JAMZ Climate Sensor: Prototype II

Deliverable G

GNG 1103

March 14th, 2021

Ogechi Ahunanya, Olivia Carnegie, Michaela Jones, Zach Léger, Julio Midence

Table of Contents:

Table of Contents:	**2**
Introduction:	**3**
Physical Prototype & Code:	**3**
CAD Prototype:	**6**
Test Plan II:	**7**
Conclusion:	**8**
Wrike Planning for Next Deliverable:	**9**

Introduction:

For this deliverable, we have started to test our components and we realized that some modifications and extra components must be added to our previous prototype. Based on our pitch presentation, we realized that JAMZ will be using a styrofoam box with a volume of $1ft^3$, so for our physical prototype we will need to change the box used into something resembling JAMZ's delivery box. Furthemore, JAMZ expressed their need for waterproof casing for the add-ons to increase durability which will be addressed. All of this has been put into consideration while improving upon and testing our prototype.

Physical Prototype & Code:

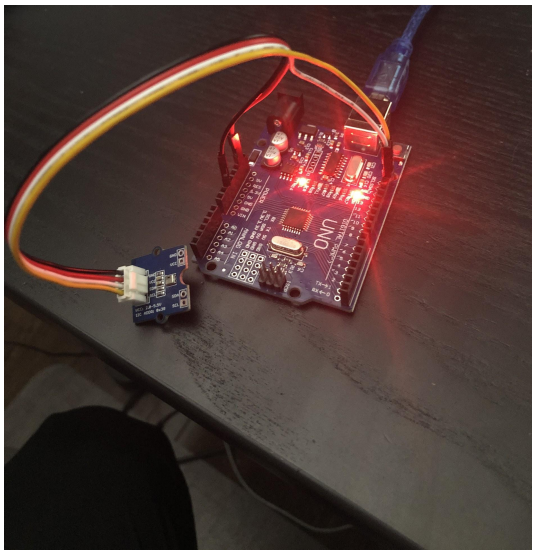

Figure 1. Trial 1 of Arduino and Sensor Testing

Figure 1 shows the wiring connection between the sensor (bottom left of the picture) and the Arduino. The wires connected properly and no errors were brought up regarding the connectivity of the wires.

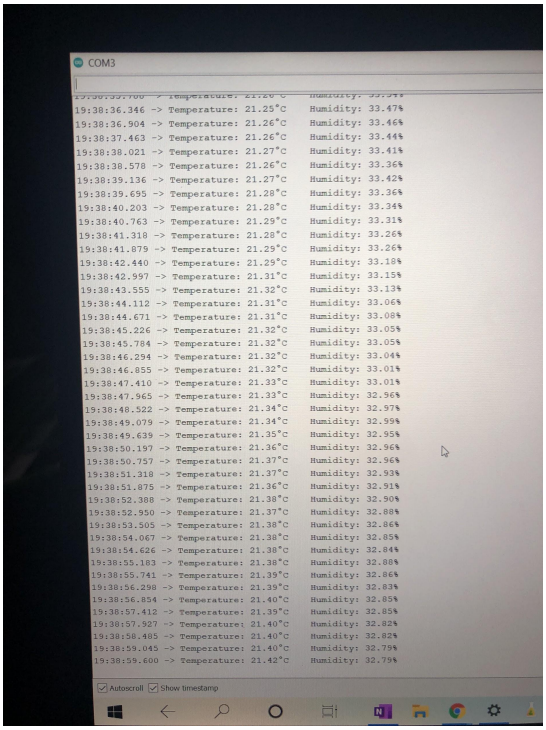

Figure 2. Sensor Results

Figure 2 shows accurate data retrieved from the AHT20 sensor and being printed off in the serial monitor. The values were compared with the temperature that the thermostat of the house was set to, and it matched very well.

