<u>Valencia Team</u>

IEEE Orlando Section

<u>Progress Report 2: February</u>

Project Addressing Climate Change:

Solar Powered Ventilation with Controlled Airflow for Parked Cars

Daniel Eisenbraun [IEEE # 98706125]

Ian Matheson [IEEE # 98579639]

Supervisor:

Dr. Masood Ejaz [IEEE # 92131157]

Department of Electrical & Computer Engineering Technology (ECET)

Division of Engineering, Computer Programming, & Technology

Valencia College

2022 - 2023

<u>Contact: Daniel Eisenbraun & Ian Matheson</u>
deisenbraun@mail.valenciacollege.edu
imatheson@valenciacollege.edu

Introduction:

This brief report covers important project developments in February. A review of spent budget and added costs will be given, followed by build progress.

Spent Budget and Added Costs:

The group incurred quite a few additional costs this month. A decision was made to swap the relays for transistors, since the relays drew too much current from the MCU and they were very bulky. Freewheeling diodes were added to the driver circuit. A lot of cabling was also ordered at the beginning of the month to facilitate proper construction. Terminal blocks were purchased to connect from the control module to the fan array cabling. A weatherproof box was also purchased to contain the control module. During testing, the Raspberry Pi Pico was damaged when an input pin contacted the positive connection of a DC power supply. These materials have been included in the updated project budget "Added Costs" section.

Progress:

Great progress was made this month. Highlights include the completion of the control module build and a test of the full system minus nozzles.

Control Module Build

Transistor Array

Following the change from the use of relays to the use of transistors, a transistor array was soldered up by Ian which would act as the means of controlling the output to the fans. An addition of Freewheeling Diodes was done to create a path for dissipation following the shutoff of the DC fans which were an inductive load.

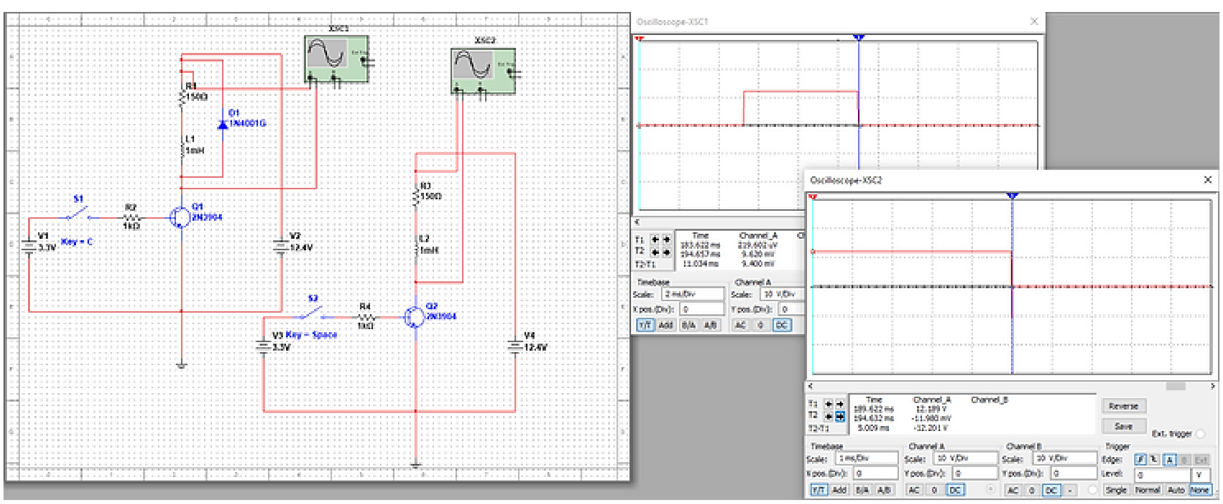


Figure 1 - Simulating transistor driver circuit with and without freewheeling diode

