The Study Buddy - A Multi-Factor Timed Lockbox

Cliff Porter, Kaiden Duvall, Denzel Stockhausen, and Idris Al-Shidhani

Dept. of Electrical and Computer Engineering, University of Central Florida, Orlando, Florida

Abstract — **This paper describes the design process used to develop the Study Buddy, a locker that uses a timer in addition to multiple factors of authentication configured by a user to securely store objects placed within. The locker contains a keypad for passcode entry, as well as an RFID reader and a fingerprint sensor as additional methods of entry, covering all three main factors of authentication. A remote was also created that communicates with the locker over Bluetooth Low Energy to allow a user to monitor the locker's condition within a certain range.**

Index Terms — **Authorization, RFID, User interfaces, Charging devices, Battery management, Bluetooth Low Energy**

I. Introduction

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.

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.

Many current commercial products that approach this issue exclusively target smartphones. We 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 have created a product with several possible uses, including self discipline, parental control, and storage. The larger size of our device in comparison to available phone lock boxes additionally allowed us to create a more flexible device that includes many methods of access that can be configured by the user.

We 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, or an RFID scanner, all built into the device and configurable via a display. This "Study Buddy", as we are calling it, also contains door sensors that detect if the container has been opened forcefully or prematurely and log the 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.

II. Specifications

The locker specifications chosen to be the base requirements of our project can be summarized in the following list:

1. Be easy to cary and weigh less than 10 lbs when empty
2. Hold 15 lbs without damage
3. Able to store a 15 in laptop, phones, and game controllers in separate compartments
4. Locker can run without power for ≥ 2 hours
5. Remote has a battery life ≥ 7 days
6. Locker runs off wall AC and battery power with ≥ 80% conversion efficiency
7. Wireless range of ≥ 15 ft
8. Wireless latency ≤ 400ms
9. Detect and notify user of unauthorized opening with ≥ 90% reliability
10. Keypad and RFID input reliability ≥ 90%

We also identified the following advanced requirements and a stretch goal.

Advanced Requirements:

1. Extend wireless range to 40 ft
2. Provide 3 5V 2A USB charging ports for users to charge their devices
3. Add a fingerprint sensor as a 4th configuration option with ≥ 80% reliability

Stretch Goal:

1. Detect when the user might need emergency access to their phone by detecting repeated phone calls and unlock or by receiving the emergency unlock signal from the remote

III. SYSTEM OVERVIEW

Our project consists of two components: first, a main locker unit with multiple compartments for storing common electronic devices, a user interface panel, and a battery backup and second, a small remote with a display, buttons and a low power consumption for long battery life.

The locker's physical design consists of one wide compartment that can fit most laptops or other long and slim devices, a small compartment meant to store smartphones or other similarly sized devices and a third compartment sized to fit other, more bulky devices such as game controllers, handheld consoles or other personal items. It is made of cut plywood sheets to ease construction complexity and a 3D printed face plate to mount all of our user interface peripherals. Electronics will be housed in the right of the locker, behind the user interface.

Figure 1: The main locker unit

Our remote has a simple design consisting of a display, 4 buttons, a AA battery pack in a 3D printed case. Its purpose is to provide remote monitoring of the state of the locker so users could check the time remaining without having to interact with the main locker and to provide emergency unlocking capabilities to override locks incase the user needs emergency access to their device or they lock themselves out of a compartment without the ability to unlock it (like in the case of a forgotten pin or lost RFID tag).

Figure 2: Remote

Figure 3 shows the final design of the main PCB for our locker which includes the various power regulators on the upper half of the board. Also included are the usb power chargers located around the lower left corner of the board. Finally, the nrf52840 microcontroller module, GPIO expander and all connections to the various peripheral devices used in our project are located around the lower right corner of the board.

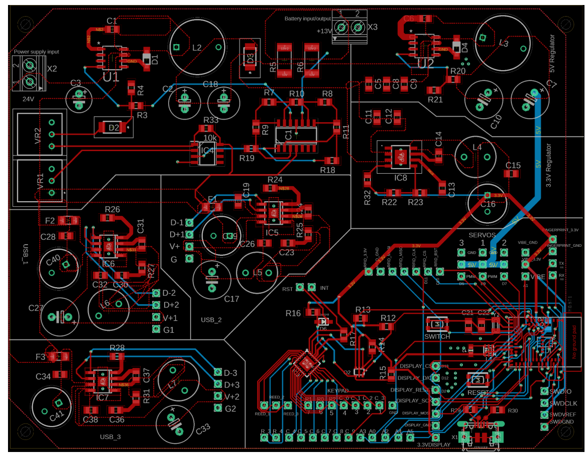


Figure 3: Main board PCB final design

IV. SYSTEM COMPONENTS

Figures 4 and 5 show the hardware block diagrams of our locker and remote, which shows our power management system in green, peripheral devices in red and control/signal devices in purple. This section will provide details on each of the components involved in our project

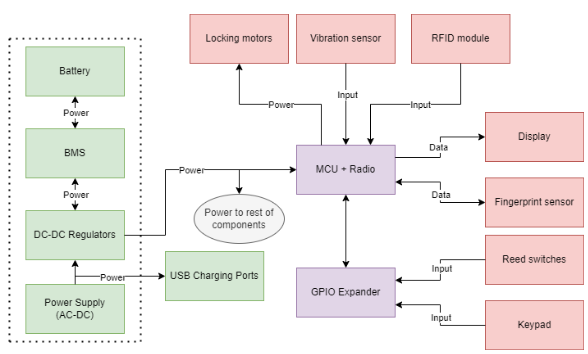


Figure 4: Locker Hardware Block Diagram

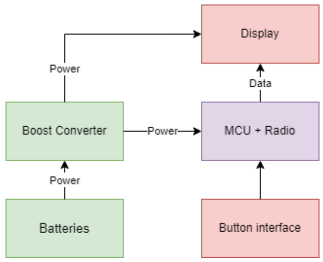


Figure 5: Remote Hardware Block Diagram

Microcontroller and Radio:

Our project uses an nRF52840 microcontroller module and uses Bluetooth Low Energy for communication.

Our microcontroller selection was largely influenced by our wireless requirement, so we compared multiple wireless standards and MCU platforms for their range, power consumption and most importantly Arduino IDE support. Due to our simple requirements for data transfer, we chose to use Bluetooth Low Energy as our wireless standard for its low power usage and the availability of premade libraries for communicating between two microcontrollers using UART over Bluetooth Low Energy.

In our research, we found the bluefruit lineup from Adafruit, an nRF52840 based series of microcontrollers which fit all of our needs for simple wireless using the Arduino IDE. It also used an MCU module with an embedded antenna which was another plus as we did not want to deal with radio signal design and interference issues.

GPIO Expander:

Our keypad and reed switches require 10 GPIO pins combined while only being simple push buttons. These devices filled up too many of our microprocessors GPIO pins to fit all of our required peripherals so we added a TCA8418 I2C GPIO expander to save these pins by only using the hardware I2C bus on our microcontroller.

User Interface:

Our Locker has a user interface panel with several devices to interact with the user which includes:

The screen on the front panel of the locker, which uses OLED technology and is used to display menus and other information to the user. The screen used on the remote also uses OLED technology and is used to display the status of the locker to the user.

The fingerprint sensor used was a capacitive one and was included as one of our advanced goals. Its use in our design was to register the user's fingerprint as a method of unlocking one of the compartments.

A 3x4 keypad for the user to interact with our user interface by using it to navigate through the menus and register a password as a method for unlocking the locker.

An RFID reader which is used as a method of unlocking the locker by allowing the user to register a tag to unlock one of the compartments.

Sensors:

Our project involves using two sensors to detect the state of the Locker, a vibration sensor and three reed switches.

We are using three reed switches attached near the doors to each compartment to sense the open/closed status of each door using a magnet. We use this state information to check that the door is closed before locking the compartment and as means to detect if the compartment has been opened without authorization by some physical means.

The vibration sensor is located in the small compartment meant for storing smartphones and is meant to detect the vibrations from a ringing phone. We use this sensor to detect if a phone is receiving repeated calls, which we identified as our criteria for an emergency situation when the user would need immediate access to their phone.

