

Automatic Secure Locker

By

Chen, Yanlin

Liu, Jiaxuan

Final Report for ECE 445, Senior Design, Spring 2019

TA: Thomas Furlong

Professor: Jing Jiang

May 1, 2019

Project No. 28

Abstract

This report presents the detailed motivation, objectives, design, development and verification processes behind our Automatic Secure Locker project. This project aims at building a locker system that satisfies an individual household's need to safely receive their incoming packages. Other than text descriptions of each module in this project, calculations, diagrams, and tables will be provided to make readers better understand this project.

Contents

1. Introduction.....	5
1.1 Background and Objective.....	5
1.2 High-Level Requirements.....	5
2. Design.....	6
2.1 Power	
Supply.....	7
2.1.1 Solar Panel.....	7
2.1.2 Battery	
Charger.....	8
2.1.3 Lithium-Ion Battery.....	9
2.1.4 Voltage	
Regulators.....	9
2.2 Control Unit.....	11
2.2.1 ESP8266 WiFi	
Module.....	11
2.3 Control Panel.....	16
2.3.1 LCD.....	16
2.3.2 Camera.....	16
2.3.3 Buzzer.....	17
2.4 Locker Module.....	17
2.4.1 Locker.....	17
2.4.2 Servo.....	18
2.4.3 Hall Effect	
Sensor.....	18
2.5 Server.....	20
3. Design	
Verification.....	21
3.1 Power	
Supply.....	21
3.1.1 Solar Panel.....	21
3.1.2 Battery	
Charger.....	21
3.1.3 Lithium-Ion Battery.....	22
3.1.4 Voltage	
Regulators.....	22
3.2 Control Unit.....	23
3.3 Control Panel.....	26
3.3.1 LCD.....	26
3.3.2 Camera.....	26
3.3.3 Buzzer.....	26
3.4 Locker Module.....	26
2.4.1 Locker.....	26

2.4.2 Servo.....	27
2.4.3 Hall Effect	
Sensor.....	27
3.5 Server.....	28
4. Costs.....	29
4.1 Parts.....	29
4.2 Labor.....	29
5. Conclusion.....	30
5.1 Accomplishments.....	30
5.2 Uncertainties.....	30
5.3 Ethical considerations.....	30
5.4 Future work.....	31
References.....	32
Appendix A Requirement and Verification	
Table.....	34

1. Introduction

1.1 Background and Objective

With the rapidly growing e-commerce industry and the increasingly maturer logistics system, more people choose to shop online for their daily needs. However, although shopping online spares people the inconvenience of going to physical stores, the risk of losing packages also becomes higher. Such a problem is more pervasive in apartments that are not equipped with a package reception system. In this case, packages are usually left at the residents' front doors, which greatly increases the risk of packages being stolen by random people walking by.

In order to eliminate this kind of package thefts, we want to design an affordable automatic secure locker that can effectively help people securely store their packages and can be put right next to their front doors. This system is constituted by an electrically controlled locker, a control panel with an LCD display, a camera and a piezo buzzer, a control unit that wirelessly connects to the internet and coordinates the operation of every other module, a server, and a power supply. When there is no package stored inside, the locker remains open until a courier puts a package inside and closes the door. During this process, the LCD display will give the courier instructions on how to use the locker. Once the door is closed by the courier, the locker automatically locks itself to prevent someone else from accessing the package inside. To retrieve the package in the locker, the owner can unlock the locker remotely from his/her cell phone or PC. Although the mechanical lock is secure enough in most cases to keep the package inside safe, in order to prevent anyone from violently damaging the locker to steal the package inside, the security module of the locker will be triggered if the locker is open when the locker is supposed to be locked. The camera will automatically take a picture of the intruder and send it to the user over the internet; the piezo buzzer will make a loud noise to scare the intruder away.

1.2 High-Level Requirements

- The locker must remain locked until the microcontroller receives an instruction from the owner to open the locker.
- If the hall effect sensor detects a door opening when the locker is supposed to be locked, the buzzer will sound and the camera will take a picture of the potential intruder and send it to the user over the local WiFi.
- The rechargeable Li-ion batteries must provide power to the system for at least 13 hours without the need for solar energy.
- The microcontroller (WiFi module) must be able to wirelessly send data with a speed of at least 5 Mbps.

2. Design

The entire automatic secure locker design can be divided into five individual modules. The power supply converts solar energy to electric energy and stores it in a lithium-ion battery to provide power to the entire project. The control panel serves as an interface for the user to interact with the locker. The security function is also integrated into this module. The control unit is the core of the entire project that controls and coordinates the operation of every other module. It is also responsible for connecting to external WiFi and uploading local data to the server. The locker module is where the packages are kept and the locking mechanism is implemented. The server is accomplished by a PHP code that takes the incoming image data and stores the image on the server. Fig. 1 is a high-level block diagram showing the interconnection between different modules of this project.

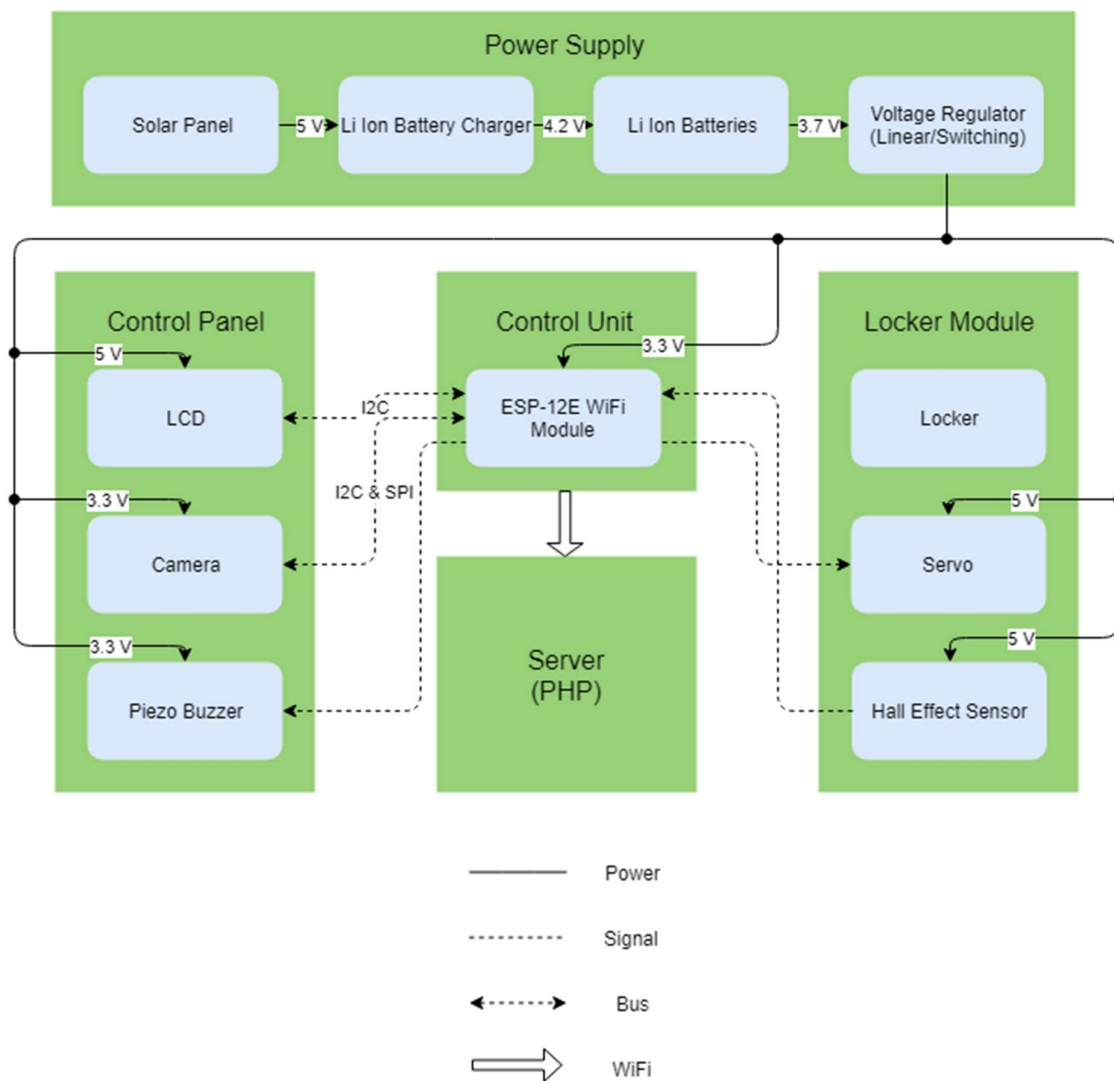


Fig. 1 High-Level Block Diagram

2.1 Power Supply

The power supply module consists of four parts: the solar panel, the Li-ion battery charging circuit, the rechargeable Li-ion batteries, and the voltage regulators. The solar panel is used to convert solar energy to electric energy that can be used by the electronics in our design; the Li-ion battery charger is used to regulate the varying voltage from the solar panel to the appropriate voltage level that the rechargeable Li-ion battery needs; the Li-ion battery is used as a buffer voltage supply that provides stable voltage to the locker; the voltage regulators include: a boost converter that steps up the 3.7 V from the Li-ion battery to the 5 V operating voltage of many electronic devices and sensors in our design; and a linear regulator that steps down the 5 V boost converter output voltage to the 3.3 V operating voltage of our ESP8266 WiFi module. Fig. 2 shows the connection of the four parts. Notice that, in order to prevent the power from flowing from the battery back to the solar panel in case there is not enough sunlight, a diode is placed between the solar panel and battery charger.

