Helmet Safety Indicator

By

Ali-Berk Eroglu

Bhavish Bhattar

Santan Katragadda

Final Report for ECE 445, Senior Design, Spring 2020

TA: Stephanie Jaster

8 May 2020

Project No. 35

Based off of Project No. 8 from Fall 2017, “Bicycle Street Notification System”

Abstract

Bicycling is an extremely popular sport and method of transportation that poses significant risks in urban conditions. Signaling as a bicyclist is difficult in certain environments and mass unfamiliarity with signals or a lack of visibility can cause a vehicle operator to harm the bicyclist. Group 8 from Fall 2017 sought to solve this problem by creating a design that took inputs from a speedometer and accelerometers to light up LEDs that were attached to the front and back of the bike. Our solution uses brake and turn button inputs to activate LED indicators on the helmet of the user for a more clear viewing angle for the vehicle operator. The device also uses an ultrasonic sensor to warn the user of driver behind them as well as drivers of the presence of a cyclist for a safer experience.

Contents

1 Second Project Motivation

1.1 Updated Problem Statement

In 2017 the number of bicyclists hit a staggering 47.5 million individuals [10], or about 12% of the total American population [4]. With this many bicyclists, cities have taken notice to accommodate cycles through separated bike lanes and paths. Even then, a vast majority of cities do not accommodate bikes and bicyclists are forced to ride alongside vehicles on the open road. In these situations, bicyclists are placed in a position of elevated risk and require more measures to ensure their safety. The two most risky situations are turning at intersections and getting overtaken by vehicles or other bicyclists [2]. Bicyclists are expected to indicate speed changes and turns via their arms, but many drivers are not aware of these signals, and at high speeds, extending one arm out of the bike is less than ideal for the bicyclist. These arm signals ultimately fail at providing a safe and effective method of indicating messages to car drivers or other bicyclists. In addition, bicyclists ride very close to cars and are only aware of cars near them when they physically turn their head, which is also a potentially risky maneuver leaving them unaware of potential threats in front of them.

In 2018, 2% of motor vehicle crash fatalities in the United States were bicyclists, resulting in 857 deaths. There were in total 45,000 reported bicycle accidents [2]. Many of these motor vehicle accidents were preventable as two of the primary causes for these accidents were drivers not properly acknowledging the bicyclists and the cities not being properly designed to accommodate bicyclists and drivers sharing the road. This shows that at least until American cities reach a higher safety standard for bicyclists, there needs to be a method that bicyclists can elevate their safety on their cycles.

1.2 Updated Solution

Our proposal is to create a safety system that is simple for riders to use and install on a variety of bikes. The safety system will have two parts: indicators for other drivers and indicators for the bicyclists. The indicators for the driver would work by having attachable triggers the user could attach to their handlebars and brakes which will send signals via bluetooth to a light attachment on the user's helmet. The lights are placed above the user's helmet so that they are directly in the line of view for cars behind the rider and they will display left and right turns as well as braking. In order to provide the rider feedback of approaching cars or other bicyclists from behind, there is an on the back of the helmet. This sensor will alert the bicyclist via a light up indicator the user can attach to the handlebars. It will also alert cars of their proximity to the cyclist by lighting up the entire LED fixture on the helmet to indicate they are approaching too close.

As urban infrastructure still lags behind the growth of bicycling, there are many solutions for providing light indicators for cycles. A vast majority exist more as a passive indicator that simply flashes to grab the attention of motorists but still relies on the bicyclist to use their arms to indicate changes in speed or direction. Even for solutions that exist to indicate turning and braking, these indicators are placed underneath the saddle of the bike, putting it in a position that is not directly in the view of a driver of a motorized vehicle. *Lumos* is one company that does provide a solution where the indicators are attached to the helmet [6]. *Lumos,* though, is a very expensive solution and, unlike our proposal, they do not offer any form of indication to the cyclist themselves. Our solution provides unparalleled safety for the bicyclist as they are able to provide clear indication to surrounding motorists, while also being able to receive information pertaining to whether vehicles are close behind them or not. In addition, by having a system that is primarily wireless, we have designed a system that is simple to install and remove. Our project was inspired by Project 8 from Fall 2017, the Bicycle Street Notification System. By being able to provide a simple to install and use system, with turn signals mounted to the user's helmet and a rider warning indicator, our solution provides safety and simplicity that eclipses the Bicycle Street Notification System.

1.3 Updated High-Level Requirements

- Helmet LED indicators are activated when turning or braking based on the trigger inputs on the handlebar and the handlebar LED indicator must activate based on the input from the ultrasonic sensor on the helmet.
- Light indicators and buttons are easily attachable and stay securely fixed to the bicycle and helmet when attached.
- The ultrasonic sensor mounted on the helmet must pick up objects and vehicles that are within 3 m behind the bicyclist.

