Spring 2019

Motorcycle Eyes

Patrick Nie
The University of Akron, pjn13@zips.uakron.edu

Alexander Ray
University of Akron, ajr114@zips.uakron.edu

Motorcycle Eyes

Final Design Report

Design Team 15

Alec Ray

Cogan Warther

Patrick Nie

Scott Zipp

Faculty Advisor

Dr. Jay Adams

4/26/2019

Table of Contents

Table of Figures

Table of Tables

Abstract

As technology continues to advance the safety factor has increased for vehicles on the roads. However, not much has been done to help improve the safety of motorcyclist. To help to solve this problem a wireless blind spot indicator for a motorcycle helmet will be designed. It will be powered off the motorcycle and have indication zones for the left, right, and rear blind spots. The system will alert the rider if a vehicle is within 7 meters of the back of the motorcycle in any of the mentioned blind spots. The radar sensors will detect the vehicle and send a signal to a microcontroller on the motorcycle which will wirelessly communicate to another microcontroller in the helmet via Bluetooth. The microcontroller in the helmet will then indicate to the rider which blind spot in occupied. This technology is already being utilized on cars and now it can hopefully be used to help insure the safety of motorcyclist as well. (CJW)

1. Problem Statement

Need

According to the Insurance Institute for Highway Safety, 4976 motorcyclists died in accidents in 2016. This accounted for about 13% of all motor vehicle deaths that year while only being about 2% of all registered vehicles. Motorcyclists are not always the most visible to other vehicles on the road and need to always be alert and aware of their surroundings. (CJW) There are also many parents that own a motorcycle. According to Statista, as of spring 2017 12.65 million people own a motorcycle in households. Assuming that most of them wear a helmet while riding that would save an estimated 1630 motorcyclist lives a year. (SMZ) Constantly having to look around to be aware of the location of other drivers can be distracting to a motorcyclist and remove his/her attention from the road ahead. Mirrors are not enough to provide a full area of coverage behind the motorcyclist. The loud noise of the motorcycle also

inhibits the method in which the blind spot detection can be clearly relayed to the motorcyclist. (AJR) In order to make this easier on the rider, a system is needed so the rider can easily see his surroundings and become less distracted while riding. (SMZ)

Objective

A motorcycle is a fraction of the size of a car. There is risk of other cars not being able to see a motorcycle as it is traveling. Many vehicles are equipped with blind spot detection and this aids the drivers of the vehicles to be made aware of other vehicles that are traveling around the perimeter of their car. This is a valid solution for a traditional car but what about the motorcycles being able to see other vehicles? With a motorcycle, there is much less surface area on the dash compared to that of a traditional car. With this smaller dashboard, even if a blind spot light or detector was to be installed, it would be difficult to see and differentiate from everything else that is already on the dash of the motorcycle. (PJN) To solve this problem, small indicator lights should be placed in the helmet in order to alert the rider to surrounding vehicles. (CJW)

Research Survey

There is currently not a device similar to motorcycle blind spot indicators in the helmet but there are a few other technologies that are designed to record riders riding style and make measurements to ensure the safety of the rider and the bike. The first is sensors located on the bike in order to determine the tilt of the motorcycle as it is being ridden. This largely comes into play while the motorcyclist is making any large turns or lane changes. The article goes on to say that "In fact, the tilt angle has a major impact in determining the lateral tire–road contact forces, which ensure the stability of a motorcycle on a curve." [1] This design utilized two sensors on the sides of the bike in order to obtain measurements of the distance to ground and calculate the

roll angle of the tires based upon those measurements. The roll angle was then stored and analyzed over a period and used to determine when the motorcycle might need new brakes or tires. This is similar to sensors for the blind spot detection as they also put sensors on the motorcycle. We will most likely be dealing with many of the same issues they did such as engine noise and road inconsistency. (CJW)

Another similar technology is used to measure the suspension of the motorcycle. A motorcycles suspension is different than that of a cars as it can be easily adjusted manually and also responds differently. A motorcycles suspension can need to be changed for a variety of reasons. Either for different riding conditions, carrying a passenger or a load of varying weight, or just for rider preference. To measure the change in suspension they placed five sensors on the motorcycle in order to obtain measurements on each of the shocks and the seat. This was then tested over several different terrain paths in order to see how the shocks performed. [2]

This could also give insight into the sensor on the motorcycle more as it will help us to see how sensors can respond to different road styles. The data could also be analyzed to see which shock level setting could be used for the most comfortable ride under each road condition based upon the rider. These are just two examples of current devices for motorcycles that also can be used to help with the safety of the bike and the rider but also to see how sensors can react to the tough conditions of the road and to help us overcome those issues. (CJW)

The basic theory behind our design is to place sensors on the side of the bike as well as right behind. We will also incorporate a method of wirelessly relaying the information to the driver. The driver will see visuals within his or her helmet if there is someone in any of their 3 blind spots that they need to be aware of. This will be a very subtle notification so that the driver is aware of the surroundings but not completely distracted by a flashy light or symbol that

continues to pop up in his or hers' helmet. This concept will require a substantial amount of testing as we do not want a vehicle to be present in a blind spot and the driver not be notified. We must test with vehicles of all sizes, including motorcycles. The helmet is a major part of all riders' safety as they are riding. There is about 40% of the United States that require the use of a helmet and 54% require some individuals to use a helmet.[3] The article also stated that "Helmets prevent fatalities and can reduce the number and severity of head injuries."[4, 5] Our hope is that this concept of placing sensors and notifying the drivers will further aid the drivers and prevent fatalities for those who wear helmets. The percentage of states that require helmets may even rise if there is seen to be an increased safety factor with the blind spot and speed notification system. (PJN)

Today, blind spot detection exists with cars and trucks. If someone has a vehicle that is equipped with the blind spot detection, the driver of these vehicles is able to see if someone is in his or her blind spot by simply looking in their side mirrors. The side mirrors are equipped with a small symbol that lights up when someone is currently driving in their blind spot. This is extremely helpful for drivers as they do not need to stretch their body/neck in order to see vehicles traveling in their blind spot. According to the article, when driving, the act of changing lanes is extremely critical. The blind spot area of any vehicle is enough to "mask" the view of an entire vehicle. The lack of checking this could result in a terrible car collision [6]. The time they take to turn and look around is time that they are not focused on the vehicle in front of them. Some cars have speedometers that project the speed on the windshield. This solves the problem of the driver having to look down in order to see his or hers' speed. Those are two advances that cars have made in order to better utilize a driver's attention and safety. We would like to try and bring these advances to motorcycles and their drivers. Motorcycle drivers have to

turn and look for other vehicles just as drivers of cars without blind spot detection have to. This, once again, takes time away from the driver being aware of what is in front of him or her. This extra time could be the difference between safety reaching your destination and a fatal crash. Motorcycles do not have the luxury of large side mirrors with visibly noticeable alerts for blind spot detection. (PJN)

Motorcycles currently do not have sufficient means of detecting traffic around them on the road. Unlike cars. Motorcycles only have two mirrors. They do not have the third rear-view mirror that allows drivers to clearly see behind them on the road. The two mirrors that motorcycles do have do not provide a large enough field of view to ensure rider safety [7]. This means that motorcyclists not only have two blind spots on either side like cars do, they also have a blind spot directly in the rear. Current technology that exists for blind spot detection sensors in cars mainly utilizes one of three detection methods. These methods include the use of cameras, radar, or sonar. The most common of these three is radar. “Although radar is the most frequently used sensing device onboard a vehicle to detect other objects in traffic applications it is not effective for low speed situations” [8]. Radar also has other shortcomings such as increased cost, increased space requirements, and increased weight. A system that makes use of a rear-view camera does exist for blind spot detection, but this method displays the video footage on the small dashboard of the motorcycle. This would require the rider to look down while riding to view the vehicle behind them. It also does not monitor the blind spots on either side of the rider [7]. (AJR)

Our design group believes that the most practical, and efficient way to detect vehicles in the motorcycle’s blind spot is with the use of ultrasonic sensors. This method does not currently exist for motorcycles, but is used in cars. The benefits of this method include being cheaper than

radar or cameras and having a wide angle of detection. Several limits do exist for this method however, such as being influenced by wind and only working at lower speeds. The limitation of only working for short distances of up to 6 meters is also mentioned for this method, but the group feels that this distance is more than adequate as it will be important to only detect other vehicles on the road in close proximity and not detect guard rails, trees, or other non-threatening objects in the blind spots [8]. (AJR)

There are currently two patents that are related to our project of a blind spot detection motorcycle helmet. The first patent is patent 13897570, motorcycle and helmet providing advanced drivers assistance. It makes a similar claim as our own, "Motorcycles have increased safety concerns… the two-wheel nature of the motorcycle the availability of standard safety system, is reduced"[9]. The idea has a display system in the helmet to provide warnings, similar to ours. Were this differs is, unlike our idea the sensors would be attached to the bike. We would only attach sensors to the helmet for a more efficient design. Another feature included in this patent is a built in speaker as another way to alert the driver. This patent includes many bells and whistles that overall improve the safety of the driver. However with our design we would simplify it and only need the driver to buy a new helmet rather than a new bike. The other patent related to our project is patent 13/671646, embedding intelligent electronics within a motorcycle helmet. This design's main purpose is to install two cameras inside the helmet, one in the front and one in the back, for a safer way to record your motorcycle ride. It states, relating to helmets mounted to the helmet, "As such, these devices can result in aesthetic problems as well as safety issues for a motorcycle rider wearing the helmet" [10]. While there main purpose is not a blind spot detector they would include it in their design. "When a vehicle is within a blind spot of a rider wearing the helmet, notification can be presented" [10]. This is exactly what we would be

doing, however we would not have the feature to record what is going on around us. Both these patents have an idea similar to our own but include much more features. With our design we would have a simpler design which will still provide the driver with a safer experience. (SMZ)

Marketing Requirements

1. The system should be small and lightweight
2. The system should have an easily identified blind spot indicator for each side
3. The system should be powered off the bike
4. The system should only detect moving cars or other vehicles
5. The system should visually alert a motorcyclist of traffic in their blind spots
6. The system should not be distracting to the motorcyclist
7. The system should be durable to weather
8. The system should operate regardless of the angle of the motorcycle relative to the ground

Objective Tree

Figure 1 shows the objective tree for the Blind Spot Detector Motorcycle Helmet. We broke down our project into two goals. The first objective is to make for an accurate system. This would include detecting only other vehicles and alert if a vehicle enters the blind spot. The other objective is for the helmet to be portable. We want a small, lightweight design that can withstand all weather conditions all powered by a single power source in the helmet.

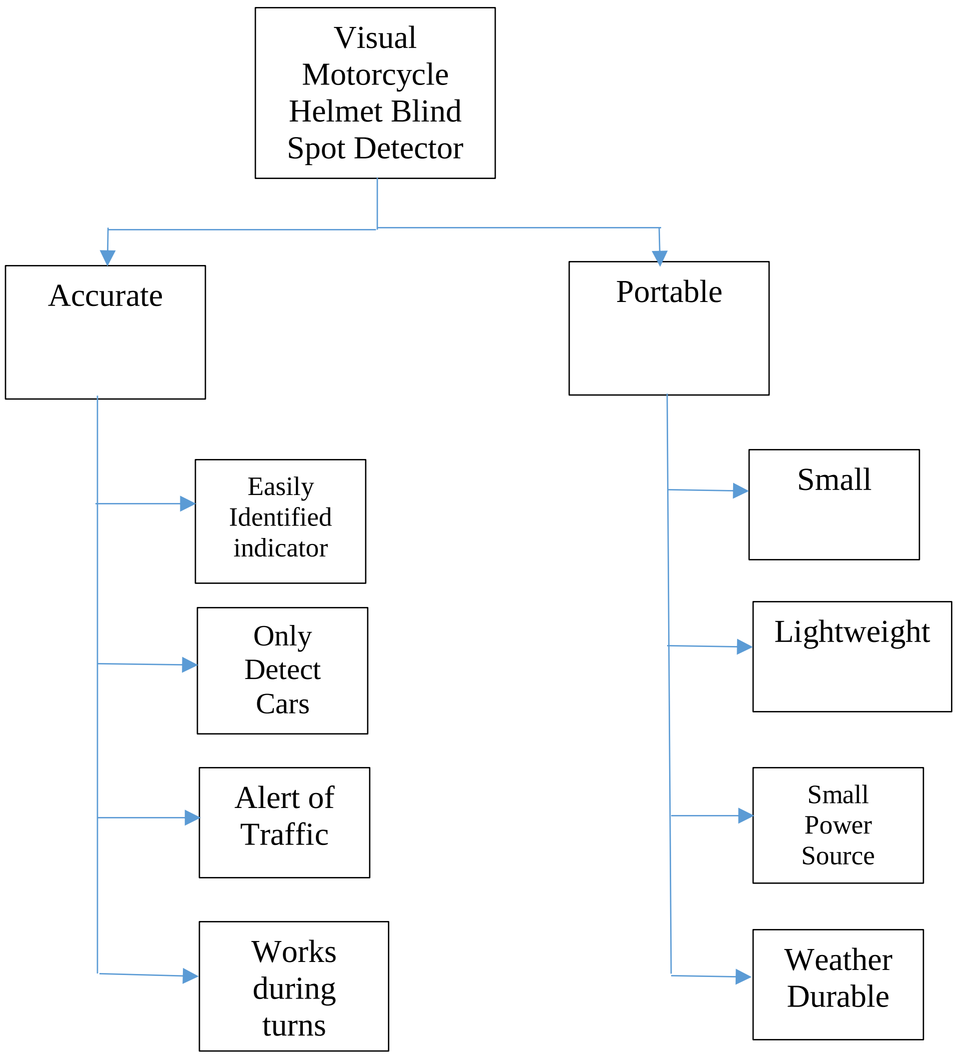


Figure 1. Objective Tree.

(CJW)

2. Design Requirements Specification

Marketing Requirements	Engineering Requirements	Justification
4,8	Must be able to detect cars in the blind spot up to 7 meters behind the bike.	This is a typical requirement of cars with similar technology.
6	Must be able to detect cars in real time without delay.	For safety the device must indicate as soon as the blind spot is occupied.
2,4,5,6	Must provide an accuracy rate of 95% or higher.	The device should give off minimal to no false positives.
4	Must work for speeds ranging from 0-70 mph.	Should be able to work under typical driving speed range.
1,3,7	Must be no larger than a 1' x 1' x 1' box.	Must be portable enough to fit on the back of the motorcycle.
6	Helmet must be wireless and have a range up to 10 feet.	This is to allow the driver to retain free movement for safety.
4,5	Rechargeable helmet battery must have a life of at least 2 hours.	To ensure the system can be useful for the duration of a typical ride length.
5,6	Indicators must not block more than 10% of the driver's field of view	This allows the driver to still be aware of the surroundings in front of them for safety.
1	Unit must weigh less than 5 lbs and helmet unit must weigh less than 2lbs.	To not add to much weight that is bothers the operator.
1,3	Must be powered off the motorcycle battery.	To not add to much weight or take up to much space on the motorcycle.

1. The system should be small and lightweight.
2. The system should have an easily identified blind spot indicator for each side.
3. The system should be powered off the bike.
4. The system should only detect moving cars or other vehicles.
5. The system should alert a motorcyclist of traffic in their blind spot
6. The system should not be distracting to the motorcyclist.
7. The system should be durable to weather.
8. The system should operate regardless of the angle of the motorcycle relative to ground.

(CJW)

3. Accepted Technical Design

Level Zero Block Diagram

Figure 2 is the level zero block diagram in designing our project. We will have three separate sensors for left, right and rear sensing and when powered and run through a system to provide an output sent to the helmet.

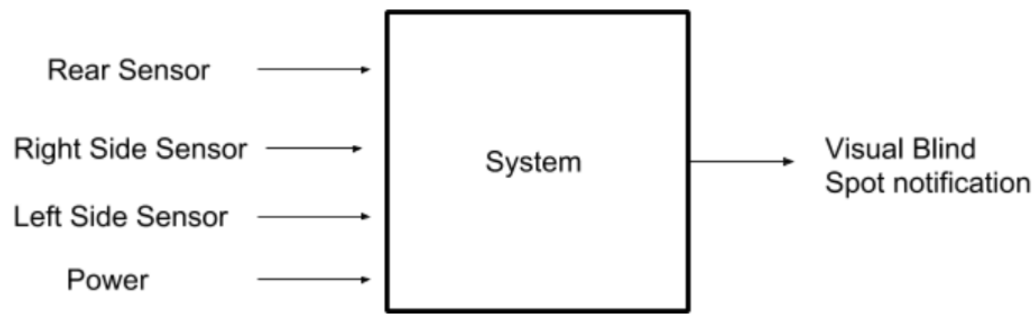


Figure 2. Level Zero Block Diagram.

Level One Hardware Block Diagrams

Figure 3 shows the level one block diagram for the hardware attached from the bike. We are using the motorcycle battery to supply 12V to a DC-DC which will in turn step the voltage down to the necessary 3.3V to power the microprocessor as well as the sensors that the microprocessor is analyzing.