Current Functioning and Tested Code:

```
#include <Adafruit_AHTX0.h>
Adafruit_AHTX0 AHT;
/*
 * Start with a Baud rate of 9600
 */
void setup()
{
  Serial.begin(9600);
  AHT.begin();
}
void loop()
{
  /*
   * This comment is taken from learn.adafruit.com:
   * sensors_event_t - This type is used to encapsulate a specific sensor reading, called an 'event',
   * and contains a data from the sensor from a specific moment in time.
   * Create a temperature and humidity value
   */
  sensors_event_t humValue, tempValue;
  /*
   * getEvent: Gets the temperature as a standard sensor event
   */
  AHT.getEvent(&humValue, &tempValue);
  /*
   * Print the temperature and humidity values in one row
   * Wait 500 milliseconds or the desired amount and print the next set of values in the next row
   */
  Serial.print("Temperature: ");
  Serial.print(tempValue.temperature);
  Serial.print("°C\t");
  Serial.print("Humidity: ");
  Serial.print(humValue.relative_humidity);
  Serial.println("%");
  delay(500);
}
```

CAD Prototype:

Going forward in our CAD prototype analysis, as well as comments from JAMZ, we have determined that we must consider waterproofing our designs. There are several ways to go about waterproofing… One, being altering our casing design in order to properly seal the sensor. Second, being altering the materials we want to use for the casing. We are confident in our first prototype casing's structure, so we will consider the second option in this case. In order to determine a new material we are working to find a material that matches our past standards for toughness and strength while now also being waterproof.

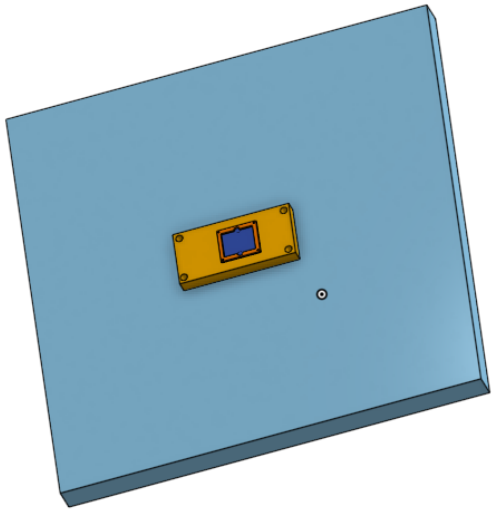

Figure 3. Sensor Casing on Drone Lid

Figure 3 demonstrates the placement of the Sensor and Casing on the interior of the drone lid. The Sensor and casing will be placed in the center of the lid and inside the box, but the sensor will not touch the box directly in order to provide optimal results.

Test Plan II:

Table 1: Climate Sensor Test Plan

Test ID	Test Objective (Why)	Description of Prototype used and of Basic Test Method (What)	Description of Results to be Recorded and how these results will be used (How)	Estimated Test duration and planned start date (When)	Analysis
1 (Code)	To ensure proper and accurate function of code with sensor.	Numerical and physical prototype. Test speed of data and response time reliability with the climate sensor	Plug in sensor and change the temperature and humidity of the surrounding environment $0 (use materials we already have)	New code already tested	The untested code did not want to print the temp and hum values when first tested. We noticed the code was missing an AHT20 sensor related function. The sensor reads the hum value first then the temp value. This was very important in determining the order the values would be passed onto our variables. Once all of this was determined, the serial monitor was printing accurate and applicable data.
2 (Wire connection)	To ensure nothing will come loose or disconnect during a delivery.	Physical prototype. Test the physical connection of the wires with the updated attachment system. Test for any kind of flight	Plug wires into all other necessary components using new additions and move/shake the prototype as if to simulate a rough flight.	Proper testing will be conducted once materials are available. Prototype testing with available	With this improved system, the wires stay connected to the sensor and to the Arduino with very little chance of them falling out. With the official materials

		path and disruptions that may occur.		materials ASAP.	this small chance will become even smaller.
3 (Weather proofing)	To ensure no weather elements can enter the box at any connection points.	Physical prototype. Test the ability to withstand different weather, such as rain, from entering the delivery box or the drone.	Spray water from a bottle onto the prototype to see if any water enters. For the prototype, use rubber blades to cover any openings and surround any components within those openings (other materials might need to be used for sooner testing).	Proper testing will be conducted once materials are available. Prototype testing ASAP.	No water is able to penetrate into the box, however, the waterproofing for the actual sensor will need more work. This isn't a big issue at the moment but adding some form of waterproofing to the sensor will ensure that, if a crash where the box breaks open were to occur, the sensor would be protected from the elements.

Conclusion:

In conclusion, this week our group was able to gather further information from JAMZ and our personal testing analysis to establish that our prototypes must be waterproof going forward. In regards to our physical prototype and the code, this week we were able to finally test the sensor. In order to generate a sufficient amount of useful feedback on waterproofing, we've planned to conduct further tests and research to ensure suitable waterproof casing is used. In order to do so, we created specific test plans to touch on the "why", "what", "how" and "when".

Wrike Planning for Next Deliverable: