

Senior Design I

Smartphone-integrated Heads Up Display for GPS Navigation in Automobiles



University of Central Florida

*College of Electrical Engineering and Computer Science
Dr. Samuel Richie, Dr. Lei Wei*

*College of Optics and Photonics
Dr. David Hagan*

100 Page Submission:

Group 6

Aaron Majdali – Optics and Photonics Engineering
(aaron.majdali@knights.ucf.edu)

Evan Hall - Electrical Engineering (ehall17@knights.ucf.edu)

Logan Glowth - Computer Engineering (logan.glowth@knights.ucf.edu)

Pedrhom Nafisi - Computer Engineering (pedrhomnafisi@knights.ucf.edu)

Table of Contents

1.0 Executive Summary	7
2.0 Project Description	8
2.1 Motivations	8
2.2 Design Constraints:.....	9
2.2.1 General Constraints	9
2.2.2 Economic Constraints	10
2.2.3 Environmental Constraints	11
2.2.4 Social Constraints	12
2.2.5 Legal Constraints	12
2.2.6 Political Constraints	13
2.2.7 Health and Safety Constraints	13
2.2.8 Manufacturability Constraints	14
2.2.9 Sustainability Constraints.....	14
2.3 Engineering Requirement Specifications:.....	15
2.4 Related Standards	17
2.4.1 IEEE 802.15.1: WPAN / Bluetooth:	17
2.4.2 IEEE 802.15.4: LR-WPANs.....	17
2.4.3 IEEE 802.11i: WPA2 and CCM.....	17
2.4.4 NMEA 0183: Data Sequencing	18
2.5 House of Quality:	18
2.6 Project Milestones:.....	19
3.0 Research Related to Project Description.....	22
3.1 Existing Projects and Products	22
3.1.1 Display Design.....	22
3.2 Attaching the HUD:	24
3.3 Display	25
3.3.1 Projector	26
3.4 Small LCD Screen to be Paired with Collimating Lens	27
3.4.1 Option 1: Adafruit ADA938 Screen.....	27
3.4.2 Option 2: ENH-DG128064-66 Transparent LCD	28

3.4.3 Option 3: TP241MC01G transparent OLED screen	29
3.5 Large LED Screen	30
3.5.1 Option 1: W050P40PH01 LCD Screen	30
3.6.3 GT-P25W	32
3.7 Heads Up Display	33
4.0 Components Selection	35
4.1 Power Supply:	35
4.1.1 Option 1: USB Micro-B Breakout Board Product ID: 1833	36
4.1.2 Option 2: BOB-12035	36
4.1.3 Option 3: 2174507-2	36
4.2 Power Regulators:	36
4.2.1 Option 1: LM317T	37
4.2.2 Option 2: LM2596 DC-DC Adjustable Buck Converter 3.2-46V to 1.25-35V Step Down Power Supply High Efficiency Voltage Regulator Module	37
4.2.3 Option 3: TPS61222DCKR	37
4.3 Boost Converters	38
4.3.1 Option 1: Super XL6009 DC-DC Adjustable Step-up Boost Power Converter	38
4.3.2 Option 2: LT1613CS5#TRMPBF	39
4.3.3 Option 3: LM2577-ADJ	39
4.5 Light Sensor:	40
4.5.1 Option 1: Adafruit ALS-PT19 Analog Light Sensor Breakout	40
4.5.2 Option 2: OPT3007YMFR	40
4.5.3 Option 3: Adafruit 161	41
4.6 Mini Speakers:	41
4.6.1 Option 1: Breadboard-Friendly PCB Mount Mini Speaker - 8 Ohm 0.2W	41
4.6.2 Option 2: Mini Metal Speaker w/ Wires - 8 ohm 0.5W	41
5.0 Navigation	41
5.1 Predetermined Routing	42
5.2 Google Cloud Platform API	43
6.0 Application	44

6.1 Application Design	44
7.0 Location Tracking:.....	45
7.1 Google Cloud Platform Location	45
7.2 Mobile Device GPS Location	45
7.3 Standalone GPS Module	46
8.0 Communication:	46
8.1 Wireless Protocol Comparison.....	47
8.2 Bluetooth Low Energy Specifications and Factsheet.....	49
8.2.1 Bluetooth Low Energy Specifications Points of Interest.....	51
8.3 Bluetooth Low Energy Architecture	52
8.4 Bluetooth Packet Format	53
8.5.1 Bluetooth Module Specifications	55
8.5.1 Bluetooth Module Schematic	56
9.0 Crash Detection:	60
9.1 Accelerometer	61
9.2 3G/GPS Module.....	61
9.3 Affects of Impact on Power Delivery and Electronics	63
10.0 Microcontroller:	63
10.1 Texas Instruments MSP430F447	65
10.2 ATmega 2560	65
11.0 Budget:	67
12.0 Testing Plan	69
12.1 Microcontroller Testing.....	69
12.2 Mobile Application Testing	70
12.3 Accelerometer Testing	71
12.4 3G/GPS Module Testing	71
12.5 LCD Screen Testing	72
12.6 LED Testing	72
12.7 BOB-12035 Testing	72
12.8 LM2596 (Step down voltage regulator) Testing.....	73
12.9 LM317T (Step down voltage regulator) Testing	75

12.10 TPS61222DCKR (Step down voltage regulator) Testing	77
12.11 LM2577 (Boost converter) Testing	78
12.12 LT1613 (Boost converter) Testing	79
12.13 Adafruit ALS-PT19 Analog Light Sensor Breakout Testing	80
12.14 Adafruit 161 Photodiode Testing	82
12.15 Breadboard-Friendly PCB Mount Mini Speaker - 8 Ohm 0.2W	84
12.16 Mini Metal Speaker w/ Wires - 8 ohm 0.5W	85
12.17 Bluetooth Prototyping and Testing.....	86
12.17.1 Bluetooth Module Bootloader	86
12.18 Communication tests.....	89
12.19 Packet Sniffing for Testing Both Nodes	90
12.19.1 Packet Sniffing with WireShark.....	91
13.0 Project Hardware Design:.....	93
14.0 Tools	96
14.1 Communication	96
14.1.1 Discord:.....	96
14.1.2 Text Messaging:	96
14.2 File Preservation	97
14.3 Other Software	97
14.3.1 LucidChart:.....	97
15.0 Conclusion	97
16.0 References:	98

Table of Figures

Figure 1: House of Quality	19
Figure 2: An off-the-shelf GPS heads up display.	22
Figure 3: An off-the-shelf ODBII heads up display.	23
Figure 4: An off-the-shelf GPS heads up display that features its own display screen.	24
Figure 5: Idea for Mounting	25
Figure 6: P1+ Mini Projector	26
Figure 7: Adafruit ADA938 Screen	27

Figure 8: ENH-DG128064-66 transparent LCD display	28
Figure 9: TP241MC01G transparent OLED	29
Figure 10: W050P40PH01 LCD screen.....	30
Figure 11: The Luxdrive Endor Star 07007-OW740-N.....	31
Figure 12: XC-10W-C LED.....	32
Figure 13: GT-P25W.....	33
Figure 14: A laser diode and a collimating positive lens. Image	34
Figure 15: A simple angled glass combiner. Image	34
Figure 16: Schematic for TPS61222DCKR	38
Figure 17: Schematic for 5 Volt to 12 Volt using LT1613.....	39
Figure 18: Schematic for the LM2577-ADJ for 5V to 12V	40
Figure 19: AES-CCM Algorithm	51
Figure 20: BLE Architecture	53
Figure 21: BLE Packet Format.....	54
Figure 22: Pinout for Bluetooth Module	Error! Bookmark not defined.
Figure 23: Schematic for ADXL335.....	61
Figure 24: ATmega 2560 Schematic.....	66
Figure 25: ATmega 2560 Schematic Continued.....	67
Figure 26: Choosing an Input Capacitor for LM2596	74
Figure 27: Schematic for LM317T	76
Figure 28: Graph for Output Current for LM317T	76
Figure 29: Schematic for ALS-PT19 Analog Light Sensor	81
Figure 30: Current and Voltage of Analog Light Sensor based on Illuminance...81	
Figure 31: Schematic for Testing Photodiode.....	Error! Bookmark not defined.
Figure 32: Bluetooth Module Bootloader	87
Figure 33: Block Diagram.....	94

Table of Tables

Table 1: HUD Design Constraints	9
Table 2: Wireless Protocol Comparison	47
Table 3: BLE Specifications and Fact Sheet	49
Table 4: PDU Packets.....	54
Table 5: MSP430 and Arduino Comparison	64
Table 6: Budget	68

1.0 Executive Summary

In today's world where mobile technology has become an essential part of our lives, it is often difficult to disconnect and put these mobile devices away for much longer than a few minutes. This becomes a major hazard when getting behind the wheel of a vehicle. There are a myriad of distractions to take into account when driving a vehicle. According to the National Highway Traffic Safety Administration, "distracted driving is dangerous, claiming 3,166 lives in 2017 alone". Distracted driving not only puts the driver's life in danger, but also the lives of the other drivers and passengers on the road, creating dangers such as speeding.

With our Senior Design project, we will design a device that provides a driver with enhanced situational awareness by displaying pertinent information in the driver's field of view. To accomplish this, we want to create our own device using knowledge of hardware and software gained throughout our college careers. This device will contain a Bluetooth module for wireless connectivity, a display module, a power delivery system, an LED array, and a speaker. The device will read information sent by the phone and use a display module to project an image onto the windshield of the vehicle, containing information such as speed limit data or navigational aids. Apart from the display module, we have also discussed adding an LED array to turn on when certain conditions are met. For example, if a driver begins to drive over the speed limit, a red LED will turn on and a speeding announcement will be played over the speaker.

There are vehicle manufacturers that have special packages that contain integrated heads-up displays, but these packages are often associated with premium prices. Additionally, these heads-up displays are not always integrated with advanced navigational awareness features such as speed limit awareness. We want to create a low cost solution that contains advanced functionality beyond what is provided with pre-existing heads-up display systems. To differentiate our device from other products, our device will have low power consumption, with a reduced footprint on the dashboard, and an easily readable and functional display.

We believe that this project will push us to learn more about the advanced systems required to make this device a reality. As seniors in Computer Engineering, Electrical Engineering, and Optics and Photonics Engineering, we will combine our fields of study to effectively and efficiently produce a product that can be used to reduce the amount of distractions and increase situational awareness while driving a vehicle.

2.0 Project Description

2.1 Motivations

The motivation for this project is to demonstrate our knowledge of optics, electrical design, and programming that we have accumulated while studying at the University of Central Florida. Classes such as Electronics, Computer Sciences, Embedded Systems, and Optics have given us in depth knowledge about the processes of engineering in our respective fields of study. It's one thing to take classes that discuss these topics, but it's incredibly beneficial to actually synthesize a physical project using this knowledge.

Upon initial group formation, we set out to determine the best project for our interests and skills. With a team consisting of two computer engineering students, one electrical engineering student, and one optics and photonics engineering student, there were a plethora of options that we could choose from that would prove to be challenging and exciting. The idea of creating a Heads-Up Display for a vehicle stuck out as one that could be well designed and implemented in the timeline of Senior Design. There are programming and hardware design aspects for our computer engineering students to tackle. The device would require power and electrical design that are taken care of by our electrical engineering student. Creating a display that can shine light off of the windshield of a car is a task that can be taken on by our Optics and Photonics student. We believe that there is enough depth in each field to provide equal opportunities for all of us to learn and contribute to creating a great Senior Design Capstone project.

One of the great things about having a Senior Design lecture is that it exposes us to a lot of strategies to making a successful design team. Events such as the Senior Design Bootcamp allowed us to come together and determine how each member of the group can contribute to the project as a whole. Referenced further in this document is a list of project milestones that we have determined to be important to the success of this project. We are going to do our best to stick to these milestones to promote timely and efficient work.

Another goal for this project is to make it as cost efficient as possible. We have not obtained any sponsorships, and as such we will be self-funding everything. As college students, bearing the cost of additional materials can be a burden. We will be sourcing materials to make the most effective product at the lowest price possible, reducing the financial strain on each member of the team.

2.2 Design Constraints:

The following constraints are being placed on the HUD device due to the factors that exist when building a self-funded design in a collegiate setting. The team must adjust our design to match what is consistent with our expected budget, timeline and restrictions set by the University of Central Florida College of Engineering and Computer Science. Other constraints exist due to the knowledge and background of each of our team members

2.2.1 General Constraints

When we started to think about how to implement the HUD Device, we set some general constraints to give us some boundaries for how it should be built. These constraints apply to both the HUD device and the mobile application. Tables 1 and 2 describe the constraints of both the HUD device and mobile application, respectively.

Constraint	The HUD Device shall:
GC.H.1	Include a custom Printed Circuit Board (PCB)
GC.H.2	Not include pre-built components such as Development Boards
GC.H.3	Be designed by December 10, 2019
GC.H.4	Be user friendly
GC.H.5	Be built by April 2020
GC.H.6	Maximize energy efficiency
GC.H.7	Increase driver safety and awareness
GC.H.8	Not interfere with the driver's view of the road
GC.H.9	Not distract the driver in any way
GC.H.10	Be reasonably designed
GC.H.11	Be funded by the students or sponsorship where applicable

Table 1: HUD Design Constraints

Constraint	The Mobile Application shall:
GC.A.1	Be developed for Android devices
GC.A.2	Developed using Android Studio
GC.A.3	Use the Google Cloud Platform API for navigational information
GC.A.4	Be able to connect to the internet from the mobile device's carrier network
GC.A.5	Have a minimalistic design to reduce distraction
GC.A.6	Be user friendly
GC.A.7	Be fully functional by April 2020
GC.A.8	Use the mobile device's Bluetooth capabilities to pair with the HUD Device
GC.A.9	Send data to the HUD Device from Bluetooth
GC.A.10	Not distract the driver in any way
GC.A.11	Be reasonably designed

Table 2: Mobile Application Design Constraints

2.2.2 Economic Constraints

Economic constraints are extremely important when considering how the project is designed and implemented. Economies are dynamically changing and can be different depending on what country or market the project is created and sold in. Our project is targeted at the United States market; however, parts are sourced from different countries and could be subject to certain taxes, importing fees, and tariffs. All these factors need to be considered when determining the value of the final product. Table 3 describes the economic constraints that are placed on this project.

Constraint	Economic Constraint
EC.1	The project shall cost no more than \$300 USD
EC.2	The project shall be funded by members of the group
EC.3	The mobile application must not add additional costs to the project
EC.4	The device and application must be created with US Market in mind

Table 3: Economic Constraints

2.2.3 Environmental Constraints

Environmental constraints are limiting factors due to the impact of used materials such as production, disposal, energy consumption, and emissions of the products involved. Our environmental constraints are pertaining to the energy efficiency of our HUD Device, the energy efficiency of the mobile device running our custom application, and the emissions of the vehicle that is being used for testing of the HUD Device. Table 4 references the environmental constraints placed on this project.

Constraint	Environmental Constraint
ENVC.1	The project shall be energy efficient
ENVC.2	The HUD Device must be powered by the vehicle containing it
ENVC.3	The vehicle used must be compliant with economic policies in place by the United States Government.
ENVC.4	The project shall not use hazardous materials

Table 4: Environmental Constraints

2.2.4 Social Constraints

Social Constraints are due to societal norms, traditions, and other factors that affect how humans interact and view the project. Our project will need to take special consideration for social constraints due to the nature of human use. Table 5 describes each social constraint placed on our project.

Constraint	Social Constraint
SC.1	The HUD Device must be user friendly
SC.2	The mobile application must be user friendly
SC.3	The project must not violate social norms
SC.4	TODO

Table 5: Social Constraints

2.2.5 Legal Constraints

As it stands, there are no federal laws that limit or control the use of a heads up display in a vehicle. Similarly, there are no laws in any of the 50 states that regulate the use of heads up displays. Because of this, there are many car manufacturers that integrate heads up displays into their vehicles, all of which are 50 state legal. Heads up displays are also not subject to laws in Canada [4]. In fact, heads up displays seem to really have no legal restrictions in North America or Europe. As this device is being designed with the United States market in mind, the GPS heads up display should face few to no legal issues. One possible point of concern could be regulations on flashing lights that are visible from vehicles. This rules out the use of red or blue LED's to illuminate the heads up display. As we are designing the system with white LED's in mind, there will be no legal repercussions.

2.2.6 Political Constraints

Political constraints are due to the implications of integrity and motivations by parties affiliated with a certain subject. There are no political intentions that surround this project, the members of the team, or the potential users of the project.

2.2.7 Health and Safety Constraints

Health and safety constraints are extremely important to keeping those who interact with the project and its components from sustaining serious injury or harm. Table 6 describes the health and safety constraints that are placed on our project.

Constraint	Health and Safety Constraint
HSC.1	Any electrical component shall be used within the designed specifications by the original manufacturer.
HSC.2	The HUD Device must not add any distractions to the driver
HSC.3	The HUD Device must not hinder the driver's view of the road
HSC.4	The mobile application must not add any distractions to the driver
HSC.5	The vehicle used for testing must be road-worthy and properly insured
HSC.6	Any electrical connections must be properly grounded

Table 6: Health and Safety Constraints

2.2.8 Manufacturability Constraints

Manufacturability constraints are due to the components required to realize the HUD Device, software for the application, or any other physical or virtual part of the project. To ensure that the project can be manufactured within our budget and timeline, each component of the project will be created or implemented using parts that are readily available or easily obtained. The College of Engineering and Computer Science at the University of Central Florida has a myriad of resources for students to use, enabling the project to be manufactured at a minimal cost. Table 7 describes the manufacturability constraints of this project.

Constraint	Manufacturability Constraint
MC.1	The project shall use parts that are readily available or easily obtained
MC.2	The project shall take advantage of labs and tools provided by the College of Engineering and Computer Science of the University of Central Florida

Table 7: Manufacturability Constraints

2.2.9 Sustainability Constraints

Sustainability constraints are due to the ability for the project to be supported and maintained after being produced and completed. Once a product has launched, it must be maintained to ensure that it remains in a useable state for the lifetime of the product. Unforeseen issues can arise long after the development stage has completed. Table 8 describes the sustainability constraints of the project.

Constraint	Environmental Constraint
SUSC.1	The project shall be supported and updated after completion
SUSC.2	The project shall allow for further development as needed

Table 8: Environmental Constraints

2.3 Engineering Requirement Specifications:

The following requirements set a defined scope for how the project will be designed and built. Each of the requirements will give the team a guideline for the end goal. Referencing this table will become useful for staying within the bounds of what we are going to create.

Requirement	
The HUD Device shall:	
R.1	Weigh no more than 1 lb.
R.2	Not exceed 5x3x2 in. in size
R.3	Run off a USB Port
R.3.1	From this port there must be a voltage step down to run about 5 Volts or lower for low power consumption
R.4	Interface with a mobile phone via a Bluetooth connection
R.5	This will be compatible with android devices
R.6	Be able to display GPS data onto a windshield or dedicated screen
R.7	Have good resolution for easy viewing
R.8	The integration of software will be done using APIs provided by Google Cloud Platform
R.9	If the data displayed, such as google maps, requires sound, there will be a speaker on the side of the device
R.10	Be able to operate within a temperature range of -20 C to 50 C
R.11	Be able to operate in and be stored in direct sunlight
R.12	Be capable of adjusting to be visible with different windshield designs
R.13	Produce an image that is visible when viewed through polarized sunglasses
R.14	Be able to automatically adjust its brightness level according to the amount of ambient light

Table 9: Engineering Requirement Specifications

2.4 Related Standards

As with any quality engineering project, there are standards that have to be adhered to in order to ensure that a product is safe, reliable, and compatible with other systems. Such standards can involve communication, data storage, or even legal considerations. Here, we list relevant standards and laws and state how we will ensure that our heads- up display conforms to them.

2.4.1 IEEE 802.15.1: WPAN / Bluetooth:

The IEEE 802.15.1 standard applies to wireless personal area networks (WPAN) and the construction of them through the use of Bluetooth technology for small, low power devices [1]. The standard contains a wide variety of clauses, message types, data-formats, and structured formats. Within the clauses state specifications for how the physical layer as well as the Medium Access Control (MAC) must operate in order to meet this standard.

2.4.2 IEEE 802.15.4: LR-WPANs

The IEEE 802.15.4 is a technical standard which defines the operation of low-rate wireless personal area networks (LR-WPANs) used by our BLE module. It specifies the physical layer and media access control for LR-WPANs, and is maintained by the IEEE 802.15 working group, which defined the standard in 2003. The standard specifies the architecture and topology of the wireless protocol.

2.4.3 IEEE 802.11i: WPA2 and CCM

The IEEE 802.11i standard specifies security requirements and procedures for wireless networks, replacing the short authentication and privacy clause of the legacy standard with a detailed security clause. This standard also deprecated the privacy and security algorithm WEP in favor of the new and improved WPA2 algorithm which uses the CCM mode that is implemented in our Bluetooth mechanism.

2.4.4 NMEA 0183: Data Sequencing

The NMEA 0183 standard is a technical standard governed by the National Marine Electronics Association, designed to standardize the format in which data is transmitted between transmitting and receiving devices. The data contains 8-bits synchronized to a 4800 Baud rate. There is one stop bit, and no parity or handshake bits. Both the NEO-6M GPS Module and the SIMCom 5230 3G/GPS Module we could be using provides output data formatted to the NMEA 0183 Standard. This means that we will need to set up our programming, hardware design, and implementation of the heads-up display and accompanying applications to support data processing according to this standard.

2.5 House of Quality:

There are many factors that need to be analyzed and discussed when designing our Heads-Up Display Device. Each of these factors has an impact on the implementation of our project. The House of Quality chart for our project, located in Figure 1, weighs the tradeoffs and effects of each factor on the outcome of our project.