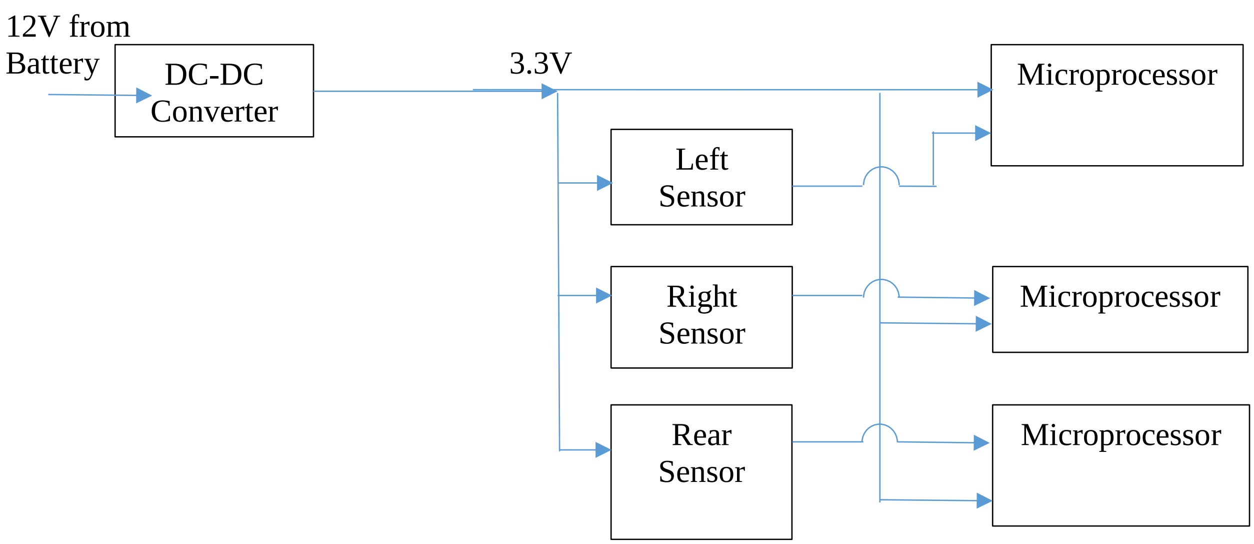


Figure 3.Sensor Block Diagram.

Figure 4 shows the level one block diagram for the hardware attached to the helmet. Here we are using a 12V rechargeable power source to power the circuit. We will have a Bluetooth module receiving a signal from the bike. It will then all go through a microprocessor to send a signal to the indicators alerting the driver of a detection in his bling spot.

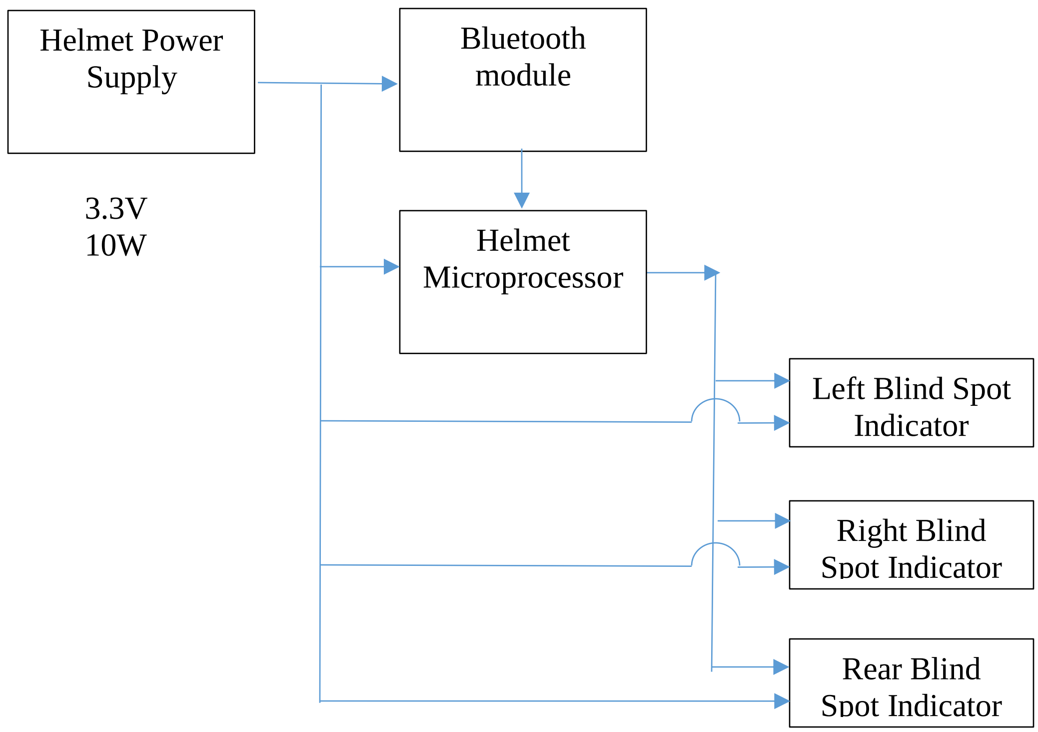


Figure 4. Helmet Block Diagram.

(CJW)

Hardware Functional Requirements

Tables 1-11 shows the hardware fictional requirements for the hardware components for both the motorcycle and helmet.

Table 1. DC-DC Converter.

Module	DC-DC Converter
Designer	Alec Ray
Inputs	12V
Outputs	3.3V
Functionality	To decrease the voltage from the battery of the motorcycle to a useable 3.3V for the controller and sensors.

Table 2. Left Sensor.

Module	Left Sensor
Designer	Cogan Warther
Inputs	3.3V
Outputs	Analog Signal
Functionality	To output a signal to indicate the blind spot is blocked

Table 3. Right Sensor.

Module	Right Sensor
Designer	Cogan Warther
Inputs	3.3V
Outputs	Analog Signal
Functionality	To output a signal to indicate the blind spot is blocked

Table 4. Rear Sensor.

Module	Rear Sensor
Designer	Cogan Warther
Inputs	3.3V
Outputs	Analog Signal
Functionality	To output a signal to indicate the blind spot is blocked

Table 5. Microprocessor.

Module	Microprocessor
Designer	Patrick Nie
Inputs	3.3V, Signals from three sensors
Outputs	Digital Signal to Bluetooth device
Functionality	To interpret the input signals and relay the output signals to the indication devices.

Table 6. Helmet Power Supply.

Module	Helmet Power Supply
Designer	Scott Zip
Inputs	None
Outputs	Power to Bluetooth device and helmet indicators
Functionality	To power the necessary devices for the helmet

Table 7. Helmet Microprocessor.

Module	Helmet Microprocessor
Designer	Patrick Nie
Inputs	3.3V, Digital signals from the Bluetooth module
Outputs	Digital signal to three blind spot indicators
Functionality	To receive the digital signal from the Bluetooth Module and relay that signal to the blind spot indicators.

Table 8. Bluetooth Module.

Module	Bluetooth Module
Designer	Patrick Nie
Inputs	3.3V, Wireless signals from the microprocessor
Outputs	Digital signal to Helmet Microprocessor
Functionality	To receive the wireless signal from the microprocessor and relay that signal to the helmet microprocessor.

Table 9. Left Blind Spot Indicator.

Module	Left Blind Spot Indicator
Designer	Scott Zipp
Inputs	3.3V, Signal from Bluetooth module
Outputs	Indication to rider
Functionality	To receive the signal from the Bluetooth module and indicate to the rider that his/her blind spot is blocked.

Table 10. Right Blind Spot Indicator.

Module	Right Blind Spot Indicator
Designer	Scott Zipp
Inputs	3.3V, Signal from Bluetooth module
Outputs	Indication to rider
Functionality	To receive the signal from the Bluetooth module and indicate to the rider that his/her blind spot is blocked.

Table 11. Rear Blind Spot Indicator.

Module	Rear Blind Spot Indicator
Designer	Scott Zipp
Inputs	3.3V, Signal from Bluetooth module
Outputs	Indication to rider
Functionality	To receive the signal from the Bluetooth module and indicate to the rider that his/her blind spot is blocked.

Level One Software Block Diagrams

Figure 5 shows the level one block diagram for the software of the motorcycle. The microcontroller will constantly look for a received signal. Once the signal is received, it must be analyzed to decipher whether it is a left, back, or right signal. The deciphered signal is then sent to the helmet. (PJN)

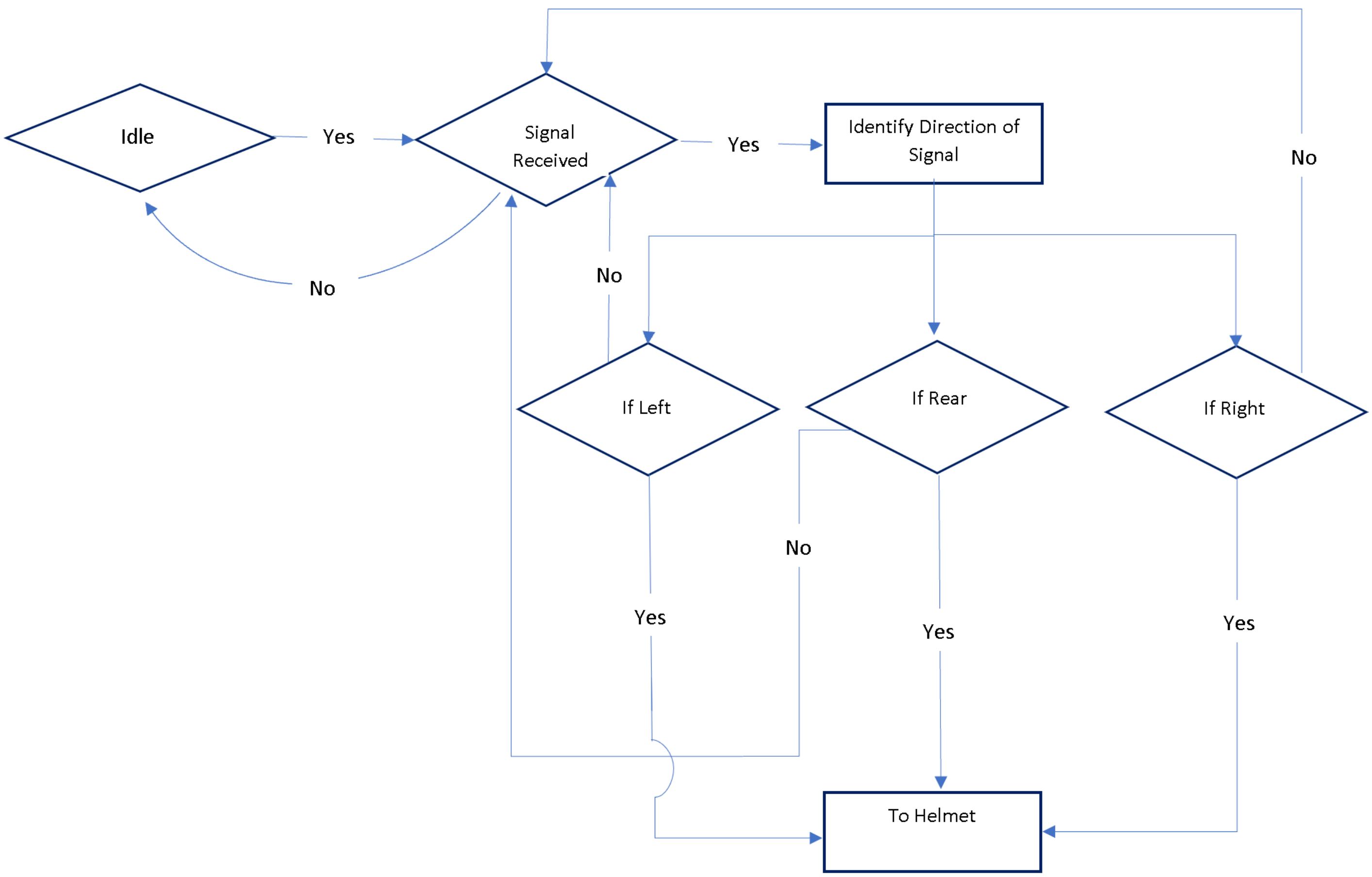


Figure 5. Level One Software Block Diagram for Motorcycle

Figure 6 shows the level one block diagram for the software of the helmet. The signal is received from the motorcycle. Sending the signal to the helmet where a left, back, or right LED is lit based on the orientation of the received signal. (PJN)

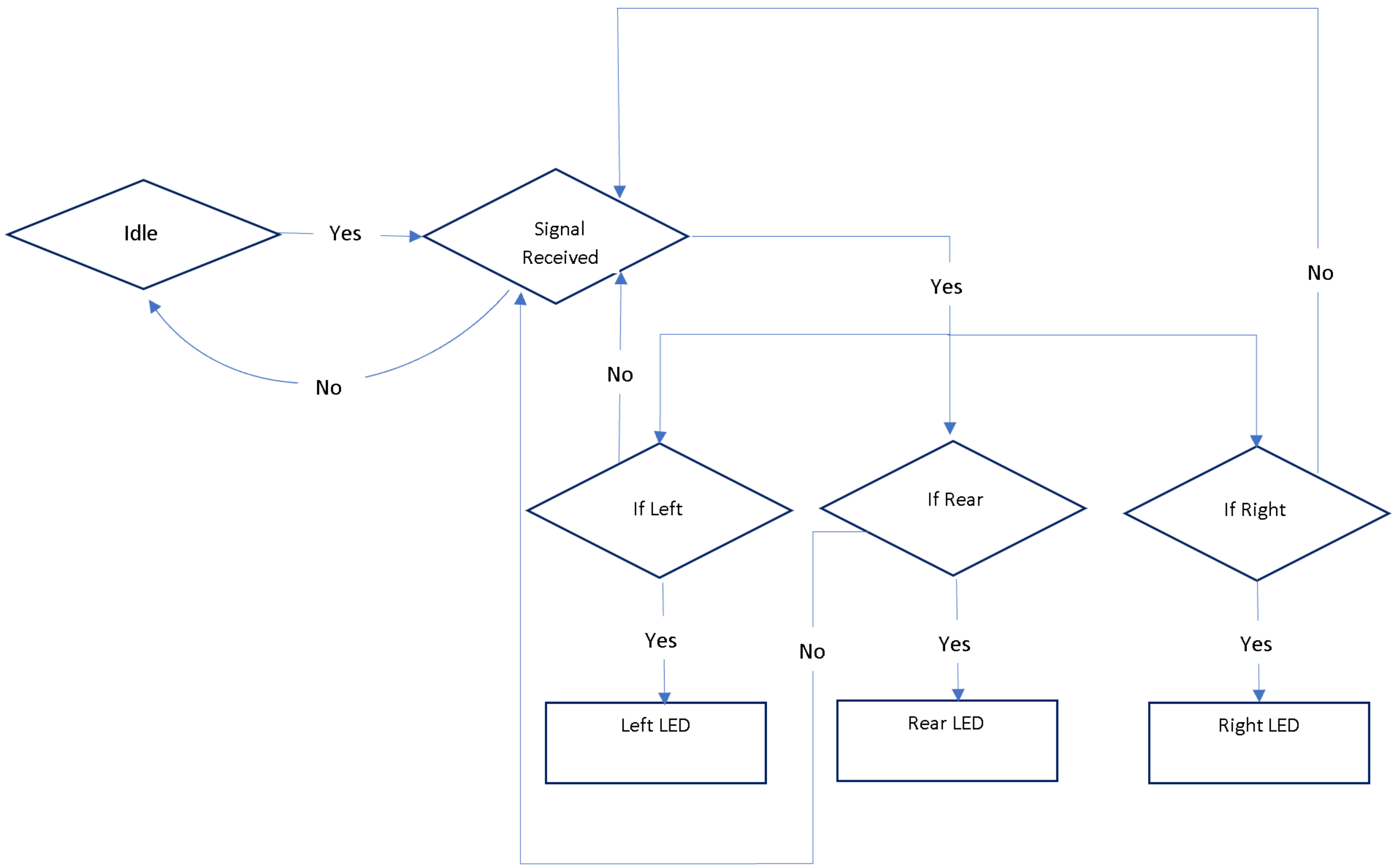


Figure 6. Level One Software Block Diagram Helmet.

Radar Chip Pseudo-Code

Set input and output pins of the radar chip

Enable I2C in slave mode

Configure chirp profiles start frequency, frequency slope, idle time, ADC start time, Ramp end time

Configure chirp configuration RAM to be added to profile with start frequency variable, frequency slope variable, idle time variable, and ADC start time variable

rlDevicePowerOn API to initialize the driver and allocate resources for the host protocol driver

rlDeviceRfStart API to initialize the RF subsystem of the radar device

Configure total TX and RX antennas through rlSetChannelConfig API

Configure number of bits per sample through rlSetAdcOutConfig API

Set up low power ADC mode for power saving through rlSetLowPowerModeConfig API

Set up Clock

Start communication with MCUs on bike

Initialize task to run in infinite loop to search for input data

If Data Present

Then FFT function will calculate the FFT of the time based radar signal\

And send to serial I2C link with corresponding Direction

Return to infinite loop task of searching for data

Else

Return to infinite loop of searching for data

(PJN)

Bike Bluetooth Module Pseudo-Code

Set input and output pins

Enable I2C in Master Mode by passing parameters to module

Initialize Clock with uS period

Define functions that are to be called when no other tasks running in system – Idle Tasks

Define the clock handler function that will service the clock expiration SWI

Initialize channel of communications with the Radar Chip

Initialize task that will run in an endless loop searching for output from the Radar Chip

Define enque function to handle multiple radar chip signals at once through FIFO structure

If output of radar chip

Then Bluetooth module will pass a ‘HI’ to Helmet Bluetooth Module

Return to infinite loop of searching for Radar Chip data

Else

Return to endless loop searching for output of radar chip

(PJN)

Helmet Bluetooth Module Pseudo-Code

Set input and output pins

Enable the I2C as slave mode by passing parameters to module

Initialize the clock with uS period

Define functions that are to be called when no other tasks running in system – Idle Tasks

Define the clock handler function that will service the clock expiration SWI

Initialize the channel of communications with the bike Bluetooth Module

Initialize task that will run in an endless loop searching for output from 1 of the bike Bluetooth modules

Define enque function to handle multiple bike bluetooth chip signals at once through FIFO structure

If receive data

Then set pin corresponding to directional LED to high

Wait set amount of time while LED is high

Return to infinite loop of searching for output from one of the Bluetooth modules

Else

Return to endless loop of searching for output of bike Bluetooth

(PJN)

Software Functional Requirements

Table 12. Sensor Received Modules.

