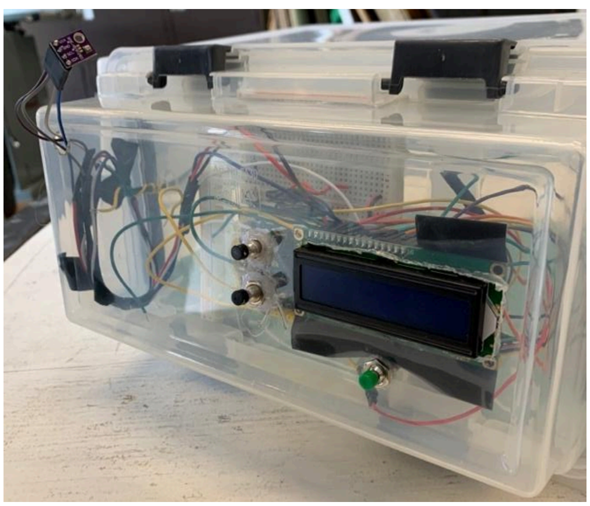
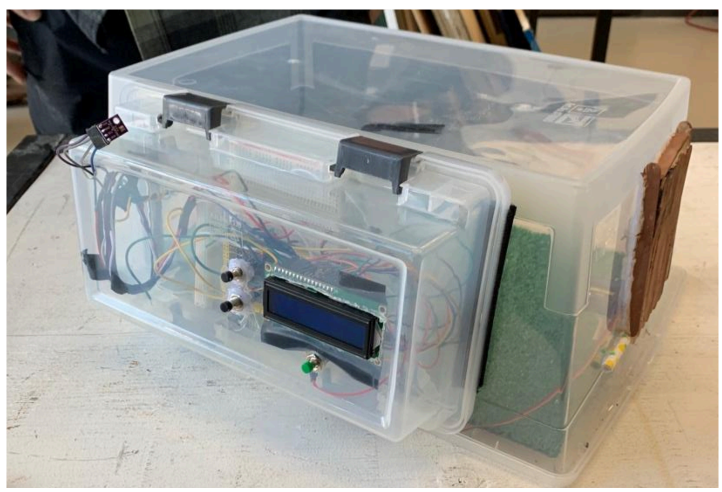
Western University Faculty of Engineering
Studio Section Yue Zhou
Team Identifier: T14-Hoop
Team Members: William Destounis, Bashshar Atif, Anees Shoaib, Aidan Sarnese, Mathew Garfat

Project Title: TAHCS

The Temperature and Humidity Control System (TAHCS) designed for Urban Roots' Hoop Houses





1 Need / Challenge

A process to automate the temperature and humidity of an indoor hoop house owned and operated by Urban Roots's, a non-profit organization, based in London, Ontario. This system must be easy to navigate for volunteers, increasing efficiency and decreasing volunteer's time commitment. This will decrease the amount of effort needed from each volunteer and will result in more consistent production and an increased incentive to work.

Blue – Need

Green – Who

Yellow – Insight

2 Final Design Documentation

Physical Components of Prototype:

Quantity	Part	Unit price	Total Price	Link	Functionality
1	Servo motor	$19.99	$19.99	link	This was used originally to show the working hatch
1	Breadboard	$13.99	$13.99	link	This was used to connect components to allow for easier installation
1	Wire	$23.88	$23.88	link	This was used to connect components
1	BMP 280	$10.46	$10.46	link	This was used to sense the internal temperature of the hoop house
1	Buttons	$9.88	$9.88	link	These were used to interface with the control system
1	LCD	$13.96	$13.96	link	The LCD was used to show information
1	Arduino	$25.99	$25.99	link	An Arduino mega was used as the control system
Total Price		$118.15			

Figure 1: Material List.

The above parts list **(Figure 1)** was assembled using the wiring diagram below **(Figure 2)**. To do this, the wires had to be soldered to the buttons and to the BMP 280 sensor. The LCD was also wired using female to male connectors to ensure that it could be attached easily to the control box frame. After having assembled the systems with wires that were longer than necessary, the wires were shortened to make the traces easier to manage. This also made it harder for wires to come undone during assembly. Our system also includes some miscellaneous building materials such as; popsicle sticks, straws, and plastic boxes.

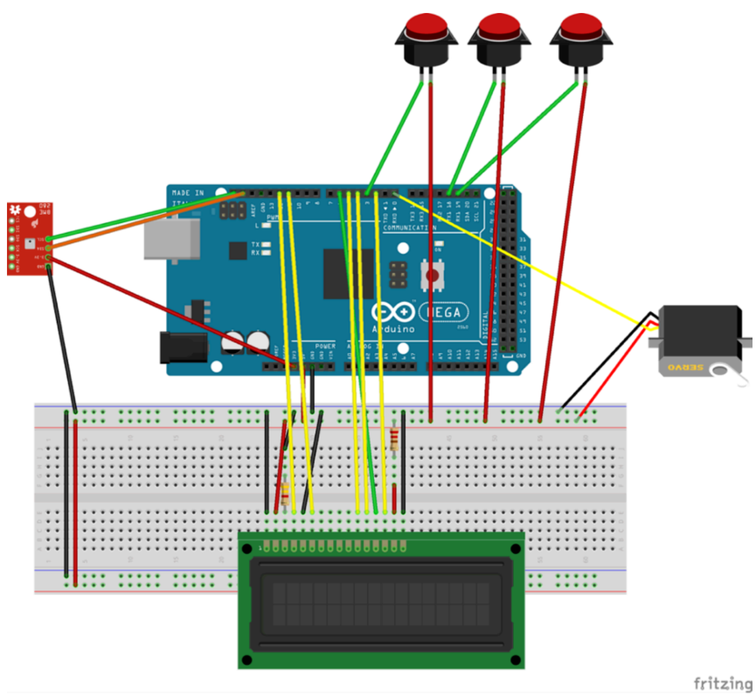


Figure 2: Wiring Diagram

After the wiring assembly was finished the breadboard and Arduino were taped on the inside of the control box on the opposite side of the user interface. Then holes were cut into the top of the control box, allowing for the LCD and three buttons to stick out in the configuration shown below **(Figure 3)**. The wires from the LCD and buttons were then shortened to again make runs easier to trace and limit failure due to wires falling out.

