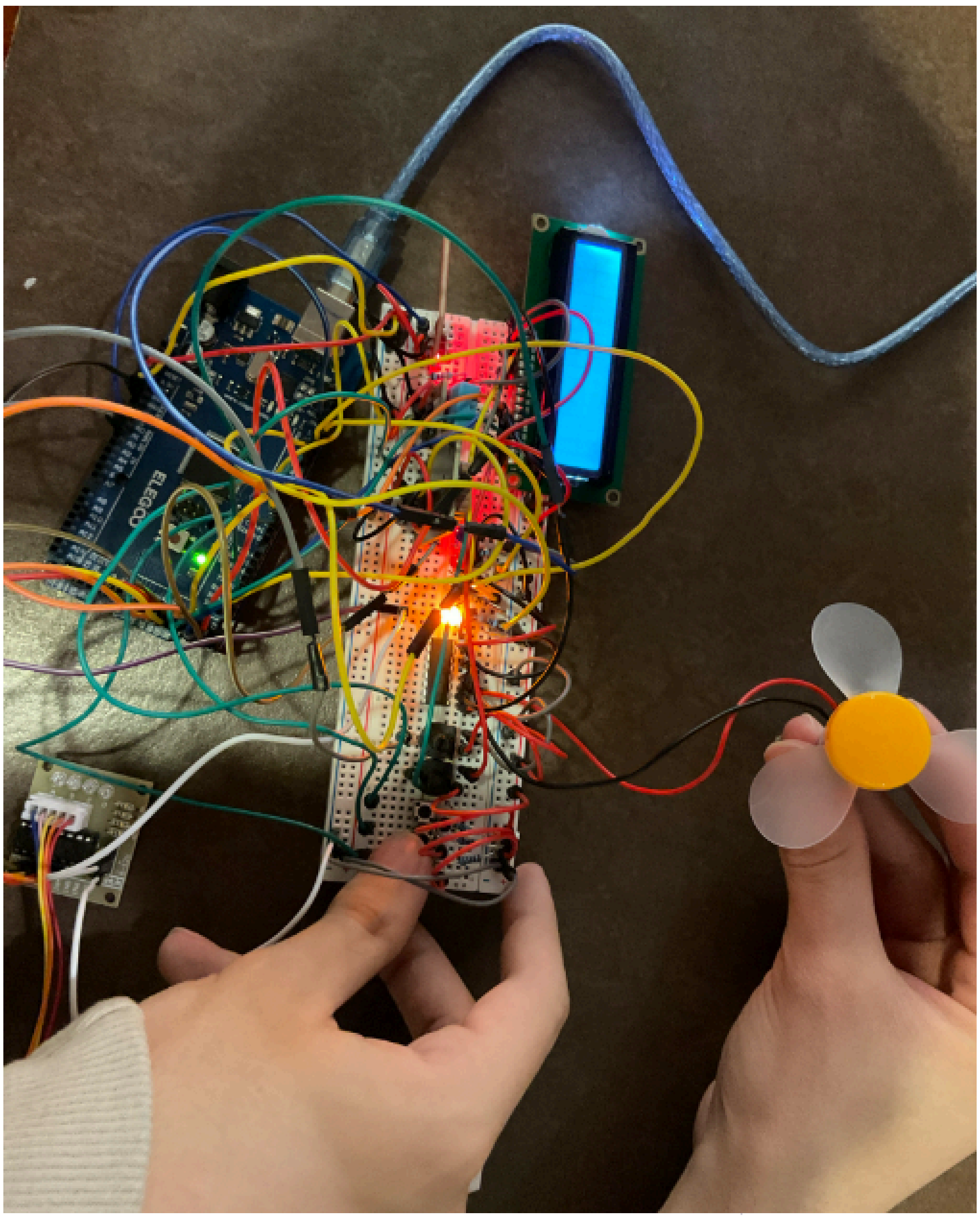
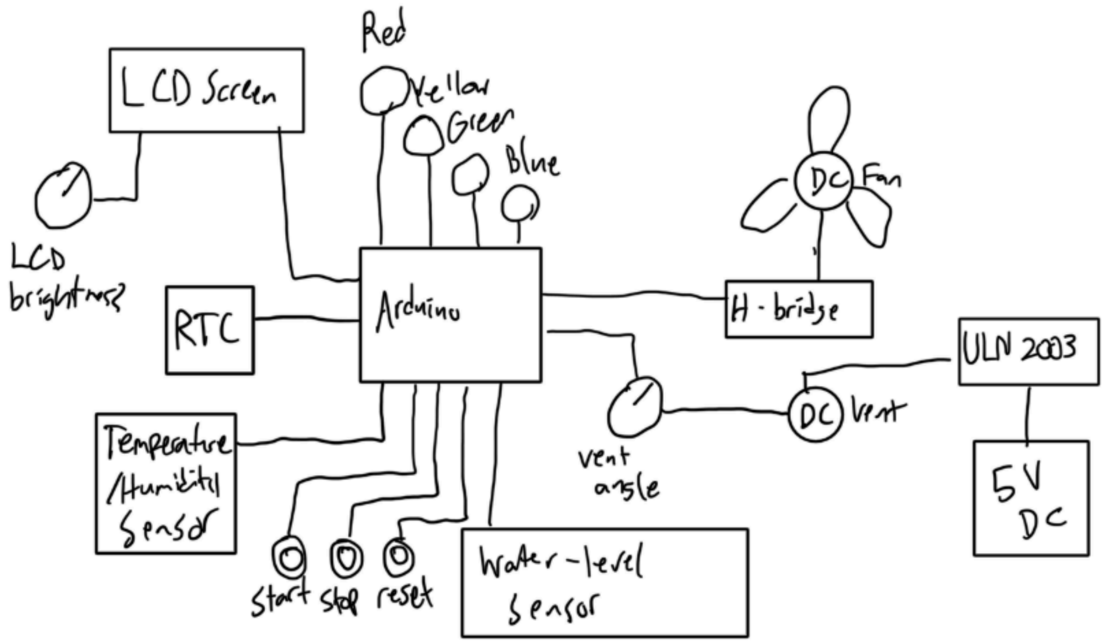
Final project report - Julian Hurley & Hunter Anderson
Group 36.

