Huddy Buddy

Group 6

Pedrhom Nafisi (Computer Engineering)

Logan Glowth (Computer Engineering)

Aaron Majdali (Photonic Science and Engineering)

Evan Hall (Electrical Engineering)

Motivation

- Driving with GPS usually involves taking eyes off the road
- Off-the-shelf solutions are focused very close to the windshield and require that the user's phone is placed on top of the dashboard
- Must not be a fixed, permanent add-on
- Combination of knowledge across disciplines

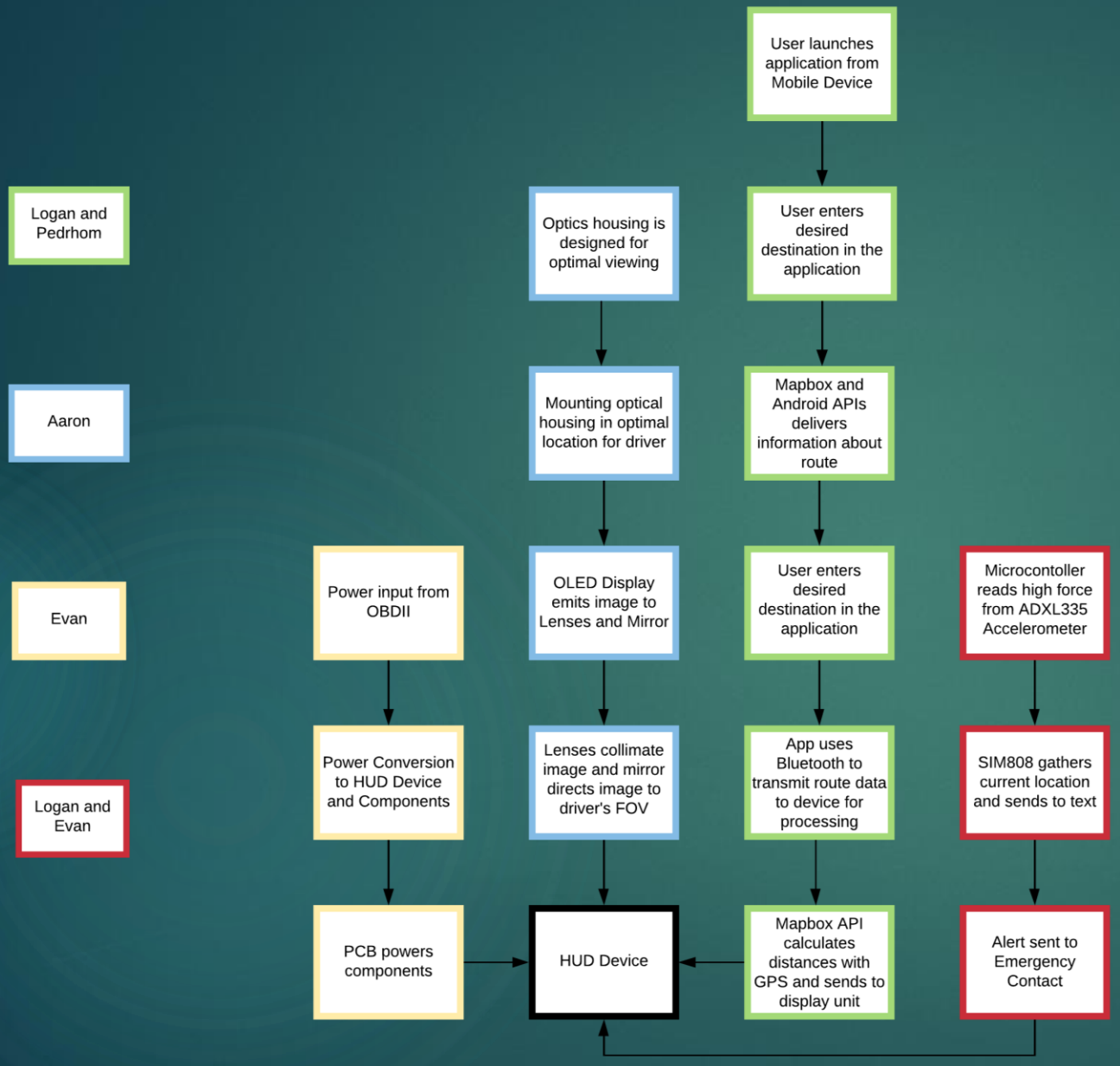
Objectives

- Design a system that displays up-to-date GPS directions from an established mapping service
- Ensure compatibility with modern vehicle OBDII ports and GPS-enabled Android-based smart phones
- Increase safety of operating a vehicle with GPS guidance and crash detection system
- Design must be cost-effective
- The Huddy Buddy must leave a positive impression on the user

