overall size of the options examined, and has a low weight, allowing us to place one on each door with minimal strain on the hinges.

3.7 RFID Technology

In order to develop a system that will react when specific physical objects are scanned across it, we researched the specifics of RFID technology. RFID, which stands for Radio-Frequency Identification, is a technology that allows tracking and identification of physical objects using electromagnetic fields. An RFID system is made up of two primary components: RFID tags and an RFID reader. The RFID tag is the transponder attached to the object the user wants to be tracked, while the RFID reader is the main unit that can identify tags and control what is done with their information. The RFID reader and tags can both be either active or passive, which affects the range, cost, and effectiveness of the system. The specific functions of the tags and the reader vary based on the type of RFID system being used.

An RFID reader generally consists of an antenna and either a transceiver or receiver, depending on the type of RFID system being used. An active RFID reader constantly transmits signals that can be detected by a tag, and receives the tag's identifying information, requiring a transceiver to function. On the other hand, a passive reader can only receive signals from a powered tag, meaning that a transmitter is unnecessary. An RFID reader can also be fixed, which allows higher power and range, or mobile, which gives the user more flexibility in many use cases.

An RFID tag is generally composed of a microcontroller and an antenna, both held together by a substrate. The microcontroller performs tasks such as storing identifying information, encoding/decoding data, modulating signals, and controlling communication. The tag's antenna is used to receive, transmit, and reflect signals as directed by the microcontroller. RFID tags generally start as inlays, which can be attached to an adhesive to become a label or encased in another material for protection. An active RFID tag is self-powered and only transmits signals to a reader, while a passive tag is powered by energy released by an active reader and acts as a transponder.

Considering the above reader and tag variants, there are two main types of RFID systems: PRAT and ARPT. PRAT, or Passive Reader Active Tag, consists of powered RFID tags that only transmit radio waves, and a reader which only serves to receive these transmitted signals. PRAT systems can have a high range of around 100 meters due to their self-powered tags, which also contributes to a higher overall cost. ARPT, or Active Reader Passive Tag, consists of a powered RFID reader that will power nearby passive tags, which means that the reader and tags act as both transmitters and receivers. ARPT systems use unpowered tags which are quite inexpensive, with the tradeoff of having less robust tags that must be much closer to the reader to function, usually within 10 meters.

Because we intended to use RFID technology as a verification method for a very small number of possible users who should have easy access to the locker, an ARPT system was the most appropriate and cost efficient option. With this in mind, two arduino-compatible ARPT RFID reader modules are compared in Table 9.

Table 9: RFID Reader Comparison

	RC522	Adafruit PN532
Operating Frequency	13.56MHz	13.56MHz
Communication Protocol	SPI	UART, SPI, I2C
Dimensions	1.57”x2.36”x0.4”	2.1”x4.7”x0.425”
Price	$7.99	$39.95

3.8 Keypad

Our research into keypads began with getting a better understanding of how they function. Most keypads effectively act as multiple switches that send a signal once one of the keys have been pressed. Each individual key can be tied to a pin on a microcontroller but this method is inefficient so instead most devices use a matrix configuration for the keys instead to reduce the number of pins used. One commonly used configuration involves setting either a row or column as input pins and the other set as output pins. All pins are initially set to low and each output pin is cycled to high until a high input pin is detected which means that the key tied to that combination of input and output pins is currently being pressed. The most commonly available keypads are 12 button (3x4 matrix) or 16 button (4x4 matrix) configurations.

After researching the keypad technologies available, before we could select a keypad, we needed to understand what was needed for our keypad. The keypad was to be used on the locker for the user to input a password as well as navigate through different menu prompts. A 3x4 keypad was most appropriate as the number keys could be used to navigate through the menus as well as be used to input passwords. This also freed up a few additional pins on the microcontroller for other features. Our device had enough space to accommodate most types of keypads available so space was not much of an issue. For ease of implementation on the device, a keypad made of more solid material was selected.

Table 10 displays 2 different keypad displays that were up for consideration. Keypad 2 was more expensive than keypad 1 but it is made of much more study material compared to keypad 1. Given that the price increase isn't significant, keypad 2 was the desirable choice.

Table 10: Keypad comparison

	Membrane 3x4 Matrix Keypad + extras - 3x4	Keypad - 12 Button COM-14662
Price	3.95	4.95
Material	Polyester Overlay	Conductive Rubber

3.9 Battery Technology

Most electronics and electrical systems in our world require a battery for them to run without direct power plugged into them, for our system we sure need one. Battery systems can vary greatly depending on what is required for a particular system we are looking at, from electric cars to cellphones to toys. As a result, we must analyze each option to determine which is the best fit for our project.

The two groups of batteries that are most commonly separated are main batteries and secondary batteries. When a primary battery's charge has been exhausted, it cannot be recharged or reused; it must be discarded. Devices like remote controls, kids' toys, and smoke alarms frequently use these batteries. These batteries generally have a single-use cell that undergoes an irreversible chemical reaction. As a result, we are unable to use the battery's produced power again as this chemical reaction is permanent.

Batteries that can be recharged after being used up are known as secondary batteries as opposed to primary batteries. As a result, they are appropriate for demanding use and extended applications. This would also apply to our system; therefore we would choose that battery type from the list of different battery technologies.

Throughout the secondary battery family, there are several ways to store and replenish the chemical energy found in each cell. Due to the fact that each battery has a different chemical makeup and use case, it is essential to comprehend why each battery would be advantageous for our system or why they would not. Among the batteries we are considering were some famous technologies like the lithium ion and nickel metal hydride, and infamous new ones like zinc ion battery technology.

3.9.1 Lithium-ion (Li-ion) Battery Technology

Lithium-ion battery technology is developing swiftly. Different battery types have certain common advantages and disadvantages. We carefully analyzed all the ways a lithium battery will improve the situation so that we could make the best choice. system overall. The primary characteristic that sets lithium batteries apart is their high energy density. The battery needs to be able to provide enough power for a constant operation of our system for the requested duration by the user. A battery that is a longer-lasting energy source is the best kind of power source.**(1)**

Low maintenance and self-discharging are the next two traits. The battery may be used with little concern about performance degradation. This more closely relates to the client. As many individuals as possible will desire a system that will function without their involvement in its upkeep. The battery discharge rate is a crucial factor because we are utilizing rechargeable batteries. Compared to other batteries on the market, lithium batteries drain at a lower rate 5% in the first four hours and a constant 1-2% every month.
The battery's cell voltage was the final important factor we examined. Because of this, we can acquire a more potent battery without giving up any space. Due to

the 3.6-volt cell voltage of lithium batteries, less cells are needed to supply the necessary voltage. For example, in Table 11: the NCR18650B 3400mAh Li-ion battery has a nominal voltage of 3.6V, which is what we are looking for. The voltage of a typical battery's cell (non-lithium) ranges from 0.8 to 2 volts, which increases the battery pack size significantly. Higher cell voltage also makes system power control easier, which is another benefit. The load characteristics, lack of priming, and widespread availability of a lower priority are further benefits.**(1)**

Table 11: NCR18650B 3400mAh Li-ion battery specifications

Rated Capacity (at 20°c)		**Min.3200mAh**
Nominal Capacity (at 25°c)		**Min.3250mAh**
		Typ.3350mAh
Nominal Voltage		**3.6V**
Charging Method		**Constant Current** **Constant Voltage**
Charging Voltage		**4.2V**
Charging Current		**Std.1625mA**
Charging Time		**4.0hrs.**
Ambient Temperature	**Charge**	**+10~+45°c**
	Discharge	**-20~+60°c**
	Storage	**-20~+50°c**
Weight (Max.)		**47.5g**
Dimensions (Max.) **Maximum size without tube**	**(D)**	**18.25mm**
	(H)	**65.10mm**
Volumetric Energy Density		**676Wh/l**
Gravimetric Energy Density		**243Wh/kg**

*A NCR18650B 3400mAh Li-ion battery specifications table that shows its characteristics table**(22)***

A lithium battery has several drawbacks in addition to all of its advantages. One that has to be highlighted is the necessity of protecting lithium batteries. Lithium batteries need to be protected from charging and draining because of their tiny size. For these kinds of batteries, over current is a serious issue that may be destructive. Many lithium batteries require a battery management system to be

installed in order to maintain them functioning correctly. As a result, the cells may be charged properly without being overcharged. Excessive voltage might harm the cell if no management system is in place.
During the charging process, while each cell is storing more power, the battery's temperature rises, so operating temperatures shouldn't go over one to two degrees Celsius.**(1)**

The lithium battery's aging is yet another drawback. Regardless of whether a lithium battery is being used or not, aging can still occur. Additionally, each time the battery is charged, aging will advance further. Batteries can frequently tolerate several charging cycles. The typical range for a lithium-ion battery's charge cycles is 500 to 1000. We might create a regular timetable to replace the battery based on the number of times it has been charged. However, the system's price will rise. Incorporating the cost of lithium batteries -which is around 40% higher than popular ones- would increases the cost above a typical value, which is a significant drawback. **(1)**

3.9.2 Nickel Metal Hydride (Ni-MH) Battery Technology

A typical usage of Nickel metal hydride batteries is high current battery cells. They therefore function well in gadgets such as laptops, digital cameras, mobile phones, portable gaming systems, and other compact but power-hungry ones. This implementation cannot be supported by a battery like a lithium-ion battery because the low internal resistance of the nickel metal hydride battery is necessary to make it feasible. These batteries can therefore supply a lot of energy to devices that need it, but they also have some drawbacks.**(2)**

When considering more than just the battery implementation in this project, we run into a few significant issues. Although there are nickel metal hydride battery full battery banks that could be used in our power system, these batteries are not suitable for extended use. This is brought on by the nickel metal hydride battery's shortcomings in a few significant areas when compared to those of its competitors. Starting, the nickel metal hydride battery rapidly drops to a very low voltage and has a very high rate of self-discharge. All battery cells have this issue, but the nickel metal hydride battery in particular does.**(2)**

This makes it challenging for this type of battery to compete in the modern world, along with their relatively long charging times compared to their competitors. Nickel metal hydride batteries have a complex charging algorithm, as it is also seen when examining this kind of battery. Because of this, creating a system where the battery and the energy to ensure the most efficient charging, a system monitors the conversion process. In addition, this battery system has the capacity to use a lot of heat, a lot more heat than other battery types. These issues collectively make this type of battery useless in our power system. **(2)**

3.9.3 Zinc-ion Battery

The zinc-ion battery is an innovation in battery technology that is relatively new. It is a relatively new product that fills the market gap between lead acid and lithium-ion batteries. Although the zinc-ion battery combines advantages of both lithium-ion and lead acid batteries, there are some significant roadblocks in its path to becoming the undisputed market leader. Zinc-ion batteries have the benefit of withstanding difficult operating conditions for extended periods of time, despite the fact that they are related to lithium-ion batteries.**(3),(4)**

The zinc-ion battery has a water-based electrolyte that prevents it from igniting, which is another benefit of the battery's chemical construction. A zinc-ion battery system is considerably safer than a lithium-ion battery due to lithium's propensity to react violently with other substances and elements, primarily as a result of the use of a flammable electrolyte. Additional parts of the zinc-ion battery include manganese and zinc, both of which are much easier to obtain, more affordable, and mineable in the United States. This battery uses both of the materials, in contrast to the lithium-ion battery, which is dependent on more expensive and hard-to-find lithium cobalt and nickel.

Zinc-ion batteries have a long-life span, and it is an important benefit. Around 15-20 years is the anticipated lifespan of these battery systems, which is longer than that of lithium-ion and lead-acid batteries. As a result, they require fewer replacements and can operate in systems for longer stretches of time, which is best for the client.**(3),(4)**

Given this, the disadvantages of the zinc-ion battery are still on par with those of other battery systems. The discharge capacity and energy density of the zinc-ion battery are lower than those of a lithium-ion battery because it is unable to compete with other batteries in this market. If the battery is stationary, the zinc-ion battery's heavier weight in comparison to the lithium-ion and lead-acid batteries is not a disadvantage.**(3),(4)**

The zinc-ion battery currently costs more than its competitors due to its recent advancement in the power sector. It will take a while for its price to stabilize at a more affordable level. As with any new technology, over the course of the next ten years, we can expect to see a significant decrease in this price. We will observe that this type of battery becomes considerably less expensive and more frequently used in systems as a result of its lower bill of materials as the cost of manufacturing these new zinc-ion batteries inexorably declines. **(3),(4)**

3.9.4 Comparison and Part Selection

Main system battery:

The main body battery for our system needs to be powerful enough to feed the system with all of its sensors, the MCU, and the lid locking motor for more than two weeks. Since the main locker box is quite big, space is not a big problem for choosing this battery, which leaves us more space to get a powerful enough battery. The shape of the battery is not important right now, so we only focused on the performance and cost of each battery technology.

For this project, we chose that our system is going to use the Lithium-ion battery technology. This is an energy dense battery that can produce maximum needed 2 amps for more than 3 Ah. One main factor in choosing this battery is having more charging and circuit protection resources available. For the needed application and its provided space that we are building the system in, a Lithium-ion battery pack is going to be more appropriate than a Ni-MH battery, since it would not take a huge amount of space and it has less cell count. Due to the lack of enough information about the zinc-ion battery and the rarity of sellers on the market and the internet, we did not consider the zinc-ion battery as one of our valid choices. Zinc-ion batteries are fairly a new technology, which means it needs more time to mature and become a valid choice or even a competitor on the market for consumers.

The following Table 12 shows a summary of the battery technology information.

Table 12: Main system battery technology comparison

	Lithium-ion	Nickel Metal Hydride	Zinc-ion
Capacity (e.g.)	(700-15000) mAh	(400-11000) mAh	>700 mAh
Life cycles	500-1000	1000	10000
Voltage	3.6 volts	1.2 volts	(0.75-1.8) volts
Max discharge current	~(5-35) A	~ (0.4-11) A	N/A
Cost/pack (average)	20$	13$	N/A

(5)

Remote controller battery:

As the remote controller does not require much power, and replacing the batteries is much easier than the main system, which is implemented in the design of the remote controller, choosing a suitable battery for the remote controller is easier.

Considering the size of the remote controller that is needed to be small enough to hold easily by one hand, batteries should be relatively small too. Thus, we are considering triple A batteries, or (AA) batteries, or coin cell batteries.

For this project, we did use the AA battery. This is a dense energy battery comparing it to a coin cell one, and since we are having a small display with an MCU and wireless technology on the remote controller, we needed a long lasting battery that we can rely on. Double A batteries can produce 0.1 amps for 25 hours, which would give the remote controller much longer lifespan. Also, double-A batteries do not weigh much, so it will be suitable for a portable small remote controller.

3.10 Voltage Regulators

The voltage regulator is a tool designed to maintain the voltage automatically. The system typically absorbs a higher input voltage and releases a more steadily streamed output voltage to assist in voltage regulation during significant changes in power generation and load fluctuation. In this case, through an active voltage regulator both our battery and the converted wall AC power will be used to supply a nice DC voltage to our system. The two main types of voltage regulators—switching regulators and linear regulators—can be used with voltages ranging from one to thirty volts. To realistically determine which is needed, we must compare them because each has advantages and disadvantages of their own.**(6)**

3.10.1 Linear Voltage Regulating

A transistor-based integrated circuit primarily made up of differential amplifiers, which we call linear voltage regulators. The input voltage must be at least greater than the output voltage for any linear voltage regulator to function properly. Two distinct classes of linear voltage regulators are shunt, and series regulators, may be considered as candidates for a suitable voltage regulator for our system.**(6)**

Shunt regulators create a path from the supply voltage to ground with the help of a simple variable resistor. Due to the fact that the current is switched away from the load and flows to ground, this kind of linear regulator is the least efficient. This type of linear regulator is less expensive because of its simpler design. These regulators are most frequently used in circuits with very low power because they can only absorb current and because the current lost in these circumstances is so small and negligible.**(7)(8)(9)**

The series regulator is still another choice. Because it outperforms shunt regulators in terms of output and design, this kind of linear voltage regulator is more widespread. To create a pathway from the supply voltage to the load in this situation, a transistor is used instead of a resistor. The device is therefore capable of operating more efficiently and with less effort when compared to their shunt counterpart. The low dropout voltage regulator is the best illustration of this implementation because it keeps proper voltage regulation even as input and output voltages approach one another.**(7)(8)(9)**

3.10.2 Switching Voltage Regulating

Alternate voltage regulators involve switching regulators. These allow us to change the duty cycle of the output voltage because they have a higher switching frequency. Energy is transferred through these devices using power switches, inductors, and diodes. An IC switching controller activates and deactivates the power switch. This increases our efficiency significantly and makes it possible for it to maintain a constant output voltage. We can do this while generating a lot less heat, which allows us to better control our regulator and make heat management simpler. However, this will result in switching regulators being more expensive than their linear counterparts. Additionally, they are more complex, requiring more external parts and greater caution when designing around them.**(7)(8)(9)**

By utilizing this kind of regulator, we can gain access to more complicated configurations, such as the widely used boost and buck converters in power systems. Switching regulator-based topologies called boost converters are able to increase an input voltage by a certain amount. However, a buck converter is used to do the opposite. Even better isolation from the input can be provided by these converters by transferring through a transformer. There is a problem with switching regulators because these types of regulators use an on and off switch, they add noise to the output, and a linear regulator would not have this issue presented at the output.**(7)(8)(9)**

3.10.3 Comparing Technologies and Part Selection

Examining the two distinct voltage regulator types allows us to see the benefits and drawbacks of each type. Linear voltage regulators have a more straightforward design, cost less money, respond to voltage changes quickly, and don't produce switching noise. However, the switching regulator comes with a higher price tag and some noise produced at its output in exchange for the benefits of greater efficiency and better performance.

Because our system would have power sources which are the wall AC signal and the internal battery which supplies a DC output, we needed a high efficiency type of voltage regulator. As we are planning to have an extended time of operation without external power from the wall to increase the portability of the system, we would need higher efficiency switching voltage regulators for the battery output power. Also, power coming from the wall is continuous and we need to consider a high efficiency switching voltage regulator to power the system, charge the battery, and provide power for the charging USB A ports. Linear voltage regulators are not very efficient when converting high voltages to small voltages, so they are not considered in our design.

Considering all the different variables to our system design, we chose switching voltage regulators for AC to DC power coming from the wall, which will be taking a DC 24 Volts at its input and providing a constant nominal 13 Volts DC at its output -like shown in Table 13 of TPS5420-Q1 IC family recommended operating conditions- to the different components in the system, and the battery charging IC circuit. Also, we needed another voltage regulator IC that is capable of operating at high efficiency but with low current output for the control electronics. For that, we chose LMR33610 for that purpose.

Table 13: (TPS5420-Q1) Voltage regulator specifications example

Absolute Maximum Ratings				
Parameter		**Min**	**Max**	**Unit**
Input voltage range	VIN	-0.3	40	V
	BOOT	-0.3	50	
	PH (steady-state)	-0.6	40	
	EN	-0.3	7	
	VSENSE	-0.3	3	
	BOOT-PH	10 V		
	PH (transient < 10ns)	-1.2 V		
Io Source current	PH	Internally limited		
I_{lkg} Leakage current	PH	10 μA		
T_J	Junction Temperature	-40	150	℃
T_{stg}	Storage Temperature	-65	150	℃
Recommended Operating Conditions				
Parameter		**Min**	**Max**	**Unit**
VI input voltage range		5.5	36	V
T_J Operating junction temperature		-40	125	℃

Characteristics and operating specification of commonly used voltage regulator (TPS5420-Q1) based on maximum and recommended ratings. ***(26)***

If the AC wall power was not available (not plugged in), then the system will need to get its power from the battery, and considering the desired long period of operation we want to achieve, we used high efficiency Switching voltage regulators to provide nice constant DC power to the different components, but not to the USB A charging ports, since they can drain the battery very fast and then the system will shut down.

3.11 Battery Management System

A piece of technology known as a battery management system, abbreviated as BMS, was developed for the sole purpose of monitoring the status of a battery pack. A battery pack is a collection of battery cells that have been electrically arranged in a row by column layout that can power a variety of electronic devices. Because of the configuration of this system, it is possible to deliver a specified range of voltage and current for an extended period of time under conditions of anticipated load.**(16)**

The term "battery" in this context refers to the entire pack, However, the overall battery pack assembly's monitoring and control features only apply to targeted cells or groups of cells known as modules. These monitoring and control features are contained within the overall battery pack assembly. These modules are an integral part of the overall battery pack assembly that is being constructed. Since lithium-ion rechargeable cells have the highest energy density of any other type, they are almost always used in battery packs for a wide variety of consumer goods, including laptops and electric vehicles. This is due to the fact that lithium-ion rechargeable cells can be used for longer periods and multiple times before needing to be replaced. When used outside of a generally small safe operating area, they can be rather unforgiving, and the results can range from compromising the performance of the battery to being extremely dangerous. In general, a small safe operating area is recommended. The job description for the BMS is undoubtedly difficult, and its overall complexity and oversight reach may extend across a wide variety of fields, including the digital, electrical, thermal, control, and hydraulic fields, among others.**(17)**

The BMS can also be used to manage the amount of power extracted from each cell, which enables the user to increase the battery's performance or extend its lifespan. In order for the system that manages the battery to function properly, it was necessary to configure a wide variety of different settings. These requirements include the following:

- The feature of protection against over discharge of a battery ensures that it will not be discharged to a level that is below a certain threshold as being risk-free.
- The protection against short circuits prevents the battery from being harmed by hasty electrical connections made between cells or between an electrode and the ground.
- The thermal protection keeps the battery from getting too hot and failing, and the thermal runaway protection will kick in and turn the battery off if the temperature of a cell rises to an unsafe level.

- "Cell balancing" refers to the process of checking that each individual cell in the battery pack has been charged to the same level. Because of this, there is a decreased chance of an uneven discharge as well as cell damage.
- Protection against the discharge or charge of current, which shields the battery from dangerously high currents flowing through it during the process of charging or discharging.
- Overcharging protection helps protect the battery from damage by preventing it from receiving an excessively high charge, which would otherwise cause the cells to become damaged or even destroyed. In the Table 14 below of a BMS is an example of an overcharge protection by cutting off power on a cell that is charged to a maximum voltage of 3.6V .In the absence of this protection, the battery could be harmed.**(17)**

Table 14: Default BMS parameter settings

parameter	value	unit
balance start voltage	3.5	V
balance end voltage	3.6	V
maximum diverted current per cell	up to 1.3 (3.9 Ohm)	A
cell over voltage switch-off	3.8	V
cell over voltage switch-off hysteresis per cell	0.015	V
charger end of charge switch-off pack	3.6	V
charger end of charge switch-off hysteresis	0.15	V
cell under voltage protection switch-off	2.2	V
cell under voltage protection alarm	2.6	V
under voltage protection switch-off hysteresis per cell	0.03	V
cell under voltage protection switch-off timer	4	s
cells max difference	0.2	V
BMS maximum pack voltage	**62.5**	**V**
BMS over temperature switch-off	50	°C
BMS over temperature switch-off hysteresis	5	°C
cell over temperature switch-off	60	°C
under temperature charging disable	-15	°C
max DC current relay @ 60 V DC	0.7	A
max AC current relay @ 230 V AC	2	A
BMS unit stand-by power supply	< 90	mW
max DC current @ optocoupler	15	mA
max DC voltage@ optocoupler	62.5	V
BMS unit disable power supply	< 1	mW
Slave unit cell balance fuse rating (SMD)	2	A
internal relay fuse (Master unit)	2 slow	A
dimensions (w × l × h)	190 x 114 x 39	mm
weight	0.650	kg
IP protection	IP32	

Lithium battery BMS showing default parameters of charging multiple cells in a pack
Copyright permission received from rec-bms; Appendix A-1

Each of these BMS features is necessary for safeguarding the battery and making certain that it will retain its capacity to perform its intended functions over time. It is possible that on occasion we will need to make some adjustments to the BMS's settings in order to adapt it to the requirements of the application that we desire. For example, if the lithium battery is going to be used in a device with a high-power output, the battery management system may need to increase the current protection setting in order to prevent damage to the cell. This is done to ensure that no cell damage of the battery is going to occur.

In addition, it is important to remember that these configurations are flexible and can be altered as necessary to accommodate the particulars of the use case. Keeping this in mind helped to ensure that we get the most out of our system. For example, if the battery is going to be used in a cold environment, the BMS may need to reduce the setting for the discharge protection in order to prevent the battery from turning off prematurely. When choosing a battery for a particular use, it is necessary to give thoughtful thought to the BMS protection features, given that these are an essential component of any lithium battery system. By developing a thorough understanding of these settings and how they work, we can make sure that our battery is operating in a way that is both safe and reliable.

For our project, we are going to use a BMS ready from the market, but it can be modified. For our battery selection, we mostly would have 3 Lithium-ion cells, 3 on series (3S) configuration. This configuration would determine the type of BMS that we did searching for for our final design.

We decided to choose from three different products of small pcb BMSs. those are ACEIRMC (HW -375), Daier (HX-3S-FL25A), and Atnsinc (4S 40A). All products can be found on Amazon and have relatively good prices. Finally, we chose Daier (HX-3S-FL25A) because it has most of the features that we considered, and can manage 3 cells in series which is enough for our design.

3.12 Power supply (AC-DC)

Direct current (DC) supplies that operate at low voltage are frequently used to provide power to electronic products and applications. The power supply of any product that is intended to function when powered by the electricity provided by the mains must make use of an AC-DC power supply in order to convert the alternating current (AC) provided by the mains to a direct current (DC). Regardless of whether it requires a DC-DC or an AC-DC power supply, there are a few requirements that are absolutely necessary to keep in mind. To begin, the DC output voltage needs to be tightly controlled within the allowed tolerances across all load conditions in order for our application to function reliably. This is true regardless of whether or not it is being loaded. This concept is referred to as line regulation. Additionally, there must not be any damaging transient voltage ripple, noise, or spikes in the output voltage because these ripples or spikes have the potential to hinder the performance of the final product.

The electrical noise may have come from somewhere else or may have been introduced by the power supply. As a byproduct of the power conversion procedure, ripple voltages, which are extremely small fluctuations, could affect the output voltage. Transient spikes may result in instantaneous voltages that are significantly higher than the output voltage when the load condition changes rapidly over a brief period of time (dV/dt). The power conversion circuitry, which is a crucial part of equipment powered by the mains, provides isolation between the input and output voltages. In the vast majority of power supplies, switched mode and linear topologies are employed. The topology that is employed most frequently is the switched mode, which includes non-isolated buck and boost in addition to isolated fly-back conversion.

As the law of nature, the presence of energy losses throughout any process of power conversion is a must, which results in a reduction in the overall efficiency of the process. When it is being used to power full load, a typical AC-DC power supply, for instance, might have an efficiency rating that falls somewhere between 80 and 95 percent. During the conversion process, which is typically triggered by switching inductors and transistors, energy is lost. This loss of energy is the cost of trying to adapt to a load. This energy loss results in the production of waste heat, which must be removed from the generator of the power. The amount of energy that an AC-DC power supply consumes while it is in standby mode or when there is no load connected to it is one of the aspects of energy efficiency that are governed by international regulations.

