AUTOBRAKING SYSTEM

By

Gary Gong

Taiqing Ling

Weichen Qi

Final Report for ECE 445, Senior Design, Spring 2021

TA: Xihang Wu

01 May 2021

Project No. 50

Abstract

We designed and built a system that could automatically detect the STOP signs ahead the bicycle and alert the rider to stop the bicycle. If no action is taken by the rider, our system will smoothly and safely auto-trigger the bicycle's brake system and force it to halt. We hope our project could help bicycle riders to be more aware of the importance of STOP signs in the street and protect the safety of themselves and other vehicle drivers. In this paper, the detailed design ideas, block diagrams, function of each modules, future modifications and the general cost of the device will be presented.

Contents

1. Introduction

The purpose of our project is to improve the safety of bicyclists and, at the same time, reduce the STOP sign violation rate in the traffic laws. Currently, the traffic laws for bicycles are relatively not comprehensive enough, which leaves potentially unsafety for bicyclists. In 2018, 857 people in the United States died in bicycle accidents. That number was the highest since 1990 [1]. In 2016, about 71% of deaths due to bicycle accidents occurred in urban areas and 30% occurred at intersections [2]. It was recommended that all bicyclists should wear a helmet but no state has so far announced such a law to rule that, which caused the tragedies.

Furthermore, as one of the important methods of transportation in the world, the bicycle has never acquired equal attention to its development as other vehicles, like the car and the motorcycle. While autonomous driving has become a mainstream in the development of vehicles, many cars have installed intelligent driving assistance like lane-keeping, blind-spot detector, auto-parking. Motorcycles also have riding assistance products. All these products help avoid traffic accidents and therefore protect drivers and riders. However, similar kinds of products are barely seen on bicycles.

Our solution was to create a riding assist for bicycles and thus lower the risk of a bicycle accident at crossing roads. In this project, we plan to build an autobrake system that will force the bicycle to stop at STOP signs. The STOP sign is such an important traffic signal that anyone who learns to drive will be taught about rules relevant to it. However, bicyclists in most countries do not need a license to ride legally and thus may lack such kind of safety awareness. Therefore, this project will help solving this problem in two directions in general: first, reminding the bicyclists about the STOP sign; second, forcing the brake if no action is taken, which means the bicyclists does not take any action to slow down. Finally, the product should be easy to install and have an affordable price.

1.1 Visual Aid

The basic design of our project is presented in a simple drawing below. To make the design capable of all our intended functions, we must use two separate parts individually: a bicycle part that contains most of the system, and a helmet part which will only be used to alert the bicyclists about STOP sign.

We will install a camera module as an ‘eye’ on the bicycle to capture frontier images as well as a hall-effect sensor on the wheel for speed measurement. These two sensors are wired and will be wireless connected to the speaker on the helmet. Once the camera detects a STOP sign, the speaker will generate warning sounds as a reminder. Then, when the STOP sign is closed enough, if still no action has been taken by the bicyclist, in which case the speed of the bicycle is not slowing down, the autobrake device will be triggered to force the bicycle to stop. An electromagnet will be used to clamp the handlebar of the bicycle. There will also be a button whose execution is of the highest priority to turn on/off the autobrake system. This is especially designed to prevent accidents due to false positives, so that if the autobrake is triggered under inappropriate conditions, the bicyclist can turn it off manually and keep riding forward. This button will be installed on the PCB and thus can be easily accessed by the bicyclist.