1.4 Updated Visual Aid

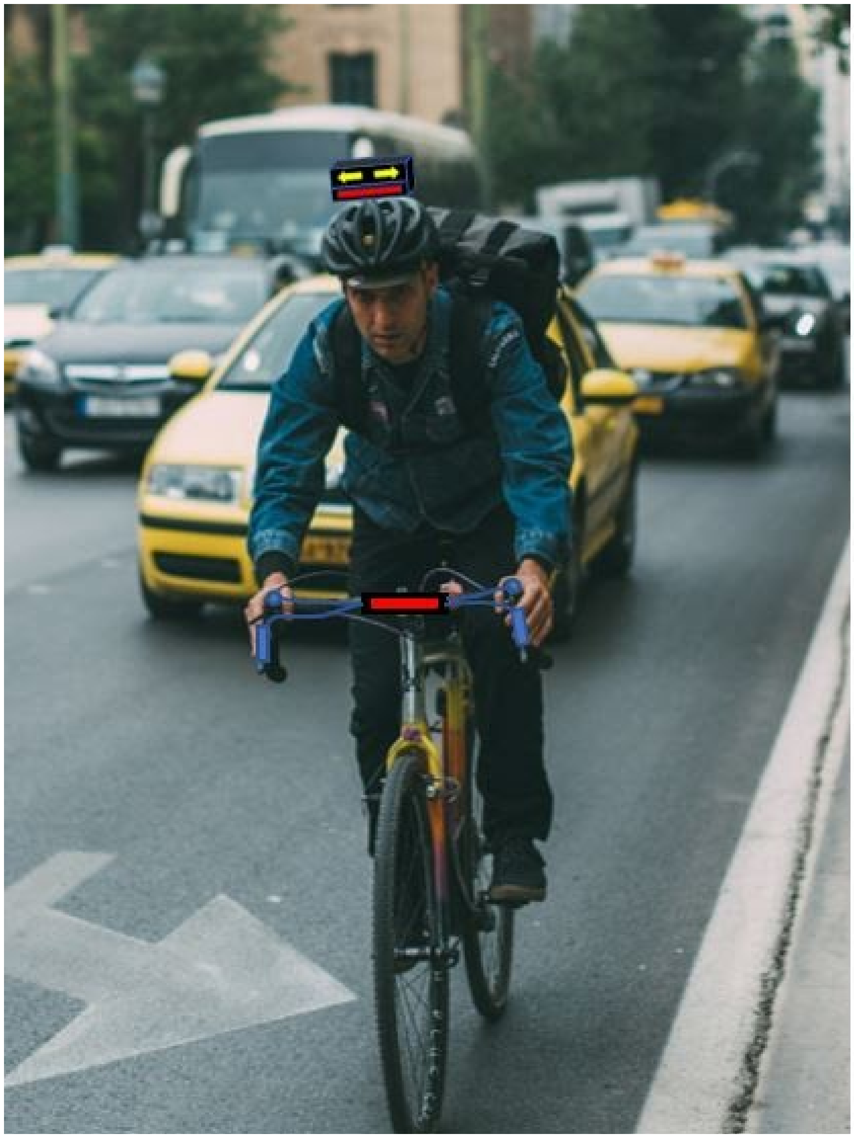

Figure 1. Visual aid of a bicyclist using the Helmet Safety Indicator at a busy intersection.

As shown in Figure 1, the user has a component of the device attached to the top of the user's helmet as well as a component that attaches to the user's handlebars. The helmet component has LED lights that indicate the direction the bicyclist is turning to vehicles and other bicyclists in front and behind the user. The helmet component also has a red LED bar that would indicate when the cyclist is braking. All the rear LED lights will glow when a motorist or bicyclist is within 3 meters of the user in order to draw attention to the bicyclist. The handlebar component has a red LED panel fastened to the handlebar to indicate when an object is within 3 m behind the cyclist. The handlebar component also has turn signal and brake triggers.

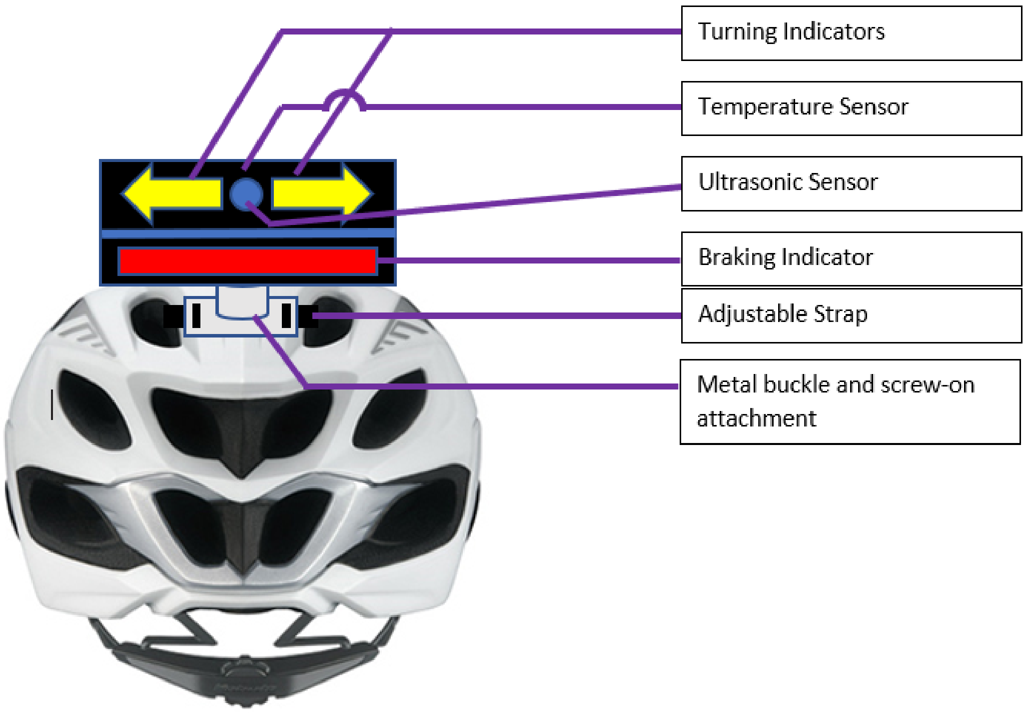

Figure 2. Rear view of the helmet component with turning and braking LED indicators.

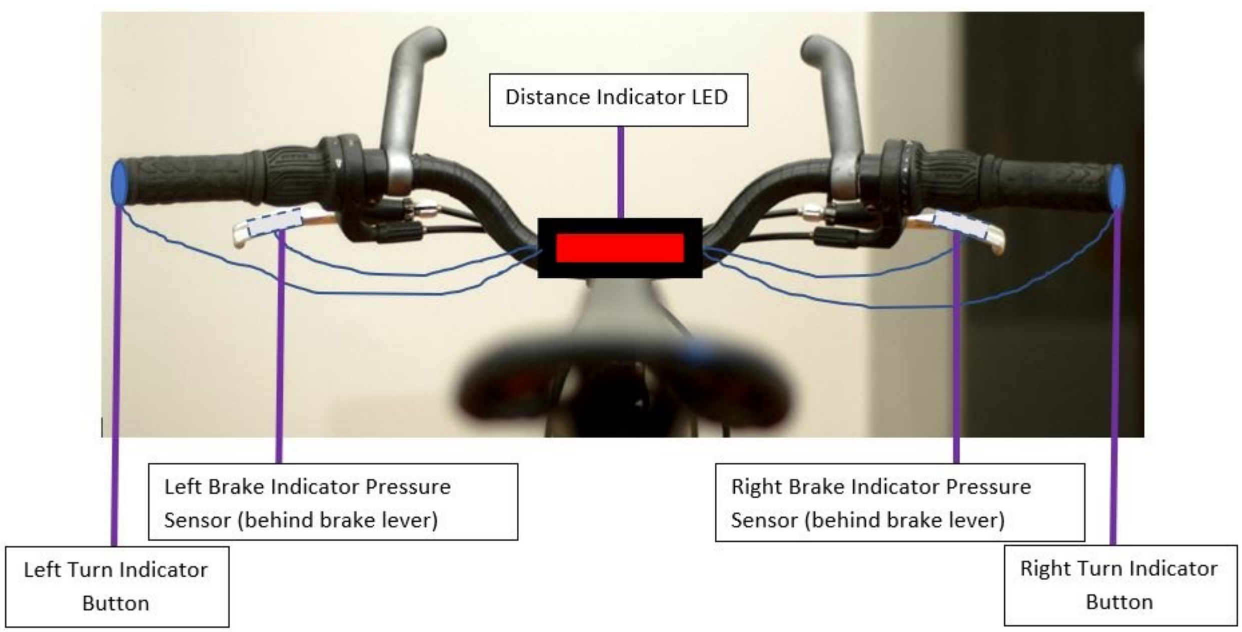

Figure 3. User view of placement of distance indicator LED, brake and turn indicator buttons on traditional bicycle handlebars.