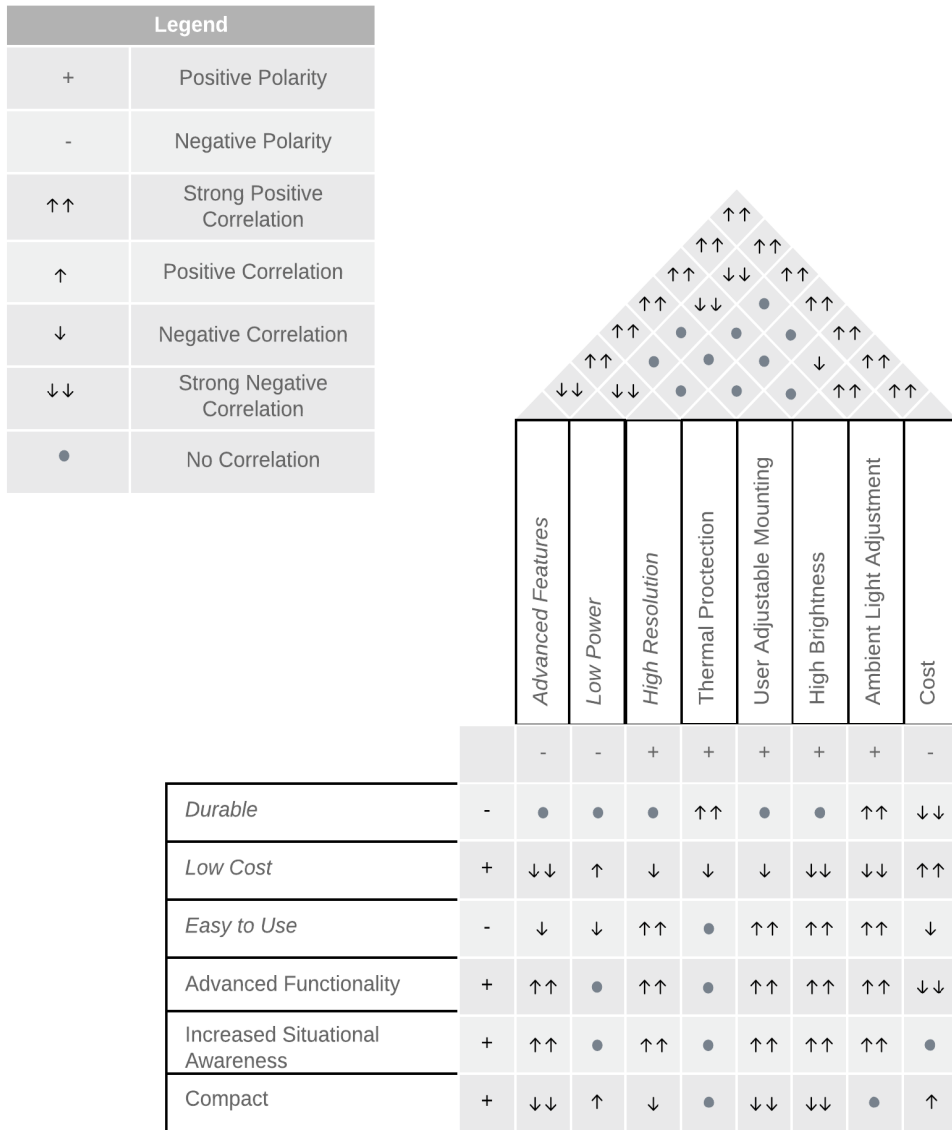


Figure 1: House of Quality

2.6 Project Milestones:

At the beginning of Senior Design I, we spent some time discussing projects that be both challenging and exciting. Once we had narrowed down what we wanted to do, we needed to create a timeline for us to design and implement the project. The senior design project spans across our final two semesters at the University of Central Florida. Projects of this scale do not happen without proper planning

and time management. Table 10 outlines our timeline for certain stages of the project to be completed throughout Senior Design I.

Task	Start Date	End Date	Status
Senior Design I			
Create Groups	8/30/19	8/30/19	Completed
Project Ideas	8/31/19	9/6/19	Completed
Role Designation	9/7/19	9/8/19	Completed
Initial Project Documentation - Divide and Conquer	9/13/19	9/20/19	Completed
Start Design Documentation	9/23/19	9/23/19	In Progress
Table of Contents	9/23/19	12/2/19	In Progress
Research Individual Parts	9/23/19	12/2/19	In Progress
Schematic Design	9/23/19	12/2/19	In Progress
60 Page Rough Draft	9/23/19	11/1/19	In Progress
100 Page Submission	11/1/19	12/2/19	In Progress
Parts Acquisition	11/30/19	1/1/19	In Progress

Table 10: Senior Design 1 Project Milestones

Upon reaching the second semester of Senior Design, the team will have already designed and planned out all of the necessary milestones to achieve success for our project. We will have already ordered and received our initial batch of parts and components. Our initially testing will have been done and completed. We are aiming to design and finalize our PCB implementation by the fifth week of Senior Design II. The testing process we used throughout Senior Design will be crucial to completing our design with minimal conflicts. Table 11 describes our expected timeline for Senior Design II.

Task	Start Date	End Date	Status
Senior Design II			
All Parts Must Have Arrived	1/1/19	1/15/19	In Progress
Schematic Implementation and Testing	1/16/19	2/28/19	In Progress
Testing Design	3/1/19	3/30/19	In Progress
Final Prototype	4/1/19	4/15/19	In Progress
Miscellaneous Time for Further Troubleshooting	4/15/19	4/20/19	In Progress
Panel Presentation	TBA	TBA	In Progress

Table 11: Senior Design II Project Milestones

We believe that the tables notated above will give us ample time to complete the project successfully. It will require every member of the group to stay focused throughout the duration of Senior Design I and II. We expect that there are going to be adverse situations that will arise throughout the design process. Throughout our college careers, we have learned how to deal with unexpected issues that can push deadlines back by days or even weeks. We are confident that this experience will allow us to handle any situations moving forward.

3.0 Research Related to Project Description

3.1 Existing Projects and Products

Parallax-free sighting systems have been in use by the military since before World War II in fighter planes. Indeed, military aircraft have been the primary use case for a heads up display for much of the technology's life. More recently, heads up displays have made their way into consumer vehicles. What's more, there are available third-party heads up display options that read out car diagnostic data and can even hold a cell phone to act as a heads-up GPS.

3.1.1 Display Design

There are different approaches to creating a heads-up display. From these, we know that brightness is a common concern with third-party models. This section will cover existing variations of heads up displays.



Figure 2: An off-the-shelf GPS heads up display.

Figure 2 shows a third-party solution that utilizes a standard smartphone as a display. The phone is held in place via a mount that also attached to a small transparent screen. The image from the phone's screen is reflected off the mount's screen and is visible to the driver. The phone requires a special app to display

GPS data and speed that can be seen by the driver i.e. the image is reversed so that the reflection is readable to the driver. Note that when in use the heads up display wholly monopolizes the phone. If the driver wishes to use the phone in any way, the phone must be removed from the mount and the app must be closed.



Figure 3: An off-the-shelf OBDII heads up display.

Another possible design for a car heads up display is shown in Figure 3. This design uses a module with a dedicated LED-lit instrument cluster. While this particular implementation is unable to display GPS data, it is able to read information from a vehicle's OBDII port. This allows the device to display fuel efficiency, speed, and tachometer data read from the vehicle itself. Note how the device uses the vehicle's windshield as the screen. This simplifies the use of the device, but introduces the problem of possibly having a display that is too dim to see in direct sunlight. This product tries to alleviate that by including a reflective film to place on the window.



Figure 4: An off-the-shelf GPS heads up display that features its own display screen.

One more example of an add-on heads up GPS is shown in Figure 4. The device is attached to the sun visor and displays its own image onto a screen. The phone is attached via USB and uses a specialized app for displaying the appropriate data. The phone can be used to play music while the GPS heads up display is used. The device is battery powered and therefore has a finite run time before it needs to be recharged. Also, the sun visor is unusable while the device is attached.

3.2 Attaching the HUD:

This is going to be a challenge because how the HUD is mounted has to be universal for any car. Some ideas that came to mind were mounting from the visor, but after thought this wouldn't work because of the distance to the windshield is too far, plus the dangling power cord is not very aesthetic. The next idea would be to use adjustable arms that are positioned onto the windshield via suction cups. After some deliberation this uses too much space on the windshield and could be a distraction/safety hazard with less viewing area. A promising idea would be to mount it, using clips or some mild sticky but removable adhesive, to the dashboard behind the steering wheel. This seems to be a tried and true method after

researching other products and seeing how they mounted their devices. The idea comes from the product in Figure 5 below.

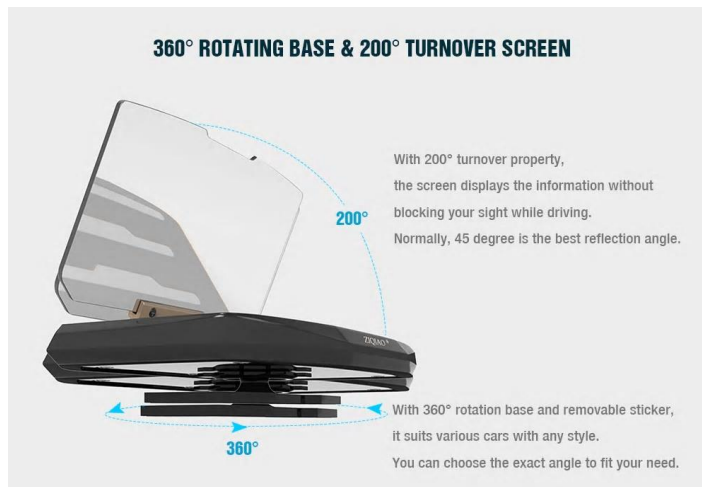


Figure 5: Idea for Mounting

This product also uses a swivel for horizontal movement whereas it would be better for our product to swivel horizontally and be able to move vertically to adjust the viewing angle.

In order to do this the first item would be the adhesive. The only good fit would be a removable adhesive that will not damage a car's interior. A candidate would be the tesa Powerstrip double sided tape. It can hold up to 2 kg (4.4092 lbs) which would be strong enough to hold the HUD in place. Another good thing about this product is if the tape is pulled it will release very easily, this way there is no damage to the interior and can be taken off if wanted. This would be put on the bottom of the base platform of the HUD. The next item would be the swivel. It needs to be able to move vertically at the minimum. This would be connected to the top of the base platform.

3.3 Display

The display is a challenge that requires a lot of thought and care. One of the primary challenges of the display will be one that can adequately show GPS and other information, all the while in bright sunlight. Consumer solutions involve a sort of reflective film that can be directly applied to a piece of glass, usually the windshield. Even if we choose to go with a separate glass screen that is attached to the device, it may be worthwhile to also include a reflective film. Because the device is to receive its power from a USB port, an otherwise adequate display may consume too much of the power budget. Another consideration is whether we want to use some sort of projector, a simple LCD screen, or a small LCD screen with a

collimating lens. The advantages of a projector are that it can be collimated easily to show an image that focuses at an infinite distance. The downsides are its higher power consumption, larger size, and larger price tag. The advantages of using a small LCD with a collimating lens are lower power consumption, smaller footprint, and being able to be focused at infinity. The disadvantages are the difficulty of finding a smaller screen with adequate brightness, lower resolution, and issues with using a collimating lens, such as distortion. The advantages of using a larger LCD are the ability to simply reflect the screen without any extra lenses, high resolution, and ease of finding one with adequate brightness. The disadvantages are higher power consumption, inability to be focused at infinity, larger size, and possibly inadequate brightness within a reasonable financial and size budget.

3.3.1 Projector

The P1+ Mini Projector shown in Figure 6 is very compact, which is extremely valuable in a space-constrained environment. However, the device itself is only capable of 30 lumens output. This isn't very bright and would easily be washed out in bright sunlight. The unit itself can be found for roughly \$100, which is more than we are willing to spend at this time. It is also a self-contained OEM solution and is therefore unsuitable for use in our project. In fact, upon further research it was found that there are few to no bare projectors to be had.

The P1+ Mini Projector was the only projector that was seriously considered. All other projectors that were reviewed were inadequate in multiple ways. Because projectors of suitable brightness and size are too expensive, too power-hungry, and too pre-built, it was decided that projectors would not work for our heads up display.



Figure 6: P1+ Mini Projector

3.4 Small LCD Screen to be Paired with Collimating Lens

3.4.1 Option 1: Adafruit ADA938 Screen

The Adafruit ADA938, shown in Figure 7, is a 1.3" 128x64 black/white OLED screen. Because the screen itself is small, it is a candidate for use with a magnifying lens to be collimated and focused at infinity. On top of that, the screen uses OLED and is monochromatic black/white. Since OLED black is done by switching off individual pixels, the contrast of this screen is excellent. As a result, the only light that would be seen would be from the active pixels and the resulting reflected image would have no extra "black" in the background. The price is not too bad, as the screen can be found for \$10 or less. However, the screen is only capable of 100 cd/m² and will not be bright enough for our purposes. What's more, the ADA938 uses OLED. OLED pixels have a finite lifespan, and so our screen can be subject to burn-in or general loss of brightness in as little as 1000 hours! While the thought of amazing contrast is tempting, the ADA938 is not likely to be considered. Of all the reasons not to use this screen, the major deciding factor is brightness.

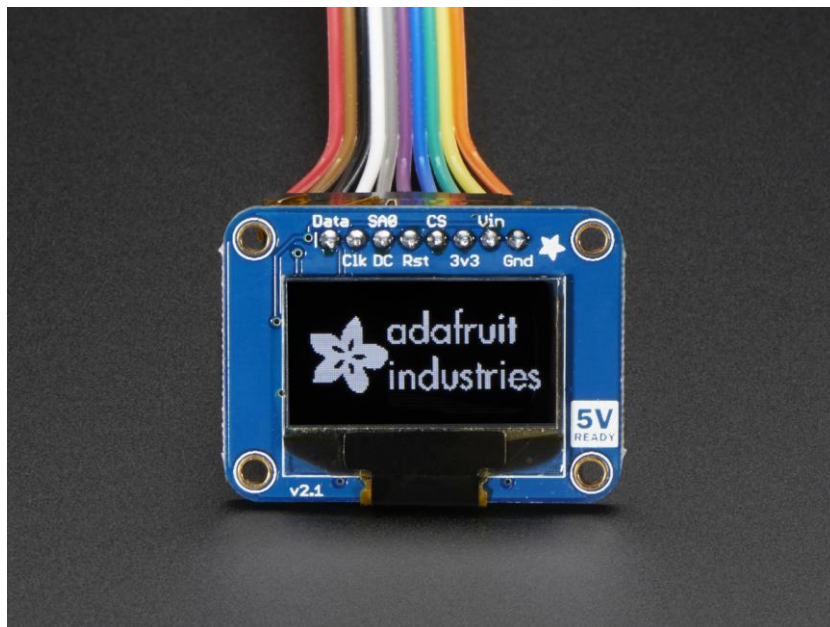


Figure 7: Adafruit ADA938 Screen

3.4.2 Option 2: ENH-DG128064-66 Transparent LCD

The ENH-DG128064-66 shown in Figure 8 is another small screen. Unlike the ADA938, this screen uses standard LCD technology and is transparent. This gives us the benefit of a less expensive display that will not suffer from burn-in and also gives us the ability to use a separate LED back light of our choosing. This screen, like the ADA938, has a resolution of 128x64, which is not a high resolution but will be readable when used. The key feature is the ability to be paired with a high-intensity LED of our choosing. This gives us the ability to pick an LED that meets our brightness and power consumption requirements while fitting within our size and financial constraints. This screen can be found for as little as \$3 online and can be ordered at that price from multiple sources. The low price gives us the option of ordering multiple screens to be combined in a 2x1 or 2x2 setup to achieve a higher resolution that can be used to create higher quality images for the driver. While that would involve extra cost and complexity, these screens are so inexpensive that it could be viable.



Figure 8: ENH-DG128064-66 transparent LCD display

3.4.3 Option 3: TP241MC01G transparent OLED screen

The TP241MC01G shown in Figure 9 is another small screen that was considered. Unlike the previous two screens, this model has a higher resolution of 128x160. This would give us a much higher resolution that would allow for more detailed graphics to be displayed. This screen is also capable of showing color images, which would give us the ability to use multiple colors in our display. One massive benefit of this screen is that, like the ENH-DG128064-66, this screen is transparent. This would allow us to choose an LED light source that would produce a satisfactory amount of screen brightness. This screen would be roughly twice as large as our other two options, at 2.4" across. The larger size would require the use of a larger positive collimator lens. This screen is also using OLED technology, which would result in a noticeably finite lifespan of our heads up display. The price of this display dwarfs that of the others we researched, as the TP241MC01G is difficult to find for less than \$80. Because of its high price, larger size, and limited lifespan, this screen will likely be passed over in favor of a more compact and affordable option.

Because a heads up display works best with a collimated image that is focused at infinity, using a small screen with a magnifying lens will likely be the solution we choose for this project. Also, due to brightness considerations, we know that a transparent LCD-type screen will be ideal because we have the freedom to choose an adequately bright LED to illuminate the screen.

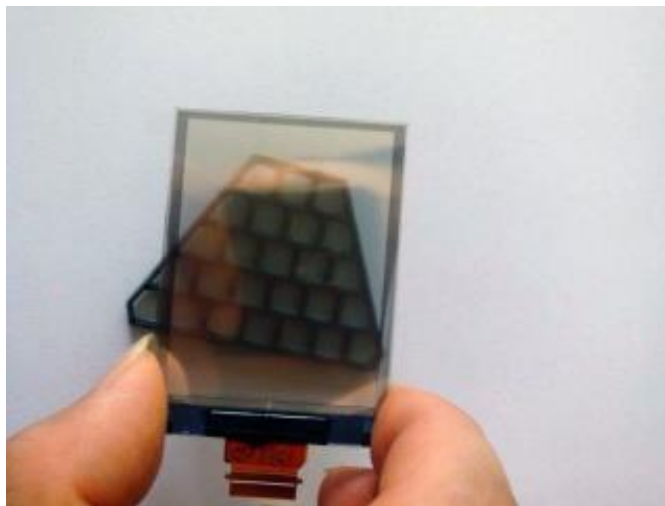


Figure 9: TP241MC01G transparent OLED

3.5 Large LED Screen

3.5.1 Option 1: W050P40PH01 LCD Screen

The LCD screen shown in Figure 10 is a 5" diagonal LED backlit LCD display with RGB capability. The screen is affordable and, more importantly, bright. With a brightness of 1000 cd/m², the screen should be bright enough to be seen in daylight. However, this display brightness may be at the lower limits of what is actually visible on a bright sunny day. What's more, this arrangement does not allow for a collimating lens. Instead, the screen will simply reflect off of a glass surface and its reflection will be focused at a finite distance. While this is common for aftermarket heads up solutions, this does not allow the image to be in focus at all distances the driver may be looking.



Figure 10: W050P40PH01 LCD screen.

3.6 LED Backlighting

If we are to use a transparent LCD or OLED screen, then we will need to use very bright LED's as the light source. In fact, the ability to choose bright LED's is one of the primary reasons for choosing a transparent LCD in the first place. By choosing bright LED's, we gain the ability to tailor the brightness requirements to our needs. We even have the option of adjusting brightness independent of the LCD display. If we so choose, we can pair the LED with a photodiode or other light-detecting device in order to automatically adjust LED brightness based on the amount of ambient light.

3.6.1 Luxdrive Endor Star 07007-OW740-N

The Luxdrive 07007-OW740-N in Figure 11 is a small PCB with 3 neutral white (4000K) LED's mounted on it. The setup is intended and marketed towards those who wish to upgrade their flashlights to something brighter. The LED setup is capable of over 500 lumens with a concentrated beam. The setup is rated at 9.0V and up to 700mA, giving it a max power consumption of 6.3W. Unfortunately, that is pretty close to our max power budget of 10W. While it may not be necessary for us to power the device at the full 700mA all the time, having a device that sucks up over half of our power budget would be a fairly large constraint. If we were to use this option, it may be necessary to redesign our input power and switch from using a USB port to using the cigarette lighter.

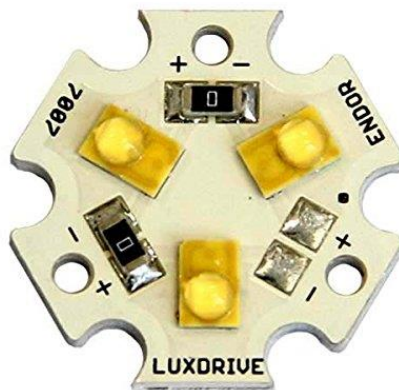


Figure 11: The Luxdrive Endor Star 07007-OW740-N

3.6.2 XC-10W-C

The XC-10W-C in Figure 12 is a single LED that is listed as consuming 10W, or the entire output of a USB port. This comes from using a 10V source at up to 1000mA. However, at full power this LED is capable of emitting 1000 lumens. This would require a heat sink, as the LED circuitry would try to protect itself and shut down due to heat after a short matter of time. This can be avoided by using a heat sink on the LED, as well as limiting the input current of the device. It is unlikely that we would need the full 1000 lumens, but having the capability is nice. This particular LED has a color temperature of 6500K, but there is a version with a color temperature of 3500K.

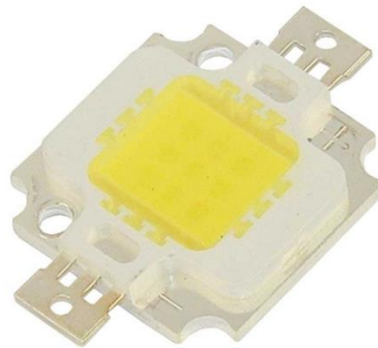


Figure 12: XC-10W-C LED

3.6.3 GT-P25W

The Getian GT-P25W in Figure 13 is another white LED, with a max forward current of 1200mA and a max forward voltage of 11V, for a total of 13.2W max power. In real operation, it runs closer to 10W. The LED is warm white, with a color temperature of 3200K. It would need a heat sink for proper operation. The advantage to this LED is its color temperature and its wide operating power range. Depending on our needs, we may not need the full power the LED provides to get a good projected image. Its color temperature means it will be easier on the eyes after prolonged use, compared to cool white.