Module	Sensor Received
Designer	Patrick Nie
Inputs	Sensor Signal
Outputs	None
Functionality	Waits for a signal to be received from the sensors. Once the signal is received, the overall process of deciphering and transmitting begins.

Table 13. Signal Orientation Modules.

Module	Signal Orientation
Designer	Patrick Nie
Inputs	Sensor Signal
Outputs	Directional Signal
Functionality	The orientation of the signal will be analyzed. Once the orientation of the signal is analyzed, the location of the sensor signal is known.

Table 14. Signal Send.

Module	Signal Send
Designer	Patrick Nie
Inputs	Directional Signal
Outputs	Bluetooth Directional Signal
Functionality	The directional signal is sent to the helmet to be analyzed

Table 15. Helmet Receive.

Module	Helmet Receive
Designer	Patrick Nie
Inputs	Bluetooth Directional Signal

<u>Outputs</u>	None
<u>Functionality</u>	Once the helmet receives the signal, the helmet process begins.

Table 16. Helmet LED

<u>Module</u>	Helmet LED
<u>Designer</u>	Patrick Nie
<u>Inputs</u>	Bluetooth Directional Signal
<u>Outputs</u>	LED Signal
<u>Functionality</u>	The directional signal triggers a left, rear, or right LED.

System Design

The overall wiring harness for the system on the motorcycle will be constructed as seen in Figure 7.

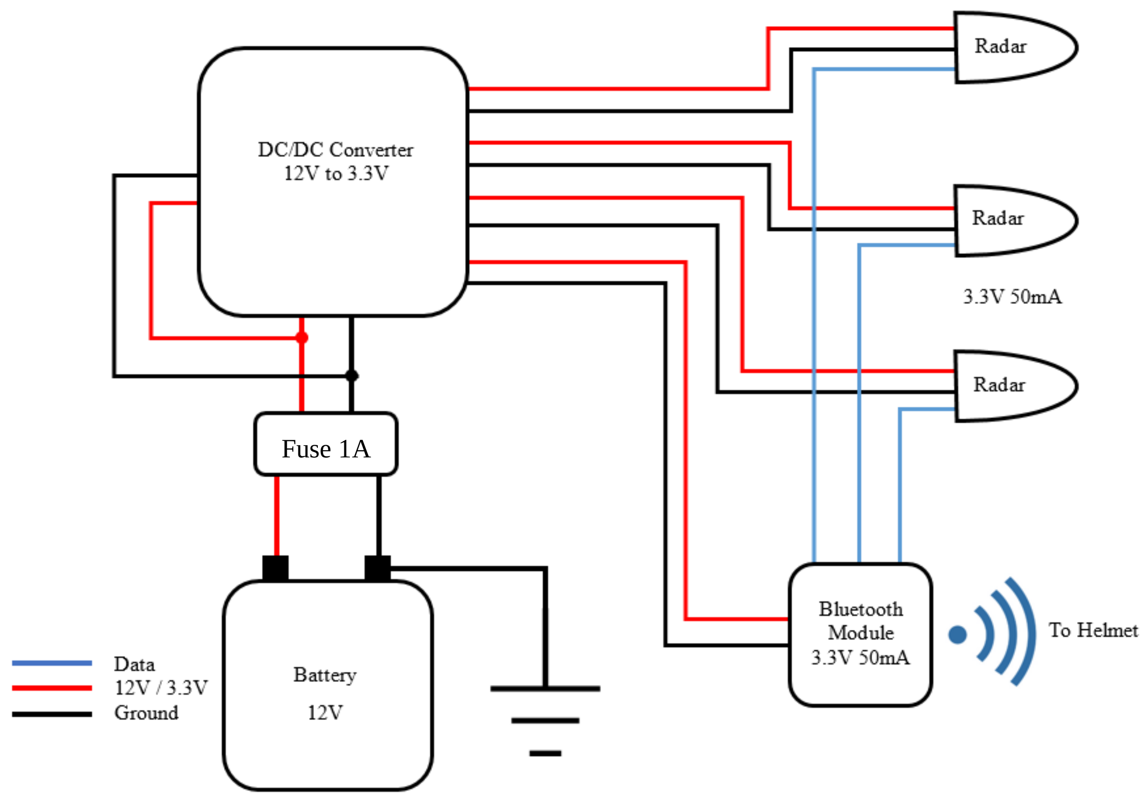


Figure 7: Motorcycle Wiring Harness.

Figure 7 shows how the system is connected and how it will operate. The motorcycle battery voltage will be reduced to a useable 3.3 volts through the use of a DC to DC converter. The necessary power was calculated using the following equation:

$$P_{system} = \left(3V_{radar} * 50mA_{radar} * 3_{number\ of\ radar\ sensors}\right) + (3V_{Bluetooth} * 50mA_{Bluetooth})$$

$$P_{system} = 600mW$$

DC to DC converters are circuits that are designed to convert one DC voltage to another of a differing value. This is done by creating a switching circuit. The voltage input is passed through a switch, operating at a specified frequency to produce a different average output voltage. An inductor and capacitor are used to provide a constant current and voltage to the load. This system required that the voltage be stepped down, this is called a Buck converter. A voltage regulator can be used to achieve the same effect. However, DC to DC converters are much more energy efficient. The design for the Buck converter can be seen in Figure 8 below.

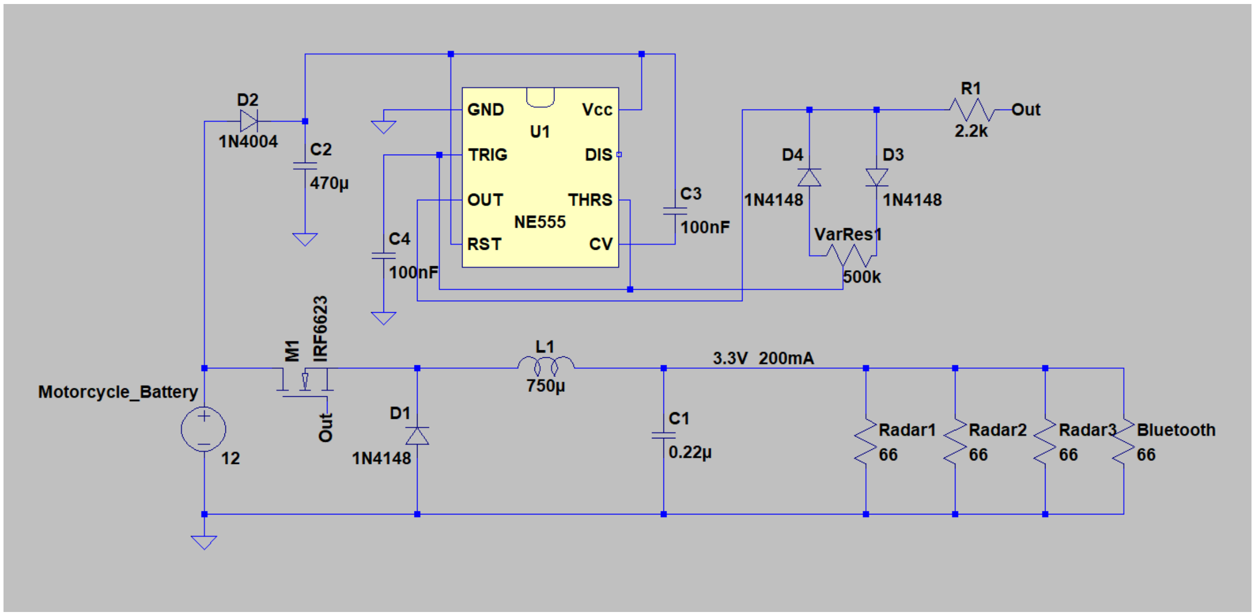


Figure 8: DC to DC Buck Converter Schematic.

A single Buck converter will suffice to power each component. Design considerations were taken to account for the maximum required current and voltage of each component. The radar chips will require 3.3 V at 50mA. The Bluetooth module contains its own DC to DC converter, but it can only operate below 5V so a converter was still required to lower the battery's 12V. Therefore, it was assumed that the Bluetooth requirements were the same as the radar modules to make calculations easier. The converter was thus designed to supply 3.3V and a total of 200mA to be divided equally among the components.

In order to step down the voltage from 12V to 3.3V and supply 200mA, the following calculations were performed. First, it was necessary to find the duty cycle required to produce an average output of 3.3V given a supply of 12V. This is done with the following equation:

$$D = \frac{V_O}{V_{in}}$$

Where D is the duty cycle. The Duty cycle was therefore calculated to be 0.275. In other words, the switch will be on only 27.5% of the time. Next, it was necessary to find the minimum required value of the inductor. A large enough inductor is required to ensure a constant current with little ripple that operates in the continuous current operating range. The minimum inductance was found to be 75uH with the following equation:

$$L_{min} = \frac{(1-D)R}{2f}$$

Where f is the chosen switch operation frequency of 100kHz. R is the total load resistance found to be 16.5 Ohms with the equation below.

$$\frac{1}{R_{total}} = \frac{1}{R_{radar1}} + \frac{1}{R_{radar2}} + \frac{1}{R_{radar3}} + \frac{1}{R_{bluetooth}}$$

An inductance of 750uH was chosen to reduce ripple and ensure that the circuit was operating in continuous current mode. There parameters are found by performing the following calculations.

$$D_1 \geq 1 - D$$

D_1 must be larger than 0.725 to be in continuous current.

$$D_1 = \frac{-D + \sqrt{D^2 + 8L/RT}}{2}$$

Where T is 1/f. An inductance of 750uH provides a D_1 value of 2.88. The current ripple is found as below

$$\Delta i_L = \frac{(V_{in} - V_O)}{L}(D)(T)$$

The current ripple was found to be 32mA giving a maximum current value of 216mA and a minimum value of 184mA. The large inductance value produced this small ripple. Lastly, the capacitor value was found using the equation below

$$C = \frac{(1 - D)}{8L(\frac{\Delta V_O}{V_O})f^2}$$

$\frac{\Delta V_O}{V_O}$ is the voltage ripple in percentage. The circuit was designed to produce a very small voltage ripple of only 5%. A capacitor value of 0.22uF was chosen since it is a common capacitor value and it provides a voltage ripple of 5.5%.

The last part of the Buck converter that needed to be designed was the actual switching circuit. This was accomplished by using a mosfet as a voltage controlled switch and a precision timer to produce a pulse wave. This timer chip provides a square pulse wave of the above specified duty ratio to control the mosfet. The precision timer chip was chosen to be a TI NE555 device. This was chosen to stay compatible with the other TI components in this system. The NE555 operates off of 12V which will be supplied by the motorcycle battery. The switching portion of the device utilizes a variable resistor in order to properly set the duty ratio. The schematic of this chip is shown below.

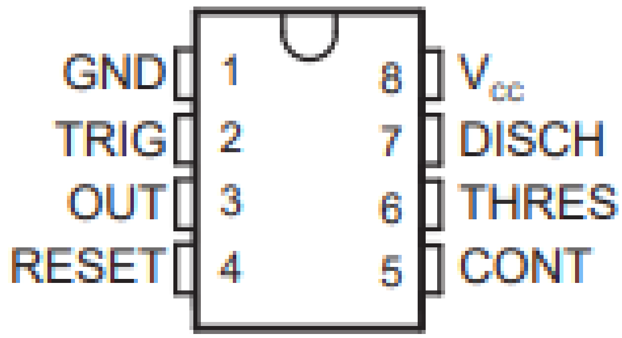


Figure 9. TI NE555 Timer.

The inputs for each pin are specified in the above schematic. Pin 7 will be unused.

Analysis into electrical noise interference led to several design considerations. The largest sources of noise will come from electrical feedback of the motorcycle electrical system (since the radar system is using the same power supply) and also from electromagnetic feedback from the revving the motorcycle engine. Thus mitigation was taken to ensure that as much noise as possible was removed from these sources. All of the components will be housed in the same enclosure, as seen in Figure 10, to reduce the wiring distance.

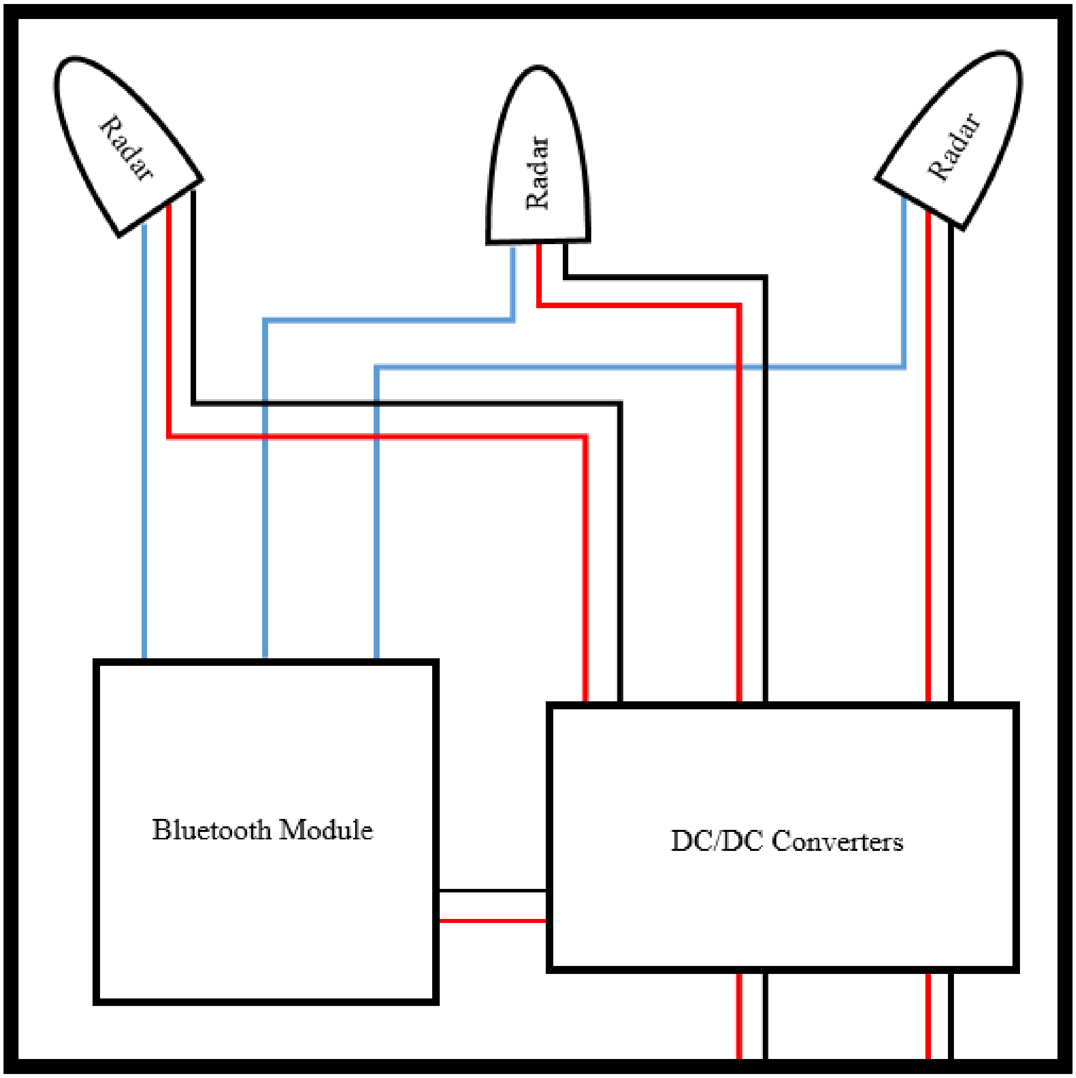


Figure 10. Motorcycle Component Enclosure.

This creates less opportunity for noise to disrupt the signals. It was also determined that the noise filter should be placed before the radar sensors and not after. Noise would most greatly affect the power lines leading to the sensors. Noise is much less of a concern for the lines leaving the sensors and going to the Bluetooth module since they are used for transmitting digital data signals. The enclosure of the components was chosen to reduce the effect of electromagnetic interference as well as ensure that the components were in a waterproof environment and able to

perform outside in inclement weather. Lastly, housing all components in the same enclosure minimized the space occupied on the motorcycle as well as the overall weight of the system.

The final mitigation technique used to reduce noise was to ensure proper grounding of the system. A single ground source should be used to make sure that there are not differences in ground potentials. These differences would allow for current to enter back into the system and alter the signals and transmission lines. When this occurs, a ground loop exists within the system. A ground loop is "a potentially detrimental loop formed when two or more points in an electrical system normally at ground potential are connected by a conducting path such that either or both points are not at the same ground potential." Therefore, to eliminate this potential issue, all ground points in the system are tied back to the battery ground.

The component enclosure will be placed on the back fender of the motorcycle as seen in Figure 11.