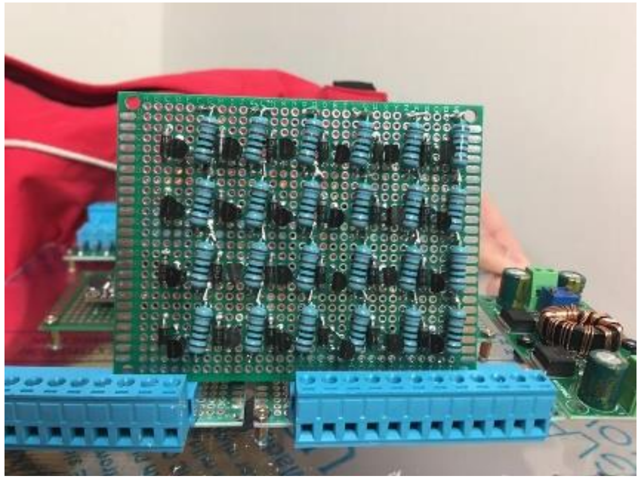


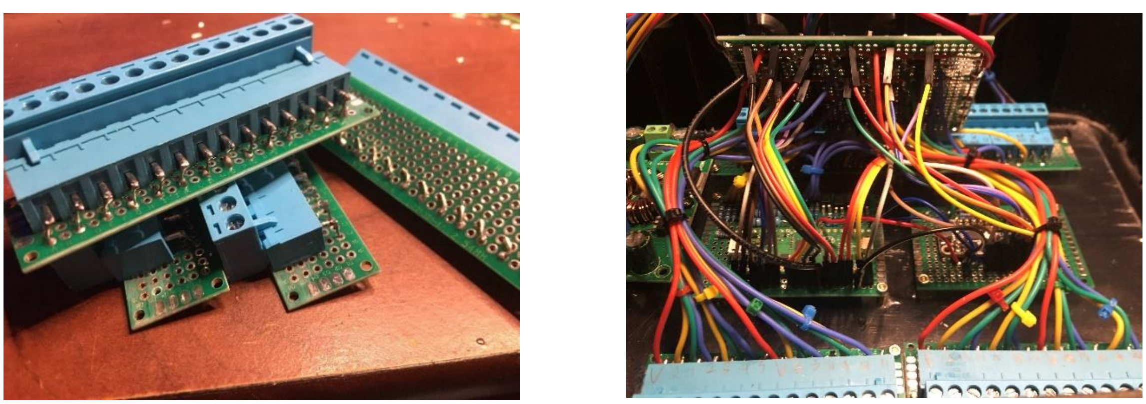
Figure 2 - Transistor array construction completed

IR Proximity Sensor Modification

Addition of current limiting resistors to each of the IR emitters was done to ensure that proper current was being fed to each emitter. The resistor at the cathode for each of the photodiodes was reduced from 1MOhms to 100kOhms to increase the input current to the ADC making it easier to read.

Wiring the Control Module

Terminal blocks were ordered to make connections easier between the control module and the fan arrays. Their pins were unfortunately too big for the group's PCBs, so the blocks were epoxied to PCBs and then jumpers were soldered to the board. While Ian soldered the transistor array, Daniel drilled some holes into a thin sheet of plexiglass and mounted all the control module components to it. Next, wires were soldered from the MCU to the transistor array. Wires were then run from the transistor array to its corresponding terminal blocks. Connections were made from the Raspberry Pi Pico to the PCF8575 GPIO expander. Finally, the connections for the sensors were wired from the MCU to the sensor terminal blocks



(a) (b)

Figure 3 - (a) Terminal Block PCBs. (b) Wiring the control module to terminal blocks

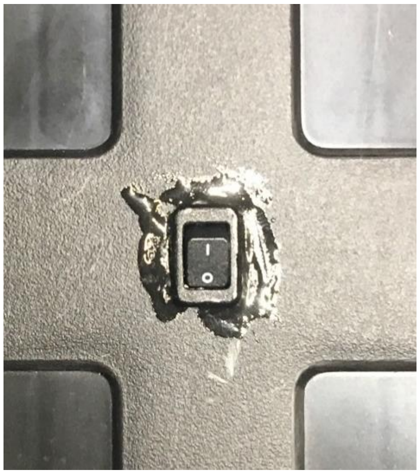
Connections were then made from the terminal blocks to the fan array cabling. All the connections for fans and sensors were run along multi-pin IP67 cable. This made it easy to connect and disconnect the fans and sensors from the control module.

