EzRack

Amanda Chanthalangsy, Vanessa Garcia, Joel Gonzalez, Trung Luu

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450 U.S.A.

***Abstract* — This document contains the details pertaining to the design and implementation of the EzRack smart bike rack system. The EzRack is a bike rack that is equipped with electronic locks and a user interface to allow users to reserve and release slots to store their bicycles. This project contains two major features that are the off-grid power system and the mobile application interface. The bike rack system will be powered by solar energy off-grid with a PWM solar charger to regulate charging. A mobile application will accompany the user interface to control the electronic locks wirelessly. Our bike rack will be environmentally friendly and convenient to install and use.**

***Index Terms* — Battery chargers, DC motors, microcontrollers, mobile applications, photovoltaic systems, user interfaces.**

I. Introduction

The University of Central Florida is one of the largest universities in America and is continuously growing. While the university's enrollment is currently greater than 60,000 students, it is also home to more than 12,000 employees and open to visitors year round. The campus spans over a thousand acres and is located in metropolitan Orlando. The University of Central Florida has become a main attraction for traffic congestion. Finding parking around the campus has also become a difficult task. Not only does the heavy traffic and parking congestion cause inconvenience, it also affects the environment by increasing emissions from the automobiles circulating around campus. To help alleviate these issues, our project aims to solve these recurring problems with an easy, eco-friendly solution that we call EzRack.

The bike racks currently available at the University of Central Florida are often crowded or located in inconvenient areas on campus. This may deter students from biking to, from, or around campus due to the hassle of securely storing their bike in a convenient location. With parking also being a hassle, the option of simply biking becomes more attractive with the implementation of a smart bike rack system. Also, the usage of mobile applications nowadays gives people a better way to plan ahead. With our electronic lock system being remotely controlled, users will be allowed to know ahead of time where they should go to park their bicycle.

Existing bicycle-sharing systems today offer a way for users to rent bicycles for use within certain boundaries. EzRack is inspired by these racks but are different in a way that allows users to use their own bikes. Instead of renting the actual mode of transportation and requiring the return to storage on the rack, our system is meant for storing any bike. This takes the responsibility away from the agency maintaining the rack from having to provide costly bicycles as well as maintaining them.

II. System Components

The following sections describe the selection and function of components used in each subsection. Each subsection plays a vital role in the system, which will be further explained in later sections.

A. Microcontroller

The microcontroller plays a central role in our system. We based our selection on considerations of the architecture, price, reliability, and ease of use. Atmel's AVR microcontrollers as well as Texas Instrument's MSP430 family of microcontrollers were considered. Due to the purposes of this project, a microcontroller with Harvard Architecture and a RISC instruction set are preferred. The ATmega328P was chosen based on its low cost and reliability compared to other microcontrollers in the group. This processor operates at 16 MHz and has 23 input/output pins, which is necessary to meet our requirements. It has an 8-bit AVR RISC architecture and 32 KB ROM storage [1].

B. Wireless Module

Wireless communication is essential in our project due to the outdoor nature of our system location. The microcontroller requires connection to the Internet to access the passwords generated by the mobile application and stored in the online database. The wireless module is the bridge between the mobile application and the MCU. The Arduino Wi-Fi Shield and ESP826612e were considered due to their wireless abilities. However, the ESP826612e was chosen for it's lower price.

With the use of Firebase-Arduino libraries, the Wi-Fi module will transmit and receive data from both the Firebase database and the MCU. Firebase real-time database is a NoSQL cloud database by Google. Data is stored as JSON and will synchronize in real-time for every client that is connected every time the data changes [2].

This allows user control from the mobile application and meets our project specifications.

C. Android Application

The mobile application developed for this project is Android-based. The Android application is designed to allow users to reserve, lock, and unlock their assigned bicycle slot directly from their android device. The Android development was chosen due to its compatibility and user-friendliness.

D. Battery

A 12V sealed lead acid battery is implemented in this design for its long lifespan and its reliability in off-grid power systems. Lead acid batteries also require a low level of maintenance. The relatively affordable cost of these batteries was also a main consideration in selecting this type. A 35Ah battery is used in this prototype to ensure enough power to operate the system on a single charge for a minimum of 8 hours.