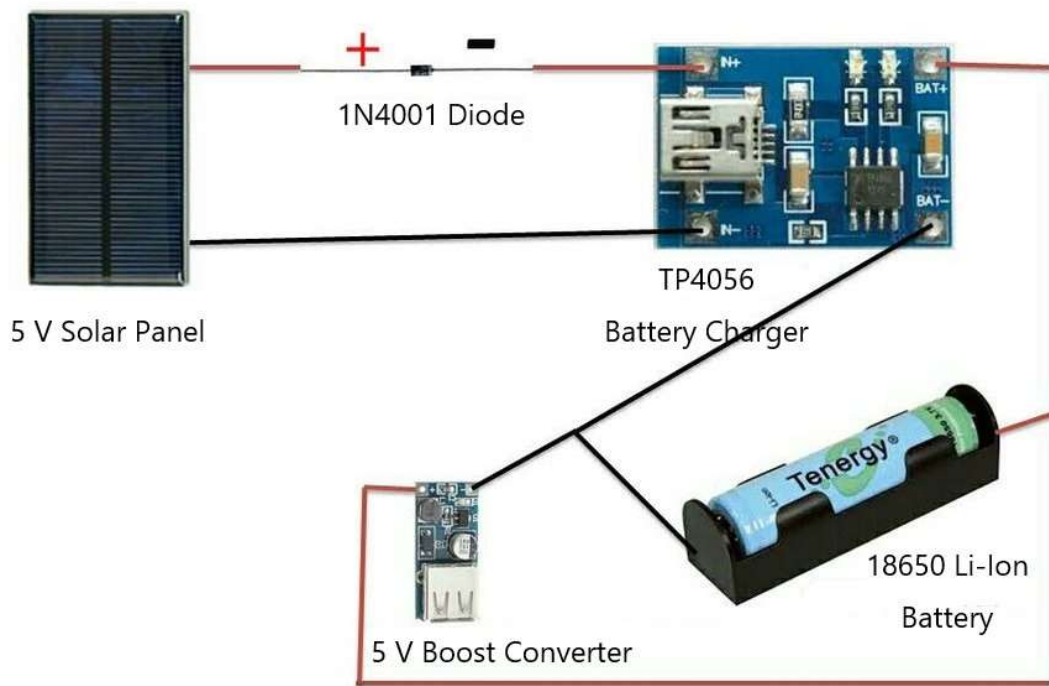


Fig. 2 Power Supply Connection [1]

2.1.1 Solar Panel

A piece of 2.5 W 5 V/500 mAh solar panel [2] is used to provide sustainable energy to the automatic secure locker. Considering the fact that our locker is designed to be placed outside the owner's apartment, using a power outlet to power the locker is not a reasonable option. Also, since we are using the ESP8266 WiFi module as the microcontroller of our design to do wireless communication, our locker has a considerable power consumption which is not efficient for non-rechargeable batteries to handle. Therefore, adopting solar energy to be our energy source becomes the best solution for us. A 5 V solar panel is chosen due to the fact

that the solar panel voltage is usually 1.5 times the battery voltage. Since we are using a 3.7 V Li-ion battery, using a 5 V solar panel is the most recommended setting. In addition, the 5 V lies well within the battery charging circuit input voltage range (4.5 V - 5.5 V). Other than the voltage requirement, the maximum current rating of this solar panel is 500 mA, which is much higher than the average current drawn by our locker (approximately 120 mA).

2.1.2 Li-Ion Battery Charger

The Li-ion battery charging circuit is essential to charge the Li-ion battery because the battery requires a stable charging voltage of 4.2 V. Any voltage higher or lower can cause an unwanted chemical reaction that leads to danger. Unfortunately, the output voltage of the solar panel is a varying value depending on the light intensity it receives. Therefore, the charging circuit is needed to provide a steady 4.2 V to charge the battery. In our design, we used the MCP73831 charge management controller IC as our battery charger. Fig. 3 and 4 show the schematic and PCB layout of the PCB designed by us. Note that R2 is a programming resistor that sets the charging current of the battery charger and the relationship between the charging current and the resistor value is:

$$I_{REG} = \frac{1000V}{R_{PROG}}[3]$$

In our case, we want to achieve the fastest charging speed, so we choose the programming resistor to be 2k Ohms to have a 500 mA charging current, which is the maximum charging current this IC can give.

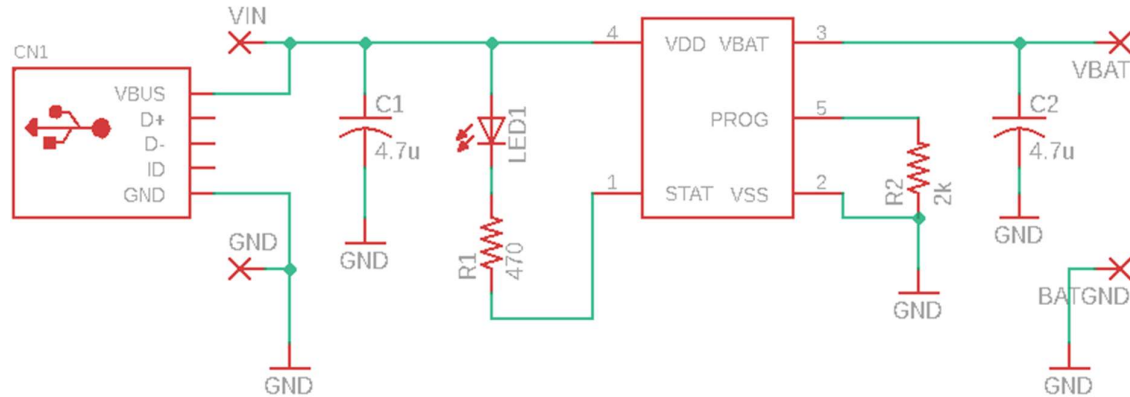


Fig. 3 MCP73831 Battery Charger PCB Schematic

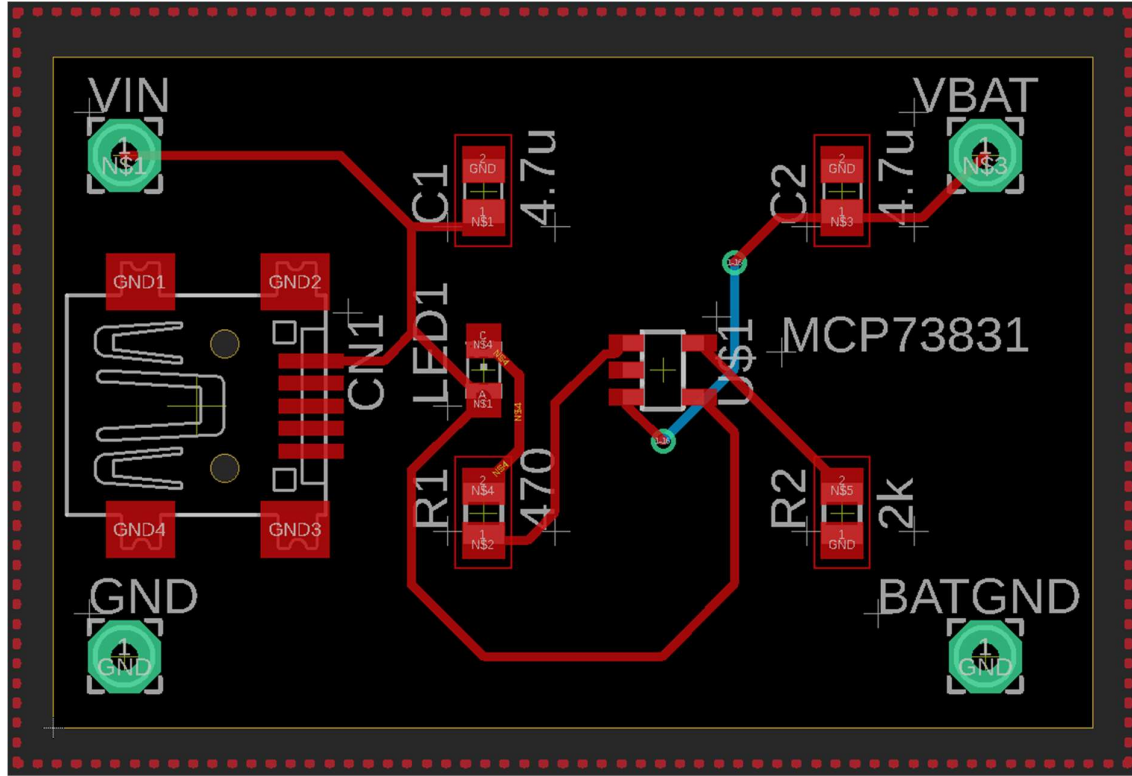


Fig. 4 MCP73831 PCB Layout

2.1.3 Li-Ion Battery

The 3.7 V Li-ion battery is the direct voltage source that powers the locker electronics. The battery capacity is chosen to be 3000 mAh to guarantee the electronics can be powered even without sunlight for more than 13 hours. In order to charge the battery, a constant 4.2 V voltage needs to be applied to the two ends of the battery [4].

2.1.4 Voltage Regulators

Two types of voltage regulators are chosen to provide two different levels of regulated voltages to electronic components used in our project.

The first voltage regulator is a 5 V switching boost converter that steps up the 3.7 V battery voltage to the 5 V operating voltage of some electronic components, including the LCD and the hall effect sensor. We want to use a switching regulator because of its very high power efficiency compared with the linear regulator. For our design, we used a L6920DB synchronous rectifier step-up converter IC [5]. This IC has an internal feedback control based on a comparator that continuously checks the status of the output voltage and adjusts the duty cycle to ensure a constant output voltage. Fig. 5 and 6 show the schematic and PCB layout of the boost converter PCB designed by us.

