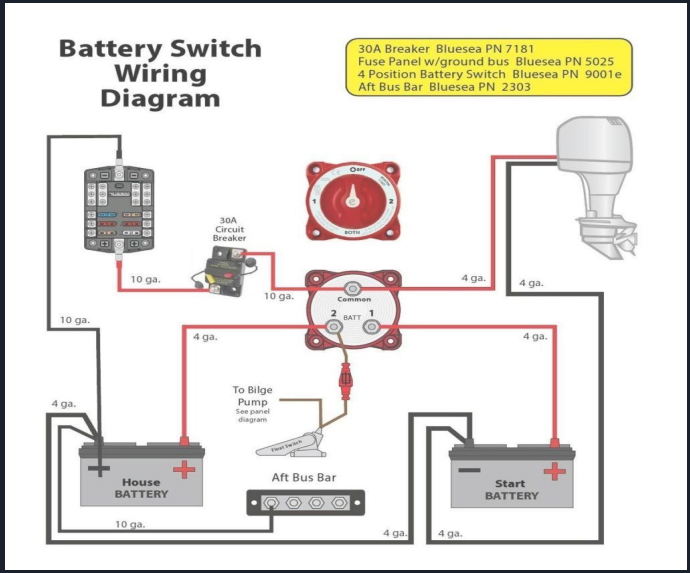
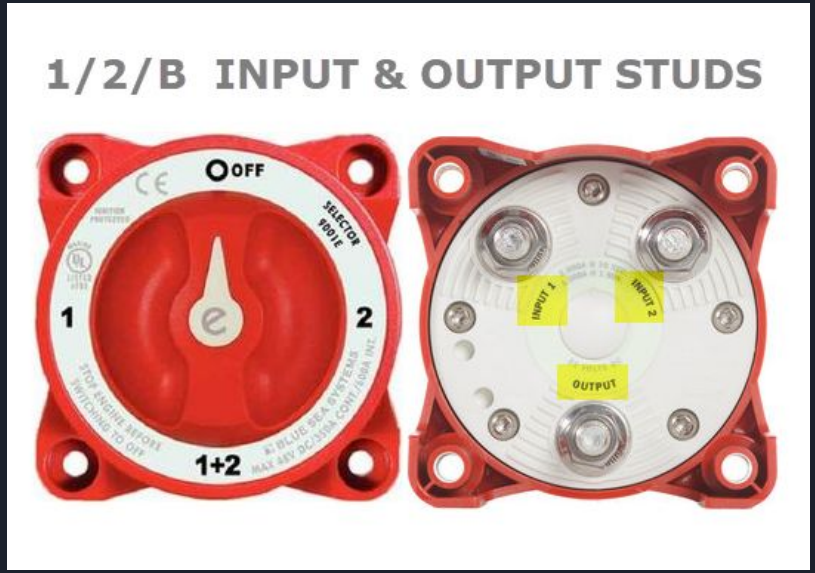
Intelligent Battery Controller

Group 21: Jed Boyer, Joey Roche, Chris Jones

Introduction

- Our goal was to build an intelligent device to switch battery options on recreational devices
- This device would be able to make decisions and switch on it’s own, to replace the existing manual transfer switch