AC-DC power conversion topologies

Linear topology is only used in a select few applications at this point in time, as it is almost never employed in modern computing. One of its most significant flaws is that it must rely on a bulky mains transformer in order to be shielded from the mains voltage and have that voltage lowered to a level where it can be used. Having said that, linear power supplies might find a place in high-quality audio applications due to the low noise characteristics that make them a viable alternative. Switched-mode topology, in any one of its many forms, is used to build the vast majority of AC-DC power supplies that are manufactured today. As opposed to a linear topology, switched-mode topologies have the advantage of utilizing a semiconductor switching circuit that operates in a frequency range, typically 20 kHz to 150 kHz, in which the physical size of the transformer can be significantly reduced. This is in contrast to a linear topology, which uses a circuit that operates depending on a transformer that is significantly bulkier . Because of the switching methods that switched-mode power supplies use, they consume less energy converging AC to DC on average than linear ones. Careful attention is required in order to avoid the formation of conducted and radiated electromagnetic interference (EMI) that are a result of the switching process. Other potential issues include noise and ripple.

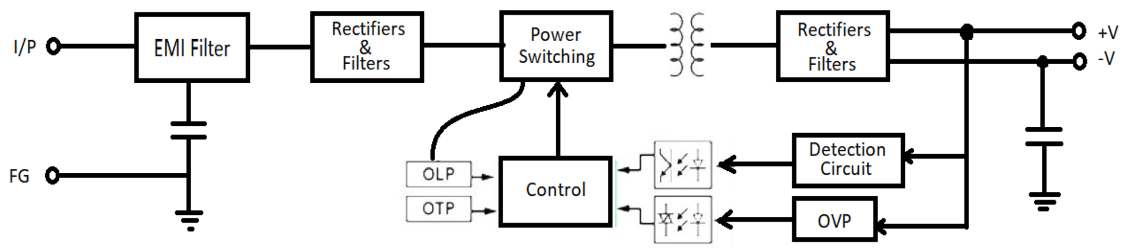

Figure 8: Block diagram of a typical isolated fly-back converter AC-DC power supply

One type of configuration that is found in AC-DC power supplies is the isolated fly-back converter (see Figure 8, above). For example, these power supplies can have an output of tens to hundreds of watts. Utilizing a compact transformer not only keeps costs to a minimum, but it also provides the input-to-output isolation mandated by laws governing product safety. The AC input is rectified and then filtered before moving on to the switching stage. The frequency and duty cycle of the driver signal which is applied during the power stage are typically determined by the load conditions. This signal typically drives MOSFETs on and off. Due to a control voltage being fed back into the driver circuit, the DC output can be controlled in this way. The primary control circuit and a DC output are electrically separated from one another by an opto-isolator. The forward converter is a further popular topology for AC-DC supplies, which DC-DC converters also make use of.

The switching losses directly affect the power supply's overall energy use efficiency, as was just mentioned. The overall system efficiency is greatly impacted by the use of zero voltage switching (ZVS), which makes sure that the switching occurs when there is no current flowing. The half-bridge converter topology is one that is frequently used in this context. Since there is less waste heat produced as a result of higher efficiencies, power supplies can be made smaller while still sustaining a greater power density. The classification of different AC-DC power supplies as Class I or Class II depends on the demands of the task at hand. The protective earth is a standard safety feature of Class I power supplies while it is not present in Class II power supplies. Wall plugs with low power ratings frequently use class II supplies.

For our locker system application, we used the switching-mode power supply, since it provides the small footprint for the system’s final PCB design. Additionally, we mostly would need a lot of power, especially while trying to implement phone charging ports that would certainly need a considerable amount of power. Trying to implement a high power rated linear power supply is not practical for our case, as the transformer would cost us much more than implementing a switching-mode power supply, not to mention the large footprint that it will take.

We decided to buy a ready to use power supply, since designing and building one is not practical for our project. Designing a flyback switching power supply needs some custom parts, which usually are hard to get and the possibility of the need to redesign and getting new custom parts is not an option, so we considered some online shopping websites to buy from and amazon was the primary market, but we also included another power supply that we already own. The following table 15 shows the considered power supplies and what we choose.

Table 15: Table comparison of power supplies

Power supply	ALITOVE	SHNITPWR	inShareplus
Maximum load (watts)	96 watts	120 watts	150 watts
Regulating technology	switching	switching	switching
Universal compatibility	(100-240) volts Yes	(100-240) volts Yes	(100-240) volts Yes

3.13 Construction material

Our research into material to be used to construct the device began into researching potential material that the locker could be built out of. Our first idea was to build the entire locker out of a wooden material. Since wood is an insulator, users should be properly protected from the internal circuitry of the device if we can keep it entirely within. We figured we may need to run wires into the storage units so we may need to properly cover it up to avoid potential damage or contact. As for the wood itself, we needed to find a type of wood that could handle the potential temperatures that parts of the device may reach as well as prevent moisture from getting in. After researching into the heating points of various woods and their ability to potentially have liquids seep through them, we believed that plywood would be more than effective for the construction of the locker. It can withstand the expected temperatures of the various parts of the device while staying affordable. As for the construction of the lock box itself, we measured out appropriate sized pieces of plywood and made use of a combination of glues and screws to build the frame of the box itself. Then we used additional wood to make each compartment within the overall frame.

Another potential material we considered was plastic using a 3d printer. We needed to find a 3d modeling software that we could make our design in. If we selected this option, several members of our group were sufficiently knowledgeable of various 3d modeling software to create the design. As for the 3d printer itself, we could have used the printer available in the UCF manufacturing lab. There were potential problems with this choice such as due to the size of the locker, 3d printing it would be both expensive and a lengthy process.

Another material considered was a metallic material. We researched potential methods into how we could make the locker out of metal. One potential design was to obtain sheets of metals and laser cut them down to the proper dimensions of the lock and box and weld them together. One problem with this design is that the metals available would be more expensive to purchase than other available options. Another problem was that none of the members of the group were sufficiently knowledgeable enough on how to weld.

With the 3 construction materials in consideration, we believed that the best material to build the locker out of was out of plywood. It would be easy to work with and relatively cheap compared to the other objects. Since our project is just a prototype, if we were to build it for commercial use, then we would likely use metal instead as it would be the most sturdy material.

3.14 Fingerprint sensors for advanced goal

Our research into fingerprint sponsor technology began into betting a better understanding on how they function. The sensors make use of a scanner to obtain an image of the fingerprint and scan it for unique patterns of where ridges meet or split. It records the exact position and angle between these ridges and makes use of an algorithm to create a unique code for the finger. There are 3 basic types of scanners used to create these images which are optical, capacitive and ultrasonic (of the 3, optical and capacitive sensors are the most commonly used and easiest ones to implement). The first type of scanner researched were optical scanners which make use of image sensors to take high-contrast images. The sensors are made of many LEDs that shine light on the finger and the sensors capture the reflected light using diodes to create the image. An algorithm is then used to see if the image is too bright or too dark to be used and the light produced from the LEDs is adjusted until a usable image is captured.

The advantages of optical sensors are that they are relatively cheap to produce and use a relatively low amount of power compared to other types of sensors. They are also easy to implement in designs using microcontrollers as many models are available that are compatible with popular embedded platforms. The disadvantage of the sensors is that they have poor security as they only need an image of the fingerprint to fool the scanner during the checking process. Some scanners come with an additional scanner that checks for biometric details but this increases the price and power consumption considerably.
The next type of scanner researched were capacitive scanners which make use of many small capacitors to create the image of the fingerprint. The capacitance

between the ridges and valleys of the finger is measured. Ridges generate a low amount of capacitance as the distance is small between the plate of a capacitor and the ridge. Valleys generate a larger capacitance due to the increase in distance as well as air between. The capacitance is passed through an op-amp circuit and analog-to-digital converter to create a digital scan of the fingerprint.

The advantage of capacitive sensors is that they have better security as they scan the shape of the object so a simple photo cannot fool the scanner. They can still be fooled by molds of a finger so some come with biometric scans to improve security at the price of increasing costs. They are typically slightly higher in price compared to optical scanners with a similar power usage depending on model. They also scan much more quickly than optical scanners.

The final basic type of scanner researched were ultrasonic scanners which use ultrasound to map a 3d image of the fingerprint. They make use of ultrasonic transmitters to produce ultrasonic waves that bounce off of the ridges and valleys of the finger and ultrasonic receivers to create the fingerprint scan. Ultrasonic sensors have the advantage of being the most secured of the scanners as the scan is too precise to be faked. The disadvantages to them are that they are slower to scan compared to other scanners and are generally more expensive and high in power consumption.

After researching the different fingerprint sensor technologies available, we figured out which type of sensor would be ideal for our project. The fingerprint sensor in our design was added as another security measure used to lock the device and can be used alongside other measures. The highest level of security was not needed as the design had other measures in place so we could go for less expensive options. This means we only needed to look at optical and capacitive sensors as the added security of ultrasonic sensors was not needed and would incur unneeded costs. When deciding between the remaining two options, both were compatible in price with optical being slightly less expensive. The key factors when deciding a sensor should therefore be power consumption and ease of implementation. The ideal sensor should use little power while in an idle state and run efficiently enough to not draw much when used. For ease of implementation, the sensors needed to be compatible with the microcontrollers used in our design.

Table 16 displays the different fingerprint sensors that were up for consideration for our design needs. When comparing the two sensors, the capacitive sensor was generally more power efficient compared to the optical sensor as well as having a slightly faster scan time. Both sensors were capable of storing over a 100 fingerprints which was more than enough for the expected number of users using the devices at any moment. The capacitive sensor's design also made it easier to implement as more resources were available online on how to make it function with our chosen microcontroller. With these facts in mind, the capacitive fingerprint sensor was more ideal to implement in our design even with the higher cost.

Table 16: Fingerprint sensor comparison and selection

	Fingerprint Reader Sensor Module DY50 FPM10A Optical Fingerprint Module For Arduino Locks Serial Communication Interface Tools	Capacitive Fingerprint Scanner/Sensor for Arduino
Price	5.36	8.03
Power	Supply voltage: DC 3.6 ~ 6.0V / 3.3V Supplying Supply Current: Current: <120mA Peak current: <140mA	Operating Voltage: 3.3V Operating Current: less than 60mA
Size	56 x 20 x 21.5mm	diameter 21mm/height 5mm
Scan speed	<1.0 seconds	300~400ms
Type	Optical	Capacitive

3.15 Phone charging for advanced goal

When talking about USB networks, one host and one device are going to be mentioned at all times. The majority of the time, the computer will be referred to as the "host," while the term "device" will be used to refer to your appliance. While power is being transferred from the host to the device, there is no restriction placed on the flow of data. Both the USB socket and the USB cable have a total of four wires and pins on them for connection purposes. In addition to the pins on the inside, which are responsible for carrying data, there is another set of pins located on the outside that supply a power supply of 5 volts.**(14)**

The host controllers found in hubs and computers are equipped with interfaces that are referred to as "A-style connectors" and have a rectangular and flat form. Because this interface makes use of friction to maintain the connection, users can connect to it and disconnect from it with a minimum of fuss. The connector is not using round pins, but flat contacts because the flat contacts are better able to withstand the repeated act of connecting and disconnecting that is required of them. Only host controllers and hubs should make use of the "downstream" connection that is provided by the A-socket connector. This connector can never be utilized for anything other than the intended purpose. On a peripheral device, the "upstream" connector position was never meant to be occupied by a cable or plug of any kind. Because a host controller or hub is supposed to supply 5V DC power on one of the USB pins, this is a very important feature to have. A-A cables are utilized for the purpose of connecting USB devices that have an A-style Female port to a personal computer (PC) or to another USB device, as

well as for the purpose of transferring data between two distinct computer systems.
If both the device and the charging port support the USB 2.0 standard, which is currently the lowest common denominator for entry-level Android smartphones, you will be able to charge it at a rate of 1.5 amps and 5 volts. Some consumer electronics that are capable of charging at a rate of 2A. If the device is compatible with one of these batteries' USB 3.0/3.1 charge ports, then it will be able to receive power at 3.0A/5V from that port. When charging an accessory that does not support either of the fast-charging specifications for USB-A, such as an inexpensive pair of wireless earbuds or another Bluetooth device, the accessory will charge slowly at either 500mA or 900mA, which is approximately the same as what you can expect when directly connecting it to the majority of PCs.**(15)**

<u>Qualcomm Quick Charge:</u>
The Qualcomm system-on-a-chip (SoC) technology, which can be found in a variety of well-known mobile devices like tablets and smartphones. The standard for its rapid charging, which is referred to as Quick Charge, has been revised a number of times. Because it is backwards compatible with earlier versions of the Quick Charge protocol, the most recent version, Quick Charge 4.0 is compatible with all previous versions of the Quick Charge protocol and can therefore be used with any Quick Charge accessory or device. In contrast to the USB PD connector, which is incapable of delivering Quick Charge 2.0 or 3.0 which utilizes USB-A as the table 17 below shows. USB-C is the only port that has the capability to use Quick Charge 4.0 and only smartphones equipped with the Qualcomm Snapdragon 8xx chipset can support Quick Charge 4.0. This chipset can be found in a sizable number of Android devices made by Samsung, LG, Motorola, OnePlus, ZTE, and Google that are sold in the North American market. Quick Charge 4.0 is only available on smartphones.**(15)**

Table 17: Different charging modes with their performance and supported connectors

Mode	Voltage	Max Current	Connector
QC 4.0 (USB-PD Compatible)	Variable up to 20V	4.6A	USB-C
QC 3.0	Variable up to 20V	4.6A	USB-A/USB-C
QC 2.0	5V, 9V, 12V, 20V	2A	USB-A
USB FC 1.2	5V	1.5A	USB-A
USB 3.1	5V	900mA	USB-A
USB 2.0	5V	500mA	USB-A

*different charging modes with their performance and supported connectors **(15)***

Quick Charge 3.0 and Quick Charge 4.0 are variable voltage charging technologies that, like USB Power Delivery, intelligently increase the amount of power that is sent to your device in order to ensure the quickest possible charging speeds without compromising security. Both of these technologies were developed by Qualcomm. But Quick Charge 3.0 and 4.0 are not the same as USB PD because they have some additional features for thermal management and voltage stepping that work with the most recent Qualcomm Snapdragon SoCs. These features allow Quick Charge to work with more recent Qualcomm SoCs. These features are designed to achieve the greatest possible reduction in heat footprint while the device is being charged. In addition, rather than using USB PD's protocol for the selection and negotiation of variable voltage, it uses a different protocol, which Qualcomm asserts is superior and provides a higher level of security for its own SoCs. Quick Charge 4.0 is approximately 25 percent quicker than Quick Charge 3.0 for mobile devices that use the most recent chipsets from Qualcomm. The manufacturer claims that the device can be fully charged in just five minutes and that it has an operating time of up to five hours.**(15)**

For our system, we used Quick Charge (OC 2.0/OC3.0) technology to offer a charging port at each individual locker. The ports connector type is a USB-A female plug that can withstand DC power at maximum 24 volts / 1 Amps each. For this goal, we did not design it fully, but partially and bought the rest of the needed components.

3.16 Emergency unlock + phone call detection for stretch goal

For our stretch goal, we planned to implement an emergency unlock feature to prevent users from having their phone stuck when they need to make or receive an emergency call. This would be an important feature if our device were to be installed in an environment such as a movie theater or class room. Since our device is primarily a tool to help with self discipline, in such an environment we needed to consider situations when the user would need immediate access to their phone and open the locker.

We came up with a few ways of detecting and triggering an emergency unlock based on different scenarios that would be related to such a situation. One idea was to detect the vibrations of a phone ringing, where the emergency situation occurs when the device detects two or three successive phone calls. Another idea was to detect the light of a strobing smartphone flashlight, where the phone tries to signal to the device that it needs to unlock the compartment it's in. The last idea we came up with was to put a feature into the remotes menu that can trigger an early unlock of the compartment after the user confirms the unlock by holding down a button.

3.16.1 Detecting repeated phones calls

In order to identify an emergency situation using repeated phone calls, we would need a way to detect a phone receiving a call. Since phones vibrate when this happens, we could figure out how to detect the vibrations from a phone call. We could do this by placing some sort of vibration sensor under the phone compartment.

There were some pretty simple vibrations sensors available in convenient modules that we could use for this. We had to find one sensitive enough to the higher frequency vibrations that phones make, but they did not have to be very reliable if we could get multiple attempts to tell if the phone is ringing. Even at 60% reliability, our chance of detecting a phone call after three repeated calls is over 90%.
So vibration sensors would be a simple and effective way of detecting an emergency situation using repeated phone calls.

3.16.2 Detecting signal from phone using strobing flashlight

While this method is simple on paper, the more difficult task is to get the phone to communicate that it is in distress.

It would be very simple to add some photoresistors in the phone compartment to detect when the phone's flashlight is flashing, but most phones do not normally strobe their flashlights, and if one does in some sort of emergency situation, it is certainly not a standard behavior.

Thus, it would be impractical to expect, or alter phones to communicate the emergency status using its flashlight.

3.16.3 User unlock using the remote

A very simple and straightforward method for the emergency unlock is to add it as a functionality to the remote's menu system. In a situation where the locker is not being used for discipline and instead simply to keep people away from their phone for a period, like a movie theater or class room, we could use the remote as an external interface for opening compartments. If the remote was mounted to a wall, someone could use it to unlock the compartment with their phone in an emergency situation. In a classroom setting, the teacher can keep the remote and trigger the unlock if a student needs to leave early, or respond to a family emergency.

3.16.4 Summary, part comparison and selection

While the flashlight idea is not practical, the vibration and remote override ideas should be effective and feasible to implement to provide the option for emergency access to a user's phone.

Vibration sensors
When we searched for common, arduino compatible vibration sensors, we only found two types; a mechanical vibration sensor module and a piezoelectric disk which gets connected to one of the MCU's analog inputs.

The mechanical sensor outputs a low when it detects vibrations and high otherwise it also has a potentiometer to tune the sensitivity. This type of sensor is reportedly prone to false positives, and would go off every time the Locker gets bumped, moved or jolted. To detect when a phone is ringing, we believed we could set a threshold for detections in the last x minutes to detect vibrations from phone calls reliably.

The piezoelectric sensor simply converts the mechanical vibration energy into an electric signal that we would have to process in software. To detect phone calls, we would have to repeatedly measure the signal from the piezoelectric disk and figure out how to tell the difference between being bumped, vibration from closing the door or pushing the buttons on the front and an actual phone ringing. This would interfere with our ability to use sleep modes as we would have to constantly (or very periodically) measure the signal from the piezoelectric disk since peeks in the signal could be very short.

Table 18 compares these two types of vibration sensors and displays our selection.
We chose the mechanical sensor as it does not require complicated and costly signal analysis and its sensitivity can be tuned.

Table 18: Comparison of vibration sensors and selection

	Mechanical sensor	Piezoelectric sensor
Cost	$1.30	$1.5
Output type	Binary High/Low	Signal waveform
Sensitivity	Adjustable	High
False alarms?	Yes, requires correction heuristic (Eg. num alarms in past 5 mins)	Depends on software algorithm

3.17 Programming Languages

Given that we selected the nRF52840 as the Study Buddy's MCU, we had to determine how we would be programming the board for use in the project. The first step of this was to determine the programming language that would be used to program our MCU.

3.17.1 C Language

Nordic, the creator of the nRF52840, provides software development kits written in the C language. The board particularly supports the C language through Nordic's nRF connect SDK. C is a procedural programming language that is extremely popular due to its versatility and the fact that many popular programming languages, such as C++, C#, and Java, have some relation to it. The popularity of the language is such that all of our group members have some experience with C, making it an appealing option if every member were to assist with the software design.

3.17.2 C++

By using the Arduino IDE, we could program the nRF52840 in a version of C++. The Arduino IDE uses a language based on a combination of C and C++. C++ is an object-oriented language created as an extension of C, using nearly the same syntax in most cases. Two of our group's members had some experience programming in C++, and they also had experience in Java and C, meaning we had sufficient knowledge to program the study buddy's MCU using the Arduino IDE. Using the Arduino IDE would give us the option of viewing many available resources to assist our software design and debugging.

3.17.3 CircuitPython

If we wished to use a high-level programming language to program the Study Buddy's MCU, we could have downloaded CircuitPython. CircuitPython is a version of MicroPython, a microcontroller-optimized offshoot of Python. Python is a high-level language that is popular for its abstraction and comprehensive standard library. However, all of our group members had very little or no experience with any version of Python.

3.17.3 MCU Programming Language Comparison

Table 19 compares the above described programming languages for the purpose of selecting one to program the final Study Buddy MCU.

Table 19: Programming language and development environment comparison

	C (nRF Connect)	C++ (Arduino IDE)	CircuitPython
Level	Middle Level	Middle Level	High Level
Group Experience	High	Medium	Low
Resource Availability	Medium	High	Low

Considering that some of our group members had experience in C++ and similar languages, and that the Arduino IDE has a large amount of supporting resources, we decided to use the Arduino IDE and its C/C++ language to program the Study Buddy's MCU.

4.0 Design Constraints and Standards

Standards are predetermined rules that must be adhered to during the process of designing or implementing something. In the field of engineering, standards play an extremely important role in the process of ensuring the design of a product that is reliable and secure. By complying to a universal standard, a product can demonstrate that it is universal, adaptable, and easily able to be incorporated into larger systems. These are all desirable characteristics. Our Locker design functions in a dependable and secure manner thanks to the fact that standards are based on tried-and-true best practices that are common in the industry.

4.1 Related Standards

There are certain standards that had to be followed when designing and building our system. These standards also impose restrictions, as is the case with all engineering-related matters. Not just the aspects that are unique to the software or hardware on their own, but also the overall design of our project was impacted by these rules and specifications. Standards are an integral part of every engineering subfield because they can act as a roadmap for the creation of a secure infrastructure.

4.1.1 Power and safety standards

Preventing harm, fire, and electrical shock is the main objective of maintaining power safety standards in electrical equipment. Both nationally and internationally, these standards are upheld. To indicate compliance within a particular economic region, products that meet the requirements of the standard are embossed with the recognizable safety mark from the associated standards organization. The CSA mark is the governing body that bestows the seal on the American and Canadian economic region. The International Electrotechnical Commission (IEC) and the International Organization for Standardization are the principal international bodies in charge of the aforementioned standards (ISO).**(11)**

4.1.2 Electricity Standards

When designing an electro-mechanical system for consumers, it is important to consider how electrical components can be separated from water. As a result of the chemical makeup of water, which makes it easy for electricity to conduct, water entering an electrical system increases the risk of component damage as well as safety concerns for anyone who comes into contact with it. Despite the fact that this project doesn't involve any outdoor activities, it is still essential to plan for specific scenarios that could arise in an office environment in order to protect everyone. A drink that was spilled on the locker box was one of these instances. In such cases, it is also necessary to protect visible components like wiring in addition to the system's core components.

4.1.3 Power Supply Standard

The Power Supply is an Essential Component that we have incorporated into our design. This is due to the fact that every election product requires some form of power in order to function, and there is a standard for power supplies that was created in order to create and control electrical safety standards. As a direct consequence of this, we arrived at the safety rules that various countries have established for their electrical systems, one can prevent electrical shocks, injuries, explosions, and fires that are all caused by electricity. These regulations are included in the electrical codes of various nations around the world. Presented further down on this page, you will find information regarding the safety standard that applies to power supply equipment such as batteries.**(10)**

<u>IEC 60950-1</u>
<u>SAFETY OF INFORMATION TECHNOLOGY EQUIPMENT</u>
IEC 60950-1 combines the first and second amendments to this standard from its second edition from 2005 (both from 2009 and 2013). With a maximum rated voltage of 600 V, the standard is applicable to office equipment and information technology (IT) equipment. Its primary goal is to protect against risks like mechanical instability, electric shock, fire, and dangerous temperatures.**(10)**

<u>IEC 60065</u>
<u>SAFETY OF AUDIO, VIDEO AND SIMILAR ELECTRONIC APPARATUS</u>
IEC 60065 is intended to prevent fires, electric shocks, and injuries in audio, video, and similar equipment, such as video projectors and photographic electronic flash units. As with previously discussed standards, not all protective measures apply to power supplies; however, those related to shock and fire require similar precautions regarding insulation, isolation, limiting voltage and/or current, and enhancing fire resistance. IEC 60065 also references IEC 60950-1 for SELV voltage limits, TNV (Telecommunication Network Voltages) for non-connected equipment, and IEC 60695-11-10 for component flammability categories.**(10)**

<u>IEC 60035</u>
<u>SAFETY OF ELECTRICAL/ELECTRONIC HOUSEHOLD APPLIANCES AND SIMILAR APPLICATIONS</u>
IEC 60335-1 is the first part of an international standard that outlines the safety requirements for devices designed for use in household appliances and similar applications with rated voltages of up to 250 V for single phase and up to 480 V for multiphase. In addition to part 1's general requirements, part 2 of the standard establishes additional requirements for particular device types. Part 2

enumerates over 100 device types due to the diversity of possible household appliances. In North America, UL 60335-1, CAN/CSA-C22.2 No. 60335-1, and NMX-J-521/1-ANCE are tri-national standards based on IEC 60335-1. The sixth edition of UL 60335-1 has been harmonized with IEC 60335-1 in the United States as part of the country's participation in the HOUS category of the IECEE CB Scheme. However, not all part 2 specifications are recognized by the United States. Part 2 requirements take precedence over part 1's general requirements. EN 60335-1 is a European standard that specifies how electrical equipment can comply with European directives such as the low voltage directive. Unlike the United States, the European Union recognizes the vast majority, if not all, of the over a hundred product-specific part 2 standards.**(10)**

<u>IEC 62368-1</u>
<u>AUDIO/VIDEO, INFORMATION AND COMMUNICATION TECHNOLOGY EQUIPMENT</u>
The IEC 62368-1 safety standards differ substantially from the IEC 60065 and 60950-1 standards, which govern the marketing of audio-visual products and computing/communications equipment in North America and Europe, respectively. As new technologies and markets continue to blur the distinctions between AV and ICT equipment, IEC 62368-1 is a unified replacement for the two previous standards. This transition is significantly more significant than a simple merger or name change because IEC 62368-1 adopts fundamentally distinct engineering principles and terminology. Similar to its predecessors, the 62368-1 standard applies to end-users, subsystems, and components such as power supplies.**(10)**

<u>IEC 61010-1</u>
<u>SAFETY OF MEASUREMENT, CONTROL AND LABORATORY EQUIPMENT</u>
IEC 61010-1 is an additional safety standard comparable to IEC 60950 and IEC 60601 that specifies requirements for measurement, control, and laboratory equipment such as x-ray machines and oscilloscopes. IEC 61010 also addresses the dangers posed by fluids, lasers, ultraviolet light, and the measuring circuit, in addition to protecting against electrical shock, fire, and mechanical or burn injuries. Not all of these dangers are directly related to the power supply of the equipment, but thankfully, the latest 3rd edition of the standard distinguishes between the general equipment requirements and the more specific measurement circuits, which are now covered by IEC 61010-2.**(10)**

4.1.4 Plugs and Sockets Standard

Today, every electronic device must be equipped with a jack-plug that can be plugged into an electrical outlet, as their operability is contingent on the availability of electrical power. AC voltages between 220 and 240 volts and AC voltages between 100 and 130 volts are the two most common types of voltages used in the indoor electrical systems of nations worldwide. The ranges of 100V, 110V, and 130V are considered to be low voltage. Some nations, including the United States and Japan, use low voltage out of concern for public health and safety. High voltage is the voltage range between 220-240 volts. Some nations, such as China (220V) and the United Kingdom, utilize high voltage because they place a premium on energy conversion efficiency. There are instances in which the voltage range of 110-130 volts is utilized, even in countries with a voltage range of 220-230 volts, such as Sweden and Russia. The American standard for wall outlets that we would use are type A and type B, one with a ground pin and the other without it.**(12)**

4.1.5 Rechargeable batteries standard

The IEEE Standard for Rechargeable Batteries governs the production of rechargeable battery packs and their utilization in a variety of applications. The standard specifies methods for the design and testing of battery cells and battery packs to reduce the likelihood of system failure. Following these instructions is necessary for the system's dependability, and failure to do so could result in battery system damage. Moreover, these rules are essential for the system's reliability. Manufacturers rely on these standards to ensure that their final products are safe for consumers' long-term use. The IEEE standard provides detailed specifications for rechargeable batteries designed for mobile computing. These requirements apply to our design. The standard focuses on a limited number of issues, some of which are within our control and others of which are pertinent to the production of these batteries. To ensure the proper production and handling of their products, the manufacturers are expected to discuss issues such as quality control, the manufacturing process, and packaging strategies. Due to the exponential increase in battery use around the world, these standards must be continuously reviewed and updated to ensure proper battery use.**(13)**

Components and battery-powered products, according to the Consumer Product Safety Commission (CPSC) of the United States, comply with the relevant voluntary standards. It is essential to consider these best practices when designing new components and products that are not yet required to comply with other voluntary standards. It is recommended to design battery-powered products using a system-based approach that takes thermal protection, charge and discharge protection, and product usage including:

- Cells that are suitable for the intended loads and conditions and were manufactured under strict quality control.
- Battery packs should include suitable battery management features, such as cell balancing, short-circuit protection, and charge control.
- Chargers that comply with any applicable voluntary standards and are appropriate for the product, its end uses, and its end products.**(13)**

4.1.6 USB Standards

The USB Implementers Forum, also known as USB-IF, is responsible for the creation as well as the ongoing maintenance of the USB standard. This is done to ensure that USB is a standard for the industry as opposed to a standard for a specific manufacturer. This corporation was established by the companies that were responsible for the development of the USB standard because those companies intend to keep making use of and improvements to the USB standard. In 1995, a conglomeration of companies, including Compaq, Digital, IBM, Intel, Microsoft, NEC, and Nortel, came together to establish the USB-IF as an industry standard. These companies were some of the very first ones to establish the organization. The organization now counts a significantly larger number of well-known companies as members, some of which are Hewlett-Packard, NEC, Microsoft, Apple, and Intel Systems. The USB-IF was able to deliver the first standard in a time frame that was considered to be reasonable. In 1996, just one year after the establishment of the organization that would later be known by that name, the very first USB standard was made available to the public.**(18)**

The universal serial bus, more commonly known as the USB, became commercially available in the year 1996. Registering on the USB-IF website is completely free for developers and doing so grants them access to the developer web-forums as well as the documentation. To participate in a working group, you will need to either be employed by a company that is a member of the organization or submit an application to become a member of the organization. The USB Implementers Forum is responsible for developing and upholding the USB specifications, which include Wireless USB, as well as managing a compliance program to ensure that USB products continue to meet a high standard of quality and are compatible with one another**(18)**. For our project we will not need a USB standard that has a massive capability of Data Transfer, however we will need a standard that has at least power transfer capability of 5V DC with a current up to 2 Amps to provide a decent power source for charging the devices that will be setting inside the locker.

4.1.7 Bluetooth low energy standards

The Bluetooth Special Interest Group (SIG) first outlined the Bluetooth Low Energy (BLE) wireless technology in 2010 as a part of the Bluetooth 4.0 standard. BLE stands out for its constrained range and low power requirements. Since then, Bluetooth Low Energy (BLE) technology has developed and is now an essential part of the Internet of Things (IoT). BLE is actually ideal for low-resource devices, like those that are frequently found on the Internet of Things. Some examples are actuators and small battery-operated sensors. The fact that BLE is widely used in consumer electronics products like smartphones, in contrast to other Internet of Things technologies, which sets it apart from. Because a smartphone can easily change into a user interface or a gateway in Internet of Things scenarios, this distinctive feature of BLE makes it easier for users to interact with BLE devices that are located in their immediate environment.**(19)**

The IETF IPv6 over Networks of Resource-Constrained Nodes (6Lo) working group is currently developing 6BLEMesh, a significant standards-based solution for BLE mesh networking. The Bluetooth Mesh standard is yet another significant standards-based choice for BLE mesh networking. The 6BLEMesh extension to RFC 7668 makes it possible for BLE mesh networks based on IPv6. In Bluetooth Low Energy (BLE) networks, IPv6 over Low Power Wireless Personal Area Network is used to specify IPv6 over star topology (6LoWPAN). The basic ideas underlying 6LoWPAN and IPv6 over star-topology BLE networks are presented in the first step of the procedure. The most important aspects of 6BLEMesh will then be described.**(19)**

The original purpose of 6LoWPAN, an adaptation layer, was to effectively enable IPv6 over IEEE 802.15.4 networks. In many cases, IEEE 802.15.4 networks, like BLE networks, contain devices with limited resources and deliver relatively low bit rates. Resource-intensive networking environments were assumed when IPv6 was being developed, but they are not at all fundamentally similar to IEEE 802.15.4 networks. In order to maximize efficiency and comply with IPv6 specifications, devices must actually add an adaptation layer between the IPv6 layer and the IEEE 802.15.4 layer. 6LoWPAN is primarily composed of the following three mechanisms:

i) compression of IPv6 and UDP headers,

ii) optimized IPv6 Neighbor Discovery (ND).

iii) fragmentation functionality.

The first two mechanisms make it possible to operate with less energy and bandwidth. The 6LoWPAN header compression technique makes use of both the intra-packet redundancy and an expectation of the values that are frequently found in the header field. A 6LoWPAN-optimized IPv6 network driver will lessen the use of multicast and enable the inclusion of energy-saving intervals by mandating that interactions be started by energy-constrained devices. Unlike the IPv6 standard, which mandates that packets be 1280 bytes in size, IEEE 802.15.4 has a maximum frame payload size of only 100 bytes. The transmission of these bigger packets is thus made possible by 6LoWPAN fragmentation.**(19)**

4.1.8 SPI standards

A standard interface bus known as SPI is used for communicating with peripherals such as analog-to-digital converters, flash memory, real-time clocks (RTCs), and sensors, amongst other devices (Serial Peripheral Interface). The Serial Peripheral Interface (SPI) bus that Motorola developed makes it possible for master and slave devices to communicate with one another in an asynchronous, full-duplex serial fashion.

The Master Out Slave In (MOSI), Master-In Slave-Out (MISO), and Slave Select (SS) lines make up a typical SPI connection. In this connection, a master and slaves are connected via the serial clock (SCK) line. Each slave has a different SS line, but they can share the SCK, MOSI, and MISO signals. High-speed data streaming is possible because overhead is kept to a minimum because the SPI interface does not define a protocol for data exchange. There are four different modes that can be formed by specifying clock polarity (CPOL) and clock phase (CPHA) as either "0" or "1," enabling flexibility in the communication between the master and the slave. The leading rising edge of the clock is used to sample data if CPOL and CPHA are both set to 0, (this configuration is referred to as Mode 0). SPI bus slave communication usually occurs in Mode 0, which is the mode that is used the most frequently. In Mode 2, data is sampled at the leading falling edge of the clock if CPOL is set to "1" but CPHA is set to "0." Similar to this, data is sampled at falling trailing edge when CPOL = '0' and CPHA = '1' (Mode 1), and when CPOL = '1' and CPHA = '1' (Mode 3), data is sampled at rising trailing edge.

3-Wire SPI Bus and Multi-IO Configurations

The SPI interface now includes additional IO standards in addition to the traditional 4-wire arrangement. These include 3-wire for fewer pins and dual or quad I/O for higher throughput. The MOSI and MISO lines are combined to form a single, bidirectional data line when the 3-wire mode is in use. Half-duplex transactions are employed in order to facilitate bidirectional communication. In addition to lowering the maximum throughput, reducing the number of data lines, and operating in half-duplex mode both have negative effects. Due to the low performance requirements of three-wire devices, pin count is frequently prioritized over performance. Multi I/O variants like dual I/O and quad I/O add additional data lines to the standard in order to increase throughput. Components that employ multiple input and output modes can match the read speeds of parallel devices while still providing fewer pins. Flash memory can now support random access and direct program execution thanks to this improvement in performance.

In addition to the conventional four-wire configuration, the SPI interface has recently been enhanced to support additional IO standards. The new specification calls for dual or quad I/O, which increases throughput, and a connection using only three wires, which reduces the number of pins. When used in the 3-wire mode, the MOSI and MISO lines are combined into a single, bidirectional data line. In order to facilitate two-way communication during the transaction process, half-duplex transactions are used. Operating in half-duplex mode and reducing the number of data lines both have negative effects in addition to lowering the maximum throughput that can be accomplished. Due to their low performance requirements, three-wire devices frequently prioritize pin count over performance. Multi I/O variants like dual I/O and quad I/O supplement the standard with additional data lines in order to boost throughput . Components with several input and output modes could indeed match the read speeds of parallel devices despite having fewer pins overall. Now that the performance has improved, flash memory can support both random access and direct program execution (execute-in-place).

4.1.9 I2C standards

The Inter-Integrated Circuit (I2C) Protocol was developed to facilitate communication between a number of "peripheral" digital integrated circuits (also known as "chips") and one or more "controller" chips. Similar to the Serial Peripheral Interface (SPI), it is only meant to be used for communicating over relatively short distances between parts of a single device. Information can be transferred between devices using just two signal wires, similar to asynchronous serial interfaces (RS-232 or UARTs, for example). I2C was first developed by Philips in 1982 for use in a range of chips they made. The initial specification only allowed for communications at 100 kHz and 7-bit addresses, which limited the number of devices that could be connected to the bus at 112. There are a few reserved addresses that are never going to be used for valid I2C addresses. The 1992 initial release of the public specification included both the 10-bit larger address space and the 400 kHz faster mode. The device will typically support I2C only once more. One example of this is the ATMega328 component, which can be found on many Arduino-compatible boards. The first of the three additional modes that have been specified is the fast mode plus, which can communicate at frequencies up to 1 MHz. Running at 3.4MHz, the second mode is also known as the high-speed mode. The third mode, also known as the ultra-fast mode, operates at a frequency of 5MHz.**(20)**

I2C in its "vanilla" form received an extension in the form of a variant that was made available by Intel in 1995 and given the name "System Management Bus." (SMBus). A format known as SMBus, which is more strictly regulated, has been developed with the goal of improving the predictability of communications between the various support ICs that can be found on PC motherboards. I2C can support devices operating at speeds between 0 kHz and 5 MHz, whereas SMBus can only support devices operating at speeds between 10 kHz and 100 kHz. This is the most notable difference between the two bus systems. In order to maximize their compatibility with embedded I2C systems, many SMBus devices will support low-speed operations. This is the case despite the fact that the SMBus protocol has a clock timeout mode that renders such operations illegal.

This is because SMBus contains a mode that prevents low-speed operations from taking place. **(20)**

I2C Hardware

SDA and SCL are the two signals that make up each and every individual I2C bus. The signal that represents the clock is called SCL (Serial Clock), and the signal that represents the data is called SDA (Serial Data) . The clock signal is produced by the bus controller that is currently being utilized; however, certain peripheral devices may on occasion cause the clock to be forced to a low state in order to prevent the controller from transmitting additional data (or to require more time to prepare data before the controller attempts to clock it out) . I2C bus drivers are what's known as "open drain," which means they can only drive the corresponding signal line to its low state . Bus contention, which happens when one device tries to drive the line high while another device tries to pull it low, is therefore not a possibility. Bus contention happens when one device tries to drive the line high while another device tries to pull it low. Because of this, there is no longer any possibility that the drivers will take damage, nor is there any possibility that the system will waste an excessive amount of power. Each signal line has a pull-up resistor connected to it so that the signal will return to its high state when there is no device asserting a low value on the line. This is done so that the signal can function properly.**(20)**

4.1.10 UART standards

A user-configurable asynchronous serial communication protocol for hardware with variable speed is called the Universal Asynchronous Receiver/Transmitter (UART). When a clock signal is not used to synchronize the bits sent from the transmitting device to the receiving end of the connection, asynchronous communication takes place. The sending of data in a parallel format is handled by the transmitting UART in conjunction with the directing data bus. The data will then be sent serially, one bit at a time, over the data line to the UART receiver end. Consequently, the data receiver will convert the serial data to parallel data. The UART's lines are used as a communication medium for sending and receiving data. Each of the transmit and receive pins on a UART device is designated for either sending or receiving data . The rate at which data is sent over a communication channel is known as the baud rate. The baud rate must be set to the same value on both the device sending and the device receiving the data if using UART . The baud rate that is set will determine the maximum number of bits that can be transferred per second over a serial port.**(21)**

Data Transmission

Transmission takes the form of a packet when UART is used as the communication protocol. The component that manages the connection is in charge of creating serial packets, controlling the hardware lines, and establishing the connection between the transmitter and receiver. Start bits, data frames, a parity bit, and stop bits make up a packet.**(21)**

<u>Start Bit/ Stop Bits</u>

Even when the UART is not actively sending data, the data transmission line is typically maintained at a high voltage level. This is done for reliability reasons. Before beginning a data transfer, the transmitting UART device will first perform a transition on the transmission line that will take it from high to low for one clock cycle. As a direct consequence of this, the transfer can then get underway. When the receiving UART detects a shift from high to low voltage, the process of reading the bits contained in the data frame at the same rate as the baud rate commences. In order to let the receiving UART device know that the data packet has been sent successfully, the sending UART will drive the data transmission line from a low voltage to a high voltage for one bit or up to two bits.**(21)**

<u>Data Frame/ Parity</u>

The actual data that is being transferred is contained in the data frame. If a parity bit is present, the value's length can range between five and eight bits. If the parity bit is not required, the length of the data frame can be up to nine (9) bits. When sending data, the least significant bit is typically transmitted first. The evenness or oddness of a number can be described using the parity concept. The receiving UART can identify whether any data was corrupted during transmission by examining the parity bit. Bits can be corrupted by electromagnetic radiation, out-of-phase baud rates, long-distance data transmission, and other factors. The receiving UART counts the number of bits with a value of 1 after reading the data frame, then determines whether the sum is even or odd. If the parity bit is a zero, also known as even parity, the bits in the data frame that are a one or logic high should add up to an even number. If the parity bit is set to 1, which indicates an odd parity, then the sum of all the logic highs in the data frame that are set to 1 should be an odd number. The UART can determine whether there were any transmission errors when the data and the parity bit match. However, the UART will notice that bits in the data frame have changed if the parity bit is a 0 and the total is odd or a 1 and the total is even.**(21)**

4.2 Project Constraints

Another factor that contributes to the success of a design is having an awareness of the realistic design constraints that must be adhered to. This is in addition to having a working knowledge of the applicable laws and standards that have been imposed by the relevant governing bodies. When it comes to putting the product to use, there are a number of potential obstacles that we will discuss in this section of the article. Because they can be broken down into the following categories, it is necessary to consider each practical constraint in its own, and they are:

- The Economic Constraints.
- The Time Constraints.
- The Environmental Constraints.
- The Social Constraints.
- The Political Constraints.
- The Ethical Constraints.
- The Health and Safety Constraints.
- The Manufacturability Constraints.
- The Sustainability Constraints.

4.2.1 Economic Constraints

The major economic constraint for this project is that we needed to fund the project on our own. As a result, we couldn't afford to spend too much on this project due to our own limited funds and agreed to collectively spend around $285. We may have been able to spend more but it would be in our best interest to stick around this price. Due to this constraint, we cannot afford to purchase the more expensive products that could help with the reliability and security of our design. We instead chose parts that offer enough reliability and security to meet our minimum demands while still staying affordable. If our product were to be commercially available, another economic constraint would be to keep the manufacturing costs of the product low enough to be competitive with other available products. We would need to keep it low enough that we could sell the product at a reasonable cost while still making a profit.

4.2.2 Time Constraints

The major time constraint for this project is that we only had 2 semesters to complete it. We had until April of 2023 to complete, test and finalize our design. Due to this limited amount of time placed, we adhered to a strict schedule to avoid running the risk of falling behind and may not have enough time to implement all that we have had planned. To accomplish these sets of goals, we followed our milestone progression chart to complete this project in time, staying ahead of the milestone progression in case a task was more difficult to complete then expected. If we failed to complete a milestone in time, that means we would have to work much more to catch up if we intended to complete this project.

Another major time constraint was part availability and shipment time. Some of the parts we wished to implement in our design may not be immediately available meaning we would need to wait until more can be manufactured. This could cut into our time for building and testing our designs. Even more time can be lost if the parts we selected became damaged or were deemed unfit for our design so caution needed to be taken when handling them. Shipment time could also cut into our time if it was significant enough from the start or if there was a delay. These factors could cut into our time to test our design and would make each mistake all the more costly in terms of time to finish.

These time constraints affected our overall project in that we may not have been able to implement everything we wished to add to the project. It is important that we at least completed our stated basic requirements in time as well as ideally some of our advanced requirements. These time constraints also affected our project in that if given more time, we could have added additional features to our project or optimized existing features. Given these time constraints however, we believed that what we outlined should be realistically achievable given no extreme errors or miscalculations.

4.2.3 Environmental Constraints

It is more important than it has ever been to think about how emerging technologies will have an effect on the natural world, and our effort is not an exception to this rule. Despite the fact that our project is dependent on the use of electricity, there is no guarantee that it would have a direct effect on the surrounding environment. As a result, it was of the utmost importance that we conduct exhaustive research into the myriad of approaches that we could take to enhance the energy efficiency of our design. Due to the specifics of our project, we were unable to utilize renewable energy sources including solar energy as our primary source of electricity, since the locker would be used inside a building.

The environmental constraints that had been placed on this project primarily take the form of restrictions that are related to the effects that this project would have on the environment rather than restrictions that the environment would impose on this project itself. Since the environment was not taken into consideration as a meaningfully constraining external force in the design of this project because it was meant to be used primarily indoors at temperatures comparable to those found in common rooms.

However, there are environmental factors that needed to be taken into consideration because this project was going to have some kind of effect on the surrounding environment. Most importantly, the materials that were used in this project may have had a negative impact on the environment as a result of the manner in which they are produced or the manner in which they are ultimately disposed of. This is a possibility because of the ways in which these processes take place. For instance, the material that is used by a 3D printer to print the components, as well as the chemicals that are present in the batteries that are required by the system, are examples. If they are not utilized appropriately, these materials have the potential to contaminate the surrounding environment and present a risk to the health of those who come into contact with them. As a result, we made it a point to take into consideration this limitation as thoroughly as possible when selecting components and materials for the project as a whole in order to reduce the likelihood of any adverse effects on the surrounding environment that could be caused by it.

4.2.4 Social Constraints

The Study Buddy is an indoor product and is intended for helping people to control themselves and for parents to control their children's urge to just play with electronics all the time and not doing healthy and beneficial activities like playing outside or their chores like cleaning and studying. So, socially it is acceptable and gives people a chance to practice a good lifestyle. If the locker were used in social places like cinemas or gyms for example, it would help keeping people's properties safe and secure. Additionally, with the advanced features of locking and unlocking the locker, it is giving an easy, fast, and practical experience for consumers to get their properties in and out of the locker with the high security it features.

4.2.5 Political Constraints

We do not believe that this project has any political constraints. As a project that we intended for people to use for themselves for the purposes of self discipline, we did not see any major way that our project would or could be influenced politically due the separate nature of its operation. Even in the situation where our project is used in a public setting, its operation is too mundane to have any sort of political ramifications. Unless some situation like our emergency unlock feature not working causes many high profile, preventable incidents or otherwise manages to change the collective perception on itself of other aspects of its functions, we do not think that these kinds of constraints apply to our project.

4.2.6 Ethical Constraints

The major ethical constraint for this project is that it may be used to lock up important devices and objects of worth. Due to the security systems used in the project, it may be difficult to remove things from the device when they are critically needed due to an emergency. If one of the functions of the device malfunctions making it difficult to open or the user is unable to personally open the device due to obscure instructions, potential harm may come to a person or property. It is important for us to make sure that the device is reliable enough that none of the locking mechanisms fail to properly unlock. If harm falls to the person or property due to our product failing, we will be ethically responsible for not ensuring the reliability of the product.

4.2.7 Health and Safety Constraints

Despite the fact that the group decided to purchase less expensive substitute materials, we came to the conclusion that these materials would only be obtained from businesses or individuals who are licensed to sell the products in question. Without the appropriate certification, sellers wouldn't be able to answer questions about how the materials are manufactured, whether or not they are compatible with the other components, and, most importantly, whether or not they are safe to use. The realization that each component has been analyzed and given the all-clear makes using the product a more secure option to pursue. This realization alone makes using the product a safer option to pursue.

The fact that this product's circuitry must be connected to the battery pack constitutes an additional safety restriction that has an effect on the product's design. There are a few components that absolutely had to be incorporated into the BMS in order to guarantee that the end product is completely risk-free. For instance, a BMS temperature sensor was incorporated into the design of the battery cells. This thermistor immediately puts an end to the process of charging the battery pack if it detects that the temperature of the battery pack has reached an unsafe level while it is being charged. By incorporating even small vents into the layout of the design for both the locker and the battery pack, it is possible to ensure that the temperature of any component will never rise to an unsafe level while it is being used. The BMS is also responsible for preventing the battery from being overcharged, which could result in an explosion or cause the user to sustain injuries. In the event that the battery cells are overcharged, the device might not function as intended. If the process of charging the battery is regulated, there will be less of a chance that the battery will suffer a catastrophic failure, which could harm the customer.

4.2.8 Manufacturability Constraints

Some manufacturability constraints for this project were part availability and cost. Since we needed to order the parts needed rather than make them ourselves, we were limited to what parts are currently being made. We had to be careful that the parts we selected to design this project wouldn't become unavailable before we could properly order them. We needed to keep in mind the amount of that part available from manufacturers as we may have needed to order more than initially needed if some become damaged for any reason. We also needed to make sure we selected parts that we could afford, further limiting what we could manufacture.

If this project was available for commercial use, we would need to keep in mind manufacturing costs to make sure we could sell the product at a profit while remaining competitive with similar products. Our manufacturing cost of around $285 means that our product would need to sell above typical market price for similar products which seem to average around $100. With extra features included in our product, it could potentially be market competitive but it would be too risky to manufacture at that cost. Mass producing the product for sale may also be problematic as we ordered various parts from different manufacturers rather than building them ourselves. This means the manufacturing of our product would depend on the availability of these parts.

4.2.9 Sustainability Constraints

During the planning stages of this project, there were a lot of sustainability considerations that needed to be taken into account, which presented some challenges to the team. To begin, given that the collective is made up of some components that are mechanically moving, it was essential to ensure that these components were crafted in a manner that would either prevent them from being obstructed or make it more difficult for them to be broken. During the process of designing, this was one of the primary concerns that were brought up. The enclosure in which the locking mechanism operates needed to be robust enough to preserve its structural stability over an extended period of time. Additionally, the enclosure needed to be able to protect itself from being tampered with, which was another factor that needed to be taken into consideration.

It is to be expected that some of the unit's components will eventually become worn out or broken given that the unit as a whole is composed of several mechanical parts that come into contact with one another as well as several doors and their hinges. The design takes into account a variety of considerations, one of which is the capability of disassembling moving parts and replacing each of the individual components. The design of the collective unit has resulted in a significant extension of the product's useful life span, which is due to the fact that the ineffectiveness of the product as a whole won't be caused by the failure of a single component or the gradual deterioration of a material over time.

Due to the fact that the vast majority of the electrical components are installed inside the locker's main body, it was anticipated that the study buddy locker's electrical system would function normally and without any issues. The main body of the locker serves as the mounting location for the wired components and the printed circuit board. This assists in protecting it from any potential environmental conditions as well as any potential physical harm from the outside.

5.0 Hardware and Software Design

Even though many design choices are based on tried-and-true design principles, some (if not most) are also left to the designer's discretion. For example, in reality, if two separate engineers were provided with the same parts and the same design goal, both of their designs would be distinct. It is crucial to understand that a superior device is produced when excellent components are combined with a substandard design. Due to this, designing a system necessitates careful consideration and close attention to every little detail. The two main components of almost all electrical or electronic systems in use today are hardware and software. The functionality of software and its capacity to perform the tasks envisioned by the product's designer depend on the physical foundation that the hardware offers. The specifics of the development procedure used to create the software and hardware for our system are the main topic of this section. In this section we concentrated on the system design as a whole while also going deeper into each individual subsystem to examine it in more detail. A key technology that was incorporated into the overall design of the product or system is the primary focus of each individual subsystem. This section will also explain and provide details for the specific systems that are a part of each subsystem.

5.1 Locker power subsystem design

The power subsystem is in charge of controlling the entire set of power controls for the locker. It is necessary for each of the devices and system components that make up the main body of the locker to be able to get their needed power from the power subsystem. The block diagram Figure 9 that can be found below illustrates how each component was chosen to carry out a particular function in the subsystem. It is essential to have an accurate understanding of the amount of power that is being consumed by each component in the locker system if one is to successfully design the power subsystem. Before we could test the functionality of other components, we first needed to ensure that the power subsystem is operational and then connect it to other system components. This subsystem, which involves many circuits, has a significant impact on the way in which the locker is able to perform its functions. Because it enables the installation and operation of the system's other components, this subsystem is essential to the system's proper operation and functionality.

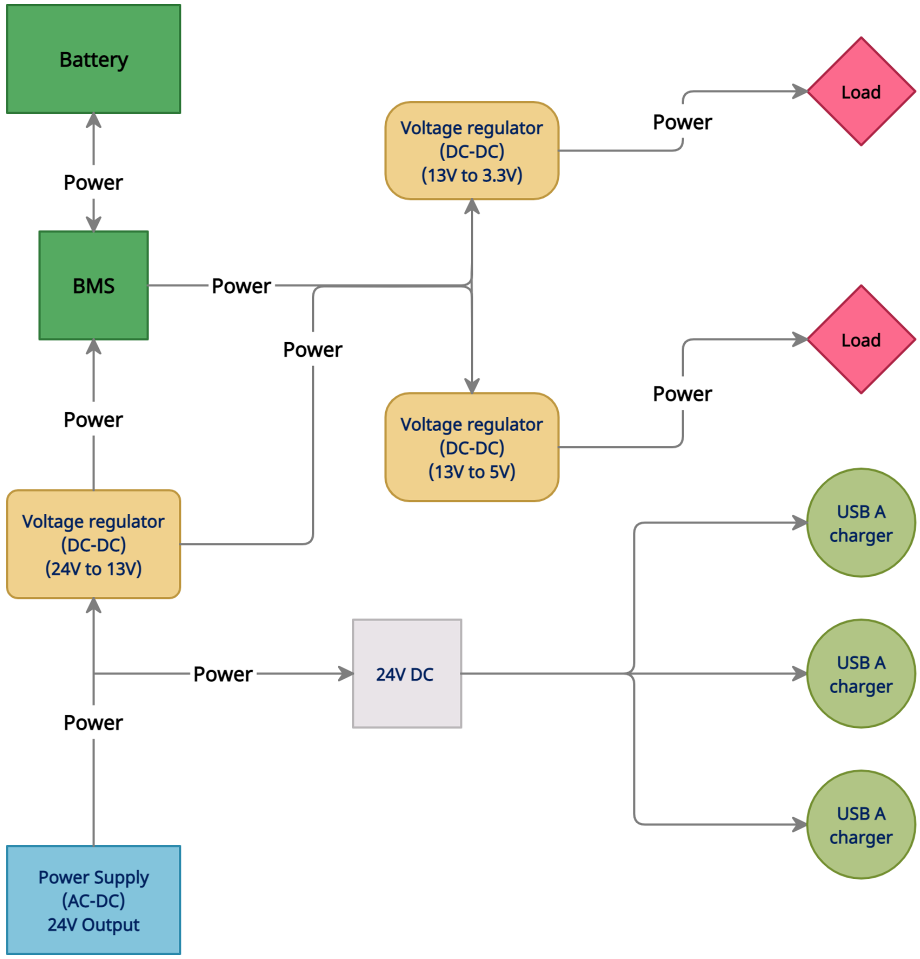

Figure 9: Functional block diagram of the power subsystem

5.1.1 Power supply

When setting up this configuration, the power supply is the component that requires the most attention to detail. If it is able to effectively deliver constant power, the system will continue to function as it was designed to do in the long term. If the power source is not present, then some of the other components of the system will not function. The AC signal coming from the wall serves as the primary source of power for this setup. We had two different goals for utilizing the wall AC signal. First, it serves as the primary power source for the locker system and provides electricity for the USB A charging ports for which the electronic devices that are stored inside can be charged from. Second, in order to get the lithium-ion battery pack ready for use, it needed to be charged from the power supply coming from the wall outlet. There are restrictions placed on the locations where electricity can be supplied by wall power. When it is receiving power by an electrical cord, the locker becomes less portable. One more reason is that there is always the possibility that the power grid will go out, and we do not want any of the electronic devices that are stored in the locker to become trapped in the event that this does take place. This indicates that the battery backup will be utilized during the times of such events.

