Bluetooth Recreational and Integrated Navigation (BRAIN) Helmet

Jordan Yamson, Ryan Mortera, Nada Algharabawi, and Stephan Morales

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

***Abstract* — The motivation for this project stems from trying to answer the dangerous distractions that navigational technology presents on the road for a motorcyclist. Rather than neglect the improving capability of a mobile phone's navigation, the solution explored involves a handsfree navigational helmet called the BRAIN Helmet. The BRAIN Helmet is the result of utilizing inexpensive and maintainable technology in a Bluetooth system to improve a motorcyclist's safety and comfort on the road.**

***Index Terms* — Audio system, Bluetooth, lithium batteries, microcontroller, mobile application, satellite navigation systems, switching circuits.**

I. Introduction

Before we hit the gas on our Senior Design Project, our team first desired to embrace Bluetooth technology and implement it into an important personal device that is used on a daily basis. Eventually we focused our attention to improving the daily ride for a motorcycle enthusiast and decided to create a Bluetooth Recreational And Integrated Navigation helmet; the BRAIN Helmet. Motorcycle helmets are typically designed without any electronic aspect but, with improving portable and rechargeable battery technologies, smart helmets are slowly on the motorcycle market's horizon. Our team's main goal for the BRAIN Helmet is to allow motorcyclists to appropriately utilize their cell phones without distracting or lowering the situational awareness of the rider. The helmet's electronic group will be mounted on the back of the helmet and attached to the helmet's integrated speakers, microphone, and heads-up display. The BRAIN Helmet will allow users to visually see navigation notifications on the minimal display while additionally being directed by the navigational audio being transmitted from their android cellular device. Additionally, users will be able to take hand free phone calls or listen to music. Overall, BRAIN Helmet users will be able to get to their destinations safely without having to pull out their phones and address incoming phone calls that may change their routes.

II. System Components summary

The BRAIN helmet system is composed of multiple components that are best summarized before going into further depth.

A. *Bluetooth Modules*

Several Bluetooth modules were considered for the BRAIN helmet. In order to properly stream audio for recreational and navigation, Bluetooth classic was necessary. However, Bluetooth Low Energy (BLE) was the primary choice for communication with a mobile Android application. The Blue Creation's BC127 had a dual mode BLE and Bluetooth Classic, but was unable to communicate enough characters in order to work for the BRAIN helmet's navigational Heads-Up Display (HUD). Due to this the BC127 was restricted to only communicating audio input and output while a BLE module was introduced strictly to communicate with the ATmega 2560 microcontroller for the navigational app.

B. *Speakers & Microphone*

The choice for the speakers and microphone didn't impact the design as much as the other parts due to just having to be able to fit within a motorcycle helmet. The thin speakers with a impedance of 8 ohms, and the electret microphone that needs the same voltage as the other parts to operate. The microphone and the speakers will be integrated directly into the helmet, one speaker on each side and the microphone

C. *Lithium Ion Polymer Battery*

The 3.7V Adafruit 328 Lithium Polymer battery is a decent sized battery with the battery capacity of 2500 mAh to keep the BRAIN Helmet operating for an entire day. It will easily fit into the electronic group housing along with the PCB and has the common JST connector allowing easy connection and replacement to a Li-Po charger.

D. *ATmega 2560*

The Atmega2560 microcontroller's main purpose is to process the data being sent from the BRAIN Helmet's Android application via the BLE Module to be discussed in a further section. It is additionally programmed to allow the rider to control the media and HUD with buttons on the BRAIN Helmet.

E. *OLED Heads-Up Display*

The HTDS-WS96 OLED display will allow us to minimize power consumption by operating at a 2.7V~6V compared to a standard LED display. This OLED display will be implemented in the system as the HUD and is small enough to attach to the helmet while not minimizing the field of view of the rider. This HUD will communicate with the ATmega2560 microcontroller to display navigational updates to the rider.

III. Details of Hardware

The hardware components summarized in section II, System Components will now be further explained with more in-depth details. A complete analysis of the hardware involved and contemplated can also be viewed on the full Senior Design report.