D. Charge Controller

To charge the battery and provide power to the system, a 12V PWM charge controller is designed using an Arduino Nano microcontroller to regulate the charging and ensure optimal battery capacity and discharge. This charge controller is designed to be capable of connecting two loads that are 5V and 12V, respectively. This ensures the proper voltage necessary to power our main MCU as well as our locking mechanisms. LED indicators are embedded into the charge controller to indicate the battery status as well as load status. An LCD display is also connected to display the current state of charge, battery temperature, and other statuses of the power control.

E. Photovoltaic Cells

In selecting the appropriate solar panel to use in our design, we considered the most cost efficient options that are able to provide sufficient power to charge the rechargeable sealed acid battery that powers our system. Since the solar panel will be mounted on top of the bike rack, size was a negligible factor. We opted for a 50W 12V monocrystalline solar panel manufactured by Renogy.

F. Locking Mechanism

An important feature of the EzRack is its embedded locking mechanism. Current electronic locks on the market are either too costly to implement in this project or do not meet our specifications for operation. In order to ensure proper operation and locking, the locking mechanism was designed using a motor driver along with a linear lock actuator and other mechanical components. This locking mechanism is designed to fit into the bike rack housing design and operate with the voltage supplied.

III. System Overview

The following section will outline the operation of the system as a whole as well as the operations of each subsystem on a surface level.

A. System Operation

In figure 1 below, the operations of all the main components in the system are shown on a high level.

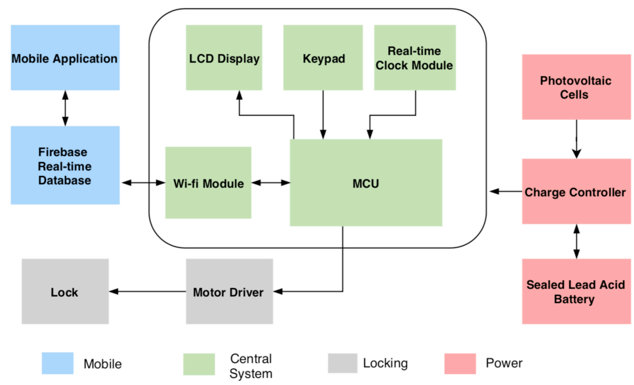


Fig. 1. Flow diagram showing the system overview of EzRack.

A kiosk will house the user interface including the MCU, wireless module, keypad, and LCD display. The interface will communicate wirelessly to a mobile application through Firebase real-time database by the wireless module. This system will transmit signals to a motor driver that will operate the locking mechanism.

B. Software Flow

In figure 2 below, the mobile application flowchart illustrates the interaction of each function.

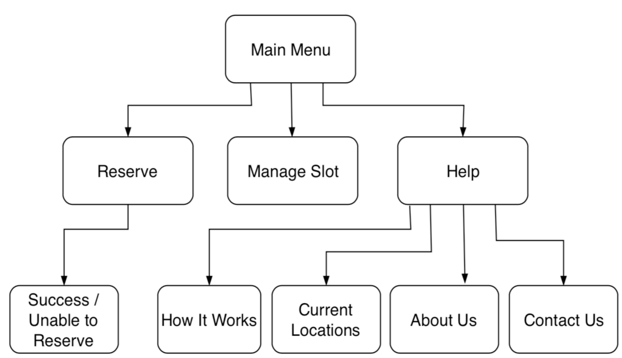


Fig. 2. EzRack mobile application flow diagram

The mobile application will allow the user to take advantage of all of the features on the bike rack. The application is user friendly as it is fairly simple. With the Android mobile app, the user will be able to access three options from the main menu. The options are to check the availably of a slot and reserve, manage their current slot, or obtain help.

Figure 3 below shows the Main Menu screen that users will encounter upon opening the mobile application.