Objective

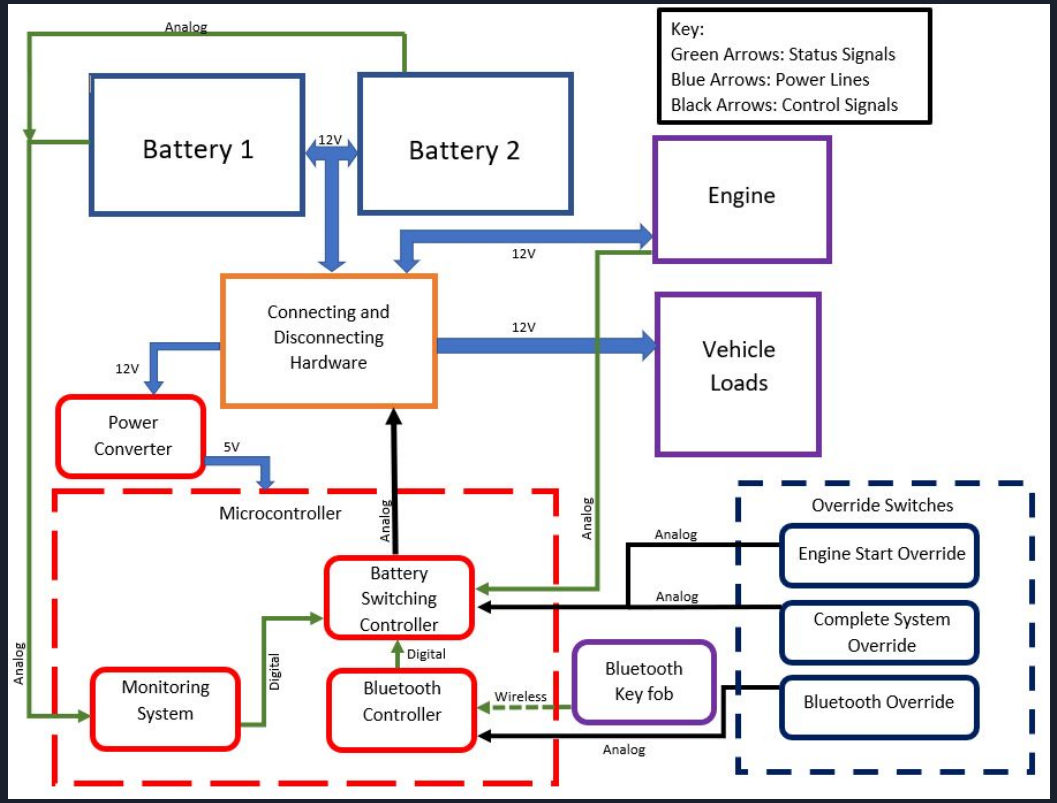
- Monitor the status of both batteries
- Varying output based on battery voltage levels and conditional triggers.
- Switch based system to maintain user control over the device
- Control battery usage to preserve the health of deep-cycle batteries.

The system solves the problem of user error in battery management leading to situations where the vehicle is unable to start. It also extends the lifetime of deep-cycle batteries.

Battery Voltage	Available Charge
12.6+	100%
12.5	90%
12.4	80%
12.3	70%
12.2	60%
12	50%
11.9	40%
11.7	30%
11.6	20%
11.3	10%
10.5	0%

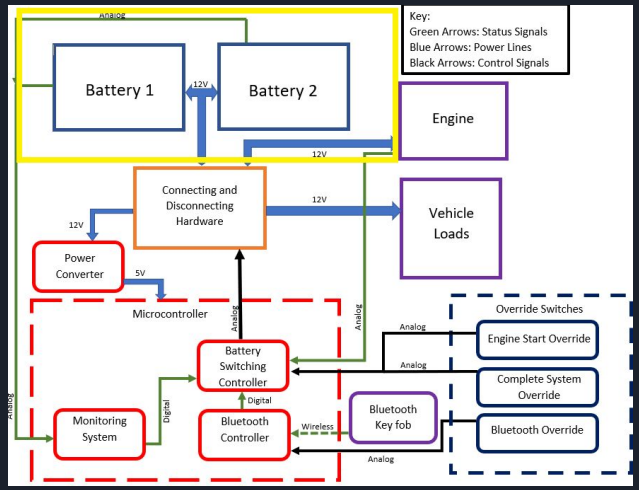
Battery Voltage
Rounded to Nearest 10th

Design (Block Diagram)