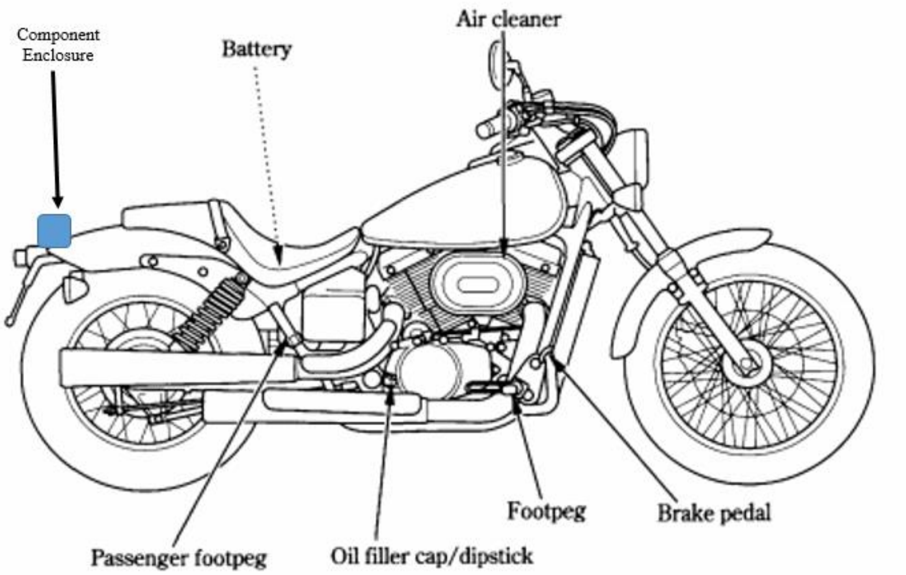


Figure 11. Component Enclosure Location.

This is the optimal location to ensure that the radar sensors have an unobstructed view of the rear of the motorcycle. It also ensures that there is as little interference as possible between the Bluetooth module in the enclosure and the Bluetooth module in the helmet to provide clear communication. [AJR]

The best way to go for the sensors is to use radar. Radar is not affected by the weather, such as snow, wind, rain, mud, and fog, as much as other sensors are. Radar can also work through a bumper on a car, so it is more easily concealed than other sensors. There is also the issue of the sensors needing to be able to work at night. Radar sensors require less lighting and are capable of still being accurate in the dark. For blind spot detectors, most cars typically use radar detectors that have a range of 20m and operate at a frequency of 24GHz [11]. A generic layout of the detection range and alerts for cars is shown below in Figure 12.

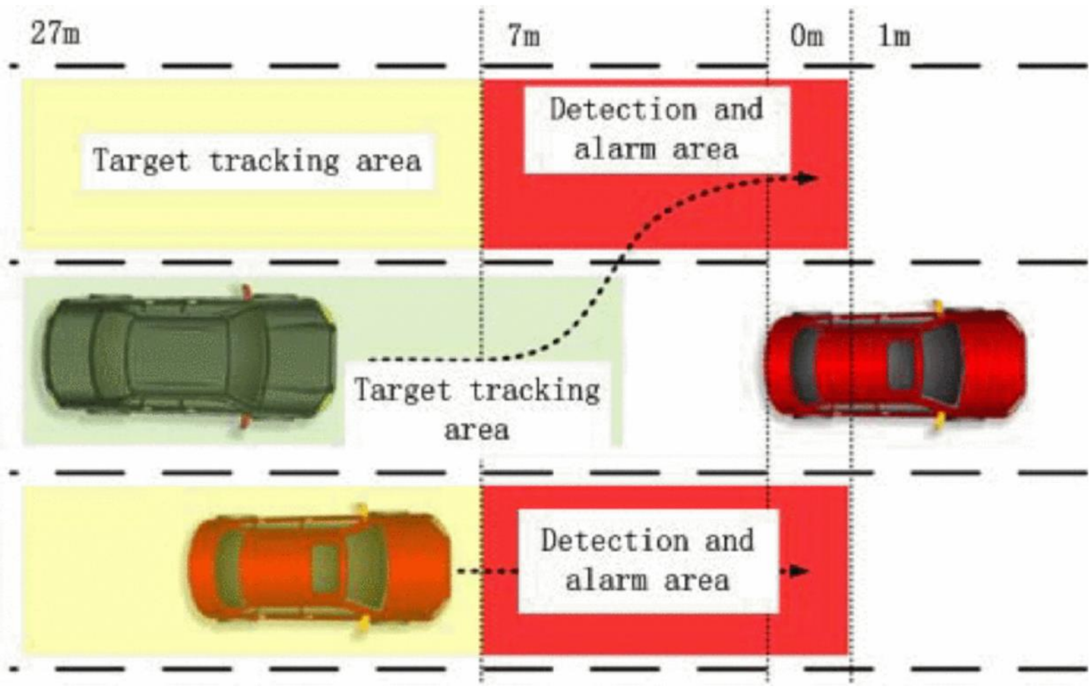


Figure 12. Blind Spot Detection Locations [12].

Minimum Sampling Rate

$$\frac{21ft}{5280ft/mile} = 3.98x10^{-3} \text{ miles}$$

$$\frac{3.98x10^{-3} \text{ miles}}{30\text{mph}} = 1.33\text{x}10^{-4} \text{ hours}$$

$$(\frac{1.33x10^{-4}\ hours}{1})(\frac{3600sec}{1hour}) = 0.48\text{sec}$$

$$\text{Fm} = \frac{1}{T} = \frac{1}{0.48} = 2.08\text{Hz}$$

$$\text{Fs} = 2\text{Fm} = 4.16\text{Hz}$$

To be able to detect a vehicle in the blind spot there needs to be at least a minimum sampling rate to do so. Assuming a vehicle will not pass at a speed greater than the motorcycles by more than 30mph, a minimum sampling rate can be calculated to detect the vehicle in the 21-foot-deep blind spot. Once calculated the frequency is doubled to give a more accurate and reconstruct able minimum frequency. [CJW]

Figure 13 is a layout of the target indication area for the motorcycle. This is similar to the blind spot detection of the car but for a motorcycle there will also be a rear blind spot indicator. The blind spot indication zones will be 21 ft. deep and 12 ft. wide as 12 ft. is the typical width of a driving lane.

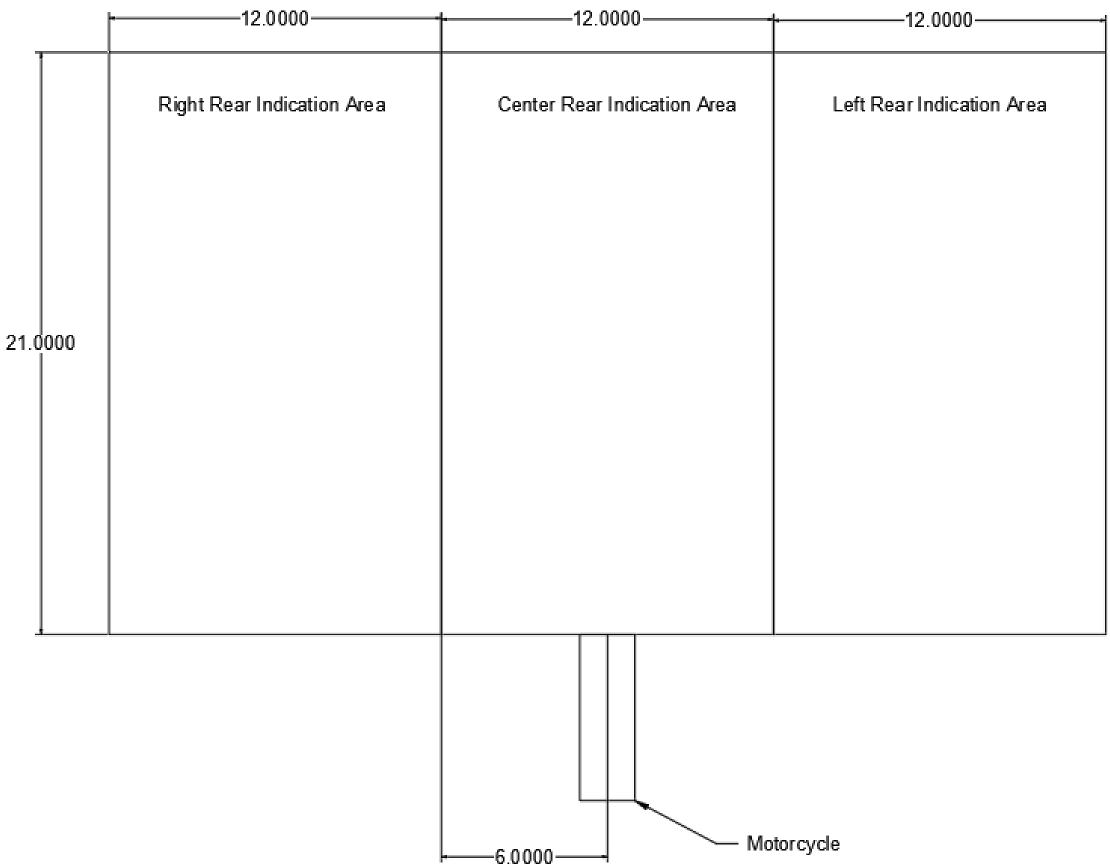


Figure 13: Blind Spot Indication Zones.

From the size of the indication zones, the maximum angle of indication the sensor must pick up as well as the maximum distance can be calculated. The calculation to determine those values for the left and right rear blind spots are shown below.

$$\theta_{required} = tan^{-1}\left(\frac{21ft}{6ft}\right) = 74\,degrees$$

$$Max\ Distance = \sqrt{21^2 + 18^2} = 27.66\,ft$$

In order to make sure the sensor is centered to point down the center of each of the indication areas, the sensor will have to be slightly rotated in its positioning on the motorcycle. This correct angle is shown in the figure below.

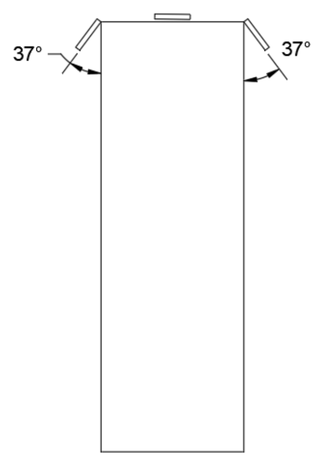


Figure 14: Sensor Orientation.

For simplicity reasons all the sensors will be made the same with the same envelop of detection and distance of detection. With only a 74 degree angle of detection, this will not cover the entire envelop of the center rear indication area. The indication area that will be able to be detected and the calculations are shown below.

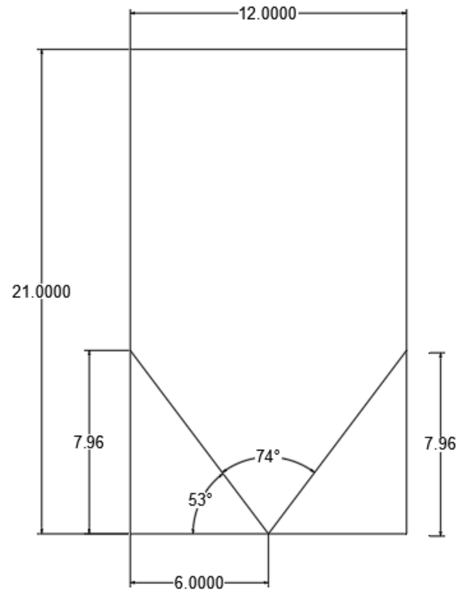


Figure 15: Center Rear Blind Spot Detection.

$$Missed\ Distance = 6 * \tan(53) = 7.96\ ft$$

$$Max\ Distance = \sqrt{21^2 + 6^2} = 21.8\ ft$$

Using the same sensor for the center rear blind spot will result in a small undetectable area in each of the corners near the motorcycle as shown in Figure 15 above. This will not be an issue as the undetectable area is at the sides of the lane and at a very close distance that not many vehicles will be occupying. A typical vehicle is also larger than the size of the undetectable area so at least a small part of the vehicle will be able to be detected. The max distance is smaller than the max distance for the left and right blind spots, so the distance will not be an issue.

Programming the chirp parameters on the radar sensors also factor into the range and accuracy of the detection. Using the chirp parameters shown in the table below, the values of max range, range resolution, and angular resolution.

Table 17: Chirp Parameters.

Chirp Parameters	
Sweep Bandwidth	750 MHz
Ramp Slope	15 MHz/µsec
Inter Chirp Duration	12 µsec
Number of Chirps	128
Chirp Duration	50 µsec
Max Beat Frequency	4.5 MHz
ADC Sampling Rate	5 Msps
Samples per Chirp	250
Range FFT Size	256
Frame Time (Total)	7.94 msec
Frame Time (Active)	6.4 msec
Radar Memory Data Required	512 kB

$$Range_{max} = \frac{IF_{max} * c}{2 * Slope} = \frac{4.5MHz * 3 * 10^8 \frac{m}{sec}}{2 * 15 \frac{MHz}{\mu\text{sec}} * 10^6} = 45\ m$$

In the range equation above "IF_{max}" is the Max Beat Frequency given in the table, “c” is the speed of light, and the slope is the ramp slope of the chirp function also given in the table

above. This set of parameters gives a range of 45 meters which is more than acceptable for the range needed.

$$Distance\ Resolution = \frac{c}{2 * BW} = \frac{3 * 10^8 \frac{m}{sec}}{2 * 750MHz} = 0.2\ m$$

The distance resolution is the distance to which two objects can be distinguished between. In this equation “c” is the speed of light and “BW” is the sweep bandwidth. The distance resolution turns out to be 0.2 meters or roughly 7 inches. This is more than sufficient to be able to clearly identify two vehicles that are driving beside each other.

$$Angular\ Resolution = (\frac{\lambda}{d * R_x * T_x * \cos(\theta)}) * (\frac{180}{\pi}) = 13.75\ degrees$$

The angular resolution is a measure of the angle to which two objects can be distinguished between. In this equation lambda is the wavelength of the signal, d is the physical distance between the two receiver antennas, "R_x" is the number of receiver antennas, "T_x" is the number of transmitter antennas, and “Ɵ” is the angle off of the perpendicular to the antenna. For this application the antennas are places at a distance $\lambda/2$ apart, the number of receiver antennas is four, the number of transmitter antennas is two, and theta is zero for simplicity. The distance between the antennas helps to determine the number of nulls in the radiation pattern of the antenna, or in other words it will help to determine the angles at which the antenna will not be able to pick up any results. This results in an angular resolution of 13.75 degrees. To put this number in perspective the total distance that is not able to be distinguished between at the center of the lane to the side is calculated. Those calculations are shown below.

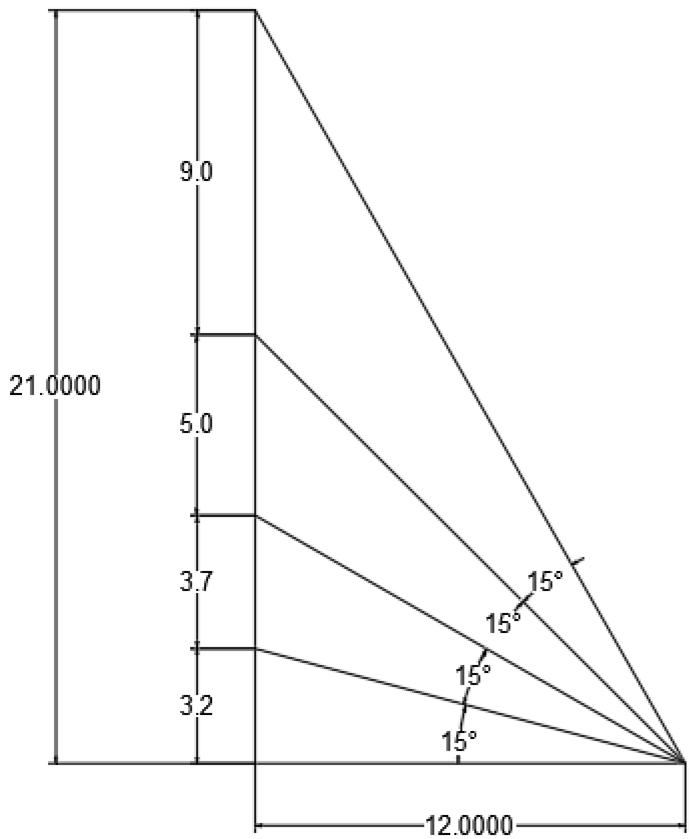


Figure 16: Angular Resolution.

Using the center of the lane directly beside the motorcycle, the angular resolution is converted to the feet in which it will not be able to tell the difference between. The angle resolution was rounded to 15 degrees for simplicity. The calculations of the distances are shown below.

$$d1 = 12 * \tan(15) = 3.2\, ft$$

$$d2 = [12 * \tan(30)] - d1 = 3.7\, ft$$

$$d3 = [12 * \tan(45)] - (d1 + d2) = 5.0\, ft$$

$$d4 = 21 - d1 - d2 - d3 = 9\,ft$$

It can be seen that this will be an acceptable angle resolution as the largest undistinguishable window is 9 ft. This is more than enough as the typical family sedan is typically anywhere from 12 to 14 feet long.

The radar antennas will need to be enclosed in order to protect them from weather and any debris that might come from the road. Radar can go through most objects but in order to cause very little disturbance in the signal the thickness of the material needs to be calculated. First, the wavelength in the material of the signal needs to be calculated using the following equation.

$$\lambda_m = \frac{c}{f\sqrt{\varepsilon_r}}$$

In this equation "λ" is the wavelength in the material, "c" is the speed of light, "f" is the frequency, and "ε" is the permittivity of the material. In this case the material used is polycarbonate which has a permittivity of $2.568x10^{-11}\,{}^{F}/_{m}$, the speed of light is $3x10^{8}\,{}^{m}/_{(\mathrm{sec})\hat{}2}$, and the frequency is 80 GHz. This equation the produces a wavelength of 740km. Then using the equation below gives that the thickness needs to be a multiple of the wavelength divided by two.

$$t = \frac{n\lambda_m}{2}$$

In this equation "t" is the thickness, "n" is an integer, and "λ" is the wavelength in the material from the previous equation. This gives a thickness of 3.7mm.