Locking Mechanism:

To physically lock the compartments we use servo motors attached to the doors of each compartment that actuate a sliding bolt into a slot in the wall of the compartment to secure the door.

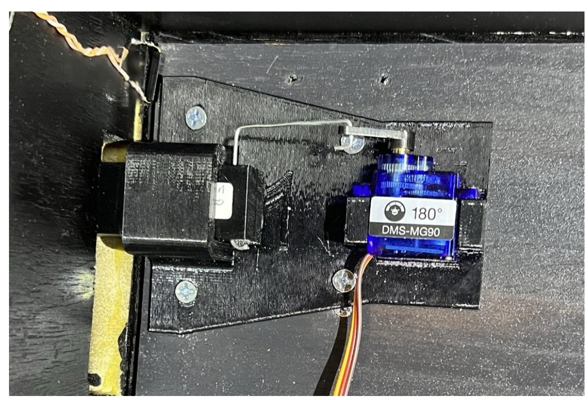


Figure 6: Locking mechanism

Power System:

Since our project has unique power delivery requirements such as a battery backup and high power usb ports, we designed a custom power sub system to meet our needs. Details of our design are in sections XIII and IX.

V. SYSTEM SOFTWARE

The main software code for the Study Buddy was written in the Arduino IDE, partially using available libraries for the RFID reader, fingerprint sensor, servo motors, and GPIO expander. The primary code employs the use of a state machine, with each state displaying one of the menus on the OLED display and looping through checks for user input. In addition, the code constantly checks the remaining time on each timer, the state of each reed switch, and the input from the remote to update the lock states and tamper detection. If call detection is enabled, this constant check also tracks the number of detected vibrations to unlock compartment 2 if a certain threshold is reached.

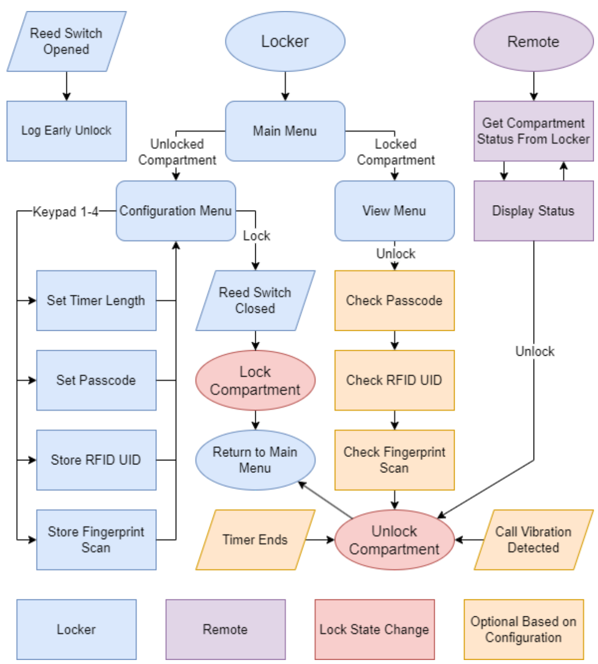


Figure 7: Block diagram showing software functionality and user input

The state machine functionality of the software is accomplished by declaring each state in the main loop and calling a state transition whenever an expected input is received. Each primary menu screen and its functionality constitutes one state, as well as every screen corresponding to an unlocking method, such as the screen prompting the user to scan an RFID tag. The user pressing the star key to go back causes the code to exit from the current state back to the state it had transitioned from.

When ordering our final PCBs, we obtained blank versions of the nRF52840 chips that, on their own, would be unable to run our Arduino code. To allow our code to function on the locker and remote PCBs, we loaded an Arduino-compatible bootloader via a J-link debugger before uploading the functional code.

VI. USER INTERFACE

The user navigates the locker's menu system primarily through keypad input that is prompted on each screen. In general, the hash key is used to confirm or continue, and the star key is used to cancel or go back. The 'Main Menu' of the device is a screen showing three lock icons numbered 1-3 that are locked or unlocked according to their corresponding compartment.

