Safety Helmet

Alejandro Velasco, Mariela Barragan,
Daniel Ram and Micheal McCoy

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

Abstract—**The Safety helmet will provide safety features for drivers by monitoring driver conditions and environment conditions. The Safety helmet will have ultrasonic sensors to detect vehicles in the driver's dead spot. Infrared camera to detect if a driver is falling asleep and a sensor to detect if a major accident has occurred. The helmet will be activated only when used therefore a button to detect if not in use will be applied. The helmet will have a display to display data such as warnings and temperature readings will be projected to the side view of the driver inside the helmet so it's not obscuring view. The main control and processing will be done with on board MCU and additional intense computation will be done on mobile devices via a mobile application and a Raspberry Pi. The MCU will collect data and transmit the data to the mobile device.**

Keywords—microcontroller, bluetooth, raspberry pi, oled, power supply, safety

I. INTRODUCTION

The reasoning behind our project is to implement important safety features to current motorcycle helmets that will aid the driver while they are traveling on the road. We are aiming to create a low cost, easy to use and accurate safety helmet that will alert the driver when other vehicles on the road are getting too close to them on their blind spots. We also want to notify the driver if they appear to be displaying symptoms showing that they might possibly be going to sleep and lastly want to create a notification system that will quickly notify their emergency contact in their phones and the authorities if a crash occurs. Many helmets currently on the market just offer one of these features and do not tailor to the safety aspect of what we are trying to focus on.

A. Goals/Objectives

The main objective that we are trying to accomplish for our project is driver safety and the ways that we as a student group can apply engineering and technological concepts to a motorcycle helmet to make it safer for the driver wearing it while they are on the road. Ultrasonic sensors, 3-Axis Gyro/Accelerometer sensors, infrared camera, Bluetooth technology and heads up display will all be incorporated onto the helmet to help with our objectives. Table 1 demonstrates what we will be striving to achieve for our project in both hardware, software and as a team.

Research, design and build a state-of-the-art safety helmet in a timely and efficient manner.
Use a minimum number of sensors to detect when objects come into close proximity and alert the driver when it happens.
Be able to turn on or off depending on if the helmet is in use.
Sensors are able to detect when a crash or accident has occurred and notify mobile applications which will notify I.C.E. contact.
Infra-red thermal camera along with eye detection technology to detect when a person is falling asleep at the wheel.
Use a microcontroller along with Raspberry Pi to control the system.
Tie all things above and be able to communicate all the data thru headset in the helmet along with a heads-up display.
Heads up display will not be in the way of the driver.
Adhere to all safety and road standards.

Table 1. Goals and Objectives

If time permits and all goals are successfully completed, we would like to add more objects and features to our project.

B. Specifications

The requirement specifications that we took into consideration are shown in table 2 below. All requirements chosen are to keep the project as user and budget friendly as possible. We are trying to incorporate the most innovative technology that we can get our hands on but also use tried and true technology that has proven to be successful in the past. Some specifications may change as we start our project and dive deeper into our research but for the most part, we should stick to what we have and be able to make our project work.

1	Simple enough for the average person to operate. Set up time should be less than 30 seconds.
2	Communicate with mobile device app via bluetooth 4.0
3	Helmet equipped with a 128x64 pixel OLED display.
4	Two ultrasonic sensors, one located on each side of the helmet to cover dead spots of up to 1.5 meters
5	A 3 Axis Gyroscope + Accelerometer sensor to detect if the rider has been involved in any kind of accident. When accident occurs MCU will notify mobile app and send SMS
6	Eye detection monitoring using a Pi NoIR camera powered by a Raspberry Pi.

Table 2. Engineer Specifications

II. HARDWARE SELECTION

In order to be able to achieve our desired specifications, various hardware components and sensors are needed. In order to detect vehicles in the blindspots, we included a proximity detection sensor into our project. For accident/collision detection, a motion sensor was included. In order for the motorcyclist to be able to see information, such as a warning sign indicating incoming vehicle or temperature, an OLED display was included in the design. For the drowsiness detection, in order to wake up the motorcyclist if drowsiness is detected, we used standard 3.5mm headphones and Raspberry PI. In order to communicate data between the MCU and the user's phone, a wireless transmission module was also included in the design. In the following paragraph, these hardware components and sensors will be discussed in more detail.

A. Proximity Detection

For our proximity detection we researched three different sensors, the HC-SR04, HRLV-Maxsonar- Ez, and the HRLV-Maxsonar- Ez. Table 3 summarizes the different proximity detection sensors that we researched.