Control Module Placed in Control Box

Once the control module was fully connected, it was placed in a box with cabling outlets. The PWM Charge Controller was mounted to the lid of the control box along with a SPDT switch. With this, the control module build was complete and ready for testing.

Figure 4 - Control module placed in control box

(a)



(b)

Figure 5 - (a) PWM Controller mounted to control box lid (b) SPDT switch installed in lid

Testing

Thermistor Testing

[Initial testing]() of the Thermistor Circuitry and Code was performed to confirm the their functionality. Each Thermistor responded to subsequent rises and falls in temperature when a heat source and ice cube were placed on the Thermistors. This creates a temperature difference between the windows which will be used to determine the airflow mode of the device.

IR Proximity Sensor Testing

Initial testing of the IR circuitry was performed and looked promising. Each of the sensors was tested against a window which can reflect the IR light. Once each was confirmed to work, we tested them with the code to ensure that if any of the IR sensors detected a window was present, that the fans would shut down as proper airflow cannot be attained if any windows are up.

New MCU

During initial testing for the Thermistors, IR sensors, and Battery Monitoring Circuitry, the Raspberry Pi Pico was damaged when contact with one of the GPIO pins of the MCU was made by the positive rail of a power supply. A new board was requested and arrived with connections to the board needing to be resoldered and testing re-performed to ensure no issues were present with the new board.

Battery Monitor Testing and Correction

Testing the battery monitor revealed that the ADC voltage was not only bouncy, but it was also inaccurate. The voltage read by the ADC is not accurate to the actual input voltage. The group saw an approximately 200mV drop in the ADC reading compared to the DMM-measured voltage. Furthermore, the drop appeared to increase as the input voltage increased. To gather data on this phenomenon, Ian set up a power supply. Starting at 12 V and increasing the voltage by 0.3 V until 15 V, Ian measured the voltage at the ADC with a DMM (Vin) and the average voltage read by the program (Vadc) over 1000 samples. Then, the % error was calculated between Vin and Vadc for each supply voltage level. A polynomic trendline was then produced for the % error using Google Sheets.

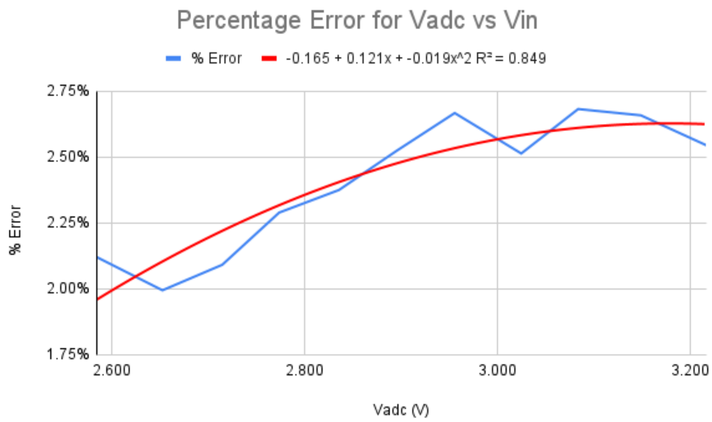
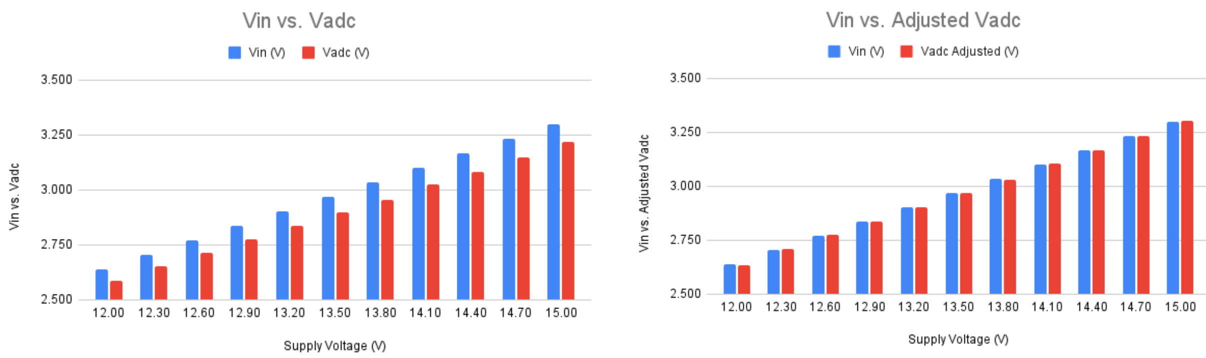