A. *Bluetooth wireless Technology*

The BRAIN helmet is made to be a wireless system to the user's smart phone, and the implementation that is used is Bluetooth technology. The BRAIN helmet system has both a Bluetooth Low Energy device as well as a Bluetooth 4.0 for two types of connections. The Bluetooth Low Energy technology is appropriate for this system for its name, low power consumption for the system. In this system, it is the backbone to the connection to the custom application that was developed. The Bluetooth 4.0 is used for high quality audio between the user's phone and our system. The specific components are as follows; Adafruit Bluefruit LE UART Friend for Bluetooth Low Energy, and BC127 for Bluetooth 4.0.

B. *Bluetooth Low Energy*

When describing these two parts of the system they both are Bluetooth technologies, but have two different purposes all together. The Bluetooth Low Energy Module, is used for low energy systems that don't need much power to turn on, in our case it is just a communication tool to send data between the android application and the BRAIN helmet. This type of Bluetooth technology is used mainly for device communications, and is optimized for data transfers. The Adafruit Bluefruit LE UART Friend Bluetooth Low Energy Module is perfect size and specifications for our system. Instead of installing this on the PCB with the BC127, it will be installed in the housing. This module will enable the transfer of data wirelessly between the application on the user's phone and our BRAIN helmet system. The strings that will be sent over the Bluetooth device are updates for the system to change the different signs on the heads-up display implemented into our system. The Bluetooth Low Energy of the system is also important when it comes to how much voltage this part of the system uses. Everything within our system uses an input voltage of 3.3 volts given from a Lithium Polymer battery.

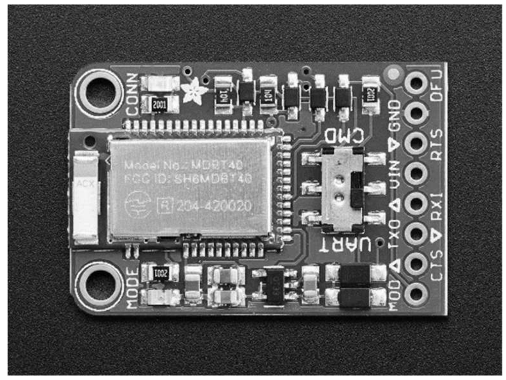

Fig. 1. Breakout Board of Adafruit Bluefruit LE. This BLE module allows easy communication between the Mobile Android Application and the ATmega2560.

C. *Bluetooth 4.0 Audio Module*

The audio module for this system is the Blue Creation 127. It is a Bluetooth dual mode, with a built in Bluetooth 4.0 and Bluetooth Low Energy. The reason why we used this Bluetooth device as an audio device is because at first, we thought we could use its BLE for the application of the system, but come to our attention that the BLE on this chip didn't have enough throughput to send all the data that needed to be sent from the app to our system.

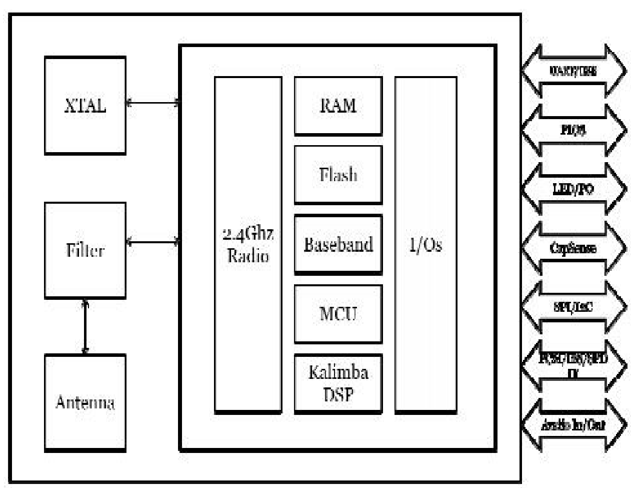

Fig. 2. Breakdown diagram of the Blue Creation 127 Bluetooth Module.

The BC127 comes pre-installed with different profiles for communication with all phones from androids to iOS devices. Its size is 11.8mm x 18mm x 3.2mm. This module has a software that can be used that is developed by Blue Creation that can be ideal for changing the audio settings on the Bluetooth device, such as increasing the audio gain for the output. The maximum data rate of this module is 3Mbps. It can work from 10 to 30 meters. This module has a built in Digital to Analog converter, with 16 bits and a DAC output sample rate of 8Khz to 90Khz. This Bluetooth Module is the main module for the audio subsystem. Wiring two 8 ohm thin speakers for the output and a microphone for the input of the audio system completed this subsystem. When using the BC127 it is just like the other parts in the system, it uses an input voltage of 3.3 volts. This audio device was used due to its ease of use and up to date software that keeps it updated within our system.