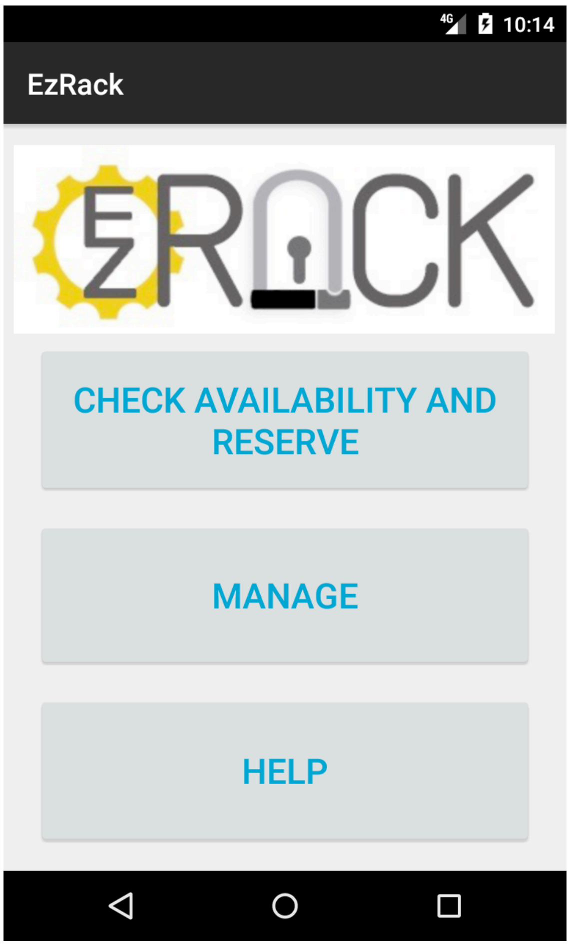


Fig. 3. EzRack User Main Menu Screen

Communication between the mobile application and the main MCU is established through the Firebase real-time database. The mobile application is the main source of control for the bike rack. With connection through the database, Firebase will communicate the appropriate information to the microcontroller through the Wi-Fi module.

The embedded software on the main MCU will communicate between all of the components in the system. It will be tasked with also tracking slot availability and operating the locking mechanism. It will be controlled not only by communication from the mobile application, but also from the keypad attached to the main kiosk of the bike rack. For users without the mobile application, a keypad has been provided to allow interaction with the bike rack. Reservations made with the mobile application can also be claimed onsite using the keypad. For users without a reservation, a five digit key can be entered to reserve an available slot.

The embedded system is comprised of six major components, the microcontroller, LCD, Keypad, RTC Module, Motor Driver and the Wi-Fi Module. The embedded system will use a 2x16 LCD display to communicate with the user. Once reserved, the slot is rendered unusable until it is either claimed with the password provided to the reservee or a certain amount of time has passed without the reservee using the password to unlock the slot, in which time it will become available for reservation through the app or use through the kiosk. To provide this feature, it will be necessary to keep accurate time. The DS3231 is low-cost, accurate, and can keep track of hours, minutes, and seconds along with day, month and year. It also compensates for leap-years and months that are shorter than 31 days. The DS3231 runs off a separate battery and will keep track of time even if the microcontroller is disconnected from its power source [4].

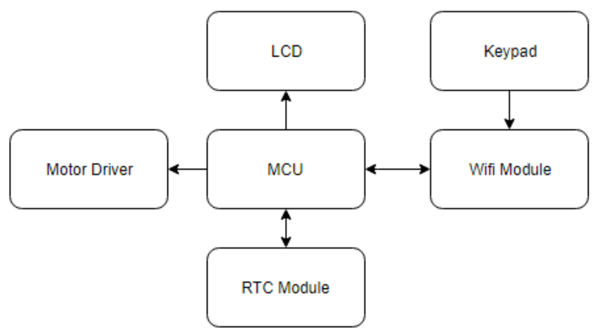


Fig. 4. EzRack Embedded System

C. Power Flow

To achieve an off-grid power system, the EzRack will be utilizing a solar-powered system. This allows the project to be competitive among other similar systems due to its energy independence. Power will be supplied to both the main communication system and the locking mechanism by a charge controller. This project utilizes a photovoltaic system to generate and store power. The solar panel will generate an electrical current, which will run through the charge controller to regulate the charging of a lead-acid battery and power the system.

