Head On

Fall 2015
10 December 2015

Group 12:
Amber Farrell, Cp.E.
Larry Herman, E.E.
Cristhian Marin, Cp.E.
Aurora Reinefeld, E.E.

Table of Contents

List of Figures

List of Tables

1.0 Executive Summary

It is an undeniable fact electrical and computer engineers have made the greatest contribution to the current technology sector. Their impact has been revolutionary and has paved the way for products that have transformed the quality of life for many. As the members of this team near the beginning of their electrical and computer engineering careers, this group wanted to create a senior design project that focuses on taking existing technology and improving its current use for the users.

In particular, we are focusing our design on the motorcycle community. The main objective for this project is to increase the riding experience of a rider by increasing his or her situational awareness and, consequently, safety. The design of this system consists of two parts. The first part incorporates a typical full face motorcycle helmet, which is then modified to contain a display. This screen will display relevant information to the user about his or her surroundings, motorcycle, and other general information. The second unit of Head On will be fitted on the motorcycle itself. This unit will contain several modules that will be used to take in data; once all of the data is gathered, it will then be transmitted to the helmet system so that it can be analyzed and displayed to the user. The combination of these two units will provide the rider with features that increase his or her enjoyment, safety, and secureness in the motorcycle.

This document describes design decisions made and the factors that lead to these decisions. First the goals, requirements, and specifications for this project will be explained. Next, a detailed description of each component of this project will be discussed. Afterwards the standards and constraints that impact the design of Head On will be analyzed and followed by an explanation of the design of the system as well as how the hardware and software of the prototype will interact. Finally, the testing procedure of the prototype will be discussed.

2.0 Project Description

One of the first steps taken when creating Head On was defining what the project would be. First a motivation for the project was determined. This helped to define what the problem was that the group wanted to solve. Based off of this motivation, goals and objectives were created. The purpose of these goals and objectives was to create a more specific list of features for the project that would help solve the problem. Finally, requirements and specifications were determined based off of these objectives. Doing this allowed the group to have concrete values to build the project off of.

2.1 Motivation

In recent years, vehicles have incorporated sophisticated electronic systems in order to improve the safety, comfort, and overall enjoyment of the ride. Notable features seen in newer vehicles include built in navigation, projected speed, directions displayed on the windshield, omnidirectional cameras for a 360 view of the vehicle, and proximity sensors to detect nearby obstacles. However, there is a noticeable absence of this kind of technology for motorcycles. While there are a few products currently on the market that aim to fill this void, they are often expensive and many do not focus on increasing the rider’s awareness of his or her surroundings. Consequently, this group decided to create the Head On system as a means to fill this void. Ideally, using this motorcycle and helmet system will to give riders easy access to data such as the proximity of potentially hazardous objects in his or her blind spots, details of the bike (e.g. cardinal direction, speed), and general details (e.g. time). Therefore, when moving forward with the design process, the primary focus will be on both the rider’s experience and safety.

2.2 Goals and Objectives

Once the motivations of the project were determined, the next step was to discuss the goals and objectives. At this stage in the design process, these statements are fairly vague descriptions of what the project should include. They will be modified in later stages of this project as the design of the system changes. Eventually, the final list of features for Head On will be developed from these goals and objectives.

One objective of this project is to allow easy charging. In order to do this, Head On will have dual charging methods so that the user can charge the system both at home and on the go. The first of these methods is a built in battery. This battery will be easy to remove and charge. This method will allow the system to be charged when at home. The second charging method will utilize solar power. Solar panels will be mounted onto the helmet in order to allow the system to charge while the rider is using it. By having two charging methods, the user will be able to use the system for a longer period of time.

Another integral goal for this project is to make sure the helmet enhances, rather than hinders the user's driving experience. In order to do this, several factors need to be considered in the design. First, the system itself should be small so that it can fit easily on the back of the helmet. This size is imperative because it must not break any safety standards for motorcycle helmets. Next, the size and location of the screen must be chosen specifically so that they are not obstructive to the user. The user’s field of view must not be seriously impacted because this would make the system counterproductive. Furthermore, the GUI that is displayed on the screen must be designed so that it is easy to read and simple to understand. The screen should not be too cluttered, and the symbols should be easy to understand. Finally, the sensors on the bike must also be placed in locations that will not be obtrusive.

Finally, the most important goal of this project is to increase the rider's safety. This will be done in two ways. The first is the system will strengthen the rider's awareness of his or her surroundings. Sensors are used to detect objects in the rider's blind spot. When an object is detected, a notification will appear on the screen indicating if the hazard is in the left, right, or rear of the rider. The second part of this goal is that the Head On system will make the rider safer by eliminating the need for the rider to look down at his or dashboard. In addition, to the notifications when there are hazards nearby, other useful information will also be displayed on the screen such as time, cardinal direction, and speed.

2.3 Requirements and Specifications

Once the goals of the project were defined, the requirements and specifications also had to be determined. These requirements and specifications add numerical values to the goals and objectives created in the previous step of the design process. Similar to the goals and objective, these values will be changed throughout the design process based on research, test results, and design modifications. In order for this project to be considered a success, these values, summarized in Table 1, must be met by the prototype. These requirements and specifications impact the hardware, software, and overall performance of the system. These requirements and specifications have been split into two categories: module specifications and power specifications.

First it is important to focus on the requirements of the modules in the system. This system should include three sensors in order to detect objects to the left, right, and rear of the user. These three values are the main blind spots of the user. The detection distance was chosen in order to ensure that the sensors can adequately reach into adjacent lanes to detect objects. This decision was based off of standard road widths in the United States. For the display, the main concerns are the size and the quality of the screen. A size of 2.2" was chosen because after some sample sketches for the layout, it was determined that all of the indicators can fit without cluttering the screen. The next component of the system that was analyzed was the microcontroller. The primary specification for this is the number of pins available. After considering the modules necessary for the system, it was determined that 20 pins would be sufficient. This amount would also leave some pins for future improvements to the design without being considered excessive. Next the wireless communication must be considered. The helmet and motorcycle units should be able to communicate with each other when the rider is near his motorcycle. Ten meters should be enough for this. It will allow the user to turn on the helmet system and have the modules connect to each other as he or she approaches the system. Next, the real time clock module should be very accurate in order to ensure that the time displayed for the user is correct. A strict accuracy of +/- 2 minutes has been chosen. Similarly, the electronic compass should be precise. Finally, the GPS should be able to start fairly quickly in order to ensure that the system can start using it immediately.

Table 1: Module Requirements and Specifications

Attribute	**Value**
Number of sensors	3
Sensor detection distance	2-400cm (1in - 13 ft)
Sensor measuring angle	30 degrees
Sensor frequency	40kHz
Sensor Resolution	0.3 cm
Display size	2.2”
Display resolution	320 x 240 resolution, 18 bit color
Microcontroller pin count	20 (14 digital + 6 analog)
Microcontroller current consumption	0.1uA-0.2mA
Wireless communication module frequency	2.4 GHz
Wireless communication module operating range	10 meters
Wireless communication module data throughput	0.7-2.1 Mbps
Real time clock accuracy	+/- 2 minutes/year
Electronic compass magnetic field range	+/-12 gauss
Electronic compass magnetic sensitivity	0.08 mgauss
Electronic compass acceleration range	+/-16 g
Electronic compass acceleration sensitivity	0.061 mg
GPS Time to first acquisition (cold start)	35 seconds
GPS Time to first acquisition (hot start)	1 second
GPS position accuracy	2.5 m
GPS velocity accuracy	0.05 m/s

Next, the power requirements for this system must be considered. This motorcycle unit will be powered by the battery on the motorcycle, so sufficient power will always be available. Consequently, the power consumption of the GPS module, Bluetooth module, and the three proximity sensors are not as much of a concern because they will all be placed on the motorcycle unit. The power specifications in Table 2 below focus on the power needs of the helmet system. Factors taken into consideration include the power needs of the modules as well as the power available from the battery and solar panels. The necessary battery voltage and capacity were decided after many calculations on what the power requirements for the modules will be. These values impact the battery and solar panel specifications. For the solar power charging, the values given Table 2 are assuming ideal conditions in which sufficient sun is available.

Another factor taken into consideration when determining the power specifications is the battery life and time to charge. The time to charge the battery varies depending on the method of charging. Specifically, charging the battery with the DCV charger should be much faster than the solar panel method. For this project it was determined that the battery life needs to be sufficient so that the rider can use the system for hours without having to worry about charging it. Consequently, a ten hour battery life was chosen so that the user will have more than enough power to use the system. Additionally, this battery life will also allow the rider to go on several shorter rides without having to charge the system in between. If sufficient sun is available for the solar panels, the system will be able to charge even while in use. Therefore, the battery life of the system will be even greater.

Additional factors that must be considered include the dimensions and measurements of the components. For example, because of the limited space for the solar panels and their location on the helmet, the size and weight must be limited. The dimensions of the panels must allow them to fit securely to the helmet in order to meet certain helmet standards. After testing various sizes on the helmet it was determined that 73mm x 94mm would be small enough to mold closely to the helmet. Based on the number of solar panels necessary, this size is also sufficient to allow all of the solar panels to fit on the top of the helmet, where they will be exposed to the most sun. The weight should not be noticeable since they will be placed on top of the use’s helmet. Therefore, it was determined that the weight of the solar panels should be 1.9g.

Table 2: Power Requirements and Specifications

Attribute	**Value**
Solar panel power	0.225 W
Solar panel output	55mA at 4.8V
Solar panel dimensions	73mm x 94mm
Solar panel weight	1.9 g
Battery capacity	2000 mAh
Battery voltage	3.7 V
Battery life	10 hours
Battery recharge time (solar)	8 hours
Battery recharge time (DCV charger)	4 hours
Sensor power consumption	75 mW
Display current consumption	100mA
Display voltage	3.3-5V
Real time clock operating voltage	3.3 V
Real time clock current consumption	110 uA (standby); 200uA (active)
Electronic compass operating voltage	2.5 V
Electronic compass current consumption	1uA (standby); 300uA (active)

3.0 Research

Before designing the software and hardware components for Head On, a great of research had to be done into every feature of it. This included doing a market analysis of similar products, researching different approaches that could be taken, and researching specific modules that could be used.

3.1 Similar Projects and Products

Research into similar products was conducted in order to determine what needs have not been met in the market. These similar products were also taken into consideration. By researching these products, it was determined what approaches would be best to avoid, as well as what approaches were more likely to result in success. This included the incorporation of certain features, the choice of components, and the manufacturing of the product.

3.1.1 Skully AR-1

Currently available for preorder is a similar product named “Skully AR--1.” At a cost of $1499, consumers can purchase a DOT/ECE certified helmet which provides motorcyclists with many useful features for on road use. Specifically, the Skully AR-1 provides riders with a transparent heads up display projected onto the visor, rear view display, full 180 degree field of view, GPS navigation, and Bluetooth connectivity. According to their website, this heads up system will allow the rider to “gain full situational awareness”. The Head On system will have a similar premise by helping increase the situational awareness of the rider through the use of proximity sensors. However, one major difference in the design of Head On is that rather than showing the rider an image of the road behind him or her, this system will simply display a symbol on the screen indicating what direction the potential hazards are coming from.

3.1.2 Helmet Tracking System

Another similar product is a previous senior design project, named “Helmet Tracking System”. This project tracks the rider using GPS and sends out texts if he or she was in an accident. This project focuses more on reacting to accidents rather than preventing the user from having them. Because the system only takes action in the event of an accident, there is no integrated display for the rider to interact with the system. Head On will focus more on the prevention of accidents. It will include a display positioned at the front of the helmet which will allow the rider to be warned visually of the proximity of other vehicles; consequently, the rider will become more aware of his or her surroundings. In a further effort to increase the rider’s safety, the Head On system will also eliminate the need for the rider to look at his or her dashboard by displaying other useful information such as, speed, cardinal direction, and time. Hence, increasing the overall awareness of the bike and surroundings, and increasing the rider’s safety. Incorporating an accident notification system, similar to the Helmet Tracking System, was considered but due to time and budget constraints it has become a stretch goal that may be revisited in the future.

3.1.3 NUVIZ

The NUVIZ heads up display system does not use the typical built in design in. Rather, this system is designed as a removable attachment which can be mounted to the rider’s existing helmet. This system includes a variety of features including access to maps and weather forecasts. It also allows the user to make and receive phone calls more safely by displaying the relevant information on the user’s helmet. Furthermore, this device also seeks to enhance the user’s riding experience by allowing him or her to play music from his or her personal music library. Basic information such as speed, time, and distance traveled are also displayed for the user. Finally, this system also gives the user the option to take pictures and videos while riding.

However, while this system offers many features that can enhance the riding experience, it fails to include features that will increase the rider’s awareness of his or her surroundings. The Head On system plans to satisfy this need by incorporating sensors that detect potential hazards in the rider’s blind spots.

3.1.4 LiveMap

LiveMap is another heads up display system that is attempting to enter the market place. The main focus of this system is hands free navigation. This system displays a map onto the visor of the helmet. While providing this assisted navigation certainly can be beneficial to the user, this system does not increase the user’s situational awareness. Additionally, the current estimated price for this system is very high, which may be unappealing to many riders. Not only will Head On cost less, it will also include more features which will not only improve the user experience, but also increase his or her awareness.

3.1.5 Reevu

The next motorcycle helmet system being analyzed is the Reevu system. This system allows the rider to view what is behind him or her, essentially acting as a rear view mirror. While this can improve the awareness of the user, there are several flaws to note. First, the system is located above the visor of the helmet, so the user must temporarily take his or her eyes off the road in order to look up at it. This is inconvenient and awkward for the user. The main point of heads up displays is to allow the user to keep his eyes looking forward. Undoubtedly looking up is less dangerous than having to look back to check for other vehicles, however, the placement of the screen is still inconvenient. In addition, the system does not offer any indication of potential hazards on the left and right of the driver. These are major blind spots which are unfortunately ignored in this system. The Head On system attempts to address the aspects left out in this design. First, the screen will be located outside of the helmet in a location that is convenient and easy to see. In addition, it will use three sensors to detect hazards to the left, right, and rear of the user, thus creating a more complete situational awareness and addressing the problem of blind spots. Finally, rather than displaying images of these blind spots, Head On will have simple indicators that are displayed when a hazard is detected. This will minimize distractions.

3.1.6 BikeHUD

Finally, the last product that was researched was BikeHUD. According to their website, this device attaches inside of the helmet. It displays information such as speed and navigation. To manually switch between views of displayed information, the system uses a series of buttons that attaches to the handlebar of the bike. However, BikeHUD, like many of the other heads up display systems discussed previously, fails to notify the user of potential hazards. In addition, because the display is located inside of the helmet, it may be difficult to read what is displayed on the screen. Once again, the Head On system will not only address the issue of notifying the user of potential hazards, the display will also be placed in a more convenient location outside of the helmet.

3.2 Sensor Research

On the road, obstacles can present themselves at any given time without warning. During the basic rider's course, motorcyclists are taught to pay attention and analyze their current surroundings for pedestrians, pets, debris, and even cars entering their lane unexpectedly. Being able to predict movement of such object can be a lifesaving skill; however, detection of these obstacles mainly applies to front facing objects, thus leaving the rider vulnerable by the limited helmet view. In order to increase the rider’s situational awareness and cover his or her blind spots, the Head On system needed to be equipped with a detection system. Primarily, the detection system would need to cover the areas a rider cannot immediately access without turning his or her head: the rear, back-left, and back-right.

Proximity sensors by definition are able to detect objects without any physical contact. These sensors feature variations in terms of maximum detection distance or ‘nominal range’ and the method used to detect a specific target. Furthermore, appropriate selection of sensor types are needed as different target materials demand different sensors. According to the American Association of State Highway and Transportation Officials, lanes must be a minimum of twelve feet wide. Consequently, when determining the requirements and specifications for this project, it was determined that the sensor should reach at least 13 feet. Assuming an ideal case in which the user is in the exact center of his or her lane, this detection distance would reach far enough into the adjacent lane that it could detect the presence of another vehicle

Several different methods of proximity detection are currently being used by the automotive industry. This includes infrared, radar, and ultrasonic detection. Infrared detection systems are only effective to the low end of the specified detection distance requirement and intense ambient lighting can further reduce the effective range. In addition, false indications can also be triggered by external light sources such as vehicle headlights. On the other hand, radar detection systems have too long of a detection distance. While they would work well for forward or rear facing detection, objects off the side of the roadway could generate false side proximity warnings. In addition, the close proximity of the sensors to one another could result mutual interference between them. Fortunately, the ultrasonic detection systems were found to have an ideal detection range for the system’s side proximity detection, as well as an adequate range for the rear proximity detection. The frequency and pulse modulation scheme ultrasonic sensors is used minimize the possibility of external interference, while the highly directional nature of the sensors prevents mutual interference of properly positioned sensors. Unfortunately, none of the sensors that are designed for use in unprotected environments were available within the budget constraints of this project. Therefore, it was decided that ultrasonic sensors would be the best choice for this project. The two models of ultrasonic sensors that met the preliminary size and cost criteria were researched in-depth in order to determine which would be the most appropriate for the Head On system.

3.2.1 HC-SR04 Ultrasonic Sensor

The HC-SR04 is a very popular ultrasonic sensor that is commonly used for robotic distance sensing. HC-SR04 is a generic part number made by multiple manufacturers. The design, as shown in Figure 1, is consistent regardless of what manufacturer it is made by, but there is a variety of documentation available from third party vendors. Due to the variety of vendors, the cost of this sensor varies. However, all of these prices are fairly low and usually are around $2.50 per sensor. Taking into consideration the budget constraint of this project and the fact that three sensors are necessary for the system, this price is fairly reasonable and falls in line with the budget.

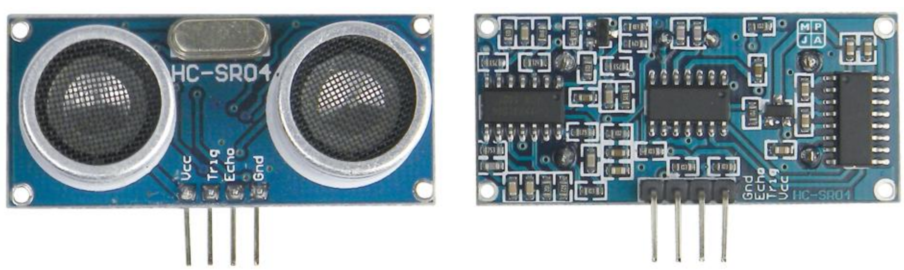


Figure 1: HC-SR04 Ultrasonic Sensor
Requesting permission in progress

Table 3 below summarizes the specifications of the HC-SR04 ultrasonic sensor. It has a small footprint and low power consumption of only 75mW. It has a 13 foot detection range and a 30 degree measuring angle which fits the project specifications perfectly. Although this range is not as great as some of the other ultrasonic sensors available on the market, it is sufficient enough to sense objects that are in adjacent lanes. Additionally, the dimensions of the sensor are fairly small which will make finding somewhere on the motorcycle to place them much easier.

Table 3: HC-SR04 Specs

Operating Voltage	+5 V
Power	75 mW
Current	15 mA
Frequency	40 kHz
Max Scan Rate	20 Hz (estimated)
Detection Distance	2 – 400 cm (1 in – 13 ft)
Resolution	0.3 cm
Measuring Angle	30 degrees
Width	45 mm
Height	20 mm
Depth	15 mm

A 10 us TTL pulse triggers the sensor to transmit a series of eight 40 kHz pulses. The sensor generates a TTL high on the output pin until the rising edge of the reflected pulse train is detected. Figure 2 illustrates this process. The pulse width corresponds to the measured distance. The distance of the object can be calculated by dividing the width of the output pulse in microseconds by 58 for centimeters or by 148 for inches. The maximum time to complete a measurement appears to be approximately 38.5 ms, but due to inconsistencies in the timing diagrams, a maximum cycle rate of 50 ms would be prudent. However, even this maximum cycle time is still acceptable for this project.

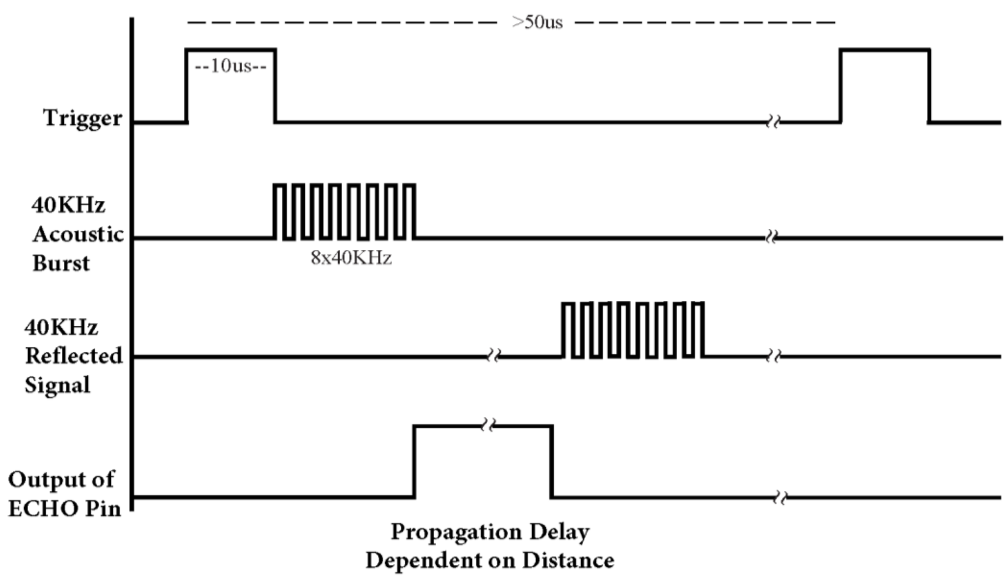


Figure 2: HC-SR04 Timing Diagram
Requesting Permissions in Progress

This sensor requires only four connections to operate. This is a fairly small number of pins and there should be more than enough available on the microcontroller. Three of these can be shared between multiple sensors resulting in only one dedicated line per sensor required at the controller. Because there will be three sensors used as part of the motorcycle system this is significant; this will reduce the pin count, thereby leaving more pins available for future expansions of the project.

The software aspect of this project must also be considered when choosing a sensor. Some members of this group already have experience programming this type of sensor. Therefore, if this sensor is chosen, it will allow more time to be spent on programming the other modules. Additionally, due to its popularity with Arduino users, there is a large amount of open source code which can be adapted to this project. This will be beneficial for the group members who do not have experience with this sensor. Finally, because of its popularity, there are also a large number of forums on line in which common problems with these sensors are discussed. This will decrease the amount of time that is spent on troubleshooting problems with the hardware and software aspects of the sensor system.

3.2.2 LV-MaxSonar-EZ MB10x0 Ultrasonic Sensors

The LV-MaxSonar-EZ Series is versatile and available in several models, each with a slightly different beam pattern based on the size, range, and sensitivity of the detection application. It is able to function on a range of supply voltages, 2.5 to 5 V, and has a very low power consumption of 5 - 15 mW. While ordinarily the power consumption should be kept as low as possible, it is not as significant for this module because it will be placed on the motorcycle, and therefore will have ample power available.

In addition to having a triggered detection cycle, this sensor can be set to free run, which causes it to continuously provide range data without external signals. The measured range data is provided in three formats: serial, analog, and pulse width, which are all simultaneously available. The serial output is in an asynchronous RS232 format at a baud rate of 9600 bpm with eight data bits, no parity bit and 1 stop bit. The sensor transmits an ASCII “R” three-digit distance and a carriage return for each measurement cycle. The analog output provides a DC voltage level between 0 volts and VCC in 512 increments proportional to the measured distance. The sensitivity is 9.8 mV per inch for a 5 volt VCC and 6.4 mV per inch for a 3.3 volt VCC. The pulse width output produces an output pulse width proportional to the measured distance with a sensitivity of 147 us per inch. Table 4 summarizes these specifications of the MB10x0.

Table 4: MB10x0 Specs

Operating Voltage	+2.5 - 5 V
Power	5 - 15 mW
Current	2 - 3 mA
Frequency	42 kHz
Max Scan Rate	20 Hz
Detection Distance	0 - 6.45 m (0 - 21.2 ft)
Resolution	1 in
Measuring Angle	variable
Width	19.9 mm
Height	22.1 mm
Depth	15.5 mm

One important factor to note is the size of this sensor. While the HC-SR04 was a small sensor, this sensor, as depicted in Figure 3, is even smaller. As previously mentioned, choosing a sensor that is small in size will make the process of placing it much simpler. The sensors will be the bulkiest part of the motorcycle unit. Ideally, the unit should be as small as possible. The simplest way to ensure this is to utilize small sensors.

Figure 3: LV-MaxSonar-EZ Series
Requesting Permissions in Progress

The sensors can be connected, triggered, and read independently, however there are also several options for connecting multiple sensors using fewer connections and requiring less processor interaction. Reducing the number of pins is desirable for two reasons. The first is that the microcontroller only has a limited number of pins available. Insufficient amounts of pins would result in either some modules being excluded from the system or the microcontroller, and consequently the design of the entire system, to be changed. The second benefit for reducing the number of pins is that more expansions could be added to the system in the future. The entire array can be triggered at the same time providing simultaneous range data. This requires only one trigger signal connection, but an individual data connection for each sensor. While it results in faster detection rates, it also increases the potential for interference between the sensors in the array depending on their position.

In the next two configurations, each sensor triggers the next one at the end of its measurement cycle. The difference between these configurations is in the external trigger signal. One configuration requires an external trigger signal for each cycle through the array. However, in the other configuration the trigger signal from the final sensor is fed back to the first sensor in the array resulting in continuous operation of the array until power is removed from the sensors. These configurations also only require a single trigger connection but could also be connected to a single data port as only one will sensor will be transmitting at a time. This eliminates the potential for interference between the sensors, but decreases the detection speed as the array must step through each sensor's measurement cycle in turn before triggering a new series of measurements. Figure 4 shows the timing diagram for the MB10x0 series sensors.

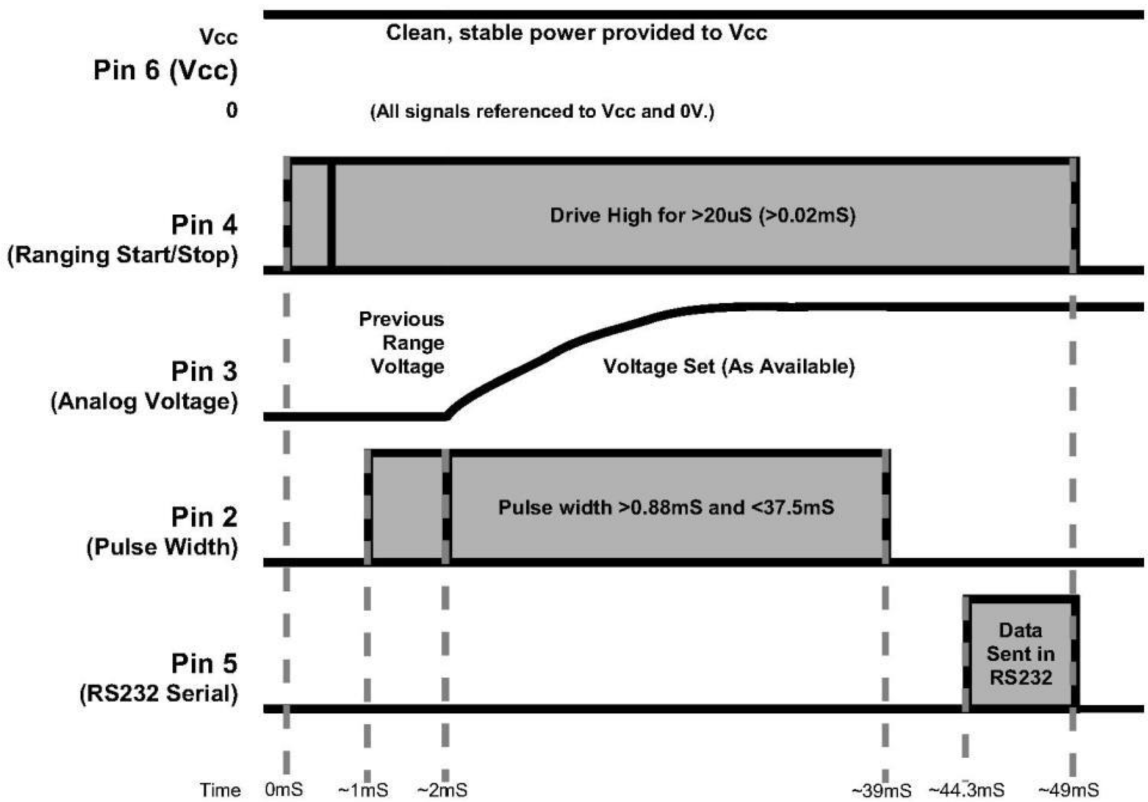


Figure 4: Timing Diagram
Requesting Permissions in Progress

The serial output is not ideal when employing multiple sensors as it would require multiple RS232 connections to the controller or additional timing and multiplexing hardware. There are several accessories available to combat electromagnetic interference and provide temperature compensation if necessary.

The various beam patterns for the MB1000 sensor running at 5.0 and 3.3V is show below in Figure 5. The accuracy of the sensor was tested using various object sizes ranging from 0.25" to 11". One important thing to note on these results is that when the sensor is running at 5V, the width and distance of the detection range is slightly larger. Therefore, if this sensor is chosen, this is most likely the operating voltage that the sensor would be set to. The selection of the beam footprint and variable gain make it ideal because the detection zone and sensitivity can be tailored to the riding environment. On the highway speeds are higher and areas are open; a longer range and higher sensitivity will increase the safety margin. While on smaller streets, a shorter range and lower sensitivity will reduce false indications. The variety of sensor configurations available will allow for modifications late in the design process should interference between the sensors prove to be a problem with the proposed sensor placement.

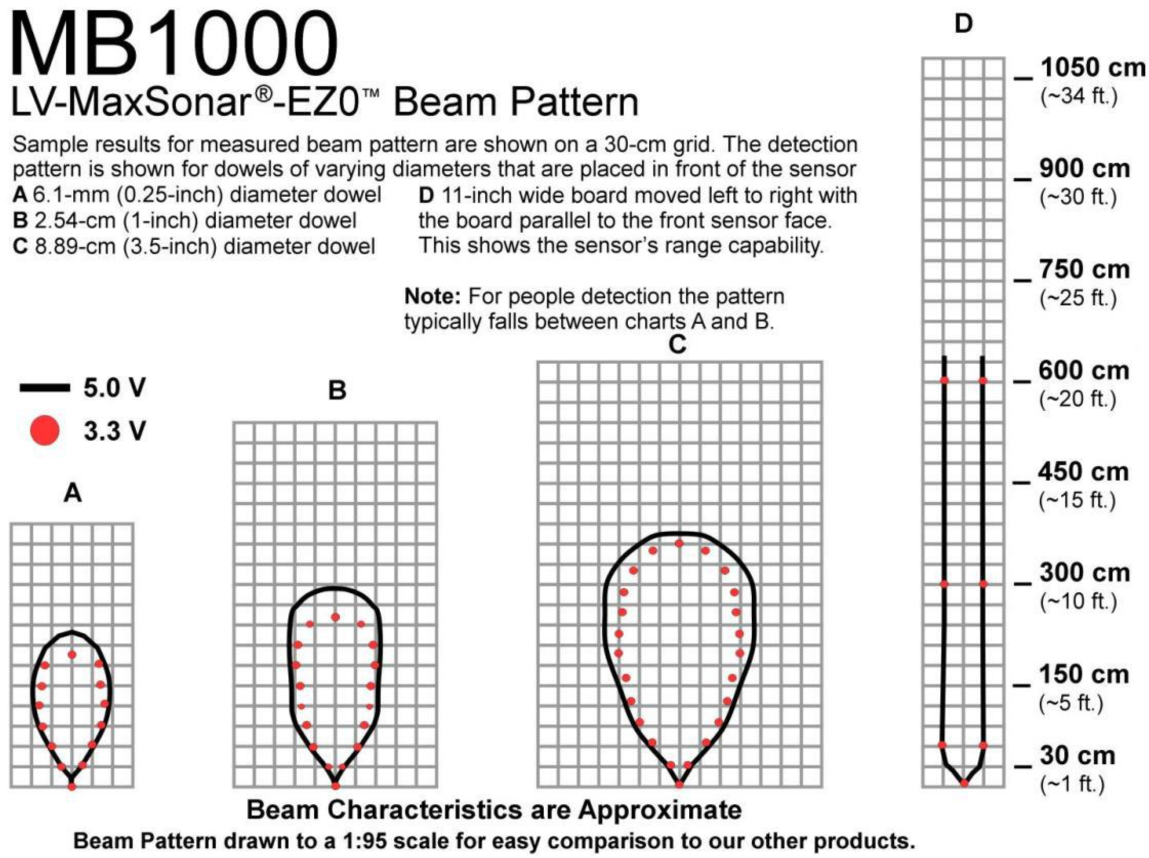


Figure 5: Detection Pattern
Requesting Permissions in Progress

The final factor considered with this sensor is the cost. Because of the increased range and improved power consumption, it is only logical that this sensor would cost more than its less advanced counterparts. On average, one MB1000 ultrasonic sensor costs approximately $25. This price is drastically higher than that of the HC-SR04. This project requires three sensors, so at this price, purchasing these sensors would expend a large portion of the funding for this project. This is a major concern because there are many other modules that must be purchased as well as manufacturing costs to consider.

3.2.3 Sensor Selection

After analyzing the requirements of this project, it was determined that the HC-SR04 ultrasonic sensor is best suited for this application. The large amount of source code available made the sensors more favorable. In addition, since some members of this group already have experience programming this particular sensor, it seemed more intuitive to choose this sensor. Although the range of the HC-SR04 is less than that of the MB1000, it is sufficient enough to detect objects in adjacent lanes. Finally, the most drastic difference between these two sensors, the price, must be considered. The price of the HC-SR04 sensor is significantly less than the sensors in the LV-MaxSonar-EZ Series, which will help the group to remain within the budget. In addition, because the HC-SR04 sensor is offered by several manufacturers, it will allow the group to find the best possible price for the sensor. Certainly, the features of the LV-MaxSonar-EZ Series sensors are greater, but this just cannot justify spending such a large portion of the funding on them, when the significantly less expensive HC-SR04 sensors provide all the necessary qualities for this project.

3.3 Screen Research

When using Head On, the user will receive all information through the display mounted to the front of the helmet. As a result, it is pivotal that the screen used is chosen carefully. The main factor that had to be considered was the size of the screen. The screen must not be so large that it obstructs the user’s view; however, it must also not be too small because all of the indicators must fit on it without being too cluttered. For each of the screens, a piece of paper that size was cut out and the indicator symbols were drawn on it. This paper was then placed in front of a motorcycle helmet, in the same location it will be in the prototype, in order to determine if it was too large. Ideally, a perfect balance must be found.

3.3.1 SparkFun Serial Enabled LCD Kit

The serial enabled LCD kit from SparkFun was one of the first screens taken into consideration when designing this system. One of the features that first made this display appealing is the availability of the built in backlight. This brightness of this backlight can be adjusted. This is important for this project because it will allow various lighting settings to be tested in order to find the optimal setting. This LCD display uses white characters on a black background. This color schematic cannot be changed; however, the high contract between the black and white would make the text easier to read. At the moment, the design of the user interface is in only black and white. However, if this display is chosen it will completely eliminate the ability to change the color scheme or give the user the ability to customize it. In addition, this screen allows up to two lines of up to 16 characters each to be displayed. These characters can be standard ASCII characters, as well as certain special characters. This format is similar to that of a seven segment display; this is significant because all four members of this group have experience programming seven segment displays. The image in Figure 6 was used to plan the layout of the user interface with this screen. The image is not created to scale because the main concern was the placement of characters. In order to emphasize that this is a 16x2 display, the location for each character was displayed. This actual display is not designed like this. Luckily the proximity indicators were able to be created using one of the special ASCII characters. Overall, the layout of the display is awkward. Because of the restrictions on the placements of characters, the data is not evenly distributed on the screen. Additionally, there are no image representations for any of the values. Finally, the overall look of the display is somewhat unimpressive. It appears very simplistic and unappealing.

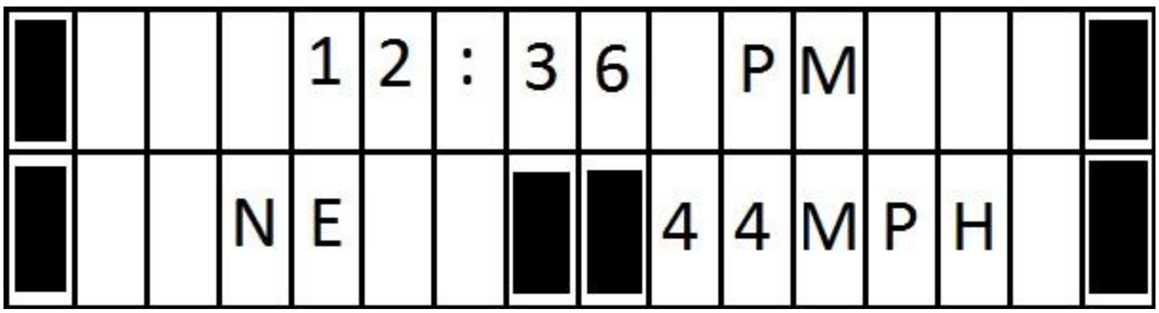


Figure 6: Segment Display Layout

For this project, the screen will display the time, speed, cardinal direction, and proximity indicators. Consequently, by limiting the information displayed on the screen to only two lines of sixteen characters, the display could be potentially cluttered, or a compromise might have to be made regarding what information is shown to the user. Furthermore, because this screen is limited to printing ASCII characters and a few special characters, the use of specially designed symbols to illustrate certain indicators is completely out of the question. Another area of concern with this screen is that it may be too wide. The placement of the screen would be difficult. Either the user would not being able to see all of the screen or the screen would block too much of the user’s field of view. However, picking this screen would also have some benefits. Its strict formatting would simplify the coding process. Additionally, the ASCII character codes would make printing the characters simpler. Finally, there is a large amount of source code available for this screen that can be used for reference.