As we were aiming for a high power rated power supply and also have a relatively small footprint, we used a switching mode power supply. The fact that switching mode power supplies include a rectifier, a filter capacitor, a series transistor, a regulator, and a transformer makes them more complex than other types of power supplies. Furthermore, switching mode power supplies usually cost more than a linear one. By using the regulator and the series transistor to transform the AC voltage into a regulated DC voltage, the unregulated DC voltage is produced. In order to enable a much smaller power supply and significantly reduce the size of the transformer, this DC is chopped to a constant high frequency voltage. Consequently, a much smaller power supply is possible. Due to the complexity of the power supply, it is unlikely to use this type of power supply for low production or cost-effective low power applications. Each transformer must be constructed in accordance with the customer's requirements.

Moving to the needed specifications for the output of the power supply, first we were aiming for a 24 volts output for a couple of reasons. Initially, the USB-type A ports can offer a charging voltage at its output of 12 volts, so we wanted to make sure that the ports receive enough input voltage to operate smoothly. Also, we would have the Battery Management System taking a steady (12.6 to 13) volts from the output of a voltage regulator after the power supply, which means we certainly have to provide a good input that is high enough to compensate for the dropout voltage of the DC-DC voltage regulator. Moreover, for lowering the costs of the final PCB for potential consumer product production, having a high voltage trace means lower current going through them, which reduces the size of the copper traces on the PCB. That is also helping with one environmental concern of using more materials than needed.

Second, the system's maximum total need of power is calculated based on the output currents needed by each component in the system. First, at the 3.3 volts voltage regulator output the estimated current needed is about 0.07 Amp, which yields 0.231 watts. At the 5 volts voltage regulator output the estimated current needed is 1 Amps, which yields 5 watts. At the input of the BMS we would need 0.5 C rating of the battery pack capacity in current to charge it, which means for 3450mAh battery pack, we would need around 1.7 Amps with 12.6 volts, and that yields 21.42 watts. Lastly, the three USB-A ports are rated at 24 watt each, which yields 72 watts total for the USB charging ports. In total, we needed at least 98.6 watts coming from the output of the power supply. After estimated calculations of the potential power losses based on the efficiencies of the different voltage regulators and other components on the power subsystem, we added another 6 watts to the total needed power. Finally, compensating for any unpredictable events, putting 30% more power on the budget makes the power supply perfect and dependable for our system, which yields a final total power rating of 128 watts, that is almost 5.34 Amps of current at the 24V output of the power supply. Figure 10 below shows the chosen power supply for the main locker. This power supply is capable of providing a nominal current of 6 Amps at 24 Volts at its output.

Figure 10: Functional power supply from inShareplus

5.1.2 Voltage regulators

In electronics, a buck regulator, also known as a step-down switching voltage regulator, is a type of voltage regulator that can be built from scratch with only a few components. This is entirely within the realm of possibility. For instance, a simple design will only consist of a single transistor, which will primarily serve the function of the switch. A diode, an inductor, a capacitor placed across the output, and a second capacitor placed across the input are the components that make up this circuit. It is likely that additional components, such as an error amplifier, voltage reference, switch driver, oscillator, and a comparator, will be required in order to implement a solution in a setting that is representative of the real world. As there is such a wide choice of well-proven, highly integrated monolithic DC-DC voltage regulators accessible at a minimal cost and with a high degree of reliability, just a small proportion of engineers choose to go the path of discrete component design. This is due to the fact that discrete component design requires a higher level of integration.

Selecting a regulator relying on the necessary specifications of the design, such as input and output voltages needed, peak load current, and maximum voltage ripple, is often easier. When developing more complicated designs, it is necessary to take into consideration a number of factors, including efficiency, frequency response, and transient response. Chip manufacturers can provide an incredible variety of options that meet the majority of the requirements, but it is not feasible for them to develop a product that is optimally suited to each and every possible circumstance. As a direct consequence of this, the designer will need to complete some additional tasks.

Although, a chip that contains a pulse-width modulation (PWM) controller, a bypass diode, and switching components such as MOSFETs is a typical place to begin when developing a low-current design (one that is less than 10 Amps), the amount of work that needs to be done is directly proportional to the level of integration of the one-piece solution(s). The designer is responsible for determining the need for an external inductor, bypass capacitors, and any other passive components that are necessary for input and output filter circuits. Even though there is a wealth of information available from manufacturers and other sources on how to go about designing a power supply based on a monolithic regulator, the process is still difficult and time-consuming. It involves computation as well as numerous loops of hardware prototype testing to see how the conceptual circuit performs in reality, and then making adjustments to precisely meet the specifications. Even though there is a wealth of information available from manufacturers and other sources on how to go about designing a power supply based on a monolithic regulator, it was possible that this process would take some time.

For our locker main system, we would need three voltage levels, so we used voltage regulator circuits. The first one is at the Battery Management System (BMS), which would need to charge the lithium-ion batteries optimally 4.2 volts for each cell in series. That means for the 3S pack that we have decided on using, we would need 12.6 volts for the whole battery pack. The other specification that the BMS needs to properly charge the batteries, is a constant current source of 0.5C rating of the battery pack capacity, which is 3450mhA. Depending on the previous voltage and current, the first voltage regulator is taking an input of 24 volts from the power supply and output a steady 13 volts with a maximum current of 0.5 Amps to be safe, since this voltage regulator is also feeding the other two voltage regulators of output 5 & 3.3 Volts. This voltage regulator is a switched-mode one, which is known for its high efficiency, and the design below can have an efficiency up to 90 %. Below is Figure 11 the schematic that we are relying on to build this voltage regulator circuit. The obtained WEBENCH circuit design has been modified to include a reverse current protection and current control feature when charging the battery pack. The current control circuit protects the battery pack from charging too quickly by pulling specific current decided by a potentiometer, and also protects the IC from being damaged.

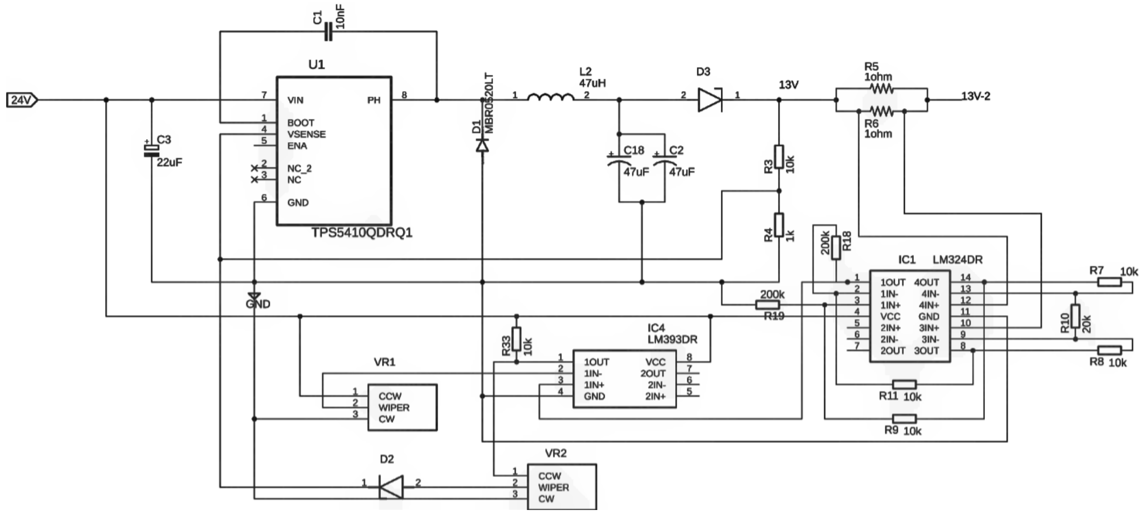

Figure 11: 13V voltage regulator circuit
*Schematic partial credit to webench**(27)***

The second and third voltage regulators feed the main control system of the locker, which consists of the main MCU, peripherals, and servo motors to lock the compartments' doors. Each one of these components needs a certain amount of power to operate properly, and since there are different voltage needs for each component, we are obligated to have two main voltage levels going to the system. The first one is the 5 volts output that is feeding most of the components including the servo motors. This switching voltage regulator will need most of the power, as the estimated current need of the 5V voltage regulator output is (0.4 to 0.7) Amps, thus the design of it is having an output of 1 Amp to be on the safe side. The other voltage regulator is also a switched-mode one with an output of 3.3 volts, but it does feed many components, and the estimated current at its output is 0.07 Amp with the extra power safety included. The Figure 12 below shows the schematics for the 5 volts output voltage regulator circuit, and the 3.3V is going to be based on another IC, that is (LMR33610) which can have high efficiency at low output currents.

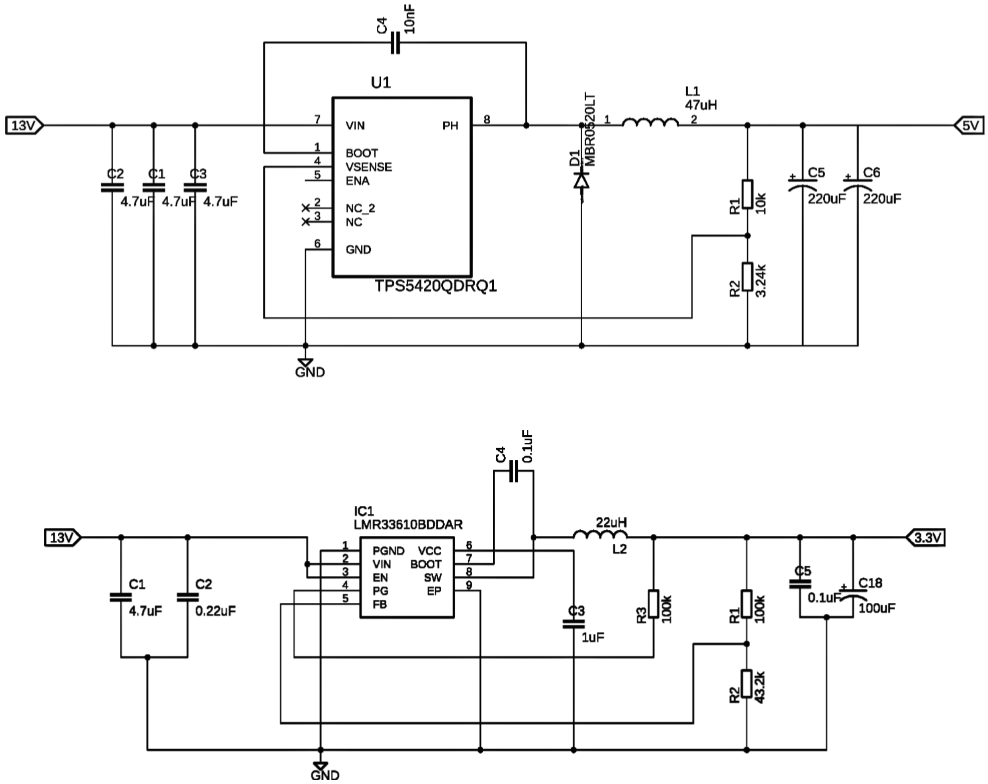

Figure 12: 5V & 3.3V voltage regulator circuits
Schematics credit to webench ***(27)***

5.1.3 Secondary power source (Battery Pack)

The lithium-ion 18650 battery has a height of 65 millimeters and a diameter of 18 millimeters. As its name implies the dimensions of the battery is measured in millimeters. They are not compatible with batteries of either the AA or the AAA size. The 18650-battery type is useful in a wide variety of devices, including those that can be recharged and those that draw a lot of current due to its high level of capabilities, which include more than 400 charge cycles and increased energy density. The 18650 Li-ion battery is a flexible battery that can be utilized in a wide variety of different applications. These include electric cars and scooters, power banks, and a variety of utility devices such as emergency lighting, torchlights, and other similar devices. This battery has managed to earn a great deal of interest from the electronic industry as a result of the numerous safety features it possesses, as well as its high output current and large energy capacity. To be more specific, a certain portion of tolerance is permitted for the length and diameter of 18650 batteries. Measurements such as 18 millimeters by 0.3 millimeters and 65 millimeters by 0.5 millimeters can be found on the datasheets and features of Li-ion cells. It is important to keep in mind the standard size, which is 18x65mm; the designers and manufacturers of electronics and batteries will take care of the rest of the details. The size of the

18650 battery is able to be utilized despite the presence of a variety of appliances or devices that inhibit the utilization of other locally developed technologies.

Although the single cell size is not that important for our design, the battery pack as a whole might present a space problem in our locker design. However, if we managed to put the individual battery cells in a flat order, the battery pack as a whole would have more tolerance towards the inner control room space inside the locker. Thus, for a configuration of three 18650s battery cells, the total dimensions of the battery pack are 54 x 65 millimeters plus some tolerances, which is a reasonable battery pack size.

The vast majority of battery chemistries can be connected to one another in either a series or parallel configuration. It is essential to use batteries with the same capacity (Ah), voltage, and type, and a designer should never mix and match batteries of different brands or sizes. A cell that is weaker than its neighbors would cause the pack to be unbalanced. Because a battery's overall strength is equal to that of its weakest link in the chain, and this is of the highest concern when the configuration in use is a series one. It's possible that some packs are connecting their components in a series while others are connecting their components in parallel. Batteries for laptops typically consist of four Li-ion cells, each with a voltage of 3.7 volts, connected in series to reach a nominal voltage of 11.1 volts. It is possible to increase the capacity of one of these cells from 2,400mAh to 4,800mAh by connecting two of them in parallel. This configuration is referred to as a 4s2p setup, which indicates that there are four cells connected in series and two cells connected in parallel across the board. If insulating foil is placed in between the cells, then the conductive metallic skin won't be able to cause an electrical short.

For our system needs, we would prefer a high voltage input to the voltage regulators with 5-volts and 3.3-volts output, since that would give us some advantages. One of them is a nominal input voltage that is high enough compared to the dropout voltage of the voltage regulators, even if the battery pack is about to drain out of power, the nominal voltage would be closer to 8 volts, which is pretty good as an input voltage to be regulated. The other advantage is that we need thinner copper traces on the final PCB design, as the current needed to carry the power to the voltage regulators is smaller due to the higher output voltage from the battery pack. For those two reasons specifically, we arranged the three battery cells into a series configuration (3S), see Figure 13 below, which would give us a maximum voltage of 12.6 volts at maximum charge, and a nominal voltage of (10.8-11.1) volts until the battery pack is near draining out of energy. Although we chose to put the battery cells in series, and that will make the total pack capacity the same as one battery cell -which is 3450mAh,- we found that this capacity is enough to reach our goal of making the locker operating time more than two hours.

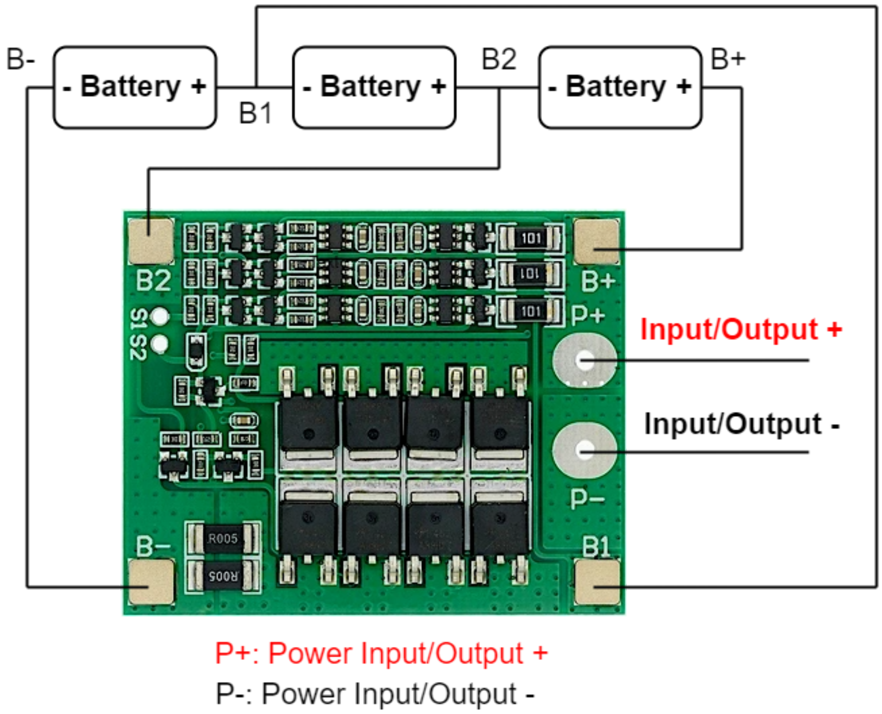

Figure 13: Battery pack configuration of 3 cells on series (3S) connected to the BMS

5.1.4 Battery Management System (BMS)

A battery management system, also known as a BMS, is essential for ensuring the continued security of a battery pack and maximizing the amount of time that a battery can serve its intended purpose without failing. The 3s (three cells in series) Battery Management System provides the user with several different types of protection, including protection against overvoltage, protection against overcurrent, protection against short circuits, protection against undervoltage, and cell charging balance. The BMS is composed of three independent modules, each of which is responsible for controlling one of three distinct cells. After that, these three modules are deftly combined with transistors and various other passive components to produce a full BMS that can deliver current of up to 20 amps and protects each individual cell.

The battery management system as seen in Figure 14 below has two integrated circuits (ICs), designated as DW01 and BB3A. In some variations of the BMS, the integrated circuits (ICs) used might be the same ones or ones that are very similar but manufactured by a different company. Nevertheless, the pinouts and functionalities of each IC are identical to one another. A single integrated circuit, designated DW01-A, is responsible for providing protection against overvoltage, undervoltage, overcurrent, and short circuits. On the other hand, the preservation of the cell's charge balance is the responsibility of BB3A IC.

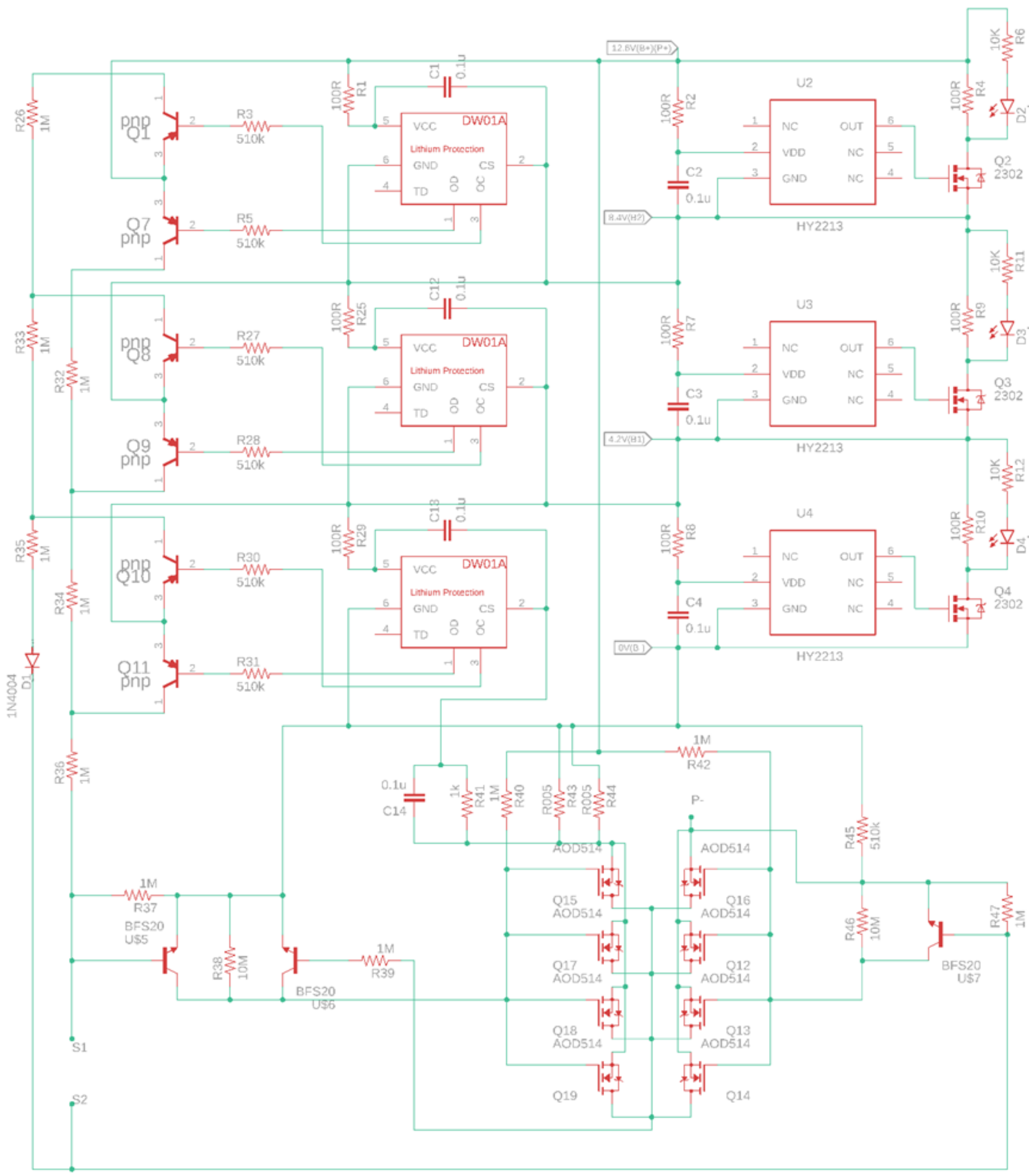

Figure 14: Full BMS circuit schematic

The DW01-A is a type of integrated circuit designed specifically for single-cell Li-ion or polymer batteries. It is in charge of the BMS's protection features, with the exception of the one that governs the charging balance. Each cell has a DW01-A attached to it so that it can keep track of how well the cell is doing in terms of its overall safety and comes packaged in a 6-pin configuration. Referring to the datasheet for the IC, we can view the functional diagram as well as any other information that may be relevant. There is an internal voltage divider circuit within the IC that serves the purpose of determining whether the cell is operating at an undervoltage or an overvoltage. Also, the comparators are able to detect both short circuits and overcurrent by comparing the voltage at the input of the CS pin and the voltage at the VSS pin.**(23)**

The HY2213 BB3A, 1 cell Li-ion battery charger balance IC is the central component of the cell balancer circuit. This integrated circuit can actively balance a cell by monitoring the charge level within the cell and adjusting the voltage accordingly. In addition to that, it possesses a delay circuit as well as a voltage detection circuit that are accurate. The HY2213 series was designed to function with a solitary lithium-ion cell, but it is also compatible with multiple-cell battery packs that contain individual lithium-ion cells in parallel within their casings. A voltage divider circuit that is connected to the input VSS and the input VDD of the IC is depicted in the functional block diagram of the circuit. Those two pins provide the overcharge detection comparator with the difference voltage of the monitored cell. Looking at the data sheet, the voltage result from the comparator is the output. The output then can be used for controlling the enhancement MOSFET in case of a cell overcharge detection, and it is possible to make use of either a P-type or an N-type MOSFET.**(24)**

The eight MOSFETs that are connected are actually split up into two groups, which are organized into two sets of four MOSFETs each. Overcurrent and overdischarge protection are the respective responsibilities of the first and second sets of circuit breakers in this circuit. Because the health of each cell degrades at a different rate, it is imperative that the protection against overcurrent or over discharge is activated by any cell in the circuit. Because the source pins and gates of the parallel MOSFETs have been coupled together, a trigger can now be applied to all of them at the same time. The drain pins of each MOSFET are connected to one another, so the circuit will only function appropriately when all eight of the MOSFETs are set to the "on" state. If any of the MOSFETs are in the "off" state, then no current will flow through the circuit, and as a result, the battery pack will not be able to power the output or charge the cells.

Taking control over the MOSFETs is achieved by updating the DW01 IC's overcharge and over discharge pins respectively. The first set of MOSFETs source is coupled to the ground. Then, the current sense pin of DW01 is also coupled to the source of those MOSFETs. As a consequence of this, the DW01 IC will activate the (OC) pin whenever it detects either an overcurrent or a short circuit and this in turn will activate the transistor pair. After that, the transistor pair will send a signal to the gate terminal, which will ultimately result in the MOSFETs being shut off. The second group of MOSFETs in the circuit that prevents the battery pack from being overcharged have their gates connected to the positive terminal of the battery pack. The (OD) pin connected to the transistor will become active once the internal voltage divider circuit of the DW01 IC determines that the battery has been overcharged.**(23)**

Due to the shortage of some of the ICs in this BMS design, we decided to buy a ready to use one that this design was based on which is the Daier (HX-3S-FL25).

5.2 Phone charging daughter boards design

Quick Charge 3.0 was released in 2015 with the goal of maintaining a greater power delivery for charging while also ensuring that power transfer was as efficient as it possibly could be. Because they are universal, all devices that support fast charging will be able to use the most recent evolutions of Quick Charge when they become available. It is important to make a note of this fact at this point because the QC and USB-PD standards are very similar in a number of ways, the most notable of which is the utilization of power negotiation protocols and variable voltage selection. However, a sizable portion of the market, including portable electronic devices such as laptops, has adopted USB-PD, which functions more as a benchmark for the industry. Quick Charge, on the other hand, is exclusive to mobile devices like phones and tablets that are run by Qualcomm's System on a Chip (SoC) and can only be used with those devices.