With the main MCU requiring an input of 5V and the lock motor requiring 12V, the charge controller is designed to be able to handle two output loads operating at 5V and 12V each. Components connected to the main MCU require 5V as well as 3.3V and therefore will be converted through buck converters connected directly on the main PCB. An Arduino Nano that will handle the charging algorithm appropriately and ensure that the load receives the necessary voltages controls the charge controller.

The figure below shows the power flow from the power system to the main bike rack components.

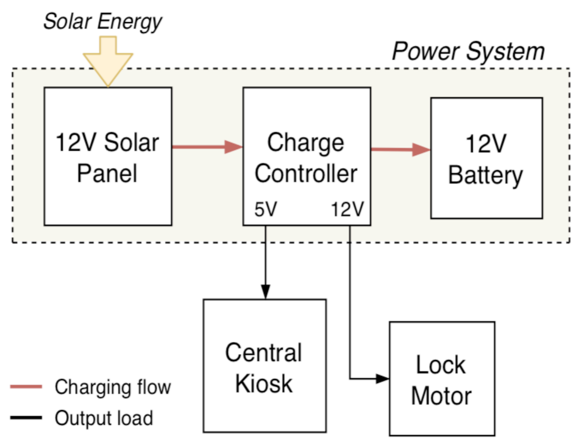


Fig. 5. EzRack Power Control Diagram

IV. Hardware Detail

In this section, the hardware components that were previously introduced will be further discussed in technical detail that supports their use in the system.

A. Embedded Systems

The software embedded into the EzRack will receive input from only two sources. One of the inputs will be a six-digit code from the dedicated mobile application. That six-digit code consists of the unique key and the opcode. The second input is received through a keypad that is attached to the bike rack itself.

The wifi module constantly polls the database for any change in the status of the bike slots. Changes can either come from the mobile application or the keypad and are treated the same. Through the mobile application, a slot can be reserved after which the database will contain a number preceded by the number one. Once the wifi module sees a number starting with one, it will alert the MCU to unlock the slot. If there is number starting with two, the MCU will be notified to unlock the slot while retaining the passcode entered by the user. If the database contains a number beginning with three, the slot will be locked. And finally, if there is a zero, the slot will be unlocked and become free to be reserved.

B. Power Control

The solar charge controller designed for our project is Pulse Width Modulated. This charging method allows the battery voltage to reach a set point then slowly reduce the charging current to help with heating and gassing of the battery. It continues to provide maximum amount of energy to the battery in a short amount of time. This is achieved by utilizing 2-step "bulk and float" charging algorithm in conjunction with current and voltage sensors. Bulk stage charging will provide maximum current available to charge the battery. As the battery's voltage increases and becomes fully charge to the set point value, float stage will reduce current and voltage to prevent gassing of the battery. The figure below illustrates this two stage charging.

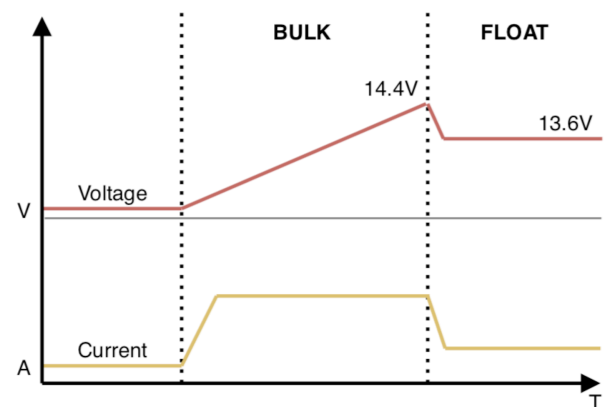


Fig. 6. EzRack Charging Graph

Bulk charge has a set point of 14.4 Volts and float charge has a set point of 13.6 Volts. Regardless of bulk charge or float charge set point, the set points are changed dynamically based on the ambient temperature of the battery from the default value of 25 degrees Celsius. If the temperature is above the 25 degrees Celsius, the charge set point is reduced. If the temperature is below the 25 degrees Celsius, the charge set point is increased.