Figure 6 - Chart of the % Error between Vadc and Vin

The equation for the trendline was then used to adjust Vadc. The correction was implemented into the battery monitor program and tested. After adding a small offset, the voltage read by the program was accurate within about 10 mV, which was expected due to the bounciness of the ADC.



(a) (b)

Figure 7 - (a) Chart comparing Vin to Vadc (b) Chart comparing Vin to Vadc with error compensation.

```
def get_voltage(raw, reps):  # Converts digital ADC reading to voltage and takes average over
specified number of repetitions
    V = []
    for _ in range(reps):
        V.append(raw*3.3/65535)
        time.sleep(0.001)
    Vadc = sum(V)/reps
    Vin = Vadc + Vadc*(-0.1662 + 0.121*Vadc + -0.019*Vadc**2)  # Sum of Vadc and the product of ADC
voltage and trendline error %
    return Vin
```

Figure 8 - Implementation of error compensation in battery monitor program

Battery Monitor Hysteresis

To keep the fans from rapidly switching on and off near battery level thresholds, a hysteresis function was added to the program. The flowchart in *figure 10* shows how it works. The ADC voltage was bouncy by a maximum of about 20 mV, so the hysteresis function checks for a 50 mV difference between the current Vin and the voltage saved from the last threshold switch. The battery mode will only switch if that criterion is met. This is observed in the program printout below, which shows that the Battery Mode remains set to 1 even after Vin falls below the Mode 1 threshold of 2.647 V.

Table 1 - Battery mode program printout showing successful hysteresis.

Vin = 2.616 V	Vprev = 0.000 V	Vbat = 11.859 V	Battery Mode = 0	Battery Previous = 0
Vin = 2.628 V	Vprev = 0.000 V	Vbat = 11.913 V	Battery Mode = 0	Battery Previous = 0
Vin = 2.634 V	Vprev = 0.000 V	Vbat = 11.939 V	Battery Mode = 0	Battery Previous = 0
Vin = 2.638 V	Vprev = 0.000 V	Vbat = 11.957 V	Battery Mode = 0	Battery Previous = 0
Vin = 2.643 V	Vprev = 0.000 V	Vbat = 11.983 V	Battery Mode = 0	Battery Previous = 0
Vin = 2.647 V	Vprev = 2.647 V	Vbat = 12.001 V	Battery Mode = 1	Battery Previous = 1
Vin = 2.647 V	Vprev = 2.647 V	Vbat = 12.001 V	Battery Mode = 1	Battery Previous = 1
Vin = 2.647 V	Vprev = 2.647 V	Vbat = 12.001 V	Battery Mode = 1	Battery Previous = 1
Vin = 2.646 V	Vprev = 2.647 V	Vbat = 11.997 V	Battery Mode = 1	Battery Previous = 1
Vin = 2.644 V	Vprev = 2.647 V	Vbat = 11.986 V	Battery Mode = 1	Battery Previous = 1