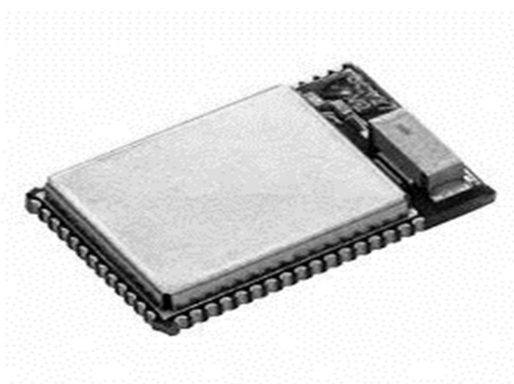

Fig. 3. Blue Creation Bluetooth 127 Module to be placed on the printed Circuit Board.

D. *Audio Sub-System*

The audio system is just a section of the project where two 8 ohm speakers are wired for outputs on the BC127 and an electret microphone is wired as input to the device. This system is for audio communication between the BRAIN helmet and the User's phone. This allows a user to take advantage of wireless cellular phone calls as well as wireless audio. The system has a decent sound clarity and loudness. When first testing, an amplifier was going to be put into the main circuit design. This amplifier was the Texas Instruments LM4902 audio amplifier. This would allow for the speakers to push a lot more volume, but this idea was ended up not needed due to the Melody software that is on the BC127. This software allowed us to boost the gain on the signals passing through the BC127 and allowed the music to be louder.

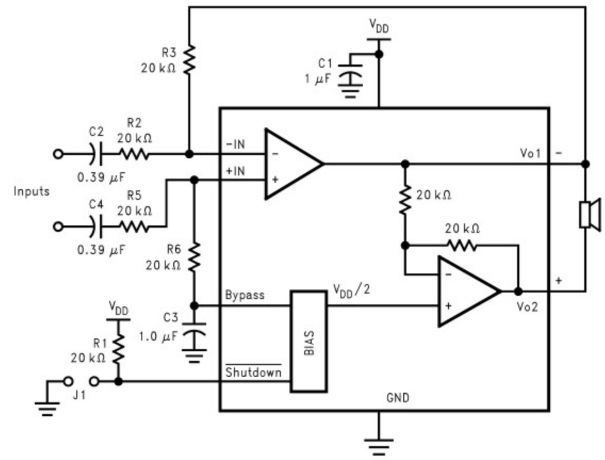

Fig. 4. LM4902 circuit for differential input audio to be amplified. This was schematic was used in our testing printed circuit Board before deemed unnecessary.

During prototyping the idea was to produce loud clear audio, however as the project drew closer to the deadline it was realized that the adjustable volume was echoed inside the helmet and if was any louder could serve as not only a distraction to a rider but as well as completely disrupt their hearing ability. As a result, the LM4902 was removed from the testing printed circuit board and the final board was allowed to be reduced in size. Another benefit of leaving the LM4902 was the additional power reserved in the system; extending the BRAIN Helmet life time between charges.

Using an electret microphone proved to be a problem when first testing. The microphone had a lot of static and noise when trying to communicate with the device. It was not clear when speaking through the microphone, so a high pass filter was created in order to clear up some of the noise that we were getting at the lower frequencies. Using 47 nF capacitors and 1 uF capacitor, the system did not have as much noise as first introduced. The microphone was a clear element needed in this section of the system in order for the user to clearly have conversations in calls or other services.

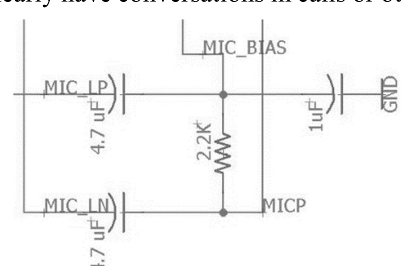

Fig. 5. High-Pass filter for microphone input to BC127.