For load control, the charge controller has a low voltage disconnect static value of 11.5V. When the battery voltage is greater than low voltage disconnect value, the charge controller will provide the load. If the battery falls below low voltage disconnect, the charge controller will no longer provide load. The charge controller has two sets of LED indicators. The first LED set is the battery indicator. When battery voltage is below 12V, the LED indicator is red indicating that battery has low voltage. When the battery voltage is between 12V and 14.4V, the battery LED is green indicating that the battery is healthy. When the battery voltage is greater than 14.4V, the LED is blue to indicate that the battery is fully charged. The second LED set is to indicate whether the charge controller is providing the load or not. When the LED is green, it indicates that the load is being provided. When the LED is red, it indicates that the load is not being provided. The following figure illustrates the charge controller components with power flow as well as data flow.

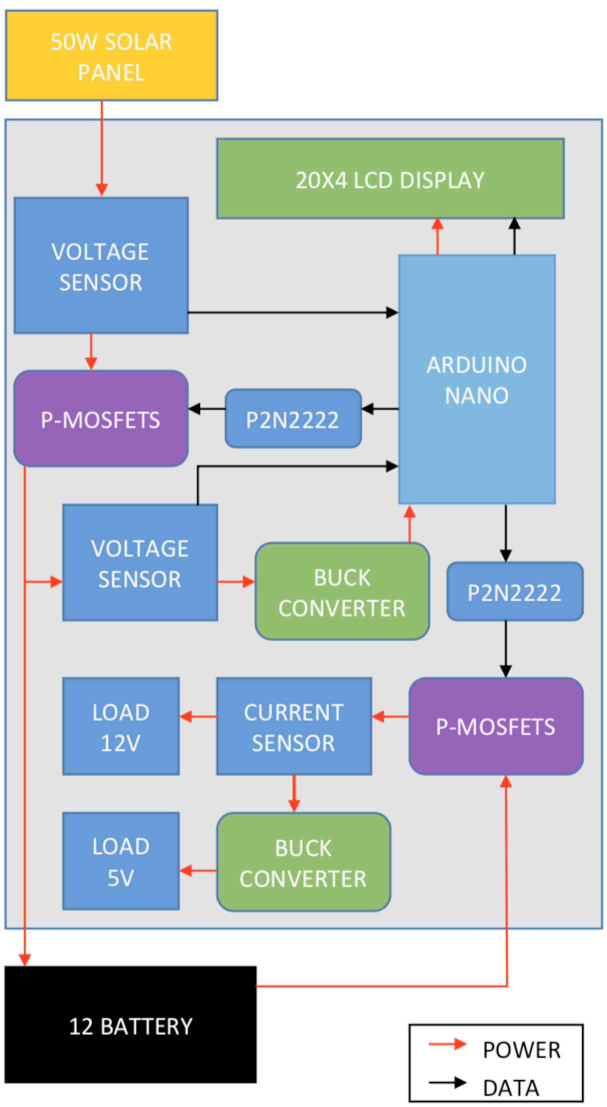


Fig. 7. Charge Controller Diagram

D. Locking Mechanism

For the design of the locking mechanism, a lever design will provide a secure lock of the bike wheel to the bike rack slot. This locking mechanism incorporates both mechanical aspects as well as electrical. A lock actuator with a push/pull motion controlled by a motor driver is built to ensure proper operation. The lock actuator has a push/pull through a distance of 1 inch. This is unsuitable for the type of locking necessary in our bike rack. To develop the necessary locking distance, the lock actuator is connected to a bar that will push a longer rod that will be able to travel through a distance of at least 5 inches and securely lock the slot.

A motor driver is designed using a L298N dual-channel H-bridge driver. H-bridge is a circuit that allows voltage to flow in either direction meaning the driver can operate the motor in a clockwise or counterclockwise motion. The device is capable of driving up to 3A per channel with a maximum power of 25W. It has an input logic voltage of 5V and is capable of driving 12V to the lock motors. With the 12V lock actuator operating at 2A, this motor driver provides adequate power to operate the lock. The L298N is a 16 pin IC with 8 pins on each side making a single chip capable of driving 2 motors [3]. It will receive inputs from the MCU to operate the locking system.