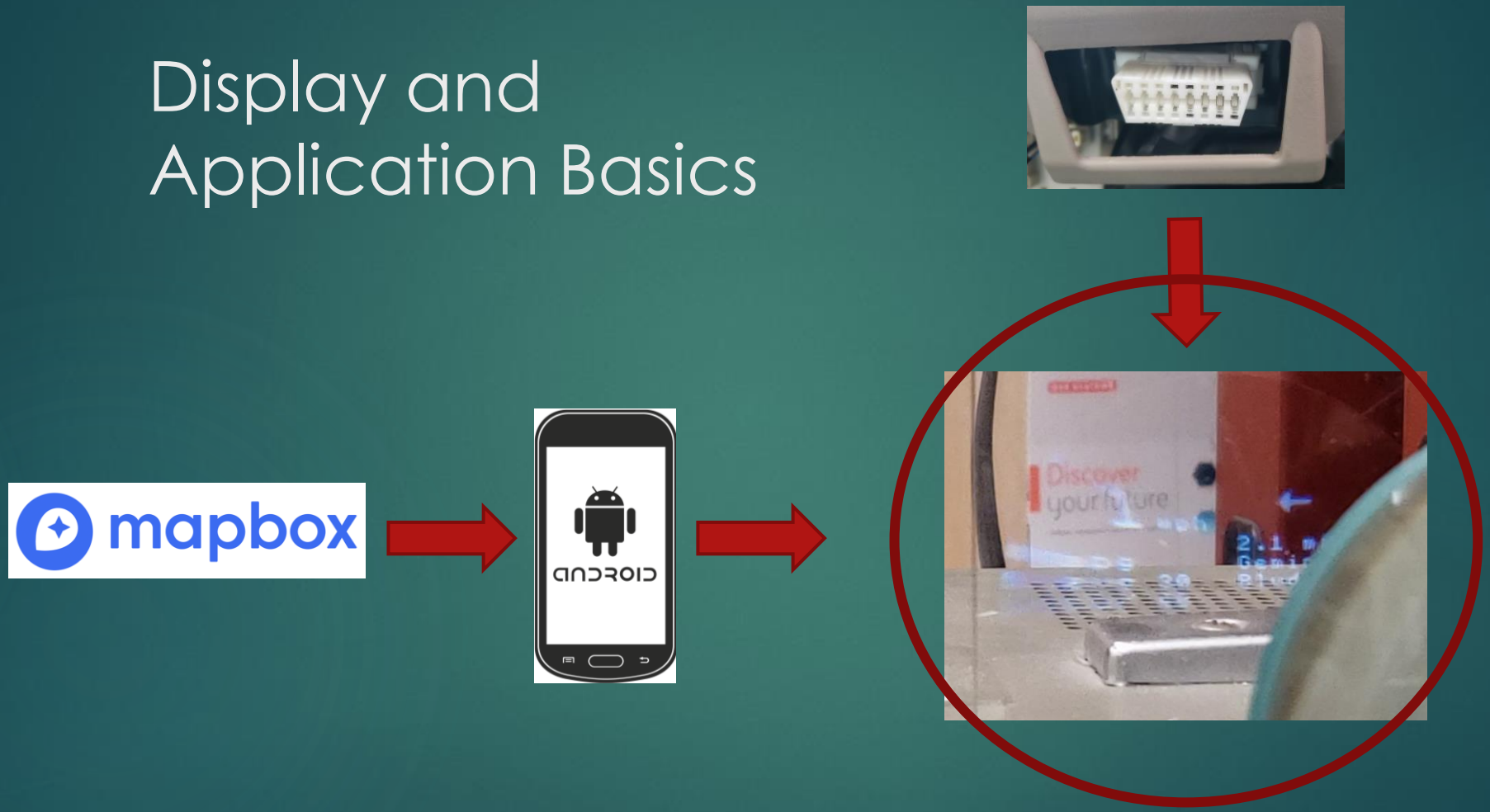
Specifications

Component	Parameter	Design Specification
Case	Size	9" x 5" x 2"
Device	Weight	< 3 lb.
Device	Temperature	-20 C to +60 C
Device	Sunlight Operability	Stored and Operated in Direct Sunlight
Power Delivery	Power Consumption	≤2.1 amps at 5 volts
Power Connector	Power Source	OBDII
Application	OS Compatibility	Android
Application	Navigation Source	Mapbox API
Wireless Connector	Connectivity Standard	Bluetooth 2.0+EDR
Image Source	Display Surface	OLED Screen
Display	Resolution	128 x 64 Pixels
Mounting System	Position	Adjustable Positioning

Block Diagram



Display and Application Basics

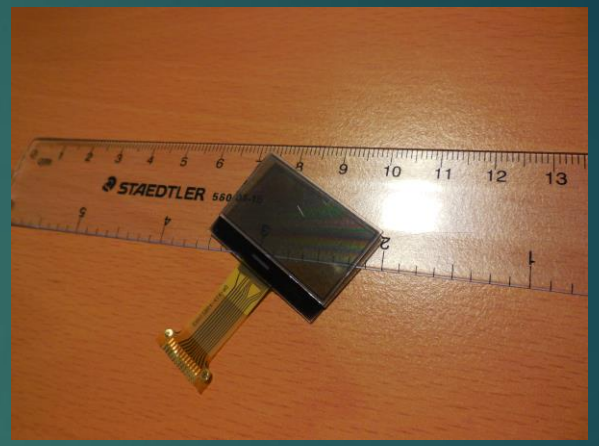


Subsystems of Huddy Buddy

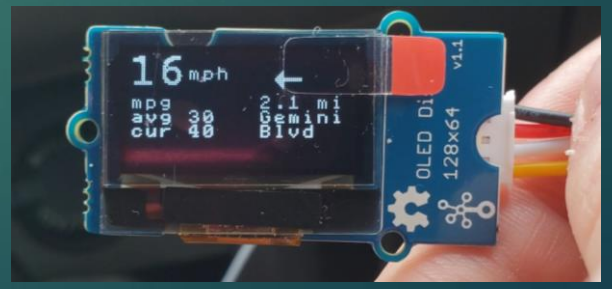
- Display Unit
  - OLED Display, Lenses and Mirror, Housing
- ATmega 2560 Microcontroller
- SIMCom SIM808 2G GSM GPS Module
- Analog Devices ADXL335 Accelerometer
- Power Delivery
  - OBDII Power Sources, Backup Battery
- PCB Development
- Android Application
- HC-05 Bluetooth Module

Display Options

- **ENH-DG128064-66 Transparent LCD**
  - 128x64 resolution
  - Paired with a high-intensity LED for illumination
  - Low cost

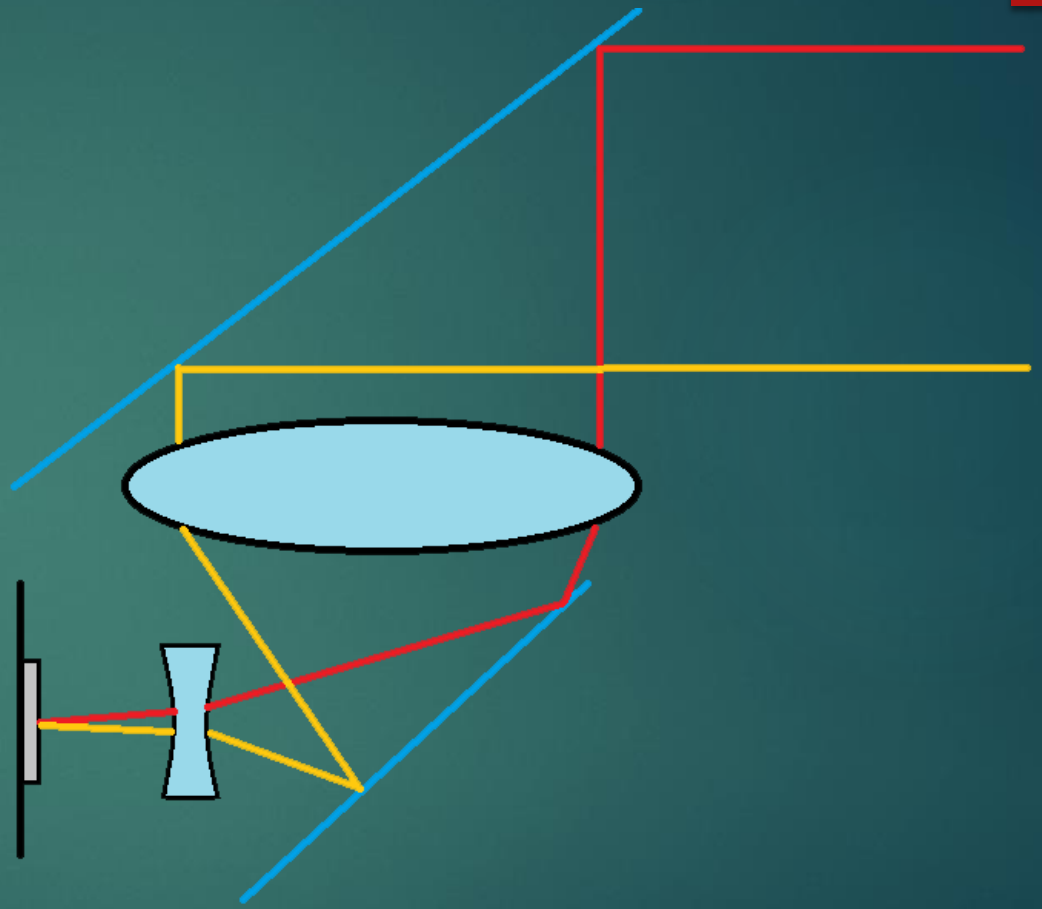


- **SeeedStudio Grove OLED 0.96” Display**
  - 128x64 resolution
  - Inter Integrated Circuit (I2C) Protocol
  - Minimum footprint on PCB (4 pins)
  - High Contrast without need for Backlight
  - Can be directly adjusted for brightness