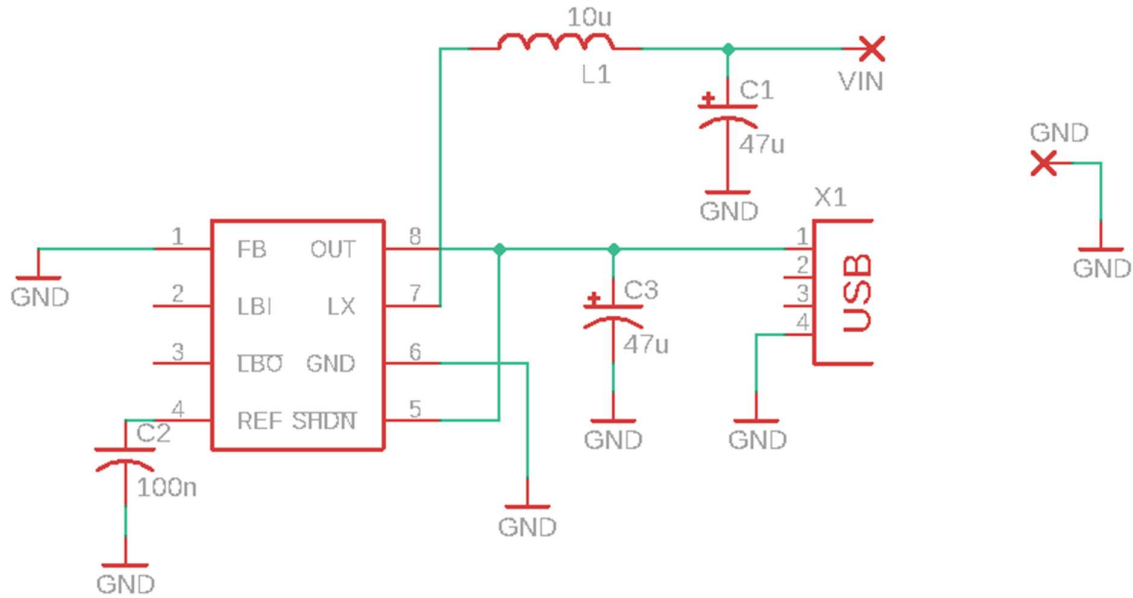


Fig. 5 L6920DB Boost Converter Schematic

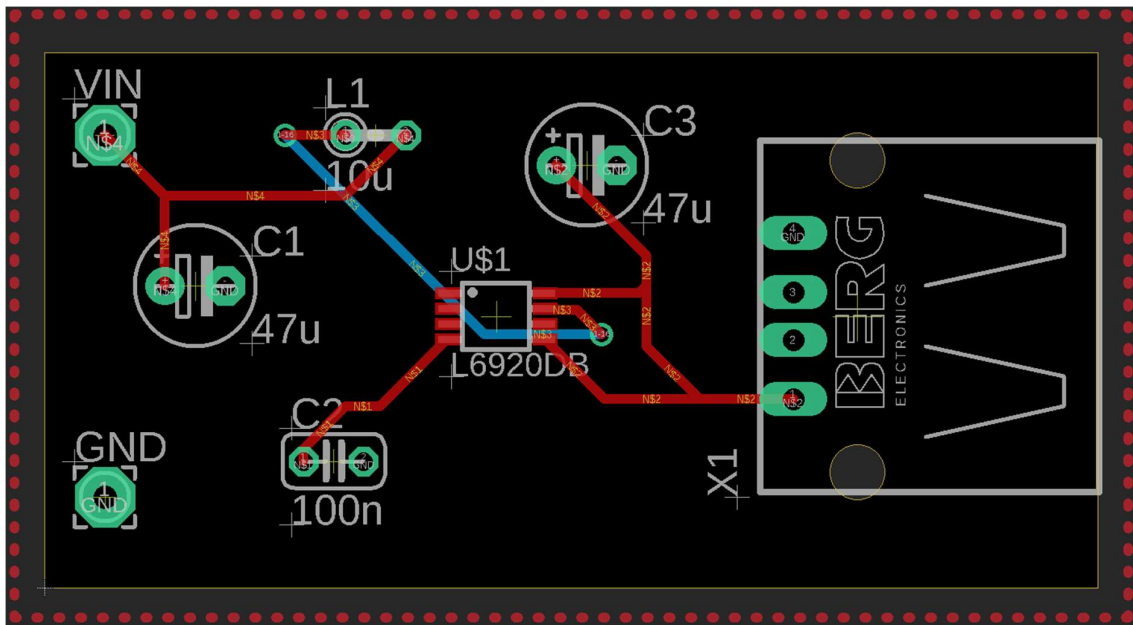


Fig. 6 L6920DB PCB Layout

The second voltage regulator we are using is a 3.3 V linear regulator that steps down the 5 V boost converter output to the operating voltage of our ESP8266 WiFi module. This linear regulator is placed on the ESP8266 microcontroller board, so its schematic and PCB layout are not presented in this section.

2.2 Control Unit

The control unit is the core of the entire design because it includes the microcontroller that controls the operation of all electronics used in our project. It is also responsible for sending data to the server over the WiFi. In this design, the ESP-12E module is used as both the microcontroller and WiFi communication tool.

2.2.1 ESP8266 WiFi Module

The ESP8266 WiFi module is the core of our project because it communicates with every other device to realize the functionality of our locker. The ESP8266 WiFi module not only serves as a portal to send and receive data over WiFi, but it also has a microcontroller that can be programmed in the same way as an Arduino development board. Specifically for our design, we chose to use the ESP-12E microchip of the ESP8266 series. It is known for its extraordinary capability, compatibility and reasonable price (< \$5).

To get a better understanding of how ESP-12E works, it is necessary to look at its pinout shown in Fig. 7.

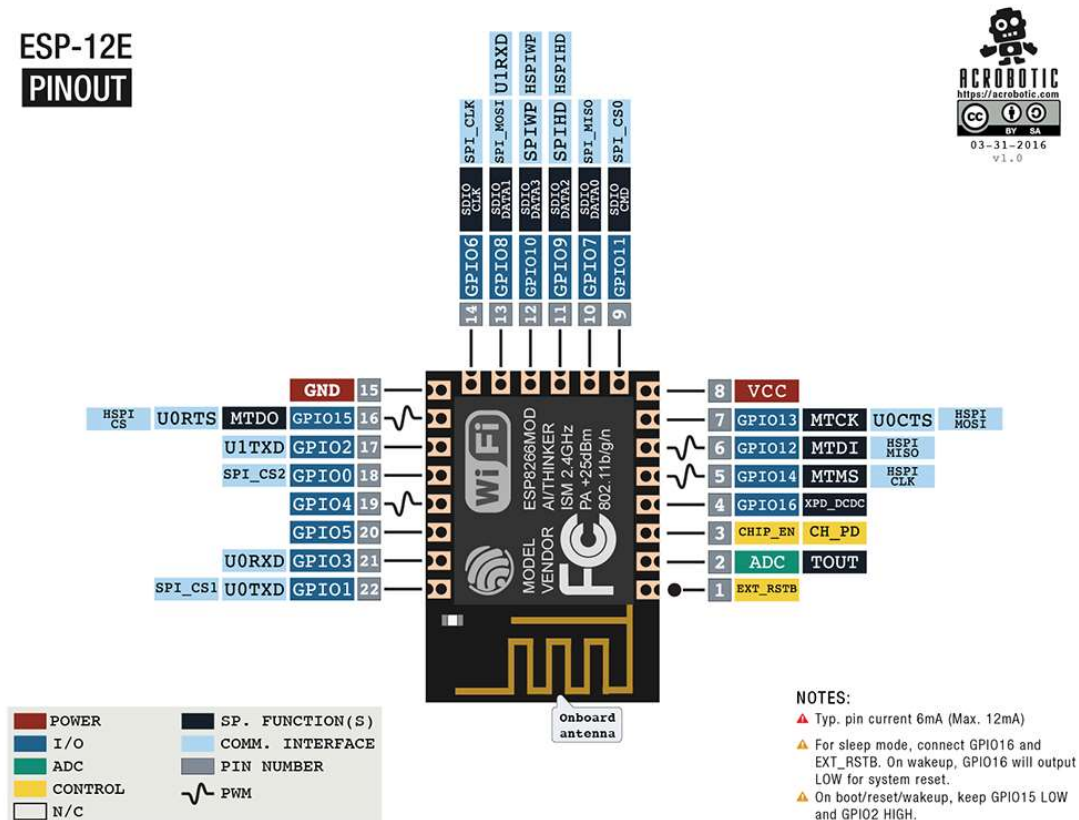


Fig. 7 ESP-12E Microchip Pinout [6]

The ESP-12E module has 17 GPIOs and 1 ADC. The amount of GPIOs provided seem to be more than enough for our need. However, one must notice that many of the GPIOs of the module have already been reserved for the chip boot-up and flash communication, which means that there is actually a very limited amount of GPIOs for free use. Table I shows the detailed pin assignment of ESP-12E in our project.

Table I: ESP-12E Pin Assignment

Pin	Usage
GPIO0	ESP-12E boot-up
GPIO1	TX / Buzzer
GPIO2	ESP-12E boot-up
GPIO3	RX / Servo
GPIO4	SDA of I2C
GPIO5	SCL of I2C
GPIO6 - GPIO11	Not usable, flash connection
GPIO12	MISO of SPI
GPIO13	MOSI of SPI
GPIO14	SCK of SPI
GPIO15	ESP-12E boot-up
GPIO16	CS of SPI (Camera)
A0	Hall effect sensor

Note that since the amount of usable GPIOs is limited, we choose to use TX and RX pins also as digital output to control the servo and buzzer because our project does not need the serial communication functionality except during the programming phase. A large number of usable GPIOs are used by the I2C and SPI protocols because our LCD uses I2C protocol and our camera uses both I2C and SPI protocols.

Another important feature of ESP-12E is its wireless communication function. It can be programmed as either a station to connect to an external access point to send data or an access point to be connected by other devices such as PCs or cellphones. In our project, since we need to use the ESP-12E module to send push notification and picture to the locker owner and use our cellphones to control the ESP-12E module to unlock the locker, we need the module to act as both the station and the access point.

Setting the ESP-12E as an access point is a relatively easy procedure because all we need to do is to use the ESP8266 WiFi library to set the ESP-12E chip to AP mode. When our

cellphone or PC is connected to the ESP-12E access point, we can establish communication between our device and ESP-12E. If we want to use our device to lock or unlock the locker by setting the position of the servo, all we need to do is to enter the IP address assigned to the ESP-12E module we are using and send certain commands to the module.

However, setting the ESP-12E as a station to send data to an external web server is much more difficult. In our project, our goal is to send the picture data stored in the camera buffer to our web server over the local WiFi. At the same time, the ESP-12E should also send a push notification to the user via a slack API bot. To understand how to accomplish that, the following two flowcharts may help.

When the image is captured by the camera, the image data is temporarily stored in the camera buffer. The ESP-12E WiFi module process the data by getting one byte of image data at a time from the buffer and concatenating the one byte of data to an existing string. When all image data is retrieved by the WiFi module, it is ready to send the data to the server. The server we are using has a regular HTTP port, therefore, the WiFi can just connect to the server and upload the image data with an HTTP post request. On the server side, there is a PHP code that takes the incoming string of image data and saves the image data in the form of JPEG files to a folder on our server.

Sending push notification to the user via the slack API is a similar process, but with one additional step. After the image being sent to the server, the ESP-12E WiFi module needs to prepare the image URL to be sent to the slack API server so that the user can view the image directly on the slack channel. However, since the slack API server only accepts requests from a secure HTTPS port. The ESP-12E WiFi module needs to set the CA certificate before doing any HTTP post request. Once that is completed, the WiFi module can now send data (image URL with a special token generated by our slack channel) to the slack API server. The slack API server will then automatically send the received information to our slack channel.