	HC-SR04	HRLV-Maxsonar- Ez	HRLV-Maxsonar- Ez
Price	$3.95	$29.95	$13.07
Size	43x 20 x15mm	22.1x19.9x15.5mm	29.5x13.0x13.5mm
Manufacturer	Digi-Key	Digi-Key	Digi-Key
Operating Voltage	5V	2.5V-5.5V	4.5-5.5V
Min/Max Range	400cm	645 cm	10cm - 80cm
Output Type	**Echo**	Analog Voltage	Analog Voltage
Current Consumption	**15mA**	2.5mA-3.1mA	30mA

Table 3: Proximity Sensors

Based on our research [1], we decided to go with the HC-SR04. The HC-SR04 Ultrasonic Module has 4 pins, Ground, VCC, Trig and Echo. The sensor emits high-frequency sound waves from the transmitter (trig pin) towards the desired object and the object picks up the sounds waves. The sound waves bounce off the object and return to the transmitter (echo pin). Since we know how fast soundwaves travel, we use the time it took for the soundwaves to return to the object in order to calculate the distance of the desired object. For example, let's say an object is 10 cm away, we know that the speed of sound is 0.034 cm/µs. If we divide $\frac{10cm}{0.034\ cm/us}$, we obtain 294 us. So the sound wave would have to travel for 294us to reach an object that is 10 cm away. But let's keep in mind that this number is double because the sound wave is traveling towards the object, and then it needs to bound back. In order to obtain the distance in centimeters we multiply our time 294us by 0.034cm/usand we obtain about 10cm.

B. Motion Detection

For our motion detection we researched three different sensors, the Gyro-Sensor SN-ENC03R0, Tilt Sensor-AT407, and MPU-6050 3 Axis Gyro. Table 4 summarizes the different motion detection sensors that we researched.

	Gyro Sensor SN-ENC03R0	Tilt Sensor-AT407	MPU-6050 3 Axis Gyro
Price	$18.80	$1.95	$9.95
Size	22.86 x 22.86 x 5.08 mm	29X5.2mm	21.2x 16.4mm x 3.3mm
Manufacturer	Digi-Key	Dii-Key	Digi-Key
Measurement Range	300 °/s	30 °/s	250 500 1000 2000 °/s
Current Consumption	3.5mA	<6 mA	3.6 mA
Operating Voltage	3V ~ 5.25V	up to 24V	3V- 5V
Axis	Single	None	Three

Table 4: Motion Detection Sensors

For motion detection, we decided to go with the MPU-6050 3 Axis Gyro The MPU-6050 is a triple-axis gyroscope containing a 16 bit AD converter chip with 16 bit of output data. This sensor is very accurate and it has a small footprint of 2 x 1.6 x 0.1 cm. It requires a power supply of 3-5V and the current required for it to operate is 3.6mA. In order to communicate, it uses the I2C communication protocol. Because of its 16-bit ADCs, we can use simultaneous sampling of the accelerometer. This device also requires less need for user calibration because of its enhanced biased and sensitivity temperature.

C. Indicators

One of the ways that we want the outside data to be able to come back to the driver is to use some sort of indicator so the rider can view this data. We have several display technologies to choose from with the most popular options being an OLED display or an LCD display. Table 5 summarizes both of these technologies. The display that we had to choose had to be small and compact in order to fit inside of the helmet but big enough so that the rider could see. We decided that the best option is to go with the SSD13060.96 Inch OLED Module. Compared to the LCD display, this OLED display has a small form factor and a better resolution. The display will show a warning symbol of the ultrasonic sensor and detects a hazard in the blindspot, it will show if Bluetooth is connected, and it will show temperature.

	SSD13060.96 Inch OLED Module	16x2 Character LCD
Voltage	3.3V-5V	5V
Resolution	128 x 64 pixels	2 lines x 16 characters, with character resolution 5 x 8 pixels
Dimension	29.28 x 27.1 mm	80mm x 35mm x 11mm
Interference	I2C	can use I2C with adaptor
Price	$5	$3
Manufacturer	Digi-Key	Digi-Key

Table 5: Display Indicator Comparison

D. Speaker

The Safety Helmet has a drowsiness detection feature. If drowsiness is detected, sound will be played in order to attempt to wake up the motorcyclist. Using standard 3.5mm headphones was sufficient. They were loud enough for the motorcyclist to hear, they are compact,and they are economical. Sound will only be played if the drowsiness is detected, for safety reasons and legality issues, the headphones cannot be used to play music.