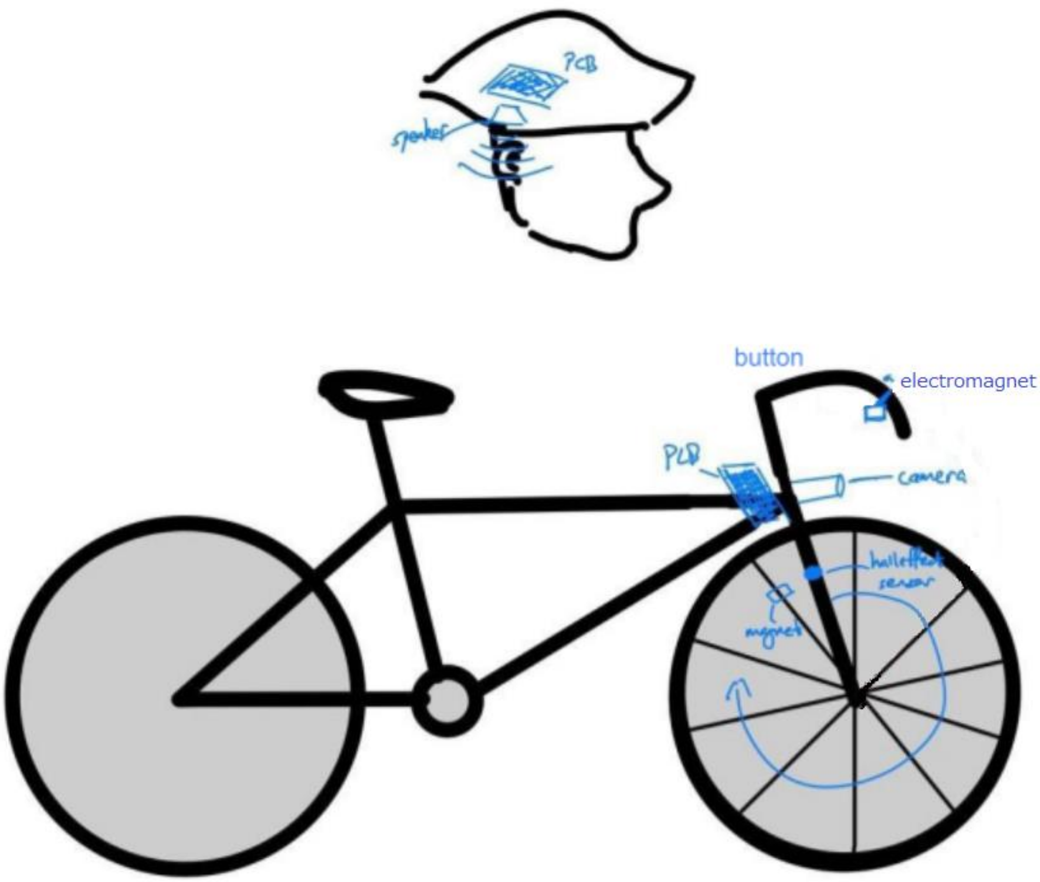

Figure 1 Simple Visual Aid for the design

2 Design

2.1 High Level Requirements

Based on our design intention, the project time scale, and the budget given, we listed three high level requirements as the accomplishment sign of the project:

1. The speaker must ring for three seconds to remind the bicyclist when a STOP sign shows up within 15 meters in the front.

2. While the STOP sign is within 5 meters, the autobrake must be triggered if and only if the speed of the bicycle is above 10km/h.

3. The false-alarm rate must be no more than 15% and the autobrake can be disabled with one-click to avoid accidents due to false-alarms.