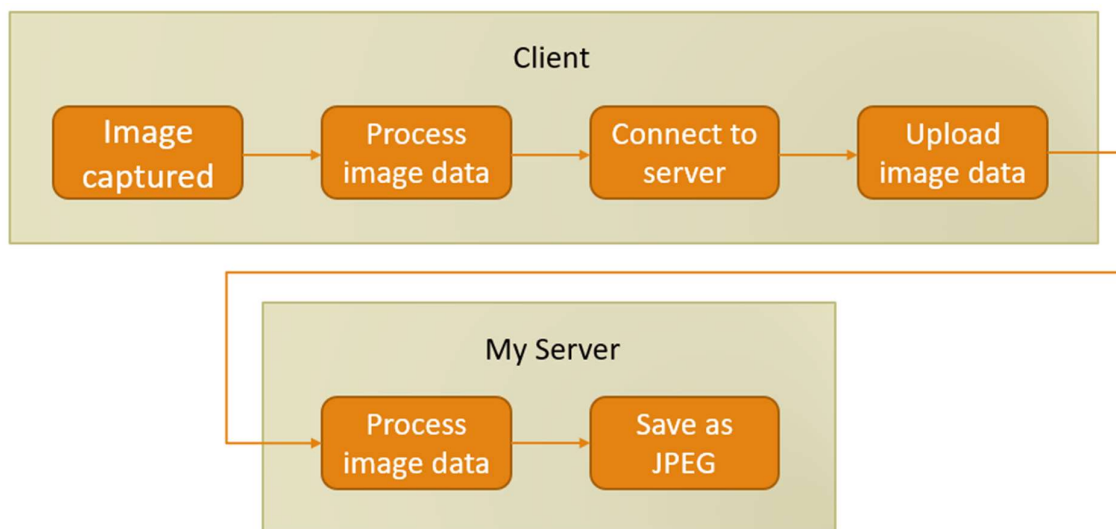


Fig. 8 Image Uploading Process

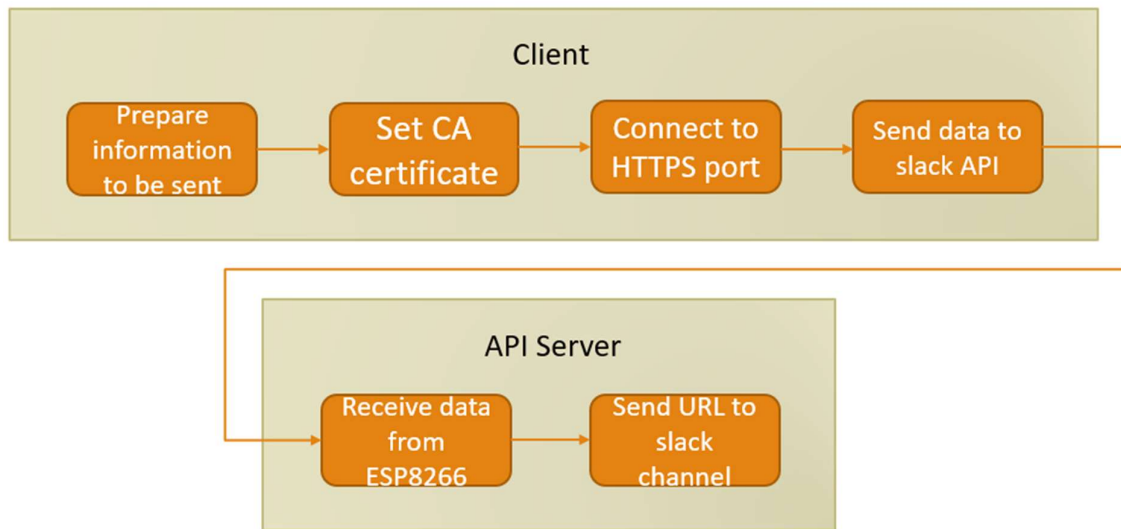


Fig. 9 Slack Notification Process

Now, we are done with the software side of the ESP-12E WiFi module setting. It is time for us to go back to the hardware side to see the design of the ESP-12E PCB. Fig. 10 and 11 show the schematic and PCB layout of our ESP-12E board. As shown on the PCB layout, the ESP-12E WiFi module, as the very core of the entire design, is placed on the left side of the PCB. On the right of the WiFi module is a 3.3 V linear regulator to provide the correct voltage to the WiFi module by stepping down the input voltage from the boost converter. Around the WiFi module, we have a couple of female headers for other electronic devices or sensors to connect to the WiFi module. One important thing to be noticed is that the USB mini B receptacle is only used for providing 5 V voltage to the PCB. One cannot connect the PCB with a USB cable to program the ESP-12E WiFi module. Instead, the programming of the ESP-12E module is accomplished by connecting an external FTDI RS232 Serial-to-USB adapter to the FTDI female header on the PCB. The 2x2 GPIO0 female headers are used to select the mode of the ESP-12E module. When the jumper wire is placed between the upper pair of the female headers, GPIO0 is connected to the ground and the ESP-12E WiFi module is set to programming mode; when the jumper wire is placed between the lower pair of the female headers, GPIO0 is connected to the 3.3 V level and the ESP-12E module is set to normal mode.

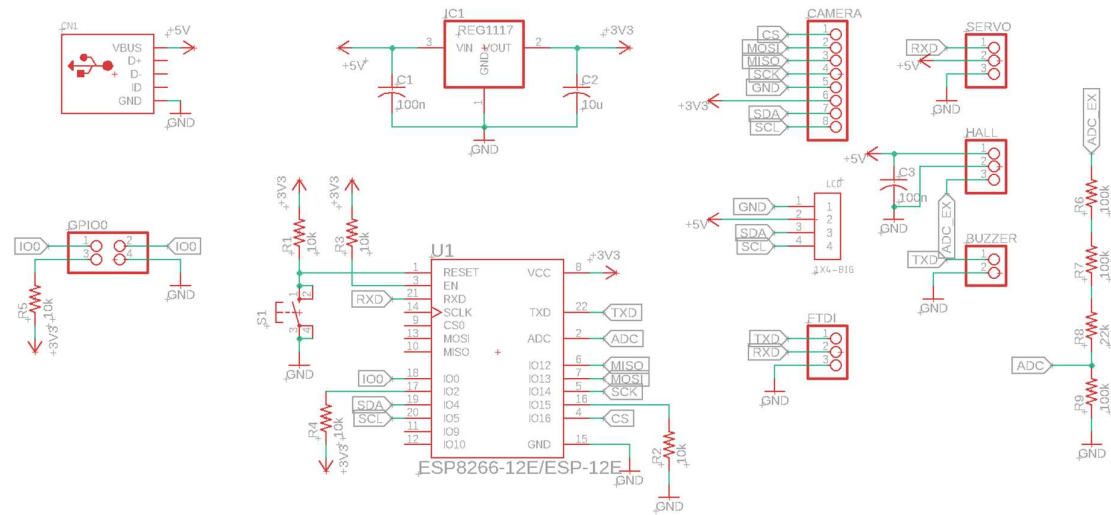


Fig. 10 ESP-12E Controller Board Schematic

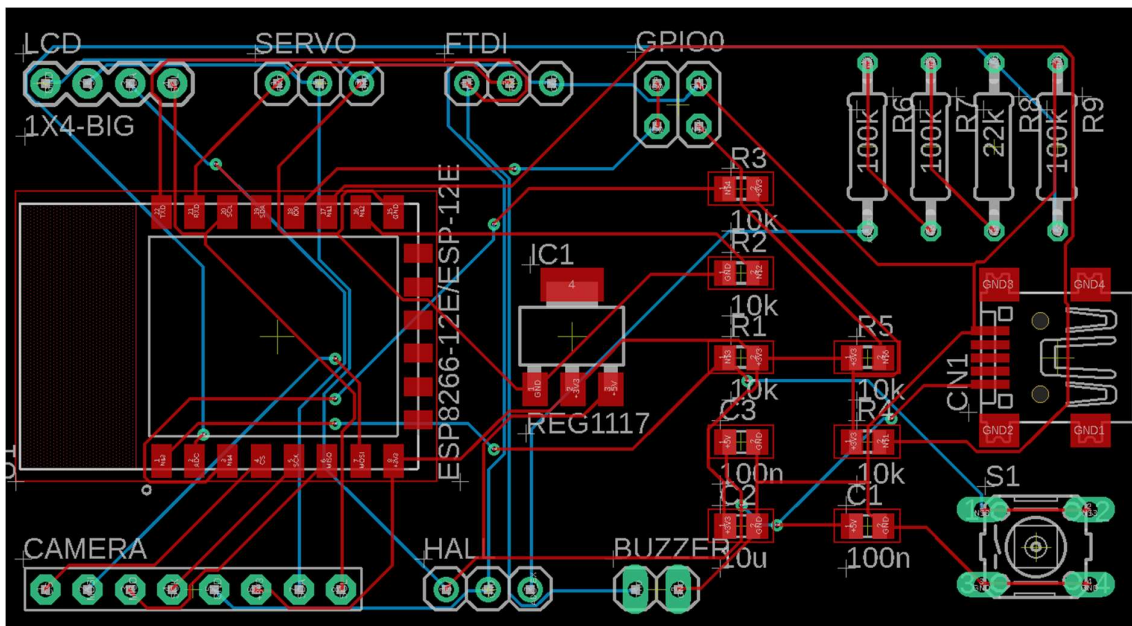


Fig. 11 ESP-12E PCB Layout

2.3 Control Panel

The control panel includes all the electronics visible to the user on the control panel box. It serves as an interface for the user to easily and conveniently operate the locker. The security module is also integrated into this panel. When a courier arrives, the LCD will instruct the user to place the package inside the locker. Once the courier does so, the camera will take a photo of the courier as a record and the LCD will notice the courier to shut the locker door. The piezo buzzer only sounds when the microcontroller detects an unexpected door opening when the servo should be in its locked position.

2.3.1 LCD

The 1602 LCD is used to display essential information to the courier to guide them on how to properly store the package inside the locker. The LCD we are using is equipped with an I2C breakout board, such that it can establish a connection with the microcontroller with only four pins: VCC, GND, SDA, and SCL. The microcontroller will send a signal to the LCD to display preset character strings to the courier. The LCD also has an LED backlight to make the characters on the LCD look clearer. Although for a 1602 LCD, only 32 characters can be displayed on the screen at a time, more information can be displayed to the user with the help of the scrolling function on the software side.

In order to control the characters displayed on the LCD, we used the `LiquidCrystal_I2C` Arduino library. In addition, the LED backlight will be turned off to save power after the package is stored in the locker because no further instruction is needed by the courier.

2.3.2 Camera

The ArduCam OV2640 2MP Camera [7] was chosen to be used as a camera to capture the face of the courier and possible intruder. This camera is powered by 3.3 V VCC and has both SPI and I2C protocol pins to communicate with the microcontroller. The SPI pins are used for the actual data transmission and the I2C pins are used for the camera configuration. Although this camera has both capturing and streaming functionality, we are only using its capturing function. To use the camera with the ESP8266 WiFi module, we used the official ArduCam Arduino library and example code distributed by ArduCam.