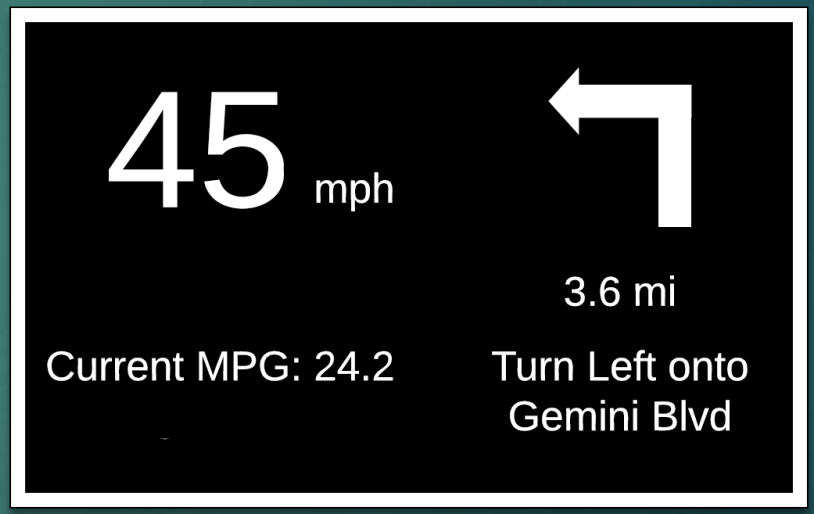
Lens System

- Heads up displays work best when focused at infinity, or "collimated"
- The best way to collimate the image is to place it at the focal point of a lens system
- When the driver looks at distant objects, the image will always be in focus



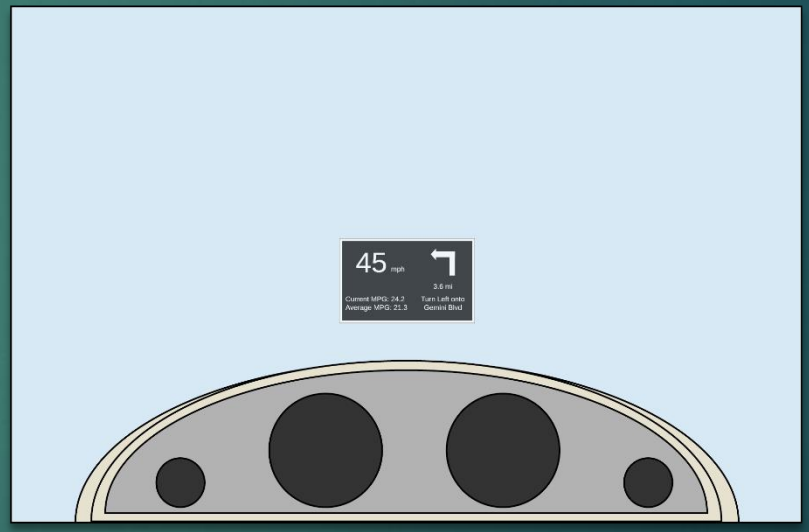
Adding information to display unit

- Speed and Fuel Economy data pulled from OBDII port in vehicle
- Navigation data from API calls to Mapbox via Android App
- Microcontroller sends data to be displayed over serial connection



Mounting device in vehicle

- Unit mounted on dashboard behind instrument cluster
- Just inside driver's field of view



Microcontroller

- **ATmega 2560 8-bit Microcontroller**
  - 54 Digital I/O Pins, of which 15 Pins provide PWM Output
  - 16 Analog Input Pins
  - 5V Operating Voltage
  - 256KB Flash Memory
  - 16 MHz Clock Speed
- Popular for embedded projects such as robotics
- Open Source Schematics

- **Texas Instruments MSP430F447 16-bit Microcontroller**
  - 48 I/O Pins
  - 1.8V – 3.6V Operating Voltage
  - 32KB + 256B Flash Memory
  - 16 MHz Clock Speed
- Easy to program with built in bootloader

Crash Detection System

- Microcontroller uses accelerometer to detect large change in forces
  - Read in values for gravity relative to x, y, and z axes
  - If a large change in values is detected, microcontroller will initiate emergency response sequence
- Microcontroller issues commands to Cellular 2G Module
  - Retrieve current GPS Coordinates
  - Send text message containing coordinates to emergency contact
- In event of power loss, backup power takes over
  - Power generation switch on impact

Crash Detection System

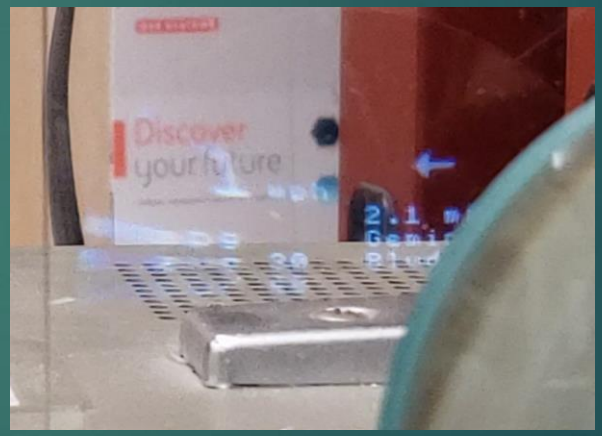
Cellular and GPS Module

- **SIMCom SIM808 GSM 2G Cellular Module**
  - Quad-Band GSM/GPRS Module with GPS for satellite navigation
  - US Cellular Service via standard SIM Card
  - Ting Mobile
  - AT Command Implementation
  - Receives commands via UART Communication
  - External GSM/GPS Antennas
  - Supply Voltage: 3.4~4.4V

Accelerometer

- **Analog Devices ADXL335 3-Axis Accelerometer**
  - ± 3g sensitivity
  - Operating voltage: 1.8V - 3.6V
  - Analog pinouts for x,y,z axes
  - 10,000 g shock survival
- **Analog Devices ADXL337 3-Axis Accelerometer**
  - ± 200g sensitivity
  - Operating voltage: 1.8V - 3.6V
  - Analog pinouts for x,y,z axes
  - 10,000 g shock survival

HUD Overview



Power Delivery

- The power supply will be through the OBD-II port that are in almost all cars today.
  - This will supply a steady 5-volt input and can deliver up to 2.1 amps.
  - Another reason for this choice is because it is possible to pull additional data from the car's sensors and display them.