2.2 Block Diagram

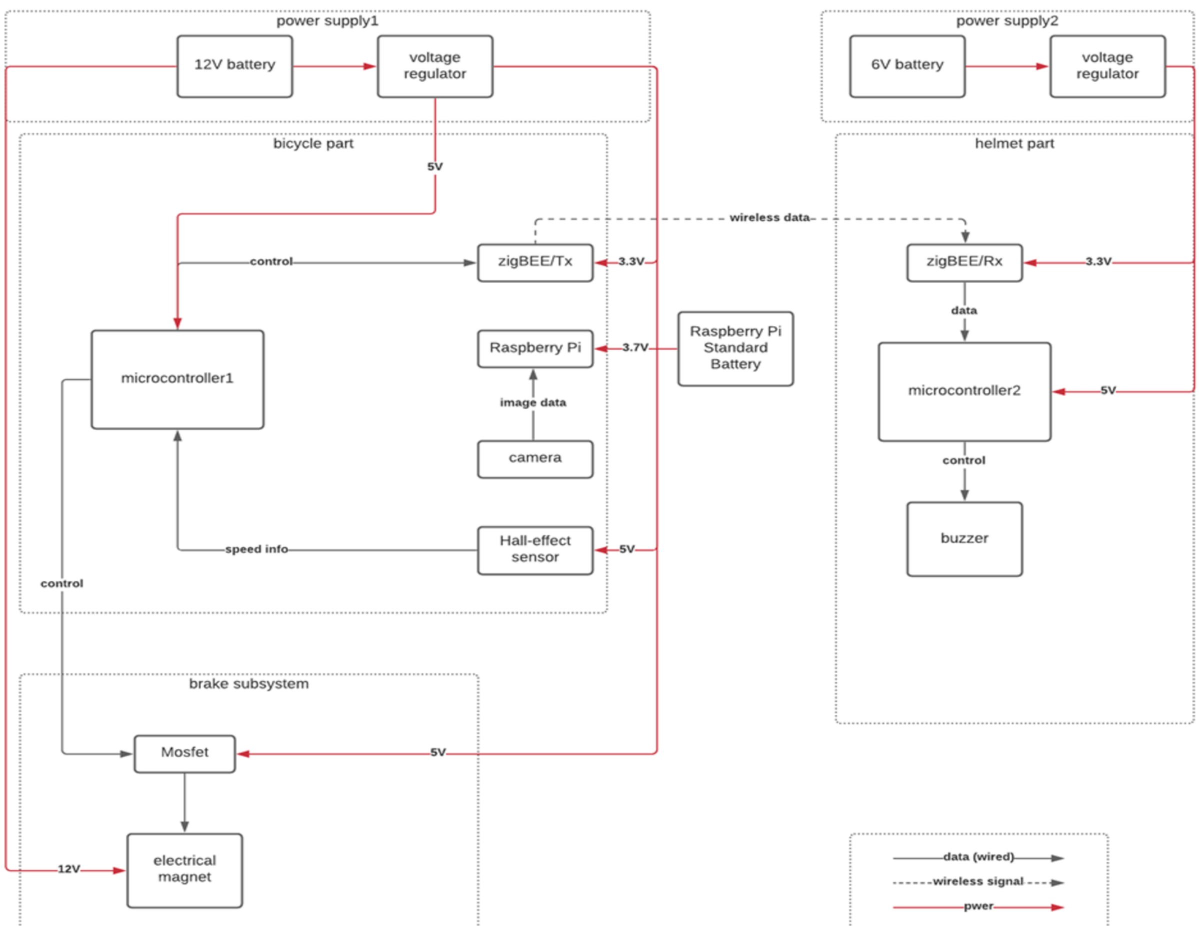

Figure 2 Block Diagram

The autobrake bicycle project is split into two parts: the detection and braking system on the bicycle and the warning system on the helmet. These two parts will be wireless connected with zigBEE module, which can have a transmission distance of 10 meters even with the lowest power supply. We plan to use a common webcam and run the 'template match' algorithm on a Raspberry Pi to detect the STOP sign. We can also determine the distance of detection by adjusting the size of convolution kernels. Once a STOP sign is detected, it will send a signal to the helmet, and the speaker will remind the bicyclist about the STOP sign. We will also measure the speed of the bicycle so that if the speed is higher than some value while the STOP sign is closed enough, which means the bicyclist does not take any action after hearing the reminder, the autobrake will be triggered.

2.3 Subsystems and Modules

2.3.1 Power Supply Module

The regulators LD1117 and LC7805CV will lower a voltage of 12V or 6V from the battery to 3.3V and 5V then supply stable power to other subsystems as shown

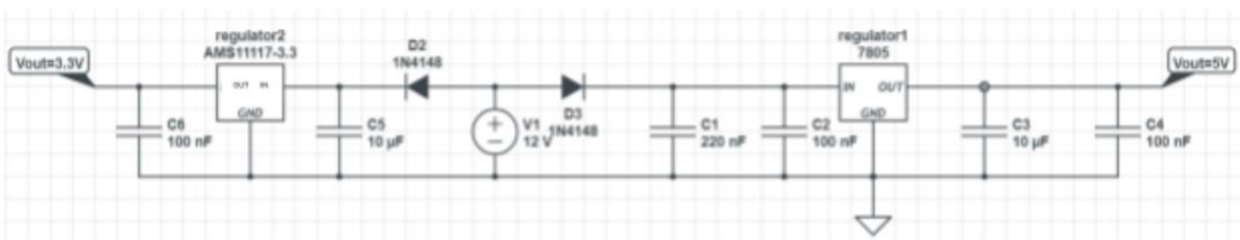

Figure 3 Power Supply Circuit

Since the voltage regulators need to be done on the PCB, we also have a simple PCB design version:

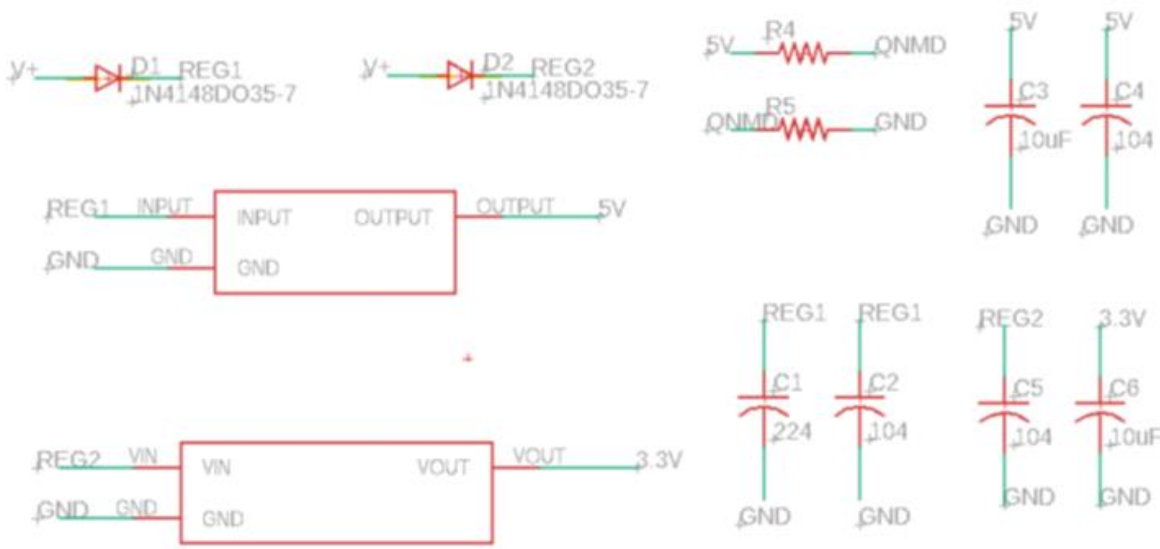

Figure 4 PCB circuit of voltage regulators

2.3.2 Microcontroller and Raspberry Pi Module

The Raspberry Pi, a small single-board computer device, will receive image input from the webcam in front of the bicycle, process it and run the template match algorithm to determine if a STOP sign is captured. It sends control signals to the microcontroller, ATmega328P, for further analysis. Based on our designed functions, the microcontroller will eventually send out control signals to the zigBEE module, electromagnet braking system, and the speaker.