As briefly mentioned above the radar sensors will be programmed with a set of chirp parameters. Chirps are a brief triangular wave signal that will be put out at a certain frequency. These chirps will then be reflected off of an object and back to the receiver antennas. However, the received signal will have a different frequency than the transmitted signal. This is what is referred to as the Doppler Effect. From this set of returning waveforms with different frequencies, the fast Fourier transform can be used to pinpoint a common range of frequencies that show at which frequency a majority of signals are returning. It is at this frequency that shows there is an object reflecting a majority of waves to the receiver antenna. From this specific returned frequency the distance an object is away as well as its velocity can be calculated. The distance and velocity can be calculated using the equations below.

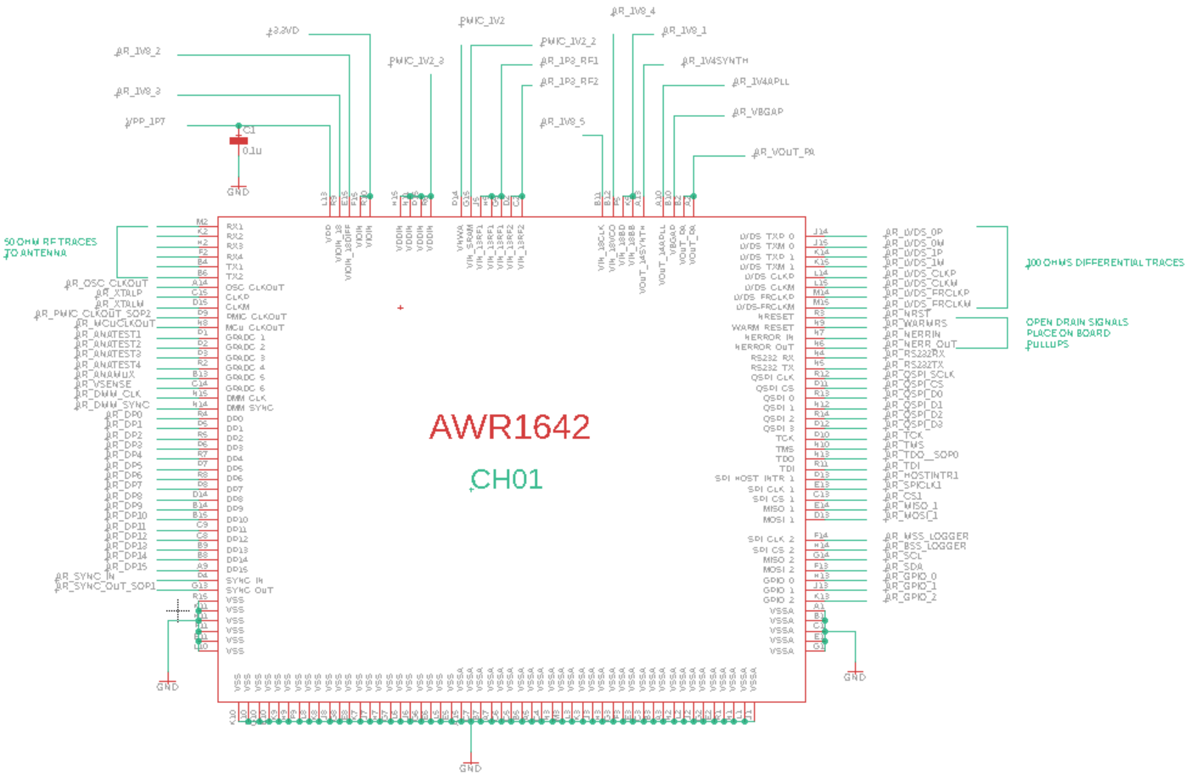
$$\Delta f = f_{reflected} - f_{transmitted} = \frac{2v}{c} f$$

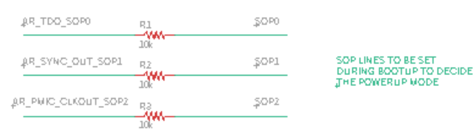
$$\emptyset = -\frac{(2d)(2\pi)}{\lambda}$$

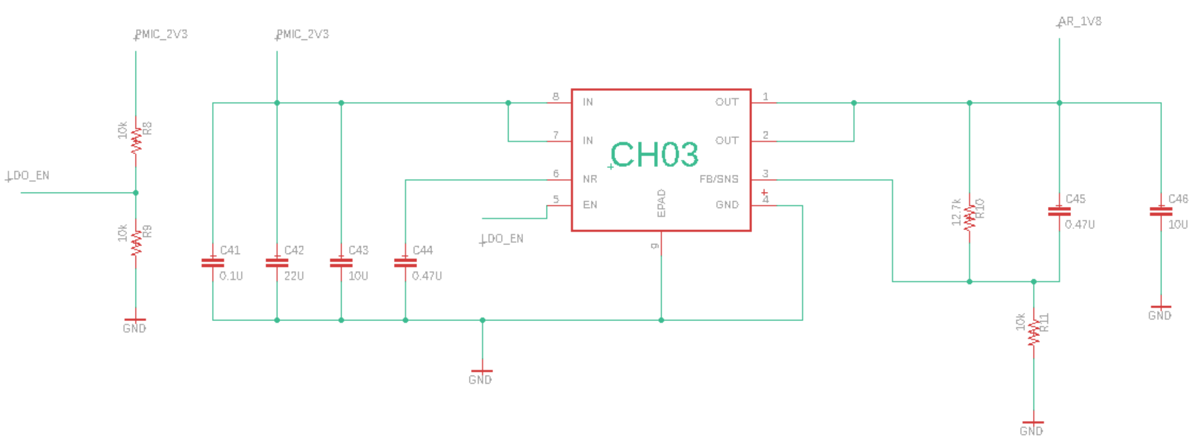
In this velocity equation first, “f” is the original frequency, “v” is the velocity of the object, and “c” is the speed of light. In the distance equation second, “Φ” is the phase difference between the transmitted and received signal, “d” is the distance to the object, and “λ” is the wavelength of the transmitted signal.

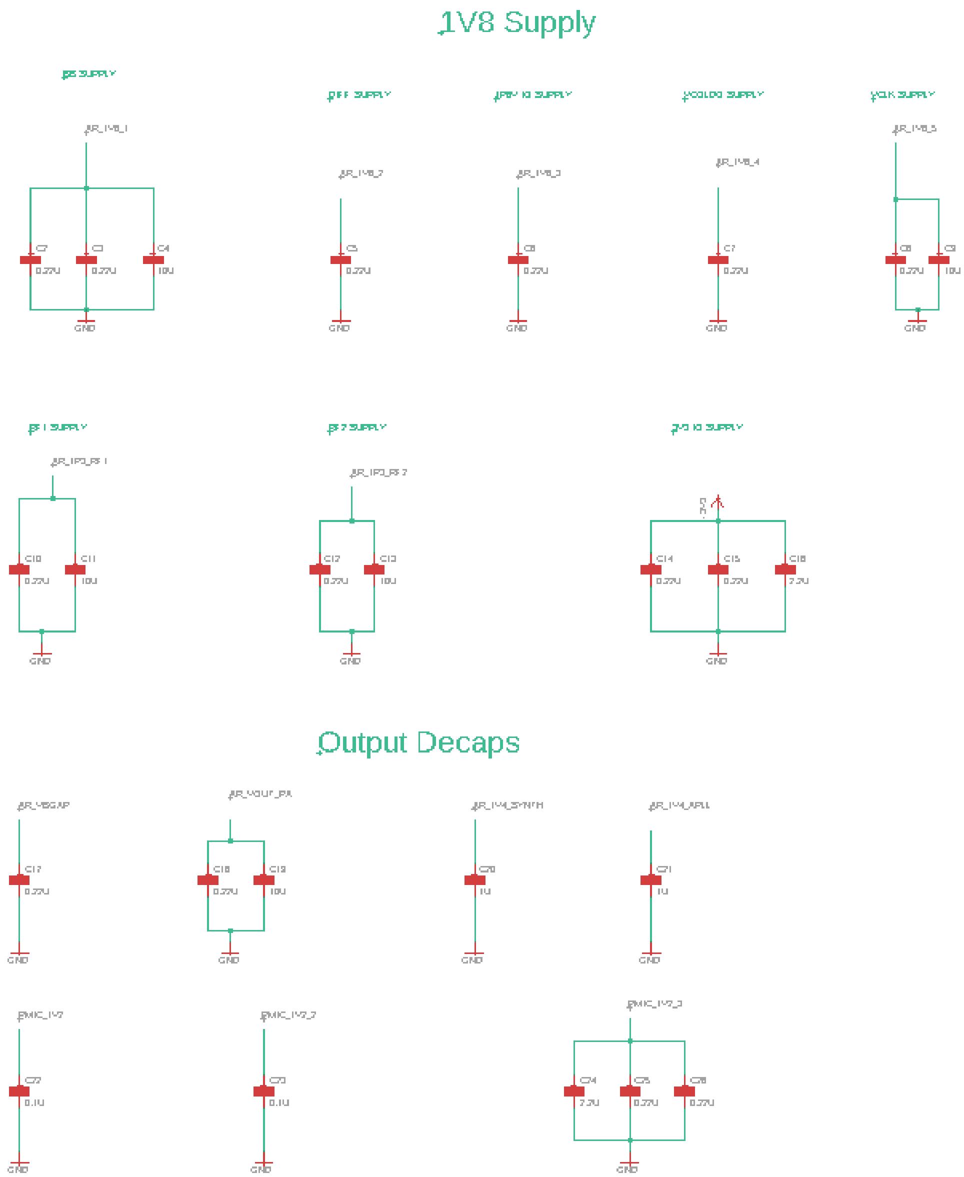
A radar sensor is needed to capture the desired range as well as meet all the specifications desired above. To build this radar sensor a printed circuit board (PCB) will be designed utilizing the AWR1642 by Texas Instruments. A 3D image of the chip must be constructed and then the pinout can be wired. A design for this is shown in the figure below. As mentioned in the calculations above the sensor will be powered off of 3.3 volts and utilize 2

transmit antennas and 4 receiver antennas. (CJW)








1V8 Supply
Output Decaps

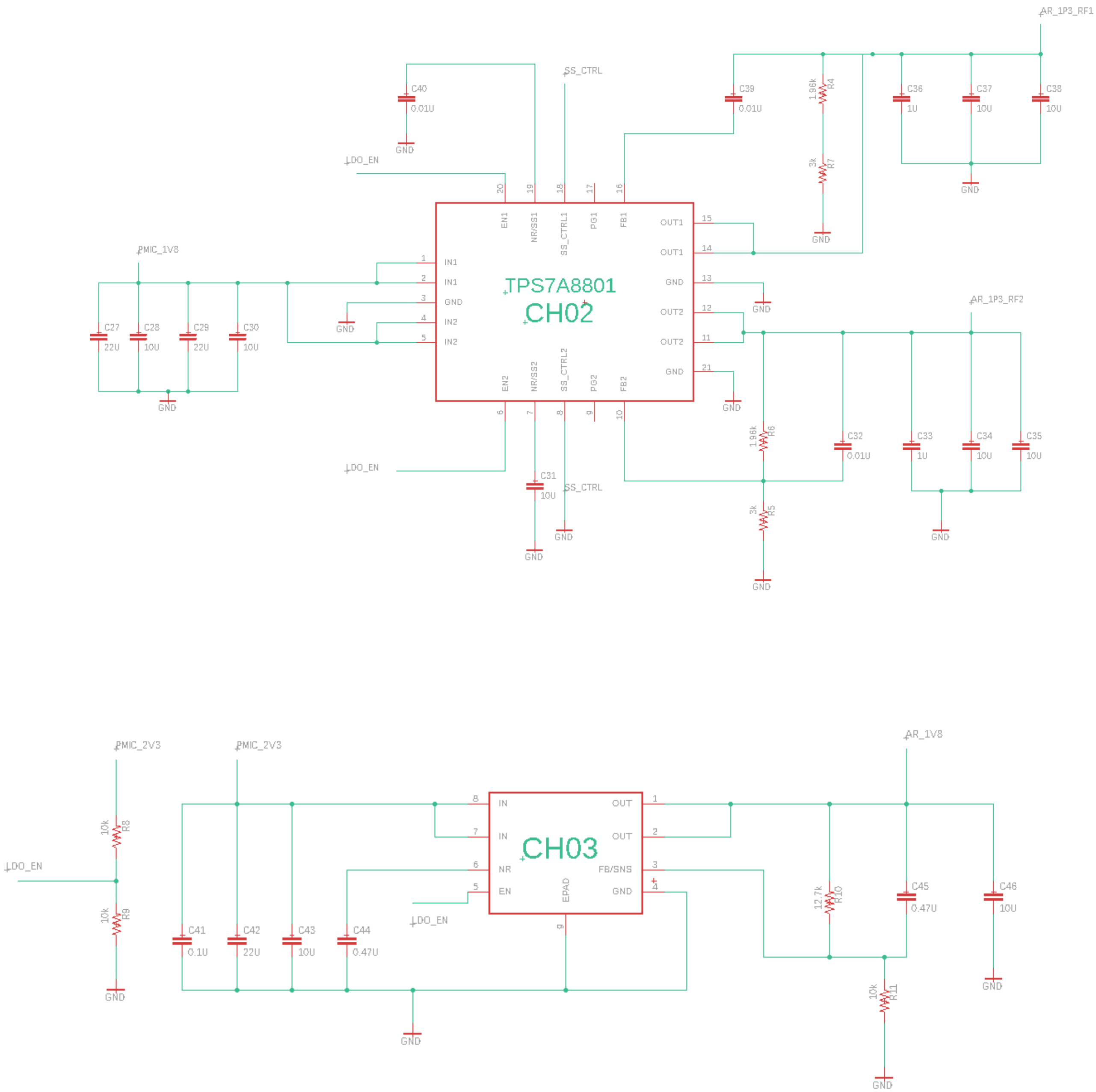


Figure 17: Wiring Schematics of Radar Chip.

For ease of programming the sensors, a microcontroller must be used for each sensor in order to be able to program the fast Fourier transform. The fast Fourier transform is used to sample and interpret the analog signal from each sensor and break it down into a single recognizable frequency that can be used to determine the speed and location of the detected vehicle. (CJW) The Fast Fourier transform takes an analog signal in the time domain and

transforms the function associated with the signal to the frequency domain. The Fourier transform uses the complex sinusoids of the different frequencies associated with the original function in the time domain. The Fast Fourier transform is a type of implementation of the DFT. DFT is the Discrete Fourier Transform. The worst case runtime or otherwise known as the Big-Oh runtime for the Discrete Fourier Transform is O(n2) where n is the samples that were taken of the input signal's function in the time domain. The Fast Fourier transform exhibits a much faster runtime. The Fast Fourier Transform has a worst case runtime of O(nlog(n)).(PJN)

Shown in Figure 18 are the graphs for the FFT. These were created using Matlab. An input signal was created with noise to represent what would be our signal from the sensors. This signal was then interpreted using the FFT function within Matlab as well as the FFT equation. This was done to ensure that our FFT calculations were correct as we were able to compare the magnitude and phase of both outputs. These outputs were identical, thus ensuring our FFT calculations were correct. (PJN)

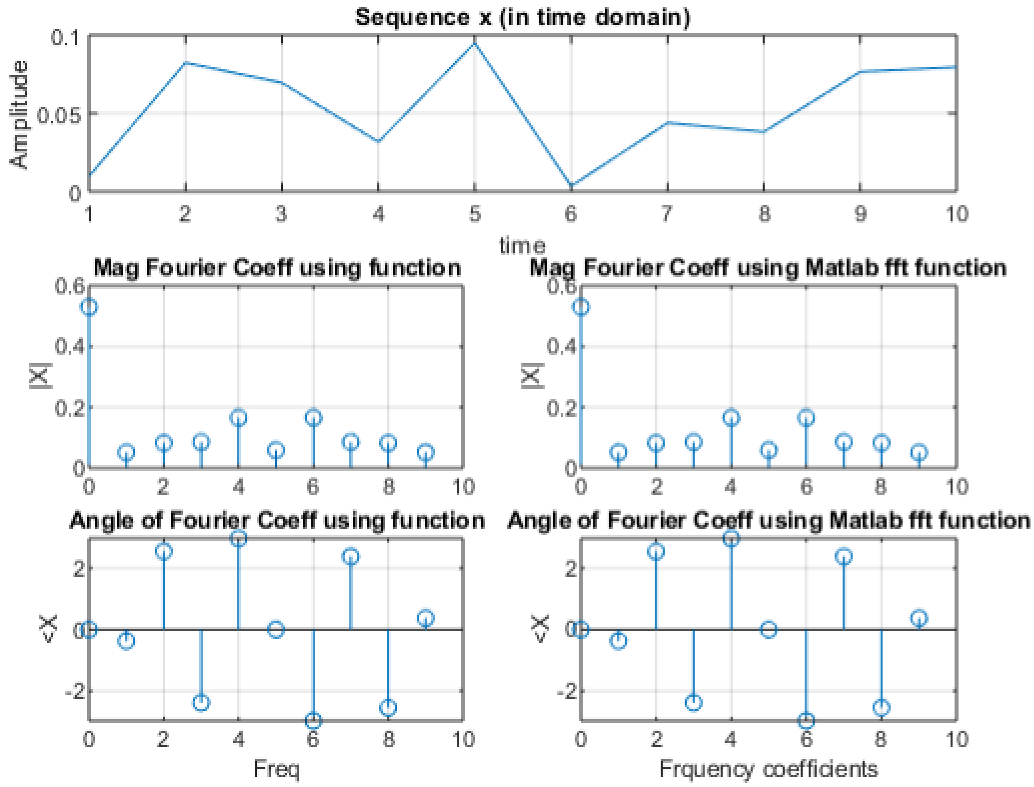


Figure 18: FFT Graphs.