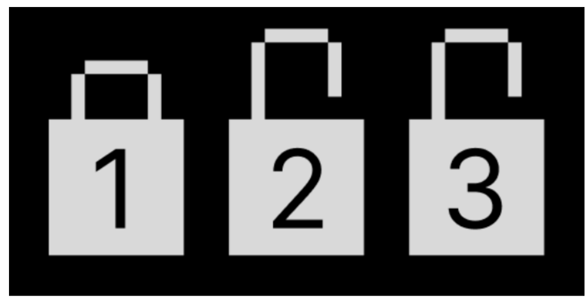


Figure 8: Locker Main Menu Screen

Pressing keys 1-3 will bring the user to a menu corresponding to that compartment, the content of which depends on the state of the lock. If the compartment is unlocked, the user is brought to the configuration menu, where they can press keys 1-4 to enable or disable the timer, passcode, RFID, and fingerprint entry methods. After enabling one or more of these and closing the door, the user can lock the compartment. If the compartment is locked when accessed from the main menu, the user will see the 'locked' screen, which displays the current time remaining if the timer is enabled as well as the option to unlock manually.

In addition to the above primary menu screens, there is an additional 'Settings Menu' the user can access by pressing star on the main menu. On this screen, the user can enable/disable the call detection unlock via the vibration sensor in compartment 2 while the compartment is unlocked. Below this option are lines indicating whether or not the tamper detection was triggered for each compartment, with an 'X' indicating the reed switch read as open while that compartment was locked.

VII. Wireless Communication

To communicate between our locker and remote we are using a UART over Bluetooth Low Energy implementation available in the Bluefruit libraries provided by Adafruit. These basic libraries provide the code required to send and receive string messages between a BLE central (our locker) and BLE peripheral device (our remote).

We took this example code and first modified it to send strings with variable information then designed a simple character based communication protocol for sending data between our devices.

For communications from the locker to the remote, we need to transmit the locked status of each compartment, including which factors it uses and how long is left on the timer.

Info			Locking Factors				Time Remaining			
L	#	l/ul	T	P	R	F	Hours		Minutes	
L	1	1	1	0	1	0	0	2	5	3

Table 1: Locker to Remote communication protocol

First, we transmit the character L to confirm that we are sending from the locker to the remote. Next we transmit a 1, 2 or 3 to identify which locker we are sending information about and its locked status. The next 4 characters are binary indicators of which locking factors are being used in the order of Timer, Pin, RFID, Fingerprint sensor. A value of 0000 would indicate that that compartment is currently unlocked. Last is the 4 characters of the timer displayed in the HH:MM format, with the highest value being 99h 59m.

As an example, the communication shown in Table 1 is indicating that locker compartment number 1 is currently locked using the timer and RFID factors and that the time remaining is 2 hours and 53 minutes.

Communication from the remote to the locker is much simpler as we only need to transmit the emergency unlock signal.

Info		
R	#	Ul
R	2	U

Table 2: Remote to Locker communication protocol

We can see in Table 2 that our remote protocol is very simple. Like the locker protocol, we first transmit an R to identify which device is sending, then the locker compartment number. Last we also transmit the character U to confirm we are sending the emergency unlock signal. The example in Table 2 is sending the unlock signal for locker compartment number 2.

VIII. Power Subsystem

The locker's power system is divided into many sections that supply the control system and the locker compartments with needed power. We can see in the figure 9 below the sections' arrangement and connection to each other.