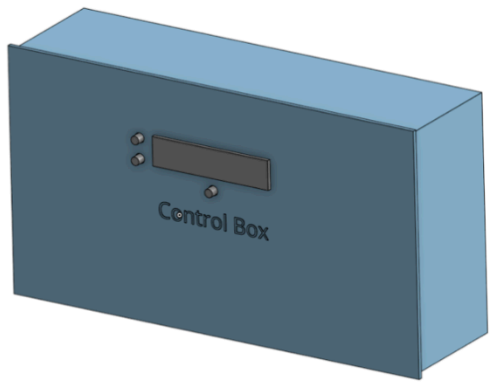


Figure 3: CAD Control Box

Another plastic box was used create a miniature hoop house **(Figure 4**). This would help show the control box's functionality. The miniature hoop house was cut on the side, allowing for a popsicle stick door to be placed over it. This door was the connected to the servo motor. Due to the motor connecting with the door on an angle, a rubber band was used to connect these two together. This allows the motor to move with a free range of motion **(Figure 5**).

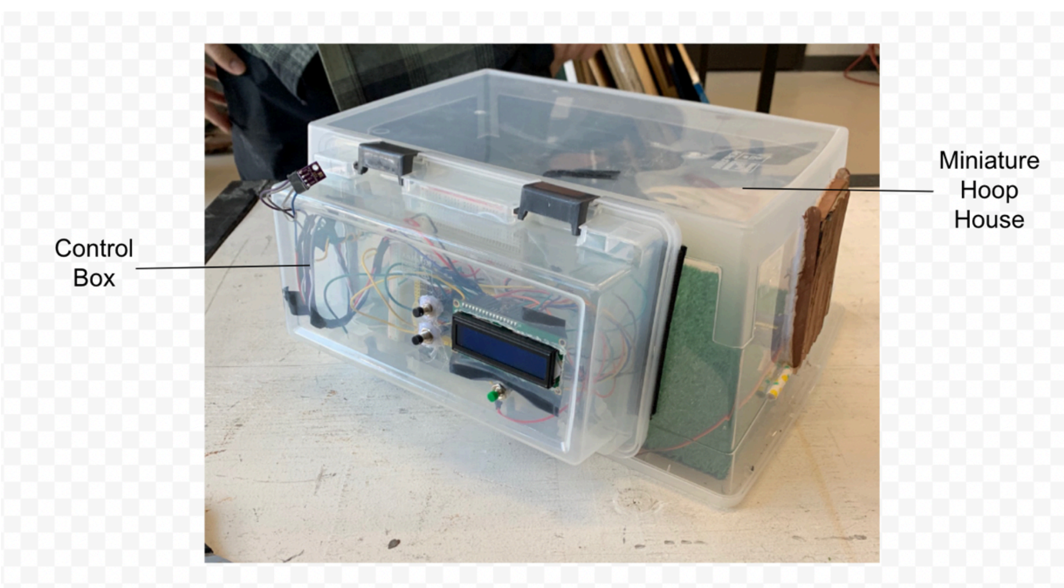


Figure 4: Control Box and Miniature Hoop House

Figure 5: Miniature Hoop House Door

The below code snippet **(Figure 6)** was used in the program to allow for the system to pause its current cycle and increment values when the buttons are pressed. The system would run the methods up, down, or cycle depending on which button changes the pin to high.

```
pinMode(19, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(19), up, HIGH);
pinMode(18, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(18), down, HIGH);
pinMode(3, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(3), cycle, HIGH);
```

Figure 6: Button code

```
void loop() {

  cTemp = bmp.readTemperature();
//  Serial.println(menu);
//  Serial.println(hTemp);
//  Serial.println(lTemp);
  if(cTemp > hTemp) {
    if(!angle == 30) {
      angle = 30;
      myservo.write(angle);
    }
  }else if(cTemp < lTemp) {
    if(!angle == 30) {
      angle = 30;
      myservo.write(angle);
```

The aside code segment **(Figure7)** shows what the system does every cycle, which happens about 60 times per second. The program begins by reading the current temperature value and saving it. It then determines whether the temperature in the hoop house is too hot, too cold, or within the range. It then either opens the hatch if the temperature is out of range or it closes the hatch if the temperature is withing the range. It also only tells the motor to preform an action if that action has not already been preformed. The code then checks to see if the current temperature is being displayed currently and if it is it updates its value on the LCD. The main cycle loop was kept short to ensure that the system does not slow down.

Figure 7: Main Loop

```
void up() {
  switch(menu) {
    case 0:
      hTemp++;
      break;
    case 1:
      lTemp++;
      break;
  }
  lcd.setCursor(0,0);
    lcd.print(hTemp);
    lcd.setCursor(7,0);
    lcd.print("High Temp");
    lcd.setCursor(0,1);
    lcd.print(lTemp);
    lcd.setCursor(7,1);
    lcd.print("Low Temp");
}
```

The final code snippet shown **(Figure8)** is an example of one of the functions called when the buttons are pressed. It uses a switch statement to keep the runtime of it short. It determines which value is being edited and then modifies that value. It then updates the LCD with the new high and low temperature values. This portion of the code had to be kept as short as possible to ensure the system is not slowed down.

Figure 8: Button code

The user interface (UI) of TAHCS is designed so that new volunteers do not have to receive formal training or read manuals to use it. The focus when designing the UI was for it to be simple and easy to use. As seen in appendix 7.3, the first menu the user is taken to upon start-up is the temperature adjusting screen. This screen simply requires one click on the up/down buttons to adjust the temperature. To switch within the “High Temp” and “Low Temp”, the user just needs to click the green button once. To switch menus to the current temperature menu, the user needs to click the green button three times. To switch back into the temperature adjustment menu, the user can again, click the green button three times.