3.3.2 Adafruit 1.8" Color TFT LCD display

The next screen that was taken into consideration was the 1.8” color TFT LCD display by Adafruit. One of the primary reasons this display was taken into consideration was the size of the screen. It is imperative that the screen is small enough that it is not obstructive to the user. As one of the smallest screens taken into consideration, it would definitely follow this constraint. However, it is also a concern that this size may result in the indicators being too crowded on the screen. This screen has a 128x160 resolution and 18 bit color. This would allow immense flexibility when designing the layout of the user interface. In addition, this screen also gives the option of using colors, which would allow the design of the user interface to be much more visually appealing. Unlike the SparkFun Serial Enabled LCD Kit, this screen does not restrict what characters can be displayed and where on the screen they can be located. Each pixel is drawn individually, which gives the layout and design of the user interface complete freedom.

Another benefit of this display is that it uses 4-wire SPI and it has its own pixel-addressable frame buffer. As a result, the display will be compatible with any microcontroller selection in the helmet unit. This will allow the microcontroller and display choices to remain independent of each other which will make the design process easier. Furthermore, there is a lot of example code that can be used as a reference when programming this display. Because none of the group members have programmed an LCD display this complicated before, this code will be a great stepping stone for determining how to program the user interface. Unfortunately, the complexity and the number of options available to customize the display mean that there will be a very steep learning curve for programming it. However, because there are so many possibilities with how the interface can be displayed, the code controlling the display will be much more complex. Additionally, while the small size of the screen is beneficial because it is nonobtrusive, it also has a drawback. The amount of information displayed on the screen may have to be limited, or else the screen may become too cluttered. This would make the display difficult to comprehend. Figure 7 below was created to illustrate this. This image was created with the same dimensions as the screen. Because the screen was so small, an extremely small font size had to be used. The large font is size 18, which for some perspective is the same size as the headings in this report, and the small font was size 10, which is even smaller than the text in this paragraph. After creating this figure, it became even more apparent that size would be a serious issue if this screen was used. Not only would the information be cramped in the screen, it would also be written extremely small, and there would be no room for future additions.

Figure 7: 1.8” Display Proposed Layout

3.3.3 Beaglebone Black Cape 4.3" LCD

The Beaglebone Black Cape LCD screen is 120.4 x 80.0 x 32.0mm. Without a doubt, the size of this display would allow for all of the information to be displayed on the screen without becoming cluttered. However, this size is also concerning because, as one of the largest screens considered, it may be too large for this project. Even with careful placement, it may block too much of the user's field of view which would decrease rather than increase the safety of the system; thus making the entire system redundant. In order to better visualize the dimensions of the display, the following diagram, in Figure 8, was created. This diagram was created to scale in order to match the 95mm wide by 53.9mm high dimensions of the screen. Clearly there is a significant amount of space available. To put this into perspective, the larger font is size 30 and the smaller font is size 16. In comparison the previous 1.8" screen considered, this screen is excessively large. Even with such large fonts used and the proximity indicators on the edges made very wide, there is still superfluous white space, thus making the layout seem somewhat awkward.

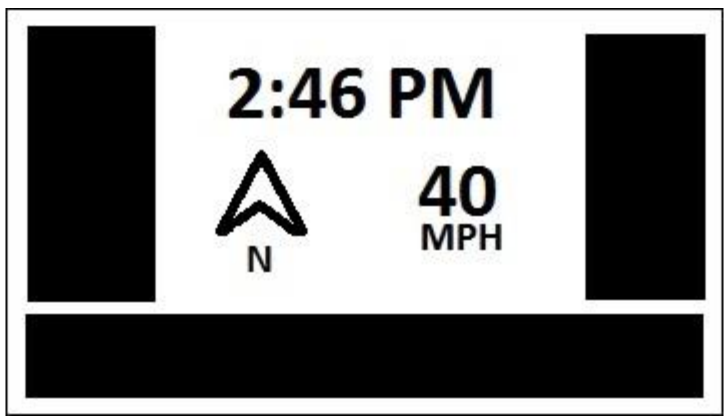


Figure 8: Beaglebone Black Cape Proposed Layout

However, this screen includes some features that could improve the design of the project. For example, this screen offers additional methods to interacting with it compared to the other screens. There are two separate methods of interaction. First, this display can be used as a touch screen. While there are no current features of the project that would require a touch screen, this option would open up many possibilities for future improvements especially if user customization is incorporated. Additionally, there are 7 buttons next to the screen that allow the user to navigate and interact with the user interface. Similar to the touch screen, there is not current need for this feature, but it does expand the number of possibilities for the design.

3.3.4 Adafruit 2.2" 18-bit color TFT LCD display

The final screen taken into consideration was the 2.2" 18-bit color TFT LCD display by Adafruit. The Adafruit 2.2" display measures at 55.23 x 40 x 2.46 mm. This screen falls in between the measurements of the Beaglebone Black Cape and the Adafruit 1.8" LCD display. Again, the project's goal is to utilize a display that is non-obtrusive to the user, yet clear and large enough for quick interpretation of the information displayed. Within these measurements, the LCD is packed with a 320 x 240 resolution and up to 18 bit color. These specs are not high resolution, but it does feature true TFT colors and improved refresh rates over commonly used screens like the Nokia 6110. By no means is the screen high resolution but it spacious and detailed enough for the various details the Head On system seeks to display. Similar to the 1.8" display, this display also has an abundance of source code available. Again, a mock layout, shown in Figure 9, was created in order to gain perspective on issues that might occur. This screen seems to be a happy medium between the 1.8" and 4.3" screens above. With size 24 and size 14 fonts, the display is not too cluttered. Additionally, there is room for additional indicators in case additional modules are added.

Figure 9: 2.2" Display Proposed Layout

3.3.5 Screen Selection

The major component taken into consideration when choosing the screen is the size. Because the screen is going to be mounted outside of the helmet, it must not be too large or else it will block the rider's view. However, the screen also cannot be too small or else it will be difficult to read. For the purpose of this project, the spark Fun Serial Enabled LCD Kit may be too wide. Either too much of the screen would not be visible to the rider, or it would cover too much of the user's field of vision. Similarly, the Beaglebone Black Cape 4.3" LCD also would be much too large. On the other hand, the Adafruit 2.2" and 1.8" 18-bit color TFT LCD displays are small enough that they would not affect the rider's ability to see.

The second major component taken into consideration is how the user interface will be designed. Because the spark Fun Serial Enabled LCD Kit uses the 16x2 character format, the placement of characters would be limited; this would result in either an overcrowded screen or certain features having to be removed due to lack of space. In addition, this screen would limit what could be drawn on it; only certain letters and special characters could be drawn rather than symbols or images. While the programming and design of the user interface would certainly be simpler if using this system, the color, character, and placement constraints are just too restricting for this project. On the other hand, Beaglebone Black Cape 4.3” LCD and Adafruit 2.2” and 1.8” 18-bit color TFT LCD displays would allow a much greater range of customization. Not only could the characters and symbols by placed anywhere on the screen, custom symbols could be designed and displayed to represent data such as speed, cardinal direction, and proximity indication.

After comparing the screens, it was decided that Adafruit’s 2.2” 18-bit color TFT LCD display would be the best choice for this project. This screen is small enough that it will not be obstructive for the rider, but still large enough that if more features were added to the project additional information could be added to the display. In addition, it will allow for more freedom in the design of the layout of the user interface by giving total freedom in the placement and design of the symbols used.

3.4 Microcontroller research

A part of developing the embedded systems that comprise the Head On project was to select central control unit that would handle data and connect to the various peripherals needed. In terms of embedded applications, four different types are available: microcontroller (MCU), digital signal processor (DSP), field programmable gate array (FPGA) and application-specific integrated circuit (ASIC). For the Head On project, only microcontrollers, digital signal processors, and field programmable gate arrays were considered.

The first embedded technology considered was microcontroller technology or MCU. As a part of the UCF curriculum, all Computer and Electrical Engineering students take part in courses that focus on microcontroller programming, thus each member of the group had prior experience with this technology. As with anything, pros and cons existed for implementing microcontrollers in the motorcycle and helmet units. The advantages for using microcontrollers consisted of the group's previous development knowledge, simple interfaces availability such as Texas Instrument’s Code Composer Studio, and efficiency when processing asynchronous tasks. In contrast, use of microcontrollers was disadvantageous due to the large effect of code on the efficiency of the system. In specific, clock cycle optimization is dependent on the code optimization, which affects performance and power consumption. In addition, the code footprint influences the count of transistors needed, which further affects power consumption. Furthermore, the microprocessors architecture design makes it unsuitable for significant streams of data processing such as video or graphic processing.

The next embedded technology considered was digital signal processor technology or DSP. This technology is commonplace in applications requiring rapid processing and feedback such as airplane landing gears and collision avoidance systems. In other words, DSP is the optimal choice for real time response systems because its architecture is designed to perform simple calculations incredibly fast. When it comes to digital signal processors, the advantages consist of simpler code than the code required for MCUs, hardware implementations of basic function, which results in optimized use of transistors and clock cycles for better performance and power consumption, and rapid processing of digital signals streams. On the other hand, DSP suffers from certain disadvantages such as reduced flexibility due to the hard wired basic functions and it falls on the higher end of the spectrum in difficulty when coding for a control portion of a control application. In addition to its use in real time response systems, DSP are suitable for audio and video oriented applications, as opposed to MCUs.

The last embedded technology considered was field programmable gate arrays or FPGAs. Field programmable gate arrays are unlike either of the other two technologies considered. Specifically, FPGAs are not processors at all, but instead these are a bulk of reprogrammable logic used to implement logic functions. There are two areas where FPGAs excel in: flexibility and speed. The former is achieved by its ability to be reprogrammed and do not have a fixed hardware resources such as those seen on PCB designs. The latter is achieved by its hardwired logic that is significantly higher than software based logic. While FPGAs impress in their ability to fit a user's design needs and provide speed, they suffer from a lack of clock cycle optimizations and high level of transistor redundancy which leads to far from optimal power consumption.

To summarize, each embedded technology considered provides the ability to process the data gathered from peripherals with its own set of advantages and disadvantages. The choice for which one to utilize came down to preference for this application as any selection was an engineering compromise. In this case, microcontroller was selected as the best suited processor for this application. It was chosen over its counterparts for its ability to handle asynchronous tasks, familiar interface, excellent performance in control applications (many real time control application perform multiple asynchronous tasks), low power consumption, and ability to do simple signal processing without the need for a dedicated DSP. To expand on this, DSP offers the benefits of fast signal processing useful for the heavy streams of data from peripherals such as the proximity sensors and GPS module; however, the data streams from these were estimated to remain on the lower end of the capabilities of DSPs ability. FPGAs offer flexibility and very fast speeds due to its hardware nature, yet suffered from high power consumption which goes against the low power consumption nature of Head On. Given this, it is important to note that either DSP or FPGA could have been used but preference, familiarity, and control application performance led to the selection of microcontrollers.

After selecting microcontrollers as the best suited technology for this application, the focus was shifted to selecting the best microcontroller for the job. The microcontroller will be the centralized unit that controls how the system handles data from peripherals, enables the helmet and motorcycle systems to operate together, and provides real time updates in the system using data. Hence, the decision of the microcontroller is vital to the design and making all planned components for the unit on the helmet motorcycle work properly together.

3.4.1 Texas Instrument’s MSP430

A requirement for electrical and computer engineers at the University of Central Florida is the course ‘Embedded Systems’ which involves a lab portion focused on programming the MSP430. As a result, one of the first considerations for a microcontroller was the Texas Instrument’s MSP430. Taking into account the group’s working knowledge with the microcontroller, inexpensive pricing for the unit, and versatility of the controller, this was a top choice.

3.4.1.1 MSP430 LaunchPad

To begin, the MSP430 LaunchPad kit is available online for as low as $9.99 directly from Texas Instrument’s and includes:

- MSP430G2553IN20 MPU
- MSP430G2452IN20 MPU
- 10-pin PCB Connector (2 male/2 female)
- 32kHz crystal
- LaunchPad Development board (MSP-EXP430G2)
- LaunchPad sticker
- Mini USB cable
- Quick Start Guide

It is apparent Texas Instrument’s wants their LaunchPad product used in many projects as this kit is probably sold at a loss to them with all the items included. The availability of the development board for the MSP430 is convenient for testing purposes while the PCB design is finalized. While the MSP430 may not have a high pin count as seen in Figure 10, it has become a recognized MCU by many hobbyists for its ultra-low power consumption. Specifically, data sheets for the MCU featured on the MSP430 LaunchPad state that at 1 MHz and 2.2 V, the MCU consumes 230 microamps, 0.5 microamps, and 0.1 microamps for active, standby, and off modes, respectively.

Consequently, it is clear the LaunchPad is suitable to remain within the power constraints of the Head On design whether it is used for the motorcycle mounted system, helmet mounted system, or both. However, even with the low power consumption, inexpensive cost, and powerful processing abilities, the pin count remains low; therefore it is difficult to consider this MCU to be implemented into the motorcycle's portion of the Head On system.

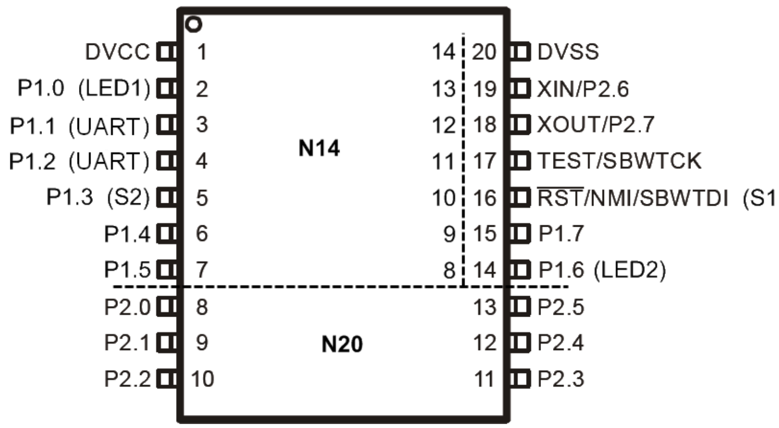


Figure 10: MSP430 LaunchPad Pins
Reprinted with Permission from Texas Instruments

3.4.1.2 MSP430FG4618

Although, the MSP430 is inexpensive, it is by no means a cheap microcontroller. The MSP430FG4618 variant features a respectable amount of flash memory at 116kB, 8kB of RAM. It provides multiple system clock options from its internal oscillator to high accuracy external crystals. All of which can be used to supply the on board real time clock. Furthermore, the controller offers 80 I/O pins, as, that allow versatility in communication with the Head On system's awareness components. The two multifunction serial port interfaces with multiple protocols such as UART, I^2C, and SPI. In addition to serial communication it includes a 12 bit analog to digital converter, two 12 bit digital to analog converters, and three programmable operational amplifiers. It also offers display options with a built in 160 segment LCD driver and regulated charge pump. All these features are powered by a 1.8V to 3.6V supply with a typical current draw of 400uA. Five power saving modes reduce the current draw down to 1.3uA in standby mode and require less than 6us to return to full operation. The MSP430FG4618 clearly has the power and pin count to perform all the functions of either of the Head On modules. Its small footprint and low power consumption also fit well within the space and power constraints of the project.

3.4.2 Atmel AVR Processors

Atmel offers a diverse line of AVR controllers that offer a wide range of memory and processing power to develop a suitable prototype. Furthermore, a variety of the chips that are available from Atmel provide the low power consumptions within the range required for this project design. The main draw to the Atmel AVR processors is their inclusion in the increasingly popular Arduino boards, which have with a fast growing community of support and projects. Two notable microcontrollers considered from Atmel were the ATmega328, as seen in the Arduino Uno, and the ATmega2560, as seen in the Arduino Mega.

3.4.2.1 ATmega328

To begin, the ATmega328 is a very popular microcontroller that appears in the Arduino Uno development board which can be attained at lost cost similar to Texas Instrument's Launchpad. The controller is priced at $3.99 from SparkFun with the Arduino bootloader already installed. This is a very low cost for a controller that is known to be powerful, reliable, and simple to work with the Arduino software. Specifically, if an Atmel processor is chosen, the project would require it to use an Arduino bootloader as the Arduino IDE since the availability of coding resources for the ATmega328 with this bootloader is substantial.

Furthermore, upon researching the ATmega328, it was discovered that it contains support for the Philips I2C protocol in addition to the Rx and Tx serial ports. This is highly beneficial because it allows flexibility to add any peripherals that may require an I2C port. There is a limitation of one I2C port on the ATmega328; however, the controller provides a 7 bit address space to allow up to 121 peripherals, assuming the addresses are all unique. Additionally, the controller features 14 digital pins (6 of which can be used for pulse width modulated output) and 6 analog pins as seen in Figure 11. Similar to the MSP430, this is a low number of pins especially since pins will be needed for the ultrasonic sensors, GPS module, and Bluetooth.

Next, the ATmega328 needs sufficiently low power in order to be implemented into the helmet unit. As mentioned earlier, the importance in power consumption comes into play in the helmet unit as the motorcycle will be used to power the system equipped onto it. According to the controller's data sheet, with the conditions of 1 MHz, 1.8 V, and 25 degrees Celsius, the ATmega328 consumes 0.2 milliamps, 0.1 microamps, and 0.75 microamps, in active, power down, and power save modes, respectively. The current prototype design plan consists of a rechargeable battery equipped to the helmet unit in conjunction with flexible solar panels to provide charge during daytime for the rider. Factoring in the power consumption values and the charging system, powering the microcontroller would not be an issue for the helmet unit.

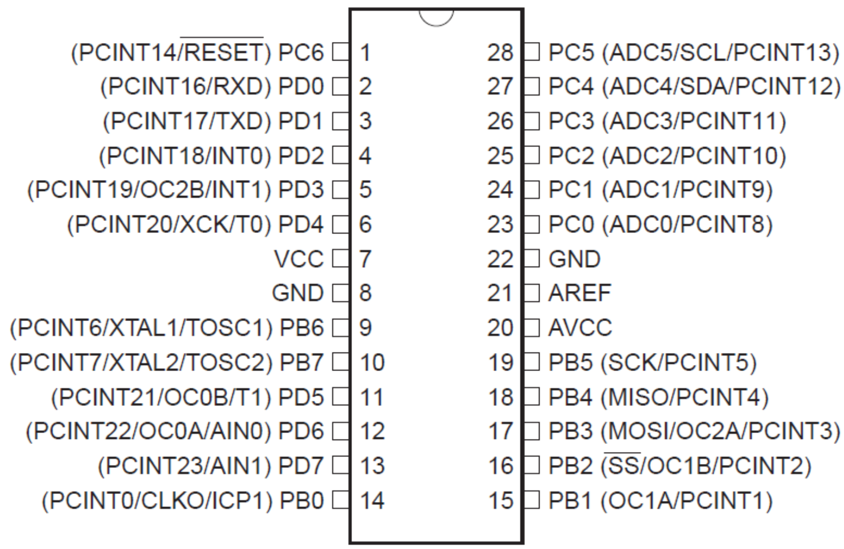


Figure 11: Pin mapping for ATmega168/328
Reprinted with permission from Atmel

3.4.2.2 ATmega1280

The next microcontroller taken into consideration in the Atmel AVR family was the ATmega1280. The ATmega1280 is all around beefier selection in comparison to the ATmega328; it provides superior memory capacity, performance, and pin count. Just as the ATmega328 is implemented into the Arduino, the ATmega1280 is featured on the Arduino Mega development board.

First, the ATmega1280 possesses a significantly higher pin count than the ATmega328. As seen in Figure 12, the ATmega1280 has 54 digital pins (14 of which can be used for pulse width modulated output) and 16 analog pins. Furthermore, the ATmega1280 also supports the Philips I2C protocol and contains multiple I2C ports. As mentioned previously, one port can support up to 128 unique addressed peripherals, yet multiple ones surely do not hurt and provide further flexibility for the design. If the ATmega1280 were used, communication with other modules, such as the ultrasonic sensors, GPS, and Bluetooth modules, would not be a concern since the pin count is more than sufficient for what has been estimated.

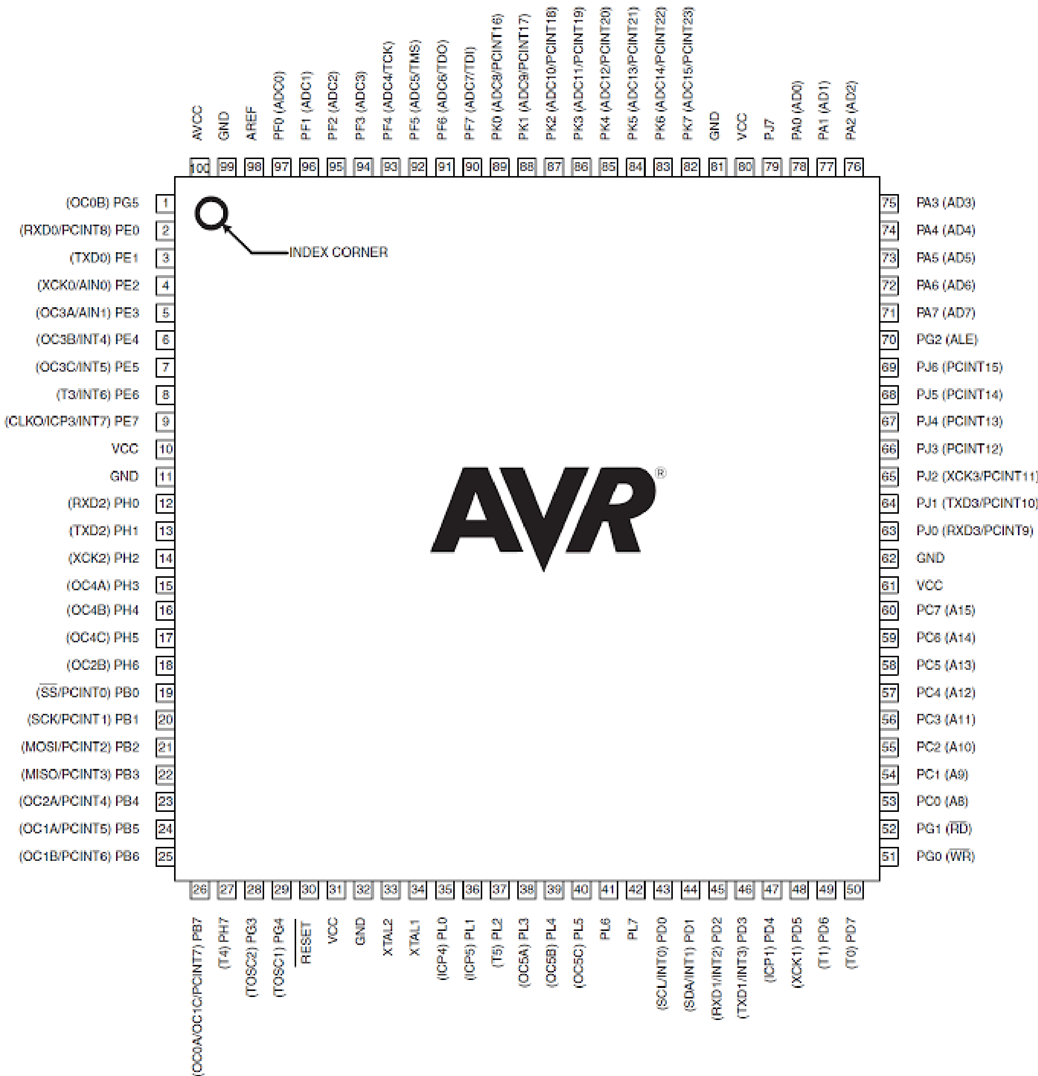


Figure 12: Pin mapping for ATmega640/1280/2560
Reprinted with permission from Atmel

As with the other microcontrollers considered, minimal power consumption was a key characteristic in considering use of the ATMEGA1280. As it is one of the larger and more powerful MCU's considered, it was expected to draw a large amount of current, yet the technical specifications were surprising. Specifically, at 1 MHz and 1.8 V the ATMEGA1280 consumes an ultra-low 500 microamps. In power down mode, it consumes an even lower current of 0.1 microamps at 1.8 V. Taking into account the ATMEGA 1280's 0 - 16 MHz speed grade, abundant pins count, large memory size, and ultra-low power consumption, it appears to be a robust chip packed with features beneficial to this implementation.

3.4.3 ARM Processors

ARM processors have increased in development and popularity over the recent years. Key features such as reduced cost, heat, and power use make it no surprise these reduced instruction set computing, RISC, based processors appear in many smartphones and tablets in today's market. Similar to Texas Instruments and Atmel, ARM Holdings' product line is copious with various chips being capable of meeting the needs of the Head On system. Further analysis of the ARM processor product line resulted in eyeing one specific MCU: the ARM STM32F405VG.

3.4.3.1 ARM STM32F405VG

First, the STM32F405VG features a healthy amount of flash memory and SRAM at 1 MB and 192+4 KB, respectively. This is enough memory to drive either unit in Head On, and still leaves a sufficient amount available for future additions to the system. To complement the large memory size, this ARM MCU features 100 pins as seen in Figure 13, 3 of which are intended for Philips I2C protocol connectivity. Furthermore, of the 100 pins on the MCU, 82 pins are available for general use which far exceeds the pin requirements of this project, thus giving further flexibility for future development and room to add more peripherals.

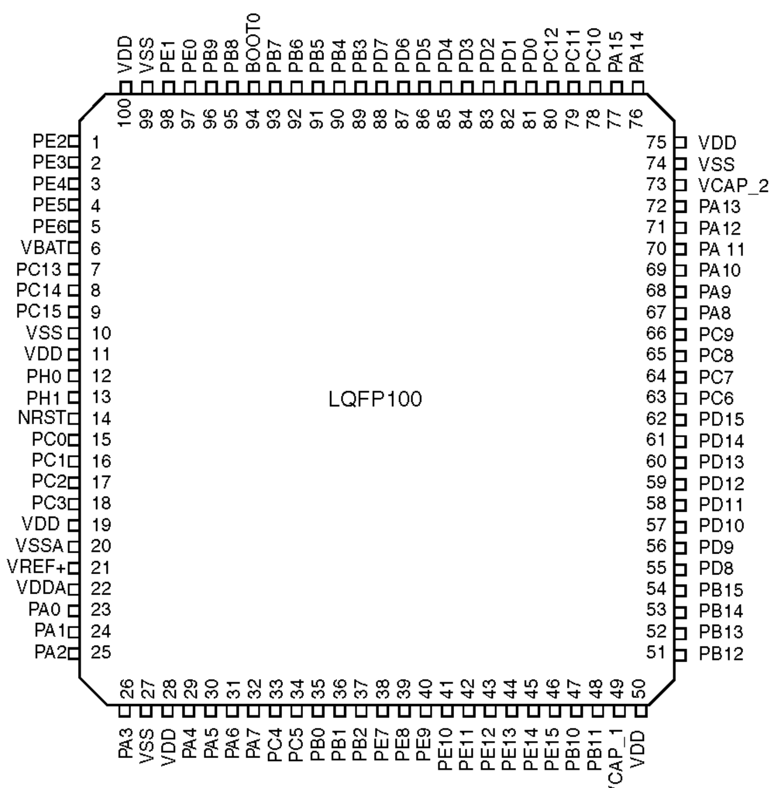


Figure 13: Pin mapping for ARM STM32F405VG
Reprinted with permission from STMicroelectronics

In addition to the flexibility it offers with the number of connections provided, the STM32F405VG must also fall below the project's power consumption requirements in order to remain a considerable option. Operating at a minimum of 1.7 V and maximum of 3.6 V, this ARM MCU draws a current of 40 milliamps, 280 microamps, and 2.2 microamps for dynamic run, stop, and standby modes, respectively. These levels appear higher than any of the other product lines previously mentioned in this document; regardless of this, these levels are reasonable and still within the threshold due to the processing performance of the ARM MCU.

3.4.4 Microcontroller Selection

Side by side, each microcontroller features its own set of advantages and disadvantages for this project. First, the MSP430 LaunchPad from Texas Instruments proved to be a potential final candidate because it featured low power consumption, small footprint, and inexpensive unit cost. In the same token, it was seen the ATMEGA328 from Atmel also featured the attractive points from the MSP430 along with I2C port connectivity and a higher pin count to successfully implement all the sensors and peripheral devices needed. The ATMEGA328 and MSP430 LaunchPad could be considered the smaller microcontrollers next to the ATMEGA1280, MSP430FG4618, and ARM STM32F405VG which featured a significant pin count increase and computation power. By implementing either one of these MCUs, we were given flexibility for current and future development of the project; however, these three microcontrollers' specs far exceed the needs of the project which led to choosing between the smaller, less powerful microcontrollers, the MSP430 LaunchPad and ATMEGA 328.

The decision between the MSP430 LaunchPad and ATMEGA328 was difficult as both microcontrollers are very much alike. The choice came down to preference. The ATMEGA328 was preferred because its popularity amongst hobbyist provides a rich community filled with relevant projects and applications that will be useful for getting all the modules and systems working and communicating properly.

3.5 Communication Technology Research

Wireless communication is defined as the transmission of information over a distance without help of wires, cables, or other electrical conductors. In modern days, one will find mobile devices that features one or more of the wireless technologies such as Bluetooth, ZigBee, and Wi-Fi. It is a crucial area of interest in the always connected modern age. Similarly, inclusion of one of these wireless communication technologies is crucial to the successful implementation of this Head On system since the helmet and motorcycle units will need to be in constant communication with one another without any interference to the rider. For the Head On system, three types of wireless communication were considered: Bluetooth, Wi-Fi, and ZigBee. In order for a decision to be made, each form of wireless communication was compared with certain project needs in mind, such as data rate, range and power consumption.

3.5.1 Wireless Communication Protocols

Before exploring what modules would allow wireless communication between the two units that comprise Head On, further research was necessary into these protocols. As mentioned previously, the three protocols considered were Bluetooth, Wi-Fi, and ZigBee. Each protocol, though similar in communicating wirelessly, has its own individual set of advantages and disadvantages in terms of security, network complexity, and operating range.

3.5.1.1 Bluetooth

One of the most recognized protocols among mobile devices and one of the first methods under consideration for wireless communication in Head On is Bluetooth. Bluetooth's popularity for connectivity is evident in today's market as it appears in headset, video game controllers, tablets, smartphones, smart watches, laptops, and even vehicles. The use of Bluetooth is featured in several close proximity environments similar to the one between the rider and motorcycle units in Head On. Specifically, Bluetooth is seen in applications such as hands-free handset use during phone calls, transfer between two nearby computers or phones, or listening to music through Bluetooth equipped earbuds.

Previously, Bluetooth was standardized by IEEE as 802.15.1; however, maintenance of the standard was ceased and "picked up" by the Bluetooth Special Interests Group. The group, consisting of over 25,000 member companies, is responsible for development of the specification, managing the qualification program, and handling trademarks. Any manufacturer that wishes to market any product as a Bluetooth device must meet the Bluetooth Special Interests Group standards first. This maintenance of the protocol offers peace of mind if Bluetooth is chosen to connect the motorcycle and helmet units of Head On together because there should not be any compatibility issues due to varying protocol standards when connecting Bluetooth certified modules.

In choosing an appropriate protocol, one of the first characteristics analyzed was range of coverage. The intention of the Head On system is to keep the rider connected during their ride, as well as allow him or her to roam around near his or her motorcycle freely, while still remaining connected to the motorcycle unit. This can be brought into fruition because Bluetooth connectivity protocol provides a range of approximately 10 meters for class 2 Bluetooth, which is the standard found in most mobile phones. In Table 5, the distance variations for all three radio classes of Bluetooth are shown as well as the maximum power consumption that they require. A closer look at the table reveals that class 2 is the halfway point in comparison to the other classes. Class 2's balance between the power consumption and operating range makes it the most suitable for the application of this Senior Design project, while still remaining within the project requirements.

Table 5: Bluetooth Power Consumption and Operating Ranges

Class	**Max Power**	**Operating Range**
Class 1	100 mW	100 m
Class 2	2.5 mW	10 m
Class 3	1 mW	1 m

Furthermore, Bluetooth networks use master/slave model to for data handling. Figure 14 illustrates this concept. These piconets, as they are commonly referred to, will have one master that can send or retrieve data from any of the slaves. Meanwhile, the slaves in the piconet can only send or retrieve data from the master and cannot “see” any other slaves in the network. The piconet on the left side of this figure represents the view of the piconet from the slave’s perspective. On the other hand, the right side of the figure represents the master’s view of the piconet. In the case of the Head On system, either the helmet or the motorcycle unit would be designated the master, and the other unit would be the slave. Because of the master’s ability to have multiple slaves, this wireless communication method would allow for easier integration of components to the Head On design in the future. Only unit that is assigned as the master would have to interact with this new slave; as a result, the original slave unit would be able to function as usual and would require no alterations to its code.

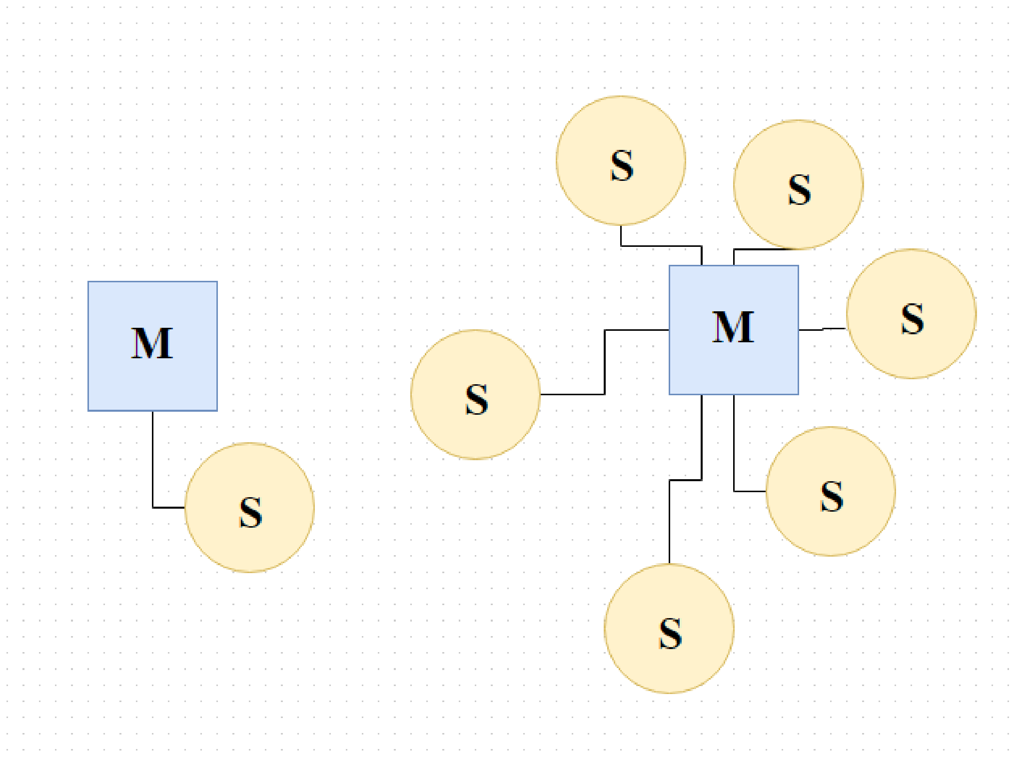


Figure 14: Depiction of the master/slave Bluetooth model

Similar to other wireless communication protocols, Bluetooth operates on the 2.4 GHz radio frequency; however, interference in this range becomes less of a concern due to the implementation of adaptive frequency hopping (AFH) which enables the connected devices to remain off frequencies used by other nearby devices. This would be beneficial to the Head On system because it would reduce the concern of possible interferences.

Another primary concern when considering which of the wireless protocol to use was power consumption. Bluetooth was built with a low power consumption design; this is evident in class 2 Bluetooth, which uses an impressively low amount of 2.5 mW at max power. This power consumption drops even further for devices that are equipped with Bluetooth low power technology, which is designed for maximum battery life instead of data transfer rates. This technology use only ½ to 1/100 of the power consumption of classic Bluetooth.