The physical design of the product will consist of two separate parts. The first part is attached to the top of the rider's helmet through an adjustable strap as shown in Figure 2. The turning and brake indicators on the helmet attachment will screw onto a metal plate attached to the adjustable strap for secure attachment. The LEDs on the turning and braking indicators are visible from 100 feet away. The handlebar component shown in Figure 3 is strapped to the rider's handlebar in a similar fashion to the helmet component and will have four wires coming out of it for the brake and turn indicator buttons. The brake pressure sensor is on the front of the brake where the user's hand will go when braking. In Figure 3, the brake indicator buttons would actually appear on the other side of the brake lever, but we have included a box for it so the location of it is understood. The helmet and handlebar components including the wires are fully enclosed and weather resistant. The battery, microcontroller, etc. is stored in the large boxes for both parts.

1.5 Updated Block Diagram

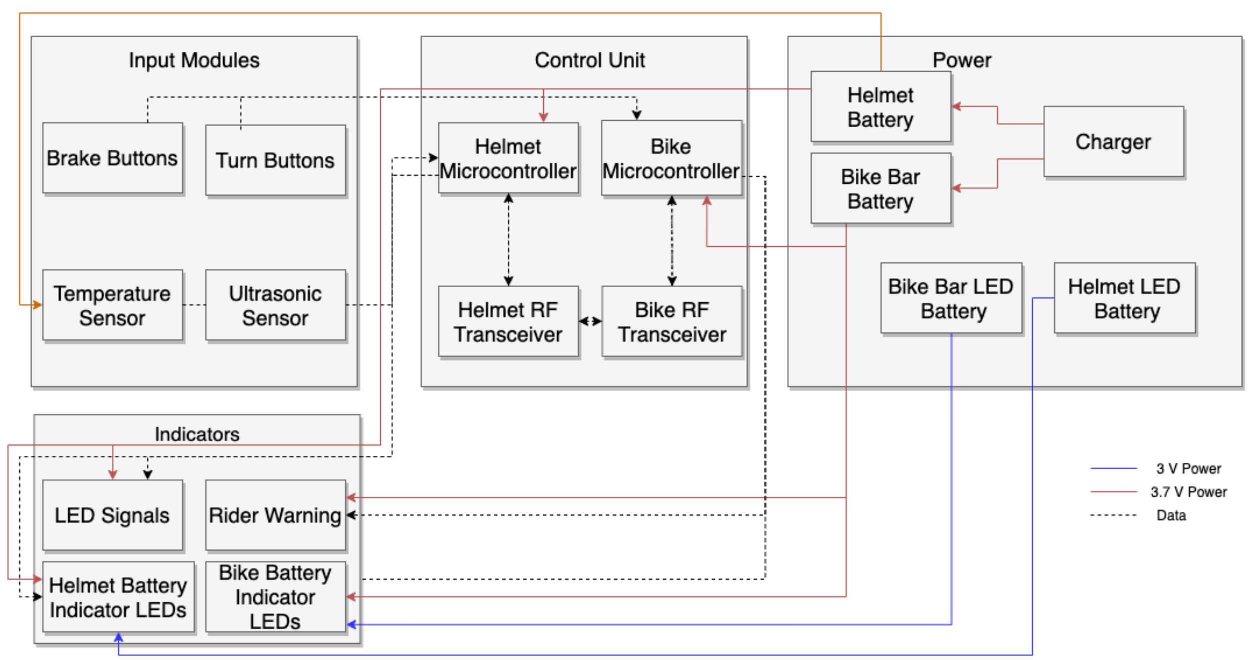

Figure 4. Overall block diagram showing power and data connections between functional units.

The device is designed with four primary submodules, as shown in Figure 4, which is spread over two separate devices. The first submodule is the input module, this consists of the buttons to indicate braking and turning as well as the ultrasonic sensor and thermal sensor on the helmet. These input modes will communicate with the control unit module. Both of the devices which are attached to the handlebars and the helmet will have their own control unit components. The inputs on the handle bar will communicate with the control unit component that is within the handlebar attachment and the sensor will communicate to the helmet's control unit component. From the control unit RF transmitters are used to send data between the two separate devices which are then relayed to the indicator submodule which will illuminate the appropriate indicators on the device. Finally, there is the power submodule, both of the components will have a 3.3 volt lithium ion battery which is charged via a Micro-USB port.

2 Implementation Details and Analysis

Due to the Covid-19 outbreak in the spring of 2020 and the corresponding lockdown, our team was unable to move forward with a majority of the implementation for our design. Due to this, our implementation primarily consists of our schematic, PCB design and potential software implementation for the ultrasonic sensor and RF transmitters. In addition, we conducted an analysis on the effectiveness of the ultrasonic sensor particularly in regards to detecting the curves of cars.