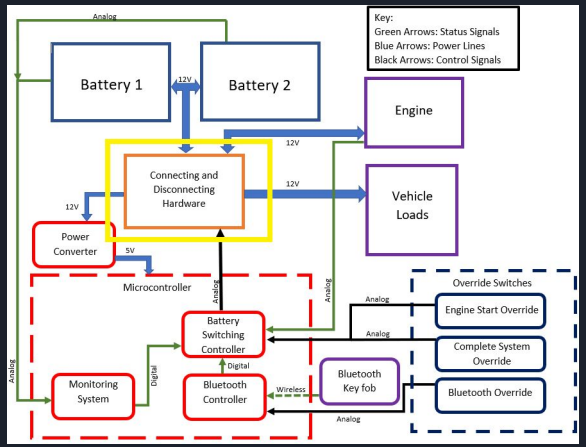
Batteries

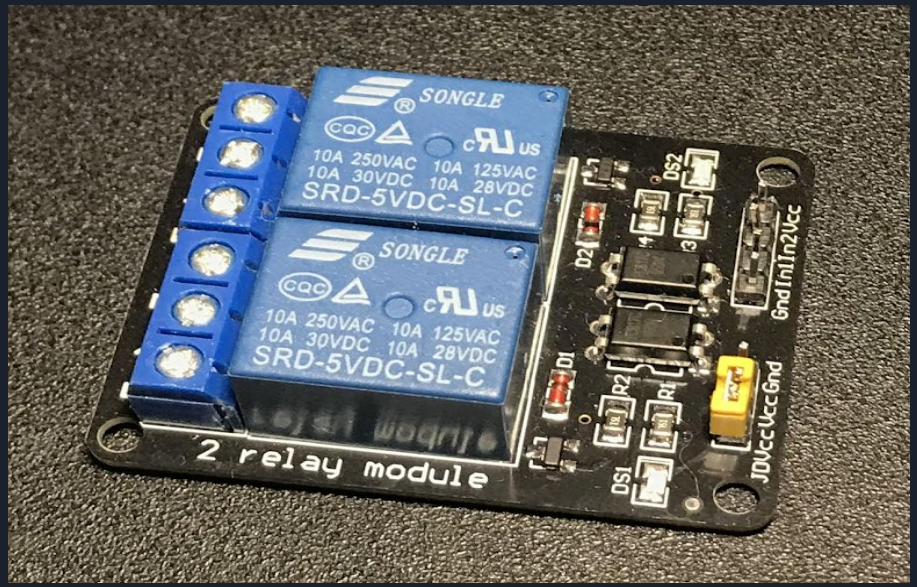
- 2 battery packs
  - Each with 8 AA batteries
  - Adds up to a total of 12V for each pack



Connecting/Disconnecting Hardware

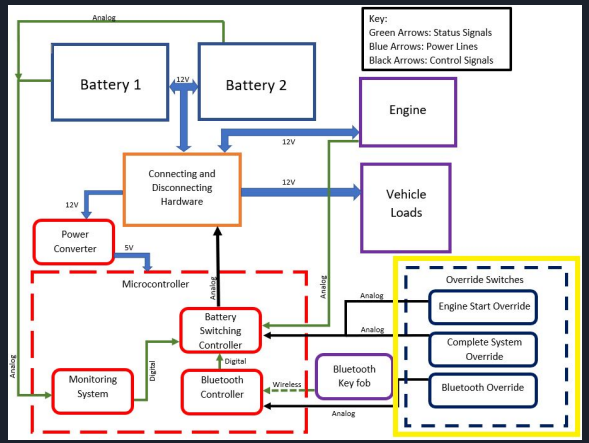
- 2 - relay module
- Switching voltage of 5V
- 20mA driver current





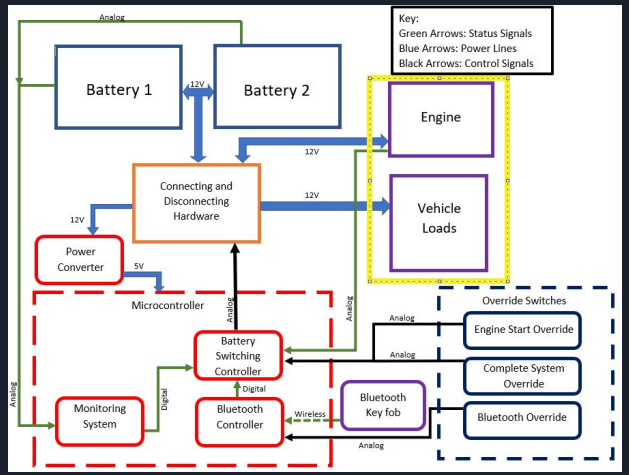
Switches

- 2 waterproof 1-position switches
- 1 waterproof 2-position switch



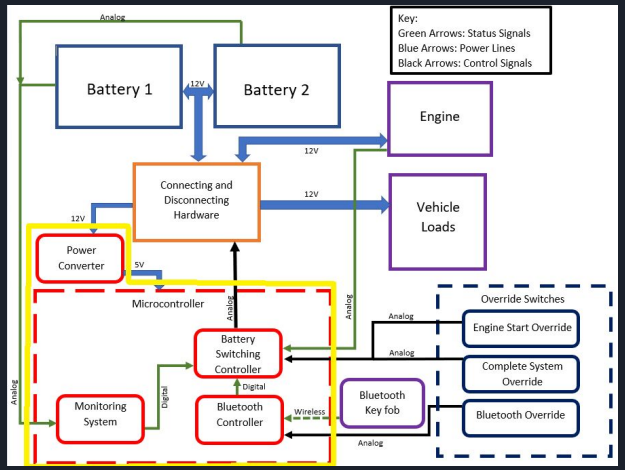
Engine/Vehicle Loads (Testbench)