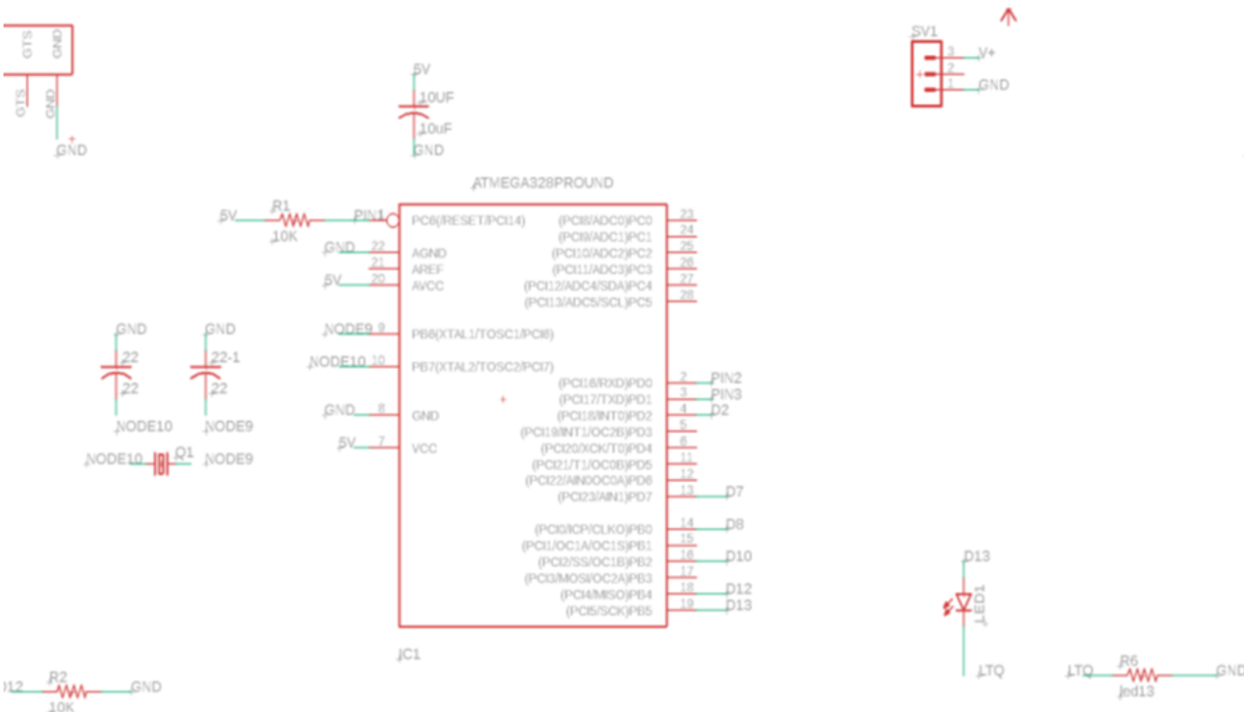

Figure 5 Microcontroller PCB circuit

2.3.3 Electromagnet Auto-Brake Module

Our design uses a combination of electromagnet and metal bar to automatically trigger the brake handles of the bicycle and thus use its original braking to system to stop the bicycle. To begin with, we did can math calculations on how much power we need to apply on the hand bars to successfully stop a riding bicycle and its bicyclist.

To be more specific, the force that stops the bicycle is friction force, which could be calculated from the equation: F = μ * N while μ here stands for the friction coefficient. According to the research articles on internet, the friction coefficient between bicycle brake pads and wheel is approximately 0.38 to 0.41, depending on the environment temperature and heat generated during the process.[3] Now we want to calculate the average energy of a bicycle in motion, using the kinetic energy equation:

$$KE = \frac{1}{2}mV^2 \tag{2.1}$$

Where m here I choose the average weight of a bicycle plus the average weight of a human. The average weight of a bicycle is about 18 pounds, equals approximately 8kg and the weight of human I choose

62kg thus the final m equals to 70kg.[4] The average speed of a bicycle rider is around 14km/h, which is 3.8m/s, so the average kinetic energy of a rider normally riding a bicycle is $1/2*70*3.8^2$ = 505.4Joles, which I take 500J for further calculation. In our design we mentioned a safety stopping distance of 10 meters, which makes the process not so sudden and gives rider enough time to respond. Next, we have the acceleration-distance equation to calculate the acceleration time pairs:

$$S = V_0 t + \frac{1}{2} a t^2 \tag{2.2}$$

We still take 10 meters for S, 3.8m/s for the V_0 and say 5 seconds for the braking time, then we get: 10 = 3.8*5 + 1/2a5*5, solving this equation we get acceleration equals -0.72m/s. Next, we use the acceleration-mass equation and friction force equation together: F = ma = μN. Put in m = 70kg, a = -0.72m/s, μ = 0.4. We get 50.4 = 70*0.72 = 0.4N, which gives N = 126N. The result means we need to apply around 60N of force on each hand bars or 120N on one of them to successfully stops the bicycle. Luckily, the product we found on amazon could provide a force of up to 200N under 5 volts.