The IP6505T IC chip

When trying to charge a smartphone that uses Quick Charge (QC), it can be difficult to do so using a regular USB car charger because it only provides 5VDC. The USB fast charging module is a boon in this respect because it is based on the IP6505T chip, which is a rectified and synchronous 24W buck converter that can be used for fast charge adapters, car chargers, and smart power strips. Therefore, the USB fast charging module is an absolute requirement. The IP6505T chip supports multiple quick charging standards and has the capability to automatically adjust the output voltage and current levels to conform to the standard for quick charging. The IP6505T chip also has a wide variety of protection features built into it. Some of these features include protection against short circuits, overcurrent, Overvoltage, and undervoltage. Some of the supported device charging standards are Qualcomm QC2.0/QC3.0, DCP (Apple, Samsung and BC1.2), Spreadtrum SFCP, Samsung AFC (Adaptive Fast Charging), MTK PE1.1/2.0, and Huawei FCP & SCP (Fast Charge & Super Charge).**(25)**

Recalling that the IP6505T integrated circuit has a synchronized switch buck regulator built into it. The output voltage can be anywhere between 3V and 12V, while the input voltage can be anywhere between 4.5V and 32V. It is able to recognize a fast charge standard as soon as it is accessed, and it can automatically adjust the output voltage to match the standard. The IC supports soft start time to avoid the inrush current during start up, which has been calibrated to a value of 10 milliseconds. The power conversion efficiency is at its highest, which is about 97% when the input voltage is 12V and the output voltage is 9V, but it depends on the output current as well as the output voltage, as it can drop below 96% if the output voltage downgraded to 5V.**(25)**

USB Charging Circuit board design

When referring to the datasheet of the IC (IP6505T), we can find the IC's pins layout and their job description, which was helpful in terms of designing the USB charging circuits. The IC pins are shown in Table 20 below.

Table 20: IP6505T IC pins and their description

Pin name	Description
GND (1)	Connect to Ground
SW (2)	DC-DC switch node, connect this pin to the inductor
BST (3)	Boost strap circuit pin, place capacitor close to the BST pin and LX pin, providing drive voltage for the gate of the upper MOSFET
NC (4)	Float pin
DM (5)	Connect to USB DM data line
DP(6)	Connect to USB DP data line
Vin (7)	Input voltage, place filter capacitor near this pin (22µF recommended for EMI input, otherwise 1µF)
Vout (8)	Output voltage feedback
GND (9)	Power and heat dissipation ground

After analyzing the IC pins' description and the IC characteristics and looking at the typical application circuit schematic that is provided in the datasheet, the full circuit should contain some main components like a solid bypass capacitor between the input voltage and ground to increase the efficiency, and it should be around 100µF. Also needed another bypass ceramic capacitor between the input voltage and ground to reduce the incoming EMI ripples and it should be around 22µF. At the output, we needed an inductor between the output and the SW pins to resist the sudden change in output current and it should be around 22µH, with a 0.1µF capacitor between the BST pin and SW pins. Additionally, we would need two output bypass capacitors with 220µF & 10µF. The rest of the circuit is shown in Figure 15 below.**(25)**

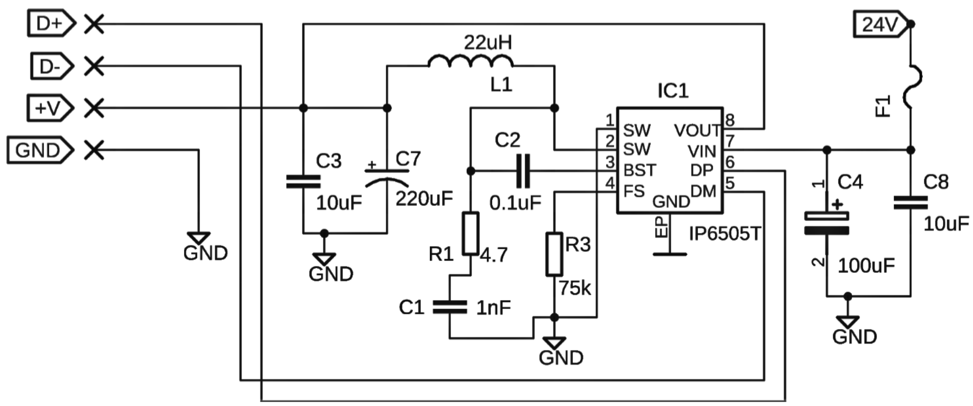

*Figure 15: USB charging circuit board, schematics credit to Injoinic Technology **(25)***

5.3 Locker control system hardware design

This section will detail the hardware design for all non-power components and the pin configurations for each one. Aside from the power sub-system, the rest of our hardware design just consists of connecting to the signal pins of our peripherals to allow for our MCU to communicate with and control them. Our peripherals use several different types of communication protocols such as UART, SPI, PWM or act as a simple switch. Some also have other requirements for certain pins to be set high or low for extra functionality that we would not need to control with our MCU and could save some GPIO pins by setting these pins with PCB traces. Figure 16 shows a high level overview of our hardware design and the types of communication we need to connect to the MCU. Note that the connections via the GPIO expander are ignored for simplification.

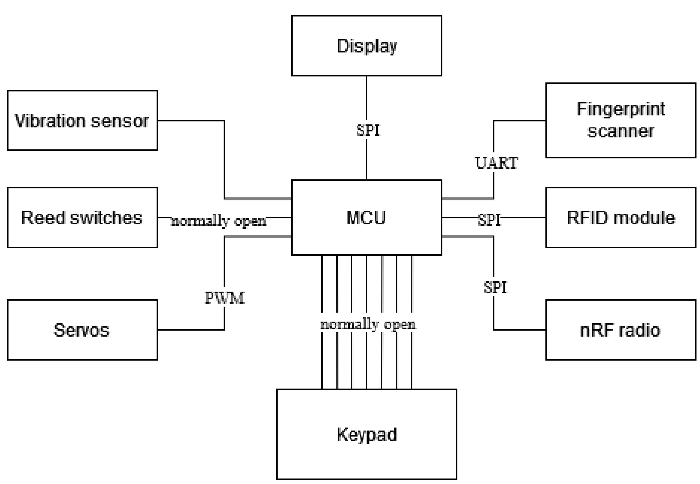

Figure 16: Locker hardware block diagram

5.3.1 Communication protocols

This section will give a brief overview of the communication protocols used in our project. Since we have multiple devices that do not use bus based protocols, we needed to consider how to connect each type of device, and needed to know which devices can share pins whenever possible. While our selected MCU has plenty of GPIO pins, the evaluation board we have may have slightly fewer pins than we need and we are basing our design on supported, arduino compatible boards which have a similar amount of pins.

UART

Universal Asynchronous Receiver/Transmitter (or UART) is a simple serial protocol for sending data short distances. Other than power or ground, it only requires one wire to send data between devices. However, UART over one wire is a simplex protocol, and needs a second, return wire to receive data and make it a dual-simplex protocol. Just about all modern processors support UART, and most embedded development platforms have dedicated pins for it.

UART only works directly between two devices, so each instance requires its own set of transmit (tx) and receive (rx) pins. Fortunately, our project only has one device that uses UART, so we took advantage of the tx and rx pins already assigned to UART and left the more general purpose GPIO pins for our other devices.

SPI

Serial Peripheral Interface (or SPI) is a widely used serial bus protocol for embedded devices. It is much faster than UART and one of its main features is that the data transfer pins can be shared between multiple devices to create a data bus. SPI differentiates between devices by using separate chip select, chip enable, or slave select pins. Only the device whose chip select pin is active will listen and respond on the data lines, so the master device must ensure that only one device is selected at a time.

SPI has three data lines which are shared between devices; a common clock, the Master Out, Slave In (or MOSI) line for sending data to devices and the Master In, Slave Out (or MISO) line. Along with the required Chip Select line, SPI is often referred to as a 4-Wire protocol.

Three of the components we selected use SPI, though we did not use the nRF24 radio module since we got the on-chip BLE radio to work. So for SPI communication we needed between 5 and 6 pins available, however, the display has extra non-SPI standard pins, which will be discussed in section 5.3.3.

PWM

Pulse Width Modulation is a technique of varying the output power of a device using a constant voltage by changing the percentage of time the device is on. It is commonly used for controlling the brightness of lights and signaling speed settings for motors. The application we are interested in is using PWM for motor control, however our application requires using PWM in a not entirely typical way.

Typically, PWM is used directly as the power input signal for the motor, where an 100% duty cycle means full speed and reducing the duty cycle means the motor controller reduces the amount of time the motor is powered, and its speed lowers. Our application is using PWM to control the angle of our stepper motors, which we will go into more details about in section 5.3.5.

For our application of PWM, we only needed one signal wire to each of our stepper motors, meaning using 3 PWM pins of our MCU to control our motors.

NO/NC and binary pins

The last kind of ‘communication’ protocol used in our project is simple normally open or closed type signals. We control, or more accurately read, these devices by simply checking their voltage state since they are all binary output devices. The devices in our project that have this property are the reed switches, our keypad and the vibration sensors. More details of each are in their respective sections below.

5.3.2 BLE radio or nRF24L01 module

In order to wirelessly communicate with our remote, we needed to communicate with one of the wireless radios we selected. If we chose to use BLE, our hardware design for the wireless radio would be very simple since the MCU we selected has its own wireless radio hardware and comes on a module with its own antenna. This simplicity is part of why we chose this MCU, but we were also apprehensive on the amount of information we could find on using the chip alone instead of as a pre-existing, arduino compatible board. This reason is why we chose to select a second wireless standard that we knew would have much more support.

Our alternative wireless hardware was using nRF24L01 modules, which communicate over SPI. This module has the 4 standard SPI pins and a separate chip enable pin does not need to be connected to our MCU if we need to save pins. If we did not connect the chip enable pin, we may have lost the ability to control its power, however, so we thought to include it as an MCU connection unless we really need to save pins. This would result in using 3 extra pins over the 3 shared for SPI.

5.3.3 Connections to the display

The display we selected was connected to our MCU using SPI communication. The display in this configuration needed to use 7 pins with 4 directly connected to the MCU. 2 of the remaining pins were connected to the power supply and ground and the final pin was used as a reset pin. To properly power the device optimally, it needed around 3.3 volts which we could easily supply. Apart from the 4 SPI pins, this display also had a data / command pin for specifying to the display what kind of data is being sent and a reset pin. This meant that the display needed 3 additional pins other than the shared ones for SPI.

5.3.4 The keypad

The keypad we chose was very easy to read and connect to our MCU. It had four rows and three columns, two of which get connected when pressing any of the twelve buttons. The row and column that get shorted together are those that the button being pressed is on, so, to connect our keypad we only needed the MCU to have access to the seven pins corresponding to the rows and columns. This means that the keypad took up the most GPIO pins of any other peripheral, but was the simplest to interface with in software. We chose to connect the keypad to the GPIO expander in order to save the pins on the MCU for other peripheral devices.

5.3.5 Servo motors

Our servo motors are one of the more unique components in our hardware design as they use a unique communication method. To signal them to turn to a specific angle, we need to send a PWM signal of 50 Hz with a duty cycle corresponding to that angle. Figure 17 shows how the angles 0 and 180 degrees are signaled to the servo. The exact millisecond values for duty cycle are in the datasheet for the particular servo, but the one we ordered did not come with a detailed data sheet. Using the low and high values for the PWM on time, the servo determines its pointing angle by measuring the duty cycle of the PWM signal as a 16-bit value. A value of 0, corresponding to a pointing angle of 0 degrees, is measured at the particular servos minimum on time (between 0.5 and 1ms), as specified in its datasheet. The high value, corresponding to a pointing angle of 180 degrees, is its max on time specification (between 2 and 2.5ms). Intermediate values, and their corresponding angles vary linearly, so to point to 90 degrees, we signal with a PWM wave with a duty cycle, or on time, half-way between the min and max specifications.
Because the servos only have 3 pins, power, ground and PWM, we only needed to use 1 PWM capable GPIO pin per servo, meaning 3 total pins.

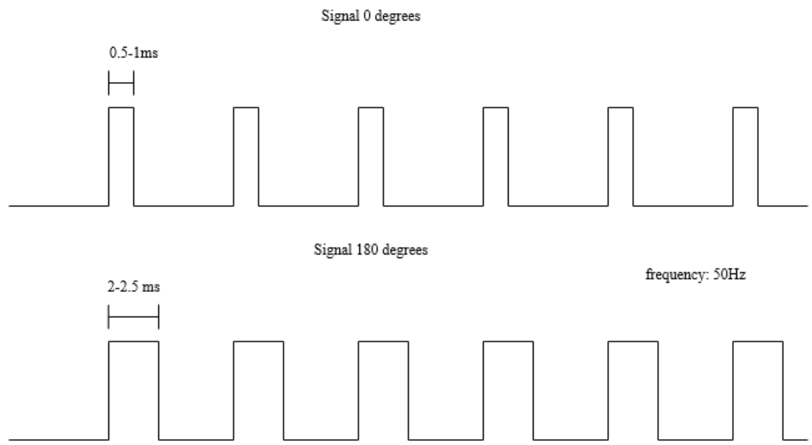

Figure 17: Diagram depicting relationship between PWM duty cycle and servo angle

5.3.6 Reed switches

Since the method we chose to detect if the compartments are open or not was so simple, we did not have to worry too much about their part in the hardware design. Our reed switches are normally open, so all we needed to do is wire one side to high voltage and the other to a GPIO pin with a pull down resistor. We included headers on our PCB to allow us to easily plug in wires to be run to each of the compartment doors.

5.3.7 RFID module

Even though our RFID module supports multiple communication standards (SPI, I2C and UART), we chose to communicate over our pre-existing SPI bus to save pins. The RFID module had additional interrupt and reset pins, though we did not need to use these in our design and set the reset pin to its inactive state by pulling it high with our 3.3V power rail. Since the RFID module is in the front panel, we connected it using headers on our main PCB and wires to the front panel. One design consideration we needed to keep in mind was that the RFID module required the tags to be close to it to function reliably, so we placed it close to the surface of the font panel.

5.3.8 Vibration sensor

Since we chose the vibration sensor type that gives a binary output, we only needed to connect its output pin to any GPIO pin to detect alarm events. The way the detection circuits of this module are setup outputs a low signal to indicate an alarm event (making it an active low device). We only detect vibrations from phone calls in the small smart-phone compartment so we only need one sensor and 1 pin to the MCU.

5.3.9 Fingerprint sensor

The fingerprint sensor selected for our design was connected to the MCU using UART communication protocol. These 2 pins were connected to the MCU while 2 other pins were connected to the power supply and ground. The optimal voltage needed to power the device was 3.3 V. As for the pins connected to the MCU, one was connected to the rx or receive data pin and the other was connected to the tx or transmit data pin.

The fingerprint sensor itself had a diameter of 20.8mm so we 3D printed a mold to hold the sensor and attached this mold to the front panel of the locker. We used an adhesive to help it stay in place and prevent it from potentially getting dislodged if the user places too much force on it.

5.3.10 Control system schematic

Figure 18 below shows our schematic design for the Study Buddy locker.

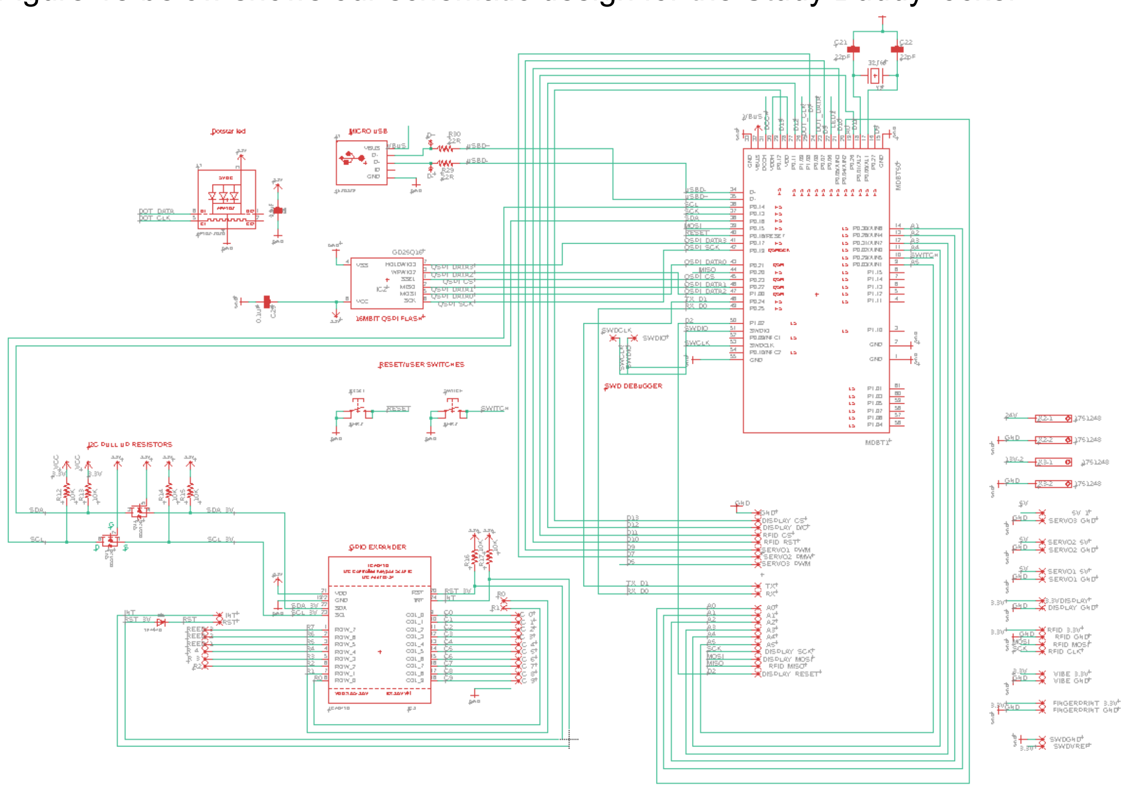

Figure 18: Main schematic for Locker control system

5.4 Remote hardware design

The Study Buddy Remote is a simple companion device that serves to extend some functionality of the Study Buddy to be used wirelessly. Its design was simple, only requiring an MCU, wireless radio, display, 4 user interface buttons and batteries but was very flexible in its use based on the software implementation. It is fairly compact and consists of a single PCB for simplicity and also provides support for the button interface.

5.4.1 BLE or nRF24 radio

Similarly to the wireless hardware described for the Locker in section 5.3.2, the remote has built in BLE hardware. Likewise, if we had used the nRF24 module, we would have used SPI to communicate with it. Since we have so few devices in the remote, we would have plenty of pins available and would use the 4 base SPI pins and the 5th chip enable pin. Space in the remote is at a premium, however, so we tried to solder the nRF24 module as close to, or directly on our remote PCB.

5.4.2 The Display

The small OLED display we chose for the remote used I2C to communicate with our MCU. I2C, or 2-Wire, as it’s commonly referred to as, is a bus communication standard for embedded devices. All devices on the I2C bus share the same data and clock wires and communicate to each other using their addresses. Since the display was the only other device on the remote I2C bus, we did not have to worry about devices talking over each other, but we still needed to know the display's I2C address to communicate with it. Since we were using I2C, the display needed 2 GPIO pins. We did not have any more I2C devices, so we did not need any additional pins to communicate with them over I2C.

5.4.3 Buttons

For the button interface of our remote, we used simple momentary buttons. These buttons have 3D printed buttons above them to protrude through the body of the remote and mount directly to our remote PCB using through holes.

5.4.4 Power

To power our remote, we use 2 AA batteries and step the 3V up to 3.3V using a boost converter design from WEBENCH. AA batteries have capacities around 2,500 mAhs. With our cells in series, this is the amount of power we have to use to reach a 7 day battery life. To meet this goal, the remote had to draw less than 14.8mA continuously on average. By using power saving modes and only waking up periodically, we were easily able to meet this power target. Using WEBENCH to help design the boost converter we arrived at the design below Figure 19, which has a very small footprint and a very high efficiency.

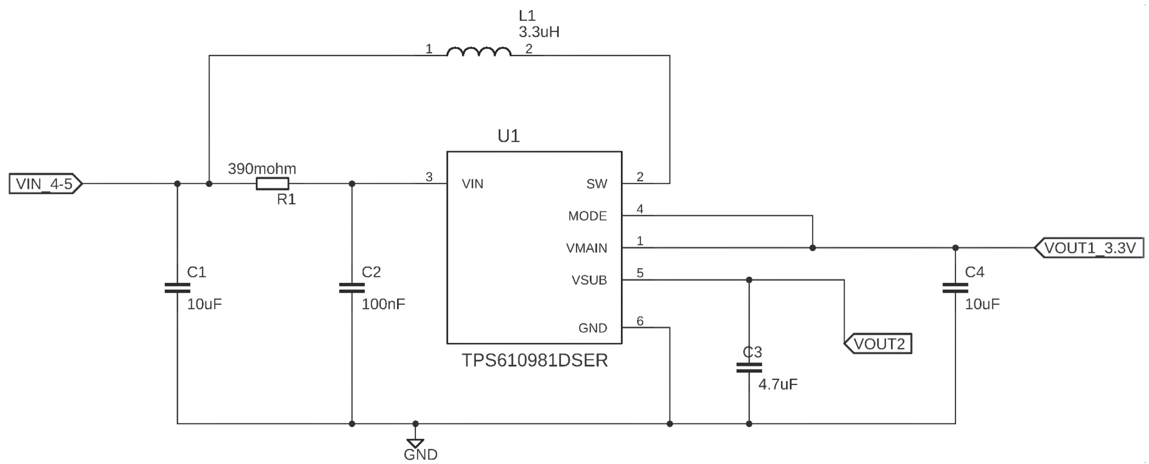

Figure 19: 3.3V Boost Converter circuit
*Schematic credit to webench**(27)***

5.4.5 Remote schematic

Figure 20 shows our schematic design for theStudy Buddy remote. With the chosen MCU we were free to use any pins to connect the display in I2C configuration. Pins for the push buttons were selected to shorten the distance of traces for the PCB of the remote to keep it small.

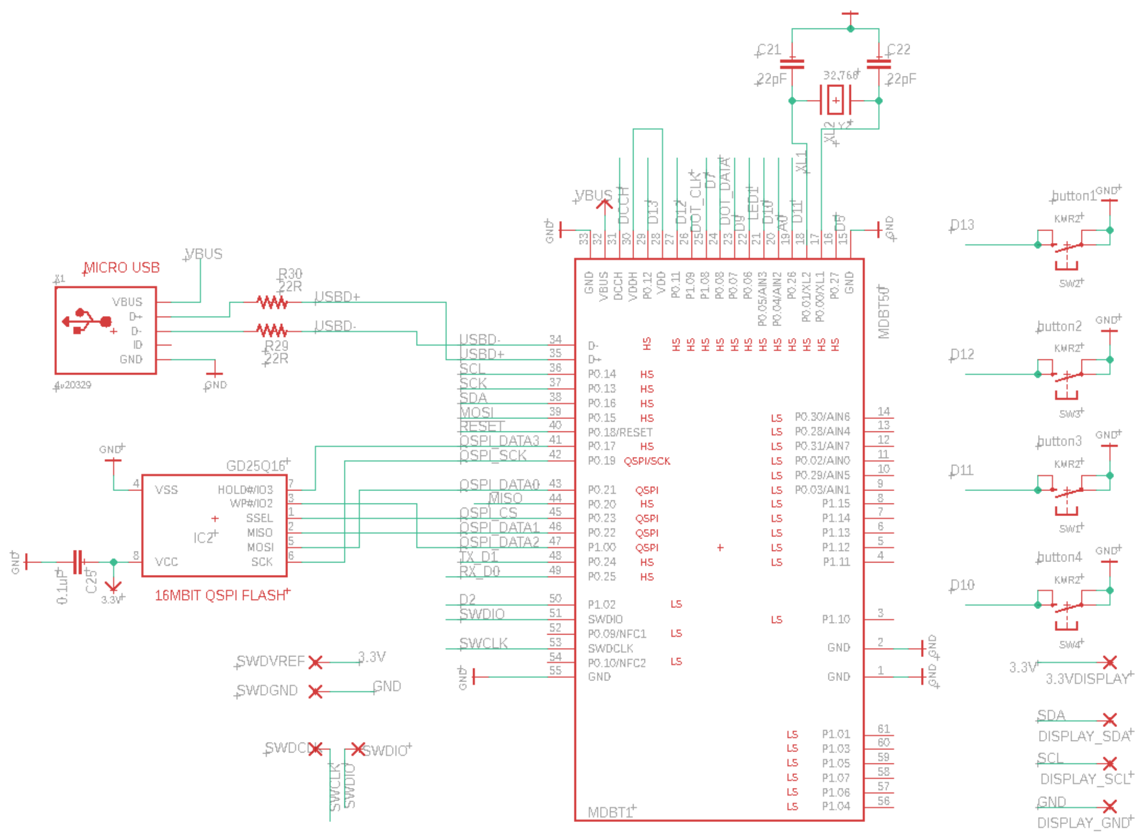

Figure 20: Main schematic for Remote

5.5 PCB Design

When starting to design our PCB or printed circuit board, we needed to first look into the available design software we could use as well as the design philosophy and considerations we should have.

5.5.1 PCB Design Software

The first software for consideration was Autodesk Eagle or Easily Applicable Graphical Layout Editor. Autodesk Eagle has the advantage of being available for free with few restrictions as well as some familiarity for the group as we all have some experience with the software from a previous course. Eagle allows the user to implement any number of parts into the program for use in designs but finding the proper parts online can be difficult for certain parts. Another advantage to eagle compared to other software is that there exists many tutorials and guides to the various functions of the program online. Compared to other design software, Eagle's library, while very modifiable, is more difficult to use and can be confusing to navigate. Eagle's rules of design checking is also less reliable compared to other software.

Another PCB design software up for consideration was Altium Designer as it was one of the more recommended software to use. The product is available for a free trial period but once that is over it needs to be paid at a rather high price for our budget. This means we need to finish our design within the free period. The library and part selection available for Designer is very extensive compared to other programs and there exists many resources for assistance on design for users of the program online.

The final PCB design software up for consideration was EasyEDA as it was more beginner friendly. The program is often recommended for those with little to no circuit or PCB design knowledge and is often used by students. There exists many resources within the community of users to make use of such as tutorials for the various tools and implementing designs. The program requires no installation and can be used online on various devices. This however does mean that an internet connection is required to work on our design. The program has a library of components to use but is smaller compared to other available software.

With all 3 design software considered, we believed that the best program to use would be Autodesk Eagle simply due to the familiarity of the program and resources already available to us. With the program selected, we next needed to understand how to properly design a PCB.

5.5.2 PCB best practices

Before designing our PCB, we need to make sure that our design will work as we intend and is capable of being manufactured as desired. To do this we should keep in mind a few basic practices. The first practice is to determine our design rules before starting our layout. We should keep in mind the capabilities of our chosen manufacturer as we may need to redesign our design if we do not properly meet their specifications and will not be able to make our design. We should make sure that our design layout limits are more conservative so that our design will be able to be manufactured and still remain reliable.

The next practice is to pay careful attention to the placement of our components and fine tune them as well as we can. Proper placement of our components will make routing between them much easier and improve our electric characteristics. It is important to consider which components are needed at certain locations and place them first as moving them may hurt other aspects of our project. Components with many connections to other components should be placed closer to the center of the design as to more easily connect and shorten traces. It is also important to avoid cross nets as a layer transition will be necessary, making the design more complex and more costly. The orientation of the components should also be kept in mind as to make routing easier and avoid potential problems with connections to devices outside the PCB.