E. Wireless Transmission

The Safety Helmet includes hardware in order to transmit data between the MCU and the mobile device. Two popular options that can achieve this is either using a bluetooth module or a wifi module. Table x: summarizes both of these technologies. In the Safety Helmet, we used a bluetooth module. The bluetooth module met the specifications that we wanted.

Bluetooth is a wireless communication standard [2] between two devices over a short distance (usually up to 10 meters) using UHF radio waves. Ultra High Frequency (UHF) radio waves between 2.400 - 2.485 GHz are used to create a personal area network that allows for a transmission of data in both directions.

	HC-05 Bluetooth Module	ESP8266 WiFi Module
Power Voltage	4V - 6V	3V - 6V
Current Consumption	30 mA	80 mA
Range	100m	480m
Support	USART & TTL	Wifi Direct &TCP/IP
Standards	IEEE 802.15	IEEE 802.15
Clock Frequency Support	2.4 Ghz	Up to 80 Mhz

Table 6: Wireless Transmission

F. Battery

We will be using the 12V onboard battery as our primary onboard power supply that motorcycles come equipped with for the Safety helmet. The reason for this is less maintenance for the user as far as making sure that the Safety helmet is always powered and ready to go. Another reason for going this route was for cost savings to the total design. We will also be using 4 AA alkaline batteries for our secondary power supply. The secondary power supply will only be used when the helmet needs to activate the accident detection system and they have disconnected from the 12V onboard power.

G. MCU

We decided to go with the MSP430 microcontroller for several reasons. First, the power consumption is significantly smaller compared to many other microcontrollers and that is due to smaller and more power efficient processors. Second, is our familiarity with the MSP430, since we already have some experience using the MSP430 in embedded systems labs and in our other personal projects it will save us time which otherwise will go towards learning a new microcontroller. Third is the cost, it's true that the cost of all microcontrollers are not expensive but since we already have the development board for the MSP430 it will save us from purchasing a new development board from another microcontroller or even designing a new development board if none are available to purchase. Lastly our familiarity with TI tools such as Code Composer Studio was the final argument needed for us to select the MSP430.

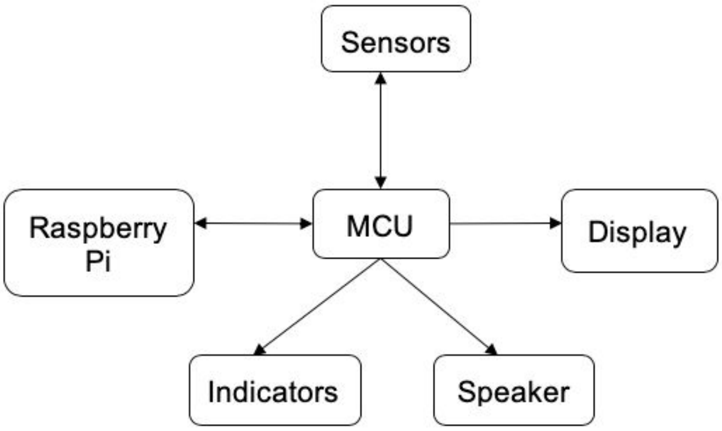


Figure 1: MCU Control System Diagram

H. Raspberry Pi

The Raspberry Pi is an open source single board computer developed by the Raspberry Pi Foundation to promote the teaching of fundamental computer science concepts. The Most recently released board (Raspberry Pi 4) features a broadcom system on chip (SoC), with an integrated ARM-Processor and graphics processing unit. The Raspberry Pi 4 is essentially a small-footprint full-fledge computer that is capable of handling large computational tasks, such as image processing. The Raspberry Pi's Broadcom processor will be able to quickly and efficiently run the software that will help detect user drowsiness.

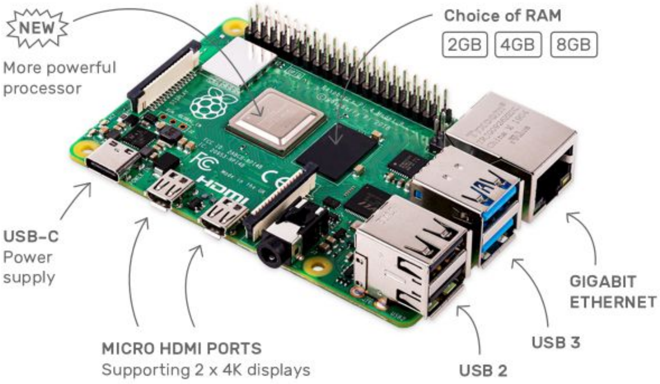


Figure 2: Raspberry Pi [3]

III. POWER SUPPLY

When deciding what power supply we would use for the safety helmet our main goal was to be able to make it as affordable as possible and be able to power the whole system of components. One of the features that we wanted was ease of use for the user and minimum maintenance to the helmet. A lot of discussion was made before coming down with a final solution.