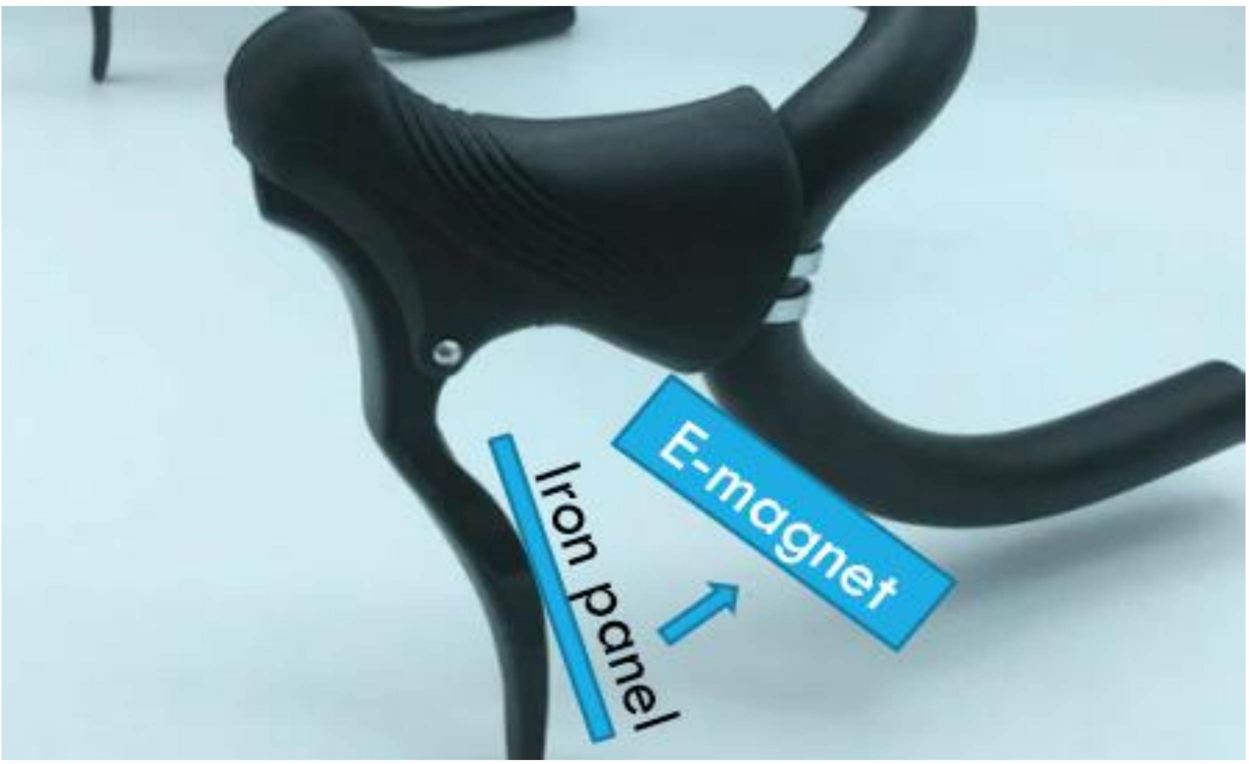

Figure 6 Visual Aid for Electromagnet Module

2.3.4 XBEE Module

XBEE can operate at 2.4GHz at 3.1+/-0.3V. The transmitter can send 250,000 bytes per second to a receiver within 30m in urban areas. We only need to use a pair of XBEEs to build point-to-point wireless connections and send 1-bit messages (1: STOP sign detected, 0: else) between the helmet and the bicycle, which are only 1m+/-50% apart. XBEE can work from -40 to 85 degree Celsius, which means it will not be constrained by extreme climates.

2.3.5 Helmet Module

For this subsystem, the helmet's PCB contains one XBEE device which is used as the signal receiver to get control signal from the bicycle subsystem's XBEE, voltage regulators to power different components, and a buzzer to make out sound that is loud enough to alert the bicyclist. The magnetic buzzer, CEM-1203 will be used to generate warning sounds on the helmet. We select it because it has two long terminals which can be easily soldered to the PCB.

2.3.6 Hall Effect Sensor

Hall effect sensor is as sensor that can detects a change in magnetic field. In our project, it is used for speed calculation. We fix the sensor near the wheel and sticks a magnet to the wheel. When riding the bicycle, the magnet rotates with the wheel and every time it goes past the hall effect sensor, the sensor will detect it. It will the report the interrupt to the Atmega, and Atmega calculates Rounds Per Minute (RPM) of the wheel based on that. Given the RPM and diameter of the wheel, the speed of the bicycle is computed.