E. *Power System*

Several aspects were considered when deciding on the power source of the BRAIN Helmet; Run time between charges, size, and output voltage were the largest factors in designing the power system of this system. After siding with the improving performance and ergonomically priced Lithium-Ion Polymer batteries over other power sources, the Adafruit328 3.7 Volts 2500 mAh battery was chosen. This allowed an ample amount of time to ride between charges; battery capacity over current drawn calculating to be over 20 hours as detailed in the full Senior Design report. As previously mentioned in regard to the LM4902, removing the two audio amplifiers for each speaker proved to improve the battery life tremendously as well. Charging the Adafruit328 was approached by looking for an effective and common method the BRAIN Helmet user would have no issue with. The answer to this was a simple Sparkfun Lithium-Ion battery charger. This charger maintains a safe charge current of 500 mA by a micro-usb connector. This charge current is limited by a resistance value on the charger circuit to ensure the battery cell does not take damage; shortening the battery lifespan or causing damage to the rider.

While the Adafruit328 regularly outputs 3.7V until drained, at max charge it holds ~4.2V. In order to regulate this voltage down to 3.3V; two options were considered. During the breadboard phase, a linear regulator was used. However, upon further investigation, it was evident that a linear regulator, though only a small drop in voltage, is still less efficient compared to a switching regulator, or buck converter. Using WEBENCH designer from Texas Instruments, it was confirmed that a TLV62565 Buck Converter is capable of a ~93% efficiency when our system is running with full streaming media.

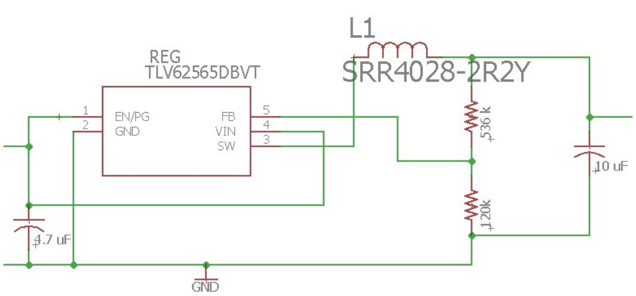

Fig. 6. TLV62565 Buck Converter Circuit used to regulate a ~3.7V Lithium Ion Battery to 3.3V outputting 500mA.

There were concerns when first introducing the switching regulator, or buck converter to the system. Research had shown that it could affect the Bluetooth modules signals broadcasting and interrupt services. However, throughout testing on breadboards after creating a mock up switching regulator power system with the Adafruit328, there was no seen interference with the Bluetooth modules communicating with the BRAIN Helmet mobile application. If a problem were to occur down the road with the BRAIN Helmet, the system will explore further options as the linear regulator was not a devastatingly low efficiency. Another advantage to not using a linear regulator though would be the excess heat that is lost in the linear regulator as the voltage drops in regulation.

F. *Microprocessor*

The microprocessor we chose is the ATmega 2560. It is available on the Arduino Mega 2560 development board provided by Arduino. Using the dev board for most of system implementation, it proved to be a great choice for our needs. It features 86 GPIO pins, more than enough for our system. If this system were to be commercialized, the extra pins would offer the product to have more features like a camera and storage for video recording or storing media. The processor contains four UART channels that will be utilized in our system. With a 16Mhz clock, 256KB flash there is enough speed and memory needed to run the system optimally. Atmel has great documentation for the processor and Arduino helped us by providing their schematic files to get a better understanding of how the development board operates with respect to hardware.

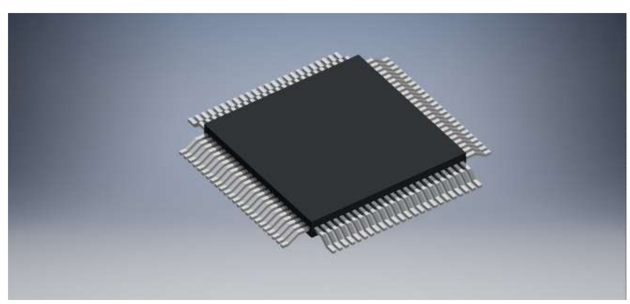

Fig. 7. ATmega2560 Processor; with the abundant amount of extra GPIO pins allows the addition for multiple features during the BRAIN Helmet's development

We chose this processor because of the high GPIO pins based on the MSP430 and MSP432 from Texas Instruments. This architecture has great documentation on Atmel's website and already have experience using their respective IDEs. This chip has the highest amount of FLASH available, which is more of a priority than RAM.