V. Software Detail

The following section further discusses the software details of the Android mobile application as well as the kiosk user interface.

A. Android Mobile Application

Users will be able to use a free mobile application to make use of the functionalities incorporated in the EzRack. The mobile application was designed to run on the android operating system.

As far as the GUI is concerned, the structure of the application is fairly simple. The top activity is the main menu. This is where the user makes their initial decision. The user will be able to view a detailed explanation of how to navigate the entire application.

For the application's backend, the algorithm is fairly simple. Based on what the user decides to do, the application will update the database accordingly. Regardless of what the user ultimately selects, the database will receive a six-digit integer from the application. The last five digits of that integer will be the key that will is used to control the status of each slot. The first digit is the control signal that will tell the device what the user wants to do. When a bike slot reservation is requested and that request is granted, the application will then update the database with the necessary fields. When the user arrives at the EzRack and is ready to secure their bike in the slot assigned to them the app will send the same key that they have entered previously to the database. It is not part of the mobile application's algorithm to determine how the EzRack will handle each situation of the bike reservation process. The information is presented in that way, however, to make that mobile application as usable as possible. The system embedded to the device is ultimately responsible for the task of determining what action the user is taking based on the control variable that is part of the six-digit integer that is receives from the database.

Android Studio was used to create the EzRack Mobile application. Most of the source code on the entire application consists of either Java Files or xml files. Although the EzRack source code features C++ support, the application is primarily written in Java. Developing a Java application on Android Studio that is compatible with Android smartphones makes the development and testing a simpler process. Adding the additional C++ functionality was a choice was done to aid with future updates.

B. Embedded User Interface

For users who wish to forego the use of the mobile application, a keypad is provided to interact with the bike rack. Any input from the keypad will engage the system to either allow the use to unlock or lock a slot that is current available on the rack. The user will enter a number followed by their five-digit passcode. The database will then be checked to see if the entered passcode is assigned to a slot. If the passcode is stored in the database, it's opcode – 1, 2, or 3- will be checked to determine what action should be taken. If the passcode is not in the database, it will be stored in the first available slot and treated as a reservation. This gives the user freedom from using the mobile application in case they do not have a mobile device readily available or do not have wireless connection from that device. Each time the user wishes to perform an action such as reserving, locking or unlocking their bike, they will need to enter their passcode. All reservations will be canceled after fifteen minutes have passed without them being claimed. The fifteen minutes allows the person who reserves the slot to arrive at the bike rack station in an appropriate amount of time to claim their slot, but also frees up the slot and allows another user to access it instead in case the original requester doesn't utilize their reservation. The operational flow of this system is depicted in the following flow diagram.