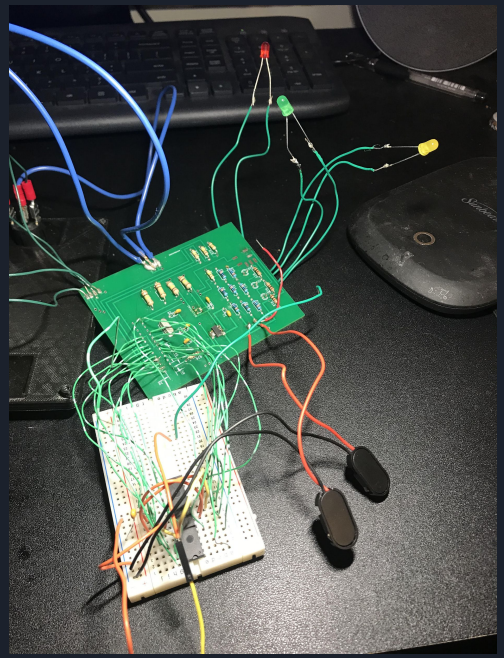
- 5 4-ohm resistors
- Multimeter for monitoring current
- Switch for either a 4 or 20-ohm line



PCB

- Standard LEDs for output
- 5V Voltage Regulator
- One inverter
- ATmega328p
- Standard resistors and capacitors





Design (Microprocessor Operation)

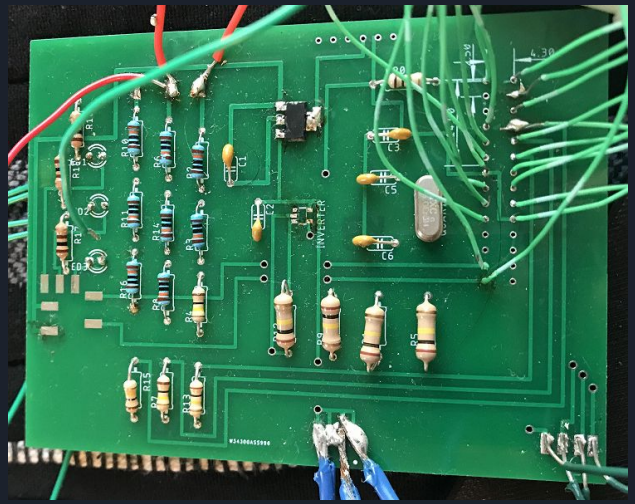
Condition:	Device On:	Batt 1 Connected:	Batt 2 Connected:
Normal operating conditions (Batt 1 Voltage > Voff, Engine is off, no override switches on):			
Connect both switch on			
Engine on signal received			
Power off			
Batt 1 Voltage < Voff (9V)			
Batt 1 Voltage < Voff (9V), Batt 2 Voltage < Vstart(11V)			
Engine override switch activated			20 sec
Post both off condition and engine on signal (Roles switched)			

```
int check_voltages(){
  float Reference_Voltage;
  float input_voltageref1;
  float input_voltageref2;
  float input_voltagerefengine;
  // take inputs from analog pins for batteries and engine on
  int analog_input_bat1 = analogRead(A0);
  int analog_input_bat2 = analogRead(A1);
  int analog_input_engineon = analogRead(A2);
  //Arduino converts analog to digital, so we must convert back to analog values
  //Change Reference Voltage to account for lower test battery voltages
```

```
Reference_Voltage = 5.0;
input_voltageref1 = (analog_input_bat1*Reference_Voltage)/1023.0;
input_voltageref2 = (analog_input_bat2*Reference_Voltage)/1023.0;
input_voltagerefengine = (analog_input_engineon*Reference_Voltage)/1023.0;
//Adjust for total battery voltage, relationship is vbatt= 4vref
input_voltage_bat1 = input_voltageref1*4.0;
input_voltage_bat2 = input_voltageref2*4.0;
engine_on_signal = input_voltagerefengine*4.0;
analog_input_enginestart = analogRead(A3);
analog_input_connectboth = analogRead(A4);
return 0;
```