2.1 Schematics

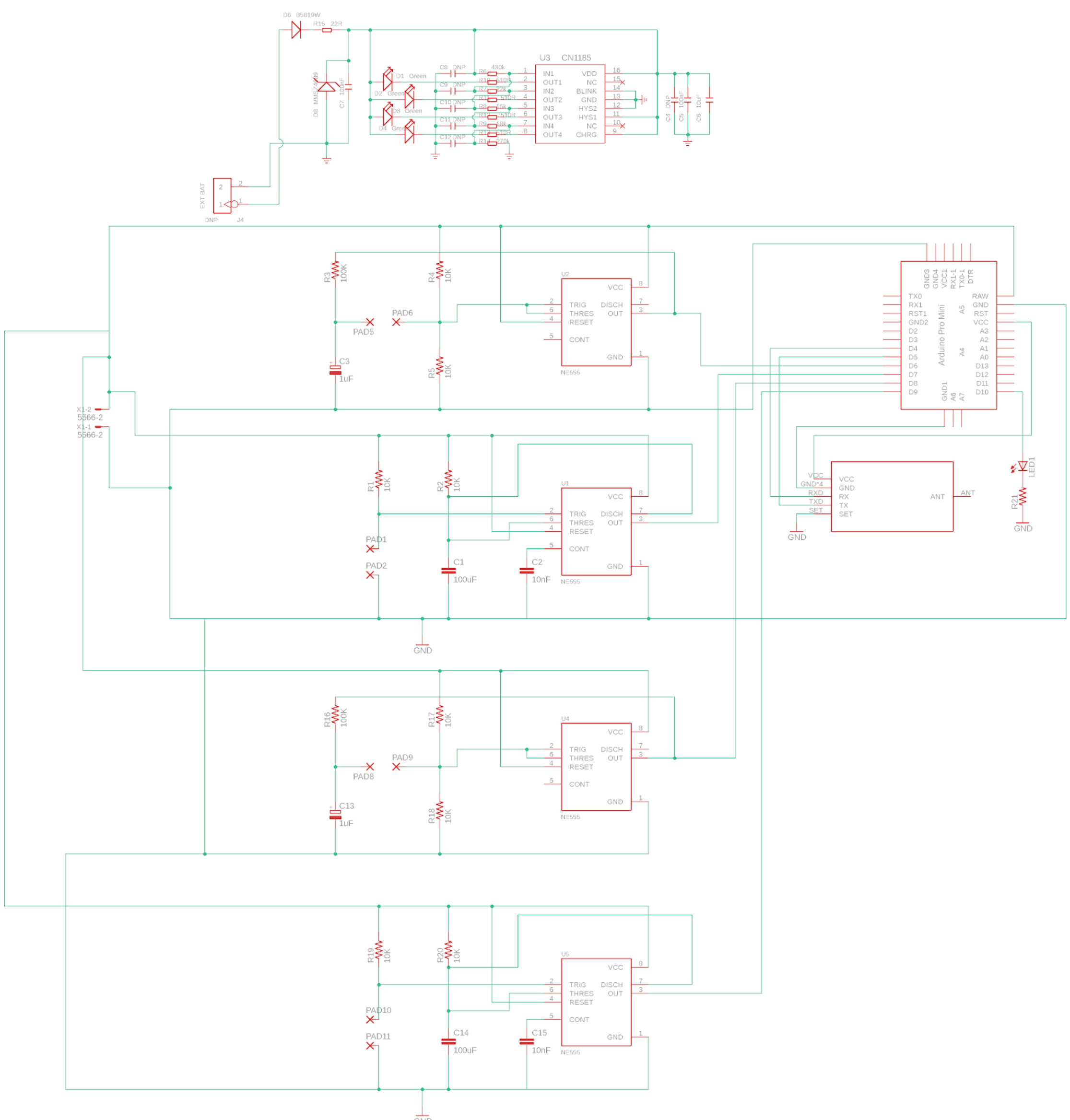

Figure 5. Schematic for the Integrated Bike Bar Circuit.

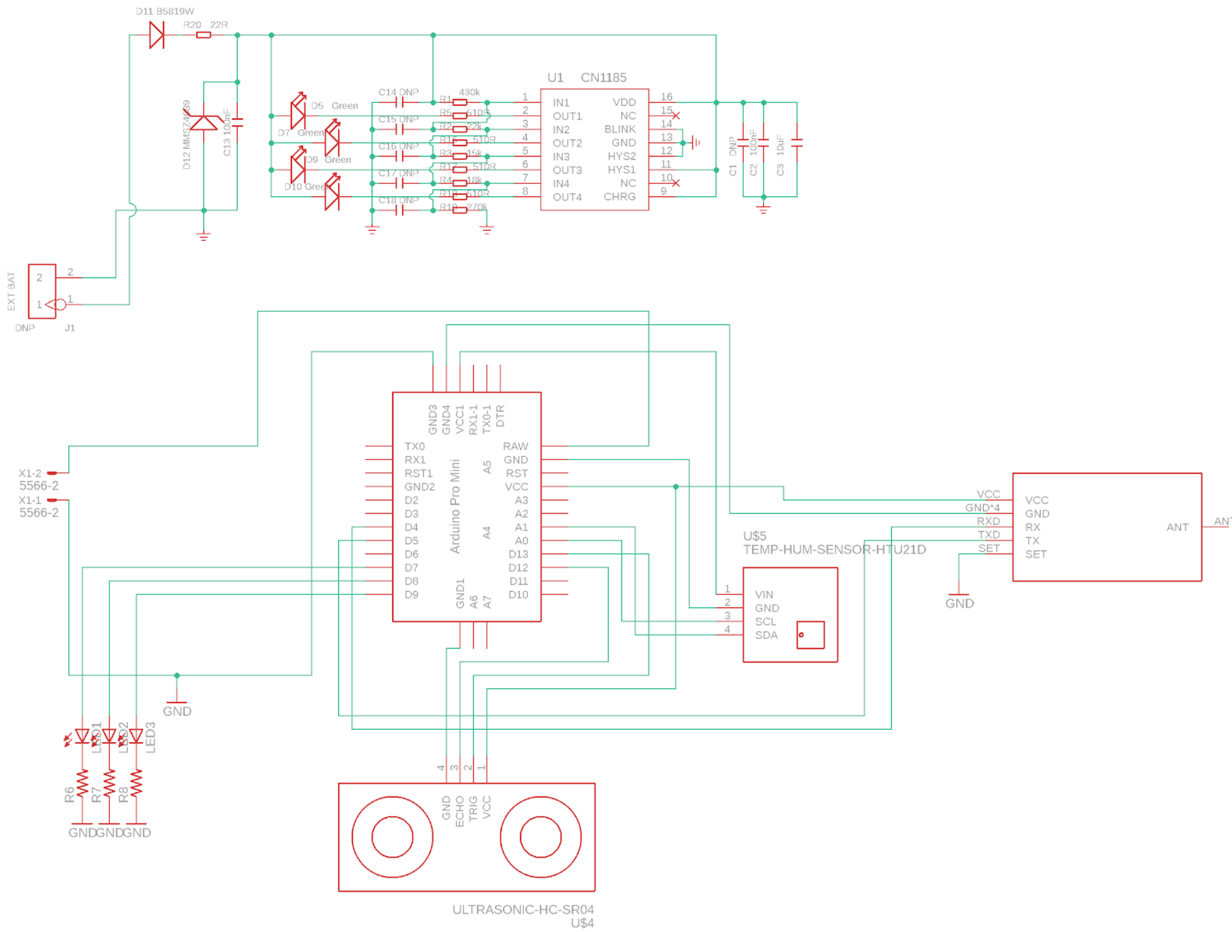

Figure 6. Schematic for the Integrated Helmet Circuit.