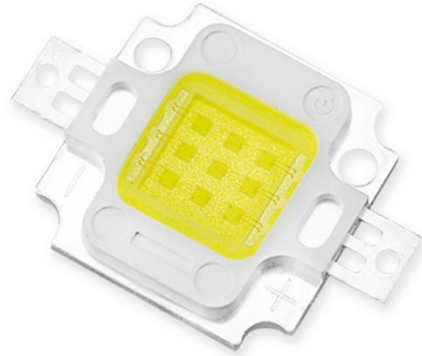


Figure 13: GT-P25W

3.7 Heads Up Display

The main feature of this device is that it is to display GPS information via a heads up display. Heads up displays have been in use, in some form or another, since World War II, where they found use in fighter planes. In 1942, the British Royal Air Force experimented with projecting information from the radar onto a flat screen that also displayed the plane's gyroscopic gunsight [6]. The inclusion of the radar readout onto the screen allowed pilots to more quickly engage targets while flying at night. The heads up display emerged again in the Royal Air Force, who coined the term "heads up display" in the late 1950's [7]. Heads up displays then went on to be included in different NATO and Warsaw Pact jet fighters. In more modern times, heads up displays can be found in commercial aircraft and even consumer cars and trucks.

Heads up displays work by using three main components. These components are the projector, the combiner, and the video generation computer [8]. The projector unit uses optical components to collimate the image. To collimate an image, a screen or other display is placed at the focal point of a positive lens or negative mirror. When an object is placed at the focal point of a positive lens or negative mirror, the resulting image is focused at infinity. Instead of the resulting rays converging at a single point somewhere past the optical component, the rays instead stay entirely parallel to each other to an infinite distance. The result of using a collimator in a heads up display is that the image is always in focus, whether the viewer is looking at an object 5 feet or 5 miles away.

The combiner is simply the medium used to overlay the collimated image with some other image, usually to overlay the heads up display image onto a view of the outside world. For this, all that is needed is a piece of transparent glass. In many heads-up display applications, the combiner is a vertical or near-vertical sheet of glass that is placed within a few feet of the user. In most jet fighter cockpits and in some commercial HUD solutions, the combiner is a vertical or near-vertical glass sheet. In some consumer cars that feature heads up displays, the windshield itself is used as the combiner.

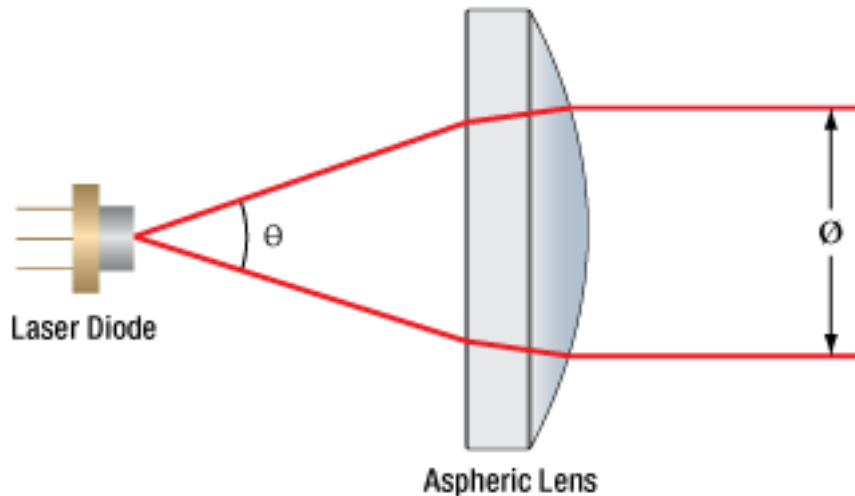


Figure 14: A laser diode and a collimating positive lens. [Image](#)

The final component of a standard heads up display is the video generation computer. The video generation computer is all of the necessary hardware and software to process data as input and give data as output that can be transformed by the projector into an image. As computers have become more advanced, video



Figure 15: A simple angled glass combiner. [Image](#)

generation computers have become faster and more powerful while also using less power and taking up less space. Because of this, video generation computers can be used the heads up displays of a wider variety of vehicles, and even in fields such as augmented reality. Modern-day systems such as Arduino or Raspberry Pi have enough power to be used as video generation computers and in many cases even have pinouts for display devices built into the board itself. Back before the transistor was in common use, such computers would use vacuum tubes that requires much more space and power. This limited the use of video generation computers to vehicles with enough space, like larger aircraft. Some of these systems are specified for different nations' militaries or corporate designs, so even otherwise modern aircraft may still be using vacuum-tube-based video generation computers.

4.0 Components Selection

4.1 Power Supply:

There are several USB types from the plug in side to the connector. The most prominent are the USB 2.0 type A and USB 3.0 type. These plug versions both provide 5 volts nominal. The difference between the USB 2.0 and USB 3.0 type A is that the USB 2.0 type A provides 500 mA compared to the USB 3.0 which can provide 1.5 to 3 A over the 5 volt bus. The other big difference was that the USB 2.0 has a throughput 480 Mb/s and the USB 3.0 has a throughput of up to 5 Gb/s. For power needs it is not as important. The next decision is whether to use a USB micro B or USB type C connection ends. The micro USB is more simplistic with only four pins, two for power and two for data transfer. The USB type C has an additional five pins, in terms of current three of those wires are for standard downstream port, a charging downstream port, and a dedicated charging port. The charging downstream port and dedicated downstream charging port supports up to 1.5 A.

To supply power to the HUD the USB 3.0 A to micro type B connector and cable will be used. The reason why is because the micro type B is cheaper and more readily available with breakout boards that allow for easy access to the two pins that are needed for power. They may have a lower rating but the good thing about USB is that you can plug any USB device into any USB cable and into any USB port. The initial idea was to use the cigarette port, but more and more cars are solely using the USB connectors in cars and doing away with the cigarette ports. If that is not an option, a typical car charger that has a USB connector will still work. Below are three choices that use a female connector USB board mount. The factors that go into this decision are cost, accountability, and time to ship.

4.1.1 Option 1: USB Micro-B Breakout Board Product ID: 1833

This first option comes from the online supplier Adafruit. They offer a breakout board with the female USB type B connector and 5 pins broken out to easily attach wires to supply power to a device. In addition to easy mounting and use they even supply a small stick of 0.1" header so it can be soldered on and plugged into a breadboard. It has through hole shielding pads for a strong connection. It costs \$1.50 and can ship in less than a week. Also, there was a video demonstration that shows how robust this connector is. The datasheet also shows that this item can withstand temperatures from -30 to 80 degrees Celsius, which helps to comply with our need for higher temperature resistance.

4.1.2 Option 2: BOB-12035

This is another breakout board from DigiKey. It seemingly offers the same as the breakout board from Adafruit, with a female USB type B connector with 5 pins broken out for easy connection. Looking at the datasheet did not give as much information. It could be assumed that it can withstand higher temperatures of a car, but it is not known for sure. This board costs \$2.50 and can also be shipped in a week. This board does not seem as robust, but it is still cheap and fits the functional requirements.

4.1.3 Option 3: 2174507-2

This third option is a little less friendly but looks more professional. This USB type B female connector is just the connector part. It does not offer the easy access to the pins as the other two options provided. This part is more prone to come off as it does not have the reliable through hole mounts rather this would just be soldered onto the PCB and could prove a problem with repeated connects and disconnects. This part is \$1.87 and can also ship within a week. Looking at the datasheet also gave less than desired information, still assuming it could resist temperatures in a car it is not explicitly given.

4.2 Power Regulators:

The next item on the list is something that can convert the 5 volts to something smaller for the smaller components. The simplest item to use for this would be a linear regulator but that can only be tuned using the adjustment pin with resistors, if that would be adequate enough. Otherwise a buck converter might have to be used a tunable potentiometer for really precise voltage regulation.

4.2.1 Option 1: LM317T

This is a basic linear regulator which can take in 5 volts and make it into something smaller. The drawback of using linear regulators is that it produces a good amount of heat. That means getting a heat sink, which means more space taken up inside the unit. The LM317T can operate between 0 and 125 degrees Celsius. If in the event the part overheats it has over current and over temperature protection. The regulator can take in a maximum of 40 volts and have a minimum of 1.2 volts minimum with tuning on the output of the adjustment pin. The output current is up to 1.5 amps. But they're very cost effective, this particular part is only \$0.64, and through digi-key can ship out on the day of purchase.

4.2.2 Option 2: LM2596 DC-DC Adjustable Buck Converter 3.2-46V to 1.25-35V Step Down Power Supply High Efficiency Voltage Regulator Module

The next option is a buck converter. The only downside to a buck converter is that they are marginally more expensive, this is a four pack so individually they would be about \$2. With that being said, the buck converter can take in 3.2 volts to 46 volts, having an output of 1.25 volts to 35 volts with a maximum of 3 amps output. The trick here is that the input must be 1.5 volts higher than the output, so not that we would need it, but it cannot be used as a boost converter. To set this up, connect the input to the input terminals and output to the output terminals. Then tune the potentiometer to the desired voltage level. Another thing is that as long as this buck converter isn't used for very long periods of time, it's heat efficient and will not require a heat sink.

4.2.3 Option 3: TPS61222DCKR

This is a tiny boost converter that would take in 5 volts and convert it to anything between 1.8 to 6 volts. It has a high efficiency above 90 percent for 5 volt input to lower output, but with higher output current. This is a good choice because it operates at -40 to 85 degrees Celsius, and will not require a heatsink to dissipate power loss. In addition it has output overvoltage, overtemperature, and input undervoltage lockout protection. This comes with its own schematic and set of equations for making the output voltage adjustable. Ideally this will provide power to most of the HUD, except the screen display. This boost converter comes from Texas Instruments and sells for one dollar and ships in five days.

The simple equation for determining output voltage is $R_1 = R_2((V_{out}/V_{FB})-1)$. V_{FB} should be at 500 mV so choosing R_1 and R_2 becomes easy.

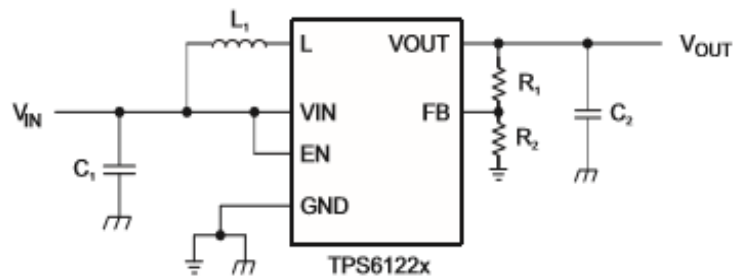


Figure 16: Schematic for TPS61222DCKR

4.3 Boost Converters

Boost converters are an easy way to create higher voltage than the input voltage. The trade off is that the higher voltage reduces the amount of current provided from the input. The basic setup of a boost converter is an inductor in series with a voltage source, with a switch to ground and then a diode and capacitor in series which are in parallel to the switch. The way it works is when the inductor is connected to ground via the switch current flows through it for a very small amount of time that allows magnetic energy to be stored inside of it, and when the switch is opened the polarity of the inductor changes so current flows through the diode and two sources are now in series which charge the capacitor. The switch that allows this to happen has to cycle on and off fast enough to not ruin the inductor, usually performed by a switching device.

Initial estimates, ranging from 10 to 12 Watts, show that the power consumption of the device will be a little higher than what a single USB port alone can provide. So to combat this a boost converter will be used to generate a higher output voltage source primarily for the bright screen display that is required. Boost converters are cheap and can either be bought or made. The downside of using them is they take up room and generate heat.

4.3.1 Option 1: Super XL6009 DC-DC Adjustable Step-up Boost Power Converter

This step-up booster is on par with what is required. Being able to take in 5 volts as the input and being able to output about 12 volts at 0.8 amps, this provides 9.6 Watts of power. This is at the lower end of the estimate, but should be able to handle the power requirements of the display. In addition, the operating temperature of this device is between -40 to 85 degrees Celsius. This has a very simple implementation design where the input voltage is connected to the input terminals and the output has output terminals. A plus side too, it that the output can be tuned with the adjustable potentiometer on the boost converter for precise output conditions. This product comes in a set of two for \$6.93 from amazon.

4.3.2 Option 2: LT1613CS5#TRMPBF

This option is a build it yourself option. The LT1613 is the integrated circuit only. To make it a boost converter the rest of the circuit is made with other elements attached to the LT1613. There is a schematic provided already to boost 5 volts to 12 volts at 130 milliamps. This gives a smaller power output of around 1.56 Watts, which is on the much lower end of the power requirement that was needed. The problem with this option is that there is a very long manufacturing lead time to get this particular piece, it would take about 8 weeks to ship. The single component is cheaper at \$4.28. Figure 17 includes the schematic for the 5 volt to 12 volt conversion.

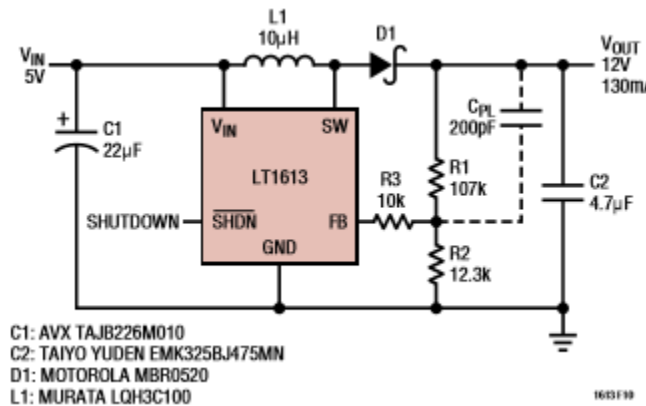


Figure 17: Schematic for 5 Volt to 12 Volt using LT1613

4.3.3 Option 3: LM2577-ADJ

The LM2577-ADJ is an adjustable boost DC-DC switching regulator. This is also a build it yourself option. This component has a wide input voltage range from 3.5 to 40 volts. With simple schematic design, the LM2577 can be made to boost 5 volts to 12 volts at 800 milliamps. This results in a power output of 9.6 Watts. This is still on the lower side of the anticipated power requirement, but should be enough to power the bright display. The LM2577 will operate at around 25 degrees Celsius when at 12 volts. In addition this part is readily available to be shipped with an asking price of \$8.24, which is more expensive but worth the delivery time. Figure 18 shows the schematic for the LM2577-ADJ.

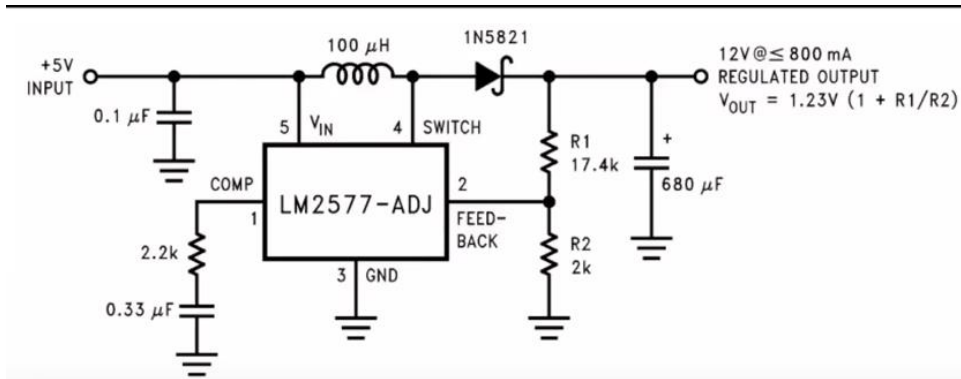


Figure 18: Schematic for the LM2577-ADJ for 5V to 12V

4.5 Light Sensor:

The reason for a light sensor is so that when the HUD is in use it knows when to dim and brighten the display. This will be useful for the transition between day and night. This does not need to be a complex light sensor rather something that's close to the human eye.

4.5.1 Option 1: Adafruit ALS-PT19 Analog Light Sensor Breakout

This is another breakout board from Adafruit. It's very simple with a power need of around 2.5 – 5.5 volts and once its on the only thing to do is read the analog voltage on the OUT pin. As light increases the voltage increases. A bonus is that it's RoHS compliant. Due to the high rejection ratio of infrared radiation, the spectral response of the ambient light sensor is close to that of human eyes. This goes for about \$2.50 and can be shipped out in less than a week. Also, it meets the temperature performance of being able to operate between -40 and 85 degrees Celsius according to the datasheet.

4.5.2 Option 2: OPT3007YMFR

This product is made by Texas Instruments. This is a good fit because it's a super thin light-sensor with a fixed I²C address. This device matches the human eye with rejecting more than 99% of infrared light. It's able to measure between 0.01 to 83k Lux, this will allow for fine tuning when dimming the display. It has a very low operating current at 1.8 μA which is what we're looking for in terms of the limited power that is available. The dimension of this chip is 0.856-mm × 0.946-mm × 0.226-mm which is space efficient but will require a PCB since it's a surface mount device. The OPT3007 will operate between -40 to 85 degrees Celsius. One last upside is that this is a smaller version of the OPT3001 that was used in the embedded systems class, so this isn't a totally new component to figure out.

4.5.3 Option 3: Adafruit 161

The Adafruit 161 is a very simple photodiode. This CdS cell respond to light between 400nm and 600nm wavelengths, peaking at about 520nm. Basically, all they can detect is if there is light or if there isn't light. For this reason, they shouldn't be used to try to determine precise light levels in lux. So, the upside is that they are very cheap, this one is \$0.95, and they are very robust so no worries of it giving out. To read this our main CPU will determine what the voltage off the photodiode is. The higher the voltage the brighter it is.

4.6 Mini Speakers:

The addition of speakers allows for visual cues as well as audio cues to be given. The idea is that if there is a turn to be taken, the user can hear that a turn will come up in so many feet, or if there is an alternative route to be taken that the option is there and can be selected when prompted. The goal in choosing a speaker is that it must use low power. With low power though, it must also be loud enough to be heard over normal traffic conditions.

4.6.1 Option 1: Breadboard-Friendly PCB Mount Mini Speaker - 8 Ohm 0.2W

This little speaker has it all. It's small and robust being only about 30 mm in diameter. The pins can fit perfectly into perfboard too. The speaker is an 8 Ohm and uses 0.2 W or less of power. The optimal temperature range is within -10 to 40 degrees Celsius. The frequency range is between 600 Hz to 11 kHz. There is a class D amplifier that will work with the speaker if that is something that is needed down the line. This speaker is \$1.85 and can be delivered in about a week.

4.6.2 Option 2: Mini Metal Speaker w/ Wires - 8 ohm 0.5W

This tiny 1-inch diameter speaker cone has an 8 Ω impedance and will be using 0.5W or less of power. This particular speaker is very simple, and its metal body is extremely lightweight. The rated frequency range is similar to the speaker above, operating between 600 Hz to 10 kHz. Also, it has a good temperature resistance being able to operate between -20 to 55 degrees Celsius. Again, this speaker can work well with class D amplifier if it is something that is needed. This speaker is \$1.95 and can be delivered in about a week.

5.0 Navigation

One of the main features we would like to implement into the heads-up display is displaying navigational information to the driver. Looking at a phone for navigation

can be dangerous and distracting as it removes the driver's eyes from the road ahead. We can eliminate this risk by displaying pertinent navigational information right in the driver's field of view. For example, if a driver's desired route has a series of turns, a counter with how far the driver is from the turn will appear, along with street names and the direction in which the driver has to turn. This will increase the driver's awareness and reduce the possibility of making a wrong turn or getting lost. To achieve this, we would need to obtain the path the user will take from starting point to destination. There are a variety of ways we could create a path from beginning to end.

5.1 Predetermined Routing

Predetermining the route ahead of traveling means we could gather coordinates for each turn along the way. The waypoints could be determined by mapping out the user's desired route using an application such as Google Earth. The route will most likely contain a series of turns that the driver will have to make in order to reach the destination. At each turn, coordinates will be created to mark the waypoints that the driver must drive through to stay on the route. These waypoints could be entered into the device and stored for processing. Using a GPS chip, we would be able to keep track of where the driver is in relation to those stored waypoints. When the driver begins his or her route, a distance variable will keep track of how far the driver is from the first point. This information will be displayed to the driver on the Heads-Up display in the form of a visible distance counter. As the driver gets closer, the distance will be shown counting down accurately. Once the driver gets to specified distances from the next waypoint i.e. 1 mile, the system will announce a turn is ahead using the built in speaker and sound recorded onto the device. Additionally, the HUD will display an arrow in the direction of the next turn as well as the street name that the driver will be turning on to. This will allow the driver enough time to make note of his or her surroundings and prepare to make the turn. Once the driver reaches the waypoint, the system will move to the next waypoint and begin counting down the distance to it.

This process of predetermining the route the driver will take and inputting the turn coordinates manually has advantages and disadvantages alike. For advantages, it drastically reduces the amount of components needed to implement the device. The driver would predetermine the waypoints of each turn, enter and store them into the device as coordinates, and then have the GPS chip determine the distances between each of the coordinates as the driver is on the route. Being able to reduce the components required to implement the device means that the cost to both manufacturer and consumer can be minimized, as well as complexity to build would decrease. This decreased complexity would allow the device to require less power, making it more efficient electronically. As for disadvantages, predetermining and storing waypoints is an extremely rigid system. It does not allow the driver a quick way to modify the route once it has been programmed into the device and started. If the driver wishes to change the route from what is already

active on the device, the driver would need to manually reprogram each waypoint for the GPS to track. We anticipate that this would be a time consuming process and would require the driver have access to a computer and the interface required to program the device. Due to the inflexibility of this approach, it is unlikely that we would implement our device in this fashion.

5.2 Google Cloud Platform API

As mentioned above, manually entering waypoints into the device is not the most efficient way to implement navigation for our HUD. A much more efficient implementation would leverage the power of pre-existing and well-established navigational platforms to assist in obtaining route information. Google Maps a mapping application for mobile devices that uses the device's internal radios, sensors, and GPS to stream information about the user's current location to Google's infrastructure, determining the user's exact location along a route down to great precision. Maps is extremely useful for determining the most efficient route between the starting point and destination. Engineers from Google have spent years developing algorithms that analyze real-time traffic data, road closures, and other variables to give the driver the safest and most reliable route. The application's user has the ability to add custom filters to the route, such as avoiding tolls or prioritizing distance against time.