Design (PCB)

- ATmega328p
- Confined to fit within 100mm x 100mm footprint
- Pads for voltage divider, switch controls, relays
- LED output to show battery status

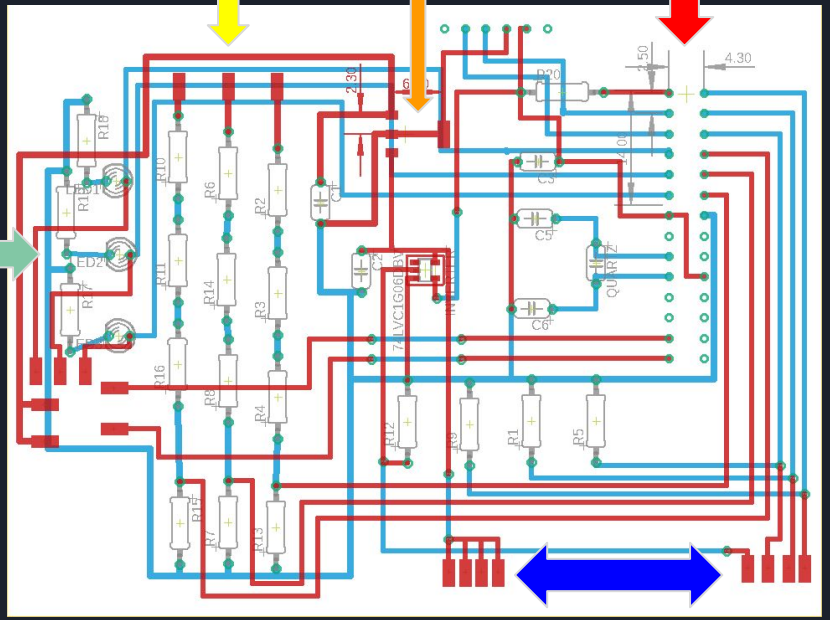


Voltage Dividers (2x battery, 1x engine start signal)

Microprocessor (ATMega328p)

5 Volt regulator

Status LEDs



Switch Pads

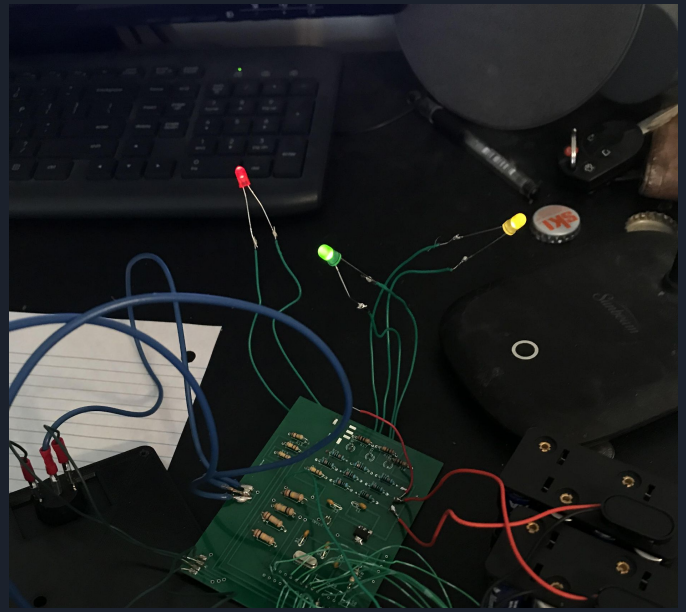
Design (Verifications)

- Could not test voltage regulator until Nov. 29th
  - Initial regulator did not produce 5V
  - Used Arduino for 5V initially, found new regulator