Since we have two separate devices, we had two separate circuit schematics on EAGLE which came out as two separate PCBs and are placed on the bike bar and helmet. The bike bar schematic shown in Figure 5 has the battery indicator LED circuit, two monostable circuits for brake buttons, two toggle switch circuits for left and right turn buttons, Arduino Pro Mini to process all these inputs and RF transceiver to transfer the data to the helmet. Also, it has the rider warning LED as the output which lights up depending on the input coming from the helmet via RF transceiver. The helmet schematic shown in Figure 6 has an ultrasonic sensor and thermal sensor as the inputs and Arduino Pro Mini to process the inputs and RF transceiver to transfer the data to the bike bar. Also, it has left turn, right turn and brake indicator LEDs as the outputs which lights up depending on the input coming from the bike bar via RF transceiver.

2.2 PCB Design

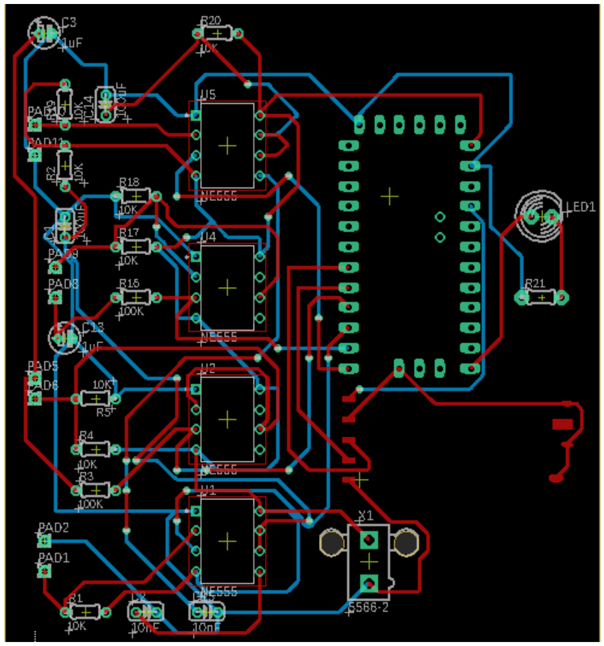

Figure 7. PCB Layout for the Integrated Bike Bar Circuit

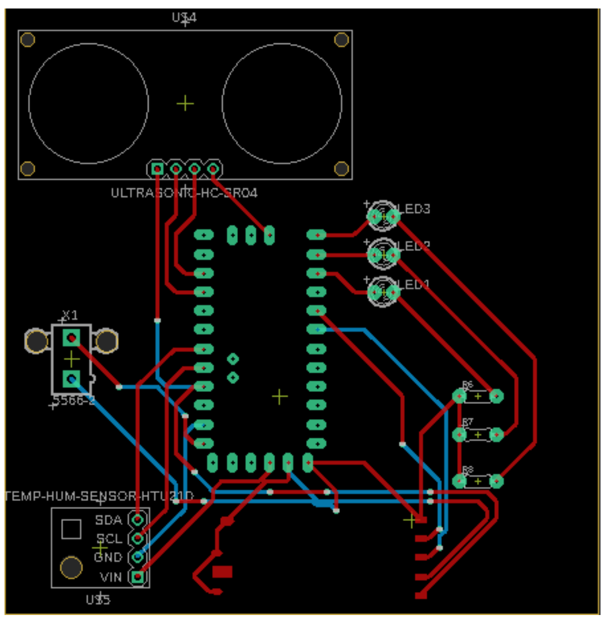

Figure 8. PCB Layout for the Integrated Helmet Circuit

In Figures 7 and 8, we created PCBs of the schematics we drew in the previous section. There are two separate PCBs since we have two devices. All the components have their own footprints except the buttons since they can be placed anywhere on the bike bar, so pads are placed for buttons to connect the buttons to the circuit.

2.3 Software

In our project we have utilized two Arduino mini microprocessors which is where we have implemented software. The primary functions that we looked into and researched for the implementation was for programming the ultrasonic sensors, the HC-SR04 as well as programming the RF transceivers. Including this software was vital in ensuring the proper communication between the helmet and handlebar component, what was integral to maintaining the lightweight design of our device. In addition the software ensured the proper functionality as well as precision and accuracy of the ultrasonic sensor.

2.3.1 HC-SR04 Program Implementation

For the Arduino Pro Mini to properly communicate with the HC-SR04, we ensured that the Arduino was able to properly initiate the device and collect the appropriate information for it, in addition we had to also ensure that there was an appropriate offset in place to ensure that any discrepancies in data that comes from changes in temperature is accounted for.

```
#define MAXRANGE 300
#define MINRANGE 20

const int trigPin = 13;
const int echoPin = 12;

HTU21D HTU;

void setup() {
      Serial.begin (9600);
      pinMode(trigPin, OUTPUT);
      pinMode(echoPin, INPUT);

      HTU.begin(); }
```

```
void loop() {

  long duration;
  float distance;
  float temp=HTU.readTemperature();
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(11);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
```

Figure 9. The first part of Arduino Code for ultrasonic sensor setup.

In Figure 9, we are showing the code used for initializing variables for the Arduino and then starting the devices. This code is based on information we found on Arduino's project hub[8]. We first define the MAXRANGE and MINRANGE at 3 m and 20 cm respectively as we had defined in our design. Next we set the trigger pin and echo pin which correspond to which pins are responsible for the transmitter and receiver of the ultrasonic pulses. Then we are identifying the HTU21D, our temperature sensor, as HTU. After the initialization, the setup is responsible for starting up all the devices being used. Next we have initialized the execution loop of the Arduino, we set up the triggering of the pin with the appropriate delays to send a signal from the trigger pin until it is collected by the echo pin to ensure that the data is properly recorded. These delays are in compliance with our requirement of an 80ms recording period. Using this we are able to deduce a duration that it took for the signal to bounce back.

```
float speedOfSound = 331.3 + 0.606 * temp;
float distance = (duration / 2*10000.0) * speedOfSound;
if( distance > MAXRANGE || distance < MINRANGE){
      Serial.print("Out of Range")}
else{
  Serial.print("compensated distance: ");
  Serial.print(compensatedDistance, 1);
  Serial.println(" cm\n");
}
  delay(80);
}
```

Figure 10. The second part of Arduino Code for ultrasonic sensor setup.

This section of code in Figure 10 is responsible for calculating the speed of sound with a variable temperature and then outputting the appropriate distances. Due to how the density of molecules change at different temperatures, the higher the temperature, the faster sound travels. This relationship is shown below in Eq (1) and is what we have coded above.

$$Speed\ of\ Sound\ (m/s)\ =\ 331.3\ +\ 0.606 * T \tag{1}$$

Here we will calculate the distance to an object in centimeters by calculating using the duration that we have in microseconds. Notice that it is divided by 2*10000. It is divided by 2 because the duration accounts for the time for the signal to go to the object and return. It is divided by 10000 because speed of sound is calculated in meters per second but the duration is in centimeters per microsecond and the distance needs to be given in centimeters.