Leveraging information from Google Maps would be the ideal scenario for implementing a navigational system into our HUD device. In order to do this, we would need to create a custom mobile application that would implement API calls to the Google Maps platform and obtain the data for a given route. For example, the driver would enter his or her desired destination into our application. The Maps API would return the most efficient route to the driver's destination at the time of computation. This information could be streamed from our mobile device to the HUD device over wireless standards like Bluetooth. The HUD device would then show the route information just as in the previous implementation.

The ability to use Google Maps API will enable us to build a deeper and more advanced implementation for navigation. It will allow us to provide real-time updates to the driver's route. This opens the door for advanced features to be implemented, such as speed limit monitoring. Google Maps has information regarding the speed limit for the road the driver is currently driving on. The speed limit is displayed on the application, notifying the driver and potentially preventing them from speeding. We would like to implement this feature into our HUD by accessing the API for the speed limit of the road the driver is currently on and displaying it in the driver's field of view. We can combine this information with the live speed read by the onboard OBD2 port of the vehicle. If the driver's current

speed goes beyond the speed limit retrieved from Maps, we would like to advise the driver to slow down, ensuring the driver is within safe driving conditions. The use of APIs would make route modification much easier, as it would only require the driver to enter the new location into our application and recalculate the route. This process is much easier than having to remap every coordinate by hand like the previous implementation. Hazards along routes such as traffic jams and accidents are often spontaneous and unplannable. We could implement a function to ping the API every few minutes to check for any hazards along the current route. If the API returns that a hazard lies ahead, we could have the API recalculate the route to avoid the hazard and keep the driver on the most efficient route. Ultimately, implementing a custom mobile application paired with Google Maps APIs and streaming the information to our HUD would provide us with the most flexibility and advanced feature set. This is the implementation we would like to strive for when building the device.

6.0 Application

To support advanced navigational features noted in the previous section, we will need to access the APIs provided from the Google Maps Cloud Platform. There are already applications that exist that can access information from the Google Maps API and the standalone Google Maps application, however we do not have access to the source code of these applications and cannot ensure that the data we need for this project will be available. Additionally, the data will need to be streamed over our Bluetooth module once the destination is chosen by the user.

6.1 Application Design

The best method for ensuring that all of the required data is being received is to build a mobile application from scratch. We will be creating an application for Android phones using the Android Studio, since the development kits are readily available and easily implemented. Cost to implement this application will be minimal as we already have access to Android devices. The application's design language is going to be simplistic and minimal, enabling the user to locate the desired destination and send it to the HUD with ease. The mobile device will communicate with the HUD by pairing the two devices over a Bluetooth connection.

The application design features a search bar at the top for the driver to enter his or her desired destination. As they are entering the destination, the Google Maps API will be suggesting destinations based on the text input from the driver. Once the desired destination appears, the driver will select it from the list. The Google Routes API will then be called to find the most efficient route from the mobile device's current location to the destination.

Once the route is returned, it will populate in an embedded map in the center of the screen. The driver will press the Send to HUD button when ready to proceed on the route. The Routes API will return each turn on the route as a coordinate, which can be used to create waypoints along the route. These waypoints will enable turn by turn navigation to be implemented in our HUD. Once the coordinates are obtained, the application will push them to the HUD from the Bluetooth connection. From this point, the responsibility for navigation is passed to the HUD device, where it will start comparing distances between the HUD's onboard GPS chip and the route's first turn coordinate. The application is not needed for navigation until a new route is desired.

The application will provide the driver with a convenient way to interface with the HUD. Since most people already own a smartphone, there will not be any additional cost to implementing this design.

7.0 Location Tracking:

Our project will require the device to be tracked by GPS coordinates to pinpoint its exact location. This will become useful for implementing turn-by-turn navigation and other advanced features. There are a few methods that we could use to implement GPS tracking in the device.

7.1 Google Cloud Platform Location

The simplest and least complex method of tracking the device's location would be to use the information taken directly from the Google Maps API running in the application on the mobile device. Since the mobile application will already be reaching out to the Google API, it would be minimal to add in the ability for the application to read the device's current location and update this to the HUD. The GPS data could be stored in the HUD device to control the contextual information displayed on the HUD.

7.2 Mobile Device GPS Location

Our second option would be analyzing data provided by the onboard GPS of the paired mobile device. API Libraries provided in the Android Studio application would allow us to build GPS tracking directly into the application. This option as well as the Google API option would limit the need for extra components, lowering the overall cost and power required to implement the HUD. Once the data from the API has been received, it would be sent over to the HUD device via Bluetooth, limiting the amount of computation needed to measure distances between waypoints along the route.

7.3 Standalone GPS Module

The third option for GPS capabilities would be to add a standalone GPS unit into the HUD's design. The GPS module works by detecting multiple satellites in geosynchronous orbit and using their relative positions to triangulate its exact position on Earth. As our HUD device will be mounted around the windshield of the vehicle, the GPS module should be able to sync with the satellites without interference. Implementing a standalone GPS chip would add power draw to the system and increase the cost of the overall design. However, it would allow us to gain experience coding for extra components that we do not have experience with, as well as add complexity to our final PCB. Standalone GPS modules are readily available on the market.

Ublox NEO-6M GPS Module: The NEO-6 module interfaces directly with the Arduino's UART over the TX/RX pins located on the module itself. The module is available for purchase on Amazon from DIYMall with an integrated GPS antenna to synchronize with the signal from the GPS satellites. Since the NEO-6M module comes implemented on its own breakout, we will need to source all of the components required to operate and implement them on our own custom PCB.

8.0 Communication:

The HUD must have a feasible way to communicate with a user's cell phone. This will allow crucial data to be transferred to and from the devices in order to maintain contact. Since the communication is to be with a user's cellphone, the protocol needs to be wireless. This means communication types such as serial and Ethernet are out of the picture. These methods, although very stable and well established in the tech development industry, will not work as running a wire through a car would be a safety hazard as well as unfeasible as most phones only have one port which is used for charging. If the device forced users to lose their one port to the HUD then it would not be a popular item in the slightest.

Fortunately, there are a lot of different types of wireless protocols available for different requirements. Popular protocols include Wi-Fi and Bluetooth, with lesser known ones such as ANT and ZigBee used for low power devices. The distance between master and slave, which is the user's phone and our device respectively, is going to be very small as it will be no larger than within a car. This allows us to choose a low power wireless protocol which fits our design best/

Based off of specifications listed below, this project will be optimal under the use of Bluetooth low energy.

8.1 Wireless Protocol Comparison

Type	Voice	Data	Audio	Video	Low-Power
Bluetooth ACL/HS	X	Y	Y	X	X
Bluetooth SCO/eSCO	Y	X	X	X	X
Bluetooth Low Energy	X	X	X	X	Y
Wi-Fi	Y	Y	Y	Y	X
Wi-Fi Direct	Y	Y	Y	X	X
ZigBee	X	X	X	X	Y
ANT	X	X	X	X	Y

Table 2: Wireless Protocol Comparison

Since the device will only need minimal communication between itself and the phone, characteristics such as audio and video streaming are not needed. The data column represents the ability to file transfer between one device and the other. Because the only real communication needed is packets to be sent over rather than whole multi megabyte files, the data column is not needed either.

ZigBee and ANT are good options with their own perks. ZigBee has a larger range than Bluetooth Low Energy (BLE), almost three times as much. ZigBee also

supports multi node connections and is much more stable with multiple connections coming into it. Since this device only needs one node as a connection and distance is not an issue, BLE has 1/10 the power consumption and most importantly is compatible with Android and Apple operating systems which makes it perfect for device to smartphone connectivity.

ANT is ultra low power, so low that it can run off of a coin cell battery for years. The problem with ANT is that it is too low power with not enough features available for this project. ANT does not support operating systems ran on phones similarly to ZigBee and so the most logical choice is to use BLE.

BLE is the fastest and cheapest 2.4 GHz wireless protocol in the market. In particular, it's the only wireless protocol that you can use with iOS without needing special certification, and it's supported by all modern smart phones.

8.2 Bluetooth Low Energy Specifications and Factsheet

Technical specification	Bluetooth Low Energy technology
Distance/range (theoretical max.)	>100 m (>330 ft)
Over the air data rate	125 kbit/s – 1 Mbit/s – 2 Mbit/s
Application throughput	0.27-1.37 Mbit/s
Active slaves	Not defined; implementation dependent
Security	128-bit AES in CCM mode and application layer user defined (2)
Robustness	Adaptive frequency hopping, Lazy Acknowledgement, 24-bit CRC, 32-bit Message Integrity Check
Connections	> 2 billion

Table 3: BLE Specifications and Fact Sheet

Technical specification	Bluetooth Low Energy technology
Modulation	GFSK @ 2.4 GHz
Latency (from a non-connected state)	6 ms
Minimum total time to send data (det. battery life)	3 ms (3)
Voice capable	No
Network topology	Scatternet
Power consumption	0.01–0.50 W (depending on use case)
Max current consumption	<15 mA
Service discovery	Yes
Profile concept	Yes
Modes	Broadcast, Connection, Event Data Models, Reads, Writes
Primary use cases	Mobile phones, gaming, smart homes, wearables, automotive, PCs, security, proximity, healthcare, sports & fitness, Industrial, etc.

8.2.1 Bluetooth Low Energy Specifications Points of Interest

Distance fits our need perfectly. Considering the distance between HUD and phone will be no more than a couple of meters, 100 meters is quite overkill. This however potentially allows us to add some functionality such as the HUD beginning to turn on once a phone connects to it. Since most people allow Bluetooth devices to automatically connect once it is within range and powered on, the option to add functionality involving greeting the user and automatically turning on within a range is added.

Security is very important, especially with wireless communication. BLE uses AES-CCM which is a mode of operating block ciphers which use a deterministic algorithm. This means that given a specific input, the encrypted output will always be the same as the key does encryption key does not change. This allows encrypting of data to be efficient and predictable to the developer to make sure it is working as intended, while also still keeping data safe as long as the key is hidden. Below is a schematic behind how AES-CCM works within the context of 128bit packets.

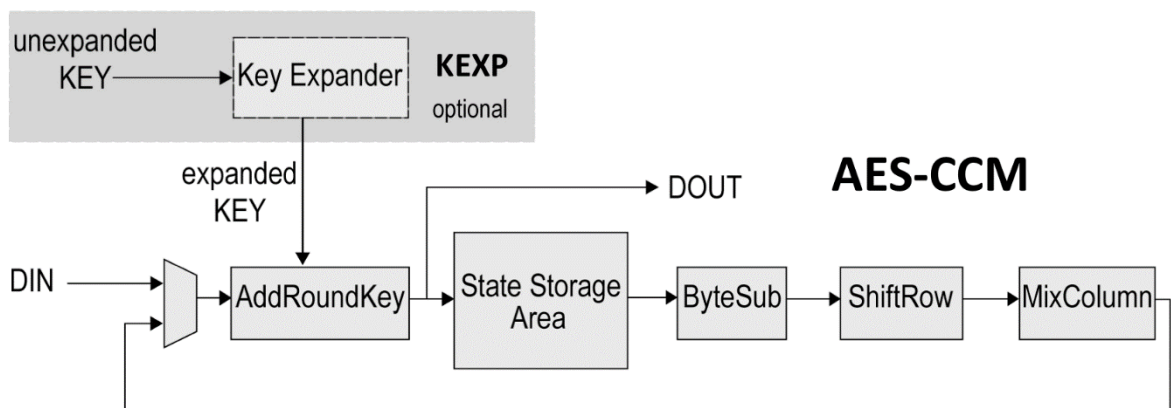


Figure 19: AES-CCM Algorithm

Speed and power of this BLE protocol fits our requirements as well. With a latency of 6ms, information gets sent at the blink of an eye. Also, the power requirements will be low as the nRF52832 does not need much power to begin with. This is listed under 8.5 Bluetooth Module Specification.

8.3 Bluetooth Low Energy Architecture

The physical layer (PHY) refers to the physical radio used for communication and for modulating/demodulating the data. It operates in the ISM band (2.4 GHz spectrum). This clock is standard for most wireless connectivity, even other protocols such as Wi-Fi use the 2.4 GHz spectrum.

The Link Layer is the layer that interfaces with the Physical Layer (Radio) and provides the higher levels an abstraction and a way to interact with the radio (through an intermediary level called the HCI layer which we'll discuss shortly). It is responsible for managing the state of the radio as well as the timing requirements for adhering to the Bluetooth Low Energy specification.

Direct Test Mode: the purpose of this mode is to test the operation of the radio at the physical level (such as transmission power, receiver sensitivity, etc.).

The Host Controller Interface (HCI) layer is a standard protocol defined by the Bluetooth specification that allows the Host layer to communicate with the Controller layer. These layers could exist on separate chips, or they could exist on the same chip.

The Logical Link Control and Adaptation Protocol (L2CAP) layer acts as a protocol multiplexing layer. It takes multiple protocols from the upper layers and places them in standard BLE packets that are passed down to the lower layers beneath it. Figure 20 gives an overview of how each layer is defined in the BLE stack.

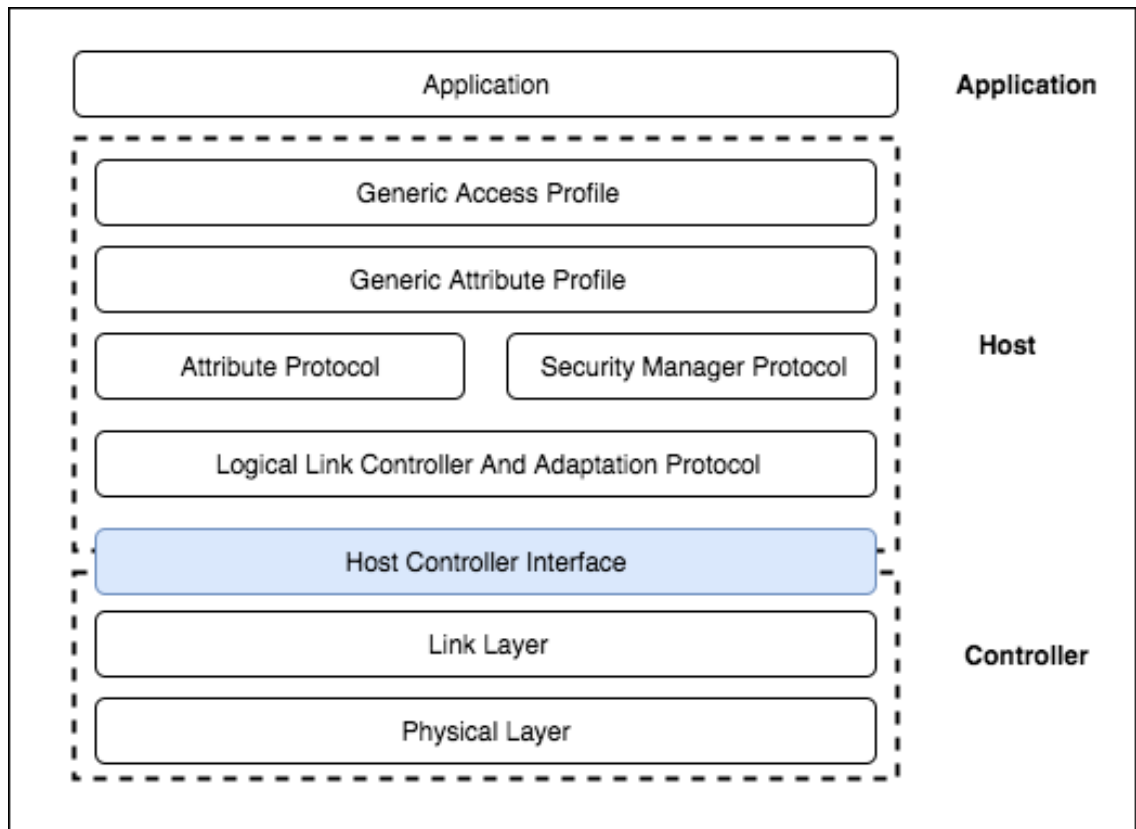


Figure 20: BLE Architecture

8.4 Bluetooth Packet Format

Bluetooth is a networking protocol that follows similar structure to other wireless networking protocols. Data is sent between transmitters and receivers through the use of a packet to package the data. These Bluetooth packets are packaged in a specific format, described in detail below and in Figure 21.

Preamble: It is used by the BLE module for synchronization of time and frequency. The preamble also performs AGC (Automatic Gain Control). It is a predefined pattern of size 1 byte which is known to the receiver. Advertising packet use "10101010" in binary. Data packet use either "10101010" (if LSB of access address is 0) or "01010101" (if LSB of access address is 1) in binary form.

Access Address: For all advertising packet is uses fixed pattern "0x8E89BED6" in hexadecimal form with size of 4 octets. or 32 bits. This address is nicknamed "bed six" and is consistently the access address across every BLE module as part of a

standard. For data packets it consists of a 32-bit random value generated by BLE device in "initiating state". The same value is used in a "connection request (CONNECT_REQ)" message.

BLE Packet

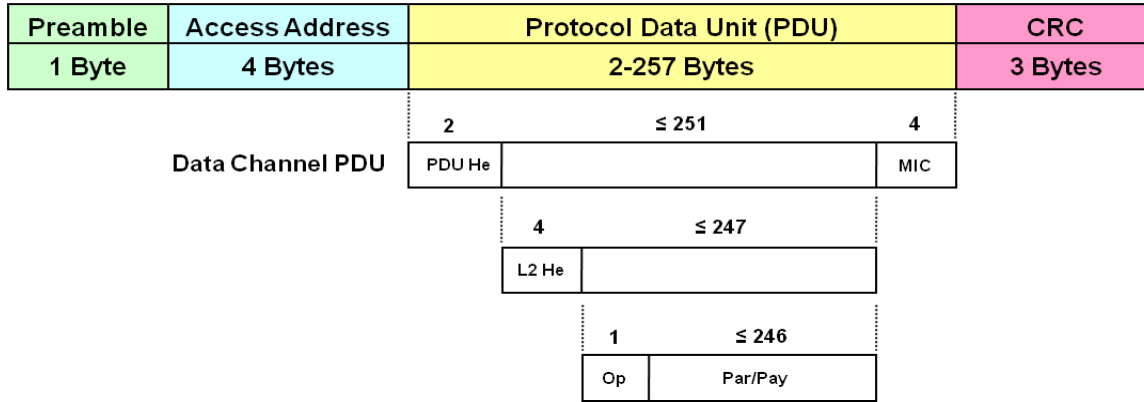


Figure 21: BLE Packet Format

PDU: It consists of either "advertising channel PDU" or "data channel PDU" as defined in the figure. The advertising channel 0000 (ADV_IND) will be the primary mode used for this project.

PDU Type	Packet Name	Description
0000	ADV_IND	Connectable undirected advertising event
0010	ADV_NONCONN_IND	Non-connectable undirected advertising event
0110	ADV_SCAN_IND	Scannable undirected advertising event

Table 4: PDU Packets

CRC: It is 24 bit in size. It is calculated over PDU. It is used for error detection of the packet. CRC is calculated using polynomial of the form $x^{24} + x^{10} + x^9 + x^6 + x^4 + x^3 + x + 1$.

This in total allows a packet to have a size of 265 bytes. The fastest a packet can be sent from a BLE module is 7.5 milliseconds. This allows an extremely fast transfer rate, however for consistency sake and for timing, we will send packets at a slower rate.

8.5.1 Bluetooth Module Specifications

32-bit ARM® Cortex™ M4F CPU

Supply: 1.7V – 5.5V

Flexible and configurable 48 GPIO

Bluetooth 5, IEEE 802.15.4, 2.4 GHz transceiver

- -95dBm sensitivity in 1Mbps Bluetooth low energy (BLE) mode
- -103dBm sensitivity in 125Kbps BLE mode (long range)
- +8 dBm TX power (down to -20 dBm in 4 dB steps)
- On-air compatible with nRF52, nRF51, nRF24L and nRF24AP Series
 - Programmable output power from +8dBm to -20dB
 - RSSI (1dB resolution) • Supported data rates:

Bluetooth 5: 2Mbps, 1Mbps, 500kbps, 125kbps

IEEE 802.15.4-2006: 250 kbps

Proprietary 2.4 GHz: 2 Mbps, 1 Mbps

Advanced on-chip interfaces

- USB 2.0 full speed (12Mbps) controller
- QSPI 32MHz interface
- High speed 32MHz SPI
- Type 2 near field communication (NFC-A) tag with wake-on field • Programmable peripheral interconnect (PPI)
- EasyDMA automated data transfer without CPU processing on peripherals • 12 bit, 200ksps ADC – 8 configurable channels with programmable gain
- 4 x 4 channel pulse width modulator (PWM) units with EasyDMA
- 5 X 32-bit timers with counter mode
- Up to 4 x SPI masters / 3 x SPI slaves with EasyDMA
- Up to 2 x I2C compatible 2-wire masters / slaves
- 2 x UART(CTS/RTS) with EasyDMA
- Quadrature decoder (QDEC)
- 3 x 24-bit real-time counters (RTC)

Operating Temp: -40°C to +85°C

The voltage of this module fits within the requirement of our supply as we are using 5V. The USB serial allows easy debugging and configuration. The speed of the CPU and transceiver are both standard and fast enough to meet the requirements for the HUD.

8.5.1 Bluetooth Module Schematic Configurations

The nRF52840 has different features that can be activated through different circuit layouts. Below are the two that were considered for this project with each having different components on and off. Schematics were taken directly from Nordic's website, who designed the chip.

VDDH vs VDD:

VDDH is the high voltage configuration while VDD is the regular configuration. VDDH is to be used when the voltage provided is more than 3.6V. The highest the voltage can be for VDDH is 5.5V. The cap at 3.6V is due to internal regulators not being able to sustain more than that voltage when in that mode.