The components attached to the processor is a Bluetooth module which will oversee multimedia controls. This will help the user connect their device to the system like connecting any other peripheral. The buttons will serve as the main input for the multimedia controls. When the user presses the buttons, it will send data via UART to the module. In another UART channel is the Bluetooth Mow-Energy module which will be used exclusively for our Android app. The module has custom characteristics which will be saving our variables. It will act as a peripheral, so it won't save anything to the application. The display is connected to the system with I2C. The display will output information for the user to see while it's in operation. It will display the distance until the next turn, the direction of the next turn, the arrival time in minutes, and the speed of the vehicle.

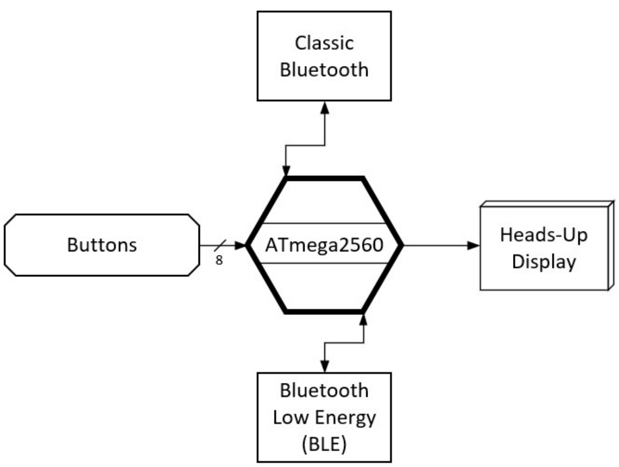

Fig. 8. This is a hardware block diagram depicting the communications between the Microprocessor and the corresponding components of the BRAIN Helmet electronics

There were a wide variety of issues and revisions before the final design. First the Bluetooth module was not low-power, so we couldn't use our custom services to store data. On the next Bluetooth module, which is the current Bluetooth media module, does have BLE built in. With this device the module does not save into the FLASH is our UUIDs, so we had to upload the BLE's database every time the system powers on. We realized that the method in charge of uploading the database is overflowing the read buffer of the device. We tried to dynamically allocate the buffer and clearing the buffer, but the device won't comply. We did use the libraries that they provided to us and override the methods, but it proved no improvements. The third Bluetooth module was the Intel Curie chip. Stephan had the Arduino 101 before the project, so we tried to implement it with two Bluetooth modules. It worked flawlessly with the libraries that Arduino provided in the IDE, but there were two major concerns, the chip is discontinued and the processor was too expensive to acquire. We used this as an alternative just in case the new Bluetooth module that we were getting does not work within our requirements. When we received the BLE module from Adafruit, that also worked flawlessly. It requires minimal time to configure the module to our standards and with the libraries that Adafruit provided. Figure X: Software flow diagram

G. *Electronic Housing and Operation*

While this project in no way requires the mechanical engineering design of a fancy casing for the BRAIN Helmet electronic group; it is necessary to have some sort of housing to protect the components. For this design specifically, resistance and shock absorption are not taken into account; as the design and functionality is the crucial factor of this Senior Design project. The housing was created using Autodesk Inventor. It was built to house our PCB on the back of the helmet. The back of the helmet is the best place for our system since it does not interfere with the safety or aerodynamic features of the helmet. The size will fit to the curvature of the helmet to not make the presentation of the design too obtrusive. Since the system has a battery, we must protect the housing from exposure to the humidity of the air and weather. Our system has three LEDs which will inform the user that the system is powered on and indicate if there is a connection between the system and the device. Also, the user will be able to charge the battery that is inside the housing using a micro-USB cable and a switch to turn on and off the system. The buttons are separated from the main PCB, it is connected to it by wires that will be hidden within the helmet. Since the helmet has removable padding and we cannot drill any hole's due to safety regulations, we wanted to maintain the integrity of the system.