Normally, the camera will be set in the standby mode to save power. Once the door is closed by the courier or the door is unexpectedly open, the ESP-12E microcontroller will send a signal to the camera and the camera will capture a picture and save it to its onboard flash memory. The maximum picture quality this camera supports is 1600x1200; however, since we need to send the picture data over the WiFi, in order to save the time for data transmission, we set the picture quality to be 640x480. The picture data will be saved in JPEG format. As mentioned in the previous section, once the picture is saved in the camera flash memory, the ESP-12E will upload the picture data to the web server.

2.3.3 Buzzer

The piezo buzzer acts as an alarm that makes a certain tone specified by the microcontroller when the locker door is unexpectedly open to force the intruder to leave the locker. Normally, the piezo buzzer will sound continuously when a 5 V voltage is applied to the two ends of the buzzer, but a certain tone can be generated from the buzzer by using a tone function and applying a delay on the software side.

2.4 Locker Module

The locker module includes a physical implementation of the locker and electronics necessary to be placed in the locker to implement the locking mechanism. The locker should be made of firm materials (wood or metal) to resist external damage. One magnet is placed near the hinge of the locker door. Once the door is properly closed by the user, the hall effect sensor will generate a higher voltage at its output due to an increasing magnetic flux density. The signal will be sent to the microcontroller to indicate that the door is closed and the servo can start to lock the door. The servo is placed on the inner side of the locker and latches with the hook on the locker door to realize the locking mechanism.

2.4.1 Locker

The locker is where the incoming package is stored. A handle is on the outer side of the locker door and a hook is placed on the inner side of the locker door to latch with the servo. The size of the locker should be large enough to fit two large flat rate USPS boxes in order to meet the basic need of a normal family.

Fig. 12 shows the physical design of the locker. The box on the locker is the control panel integrated with the security module. The left part of the control panel is the solar panel; the middle part of the control panel is the camera; the right part of the control panel is the LCD and piezo buzzer. Most electronics of the project including the PCB will be placed inside this box. Underneath the box is the actual locker with the hall effect sensor and micro servo inside.

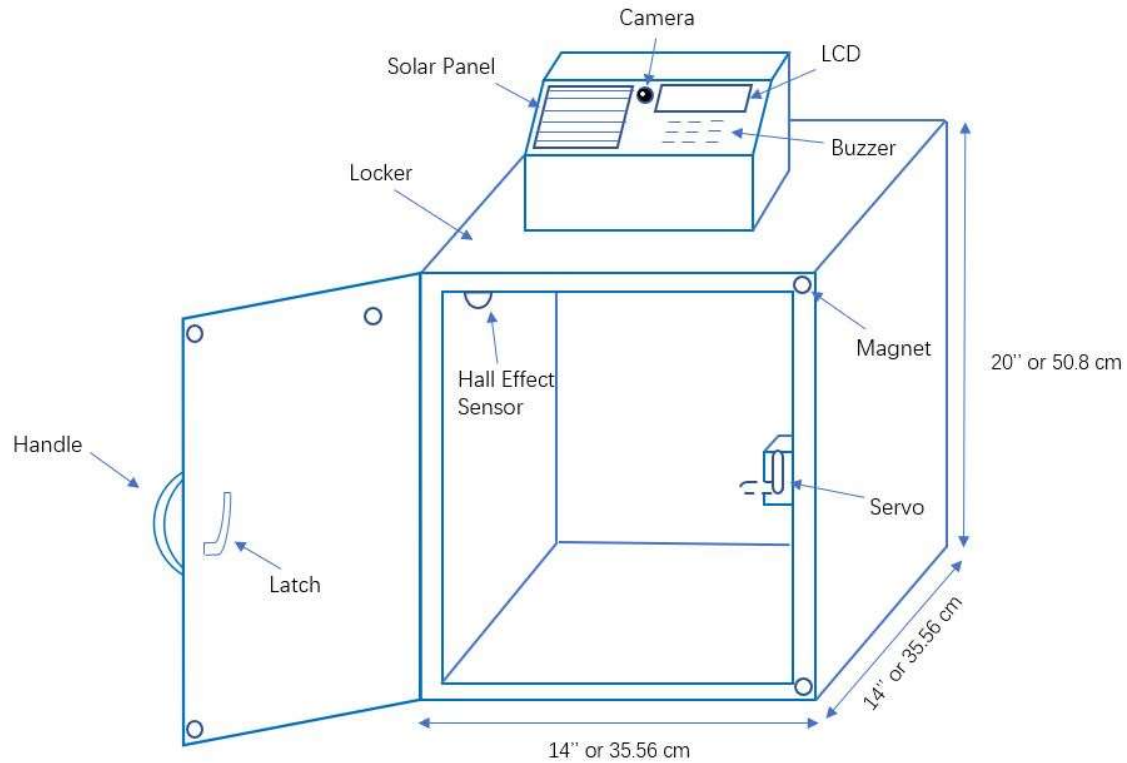


Fig. 12 Physical Design of the Locker

2.4.2 Servo

The micro servo is a necessary component to lock the locker. The micro servo works with the hook on the inner side of the locker door to lock the locker. When the locker is open, the servo stays at its 0-degree position; when the locker is closed, the microcontroller will send a signal to the servo to make it turn 90 degrees to the left to latch with the hook on the door such that the door cannot be opened from the outside. Arduino Servo library is used to set the angle of the micro servo.

2.4.3 Hall Effect Sensor

The hall effect sensor is placed at the edge of the locker door to detect whether the door is closed or not. When the door is closed by the courier, the hall effect sensor will detect an increasing magnetic flux density because of the magnet on the inner side of the door. This increasing magnetic flux density will increase the output voltage of the sensor. For the hall effect sensor we are using, at 25 Celsius degree temperature, each unit of magnetic flux density (G) will produce 5 mV of the output voltage. This output voltage will be fed into the only ADC on the ESP-12E chip. On the microcontroller side, it will determine whether the door is closed or not based on the hall effect sensor output voltage magnitude and send a signal to the servo and camera to make them rotate 90 degrees or capture the picture

respectively. The following schematic shows the necessary connection of the hall effect sensor circuit:

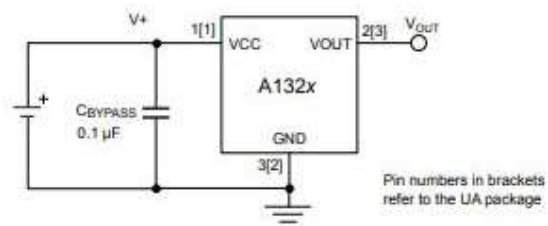


Fig. 13 A1324 Hall Effect Sensor Application Circuit [8]

2.5 Server

Although the server design is not the most important part of this project, it is necessary to realize certain functions of our locker. Technically, there are two servers involved in our locker design.

The first server is the one to store the images taken by the security camera of the locker. On the server side is a PHP code that takes the incoming string of image data from our ESP-12E WiFi module and saves the image data in the JPEG format to a specific folder on our server. The second server is the slack API server that accepts HTTP post requests from our ESP-12E WiFi module and sends data to our slack channel.

3. Design Verification

3.1 Power Supply

3.1.1 Solar Panel

To test whether the solar panel satisfies our need to charge the Li-ion battery, we tested the open circuit voltage of the solar panel in two different weather conditions (both measurements were taken in the afternoon).

Table II: Solar Panel Open Circuit Voltage

Date (Weather)	Output Voltage
April 21, 2019 (Sunny and Clear)	5.25 V
April 23, 2019 (Mostly Cloudy)	4.76 V

From the data shown in Table II, we can see the output voltage of the solar panel depends quite heavily on the intensity of the sunlight during the daytime. However, although the voltage level of the solar panel output varies with the weather, both of these two output voltages were tested to be enough to charge our 3.7 V Li-ion battery. Therefore, we are affirmed that this solar panel satisfies our need to provide solar energy to our project.

3.1.2 Battery Charger

We tested the MCP73831 battery charger PCB by connecting the input of the battery charger to a desktop USB port that provides a constant 5 V voltage and connecting the output of the battery charger to the two ends of the Li-ion battery. The battery charger stand-alone seems working because when there is a 5 V input voltage, the battery charger has a constant 4.2 V output voltage which is exactly the same as the charging voltage of the Li-ion battery. However, in the actual test, after we connected the output voltage of the battery charger to the battery, the current flowing through the battery charger IC was too high that it burned not only the battery charger IC but also the desktop USB port. This was unexpected because the maximum current supported by the USB 3.0 port is 900 mA, which is a lot higher than the battery charger charging current. In addition, at that time, the battery was only connected to the boost converter PCB, which means that the power consumption of the battery was very limited. The only explanation may be because the MCP73831 battery charger IC does not allow us to charge the battery and use the battery at the same time.

In order to fix this problem, we replace the MCP73831 battery charger PCB we designed with a commercial TP4056 Li-ion battery charger purchased on Amazon. A schematic of the TP4056 battery charger PCB is shown in Fig. 14. This battery charger has a higher charging current rating (1 A) and also provides a 4.2 V charging voltage when it is connected to a 5 V voltage source. When we tested the TP4056 battery charger in our final design, it worked perfectly and our Li-ion battery can be charged with no problem.

TYPICAL APPLICATIONS

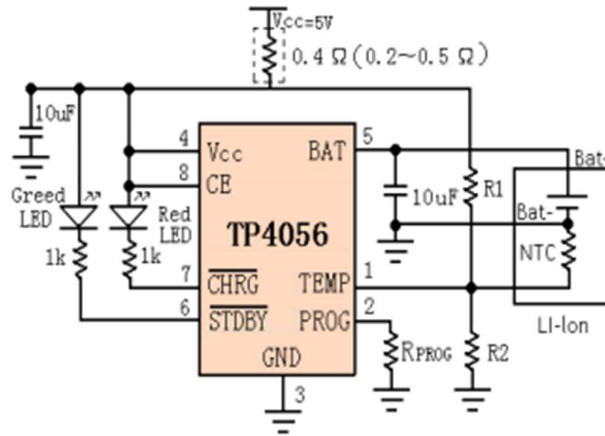


Fig. 14 TP4056 Battery Charger Application Circuit

3.1.3 Li-Ion Battery

The verification of the Li-ion battery was done by measuring the voltage of the battery using a multimeter at different capacities of the battery to see whether it provides a sufficient voltage to our locker. When the Li-ion battery was fully charged, the output voltage of the Li-ion battery was 4.12 V; after the Li-ion battery was discharged at a rate of 200 mW for half an hour, the output voltage of the Li-ion battery was 3.87 V.