A. Primary Onboard Power

At first we wanted to use a lithium-ion rechargeable battery. One of the things that swayed us from doing this was the need for the power supply to power the whole safety helmet. The Raspberry Pi is a power hungry device that needs up to 3A and 5V to work and the lithium-ion battery would not be able to power it long enough for the average motorcycle ride and the user would have to constantly charge the battery after every ride. The rest of the safety helmet components can all run at full potential with little power. Most of them are 3.3V or 5V with less than 1A current needed to perform. With this information we decided to use the 12V onboard lead acid battery that the motorcycles already come equipped with. Doing this will allow us to make the safety helmet more affordable and one less thing that the user has to worry about as far as maintenance is concerned with the helmet. This helps with powering the whole system including the Raspberry Pi and allowing the user to not have to recharge the safety helmet after each use.

B. Secondary Power

One of the main features of the safety helmet is the accident detection feature. If an accident occurs and the driver is ejected from the motorcycle, after impact the safety helmet will detect what has just happened and notify your emergency contact via your phone app. More of this feature will be discussed in depth later in this report. Once the driver is ejected the safety helmet will become disconnected from the main 12V onboard power supply and will have to resort to a secondary power supply in order to activate the accident detection feature. In order to solve this problem we added a secondary power supply consisting of 4 AA alkaline batteries. The MPU-6050 Axis Gyro only runs off of 3.3V and 3.6 mA which the backup battery can easily supply when needed. A circuit consisting of two 1N4001 diodes was placed in between both power supplies to aid in the activation of the backup power supply when it detected that the main 12V power supply had been disconnected. Figure 3 is a schematic of the circuit.

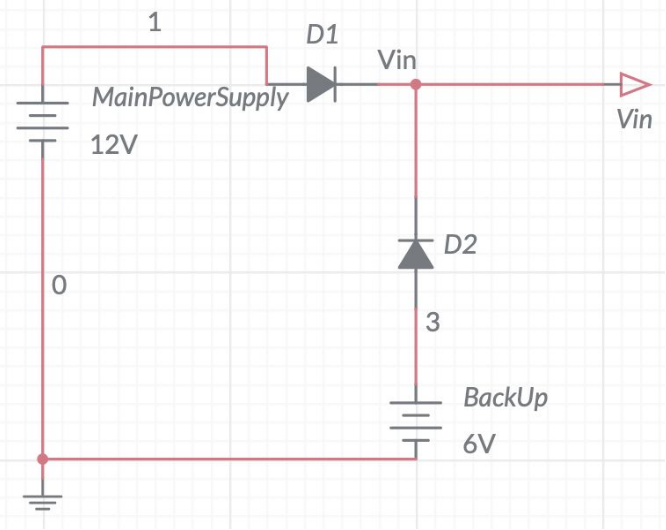


Figure 3: Secondary Power circuit

C. Voltage Regulators

Since we will be using the onboard 12V battery to power the safety helmet we will need to regulate the 12V down to the required 3.3V and 5V that are needed to supply the safety helmet components. We decided to use Diodes Incorporated fixed three-terminal linear regulators for both the 3.3V and 5V outputs and create our own PCB power supply via Kicad schematic/PCB design software. The data sheets guided us on how to properly design the circuit and what components to use so we can take advantage of the maximum ratings. Figure 4 below demonstrates the schematic used along with the components.