Gathering all details of the Bluetooth protocol's operation and characteristics, a list of key advantages and disadvantages for the protocol consist of:

Advantages:

- Low power consumption for class 1 and 2 radios
- Low connection complexity for easy pairing and connections
- Reduction of clutter on frequencies through adaptive frequency hopping
- Various operating ranges to meet any project needs up to 100 meters

- Direct line of sight not required for paired devices

Disadvantages:

- Utilizes cluttered 2.4 GHz frequency band
- Low data transfer speeds compared to other protocols (25 Mbps for Bluetooth 4.0 vs 250 Mbps for Wi-Fi)
- Lower security compared other protocols such as Wi-Fi and ZigBee

Among the various protocols considered for communication between the helmet mounted microcontroller and motorcycle mounted microcontrollers, Bluetooth was one of the easiest to configure a connection. Bluetooth makes pairing quick and hassle free by allowing devices to “remember” previous bonds, thereby allow future re-connecting without going through the pairing process again. However, with a low connection complexity comes one hindrance as the Bluetooth protocol contains a less secure network platform limited to key matching. Despite this, the lower security of the protocol is shadowed by its performance in power consumption and operating. Specifically, class 2 Bluetooth devices which possess low power consumption levels at 2.5 mW for a maximum operating range of 10 meters. All things considered, Bluetooth is an excellent fit for this project due to its balanced features.

3.5.1.2 Wi-Fi

Another familiar and popular protocol taken into consideration is wireless fidelity, otherwise known as Wi-Fi. The protocol is controlled by the Wi-Fi Alliance which defines it as any wide area local network product based on the IEEE 802.11 standards. It is common to for Wi-Fi to operate on 2.4 GHz radio bands and more recently 5 GHz radio bands. The development of Wi-Fi is mature, thus it can be trusted for establishing a reliable connection between the two units that comprise Head On.

When it came to considering the Wi-Fi protocol, two key features made it shine among all the other wireless communication technologies: data rate and connection distance. To begin, the range for an access point using a stock antenna under either the 802.11b or 802.11g standards can reach up to 100 m considering very little interference. This is a highly impressive range, yet it significantly exceeds the project requirements as the hope is that the rider is never separated that far from the motorcycle. Next, the data transfer rates for Wi-Fi eclipsed other mentioned protocols as it can provide speeds similarly high to a wired connection. For one of the IEEE standards, 802.11ac, data rates can go up to 433.3 Mbit/s per spatial stream. In the case of this system, data transfer rates must be sufficient enough to allow a consistent transfer of data from the ultrasonic sensors and GPS to the helmet system with minimal to no data loss.

While Wi-Fi excelled in the data rate transfer rates and connection, it is also important to determine if this protocol has a low enough power consumption.

Unfortunately, in the case of Wi-Fi, it the largest power consumer among all the protocols that have the same data transfer rates due to the range of the signal it emits.

Taking into account all details of how Wi-Fi operates and the characteristics of the protocol, a list of key advantages and disadvantages was constructed:

Advantages:
- Signal is able to reach long distances of up 100 m
- Can operate on 2.4 GHz or 5 GHz radio frequency
- High data transfer rates that were comparable to wired connections
- Variability in security provided by various security schemes

Disadvantages:
- Complexity in connectivity compared to other protocols such as Bluetooth
- Highest power consumption among considered protocols
- Potential interference issues due to radio wave propagation

This list detailed several key points for and against the Wi-Fi protocol. The range provided and data rates excel among all the protocols, however this comes at high cost in regard to energy consumption. While there is no denying that the data rates and connectivity range are impressive, the project currently does not require such features since no video streaming is currently planned and the system is designed to be used in close proximity on rides. As it stands, it is not sensible to use Wi-Fi given its excessive specs and power use.

3.5.1.3 ZigBee

One other wireless protocol considered is the less familiar, compared to Bluetooth and Wi-Fi, ZigBee protocol. ZigBee is a fairly recent technology, yet it is maturing as development has remained strong since its conception in 1998. Following creation, it has been standardized under IEEE 802.15.4 in 2003 and recently received a revision in 2006. Similar to Wi-Fi and Bluetooth, the ZigBee protocol development, research, and certification is handled by an association of members consisting of business, university, and government agencies known as ZigBee Alliance.

ZigBee combines features that can be found in both Bluetooth and Wi-Fi. To be specific, ZigBee connections are secured by a standard AES-128 security scheme, as seen in Wi-Fi, while providing low power consumption levels. The protocol also transmits on the 2.4 GHz radio band globally with access to sixteen 5 MHz channels. Unlike other protocols, which implement special techniques to ensure transmission is successful and free of interference, ZigBee does not utilize frequency hopping technology. Rather it depends on automatic retransmission to deliver data packets. Due to this, special care is required in

ZigBee implementations in order to ensure that no complications arise as a result of interference.

Various characteristics were essential in choosing the appropriate wireless communication protocol. First and foremost the protocol used needed to maintain low power consumption. In the case of ZigBee, it excels in low power consumption thus it is implemented into many low data rate applications requiring maximum battery life and security. Additionally, the ZigBee PRO product line uses a Green Power feature that allows communication using only 100-500 microjoules of energy. This makes ZigBee the best choice for low power consumption; however, other specifications such as data rates and operating range remain. As previously mentioned, ZigBee is featured in low data rate applications as the protocol has a defined rate of 250 kilobits per second with a range of 10 - 100 meters. Complications arise with the data rate as it is only a fraction in comparison to Bluetooth and Wi-Fi's data rates. However, the wireless communication is only needed to transmit data gathered from the sensors and GPS module thus 250 kilobits per second would still be sufficient in this application. One caveat with the range is devices connected through ZigBee must maintain line of sight in order for a successful connection to be achieved. When one takes into consideration the movement of the rider and any gear, it is seen that the line of sight between both units cannot be guaranteed at all times, thus jeopardizing both the connection and the system's effectiveness in assisting the rider.

In summary, the advantages and disadvantages of the ZigBee protocol are listed as:

Advantages:

- Operating range up to 100 meters depending on conditions
- Simple, noncomplex connection setup
- Secure network provided by AES-128 security scheme
- Lowest power consumption among protocols considered
- Low network acquisition delay at 30 ms
- Able to connect to multiple nodes

Disadvantages:

- Operates on cluttered 2.4 GHz radio band
- Susceptible to interference due to lack of frequency hopping technology
- Requires clear line of sight for connection

Overall, ZigBee is a well-rounded wireless communication useful for handling sensor data transmission. Featuring 128 bit symmetric encryption, simple connection setup, and a quick network acquisition, it provides peace of mind for the user in terms of no other nearby persons intruding into the Head On system and a simple, reliable connection. ZigBee continues proving itself as effective by providing the secure connection while drawing ultra-low current levels compared to the other protocols discussed in this document. However, with this in mind, ZigBee also has downsides in the form of low data rates which limits the flexibility of expansion later on for the project for high bit rate features such as video feed. Additionally, the protocol requires a clear line of sight between the modules that cannot be guaranteed especially in the motorcycle-rider application. Unless the unit placement can be designed free from any obstructions, use of ZigBee can prove disastrous towards the goal of aiding the rider.

3.5.1.4 Wireless Communication Protocol Selection

In selecting the best suited wireless communication protocol for the Head On system, certain factors were vital to ensure a reliable connection and guarantee that all data was processed between the microcontrollers in real time. These heavily weighted factors included: power consumption, data transfer rates, and operating range. Other factors that were also considered in the decision but were not emphasized as much include the network complexity, security levels, and interferences of the protocols. To illustrate, Table 6 below displays the differences between the Wi-Fi, ZigBee, and Bluetooth protocols considered based on various factors. These differences were analyzed in order to determine which protocol would be best suited for this project.

Table 6: Comparison of Wireless Data Transmission Protocols

Attribute	Wi-Fi	ZigBee	Bluetooth (Class 2)
Standard	802.11 (a,b,g,n)	802.15.4	802.15.1
Frequency	2.4 and 5 GHz	2.4 GHz	2.4 GHz
Operating Range	100 meters	10 - 100 meters	10 meters
Data Throughput	25 Mbps (g)	250 kbps	0.7 - 2.1 Mbps
Network Acquisition	3-5 s	30 ms	< 10 s
Security	High	High	Low
Relevant Power Consumption	High	Very Low	Medium
Example Battery Life	Hours	Months to years	Months to years

At a glance, Bluetooth stands out as the mid-range among the three protocols considered. Specifically, Bluetooth offers mid-levels of power consumption at 2.5 mW and an operating range of 10 meters for class 2. Furthermore, Bluetooth also offers flexibility with a data transfer rate of 25 Mbps, which ZigBee lacks at a limit of 250 kbps. One caveat of the Bluetooth protocol though is the low security scheme limited to key matching. However, this is a small compromise as the Head On system is designed as a situational awareness assistant rather than a means to transfer sensitive data or act as a security system for the motorcycle. Considering all the attributes of Bluetooth in relation to Wi-Fi and ZigBee, it was determined that Bluetooth is the best suited protocol for this project in order to keep the reliable, real time data displayed while still maintaining a low power system.

3.5.2 Wireless Communication Modules

After exploring various options for wireless communication technology that could be implemented, Bluetooth was chosen as the best fit for this application. Now, further research is required in order to select the best suited modules that will allow Bluetooth connectivity to be incorporated into the microcontrollers. The Bluetooth module selected needed to meet certain criteria. One of the first requirements of the Bluetooth modules is that it possesses a capability for UART serial communication. Other requirements include sufficient operating range, low power consumption, and a minimal design.

3.5.2.1 RN42

One of the first modules considered is the RN42 Bluetooth module. Specifically, the RN42 module considered is manufactured by Microchip, as seen in Figure 15, which comes in a small form factor measuring at only 13.4 x 25.8 x 2.4 mm. Microchip directly sells most models of RN42 for an affordable cost of $15.27. Earlier, it was discussed that Bluetooth technology is classified into one of three various classes. This specific module is classified as a Class 2 Bluetooth module; therefore, it would meet the necessary operating range as Class 2 is capable of operating up 10 meters apart. Additionally, the RN42 is certified by the FCC and Bluetooth SIG. Being a certified Bluetooth device, it reassured the group this part would bear no compatibility issues with other Bluetooth connections and functionality would meet the standard requirements.

In terms of connection, the RN42 module offers several favorable features. In particular, the RN42 supports various Bluetooth stack profiles which include SPP, HID, and DUN. Additionally, interfacing is made simple with this module with the ability to utilize ASCII commands over UART or use a USB connection interface.

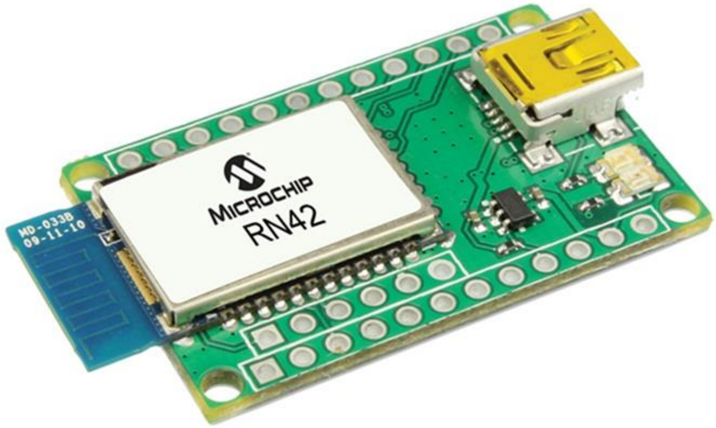


Figure 15: RN42 Bluetooth Module from Microchip
Requesting Permission in Progress

As with all of the other modules in this project, power consumption is a major concern. The RN42 only uses 26 microamps, 3 milliamps, and 30 milliamps when it is in sleep mode, connected, and transmitting respectively. The module will mainly be in transmitting mode since the microcontroller on the motorcycle and helmet units will be continuously communicating data when the system is used. While this does cost the greatest amount of power among the three power modes, this value is still quite small; therefore, the power consumption of this module will be minimal.

In summary, the key features that made the RN42 favorable for this application include:

- Class 2 Bluetooth with a max operating range of 10 meters
- Embedded stack profile support for: GAP, SDP, RFCOMM, L2CAP, SPP, HID. and DUN
- 128 bit encryption
- Low power consumption:
    - Sleep mode: 26 microamps
    - Connected: 3 milliamps
    - Transmitting: 30 milliamps
- Modular certified for the FCC
- Bluetooth SIG qualified
- Sustained data rate of 1.5 Mbps and 3.0 Mbps burst

3.5.2.2 HC-05

Another module Bluetooth communication module considered is the HC-05. The HC-05 is a very popular module among hobbyist because of its low cost. One can find that the HC-05 is widely available from various distributors, but the specifications remain similar among the various companies. For this project, the HC-05 module under consideration is made by Itead.

To begin, the HC-05 module is available for the low price of $6.00 making it an inexpensive device for linking the two microcontrollers that will comprise Head On. Use of this chip would minimize the size of the units mentioned as it is also available in a small form factor with the measurements 2.7 x 1.3 x 0.8 cm. This is ideal because one of the goals of this project is to keep the size of the motorcycle and helmet units small so that they will be non-obtrusive.

Next, the HC-05 is a Bluetooth SPP (Serial Port Protocol) module with a Class 2 classification similar to the RN42, so it will allow for an operating range of 10 meters. Another notable feature of this module is that unlike its even numbered counterparts (HC-04 and HC-06), which are set to either master or slave mode from the factory, the HC-05 can be configured to either mode according to the user's need. Connectivity is a breeze with this module as it comes with an auto connect feature that connects to the last known device. In terms of connecting the HC-05 to a microcontroller, it uses 6 pins: VCC, GND, TX, RX, KEY, and LED. These pins can be seen in Figure 16 below. Keeping in mind the low pin count on the ATmega328, this chip helps meet the physical constraint as only five of the six pins are needed for proper serial to Bluetooth data communication since LED is not necessary.

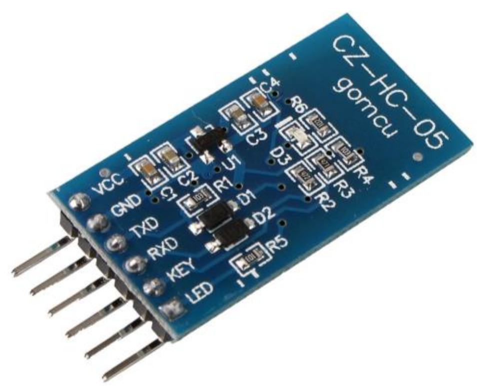


Figure 16: A generic HC-05 Bluetooth module
Requesting Permission in Progress

As with the RN42 module, power consumption must be taken into account when considering this module. The maximum power consumption of this module is 30-40 mA, and it occurs during pairing. This is not a major concern for this project because pairing will only have to occur when starting up the system. Unlike the RN42, the HC-05 does not have different current consumption for sleep, connected, and transmitting modes. In fact, a sleep mode does not exist. While this would normally be concerning, the current consumption of these modules is only 8 mA (after pairing). Because of this steady value of current consumption, calculating the amount of power needed for these modules will be significantly easier.

Key features that made the HC-05 favorable for this application include:

- Class 2 Bluetooth with max operating range of 10 meters
- Support for both master and slave mode
- UART interface with programmable BAUD rate
- Small footprint and affordable
- Bluetooth Spec v2.0+EDR compliant
- Maximum power consumption of 30-40 mA
- RoHS (Restriction of Hazardous Substances Directive) compliant

3.5.2.3 SaBLE-X

The newest of all the Bluetooth modules considered is the SaBLE-x module from LSR. The SaBLE-x is appropriately named after the most recent version of Bluetooth technology, Bluetooth Low Energy (or Bluetooth v4.0+ or Bluetooth Smart) which it makes use of. Bluetooth Low Energy devices are unlike their predecessors as focus is placed on improving cost, security, interoperability, and most of all power consumption. These BLE devices are designed to run for long periods on battery sources or energy harvesting devices which quickly made it attractive for either the helmet unit which is battery powered or both units.

First, it was important to meet certain constraints when choosing the right Bluetooth module. Specifically, constraints consisted of price, size, and operating range. In comparison to other modules, SaBLE-x was pricier yet affordable at $16.52 for one module from digikey.com. The module comes in a small form factor, as seen in Figure 17, with measurements 11.6 x 17.9 x 2.4 mm which is useful when trying to maintain minimal design for placement on the helmet or motorcycle. Though small and inexpensive, the SaBLE-x does possess significant operating range. Operating under Class 2 specifications, this module can communicate within a 10 meter range similar to the other two considered modules which is more than satisfactory for keeping the rider and motorcycle connected.

Figure 17: SaBLE-x module from LSR
Reprinted with permission from LSR

Next, the features of the SaBLE-x module were considered to determine whether it fit the Head On application or not. Two features that immediately stood out about this board during research on it were the availability of a development board package from LSR and the implementation of an ARM Cortex M3 microcontroller. The development board is useful for initial prototyping of the Head On unit and most helpful when it comes to testing phase of the project. The availability of a microcontroller on the Bluetooth module is also helpful as it allows flexibility to place the application in its entirety on the MCU or use an external MCU in conjunction with this module. Further features of this module consisted of upgraded Bluetooth support, ultra-low power consumption, and it meets various compliances. Specifically, the SaBLE-x module is compliant with Bluetooth 4.1 standards and is capable of Bluetooth 4.2 for future use. While supporting the most recent version of Bluetooth technology, this module maintains low power consumption with 8.4 mA in transmit mode, 7.4 mA in receive mode, and 1 μA in standby mode as it operates on a voltage range of 1.8 V to 3.8 V. In addition to low power and updated Bluetooth technology support, the module meets specific compliances similar to the other two modules analyzed. In specific, the SaBLE-x meets FCC, REACH, and RoHS compliances, which is important as the embedded devices will work in close proximity to users.

Overall, LSR's Bluetooth module is an impressive piece of hardware at a low cost that made a notable impression with features that include:

- Class 2 Bluetooth support for operating range of 10 meters
- System On Chip (SoC)
    - Integrated ARM Cortex M3 and Cortex M0
    - 128 kB Flash / 20 kB SRAM
    - Onboard 32 kHz and 24 MHz crystals
    - Drivers, BLE controller, and bootloader on ROM
- Low power consumption
    - 8.4 mA in transmit mode
    - 7.4 mA in receive mode
    - 1 μA in standby mode
- Development board available for testing
- Bluetooth 4.1 compliant, Bluetooth 4.2 capable
- FCC, REACH, and RoHS compliant

3.5.2.4 Wireless Communication Module Selection

The final choice for the Bluetooth module was a critical as this piece of hardware is the communication point between the two units of Head On. The modules were compared based on specific criteria such as footprint, Bluetooth version, power consumption, price, and additional features. The operating range of each module was also considered; however, all considered modules carried a Class 2 classification which provides 10 meters of operating range thus all were sufficient in this category.

The first module considered was Microchip's RN42 Bluetooth module. The footprint and price of this chip were low at a size of 13.4 x 25.8 x 2.4 mm and price of $15.27 per unit. Power consumption was low with draws of 26 microamps, 3 milliamps, and 30 milliamps when it is in sleep mode, connected, and transmitting respectively. The RN42 also carried the benefits of FCC compliance and Bluetooth SIG certification. Additionally, this module featured multiple stack supports which include GAP, SDP, RFCOMM, L2CAP, SPP, HID, and DUN. One drawback about this module though was the Bluetooth version support. Specifically, the RN42 supports 2.1 while the most current Bluetooth technology is version 4.1.

The second module observed was the popular HC-05. There was a lot of flexibility as far as sources with this module since it is a generic design, yet the one compared was supplied by Itead. The footprint of the module itself was a little bit larger than the first coming in a size of 2.7 x 1.3 x 0.8 cm. In contrast, the price was about half the price at around $8. The draw for the HC-05 was low at a max consumption of 30-40 mA and only 8 mA once paired. Furthermore, pin count was not an issue with the HC-05 as it only contains 6 pins of which only 4 are critical. The low pin count and ability to set the master and slave on these modules provide flexibility to the design of the two units in terms of selection of components that may require more pins and which unit provides control for the pairing. Similar to the RN42, this module lacked in the area of the Bluetooth support as it is limited to the older 2.0 version.

The final module considered is the most powerful of the lineup which is the SaBLE-x from LSR. Coming in a small form factor of 11.6 x 17.9 x 2.4 mm and only $16.52 per unit, this module packs a powerful punch. Specifically, unlike the other two modules considered, which were dedicated Bluetooth modules, the SaBLE-x expands on this by providing the ability to be used as a dedicated Bluetooth module as well as function as the microcontroller with its ARM Cortex processor, FLASH, SRAM, and onboard crystals. To add to this, the SaBLE-x module featured the newest Bluetooth technology with version 4.1 support and version 4.2 capabilities. Being a Bluetooth Low Energy device, this module was able to carry powerful hardware at no cost to power consumption as exemplified with draws of 8.4 mA in transmit mode, 7.4 mA in receive mode, and 1 μA in standby mode. Prototype testing of the design is also a breeze with this module with the availability of a development board. An additional benefit of the SaBLE-x was its FCC, REACH, and RoHS compliance.

The decision between all Bluetooth modules was a tough one as all the modules are relatively in the same range in terms of price, operating range, and size. The most significant differences were the features. After gathering the advantages and disadvantages of each module, it was determined the HC-05 was the best fit for this application. The reasons for this being it contained sufficiently low power consumption which is crucial for the helmet unit, was half the price of both its competitors, and pin count remained the lowest of all units at only 6 pins for the

HC-05. Its lack of newer Bluetooth technology is disappointing with support for the aged version 2.0; yet, this is overlooked by its advantages and the fact it will only be paired to another HC-05 module in this application rather than a phone or tablet.

3.6 Battery Research

When considering the battery needs for this design, it was immediately obvious that lithium ion technology was to be used. This technology is most common today in powering small electronics due to their small size and the low self-discharge they have. Since the design is to be placed on a motorcycle helmet it was important to have the battery be small in size and have a good recharge rate, both of which are featured heavily in lithium ion technology. In addition, it was important to make sure that these batteries are appropriate when using solar power to charge. However, before a decision could be reached, a more in depth analysis into lithium ion batteries (Li-ion) and lithium polymer batteries (LiPo) was necessary.

3.6.1 Lithium Ion Batteries

Extremely common in the market right now, Li-ion batteries are good way to power small electronics. They have a high energy density and a low self-discharge rate, which makes them a good choice for the purposes of this design. However, these batteries can become volatile, catch fire, or even explode if the charge exceeds what the battery is rated for. Since the battery is to be mounted on the helmet, it is paramount that it be as safe as possible for the rider using the system.

3.6.2 Lithium Polymer Batteries

Lithium polymer batteries are quite similar to lithium ion batteries. In fact, they are almost exactly the same. However, lithium polymer batteries are considered safer than their more volatile counterparts. Whereas, lithium ion batteries are saturated with electrolytes to aid with the creation of ions, lithium polymer batteries use a dry polymer electrolyte. Hence, the risk of electrolyte leakage and the risk of explosion are significantly reduced. Furthermore, due to the polymer technology they are much easier to manipulate into different shapes and sizes; this is significant since the battery must fit on the helmet system. A couple of drawbacks for this type of battery are that it has a lower energy density than the traditional lithium ion batteries and that they are more expensive.

3.6.3 Battery Selection

After considering the aforementioned attributes, it was decided that the lithium polymer battery would be best suited for the purposes of this design. Its versatile shape and reduced risk of explosion make ideal to be placed on a motorcycle helmet that is to be worn by a rider. The chosen battery is a 3.7V 2000 mAh lithium polymer battery sold by SparkFun. At 0.2 5 x 2.1 x 2.4 inches, this battery would be easy to place on the helmet and would not take up a lot of the already restricted space. Another point of interest is the weight of the battery. Because the battery will be placed on the helmet module, it should be as light as possible. If the battery is too heavy it will make the system uncomfortable for the user. Based off of calculations made regarding the power consumption of the modules on the helmet system, the necessary voltage for this battery was determined. The battery chosen provides sufficient power for the microcontroller, screen, and modules that make up the helmet system. The specifications of this battery are summarized in Table 7 below.

Table 7: Battery Specifications

Item	Value
Nominal Capacity	2000 mAh
Nominal Voltage	3.7 V
Weight	37 g
Cell Voltage	3.7-3.9 V
Discharge cut-off Voltage	2.75V
Impedance	< 300 mOhms
Dimensions	0.25” X 2.1” x 2.4”

3.7 Energy Harvesting System

A primary part of the power system for Head On is the use of solar cells. However, in order to safely harvest the solar energy created by the solar cells, an energy harvesting system must be incorporated into the design. The general energy harvesting system, shown in Figure 18, was used as a basis for the harvesting system of Head On. It will be modified as necessary in order to adapt to this system. For instance, while this figure does display the use of a temperature sensor, this is not included in the Head On design. Rather, other modules such as the display, electronic compass, and real time clock will be included.

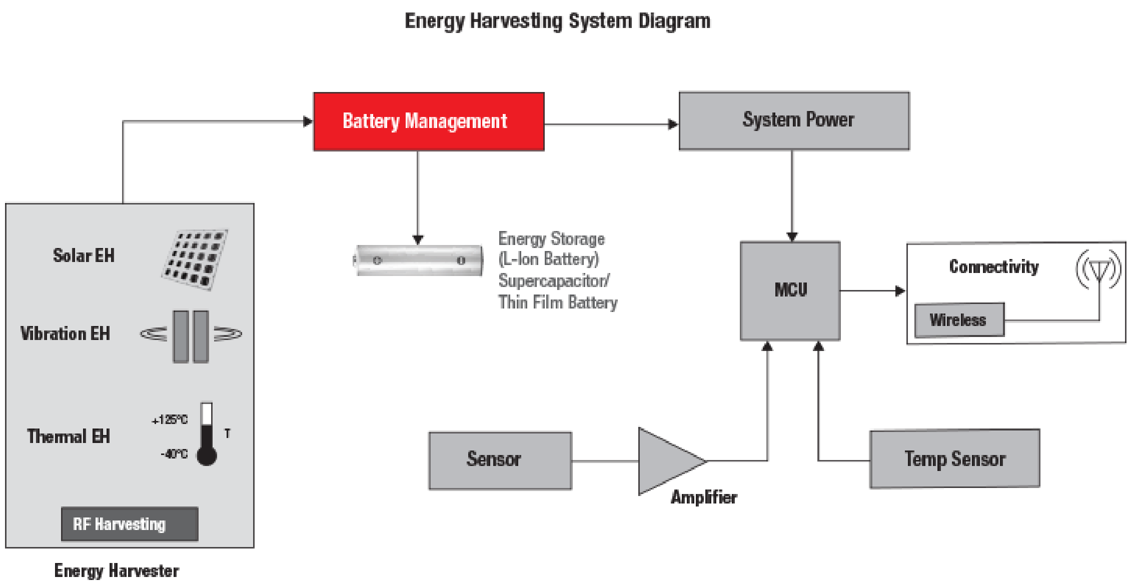


Figure 18: General Energy Harvesting System
Reprinted with Permission Texas Instruments

Once energy is extracted from the solar panels, it will be converted into electric power so that it can be used by the system. However, this brings up the concern of overcharging the lithium polymer batteries. In order to ease this concern, a few options for battery management were explored.

3.7.1 SparkFun Sunny Buddy - MPPT Solar Panel

Since the battery chosen for this design is sold by SparkFun, a quick inquiry into whether it is appropriate for charging with power harvested from solar panels was necessary. SparkFun's customer support responded in the affirmative and suggested the use of the Sunny Buddy to aid with the charging of the lithium polymer battery selected. Sunny Buddy is a maximum power point tracking solar charger, specifically designed to charge a single cell lithium polymer battery. Part of the power system for Head On will include a lithium polymer battery, so this module was a definite contender to incorporate in the design. This product, which is sold by SparkFun, is meant to have a maximum charge current of 450 mA, a maximum input voltage of 20V, and a minimum input voltage of 6V. It comes with a pre-installed barrel connector and a 2 pin JST connector. The typical design using the LT3652 for a 2A single cell lithium polymer battery is shown below in Figure 19. Since this design will use a 2A single cell lithium polymer battery, this typical design for the Sunny Buddy would be very close to the actual design. However, a few changes would still need to be made in order to apply the design to this system. For example, since the minimum input voltage for the LT3652 chip the Sunny Buddy uses is higher than the output voltage of the panels being utilized, a boost would need to be given to the output voltage in order for the design to work properly. This can be done in one of two ways. One option is to change the solar panels currently chosen for the design to ones with sufficiently higher output voltages. The alternative is to modify the typical design of Sunny Buddy's LT3652 charger, show in Figure 19. By adding a boost converter at the output of the solar panels and before the input of the Sunny Buddy, the appropriate voltage could be reached. At which point the actual components of the Sunny Buddy typical design should work properly, but more testing would be needed to be sure.

Another alternative would be to switch out the chip used by the Sunny Buddy. The LTC3105, which is manufactured by Linear Technology like the LT3652, is similar and has a minimum input voltage of 250 mV. This would fit the design better, but would basically mean getting rid of the Sunny Buddy as a whole.

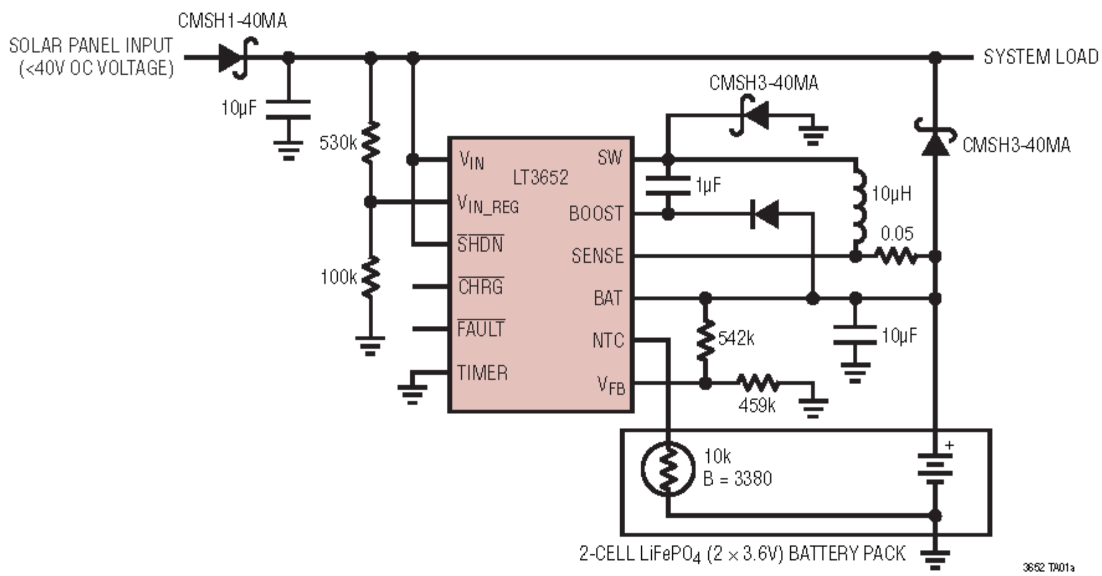


Figure 19: LT3652 Typical Design
Requesting permission in progress

The pin configuration of Sunny Buddy is illustrated in Figure 20. At a measly 3mm x 3mm, the Sunny Buddy chip would fit very easily into the design of Head On. Due to the low pin count, only twelve pins, and the extremely small size of the chip, finding a place on the system for the chip will be relatively easy, thus allowing more focus to be put on placing the other modules of the system.

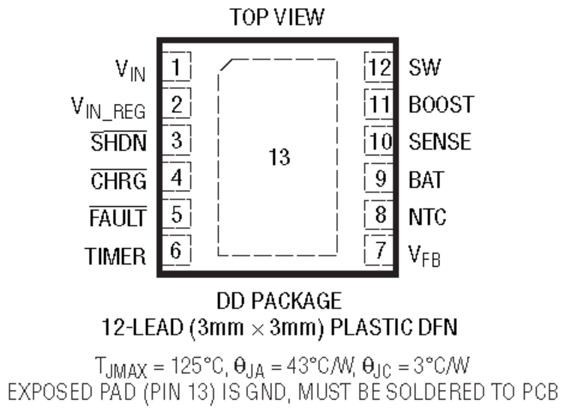


Figure 20: LT3652 Pin Configuration
Requesting permission in progress

3.7.2 Texas Instruments bq25570 Nano Power Converter

This device, which is manufactured by Texas Instruments, is designed to harvest power from sources, such as solar cells, and manage it in order to avoid overcharging a rechargeable battery. It contains a secondary power rail for other networks that might require the harvested power. One of the main factors that led to the consideration of this module is that it boasts a small footprint, which is an important constraint in this design. The BQ25570 is packaged into a 3.5-mm x 3.5-mm package and could readily be placed along with the rest of the system. As shown in Figure 21 the BQ25570 RGR package has only 20 pins. Both the size of the module and the number of pins are greater than Sunny Buddy. Of these pins, the input voltage for the VIN_DC, VOC_SAMP, VREF_SAMP, VBAT_OV, VRDIV, OK_HYST, OK_PROG, VBAT_OK, VBAT, VSTOR, LBOOST, EN, VOUT_EN, VOUT_SET, LBUCK, and VOUT pins ranges from -0.3V to 5.5V. The maximum input power is 510 mW.

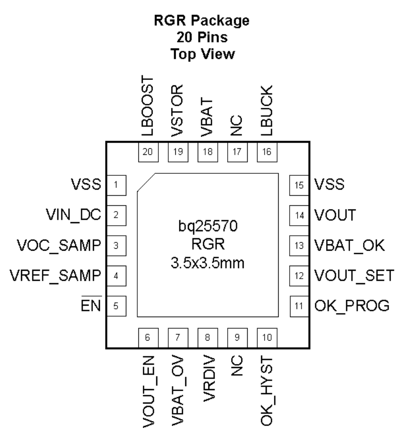


Figure 21: RGR Package 20 Pins Top View
Reprinted with Permission from Texas Instruments

A typical design application for the BQ25570 can be below in Figure 22. This design will be modified slightly in order to produce the desired output in the Head On system. For example, one of the modifications will be on the host pins. Since this project does not require the BQ25570 or its output to have the ability to turn off, the host control pins can be hardwired in their on state. In addition, another possible modification will involve the input filtering. Since the solar panels' supply voltage is very close to the required output voltage, this input filtering may need to be modified in order to compensate for input voltage fluctuations. However, this cannot be determined without testing further testing. At the moment, the remainder of the design will be kept identical to the layout in Figure 22.

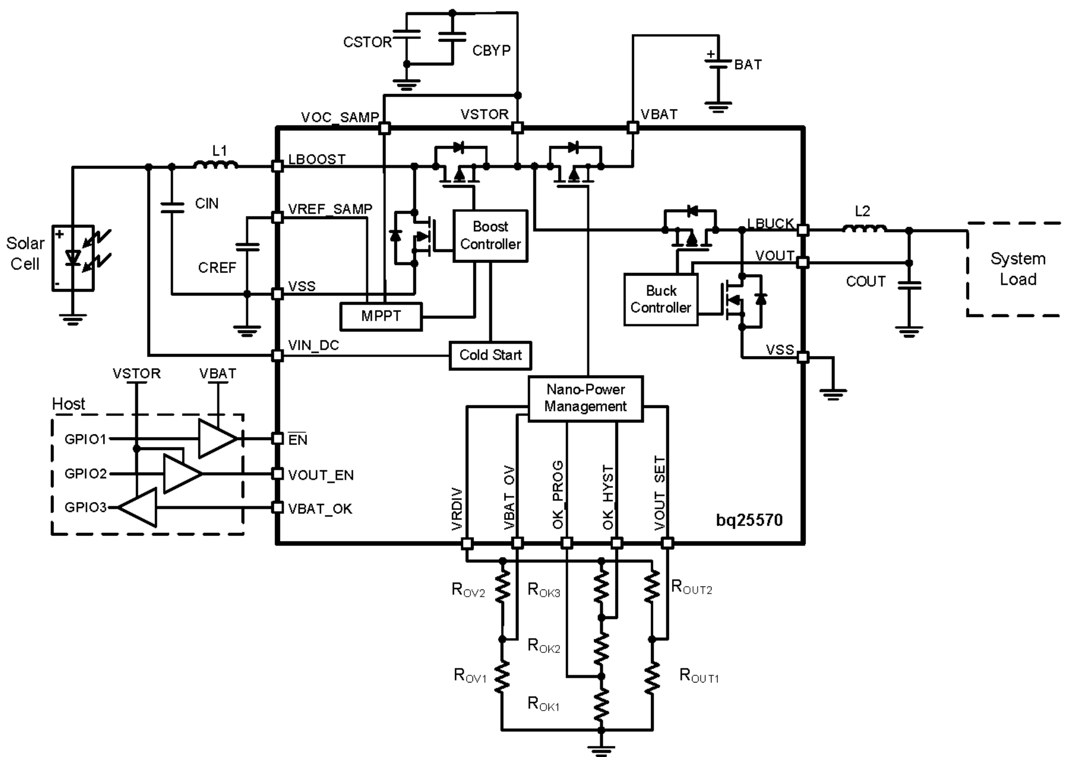


Figure 22: Typical Design for Solar Applications
Reprinted with Permission from Texas Instruments

3.7.3 Energy Harvesting System Selection

After some consideration, the BQ25570 Nano Power Boost Charger and Buck Converter was deemed to be most appropriate for the design. As noted above, the Sunny Buddy has a minimum input voltage of 6V and the chip the Sunny Buddy utilizes, the LT3652, has a minimum operating voltage of 4.95V. The output voltage of the panels that will be used is 4.8V. For this reason the Sunny Buddy was considered unsuitable for this project, unless the components were changed to accommodate the lower input voltage. Furthermore, the BQ25570 allows more control over the design and the handling of the harvested power, whereas the Sunny Buddy only has the input port for the panel output and the barrel connector for the battery to be connected to. Finally, the BQ25570 has the secondary power rail for the components to be connected to and the primary for the battery, thus making it the most suitable choice for this system.

Before changing the values of the components for the BQ25570, the values suggested by the data sheet will be used. This will allow for testing and adjusting until the final values can be settled. Again, the enabling function will not be engaged for the sake of this design because there is no desire to turn the BQ25570 off while the system is still in use. The table below outlines the values of the components, as detailed by the BQ25570 data sheet.

Table 8: BQ25570 Design Component Values

L2	10 uH
CBYP	.01 uH
Cref	10 nF
Cin	4.7 uF
Rov1	5.61 MOhms
Rov2	7.32 MOhms
Rok1	5.62 MOhms
Rok2	5.49 MOhms
Rok3	1.87 MOhms
Rout1	8.66 MOhms
Rout2	4.22 MOhms

3.8 Solar Panel Research

Due to the size and shape constraints inherent in designing a motorcycle helmet system, the most important aspect when searching for the solar panels was that they be flexible and easily molded in order conform to curves and grooves of an average motorcycle helmet. This is a major concern as there are several motorcycle helmet standards in place which require any additions to helmets to be no more than a few millimeters above the helmet. Furthermore, the panels must be thin and lightweight since the user will be directly affected by any increase in weight of the helmet. After researching the available products, two companies rose as front runners; both PowerFilm Solar and Solbian specialize in flexible solar panels that can be purchased and integrated by anyone. Geared towards personal use atop cars, recreational vehicles, or boats the flexibility and versatility are a top priority for the products these companies offer. However, after further inspection, it became clear that PowerFilm Solar simply had a greater variety of panels to choose from; thus, increasing the likelihood of finding a panel that would suit the specific needs of this design. In addition, due to the design constraints of this project, mainly the size and shape of the helmet, Solbian was deemed unsuitable. The products offered by Solbian were simply too big to be properly placed on the helmet. Consequently, it was several PowerFilm products which were considered before the final selection was made.

3.8.1 R7 7 Watt Rollable Solar Panel

The first solar panel considered was the 7 watt rollable solar panel. Advertised as a good solution for powering electronics on a boat, it is weather resistant, extremely flexible, and easy to use. These characteristics are particular important for the Head On system. The curvature of the helmet is fairly large so this would require the panels to have a greater flexibility in order to follow this curvature. Furthermore, the head on system will be used outdoors, and is therefore exposed to various weather conditions. Finally, because the solar panels are only one of many components in the Head On design, they must not be too complex; this project has a very specific deadline which must be met so an exorbitant amount of time cannot be dedicated to just this one feature. Another feature to note for this solar panel is that it comes with a female car charger adapter, so it simply needs to be rolled out and plugged into anything, from a battery to an electronic device.

Table 9 summarizes the specifications of this particular panel as detailed by the PowerFilm website. The operating voltage, current, and power all are sufficient for the Head On system. The surface area, though large, would fit the helmet as long as the panel is significantly flexible. As a result of this larger size, the weight is also a bit of a concern. Although it is only slightly more than half a pound, this solar panel will be placed on the helmet so ideally the weight should be as small as possible.

Table 9: R7 Rollable Solar Panel Specs

Operating Voltage	15.4 V
Power	7 Watts
Current	0.45 Amps
Width	368.3 mm (14.5 in)
Length (unrolled)	584.2 mm (23 in)
Weight	0.269 kg (0.594 lbs)

3.8.2 MPT6-75

The MPT6-75 is part of PowerFilm Solar's OEM series. The OEM series, which is specifically geared towards versatility, has several different panels that could be used for the Head On design. The MPT6-75 is the panel in this series with the highest voltage and current output while still being small enough in size to be accommodated on the motorcycle helmet. However, since each panel is putting out only 50 mA there would need to be several of them connected in order to meet the current requirements of the system as a whole. The size of the MPT6-75 would mean that not all parts of the panel will be receiving full amounts of sunlight since they would begin to curve into the sides of the motorcycle helmet. The specs for the MPT6-75 are summarized in Table 10. One of the most significant specifications to consider is the size. There is a very limited surface area on the helmet that is exposed to sunlight. Figure 23 illustrates the diminished size of the MPT6-75.

Table 10: MPT6-75 Specs

Operating Voltage	6 V
Wattage	.3 W
Current	50 mA
Typ Voc	8 V
Typ Isc	65 mA
Typ Output at AM 1.5	55 mA at 6 V
Length	114 mm (4.49 in)
Width	73 mm (2.87 in)
Weight	2.3 g (.08 oz)
Total Thickness	0.22 mm

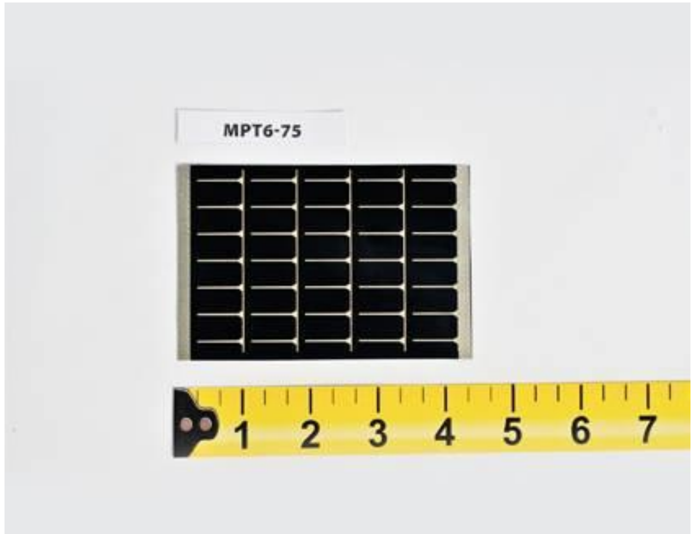


Figure 23: MPT6-75
Requesting permission in progress

3.8.3 MPT4.8-150

This small panel is part of the OEM series of the PowerFilm Solar products. As shown in Figure 24, it is much smaller in size, and therefore increasing its versatility. It can be connected with other products from this series in order to achieve a higher output voltage or output current. The smaller size also would allow for the panels to be placed in different configurations on the helmet in order to more easily achieve the desired output. However, one area of concern for this panel is that it does not have a UV stabilized surface so it would either need to be changed out after a year, or a request would need to be made to the company to manufacture a panel with the same output and size, but with a UV stabilized surface.

Figure 24: MPT4.8-150
Requesting permission in progress

Table 11 below identifies the specs for this particular OEM solar panel. If several panels were used, the operating voltage, wattage, and current would be sufficient for the system. Additionally, because of its smaller size, the flexibility of the panel would not have to be as great, and the panel would most likely fit more closely. Finally, another point of interest for this panel is its thickness. Due to standards in motorcycle helmet designs, all components of the system must be as thin as possible so that they do not protrude too much from the helmet. At only 0.22mm, the MPT 4.8-150 is in accordance with these standards and would hardly protrude from the helmet.

Table 11: MPT 4.8-150 Specs

Operating Voltage	4.8 V
Wattage	0.48 W
Current	100 mA
Typ Voc	6.4 V
Typ Isc	130 mA
Typ Output (at AM 1.5)	110 mA at 4.8V
Width	146 mm (5.75 in)
Length (unrolled)	94 mm (3.7 in)
Weight	3.9 g (1 oz)
Total thickness	0.22 mm

3.8.4 MPT4.8-75

The MPT4.8-75 is another panel from the OEM series of the PowerFilm Solar products. Table 12 summarizes some of the important specifications for this product as shown on the PowerFilm Solar website. It is even smaller than the MPT4.8-150, which makes it even easier to place on the helmet. However, as a result, the current output is half of that of the MPT 4.8-150. This would mean that even more solar panels would be necessary to produce the required current output. This could be a concern because not all areas of the helmet will have enough exposure to the sun, and depending on the amount of solar panels needed, they may not all be able to fit on the top of the helmet.

Table 12: MPT 4.8-75 Specs

Operating Voltage	4.8 V
Wattage	0.225 W
Current	50 mA
Typ Voc	6.4 V
Typ Isc	65 mA
Typ Output (at AM 1.5)	55 mA at 4.8 V
Width	73 mm (2.87 in)
Length (unrolled)	94 mm (3.7 in)
Weight	1.9 g (0.07 oz)
Total thickness	0.22 mm

Once again, with the use of several of these solar panels, the operating voltage, wattage, and current necessary could be achieved. The length, width, and weight of this panel are the smallest of all the panels under consideration. Consequently, allowing for more freedom when placing the panels. This compact length of the panel is demonstrated in Figure 25. Finally, the last specification to consider is the total thickness. Just like the MPT4.8-150, the thickness of this panel is 0.22mm, which follows the standards for external attachments for motorcycle helmets.

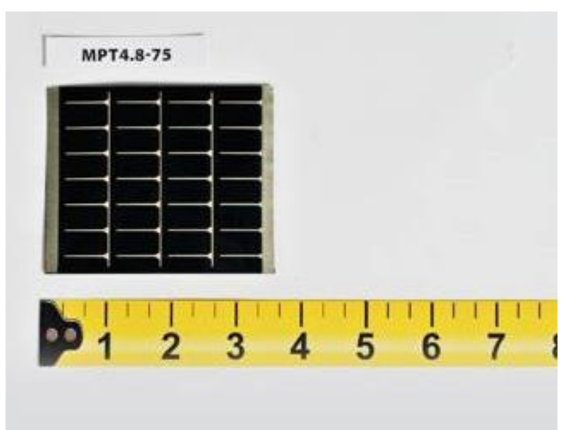


Figure 25: MPT4.8-75
Requesting permission in progress

3.8.5 Solar Panel Selection

After giving consideration to the above stated facts, as well as the low power consumption of the electronics on the helmet, the MPT4.8-75 was chosen for the design. Its small size will make it easier to place on the helmet while providing the necessary power output needed for the design. The MPT6-75 has the same current output as the MPT4.8-75, but has a bigger output voltage. Since the chosen components do not require such a high voltage in order to operate, the MPT4.8-75 is better suited to fit the size constraints. Furthermore, this selection will help keep the overall cost down. The MPT4.8-75 can be found on Amazon.com in a pack of ten for $30. Hence, the versatility, efficiency, and overall cost effectiveness of the MPT4.8-75 makes it the best choice for this design.

3.9 Step-Down (Buck) Converter

Power for the bike module will be fed from the motorcycle’s electrical system. The motorcycle provides a nominal 12V supply which must be stepped down significantly without creating overheating issues. Hence, it was clear from the beginning that some sort of buck converter would be necessary in order to ensure all the components received the appropriate voltage with minimal heat generation. The power available is well beyond the needs of module making the voltage regulator selection much less critical so they were not researched in depth.

The family of buck converters offered by Texas Instruments was looked at first. The TPS5490, TPS5491, and TPS5492 are all buck converters offered by TI with very similar characteristics. In fact, the main difference between the three is the maximum output voltage. The TPS5490 has a maximum output voltage of 16.2V, the TPS5491 has a maximum output voltage of 15.3V, and the TPS5492 has a maximum output voltage of 14V. Since the desired output voltages will be 3.3V and 5V it is unnecessary to have a higher output voltage from the buck converter. Therefore, from the TI family of buck converters the best choice would be the TPS5492.

Similar products to the TI buck converters are the LTC3622 by Linear Technology and the BD93291EFJ by Rohm Semiconductor. The maximum output voltage for the LTC3622 is the input voltage and the maximum output voltage for the BD93291EFJ is 5V. Therefore, it is clear the TI products have a bigger range, which makes them more desirable. Furthermore, the TI products have resources available that aid in design, such as the Eagle Cad files and samples of the chips which can be requested. For these reasons, the TPS5492 was chosen as the buck converter for the bike module.

3.10 Real Time Clock (RTC)

One of the features of the Head On system is to provide the current time on the display. Processors have an internal clock that can be adjusted whenever connected to computer as it has access to the system's clock. This internal time clock resets itself whenever powered off, thus a solution was needed to ensure accurate time was displayed upon every use of the Head On system. A solution was found in the form of a Real Time Clock which will be used to maintain accurate time for the MCU while the system is powered off. This will enable the MCU to display the correct time to the user upon start up. While there is a wide range of RTCs available most have many more features than required for this application. The parameters of interest are an accurate time of day, low power consumption, battery backup capable, small footprint, and ease of interface. These parameters were examined in detail for the DS3231, DP8573A, and the BQ32000 real time clocks.

3.10.1 DS3231 RTC

The DS3231, manufactured by Dallas Semiconductor, is a popular device used in several off the shelf RTC modules designed to plug into common MCU platforms. It utilizes an internal Temperature Controlled Crystal Oscillator (TCXO) to maintain a clock accuracy of +/- 2 minutes per year. An internal register is available to compensate for drift due to the aging of the oscillator. The clock time is selectable in either 24-hour or 12-hour AM/PM format. For the purpose of this project, the 12-hour AM/PM feature would be used, but the 24-hour form could be used in future upgrades where the user chooses the mode of clock he or she would like to use. The device can operate on a supply voltage of 3.3V; drawing 110 uA in standby mode and 200uA in active mode. It is equipped with direct connection for a backup battery. Although several of the connections are not used, it is offered only in a 16 pin surface mount package. The device is capable of connecting to a variety of MCUs using a 400 kHz I^2C serial interface.

3.10.2 DP8573A RTC

The DP8573A by National Semiconductor is also designed for microprocessor designs where accurate time and date are required. There is not a time keeping accuracy given for this RTC as it is dependent on the external crystal and loading capacitor used. An accuracy of +/-10 minutes per year is generally achieved by these systems when properly tuned. It requires a 4.5V supply voltage and draws approximately 140 uA when active and 3.3V at 12uA in standby. There is a direct connection for battery power and internal detection and switching is provided in order to prevent power interruptions. It is available in either a dual inline 24 pin package or a 26 pin plastic chip carrier. The larger footprint of this device is to accommodate address and data lines as it does not support the I^2C protocol.

3.10.3 BQ32000 RTC

Finally, the last real time clock considered is the BQ32000. The BQ32000 RTC from Texas Instruments is marketed as a replacement for many commonly used RTCs. Similar to the DP8573A, the accuracy of this RTC is also dependent on accuracy of external components. An accuracy of better than +/-10 minutes per year can be achieved with proper component selection. The device uses a 3.3V supply voltage drawing 100uA when in active mode but as little as 2V at 1.2uA in standby. In addition to a direct connection for battery operation it can provide a trickle charge to the backup battery while being powered by the primary supply. This allows for the use of a rechargeable battery or supercapacitor eliminating the need for user battery replacement. The device comes in an 8 pin dual inline package and all connections are required for operation. Communication with the MCU is requires only 3 pins using a 400 kHz I^2C Serial Interface.

3.10.4 Real Time Clock Selection

The RTC, while essential to system operation, is not a primary module and should be as simple as possible to implement. I^2C serial interface on the DS3231 and BQ32000 is a much more efficient method of transmitting data than the data bus of the DP8573A. It is faster, uses fewer connections, and can be read easily by the Atmega328 MCU. Due to the I^2C interface both the DS3231 and BQ32000 have smaller footprints. While the BQ32000 is half the size of the DS3231 it also requires additional external components negating its size advantage. The external crystal and load capacitors are critical to the operation and accuracy of the BQ32000. They must be carefully selected and tuned to achieve a desirable accuracy. The oscillator design is very sensitive to lead length, trace size, and connection quality making implementation more time consuming. In addition the crystal is external to the oscillator making it very sensitive to temperature changes. As the final system will be operated outdoors temperature extremes can be expected. The DS3231 eliminates both the problematic implementation and the temperature sensitivity by utilizing a completely onboard TCXO as shown in Figure 26. Additionally, the DS3231 supports both 24-hour and 12-hour time. This reduces the number of calculations that will have to be made by the software. At the moment, the design for the user interface incorporates the 12-hour clock; however, in the future this dual time format could be made an option for the user to customize the system. The efficient MCU interface, the modest device footprint, and the onboard oscillator of the DS3231 make it the best choice for this application.

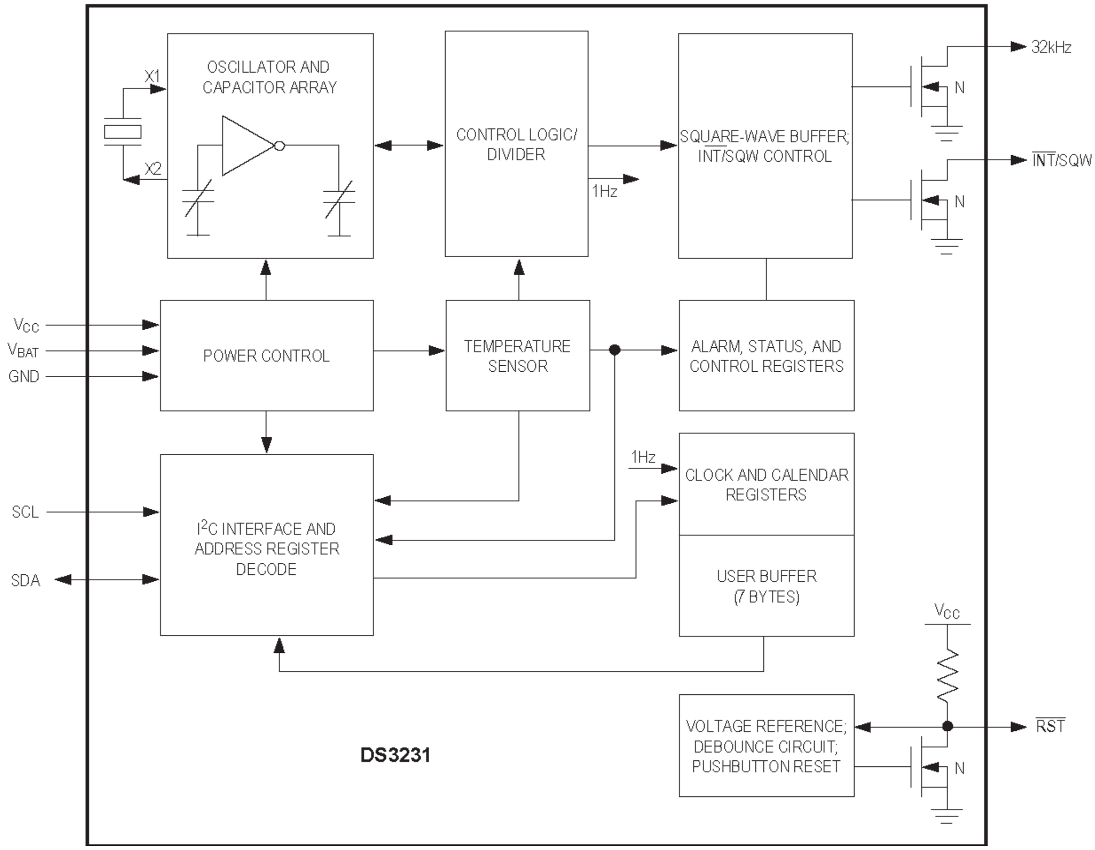


Figure 26: DS3231 RTC Internal Schematic
Requesting permission in progress

3.11 Electronic Compass

Directional information can be provided by the GPS module but that information can become invalid when the rider is stopped or changes direction in a narrow radius. In order to provide the rider with accurate compass heading information and electronic compass module was incorporated into the design. The compass was installed in the helmet module to reduce the data transmitted over the wireless connection and to correlate the compass heading to the rider’s line of sight versus the position of the bike. Since the rider’s head changes in elevation as well as azimuth during use the system must be able to sense both changes in orientation to provide an accurate heading. In order to achieve this only 3-axis magnetometers which also incorporated a 3-axis accelerometers were reviewed for this application. The sensors that were considered were the FXOS8700CQ, MC6450, and the LSM303D.

3.11.1 FXOS8700CQ 6-Axis Sensor

The FXOS8700CQ 6-Axis Sensor with Integrated Linear Accelerometer and Magnetometer manufactured by Freescale Semiconductor is designed for use in personal navigation devices and unmanned aerial vehicles. It is produced in a 16 pin QFN package and operates over a wide temperature range. The accelerometer measures along three axis and has three selectable ranges from +/-2g to +/-8g. The magnetometer also measure in three axes with a measurement range of +/-1200 uT with a sensitivity of 0.1uT. Both sensors have a selectable output data rate from 1.563Hz 800Hz available for I^2C or point to point serial interface The data rate is limited to 400 Hz when interlacing the data from both sensors in hybrid mode. It requires a 2.5V supply but the current consumption varies significantly depending on the measurement mode and output data rate. The maximum current draw of any configuration is 1072uA. However for this application the sensor would most likely be operated in hybrid mode and draw a current of 183uA.

3.11.2 MC6450 6-Axis eCompass

The MC6450 6-axis electronic compass from mCube is also a combined 3-axis accelerometer and 3-axis magnetometer geared toward portable navigation applications. Both sensors and an I^2C serial interface are combined in a 16 pin LGA package. The accelerometer has a single measurement range of +/-8g. The magnetometer measures +/- 2400 uT with a sensitivity of 0.15uT. The sample rate of both sensors is programmable from 0.5Hz to 100Hz and the data output is available at up to 400kHz over the I^2C serial connection. The sensor uses a nominal 2.8V supply and draws 3uA in standby and between 60uA and 2500 uA depending on the output data rate.

3.11.3 LSM303D eCompass Module

The LSM303D Ultra-compact high-performance eCompass module by STMicroelectronics is another integrated 3-axis accelerometer and 3-axis magnetometer. It is designed for use in tilt-compensated compasses as well as several other position and rotation detection applications. In addition to the accelerometer and magnetometer the LGA-16 package also houses an embedded temperature sensor and a SPI/I^2C serial interface with a FIFO data buffer. The accelerometer has five measurement ranges from +/-2g to +/-16g full scale. There four available ranges on the magnetometer from +/-200uT to 1200uT with a sensitivity of 0.008uT to 0.479uT depending on the range selected. The output from all the sensors is accessible via either an I^2C or SPI serial interface. The built in FIFO buffer is intended to reduce power consumption. By having the sensor data ready for transmission it eliminates the need for the MCU to constantly poll the sensor for data. The sensor operates on a 2.5V supply drawing 1uA in standby and 300uA during normal operation.

3.11.4 Electronic Compass Selection

All the sensors researched had very similar specifications and operating requirements. The ranges and sensitivities of each were more than adequate to provide an eight point compass heading. Each also provided an I^2C serial interface making programming simpler and to maintain consistent communication protocols across the system. The two main features that make the LSM303D the best choice are the FIFO data buffer and the consistent power draw. The buffer eliminates the need for choosing between output data rate and power consumption. While the power consumption of the LSM303D is not the lowest possible of the three devices it does provide the greatest access to sensor data for the current required.

3.12 Global Positioning System (GPS)

Several options were considered for acquiring speed and heading information for display to the user. The primary methods to measure speed were using the motorcycle speed sensor, a radar based system, or the Global Positioning System. Reading the speed directly from the motorcycle would be complicated and very invasive procedure most likely requiring professional installation. In addition speed measurement devices vary between vehicles requiring different signal processing based on the motorcycle used. While a RADAR system would provide a consistent speed signal it would also require an additional sensor unit on the front of the motorcycle increasing the complexity of the system and make installation more cumbersome. In addition, neither of these options provides heading information. It was found that GPS modules about the size of a quarter were available which provided both accurate speed and heading information. Furthermore they could be incorporated directly into the circuit board design making them an ideal choice for this application. Three devices were reviewed for their suitability; the PA6C, LS20032, and the GP-735.

3.12.1 PA6C LadyBird-3 GPS Module

The PA6C LadyBird-3 manufactured by GlobalTop Technology has a 66 channel receiver. It provides a position accuracy of 2.5 meters and a velocity accuracy of 0.05 meters per second. The module typically takes 33 seconds to acquire data from a cold start and 1 second when a backup battery is used to enable a hot start. The PA6C also has and built in antenna and a 9600 baud rate for serial data transfer. In addition to transmitting data in real time it has an onboard data logging feature that can up to two days of continuous data. The device operates on a 3.0V to 4.3V supply and draws a max current of 25mA during satellite acquisition but only 20mA while tracking. It has an automatic power saving mode that can reduce the current draw to as low as 180 uA by using an algorithm to balance tracking accuracy and power consumption based on motion. The entire device in housed in a 16mm surface mount package.

3.12.2 LS20032 GPS Smart Antennae Module

The LS20032 GPS Smart Antenna Module from LOCOSYS Technology is very similar to the PA6C. It also uses a 66 channel receiver to achieve a 2.5 meter position accuracy. The cold start acquisition time is typically 32 seconds while the hot start time is one second using the built in micro-battery. It has up to a 10 Hz refresh rate and transmits data over a 9600 baud serial connection. The data logger function stores GPS data in internal flash memory however the datasheet does not give any indication of how much storage is available. The module operates on a nominal 5V supply and draws 19mA during normal operation. The package is a bit bulky measuring 30mm x 30mm with components mounted to the underside preventing a flush mounting to the board.

3.12.3 GP-735 Smart Antenna Module

The final GPS unit was the GP-735 GPS Smart Antenna Module produced by ADH Technology. It uses a 56 channel receiver to track its position within 2.5 meters. In addition to tracking positions, it also provides velocity with speed accuracy better than 0.1 meter per second and heading error of less than 0.5 degrees. The rated accuracies are achieved after 29 seconds from a cold start. From a hot start using the onboard battery backup accurate data can be read after only 1 second. The device has a built in patch antenna but has connections for an external antenna if desired. The output data rates of 1Hz and 10Hz using the USB interface and up to 9600 bps over the serial interface. The power consumption during continuous tracking is only 37mA and operates on a 3.1V to 5.5V supply. The size of the device is quite large at 35mm x 8mm x 5.2mm including the connecting jack at one end.

3.12.4 GPS Module Selection

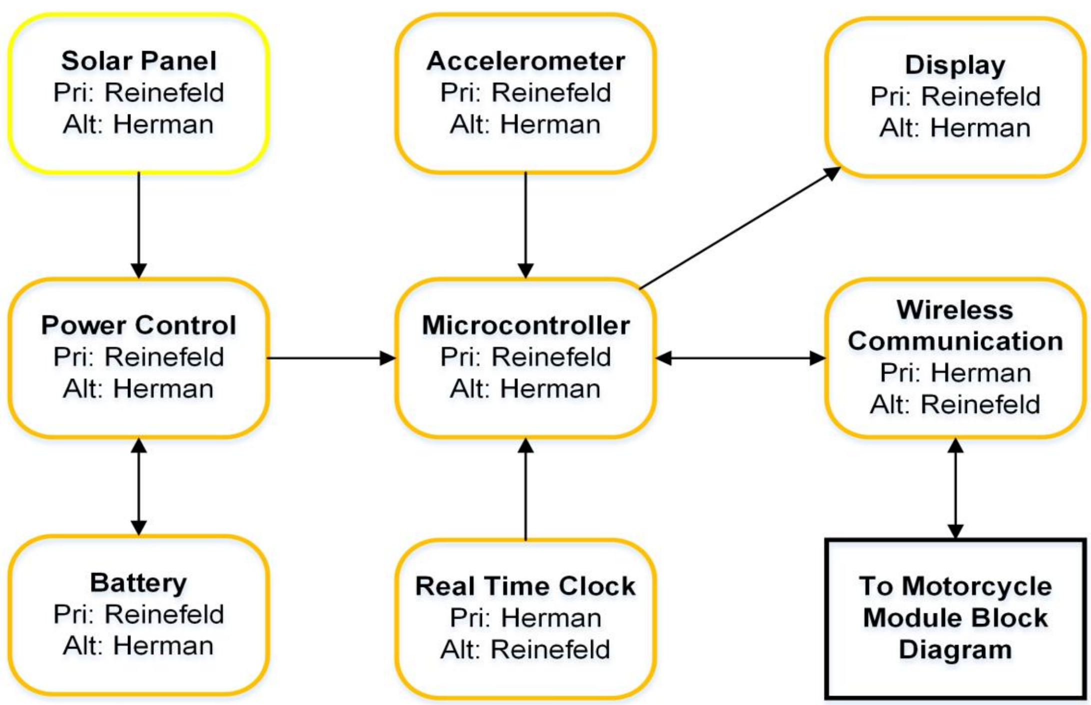
All of the GPS modules reviewed were very similar in accuracy and data transfer rates. The GP-735 had a slightly faster cold start time than the other models but the highest power consumption of the three. Its footprint was also bulky due to its overall size and large connector. The PA6C and the LS20032 were almost identical in all their features and specifications except power consumption and size. The PA6C draws slightly more current during normal operation but also operates at a lower voltage resulting in about 16mW lower power use than the LS20032. The LS20032 is almost twice the size of the PA6C and has components on the bottom side of the module that would make mounting difficult. In contrast the PA6C will surface mount to the main board providing stable physical and electrical connections. For these reasons the PA6C LadyBird-3 is the better choice for this application.

4.0 Block diagrams

In order to demonstrate how the hardware of the helmet and motorcycle modules interacts with each other, block diagrams were created. In addition, block diagrams were also created for the software aspects of the two modules. These diagrams explain how the software communicates with the hardware, analyzes the data received, and outputs the results.

4.1 Helmet module hardware

Figure 27 below summarizes how the hardware components in the helmet module interact. The solar panel and battery both are regulated by the power controller. This power is then used by the microcontroller in this module. The microcontroller will receive data from the motorcycle system using wireless communication, as well as the accelerometer and real time clock modules located on the helmet system. This data will then be sent to the display. Using wireless communication, the microcontroller on the motorcycle system communicates the data gathered from the sensors to the microcontroller on the helmet module. Based on the data the microcontroller receives, the information is then displayed on the screen.



Block Status			
	To Be Acquired		Design
	Acquired		Prototype
	Research		Completed

Figure 27: Block Diagram of Helmet Module Hardware

4.2 Motorcycle Module Hardware

Figure 28 below summarizes how the hardware components of the motorcycle module interact with each other. The power regulator will be used to send the appropriate amount of power from the power system to the microcontroller. This microcontroller will communicate with the left, right, and rear sensors in order to detect objects. Additionally, the microcontroller will receive data from the GPS and electronic compass modules. Once all the data is collected, it will be transmitted to the microcontroller on the helmet system using wireless communication.

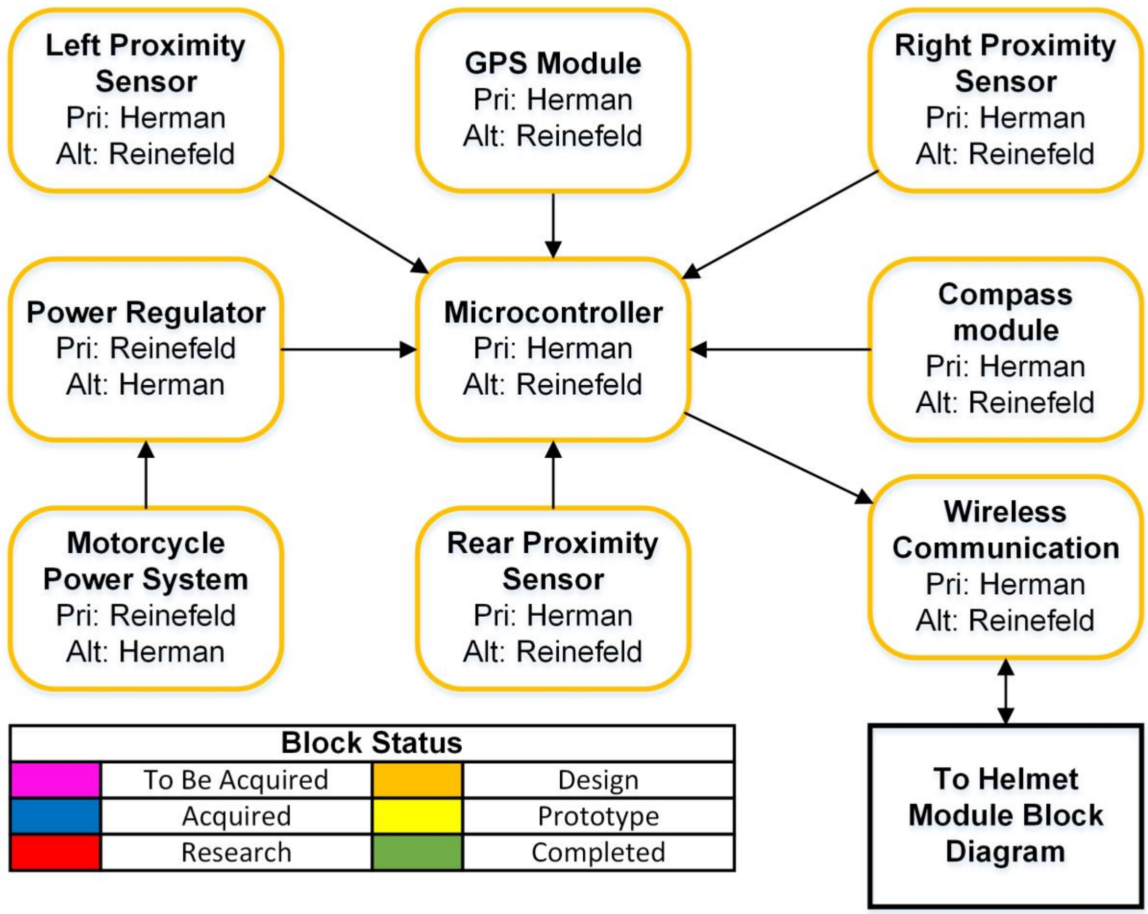


Figure 28: Block Diagram of Motorcycle Module Hardware

4.3 Helmet Module Software

Figure 29 below illustrates the software components of the helmet module. Using wireless communication, the microcontroller on the helmet module will receive the data that was collected by the motorcycle system. In addition, data from the real time clock module and the accelerometer located on the helmet system will also be gathered. Based on this data, the appropriate functions will be called to format the display. Finally, the display will be updated with the new data.

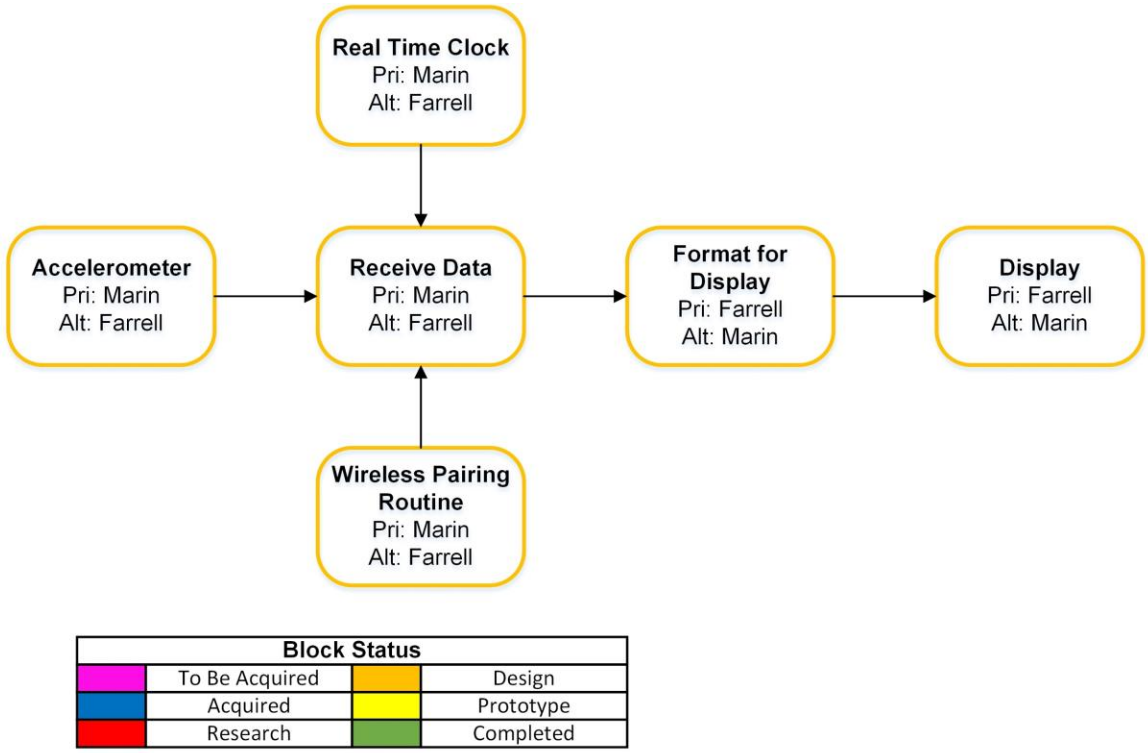


Figure 29: Block Diagram of Helmet Module Software

4.4 Motorcycle Module Software

Figure 30 below illustrates the software components of the motorcycle module. A cycle will be created in order to prioritize the signals and timing for the gathering of data from the modules. The GPS module will be used to gather the speed and cardinal direction of the rider. Next, the sensor inputs will be gathered one at a time. Once this is complete, the system will determine if it needs to use the compass module to find the cardinal direction. If the rider was not moving when the GPS module was analyzed, then a valid cardinal direction would not have been received. In this case, the compass module would be used to find this value. After receiving all the data, the values will be compared to the threshold values in order to determine if there was a change and if so what that change was. Appropriate indicator signal values were then calculated. Finally, the data will be transmitted, using wireless communication, to the helmet system.