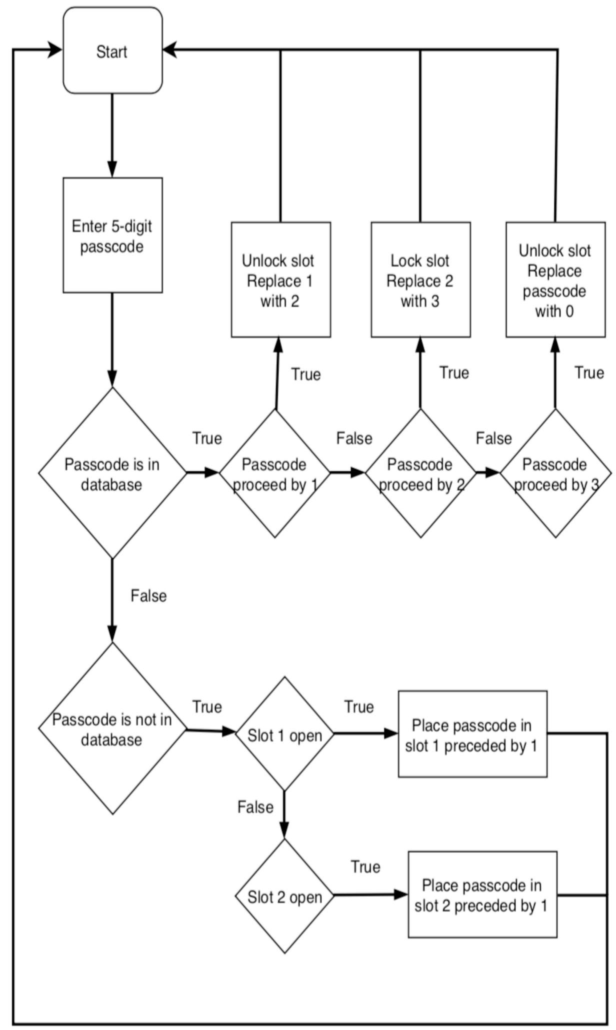


Fig. 8. Operational flow diagram

VI. PCB DESIGN

The EzRack system is designed having two separate Printed Circuit Boards. The first PCB includes the main MCU along with the user interface components as well as the wireless module. The second PCB included the charge controller and the voltage regulators. Two separate PCBs were necessary due to our rack design.

The first board will be mounted in the user kiosk housing while the second board must be located near the battery housing. This is to ensure the charge controller is able to read the temperature of the battery through the temperature sensor. The schematics of all the components were first developed and implemented on breadboards. This was done to test and ensure proper connections and functionality. These designs were then further developed into PCB designs using the EAGLE PCB design software.

These PCBs have been printed by JLPCB for their quick turnaround times and delivery. Figure 9 and 10 represent the two printed circuit board layouts developed for this project.

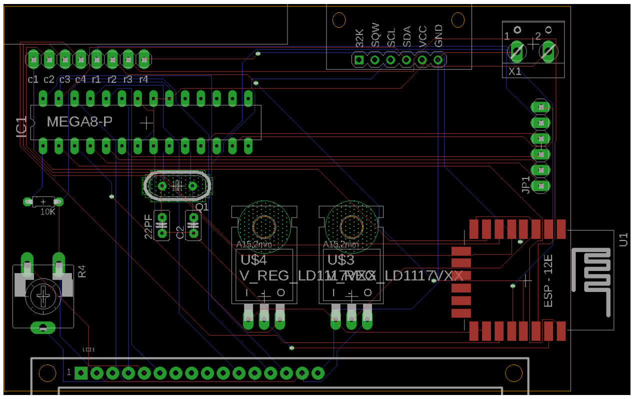


Fig. 9. Embedded PCB layout

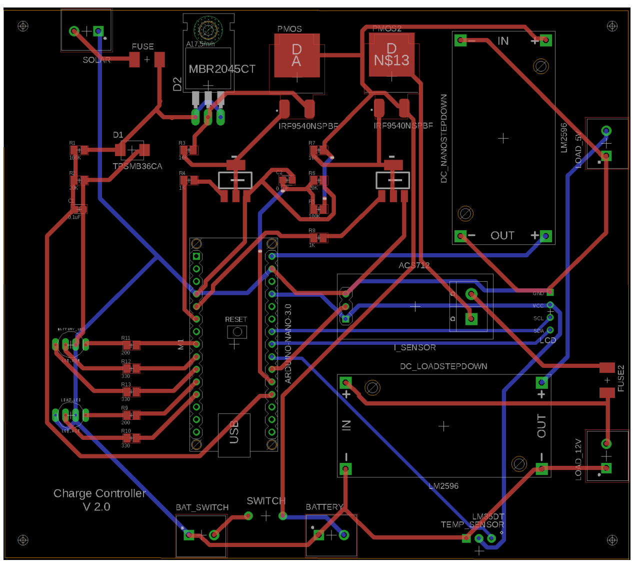


Fig. 10. Charge Control PCB layout

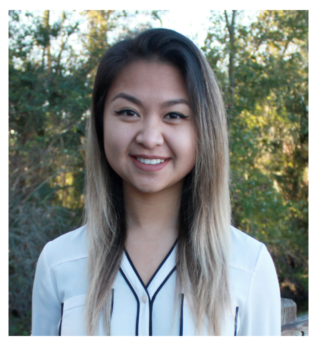
VII. Conclusion

This document outlines and reviews the key features and functions of our Senior Design project. EzRack is a product of the technical concepts and personal motivations that each group member has obtained throughout their time at the University of Central Florida. Over the course of two semesters, the team has been able to cooperate and communicate with one another to build a fully functioning prototype of a design carefully researched and planned. Although problems were faced throughout the developments of the project, they were ultimately overcome.