The Bluetooth module that was chosen for the system was the CC2650MODA Bluetooth Module. Figure 18 displays a CADD rendered drawing of the chip displaying the 32 pins. BLE 4.2 Module which supports 2.6 times higher throughput than BLE 4.1. BLE stands for Bluetooth low energy. Bluetooth low energy provides a similar range to traditional Bluetooth versions while consuming less power. This is going to be beneficial as we maintain a charge within the helmet. The module supports I2C and can be programmed as the master or the slave. This becomes beneficial when we are programming and configuring the radar sensor chips. The 3 chips should be able to communicate properly to the Bluetooth module as one is the slave and one is the master. The module accompanies the code composer studio for easy of configuration. With 15 GPIO's this is more than efficient for our 3 radar chips.(PJN)

The device supports idle mode while the peripherals and the radar sensors remain clocked. However the CPI and memory are not clocked and there is no code that is executed. A programmed interrupt from one of the GPIO's triggers the active mode within the processor. This process of transitioning from idle mode to active mode based on an interrupt will decrease power consumption and thus increase the helmet's time between charges.(PJN)

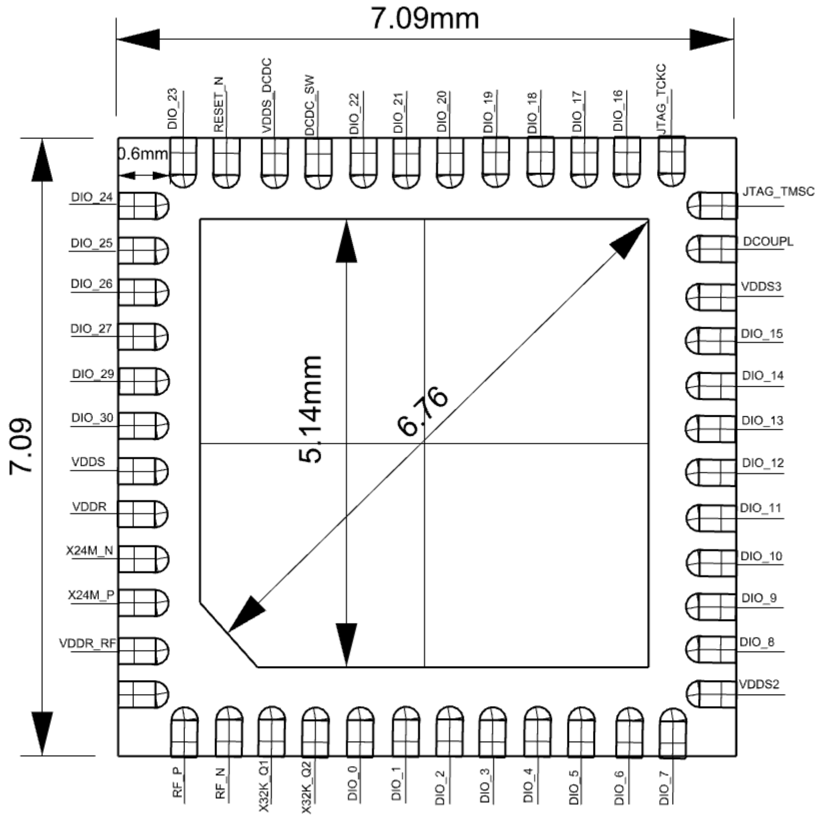


Figure 19: Bluetooth Chip.

Figure 20 shows the mechanical drawing for the helmet view for the system. The drawing shows three small LEDs across the top of the visor on the helmet. This will allow us to alert the driver that a vehicle has entered that sides corresponding blind spot.

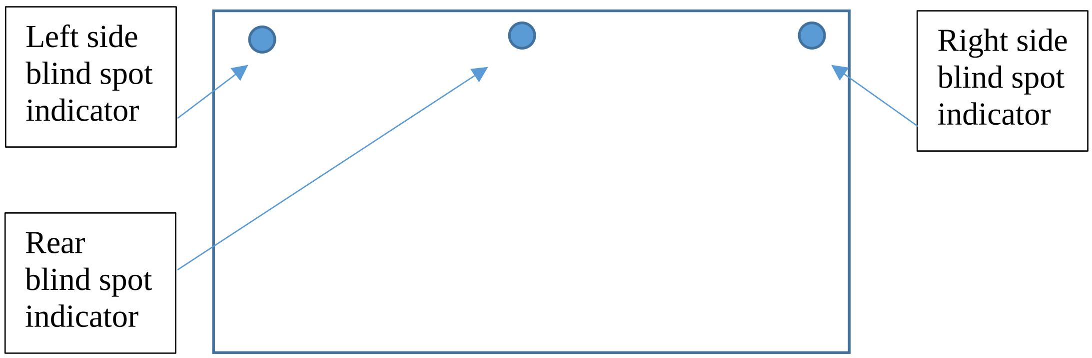


Figure 20: Blind Spot Notifications.

For the power source on the helmet we will be using a rechargeable circuit using a Lithium Ion battery or Li- Ion for short. The reasoning for us to be using a rechargeable power source instead of a non-rechargeable one is one, for consumer convenience. Without a rechargeable power source the user would be forced to either need to have a cord running to the power supply of the bike or constantly be forced to change the battery which is not convenient or ECO friendly. The problem of the cord would be a hazard for the driver by limiting their mobility. The other reason for using a Li-Ion rechargeable circuit is the size of this. Another hazard for the driver would be the weight of the helmet. Since the circuit is attached to the helmet, it is added weight for the driver. So by using this lightweight rechargeable circuit we can limit the strain of wearing a helmet while riding. (SMZ)

Li- Ion batteries use an intercalated lithium compound as one electrode material unlike the metallic lithium in non- rechargeable Li- Ion battery. This allows for ionic movement which is the ability for a charged particles to move through a medium in response to an electric field. When the battery is fully charged and is put into application the battery will start to discharge. While the ions are discharging, they are moving from the negative electrode through an electrolyte to the positive electrode. Once the ions on the negative side are to the positive side

the current stops. To get the battery back to normal you charge it which forces the ions to move from positive to negative.

There are a lot of safety precautions needed when dealing with Li-Ion batteries. One thing to note is that today's Li-Ion batteries they use lithium- based intercalation compounds because pure lithium is highly reactive making an overall safer for use. Another factor to consider is the charging of the circuit. To an extent, increasing the charging current decreases charge time however this is not the case. Since the ions in the battery have a finite mobility, "Increasing the current passed a certain threshold doesn't shift them any quicker. Instead, the energy is actually dissipated as heat" [14]. Also, charging at high current can eventually cause the ions to embed themselves into the negative side of the battery overall ruining the battery.

To account for this, Li- Ion battery charging follows a profile designed to ensure the safety and performance of the battery. If a battery is discharged to below 3V, a pre- conditioned charge around 10% of a full charge. This prevents overheating of the battery. Another safety feature is when the battery reaches 4.2V, the charger switches to constant voltage phase, which eliminates overcharging. To counter the self-discharging battery still in the circuit a top up charge is activated to keep the battery between 3.9V and 4.2V. If you continually overcharge the battery, the ions stop moving converting all the electrical energy to thermal potentially leading to a possible explosion. (SMZ)

It is very easy to calculate the power of the helmet. The components of the hardware in the helmet is a Bluetooth receiver and three LED's. Putting all of these together we get a power rated at .231W shown below.

$$Power\ of\ Helmet$$

$$= (3V_{Bluetooth} * 10mA_{Bluetooth}) + 3(3V_{Led} * 20mA_{Led}) + 2(3V_{Buzzer} * 3.5mA_{Buzzer}) = .231W$$

All of this can be put in a simple circuit as shown in figure 21.

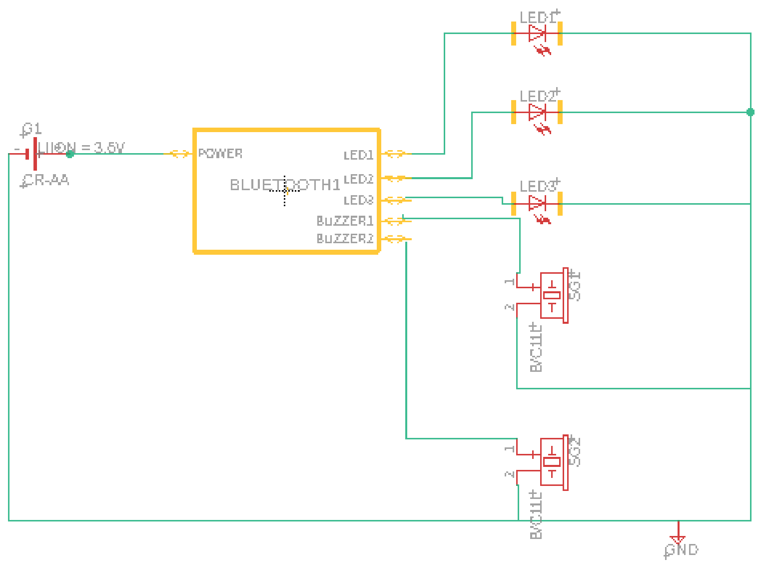


Figure 21: Helmet Circuit.

As shown in the circuit above a single 3.6V Li- Ion battery to power the circuit. This is enough to power the Bluetooth receiver as well as the LED's and buzzers for the circuit. The receiver will receive the signal from the bike and light up the corresponding LED and buzzer in the helmet. Along with this circuit we will also have a rechargeable circuit for the Li Ion battery.

We will be using the MAX8900A switch mode Li+ Charger. This device allows us to monitor the batteries temperature so we can assure we do not overcharge the battery. The device can be seen below in Figure 22.

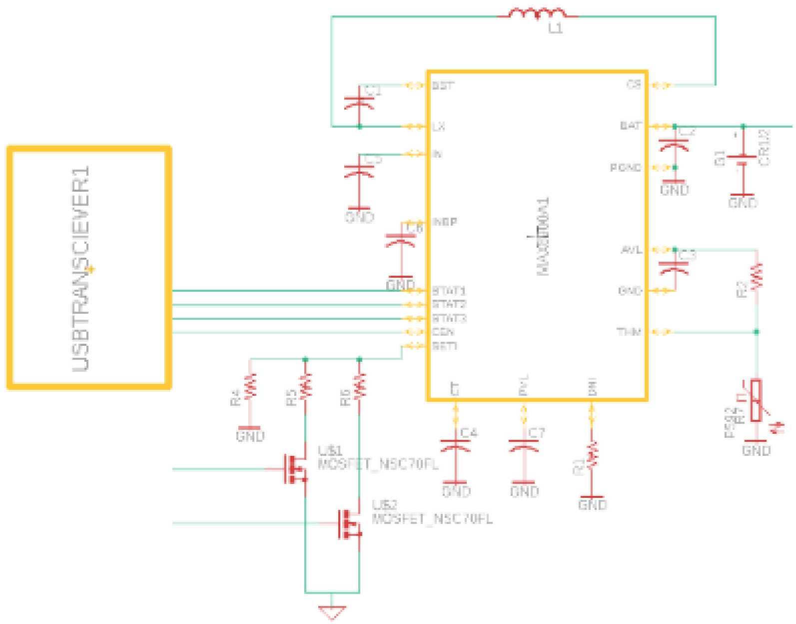


Figure 22: MAX8900A.

This device is attached to a USB transceiver and connector to make for a simple way for users to charge the device. The battery we will be charging will have a life of 2600mAh which will give us at least a two-hour lifespan before it would have to be recharged due to the limited current in the circuit. A safety precaution is built into the system with the pin CEN. When CEN is high or when the input voltage is out of the range of -22V-+22V the system will become disable.

Another safety feature built in is the Thermistor Monitoring. There are four temperature thresholds that the battery can operate at, T1, T2, T3, and T4. If the thermistor receives an input less than T1=0°C or greater then T4= 60°C again the system will shut off. Values of T2 and T3 will tell the charger to either raise or lower current to limit stress in the battery. The relationship of thermistor resistance to temperature is shown below.

$$R_{Thrm} = R_{25} * e^{\beta\left(\frac{1}{T+273℃}-\frac{1}{298℃}\right)}$$

Where Rthrm is the resistance in Ω of the thermistor at temperature T, R25 is resistance at 25°C, β is the material constant of thermistor and finally T is the temperature of thermistor. We also need to pick out inductor values for the circuit. Since we are operating in between an input voltage range of 3.4V – 8.7V it is recommended we use 1uH. (SMZ)

The MOSFETs connected to SETI can drive the discharge current up or down depending on when they are turned on. When the MOSFET is on the associated resistor or resistors are added in parallel. This causes the total SETI resistance to decrease which causes the current to increase. This can be shown below. (SMZ)

$$I_{FC} = \frac{3405V}{R_{Seti}}$$

System Implementation

When going to implement the design there were many issues that arose and had to be worked around. The first issue came with the radar sensors. The plan was to use the AWR1642 chip to design a printed circuit board (PCB) with built in patch antennas to be programmed and sense the vehicles in the blind spot. The first problem with that was that the device operates at 80 GHz which is to complex and costly to create a PCB of the chip so direction was moved to programing the evaluation board of the AWR1642 that was purchased from Texas Instruments. This proved very difficult to program as it was designed to track up to thirty cars at once and only display an output when the vehicle is within the certain output range. We were able to get the TI sensor working at short range with a Matlab add on from Texas Instruments but we were not able to get it to send an output as we desired. With making no real progress with the AWR1642 chip we decided to turn to LIDAR. A LIDAR sensor was purchased and began to be programmed. This was difficult to use with the PIC24 microcontrollers as it was first attempted

with I2C before realizing the PIC has difficulties accepting inputs. The three sensors were programmed with Pulse Width Modulation (PWM) which was able to get all three sensors working simultaneously in real time. The signal was interpreted and used to send an output bit high which was send to the Bluetooth module in order transmit to the helmet. There was originally some noise in the output signal which caused the output bit to flicker but that was resolved by slowing down the operation of the sensors. One design requirement was to have the sensors work in real time so the sensors were only slowed down marginally so that there is no difference between when a car is in the blind spot and when the light comes on that is detectable by the human eye. (CJW)

The DC to DC converter design was initially successful. The PCB board design is pictured below in Figure 23.

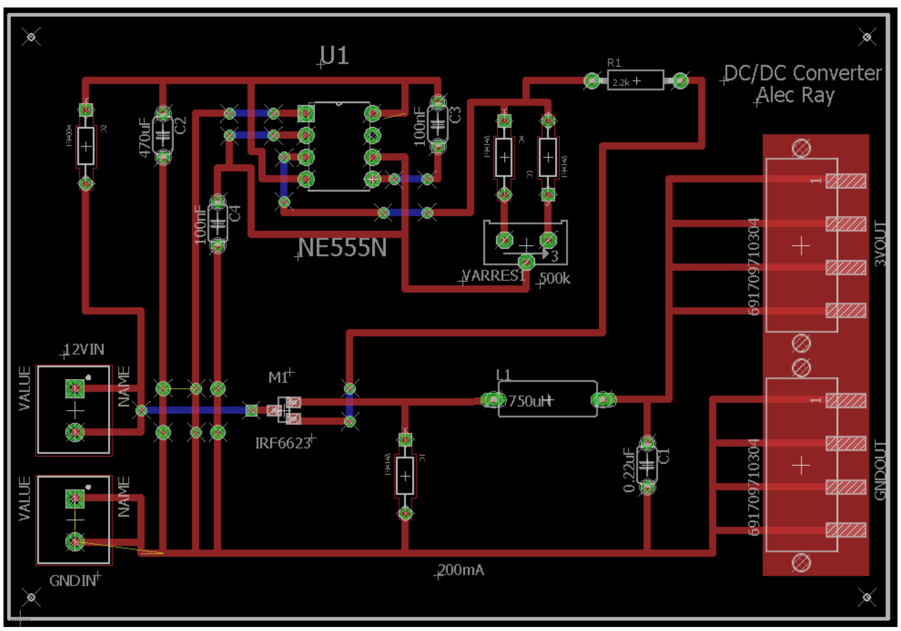


Figure 23: Converter PCB Design.

Testing the circuit produced excellent waveforms displaying the effectiveness of the switching circuit and the output signal. These are shown below.