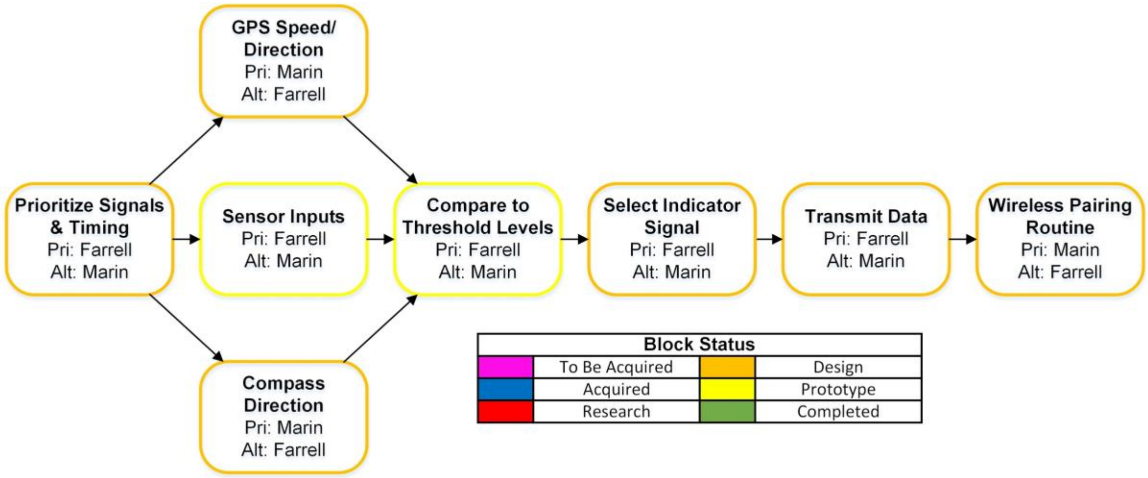


Figure 30: Block Diagram of Motorcycle Module Software

5.0 Related Standards

When designing the system, it was critical to take into consideration relevant constraints that would impact the system. These constraints not only impact the goals, requirements, and specification of the project, but also determine what features of the design are not feasible. The major constraints that are focused on in this project are the constraints impacting the design of the helmet and road standards.

5.1 Department of Transportation Helmet Safety Standards

Because half of the system will be placed on the motorcycle helmet, it is critical that it does not go against any standards for helmet design. There are two main groups who develop standards for motorcycle helmet design. The first of these organizations is the Department of Transportation. Specifically, the standard in consideration is the Federal Motor Vehicle Safety Standard No. 218 (49 CFR Sec. 571.218). In S5.4, it is explained that the rider must have a 105º field of view on either side of the mid-sagittal plane. As shown in Figure 31, this plane, which is also called the longitudinal plane, is located in the center of the helmet. This standard will impact the placement of the LCD display.

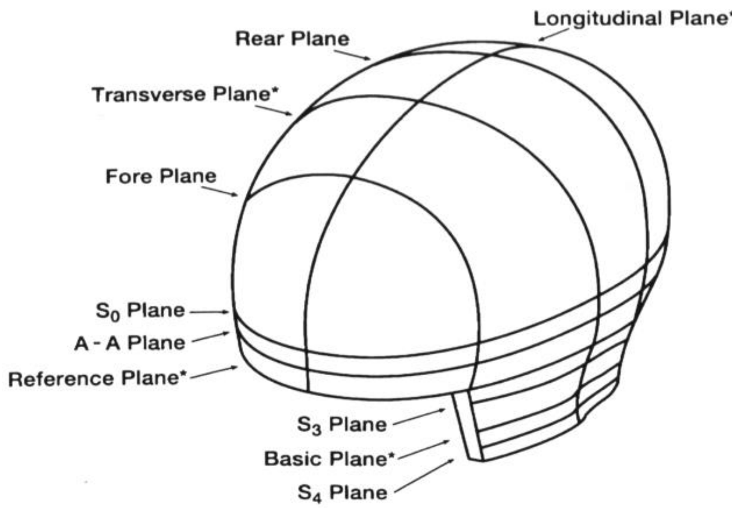


Figure 31: Motorcycle Helmet Layout
Reprinted with Permission from The Snell Memorial Foundation

Additionally, Standard No. 218 section 5.5 describes the standards for projections inside and outside of the helmet. The PCB board and its attached modules can be considered as a projection. This standard explains that if the motorcycle helmet includes a projection on the outside, it must be a portion of an accessory that is necessary. In this case, the Head On system is the accessory, and the PCB board on the helmet is a portion of the system that is absolutely necessary. Without this PCB board, the entire Head On system is redundant as it is the MCU on this board that processes all of the data and causes the results to be displayed on the screen. This PCB board must also not protrude more than 5mm from the helmet in order to follow this standard. Furthermore, the solar panels must also be flexible enough to mold to the helmet as flat as possible so that they meet the 5mm requirement.

5.2 Snell Memorial Foundation Helmet Safety Standards

The second organization that creates motorcycle helmet standards is the Snell Memorial Foundation. According to the M2015 Helmet Standard developed by the Snell Memorial Foundation, if anything is added to the helmet and it is greater than 7 mm, it must be removable. In the case for this project, the PCB and modules that are going to be attached to the helmet must be removable, as well as the display that is placed in front of the helmet. The solar panels on the other hand, will be less than 7mm so they will not have to be removable. However, the standard states that any addition to the outside of the helmet that is less than 7mm must fit the helmet smoothly in order to reduce the amount of friction it produces. Therefore, it is imperative that the solar panels are extremely flexible so that they can mold to the curvature of the helmet. Additionally, the Snell Memorial Foundation also has helmet standards regarding the peripheral vision of the rider. The rider must have the ability to see 210º horizontally, 7º up, and

30º downward. This particular standard will impact the placement and size of the display.

5.3 AASHTO Standard Road Widths

The American Association of State Highway and Transportation Officials determines the minimum width for traffic lanes in the US. This standard is relevant to this project because in order to be successful, the system must be able to detect vehicles that are in adjacent lanes. Different types of roads have different standard widths; therefore, for the purpose of this project the maximum road width will be taken into consideration. Single lane ramps, whose width is significantly larger than other roadways, will not be taken into consideration since there will not be other vehicles adjacent to the rider; for all other roadways, the maximum width is twelve feet. Therefore, when choosing the proximity sensor, this maximum value will be taken into account.

6.0 Design Constraints

When designing Head On, it was important to take into considerations the constraints that would impact the project. Time, funding, and resources were three of the major constraints for this project. In addition, standards, such as helmet safety standards also impacted the design. These constraints impacted what would and would not be feasible. After identifying and analyzing them, certain features from the initial design were discarded entirely from the design, while others became stretch goals.

6.1 Economic Constraints

When designing the Head On system, the costs of the parts and manufacturing had to be taken into consideration. An estimated budget for the prototype and final product was determined. Once it was determined what funding the project would receive, the budget was re-analyzed in order to ensure that the project remains in budget. This constraint resulted in some of the parts being removed and others changed. Many similar products that are already out in the market are quite expensive. As a result, it is important that Head On is more economical. Consequently, a balance had to be found between cost and accuracy of parts.

6.2 Time Constraints

Since Head On must be completed in only two semesters, time was a major constraint of this project. Numerous features could be added to this project if time were not an issue. However, a working prototype must be built and programmed in only two semesters; as a result, the list of features to be included had to be narrowed down to the most important ones. Those features that did not make the cut then became stretch goals that would be implemented if sufficient time remains at the end of Senior Design II. It is imperative that adequate time is allocated after the final prototype is created, in order to allow for testing of the system. Based on the results of the system, some problems with the design may be discovered and changes may have to be made.

6.3 Health and Safety Constraints

One of the goals of Head On is to increase the safety of the rider. As a result, it is imperative that no part of the system poses any safety concerns for the rider. One area of concern is the field of view of the user. Because the display of the system will be mounted outside of the helmet, it must be positioned in such a way that it does not become an obstruction. The size of the screen must also be limited in order to ensure that it does not block the user’s view. There are several standards that dictate how motorcycle helmets must be designed. This includes specifications on attachments to the helmet. It was important to design the helmet with these constraints in mind. Additionally, the power system was an area of concern, especially the portions of the system that are on the helmet. Batteries have the potential to become volatile, catch fire, or explode; therefore, it is imperative that the battery chosen has the lowest possible chance of injuring the rider in these or any other ways.

6.4 Manufacturability Constraints

When creating the system, the design had to be feasible. This includes the creation of the components, such as the PCB, as well as the assembling of the final prototype. The design of the PCB must be realistic so that it can be printed. In addition, because the resources that are available to this group are limited, the prototype must not require equipment that the group does not have at their disposal.

6.5 Sustainability Constraints

Because the parts of the Head On system components will be exposed, environmental factors play a major role in the sustainability of the system. All of the components of the system must be able to function in a variety of weather conditions including rain, snow, and wind. For example, in the case of rain and snow, the modules themselves must either be waterproof or must be put into a waterproof casing in order to ensure that they remain functional. In addition, the components must not extend too far from the motorcycle and helmet or else the force from the wind could damage them. Finally, the lifespan of all of the components must be similar in order to ensure that the device does not fail due to one part. Ideally, the parts should have a lifespan equal to that of a typical motorcycle helmet, five years.

6.6 Environmental Constraints

For this project, one of the major focuses is the use of renewable energy in the form of solar panels. Therefore, the success of this system is partially impacted by the availability of sunlight. Solar panels must be placed on the helmet in locations that will optimize their efficiency and exposure to light. In the chance that sun is unavailable, an alternative energy source must be used. While the functionality of Head On is impacted by the environment, its creation and use does not directly impact the environment in any significant way.

6.7 Social Constraints

The Head On system must be designed in a way that it can be used on a variety of helmet and motorcycle sizes so that it can be available to all motorcycle riders. For example, the display must have the ability to attach to any motorcycle helmet.

6.8 Political Constraints

There were no political constraints that impacted the design of the Head On system.

6.9 Ethical Constraints

The motivation for creating Head On is to benefit riders who use it. Therefore, it is important to consider the health and safety of those who use the system. Additionally, no components of the system must use patented designs without appropriate permission. This includes both the hardware and software components.

6.10 Legality Constraints

Head On does not have any legality constraints.

7.0 Project Designs

Upon completion of the research phase of this project, the design phase began. This phase consisted of two parts: software design and hardware design. Because the hardware and software aspects rely heavily on each other and due to time constraints, these two parts were completed simultaneously. The hardware design was completed using schematic creation software. On the other hand, the software design was done by writing programs on the Arduino IDE and testing them using an Arduino UNO.

7.1 Hardware Design

In order to make the system less cumbersome it was designed as two separate units that are connected via Bluetooth. One module was positioned on the motorcycle; in order to take advantage of the ample space and power available on, this module housed the major functions of the system. Consequently, the other module, which is mounted on the helmet, was restricted to the bare necessities including a self-contained power supply.

7.1.1 Helmet Module

The helmet module consists of four solar cells, power supply, battery, real time clock, microcontroller, Bluetooth module, and an LCD display. All the components except the solar cells and the LCD screen were integrated onto a single board mounted in a housing module at the lower rear of the helmet. The module has a power switch and two buttons for setting the clock time. The solar cells were affixed in an array at the top of the helmet. The screen was mounted to the helmet using commercially available camera mount designed for use on motorcycle helmets.

7.1.1.1 Power System

The power supply of the system, displayed in Figure 32, is integral for the functionality of the Head On system. This system is fed by four MPT4.8-75 solar panels located on the top of the helmet. In order to minimize power fluctuations common in solar powered systems a filter network was used between the solar panel output and the input of the boost charger. Energy harvested from these solar panels is used to recharge the integrated lithium polymer battery, thus extending the operation time of the system. The onboard power is managed by a Texas Instruments BQ25570 Nano Power Boost Charger and Buck Converter for Energy Harvester Powered Applications. This device integrates solar power regulation, smart battery charging, and output power regulation all into one package. The output voltage and battery charging range are set by three interconnected voltage divider networks. The battery charging output does not require external filtering. The output supply voltage however used a simple inductor and capacitor network to smooth any ripple remaining on the output. Certain features of the BQ25570 are able to be controlled externally but were not necessary for this application. Only one control line was connected to pin 6 of the Atmega328 for the battery status indicator. All the functions and outputs were hardwired to provide the required 3.3V supply voltage.

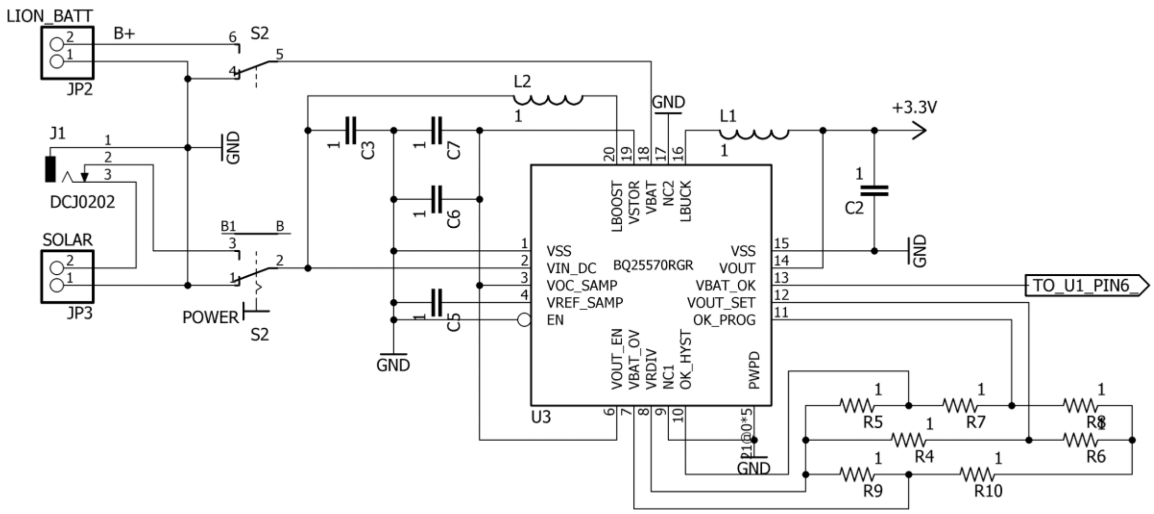


Figure 32: Helmet Module Power Supply Schematic

Due to the current requirements of the different modules going into the system, 100mA, .2mA, and 40mA for the screen, microcontroller, and Bluetooth module respectively, four panels were deemed necessary. The 200mA provided by the four panels will sufficiently cover the requirements of the modules. In order to ensure a maximum output from the solar panels will be achieved, they were placed at the apex of the helmet as shown in Figure 33 and Figure 34. This location will allow the sunlight can hit them more directly. Additionally, it is important to note due to the flexibility of the panels chosen, they were able to mold the curves of the helmet so that they follow the standards for motorcycle helmets.

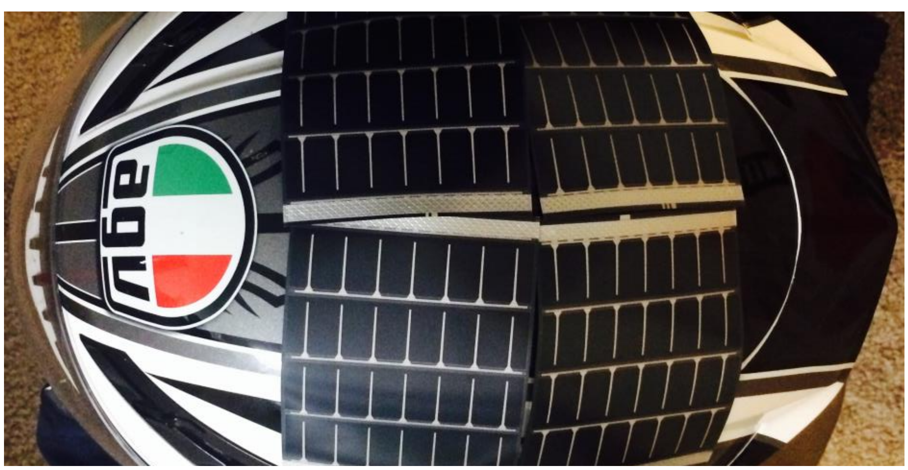


Figure 33: Solar Panel Placement Top View

Figure 34: Solar Panel Placement Side View

7.1.1.2 Control System

The helmet system is controlled by an Atmega328 microcontroller as seen in Figure 35. Its timing is supplied by the 32kHz output from a battery backup real time clock module. The RTC also maintains the display clock time when the system is powered off through the use of a CR2032 watch battery. The controller receives data from the motorcycle module via a HC-05 Bluetooth module connected to a software simulated serial port on pins 4 and 5 of the MCU. This was done to maintain consistency of the wiring and programing between the helmet and motorcycle modules. While the MCU's physical serial port, pins 2 and 3, was available for use on the helmet module, that connection was reserved for the GPS unit on the motorcycle module. The data from the motorcycle module is processed by the MCU and delivered to the screen over a ribbon cable along the outside of the helmet.

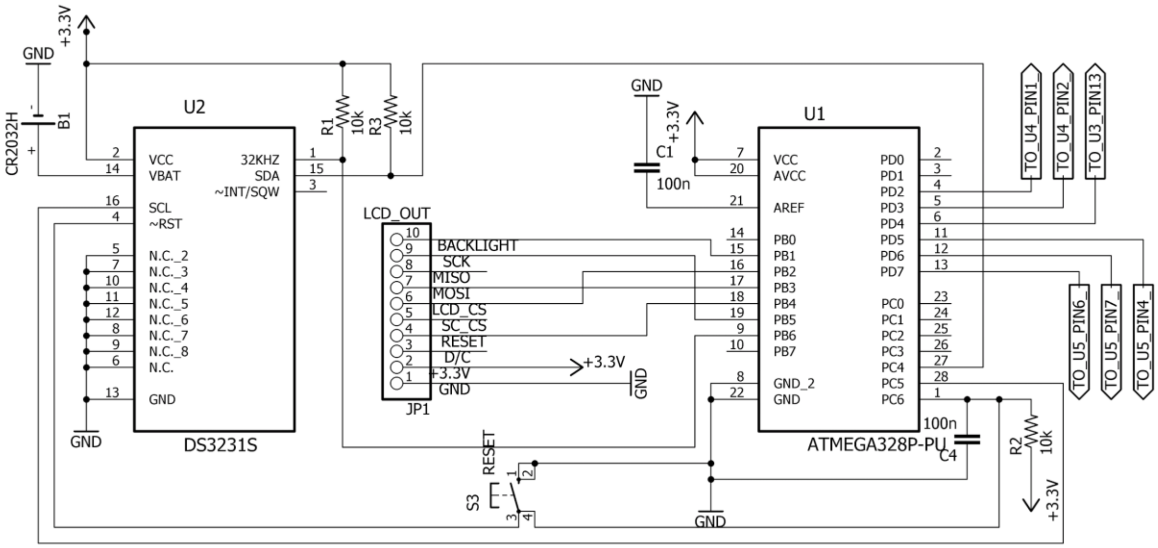


Figure 35: Helmet Module Control System Schematic

7.1.1.3 LCD Screen

The Adafruit 2.2 TFT LCD is mounted outside of the face shield using a modified GoPro helmet mount and connected to the control system via a ribbon cable. If the mount cannot be easily modified or the modifications do not secure the display enough, a new mount will be designed and created using 3D printing. The original design mounted the screen inside the helmet however this proved to be problematic for several reasons. The first concern was the minimum focal distance of the user. The internal mounting of the screen would have placed it too close for the user’s eyes to focus properly. This would have made the system a burden to use and would have hindered rather than enhances the user’s riding experience. Another concern with placing the screen inside of the helmet was finding a location with sufficient room. There is limited space inside the helmet, which would restrict where the screen could be mounted. The screen also could not obstruct the user’s field of view, thus eliminating the majority of the area located directly in front of the user’s face. The other major concern was due to the necessity for actual road testing. While the screen could be secured for normal motion, the safety of the rider could not be assured under all conditions. Consequently, it was decided that the best option is to mount the screen outside the helmet. Doing so will increase the focal distance to the screen and allow the safety features of the helmet to protect the user should he or she encounter unforeseen circumstances. The exact location of the display will be determined after various tests in order to ensure an optimal field of view is achieved. There are standards in place which dictate a minimum field of view necessary for helmets. Special care will be taken to ensure that these constraints are met. Additionally, this position must also allow the reader to easily read the information on the screen.

7.1.1.4 Electronic Compass

The electronic compass module mounts directly to the printed circuit board on the helmet unit. This placement results in a near vertical position of the sensor during use. This shifted the reference axis, displayed in Figure 36, requires the MCU to compensate when analyzing the data.

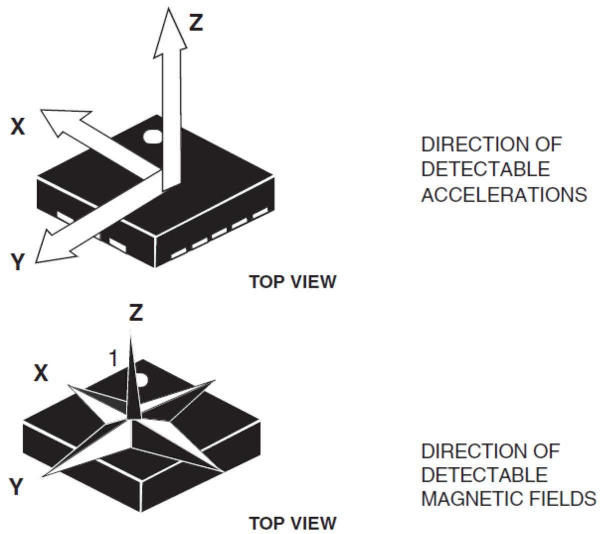


Figure 36: Detection Axis
Requesting Permissions in Progress

Figure 37 illustrates the connections between the MCU on the helmet module and the LSM303D compass module. The first thing to note in this schematic is that the MCU simulates the necessary data ports over three digital input/output lines. Additionally, it is also important to point out that the interrupt connections on the compass module were not needed.

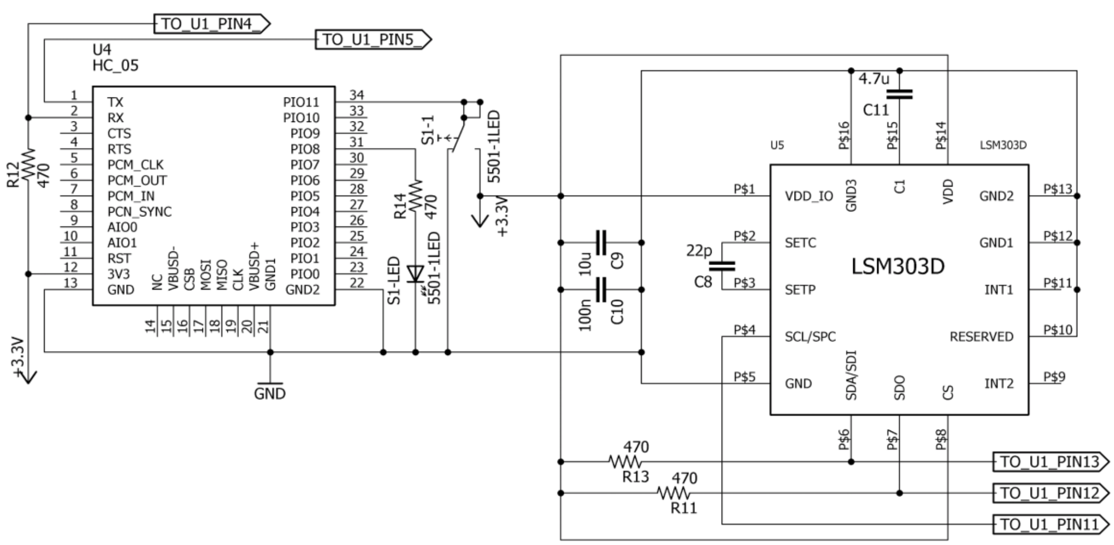


Figure 37: Helmet Module Bluetooth and Compass Schematic

7.1.1.5 Bluetooth Transceiver

As stated in section 7.1.1.2, the Bluetooth module was connected to two digital input/output lines on the MCU, as seen in Figure 37 and the necessary functions were simulated using software. While the final helmet module was designed to be slaved to the motorcycle module a button for manual pairing was also included for development and troubleshooting purposes.

7.1.2 Bike Module

The bike module consists of a dual buck converter, three sensors, a microcontroller, a Bluetooth module, a real time clock and a GPS tracker. All components except for the three sensors will be integrated into one board. This board will then be placed at the tail end of the bike in a location that is non-obstructive to the rider. The three sensors will be pointed to the left, rear, and right of the user in order to detect objects in the blind spots of the rider.

7.1.2.1 Power System

Unlike the helmet system, the bike system will not be powered by solar panels. The entire system will be powered by the existing 12V battery on the bike and the power will be managed by the TPS5492 1.5-A/2.5-A Dual, Fully-Synchronous Buck Converter With Integrated MOSFET by Texas Instruments. The input to the TPS5492 will be the 12V from the battery on the bike and in turn it will provide 3.3V to the real time clock, 5V to the microcontroller, 3.3V to the Bluetooth module, 5V to the sensors, and 3.3V to the GPS. There will be lines running from the TPS5492 to pins 1 and 2 of the real time clock chip, pins 1, 7, and 20 of the microcontroller, pins 12 and 34 of the Bluetooth module, pin 1 of the GPS chip, and pin 1 of the sensors. This configuration can be further explored by looking at Figure 38.

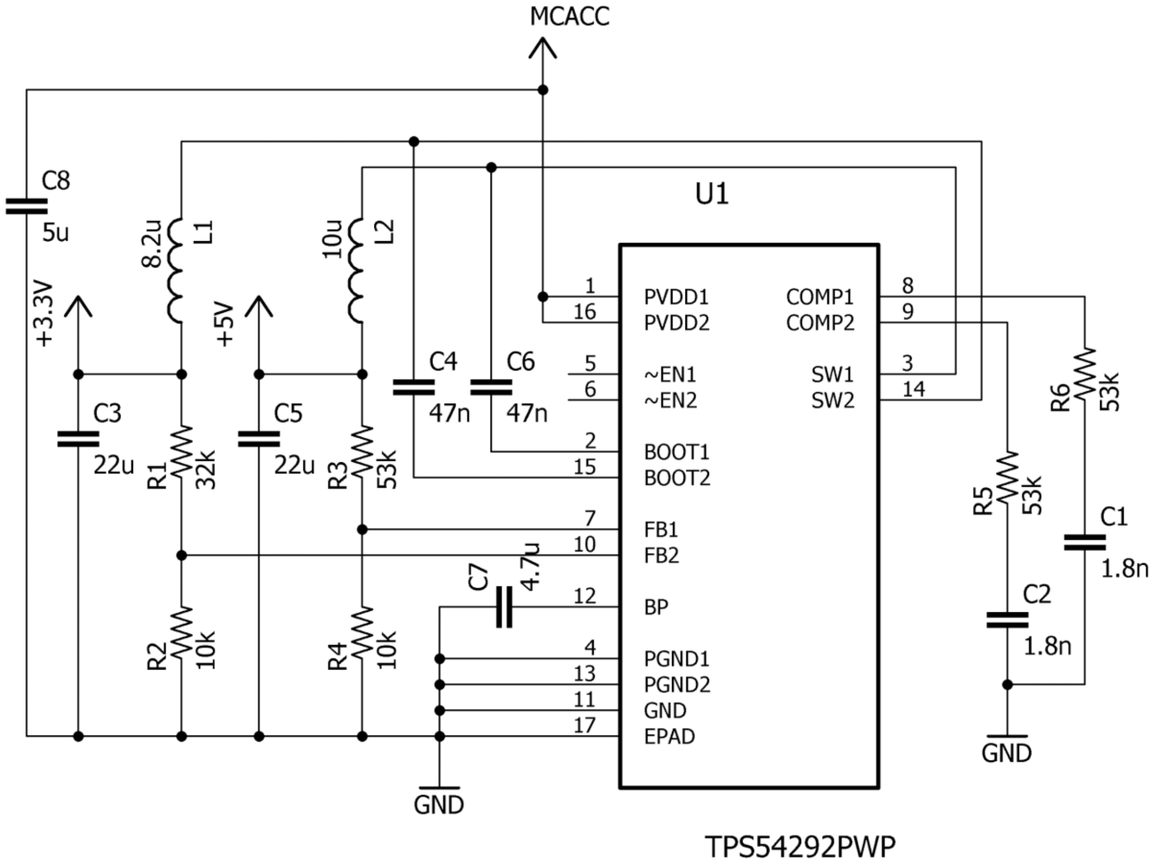


Figure 38: Bike Module Power Supply Schematic

7.1.2.2 Control System

Much like the helmet system, the bike system will be controlled by an Atmega328 microcontroller as shown in Figure [insert number here]. The microcontroller will receive data from the three sensors, process it, and then send it to the Bluetooth module for transmitting to the helmet system. Since the helmet system has a limited power supply and the screen to contend with, it was decided that the microcontroller on the bike would be better suited to interpret the data from the sensors. The microcontroller will be connected to the sensors via pins 14 through 19 and 21, to the GPS via ins 2 and 3, to the Bluetooth module via pins 4 and 5, and lastly to the real time clock via pins 9 and 27. This configuration can be further examined in Figure 39 shown below.

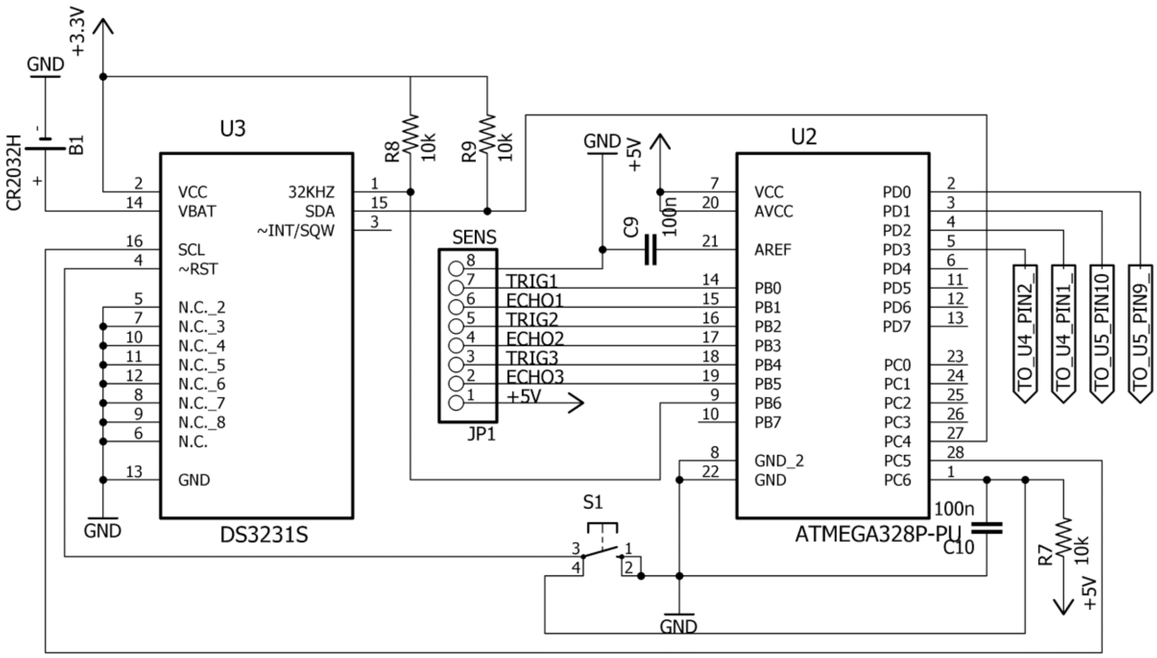


Figure 39: Helmet Module Control System Schematic

7.1.2.3 Global Positioning System

The Global positioning module was incorporated as an autonomous method for the system to provide accurate speed information to the user. The module performs all the necessary GPS functions internally so it requires very few connections to the rest of the system. It provides access to all the collected data via an I^2C serial interface which was connected to the physical serial port at pins 2 and 3 of the MCU through a dampening resistor on each line. To minimize EMI additional filtering of the input power was provided by a ferrite bead on the input line and two capacitors in parallel to ground minimize EMI.

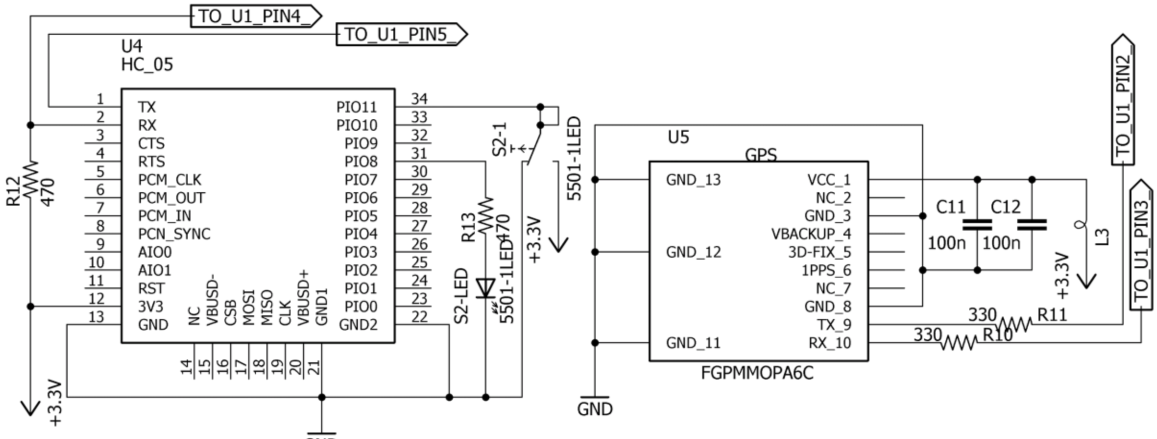


Figure 40: Bike Module Bluetooth and GPS System Schematic

7.1.2.4 Bluetooth Transceiver

The installation of the Bluetooth transceiver in the bike module is virtually identical as described in section 7.1.1.5 of the helmet module and depicted in Figure 40. The only difference in their use is the bike module was set as the master device.

7.1.2.5 Sensor Array

The ultrasonic sensor array is the cornerstone of the system. It provides the critical object distance data to be processed for the user. Each module was connected to the main board using four wires to allow for adjustments to the sensor positions. Each trigger and echo pin from the sensors were connected to separate pins on JP1 and digital I/O pins on the MCU. The 5V supply lines however, were combined at a single pin on JP1 connected to the 5V supply buss. The ground wires were also connected in a similar fashion to the ground buss. The rear facing sensor was placed first and each of the lateral sensors was situated at a 45 degree angle relative to the rear sensor as seen in Figure 41. This angle was sufficient to minimize crosstalk between the rear and lateral sensors yet gave full coverage of the users blind spots.

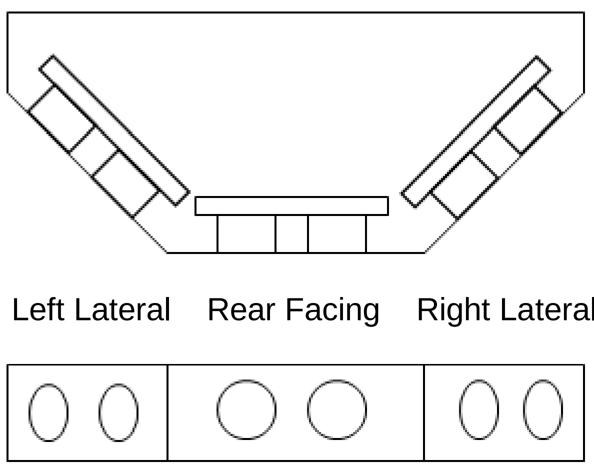


Figure 41: Sensor Placement Representation

7.1.3 Module Housings

The Head On system was designed for use in an outdoor environment which is subject to the elements, temperature extremes, and rough handling. While the prototype housings were not designed to be water resistant, every effort was taken to minimize the impacts of heat and rough handling. The primary focus was to prevent damage and ensure proper operation through the testing and presentation phases of development. Improvements to the modules' housings in order to allow it to survive long term field conditions would need be addressed during design refinement. Clear covers were used in order to allow internal components to be reviewed by project evaluators.

7.1.3.1 Helmet Module Housings

The main circuit board of the helmet module was housed in a generic, rectangular, ABS plastic case with a clear cover. In order to keep the footprint small, the battery was secured under the circuit board using neoprene padding to prevent movement. The case was attached to the back of the helmet with an adhesive to avoid compromising the helmet. The power input from the solar panels was run down the back side of the helmet between the module and the helmet, and fed in through a stress relief grommet at the bottom of the case. A slide switch was placed through the case on the left side for system power. On the opposite side, two push button switches were installed for setting the clock time. No external access to the MCU reset or Bluetooth pairing buttons was provided. The output cable was fed through a stress relief grommet at the bottom side of the case and secured along the lower edge of the helmet. The cable was wrapped around the arm of the GoPro mount to prevent getting caught on the user’s hand and to reduce movement during testing. The cable then runs the length of the screen where it was looped back on itself before connecting to the screen to minimize stress on the solder joints due to movement of the screen. This cable was sandwiched carefully between the screen and its cover. A simple flat cover plate with a neoprene seal around its edge was used to on the back side of the screen to provide a mounting connection and to protect the cable and connections from damage. A generic case was not used because the size required to fit the screen would have significantly increased the display size, thus reducing the user’s visibility. A full custom fit enclosure would have been preferred but the materials and equipment were not available to produce the clear front needed for the screen.