Although the two voltages we measured at different battery capacities are different, the voltage of the battery is always higher than 3.7 V. Since the voltage regulators we are using can regulate the varying battery voltage to the level we want (3.3 V / 5 V), this battery satisfies our needs.

3.1.4 Voltage Regulators

The 5 V boost converter PCB we designed was tested by measuring the output voltage of the converter after connecting the 3.7 V battery to the input of the converter. When no load was connected to the output of the 5 V boost converter, the output voltage of the converter was measured to be 4.97 V, which is very close to the voltage level we want. However, when we tried to connect a stand-alone ESP8266 NodeMCU to the output of the boost converter, the output voltage drops to only 1.72 V. We could not find out what caused the voltage regulation of the boost converter failed because the current drawn by the NodeMCU was definitely lower than the boost converter current rating and the design of the boost converter PCB followed exactly the application circuit provided by the IC manufacturer.

In order to fix the boost converter issue, we purchased another 5 V boost converter PCB from Amazon. This boost converter outputs 5.02 V when it is connected to a 3.7 V Li-ion battery and the output current rating is 650 mA, which is far higher than what we need. The voltage regulation of this converter is also working because the output level remains at around 5 V when we used this boost converter to provide power to our ESP-12E microcontroller PCB.

3.2 Control Unit

In order to test whether the control unit module functions as what we expected, the first thing we tested is to see whether the ESP-12E module can be switched between the programming mode and the normal mode with no problem. When GPIO0 of the ESP-12E module was connected to the ground by placing a jumper wire between the upper pair of GPIO0 female headers, which means that the module was set to be in programming mode, we were able to upload code from Arduino IDE to the ESP-12E WiFi module at the highest writing speed of 921600 successfully. When GPIO0 of the ESP-12E module was connected to the 3.3 V level, which means that the module was set to be in normal mode, the module could execute the code uploaded with no problem. The reset button on the ESP-12E PCB was also able to reset the ESP-12E WiFi module. The ADC of the ESP-12E module could read output from the hall effect sensor correctly.

To test whether our ESP-12E module can connect to our home WiFi, sends image data from the camera to the server and sends a slack notification to the user, we first uploaded our entire code to the ESP-12E microchip. Since there was no previously saved WiFi setting for the module to connect to, the ESP-12E module turned itself to AP mode to allow our WiFi connected device to connect to it so that we can be redirected to the following pages to configure the WiFi setting.

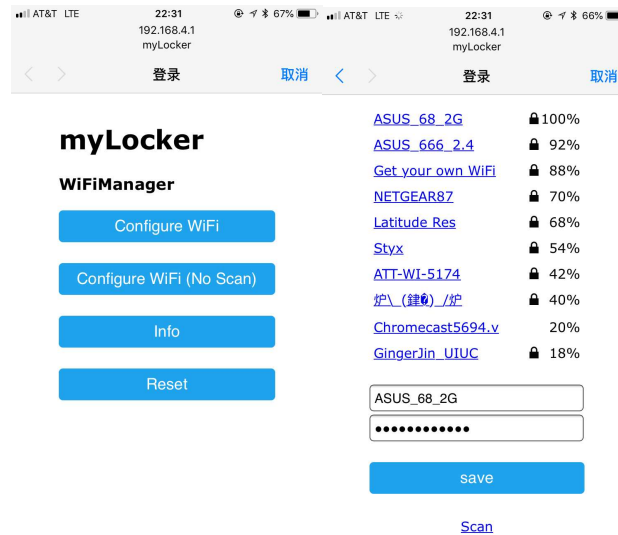


Fig. 15 and 16: ESP-12E WiFi Configuration Interface

By pressing the “Configure WiFi” button on the cellphone, the user will see a list of available WiFi networks to connect. The user can select any one of the WiFi networks and enter the password to connect the ESP-12E module to the home WiFi. Once that is done, the ESP-12E module will turn itself back to the station mode so that it can upload data to our server over the saved WiFi network.

To see whether the ESP-12E WiFi module can upload image data to our server and send notifications to our slack channel, we tested the module in our physical locker. It turned out that both of these goals were accomplished.

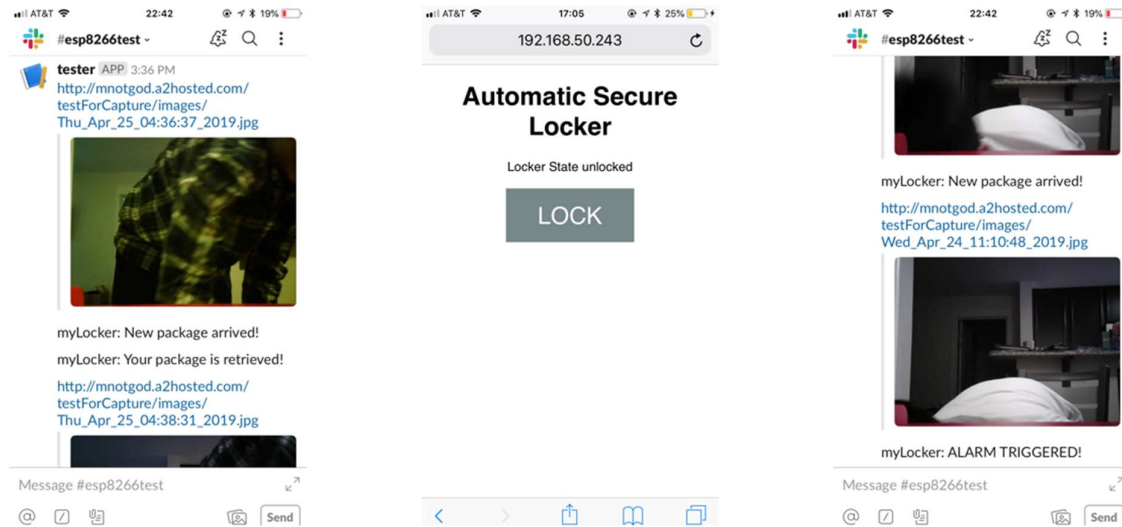


Fig. 17 Slack Notification and Test Results

As shown on the left and right screenshots of Fig. 17, the ESP-12E module can send image data to the server and send the image URL to our slack channel with no problem. The ESP-12E module can also send notification such as “New package arrived!”, “Your package is retrieved!” and “ALARM TRIGGERED!” to the slack channel to notify the user of the current state of the locker.

The last thing we need to test the ESP-12E WiFi module is whether the user can wirelessly control the locker over the local WiFi. We tested this feature by entering the IP address of the ESP-12E WiFi module. We were then redirected to an HTML generated page shown on the middle screenshot of Fig. 17. As shown on the screenshot, the webpage shows the current state of the locker and a button in the center of the webpage enables the user to lock or unlock the locker.

After all these tests, we were affirmed that our ESP-12E module functions well as a control unit to control other electronic parts and communicate with online servers for wireless data transmission.

3.3 Control Panel

3.3.1 LCD

To test whether the LCD functions correctly, we connected the four pins of the I2C breakout board of the LCD to an ESP8266 NodeMCU development board. Then, we uploaded a sketch from Arduino IDE on our computer to the NodeMCU to see whether the LCD functions as what we expected.

The testing result was that characters could be displayed on the 1602 LCD screen clearly and the common functions used to control an LCD such as clear, setCursor, scrollDisplayLeft and backlightOff can be run correctly. The backlight of the LCD also enables the user to see the characters clearly even during the night. These results proved that the 1602 LCD with I2C adapter works with no problem.

3.3.2 Camera

In order to test whether the ArduCam OV2640 camera works properly, we connected the camera with an ESP8266 NodeMCU development board. Then, we used the example code provided by the ArduCam distributor to see whether the camera is able to capture pictures and whether we can change the resolution of the pictures taken by the camera. The test was successful and we were able to capture the image and change the resolution of the captured image from the software side.

3.3.3 Buzzer

The test for the buzzer is done by connecting the positive end of the buzzer to 5 V VCC and the negative end of the buzzer to the ground and place the buzzer inside the locker to hear whether the buzzer can sound with enough loudness for people around the locker to hear the noise. The test was successful because the sound from the buzzer was loud enough for people around the locker to pay attention.

3.4 Locker Module

3.4.1 Locker

Fig. 18 and 19 show the actual locker we used for our final demonstration. The locker has two levels. The lower level is the locker to receive and store packages; the upper level is used to store PCBs and other electronic components used for this locker. Inside the lower level of the locker, we have the micro servo that latches with the hook on the inner side of the locker door. On the top right corner, we have the small hall effect sensor that detects the change of magnetic flux density when the door is closed. On the inner side of the locker door, there is a magnet. Inside the upper level of the locker, we have our battery, power PCBs and microcontroller PCB. On top of the locker are our solar panel, camera, and LCD screen.



Fig. 18 Physical Appearance of the Final Product

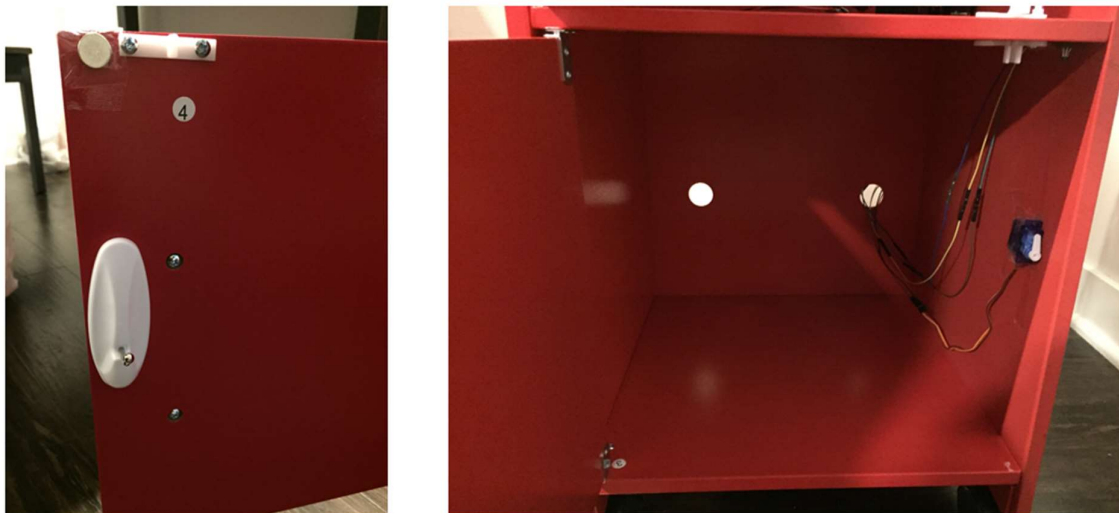


Fig. 19 Lower Level of the Locker and the Locker Door

3.4.2 Servo

To test whether the servo works as we expected, we connected the servo to an ESP8266 NodeMCU development kit. The signal pin of the servo is connected to one of the available GPIOs of the NodeMCU. We uploaded a simple sketch from the Arduino IDE to the NodeMCU to test whether the servo can turn from its 0-degree position to its 90-degree position and turn from its 90-degree position back to its 0-degree position. The test was successfully done because the servo can follow the sketch instructions and turn between the 0-degree position and the 90-degree position almost immediately.