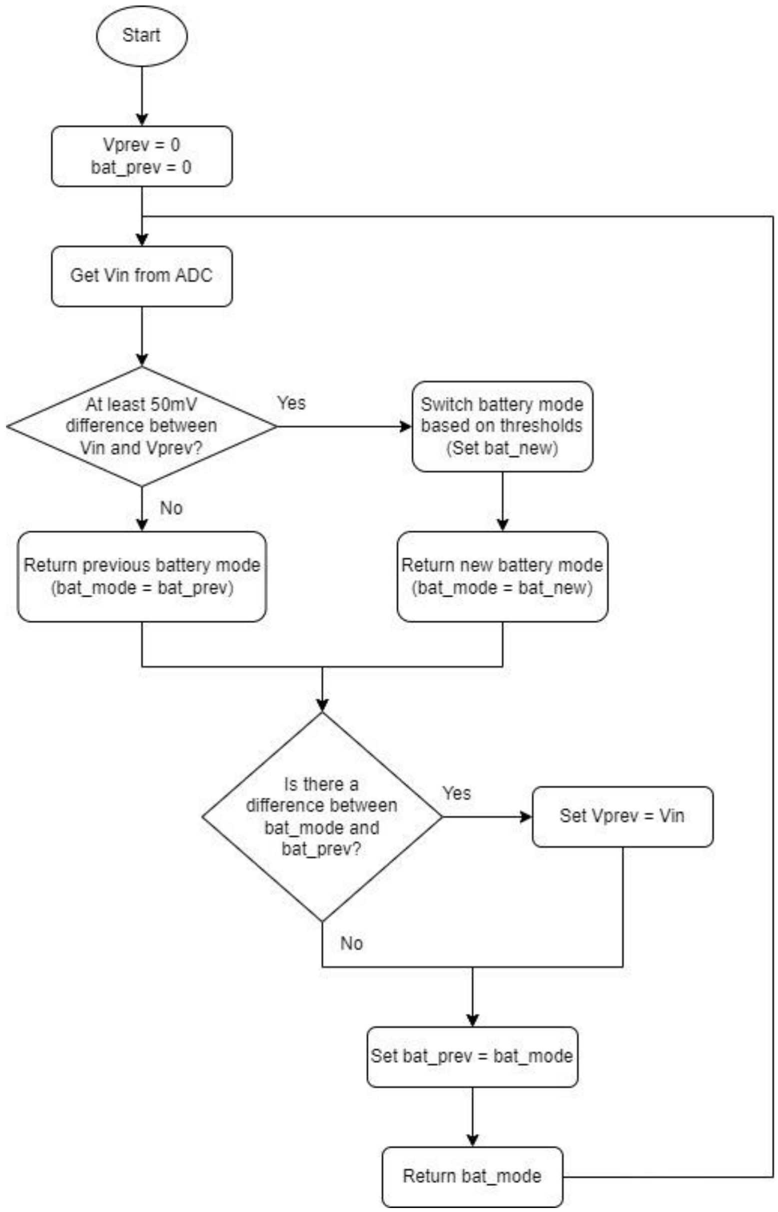
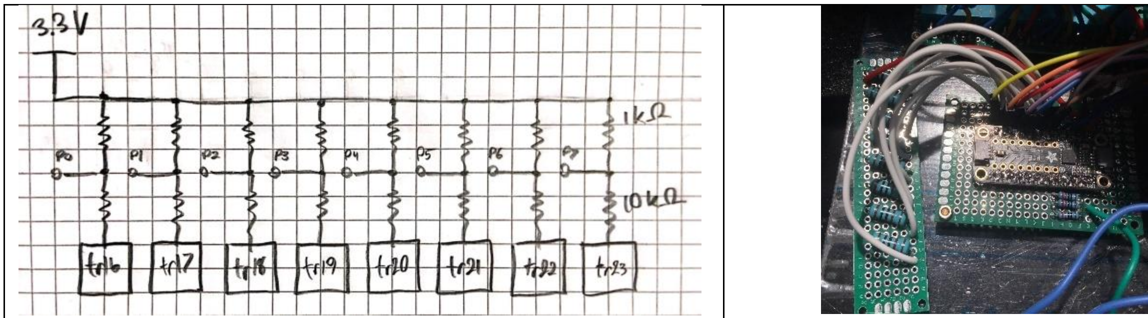


Figure 9 - Flowchart for battery mode hysteresis.