7.1.3.1 Bike Module Housing

The unique layout of the sensor units in the bike module prevented the use of a generic case. A custom case designed to accommodate the sensors and the main circuit board was designed and then 3D printed. An octagonal design was chosen to provide the correct mounting angle for the sensors as well as a symmetrical, aesthetically pleasing design. A simple, clear, flat panel was cut to fit as a cover for the module. To avoid the need to use adhesives on the test vehicle, mounts for both wire tie straps and suction cups were designed into the bottom of the housing. No user interface is necessary with this module as it is powered on and off by the motorcycle ignition switch. The only external connection on the case was the two wire power supply lead. The positive lead was to be run to an accessory power receptacle on the fuse box and the negative lead to an appropriate ground.

7.2 Software Design

When designing the software for Head On, the agile method will be used. This method will allow for changes to be made more easily to the design. Because Head On must be created in only two semesters, it is imperative that any changes can be made quickly and easily. As demonstrated in the Figure 42 below, the design of the product can be changed if the testing, coding, or analysis stages make it necessary. Additionally, the code itself can also be changed easily if the results of testing or analysis of requirements determine that there is a need to.

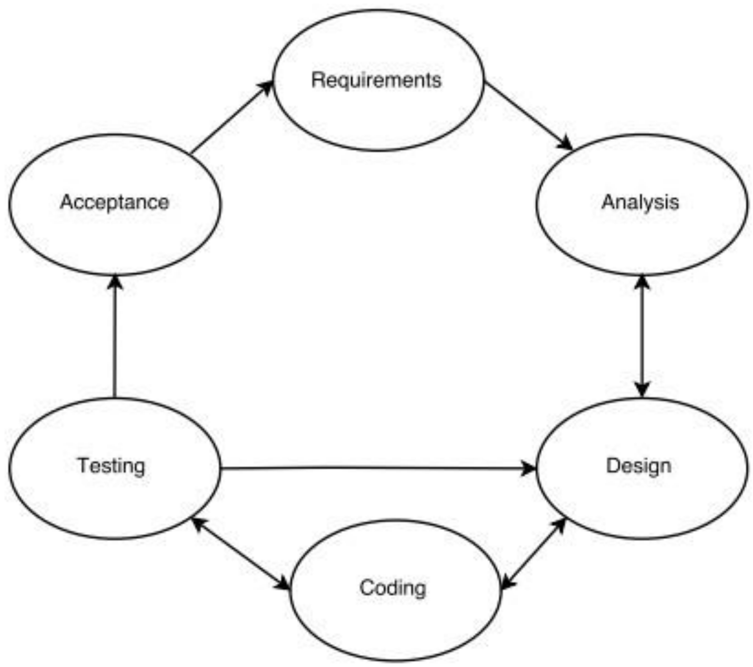


Figure 42: Agile Methodology

The following sections provide both a general and detailed overview of the operation of this system and the interactions of its components.

7.2.1 Helmet System

The modules communicating to the microcontroller in the helmet system include the display, the real time clock, the compass module, and the wireless communication module. While the helmet system does have modules from which it gets data from, the main focus of this system is the analysis of data and the calculation of what changes must occur in the display based on this data. Therefore, since this includes data from the motorcycle and helmet modules, all of these modules will be discussed in the following section.

7.2.1.1 UML Diagrams/Flowcharts

To begin, the MCU on the helmet module must process the sensor data that was transmitted from the MCU on the motorcycle. Figure 44 gives an overview of the process for updating the proximity indicators on the display based on data received from the sensors. Unlike the other modules whose numerical values are displayed on the screen, the proximity sensor indicators only have two modes: on or off. Starting with the left sensor, the analysis phase will commence. For the sake of simplification, the analysis phase in Figure 44 is represented as only a single action instead of a series of steps. Rather, the flowchart in Figure 43 explains this process in detail. In this analysis phase the value will be compared to a certain threshold in order to determine if an object was present or not. If the distance detected by the sensor is below the threshold, a value of 1 will be assigned for this sensor. If the distance was above the threshold, meaning, an object is not next to the rider, then a value of 0 will be assigned. Because this process is the same for all three sensors, it will be written as a function. The function will be called for each sensor and the reading for that sensor will be passed as parameter. Finally, the Boolean value will be returned, depending on if an object was detected, and saved in a variable.

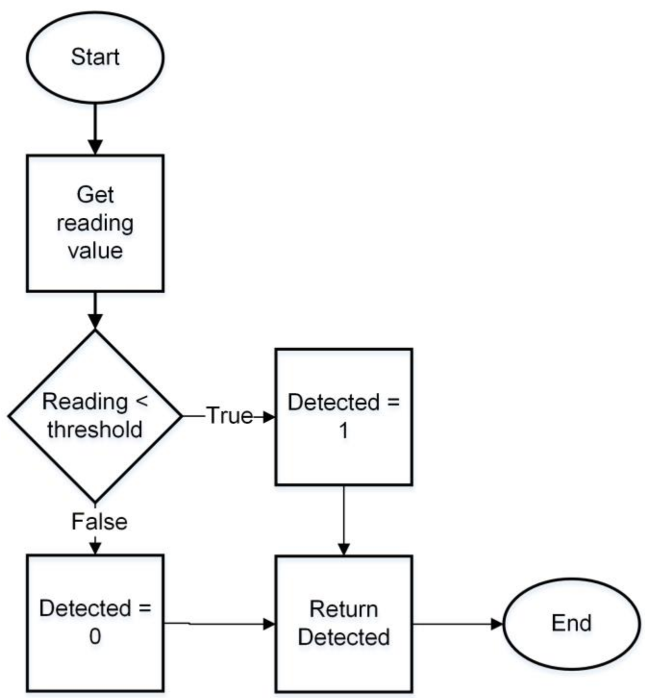


Figure 43: Sensor Analysis Phase

Now that tis analysis phase is complete the indicator updating process can continue. Now, the previous value received, which is also stored in a variable, is compared to the Boolean value that was returned by the function. If they are the same, then this portion of the screen does not need to be updated. However, if they are different, the corresponding proximity indicator must be updated. This means that if an object was detected by this sensor, the proximity indicator will be displayed and if not the indicator will be removed. This process is repeated for the rear and right sensor. Once all three stages are complete, the new values will replace the old values so that they can be used for comparison in the next cycle. Finally, the screen is then updated so that the indicators now reflect the most recent data. Immediately upon completion of the cycle, it will begin again with the newest set of data. Note in the diagram that the routine for each sensor is identical. As a result, a general function will be created that can be used by any sensor. The function will be called once for each sensor within the main program loop; the parameters passed into the function will be the previous value and the current measurement.

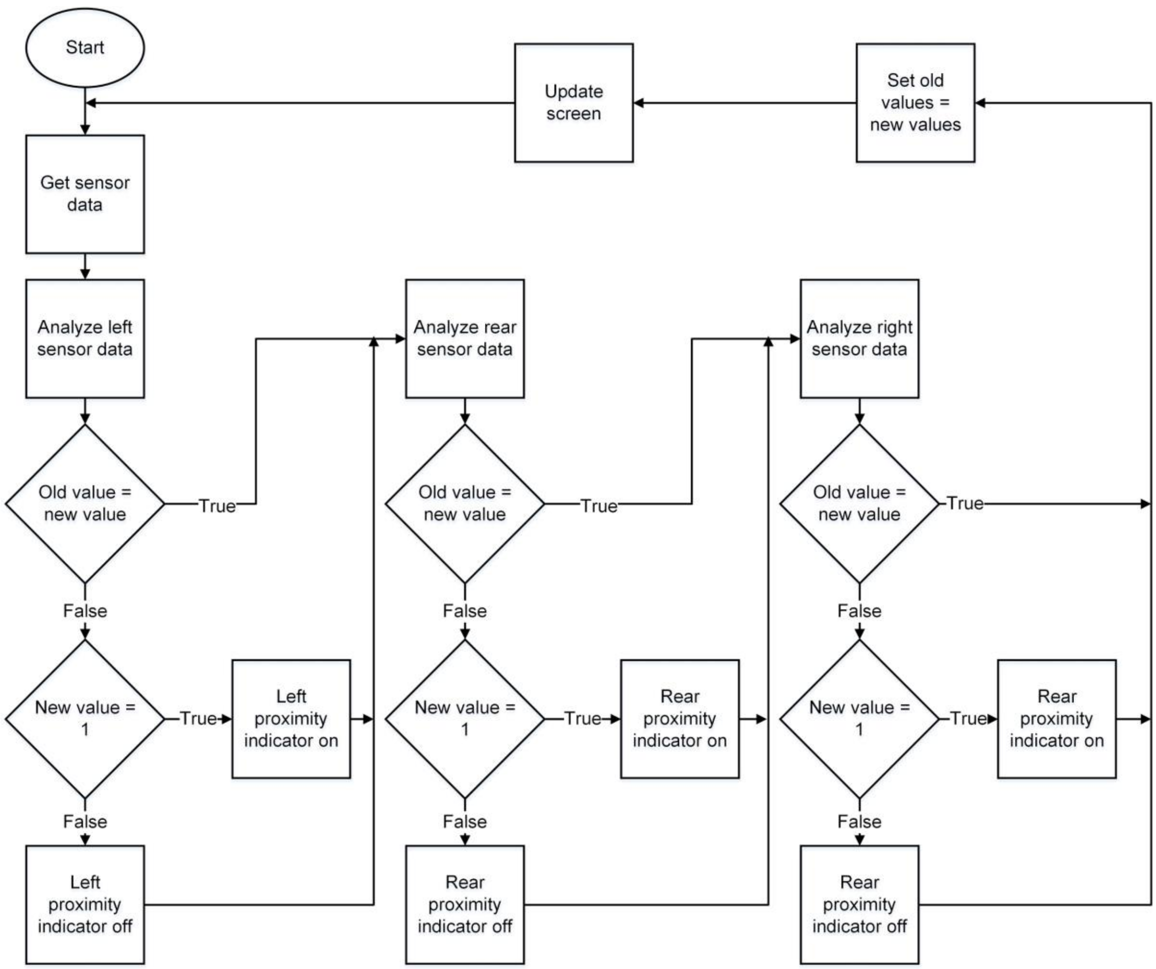


Figure 44: Update Proximity Indicators

The process for updating the speed indicator, explained in Figure 45, is significantly simpler than the other indicators because it contains only one value and a comparison to a threshold does not need to be made first. . The speed will be calculated using the GPS module on the motorcycle system. Once the new speed value is sent to the helmet module, this new value will be compared to the old value which is stored in a variable. As with the other indicators, if there is no difference in value, the indicator does not update; if the values differ the new value is displayed on the screen. Regardless of whether or not the value changes, the unit portion of the indicator will always remain constant. The final step in the cycle is to update the variable holding the old value so that now it holds the value that is displayed on the screen; this will allow the next value measured by the module to be compared to what is on the screen.

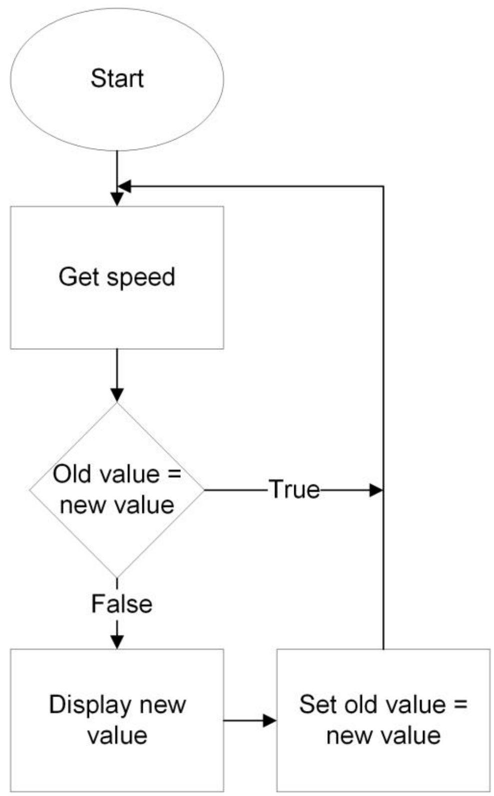


Figure 45: Speed Program

The next software aspect to consider is the compass module, which is located on the helmet system. One thing to note is that because it is located on the helmet system, the resulting direction will be the direction the rider's head is facing rather than the direction the bike is facing. The compass module was chosen to calculate this data rather than the GPS module because, unlike the GPS module, it can determine the direction even if the user is not moving. As shown in Figure 46, a measurement will be taken by the compass module in order to determine the direction the rider is facing. This new measurement will then be compared to the previous measurement. Similar to the sensor process, if the new and old values are the same, the screen will not need to be update. However, if there is a difference, both the value (i.e. N, S, E, W, NE, NW, SE, or SW) and the compass symbol will have to be updated. A series of comparisons will be made to determine what the new value and symbol should be based on the new measurement. Finally, the screen will be updated to represent the change.

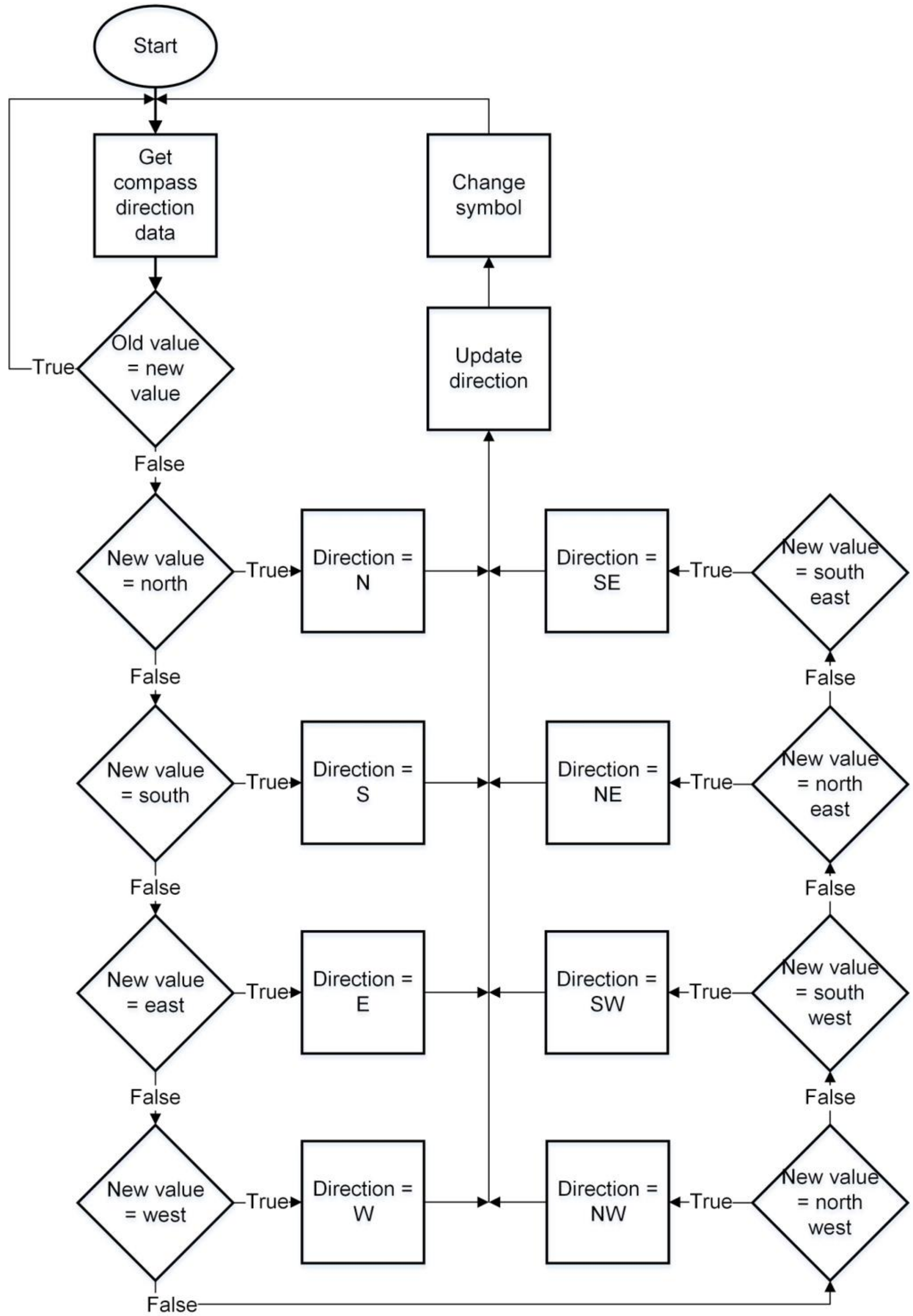


Figure 46: Update GPS Indicator

Finally the last component of the display is the clock. The software process for processing the data from the real time clock module and updating the display is shown below in Figure 47. It will occur about once per second. This starts with the smallest unit of time that will be displayed, the minute. If the minute has not changed yet, then the other larger values do not need to be checked; none of the values on the screen will be updated. If the minute has changed, the next step is to check the hour. If the hour has not changed, only the minute will be updated. If it has been changed, the period will be checked. Because the Head On system will be using the 12-hour clock format, it will use the two periods: AM and PM. If the periods are the same, only the minute and hour will change; otherwise, the minute, hour, and period will change. All the values will be stored in variables so that they can be referenced in the next iteration of the cycle. Finally, the screen will be refreshed so that it shows the new value.

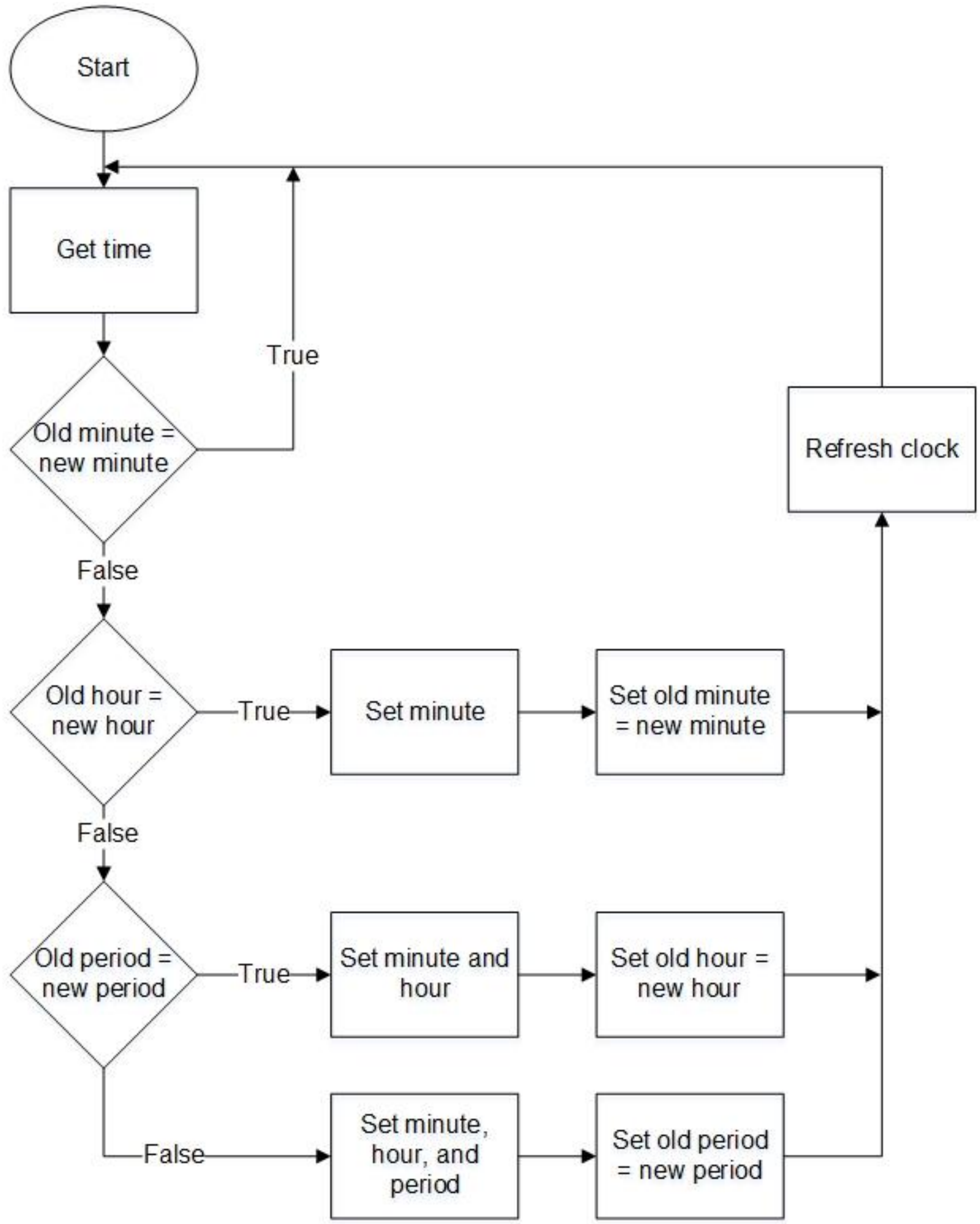


Figure 47: Update Time Indicator

7.2.1.2 GUI Design

When designing the graphical user interface, it was imperative to consider the user experience. The two computer engineering students in this group have taken a course called Human and Technology Interaction. In this class the proper way to design a user interface were analyzed. This design process includes applying the “Eight Golden Rules of Interface Design” as described by Ben Shneiderman and Catherine Plaisant. Figure 48 illustrates the layout of the user interface that was designed after applying each of these rules.

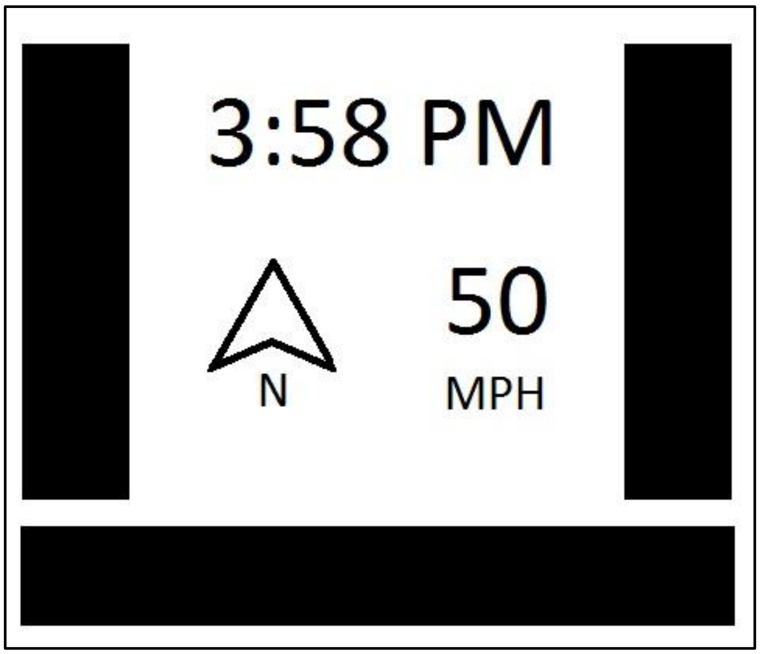


Figure 48: GUI Design

The first rule considered when designing the interface was to strive for consistency. Overall, the user interface is kept consistent by keeping the color scheme uniform. All symbols and text are displayed in black and the background is kept white. Each time the screen refreshes to display the new values, this color scheme remains the same. Additionally, the font type used is kept the same throughout. Finally, the size is one of two sizes. All components of the time indicator and the numerical portion of the speed indicator are written in a larger font. On the other hand, the units of the speed indicator and the text portion of the cardinal direction indicator are both written in a slightly smaller font.

Next, the user interface must cater to universal usability. This means that the system must be designed in such a way that users with varying skill levels are capable of using it. In order to achieve this, the symbols and layout must be easy to understand and interpret. For the time indicator, the 12-hour clock is used as opposed to the 24-hour clock format because, in general, people are more familiar with this format. Similarly, the speed indicator is displayed in miles per hour because this is the common unit of speed when driving in the United States. The cardinal direction value was chosen to match the values typically used on a compass rose. Additionally, the cardinal direction symbol is designed to look like a simplified compass needle. Finally, since there is no standard symbol for proximity indicators, a simple black rectangle will be used. Three were created in order to correspond to the three sensors on the motorcycle.

The next “Golden Rule” to consider is that the system must offer informative feedback to the user. Aside from turning the system on and off, the user does not directly interact with the system. Therefore, when this rule was applied, more focus was put on giving the user feedback based on the data the system gathers rather than actions of the user. When new data is collected from the modules, the updated information will be displayed on the screen for the user in the form of the proximity, time, cardinal direction, and speed indicators. The sensors and modules on the motorcycle and helmet system will be continuously gathering this data, so the screen will also be continuously refreshing to show the user these changes.

As mentioned previously, the current design for the Head On system does not involve options for user interaction with the user interface. Therefore, the next rule of interface design, designing dialogs to yield closure, is not a concern. The user’s interactions with the screen will be very passive. He or she will merely observe the screen as it updates the values.

Additionally, error handling based on interactions from the user is also unnecessary. One of the original ideas for the Head On system was to incorporate more user interaction through the use of buttons or touch screens. However, this increased user interaction could result in errors. Therefore, by restricting the ways the user can directly interact with the system, errors are prevented.

One of the few ways the user can interact with the system is through the power button. The user can shut off the display along with the rest of the system by pressing this button. This system and display can easily be turned back on by pressing the power button again. In a similar fashion, the system can also easily reverse actions. The data that is currently displayed on the screen can be removed when new data is gathered by redrawing the appropriate symbols and values. This ability to update the screen is pivotal to the success of the system because new values are constantly being received from the modules, and the associated indicators on the screen must reflect changes.

When designing the system, it was important to ensure that it supports the user’s internal locus of control. Once again, the user is not initiating the actions so instead the focus shifted to ensuring that the user does not feel overwhelmed by the system. Data gathered from the modules should be made easily available to the user. In essence, this is the entire purpose of the head on system, and specifically, the display module. Unlike seven segment displays, the LCD screen chosen for this project allows the layout to be more free-flowing. Consequently, all of the data can fit on the display. When designing the interface, all of the modules were shifted around until an appropriate layout was discovered. The user can be assured that all of the information from the modules is being displayed on the screen.

Finally, and possibly most importantly, the system must be easy to use in order to reduce short-term memory load. The indicator symbols that were chosen were fairly simplistic and self-explanatory. The proximity indicators are solid bars that are located on the left, bottom, and right of the screen in order to correspond with the left, rear, and right of the user. Both time and speed are written in using numbers. This format can easily be understood by any user. The cardinal direction is written in both text form and symbolic form, using the compass symbol. The user may not be as accustomed to directions as he or she is with time and speed. Therefore, both of these methods of displaying the value were incorporated. Overall, the simplicity of the system will reduce short term memory load of the user because there are no complex symbols that the user will have to learn to interpret.

7.2.1.3 Sensor Analysis and Processing

Using the wireless communication module, the microcontroller on the motorcycle will transmit the readings from the sensors to the helmet system. Once this data is received by the microcontroller on the helmet, it will be processed. The new values will be compared to the old values. If a change is indicated, this change will then be reflected to the user via the display. The microcontroller will send the necessary commands to the display in order to update the screen.

7.2.1.4 Libraries

A critical portion of the design process is picking out libraries that will be necessary for the program to work correctly. This must be done for each module that the helmet and motorcycle system interacts with. The major point of focus in the helmet system is the display. All of the information is displayed to the user must be done so through this display. Without the help of libraries, programming this system would be a daunting task.

7.2.1.4.1 Display Libraries

The first library that will be utilized in the system is Adafruit_ILI9340.h. This library is specifically designed for the Adafruit 2.2” LCD display that has been chosen for this project. This library already has basic colors defined in it. In addition, it provides several functions that will make the process of drawing the user interface much simpler. This includes functions that can be used to draw lines, fill the screen, and draw pixels.

The next library that will be used is the Adafruit_GFX.h. This library offers even more functions that can be used in the drawing process. Specifically, it contains functions that allow you to draw and fill basic shapes such as circles, triangles, and rectangles. For this project, drawRoundRect(int16_t x0, int16_t y0, int16_t w, int16_t h, int16_t radius, uint16_t color) and fillRoundRect(int16_t x0, int16_t y0, int16_t w, int16_t h, int16_t radius, uint16_t color) can be used to draw the proximity sensor indicators that will be placed on the left, right, and bottom of the screen. In addition, the drawTriangle(int16_t x0, int16_t y0, int16_t x1, int16_t y1, int16_t x2, int16_t y2, uint16_t color) and fillTriangle(int16_t x0, int16_t y0, int16_t x1, int16_t y1, int16_t x2, int16_t y2, uint16_t color) can be used to create the compass symbol portion of the cardinal direction indicator. This library also features functions that can be used to display text on the screen. The color, size, and position of the text can all be altered easily using setTextColor(uint16_t c), setTextSize(uint8_t s), and setTextWrap(Boolean w) respectively. This will make displaying the time, speed, and cardinal direction much easier.

7.2.1.4.2 Real Time Clock Libraries

Two libraries will be used to interact with the real time clock. The first of these libraries is Time.h. This library is specific for Arduino and has the functions hour() and minute() that can retrieve the current hour and minute respectively. The hour is returned in the 24 hour format, so it would have to be converted to the twelve hour format and the time period (i.e. AM/PM) would have to be calculated. Additionally, the now() function is available which will also retrieve the current time. The other library that could be used is DS3231RTC.h. This library open source adds additional functionality to the time library, and it is designed specifically for the DS3231RTC real time clock module that is being used in this project. The read() function can be used to obtain the second, minute, hour, weekday, day, month, and year. Not all of these values are needed for the program; however, this would reduce the number of functions that must be called in order to obtain the data.

7.2.1.4.3 Communication Libraries

In order to communicate with the real time module, appropriate libraries must be included. Arduino has to Wire library that can be used to facilitate communication with I^2C devices. Begin() can be used to set the master and slave when initializing the modules. The beginTransmission() and endTransmission() allow for data to be transmitted across the I^2C bus.

In order to allow the display to communicate with the microcontroller, the SPI.h library for Arduino will also be used. The begin(), beginTransaction(), end(), and endTransaction() functions will allow the SPI bus to be initialized and disabled. Once the bus is initialized, data can be transferred from the microcontroller on the helmet system to the display using transfer(). Should it be necessary, the SPI library also allows for interrupts to occur with the function usingInterrupt().

7.2.2 Bike System

The main software aspect of the motorcycle system is receiving the data from the sensors and GPS module and transmitting them using wireless communication to the helmet system. The main focus of the motorcycle system is retrieving data from the modules and transmitting it to the helmet system.

7.2.2.1 UML Diagrams/Flowcharts

Figure 49 explains the logic behind the code that is controlling the sensors. While there are other modules communicating with the microcontroller, as well as other participants, such as the user and the objects being detected, this flowchart does not include them; rather this diagram provides a more in depth look at the interactions between the three sensors and the microcontroller on the motorcycle. At the startup, the echo and trigger pins on the sensors will be initialized. This step will only be executed once, and upon completion, the remainder of the program will run in a loop; this is the portion of the program that will continuously detect whether or not an object is present. Starting with the first sensor, the trigger pin will be set to low, then high, then back to low. The timing of this is significant. Each time the pin should be set to low for approximately two microseconds; this will ensure that the system recognizes the change from low to high, and vice versa. On the other hand, the trigger pin must be set to high for 10μs; this value is specified by the datasheet for the ultrasonic sensor used in this project. This pattern causes the sensor to send out a signal which will be used to detect any objects. At this point, the program will then send a signal to the echo pin so that the sensor can then detect if the signal bounced off of anything. This process is then repeated for sensors 2 and 3. Once all three sensors have completed the cycle, it will begin again with sensor 1. One significant thing to note is that for each sensor, the process of detecting an object is identical. Consequently, when programming this portion of the cycle, a general function can be used. The trigger and echo pins, which will be defined in the initialization of the program, will be passed as parameters of the function.

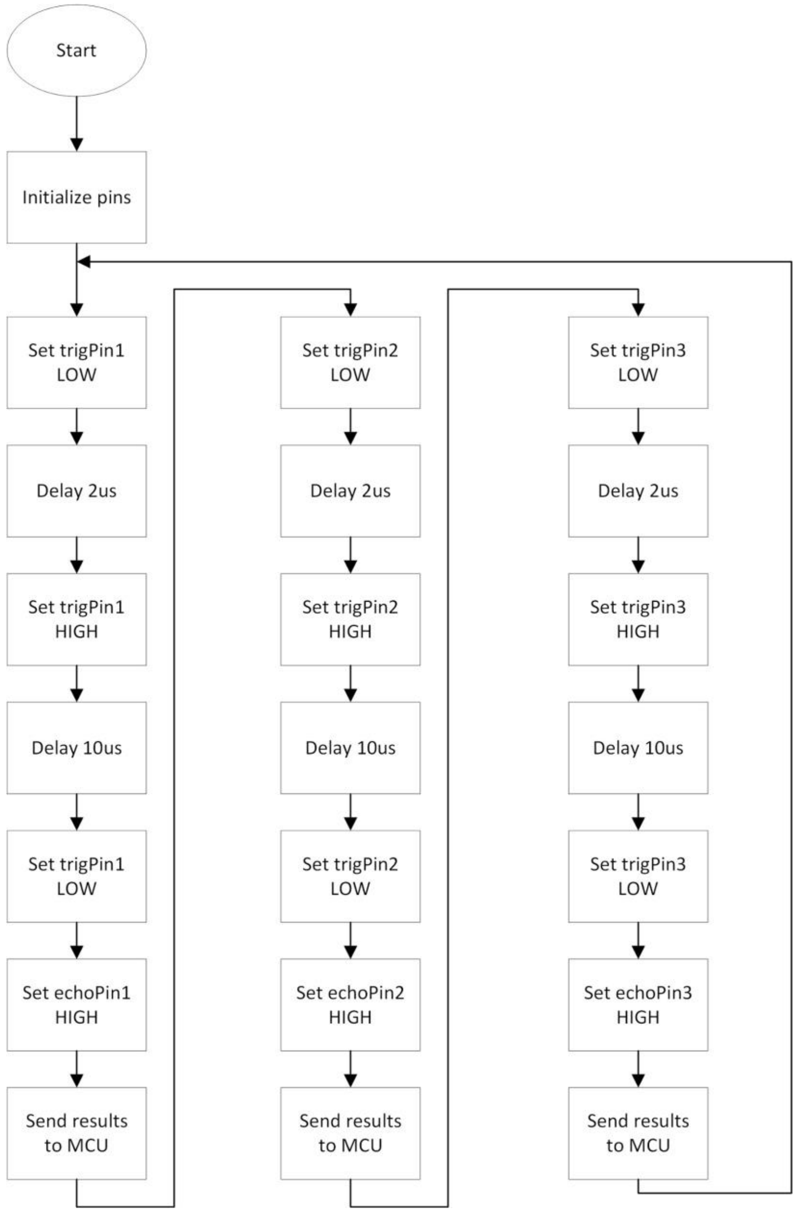


Figure 49: Sensor Program

Now that the in depth process occurring between the microcontroller and sensors has been analyzed, the other external “actors” in the system can be analyzed. The sequence diagram in Figure 50 takes into account not only the microcontroller on the motorcycle and the three sensors, but also the rider, the microcontroller on the helmet, the display, and the surroundings. The first actor to consider is the user; after turning the system on, which will only occur once, the user’s only interaction with the system is viewing the data displayed on the screen. The rest of the interactions in this object detection process are between the components of the system and the environment that it is being used in. As explained in detail in the previous diagram, the majority of the actions in this process occur between the microcontroller on the motorcycle and the sensors. For each sensor, a signal is sent so the sensor will send a response, then another is sent to allow the sensor to detect a response. If an obstacle is present, the signal that is sent out into the surroundings will bounce off the object back to the sensor. Regardless of the results obtained, the data is always sent back to the motorcycle module, and is then transmitted to the helmet module. Upon arrival in the helmet module, the data will be analyzed and the display will be updated accordingly. At this point, the user will come back into play, and he or she will be able to see any changes in the display. Though this process does appear lengthy, the interactions between the sensors and the microcontroller occur in a few microseconds. Therefore, from the point of view of the user, the updates displayed on the screen occur in real time.