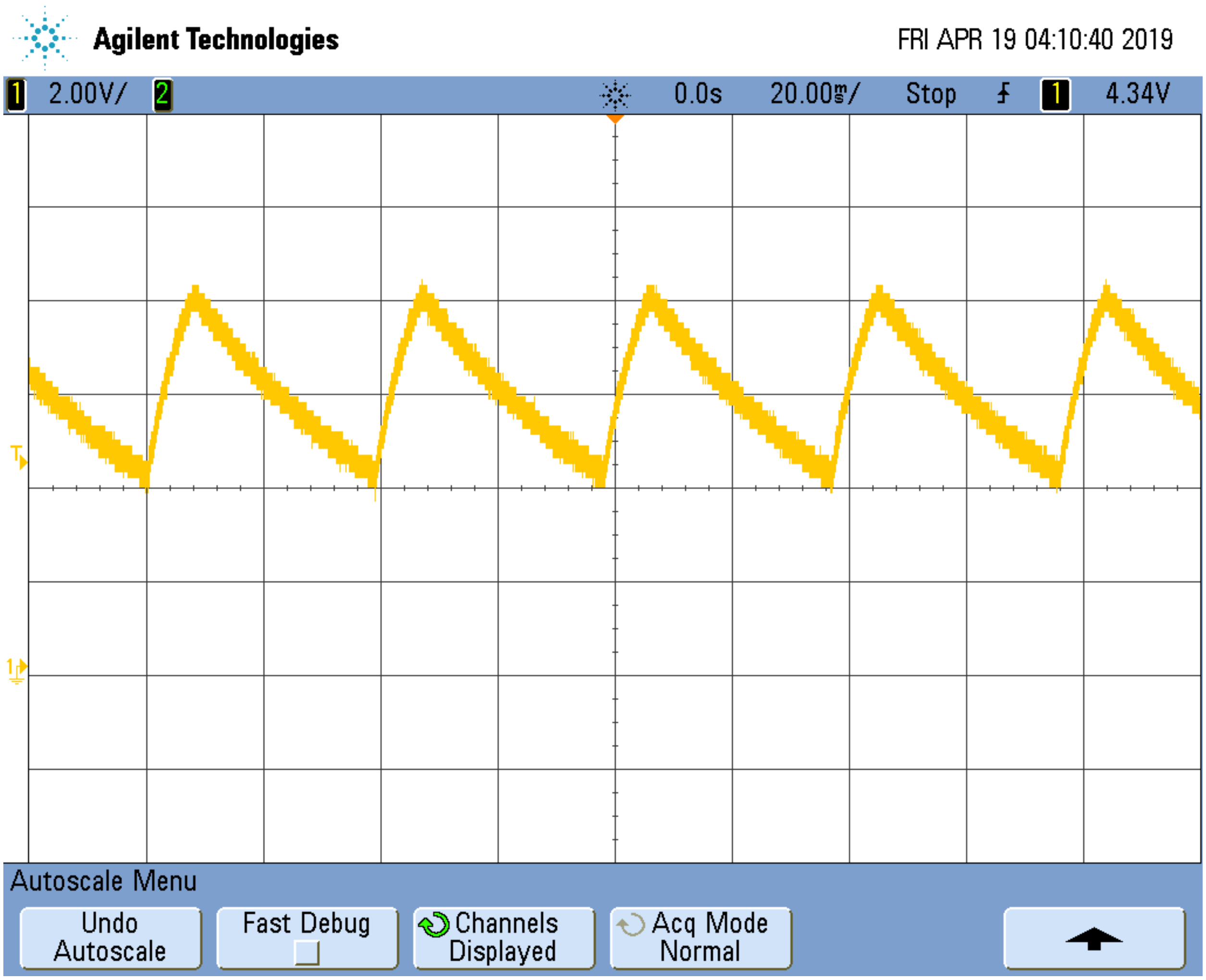


Figure 24: Triggering Signal.

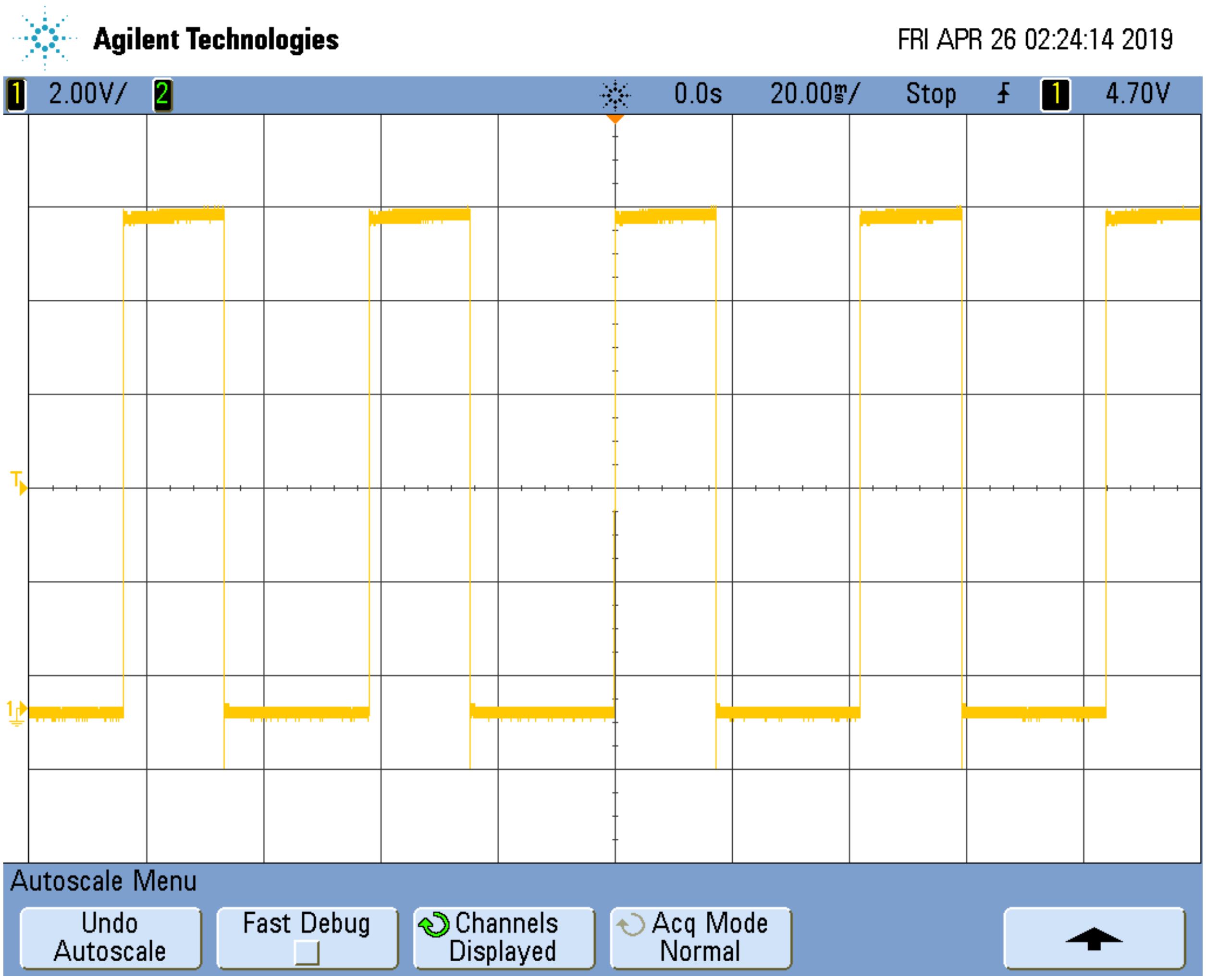


Figure 25: PWM Signal.

The output voltage signal was achieved at a 6% ripple.

However, another design issue arose when the DC to DC converter was hooked up to the sensors. Since the original design had planned on using the AWR1642 chip, the calculations for the converter were carried out with the idea of supplying less current to the system. With the new sensors and boards, a new MOSFET was needed in the converter circuit so that the necessary current and power could be supplied. Once this was achieved however, the system experienced a large amount of noise, presumably from the ripple of the DC converter. As a solution, it was determined that the evaluation board can be used as a converter to power the sensors, and the DC

to DC converter was then repurposed in the rechargeable battery circuit. Now, the new design allows for the battery circuit that is located in the helmet to be charged off of the 12V from the battery. This is carried out by stepping down the 12V to the necessary 5V needed for the system. The slight design changes still met all of the original marketing requirements of space and weight. (AJR)

After careful analysis it was found that this chip is not suitable for this application. When first running this application with a simulated battery using a DC power supply and a 2Ω resistor in parallel connected between BAT pin and GND. This was not allowing for accurate readings in the circuit because it was just measuring the DC current across the resistor. This value was the correct current so when I switched the simulated battery for the 3.7V Li Ion battery the chip was drawing to much current and fried the board. Upon further tests with a different undamaged chip and attaching it to a breadboard results could not be repeated. Do to this and not being able to check if part was damaged do to type of device (30 Array Bumps) it was decided to switch to the device MCP73831/2 SOT-23-5. This is an integrated circuit controller for LI-ION batteries up to 4.2V.

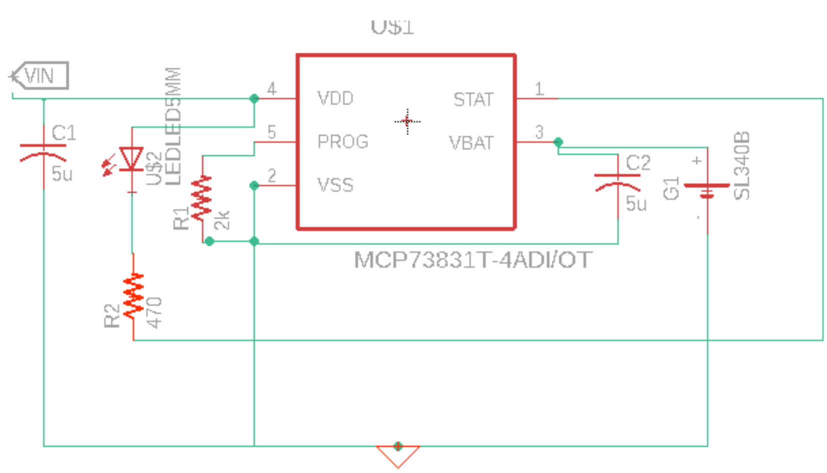


Figure 23: Li- Ion Battery Charger MCP73831/2 SOT-23-5.

To program this device there are a few prequalification to go through to make sure the battery is properly charged. The first prequalification is R1 (Rprog). To set this we need to know the charge current of the battery (Ireg) which is 0.52A. From this we can use the equation below to set Rprog found to be 2kΩ.

$$\text{Ireg} = \frac{1000V}{Rprog}$$

The other prequalifications are the capacitors. It is recommended to use ceramic capacitors do to the reverse polarity of current from the input power supply and the Li Ion battery. In the datasheet it is recommended to us 4.7uF capacitors for a current of 500mA. Since it is needed to have 520mA we need to use a slightly higher capacitance so it is chosen to be 5uF or greater for the greatest AC stability. This also provides compensation when there is no battery load, compensating for the inductive nature of the battery.

The final resistance value needed is R2 (Rled). This is the resistance that limits the current to the LED. This is needed so the LED does not exceed its current limit. This step is not as important as the others but without it would damage the stat pin which is responsible for showing the device is charging, done charging or low current. It was recommended by the datasheet to use a 470Ω resistor. (SMZ)

Once all of this is completed the circuit will be able to provide a current of 500mA for about a 4 hour charge time. But due to low current flow of the circuit the battery can be run for over 50 hours. The chip also has an internal switch to prevent over charge and not going to damage the battery or over heat.

While the old design using the MAX8900A chip would be able to provide a similar result it was found using the MCP73831 the device can be ran at a similar efficiency, the MCP73831 is a much smaller and overall cheaper design if needed to become a regularly produced product. (SMZ)

For the Bluetooth implementation that is to be used to transmit signal from the bike to the helmet a Texas Instruments CC2650 Module was to be used. Through analyzing the chip and understanding the operating specifications it was decided to be the best option. Upon using this chip and programming the device it was found to not perform. The device had a Real Time Operating System (RTOS) . This made it extremely difficult to program without affecting the processes that were occurring. The device was able to be programmed to be seen from a mobile device. However, when it came time to transfer the packets of data across a Bluetooth connection to another device, this is where there were difficulties. The configurations and code that was required to do so was troublesome and not performing correctly.

Slightly after the decision to use LIDAR sensors and a PIC24fj1024gb610 PIM on an Explorer 16/32 evaluation board it was decided to switch to a Microchip Bluetooth module. The module that was used was an RN4871 Click. This utilized UART communication and was programmed using an Explorer 16/32 evaluation board. The Explorer board used the MSP 2221A serial to UART chip. This enabled the Bluetooth module to be programmed using a normal USB connection to a PC. The COM port that was used by the USB was identified and this port was used to communicate with the chip through a terminal emulator running on a PC. The terminal emulator used the configuration of a baud rate of 115200 and 8 data bits.

Once able to connect to the RN4871 module, the module could be programmed using ASCII commands. While in command mode, the name, features, pin configuration, and Generic

Attribute Profile (GATT) settings. The module on the bike is configured as the client and the module in the helmet is configured as the server. The Universally Unique Identifier was configured on the server as well as the characteristics that were to be used for the communication. This is then communicated with the client and used as services to send and receive data. The server module looks for a signal from the client every 25 mS. Once it receives a signal from the client, it then uses this value to assign to the GPIO pin. The signal that was sent from the client is based on the reading of the GPIO pin. Once the pin senses a high signal from the sensors, a write is performed to the server with this value. (PJN)

Table 18: Parts List.

Qty.	Refdes	Part Num.	Description
1		AWR1642BOOST	Evaluation board for AWR1642 chip
3	Ch01	AWR1642ABIGABLR Q1	Single Chip Radar
1		2001-2007	Motorcycle Battery
3	Ch03	TPS7A8101DRBT	High Bandwidth PSRR, 1 A Linear Regulator
3	Ch02	TPS7A8801QRTJRQ1	Automative Voltage Regulator
1		MAX8900AEWV+T	IC BATT CHRGR LI-ION/POLY 30WLP
2	B01,B02	296448051ND	Low Power Bluetooth Module
1		638-1092-1-ND	USB Transceiver
1		WM6294CT-ND	USB Connector
1			Li- Ion Battery
3	U1	NE555	Single Precision Timer
2	C1	C1	Capacitor 220nF
2	C2	C2	Capacitor 470uF
2	C3	C3	Capacitor 100nF
2	C4	C4	Capacitor 100nF
2	D1	1N4148	Diode
2	D2	1N4004	Diode
2	D3	1N4148	Diode
2	D4	1N4148	Diode
2	L1	L1	Inductor 750uH
3	M1	IRF6623	N Channel Mosfet
2	R1	R1	Resistor 2.2K Ohm
2	VarRes1	652-3299Y-1-504	500k Ohm Variable Resistor
Qty.	Refdes	Part Num.	Description
1	usbtransceiv er	638-1092-1-ND	USB Transceiver
1	usbtransceiv er	WM6294CT-ND	USB Connector
1	BAT		Li- Ion Battery
3	U1	NE555	Single Precision Timer
3	USB1	10118193-0001LF	USB - Micro Female
3	CON1	609-3704-2-ND	Female Power Socket
1	usbtransceiv er	AT32UC3B1256-AUT	32-bit Microcontrollers - MCU 32-bit
1	RT	NCU15XH103D60RC	NTC Thermistors
1		AWR1642BOOST	Evaluation board for AWR1642 chip
3	Ch01	AWR1642ABIGABLR Q1	Single Chip Radar
1		2001-2007	Motorcycle Battery
3	Ch03	TPS7A8101DRBT	High Bandwidth PSRR, 1 A Linear Regulator
3	Ch02	TPS7A8801QRTJRQ1	Automative Voltage Regulator

1		MAX8900AEWV+T	IC BATT CHRGR LI-ION/POLY 30WLP
2	B01,B02	296448051ND	Low Power Bluetooth Module
3	M1	IRF6623	N Channel Mosfet
2	VarRes1	652-3299Y-1-504	500k Ohm Variable Resistor
2	L1	HM3341-ND	Inductor 750uH
4	T1, T2	277-1667-ND	2 Position Terminal Block
4	T3, T4	1297-1110-1-ND	4 Position Terminal Block
14	Term	6911371710006	Wire to Board Terminal Block
1	Debug	TMDSEMU110-U	TI - XDS110 JTAG Debug Probe
2	DC Converter		PCB Board
	Bluetooth Circuit		PCB Board
1	Rth	TO103J2J	THERMISTOR NTC 10KOHM 5% TO-220
1	Bat	140-783	Single 18650 Battery Holder with Wire Leads
	Rechargeable Battery		PCB Board
3	U1	NE555	Single Precision Timer
2	VARES 1	3299Y-504LF-ND	500k ohm potentiometer 25 turns
1	Rth	TO103J2J	THERMISTOR NTC 10KOHM 5% TO-220
1	Bat	140-783	Single 18650 Battery Holder with Wire Leads
1	Rechargeable Battery		PCB Board

(AJR)

4. Budget

Table 19: Budget.