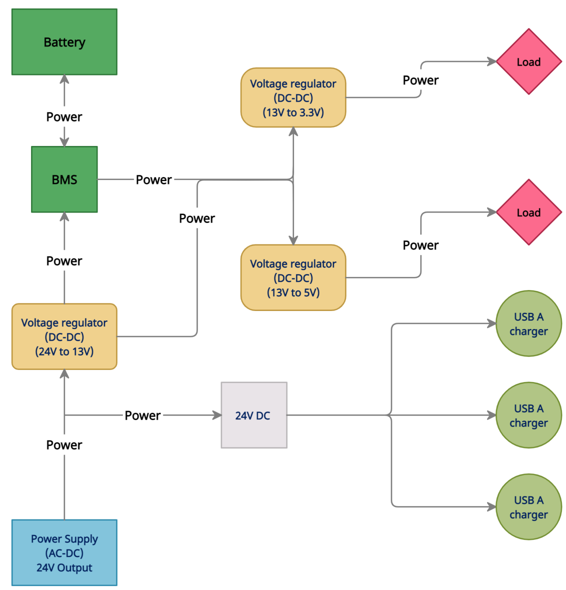


Figure 9: Block diagram showing all sections of power subsystem

The power supply is the main source of power to the entire main locker and also responsible for charging the secondary power source for the control system. For the specific power needs of all the components working at the same time, the total power consumption added with losses reached around 94 watts in calculations. Adding 30% additional watts for a good measure we have around 122 watts, so we chose a AC to DC power supply that can handle around 130 watts, but we had an LED power supply already that can handle 150 watts, which was more than enough for our project.

Voltage regulators are the perfect solution for controlling the voltage level to use it for different purposes without losing much efficiency. In our power design we needed three voltage levels, one to charge the battery pack with around 13V, one to power the servo motors with 5V, and one to power the rest of the control system with 3.3V. Using Webench designing tool from Texas Instruments to help us design the regulating circuit using a chosen ICs which were (TPS5420-Q1) for 13V and 5V outputs, and (LMR33610) for 3.3V output From Texas Instruments.

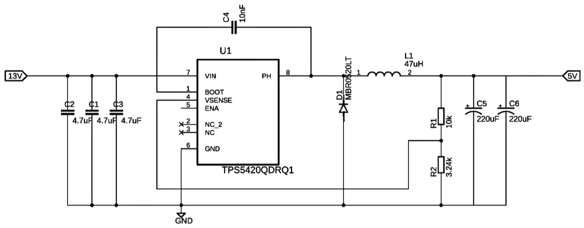


Figure 10: 5V regulator circuit. Credit: Webench [1]

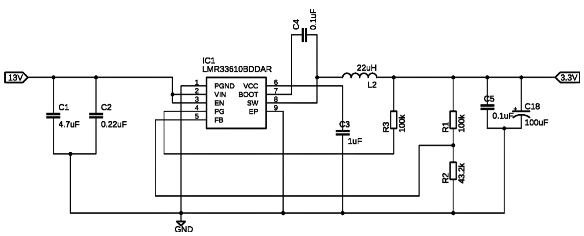


Figure 11: 3.3V regulator circuit. Credit: Webench [1]

To meet our specifications we had to have a secondary power source that would be the backup if the AC wall power goes out, or the user wanted to put the locker in a place without power for some time. We chose 18650s li-ion battery cells to make a battery pack with 11.1V nominal voltage and a capacity of around 3400mAh. This battery pack is going to supply the control system with power through the 5V and 3.3V outputs voltage regulators for long periods without main power. One of the challenges was to limit the current going to the battery pack while it is charging to maximize its life expectancy and to protect the 13V output regulator IC from overheating, so we added a current limiting circuit using Op-Amps that is going to feed the sensing pin in the IC.

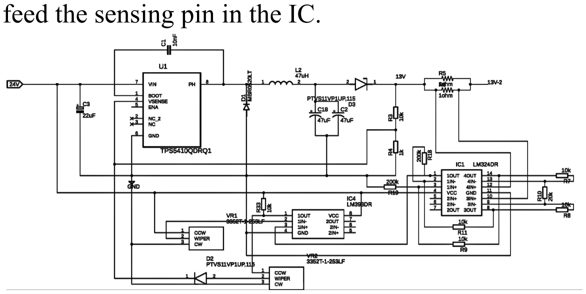


Figure 12: 13V regulator with Current limiting circuit. Partial Credit: Webench [1]

To protect the battery pack from different conditions like overvoltage, undervoltage, overcurrent discharge, and short circuit, we chose to get a Battery Management System (BMS) that is compatible with managing three cells in series (3S) and also has many features that we need. We chose the (HX-3S-FL25A) from (HX-3S-FLxxA) BMS series as it has most of the features we need.