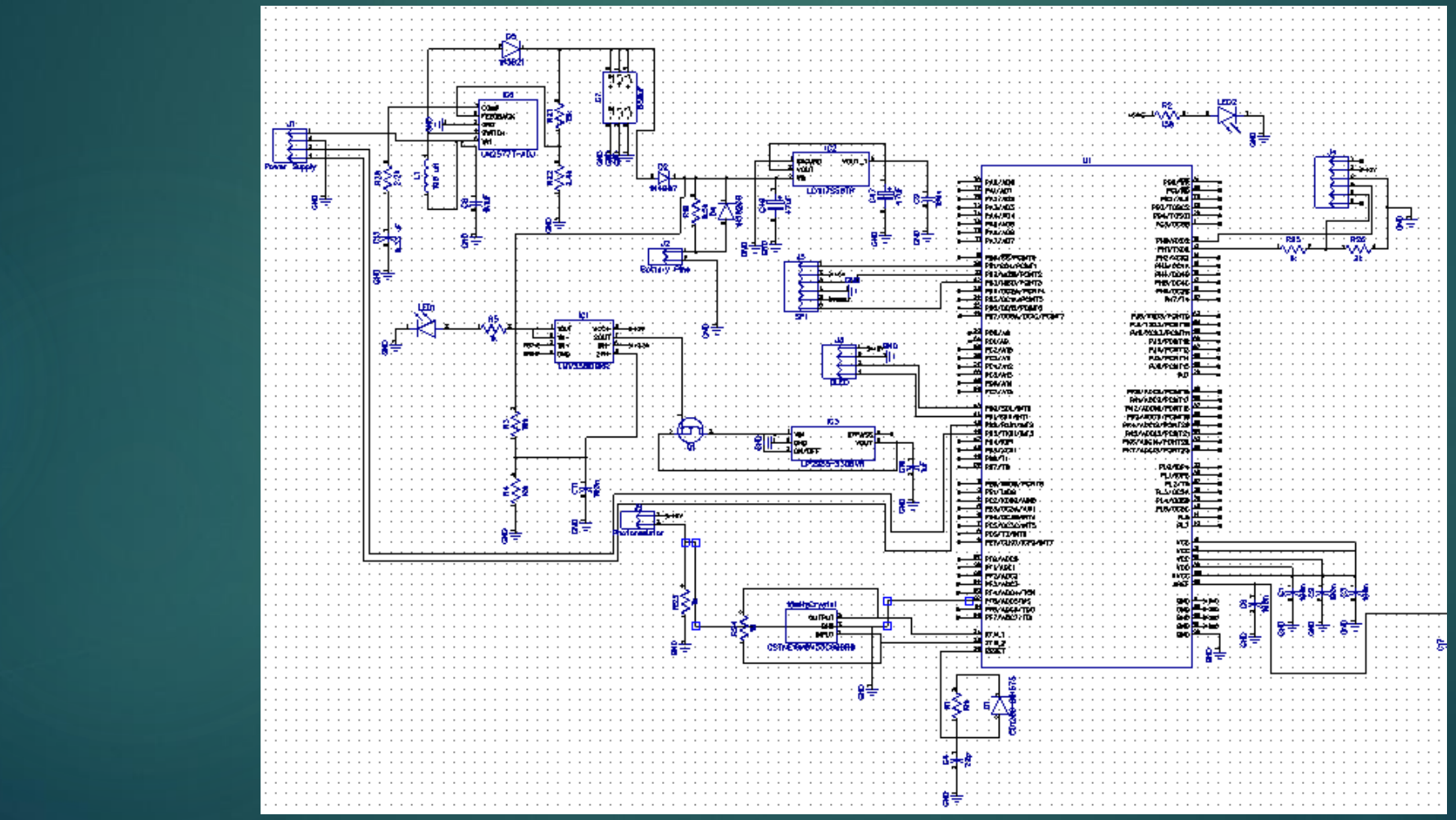
Power Delivery

- Most of the components will be driven off the 5-volt input but for a few of the peripheral parts, they require either more or less voltage is needed and to do that a buck or boost converter is used.
  - The LM2577-ADJ was supposed to be used for the LED backlight which will supply 9 volts and up to 800 mA if needed.
  - The LD1117S50TR was used instead of the NCP1117ST50T3G, listed on the ATMega2560 schematic, because it was not a part that could be readily sourced. It has the same specs, with a slightly better dropout voltage and better current handling.

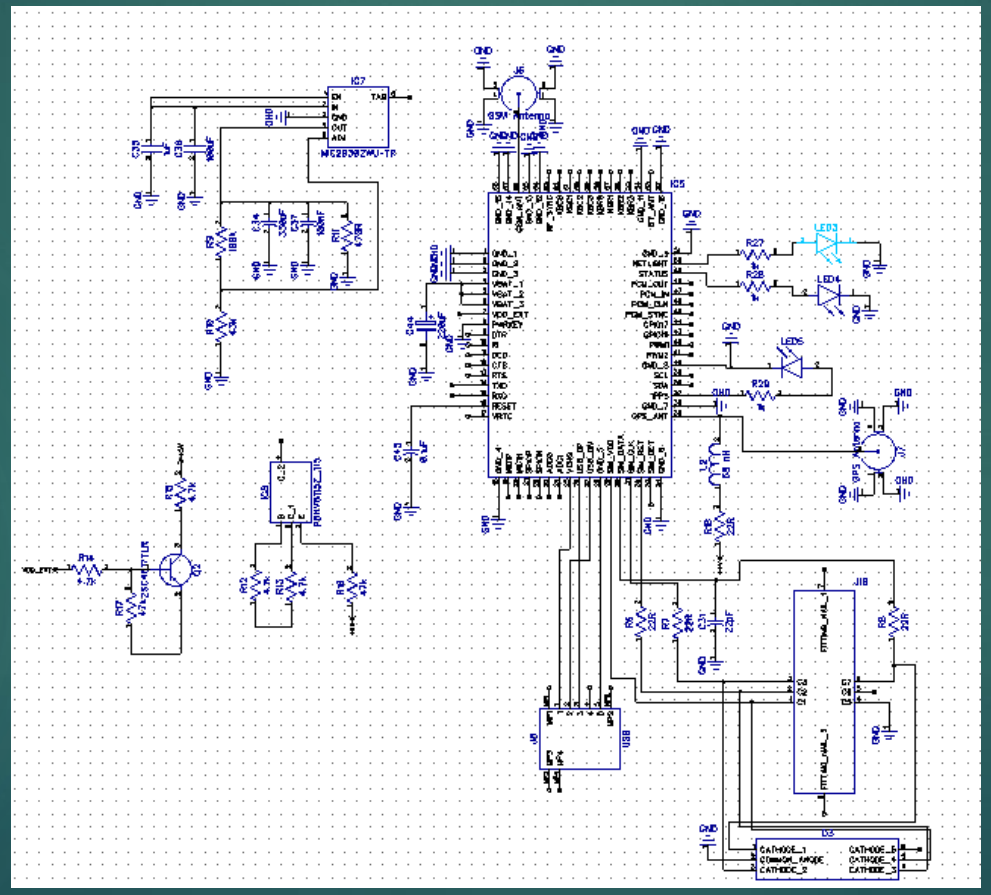
PCB Development

- Being a beginner with designing PCBs there are a lot of designer tools out there and the one I went with is called DipTrace. It is very easy to use and in my opinion is a lot like Multisim which most everyone is familiar with.
  - One small drawback is having to make individual part schematics and footprints, but there are tools to easily make the parts that were not included in its already extensive library.
  - The supplier for most of the RLC components can be obtained through Mouser.com. The actual chips may come from separate vendors. The PCB will be created through JLCPCB.