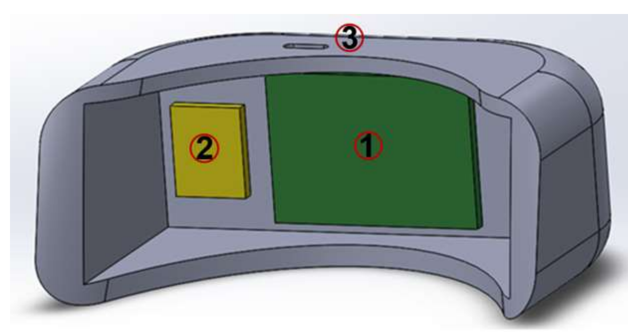

Fig. 9. This is a screenshot of an Autodesk Inventor rendering of the BRAIN Helmet electronic group housing.

In reference to Figure 9, there are multiple aspects to consider for the housing of the BRAIN Helmet system. The Battery, noted by the number 2 on Figure 9, will sit mounted inside the housing, next to the Printed Circuit Board, noted 1. The micro-usb charger will be accessible from the outside of the housing to allow the user easy access recharging after a ride. The Printed circuit board (PCB) will be installed onto the housing via three M3 screws and corresponding stand off screws attached to the housing. Due to development stages, the housing will initially be attached to the helmet via a removable source such as heavy-duty Velcro or double-sided tape.

IV. Details of Software

This section will describe in further detail the process of programming and application development in order to get the BRAIN Helmet to operate. When the system is powered on it will start initializing the Bluetooth modules and the display. In the loop of the program it will check for an input from the buttons and it will send the response to the module. After checking for inputs, it will poll the data stored on the UUIDs and will obtain the data and store it locally.

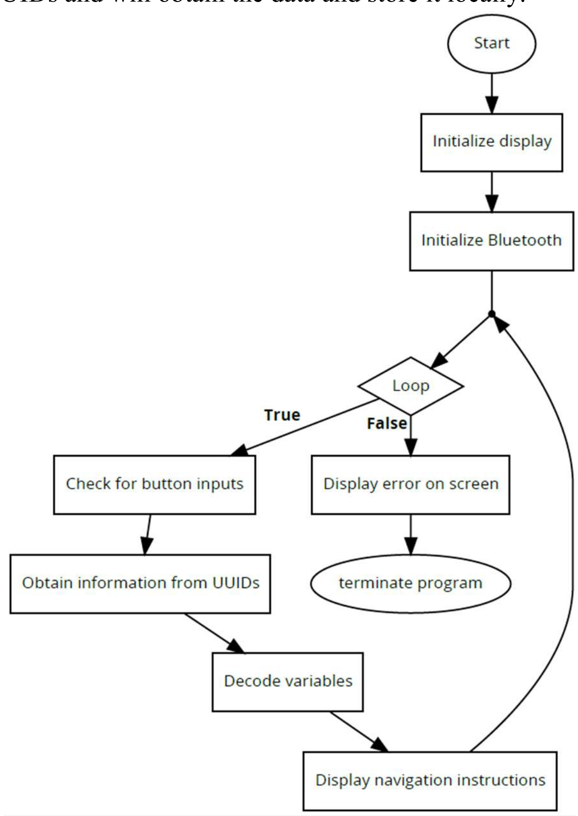

Fig. 10. A logic flow diagram of the BRAIN Helmet software in terms of communicating between navigation and the Heads-Up Display

With the local data accessible, it will convert the byte array into integer numbers and store them into their respective variables, so the display will update the screen with the most updated information. If the system did not get any navigation instructions it will display its default values.

A. *Connection between Microcontroller and the Mobile Application*

The Microcontroller is a small device that has multiple functions which controls the devices that are connected to it. Our project microcontroller is connected to multiple devices which are speakers, microphone, power supply, PCB and Mobile Application. Microcontroller and Mobile application have wireless communication via Bluetooth. This connection is mainly established to transfer back and forth streams of audio and string data. Thus, we have decided to develop a user interface to make it easier for users to communicate with the main module. After doing an extensive research and market analysis, we have decided to develop an application for Android devices since of the many excellent facilities it provides in addition to the large market share it has.