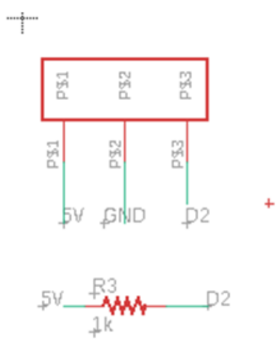

Figure 7 Hall-effect Sensor Module

2.3.7 Final PCB Designs

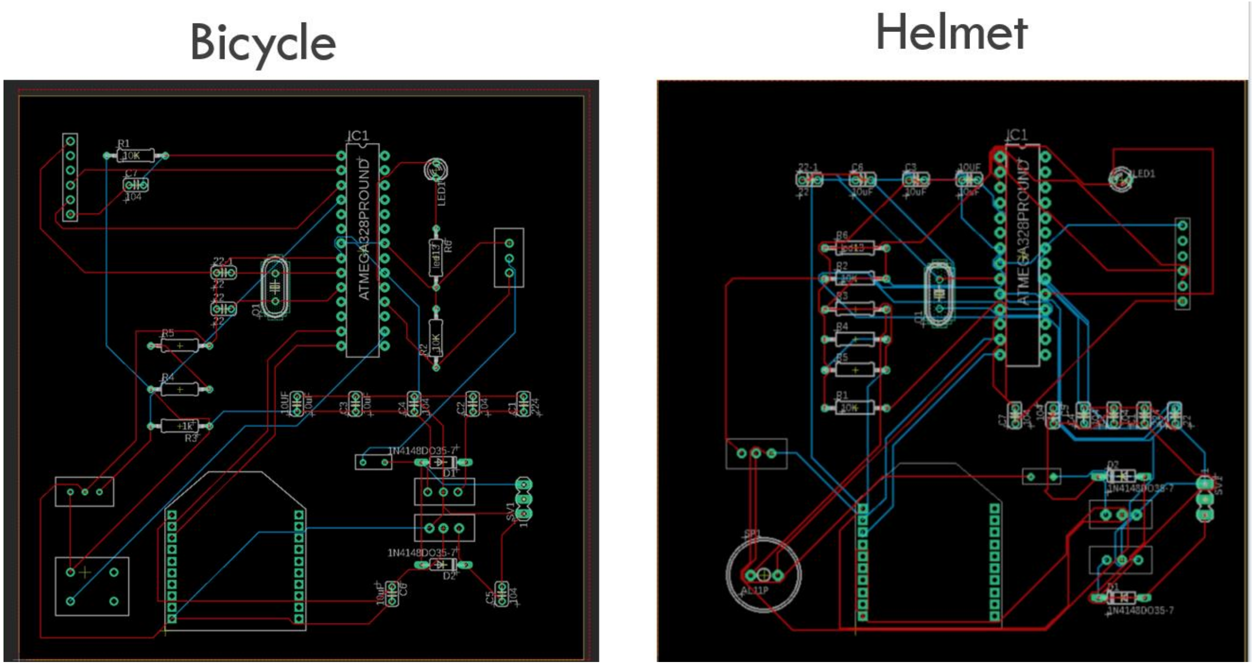

Figure 8 The PCB layouts for two subsystems

2.4 Template Match Algorithm

Template Match is the algorithm that we use for STOP sign detections. This algorithm searches for a position on the raw images that matches the template by using correlation coefficients. The correlation coefficient measures the correlation relationship between, two sets of data, 1 for positive and -1 for negative. In our case of STOP sign detection, we traverse through every pixel on the frame image and calculate corresponding correlation coefficients with the STOP sign template. We then gather all the coefficients to build a correlation matrix, in which the coordinates with the maximum value represents the matched location. Considering that in real life cases, the STOP sign captured by the camera could more or less be tilted depending on the position of the camera and it is not likely to match perfectly with the template, we are not expecting a maximum correlation of 1 in the detection result. We did several experiments on different video clips and found out that a threshold correlation of 0.58 gives the optimal result, which has absolutely 0% false alarm rate and could detect STOP signs of 8 different sizes at various distances within 17 meters.

2.4.1 Selection of Templates

In United States, all the STOP signs have a standard size of 75cm across opposite flats of the octagon. Therefore, at a fixed distance, the STOP sign image captured by the camera should have a fixed size.

$$distance\ to\ STOP\ sign\ (mm) = \frac{size\ of\ STOP\ sign\ (mm) * image\ size\ (pixels) * focal\ length\ (mm)}{size\ of\ STOP\ sign\ on\ frame\ (pixels) * size\ of\ camera\ lens\ (mm)} \qquad (2.3)$$

We can then calculate the size of template we need for detection at different distances.

Figure 9 STOP sign templates

Figure 8 shows the STOP sign templates we used. The smallest template has as size of 16*9 pixels and corresponds to a distance as far as 17 meters.

2.4.2 Pseudo Codes

The following is the pseudo code showing how we use this template match algorithm in STOP sign detection. Notice that we create an independent thread for the camera which keeps updating the frame image. This guarantees that the frame fetched by the main thread, which is used for further image processing, is always updated.

```
Main Thread:

* Create a list containing all template images

* Fetch the latest frame

* Found = False

* For each template in list:

        * Compute correlation_matrix with the lates frame

        * if correlation_martrix.max() > 0.58:

                * Found = False

                * break

* if Found is True:

         * Serial write a signal to Atmega through USB port

* Repeat from step 2

Camera Thread:

* Create a shared variable called latest_frame

* Always stores the captured frame image into latest_frame
```

3. Design Verification

For the design verification part, please refer to Appendix. We break our high-level requirements into small modules and set up the requirements for each one. Appendix A contains the requirements and verifications.

We are able to satisfy most of requirements except for the Electromagent module. In our final demonstration, the electromagnet was not able to make the iron plate attach to it, while it can hold the iron plate once it is already attached. Please refer to Uncertainties section for details. Also, we still cannot do the roadtest, since our raspberry pi cannot generate a high fps for stop sign detection, and we also want to calculation speed to very fast in order to brake the bicycle on time. Once we can fix that, I believe we can do roadtest as soon as possible.