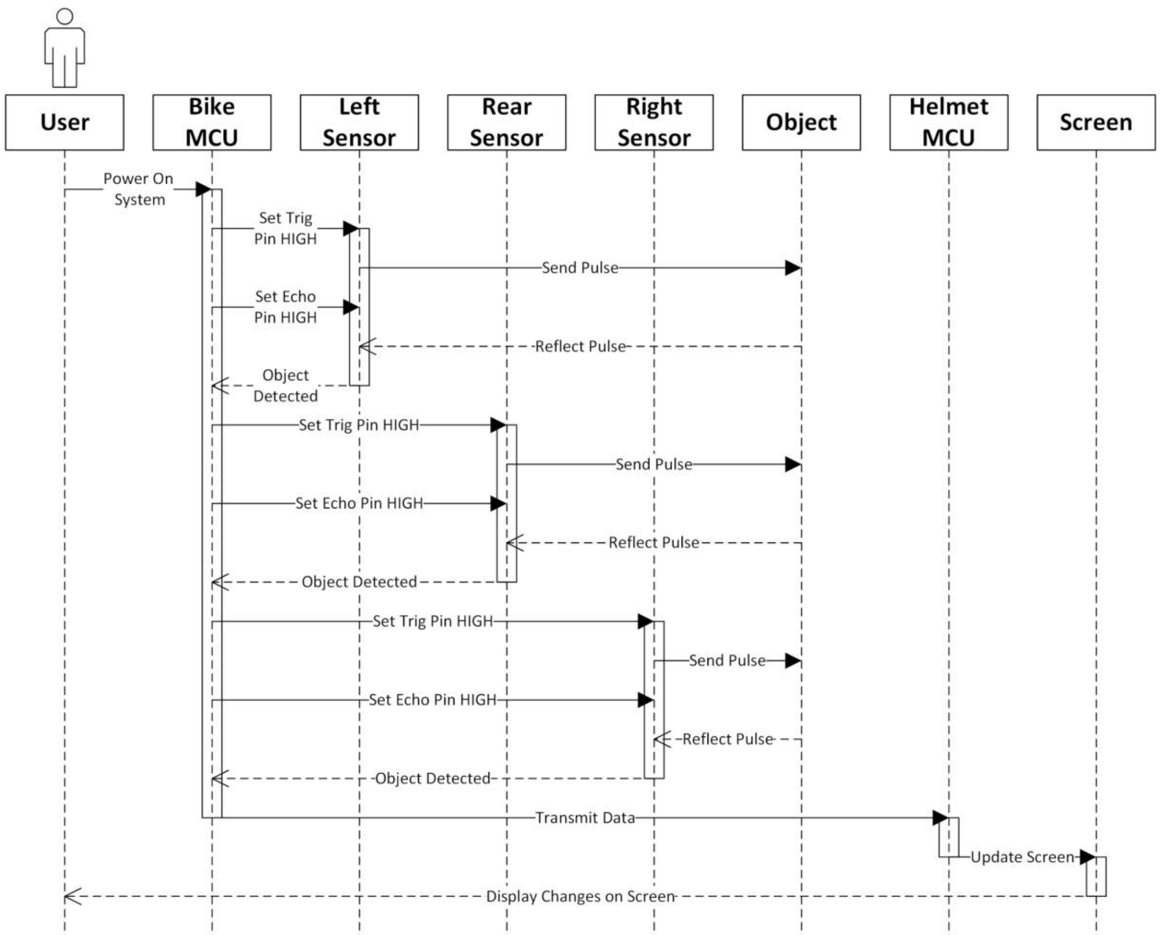


Figure 50: Sensor Sequence Diagram

The other module that the MCU on the motorcycle must communicate with is the GPS module. As outlined below in the sequence diagram in Figure 51, the GPS module is used to obtain the speed of the rider. This process is extremely simple. The microcontroller on the motorcycle begins by calling the appropriate function from the Adafruit GPS library to request the speed. The GPS module then responds by returning this speed value. Finally, this value is transmitted wirelessly to the microcontroller on the helmet system.

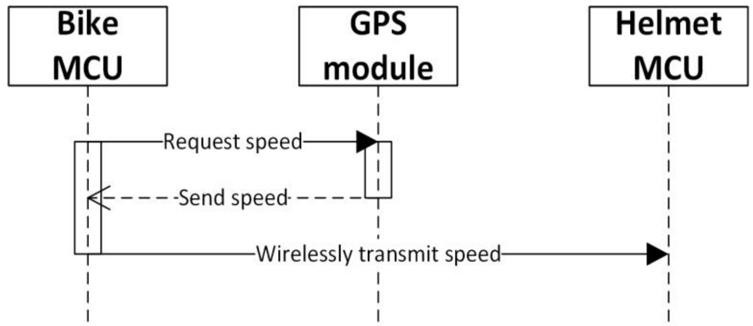


Figure 51: GPS Sequence Diagram

7.2.2.2 Algorithms

The main algorithm of the motorcycle helmet system is used to detect calculate the distance of an object. The distance of the object is not automatically calculated by the sensor; rather, the value returned is the amount of time, in microseconds, that occurred between the sending of the signal and the return of the echo. This value can then calculate the distance to the object in either centimeters or inches using the following formulas:

Distance (cm) = time/58

Distance (in) = time/148

For the purposes of testing, the distance was calculated in centimeters. Because centimeters are smaller than inches, it made testing the accuracy of the sensors more accurate. For each sensor, the MCU on the motorcycle will convert the value it receives from the sensor into cm then send it to the helmet system.

7.2.2.3 Sensor Analysis and Processing

The code for the sensors will consist of two main parts: taking the measurement and analyzing the readings. When taking the measurement, each sensor must take its measurement individually. Figure 52 below shows the process for using the echo and trigger pins on the HC-SR04 ultrasonic sensor. The microcontroller must send a trigger signal to the current ultrasonic sensor for at least ten microseconds in order to indicate that the sensor should produce an ultrasonic signal. At this point, the program will tell the sensor to read the echo pin to determine if the signal was returned. If the echo pin reads high then there is an object within range; if not, there is no object. This cycle will be executed continuously for all three sensors.

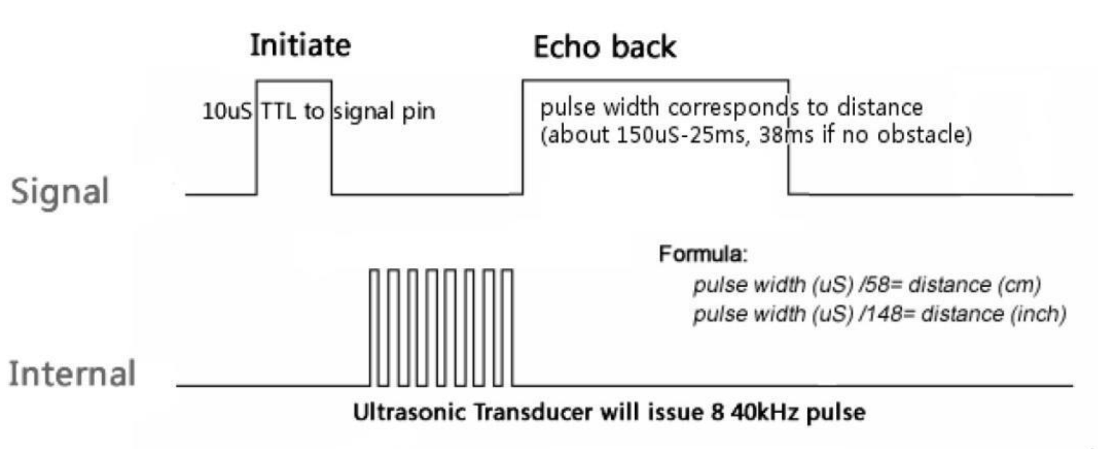


Figure 52: Trigger and Echo of the HC-SR04
Requesting permission in Progress

7.2.2.4 Libraries

The motorcycle will use some of the same libraries as the helmet system; however it will also have to have some libraries that were not included in the helmet system. Specifically, certain modules, such as the GPS module, will require special libraries in order to simplify the programming process.

7.2.2.4.1 GPS Libraries

For the GPS module that was chosen for this project, Adafruit has provided the Adafruit_GPS.h library. The function in this library that is the most pivotal for this project is the parse function; it takes in the data from the GPS module and uses parsing to convert it into a more understandable and usable form.

7.2.3 Overall System

Because the helmet and motorcycle systems work together, it is also important to view the system as a whole. This includes how the two systems communicate with each other, as well as how the overall system works.

7.2.3.1 UML Diagrams/Flowcharts

The following sequence diagram in Figure 53 gives a broad overview of how the various modules will transmit data to the microcontrollers and display this data to the user. The microcontroller on the helmet system will receive information from the ultrasonic sensors and GPS module. Simultaneously, the microcontroller on the helmet system will be receiving the data from the real time clock module and electronic compass modules. It is important to note that the helmet system will finish retrieving data before the motorcycle system does. As a result, there will be no conflict when the motorcycle module transmits its data to the helmet system using the wireless communication modules. Additionally, this method will reduce the wait time for the screen to update and the amount of time the microcontroller is not in use. When all the data has been received, the microcontroller on the helmet will send the necessary changes to the display, and these changes will be updated for the user to see.

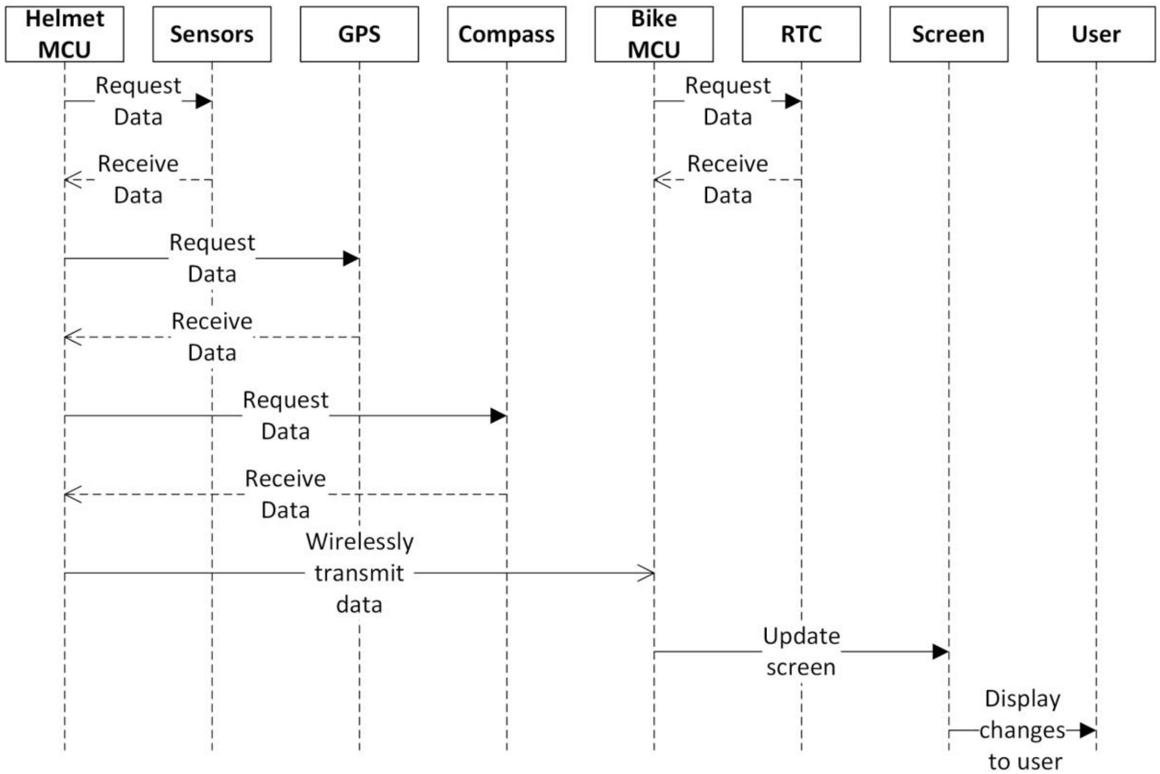


Figure 53: Head On System Sequence Diagram

8.0 Prototype Construction and Coding

The Arduino Uno will be used initially when creating the prototype because it contains the same microcontroller as the Head On system. This will allow the software to be created and tested without having to wait for the printed circuit board to be completed. Modules will be connected to the development board using a breadboard, jumper wires, and if necessary header pins. The code for these modules will then be written and tested. Once the accuracy of the code has been determined, it will then be uploaded into the helmet and motorcycle modules. These modules will consist of the PCB boards and the same modules tested on the development boards.

Simultaneously, the printed circuit boards will also be created. PBC diagrams and schematics will be created for the helmet and motorcycle systems. These diagrams will then be sent to OSH Park. Once the complete circuit boards arrive, the prototype will be built. All equipment used will come from UCF facilities such as the Texas Instruments Innovation Lab and the Senior Design Lab. All the necessary modules will be placed on the circuit boards using the surface mount soldering system and reflow oven in the Senior Design Lab. Housing units will be 3D printed using equipment from the Texas Instruments Innovation Lab. Each circuit board will then be placed inside of their unique housing units in order to protect them from the environment. The Bluetooth connection will be set between the two units so that they can communicate with each other. Finally the power system will be set up. The solar panels will be placed at optimal positions on top of the motorcycle helmet. These solar panels will be connected in order to produce the necessary power for the battery which is also on the helmet system. On the other hand, the motorcycle system will be powered by the motorcycle battery itself.

8.1 Master Parts List

After considering a variety of parts for each part of the project, a list of final parts was determined. These selected modules are displayed in Table 13. These parts will be incorporated into the final prototype of Head On.

Table 13: Master Parts List

Type	**Module Chosen**
Sensor	HC-SR04
Screen	Adafruit 2.2" 18-bit color TFT LCD display
Microcontroller	ATmega328
Wireless Communication	HC-05
Lithium Polymer Battery	585460
Energy Harvesting System	BQ25570
Solar Panel	MPT4.8-75
Boost Converter	TPS54292
Real Time Clock	DS3231
Electronic compass	LSM303D
Global Positioning System	PA6C LadyBird-3

8.2 Facilities and Equipment

All specialized facilities and equipment utilized were provided by the University of Central Florida College of Engineering and Computer Science. 3D printing was performed in the Texas Instruments Innovation Lab in Engineering II. Surface mount soldering was accomplished using the surface mount soldering system and reflow oven in the Senior Design Lab in Engineering I. Additionally, the electronic test equipment in the Senior Design Lab and the Smart Lab in the Harris Engineering Corporation building, was used for electrical testing and troubleshooting. No other specialized equipment or facilities were used in designing Head On.

8.3 PCB Vendor and Assembly

When considering which PCB vendor to send our design to there were several factors that needed to be taken into consideration. To begin with, the time it would take to get the printed circuit boards back. Since this design is time sensitive, it is important to allow plenty of time for the board to be made and sent out, for the design team to test it, and for any potential problems to be addressed. Furthermore, the price needed to be taken into account. More than one PCB will be ordered in case one of them is faulty. This will allow the design team to pinpoint whether any potential faults are due to the design or the actual printed circuit board. Hence, it is important for the boards to be cost effective since more than one will be ordered.

Highly recommended by other students, the first vendor considered was OSH Park. For starters, the file submission type for OSH Park is Eagle CAD. Since that is the same software being used to design the printed circuit board for this project it would make for an easy transition. Furthermore, OSH Park’s most basic order includes three copies of the design sent in. As mentioned above, more than one board will be ordered, so OSH Park’s policy of sending three copies with every order would fit right in with that idea. However, it should also be noted that OSH Park requires that every single order sent in is for quantities in multiples of three. Meaning, more boards need to be ordered then it would have to be six total and so on. For the standard two layer order OSH Park’s price is five dollars per square inch. Since this design already has size constraints, the total for the circuit board would not be overly expensive.

Another vendor considered was Advanced Circuits. They are the third largest PCB manufacturers in North America, which lends credibility to their products. One of the features they have is a special program for engineering students who wish to order from them. They provide free PCB layout software called “PCB Artist” and give discounts to students on the pricing. There is no minimum number of boards that need to be ordered and no restrictions on how many should be ordered at a time. However, the price for each two layer printed circuit board is thirty- three dollars. Despite the fact that the boards would probably be of better quality, since more than one board would be ordered, Advanced Circuits would likely be too far outside the budget.

The last vendor considered for the printed circuit boards was Bay Area Circuits. Known for printing boards at a fairly inexpensive price, they are well known amongst students and hobbyists alike. Much like Advanced Circuits they have a special student rate that can be accessed by registering with a university email. The deal for students is thirty dollars for a two layer printed circuit board. Although this is still pretty expensive, it is important to note that the turn time is five days. Should a circuit board be needed in an extremely timely manner, Bay Area Circuits would likely be the best choice.

Although not recommended, a final option would be to create a printed circuit board and simply do away with vendors all together. However, this would increase the possibilities of making mistakes that could derail the design. Based on the above reasoning, the best choice for procuring the printed circuit boards for the project is to choose OSH Park as the vendor. They are by far the most inexpensive and their policy of sending three copies at once would be useful for testing purposes. However, since the turn day for OSH Park is 12 days, if time were to become an issue, then either Advanced Circuits or Bay Area Circuits would be the best choice.

8.4 Final Coding Plan

Two sets of code will be created: one for the helmet system and the other for the motorcycle system. Once the codes have been tested using the Arduino Uno and it is confirmed that they work properly, they will then be uploaded into the microcontrollers on the two systems. The code for the motorcycle system will focus on processing the sensors, GPS module, and electronic compass module; additionally, it will also include code to transmit this data to the helmet system using the wireless communication. The code for the helmet system will focus on taking this data received, as well as the data it gathers from the real time clock module and making decisions as to how the display must be updated. The code for the helmet system must be able to call the appropriate functions in order to update the LCD display with the newest information.

9.0 Project Operation

The following sections acts as a user manual for the basic operations of the Head On system. This includes explanations on how to change between power modes, how to charge the system using either charging method, and how to understand the user interface.

9.1 On Mode, Sleep Mode, Off Mode

This system has three power modes. The first of these is on mode. This mode consumes the most power. It occurs when the system is transmitting and receiving data. The sensors, GPS, and other modules are all actively taking measurements and the microcontrollers are receiving them. This mode can be initialized by pressing the push button on the helmet module. In order to use the system, the rider must put the system in on mode. The second mode is sleep mode. This mode occurs when the system is not in use. If the rider remains in one location and there is no change in objects around the rider, the system will go into sleep mode. This system uses less power than on mode, but more than off mode. The final mode is off mode. In this mode, the system turns off completely. This mode can be activated by pressing the push button when the system is on. In this mode, the system is using the least power. When the system is not in use the user should put the system into off mode in order to conserve battery power.

9.2 Charging

There are two methods for the user to charge this system. The first is through the removable battery in the helmet system. In order to charge this battery there will be a barrel jack for the user to connect to. Much like with older cell phones, the user simply has to plug in a charger with a barrel connector to a regular wall plug and connect it to the system. The second method requires no actions from the user. The helmet system will automatically charge when the solar panels are in the presence of sunlight. Similar to the battery charging method, the system does not have to be powered on in order to charge using the solar panels. The charging rate using the solar panels will vary depending on how much sunlight is available.

9.3 User Interface

When designing the user interface, the top priority was ease of use. Efforts were focused on making sure the rider would be able to glance at the display at any moment in time and quickly decipher all the data gather at the moment. Furthermore, the symbol designs must be easy to understand and the display must not be too cluttered. The placement, size, and design of the symbols were chosen with the user experience in mind. Figure 54 shows an initial proposed sketch for the user interface in Head On.

In the center, information on the rider is contained. To be specific, the center of the display contains the current time, cardinal direction, and speed in miles per hour. All the information in the center is displayed in large font for improved clarity and readability. Additionally, arrow designs are implemented into the cardinal direction to complement the cardinal value displayed.

On the sides and lower edge of the display, proximity bars are added. Each of the proximity bars' placement on the screen corresponds with the placement on the motorcycle such as left, right, and rear facing. These proximity bars are dependent on the information retrieved from the ultrasonic sensors. For example, at all times that the rider is at a far enough distance from any object according to the logic used for the sensor, the corresponding bar will not be displayed. In contrast, when as an object approaches into the sensor's operating range, the bars will dynamically increase according to the distance it detects.

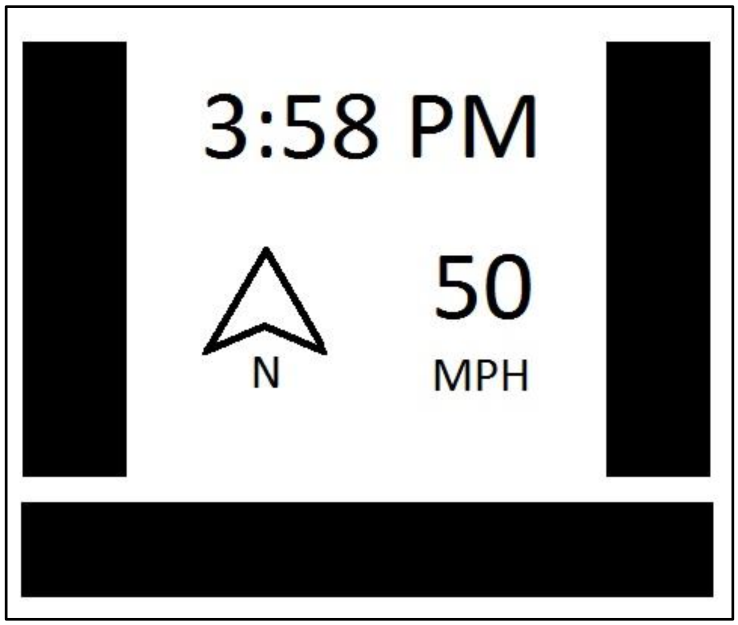


Figure 54: User Interface Layout

Overall, efforts are placed into making the Head On display easy for the user to use while providing useful and possibly lifesaving, information during use. Users should be able to glance at the display to see the time, their cardinal direction, speed, and any possible dangers entering his or her vicinity without distracting the rider.

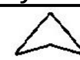
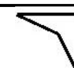
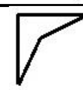
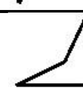
9.3.1 Speed Indicators

The speed of the user can be found in the center of the screen, below the time and to the right of the direction indicator. This indicator consists of two parts. The first is the numerical value of the speed, and the second is the units that the speed is being measured in. This value displayed will be the value that is measured by the GPS module on the motorcycle.

9.3.2 Cardinal Direction Indicator

The cardinal direction indicator is also located in the center of the screen. It also consists of two components. The first is the compass symbol that will indicate the direction. Next is the value of the direction represented by N, E, S, W, NE, NW, SE, and SW corresponding to the four cardinal and four intermediate compass directions. Table 14 summarizes the directions and their corresponding symbol. Similar to the speed indicator, this value will be measured by the GPS module found on the motorcycle. However, the electronic compass module will also be used for when the rider is not moving.

Table 14: Cardinal Direction Values

Direction	Letter Representation	Symbol
North	N	
East	E	
South	S	
West	W	
Northeast	NE	
Northwest	NW	
Southeast	SE	
Southwest	SW	

9.3.3 Time Indicator

The time indicator on the display is located on the top of the screen. The time is displayed in standard 12-hour clock form with the hour and minute, followed by the period, either AM or PM. Because of the real time clock module utilized in the design, the time displayed on the screen will have an accuracy of +/- 2 minutes/year.

9.3.4 Proximity Sensor Indicator

The proximity sensor indicators are displayed as solid black rectangles, and can be found on the left, right, and bottom of the screen. These bars indicate the location of potential hazard. The left, right, and bottom indicators correspond to the left, right, and rear sensors respectively. These bars are only displayed on the screen when a hazard is present. In the event that more than one sensor detects a hazard, the appropriate combination of indicators will be displayed.

10.0 Prototype Testing

In order to confirm that Head On does not have any software or hardware problems, a series of tests will be run on each component. In order to ensure that the testing is thorough, a wide variety of scenarios will be implemented. After testing, the results will be analyzed and any areas of concern will be addressed. If necessary, changes will be made to the design. This process of testing will then be repeated until the results are satisfactory and all of the desired goals, requirements, and product specifications developed for this project have been met.

This process is made possible by the use of the agile method. During any stage of this process, if the need arises to make changes to the design of the product, they can easily be implemented without having to start from scratch. Therefore, when the prototype is tested the issues that are discovered can be dealt with in a more timely fashion.

10.1 Testing Environment

Head On will be tested in two types of environments. First it will be tested in a lab setting in order to ensure that the software is error free. Once it is confirmed that the software errors have been resolved, the prototype will then be tested on the road. This will allow real life scenarios to be encountered and troubleshot. The following real life scenarios will be tested:

a. Daylight
b. Night
c. Low traffic
d. High traffic
e. Low speeds
f. High speeds

10.1.1 Software Testing Environment

The Arduino UNO uses the ATmega328 microcontroller. Since this is the same microcontroller as the prototype, this development board will be used to test code that is running the helmet and motorcycle module. Rather than testing the software on the prototype, it will be tested on the development board first. This will help isolate any errors in the software, rather than trying to determine problems with the software and hardware simultaneously.

10.1.2 Hardware Testing Environment

Once the software has been tested, the hardware components of the prototype can then be tested. First the components will be individually placed on a breadboard and connected to the microcontroller to ensure that the components are getting the necessary ranges. This testing will be done as soon as the parts are purchased so that time is not wasted building a prototype with insufficient parts.

After the success of the components is verified, the prototype will be developed and the system will be tested in a real life environment. The helmet and motorcycle module will be used by a rider when driving a motorcycle in various scenarios.

The first two testing scenarios will focus on the ability of the user to use the system with or without light. This will ensure that the rider can use the system both during the day and at night. The primary component that will be focused on will be the screen since that is how the user receives information from the system. The first scenario that will be tested is the use of the system in daylight. The main purpose of testing in this environment is to ensure that glare from the sun will not impact the user’s ability to read the screen. Next the system will be tested at night. This is to ensure that the backlight of the LCD screen is sufficient enough to allow the screen to be visible when there is very little light.

The next two testing scenarios will focus on the how well the sensors can detect and process objects. Initially the system will be tested in low traffic. Because there will be less hazards around the rider, the sensors will be triggered less often. This will allow us to ensure that the system is not getting false positives. Once it is determined that the system works properly in low traffic, it will then be tested in high traffic. These test results will be used to confirm that the system is capable of detecting several objects at once and communicating this to the screen.

The final two scenarios under consideration will focus on testing the features of the device that rely on the GPS module; this includes the cardinal direction and speed. To start, the system will be first tested at low speeds. This will ensure that the cardinal direction and speed are accurate. Then once the results for this environment are satisfactory, the system will be tested at high speeds. This will test how quickly the system will react. When accelerating to a higher speed, the speed indicator on the screen should also be changing. In addition, if the rider changes direction at high speeds, the cardinal direction indicator should also change accordingly.

10.2 Testing Procedure

In order to determine that the final prototype is fully functional, both the hardware and software must be tested for each component of the system as well as the overall functionality of the system. Any errors discovered during the testing phase will result in hardware and software analysis as well as possible design modifications.

10.2.1 Unit Testing

Each component must be tested individually to ensure that it is working properly. This will allow errors to be pinpointed to a particular module or portion of the program. This testing will occur in both the lab setting and the real life scenarios described previously.

10.2.1.1 Wireless Communication Testing

Successful wireless communication is critical to this application as it is the linking factor between the helmet and motorcycle unit. As with all other components of Head On, the Bluetooth modules had to be reviewed for failure in hardware or data loss during communication. As a part of the testing procedure of the wireless communication, an Arduino Uno was used as it incorporates the ATmega 328 microcontroller.

To begin, one of the HC-05 Bluetooth modules was installed on the Arduino Uno, which was powered by an external 5 V power supply. The wiring from the Arduino Uno to the HC-05 is illustrated in Figure 55. Once the wiring was done, testing was done through a loopback test. By definition, a loopback test is a test in which a signal is sent from a communications device and sent back (loopback) to determine device functionality. For loopback testing on the HC-05, an additional wire was needed. Specifically, a wire was needed to short circuit the Rx and Tx pins in such a way the receiver and sender are connected directly. From here, the terminal was used to verify proper functionality of data transfer.

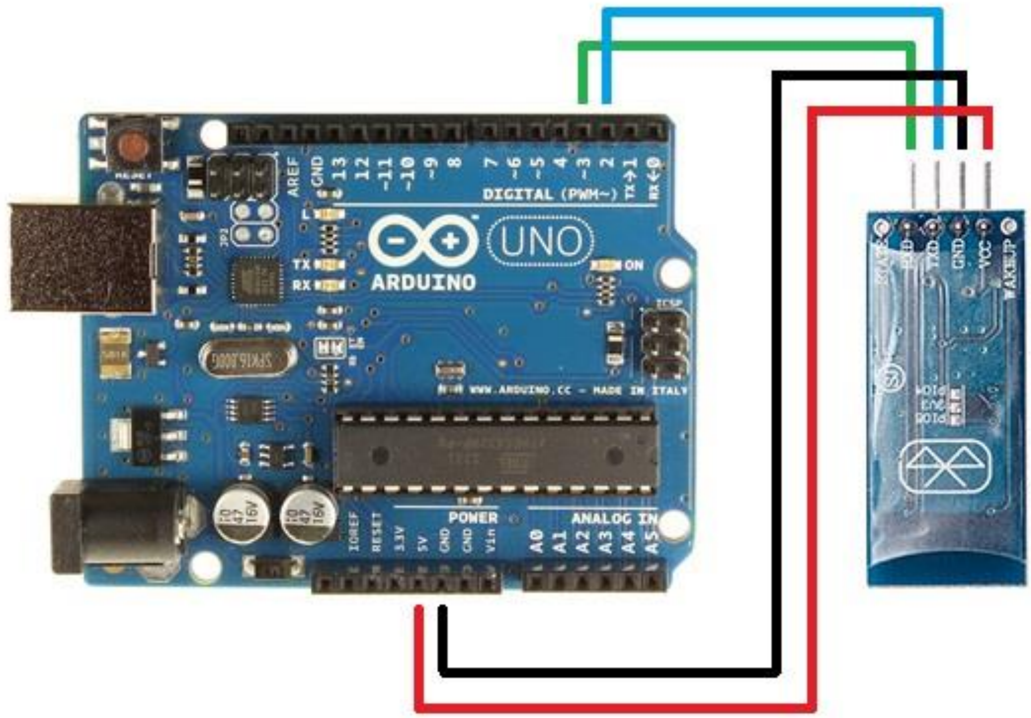


Figure 55: Wireless Communication Testing Layout
Requesting permission in progress

Next, the Arduino Uno was connected to a PC through USB and a new terminal instance was used to verify input and output. Before testing the Uno with HC-05, a new sketch or program was uploaded with the intent of sending a signal and receiving the same signal in return. In specific, the sketch was designed to incorporate a while loop that would transmit an arbitrary message every 2 seconds. For testing purposes, the message transmitted was “This is a test.” Assuming the HC-05 is working properly in the loopback test, the expected output would be a stream of “This is a test” messages displayed across the terminal until the program is terminated. After the first HC-05 Bluetooth module was tested for functionality the same steps were repeated on the remaining HC-05 module.

Additionally, the wireless communication module will be tested when it is integrated in the prototype. If the motorcycle successfully transmits the data it receives from the modules to the helmet system, then the wireless communication module and software will be considered successful. This will be verified through the LCD display. If the values are updating then the communication is successful.

10.2.1.2 Screen/GUI Testing

Each indicator on the screen will be tested individually to ensure that it is and is not displaying when appropriate. In addition, the size, shape, and location of the indicators will also be verified. When an object is detected by one or more of the sensors, the appropriate proximity warnings must be displayed on the screen. This includes all possible combinations of sensors detecting objects. The sensor indicator should only be displayed when an object is detected; therefore, when the object is no longer there, the indicator should no longer be displayed. It must also be tested that the time displayed on the screen is accurate. The time should match other clocks that are known to be correct. In addition, it should display in the twelve hour format with AM or PM depending on the time of day. In addition, the speed will be tested. When accelerating the speed on the indicator should also be increasing, and the opposite should occur when decelerating. If the user is at a standstill the display should indicate that the current speed is 0 miles per hour. At all times, the speed on the screen should match the speed that is on the rider's motorcycle. Finally, the cardinal direction will be tested. When the user makes a turn, the cardinal direction indicator should also change accordingly.

First the screen will be placed on a breadboard using break away headers. It will then be connected to the Arduino UNO using jumper wires. The other modules will be connected to the breadboard. A program will be created in the Arduino IDE to call certain draw functions for the screen in response to the inputs from the modules. All of the modules will be tested with the screen in order to verify that the indicators are being drawn correctly. These values will also be sent to the serial terminal so that it can be verified that the values displayed are correct. This screen testing should occur after the individual testing of the other modules is done in order to ensure that the modules themselves are working properly.

Finally, once the prototype is built the prototype will be tested in the various environments and scenarios described in the sections above. At all times during this test, the values on the screen should be updating. Additionally, the layout of the screen should remain consistent. If any of the indicators are displaying incorrectly or the values are not updating when they should be, the hardware and software components of the screen will be analyzed again in the lab setting in order to determine what the source of the error is. Ideally, the format of the screen should resemble the user interface layout depicted in Figure 54 earlier in the document. The prototype will be tested in various lightings. Currently, the only lighting source for the display is the backlight. This backlight is adjustable, so the ideal setting must be determined for the various times of day. The rider should be able to use the system whenever he or she wants, so the screen must be easily readable.

10.2.1.3 Solar Panel Testing

Although there are two forms for charging the battery on the helmet system, the main form of charging is via the solar panels adhered to the motorcycle helmet itself. Therefore it is important to make sure that the panels are providing the desired voltage and current output. Since the panels are to be molded to the shape of the helmet, some panels will be slightly angled and, therefore, not receive direct sunlight. Furthermore, the panels will be subject to varying levels of sunlight due to weather and the location of the rider.

Given the above mentioned challenges, the solar panels will have to be tested in order to account for both. To start off, the panels will be placed on the helmet in the configuration that best exposes all of them to direct sunlight. Using a multimeter the voltage and current will be measured from the combined panels. Then the panels will be connected to the BQ25570 and the lithium polymer battery in order to ensure that the battery can be properly charged in a timely manner using the solar panels.

In order to correctly gauge the output levels of the panels they have to be tested in different light levels. With the panels in the configuration that best allows them to be hit by direct sunlight, they will be placed in varying levels of light. This can be achieved by placing the helmet outside during sunrise, mid-morning, noon, midafternoon, and sunset. To further test how the system will perform in different light levels, the helmet will be placed in the shade during the aforementioned times of day and tested again. Finally, the same testing will be done on a cloudy day.

The testing outlined above should give a comprehensive analysis of how the panels will perform on the Head On system. However, it is important to note that the current configuration of the panels allows for room for an extra panel should the tests prove that the required output is not being reached.

10.2.1.4 Speed Testing

One of the features of Head On is to allow the user to digitally visualize their current travel speed in miles per hour, thus reducing the need to look down at the gauges. To allow this feature to be useful to the rider, it needed to be accurate to prevent the user from conflicting with speed limits. The testing for this component is not as simple as looking down at the gauges and comparing it to the digital output of Head On. The reasoning for this is that a stock motorcycle with no changes to gearing will have inaccuracies from the factory on the speedometer. This variation on speedometer is generally around 10% above the true speed for a majority of motorcycle manufacturers and is further affected by gearing changes and different or worn out tires. Therefore, an alternative method for testing must be created. To test this portion, a little creativity was used using a car, laptop, Android phone, Arduino Uno, and GPS module.

To begin, the Arduino Uno was wired to the GPS module per the data sheet. The Arduino Uno was then connected to a laptop through USB for both power supply and data transfer. Then a new sketch was uploaded to the Arduino Uno with the intent of displaying the GPS modules tracked speed. Specifically, a similar testing program was used as seen in the wireless communication testing with a while loop that displays a message every two seconds; however, instead of “This is a test”, the program would display the GPS speed as parsed by the Adafruit GPS library function speed(). The next part of the testing configuration involved the Android phone which will use its internal GPS feature to display speed. The Android phone in this case is the control as it can be trusted to display accurate location and speed. Furthermore, a speed tracking application needed to be installed on the phone as the preinstalled Google Maps application did not provide speed functionality. Thankfully the Google Play Store has an abundant number of free GPS based speedometer apps from which SpeedView was selected due to its highly rated reviews for accuracy and functionality. SpeedView provides a clean and simple user interface which displays the speed in digital and gauge formats as seen in Figure.

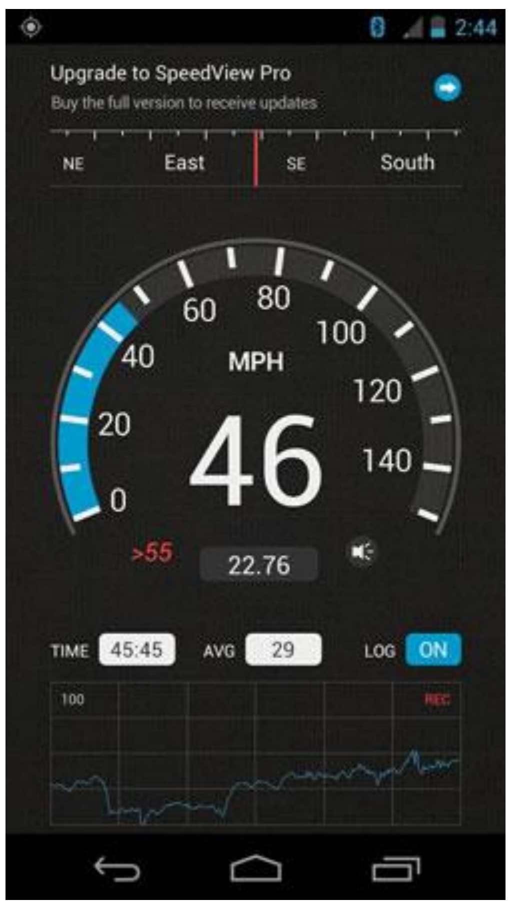


Figure 56: SpeedView Display

Finally, with all portions of the testing environment set, a moving vehicle was used with the laptop, Arduino Uno, and Android phone operating within it. On the PC side, a new instance of the terminal was then initialized and the Arduino sketch was run. On the Android side, SpeedView was initialized with high accuracy location mode enabled, which uses a mix of GPS, Wi-Fi, Bluetooth, and cellular networks for information. As the vehicle approached constant speeds, the values displayed on the terminal from the GPS module were validated by comparing them to the output of SpeedView.