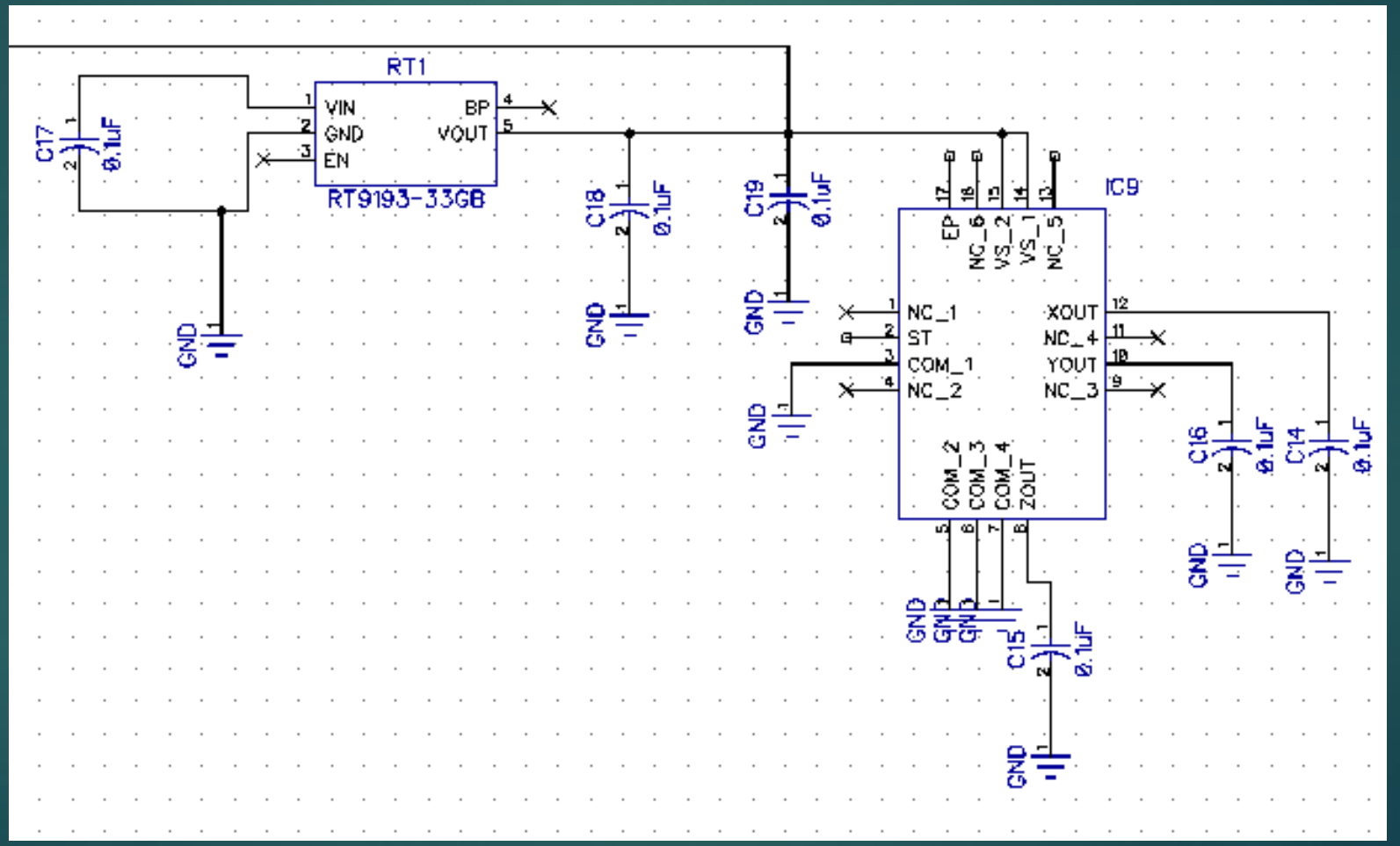
PCB Development (ATmega2560)



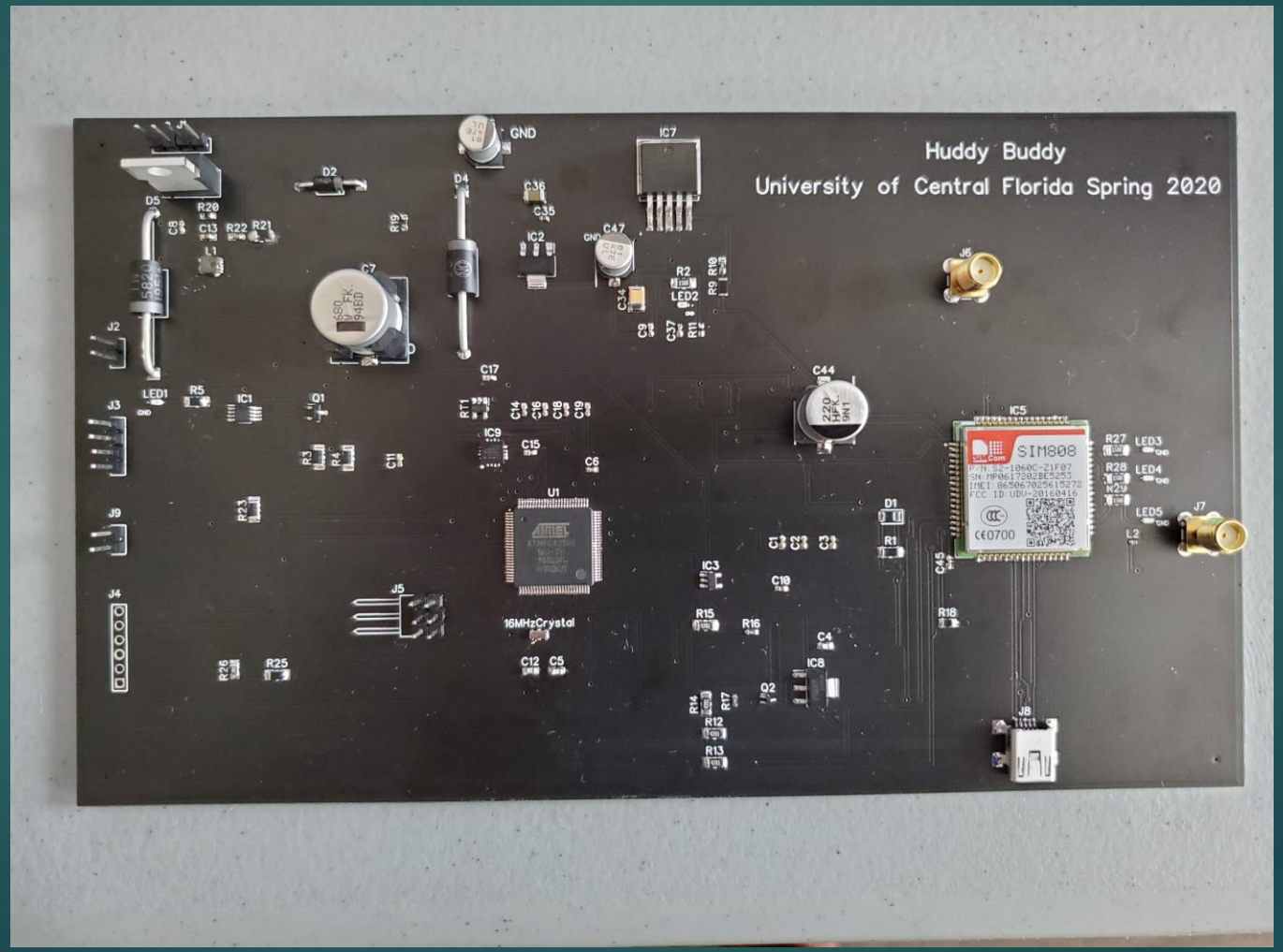
PCB Development (SIM808)