Qty.	Part Num.	Description	Cost	Total Cost
1	638-1092-1-ND	USB Transceiever	1.28	1.28
1	WM6294CT-ND	USB Connector	1.30	1.30
1		Li- Ion Battery	7.29	7.29
3	NE555	Single Precision Timer	0.25	0.75
3	10118193-0001LF	USB - Micro Female	0.45	1.35
3	609-3704-2-ND	Female Power Socket	0.67	2.01
1	AT32UC3B1256-AUT	32-bit Microcontrollers - MCU 32-bit	6.28	6.28
1	NCU15XH103D60RC	NTC Thermistors	0.30	0.30
1	AWR1642BOOST	Evaluation board for AWR1642 chip	299.00	299.00
3	AWR1642ABIGABLRQ1	Single Chip Radar	66.92	200.76
1	2001-2007	Motorcycle Battery	52	52
3	TPS7A8101DRBT	High Bandwidth PSRR, 1 A Linear Regulator	2.44	7.32
3	TPS7A8801QRTJRQ1	Automative Voltage Regulator	4.09	12.27
1	MAX8900AEWV+T	IC BATT CHRGR LI-ION/POLY 30WLP	4.22	4.22
2	296448051ND	Low Power Bluetooth Modual	12.09	24.18
3	IRF6623	N Channel Mosfet	0.87	2.61
2	652-3299Y-1-504	500k Ohm Variable Resistor	4.77	9.54
2	HM3341-ND	Inductor 750uH	5.64	11.28
4	277-1667-ND	2 Position Terminal Block	0.43	1.72
4	1297-1110-1-ND	4 Position Terminal Block	5.23	20.92
14	6911371710006	Wire to Board Terminal Block	3.13	43.82
1	TMDSEMU110-U	TI - XDS110 JTAG Debug Probe	102.80	102.80
2		PCB Board	18.50	37.00
		PCB Board		20.00
1	TO103J2J	THERMISTOR NTC 10KOHM 5% TO-220	2.63	2.63
1	140-783	Single 18650 Battery Holder with Wire Leads	1.99	1.99
		PCB Board	2.00	2.00
3	NE555	Single Precision Timer	0.25	0.75
2	3299Y-504LF-ND	500k ohm potentiometer 25 turns	3.39	6.78
1	TO103J2J	THERMISTOR NTC 10KOHM 5% TO-220	2.63	2.63
1	140-783	Single 18650 Battery Holder with Wire Leads	1.99	1.99
1		PCB Board	2.00	2.00
			Total	**$890.77**

AJR

5. Project Schedule

		Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1			SDP	**58 days**	**Thu 9/27/18**	**Mon 12/17/18**		
2			**Midterm Report**	**10 days**	**Thu 9/27/18**	**Wed 10/10/18**		
3			Gannt Chart	3 days	Thu 9/27/18	Mon 10/1/18		Alec Ray
4			Accepted Technical Design	3 days	Thu 9/27/18	Mon 10/1/18		Scott Zipp
5			Software Block Diagram	3 days	Thu 9/27/18	Mon 10/1/18		Pat Nie
6			Hardware Block Diagram	3 days	Thu 9/27/18	Mon 10/1/18		Cogan Warther
7			FR Tables	3 days	Thu 9/27/18	Mon 10/1/18		Cogan Warther
8			Design Requireme	3 days	Thu 9/27/18	Mon 10/1/18		Cogan Warther
9			**Project Analysis Topi**	**10 days**	**Thu 9/27/18**	**Wed 10/10/18**		
10			Analysis into Noise	10 days	Thu 9/27/18	Wed 10/10/18		Alec Ray
11			Analysis into DC to DC Conversion	10 days	Thu 9/27/18	Wed 10/10/18		Alec Ray
12			Radar Sensor Research	7 days	Mon 10/8/18	Tue 10/16/18		Cogan Warther
13			Radar Sensor Decision	1 day	Tue 10/16/18	Tue 10/16/18		Cogan Warther
14			Geometry of Sens	3 days	Thu 9/27/18	Mon 10/1/18		Cogan Warther
15			Antenna Design Research	7 days	Mon 10/15/18	Tue 10/23/18		Cogan Warther
16			Bluetooth Researc	7 days	Mon 10/15/18	Tue 10/23/18		Pat Nie
17			Programming Research	7 days	Mon 10/15/18	Tue 10/23/18		Pat Nie/ Cogan Warther
18			Microcontroller Functionality	3 days	Thu 9/27/18	Mon 10/1/18		Pat Nie
19			Helmet Component	3 days	Thu 9/27/18	Mon 10/1/18		Scott Zipp
20			**Midterm PowerPoint**	**7 days**	**Tue 10/2/18**	**Wed 10/10/18**		
21			Slides	7 days	Mon 10/1/18	Tue 10/9/18		Team
22			Rechargable Circuit	2 days	Mon 10/1/18	Tue 10/2/18		Scott Zipp

		Task Mode	Task Name	Duratio	Start	Finish	Predecessors	Resource Names
23			Rehersal	3 days	Wed 10/10/18	Fri 10/12/18	21	Team
24			**Team Poster**	**8 days**	**Wed 10/10/18**	**Sun 10/21/18**		
25			Team Poster	8 days	Wed 10/10/18	Sun 10/21/18		Pat Nie/ Cogan \
26			**Final Design Report**	**35 days**	**Thu 10/11/18**	**Wed 11/28/18**		
27			Component Layout on Motorcycle	5 days	Thu 10/25/18	Wed 10/31/18		Alec Ray
28			Bluetooth Location Analysis	5 days	Thu 10/25/18	Wed 10/31/18		Alec Ray
29			Power Calculations	5 days	Thu 10/25/18	Wed 10/31/18		Alec Ray
30			Wiring Harness	5 days	Thu 10/25/18	Wed 10/31/18		Alec Ray
31			Info on Bluetooth Module	6 days	Wed 10/31/18	Wed 11/7/18		Pat Nie
32			FFT Research	6 days	Wed 10/24/18	Wed 10/31/18		Pat Nie
33			Analysis into Noise Filter	5 days	Thu 11/1/18	Wed 11/7/18		Alec Ray
34			Analysis into Proper Grounding Techniques	5 days	Thu 11/1/18	Wed 11/7/18		Alec Ray
35			Component Wiring Diagram	6 days	Thu 11/1/18	Thu 11/8/18		Alec Ray
36			Info on Lithium Ion	2 days	Sat 11/3/18	Mon 11/5/18		Scott Zipp
37			DSP FFT MATLAB	9 days	Wed 10/31/18	Mon 11/12/18		Pat Nie
38			Block Diagrams	4 days	Thu 10/11/18	Tue 10/16/18		
39			Schematics	8 days	Wed 10/17/18	Fri 10/26/18	38	
40			Rechargable Circuit Schematics	14 days	Sat 11/10/18	Wed 11/28/18		Scott Zipp
41			Psuedo Code	8 days	Mon 10/29/18	Wed 11/7/18	39	Pat Nie
42			Bluetooth Chip Eagle CAD	21 days	Mon 11/19/18	Mon 12/17/18		Pat Nie
43			Parts List	3 days	Thu 11/8/18	Mon 11/12/18	41	Team

44			Materials Budget	3 days	Tue 11/13/18	Thu 11/15/18	43	Team
45			Gantt Chart	2 days	Fri 11/16/18	Mon 11/19/18	44	Team
46			Radar Application Calculations	7 days	Mon 11/5/18	Tue 11/13/18		Cogan Warther
47			PCB layout Research	5 days	Sun 11/11/18	Thu 11/15/18		Cogan Warther
48			DC to DC Converter Design	10 days	Thu 11/15/18	Wed 11/28/18		Alec Ray
49			PCB Design Eagle CAD	21 days	Mon 11/19/18	Mon 12/17/18		Cogan Warther
50			Chirp Parameters/ Doppler Calculations	5 days	Mon 11/12/18	Fri 11/16/18		Cogan Warther
51			Abstract	3 days	Tue 11/20/18	Thu 11/22/18	45	Cogan Warther
52			Conclusions	3 days	Fri 11/23/18	Tue 11/27/18	51	Scott Zipp
53			Format Final Report	2 days	Fri 10/26/18	Sun 10/28/18		Alec Ray
54			**Final PowerPoint**	**14 days**	**Mon 10/22/18**	**Thu 11/8/18**		
55			Slides	7 days	Mon 10/22/18	Tue 10/30/18		
56			Rechargable Circuit	1 day	Thu 11/22/18	Thu 11/22/18		Scott Zipp
57			Rehersal	7 days	Wed 10/31/18	Thu 11/8/18	55	Team

Task Name	Duratio	Start	Finish	Predecessors	Resource Names
SDP Spring 2019	**85 days**	**Tue 1/15/19**	**Sat 5/11/19**		
Software	**85 days**	**Tue 1/15/19**	**Sat 5/11/19**		
Radar PCB	21 days	Tue 1/15/19	Tue 2/12/19		Cogan Warther
Microcontroller PCB	21 days	Tue 1/15/19	Tue 2/12/19		Pat Nie
Program Radar Sensor	51 days	Tue 2/12/19	Tue 4/23/19		Pat Nie/ Cogan Warther
Program Microcontroller/Blueto Module	51 days	Tue 2/12/19	Tue 4/23/19		Pat Nie/ Cogan Warther
Hardware	**81 days**	**Tue 1/15/19**	**Tue 5/7/19**		
DC/DC Converter PCB	14 days	Mon 1/21/19	Thu 2/7/19		Alec Ray
TI NE555 Timer Testing	7 days	Thu 2/7/19	Fri 2/15/19		Alec Ray
Rechargable Circuit PCB	21 days	Thu 1/17/19	Thu 2/14/19		Scott Zipp
Helmet Circuit	7 days	Thu 2/14/19	Fri 2/22/19		Scott Zipp
DC/DC Converter Testing	**58 days**	**Fri 2/15/19**	**Tue 5/7/19**		
Testing Converter with Power Supply	14 days	Fri 2/15/19	Wed 3/6/19		Alec Ray
Testing Converter with Radar Sensors	16 days	Wed 3/6/19	Wed 3/27/19		Alec Ray, Cogan Warther
Testing Converter with Bluetooth Module	14 days	Wed 4/10/19	Mon 4/29/19		Alec Ray, Pat Nie
Install Radar on Motorcycle	11 days	Tue 4/23/19	Tue 5/7/19		Cogan Warther
Testing	14 days	Fri 2/22/19	Wed 3/13/19		Scott Zipp
Assembly	14 days	Sat 3/16/19	Wed 4/3/19		Scott Zipp
Install Microcontroller and Bluetooth on Motorcycle	11 days	Tue 4/23/19	Tue 5/7/19		Pat Nie
Wiring Harness	**7 days**	**Mon 4/29/19**	**Tue 5/7/19**		
Assembly	7 days	Mon 4/29/19	Tue 5/7/19		Alec Ray

6. Team Information

Project Leader – Scott Zipp EE

Archivist – Alec Ray EE

Hardware Manager – Cogan Warther EE

Software Manager – Patrick Nie CpE

7. Conclusions and Recommendations

The creation of a motorcycle blind spot detection system will help to ensure motorcycle operator safety and reduce vehicular accidents. The design outline in this report demonstrated multiple aspects of electrical and computer engineering including, but not limited to power design, Bluetooth operation, circuit construction, signal analysis and conversion, and microcontroller programming. Further analysis will be conducted to determine the proper components needed to ensure that the system is operational and meets all necessary design requirements. The system met most of the design requirements set in the introductions. Multiple issues arose on the software side and those issues were resolved in a timely manner while keeping the design requirements in mind and still working to meet those specifications set forth in the research phase of the project. (CJW)

8. References

[1] M. Norgia, I. Boniolo, M. Tanelli, S. M. Savaresi, and C. Svelto, “Optical sensors for real-time measurement of motorcycle tilt angle,” *IEEE Trans. Instrum. Meas.*, vol. 58, no. 5, pp. 1640–1649, 2009.

[2] C. Liguori, V. Paciello, S. Member, and A. Paolillo, “ISO / IEC / IEEE 21451 Smart Sensor Network for the Evaluation of Motorcycle Suspension Systems,” *Sensors Journal, IEEE*, vol. 15, no. 5, pp. 2549–2558, 2015.

[3] Traffic Safety Facts Laws, Motorcycle Helmet Use Laws. Washington, DC: National Highway Traffic Safety Administration, 2006.

[4]Goodwin A, Kirley B, Sandt L, Hall W, Thomas L, O’Brien N, Summerlin D. Countermeasures that work: A highway safety countermeasures guide for State Highway Safety Offices. 7th edition. (Report No. DOT HS 811 727). Washington, DC: National Highway Traffic Safety Administration; 2003.

[5]Kraus JF, Peek C, McArthur DL, Williams A. The effect of the 1992 California motorcycle helmet use law on motorcycle crash fatalities and injuries. JAMA. 1994;272(19):1506–11.

[6]Chiung-Yao Fang, Jui-Hung Liang, Chiao-Shan Lo, Sei-Wang Chen, "A real-time visual-based front-mounted vehicle collision warning system", *Computational Intelligence in Vehicles and Transportation Systems (CIVTS) 2013 IEEE Symposium on*, pp. 1-8, 2013.

[7] Deigmoeller, J., Janssen, H., Fuchs, O., & Eggert, J. (2014). Monocular rear approach indicator for motorcycles. *VISAPP 2014 - Proceedings of the 9th International Conference on Computer Vision Theory and Applications*, *3*, 474–480.

[8] Song, K.-T. S. K.-T., Chen, C.-H. C. C.-H., & Huang, C.-H. C. H. C.-H. C. (2004). Design and experimental study of an ultrasonic sensor system for lateral collision avoidance at low speeds. *IEEE Intelligent Vehicles Symposium, 2004*, (3), 647–652.

[9] Continental Automotive System Inc, "Motorcycle and Helmet Providing Advance Driver Assistance", 13897570, 2013.

[10] P. Aloumanis and E. Aloumanis, "EMBEDDING INTELLIGENT ELECTRONICS WITHIN A MOTORCYCLE HELMET", 13/671646, 2016.

[11] Ginsburg B, Ramasubramanian K, Rao S, Singh J. "AWR1642 mmWave Sensor" Texas Instruments. 2017.

[12] G. Liu, L. Wang and S. Zou, "A radar-based blind spot detection and warning system for driver assistance," *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, Chongqing, 2017, pp. 2204-2208

[13] Texas Instruments, "AWR1642 Single-Chip 77- and 79-GHz FMCW Radar Sensor," AWR1642 datasheet,May 2017 [Revised April 2018].

[14] Digi Key, "A Designer's Guide to Lithium Battery Charging," Sept 2016

9. Appendix

FFT Code:

```
function [X]=DFT_nik(x)
%varying the degree of noise
NM = 0.1;
N = 10;
%creating a vector with size N to be used for the time of the sin wave
t = 1:N;
%setting the noise
noise = rand;
%50 Hz sin wave as the input with the noise
x=sin(pi*50*t) + NM * rand(1,N);
%analyzing the signal throughout the time range
for k = 1:N
 X(k)=0;
 for n = 1:N
 X(k)=X(k)+x(n).*exp(-1j.*2.*pi.*(n-1).*(k-1)./N);
 end
end
f=0:k-1;
subplot(311)
plot(x)
title('Sequence x (in time domain)')
xlabel('time')
ylabel('Amplitude')
grid;
subplot(323)
stem(f,abs(X))
title('Mag Fourier Coeff using function')
```

```
ylabel('|X|')
grid;
subplot(325)
stem(f,angle(X))
title('Angle of Fourier Coeff using function')
xlabel('Freq')
ylabel('<X')
grid;
subplot(324)
stem(f,abs(fft(x)))
title('Mag Fourier Coeff using Matlab fft function')
ylabel('|X|')
grid;
subplot(326)
stem(f,angle(fft(x)))
title('Angle of Fourier Coeff using Matlab fft function')
xlabel('Frquency coefficients')
ylabel('<X')
grid;
```

Sensor PWM Code:

```
/*
 * File:   newmainXC16.c
 * Author: cjw88
 *
 * Created on April 10, 2019, 1:21 PM
 */


#include "xc.h"
void ms_delay(int ms) {
    TMR1 = 0;
    T1CONbits.TON = 1;
   while (TMR1 < 250 * ms) {
    }
```

```
    T1CONbits.TON = 0;
    TMR1 = 0;
}

int main(void) {
    int a;
    int b;
    int c;
    //int enable1;
    //int enable2;
    //int enable3;
    //int distancecm;
    PORTA = 0x000; //clear port A
    PORTB = 0x000; //clear port B
    PORTC = 0x000; //clear port B
    TRISAbits.TRISA1 = 0;
    TRISAbits.TRISA2 = 1;
    TRISAbits.TRISA3 = 0;
    TRISAbits.TRISA4 = 0;
    TRISAbits.TRISA5 = 1;
    TRISAbits.TRISA6 = 0;
    TRISAbits.TRISA7 = 0;
    TRISAbits.TRISA0 = 1;
    TRISAbits.TRISA10 = 0;
    TRISBbits.TRISB1 = 0;


    T1CON = 0x10;

    //ms_delay(100);
   //PORTAbits.RA1 = 0;

    while (1){

        //Sensor 1
         TMR1 = 0;
        PORTAbits.RA4 = 0;
        while (PORTAbits.RA5 == 0){}
            T1CONbits.TON = 1;
          while (PORTAbits.RA5 > 0){}
        T1CONbits.TON = 0;
        a = TMR1;
        a = a/2.5;
        if (a < 30){
            //enable1 = 1;
            PORTBbits.RB1 = 1;
```

```
}
else{
   //enable1 = 0;
   PORTBbits.RB1 = 0;
}
PORTAbits.RA4 = 1;
ms_delay(100);


//Sensor 2
TMR1 = 0;
PORTAbits.RA1 = 0;
while (PORTAbits.RA2 == 0){}
   T1CONbits.TON = 1;
 while (PORTAbits.RA2 > 0){}
T1CONbits.TON = 0;
b = TMR1;
b = b/2.5;
if (b < 20){
PORTAbits.RA3 = 1;
}
else{
PORTAbits.RA3 = 0;
}
//PORTAbits.RA1 = 1;
ms_delay(100);

//Sensor 3
TMR1 = 0;
PORTAbits.RA7 = 0;
//PORTAbits.RA10 = PORTAbits.RA0;
while (PORTAbits.RA0 == 0){}
  T1CONbits.TON = 1;
 while (PORTAbits.RA0 > 0){}
T1CONbits.TON = 0;
c = TMR1;
c = c/2.5;
if (c < 20){
PORTAbits.RA10 = 1;
}
else{
PORTAbits.RA10 = 0;
}
PORTAbits.RA7 = 1;

ms_delay(100);
```

```
    }
    return 0;
}
```

Bluetooth Module Scrips:

Write Script:

```
@PW_ON
C,0,D88039F8D8DA
@CONN
SM,1,000E
@TMR1
CI
SM,3,0040
@TMR3
$VAR1 = |I,08
CHW,0072,$VAR1
SM,2,0040
@TMR2
$VAR2 = |I,10
CHW,0074,$VAR2
SM,3,0040
```

Read Script:

```
@PW_ON
SM,2,0040
@TMR2
|O,08,%0072
SM,3,0040
@TMR3
|O,10,%0074
```

SM,2,0040

Data Sheets

NE555 - http://www.ti.com/lit/ds/symlink/ne555.pdf

https://datasheets.maximintegrated.com/en/ds/MAX8900A-MAX8900B.pdf

https://www.digikey.com/product-detail/en/microchip-technology/USB3317C-CP-TR/638-1092-1-ND/2166044

https://www.digikey.com/product-detail/en/molex-llc/0482580001/WM6294CT-ND/2804764

CC2650MODA - http://www.ti.com/lit/ds/symlink/cc2650moda.pdf