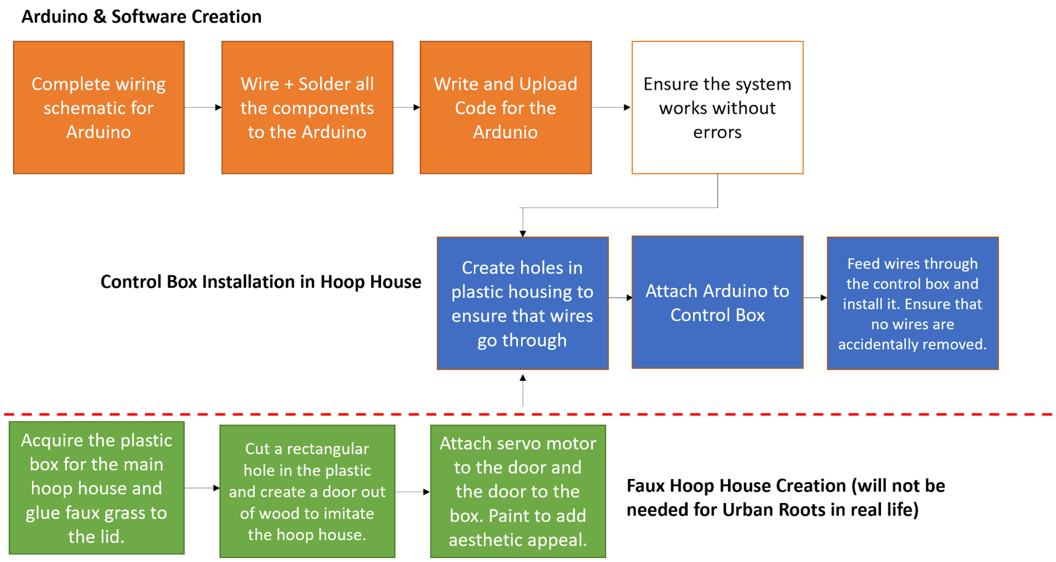


Figure 9: Flow Chart

The production process of our system is what makes it more practical to install compared to more complicated designs. The production process for TAHCS is two-fold, one is concerned with the creation of software and hardware relevant to the Arduino, and the other is concerning the creation of the faux hoop house and door. In real life the creation of the green boxed steps in Figure 9 will not be necessary however the steps with blue and orange steps will are necessary and laid out above. The explanation of the steps concerned with production are present in the flowchart above **(Figure 9)**.

3 Testing and Validation

3.1 Objectives (and Functional Capabilities)

#	Objective	Assessment	Evidence
1	Easy communication between the volunteers and the temperature and humidity	Met	Our system includes a control panel with a liquid crystal display (LCD) and a variety of buttons. These buttons allow the user to input the acceptable temperature and humidity range. As seen in ***Figure9,***

#	Objective	Assessment	Evidence
	control system of the hoop house.		we used an LCD to display the temperature range from high to low. Two black buttons were used to adjust the numerical values of the temperature and humidity. A green menu button is used to switch between the temperature and humidity settings. This display is very efficient and easy to understand.
2	Ensuring that the position of the door of the hoop house corresponds to the temperature sensor.	Met	The hoop house door is connected to a servo motor, which is connected to the Arduino. The temperature and humidity sensor communicates with the Arduino and depending on the input it will open or close the door. As seen in ***Figure11,*** when the temperature sensor is held and detects heat, the hoop house door opens via a servo motor. This will let out excess heat and humidity from the hoop house. Alternatively, when the hand is removed, as seen in ***Figure12,*** the sensor detects a normal room temperature and shuts itself to maintain that temperature.
3	Accuracy of measuring and recording the temperature inside the hoop house.	Met	The current temperature inside the hoop house is displayed on the LCD, informing volunteers when the door will be opening or closing. As seen in ***Figure 9*** and **Figure10,** the sensor is connected to the LCD and the bread board in order to record the temperature of the hoop and report it. The white wire connects to a computer, where it will show a record of the previously recorded temperature and humidity's.

#	Objective	Assessment	Evidence
4	The device should be easy to setup for the farmer.	Met	Our design is made from the strategic connection of simple components. Urban Roots can easily connect each external component (temperature and humidity sensor and the servo motor) to the control panel. As seen in ***Figure13,*** our design supports a simple servo motor connected to the hoop house door. This servo motor is connected to our control panel, that consists of an LCD and a coded Arduino. Urban Roots would just be tasked with connecting all these components, which is a simple task when following step by step instructions.
5	The device should be user connected.	Met	Our system can receive user input, making it much easier for volunteers to control the temperature and humidity in the hoop house. As seen in ***Figure 9,*** Our device hosts a conventional 3 button mechanism which allows the user to set the desired temperature and humidity level. It is important for our system to accept user input as the desired temperature and humidity may change depending on what plants they are currently growing.
6	Making sure that the system automates to the corresponding temperature	Met	Our system detects when the temperature or humidity goes higher or lower than the inputted desired range. It then uses this input to decide if the hoop door should open or close. As shown in ***Figure9, Figure11 and Figure12***, our device can successfully complete this task. When the sensor detects a higher temperature the servo motor responds and when it lowers it retracts. The inputted acceptable temperature and humidity range is essential to this system, ensuring that these values are controlled.

Figure9: Full lcd setup with control buttons.

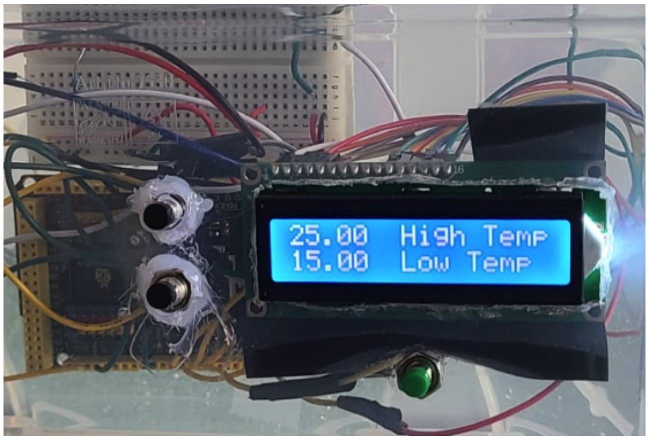


Figure10: System sensor (Outside for testing Purposes).

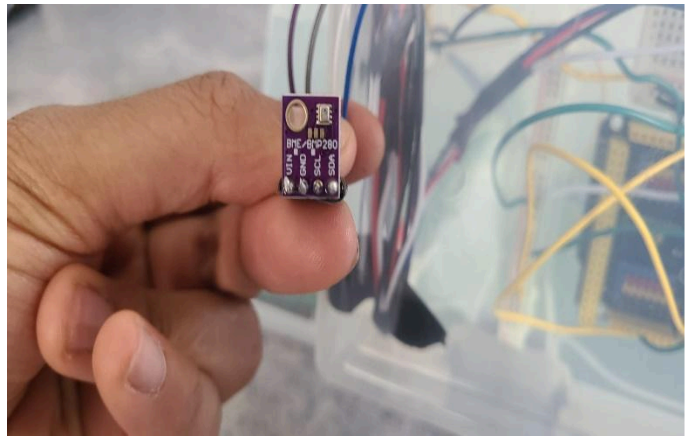


Figure11: System detecting heat.