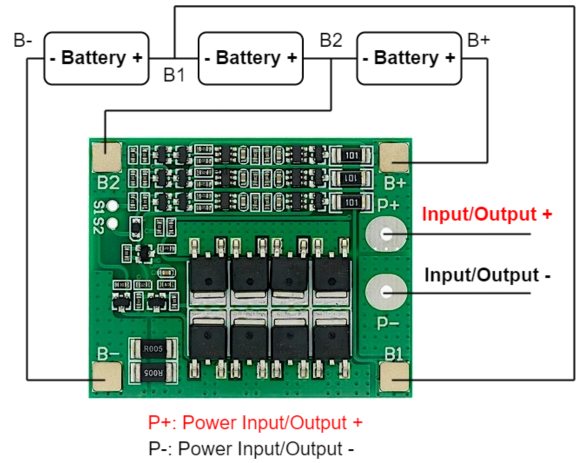


Figure 13: Battery pack wiring to the BMS

IX. USB PHONE CHARGING

For more convenient user experience while their electronics are inaccessible inside the locker, we decided to add USB charging wires to the three compartments with a female USB A end to give more flexibility to the private user. If the locker is going to be used in a public place, then we will provide wires that are connected to the female USB plug, but held tightly to prevent them from being stolen. The IC that we chose is the (IP6505T) from Injoinic Technology, which can provide Quick Charge (QC3.0 / QC2.0) to the connected device and communicate with it through the data pins (D+ / D-).

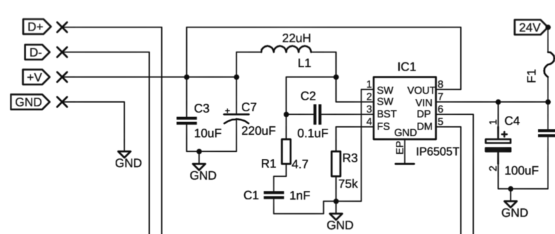


Figure 14: USB charging circuit. Credit Injoinic Technology [2]

To power the remote we chose to have two AA batteries as the source of power. However, the terminal voltage of two AA batteries in series is only 3V and might be lower, so we used an IC from Texas Instruments (TPS610981) to configure a boost converter circuit using webench to step up the voltage to 3.3V , which what we need to power all the remote components.

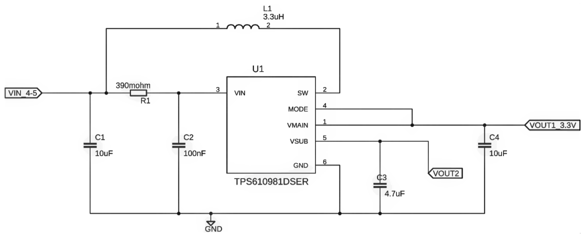


Figure 15: 3.3V Boost circuit. Credit: Webench [1]

X. Security and safety features

The Study Buddy's locker employs a few extra security and safety features beyond the standard lock and entry methods.

First is the use of reed switches positioned in the frame of each compartment door that align with magnets placed on the doors themselves. These reed switches serve two purposes in the security and functionality of the locker: ensuring the door is closed before locking and acting as tamper detection. If one of the normally open reed switches reads as open when the user is in the configuration menu, they will be prompted to close the door before they can enable the lock. This is to prevent a locking bolt from ever being extended while its door is open, which would prevent it from being closed for the lock duration. In addition, if the state of the reed switch changes while the lock is enabled, the locker logs this in the settings menu to inform the owner that the device was tampered with.

The other extra locker feature is the vibration sensor placed at the bottom of compartment 2 of the locker. If the user has enabled call detection in the settings menu, compartment 2 will unlock when the vibration sensor detects consistent vibrations for a long enough period of time. This serves to detect if a phone locked in the device is being called multiple times in a row, which may constitute an emergency where the phone is needed and the locker cannot be unlocked by the owner.

If an override of the locker's normal unlocking logic is needed, due to an emergency or a user being unable to unlock due to a forgotten passcode or missing RFID tag, the Study Buddy's remote serves as a master key to all of the locker's compartments. By selecting the desired compartment and pressing the 'unlock' button on the remote, the user is brought to a screen where they are prompted to press the unlock button again to confirm the override. If they do so, the chosen compartment will immediately unlock regardless of which unlocking methods are enabled.