Next, it's important to consider how to route power and ground traces so that our signals remain clean. Our power and ground should be on separate layers of each other though that depends on how many layers our design will have. If we can afford to have more layers, it is recommended to use a ground layer rather than ground traces. If we add a ground layer to our design, it is important to note that our bluetooth module requires there to not be a ground pad next to the module as specified in its datasheet. As discussed in the previous practice, this means that we will need to place the module close to the section of the PCB that contains the no ground pad. We should also make sure that we use a common power rail for components that need direct connections to power. We also need to make sure that the width of these power traces are wide enough as too thin traces may cause heating issues if the current flowing through them is too strong. Trace width is also important to consider as some manufactures cannot reliably manufacture traces that are too thin. We should keep in mind the width of our traces when choosing a manufacturer and make sure that their widths are larger than the minimum set by the manufacturer to make the process easier.

Next, it's important to keep components separate from each other when they can potentially interfere with each other. Some components can produce high temperatures so it is important to keep these components spread out throughout the PCB so that the heat can be properly dissipated. The use of heat sinks may also help combat this issue though too many can make their design more difficult to manufacture. We should keep in mind the thermal resistance of our components found in their datasheets and attempt to calculate their potential temperatures in ideal operations. If a component is in danger of reaching high temperatures, we should be prepared to add heat sinks close to the component. Failure to properly account for potential heating issues may cause permanent damage to the overheated component and ruin the overall design. We must take extra careful consideration as our PCB board will be housed in a small enclosed space so heat will not easily dissipate. Besides heat, some components can interfere with others by generating noise that can interfere with more sensitive ones. To combat this, many designs group their analog and digital components separate from each other.

To finish up the design, we should pay careful attention to our design to make sure there are no errors that are undesirable as any uncaught errors may lead to our PCB failing to be manufactured or not behaving as expected. We should make use of Eagle's Design Rules check and Electrical Rules Check to make sure that our design is within the proper constraints and correct any errors caught by both. Eagle’s Rules checks should be good enough to catch most errors though we should check and modify them if some of the checks do not meet our manufactures requirements, especially for trace widths.

5.5.3 PCB design philosophy

With the previous practices in mind, we need to find a balance between what will be most optimal for our electrical characteristics and the manufacturability of our design. In order to reach desired values for trace resistances and power consumption for the overall design, we will need to make use of larger trace thicknesses and more conductive inks within the PCB. Using these would make the manufacturing process much more difficult and more expensive for the manufacturer.

5.5.4 Locker Main PCB Design

Figure 21 shows our final design for the locker’s main PCB which includes the voltage regulators needed in our design. It also includes the 3 usb charging circuits. Finally, it includes the MCU and GPIO expander and the headers for connecting the peripheral devices connected to the board.

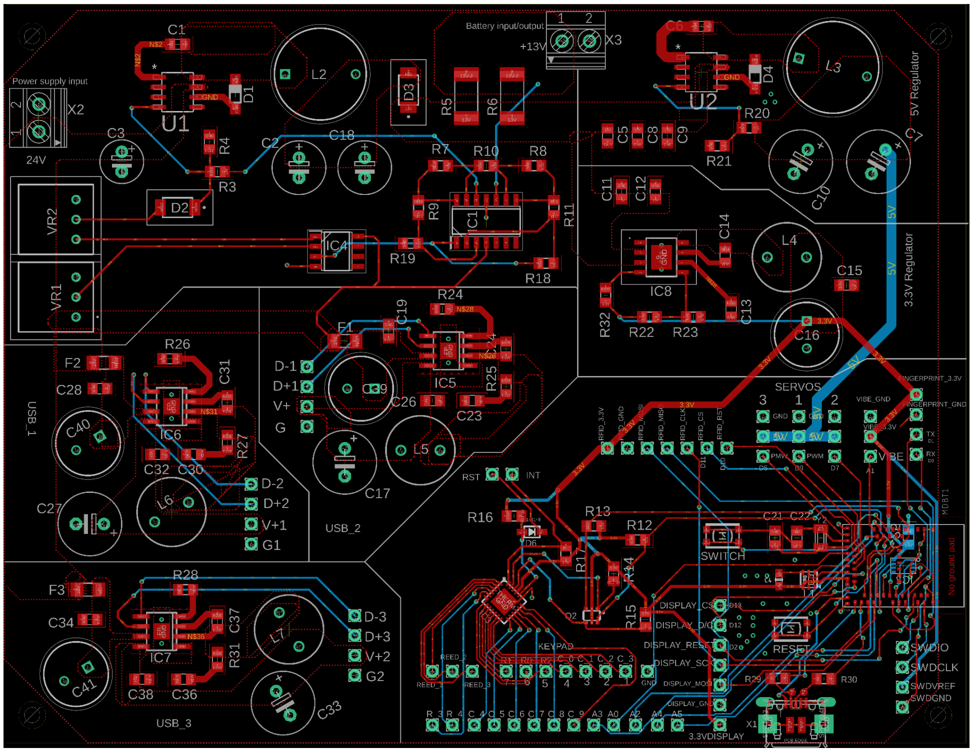

Figure 21: Locker Main PCB Design

Figure 22 below shows the final constructed locker PCB that was used in the final functional Study Buddy design.

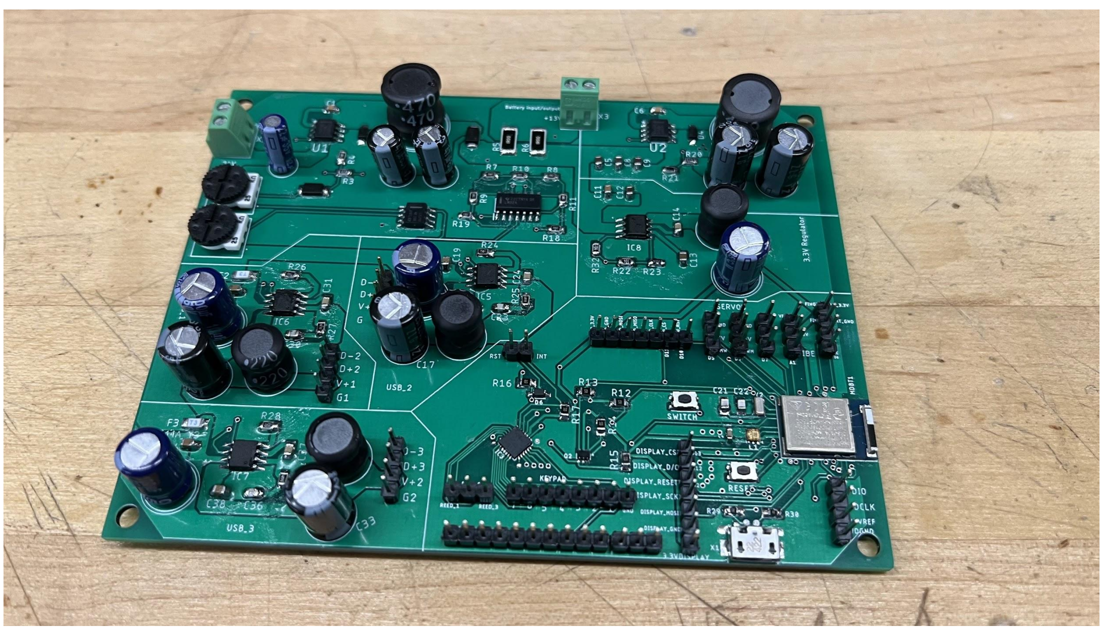

Figure 22: Final constructed locker PCB

5.5.5 Remote Main PCB Design

Figure 23 shows our final design for the remotes's main PCB which includes connections for the battery pack as well as the needed boost converter. It also includes the 4 push buttons and the MCU.

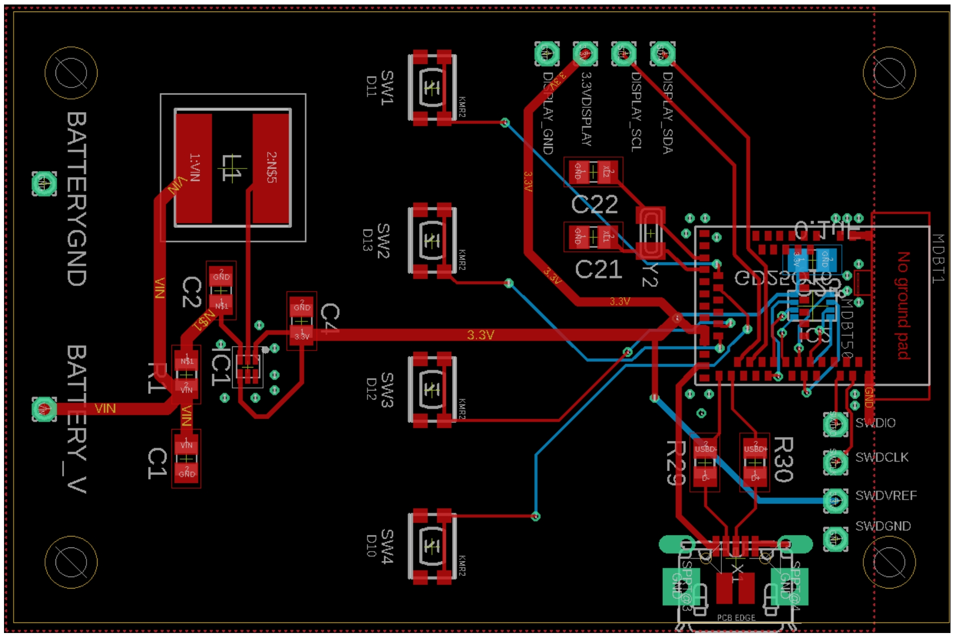

Figure 23: Remote Main PCB

Figure 24 below shows the final constructed version of our remote PCB that was used in the Study Buddy remote.

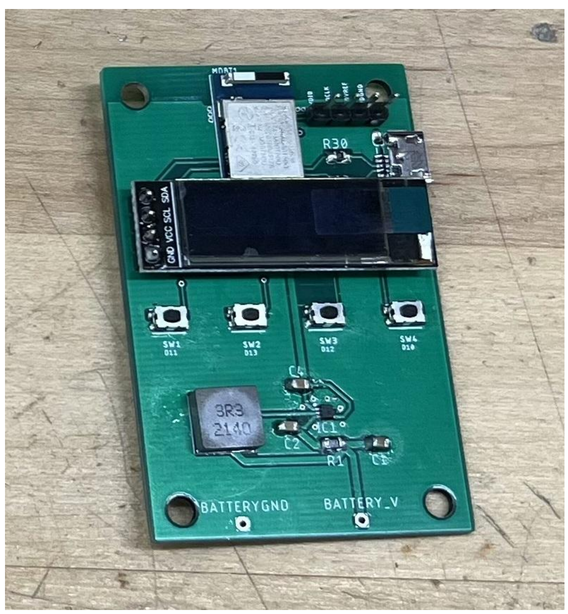

Figure 24: Final constructed remote PCB

5.6 Software design

The software of the Study Buddy's locker serves a few primary purposes: controlling and monitoring the lock state of each door, interpreting the signals from each connected input device, maintaining the built in timer, deciding what to display on the screen at any given moment, and facilitating communication with the Study Buddy remote. The remote's software is considerably simpler, mainly serving to control the display, interpret user input from buttons, and control communication with the locker via Bluetooth Low Energy or an nRF24 radio module.

Because we used the Arduino IDE to program the Study Buddy's software, we had access to several Arduino libraries that serve to abstract away some complicated code, such as the specifics of the I2C and SPI communication protocols. This, in general, allowed for smoother programming of the MCU's interactions with the Study Buddy's various input and output devices. In addition, specific libraries exist for several of our components, which further improved the overall software design process

The below sections describe in more detail the general design philosophy of the Study Buddy's software and discuss the specifics of the software functionality and available libraries for each of the relevant components.

5.6.1 User Input

The user input methods of the Study Buddy locker are a keypad, an RFID reader, and a fingerprint sensor. With simple usage of the device, the keypad corresponds to a single saved numeric password, the RFID reader corresponds to a single saved identification code stored on an RFID tag, and the fingerprint sensor corresponds to a single saved ID that the sensor generates from a recognized print. In reality, multiple of each of these saved identifiers are stored in the MCU's memory, and the software is able to identify which specific input ID corresponds to each of the three locker compartments.

The user is able to select which unlocking methods they wish to enable when configuring a compartment before locking it. Because each of the locker's three compartments are controlled individually, the software will store three values for each compartment, one for each unlocking input method.

Keypad Software and Decoding

As mentioned in section 5.3.4, the keypad on the Study Buddy locker is multiplexed, covering all twelve buttons with pins in four rows and three columns, having seven pins in total. In order to correctly interpret user input from the keypad, our software decodes the signals from these pins fast enough to recognize when any button is pressed and identify it. To do this, the software sets all of the column pins high in order to detect when a button is pressed. As soon as a button has been pressed, one of the rows will read high, and an interrupt will trigger, calling a function that quickly checks each column in sequence to identify which button was pressed.

Using the above method, we can detect a password input from a user via the keypad. When selecting passcode as an unlock condition before locking a container, the user is prompted to enter a four number passcode and press the hash symbol to submit. Upon doing so, their passcode will be stored as a variable in memory. The software will not attempt to hash or encrypt the password in any way, as being able to access the MCU's memory implies that someone already has access to the locker's internals, making entry by stealing the passcode unnecessary. When attempting to unlock a locked compartment with the keypad unlock method enabled, one must enter a code into the keypad which will be compared to the one stored in memory. If correct, the passcode entry check will be passed and the locker will move on to other enabled unlock methods, if there are any. If the entered code is incorrect, the user will be prompted to either try again or to return to the locker's main menu.

RFID Software and Libraries

The RC522 RFID reader selected for the Study Buddy locker has an easily accessed Arduino library "MFRC522.h" that contains several useful functions that make reading from and writing to compatible RFID tags and cards significantly easier. The "IsNewCardPresent()" function is able to detect when a new card is held near the reader, the "ReadCardSerial()" function selects the card and accesses information stored on it, and the "uid.uidByte[]" function reads the card's unique identification number (UID).

Upon selecting the RFID option as an unlocking method before locking a compartment, the user will be prompted to scan a valid card or tag across the reader. Upon doing so, the IsNewCardPresent() function will indicate to the software that a card is being scanned, and the card's UID will be accessed via ReadCardSerial() and stored in memory via uid.uidByte[]. When attempting to unlock the compartment, the user will be prompted to scan the correct RFID tag or card to move on. Using the same functions as above, the scanned card's UID will be compared to the one stored in memory. If the UID matches, the card is correct, and the unlocking protocol will move on to the next step. If the UID does not match, the user will be prompted to ensure they scan the correct card or to return to the locker's main menu.

Fingerprint Sensor Software and Libraries

The selected fingerprint sensor for the Study Buddy locker is the Adafruit 4750 capacitive fingerprint sensor. Adafruit provides the libraries "Adafruit_Fingerprint.h" and "Adafruit_Fingerprint.cpp" for their fingerprint sensor products that include functions for enrolling new fingerprints and searching for a currently scanned fingerprint in memory. Enrolling a fingerprint on this sensor means assigning an ID number to the new fingerprint being scanned and storing it in memory. Searching a fingerprint here means comparing a currently scanned fingerprint to those stored in memory and identifying which ID, if any, matches the current print.

For the Study Buddy, we decided to not save user fingerprints between uses of the locking compartments. This is in order to ensure that users know for certain what fingerprint will unlock a compartment in any given circumstance, and to avoid a user accidentally locking a compartment with another person's print as an unlock condition.

To enroll a fingerprint in the Study Buddy, the user will have to select the fingerprint sensor as an option before locking a compartment, which will begin the enrollment process in the software. The user will be instructed to place their finger on the sensor and hold until scanned, then to remove their finger and place it back again. This double scan is inherent to how the fingerprint sensor works, as it is designed to ensure it recognizes the same print every time, and takes multiple samples to compare against. Once the user has completed the fingerprint enrollment, the locker configuration process will continue. If the finger scanned is one that is already stored somewhere in the sensor, that ID will be used and the process will complete with one scan. The print they enrolled will be saved under the ID corresponding to the locker they are configuring, meaning that only the first three fingerprint IDs will be used by the Study Buddy. When attempting to unlock the locked compartment, the user will be prompted to place their finger on the sensor once again, this time only once if the scan is successful. If the scanned print matches the print stored under the corresponding compartment's ID, the unlocking process will continue. If the scan does not match, they will be prompted to scan again or return to the locker's main menu.

5.6.2 Timer

In addition to the three unlocking methods that take direct user input, the Study Buddy gives the user the option of enabling a timed unlock based on a time set by the user. To do this, we use the nRF52840's TIMER peripheral and its corresponding "timer.h" library to set a timer of a length taken by user input in hours and minutes that will trigger an interrupt upon reaching 0. This interrupt calls the unlock function for the timer's corresponding compartment and unlocks the door.

The timer’s current progress isbe visible on the specific compartment’s menu screen on both the locker and the remote, and will count down continuously even if other compartments are configured or timers are started during the countdown. This is possible due to the presence of multiple timer instances in the nRF52840’s TIMER peripheral that can be configured individually and that can trigger different interrupts.

5.6.3 Detecting ringing phones for emergency call detection

Our project's stretch goal was to be able to unlock the locker early in an emergency situation. One of the methods we chose to do this is by detecting when a cell phone is receiving repeat phone calls by sensing the vibrations of a ringing phone. We chose a vibration sensor that outputs a low signal when it detects vibrations and has an adjustable sensitivity. This sensor is known to give false alarms and can detect slight changes in motion such as being moved or bumped. In order to achieve our required greater than 65% reliability we had to balance the vibration sensor’s sensitivity with the method we use to discriminate between non-phone events and the vibrations from a ringing phone receiving a phone call.

Our general approach to this problem was to keep track of how many detections there have been in the last x seconds. The idea behind this approach is that a ringing phone starts and stops vibrating many times, which we should be able to detect as multiple vibration events from our sensor. At any time, the software knows how many vibration sensor toggles have occurred in the last minute, and by setting a threshold corresponding to about 1.5 phone calls (found to be about 2500), we give the software an exact number to check for, at which point the compartment will open.

By discriminating between random vibrations and a phone call using the amount and regularity of the vibrations, we are able to reliably detect if a phone call is occurring. If we detect two likely phone calls in succession we can be reasonably confident that someone is trying to reach the owner of the phone and trigger an emergency unlock.

5.6.4 Locking and Unlocking

Because the locks on the compartment doors of the Study Buddy are controlled by servo motors, the device’s software needs to be able to rotate the motor to set positions defined as the “locked” and “unlocked” states. To avoid manually working with the PWM signals of the motors, we used Arduino’s “Servo.h” library which abstracts the motor control signals to simple provided functions. Using the Servo.write() function, we can directly choose the angle (from 0 to 180 degrees) that the motor should move to, meaning that we can easily test a motor at different angles to find what angles will correspond to “locked” and “unlocked”.

Before locking a compartment, the unlock conditions must be set by the user and the corresponding door must be closed. These conditions are checked by the software before calling the locking function. To call the locking function and lock a compartment, the user must choose a compartment to lock, enable at least one of the unlock conditions (timer, keypad, RFID, and/or fingerprint), configure the conditions they selected as defined in sections 5.6.1 and 5.6.2, and then close the compartment door. Upon closing the door, the magnet attached to the door comes to rest directly next to a reed switch situated by the doorframe, sending a high signal to the corresponding input pin and allowing the locking function to be called via a button press by the user. This causes the motor to push the locking bolt into position in the locker’s frame, preventing the door from opening until unlocked. Now, in the “locked” state, all user interaction with the door via software is limited to unlocking it as dictated by the enabled settings.

To unlock a compartment without the remote override or enabled timer, the user must first select a locked compartment from the locker’s menu. Upon selection, the user will be prompted to verify all of the input methods enabled for the compartment in sequence, as defined in section 5.6.1. Once all inputs have been verified, the unlock function will be called, and the motor will pull the locking bolt out of the locker’s frame. Once unlocked, the compartment will return to the “unlocked” state, allowing it to be configured and locked once again. If the remote’s unlock override is used for a compartment, or if a compartment’s enabled timer reaches zero, the unlock function will be called directly without checking the unlock conditions.

5.6.5 Locker Menu Design

To configure, lock, and unlock the compartments of the locker, the user must be able to select compartments and view instructions for configuring the unlocking conditions. To do this in a way that is intuitive and easy to follow along with, the locker’s software displays menus with settings and instructions on the locker’s 128 by 64 pixel OLED display. The user will be able to navigate between menu screens and select displayed options using the locker’s keypad. Table 21 below describes each of the menu screens, their purposes, and what inputs are accepted on each screen.

Table 21: Locker Menu Screen Descriptions

Screen	Purpose	Input
Main Menu	Display state of each compartment	1, 2, 3 to open menu for specific locker, ✱ for settings
Settings	Toggle call detection and check tamper history	# to toggle call detection, ✱ to return
Lock Compartment X	Configure locking conditions of compartment X	1 to toggle timer, 2 to toggle passcode, 3 to toggle RFID, 4 to toggle fingerprint, # to lock, ✱ to return
Configure Timer	Input time for timed unlock	Numbers to input time, # to enter, ✱ to return
Input Passcode	Input passcode for locking/unlocking	Numbers to input code, # to enter, ✱ to return
Scan RFID Tag	Prompt to scan RFID tag for locking/unlocking	RFID reader to input, ✱ to return
Scan Fingerprint	Prompt to scan fingerprint for locking/unlocking	Fingerprint sensor to input, ✱ to return
Unlock Compartment X	View remaining time or choose to unlock compartment X	# to unlock, ✱ to return

Figure 25 below shows mockups of most of the locker display menu screens, displayed in the order of the table above (with the exception of the fingerprint screen, as it is almost identical to the RFID screen). Note that the text on the final device appears differently, as it uses a library font appropriate for the 128 by 64 pixel screen size.

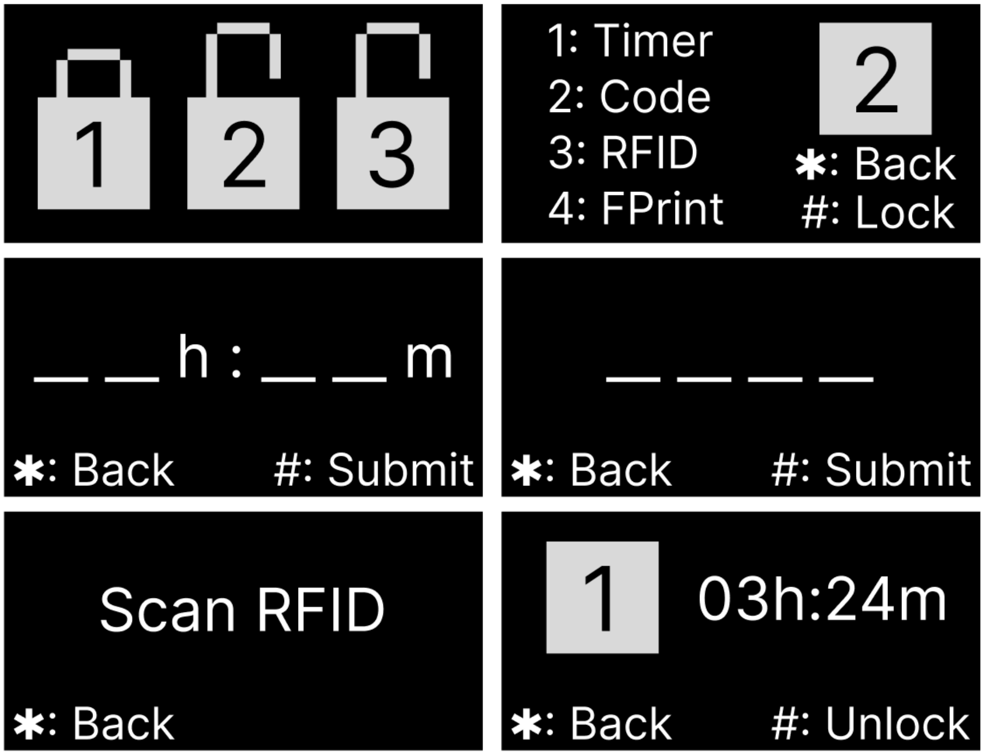

Figure 25: Locker Menu Mockups

5.6.6 Remote Menu Design

The Study Buddy remote is able to display the current state of each compartment and the remaining time for a timer-locked compartment on its 128 by 32 pixel display. The user is able to scroll between three separate menu screens, each corresponding to one of the three locker compartments, using a left and right button. There is also an option to override the unlock conditions of a chosen locker via a third button on the remote. The user is asked to confirm if they want to unlock a compartment, and does so by pressing the unlock button again, with the fourth button acting as a back button. Figure 26 below shows mockups of the locker's numbered menu screens and unlock confirm screen. Note that the font and font size differ on the final device.

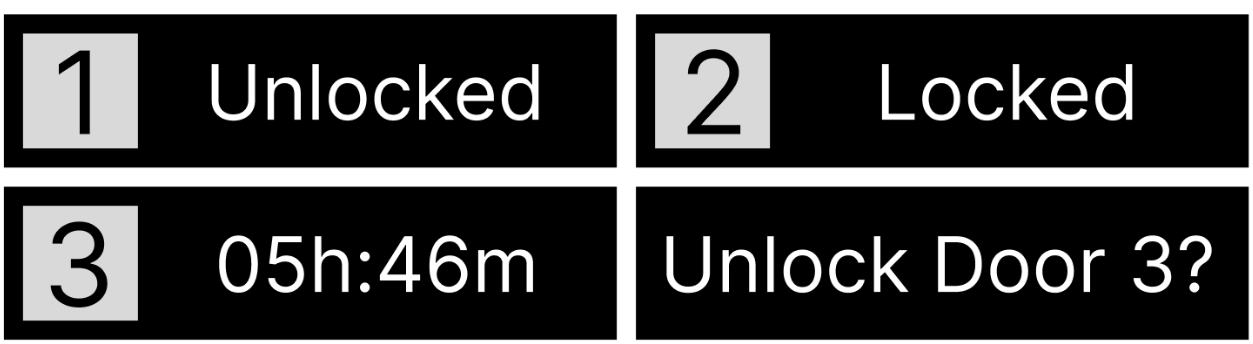

Figure 26: Remote Menu Mockups

5.6.7 Locker/Remote Communication

To communicate between the Locker and Study Buddy remote, we rely heavily on premade libraries for the nRF52 BLE radio. This allowed us to spend more time on what we wanted to send rather than how to implement the communication from scratch.

The main substance of our communication consists of the Locker sending regular updates to the Study Buddy remote containing information on the current status of the Locker. This mainly includes the current locked state of each compartment with information on the time remaining and information on how the lock was configured. The Study Buddy remote then displays this information to the user via the menu system.

The Study Buddy remote can then take input from the user and send an emergency unlock message back to the Locker with information on which compartment it should unlock.

5.7 Locker physical construction and design

Since one of our requirements is to hold common electronics such as laptops and handheld game consoles, the construction of our locker required lots of material to fit these large dimensional requirements. So, in order to keep costs and complexity down we chose to make the bulk of our locker out of plywood, since it is cheap, comes in large sheets and is easy to cut. This allowed us to save lots of time in the physical construction of our locker to instead spend on the electronics, software, and our advanced and stretch goals.

To join our pieces of plywood, we used wood screws to hold each wall together, including the interior walls joining each of the compartments. We used a separate type of crew to connect the back of the locker, which is a sheet of acrylic to allow us to see into the compartment from behind to show the actuation of the locks.