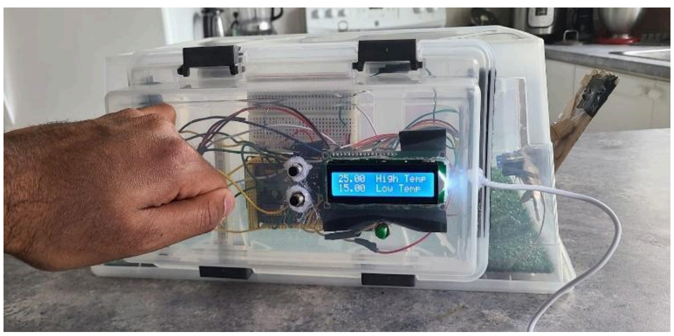


Figure 12: System at room temperature.

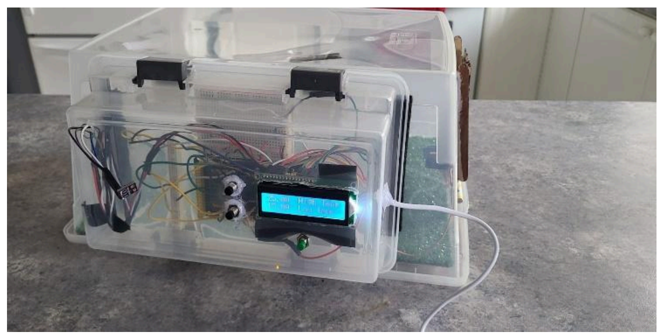
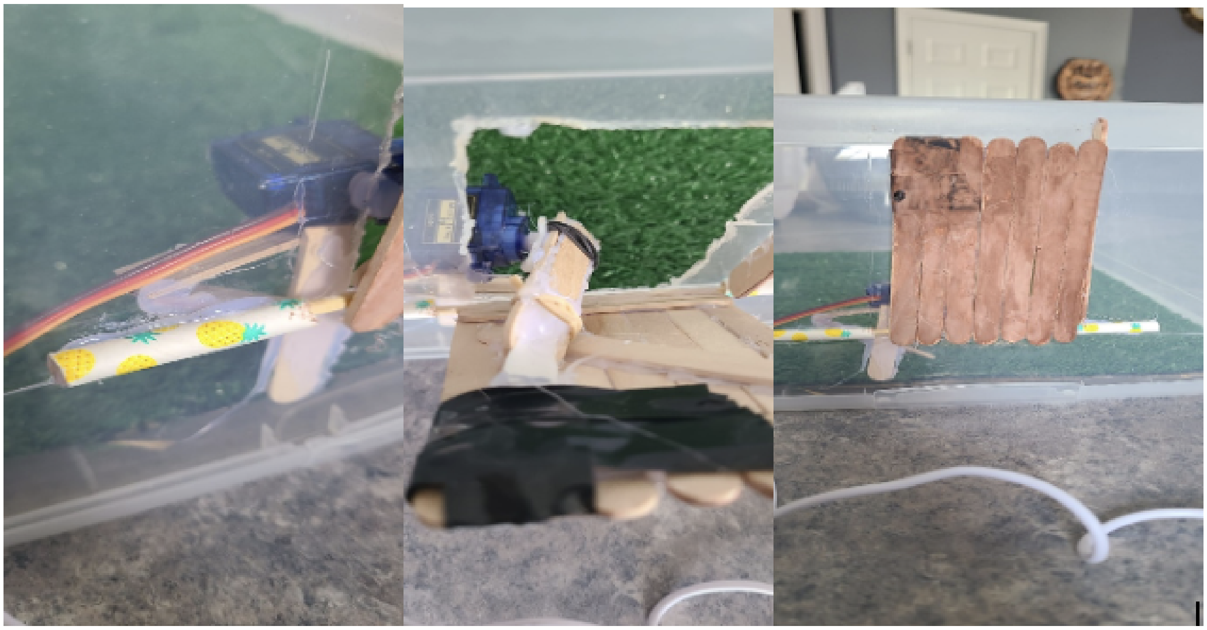


Figure 13: Simple hinge mechanism connected to the motor and the door.



3.2 Testing and Validation: Constraints

#	Constraint	Assessment	Evidence
1	Must be Cost Effective	Met	Our design uses cheap materials, relative to industry alternatives. This is because we are purchasing each component separately and connecting it ourselves. This is very important as Urban Roots is a non-profit organization with a small budget. For example, our entire system will cost approximately $120. This is significantly cheaper than industry alternatives. Some alternative products can range up to $740 in price **(Figure14)**.
2	Determine When to Actuate Motor	Met	Our system can determine when to actuate the motor, controlling the hoop house door. This is essential as it is how our system regulates the hoop's temperature and humidity. This is done through the coding of the Arduino and the user inputted range of acceptable values. When the sensor detects a higher temperature/humidity, it will open the door (**Figure 11).** When the sensor detects a temperature/humidity within the accepted limits, it will shut the door to maintain the conditions of the hoop house (**Figure 12).**
3	Sense Internal Humidity and Temperature	Met	After coding and wiring our Arduino, the LCD display was able to show the current temperature and humidity within a room **(Figure9).** This shows how our sensor was functional and served its purpose in our design. This is very important as an accurate

#	Constraint	Assessment	Evidence
			sensor reading is essential for our system to operate correctly.
4	Weather Resistant	Met	The material used in our prototype will be able to withstand a strong and changing climate. This is important as our system must be practical and have a long lifespan. The control box (**Figure 15)** encloses all the delicate wiring, protecting it from the elements.
5	Actuate a Motor	Met	Our group was able to fulfill this constraint. This is an essential task to complete as it is the direct method of controlling the temperature and humidity. We used a relay to control motor function, allowing us to open and close the hoop door. This motor served as a replacement to a human volunteer manually opening and closing the window **(Figure 16).**