XI. Design constraints & difficulties

One time-related constraint that impacted the difficulty of finishing this project was that our schedules did not often overlap making it difficult to plan meetings where everyone could be present. This limited the number of times we could meet each week to work on the project as well as the length of the meetings.

We also encountered some difficulties during the course of our project relating to faulty/damaged parts, design faults or other shortcomings that we had to learn about. The remainder of this section will detail some of these difficulties.

One of our major concerns going into this project revolved around our MCU choice. We chose the nRF52840 mainly due to its readily available and easy to use Bluetooth Low Energy libraries for the Arduino IDE. These were important to use as they would save us time having to re-implement code for BLE and all of our peripherals. The main catch was that using the Arduino IDE on the nRF52 platform required a special bootloader on the chip that was not present on the blank modules that we would have to use in our project. Our concern was that there was very little information on how to load the special bootloader onto our chips since it was outside the use case of buying the premade boards to use with the Arduino IDE. The supposed official way was to use the nRF52 Developers Kit, which included a J-Link device for programming external boards. We included the appropriate software debug pins as pads in our first pcb prototype but could not connect to it with our development kit. After lots of testing, we concluded that the J-Link on our development kit was not working correctly and ordered a proper J-Link which would work. Luckily our new J-Link worked properly and told us how to properly connect to our MCU. We also included more discrete headers for easier programming and successfully programmed our boards in the Arduino IDE.

After assembling our PCB, some of our peripherals wouldn't work and we had intermittent difficulty connecting to our MCU to program it. We discovered that some of our GPIO pins were not receiving signals like we expected and were likely not connected. After resoldering the MCU we had even more connection problems and decided to remove the QSPI flash chip on the back side of the MCU, as it may have been overheated or been interfering with nearby traces and was not needed in our design since the nRF52840 had enough internal flash for our project. This fixed our connection issues but also allowed the dotstar LED to spontaneously turn on. We believe that the dotstar may have been wired wrong and damaged the adjacent

GPIO pins inside the MCU as they no longer worked. We decided to remove it as it was not functioning as we expected it to and only increased our power consumption. After replacing the MCU once more, we successfully confirmed the functionality of all of the locker's peripherals.

The 13 volts output regulator's current limiting circuit design should have had a pull up resistor to enable the comparator chip LM393 to pull the output to high-state when monitoring the voltage across the sensing resistors. We have changed the final PCB design to include the pull-up resistor, but we did not have time to order a new PCB, so as a fast and practical solution we have added a through hole resistor by soldering it to two of the potentiometer legs which are connected to Vcc and Vout of the comparator.

XII. Conclusion

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 for 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.

Acknowledgement

We would like to acknowledge professors Lei Wei and Samuel Richie for their oversight and assistance during the course of our project, the many professors and staff of UCF and the Department of Electrical and Computer Engineering that have guided us during our education, and Steve Porter for assisting us with the construction of our locker.

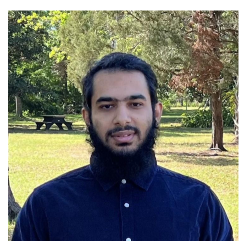
Biography

Cliff Porter is a senior pursuing a bachelor's degree in Computer Engineering at the University of Central Florida expecting to graduate in Spring 2023. He intends to pursue a career in robots or consumer electronics near Orlando after graduating.

Kaiden Duvall is a senior studying Computer Engineering at the University of Central Florida, graduating with a bachelor's degree in Spring 2023. He intends to pursue a career focusing on software in the Orlando area after graduation.

Denzel Stockhausen is a senior student at the University of Central Florida and plans to graduate in May of 2023 with a bachelor's degree in Electrical Engineering. He plans to pursue a career somewhere in the state of Florida.

Idris Al-Shidhani is currently a senior student at University of Central Florida, and he plans to graduate in May of 2023 with a Bachelor of Science in Electrical Engineering. He plans to continue studying power electronics to develop reliable power control designs for future power storage technologies and power harvesting.



References

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

[2] "INJOINIC IP6505T." www.lcsc.com, https://www.lcsc.com/product-detail/DC-DC-Converters_INJOINIC-IP6505T_PRO_C515677.html.