In order to meet our size requirements, we specified a rough set of dimensions for the compartments to be able to fit common types of electronics. The dimensions we used for a rough CAD mockup were 16 inches wide and 2 inches tall for the laptop compartment and 12 inches wide and 4 inches tall for the larger top compartment. These were chosen after approximately measuring the types of devices we wanted to be able to fit, such as a laptop and handheld gaming appliances like controllers or handheld consoles. We also took note to ensure that the small compartment will be able to fit most smart phones.

Using these dimensions, we constructed a simple CAD model to visualize design choices and work on the design of our mechanical component, the locking mechanism. Figure 27 shows the dimension-visualization state of our CAD design.

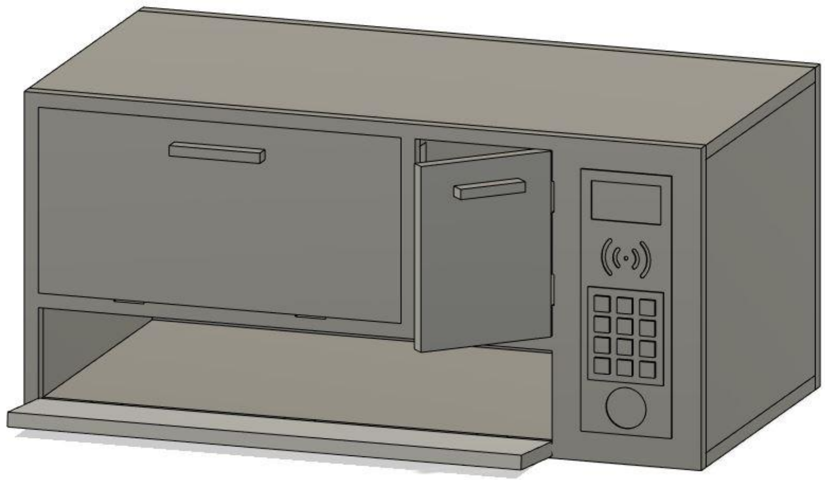

Figure 27: Rough dimensional CAD mockup of the Locker

<u>Attaching the doors</u>
To allow for our compartment doors to open, we used cheap hardware store metal hinges attached to the sides of the compartment door panels. These hinges are more than capable of supporting the weight of the doors plus the weight of the locking mechanism. Weeasily made space to mount the hinges by including small tab cutouts in the front panel and doors (shown in Figure 24).

One design decision we had to consider with the hinges was the direction each door should open. We knew that the bottom laptop compartment should open down to allow easy access, but were not so sure about the direction the other two should open. We started by modeling the hinges at the bottom of both top compartments, but found that it made opening the smaller smartphone compartment awkward. Instead we decided to have all the compartments open by their longer side, with the bottom and top left compartments opening down and the top right compartment opening to the right.

<u>Cutout for user interface devices</u>
We did not want to mount the RFID reader on the outside of our device, visible to the user, but we were also worried about it not being able to read any tags though the entire thickness of the plywood.

If we wanted to fix this issue with the least complexity, we would need access to a handheld or CNC router to carve the pockets in the plywood. Since we did not want to risk not being able to access one and due to none of our team members having any experience using a hand or CNC router, we chose to find another way of bringing the RFID sensor close to the surface of our face plate.

This led us to the idea of using a separate 3D printed insert which we could design to have recess and pockets for all of our user interfacing devices. This would allow us much more control and precision to the look and fit of our front panel and also be much simpler to build. We would design the cutout to fit the protruding parts of our keypad and fingerprint sensors snugly for a clean look, create a bezel around our display so only the actual display is visible from the front and add mounting holes to easily secure all of our devices.

Since it is under 9in in length, we could even use the 3D printer one of our teammates already has to avoid having to wait on getting our part printed from a print queue like is used in the TI Innovation Lab.

This idea would also allow us to easily remove the cut-out to access the electronics cavity or replace any of the user interface devices should they break. When we were done testing our project, we could simply screw the front panel in place. Figure 28 shows the constructed version of our cut-out faceplate.

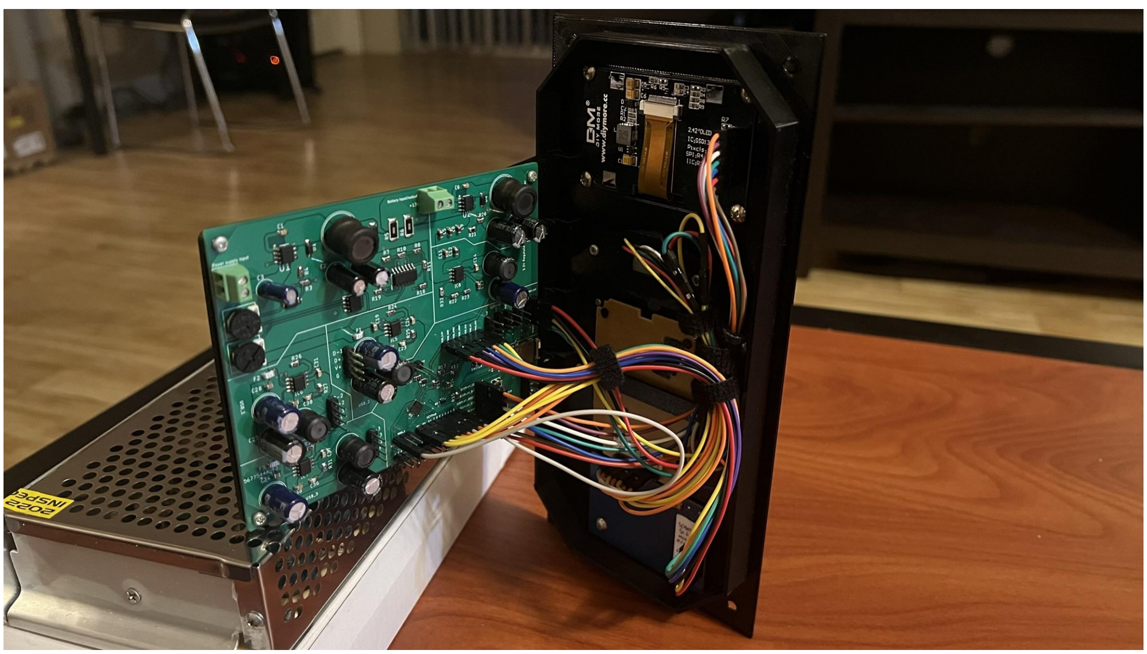

Figure 28: Back side of the user interface cut-out design

<u>Wiring</u>

Since our design required some of our devices be outside of the electronics compartment, we needed to run wires throughout our Locker. Specifically, the reed switches, servos, phone charging ports and vibration sensor needed to be placed in the compartments and have wires run to them. In a more professional version of our project we would ideally run all the wires inside of the walls of the compartments, but for our prototype we included small holes in our plywood panels to run wires though. Once we finished testing all of the external components we secured the wires by velcroing them into the corners of the compartments.

Final Construction

Using the methods described above, we constructed the actual locker relatively quickly once our PCBs had been tested and our components prepared. Figure 29 below is an image of the final constructed locker.

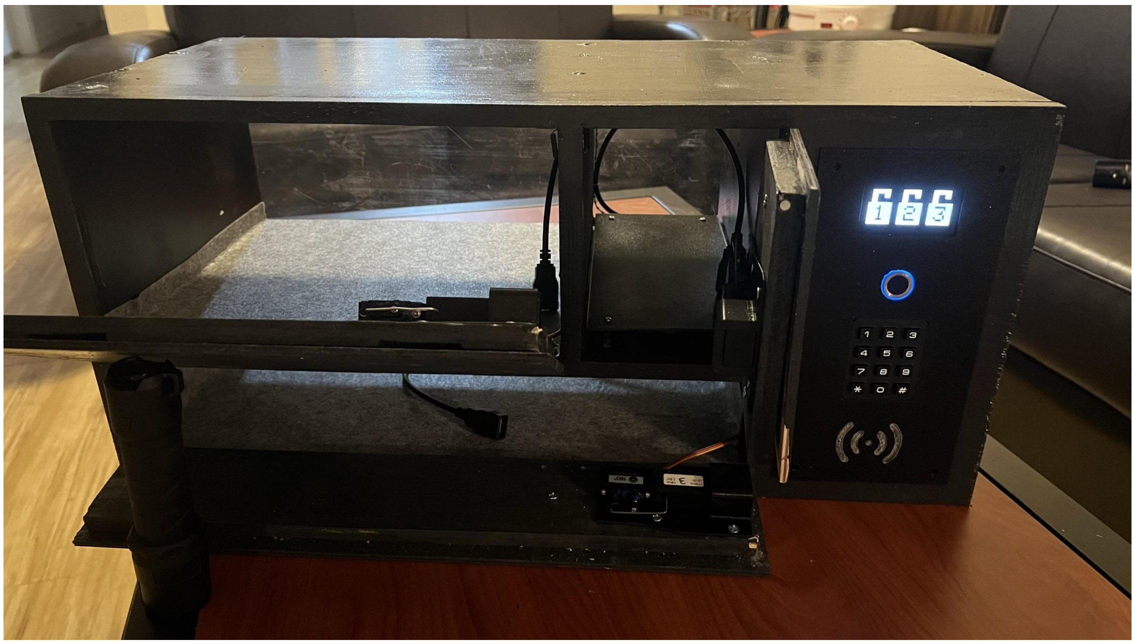

Figure 29: Final constructed locker

5.8 Remote physical construction and design

The physical design of our remote did not have any design requirements related to it, so its size and shape depended only on what we wanted to put in the remote. We also designed it to have a convenient form factor, though the main consideration we had when coming up with a design was the size of the screen we chose and being able to fit 3 AAA batteries. All in all, our remote is a small device with a screen and buttons.

Since it is small, we used 3D printing to make all of the physical components of the remote's body to achieve a much more polished result than gluing all of our components to a board, punching holes in a plastic box, or using hot glue and wood sheets to make the remote body. 3D printing also allowed us to add precise features like screw holes to mount our PCB to.

For our first design pass, we added our design dimensions of interest (the display's size and 3 AAA batteries side by side) into a sketch and created a rough shape for the remote around them. We added 4 buttons to navigate the interface on the display and some space for a PCB as well. Figure 30 shows the final constructed remote.

Figure 30: Final constructed remote

5.9 Locking Mechanism Physical Design

To ensure the security of the locker's compartment's we needed to design a locking system that is durable and secure relative to the locker's structure and that can be electronically controlled with minimal signals (as the locking devices will be located farther away from the MCU than most of the other components).

To accomplish the security function, we use a sliding bolt lock, made with a durable 3D printed material. Each bolt lock is positioned on a compartment door such that, when its door is closed, the locked bolt will slide into holes cut into the locker's frame. With this locking method, the weakest link in the locking mechanism is a bolt made out of a durable 3D printing material, meaning that the compartments cannot be easily forced open.

To accomplish the electronic control of the locking system, use a small positional servo motor to control each sliding bolt. When given the correct signal from the MCU, the motor will toggle between two set angles, which correlate to the unlocked and locked positions. The motor is positioned such that it is upright relative to gravity and its back surface is attached to the door, with the servo horn in line with the locking bolt and attached to it with a stiff wire. In this configuration, shown in Figure 31 below, the motor can be rotated between two set positions, directly pushing or pulling the bolt with it. Because the servo motor and locking bolt are both attached to the door on the inside, the motor is not at risk of damage if its door is forcefully pulled while locked.

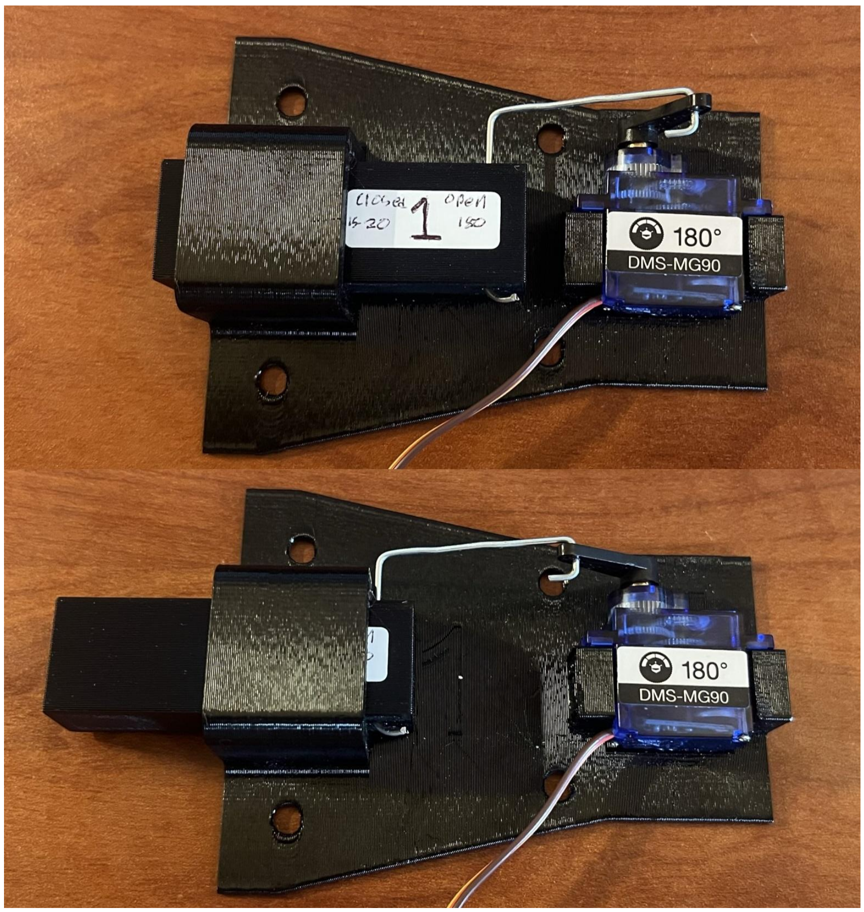

Figure 31: Locking Mechanism Final Design

6.0 Testing and Evaluation Plans

This section details our plans for testing and evaluation of all of our parts and components. The purpose of this testing was to ensure the parts we ordered and received both function properly and satisfied our design requirements. We also wanted to compare certain technologies we identified in our research but could not decide on without testing them ourselves. The result of having completed our testing and evaluations is a set of demo code for each component that we used to build on to create the functionality of our full project.

6.1 MCU testing

Our testing plan for our MCU platform involved both evaluating that we could plug in all of our peripherals and get them to work as expected and testing to ensure that we did not face any major difficulties when we had to switch to using blank chips and our custom PCB design.

To test the general functionality of our MCU for our project, we first soldered headers onto and attached our evaluation board to a breadboard. Then we used header wires to connect all of our different peripherals to test them individually, then tested all of the SPI devices together to check that they did not interfere with and worked with each other. While testing each device, we also tested that our development environment worked using premade Arduino IDE libraries and figuring out what workarounds we needed to do. The result of these tests was our confidence that we could easily develop our prototype using the evaluation board we picked using premade libraries.

A secondary concern we investigated during our testing and evaluation of the MCU was how many GPIO pins our design needed in total. As we tested our peripherals, we kept a tally of GPIO pins used and for what. After we finished testing we took note which pins could be shared between devices and then calculated the total number of pins required. We used this figure, and information on which pins were needed and for what to compare with both our evaluation board and the similar board we were basing our design on to see if we needed to reduce the number of pins we required.

Although both the MCU and the module we chose have over 40 GPIO pins, the arduino compatibility ‘version’ of our MCU only supported the generic pins common to all arduino uno derived boards. This meant if we wanted to use these extra GPIO pins in our design, we either needed to abandon the arduino IDE and switch to using the nRF Connect SDK or figure out how to modify the arduino support configuration to essentially create a full new board definition for the Arduino IDE. This could be easily avoided, however, if we noticed we needed more GPIO pins by using a ‘GPIO expander’ device, which essentially is a chip that has its own GPIO pins that can be controlled over i2c. This way, we could move all of our simple on/off devices to the GPIO expander potentially turning 11 GPIO pins into 2.

The last thing we needed to consider while evaluating our MCP platform is how we could get arduino support for the blank module we had to use in our custom PCB. These modules did not come with bootloaders on them, and thus had USB support disabled until we loaded one. To flash the bootloader onto our chips we needed to use a J-Link device over the debug pins of our MCU. Our design reference board had these pins available already and one of our group members had the official nRF52 development board (shown in Figure 32), which include a J-Link to program standalone chips.

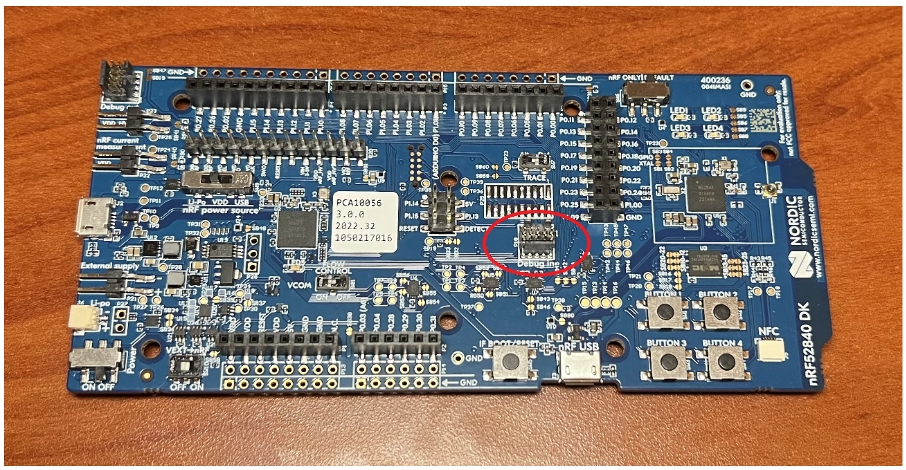

Figure 32: A nRF52840 Development Kit with J-Link connector circled

If we could not get the arduino compatible bootloader onto our chips, we could resort to buying a version of our reference board with the same, but pre-programed module, and remove it to solder onto our PCB.

6.2 Wireless testing

Testing our wireless standards was important for us since we had a concern that we would not be able to meet our wireless range requirement using our low power radios. This is why we chose a second wireless standard in case our primary choice was either too difficult to implement or could not meet our range requirements. As such, our wireless testing and evaluation would ensure we could meet our wireless range requirements and to compare our wireless standards to see which was the easiest to work with, had the best performance and checked power consumption for our battery life requirements.

Our main standard we chose was Bluetooth Low Energy (or BLE) since the radio was available built into our MCU and the module we chose had an integrated antenna, making adding it onto our design trivial. It also had better low power performance than the nRF24 module we chose and had official support in the Arduino IDE.

To test if BLE worked for our project we first needed to connect both of our evaluation boards up to a breadboard and load one of the BLE examples from the arduino library. After verifying that we could send data between our evaluation boards, we created a test program that would send and receive data in a similar manner we planned to use for our project. Using a laptop to monitor the output from the evaluation board standing in for our remote, we would then walk around a few different environments (in open, indoor and RF noisy environments) to test the range and reliability of BLE. If we got good results at close range, but started dropping messages closer to our target range we would experiment adding resilience to our wireless communication using multiple sends, repeat send after timeout and acknowledgement messages to try and improve reliability without messing with more advanced configuration or re-creating a more robust (and complex) protocol like Wi-Fi or TCP.

If we found that using BLE was too difficult or could not get satisfactory results from it, we would attach our nRF24L01 modules to our evaluation boards over SPI. We would then repeat similar evaluation steps to BLE by loading library examples and testing at close range, moving to longer range testing with a more representative program and trying to add reliability if we would get poor performance.

After testing, we would select the wireless standard we found the easiest to use, giving preference to BLE (due to the lack of hardware design work). If we could not get the range and reliability as needed by our design requirements, we would choose the best performing and easier to work with standard and consider reducing our requirements.

6.3 Displays testing

To test to see if the displays we selected would be functional in our design, a breadboard circuit was built. The Arduino Nano would be used to power and program the display in place of our final design’s power supply and control chip. The 3.3 V pin was used to power the display used in each test run. Both of the displays selected would use SPI configuration and were tested individually. We made use of a library provided by the manufacturer of the display and ran a test program on the Arduino Nano. We tested to see if the full display would properly light up to ensure that there were no damaged pixels. A second test was made to both displays to check if they were capable of properly updating with new information received from the microcontroller.

6.4 Fingerprint scanner testing

Testing the base function of the fingerprint sensor could be done via a Windows software provided by Adafruit, shown in Figure 33 below. Using their software, the sensor could be connected to a Windows device via a USB port and a USB-serial converter, then tested using minimal connection to our testing microcontroller. The windows software required uploading a blank sketch (or a different test sketch provided by Adafruit) to the testing MCU and connecting its RX and TX pins to the fingerprint sensor. After doing so, the SFG software could be used to easily test the functionality of the enroll and search functions of the sensor.

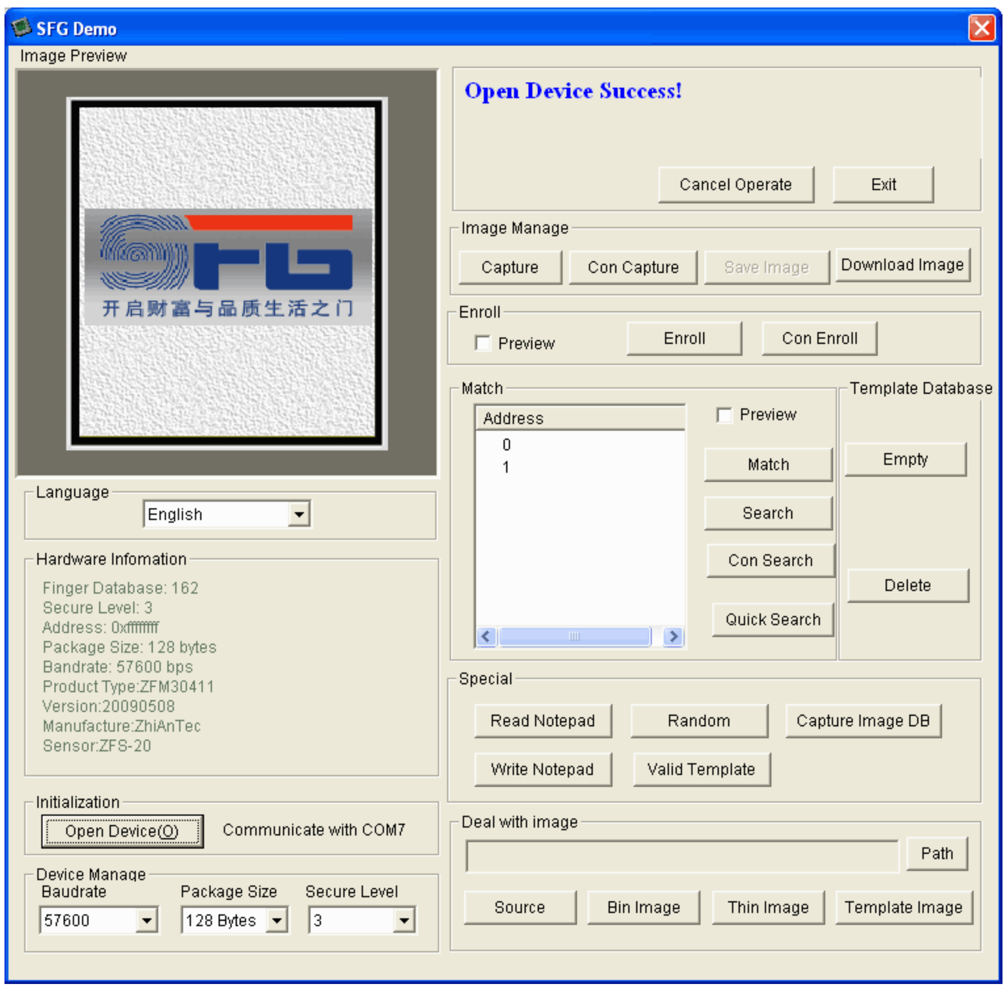

Figure 33: Testing Interface of Adafruit's provided Windows SFG software

To test to see if the fingerprint module we selected would be functional and compatible in our specific design, a breadboard circuit was built. The Arduino Nano was used to power and program the module in place of our final design's power supply and control chip. The module interfaced using UART configuration and used the Nano to configure the module. We tested the fingerprint sensor (via the provided Adafruit_Fingerprint enroll) to see if it could properly enroll fingerprints scanned over several trial runs. We checked to see if the sensor was reliable enough to be used in our design. Ideally the reliability score should have been above 80% to be acceptable and if it was not, changes would have been made to the code of the program to see if the sensor could be improved. If it still did not pass, then a change in sensors would have been necessary. If it passed, we moved on to the next test.

The second test to see if the fingerprint module was functional with our design involved checking to see if the sensor could properly recognize a fingerprint scored in its database. It is important to check that the sensor can recognize the same fingerprint with high reliability to avoid the user potentially getting locked out of the device so extreme care was taken when performing the test. The sensor should have around the same reliability score recognizing the same fingerprint as scanning a new one. If it failed, changes to how the sensor scans and stores information may have been necessary and if the reliability score still did not improve with future tests, another sensor would have been necessary.

6.5 Keypad testing

To test to see if the keypad we selected was functional and compatible in our design, a breadboard circuit was built. The 7 pins of the keypad were connected to the Arduino Nano and we ran a program that would take inputs from the keypad and display the results on a screen.

To check which buttons were being pressed, we would raise each row one at a time to high then measure if any of the 3 columns were high as well. Because of how the keypad was wired internally, the only way we would measure a high signal on a column is if a button in that column is being pressed. Since we only raised one row at a time, the columns that were high corresponded to a pressed button in that column and the row currently being held high. By iterating through each row we could detect if every button was being pressed independently without having to worry about collisions, or signals from two buttons being pressed only looking like one.

As we went through each row we would save buttons that have been detected into an array. The expected result of this test code was for the program to display the correct state of which buttons were currently being pressed.

6.6 RFID Module and Tags testing

To test the RC522 RFID reader module, we connected it to power and connected the four SPI pins (SDA, SDK, MOSI, and MISO) to our MCU evaluation board.

With the MCU and RFID reader properly connected, we installed the MFRC522 library in the Arduino IDE. This library comes with text code we could modify to easily check the functionality and effective range of the RFID reader and its tags. When running this example code, holding a tag close to the reader module would output the tag's UID to the console, along with other information such as the available space for writing to the tag, as shown in Figure 34 below.

```
Firmware Version: 0x92 = v2.0
Scan PICC to see UID, SAK, type, and data blocks...
Card UID: F9 8A D2 A3
Card SAK: 08
PICC type: MIFARE 1KB
```

Figure 34: Example Output of MFRC522 Library RFID Test Code

By running the above test with all of our available RFID tags and cards, we could confirm their UIDs, effective ranges, and the general functionality of the RFID reader module. Once we had confirmed that the UID of each tag read correctly, we were certain that our final software would be able to identify and store the UID of any compatible RFID tag, which is the ID we used to identify tags for locking and comparing tags from unlocking the locker's compartments.

6.7 Servo Motors testing

To test the functionality of our servo motors, we first needed to connect the power, ground and the PWM signal wires to our MCU evaluation board. After setting up our MCU we loaded one of the servo control examples from the Arduino servo library. This code tested our servos by setting their positions, waiting, then moving them again. We only needed to change the example code minimally to get our servo motors to move, only needing to change the pin definitions to match our setup. The result of this test verified that all of our servo motors functioned and were compatible with the Arduino servo library.