Figure14 : Industry Alternative System

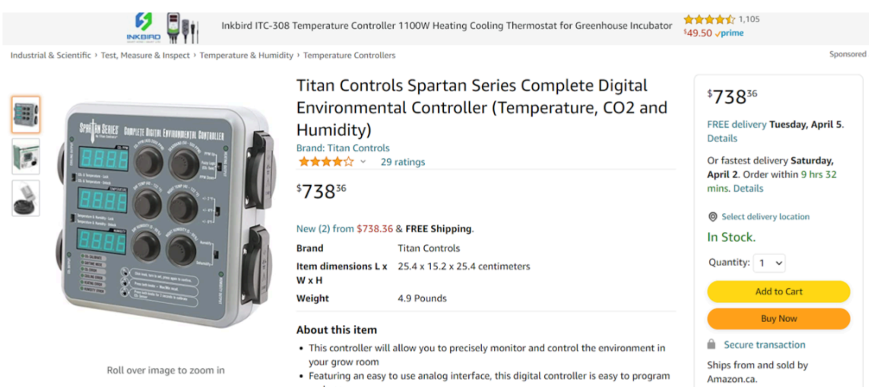


Figure15 : Prototype Control box

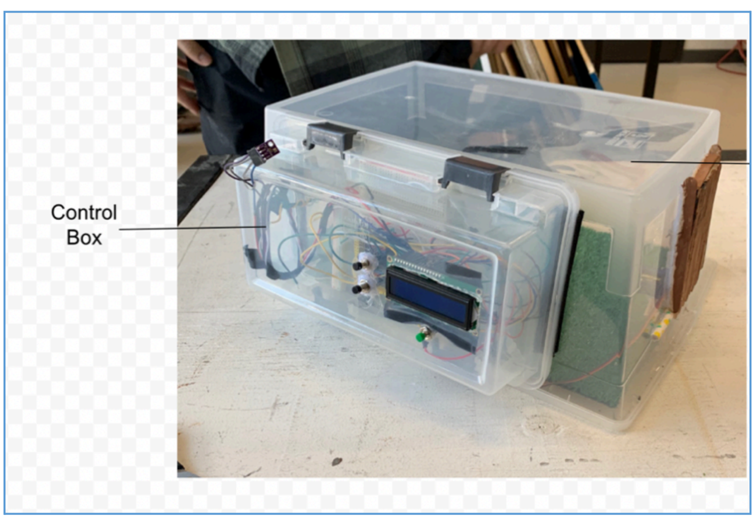
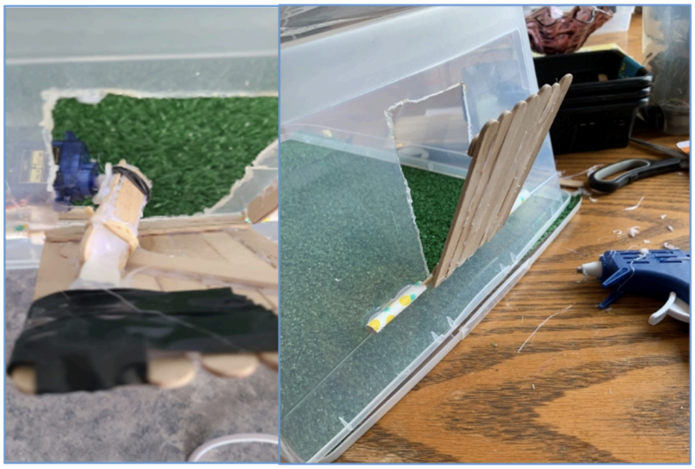


Figure16 : Motor Connected to Hoop Door



4 Comparison

4.1 Comparison 1

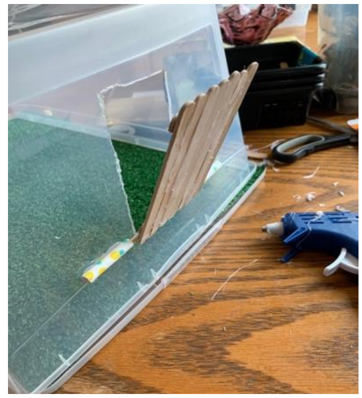
Arduino system and status quo / Arduino system is automated, and status quo requires manual labour/ Link (as applies): https://www.amazon.ca

Practicality Comparison	Our final solution to Urban Roots' temperature and humidity control problem is much more practical than their current status quo solution. Urban Roots is currently managing the temperature and humidity of their hoop house manually. A volunteer is opening and closes a window when humidity and temperature values are high. This is a poor, inefficient system, that wastes lots of volunteer time. Our solution automates this process using an Arduino and numerous peripherals. It uses a temperature and humidity sensor, connected to an Arduino, to control a motor. This motor will open and close the window depending on the inputted temperature and humidity limits. The required materials for this solution are very cheap relative to alternatives. This saves lots of time and resources for Urban Roots.
Comparison of Strengths	A strength of our design is that our system will save Urban Roots a considerable amount of time. Volunteers can now use their talents in other areas, rather than waste them completing a time-consuming task. This incentivizes volunteering at Urban Roots as there will be more time spent working on rewarding tasks. A strength of their current system is that it is reliable. A volunteer opening and closing the door in some cases is more reliable than an electronics system which may be prone to errors or malfunction.
Comparison of Weaknesses	A weakness of our proposed solution is that there may be malfunctions. If any damage were to occur to our system it may cause a system malfunction, preventing it from working. Urban Roots's current system's weakness is that it is very inaccurate. For any changes to be made to the internal temperature and humidity of the hoop house, they need a volunteer present, to open and close the window. If the temperature changes rapidly, internal conditions will not be fixed until someone notices this problem and manually fixes it. This may cause serious and irreversible damage to their crops.