However, one disadvantage we found during the actual test is that, although the servo is able to lock the locker, if the intruder tries to pull the handle on the locker door with a strong force, the servo inside the locker may break easily. In this way, the packages stored inside the locker can no longer be considered safe, even though we have the security camera and alarm.

3.4.3 Hall Effect Sensor

For the hall effect sensor we are using (Allegro A1324), when there is no magnet near the sensor, the digital output received by the microcontroller is about 780, which is equal to 2.51 V in terms of the analog voltage. According to Fig. 20, we can see this value is close to the quiescent voltage output of the sensor provided by the manufacturer. In our software side, we set the threshold to be 900, which is 2.9 V. If the output voltage of the sensor goes beyond this threshold, the microcontroller knows the locker door has been closed and it will send a signal to the servo to turn the servo 90 degrees to the left to lock the door. In our actual test, when the door is closed by the user, the digital value of the hall effect sensor output goes beyond 1024 (or 3.3 V in analog value) to “tell” the microcontroller the door has been closed.

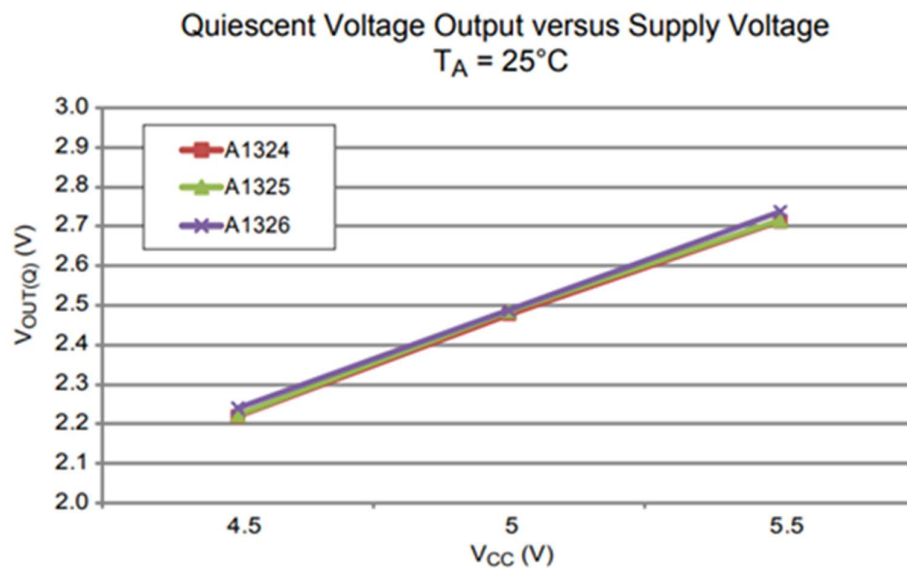
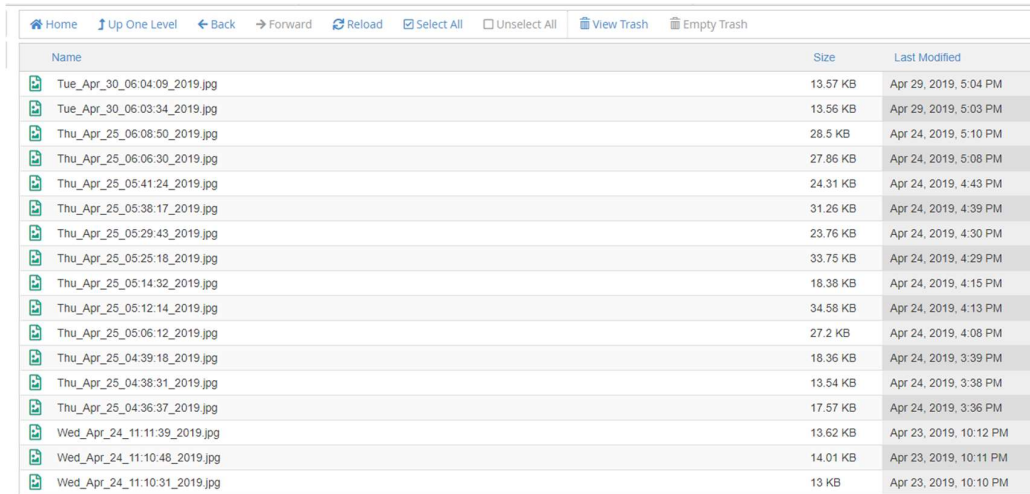


Fig. 20 Quiescent Voltage Output vs. Supply Voltage [8]

3.5 Server

Although the server is part of our entire design, the test for the server has already been carried out while we tested the functionality of our ESP-12E module. Fig. 21 shows a screenshot of the CPanel file system of our server. We can see from the screenshot that images were saved as JPEG files on our server folder.



The screenshot displays the CPanel file manager interface. At the top, there is a navigation bar with links: Home, Up One Level, Back, Forward, Reload, Select All, Unselect All, View Trash, and Empty Trash. Below this is a table listing files in a directory. The table has three columns: Name, Size, and Last Modified. Each file name starts with a small icon representing a JPEG image, followed by a timestamp in the format 'Day_Abbr_Month_Day_HH:MM:SS_YYYY.jpg'. The files are sorted by last modified date, with the most recent at the top.


















Name	Size	Last Modified
 Tue_Apr_30_06:04:09_2019.jpg	13.57 KB	Apr 29, 2019, 5:04 PM
 Tue_Apr_30_06:03:34_2019.jpg	13.56 KB	Apr 29, 2019, 5:03 PM
 Thu_Apr_25_06:08:50_2019.jpg	28.5 KB	Apr 24, 2019, 5:10 PM
 Thu_Apr_25_06:06:30_2019.jpg	27.86 KB	Apr 24, 2019, 5:08 PM
 Thu_Apr_25_05:41:24_2019.jpg	24.31 KB	Apr 24, 2019, 4:43 PM
 Thu_Apr_25_05:38:17_2019.jpg	31.26 KB	Apr 24, 2019, 4:39 PM
 Thu_Apr_25_05:29:43_2019.jpg	23.76 KB	Apr 24, 2019, 4:30 PM
 Thu_Apr_25_05:25:18_2019.jpg	33.75 KB	Apr 24, 2019, 4:29 PM
 Thu_Apr_25_05:14:32_2019.jpg	18.38 KB	Apr 24, 2019, 4:15 PM
 Thu_Apr_25_05:12:14_2019.jpg	34.58 KB	Apr 24, 2019, 4:13 PM
 Thu_Apr_25_05:06:12_2019.jpg	27.2 KB	Apr 24, 2019, 4:08 PM
 Thu_Apr_25_04:39:18_2019.jpg	18.36 KB	Apr 24, 2019, 3:39 PM
 Thu_Apr_25_04:38:31_2019.jpg	13.54 KB	Apr 24, 2019, 3:38 PM
 Thu_Apr_25_04:36:37_2019.jpg	17.57 KB	Apr 24, 2019, 3:36 PM
 Wed_Apr_24_11:11:39_2019.jpg	13.62 KB	Apr 23, 2019, 10:12 PM
 Wed_Apr_24_11:10:48_2019.jpg	14.01 KB	Apr 23, 2019, 10:11 PM
 Wed_Apr_24_11:10:31_2019.jpg	13 KB	Apr 23, 2019, 10:10 PM

Fig. 21 CPanel File System of the Server

4. Costs

4.1 Parts

Description	Part #	Quantity	Cost
ESP8266 WiFi Module	ESP8266 ESP-12E	1	\$4.99
18650 Li-ion Battery	PRT-13189	1	\$6.50
5 V Boost Converter	TE110	1	\$6.98
Li-ion Battery Charger	TP4056	1	\$1.20
3.3 V Linear Regulator	LD1117AS33TR	1	\$0.79
1602 LCD with Backlight	SunFounder LCD1602	1	\$6.49
Camera	ArduCam Mini OV2640	1	\$25.99
LCD I ² C Breakout Board	HiLetgo 3-01-0080	1	\$2.99
Solar Panel	ALLPOWERS 2.5W 5V/500mAh	1	\$8.99
Hall Effect Sensor	A1326LUA-T	1	\$2.10
Passive Components	-	-	~\$10.00
PCBs	-	1	\$1.00
Locker	KidKraft Medium	1	\$52.36
Total	-	-	\$130.38

4.2 Labor

Total labor cost = (2 x 40 \$/hr) x 2.5 x 10 hr/wk x 12 wks = \$ 24,000

5. Conclusion

5.1 Accomplishments

Through the three months of project development, we successfully demonstrated the basic functionality of the locker: the locker remained locked until the owner unlocks it over WiFi; the locker can send pictures and push notifications to the owner in case there is any status update; the locker can be powered by the solar energy. Although there were some parts of the design not working as what we expected, after replacing some parts with the off-the-shelf components, we were still able to finish the project within the tight time schedule.

5.2 Uncertainties

One of the biggest uncertainties about our final product is whether the locker can always keep functioning without the need for the user to replace the battery frequently.

Although theoretically, the Li-ion battery itself is able to provide power to the locker without sunlight for more than 13 hours, we were not able to test the locker performance in the long term to determine whether this is always the case.

If this product is going to be commercialized one day, necessary long term tests are required before this product can be distributed to the market.

5.3 Ethical Considerations

As engineers who build projects for the welfare of the entire human community, we force ourselves to strictly follow the IEEE Code of Ethics [9]. First of all, this secure locker project is designed to help protect people's property, which corresponds to code 1 of the IEEE code of ethics:

to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, and to disclose promptly factors that might endanger the public or the environment;

In addition, the project is designed for the use of all groups of people, with no discrimination either religiously or racially, etc. Therefore, it corresponds to code 8 of the IEEE code of ethics:

to treat fairly all persons and to not engage in acts of discrimination based on race, religion, gender, disability, age, national origin, sexual orientation, gender identity, or gender expression;

However, due to the limitation of the design, this locker is not recommended to use by people with blindness. We hope that we could provide more accessible versions of this product in the future.