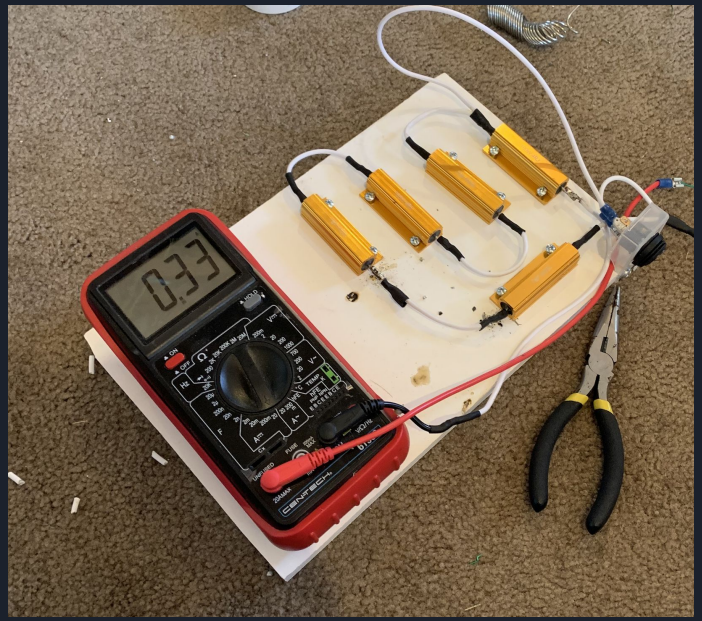
Requirement:	Functional:	Verification / Reasoning:
Regulator Works for input Voltages from 6-13V		We did not have access to lab to vary inputs with Power Supply
Microprocessor polls every 20-30 Sec		Can be shown in code: 20 sec pause after each update
Microcontroller makes status decisions and sends control signals		Device operation makes this obvious: See breakout table
Bluetooth switch works		Signal is being sent to PCB, non-functioning Bluetooth module limits usefulness
Relays work		Bad component prevented this; Our relay module does not switch
Both batteries connected switch works		Device operation makes this obvious: See breakout table
20 sec engine start override switch works		Device operation makes this obvious: See breakout table
Power off switch works		Device operation makes this obvious: See breakout table
Bluetooth module is operational, works in 20-30ft range		We were unable to make the Bluetooth module operational

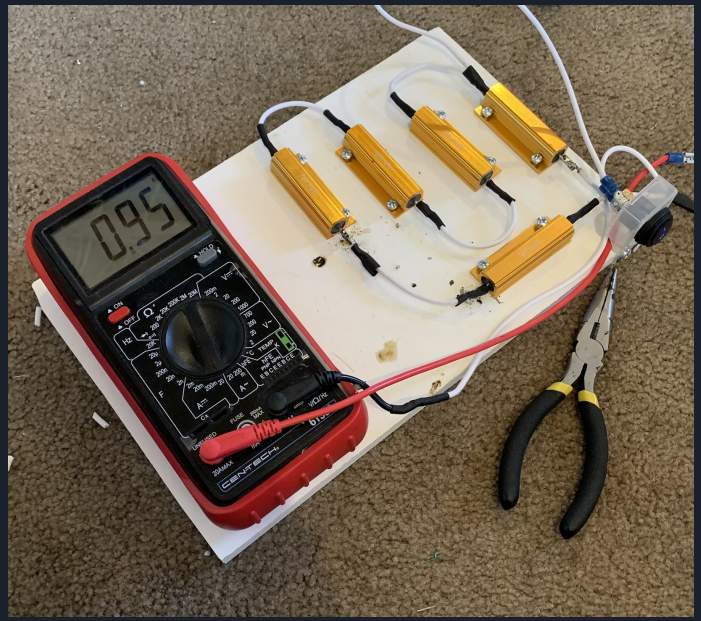
User Interface Verification

- The switches perform their intended functions and outputs to 3 LEDs
  - Red LED for battery 1, green for power to board, and yellow for battery 2
- Both batteries on shown in picture
- These signals would turn on a relay