4. Costs

4.1 Parts

Table X Parts Costs

Part	Manufacturer	Part No.	Quantity	Actual Cost ($)
3.3V regulator	sparkfun	LD1117	**2**	**1.95**
5V regulator	sparkfun	LC7805CV	**2**	**0.95**
Microcontroller	adafruit	ATmega328P	**2**	**5.95**
zigBEE	digi-key	XBEE	**2**	**21.95**
Raspberry Pi	amazon		**1**	**56.99**
Hall effect sensor	sparkfun	US1881	**1**	**0.95**
Webcam	logitech		**1**	**29.99**
Electromagnet	amazon		**1**	**13.99**
Buzzer	sparkfun	CEM-1203	**1**	**1.95**
			Total Cost:	**165.47**

Table 1 Cost for parts

5. Conclusion

5.1 Accomplishments

Our project has progressed exceed our expectation in many degrees, both good and bad. In general, the final prototype is capable of all our high-level requirements with only some small flaws and some possible improvement directions. The good part is that we successfully finished the template matching algorithm of the project, and that software part performs beyond our expectation in both reaction speed and accuracy. The final algorithm could analyze one photo within 0.03 to 0.24 second on the computer, within 0.3 second when running on the actual Raspberry Pi. Furthermore, the false-alarm rate of our software is almost 0 in our real test. The unexpected bug is that although the electromagnet could supply force so strong that hold the metal tight enough to stop the bike, the magnet and the metal bar have problem to touch each other from their original place. Still, we proved that with some modifications this system would eventually accomplish our goal. In conclusion, our prototype fulfills all the functions in the lab test and stand still test, but it is not so ready for the real road test that requires someone really riding the bicycle in front of a stop sign.

In addition, we believe the accomplishment is not only physical, but also mental. We all learned a lot in this process of come up with an idea, find a solution, design product, and try to build it in real life. We used PCBs in our designs, which honestly I never got in touch with before, this improves our ability to use new materials together with old knowledge in order to finish certain tasks.

5.2 Uncertainties

The main uncertainties in our project are from three problems in total: the camera, the electromagnet, and the algorithm’s ability to detect tilted STOP signs.

Firstly, during our outdoor test we found that the pictures taken by the camera were sometimes overexposed. The photo sent into the software for analysis is covered in large white spots that greatly influenced the accuracy of the detection. But the problem is not caused by the code bug, it is because we input wrong photos as the resource. This problem is especially serious when the bicycle is facing a shining sun or at special time periods like the sunrise or sunset.

Secondly, as mentioned before, the electromagnet and metal device had some collaborate issues. It seems that the attraction force, in other words, the magnetic field between the electromagnet and the metal decreases drastically when they apart each other. To automatically trigger the bicycle’s brake system, we need to separate them for at least 2.5 centimeters at first. With a distance of that range, there is still attraction force between them, but not so big as they are touching each other, and also not big enough to make them touch each other.

Thirdly, since we use the template matching algorithm for the software, the range of our templates determines the detection range of our final product. In the project we used 8 templates for the accuracy and speed of the algorithm, getting almost 0 false-alarm rate within 0.3s of running time. However, since most of our templates are quite similar, the algorithm is not able to clearly detect a STOP sign if it is tilted in some degree. A same STOP sign upside down is a completely different picture for the algorithm, thus a STOP sign not directly facing the camera will lead to a tilted picture taken and affect the detect accuracy.

5.3 Ethical considerations

To begin with, since our project is completely based on the bicycle itself, the safety issue of the riders always stays at the first place during the entire design process. We strictly followed the IEEE Code of Ethics number 1: *To hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, to protect the privacy of others, and to disclose promptly factors that might endanger the public or the environment*.[6] To give the highest control priority to the bicyclists themselves, we implemented the Enable/Disable button on the PCB and placed it right in front of the bicycle so that the bicyclist can easily access the button and turn off the auto brake system depending on the bicyclist's own consideration of the road traffic conditions.

Furthermore, the first intention of our project is not to force the bicyclists to obey the detection result. We believe that everyone must have entire control of the bicycle when he/she is riding it, thus our project is more like an assistant for helping riders identify stop signs when they do not notice the potential danger. In addition, the project is not auto driving related, it is aimed to lower the risk for bicycle riders instead of letting bicycle riders ride with lower precaution. The user who uses this product may think that they are safe but less aware of the stop sign on the roads. This is not the intention of the project and can cause serious issues to the traffic safety. In short, no matter what kind of riding assistant system we built, the actual behavior of the riders on the road is still the most determinant factor to their own safety.

Finally, the IEEE Code of Ethics number 5 mentioned "*To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, to be honest and realistic in stating claims or estimates based on available data, and to credit properly the contributions of others*."[6] We made accurate and honest claims about all the data collected in our real tests. We honestly accepted the criticism from others and took serious consideration about the errors found by peer reviewers. Besides, we tried our best to minimize the criticism and fixed errors as many as we can. For those unfixed bugs, we gave potential future improvements for all of them.

5.4 Future work

Our future work about the project will mainly focus on how to solve the uncertainties listed above and improve the performance of the prototype as an integrated large system. We already had some possible solutions for those problems.

Since the over exposure problem is caused by the sun and the camera together and we could not make the sun to always shine in our preferred direction, we could only modify and improve the performance of the camera. To begin with, we might try a better camera device, with higher resolution and lower ISO value. The ISO value determines the camera sensor's sensitivity of light, with a lower ISO value, it will be much harder for the photo to experience overexposure.[5] However, a better camera means the pictures sent into algorithm for analysis will be larger and that definitely costs more time for the software to run through process and get a result.