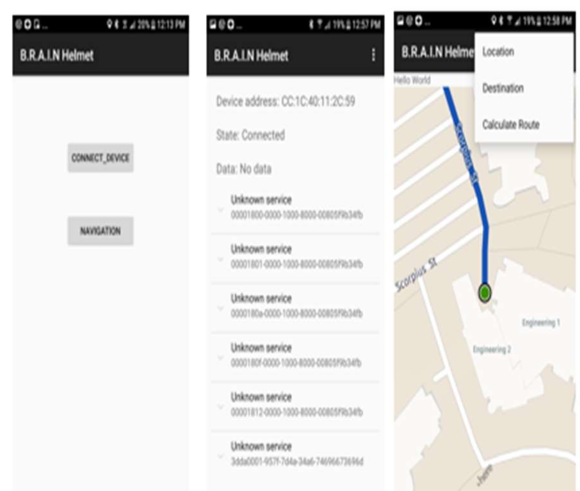

Fig. 11. This figure depicts three screens seen on the developed BRAIN Helmet Mobile application; a start up screen, a device information screen, and navigation.

B. *Android Studio*

We have chosen Android which is one of the most popular mobile operating systems designed by Google to build our Mobile Application. This mobile operating system is designed particularly for touchscreen devices such as smartphones and tablets. User Interfaces based on

Android require direct interaction which involves tapping and swiping. For Text inputs, a virtual keyboard will be provided. The response will be immediate to user actions. The home screen can be made of multiple pages that user can go back and forth. The default Android's user interface has a status bar on the top of the screen which includes information about the device and its connectivity. Also, a Notification screen is included to show important updates and information. Most Applications are written using Java programming Languages and Android Software Development Kit (SDK). This kit involves Software Libraries, Debugger, and an Emulator based on QEMU. Android Development Tools (ADT) was used by Eclipse to support Google's Integrated Development Environment (IDE). Android Studio is an Integrated Development Environment which was developed by Google. This Integrated Development Environment is based on IntelliJ IDEA which is a Java Integrated Development Environment for developing computer Software designed by JetBrains which is a software development company whose tools are targeted towards software developers and project managers. Android is supported by different kinds of Hardware such as x86, x86 - 64 MIPS64, and MIPS. ARM is the most popular hardware that supported Android's user interfaces. Non-compulsory hardware components may be combined with Android devices such as thermometers, GPS, accelerometers, pressure sensors, and gaming controls.

The IDE that Android Studio provides is a user-friendly interface that can be used by any good programmer. It is simple enough that any experienced programmer can learn to use it in a short period. There are many useful tools that Android Studio provides for contributors to take advantage of in order to develop an efficient application with the minimum amount of code, time, and effort in addition to the best experience for users. This operating system has an official integrated development environment called Android Studio. Android Studio programming depends on Java language and the function and packages available by Java. Fortunately, our software team has a good experience programming with Java, so Android Studio is an excellent choice to work with.

C. *The Main Functionality of the BRAIN Helmet Mobile Application*

The main purpose of Mobile Application is to send and receive data. The most important function that needs to be implemented is the Bluetooth connection. The user should be able to click the device connection button on the main screen and browse the available device which should include the Brain Helmet's device. After establishing a successful connection, the BRAIN Helmet user can use this system, as the name suggests, for recreation and not just for navigation. The successful Bluetooth connection allows the user to make phone calls with the helmet on, listen to their streaming music, podcasts, or even navigate their phone with their voice if applicable. The entire premise of the BRAIN Helmet is to allow the user to embrace technology and his mobile device's abilities without pulling the phone out on the road or at a stop light.

D. *Mobile Application Navigation*

The other important function that should be implemented in our application is the navigation function. This function is based on calculating the route to the desired destination and sending multiple important information to the motorcycle rider such as expected arrival time, velocity, distance, and directions. This crucial information will be constantly processed and delivered to the BRAIN Helmet user in two ways: they will hear the audio as one would in a car GPS and they will see it on the minimal, but effective, heads-up display which will be attached to the front of the helmet and constantly receiving and transforming data from the phone to the microcontroller and vice versa.