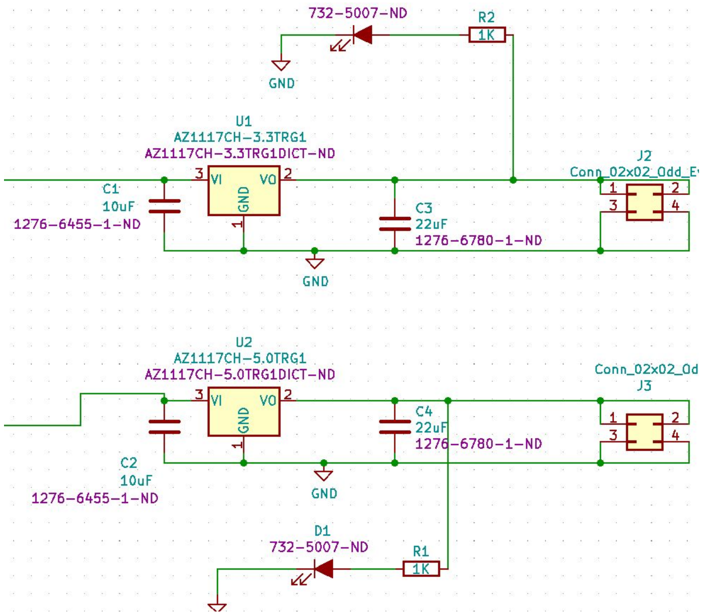


Figure 4:. Power Supply Voltage Regulators

D. Reverse Polarity Protection

To finalize the design of the PCB for our power supply we decided to implement reverse polarity protection to our schematic in order to protect our main PCB and components from accidentally connecting the power supply with incorrect polarity. Texas Instruments offers a great solution with their LM74610-Q1 Smart Diode Controller and pairs it with a N-Channel MOSFET to offer us the best protection. The LM74610-Q1 is designed to drive the MOSFET to emulate an ideal diode rectifier. The supplied data sheet supplies the schematic and analysis that should be used to fully take advantage of the components. Texas Instruments offers these tools and components all for free which is another plus in the construction of the Safety Helmet. Figure 5 demonstrates the schematic in KiCad which we found in the Texas Instrument data sheets.

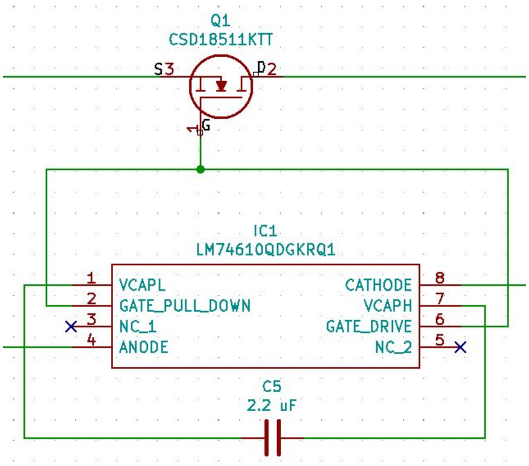


Figure 5: Reverse Polarity Schematic

IV. MICROCONTROLLER

The microcontroller will be the main control of the safety helmet system. All peripherals are connected to the microcontroller and they will be accessed and updated using a timer. The microcontroller has a built-in RC oscillator which can be set up to 16MHz and auxiliary clock which can be used for external clocks. Additionally we added a main switch for the power and a button to be used as a detection mechanism when the helmet is in use.

A. Peripherals and Configurations

Since we are using the main onboard battery which is constantly charged while driving we decided to increase the main clock to 16MHz because power consumption is not an issue and we needed fast computation in order to update the OLED display in a timely manner. Since the OLED display uses I2C protocol which is relatively slow the fast clock will compensate for that. The I2C clock is set to 1MHz which is the maximum the OLED can handle.

Since all our tasks and procedures rely on accurate timing we decided to use a 32KHz crystal oscillator which is known to be more accurate than RC oscillators. We also added two LEDs on our PCB for indication. Red LED is flashing to indicate that the microcontroller is powered on and the green LED is set to on when the helmet is in use which means that someone is wearing the helmet.

B. Software Structure

Once the microcontroller is powered on, it will first initialize MCU settings by turning off the watchdog timer, setting the main clock to 16MHz and enabling I/O pins. After that it will initialize the OLED display, peripherals, timer and timer clock. It then will enter the main loop, inside, there are procedures that are responsible for retrieving and transmitting data to sensors and peripherals. These procedures will be executed in a timely manner which will be determined by the timer and inputs from other peripherals as shown in figure 6.

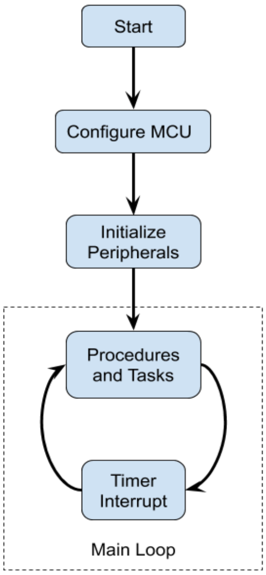


Figure 6: Software Flowchart

With this structure we have more control over the timing of each procedure and task.