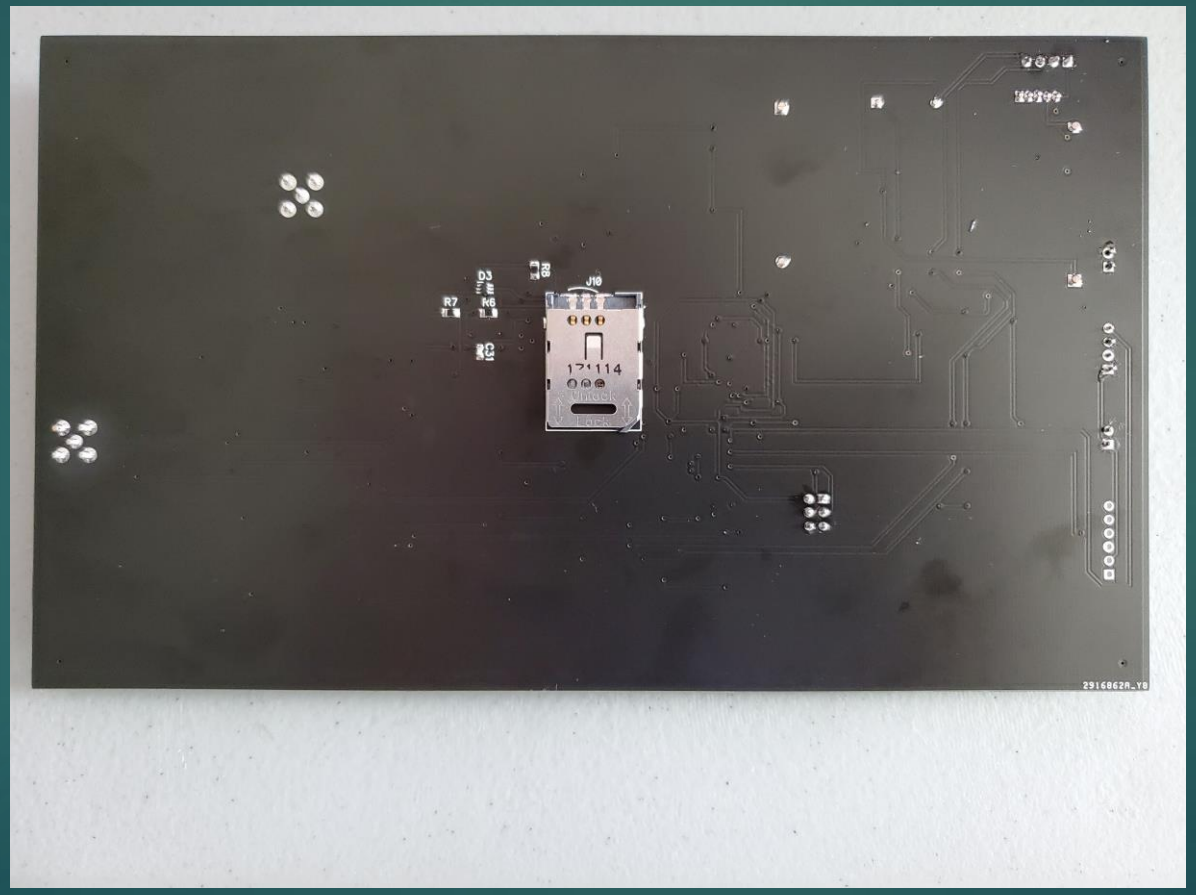
PCB Development (ADXL335)



Huddy Buddy Version 1.0 (Front)



Huddy Buddy Version 1.0 (Back)



Complications with PCB

- First of all, there was the delay of getting our PCB constructed once COVID-19 shut everything down. With small parts such as the ATMega2560, Sim808, and ADXL335, the backbone of the project, being very difficult for beginners and without the right tools, we sought assistance from a company called Quality Manufacturing Services who offer free help for college students on their senior design projects. No response was given starting from March, 20th until April 1st. Once I put in the request for help the PCB was taken in on April 3rd and not returned until April 10th due to low staffing.