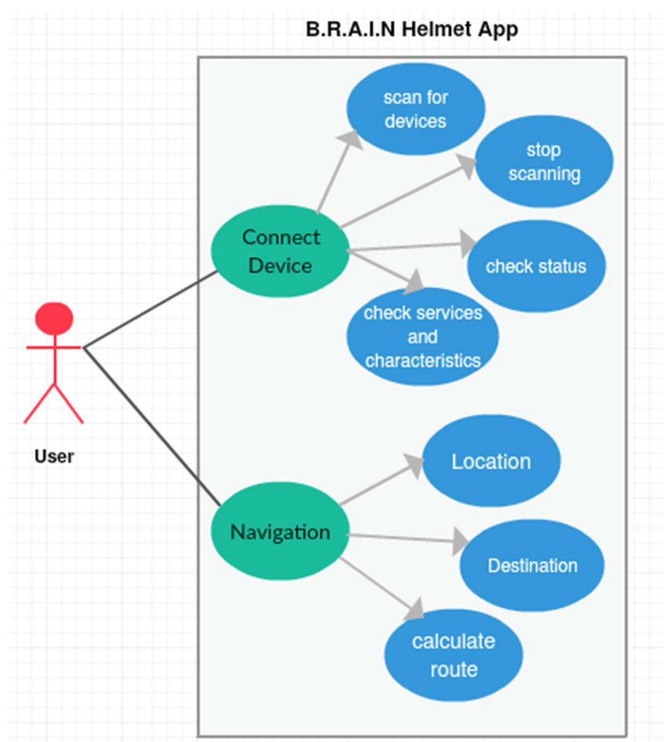

Fig. 12. Breakout diagram of the opportunities and services the BRAIN helmet application provides for the user.

E. *Software Test Environment*

Android Studio is the official building environment for Android Applications. Android Studio is designed to give the programmers a great experience by providing many features and testing methods. There are two different types of testing that can be made using Android Studio. Programmer can test the code by creating a JUnit test that runs on the local JVM. Also, the Application can be tested directly on mobile device. In order to test our software application, software team will do continues testing on Android Studio emulator. Also, they will do some direct test on the mobile device itself between each new stage and progress they will make on the application.

F. *Real Time Application Updating*

The following functionalities need to be met when designing the software: collecting data from real time map, performing some calculations on that data, checking the status of the Bluetooth connection, and sending results of calculations to the mobile device. Since the navigation is real time functionality, we keep checking for updates and send it instantly to main module; allowing the relatable data to be displayed for the user as shown in Figure 13.

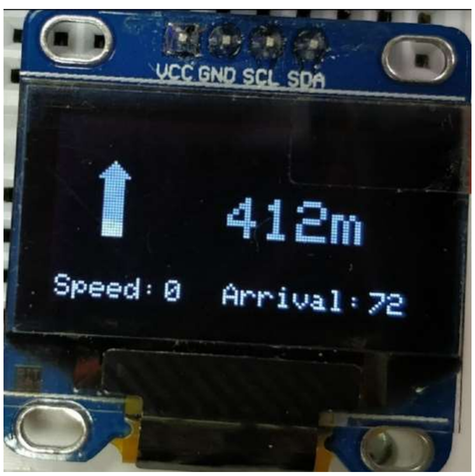

Fig. 13. Heads-Up Display with navigational information.

V. Board Design & Realization

The BRAIN Helmet will consist of two printed circuit boards. The main board will be compacted and packaged with primarily 0805 SMD parts covering the large majority of the board; excluding larger 3mm through hole LEDs, pin headers, and the ATmega2560's crystal. The second board will hold 8 switches and have through hole headers as well to connect to the primary board. Perforated prototyping boards, or vector boards, were originally contemplated but in order to best reduce the size of the BRAIN Helmet electronic group and housing, we have chosen to take the harder route of surface mounting the large majority of components.

The Engineers

Ryan Mortera is a 23-year old graduating Electrical Engineer student. Ryan wants to pursue a career in Electrical Engineering right after he graduates from the University of Central Florida this Fall. Ryan wants to further his education by pursuing a Master Degree.

Stephan Morales is a 23-year old graduating with a bachelor's in Computer Engineering from the University of Central Florida. He wants to pursue a career as an embedded engineer after he graduates in Fall 2017.

Nada Al-Gharabawi is a senior at the University of Central Florida. She is graduating with her Bachelor's in Computer Engineering in December of 2017. She is interested in Software Design and Embedded Systems.

Jordan Yamson is an Electrical Engineering student graduating at the age of 23. Following graduation from the University of Central Florida, Jordan will start his career with Lockheed Martin as an Electrical Engineer associate. While Jordan does not have a set desire to specialize in analog or digital yet, he is taking a position centralized in analog design but with the opportunity to learn and assist in digital design.