V. EYE DETECTION MODULE

Over the last several decades advances in computer vision and processing hardware have shaped the way we interact with the world. Today many commercial devices leverage computer vision in order to better interact with the environment and users. The Safety Helmet system uses computer vision in order to keep the user awake while operating their motor vehicle. The components of this module work together to detect if the user is falling asleep, and attempt to wake the user through an audio alert.

A. Processing Hardware

The computer vision algorithms that detect whether the user is falling asleep are highly compute-intensive, requiring proper hardware to execute all the instructions needed in real-time. Since the Safety Helmet system must be portable, the hardware responsible for the eye detection must have a relatively small footprint. The best candidate that fulfilled all of our requirements was the Raspberry Pi4. This system is a 64-bit ARM SoC (System on Chip), that clocks at 1.5 GHz. The Raspberry Pi4 not only provided the computational power but also several peripheral interfaces and ports that proved useful when developing the audio alert to wake the user.

B. Camera

The system needed a camera that has the ability to overcome low light conditions, in order to feed high quality images to the processor. The camera used for the module was a NoIR camera created by the Raspberry Pi Foundation. The NoIR camera operates just like a regular camera but without the infrared filter included in the lens assembly. The absence of the infrared filter allows the camera to capture high quality images in low lighting environments. The NoIR camera contains 8MP and can operate at a max frame rate of 1080p at 30 fps. A big plus of using this camera with the Raspberry Pi4 was the ease of integration and compatibility, because of the simple CSi interface [4].

C. Face Alignment Algorithm

The face alignment algorithm is the foundation of the drowsiness detection . The problem statement that the algorithm tries to solve is given an image of a face, outline all the main features of the face in the image, such as the mouth, eyes, and nose. The algorithm tries to solve this problem through the use of machine learning. Given many prelabeled images of faces. The first step in the algorithm is to locate the user's face within the image. Once the face is found, the next step is to extract a number of pixels within the face region and apply a regression stage to those pixels. After each stage of regressions the feature points are updated to match the outline of the facial features. The face alignment algorithm used is generated from the research paper "One Millisecond Face Alignment with an Ensemble of Regression Trees" by Vahid Kazemi and Josephine Sullivan [5].

D. Eye Detection Algorithm

The implementation of the face alignment algorithm can be found within the Dlib library, which houses many machine learning and image processing algorithms. The face alignment algorithm returns 68 points that represent the features of the user's face, we can index a subset of these points to only retrieve the eye regions. Once we have the points that outline the eyes, we can then perform an absolute difference calculation from the points that are vertical to the eye as well as horizontal. From here, we can take the aspect ratio, which will be the absolute difference of the horizontal points over the vertical points. As the user close's their eyes we can expect the aspect ratio value to get larger, and vice versa when their eyes get wider. The algorithm will compare the calculated aspect ratio to a set threshold, which will allow the system to know if the user is about to fall asleep. If the user is detected to be falling asleep, a high pitch sound will play in the ears of the user in an attempt to wake them.

VI. SCHEMATICS/PCB

There are many softwares available that can be used in order to design schematics and PCBs. For our project, the schematics as well as the PCB for the Safety Helmet were all designed in Kicad. Kicad is popular software that is used to design schematics and PCBs. It is an open source software and it is user friendly.

A. MCU Schematic

Figure 7 Shows the MCU schematic. The schematic footprint was obtained from the TI website. The MCU schematic shows the different pins that we used to connect our different components and sensors.

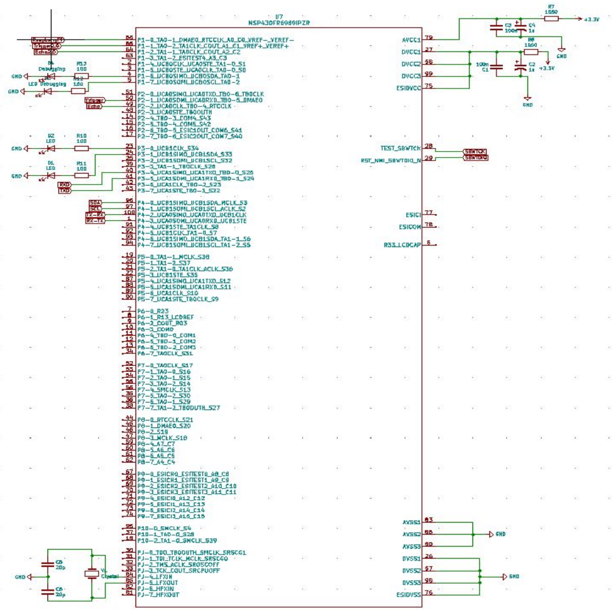


Figure 7: MCU Schematic

B. Peripherals Schematic

Figure 8 shows a block diagram schematic of the different peripherals and sensors that are connected to the MCU. Outlined in red are the different sensors that were connected to the MCU. Outlined in green are the power regulators ,and outline blue are the programming pins.