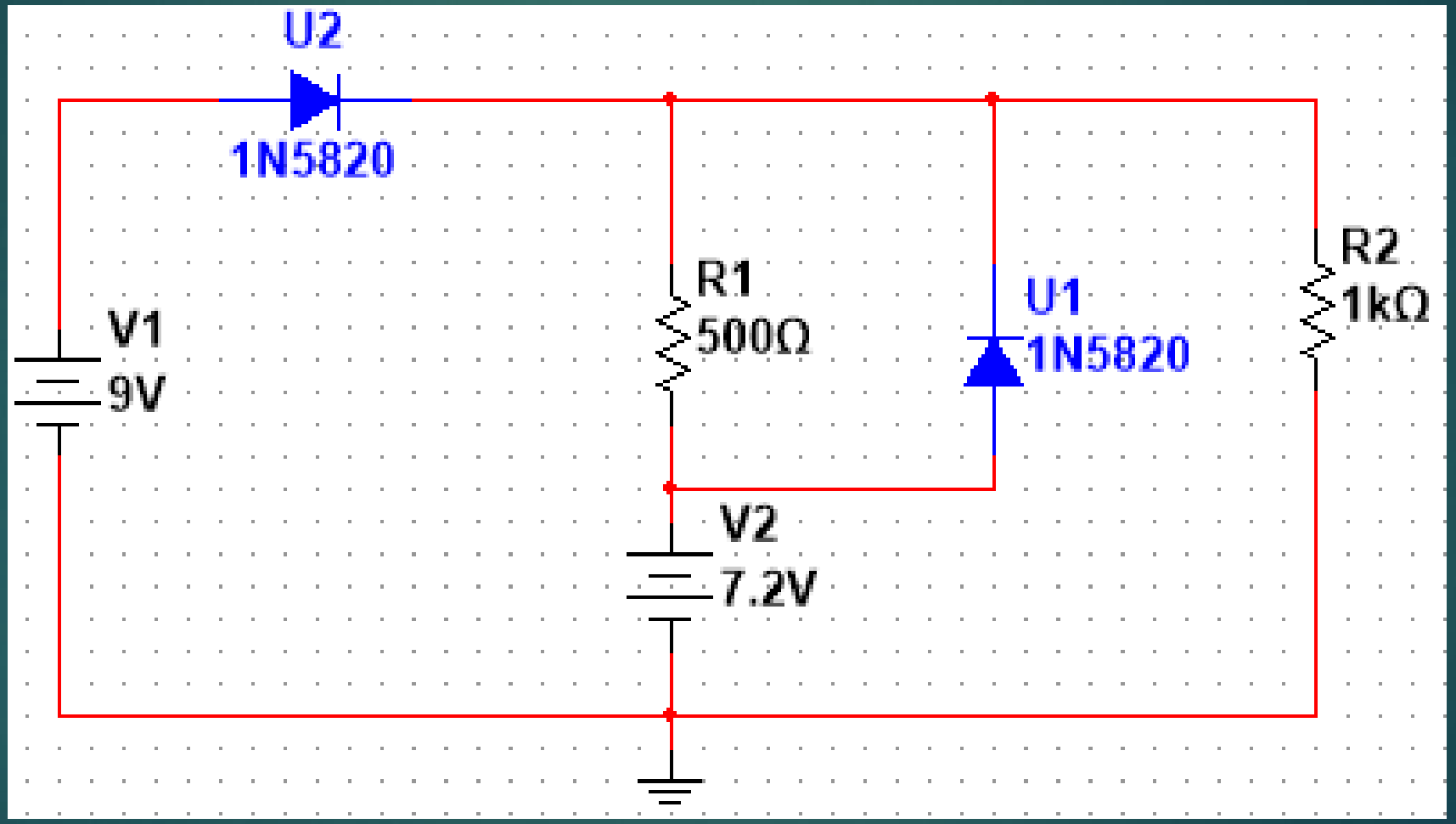
Complications with PCB

- Once the PCB was returned, testing began. A connection between the ATMega2560 and the OLED was established which was a big success since the Arduino Uno was used as the bootloader. From there we tried to test the whole system but something crucial burnt. An inductor that turns out cannot handle the total load current started to smoke and could not test further. I believe the Sim808 started to pull more current than anticipated.

Backup Battery System

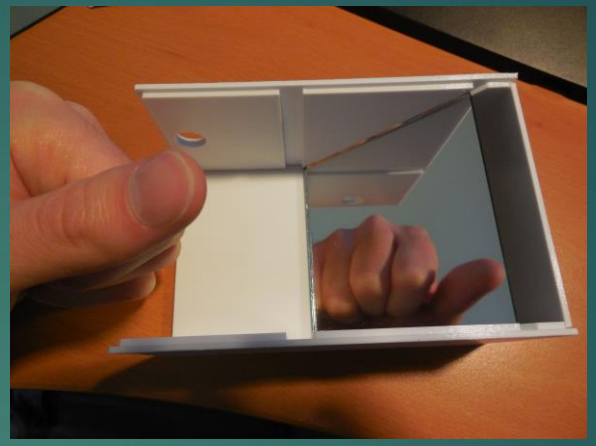
- In the event of a crash and no power can be provided to the SMS module, there still needs to be power to allow for the emergency text message or email to be sent out to the respective emergency contacts.
  - The normal DC power supply will be connected to a blocking diode rated at a current higher than the power supply. Then the rechargeable battery is connected to the circuit with a resistor and another diode. The value of the resistor is very important because while the battery isn't being used its going to be charged.

Backup Battery System

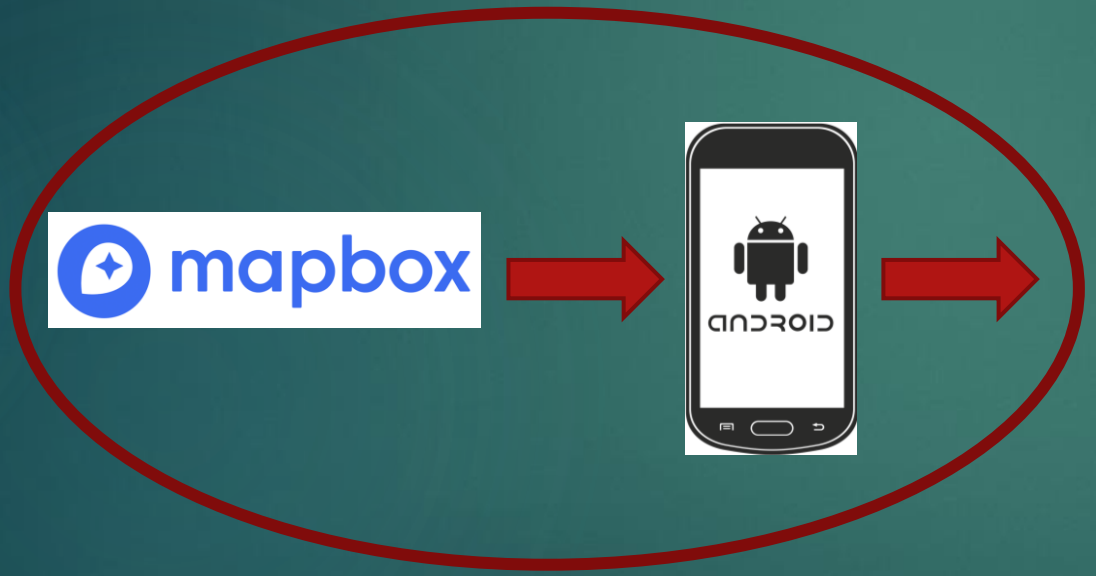


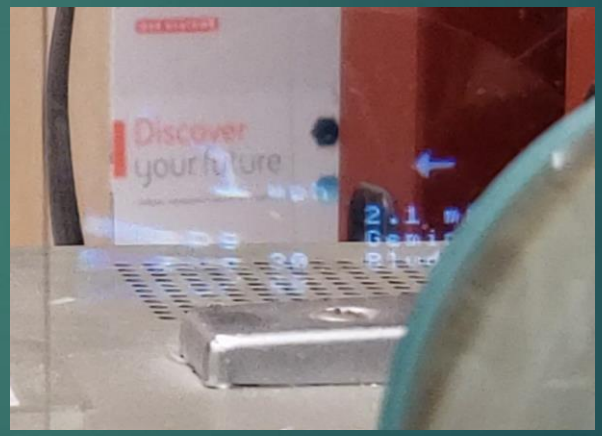
Display Enclosure

- 3-D Printed Enclosure
  - Made to calculated measurements
  - Holds lenses, mirror, and display at desired locations
  - Could reasonably made from common materials like Delrin, but is 3-D printed due to current events