Other Comparisons	

Door that must be opened and closed

The prototype version of the hoop house and the door

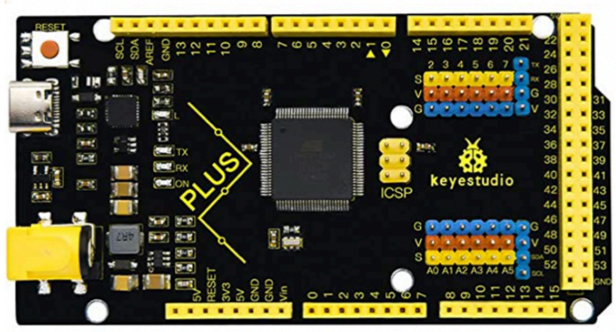


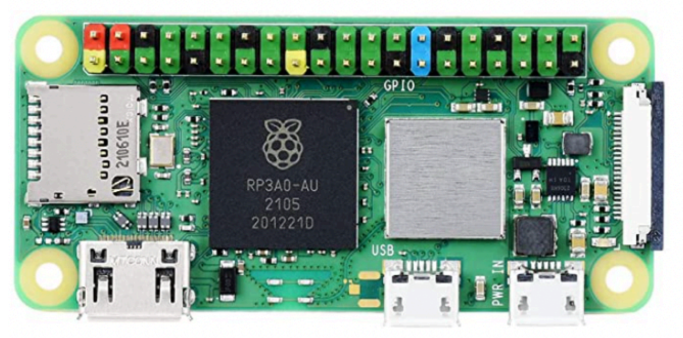
4.2 Comparison 2

Arduino system and Raspberry-Pi System/ Raspberry-Pi is an alternative microcontroller / Link (as applies): https://www.amazon.ca

Practicality Comparison	Both Arduino and Raspberry-Pi microcontrollers can be used to automate the temperature and humidity control of Urban Roots's hoop house. Although, the price of an Arduino is significantly cheaper than a Raspberry-Pi and provides the necessary capabilities to complete this. This makes the Arduino system more practical for our final design.
Comparison of Strengths	The Arduino system is very cheap relative to other system alternatives. This is a major strength as Urban Roots is a non-profit with a low budget. The Raspberry Pi however provides more and greater capabilities than the Arduino such as stronger computing power, easier compatibility with Wi-Fi, Bluetooth, and telecommunication networks. The Raspberry Pi can also be coupled with a touch screen display with detailed information and a better user experience. All of these features are extremely attractive however the costs are quite expensive with Raspberry Pi screen being $103, and the microcontroller starting at $75.
Comparison of Weaknesses	A weakness of the Arduino circuit is that it may be too simple which results in a loss of features. Having a lack of features such as wireless control and monitoring may cause inconvenience to Urban Roots volunteers. In contrast, the weakness of the Raspberry Pi system is that it is more expensive. Price is a major factor for non-profit organizations, and it is important to make the automated system cheap.
Other Comparisons	

Arduino Raspberry-Pi





5 Potential Improvements

Our current TAHCS design was created with an emphasis on simplicity and consistency. This thinking helped us produce a practical system that could be easily and immediately implemented. Simplicity was a focus as we discovered that a greater number of features increases the number of potential errors. After a thorough evaluation of our design, we discovered three main improvements that with more time we could have implemented. These innovations would improve the ease of installation, functionality of the solution, and cost of the system. The improvements include the integration of printed circuit boards, Wi-Fi enabled applications, and richer installation resources.

A major improvement to our design would be to implement printed circuit boards (PCBs) which would significantly reduce costs and production times. Replacing the entire Arduino microcontroller, bmp 280 temperature & humidity sensor, and the breadboard circuits with a single printed circuit board would significantly reduce the costs and time it takes to create the TAHCS system (if made at an electronics lab which does not require mass production).

Our group originally planned to incorporate app and Wi-Fi connectivity to our design. We decided to iterate away from this idea because of the associated costs and the labour required to procure such features. With the ability to create printed circuit boards, having more features such as Wi-Fi connectivity and a remote camera would be significantly cheaper. Furthermore, with more time an external application (mobile app or website) could be created to establish control over the TAHCS at the hoop houses regardless of the location of the user. These internet-based applications will require advanced programming knowledge of databases, and front-end development in order to make a seamless and dependable experience for our clients. This is something that would enhance our system significantly, increasing user connectivity and ease of access.

After reviewing our design, a potential concern is the difficulty of installing this system. The installation process should be as simple and easy to follow as possible, considering our client might not have the required prerequisite knowledge to work with Arduino’s. With more time we could have created video tutorials, and in-depth manuals. This would allow us to fully empathise with our client’s need of simplicity and ease of installation.

With the main limitation of time, these alternate improvements were not possible, however in future iterations of the TAHCS, these features would be implemented.

6 References

"12 pcs solderless prototype Breadboard 6 X 170-tie points Mini Modular Breadboard Kit & 6 x 25 tie-points mini modular Breadboard Kit," *Amazon.ca: Electronics*. [Online]. Available: https://www.amazon.ca/dp/B091CFG7XV/?coliid=I21B6WNK9EU3JQ&colid=3FJW8X504FZ0Z&ref_=lv_ov_lig_dp_it&th=1. [Accessed: 04-Apr-2022].

"Cylewet 12Pcs 1A 250V AC 2 pins SPST momentary mini push Button Switch Normal Open (Pack of 12) CLT1078," *Amazon.ca: Tools & Home Improvement*. [Online]. Available: https://www.amazon.ca/dp/B07DS6QMRW/?coliid=IWC0R45AUBAE8&colid=3FJW8X504FZ0Z&psc=1&ref_=lv_ov_lig_dp_it. [Accessed: 04-Apr-2022].

"Dciustfhe 3Pcs BMP280 5V digital temperature barometric pressure sensor module, Temperature Sensor Atmospheric Pressure Board," *Amazon.ca: Industrial & Scientific*. [Online]. Available: https://www.amazon.ca/dp/B09J597CHS/?coliid=I9OHAD1QIGCU&colid=3FJW8X504FZ0Z&psc=1&ref_=lv_ov_lig_dp_it. [Accessed: 04-Apr-2022].

"Electrical wire CBAZY™ hook up wire kit (solid wire) 22 gauge 6 colors 26.2 feet each wires 22 AWG," *Amazon.ca: Tools & Home Improvement*. [Online]. Available: https://www.amazon.ca/dp/B0791BNDY2/?coliid=IBZW0P0WFJB8C&colid=3FJW8X504FZ0Z&ref_=lv_ov_lig_dp_it&th=1. [Accessed: 04-Apr-2022].