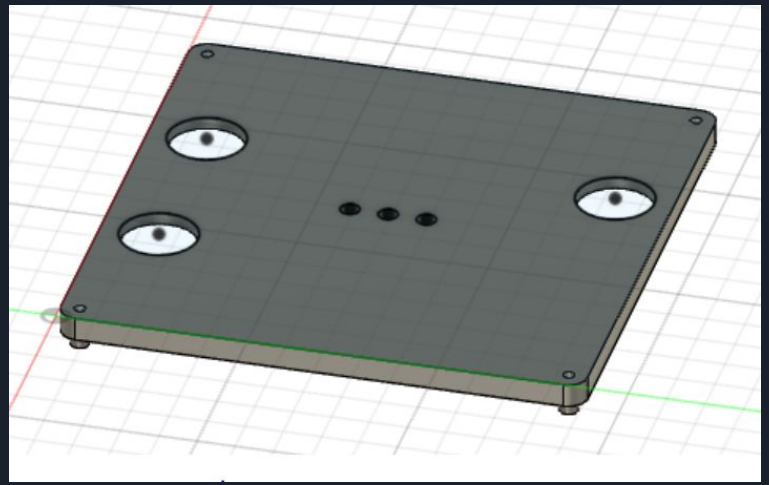
Test bench for Validation

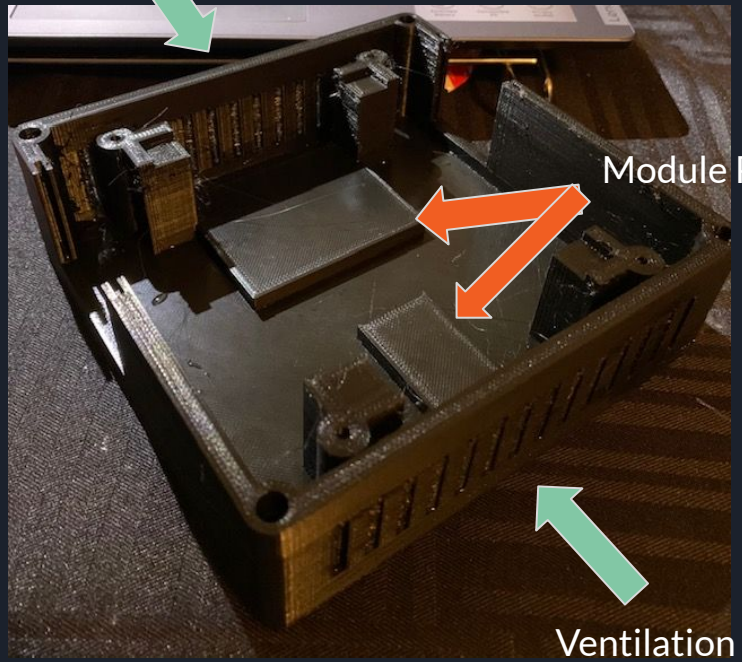




Design (Physical Case)

- Tried to keep dimensions similar to existing switches (Approx. 4” x 4”)
- Physical Dimensions: 5.43” x 5.43” x 1.92”





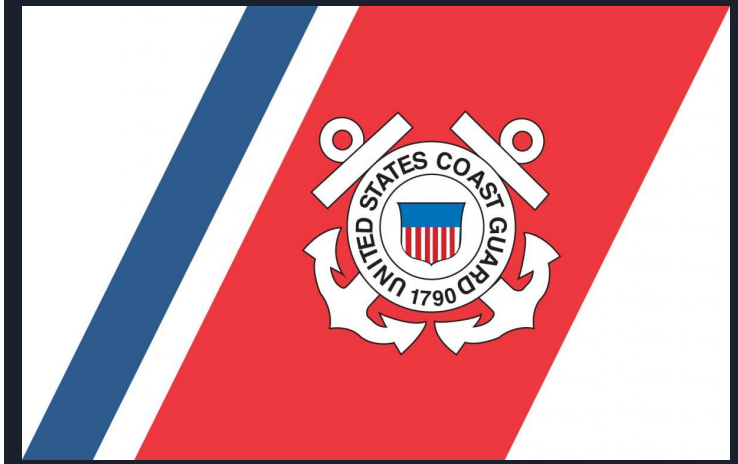
Conclusion (Needed Improvements and Additions)

- Working relay module
- Test the system using the designed testbench
- Fix mounting holes for ATmega
- Get a proper inverter (digital zero rather than high z output)
  - Truth table shown in picture
- Possibly get Bluetooth to work (add-on to system)
  - One with RSSI for distance gauging

INPUT A	OUTPUT Y
L	Hi-Z
H	L

Continuation (Future Plans for Full Scale Device)

Future Work (Water proofing, Scale Up, Regulatory Approval)



UL LISTED

ISO International Organization for Standardization