2.3.2 HC-12 Program Implementation

The HC-12 is the RF transceiver that we are utilizing, and we are programing the Arduino Mini to accept and read out data that the transceiver receives as well as send out messages. This code is programmed into both Arduino Minis that are in the device as both of them will interface with the HC-12 RF transceiver.

```
const byte HC12RxdPin = 4;
const byte HC12TxdPin = 5;


SoftwareSerial
HC12(HC12TxdPin,HC12RxdPin);

void setup() {
  Serial.begin(9600);
  HC12.begin(9600);
}
```

```
void loop() {
  if(HC12.available()){
    Serial.write(HC12.read());
    }
  if(Serial.available()){
    HC12.write(Serial.read());
  }
}
```

Figure 11. The Arduino code for the usage of the RF Transceiver.

The code shown in Figure 11 is utilized to set up the RF transceiver so that it can send messages between the two components [9]. It first sets pins 4 and 5 to receive and transmit data respectively. It then serializes the pins in order to avoid race conditions. Finally for the initialization, the serial pin and the transceiver are initiated. The next part is the main execution loop. The if statement checks whether there is data stored in the HC-12 data buffer. If it is the case then it accepts it and reads it to the Arduino, where the data received is used. If there is data in the Arduino's data buffer then it writes to the HC-12 data buffer and is transmitted. This simple loop will handle communication between the two transceivers.

2.4 Distance Detection Optimization

In addition to programming we need to ensure, physically, that the sensor is also able to detect within the necessary range. As stated we are utilizing the HCSR-04 sensor, which is placed atop the helmet, and has a detecting range between 20 cm and 4 m. The requirement for the sensor is that it is able to detect objects, primarily vehicles and other bicyclists that enter within 3 m of the rear of the bike.

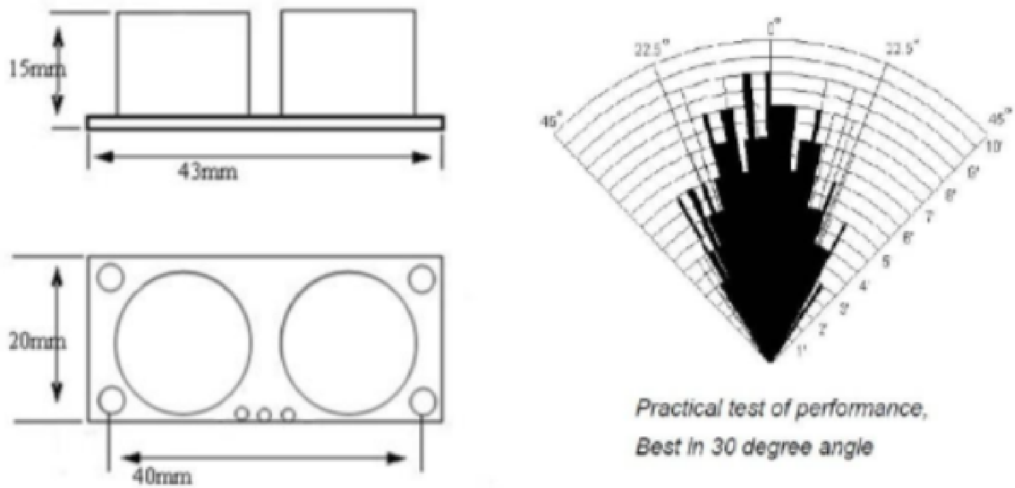

Figure 12. The performance vs. the angle of the HC-SR04 ultrasonic sensor [7].

We looked into optimizing the detecting angle in order to maximize the range at which the sensor was able to detect objects. Since the helmet and the approaching object for the ultrasonic sensor to detect won't stay at the same angle due to the bicyclist shifting their seated position or head, the performance of the ultrasonic sensor was analyzed at the possible range of angles. Figure 12 illustrates the performance testing graph of the sensor at different angles. It shows that ideal detection is done at a 30 degree angle, but is also still good at the range between 20-40 degrees. Keeping the sensor's detection within this range is achieved via the helmet mount. The design will optimize maintaining the detection angle between the 20 - 40 degree range while the rider is sitting on their cycle saddle.

3 Second Project Conclusions

3.1 Implementation Summary

In the workflows for the implementation we have seperated the work between hardware and software. Ali-Berk focused on the hardware creating the schematics as well as translating them with the design constraints into a viable PCB. He designed the schematics for the circuits in both the helmet and handlebar components of the device. Santan established the requirements for the proper functioning of the hardware components and provided insight into how the schematics are assembled as well as worked on creating the RF transceiver code for the Arduino so that messages are sent between components properly. Bhavish focused primarily on the software component of the design he worked on programming the Arduino to collect the data from the ultrasonic sensor and correcting the data to ensure the precision necessary, he also provided the requirements that the schematic needed to meet for the proper functioning of the ultrasonic sensor, RF transceiver and thermal sensor.

3.1.2 Hardware Summary

The first major component of our setup was to build up the necessary circuits before doing any software implementation for Arduino Pro Mini. Since we already built up a voltage indicator circuit for our Protect-U project, we also used it in this project, both for the helmet device and the bike bar device. The voltage indicator works with CN1185, which is a four-channel voltage monitoring chip that gives a quantized output depending on the input coming from the battery.

In this project, designed monostable circuits for brake inputs, and toggle switch circuits for turn inputs and they're connected to the digital pins of Arduino Pro Mini. The reason for building up a monostable is that we wanted the brake indicator to light up for one more second after the brake button gets released, so that we arranged the time constant RC to specify the time. We also built up a toggle switch circuit for turn indicators, which is implemented by an equal quantization of supply voltage, threshold voltage and trigger voltage of the 555 timer chip. The ultrasonic sensor, RF transceivers and LEDs are also connected to the digital pins of the Arduino Pro Mini, and the thermal sensor for the ultrasonic sensor is connected to an analog pin.

3.1.2 Software Summary

The second major component of our setup was including the software portions for our design. We have typed out code for the Arduino Mini in order to properly set up the ultrasonic sensor and the RF transmitters. Initiating these devices are the most software intensive sections of the project, thus having down the basic code to get these components up and running is the most important step we can take from a software perspective.