"KEYESTUDIO 5V MEGA 2560 R3 Microcontroller Development Board ATMEGA2560 CP2102 + type-C USB cable starter kit for Arduino Project Set," *Amazon.ca: Electronics*. [Online]. Available: https://www.amazon.ca/dp/B08V4RCRS2/?coliid=I3PQHVS1SE61AR&colid=3FJW8X504FZ0Z&psc=1&ref_=lv_ov_lig_dp_it. [Accessed: 04-Apr-2022].

"KEYESTUDIO IIC 12C 1602 LCD 16x2 display blue backlight module for Arduino Mega R3 2560 raspberry pi," *Amazon.ca: Electronics*. [Online]. Available: https://www.amazon.ca/KEYESTUDIO-Display-Backlight-Arduino-Raspberry/dp/B0177XQE7K/ref=sr_1_5?crid=1AFWA32KJMLD4&keywords=lcd%2Barduino%2Bdisplay&qid=1649017330&sprefix=lcd%2Bar%2Caps%2C65&sr=8-5. [Accessed: 04-Apr-2022].

"Miuzei 5 pcs SG90 servo motors kit mini SG90 9G servo motor for RC Robot helicopter airplane car boat remote control blue," *Amazon.ca: Toys & Games*. [Online]. Available: https://www.amazon.ca/Miuzei-Motors-Helicopter-Airplane-Control/dp/B07Z16DWGW/ref=sr_1_6?crid=

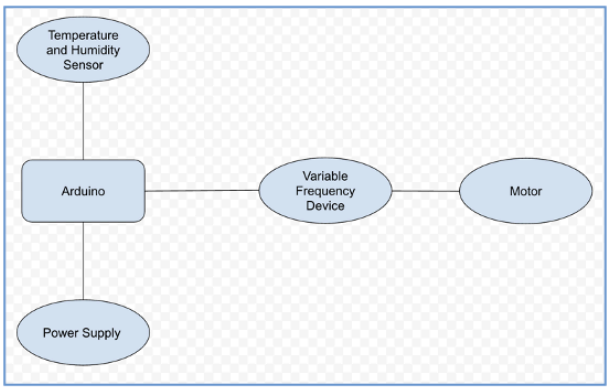
3AHNFV1OK54RQ&keywords=servo&qid=1649017482&sprefix=servo%2Caps%2C70&sr=8-6. [Accessed: 04-Apr-2022].

“Waveshare Raspberry Pi Zero 2 WHC five times faster 1GHz quad-core ARM cortex-A53 CPU with WIFI bluetooth 4.2 ble with Pre-Solder Color Coded pinheader,” *Amazon.ca: Electronics*. [Online]. Available: https://www.amazon.ca/Waveshare-Raspberry-Pi-Zero-WHC/dp/B09LCKPTHY/ref=sr_1_2_sspa?crid=1LJJA79XOCGGJ&keywords=raspberry%2Bpi&qid=1648935294&sprefix=raseberry%2Bpie%2B%2Caps%2C72&sr=8-2-spons&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUE5OE9HNU1GNUNFUTEmZW5jcnlwdGVkSWQ9QTA0MzI1OTAxQjlFWFk0Q0tHNEVLJmVuY3J5cHRlZEFkSWQ9QTA2MTk5NjJMQ0lYTUswMEVFUE4md2lkZ2V0TmFtZT1zcF9hdGYmYWN0aW9uPWNsaWNrUmVkaXJlY3QmZG9Ob3RMb2dDbGljaz10cnVl&th=1. [Accessed: 04-Apr-2022].

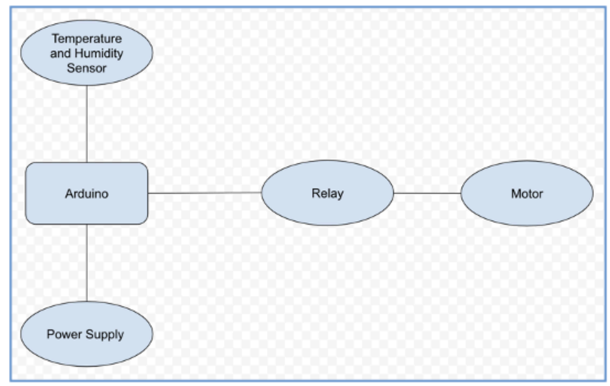
7 Appendix A – Examples of Design Documentation