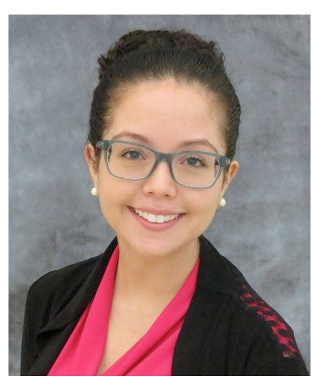
The goal of our project was to be utilized for bicycle storage around the University of Central Florida. The final production and realization of our project is a small-scaled version of what we envision the EzRack to be. This project was fully funded by the group members and therefore the budget was kept as low as possible. In order to achieve all of the specifications while maintaining an affordable budget, many components were chosen for their low cost along with reliability. The housing for the bike rack was built using materials that not only could offer durability but also were easily manipulated to fit our desires. Future implementations of this design may incorporate more bike slots and a more sophisticated build.

Even though the group was comprised of both Computer and Electrical engineering students, each member had the same responsibility of problem solving. With the completion of the project, each member is able to demonstrate an understanding of all of the components contained in the system. Work distribution allowed each member to gain experience as well as a sense of accountability in the team that will be beneficial in future endeavors.

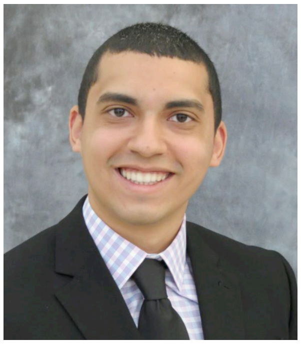
Biography



Amanda Chanthalangsy is currently a senior Electrical Engineering student at the University of Central Florida. She will be graduating in the Summer Semester of 2018. She plans on obtaining a Master's Degree in Business Administration in the near future and ultimately pursuing a career in sales engineering.



Vanessa Garcia De Quevedo is currently a senior Computer Engineering student at the University of Central Florida. She will be graduating in the Summer Semester of 2018. She plans on obtaining a Master's Degree in Computer Engineering in the near future.



Joel Gonzalez is currently a senior Computer Engineering student at the University of Central Florida. He will be graduating in the Summer Semester of 2018. He will be pursuing a career in computer software and hardware engineering. Joel also hopes to obtain a master's degree in Computer Engineering in the near future.

Trung Luu is a Computer Engineering student at University of Central Florida that is graduating in the end of Summer semester of 2018. Aside from engineering school, he was also part of the 920th Aeromedical Staging Squadron as a medic at Patrick AFB. His interests in computer technologies motivated him to pursue computer engineering. He is currently seeking an opportunity to continue to serve and to protect The United States of America through an employment with The Department of Defense. His near future plan is to further his education through a Master's Degree in System Engineering. You can reach him at Trung_LuuM@knights.ucf.edu.

ACKNOWLEDGEMENT

We wish to offer our gratitude and appreciation for the support and knowledge of the ECE Department at the University of Central Florida. This project has been made possible with the guidance of both Professor Samuel Richie and Professor Lei Wei throughout the past two semesters. Additional thanks to the professors who have kindly agreed their time to review our project.

REFERENCES

[1] "ATmega328P." *Microchip Technology Inc*, www.microchip.com/wwwproducts/en/ATmega328P.

[2] "Firebase Realtime Database | Firebase Realtime Database | Firebase." *Google*, Google, firebase.google.com/docs/database/.

[3] RakeshRon. "L293D Motor Driver IC." *RON ROBOTICS*, 20 Apr. 2018, www.rakeshmondal.info/L293D-Motor-Driver.

[4] "DS3231 Extremely Accurate I²C-Integrated RTC/TCXO/Crystal - Maxim." *The Fundamentals of an ECDSA Authentication System - Tutorial - Maxim*, www.maximintegrated.com/en/products/digital/real-time-clocks/DS3231.html.