EXTSUPPLY:

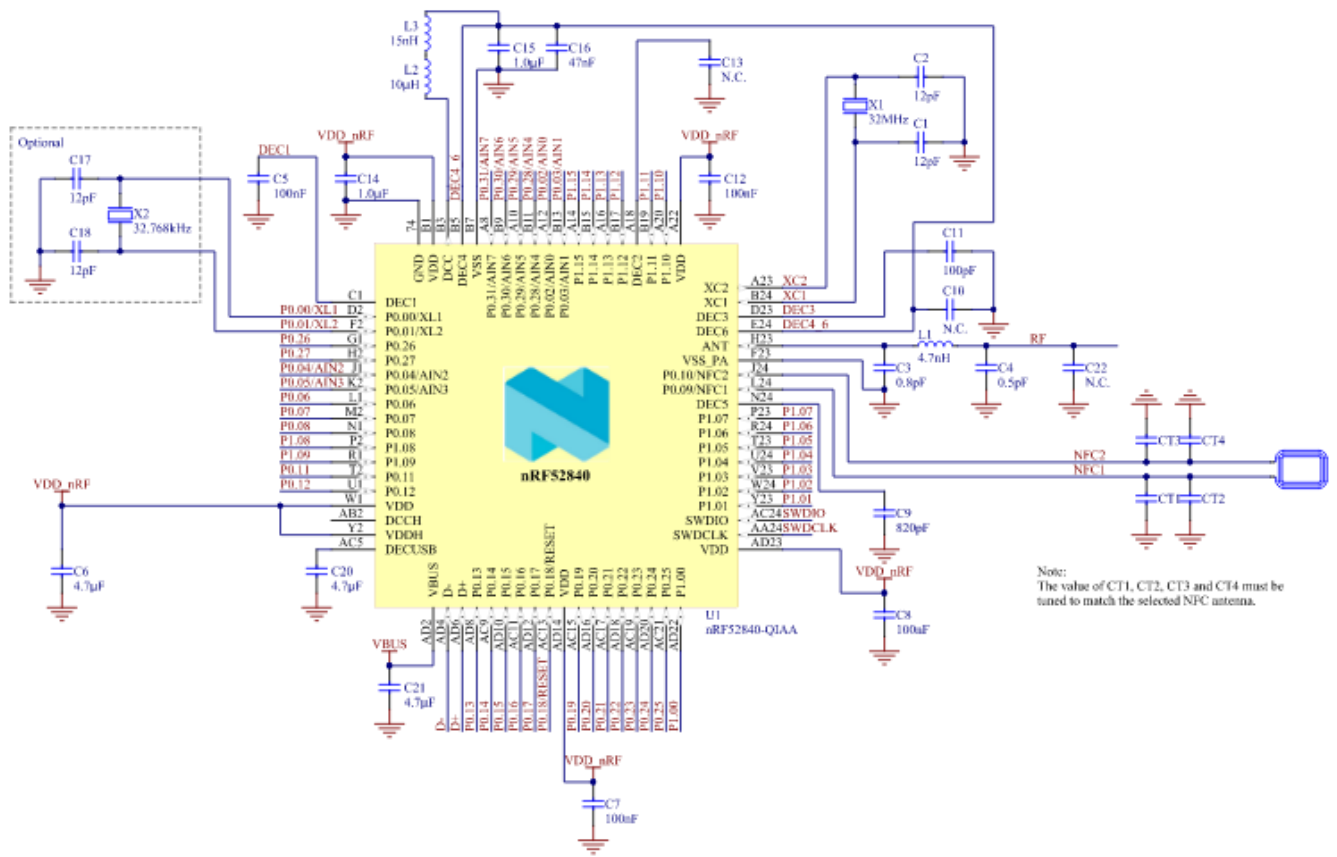
An external supply is only available when running in VDDH mode. This is because external supply only needs to exist if you need certain features that need that 3.6V voltage where 5.5V is too high. For example, if the chip was powered by USB, the chip must be in VDDH mode and the VDD pins can now act as an external supply to external circuitry.

DCDCEN0 and DCDCEN1:

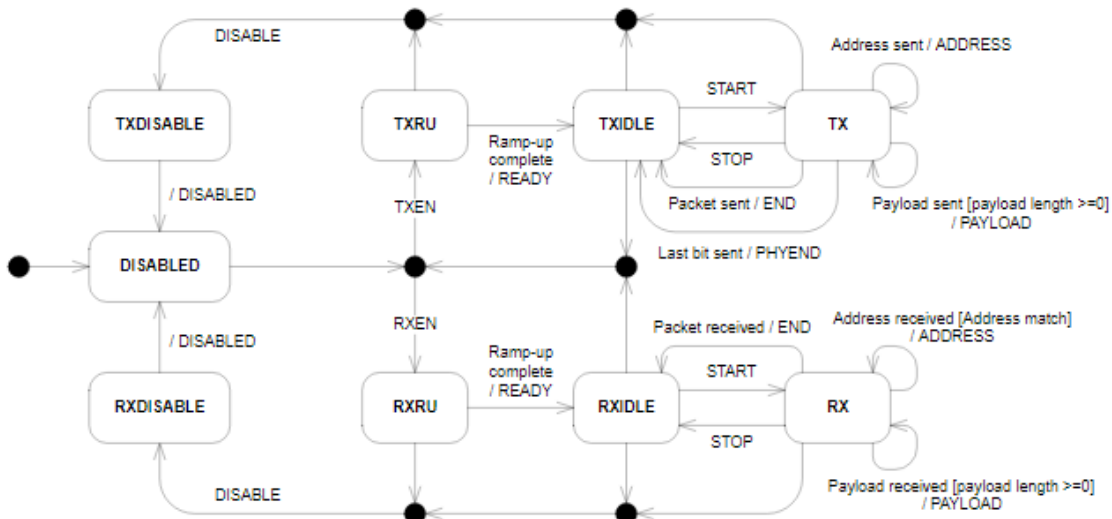
When DCDCEN1 is set, it allows low current DC/DC conversion regulator to be activated for REG1 and DCDCEN0 for REG0. A figure below shows the schematic of the DC/DC and LDO interact.

USB:

Allows the chip to be powered through the Universal Serial Bus without an external power source, note that this forces you to run in VDDH mode if this design choice



Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDCEN0	DCDCEN1	USB	NFC
Config 2	N/A	Battery/Ext. regulator	No	No	Yes	Yes	Yes



9.0 Crash Detection:

When we initially discussed senior design project ideas, making an innovation-packed project was our first priority. Given that the project we chose is focused around driver safety and awareness, adding functionality that will increase these concepts is important to us. The HUD will allow us to reduce distractions and prevent accidents caused by distracted driving. However, accidents can happen at any moment due to factors that we cannot plan for nor have control of. In the event that an accident was to occur, response time is paramount to ensuring those involved are given the care they need. In addition to the reduced distraction provided by our HUD device, we would like to add a crash detection system to our design.

Creating a crash detection feature will allow us to gain experience with some systems we were not initially familiar with at the beginning of Senior Design. The following subsections will discuss each feature required to implement safety features in the event that a crash does occur.

9.1 Accelerometer

Vehicular crashes are often paired with impacts that produce strong and abrupt forces. These forces can be detected using an accelerometer to analyze the gravity relative to the position of the sensor. When a change in force occurs, the sensors on the accelerometer report the differences as data points and the microprocessor can determine if the impact is enough to constitute a crash.

The Adafruit ADXL335 Accelerometer is a triple-axis accelerometer that comes with pinouts for x, y, and z axis measurements. It has an onboard voltage regulator that steps down 5V to the 3.3V required to power the accelerometer. The ADXL335's datasheet states that the chip can withstand forces up to 10,000g's, which is well within the impact forces we would expect from the typical car crash. Once the accelerometer detects an impact, the HUD device will send a command to an SMS module located on the chip. The SMS chip will send a text message containing the device's current location and details about the crash to an emergency contact programmed by the user.

Adafruit sells the ADXL335 on a breakout board with the required components required to implement it. We will be utilizing the schematics provided by Analog Devices in Figure X to implement the ADXL335 onto our custom PCB.

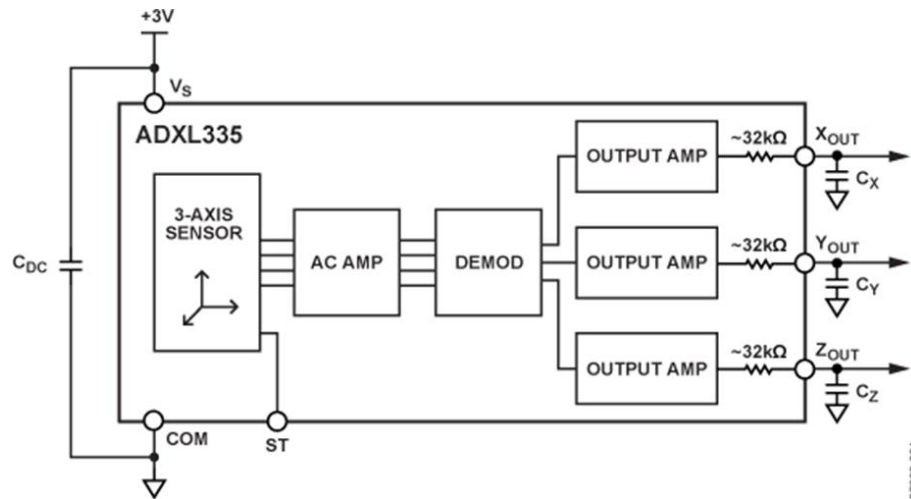


Figure 22: Schematic for ADXL335

9.2 3G/GPS Module

After the accelerometer detects an impact strong enough to constitute a crash, communicating that a crash has occurred to an emergency contact with minimal delay is paramount to providing with the driver the greatest amount of safety

features. Initially, we considered implementing both cellular networking and GPS components to the final design. We thought that these components would be separate and require their own space on the final PCB. Upon researching cellular networking components, we discovered the Adafruit FONA 3G module, which implements a SIMCom 5320A GSM wireless networking chip. The 5320A connects to a wireless cellular network over a GSM connection with a SIM card. The 5320A takes advantage of a 3G network, providing the ability to send data, SMS, MMS, voice, and other data from the cellular connection.

To use the wireless cellular network, the team will need to acquire a SIM card with an active plan. In the United States, there are regulations that control many aspects of cellular network operations. The Adafruit website states that the FONA 3G module must be used in accordance with AT&T's 3G service, however the module may work with 2G networks as well. Adafruit has partnered with Ting, a mobile virtual network operator that leases network space from established telecom companies. Adafruit provides a package that includes a free Ting SIM card with activated 2G service when purchasing the FONA 3G module. This is ideal for our project, as it minimizes the total cost of implementation and prevents the possibility of having a monthly subscription model to implement a cellular connection service.

Finding the exact location of the cellular module can be done via triangulation of the cellular connection. The cellular module will locate nearby cellular towers and find the strongest connection between them. This data can be analyzed and measured against locations of cellular towers to determine the exact location of the module. However, this process is arduous and requires access to locations of network cellular tower locations for which the module is connected to. Fortunately, the SIMCom 5320A implements built-in GPS capabilities onto the chip. This will allow the team to access the location of the device by connecting to geosynchronous satellites that provide location data with great precision. We can use this data to analyze the driver's current location along the desired route as mentioned in Section 7.0.

Adafruit provides documentation that includes schematics for how the FONA 3G module is implemented. This will allow us to purchase the SIMCom 5320A and add it to our custom PCB. Since it comes with a free Ting 2G enabled SIM card, we can minimize the cost associated with using a cellular network connection. The ability to combine cellular and GPS onto one module reduces the components required to implement the functionality of communicating with an emergency contact in the event of a crash.

9.3 Affects of Impact on Power Delivery and Electronics

Large impacts during a car crash are often strong enough to reduce the vehicle's ability to continue under its own power. Once the engine stops running, the vehicle is no longer able to use the alternator to produce electricity. For our device to work, we rely on the vehicle's alternator to provide enough power for the device and accompanying components.

Since it is critical to have the driver's location via SMS text at the time of impact, we need to have the device stay active long enough for that signal to be sent. To ensure reliability of our electronics during and after the event of a crash, we may have to look at adding a secondary power source into our design. We would not need to have the device powered for a long time, only a short enough duration to complete the SMS message containing the driver's current position to his or her emergency contact. Small lithium polymer batteries are cheap and would provide enough power to allow the microcontroller and accompanying electronics for a few seconds after the main car power source becomes unavailable.

We are currently looking into the best way to implement the secondary power source into our design. Our initial thought is to have a small battery attached to the power delivery system on an inactive circuit. If the accelerometer detects an impact, there should be enough time to switch the power from the main source of the vehicle to the secondary source.

As previously mentioned in this section, crashes are typically accompanied by strong forces that can sustain serious damage to the vehicle. Our HUD device must be designed in such a way that it will survive the impact and operate normally. When designing and building our HUD device, we need to ensure that all components are properly secured to the PCB and that all wiring and solder points will not break free.

10.0 Microcontroller:

The HUD device will need to receive data from the GPS module multiple times per second and calculate the distance between the received data and the waypoints along the route. This requires processing power and storage to manipulate and store the calculations. Once these calculations are complete, the display will need to be updated to reflect the most current information to the driver.

As we began to research microcontrollers needed to handle all the information from the GPS module, APIs, SMS module, and drive the display unit, it became apparent that a powerful microcontroller would be needed to provide the driver with the best user experience. The microcontroller has to have UART communication to interface with all of our modules. There is a common thread in projects that implement similar designs, a majority of them use Arduino or TI MSP based microprocessors. Upon researching Arduino and TI microprocessors, there are two that stood out. A comparison of these microcontrollers is listed in Table 7 below.

Microcontroller	MSP430F447	Arduino ATmega 2560
Price (\$)	8.50	38.50
Processor Speed	16 MHz	16 MHz
Data Bus Bandwidth	16-bit	8-bit
RAM	1 KB	8 KB
Flash Memory	32KB + 256B	256 KB
UART Channels	3	4
I/O Pins	48	54
Operating Temperature	-45 C to 85 C	-40 C to 85 C
Operating Voltage	1.8 V to 3.6 V	5 V
Special Features	Integrated LCD Controller	Open Source

Table 5: [MSP430](#) and Arduino Comparison

10.1 Texas Instruments MSP430F447

The Texas Instruments MSP430F447 is unique as it has its own integrated LCD controller. This would prove useful when driving the Heads-Up Display unit. A segment of the microprocessor would solely be devoted to this display, eliminating the need for the whole processor to calculate the updates to the display a few times per second. This would free up used memory, power, and allow the microprocessor to do calculations on other components and tasks.

10.2 ATmega 2560

The ATmega 2560 seems to be the most likely candidate for what we would implement into our project. The hardware for the Atmega 2560 is open source, allowing us to create our own implementation of the microprocessor without infringing on copyrighted or patented designs. Additionally, this microprocessor is well known for their flexibility and portability for projects containing embedded design similar to our project narrative, typically utilizing something like an Arduino based system. Arduino provides a development kit that includes the ATmega 2560 fully implemented. The development kit includes custom PCB files and other documentation for creating custom components using the technology they provide. To abide by the regulations set by the College of Engineering and Computer Science, the development board for the Arduino Mega 2560 would only be used for design and testing. Our custom PCB will be designed to contain the Mega 2560 and all its components required to operate correctly and efficiently. Figures **24 and 25 below** are the schematics needed to implement the ATmega 2560.

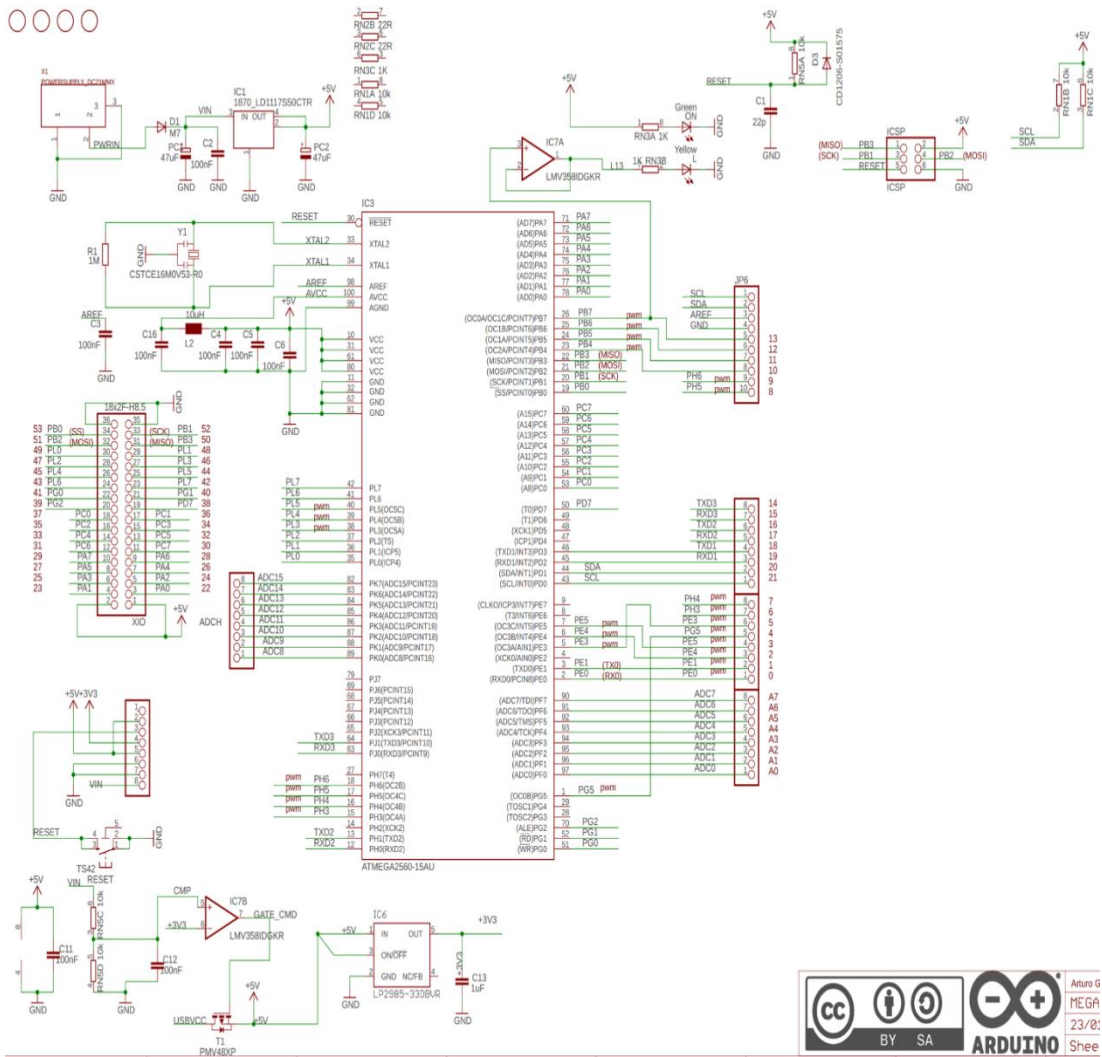


Figure 23: ATmega 2560 Schematic

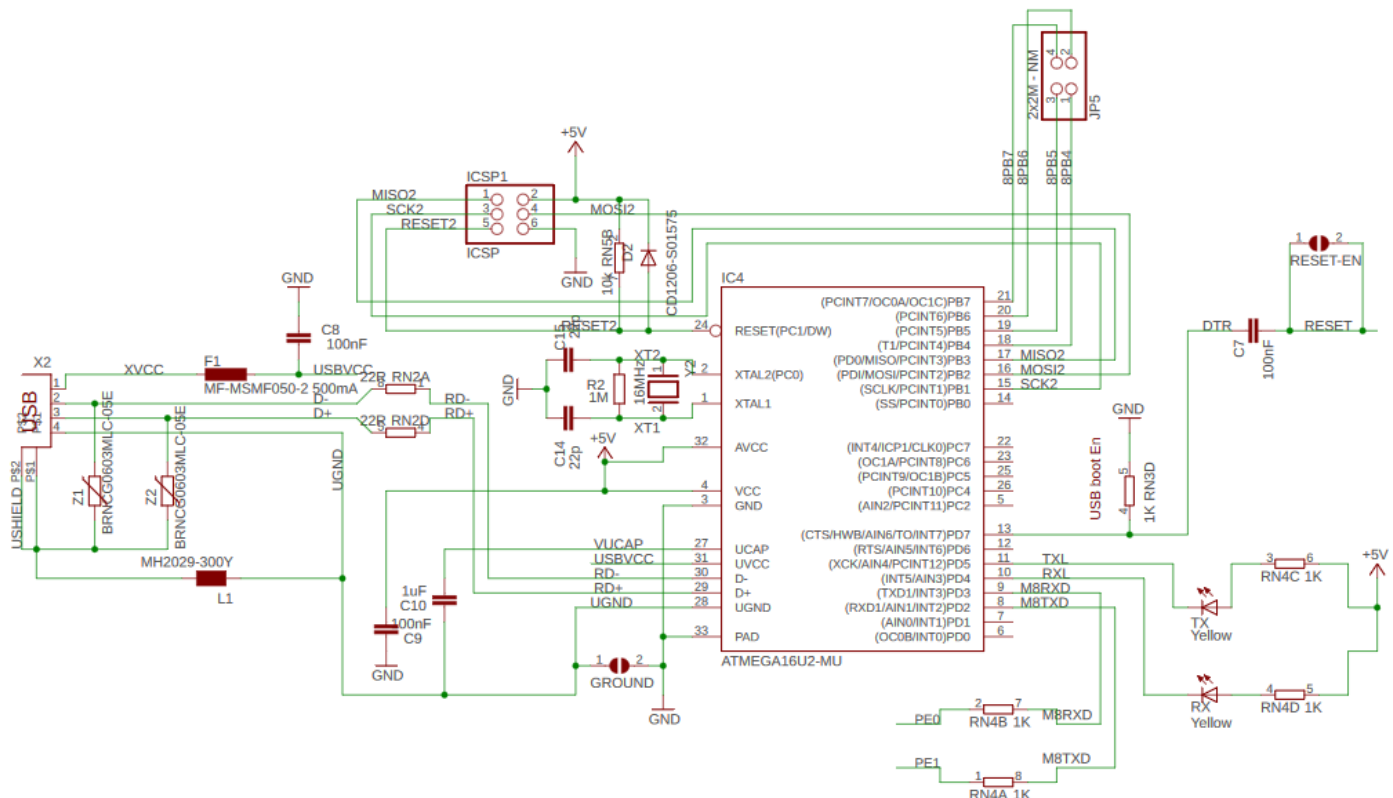


Figure 24: ATmega 2560 Schematic Continued

11.0 Budget:

To implement this project, we will need to research and obtain materials that will meet our specifications in order to produce an effective heads-up display device. We will need a PCB, a power delivery system, microprocessors, LEDs, soldering equipment, a Bluetooth module, and a speaker. At this point, we have not obtained funding or sponsorships for anything involved with this project. As such, the group will be self-funded and will split the cost required to complete the device. To implement navigational aids, we may have to acquire a license to access the Google Maps APIs. This is something that will be researched to determine if this cost is realistic.