We then went on to test our servo's ability to move a sliding bolt when resistance is applied. This test showed us that the servos we selected were capable of sliding the locking mechanism closed even if it is not properly aligned and had higher than ideal friction. IF our servo motors were able to still move the locking mechanism closed with resistance, we were confident that our compartments would not fail to shut or reopen. If they failed to move the locking mechanism with resistance applied we would have had to redesign our locking mechanism to have a higher mechanical advantage or ensure it did not bind if incorrectly aligned to protect our motors from damage.

6.8 Reed Switches testing

To test that our reed switches are functional, we needed to connect them to the probes of a multimeter set to continuity mode. Since the reed switches we selected are normally open, we would not get a continuity reading until we moved a magnet near them. After testing all of our reed switches, a successful result was found in that all of them changed from an open circuit to setting off the continuity probe when a magnet is brought near them.

We also needed to test how close the magnet needs to be before triggering the reed switches and how far away it can get before it goes back to an open circuit. This information was important to let us know how we need to mount our magnets and reed switches in the compartments so that we can detect when the compartment has been opened very soon after it starts to move. This was found to be a matter of millimeters, which was trivial as the magnets rested directly against the reed switches in the final design. By positioning our reed switches and magnets at the right distance, we could ensure that a compartment door has been opened before any sort of tool can be insured into the gap and potentially interfere with our knowledge of the state of the compartments.

6.9 Power Supply & Voltage Regulators testing

After we got all the needed parts for the power supply and the different voltage regulators, we tested each individual part to make sure they are indeed the right ones. Then, we connected them together using wires. After making sure all components were connected together in the right way, we started testing the needed and the limit characteristics of the power supply and the voltage regulators. For the power supply, we tested the rated typical use and maximum power output, rated current, and rated voltage. We also needed to test for the output EMI or switching noise.

The procedure began by connecting the power supply to the wall AC 120 volts signal, then using a multimeter to check if the output voltage is 24V. Using a variable load unit from the lab, we tested the capability of the power supply to output 150 watts as its maximum rating, and saw the output current. We also needed to measure the temperature of the unit, as very high operating temperatures can lead to fire hazard, especially in the locker as the model is made of wood.

For the voltage regulators procedure, we started by applying the intended input voltage in the design. Then we could start measuring the output voltage to see if they are the rated ones and using a variable load unit from the lab enabled us to see if the maximum current rating of the full design is indeed true. Additionally, we needed to measure the output voltage using an oscilloscope, so we could see if the EMI at the output would be an issue that causes the MCU and other sensors in the system to not function properly. We also needed to measure the temperature of the components, so they do not create a hazard while operating. After gathering all information from the input and output, we could calculate the efficiency of each voltage regulator and compare it to the rated one and the planned one in the design section, which is about 85% for both regulators.

6.10 BMS testing

6.10.1 Battery pack

Preparing the battery pack

As the configuration of the Battery Management System was chosen to be three battery cells in series (3S), the battery pack has to be the same to make sure the BMS and the battery pack will function properly. Starting from the 0V node or (B-), which should connect to the first cell's negative terminal. Then moving to the 4.2V node or (B1), which should connect to the first cell's positive terminal and also the second cell's negative terminal. Then moving to the 8.4V node or (B2), which should connect to the second cell's positive terminal and also the third cell's negative terminal. Lastly, moving to the 12.6V node or (B+), which should connect to the third cell's positive terminal. In the BMS we also have two connecting points called (P+) and (P-), which is the place to connect the power in for charging the battery pack and the power out for providing voltage and current to make the needed application operate. Noting that the connection point (P+) is connected to the third cell's positive terminal which is also the (B+) node, but the (P-) power connecting point is not sharing with the first cell's negative terminal (B-), and that for some protection features.

Charging the battery pack

When charging the battery pack we needed to use the constant current and constant voltage method. For this pack the needed charging voltage is the pack's total voltage when fully charged which is 12.6V. The constant charging current can be any amount from 0.3C to 1.5C of the full battery pack capacity, so we chose to use 0.5C current, which is about 1.7A to make sure we maintain a longer battery pack lifespan. Noting that the power supply does not know what current value the limit is we want to use and might push a current of 2C or more, which can cause a fire hazard. So when charging the battery pack, we had to use a power supply that has a constant current output feature, which means we had to add the constant current feature into the BMS input power.

6.10.2 BMS testing

The Battery Management System's main role is to overlook the battery pack and each individual battery cell in it, which meant it had to have many features and operating characteristics to function properly. After we got all the needed parts for the BMS including the batteries, we tested each individual part to make sure they were indeed the right ones. Then, we connected them together using wires. After making sure all components were connected together in the right way, we started testing the needed and the limit characteristics of the BMS.

Overvoltage protection

Testing the Overvoltage protection feature of the BMS should be easy. First we monitor each battery cell's voltage individually by a separate voltmeter, then we connect the power input terminals to higher voltage than the rated one which is 12.6V giving each cell 4.2V, and see if the individual battery cell's voltage changes to add up to the input voltage. If it did not change, then that means the Overvoltage Protection feature works, if the individual battery cells' voltage changed to add up to the input voltage, then the protection feature failed to activate.

Over discharge voltage protection

For testing the BMS over discharge voltage protection feature, we needed to connect the battery pack to a variable load unit from the lab and let it discharge and monitor each cell's voltage. After the rated protection voltage range of 2.3V to 3V, we should see the BMS shutting off the power outputs by turning the first set of power MOSFETs off and a current draw of 0A at the variable load unit. The BMS should not turn the power on again until the battery pack is charged above the rated voltage.

Short circuit protection feature

We needed a load that we can tell if it turns on or off like an LED light for testing the short circuit protection feature. We just needed to connect the load to the power output of the BMS, then short circuiting the two ends with a wire. If the LED light turns off, then the short circuit feature works, if the LED did not turn off, then the feature fails to activate, and the battery pack is in big hazard of exploding or catching on fire.

6.11 QC USB charging circuit ports testing

After we got all the needed parts for the USB charging circuit, we tested each individual part to make sure they were indeed the right ones. Then, we connected them together using wires. After making sure all components were connected together in the right way, we started testing the needed and the limit characteristics of the input voltage. For the output, we used a USB meter that has a built-in QC 2.0/3.0 simulator that sends a signal to the IC inside the charging circuit. We also used a multimeter and multiple devices ranging from smartphones to power banks to even small gadgets like smart watches to see if the chip can send the required output voltage and charge the devices correctly. All those methods enabled us to see if the maximum rated characteristics of the input and the output are true and enabled us to have a chance of redesigning it if they had not met our goals.

After we bought a product that is very similar to the designed one from the market, we started testing it by using a power supply from the lab and inputting variant voltages to the USB charging chip and we connected a Samsung device that is compatible with QC 3.0 to it, as shown in Figure 35. Using a multimeter from the lab, we could see that the output voltage was always set to 9V whether the input voltage was 12 V or 24 V as seen in Figure 36 below, which meant that the buck converter worked great and could handle variant input voltages.

Figure 35: Demonstration of quick charging using QC board

Figure 36: Charging voltage applied to module and current draw

6.12 Vibration Sensor testing

The testing and evaluation of our vibration sensors was done largely to see if we could use them to reliably detect phone calls with few false positives. We also experimented with different ways of interacting with it to create our own phone-detecting code to abstract away some of the basic control and give us a more library-like experience when developing our prototype since we could not find much (or any) information about the modules online.

The first step in testing our vibration sensor was to connect it to a digital GPIO pin on our evaluation board. We then wrote a simple program that outputs the value of its signal pin to the console to verify and understand its behavior. Then we started to test how it responds to different kinds of vibrations such as bumps, shaking with our hand, shaking/ tapping the surface it's on, and an actual cell phone ringing in different locations on and around it and test how sensitive it is. We also tweaked its sensitivity using the trim potentiometer on the module.

After testing how it reacts to actual phone calls and various false stimuli, we implemented some of the different detection strategies outlined in sections 3.16.1 and 5.6.3 for reliably detecting phone calls and tweaked them until we met our reliability requirement and had a low false positive rate. In the end, we decided to poll the sensor constantly, keeping a log of the number of toggles within the last minute. If this number ever reaches 2500, the compartment unlocks, as this constitutes about 1.5 phone calls worth of vibrations.

7.0 Administrative Content

The following section will overview the money and time management aspects of our project, specifically discussing the budget, financing, and milestones for the Study Buddy's design.

7.1 Project Budget and Financing

Table 22 below shows estimates of the cost of components and total overall cost of the Study Buddy's construction. The study buddy is not a sponsored project, thus all members personally contributed an equal amount toward component and material costs

Table 22: Estimated Cost of Project Materials

Component	Quantity	Estimated cost
Keypad	3	$ 15.00
Large display	1	$ 27.00
Small display	4	$ 14.00
Fingerprint sensor	2	$ 40.00
RFID reader + tags	3	$ 24.00
Vibration sensor	5	$ 6.00
Rechargeable batteries	6	$ 30.00
Remote battery	2	$ 6.00
BMS	2	$ 10.00
Locks	3	$ 13.00
Small motors	6	$ 35.00
Reed switches + magnets	10	$ 17.00
Radio modules	10	$ 14.00
USB charging ports	4	$ 12.00
Controller chips (MCU)	4	$ 50.00
Push buttons (multi pack)	1	$ 13.00
PCB (main)	1	$ 15.00
PCB (remote)	1	$ 15.00
Construction material	1	$ 12.00
Total		**~$ 368.00**

Table 23 below shows the actual final costs of the materials for the Study Buddy locker, and Table 24 shows the final costs of materials for the remote.

Table 23: Locker Final Bill of Materials

Locker		
Component	Quantity	Actual cost
Plywood sheets	2	$ 49.56
Hinges	3	$ 4.92
3D printed material	273.8g	$ 6.02
Large display	1	$ 26.99
Fingerprint sensor	1	$ 19.95
Keypad	1	$ 4.95
RFID reader + tags	1	$ 7.99
Vibration sensor	1	$ 1.30
Servo motors	3	$ 17.70
Reed switches + magnets	3	$ 4.55
BMS	1	$ 10.69
Rechargeable batteries	3	$ 15.00
USB cables	3	$ 5.69
Power supply	1	$ 19.99
PCB	1	$ 1.60
Controller chips (MCU)	1	$ 12.95
PCB components	-	$ 40.48
Total		**$ 250.33**

Table 24: Remote Final Bill of Materials

Remote		
Component	Quantity	Actual cost
3D printed material	38.85g	$ 0.85
Small display	1	$ 4.00
AA batteries	2	$ 1.16
PCB	1	$ 0.40
Controller chips (MCU)	1	$ 12.95
PCB components	-	$ 5.26
Total		**$ 24.62**

7.2 Project Milestones

Table 25 below shows the important due dates and milestones in the project.

Table 25: Project Milestones

Milestone	Date
Divide & Conquer Complete	10/7/22
Testing Components Ordered	11/8/22
Main Schematic Complete	12/4/22
SD1 Document Complete	12/5/22
Breadboard Prototype Complete	2/1/23
First Locker PCB Design Ordered	2/11/23
Critical Design Review	2/24/23
Power System Tested	3/3/23
Main Software Completed	3/18/23
Middle Term Demo	3/20/23
Revised Locker PCB Ordered	3/27/23
Remote PCB Ordered	3/27/23
Final PCBs Tested	4/10/23
Physical Construction Complete	4/12/23
Final Presentation	4/19/23
SD2 Final Document Complete	4/22/23

8.0 Study Buddy Operation

The following section will serve as a user's manual for the Study Buddy, describing the features of the device overall, as well as giving detailed instructions on how to use the locker and remote for a user completely unfamiliar with the design. Labeled diagrams of the physical components are included to ensure the user can easily identify all of the major functional components of the Study Buddy system. In addition, instructions for a full single use of the system, from configuration to unlocking, is included in section 8.3.

8.1 Locker Design and Features

Figure 37 below shows a labeled isometric view of the Study Buddy's locker. Label 4 refers to the entire rectangular panel on the right side of the device, rather than one of the individual components.

Figure 37: Labeled Locker Overview

Labels 1, 2, and 3 indicate the storage compartment corresponding to that number. These numbers are used when configuring and unlocking compartments to clearly indicate to the user which compartment is being referred to. Label 4 indicates the locker's input panel, which Figure 38 below shows in more detail.

Figure 38: Labeled Locker Input Panel Mockup

In the above figure, the component indicated by each label is the individual part located to the left within the larger panel. Label 1 indicates the display, which will prompt the user to select a compartment and will give instructions when configuring the locker for use. Label 2 indicates the fingerprint scanner, which can be used as an optional unlocking method if the user enables it. Label 3 indicates the keypad, which is used for navigating the locker’s menu and for inputting passcodes and timer values for configuration or entry. Label 4 indicates the location of the built in RFID reader, and is where the user needs to hold their corresponding RFID tag to use the reader as an unlocking method.

8.2 Remote Design and Features

Figure 39 below shows a labeled view of the Study Buddy's remote. Labels 2 through 5 each refer to the single button directly above them. Other than the labeled components and the remote's battery tray, there are no other components on the remote that the user will need to interact with.

Figure 39: Labeled Remote Mockup Overview

Label 1 indicates the remote's display, which shows the lock state of the selected locker and prompts the user to confirm an unlock override. Labels 2 through 5 indicate the remote's input buttons, which are used to scroll through the compartments, activate and confirm the unlock override, and return to the previous screen. The exact button assignments are covered in section 8.4. Label 6 indicates the remote's built-in power switch.

8.3 Locker Configuration and Usage

The below steps detail how a user can configure a lock, lock a compartment, and unlock a compartment on the Study Buddy locker. For information on how to use the Study Buddy remote, please see section 8.4.

1) Plug the locker into a standard wall outlet and wait for the display to turn on.
2) Open the compartment you wish to lock (labeled 1 through 3) and place anything you would like to lock inside.
3) Close the door of the compartment.
4) Press the keypad button corresponding to the locker you are using.
5) Press the keypad button corresponding to the first unlock method you would like to enable. At least one method must be selected before you are able to lock the compartment. The instructions for each method are given below.
    a) Timer: Input the hour and minute values of the timer. The locker will automatically unlock once this timer reaches 0, regardless of what other input methods are enabled. Press # to submit the time, or ✱ to cancel.
    b) Passode: Input the four digit code you would like to be able to unlock the compartment with. Press # to submit the code, or ✱ to cancel.
    c) RFID: When prompted, hold your compatible RFID tag next to the RFID reader. The tag can be removed when indicated, but may need to be held up again if prompted. Once your tag has been accepted, the unlock method will be fully configured. Press ✱ to cancel.
    d) Fingerprint: When prompted, place your fingertip against the fingerprint scanner. You may be prompted to remove and replace the finger in order for the device to confirm your print, or if there is a problem with the scan. Once your fingerprint has been accepted, the unlock method will be fully configured. Press ✱ to cancel.
6) To remove an unlock method that has been enabled, press the keypad button corresponding to that method. If you wish to enable it, you must go through the configuration process for that method again.
7) Once you enabled the methods you would like to use, press # to lock the compartment door.
8) To unlock the compartment using the passcode, RFID, and/or fingerprint methods, select the desired compartment from the main menu once again.
9) Press # to begin the unlock process. Each of the enabled unlock methods will be prompted, and the user must input the correct code, scan the correct RFID tag, and/or scan the correct fingerprint in the order given.
10) Once all unlock methods have been confirmed, open the unlocked compartment.

The above instructions apply to each of the three compartments individually. All three can be locked by different locking methods simultaneously, as well as having different timers. The expected user inputs from the keypad are labeled on every screen of the display.

8.4 Remote Usage

The Study Buddy's remote is simple, mainly serving as a way to check the status of each compartment and unlock them instantly as an emergency override method. The middle two buttons are for scrolling left and right through each of the three compartment screens. A screen will say "Unlocked" if a compartment is unlocked, "Locked" if it is locked without the timer enabled, and will list the current remaining time if it is locked with the timer enabled.

To use the emergency override, press the fourth button when on the screen of a locked compartment. You will be shown a prompt asking if you are sure you want to unlock the compartment. Pressing the unlock button again will unlock the compartment, overriding any unlock methods that were enabled on it. Pressing the first button will instead cancel the unlock, bringing the user back to the compartment screen.

8.5 Extra Features

The phone compartment (number 2) of the locker comes with a vibration sensor built into the bottom of the compartment. This can be enabled as another unlock method in the settings menu, and will open the compartment instantly if triggered, similarly to the timer. This is intended as a way to unlock the compartment in an emergency, but relies on the user to understand when a particular phone would vibrate and if that constitutes a scenario worth giving access to the phone. Because a phone may vibrate in a non-emergency scenario, this method can be disabled and the remote can instead be used for emergency scenarios.

There is also an extra menu on the locker that can be accessed by pressing ✱ from the main menu. This extra menu will indicate if the locker detected an unauthorized entry of a compartment that was locked, and will also allow the user to toggle phone detection for locker 2. Pressing ✱ on this menu will bring the user back to the locker's main menu.

9.0 Project Summary

As the capstone project of our engineering education, the design and creation of the Study Buddy serves to round out the skill sets of our team's members to prepare us for working in a real engineering environment in the near future. Throughout the design stages of the Study Buddy, we as a team have gained several skills that can only come from real design experience, and not from following instructions or memorizing material. In order to work effectively as a team, the skills of some of our members have been spread to the rest of us, such as power design, software design, and experience with wireless communication. In addition, we have all learned about topics that we had not needed previously, such as AC/DC conversion, battery management, schematic design, PCB design, and user interfaces. The general design principles needed to create a user-friendly device will certainly benefit all of us in our future workplace, as well as some more general engineering skills such as searching for components, reading and understanding datasheets, and researching interesting technologies.

The Study Buddy is not intended to be a revolutionary product, but instead is meant to fill a niche for a device that can store and lock away objects temporarily to help with discipline and work ethic. By providing multiple input methods, the device is flexible enough to suit a potential user's needs, while also being interesting to design from a hardware and software perspective. The addition of the remote allowed us to explore wireless standards and design and can make for a helpful tool for users who do not wish to directly monitor the locker. We also gained experience in power design with our use of wall charging and batteries, which are skills valuable to any electronic product designs we may contribute to in the future.

The process of finding useful information about technologies and components has been interesting, as available information can often be inconsistent or misleading depending on the source. Overall, we have improved our ability to pick through information to find what is useful given our specific circumstances and requirements. We have also learned effective ways of communicating ideas, as well as translating our designs to diagrams and schematics to effectively demonstrate our progress. Our early documentation was invaluable when we constructed our prototypes and our final design. Through the combined efforts of our group, we have produced a design that we are proud of, and were able to create a functioning version of the Study Buddy as we envisioned it. We enjoyed the challenges and learning experiences we encountered on our design journey, and hope to be able to apply the skills we acquired professionally in the near future.

References

1) Notes, Electronics. “Lithium Ion Battery Advantages & Disadvantages.” *Electronics Notes*, https://www.electronics-notes.com/articles/electronic_components/battery-technology/li-ion-lithium-ion-advantages-disadvantages.php.

2) “Nickel Metal Hydride.” *Nickel Metal Hydride - an Overview | ScienceDirect Topics*, https://www.sciencedirect.com/topics/engineering/nickel-metal-hydride.

3) “Zinc-Ion Battery.” *Zinc-Ion Battery - an Overview | ScienceDirect Topics*, https://www.sciencedirect.com/topics/materials-science/zinc-ion-battery.

4) Power. “Zinc-Ion Batteries Are a Scalable Alternative to Lithium-Ion.” *POWER Magazine*, 8 Jan. 2021, https://www.powermag.com/zinc-ion-batteries-are-a-scalable-alternative-to-lithium-ion/.

5) “Nickel Metal Battery.” *Your Trusted Battery Power Supply Partner in China!*, https://www.huajubattery.com/ni-mh-metal-battery/.

6) Keim, Robert. “What Is a Linear Voltage Regulator? - Technical Articles.” *All About Circuits*, https://www.allaboutcircuits.com/technical-articles/what-is-a-linear-voltage-regulator/.

7) *Linear and Switching Voltage Regulator Fundamental Part 1*. https://www.ti.com/lit/an/snva558/snva558.pdf.

8) Author Cadence PCB Solutions, et al. “Using a Switching Regulator vs. Linear Regulator for DC-DC Conversion.” *Using a Switching Regulator vs. Linear Regulator for DC-DC Conversion*, 13 Oct. 2022, https://resources.pcb.cadence.com/blog/2020-using-a-switching-regulator-vs-linear-regulator-for-dc-dc-conversion.

9) “#772 Basics: Switching Power Supplies (Part 1 of 2).” *YouTube*, YouTube, 22 Apr. 2021, https://www.youtube.com/watch?v=9bFVTa8FEM4&list=WL&index=1.

10) “Power Supply Safety Standards, Agencies, and Marks.” *CUI Inc*, 20 July 2020, https://www.cui.com/catalog/resource/power-supply-safety-standards-agencies-and-marks.

11) “Differences between International Safety Standards for Power Supplies.” ASME, https://www.asme.org/topics-resources/content/how-to-manage-regulations-for-power-supplies.

12) "Power Plug & Outlet Types A & B." *WorldStandards*, 24 Jan. 2022, https://www.worldstandards.eu/electricity/plugs-and-sockets/ab/.

13) "IEEE SA - IEEE Standard for Rechargeable Batteries for Multi-Cell Mobile Computing Devices." *IEEE Standards Association*, https://standards.ieee.org/ieee/1625/4382/.

14) "USB Hardware." *Wikipedia*, Wikimedia Foundation, 5 Oct. 2022, https://en.wikipedia.org/wiki/USB_hardware.

15) "USB Power Delivery Is the Fastest Way to Charge IPhone and Android Devices." *ZDNET*, https://www.zdnet.com/article/usb-power-delivery-the-fastest-way-to-charge-iphone-and-android-devices/.

16) "Lithium Ion Battery Management and Protection Module (BMS) Teardown - Schematics, Parts List and Working." *Circuit Digest - Electronics Engineering News, Latest Products, Articles and Projects*, https://circuitdigest.com/electronic-circuits/lithium-ion-battery-management-and-protection-module-bms-teardown-schematics-parts-list-and-working.

17) Tritek. "What Is a BMS Protection Board of Lithium Battery?" *Ebike Battery Manufacturer*, 14 Oct. 2022, https://tritekbattery.com/bms-protection-settings-of-the-lithium-battery/.

18) "USB: Port Types and Speeds Compared: Tripp Lite." *Tripp Lite Website*, https://tripplite.eaton.com/products/usb-connectivity-types-standards.

19) Darroudi, Seyed Mahdi, et al. "Bluetooth Low Energy Mesh Networks: A Standards Perspective." *IEEE Xplore*, https://ieeexplore.ieee.org/abstract/document/9071998.

20) Person. "I2C Communication Protocol: Understanding I2C Primer, PMBus, and SMBus." *I2C Communication Protocol: Understanding I2C Primer PMBus and SMBus | Analog Devices*, https://www.analog.com/en/analog-dialogue/articles/i2c-communication-protocol-understanding-i2c-primer-pmbus-and-smbus.html.

21) Peňa, Eric, and Mary Grace Legaspi. "UART: A Hardware Communication Protocol Understanding Universal Asynchronous Receiver/Transmitter." *UART: A Hardware Communication Protocol Understanding Universal Asynchronous Receiver/Transmitter | Analog Devices*, https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html.

22) Table 11: NCR18650B 3400mAh Li-ion battery specifications: https://www.imrbatteries.com/content/panasonic_ncr18650b-2.pdf.

23) DW01A Datasheet. https://datasheet.lcsc.com/lcsc/1901091236_PUOLOP-DW01A_C351410.pdf.

24) *HY2213 Data Sheet*. https://datasheet.lcsc.com/szlcsc/HYCON-Tech-HY2213-BB3A_C113632.pdf.

25) "INJOINIC IP6505T." www.lcsc.com, https://www.lcsc.com/product-detail/DC-DC-Converters_INJOINIC-IP6505T_PRO_C515677.html .

26) *" WIDE-INPUT-RANGE STEP-DOWN SWIFT™ CONVERTER." Texas Instruments, https://www.ti.com/lit/ds/symlink/tps5420-q1.pdf?ts=1682189304915&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FTPS5420-Q1 .*

27) "WEBENCH® Power Designer." *WEBENCH® Power Designer | Overview | Design Resources | TI.com*, https://www.ti.com/design-resources/design-tools-simulation/webench-power-designer.html.

28) Woodford, chris. "How Do Fingerprint Scanners Work." Explain That Stuff, 17 Mar. 2022, https://www.explainthatstuff.com/fingerprintscanners.html#:~:text=The%20scanner%20uses%20a%20light,turn%20it%20into%20a%20code.

29) Heikkila, Tuomas. "Design Philosophy in Printed Electronics Manufacturing." Altium, 3 Dec. 2022, https://resources.altium.com/p/design-philosophy-printed-electronics.

30) Peterson, Zachariah. "Top 5 PCB Design Rules You Need to Know." Altium, 3 Dec. 2022, https://resources.altium.com/p/pcb-layout-guidelines.

31) Singh, Arjun. "Breakthrough in the World of Fingerprint Sensors: Optical vs Capacitive." Mantra Softech India Pvt Ltd, https://blog.mantratec.com/Optical-vs-Capacitive-Fingerprint-Sensors.

32) Agnihotri, Nikhil. "What Are the Different Types of Fingerprint Scanners?" Engineers Garage, https://www.engineersgarage.com/different-types-fingerprint-scanners-optical-capacitive-ultrasonic-in-display/.

33) Tyson, Jeff. "How Lcds Work." HowStuffWorks, HowStuffWorks, 17 July 2000, https://electronics.howstuffworks.com/lcd.htm.

34) Woodford, Chris. “How Oleds (Organic Leds) Work.” Explain That Stuff, 1 Sept. 2022, https://www.explainthatstuff.com/how-oleds-and-leps-work.html.

Appendix A

Figures & Tables references & permission

1)

Table 14: Default BMS parameter settings:

https://www.rec-bms.com/datasheet/UserManual9R.pdf

Permission email:

Re: Copyright Permission Request

MA **Maja Pozar Andrejasic <maja@rec-bms.com>**
11/9/2022 2:40 AM

To: Idris Saif Salim Al-Shidhani

Dear Idris,

Sure. We are giving you the permission to use the photos from our manuals for your student project.
Thanks for asking.

Good luck with your project.

Best regards,
Maja Pozar Andrejasic, CEO
--
Think before printing!

REC d.o.o.
Novi trg 9
SI - 6230 Postojna
----
e-mail: info@rec-bms.com
web: www.rec-bms.com

On Tue, Nov 8, 2022 at 11:18 PM Idris Saif Salim Al-Shidhani <Idris.alshidhani@knights.ucf.edu> wrote:

> Hello,
>
> My name is Idris Al-Shidhani, I am an electrical engineering student at University of Central Florida. I am working on my senior design project and I wanted to use a figure of the default specifications of **BATTERY MANAGEMENT SYSTEM 4 – 15S** that I attached with this email for one time usage, which I know is copyrighted. I am asking for a permission to use it in my senior design report.
>
> The website that I took the figure from is:
>
> https://www.rec-bms.com/datasheet/UserManual9R.pdf
>
> Thank You