10.2.1.5 Clock Testing

The clock testing for the system will be much less complicated than the other modules. The real time clock module will be connected placed on a breadboard and connected to the Arduino Uno using jumper wires. A new program will be created in which the minute, hour, and period are continuously retrieved from the module. Every time the value of the time changes, it will be displayed on the serial monitor. These values will be compared to an online clock to verify accuracy. In this case, time.gov will be used as it provides the time as provided by three time agencies of the United States: a Department of Commerce agency, the National Institute of Standards and Technology (NIST), and the U. S. Naval Observatory (USNO). This process will be done for several minutes in order to ensure that the clock is changing at the proper time. Additionally, testing will be done several times to confirm there are no significant deviations in timekeeping with continuous use of the real time clock module.

10.2.1.6 Sensor Testing

The sensors will first be tested in the lab setting. This testing method will serve two purposes. First, it will allow the maximum range of the sensors to be verified. This is imperative because the sensors must be able to sense far enough that it can determine if an object is present in one of the blind spots. If the sensors do not provide the necessary range needed for this project, another sensor will be chosen. The second purpose for this testing is to ensure that the sensors are cycling through correctly. When testing the sensors in this setting, the three sensors will be placed on a breadboard and connected to the Arduino Uno using jumper wires. The grounds of the three ultrasonic sensors will all be connected to the ground on the Arduino UNO. Similarly, the VCC of the sensors will all be connected to the +5V on the Arduino. Finally, the trigger and echo pins for each sensor must be connected to their own pins on the development board. The exact testing program created will vary depending on what pins the echo and trigger pins are connected to. A program will be created in the Arduino IDE in which, one by one, the sensors receive the appropriate trigger and echo signal to detect an object. This value will be converted to centimeters and store in a variable. Once this is completed for all three sensors, the values will be displayed on the serial monitor. Objects of varying shapes and sizes will be placed at varying distances in front of different sensors in order to verify the range of the sensors. Once a desirable range is found, actual software that will be used by the prototype will be tested. Slight modifications will have to be made in order to have the values display on the serial monitor, however, the majority of the program will remain the same. This test will verify that all three sensors are detecting objects at the proper time and confirm that the sensors are not interfering with each other.

Once the prototype is built, the sensors will be tested in the various real life scenarios. Various traffic patterns and speeds will be tested. Since the serial monitor cannot be used in this scenario, the display mounted to the front of the helmet must be used instead. Therefore, it is imperative that before testing the sensors on the prototype, the display must have successfully tested in the lab setting. The both the low and the high traffic patterns will help verify that the sensors are only updating when there is a change in their environment. Testing the sensors while driving at various speeds will help confirm that the sensors are quickly updating based on changes in the objects around the user. For example, if the rider is driving on the highway, and he or she passes a car, the sensors should be able to quickly sense that vehicle and change the indicator on the screen. Once that vehicle is no longer present, the sensor should then detect this and remove the indicator from the screen. In order to be effective the sensors and the display updates must be extremely quick.

10.2.1.7 Cardinal Direction Testing

In order to test the cardinal direction in the lab setting, the compass module will be attached to the breadboard using breakaway header pins. Once the module is secured, jumper wires must be used to connect these headers to the Arduino Uno. After completing the setup, a program will be created in the Arduino IDE which continuously retrieves the cardinal direction from the module and displays the value on the serial monitor. The module and breadboard will be randomly pointed in all four cardinal directions and all four intermediate directions in order to ensure that the compass module is correct.

Additionally, because the compass module will be oriented at a different angle when on the motorcycle helmet, once the prototype is built the compass module will be tested again. This will ensure that the program interacting with the compass module is interpreting the results correctly. Similar to the lab testing procedure, the rider will turn in different directions in order to verify all the cardinal and intermediate directions. The result will be displayed on the LCD screen on the front of the helmet.

11.0 Administrative Content

One of the keys to the success of this project was proper administration. This included creating milestones to keep the group on track, keeping track of budgets and financing, and distributing the work between all four team members based on his or her skills.

11.1 Milestone Discussion

In order to successfully complete Senior Design I and II, a series of milestones was created. This aided in keeping the project on track so that all deadlines could be met. Table 15 and Table 16 illustrate the milestones that were set for Senior Design I and II, as well as the deadline associated with each one. Not all of these deadlines are set in stone and they may be changed if necessary.

In Senior Design I, the primary focus is researching and designing the hardware and software components of the project. This includes researching similar products, determining possible ways to implement the Head On features, and investigating specific modules that could be used. Once sufficient research was performed, the prototype design process began. All of this research and design decisions were incorporated into the Senior Design I paper. At the end of the semester, this Senior Design I documentation is the sole method of evaluating the success of the project up to that point. A working prototype will not be necessary until the end of Senior Design II.

Table 15: Senior Design I Milestones

Milestone	Deadline
Research possible project ideas	August 28, 2015
Choose project topic	August 31, 2015
Perform market analysis of similar products	September 9, 2015
Develop list of possible features	September 9, 2015
Research ways to implement the agreed upon features	September 11, 2015
Submit project proposal document	September 15, 2015
Propose idea to local companies for funding	September 22, 2015
Research chips and compatible modules	September 30, 2015
Design PCB for motorcycle and helmet system	October 31, 2015
Design UML diagrams for software	October 31, 2015
Begin programming microcontroller to test sensors	November 15, 2015
Finish writing Senior Design I paper	November 30, 2015
Proofread Senior Design I paper	December 8, 2015
Submit Senior Design I paper	December 10, 2015

In Senior Design II, the primary focus is the building and testing of the prototype. This builds upon all of the research and design that was performed in Senior Design I. By the conclusion of Senior Design II, the Head On prototype must be fully functional in order to complete the class.

Table 16: Senior Design II Milestones

Milestone	Deadline
Acquire parts for prototype	January 30, 2016
Obtain PCB for motorcycle and helmet system	January 30, 2016
Perform unit testing for each module	February 10, 2016
Develop prototype with all modules transmitting and receiving data	February 15, 2016
Test prototype for accuracy and determine changes that need to be made	February 17, 2016
Finalize design of hardware and GUI	February 20, 2016
Build and program final product	February 29, 2016
Troubleshoot problems	March 15, 2016
Test for accuracy and make necessary modifications	March 31, 2016
Finish final documentation	April 15, 2016
Demonstration	TBD
Ensure all requirements set in senior design are met	April 29, 2016

Overall, there are three main phases to this project: research, design, and testing. Each of these phases contributed to the overall completion of the project. Because the agile method is being used for this project, it is impossible to give an exact date for when each phase will end. Depending on the information gathered at each stage, the project may move on to the next stage or may return to the previous stage in order to make modifications.

In the first phase, which extended through the majority of Senior Design 1, a project was chosen and the goals, requirements, and specifications were determined. After this, the research into relevant technologies and possible modules to use with this system was determined. After much research, all the modules were chosen. However, along the way goals and features of the system were modified, added, and removed as a more realistic idea of what would be feasible was developed. Thanks to the fluidity associated with the agile method, these changes could be easily made without having to restart the project from scratch.

At this point phase two, the design phase, began. Phase two started in Senior Design 1 and would extend into Senior Design 2. This phase mainly consists designing and creating the hardware and software of the system. This included designing the PCB, creating schematics, designing the user interface, and beginning to program the microcontrollers. Because the design phases for the software and hardware components are occurring simultaneously, all of the programming was done on a development board; this alleviated the need for the two computer engineering students to have to wait until the prototype was built to start testing. Once all of the parts were acquired, the process of creating prototypes began. This process allowed the design to be actualized.

Once a prototype is created, the testing phase will ensued. This will occur only in Senior Design 2. During the testing phase, each component of the system will be tested for accuracy. A variety of tests will be run in multiple environments in order to ensure that the goals, requirements, goals, and specifications are met. This includes testing the components in a lab environment using the development board as well as testing the actual prototype while on the road. If it is determined that one of the requirements set for the project has not been met, the project will return to the design phase in order to make necessary modification. This includes both software and hardware modifications. In some cases, if it is determined that the component itself is the problem and not the way it is integrated into the design, the project may return to the research phase in order to resolve the problem. Once again, this fluid movement between phases is made possible by the implementation of the agile method. This process of testing and making modifications based on results will be repeated until all tests in phase three result in success.

11.2 Sponsors

At the start of Senior Design I, the group applied for funding from Boeing and Leidos. A funding proposal was created detailing the project description, estimated budget, and plan for moving forward. After reviewing the proposal that was submitted, Boeing/Leidos generously agreed to donate $532.48 for this project.

11.3 Budget and Financing

Throughout the creation of this project a budget had to be maintained. This budget was calculated based off of estimated costs for parts, the funding received from sponsors, and the amount each group member was willing to contribute.

11.3.1 Estimated Budget

A critical part of the design process was determining a budget for the project. In order to determine the budget of the project, the parts that would have to be acquired for the prototype and final product had to be ascertained as well as what equipment and services would be needed to build them. In addition, an estimate of the amount of funding that would be received also had to be taken into consideration since the remainder of the funding would have to be provided by the team members. Table 17 displays the estimated cost of parts that will be used for creating the prototype and final product.

Table 17: Budget

Part	**Quantity**	**Unit Price**	**Total Price**
Motorcycle for testing	1	Acquired	$0.00
Motorcycle helmet	1	$50.00	$50.00
Printed circuit board	10	$5.00	$50.00
Microcontroller	2	$10.00	$20.00
GPS module	1	$50.00	$50.00
LCD display	1	$15.00	$15.00
Solar panel	8	$15.00	$120.00
Battery	1	$40.00	$40.00
Sensor	3	$10.00	$30.00
Wireless module	4	$9.00	$36.00
Power regulator	4	$15.00	$60.00
USB TTL board	1	$6.00	$6.00
Breadboard	1	$3.00	$3.00
Jumper cables	1	$3.00	$3.00
Development board	2	$25.00	$50.00
GPS module	1	$50.00	$50.00
Real time clock	1	$2.00	$2.00
Electronic compass	1	$14.00	$14.00
Energy harvesting system	1	$8.00	$8.00
Break away headers	1	$3.00	$3.00
Total:			**$610.00**

In an effort to further analyze the budget, the pie chart below breaks down the estimated costs spent on each item as a percentage of the total. The majority of the spending for this project will be on power related components, specifically the solar panels and power regulator. Though solar panels are by no means the most expensive component of this project, several of them are needed for the prototype as well as for testing purposes. Similarly, an individual power regulator is not expensive; however, several are needed for the system. The majority of the modules fall in the midrange of spending. Though some of these, such as the GPS, are in fact the most expensive per unit, only one will be needed for the system. Most of the testing equipment used for the development of the software is on the low end of the spending spectrum. Several of these components are so inexpensive, only a couple of dollars, that their cost is almost negligible when considering the entire spending budget.

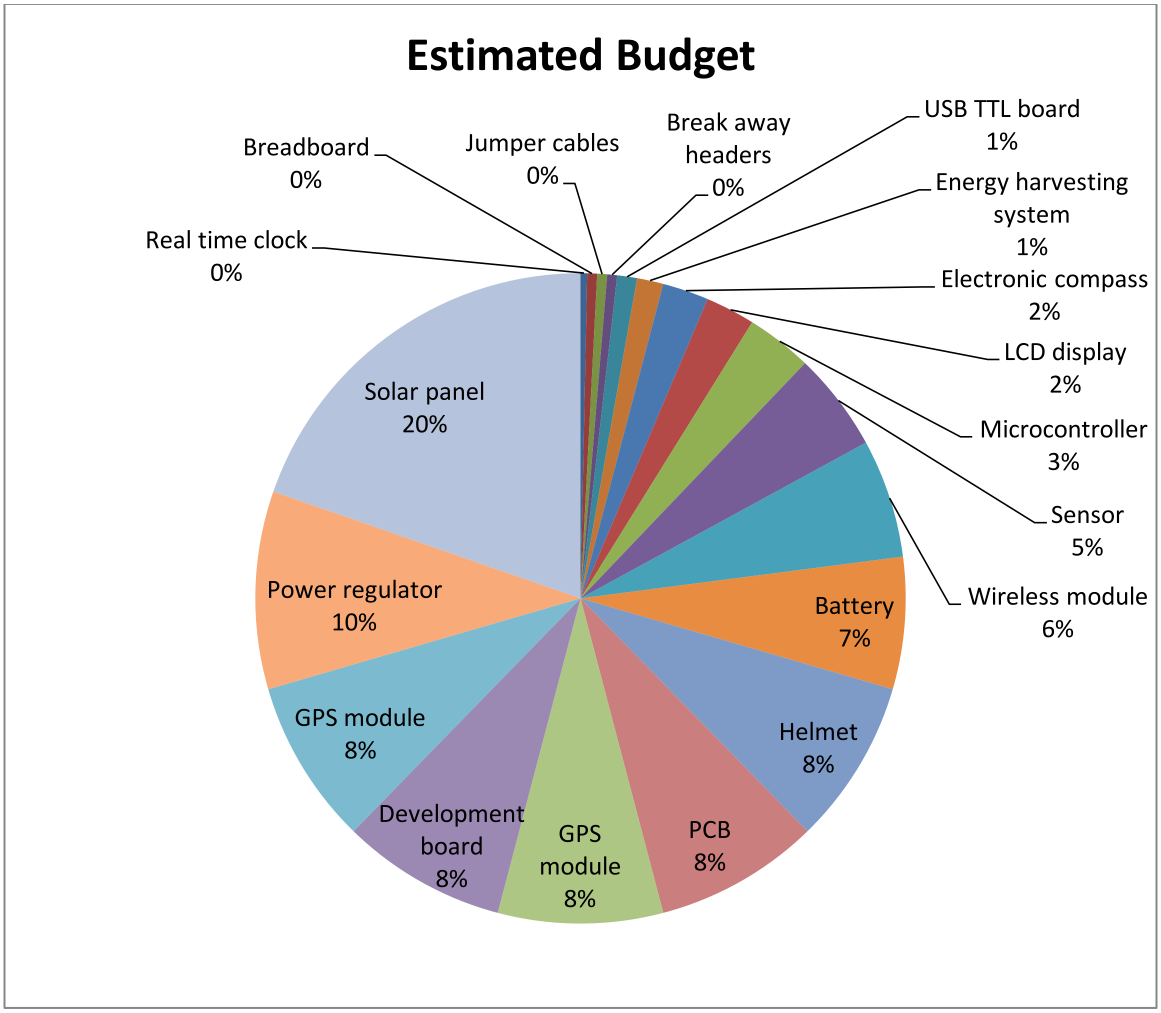


Figure 57: Budget Percentages

11.3.2 Financing

Once an estimated budget was determined, the group then began looking into possible sources of funding. A funding proposal was created outlining the project and how any funding received would be used. After submitting necessary documentation, Boeing/Leidos agreed to sponsor this group by generously donating $532.48. It was agreed by all member of the group that if this project exceeds this amount, then the difference will be split evenly between all four members. Fortunately, the cost of producing the prototype was lowered due to the resources available to UCF students. This included the use of tools in the Texas Instruments Innovation Lab as well as the Senior Design Lab. As a result, a larger portion of the funds could be dedicated to the purchasing of the components for the system. Furthermore, some of the group members already had equipment that was used for testing, such as development boards, breadboard, etc., as well as a motorcycle and motorcycle helmet. Finally, because similar products were available from different distributers, lower prices were able to be found.

11.3.3 Bill of Materials

The following table contains the complete list of parts purchased or already acquired that were used in the prototype and final product of Head On. At the moment, while all of the parts have been chosen, not all of the parts have been purchased. Those parts not purchased yet have been indicated by the TBD in the quantity and total price columns.

Table 18: Bill of Materials

Part	**Quantity**	**Unit Price**	**Total Price**
Motorcycle	1	Acquired	$0.00
Motorcycle Helmet	1	Acquired	$0.00
Arduino Uno	2	Acquired	$0.00
Bread Board	2	Acquired	$0.00
Jumper Wires (set of 20)	2	Acquired	$0.00
36-pin 0.1" Female header (pack of 5)	TBD	$2.95	TBD
HC-SR04 Ultrasonic Sensor (pack of 3)	TBD	$7.50	TBD
Adafruit 2.2" 18-bit color TFT LCD display	TBD	$24.95	TBD
ATmega328	TBD	$1.63	TBD
HC-05 Wireless Communication Module	TBD	$8.99	TBD
3.7V 2000 mAh lithium polymer battery	TBD	$12.95	TBD
bq25570 Nano Power Boost Charger and Buck Converter	TBD	$7.20	TBD
MPT4.8-75 Solar Panel	TBD	$10.21	TBD
DS3231 Real Time Clock	TBD	$1.99	TBD
LSM303D Electronic Compass Module	TBD	$14.99	TBD
PA6C LadyBird-3	TBD	$12.00	TBD
PCB board	TBD	TBD	TBD
Total:			**TBD**

11.3.4 Financing Issues

Originally, this group planned on entering the TI Innovation Challenge North America Design Contest. Groups who entered this contest received a $100 coupon to use on Texas Instruments products for their project. In order to be eligible for this contest, the group must use at least two TI analog integrated circuits and a TI processor in the project. However, after analyzing a variety of modules from other vendors, it was determined that the other products would be more suitable for this project. Unfortunately, this choice resulted in the loss of this possible source of funding.

11.4 Team Organization

The Head On project team consists of two electrical engineering and two computer engineering seniors at the University of Central Florida. Because this project consists of both a hardware and software component, the work was divided among the two subgroups accordingly. The two computer engineering students were tasked with all software aspects of the project. During the research phase of the project, the computer engineering students focused on finding sensors, microcontrollers, and screens since they would be the ones that would program them. Once the modules were chosen, these two students then began programming the microcontrollers in the helmet and motorcycle modules, as well as the user interface of the display. On the other hand, during the research phase, the two electrical engineers were responsible for all aspects related to power. This included but was not limited to, the solar power cells, the battery, and power regulators. Once the research phase was complete, these two students then focused on designing and creating the printed circuit boards for the helmet and software modules as well as determining how to best utilize and distribute adequate power to all components.

During the research phase of this project, weekly meetings were held in order to compare findings. The success of the project relied heavily on the communication between group members since the modules being chosen by each group must work together. Appropriate changes to the modules chosen were made when necessary. During these team meetings, each group member also discussed his or her progress, as well as what aspects of the project still needed to be completed. This allowed the group to gain a realistic idea of how much work remained. At the conclusion of each meeting, a goal was set for what each member should complete before the next meeting. In addition to these full group meetings, smaller meetings were also held in the computer and electrical subgroups. Each subgroup would meet to discuss their components of the project. Similarly, work would be divided within the subgroup members.

12.0 Conclusion

In conclusion, when approaching this project a great deal of planning and research had to take place. Goals, requirements and specifications had to be determined. Next research was performed on what modules to incorporate into the system. Once the final choices of modules were determined, the process of designing the hardware and software components began. PCB diagrams, flowcharts, UML diagrams, and schematics were created. During this process, many features were edited and others were removed completely. Once these diagrams and schematics were created, the prototype was built and tested. The design of the Head On system was then modified as necessary based off of the test results.

Head On does have possible features that could be added to it in the future in order to improve the user's experience even further. One feature includes a forward facing sensor. However, in order for this to be done, a sensor would have to be found that can detect objects much farther than the sensors used for the left, right, and rear. Two possible solutions to extend the detection range are RADAR and LIDAR systems which have ranges have ranges from 50 ft to over 300 ft. The rider will must have adequate time to react to the detection of the object. Improving the efficiency of the software would not sufficiently increase the amount of time the user has to react, so it is the sensor that must provide this needed time.

Another additional feature that could be added to the system is an indication of how close the hazards are to the rider using a colored indicator system. The color of the corresponding indicator on the screen could change color depending on the proximity of the object. For example, an orange indicator could indicate that the object is 15 feet away, while a red indicator could indicate that the object is 5 feet away. Ideally, this indicator could also take into account the rider's speed. For example, at higher speeds an object may be considered more highly hazardous at a further distance.

Furthermore, in the future this project could be improved by allowing more customization from the user. This could include allowing the user to choose a color scheme for the user interface or change the layout of the display by adding or removing features. Additionally, the indicators themselves could be changed. For example, the user could choose to display the time in the 24-hour format or the speed in kilometers per hour. However, in order to implement this, the user would have to have some sort of way to interact with the system such as buttons or touch screen capabilities.

Finally, one of the concepts that was originally considered but later removed in the design process was the emergency notification system. Similar systems have been integrated in cars and in some of the helmet systems discussed in the beginning of this report. While the main focus of this system is to help the rider stay safe, this added feature would be incredibly beneficial to the user if he or she should get into an accident.

Overall, it is the goal of the Head On system to assist the user in becoming a safer driver through the use of hazard detection and dashboard integration with the display. Because of time constraints of this project, many features had to be removed from the system. However, these features could be added to the system in the future in order to further improve Head On.

13.0 Appendices
13.1 Appendix A: Permissions to Use

CM Cristhian Marin
4:38 PM

PERMISSION TO USE
To: pr@microchip.com

I am a computer engineering student at the University of Central Florida. I am requesting permission to use an image of the RN42 Bluetooth module from the product brochure in my senior design paper as a reference to available wireless communication hardware available for use in our helmet design.

Thank you for your time.

Cristhian Marin
University of Central Florida '16
Computer Engineering
cristhian@knights.ucf.edu | 321.750.6069

MM Michael MARKOWITZ
12/7/2015 4:46 PM

RE: Permission to use
To: Cristhian Marin

Permission granted, Christian.

I hope you chose to use this MCU!

life.augmented

Michael Markowitz | Tel: +1 781 591 0354
Corporate External Communications | Director Technical Media Relations

STMicroelectronics

ST online: www.st.com | Follow us on twitter: @st_world or, for media, @st_news

This communication is confidential and intended solely for the addressee(s). If you are not the intended recipient, you should not review, retain, copy or distribute the e-mail itself or the information it contains. In such case, we kindly request you notify the sender by replying to this transmission and delete the message without disclosing it. Thank you!

From: Cristhian Marin [mailto:cristhian@knights.ucf.edu]
Sent: Monday, December 07, 2015 4:19 PM
To: Michael MARKOWITZ
Subject: Permission to use

Dear Mr. Markowitz,

I am a computer engineering student at the University of Central Florida. I am requesting permission to use the pinout diagram for the STM32F405VG in my senior design paper in reference to the available choices for microcontrollers to implement in my team's helmet design.

Thank you for your time.

Cristhian Marin
University of Central Florida '16
Computer Engineering

CP Chugh, Paramjyot
12/7/2015 4:59 PM

RE: PERMISSION TO USE IMAGE
To: Cristhian Marin

Christhian,
Thanks for checking with us on the use of Atmel materials. Kind of usage that you have described below would be considered fair use, so you are authorized to use this material provided you provide acknowledgment in your material about the source of these images.

Regards,
Paramjyot
(408) 487-2706

From: Cristhian Marin [mailto:cristhian@knights.ucf.edu]
Sent: Monday, December 07, 2015 12:46 PM
To: DL-Atmel-US-PR
Subject: PERMISSION TO USE IMAGE

I am a computer engineering student at the University of Central Florida. I am requesting permission to use the images for the ATmega 328 and ATmega 1280 pinout diagrams in my senior design paper in reference to choices for microcontrollers available to implement in our helmet design.

Thank you for your time.

Cristhian Marin
University of Central Florida '16
Computer Engineering
cristhian@knights.ucf.edu

SN Stephanie Newsted
5:25 PM

RE: LSR Marketing Department
To: Cristhian Marin

Hi Cristian,

Yes, you may go ahead and use our SaBLE-x image. Please let me know if you need anything additional.

Best,
STEPHANIE NEWSTED | MKTG + COMM ASSOCIATE
P: 262.228.6906 | snewsted@lsr.com

From: Cristhian Marin [mailto:cristhian@knights.ucf.edu]
Sent: Monday, December 7, 2015 4:19 PM
To: Stephanie Newsted
Subject: Re: LSR Marketing Department

Hi Ms. Newsted,

We will be using the image solely in internal documentation; however, I would like to note the paper will be made available in PDF format for our peers and future students to view.

Regards,
Cristhian Marin

On Dec 7, 2015 5:12 PM, Stephanie Newsted <snewsted@lsr.com> wrote:
Hi Cristhian,

Thank you for emailing us in regards to photo rights. Will you be posting this on a website at all or will this only be used in internal documentation?

STEPHANIE NEWSTED | MKTG + COMM ASSOCIATE

CM Cristhian Marin
12:25 PM

PERMISSION TO USE IMAGE
To: support@innogear.net

I am computer engineering student attending the University of Central Florida. I am requesting permission to use the image of the HC-05 Bluetooth module as a reference in my senior design paper to discuss on available wireless communication devices .

Thank you for your time.

Cristhian Marin
University of Central Florida '16
Computer Engineering

SJ Steve Johnson <sdj@smf.org>
To: Amber Farrell;
Reply all
Tue 12/8/2015 11:34 AM

Inbox

Dear Amber,

You are welcome to use Figure 1, ISO Head form for you senior design project.

Sincerely,

Stephen Johnson
Administrative Director
Laboratory Manager

Snell Memorial Foundation, Inc.
3628 Madison Ave, Suite 11
North Highlands, CA 95660
916-331-5073 tel. 916-331-0359 fax
888-SNELL99 toll free
sdj@smf.org
smf.org

AF Amber Farrell
To: info@smf.org;
Reply all
Mon 12/7/2015 6:26 PM

From: Amber Farrell
Sent: Mon 12/7/2015 6:26 PM
To: info@smf.org;

Permission

I am a computer engineering student and I am currently working on my senior design project at the University of Central Florida. I am requesting permission to include Figure 1, found in the link below, as part of our research paper.

http://www.smf.org/standards/k98/k98std

Thank you.

Amber Farrell
Computer Engineering
University of Central Florida
farrell.amber@knights.ucf.edu

Permission

AF Amber Farrell
To: info@iteadstudio.com;

Reply all

Fri 11/6/2015 4:58 PM

From: Amber Farrell
Sent: Fri 11/6/2015 4:58 PM
To: info@iteadstudio.com;

Permission

Hello my name is Amber Farrell and I am working on my senior design project at the University of Central Florida. As part of our project I would like to include the sequence chart for the HC-SR04 ultrasonic ranging module found in the link below:

http://www.electroschematics.com/wp-content/uploads/2013/07/HC-SR04-datasheet-version-2.pdf

Thank you.

Amber Farrell
Computer Engineering Student
University of Central Florida
farrell.amber@knights.ucf.edu

Texas Instruments Incorporated

Copyright http://www.ti.com/corp/docs/legal/copyright.shtml

TI further grants permission to non-profit, educational institutions (specifically K-12, universities and community colleges) to download, reproduce, display and distribute the information on these pages solely for use in the classroom. This permission is conditioned on not modifying the information, retaining all copyright notices and including on all reproduced information the following credit line: "Courtesy of Texas Instruments".

Terms of Use http://www.ti.com/corp/docs/legal/termsofuse.shtml

TI further grants to K-12 educational institutions, universities, and community colleges permission to download, reproduce, display, and distribute TI Services solely for use in the classroom, provided that such institutions identify TI as the source of TI Services and include the following credit line: “Courtesy of Texas Instruments.”

Mon 12/7/2015 10:22 PM

Larry Herman

Permission to use images

To 'mpja@mpja.com'

To whom it may concern,

I am a student at the University of Central Florida's College of Engineering and Computer Science. I am writing on behalf of my senior design project team. We are required to produce a design document as part of our final design project. The document is for educational purposes only and we would like to include the following images from your *HC-SR04 User Guide* with your permission.

Section 3 - Product Views: Front and Back
Section 7 – Module Timing: HC-SR04 Ultrasonic Module

Please let us know if we can use some or all of this images. Thank you in advance for your support of our project.

Sincerely,
Larry Herman

Mon 12/7/2015 10:45 PM

Larry Herman

Permission to use images

To 'info@maxbotix.com'

To whom it may concern,

I am a student at the University of Central Florida's College of Engineering and Computer Science. I am writing on behalf of my senior design project team. We are required to produce a design document as part of our final design project. The document is for educational purposes only and we would like to include the following images from your *LV-MaxSonar-EZ* datasheet with your permission.

Product Image
Timing Diagram
MB1000 Beam Pattern

Please let us know if we can use some or all of this images. Thank you in advance for your support of our project.

Sincerely,
Larry Herman

Mon 12/7/2015 11:10 PM

Larry Herman

Permission to use images

To 'gsisco@linear.com'

To whom it may concern,

I am a student at the University of Central Florida's College of Engineering and Computer Science. I am writing on behalf of my senior design project team. We are required to produce a design document as part of our final design project. The document is for educational purposes only and we would like to include the following images from your *LT3652* datasheet with your permission.

Typical Application - 2A Solar Pane With 7.2V $LiFePO_4$ Battery and 17V Peak Power Tracking
Pin Configuration – DD Package

Please let us know if we can use some or all of this images. Thank you in advance for your support of our project.

Sincerely,
Larry Herman

Mon 12/7/2015 11:27 PM

Larry Herman

Permission to use image

To 'Ferda.Millan@maximintegrated.com'

Dear Ferda Millan,

I am a student at the University of Central Florida's College of Engineering and Computer Science. I am writing on behalf of my senior design project team. We are required to produce a design document as part of our final design project. The document is for educational purposes only and we would like to include the Block Diagram image from your *DS3231* datasheet with your permission.

Please let us know if we may use this image. Thank you in advance for your support of our project.

Sincerely,
Larry Herman

Mon 12/7/2015 11:36 PM

Larry Herman

Permission to use image

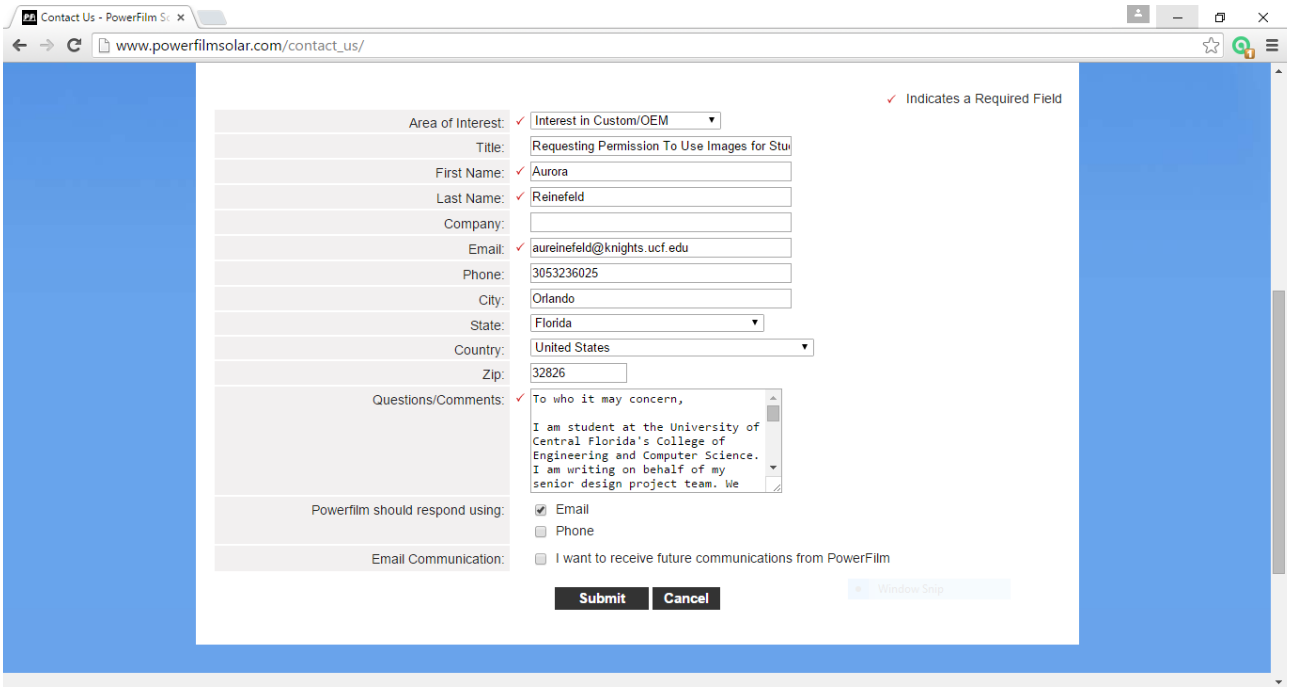
To 'AME_BO_West_Inquiries@list.st.com'

To whom it may concern,

I am a student at the University of Central Florida's College of Engineering and Computer Science. I am writing on behalf of my senior design project team. We are required to produce a design document as part of our final design project. The document is for educational purposes only and we would like to include Figure 2. Pin connections from your *LSM303D* datasheet with your permission.

Please let us know if we may use this image. Thank you in advance for your support of our project.

Sincerely,
Larry Herman


Contact Us - PowerFilm S
www.powerfilmsolar.com/contact_us/
✓ Indicates a Required Field
Area of Interest: ✓
Interest in Custom/OEM
Title:
Requesting Permission To Use Images for Stu
First Name: ✓
Aurora
Last Name: ✓
Reinefeld
Company:
Email: ✓
aureinefeld@knights.ucf.edu
Phone:
3053236025
City:
Orlando
State:
Florida
Country:
United States
Zip:
32826
Questions/Comments: ✓
To who it may concern,
I am student at the University of Central Florida's College of Engineering and Computer Science. I am writing on behalf of my senior design project team. We
Powerfilm should respond using:
Email
Phone
Email Communication:
I want to receive future communications from PowerFilm
Submit
Cancel

13.2 Appendix B: Bibliography

Shneiderman, B. & Plaisant, C. (2010). Designing the User Interface. Fifth edition. Boston, MA: Pearson Higher Education.

Snell Memorial Foundation. "Snell Foundation - M2015 Helmet Standard." *M2015 Helmet Standard*. Snell Memorial Foundation, 2015. Web. 07 Dec. 2015.

United States. U.S. Department of Transportation. Federal Highway Administration.*Mitigation Strategies For Design Exceptions*. By Federal Highway Administration. Federal Highway Administration, 15 Oct. 2014. Web. 7 Dec. 2015.

United States. National Highway Traffic Safety Administration. Department of Transportation. *49 CFR 571.218 - Standard No. 218; Motorcycle Helmets.*By Department of Transportation. Department of Transportation, 1 Oct. 2011. Web. 7 Dec. 2015.

13.3 Appendix C: Figure References

[1] http://www.mpja.com/download/hc-sr04_ultrasonic_module_user_guidejohn.pdf
[2] http://www.mpja.com/download/hc-sr04_ultrasonic_module_user_guidejohn.pdf
[3] http://www.maxbotix.com/documents/LV-MaxSonar-EZ_Datasheet.pdf
[4] http://www.maxbotix.com/documents/LV-MaxSonar-EZ_Datasheet.pdf
[5] http://www.maxbotix.com/documents/LV-MaxSonar-EZ_Datasheet.pdf
[6] Original work
[7] Original work
[8] Original work
[9] Original work
[10] http://www.ti.com/lit/ug/slau318f/slau318f.pdf
[11] http://www.ti.com/lit/ds/symlink/msp430fg4618.pdf
[12] http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf
[13] http://www.atmel.com/images/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf
[14] http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/DM00037051.pdf
[15] Original work
[16] http://www.microchip.com/wwwproducts/Devices.aspx?product=RN42
[17] http://penguink.com/index.php?route=product/product&product_id=26631
[18]https://www.lsr.com/embedded-wireless-modules/bluetooth-module/sable-x-ble-module
[19] http://www.ti.com/lsds/ti/power-management/energy-harvesting-and-solar-charging-overview.page
[20] http://cds.linear.com/docs/en/datasheet/3652fd.pdf
[21] http://cds.linear.com/docs/en/datasheet/3652fd.pdf
[22] http://www.ti.com/lit/ds/symlink/bq25570.pdf
[23] http://www.ti.com/lit/ds/symlink/bq25570.pdf
[24] http://www.powerfilmsolar.com/products/?mpt675&show=product&productID=271541
[25] http://www.powerfilmsolar.com/products/?mpt48150&show=product&productID=271540
[26] http://www.powerfilmsolar.com/products/?mpt4875&show=product&productID=271539
[27] https://www.maximintegrated.com/en/products/digital/real-time-clocks/DS3231.html#popuppdf
[28] Original Work

[29] Original Work
[30] Original Work
[31] Original Work
[32] http://www.smf.org/standards/k98/k98std
[33] Original Work
[34] Original Work
[35] Original Work
[36] Original Work
[37]: http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/DM00057547.pdf
[38] Original Work
[39] Original Work
[40] Original Work
[41] Original Work
[42] Original Work
[43] Original Work
[44] Original Work
[45] Original Work
[46] Original Work
[47] Original Work
[48] Original Work
[49] Original Work
[50] Original Work
[51] Original Work
[52] Original Work
[53] http://www.electroschematics.com/wp-content/uploads/2013/07/HC-SR04-datasheet-version-2.pdf
[54] Original Work
[55] Original Work
Figure 55: Wireless Communication Testing Layout
[57] https://play.google.com/store/apps/details?id=com.codesector.speedview.free&hl=en
Figure 57: Original Work