In the code for the ultrasonic sensor, we were able to optimize the functionality in two ways. First we had taken a step to discard data registered outside of the range of 20cm to 4 m, which reduced the ultrasonic sensor triggering in situations where it should not be. This is an important step to take because it upholds the integrity of the information coming from it, as the more times it misfires, the less the user will consider information it is displaying. We also programmed necessary elements to include a thermal sensor and programatically use it to correct the output of the ultrasonic sensor. The speed of sound can drastically vary in different temperatures and this can lead to a significant error in the distance being output by the ultrasonic sensor. In order to combat this we utilized how the speed of sound varies based on the temperature to adjust the distance output by the ultrasonic sensor, to make sure the read out on the Arduino was more precise. Finally we also typed up the code necessary to utilize the RF modules to send messages between Arduinos. This sets up the connection between the two components and makes them have a lightweight and easy to utilize design.

3.2 Unknowns, Uncertainties, Testing Needed

There are many additional obstacles to completing the development of this product due to the current pandemic. Shipping and manufacturing of parts and PCBs are limited and may take extended periods of time to arrive. The team is also unable to use the lab or meet in person as members live in different countries. Without the materials, the team is unable to build the product or subsystems to test them physically.

We could not test the ultrasonic sensor as we needed the physical part. A simulation of the code for the microcontroller does not properly simulate the physical sensor and any potential adjustments. In order to test this, we would connect the ultrasonic sensor to the Arduino and run the provided code. We would test the angle and distance at which the ultrasonic sensor picks up curved objects resembling a car. We also could not simulate the reaction of the ultrasonic sensor to curved objects. Our situation is similar for the temperature sensor as a simulation would not accurately test the physical sensor.

The team is also not sure of the visibility of the LED indicators during the daytime. This will require testing of the LEDs with the outer lens. Many different people will need to confirm it's visibility for verified results.This may lead to the addition of LEDs and alternative batteries. The team did not have access to LEDs or the batteries to physically test this component as well. The indicators are tested for visibility from at least 100 feet away.

Our team plans to use a 3D printer to manufacture the outer casing of our product. The creation of this casing is one of the final parts of the project. Once the fastening device, and all product materials and subsystem requirements have been created and tested, the finalized dimensions of the complete product are used to create the CAD models for the 3D printer. The team did not have CAD and felt that the simulation of the 3D model is not useful in testing the actual durability of the outer casing. The team would test the casing under heat of up to 100 degrees Fahrenheit and underwater to ensure that the device is weatherproof.

3.3 Ethics and Safety

When working on developing this device, our group will take many precautions to ensure our safety and the safety of those around us. All members of the group have completed and passed the required lab safety training and will follow Campus Environmental Health and Safety policy #RB-13, in which we are responsible for maintaining and creating a healthy and safe environment for our team and the UIUC community [3]. We will make sure to not modify circuits while the power is connected, as well as be aware of any burning parts to make sure we do not start a fire. The lab is also equipped with a fire extinguisher in case of emergency. Additionally, our group will have at least two members present when working in the lab to help prevent accidents. In the situation the group needs to work with equipment that the members of the group are unfamiliar with, we will ensure that there is the necessary supervision to validate the proper usage of the equipment. This is in accordance with #6 of the IEEE code of ethics in which we will only undertake tasks for others only when qualified by training or experience [5]. Finally, the group will not use other individual's work without the proper citations, and certify that the idea and design of the project is original and unique [1].

Since this device is used outdoors and can potentially have close contact with the human body, extra precautions are necessary to protect the user. The outside of the helmet LED box and the handlebar LED box will have insulating, waterproof, heat-resistant casing that will make sure no water can come in contact with the electrical components stored inside. Heat resistant plastic is used to prevent the casing from warping under heat from the sun. This will minimize the risk the user faces when using the device while it is raining outside as well.. Although lithium ion batteries have become the standard in rechargeable consumer products, there is still a level of risk when using them. The team will use a battery purchased from a reputable company with a protection circuit built into the battery. The protection circuit will prevent the battery voltage from getting too high or too low and will cause the battery to cut out at 4.0V. This also prevents the other circuitry in the device from shorting or being damaged by improper power input. The noise susceptibility of the 5.0V battery is lower than a 3.3V battery, making it more stable. The team will use the manufacturer recommended charger for the battery to prevent malfunctions when charging the battery. The charging will have three stages: a preconditioned charge, constant-current fast charge, and a constant-voltage trickle to top the battery off. These stages prevent overcharging of the battery. The product will also have a voltage detection circuit that can visibly notify the user when the battery has charged. The exposed wires on the handlebar will also have weatherproof, heat resistant insulation to minimize risk from exposure to the elements.

Since we are promoting this product as a replacement to traditional hand signals, it is imperative that our product works under duress and can work for up to eight hours. The team will conduct rigorous testing on prototypes to ensure consistent results. Bicycle signaling is required by law in the United States. In order to prevent situations where the device has lost battery and signals are not being output, visible battery indicators are on the handlebars. Riders are encouraged to monitor the battery levels of both the helmet and handlebar components to ensure that the device is still charged and responding. This is in accordance with #5 in the IEEE Code of Ethics in which we are urged to improve the understanding of individuals on the societal implications of our technology [5]. Additionally, in compliance with #1 in the IEEE Code of Ethics, we will ensure that all safety precautions are taken in the construction of the device to ensure that there is very minimal risk of harm to the user [5]. If any such risk exists, information would be provided to the user to inform them of the proper usage to avoid said risk.

Another consideration is that the distance sensor on the handlebars component of the product may distract the user if it is triggered by objects that are very far away or not directly behind the user. Consecutive misfires may cause the user to disregard the light entirely and pose a risk. The team will thoroughly test the precision of the ultrasonic sensor with different objects to ensure accurate distance reading. In addition a temperature sensor is used to improve the microcontroller's understanding of the ultrasonic sensor readings. Temperature modifies the readout of the ultrasonic sensor as, changing temperature modifies the speed of sound. By using the temperature sensor to adjust for this, we ensure that the device is capable of measuring accurately regardless of the temperature the rider is using it in.

3.4 Project Improvements

Given more time we plan to improve this project in three primary ways, through a more sleek and aerodynamic design for the helmet mount, switching out the ultrasonic sensor for a more exact optical sensor such as LiDar and a focus on trying to reduce costs.

For a more aerodynamic design we plan to reduce the height of the camera above the helmet and instead try to create a mount for the rear of the helmet. The current design opted for the helmet mount because that was an easier way to guarantee universal usability given the time constraints that we had. The back of a helmet is the ideal location to mount the lights and sensors but unfortunately it is much harder to design a universally viable mount for the rear of a helmet given the vast array of different helmet shapes out there. With the extra time of a year, we would have the time to iterate on a design that can fit more on the rear of the helmet, creating a more sleek design that is far more aerodynamic than the indicator sitting above the helmet.