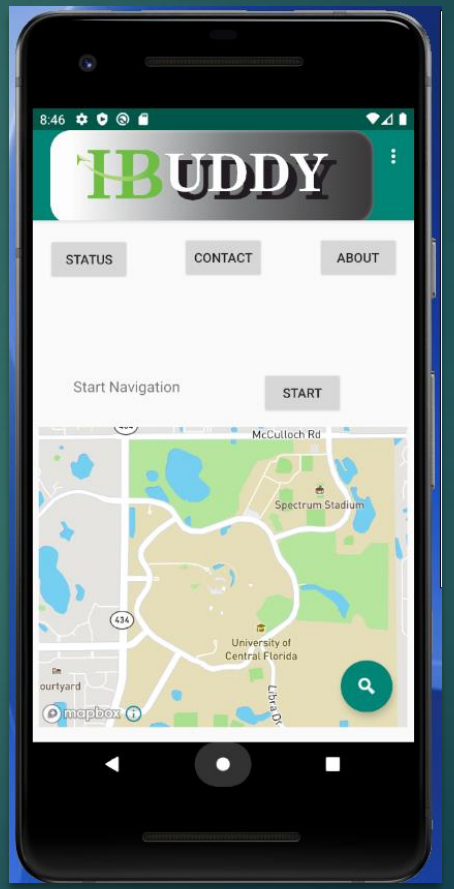
HUD Overview





App Development

- Android Studio using Java
- Mapbox for API calls and data
- Choose location to route to based on current location of device
- Allows checking of Bluetooth connection status
- Designate Emergency Contact



Bluetooth

HC-05 Master/Slave Module

- 2.4 GHz Transceiver
- 3 Mbps, Long Range
- Bluetooth 2.0+EDR
- Arduino development environment
- Low cost, very common part

nRF52840

- Bluetooth Low Energy
- Nordic SDK environment
- More expensive, smaller size

HC-05

nRF52840

Testing

- **Microcontroller**
  - Created Arduino Sketches for each component
    - SIM808 Cellular Module – Adafruit Fona Library
    - ADXL335 Accelerometer – Adafruit Accelerometer Functions
    - OBDII UART Adapter – Freematics OBDII Library
    - Grove OLED Display – u8g2 Display Library
  - Combined all of the sketches into one, connected all the devices to the module with the breadboard
  - Successfully able to pull speed and fuel economy from OBD, detect a force that would constitute a crash, get the current GPS coordinates for the Huddy Buddy and send to an emergency contact

Testing (cont.)

- **HC-05 Bluetooth Module**
  - Hooked up to an Arduino Nano to configure and test initial setup with AT commands through serial
- **Android Application**
  - Android Studio development environment on a Google Pixel 2 emulator with a Samsung Galaxy S8 for physical testing.
- **Custom PCB**
  - Used an Arduino Uno as boot loader to push the sketches to the PCB. Connected OLED screen to PCB. Supplied 5-volts to PCB to power on. Disconnected 5-volts main supply to test backup battery. Used photodiode to print lux values on PC.
- **Collimating Lens System**
  - Aligned lenses and display with a plexiglass beam splitter to determine ideal spacing of lenses relative to calculated spacing

Work Distribution

name	Optics and Display	PCB and Power Delivery	HUD Programming	App Development
Aaron	✓		✓	
Evan		✓	✓	
Logan		✓	✓	
Pedrhom			✓	✓

Financing

- Exclusively Student Funded
  - Logging all items purchased
  - Splitting overall cost at end of project
- Use of tools provided by the University of Central Florida
  - TI Innovation Lab
  - Senior Design Lab

Budget

Item	Description	Price ($)	TOTAL ($)
PCB	Implement all hardware needed onto PCB	40	455
Display Unit	Display information to driver on windshield	100	
Power Delivery System	Provides power to system	~45	
Microprocessor	Processes data and sends to display	50	
LEDs	Show power states and pertinent information	15	
Soldering Equipment	Needed to implement electrical components	Provided by UCF	
Bluetooth Module	Receive data from phone over Bluetooth connection	10	
Bluetooth Testing Development Board	Allows debugging and initial exposure to the HC-05 through Arduino Nano	15	
Oscilloscope	Allows testing of RF portions and voltage differences	Provided by UCF	
Speaker	Play recorded sounds in specific situations	~10	
Mobile Smartphone	Needed to host custom Application	N/A	
GPS/Cellular Module	Feed location data to system and send SMS messages	90	
Battery	Used as backup power source	20	
OBDII Connector	Needed to get data from vehicle's OBDII Port	45	

Bill of Materials

Item (quantity)	Vendor	Description	Price ($)
SeeedStudio Grove OLED Display	SeeedStudio	Display information to driver on windshield	18.80
Adafruit ADXL335 Accelerometer	Adafruit	Processes data of forces on chip	~14.95
Arduino Mega 2560	Amazon	Processes data and sends to display	28.50
Soldering Equipment	N/A	Needed to implement electrical components	Provided by UCF
HC-05 Bluetooth Module	DSDTech	Receive data from phone over Bluetooth connection	10.00
Arduino Nano Bluetooth Testing Development Board	Arduino	Allows debugging and initial exposure to the HC-05 through Arduino Nano	15.00
Oscilloscope	N/A	Allows testing of RF portions and voltage differences	Provided by UCF
Mobile Smartphone	N/A	Needed to host custom Application	N/A
SIM 808 GPS/Cellular Module	Amazon	Feed location data to system and send SMS messages	26.98
Battery	18650batterystore.com	Used as backup power source	20.00
Freematics OBDII Adapter	Freematics	Needed to get data from vehicle's OBDII Port	39.98

Bill of Materials

Item (quantity)	Vendor	Description	Price ($)	TOTAL
PCB	JLCPCB	Implement all hardware needed	~35.00	392.20
ATMega2560 (x2)	Mouser	Arduino based microprocessor used to compile all commands and procedures for making Huddy Buddy work	23.70	
ADXL 335 (x2)	Mouser	3-axis accelerometer used to detect a crash	11.40	
Sim808 (x2)	Adafruit	Allows capability of using GPS and GSM, sending SMS messages	59.90	
LM2577T-ADJ (x2)	Mouser	Steps 5-volts to 9-volts	12.20	
MIC23902WU-TR (x2)	Mouser	Used to step down 5-volt to 4-volt for Sim808	4.28	
RT9193-33GB (x2)	Digi-key	Used to step 5-volts down to 3.3-volts for ADXL335	1.00	
LD1117S50TR (x2)	Mouser	Steps the 9-volts down to 5-volts for ATMega2560	0.80	
47388-2001(Sim Holder) (x2)	Mouser	Allows testing of RF portions and voltage differences	3.18	
Collective rest of parts and passive components (x2)	Mouser	Extra parts that make the main components work correctly	~18.00	
PH0534NCV	EISCO	3-inch Double-convex lens	12.59	
PH0536HCC	EISCO	2-inch Double-concave lens	9.49	
1613-59	Mirrors	3-inch square mirror, 5 pieces	6.53	
Enclosure	B. Miller	Custom 3-D printed enclosure	20.00	

Challenges

- Dealing with consequences of CoVID-19
  - Delayed shipments
  - Remote work
- Selecting a display with high enough brightness for daylight operation
- Ensuring compatibility with a wide variety of vehicles and smartphones
- Ensuring correct sizing of 3-D printed components
- Unexpected issues with components
  - Durability of SIM5320 3G GSM Module
  - Configuration of SIM808 Module and OBD Library
  - Software Debugging

Huddy Buddy