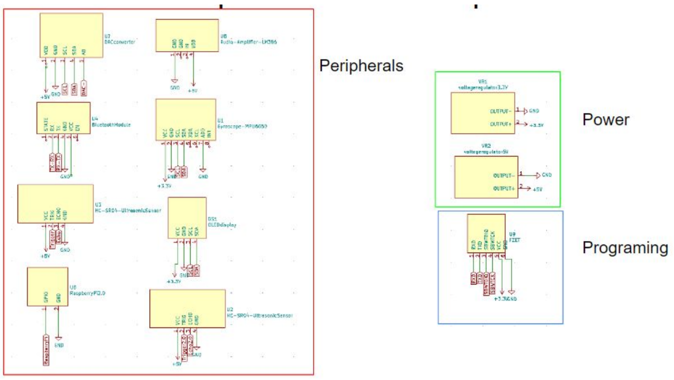


Figure 8: Peripheral Schematic

C. PCB

Figure 9 illustrates the PCB that was designed for the Safety Helmet. It’s a 2 layer PCB. It has a small factor measuring 40.8 x 40.6mm. As shown in the figure, surface mount components were used in order to make the PCB smaller, connectors were soldered into the PCB in order to connect the different modules and sensors. The PCB as well as the different sensors were placed in the back of the helmet. The PCB was ordered from Osh Park.

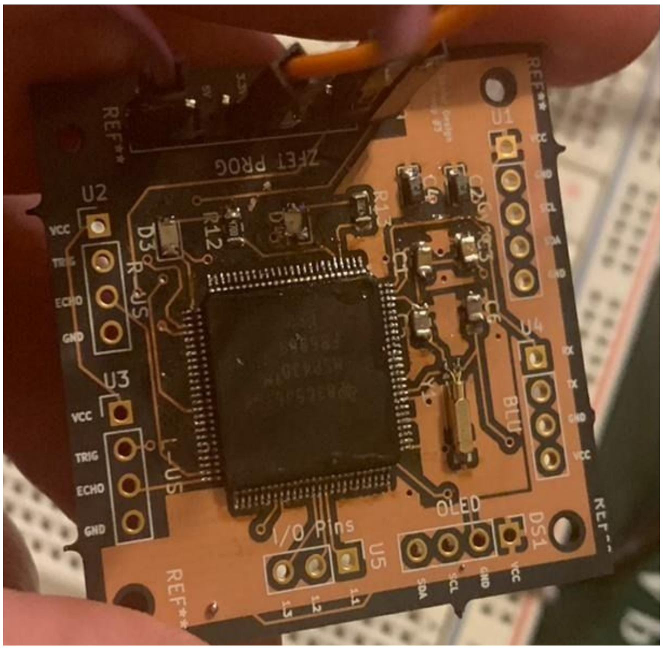


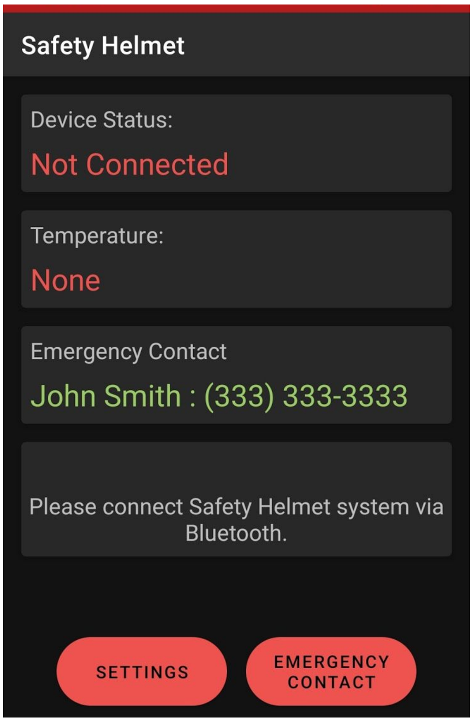
Figure 9: PCB

VII. MOBILE APPLICATION

The mobile application for the Safety Helmet System was developed with Android Studio using Java. The application allows the user to interact with peripherals on board the helmet as well as play an important role in signaling for help if the user were to encounter an accident.