Our project has all components and functionality necessary for a fully-featured swamp cooler. It has a 4 state system consisting of active, idle, disabled, and error. Active and idle states are active only when there is no error triggering the error state or the system isn't disabled. Active means the value recorded by the temperature and humidity sensor are above the threshold, signaling to start cooling. In its active state, which is triggered by being in the idle state and the temperature exceeds a threshold, the blue LED lights up, the temperature and water level sensors are recording and periodically updates every minute on the LCD screen, and the fan motor blows. In idle, which is achieved by pressing the start button or the reset button when in the error state while the water level is high, the fan motor is not on and the green LED is illuminated, as well as continuing to periodically display information every minute about the water level and temperature. In the error state, which is achieved when either the idle state or active state detects the water level being too low, which indicates that it wouldn't be able to function properly as doesn't have enough water, nothing is activated and the red LED is illuminated and an error message is displayed to the LCD. Just like in the error state, the disabled state, which is achieved by the stop button being pressed, has nothing activated and the yellow LED is illuminated, and the LCD is cleared, but you also can't control the vent. In all states but disabled, the user can move potentiometers to change the display brightness of the LCD or the vent angle. Whenever the vent is moved or a state is changed, information is sent to the console regarding the real time the change took place as well as if the vent was moved or what state the system changed to. The components used are an H bridge IC to power the fan motor, the 5V power board to power the vent DC motor, a driver circuit to help control the DC motor, the vent motor itself, which angles the direction of the cooled output air, 3 buttons to change from specific states, a water level sensor, a temperature/humidity sensor, a realtime clock, 4 colored LEDs, a LED backlit LCD screen, 2 potentiometers to control LCD brightness and vent angle via analog input, and the arduino to interface the components with the logic of our program. Diving into the overall functionality of the system, the water level sensor detects and displays how high a body of water is, the temperature/humidity sensor detects and displays the temperature of the air, and the fan is active when the system is in the active state as a result of the recorded temperature exceeding the threshold, otherwise the fan is off. Furthermore, whenever the potentiometer controlling the vent is turned to the left, the vent will rotate to the left until the potentiometer is returned back to its default state. As a result, when the potentiometer is turned to the right, the vent will rotate to the right. This is possible as a result of the driver circuit which helps send the necessary information to have the vent turn left or right. Meanwhile, the H bridge is able to turn on the fan and control its speed via a signal received by the arduino. The other potentiometer controls the brightness of the LCD. The LCD helps us see the information being recorded by the water level sensor as well as the temperature sensor, or if there has been an error in the system. The real time clock syncs up with the real time of the day and allows us to have access to that information to keep track of the times that our system is either modified via the vent or has a state change. The power needed for the system is provided by the arduino’s USB connection and the other power board's 5V output. The operating temperature minimum and maximum are constrained by the highest minimum and lowest maximum of any of the components.



(Assume all components are also connected to power and ground)



Specification Sheets:

https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf
https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf?srsltid=AfmBOoonAj_6PW0LFpuOCEo32IoklQ6pHYuOQu6URKtzpPBUIBJaXz0Z
https://curtocircuito.com.br/datasheet/sensor/nivel_de_agua_analogico.pdf?srsltid=AfmBOoozt28erXcb7HyW2ygEbxPYeKWKdobSibTydCtRG8kaCeXKwzvD
https://www.analog.com/media/en/technical-documentation/data-sheets/ds1307.pdf
https://www.mouser.com/datasheet/2/758/stepd-01-data-sheet-1143075.pdf?srsltid=AfmBOorW52nsG63ijiGWpjk98vcz4y3CfGfNrqpbL7IRb8_to1mK_3QI
https://www.farnell.com/datasheets/1863913.pdf
https://www.hadex.cz/spec/m513.pdf

https://www.ti.com/lit/ds/symlink/l293d.pdf?ts=1765466837433&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FL293D
https://www.vishay.com/docs/37484/lcd016n002bcfhet.pdf
https://www.handsontec.com/dataspecs/mb102-ps.pdf