The second major improvement is to try to utilize an optical sensor that is more exact and more capable of detecting objects through slight obstructions such as inclement weather. We opted to utilize ultrasonic sensors due to their inexpensiveness as well as the familiarity we had with the technology but unfortunately the inexpensiveness and ease of installation comes at the cost of accuracy. While we were able to find work arounds to some of these issues of accuracy, such as how temperature affects readouts, and programmatically eliminate them, considering the importance of accuracy to a safety device, with more time we will look into optical distance sensors for the Driver Warning component.

Finally, it is very important to look into reducing costs for the device. The device has a cost of $188 to produce and we would hope to bring this down significantly. There are many devices where we could look into reducing the cost, such as utilizing cheaper microcontrollers instead of the Arduino Mini or finding cheaper batteries that the design can use. By having the time to make small price optimizations to each of the components on the parts list, we can provide the device at a much more affordable cost to the consumer.

4 Progress Made on First Project

For Protect-U, our initial senior design process, we had made a lot of progress, particularly with the planning of how we would assemble the system. After our design review, the steps we took prior to starting the second project was implementing the feedback we received from the design review into our project, and mapping out the first iteration of our PCB design.

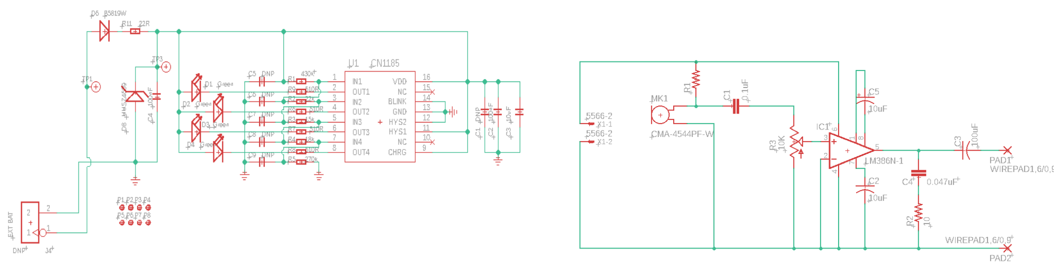

Figure 13. Voltage Indicator Circuit and Microphone Amplification Circuit

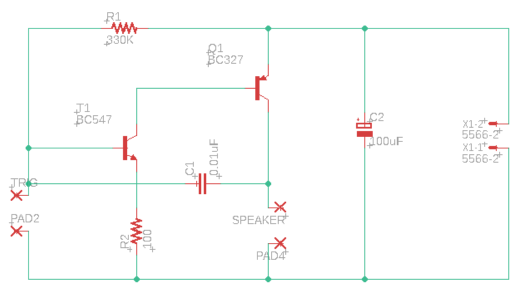

Figure 14. Speaker Circuit

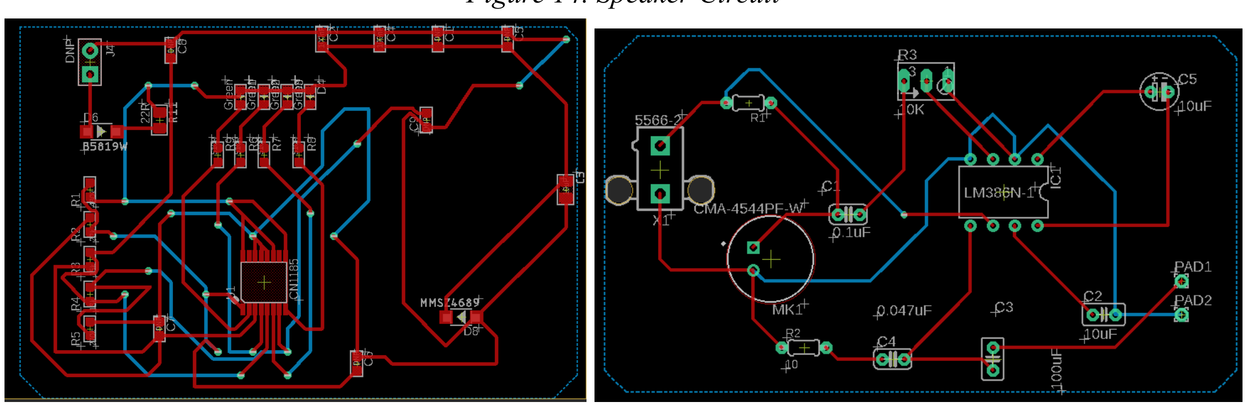

Figure 15. Voltage Indicator Circuit and Mic Amplification Circuit

In Figures 13 and 14, we have shown the schematics that we had put together for our Protect-U design. Upon completion by that design, we moved forward to creating the PCBs for the design which are shown in Figure 15.

References

[1] “Article 1, Part 4 - Academic Integrity Policy and Procedure.” *Part 4, Article 1 " Student Code " Illinois*, Illinois.edu, studentcode.illinois.edu/article1/part4/1-402/. [Accessed: April 2, 2020]

[2] Bicycle Accidents in the United States. (2019, December 1). [Online]. Available: https://www.peoplepoweredmovement.org/bicycle-accidents-in-the-united -states/ [Accessed April 2, 2020]

[3] “Campus Environmental Health and Safety,” Campus Administrative Manual. [Online]. Available: https://cam.illinois.edu/policies/rp-13/. [Accessed April 17, 2020]

[4] Fuller, S. (n.d.). Topic: Cycling. [Online].Available: https://www.statista.com/topics /1686/cycling/. [Accessed April 2, 2020]

[5] Ieee.org, "IEEE IEEE Code of Ethics", 2020. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: April 2, 2020]

[6] Lumos Helmet - A Next Generation Bicycle Helmet. (n.d.). [Online]. Available: https://lumoshelmet.co/. [Accessed: April 2, 2020]

[7] Raspoid API. (n.d.).[Online]. Available: http://raspoid.com/download/datasheet/HCSR04. [Accessed: April 17, 2020]

[8] “Ultrasonic Sensor HC-SR04 with Arduino Tutorial”. (n.d.). [Online]. Available: https://create.arduino.cc/projecthub/abdularbi17/ultrasonic-sensor-hc-sr04-with-arduino-tutorial-327ff6. [Accessed May 4, 2020]

[9] “Understanding and Implementing the HC-12 Wireless Transceiver Module - Projects.” (n.d.). [Online] https://www.allaboutcircuits.com/projects/understanding-and-implementing-the-hc-12-wireless-transceiver-module/ [Accessed: May 4, 2020]

[10] Walljasper, J. (2018, February 19). 10 Reasons Bicycling Will Continue to Soar in Popularity. [Online]. Available: https://www.strongtowns.org/journal/2018/2/9/12-reasons-bicycling-will-continue-to-soar-in-popularity. [Accessed: April 2, 2020]