For the electromagnet part, we have two different plans. First is to stick up with the current design and improve the hardware fitness rate. So far the metal we used in the project is still some randomly found iron plate, which is not suitable in thickness, material, and shape. To actually make this design work as expected, we need a metal bar perfectly fits into the gap between two bars and as thick as possible, which requires customized order to have one potentially working sample. Another method is that we could change to other designs using other device, like a servo to pull the braking line or some special device that could automatically clamp together once trigger to replace the electromagnet and metal.

Finally, we already know how to make the algorithm able to detect tilted STOP signs. The answer is quite simple: adding more templates. As we are only using 8 templates right now, the detection range will be much larger if we use 20 templates instead, but that modification will slow down our detection speed greatly. Compared with a wider detection range including different angles of tilted signs, we prefer the detection speed since time is critical in the design of our project, it is closely related to the time left for respond to the bicyclist, thus directly determines the safety of the rider. We could possibly improve the detection speed even faster by using black and white pictures instead of colorful pictures, but that requires us to change the camera used as well.

Références

[1] S. Gustafson, “Why bicyclist deaths are at the highest levels in 30 years” Autoblog, 04-May-2020. [Online]. Available at: https://www.autoblog.com/2020/05/04/cyclist-traffic-death-tracking-outside-magazine/#:~:text=Tracking%20news%20databases%20and%20local,trains%20during%20the%20coronavirus%20 pandemic. [Accessed: 04-Mar-2021].

[2] “U.S. Bicycle Accident Statistics,” PennyGeeks, 31 May 2019. [Online]. Available at: https://pennygeeks.com/legal-resources/statistics/bicycle-accidents/#:~:text=Almost%20467%2C000%20individuals%20were%20hurt,reaching%20nearly%20820%20in%202015. [Accessed: 04-Mar-2021]

[3] A. W. Orlowicz, “Coefficient of Friction of a Brake Disc-brake Pad Friction Couple”, Research Gate, April 2016. Available: https://www.researchgate.net/publication/311939553_Coefficient_of_Friction_of_a_Brake_Disc-Brake_Pad_Friction_Couple.

[4] “What is the average weight of a road bike”, ebikegeneration, December 2017. [Online]. Available: https://ebikegeneration.com/blogs/news/what-is-the-average-weight-of-a-road-bike.

[5] Jimmy Chin, “A beginner’s guide to understanding overexposure and underexposure”, MasterClass Articles, 2021. [Online]. Available: https://www.masterclass.com/articles/basic-photography-101-a-beginners-guide-to-understanding-overexposure-and-underexposure#4-ways-to-fix-an-overexposed-or-underexposed-photo.

[6] ieee.org, “IEEE Code of Ethics”, 2021. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html.

Appendix A Requirement and Verification Table

1. Power Supply Unit:

Requirements:	Verifications:
1. The two regulators must be able to have a stable 3.3V+/-5%, 5V+/-5% output voltage at 12V input.	a). The 3.3V regulators successfully provide power to the XBEEs and do not overheat. b). The 5V regulators successfully provide power to the Atmegas and do not overheat.

2. Detection Unit:

Requirements:	Verifications:
1. Provided that the photo image is of enough quality, the maximum distance of STOP sign detection is at least 15 meters.	a). Run the python program on a pre-recorded code and verify if the first match is far enough.
2. The false alarm rate is below 15% to guarantee accuracy.	a). Label the detection position on the images and verify all matches are true positives.
3. The raspberry pi reads real time photos from the webcam and convert them to grayscales.	a). Plot each image captured and verify if they are real time.
4. The processing time on a clipped image is within 0.5 seconds	a). Use an ipad as the monitor of raspberry pi and prints out the time taken for each image processing.
5. Correctly sends a one-byte signal to the control unit on the bicycle at each template match.	a). Whenever the control unit on the bicycle receives the signal, it sends another one-byte signal to the helmet through XBEEs. b). Verify if the buzzer rings

3. Control Unit:

Requirements:	Verifications:

1. The control unit on the bicycle reads a signal from raspberry pi and sends it to XBEE transmitter. The control unit on the helmet then receives the signal form the XBEE end device and rings the buzzer.	a). If the buzzer rings, it means both the wireless communication between the two XBEEs and the wire communications between each XBEE and its control unit works fine.
2. Reads the speed from hall-effect sensor and activates the brake if the STOP sign is closed and the wheel is spinning at a speed over 10rpm.	a). Run the program on a prerecorded video and. b). Manually spin the wheel to simulate riding on the road. c). At the moment when the brake is triggered, verify that the STOP sign is closed enough. d). Replay the video and do not spin the wheel, this time the brake should not be triggered.
3. The button switch allows the user to disable the entire system with one click.	a). After the brake is triggered, we can lift it with one click on the button. In this case we decide it is a false alarm. (although it is not likely to happen) b). Redo the demo with the system disabled, nothing should happen.

4. Electromagnet Autobrake Module:

Requirements:	Verifications:
1. Stops the bicycle when correct control signal is received from the microcontroller.	a). Spin the wheel until the brake is triggered. b). Continue to spin and feel the friction on it.

5. Buzzer Unit:

Requirements:	Verifications:
1. The buzzer should generate warning sounds to remind the user about the STOP sign	a). Rings it and verify the sound is clear.

Table 2 RV table