A. Functionality

Users of the mobile application have the ability to modify brightness levels of the OLED display as well as change the units of the temperature values that are streamed from the temperature sensor built into the gyroscope. Before using the application the user must set an emergency contact that they would like to be notified in case of an accident. The user can edit and save new emergency contacts as well as custom emergency messages. All the emergency contact information is saved locally on the user’s device via shared preferences. In the case that the user does encounter an accident, the application will use the SMS (Short Message Service), to forward a text message to the user’s desired emergency contact. The text message will contain a request to receive immediate help.



Safety Helmet

Device Status:
Not Connected

Temperature:
None

Emergency Contact
John Smith : (333) 333-3333

Please connect Safety Helmet system via Bluetooth.

Figure 10. Dashboard User Interface

B. User-Interface

The mobile application user-interface aims to follow the Google Material Design Guidelines. The application consists of three activities: Dashboard, Configuration, and Emergency Contact Activities. The dashboard activity displays the user’s emergency contact’s name and phone number, as well as temperature data streamed from peripherals on board the helmet. Warnings are also displayed to let the user know if an emergency contact has not been set and/or if a bluetooth connection has not been made to the Safety Helmet system. The configuration activity contains the controls that allow the user to connect to the Safety Helmet via bluetooth as well as change temperature units and brightness levels on the OLED display. The emergency contact activity provides a form that allows the user to edit and save contact information.

VIII. CONCLUSION

The purpose of the Safety Helmet was to design a helmet that will make motorcyclists safer. We achieved this by incorporating numerous features such as proximity detection, drowsiness detection, and collision detection. Currently there are similar products available in the market but they are expensive. We wanted the Safety Helmet to be affordable but at the same time reliable.

Designing and building the Safety Helmet has been an overall challenging and rewarding experience.One of the major drawbacks that we encountered was not being able to use the labs at school because of Covid-19 and a delay in getting testing equipment. Designing and testing the Safety Helmet has allowed us to incorporate all the skills that we have learned in studying Electrical and Computer Engineering.

IX. ENGINEER BIOGRAPHY

Alejandro Velasco:

Alejandro is a graduating Electrical Engineer student from the University of Central Florida. After graduation he is pursuing a career in the RF field and possibly open to coming back to UCF to pursue a Masters degree.

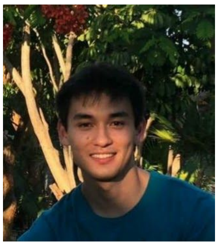
Mariela Barragan:

Mariela is a graduating Electrical Engineering student from the University of Central Florida. She is currently interning at Phasor Engineering LLC. After graduating, she hopes to eventually work in space programs. In the future, she plans to pursue a Masters degree.

Daniel Ram:

Daniel Ram is a graduating Computer Engineer from the University of Central Florida currently looking for a full-time position. His main focus is real time embedded systems and also interested in FPGAs. After graduation he's interested in advancing his knowledge in RT embedded and FPGAs. In the future consider a master's degree.

Michael McCoy:



Michael McCoy is a senior studying computer engineering at the University of Central Florida. He hopes to take on more projects that involve embedded systems. After gaining some more work experience, he plans to pursue a master's degree in computer engineering.

X. REFERENCES

[1] Kinney, Thomas A. "Proximity Sensors Compared: Inductive, Capacitive, Photoelectric, and Ultrasonic." *StackPath*, 1 Sept. 2001, www.machinedesign.com/automation-iiot/sensors/article/21831577/proximity-sensors-compared-inductive-capacitive-photoelectric-and-ultrasonic.

[2] X. Wang *et al.*, "A 0.9–1.2V supplied, 2.4GHz Bluetooth Low Energy 4.0/4.2 and 802.15.4 transceiver SoC optimized for battery life," *ESSCIRC Conference 2016: 42nd European Solid-State Circuits Conference*, Lausanne, 2016, pp. 125-128, doi: 10.1109/ESSCIRC.2016.75982

[3] Foundation, Raspberry Pi. *Buy a Raspberry Pi 4 Model B – Raspberry Pi*, 2012, www.raspberrypi.org/products/raspberry-pi-4-model-b/.

[4] Working Group, Camera. "MIPI Camera Serial Interface 2 (MIPI CSI-2)." *MIPI*, 10 June 2020, www.mipi.org/specifications/csi-2.

[5] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, 2014, pp. 1867-1874, doi: 10.1109/CVPR.2014.241.