7.1 Flowchart

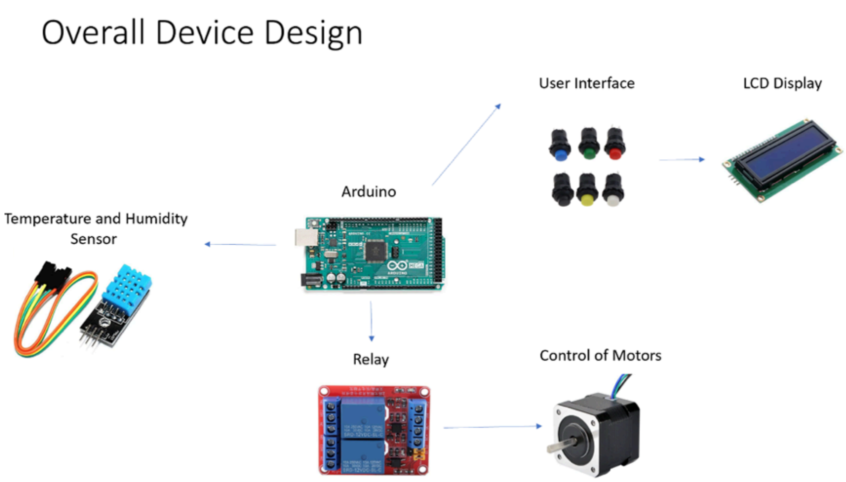
Final System Diagram



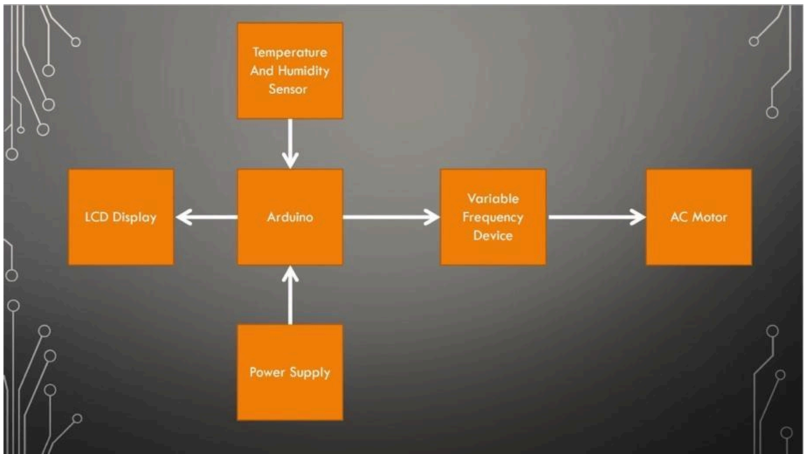
Presentation System Diagram



Prototype flow chart

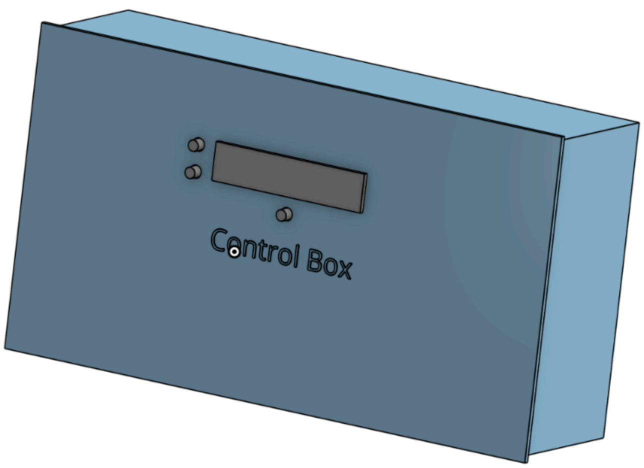


Final design flow chart

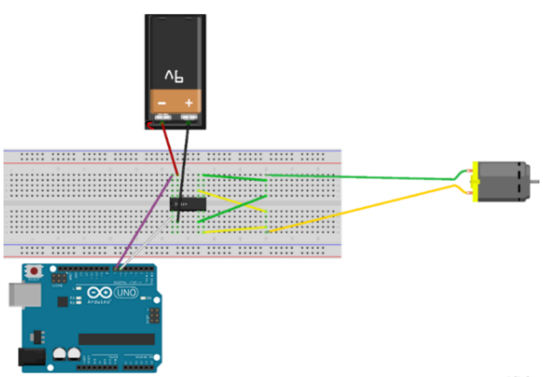


7.2 CAD Models or Drawings

Onshape:
(https://cad.onshape.com/documents/20ee39fd900c3b1d35ef262f/w/d96df70a564f7de0f733ed73/e/c24f7ffdb3dff3b3bd84eb05?renderMode=0&uiState=6248c08bbfdab14f2aad8bb8)

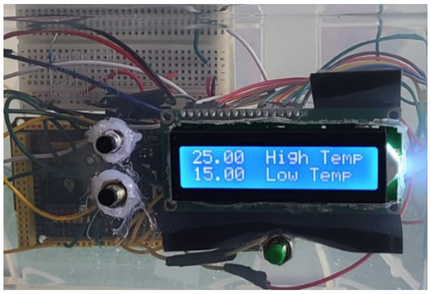


LCD Wiring Diagram

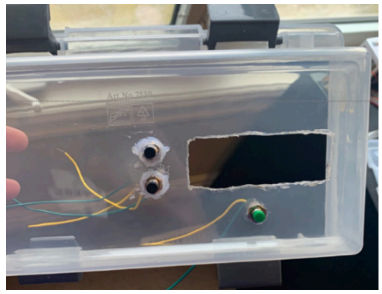


7.3 Apps Workflow/Interface

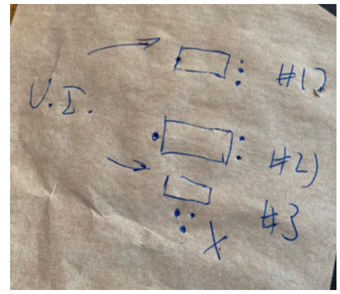
LCD Display & Buttons



Buttons Layout

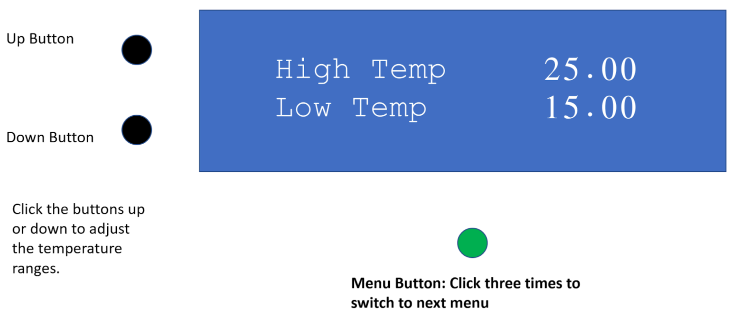


Button's Ideation

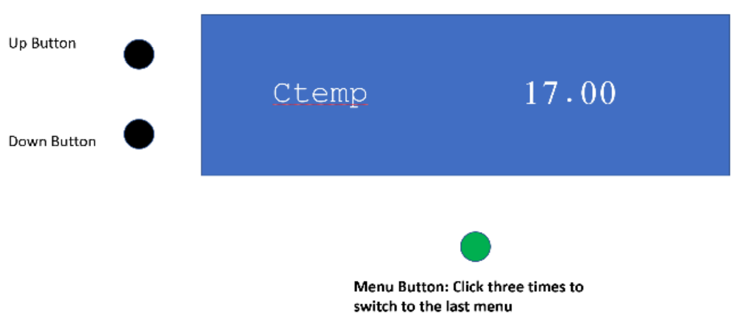


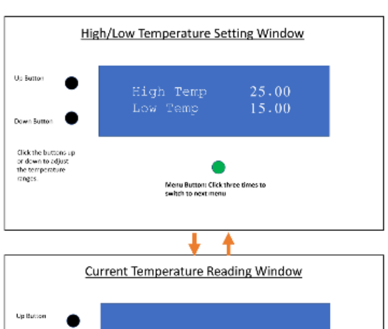
User Interface User Guide

<u>High/Low Temperature Setting Window</u>



<u>Current Temperature Reading Window</u>





As made evident with this user interface flow diagram the program is very simple and easy to use. With a lack of too many screens, buttons, and functions any user can naturally use TACHS without getting