We have determined that a rough estimate of \$300 will be enough to completely implement the project.

Item	Description	Price(\$)
PCB	Implement all hardware needed onto PCB	40
Display Unit	Display information to driver on windshield	~102
Power Delivery System	Provides power to system	~45
Microprocessor	Processes data and sends to display unit	38.50
LEDs	Show power states and pertinent information	~5
Soldering Equipment	Needed to implement electrical components	Free if borrowed from school
Bluetooth Module	Receive data from phone over Bluetooth connection	11
Bluetooth Testing Development Board	Allows debugging and initial exposure to the nRF52840 through Arduino and Nordic SDK	24
J-Link OB ARM Programmer	Flashes firmware and sends code to blank Bluetooth module.	5
Oscilloscope	Allows testing of RF portions as well as Voltage differences	Free if borrowed from school
Speaker	Play recorded sounds in specific situations	~9.90
Mobile Smartphone	Needed to host custom Application	N/A
GPS/3G Module	Feed location data to system	90

Table 6: Budget

12.0 Testing Plan

A successful project needs to go through various testing phases to ensure each component is working properly and the intended outcome is achieved. As mentioned in the above sections, there are numerous components that contribute to the operation of the HUD device. Each of these components needs to be thoroughly inspected, tested, and verified to guarantee that the HUD device can be completed on time and avoid last minute changes. The following sections will describe the test plan for each one of the components we intend to use.

12.1 Microcontroller Testing

The ATmega 2560 Microcontroller is the heart of the operation of the HUD Device. Without proper functionality of the microcontroller, we would not be able to operate the display that provides the driver with information required to reduce distractions while driving. We would not be able to transfer data between the mobile application and the HUD device over Bluetooth protocols. Data from the accelerometer would not be processed if a crash were to occur, reducing the ability for our project to help those impacted by the crash.

For these reasons, we need to be certain without doubt that our microcontroller is working properly. One of the benefits of choosing a microcontroller as popular as the ATmega 2560 is that there is a myriad of online resources that discuss how to properly develop and maintain the platform. Documentation about implementing the microcontroller is readily available in detail from the manufacturer. This information will become paramount when implementing and troubleshooting the microcontroller and its functionality.

To test the implementation of the ATmega 2560, the team will initially purchase a Development Kit that includes a fully integrated PCB. Development kits provide resources such as easy-to-use IDE's and verified components to ensure proper functionality. This will allow use to verify the use of the other components using methods discussed later in this section. Once we have verified that the components, we have selected will work with the ATmega 2560, we will then look to implement the microcontroller into our own custom PCB.

Upon completion of the verification phase, we will enter the testing phase for our own microcontroller implementation. The ATmega 2560 is an open-source hardware platform, which means we would have access to the documents the

manufacturer uses to create the final product. Adding the ATmega 2560 to our custom PCB will require us to add in all the required features from the development board ourselves. This includes power delivery, firmware flashing, pushing code to the microcontroller, and other important tasks to operate the microcontroller. We will test the results of our custom PCB design and compare them against the development board to verify that our design is optimal for the project.

A large portion of our parts will be ordered from online suppliers that source their parts from various countries around the world. Sometimes faulty parts make it through the supplier's production line and get delivered to the consumer. Before using any components on our final design, we will test each resistor, capacitor, pinout, and module with tools such as a multimeter to verify that the microcontroller is meeting the optimum conditions to operate properly.

12.2 Mobile Application Testing

The mobile application we are going to produce for the project plays an integral part in creating a functional navigational aid. We must ensure that the driver has a consistent and beneficial experience while using the application. There are a few aspects of the application that will need to be tested and verified.

Firstly, we will need to test the application's design itself. As mentioned in Section 6.1, we will be designing and building the mobile application using the Android Studio Development Suite. This software includes an IDE and other tools, making the software design and testing process of custom applications much more efficient. Throughout the development of the application, we will be using Android Studio to push the application to our mobile device, allowing us to test the application with the hardware we intend to use for the final product. The group will test a few different designs to choose the best one for this project.

Next, we will need to test the application's functionality. We will test this by entering multiple destinations into the application to verify it is being received as expected. Then, we will send the data to the Google Cloud Platform APIs and verify the results to be sure it is what we would expect. Since we are relying on a third-party service to provide the APIs required to receive the data, we will need to check that Google's services are online and functioning when running into any issues. We will then ensure that the data is transmitted over the mobile device's Bluetooth radio and received by the HUD device as expected.

Android is a fragmented operating system that contains multiple active versions across many different manufacturers. Initially, we will be developing and testing for the Samsung Galaxy line of mobile devices, narrowing the scope of testing we will need to do. At the time of application creation and testing, we will tailor the application to take advantage of the current API packages provided by the Android platform. As the application ages and progresses, we may investigate making the application device-agnostic, allowing greater compatibility and larger reach, but requiring a greater amount of testing to ensure the application will work no matter what device it is running on.

12.3 Accelerometer Testing

The accelerometer's accuracy is extremely important considering we are using it as a safety mechanism. Any erroneous measurements or failures would prevent the project from enabling greater safety for the driver. Once the accelerometer components are received, we will test its accuracy to ensure it is measuring the expected outputs. According to omnicalculator.com, a person that weighs 165 pounds travelling at 45 miles per hour will experience roughly 10g's of force when given a stopping time of 0.2 seconds. We will need to analyze crash impact data to determine the accuracy of this measurement. Once we have an accurate measurement for G-force required to constitute a crash, we will test the accelerometer to simulate a crash and measure the results over the microcontroller.

12.4 3G/GPS Module Testing

Once we have acquired the Adafruit FONA 3G Module, we will begin testing the ability to connect to the 2G network using the included 2G Ting SIM card. Using the ATmega 2560, we will connect the Fona 3G module to the microcontroller over the I/O pins. We will test the ability to trigger a text message over the microcontroller, as well as analyzing the incoming GPS data from the module and verify that is accurate with the location of the module within reasonable accuracy.

12.5 LCD Screen Testing

When the LCD screen has arrived and we have determined the proper connectors for it, we will be able to evaluate the performance of the screen. Metrics we will be looking at will be light transmission, contrast, resolution, and refresh speeds. The LCD is listed as having a resolution of 128x64, and we will be able to evaluate how easily such a resolution can be seen by the user, as well as how much information can be packed into the screen. Light transmission and contrast can be evaluated with a bright flashlight. Our main concern with regards to contrast is whether or not the projected image will be too “washed-out” to be visible. If the contrast is unsatisfactorily low, then the finer details of the projected image will be difficult or impossible to be seen by the driver.

12.6 LED Testing

Once proper LED's have been selected, we will be able to power them and evaluate how well they will work for our application. The main metrics we will be watching are brightness, power consumption, and heat output. The main challenge will be finding a proper setup for testing the LED's. If possible, we can also evaluate the LED's with the LCD display and how well the LED's are able to illuminate the information displayed on the screen. The challenge will be finding LED's that fit in our power budget while being able to produce an image that is visible in sunlight. By their nature, LCD's tend to block at least half of all light that enters them due to polarizers. These polarizers are necessary for function and must be compensated for when selecting a suitable LED light source.

In order to test the LED's, they must first be hooked up to a power meter to gauge what their maximum operating power consumption is. We must then determine whether or not our power budget will allow us to use that LED. Once that is completed, we will be able to evaluate their effectiveness at producing a visible projection.

12.7 BOB-12035 Testing

This test is to prove that the main power supply of our project is working properly and to our standards. This test will be conducted with a wall charger that has the same specifications of the car charger we will be using, which is a 5-volt, 2.1-amp

fast charger. In order to make sure our connector works, the wall mount will be inserted with a USB type B cable connected to the connector. A multimeter will be used to test the output at the pins. The only way this part passes inspection is if it outputs 5 volts and 2.1 amps anymore or any less and the test is a fail and a new part will be ordered on the assumption it was damaged on arrival.

12.8 LM2596 (Step down voltage regulator) Testing

The point of this test it to configure the voltage regulator needed to step down the 5-volt input for the smaller parts of the system. This works with a combination of diodes, capacitors, and inductors, with the main part being the LM2596. There will be strict tolerances for each individual part. The goal is to output 3.3 volts at around 2 amps. In order to do this, the datasheet gives specific parts that should be used following the schematic from figure __ and the corresponding tables below for part selection.

Table 3. LM2596 Fixed Voltage Quick Design Component Selection Table

CONDITIONS			INDUCTOR		OUTPUT CAPACITOR			
OUTPUT VOLTAGE (V)	LOAD CURRENT (A)	MAX INPUT VOLTAGE (V)	INDUCTANCE (μH)	INDUCTOR (#)	THROUGH-HOLE ELECTROLYTIC		SURFACE-MOUNT TANTALUM	
					PANASONIC HFQ SERIES (μF/V)	NICHICON PL SERIES (μF/V)	AVX TPS SERIES (μF/V)	SPRAGUE 595D SERIES (μF/V)
3.3	3	5	22	L41	470/25	560/16	330/6.3	390/6.3
		7	22	L41	560/35	560/35	330/6.3	390/6.3
		10	22	L41	680/35	680/35	330/6.3	390/6.3
		40	33	L40	560/35	470/35	330/6.3	390/6.3
	6	22	L33	470/25	470/35	330/6.3	390/6.3	
	2	10	33	L32	330/35	330/35	330/6.3	390/6.3
5	3	40	47	L39	330/35	270/50	220/10	330/10
		8	22	L41	470/25	560/16	220/10	330/10
		10	22	L41	560/25	560/25	220/10	330/10
		15	33	L40	330/35	330/35	220/10	330/10
		40	47	L39	330/35	270/35	220/10	330/10
		9	22	L33	470/25	560/16	220/10	330/10
	2	20	68	L38	180/35	180/35	100/10	270/10
	40	68	L38	180/35	180/35	100/10	270/10	

Table 9: Choosing an Inductor and Capacitor for LM2596

VR	3-A DIODES				4-A TO 6-A DIODES						
	SURFACE-MOUNT		THROUGH-HOLE		SURFACE-MOUNT		THROUGH-HOLE				
	SCHOTTKY	ULTRA FAST RECOVERY	SCHOTTKY	ULTRA FAST RECOVERY	SCHOTTKY	ULTRA FAST RECOVERY	SCHOTTKY	ULTRA FAST RECOVERY			
20 V	SK32	All of these diodes are rated to at least 50V.	1N5820	All of these diodes are rated to at least 50V.		All of these diodes are rated to at least 50V.	SR502	All of these diodes are rated to at least 50V.			
			SR302				1N5823				
	MBR320				SB520						
30 V	30WQ03		1N5821							SR503	
	SK33		MBR330				50WQ03			1N5824	
			31DQ03							SB530	
			1N5822							SR504	
40 V	SK34		SR304				50WQ04			1N5825	
	MBRS340		MBR340								
	30WQ04		MURS320		31DQ04		MUR320			MURS620	SB540
50 V	SK35	30WF10	SR305			50WF10		HER601			
or	MBRS380		MBR350		50WQ05		SB550				
More	30WQ05		31DQ05				50SQ080				

Table 10: Choosing a Diode for LM2596

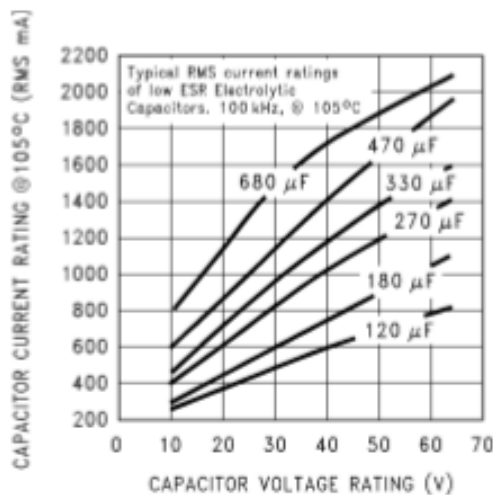


Figure 25: Choosing an Input Capacitor for LM2596

After appropriate part selection is made then each part must pass its individual tolerance test to make sure the part will perform as desired. In order to achieve the 5-volt, 2.1-amp configuration the following parts will be used; a 22 uH inductor, with an output capacitor of 470 uF / 25 V. The diode must be selected so that the current rating is 1.3 times higher than the output current which is 1.3 times 2.1 which is 2.73. Knowing that, the 1N5820 schottky diode will be used because of its maximum repetitive reverse voltage of 40 volts and average rectified forward

current of 3 amps. Finally, the input capacitor must be selected such that its voltage rating is 1.25 times greater than the input voltage with a ripple current rating of about half the load current; this means a 680 uF capacitor with 100 voltage rating and ripple current rated at 1280 mA will be used. Because the output is across the capacitor the use of an oscilloscope will be able to show the output voltage. Then the current will be shown using a multimeter. The table below will be used to determine if the parts are within tolerance and if the circuit passes the testing phase.

Part	Required Value	Actual Value	Pass/Fail
Input Capacitor	680 uF		
Output Capacitor	470 uF		
Inductor	22 mH		
Circuit Testing			
Output Voltage	3.3 Volts		
Output Current	2.1 Amps		

Table 11: Part Testing for LM2596

12.9 LM317T (Step down voltage regulator) Testing

The LM317T is another step-down voltage regulator that will also have to be tested to see if it is a part to be used or not. The LM317T is a linear regulator and is simple in design. Below is a schematic of how it would look.

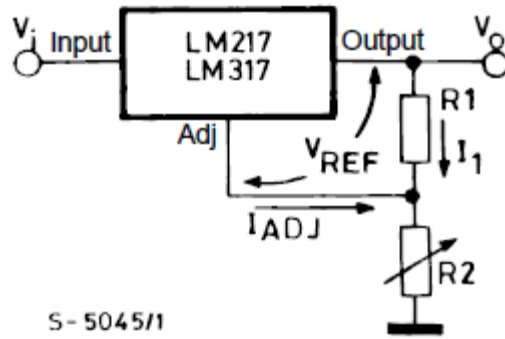


Figure 26: Schematic for LM317T

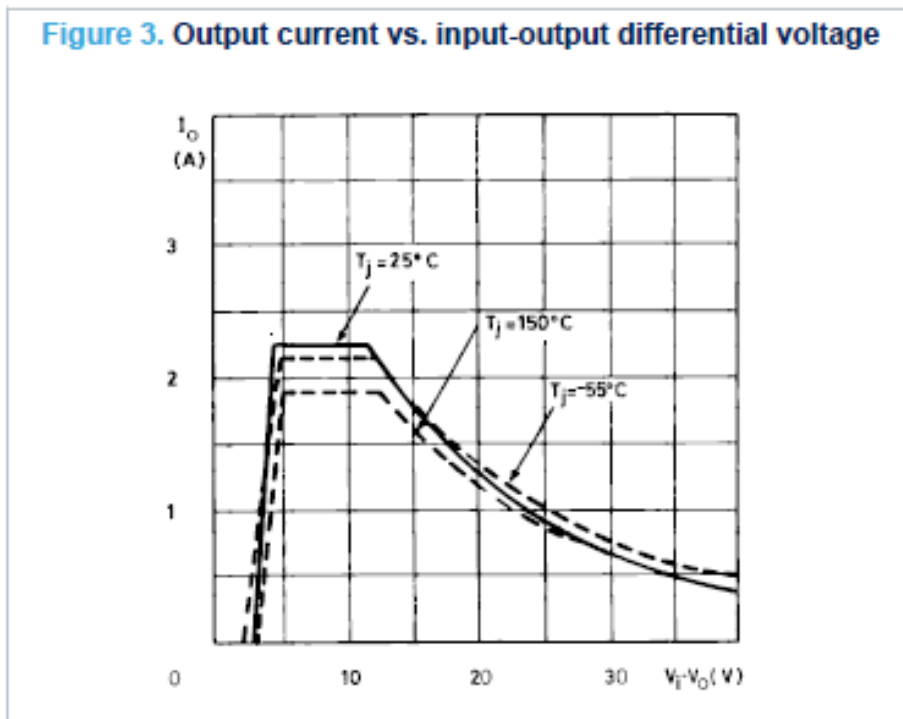


Figure 27: Graph for Output Current for LM317T

The equation for output voltage is $V_o = 1.25(1+(R_2/R_1)) + I_{ADJ}R_2$. Since at the most I_{ADJ} is $100 \mu\text{A}$ the $I_{ADJ}R_2$ term can be ignored. The desired output is 3.3 volts. Solving for R_1 and R_2 gives values of $1 \text{ k}\Omega$ and $1.640 \text{ k}\Omega$ respectively. This would have an output current of about 2 amps. In order to test this, the resistors must be checked to confirm they are the correct values. Then the output will be tested to ensure the desired results. The table below shows whether if the parts pass or fail.

Part Testing			
Parts	Required Value	Actual Value	Pass/Fail
R₁	1000 Ω		
R₂	1640		
Circuit Testing			
Voltage Output	3.3 Volts		

Table 12: Results for LM317T

12.10 TPS61222DCKR (Step down voltage regulator) Testing

The TPS61222DCKR is the last step-down voltage regulator to be tested. It has a simple schematic layout that was provided in the datasheet shown in Figure XX. The input is 5 volts and the output should be about 3.3 volts. The output is determined by a voltage divider and has the equation below.

$$R_1 = R_2 \times \left(\frac{V_{OUT}}{V_{FB}} - 1 \right)$$

V_{FB} is about 500 mV so that means that the resistor values of R_1 and R_2 must be 5.6 kΩ and 1 kΩ respectively. The test will be conducted to ensure that each part is the correct value that is needed and then to test whether the desired operation has occurred or not. Because the output is across the capacitor the use of an oscilloscope will be able to show the output voltage. Then the current will be shown using a multimeter.

Part Testing			
Part	Desired Value	Actual Value	Pass/Fail
R₁	5.6 kΩ		
R₂	1 kΩ		
Inductor	4.7 μH		
Capacitor	10 μF		
Circuit Testing			
Output Voltage	3.3 Volts		
Output Current	1.5 Amps		

Table 13: Results for TPS61222DCKR

12.11 LM2577 (Boost converter) Testing

This part will assist in providing higher voltage to the LCD screen and the backlight. It needs to take 5 volts and boost it to 12 volts. Looking at the datasheet for the LM2577 a typical application of 5 volts to 12 volts is already provided as shown in figure XX from earlier. Listed in the figure XX is one 0.1 μF, one 0.3 μF, and one 680 μF capacitor. There is one 100 μH inductor. The three resistors have values of 17.4 kΩ, 2 kΩ, and 2.2 kΩ. All of these parts must be tested individually for accuracy purposes. Then the circuit, as a whole, will be tested for the desired voltage and current output. The table below will be used to document the testing parameters.

Parts Testing			
Part	Desired Value	Actual Value	Pass/Fail
R₁	17.4 Ωk		
R₂	2 Ωk		
R₃	2.2 Ωk		
C₁	0.1 μF		
C₂	0.33 μF		
C₃	680 μF		
L	100 μH		
Circuit Testing			
Voltage Output	12 Volts		
Current Output	800 mA		

Table 14: Results for LM2577

12.12 LT1613 (Boost converter) Testing

The purpose of this boost converter is to take 5 volts to 12 volts. It needs to provide a higher voltage to the LCD screen and the backlight. Looking at the datasheet it already comes with a schematic for 5 volt to 12 volts boost converter. The figure XX from earlier shows how. In the schematic all resistors, capacitors, and inductors will need to be tested to ensure that they perform at the rated value. In addition, the circuit will need to be tested to ensure that the output voltage and current are at the desired values. The table below will record the data for testing.

Part Testing			
Part	Desired Value	Actual Value	Pass/Fail
R₁	107 kΩ		
R₂	12.3 kΩ		
R₃	10 kΩ		
C₁	22 μF		
C₂	4.7 μF		
C_{PL}	200 pF		
Circuit Testing			
Output Voltage	12 Volts		
Output Current	120 mA		

Table 15: Results for LT1613

12.13 Adafruit ALS-PT19 Analog Light Sensor Breakout Testing

The light sensor's goal is to determine if when the backlight dulls or intensifies. The way it works is by connecting the light sensor to a voltage source of around 2.5 to 5.5 volts. Then on the analog output you can either measure the voltage or the current that is measured. The increase or decrease in either voltage or current shows if it is getting duller or brighter. The table below shows how the values are to be interpreted off the analog output. The figure below that shows a typical setup for getting a readout.

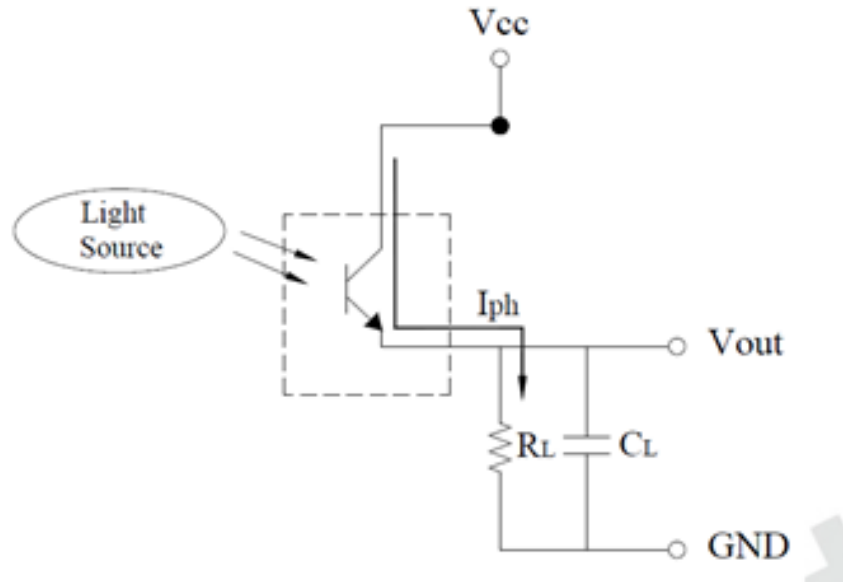


Figure 28: Schematic for ALS-PT19 Analog Light Sensor

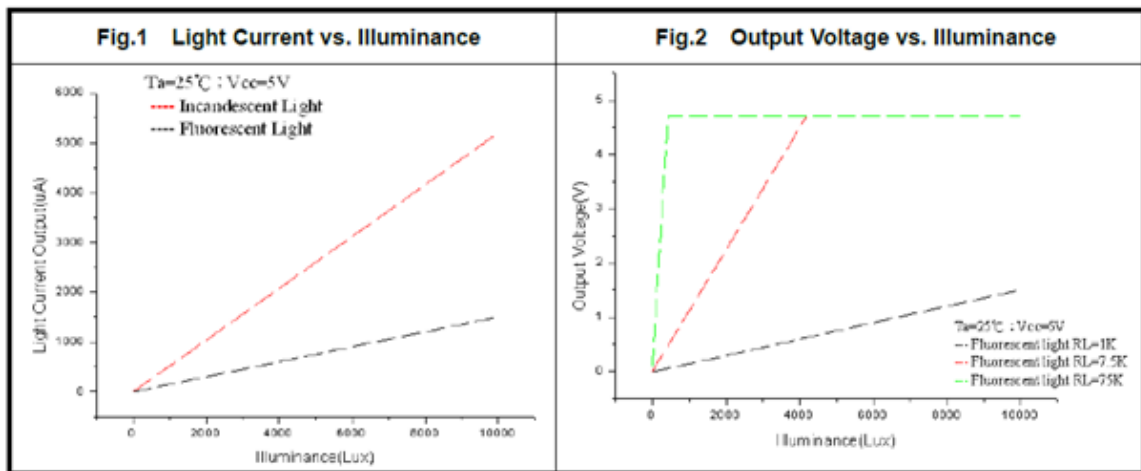


Figure 29: Current and Voltage of Analog Light Sensor based on Illuminance

12.14 Adafruit 161 Photodiode Testing

The photodiode is a simple component that shows the level of lux by changing the resistance like a variable resistor. The higher the resistance the lower the level of light and the lower the resistance the brighter the level of light. It creates a resistance from 200 kΩ to 10 kΩ. The voltage input is up to 100 volts and uses less than 1 mA of current on average. To test if the photodiode works all that needs to be done is connect each end to a multimeter set to resistance. Then to test it in a circuit a simply connect one side of the photodiode to a voltage source connected to a pull-down resistor with a value of 1 kΩ to 10 kΩ. If you need to get better readings to differentiate between bright and really bright use the 1 kΩ. Using the 10 kΩ resistor give different light level ranges. In figure 31 it can be seen how the photoresistor changes its resistivity with the amount of light that shines on it. The following equation shows the relation between the combined resistance of the resistor and the photocell regarding voltage output. $V_o = V_{cc}(R / (R + \text{Photocell}))$

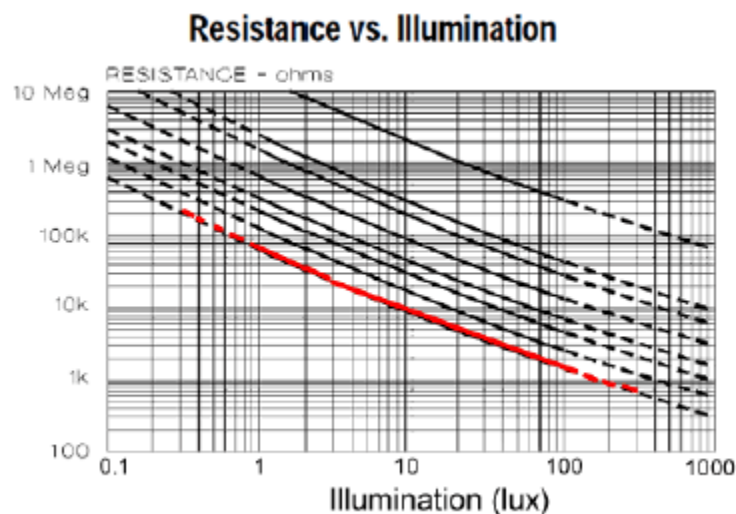


Figure 31: Light vs Illumination

The two tables below will show how the photodiode responds using a 10 kΩ resistor and a 1 kΩ resistor. There is a disclaimer saying that not all the photodiodes are made the same having around 50% in discrepancy even within the same batch. The same type of testing will be performed with a halogen light like in the ALS-PT19 Light Sensor.

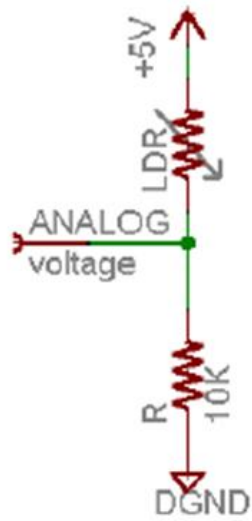


Figure 32: Schematic for Testing Photoresistor

10 kΩ Pull-Down Resistor			
Level of Light	Inches away from light sensor	Voltage Output (V)	Current Output (mA)
Dark (Completely covered)			
High Intensity	1		
Medium to High Intensity	2 to 5		
Medium Intensity	6		
Low to Medium Intensity	7 to 10		
Low Intensity	10+		

Table 16: Photodiode with 10 kΩ pull-down resistor

1 kΩ Pull-Down Resistor			
Level of Light	Inches away from light sensor	Voltage Output (V)	Current Output (mA)
Dark (Completely covered)			
High Intensity	1		
Medium to High Intensity	2 to 5		
Medium Intensity	6		
Low to Medium Intensity	7 to 10		
Low Intensity	10+		

Table 17: Photodiode with 1 kΩ pull-down resistor

From this data it will be determined if 1 kΩ or a 10 kΩ resistor will be used and if these are not adequate results maybe another resistor will be used.

12.15 Breadboard-Friendly PCB Mount Mini Speaker - 8 Ohm 0.2W

Testing speakers is a simple enough task. This can be accomplished without an amplifier or source of sound. To test this the only tools that will be needed is a dc power source and a multimeter. Since this is an 8 ohm speaker if the multimeter is in resistance setting, placing the probes on the speaker input and output should produce about 8 ohms. Then to check that the phase is good simply take a dc source around 1 volt since these are tiny speakers and apply a voltage to the input and output terminals. Having positive to positive and negative to negative will make the cone stick out and then reversing the polarity will make the cone stink inwards. Then a continuity check would be the last thing to test to make sure the speaker is intact. The table below will show the results of the tests.

Test Performed	Desired Outcome	Actual Outcome
Ohm Test	8 Ω	
1 Volt Positive Polarity	Cone pops outwards	
1 Volt Negative Polarity	Cone pops inwards	
Continuity Test	Beep from Multimeter	

Table 18: Results for Testing the Breadboard-Friendly PCB Mount Mini Speaker - 8 Ohm 0.2W

12.16 Mini Metal Speaker w/ Wires - 8 ohm 0.5W

The same type of test will be performed for the metal 8 ohm 0.5 Watt mini speakers. To test this the only tools that will be needed is a dc power source and a multimeter. Since this is an 8 ohm speaker if the multimeter is in resistance setting, placing the probes on the speaker input and output should produce about 8 ohms. Then to check that the phase is good simply take a dc source around 1 volt since these are tiny speakers and apply a voltage to the input and output terminals. Having positive to positive and negative to negative will make the cone stick out and then reversing the polarity will make the cone stink inwards. Then a continuity check would be the last thing to test to make sure the speaker is intact. The table below will show the results of the tests.

Test Performed	Desired Outcome	Actual Outcome
Ohm Test	8 Ω	
1 Volt Positive Polarity	Cone pops outwards	
1 Volt Negative Polarity	Cone pops inwards	
Continuity Test	Beep from Multimeter	

Table 19: Results for Testing the Mini Metal Speaker w/ Wires - 8 ohm 0.5W

12.17 Bluetooth Prototyping and Testing

The Bluetooth communication is tested using RSSI readings and serial monitor. Using a free smartphone application, we can determine RSSI commands that will verify an established Bluetooth connection. A signal strength of -30dBm or more is considered according metageek documentation and will be used as a reference to determine the HUD's communication signal strength.

To check the validity of data, a serial monitor on Arduino IDE along with a Command Line prompt can be used. The data bytes transmitted are ASCII characters and because

ASCII is a global standard, it makes deciphering the code easier. A good software test will be to run a loop that transmits packets continuously and the receiving end sends an acknowledgement in return to validate that the data has been received. A counter should be incremented every time a packet is received and sent to track the number of packets exchanged and test if any packets were lost during transmission. If a packet is lost, debugging begins to see if the issues are due to RF or digital portions.

Oscilloscopes that display Bluetooth RF Frequencies are useful when troubleshooting and analyzing Bluetooth compliant devices. Although this hardware is not available to us, a regular oscilloscope will be more than sufficient for testing as the specification sheet provides us all the expected values needed to be read.

12.17.1 Bluetooth Module Bootloader

The nRF52840 module is a blank chip that must be flashed with code before use. To do this, we use a combination of a SWD (Serial Wire Debugger) Programmer with its hardware setup similar to the figure below, as well as DFU Bootloader tools in order to configure the kernel and other important aspects of startup code.

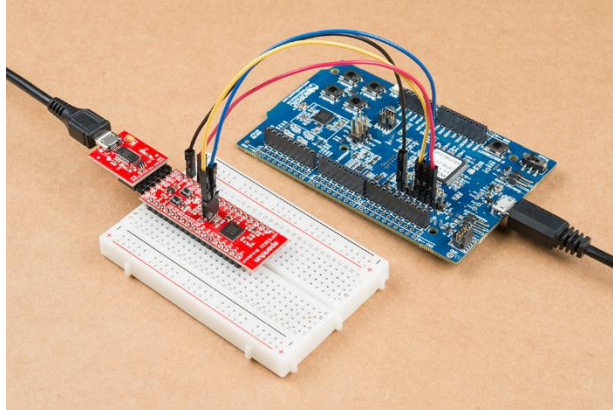


Figure 33: Bluetooth Module Bootloader

Nordic's SDK for boot loading uses a Python based tool for updating via serial called `adafruit-nrfutil`. This requires Python 3 to be installed on a computer and installed via PyPI through Command Line. Once all the necessary programming tools are installed and hooked up, we will use Nordic's Secure DFU bootloader mode in order to update firmware on the device and make it secure.

The way the DFU is secure is through the use of cryptography keys. Two keys are used, a public key and a private key. This is generated through the `nrfutil` library installed prior. The commands to do this are as follows.

```
nrfutil.exe keys generate private.key
```

This creates the `private.key` file that will be used for decryption. After this key is created, a public key is made through command line using the `private.key` file.

```
nrfutil keys display --key pk --format code private.key --out_file public_key.c
```

Now the respective public key that is used for encryption is created based off of the private key. With the two keys generated begins the compiling and building of the bootloader itself. An external open source library called `uECC` on github. The `uECC` library implements a small and fast Elliptic-curve Diffie–Hellman (ECDH) and Elliptic Curve Digital Signature Algorithm (ECDSA) algorithm that is made for specifically public key based cryptography, which is what we made earlier. This library is compatible with our 32-bit ARM processor within the BLE module and is written in C so everything is compatible. This library is then cloned into a SDK folder and compiled. Once this is compiled, the command “make” will build the bootloader based on the public key.

A DFU .zip packet is required for the DFU master to send new image file to the DFU target. With the hex file generated prior, the zip file is made with the following command

```
nrfutil pkg generate --hw-version 52 --application-version 1 --application
nrf52840_xxaa.hex --sd-req 0x98 --key-file private.key app_dfu_package.zip
```

--hw-version: By default this should match with the chip. Since we are using a nRF52xxx chip, we use "52".

--application-version: By default the start number for application version is 0. To be able to update new application, the application version should equal or greater than the one the bootloader stored. This case we use 1.

--sd-req: In this case the application runs with Softdevice S132 v4.0.2. The code number for this softdevice version is 0x98. This was found by typing *nrfutil pkg generate -help*.

--application : Tells nrfutil that you are going to update the application and the application image is provided.

With the final zip file created, the zip file is sent to the IDE used and update via serial.

With the bootloader applied to the device, any application code sent to the device must take into consideration that the boot loader exists in memory. Below is a memory map of addresses of the nRF52840.

Description	Start Address	End Address	Size (KB)
Bootloader Settings	0x000FF000	0x000FFFFF	4
Master Boot Record Params	0x000FE000	0x000FEFFF	4
Bootloader	0x000F4000	0x000FDFFF	40
Application Code	0x00026000	0x000F3FFF	824
SoftDevice	0x00001000	0x00025FFF	148
Master Boot Record	0x00000000	0x00000FFF	4

Table 20:

12.18 Communication tests

When packets are sent over from the device to the phone and vice versa, we must confirm that the data being sent over are not garbage values and that packets do not overlap each other. This is done through multiple ways. First we confirm the validity through RSSI, or “Received Signal Strength Indicator,” which is one of two ways to confirm the packet. The second is through measurement of the RF signal. Because Bluetooth is a radio frequency-based technology, an oscilloscope is used to measure the signal is dBm.

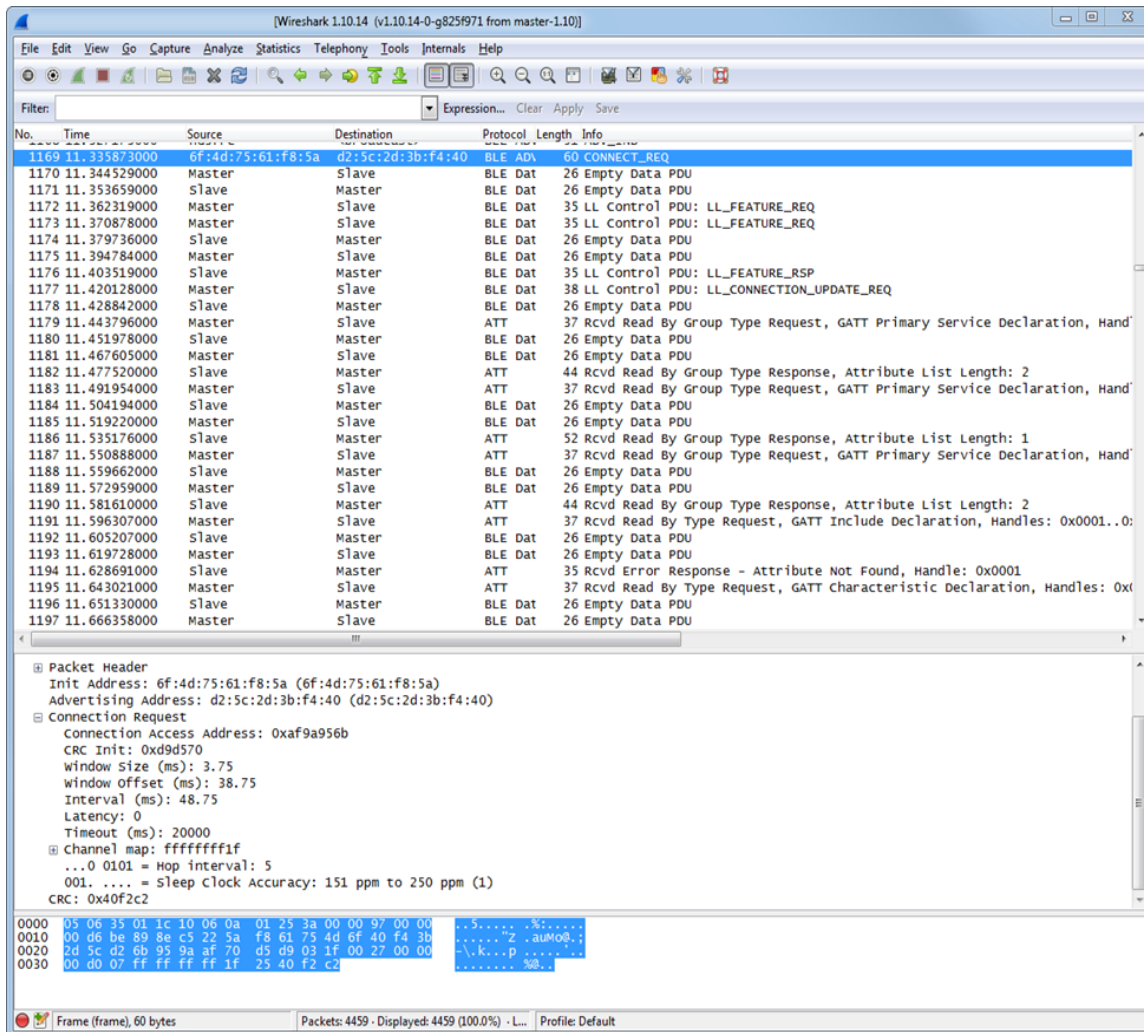


There exists free applications on the Apple AppStore that allows testing of Bluetooth devices in order to gain knowledge of what hex values are being sent. One of these applications is called LightBlue.

The HUD's BLE will be configured as a peripheral that has data the phone, the central, needs. With this application, we can send packets with a delay as low as 50 milliseconds and as long as 2 seconds. The data on Peripherals are organized according to their respective Profile. This profile contains a variety of services and characteristics. Characteristics are the holders of data, and can be accessed in 3 ways- Read, Write, and Notify. With these three types of data manipulation, we can fully test out the BLE module on a high level and make sure that distance is not a problem as well as any kind of interference.

Within the application we can find addresses to read and write to. In the figure above we can see their example has an address that starts at 0xFF10, and from there they can look at nearby addresses and determine what type of address it is. This way we can access data values such as RAM.

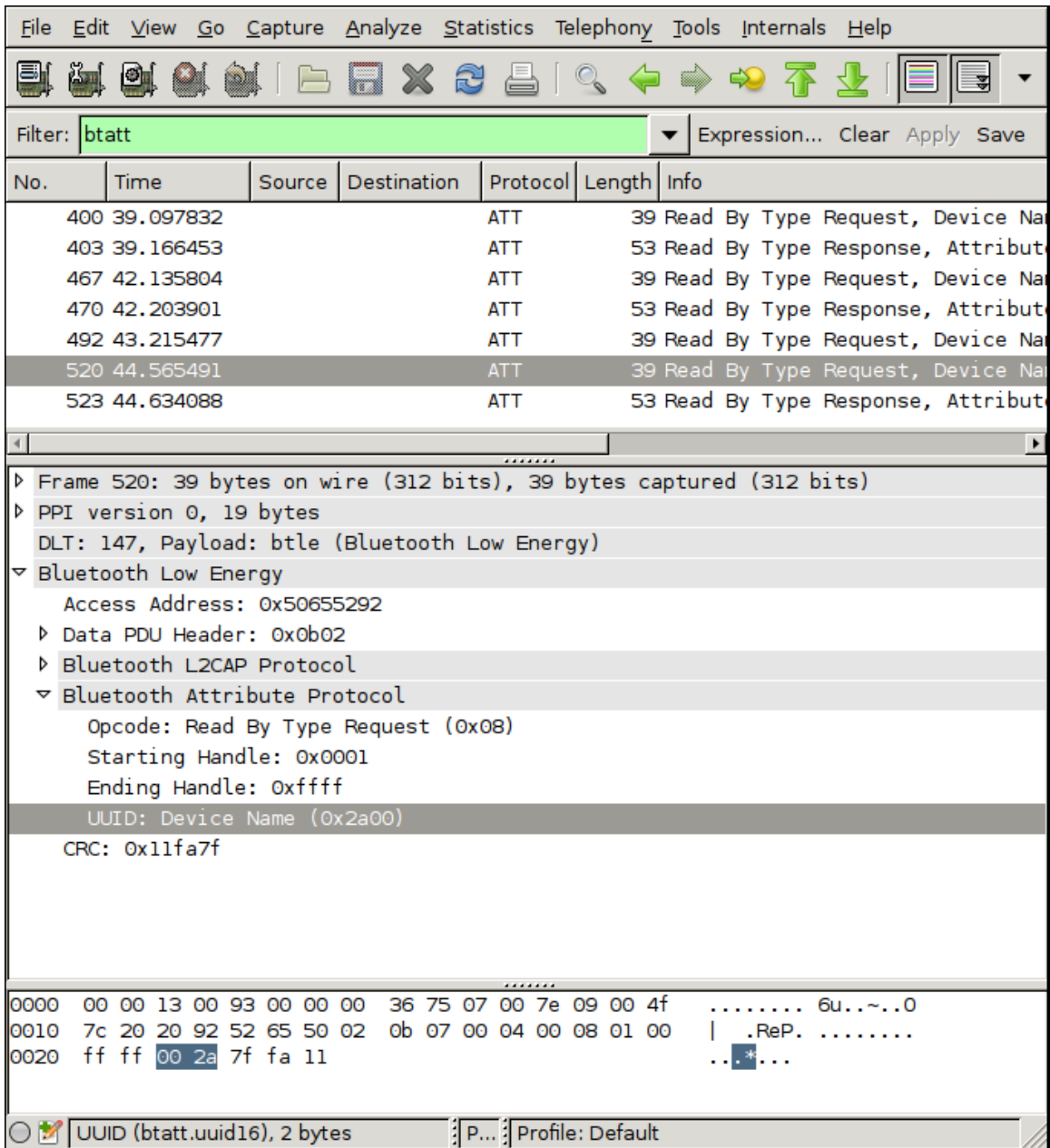
12.19 Packet Sniffing for Testing Both Nodes



This sniffer uses a Nordic evaluation board with special firmware programmed to the device. The nRF52840 captures the BLE packets, adds time, RSSI, and other metadata, and forwards them to Wireshark. The advantage here is that Wireshark is an industry standard tool that allows anyone to view your information. Wireshark is a free and open-source packet analyzer available for Android devices. Having software to check packets on both Android and iOS operating systems allows us to test on two of the most common smartphone operating systems on the market

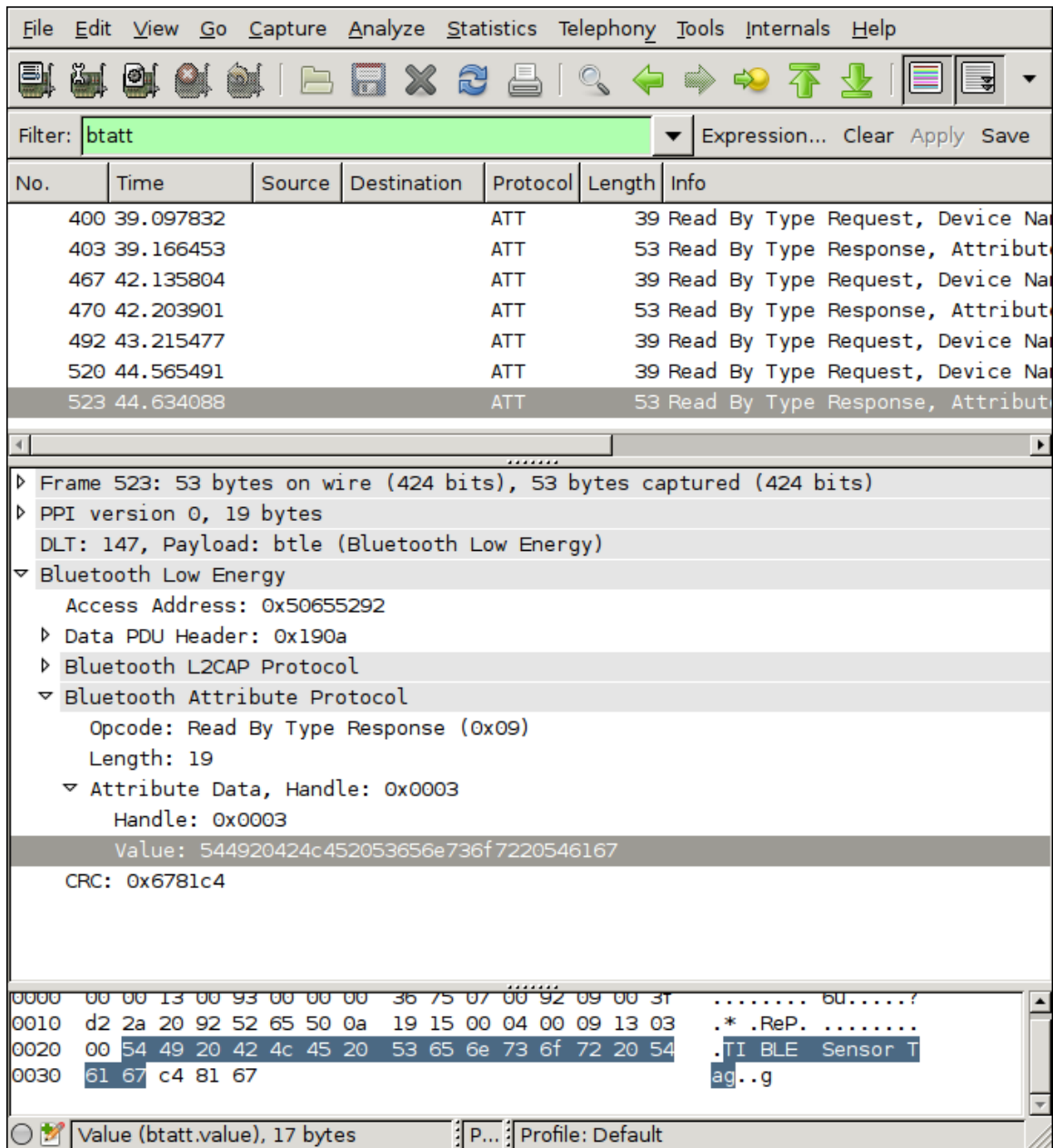
today. LightBlue and Wireshark will provide all the Bluetooth packet testing needed once the RF portions are confirmed to be operational and stable.

12.19.1 Packet Sniffing with WireShark



This is an example of what bluetooth traffic is formatted like when testing packets for sending and receiving. This excerpt is taken from the command prompt portion

of WireShark. Important information to take note of is the access address value, the attribute protocol that tells what the packet's characteristics are, as well as Opcode that says whether it is a packet being sent or received. The example above is a request from the computer to the BLE module asking for what the device name is. In the bottom where "00 2a" is highlighted, this is the portion that is being sent and specifically asks for the device name. When translated out of hex, it represents ".*" which does not mean anything to us but to the device, its recognized at requesting the name.



This next excerpt shows a packet received from the Bluetooth module in response to the packet sent above. The access address of the BLE is not surprisingly the

same as all communication is based off that address. The response is in hexadecimal which is highlighted above. Wireshark lists that the total length of the packet as 19. When the hexadecimal string is translated, it reads “TI BLE Sensor Tag” which is the name of the TI CC2540 that is used in this example. These two screenshots show that Wireshark is very powerful, handy, and fits our needs perfectly.

13.0 Project Hardware Design:

The hardware mentioned in this section are the physical components that will be used to power and control our heads-up display. This section will provide information concerning each section of hardware design.

13.1 Block Diagram

Our project requires a broad scope of knowledge to implement each subsystem to the final product. The group has broken down each task by topic and system, allocating our knowledge and resources efficiently. The block diagram in Figure 33 describes each task with how it fits into the final product

The left side of the block diagram, highlighted in yellow, contains all the electrical subsystems required to power the HUD device. Evan Hall, our Electrical Engineering student, is tasked with designing the power delivery and PCB aspect of the final design. Evan will design a system to deliver power from a USB port in the vehicle to the HUD device. As mentioned in Section 4.0, we will be implementing different electrical components to power each component of the design.

The middle section of the block diagram, highlighted in blue, contains all the display focused components of the HUD device. Aaron Majdali, our Optics and Photonics Engineering student, will be designing and implementing the display unit, ambient light sensor, optical focusing system, and the mounting of the display within the vehicle.

The right section of the block diagram, highlighted in green, contains all the programming and microcontroller hardware required to implement the HUD device. Logan Glowth and Pedrhom Nafisi, our Computer Engineering students will be tasked with designing and building the mobile application, computing components, and data transferring methods

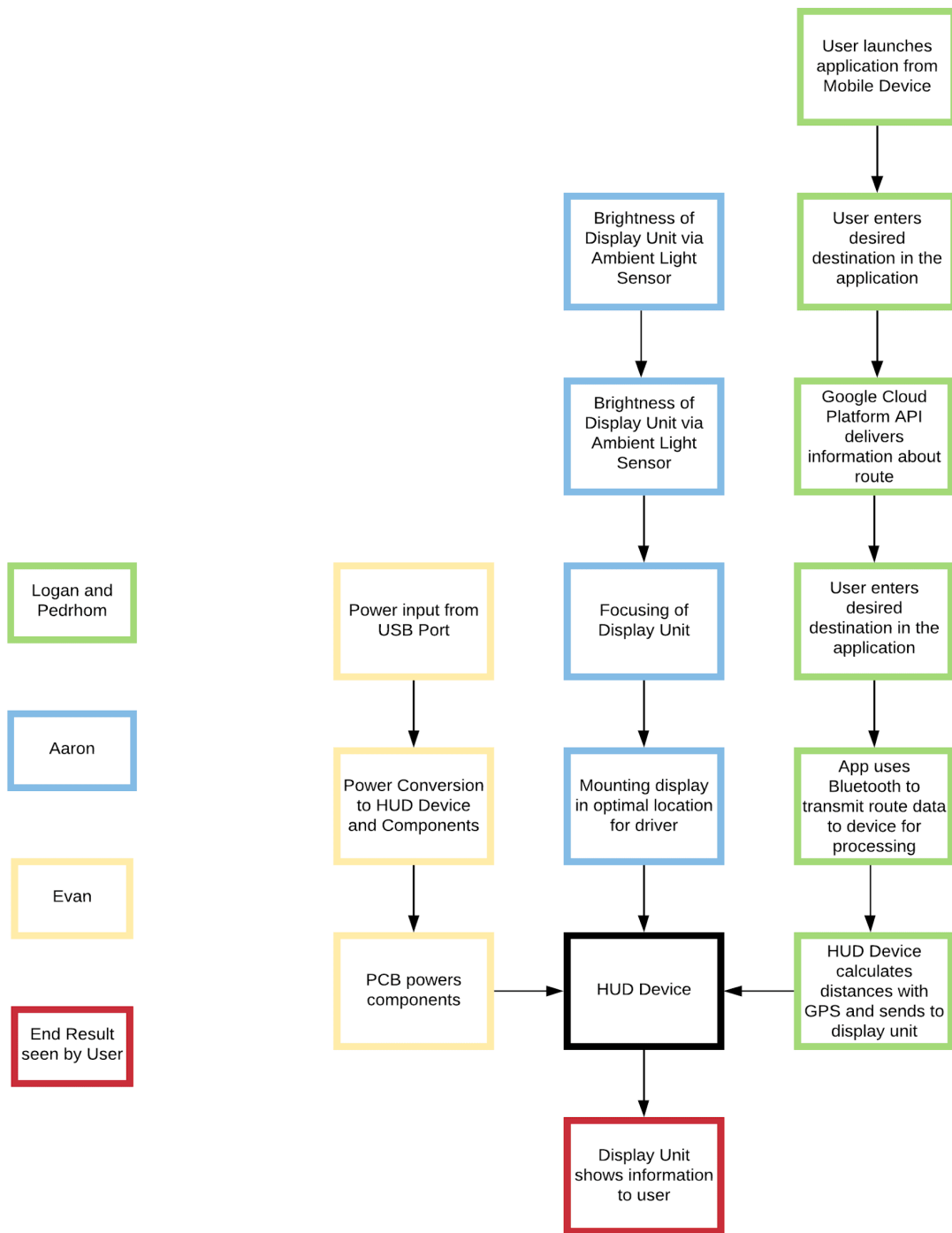


Figure 30: Block Diagram

13.2 Display System

The primary optical elements of a heads up display are a projector and a combiner. The projector is anything that can project an image onto the combiner, which is a transparent surface that will show the user both the outside world and the projected image. The first step in determining what components to use for the heads up display was to determine how we wanted to display the data. At first, it was thought that the display could merely be a bright screen reflecting off of a windshield or a smaller vertical piece of glass. Upon consultation with several experts, it was decided that our heads up display would feature a collimated image, generated via a smaller screen and a positive lens.

13.2.1 Projector Design

Our heads up display will feature a small transparent monochromatic LCD display, illuminated by a bright LED. The LCD, an ENH-DG128064-66, will be located one focal length away from a positive lens. Selection of the positive lens will be based on several factors. The lens must be cost-effective to accommodate our budget, it must have a large enough aperture to properly display the image from the LCD, it must have enough magnification to allow the information to be easily seen by the driver, and it must have a short enough focal length for us to fit the system into our specified size. The LED that we choose must be able to pass enough light through the LCD to be seen in the combiner on a sunny day. Fortunately, there are many available LED's on the market that can fulfill our brightness needs. Our main concerns will be brightness and power consumption. As bright LED's can produce a lot of heat, and this heat can affect the LED's performance and health, it may be necessary to put a simple heat sink on the LED. These three main components will be mounted into opposite sides of the housing. It should be possible to mount the PCB and other components underneath the LED, LCD, and lens. If an aluminum housing is used, it may even be possible to mount the LED in such a way that the housing itself is the heat sink for the LED. The LED may also need a diffusor, so that the light is evenly spread across the LCD screen. The LCD can simply be mounted, via a light clamping at each corner or even adhesive. The lens can be clamped or glued in place. If glue is used, it must be a kind that is not sensitive to temperature and sunlight.

13.2.2 Combiner Design

Further testing will determine what will be used as the combiner. The combiner needs to be a piece of glass that sits directly in front of the driver and can reflect the heads up display image without obstructing the driver's view of the road. The two options we have are using the windshield, or using a small sheet of glass that will sit close to vertical. Regardless of the option chosen, it may be necessary to use a reflective film. Such a film can be purchased online and is easily applied to a glass surface. The film would allow more light from the projected image to be reflected towards the driver. If we choose to use a vertical piece of glass as the

combiner, then the device must be oriented so that the reflected projection can reach the driver. Likewise, if the windshield is used as the combiner, the device will have to project nearly vertically to produce a reflection that can be seen by the driver. If we do decide to use a vertical glass screen as the combiner, then our device will have to be modified to mount the glass.

14.0 Tools

Implementing a project of this scale becomes much easier and more efficient with the aid of tools at our disposal. The College of Engineering and Computer Science provides a myriad of resources to assist with the design and implementation of the Senior Design project. The following section is devoted to the tools our team will take advantage of throughout the course of Senior Design.

14.1 Communication

Clear channels of communication play a pivotal role in staying organized and maintaining the set schedule. The team will be utilizing a few different communication tools to ensure the project stays on track and is completed within the given timeline.

14.1.1 Discord:

Discord is a free VoIP application that provides the ability for users to talk, video call, chat, share files, screen share, and much more. Many of us already use Discord as a primary communication platform. Discord contains the functionality needed to archive group discussions in the event that referencing them is needed. This will be the preferred communication platform moving forward

14.1.2 Text Messaging:

Text messaging has been the group's initial form of communication since the beginning of Senior Design I. Sharing content across MMS is difficult as it compresses images and video to the point that it is not very usable. We will be shifting to utilize Discord moving forward.

14.2 File Preservation

Technology is not perfect, and accidents happen. In an event where a computer crashes and data are lost, we want to ensure all our hard work is backed up and protected. We will be utilizing Google Drive to store all the documents and related materials to the project. Google Drive is free and provides us with enough storage to preserve all our data.

14.3 Other Software

14.3.1 LucidChart:

LucidChart is a web-based application that allows users to collaborate in creating charts, diagrams, and other related tools. Some of the figures in this document were created using LucidChart. LucidChart is a paid service but provides student licenses to users that apply with their University's domain email address.

15.0 Conclusion

Over the next two semesters, we will research, design, and build a Heads-Up display for a vehicle. The Heads-Up Display will be designed to increase driver awareness and safety when driving. Information will be displayed to the driver in a fashion that will not impede with the driver's view. The application we will create will allow the driver to choose a destination. The device will contact the Google Maps API and provide an efficient route to the destination. The device will then calculate the driver's current location using the attached GPS module and compare this location to waypoints along the route. The display unit will provide the user with a visual representation of the route and guide them to the destination with turn-by-turn directions.

The device is designed to reduce the likelihood for a driver to be distracted while driving. However, there is always the possibility for other drivers to be distracted. In the event that a crash occurs, we will have an impact detection system that will trigger when a strong or sudden impact is registered. This system will send an automated SMS text message to the user's designated emergency contact with information regarding the user's last reported location at the time of the impact. We hope to facilitate the ability for those impacted by the crash to get the help they need as soon as possible.

The device mentioned above will be designed and realized utilizing the knowledge that each member of the team has accumulated throughout their time at the University of Central Florida. We hope that this device will provide us with

challenging and exciting design decisions, while promoting a positive effect on our community by producing a project focused on increased safety and awareness.

16.0 References:

Note: Images used throughout this document are pending approval for use. They are the creations of their rightful owners and are subject to change to abide by the copyright regulations and the wishes of the image creators.

SENT FOR APPROVAL:

Arduino ATmega 2560 Schematic

ADXL335 (<https://www.analog.com/en/products/adxl335.html#product-overview>)

HAVE NOT CONTACTED FOR APPROVAL YET

[1] IEEE 802.15.1, IEEE, 2005.

[2] "Bluetooth 5 & BLE: Achieving maximum throughput", 2017. [Online]. Available: <https://www.novelbits.io/bluetooth-5-speed-maximum-throughput/> [Accessed 27/10/2019]

[3] "Bluetooth Low Energy Technology - Technical Information", 2014. [Online]. Available: <https://web.archive.org/web/20140214120404/http://www.bluetooth.com/Pages/low-energy-tech-info.aspx>

[4] <https://californiaconsultants.org/wp-content/uploads/2014/05/CNSV-1205-Decuir.pdf>

USB Micro-B Breakout Board Product ID: 1833. Digital image. https://www.adafruit.com/product/1833?gclid=Cj0KCQjw6eTtBRDdARIsANZWjYbPv2PNecbHn-w0ufd4guBG1Gbflosh96lIN8Usan008xAM-ppynC4aAiOUEALw_wcB

BOB-12035. Digital image. <https://www.digikey.com/product-detail/en/sparkfun-electronics/BOB-12035/1568-1194-ND/5673780>

2174507-2. Digital image. https://www.te.com/usa-en/product-2174507-2.html?te_bu=Cor&te_type=srch&te_campaign=ggl_usa_cor-ggl-usa-srch-selectmktg-fy20-genpl_sma-

[1176_1&elqCampaignId=82989&mkwid=s%7Cpcrid%7C379682292390%7Cpkw%7C2174507-2%7Cpmt%7Ce%7Cpdv%7Cc%7Cslid%7C%7C%7C%7Cpgrid%7C77094862456%7Cptaid%7Caud-428581635047:kwd-303524422505%7C&gclid=Cj0KCQjw6eTtBRDdARIsANZWjYYUoTOlqH_kF4HvT1HfXuOvi64KU3XV9RP9lhCDD7UyvA6FJD83xxYaAkuaEALw_wcB](https://www.amazon.com/dp/B01FDEWS7U/ref=asc_df_B01FDEWS7U/?tag=hyprod-20&linkCode=df0&hvadid=167146065113&hvpos=1o1&hvnetw=g&hvrnd=13668942640763954150&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmld=&hvllocint=&hvllocphy=9057291&hvtargid=aud-798931705416:pla-304542141486&psc=1)

Texas Instruments. *TPS6122x Low Input Voltage, 0.7V Boost Converter With 5.5µA Quiescent Current*. PDF. October 12, 2019

Super XL6009 DC-DC Adjustable Step-up Boost Power Converter. Digital image. https://www.amazon.com/XL6009-Adjustable-Step-up-Converter-Module/dp/B01FDEWS7U/ref=asc_df_B01FDEWS7U/?tag=hyprod-20&linkCode=df0&hvadid=167146065113&hvpos=1o1&hvnetw=g&hvrnd=13668942640763954150&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmld=&hvllocint=&hvllocphy=9057291&hvtargid=aud-798931705416:pla-304542141486&psc=1

Linear Technologies. *LT1613 1.4MHz, Single Cell DC/DC Converter in 5-Lead SOT-23*. PDF. October 15, 2019

Texas Instruments. *LM1577/LM2577 SIMPLE SWITCHER® Step-Up Voltage Regulator*. PDF. October 15, 2019

TO220 Bolt On Heatsink. Digital image. <http://www.reuk.co.uk/wordpress/electronics-and-electricity/buy-to220-bolt-on-heatsink/>

Adafruit ALS-PT19 Analog Light Sensor Breakout. Digital image. https://www.adafruit.com/product/2748?gclid=Cj0KCQjw6eTtBRDdARIsANZWjYbF6W_jP3p3-3inJj6vZAhEjTdL001FtJpAdnoUMf2bSY44IGcK_ykaAgJkEALw_wcB

Idea for Mounting. Digital Image. <https://www.skrten.com/products/skrten-navigation-hud-display>

[6] White, Ian. *The History of Air Intercept Radar & the British Nightfighter*. Pen & Sword, 2007.

[7] Kim, John. *Rupture of the Virtual*. DeWitt Wallace Library, Macalester College, 2016.

[8] Previc, Fred and Ercoline, William. *Spatial Disorientation in Aviation*. AIAA, 2004.

ublox. *NEO-6: u-blox 6 GPS Modules Data Sheet*. PDF. October 30, 2019.

[9] <https://www.adafruit.com/product/4078>

[10] <https://www.argenox.com/library/bluetooth-low-energy/ultimate-guide-to-debugging-bluetooth-smart-ble-products/>

[11] <https://www.extremetech.com/computing/115251-how-usb-charging-works-or-how-to-avoid-blowing-up-your-smartphone>

DOESN'T REQUIRE APPROVAL

“Digital Micromirror Device.” *Wikipedia*, Wikimedia Foundation, 29 July 2019, https://en.wikipedia.org/wiki/Digital_micromirror_device

“DLP Products – Getting Started.” *DLP Getting Started | DLP Products | TI.com*, <http://www.ti.com/dlp-chip/getting-started.html>.

“nRF52840 Info Center”

https://infocenter.nordicsemi.com/index.jsp?topic=%2Fstruct_nrf52%2Fstruct%2Fnr52840.html&cp=4_0

“U Drive. U Text. U Pay.” *NHTSA*, 8 May 2019, <https://www.nhtsa.gov/risky-driving/distracted-driving>.

Czernia, Dominik. “Car Crash Calculator.” *Omni*, Omni Calculator, 22 Feb. 2019, <https://www.omnicalculator.com/physics/car-crash-force>.

[4] “Transport Canada Guidelines to Limit Distraction from Visual Displays in Vehicles.” *Transport Canada*, Government of Canada, 27 February 2019, <https://www.tc.gc.ca/en/services/road/stay-safe-when-driving/guidelines-limit-distraction-visual-displays-vehicles.html>