Fan Control Testing and Troubleshooting

Following confirmation of the IR sensors and Temperature sensors being functional in a controlled environment, the load was wired up to the transistor array. Manipulating the code, fans were activated to ensure all intake and exhaust fans attached to each of the vent visors was being properly controlled. During this initial test, it was determined that the signal from the PCF8575 was not sufficient to trigger the base of the transistors causing the intake and exhaust fans for windows 3 and 4 to not activate. The fix was to attach a resistor array to the output of the GPIO pins of the expander board. Following integration of the resistor array, all transistors were now capable of being triggered.



(a) (b)

Figure 10 - (a) Pullup Resistor schematic for PCF8575 (b) Connection of pullup resistors to PCF8575

Sensor Fan Control Tests

Once all fans were confirmed functional and controllable via the code, we ran a test running the fans off of the temperature data received by the Thermistors to check if air flow mode switching would occur when temperature differences were present between the windows. As well, the ability for the IR code to disable the load was tested. Both were confirmed to work as intended.

Full System Mock Tests

After all sensors were confirmed to be working and that control of the fans was possible through the code, a mock test was performed by attaching the vent visors to the test vehicle and integrating the Solar Panel, Car Battery, and PWM Charge Controller which would provide power to the system. Initial testing indicated that the Thermistor Circuitry and Fan Controls were nominal, however issues related to the Battery Level Voltage monitoring and IR Sensors were found. The Battery Level Monitoring was off by ~100mV from the expected value during the mock test and further correction in the code would be needed to offset the difference as the voltage going into the ADC was confirmed to match that of the battery voltage on a DMM. The IR sensors were found to have an exceptionally weak signal at the input to the ADC from the photodiodes. Initial assumptions were that the 3v3 pin of the Pico may not be providing sufficient current for the emitters to function properly. Additional troubleshooting would be performed to find the root cause of the issue.

On 2/27/2023, a secondary mock test was performed following corrections made to the Battery Level Monitoring code and a possible solution to the IR sensor issues experienced being implemented. The Battery Level Monitoring code was now successfully reading the proper voltage from the battery with only a 10mV to 20mV inaccuracy on occasion. The IR sensors were still experiencing issues, however. It was determined during the second mock test that the IR sensors could not be used in their current configuration as the ambient IR light provided by the sun was sufficient to saturate the photodiodes making it difficult to detect any of the reflected IR light from the emitters.