Another thing to be aware of is that when a package is placed in the locker or the security system of the locker is triggered, the camera needs to take pictures of the person standing in front of the locker. This might offend the person's right. Therefore, it violates code 9 of the IEEE code of ethics in some ways:

to avoid injuring others, their property, reputation, or employment by false or malicious action;

In order to minimize the negative impact of the camera module as much as possible, a notice will be labeled in a conspicuous position of the locker to tell the users that their actions might be recorded if they intend to open the locker without permission.

5.4 Future Work

Although ECE 445 has come to an end, the development of our automatic secure locker has not finished yet. The project presented right now is a prototyping version that only shows the potential of this product. In order for this project to truly become a commercialized product, there is still a lot of future work to be done:

1. The micro servo is not suitable to be the lock of our locker because it is not firm enough when the intruder tries to pull the door handle very hard.
2. The physical locker used for our final demonstration is not the same as what we proposed in the design phase. If cost is no longer a problem, a better locker made of metal with an appearance like what is shown in Fig. 12 should be used to be our actual locker.
3. The design of our power PCBs is problematic and does not meet our original expectations. Further design and tests are required to build a better battery charger PCB and boost converter PCB to satisfy our needs so that we do not have to purchase the off-the-shelf products online.

References

- [1] Instructables, “SOLAR POWERED ARDUINO WEATHER STATION”. [Online].
Available:
<https://www.instructables.com/id/SOLAR-POWERED-ARDUINO-WEATHER-STATION/>.
[Accessed: 3/4/2019]
- [2] Amazon, “ALLPOWERS 1PC 2.5W 5V 500mAh Mini Solar Panel Module Solar System Epoxy Cell Charger DIY 130x150mm (Solar Panel Only) ”. [Online].
Available:
https://www.amazon.com/ALLPOWERS-Encapsulated-Battery-Charger-130x150mm/dp/B00CBT8A14/ref=sr_1_3?crid=37OKGMZCRQW2W&keywords=5v+solar+panel&qid=1553489620&s=gateway&srefix=5v+solar%2Caps%2C148&sr=8-3.
[Accessed: 3/24/2019]
- [3] Microchip, “Miniature Single-Cell, Fully Integrated Li-Ion, Li-Polymer Charge Management Controllers”. [Online].
Available:
<http://ww1.microchip.com/downloads/en/DeviceDoc/20001984g.pdf>.
[Accessed: 3/7/2019]
- [4] SparkFun, “ICR18650 2600mAH datasheet”. [Online].
Available:
<https://cdn.sparkfun.com/datasheets/Prototyping/ICR18650%202600mAH%20datasheet.pdf>.
[Accessed: 3/7/2019]
- [5] ST, “Synchronous rectifier step up converter”. [Online].
Available:
<https://www.st.com/content/ccc/resource/technical/document/datasheet/38/68/3c/50/c4/07/4b/83/CD00055960.pdf/files/CD00055960.pdf/jcr:content/translations/en.CD00055960.pdf>. [Accessed: 3/24/2019]
- [6] ACROBOTIC, “What Is The ESP8266 And Why Is It So Popular?”. [Online].
Available: <https://learn.acrobotic.com/tutorials/post/what-is-the-esp8266>.
[Accessed: 3/21/2019]

- [7] Arducam, “2MP: OV2640”. [Online].
Available: <http://www.arducam.com/products/camera-breakout-board/2mp-ov2640/>.
[Accessed: 2/15/2019]
- [8] Allegro MicroSystems, “Low Noise, Linear Hall Effect Sensor ICs with Analog Output”.
[Online].
Available:
<https://www.allegromicro.com/~media/Files/Datasheets/A1324-5-6-Datasheet.ashx>.
[Accessed: 3/7/2019]
- [9] IEEE, “IEEE Code of Ethics”. [Online].
Available: <https://www.ieee.org/about/corporate/governance/p7-8.html>.
[Accessed: 2/16/2019]

R&V Tables

1 Solar panel

Requirement	Verification
<ol style="list-style-type: none">1. The open circuit voltage of the solar panel when it is exposed under sunlight in a partly cloudy day should be at least 5 V.2. The current drawn from the solar panel should be at least 250 mA when the voltage output under the load of the solar panel is 5 V.	<ol style="list-style-type: none">1. On a partly cloudy day, place the solar panel under sunlight, measure the voltage across the output terminals of the solar panel using a voltmeter to see if the voltage is higher than 5 V.2. Terminate the solar panel with a resistive load such that the voltage across the load is 5 V. Measure the current through the resistive load with an ammeter to see if the current is higher than 250 mA.

2 Li-ion battery charger

Requirement	Verification
<ol style="list-style-type: none">1. When the Li-ion battery charger is connected to a not fully charged Li-ion battery, the output voltage should be between 4.1 V and 4.3 V when an input voltage of 4.5 V - 5.5 V is applied.	<ol style="list-style-type: none">1. Discharge a 3.7 V rechargeable Li-ion battery to $\frac{2}{3}$ of its capacity, connect it to the battery charging circuit. Apply a 5 V input voltage to the battery charger, use a voltmeter to measure if the output voltage of the charging circuit is between 4.1 V and 4.3 V.

3 3.7 V Li-ion battery

Requirement	Verification
<ol style="list-style-type: none">1. The Li-ion battery should have an output voltage between 3.6 V and 3.8 V when it is not depleted.	<ol style="list-style-type: none">1. Measure the voltage across a fully charged Li-ion battery using a voltmeter to see if the voltage is around 3.7 V.2. Discharge the battery at a rate of 150 mA for 14 hours, then

	measure the voltage across it again, check if the voltage is still around 3.7 V.
--	--

4 Boost converter

Requirement	Verification
1. The boost converter must output a steady 5 V voltage when it is connected to a 3.7 V Li-ion battery.	1. Connect a 3.7 V Li-ion battery to the input of the boost converter, measure the output voltage of the converter to see if the voltage is between 4.95 V and 5.05 V.

5 Microcontroller

Requirement	Verification
<ol style="list-style-type: none"> 1. The ATmega328p must transmit and receive data over I2C and SPI buses to control the operation of a certain I2C or SPI device. 2. The ESP8266 must transmit and receive data via a local WiFi network. 	<ol style="list-style-type: none"> 1. Connect the OV2640 camera to the ATmega328p chip and upload code to the microcontroller over USB connection. If the SPI interface can be setup between the microcontroller and the camera can be detected by the microcontroller, the I2C and SPI buses should work properly. 2. Connect the ESP8266 module to the local network. Connect an LED to one of the GPIOs of the ESP8266 module, control the output state of the GPIO from a web browser to see if the LED can be turned on and turned off via TCP remotely.

6 LCD

Requirement	Verification
1. The LCD should communicate with the microcontroller over the I2C	1. Power the LCD by connecting VCC and GND pins to 5 V and ground

<p>bus to display correct information to the user.</p> <ol style="list-style-type: none"> 2. The contrast of the LCD should be set to an appropriate value to clearly display the characters. 3. The backlight of the LCD should be bright enough for the user to read during the night time. 	<p>pin of the ATmega328p, connect the two I2C pins SDA and SCL to corresponding pins of ATmega328p chip, use functions from LiquidCrystal_I2C.h library to send displayed data to the LCD to display. If all characters are displayed correctly and clearly, the LCD works.</p> <ol style="list-style-type: none"> 2. Remove external light as much as possible to see if the characters on the LCD are still visible to the user.
---	---

7 Camera

Requirement	Verification
<ol style="list-style-type: none"> 1. The camera should remain idle to save power normally. When the locker is opened by a courier, the camera should take a picture of the courier to keep as a record; when the locker is opened unexpectedly, the camera should take 10 consecutive pictures of the intruder. The camera should be at least 2 megapixels to ensure the definition of the photos. 	<ol style="list-style-type: none"> 1. Connect power, I2C and SPI pins of the camera module to the ATmega328p. 2. Upload code from the Arduino IDE to ATmega328p over USB, open the host application provided by ArduCam to capture a photo with size 800x600. Check if the SPI interface can be set up and camera can be detected.

8 Piezo buzzer

Requirement	Verification
<ol style="list-style-type: none"> 1. The buzzer should make a tone controlled by the microcontroller with a loudness high enough to be heard 10 meters away from the buzzer. 	<ol style="list-style-type: none"> 1. Connect the buzzer to one of the digital I/Os of ATmega328p chip, upload code that makes the buzzer play a certain tone, gradually step back until we can no longer hear the sound played by the buzzer. Measure the distance with a ruler

	to make sure it is more than 10 meters.
--	---

9 Locker

Requirement	Verification
<ol style="list-style-type: none"> 1. The locker should be large enough to contain two USPS large boxes, therefore, the dimension of our locker should be at least 35.56 cm x 35.56 cm x 50.8 cm [6]. (However, for the demonstration purpose, the actual dimension of our locker may be smaller than expected.) 2. When the locker is closed, it should not be opened without at least 10 N of pulling force being applied to the handle. 	<ol style="list-style-type: none"> 1. Take two USPS large boxes and place it into the locker to see if the locker can fit the two boxes. 2. Hold a force gauge and place the hook of the force gauge to the door handle, pull the force gauge to open the door, record the force being applied at the moment when the door is open. Check if the force is at least greater than 10 N.

10 Servo

Requirement	Verification
<ol style="list-style-type: none"> 1. The servo should start at 0 degree position, and be able to rotate to the 90 degree position in about 1 second with no load applied. 	<ol style="list-style-type: none"> 1. Power the servo with a different 5 V power rail other than the one from the microcontroller, connect the servo signal input to one of the digital pins on a functioning Arduino Uno board. 2. With the help of Arduino Servo library, write code to set the initial position of the servo to 0 degree and then rotate it to 90 degree and upload the code to the board. If the operation is successfully done, the servo should be working.

11 Hall effect sensor

Requirement	Verification
<p>1. The hall effect sensor should send valid analog signal to the microcontroller based on how close the distance between the magnet and the sensor. There should be a distinguishable difference between the value we get when the door is closed and the value we get when the door is open.</p>	<p>1. Place the hall effect sensor on a breadboard such that it faces directly with a magnet. Set up the circuit in Fig. 6 and connect the output of the sensor one of the digital pin on a functioning Arduino Uno board.</p> <p>2. Write code to use the serial monitor of Arduino IDE to check whether the Hall effect sensor returns signals that are proportional to the distance between the sensor and the magnet.</p>