Figure 11 - Full system mock testing

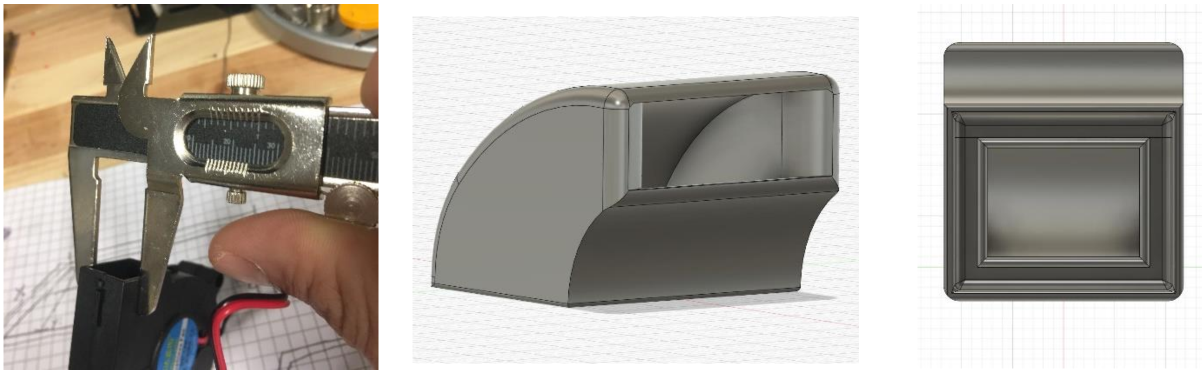
IR Issues

The IR issues experienced during the first mock test were initially believed to be a result of insufficient current draw from the 3v3 pin of the Pico, but this was confirmed to be a non-issue as the circuitry attached would only draw approximately 124mA at any one time while the 3v3 pin is capable of supplying up to 300mA safely. With that option off the list of potential issues, voltage supplied to each emitter and the cathodes of the photodiodes was confirmed to be sufficient. After these tests, it was determined that the proximity of the IR emitter to the photodiodes physically was causing the issues experienced. The 12 degree arc of the emitter was causing the photodiode adjacent to it to become saturated with IR light before any reflected IR light can be detected. A simple boundary layer was placed between the emitter and photodiode which resolved the issue in a controlled environment. It was also determined that the emitter for window one was experiencing issues related to the voltage difference between the ambient signal and IR signal which would need resolved.

When the secondary mock test was performed, the IR issues persisted despite the addition of a boundary between the emitter and photodiodes. It was determined that the ambient IR light provided by the sun was sufficient to saturate the photodiodes and confirmed via a controlled test where in a room within minimal sunlight, light would be let in while monitoring the voltage across the photodiodes. A solution to this issue will be researched by the team.

Nozzle Design

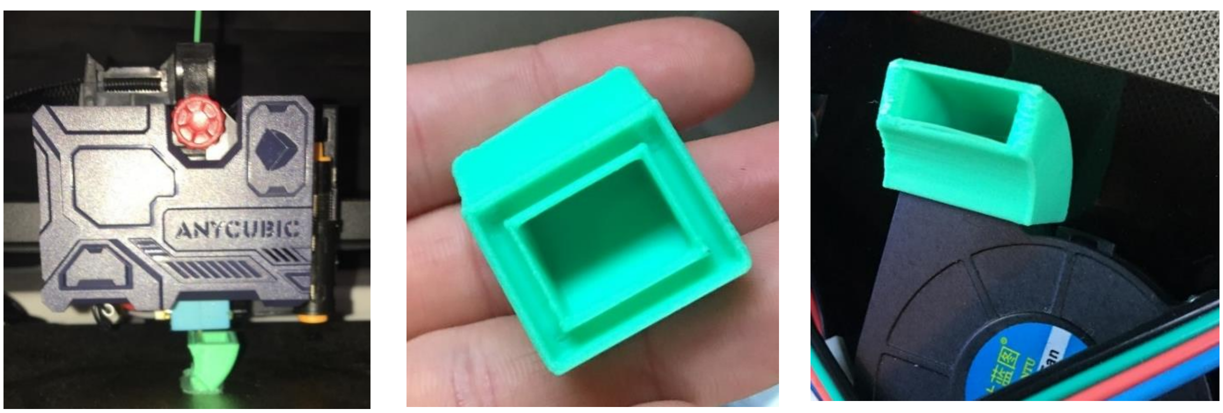
Ian designed a nozzle for the intake fans. He used Autodesk Fusion 360 to sketch a CAD model and then exported an STL file for 3D Printing.



(a) (b) (c)

Figure 12 - (a) Measuring the fans dimensions (b) CAD nozzle outlet (c) CAD nozzle inlet

Ian employed the help of his friend Duncan Kurtz for 3D printing. Duncan offered to print the prototype and even offered to print all 16 intake nozzles. These will be completed in the coming days, just in time for full system temperature testing.



(a) (b) (c)

Figure 13 - (a) 3D printer doing what it does (b) nozzle inlet (c) nozzle fitted on the fan

Summary

Overall this month, a great deal of progress toward meeting the goals set forth for the project was made. The construction of the control module and integration with the vent visors was made very easy by the use of IP65 and IP67 cabling. Overall costs associated with the project rose as material requirements caught up with our goals. We were able to test and troubleshoot all major components for the project with only the IR sensors still having issues at the end of the month that will be resolved in March. Despite these issues, we will be able to initiate airflow and temperature testing throughout the month of March.