Swamp Cooler – CPE301 Final Project

Zach Zimmermann

Matt Facque

Project Overview

The swamp cooler upon startup initializes to 'idle' mode, which is denoted by the green LED. From this point, the water reservoir is in an ok state (there is enough water to run the swamp cooler) and the user has multiple options to run the swamp cooler. If the 'standby' button is pressed, the system reverts to standby mode which disables interrupts, disables the fan motor, the temp/humidity sensor, disables the water sensor, and for a visual cue of 'standby' status, lights the yellow LED. If there is a problem with the water reservoir, then the system will revert to 'error' mode which is denoted by the red LED. In the 'error' state, the temp/humidity sensor is disabled, the fan is disabled, and the vent is closed. In the 'error' state, if the water level problem is resolved the swamp cooler will revert into 'idle' mode.

During 'idle' mode, the temp/humidity sensor will be constantly taking readings and displaying the results. If the sensor reports high temperature, the system will go to 'operation' (blue LED) state. The system will enter 'operation' mode if the registered temperature of the sensor is greater than 70 degrees Fahrenheit and there is an acceptable level of water in the reservoir. In 'operation' state, the user can press the 'standby' button which will cause the system to go to the 'standby' state. Furthermore, if there is a problem with the water reservoir, the system will enter the 'error' state.

More information on the states of the system. 'Idle' state is the state the system will initialize into. It is characterized by high water levels and low temperature levels. 'Standby' state is the user operated state where the system sensors are disabled, system interrupts are disabled, vent position is moved to close, and the onboard fan is turned off. 'Standby' state is caused by a button press. 'Error' state is characterized by a problem with the water reservoir. If the reservoir is low on water, the system will enter 'error' state. In 'error' state, the fan will be disabled, and the vent will be closed. Once the reservoir is filled again, system normal functionality will resume. The water level of the reservoir is monitored by the water level sensor. Finally, 'operation' state is characterized by high temperature (monitored by the temperature/humidity sensor) and high water levels (monitored by the water level sensor). Due to the high temperature, the vent will be moved into the open position and the fan will turn on. After operation for an extended duration of time, the temperature as read by the sensor should lower in which case the swamp cooler can revert to 'idle' state. If during operation, the water level in reservoir drops to a low state, the system will enter 'error' state until the water level is increased.

The system will report to the onboard LCD and to an external serial monitor to log operation information. The LCD will display a constant value for ambient temperature and humidity (read from the temperature and humidity sensor) and the external serial monitor will report on operation times. The operation times will log as month/day/year as well as whether the fan is enabled or disabled.

System Specifications

5 V (system on-board power supply)

5 V (external power supply)

4x LED (red, blue, green, yellow)

LCD module

10KΩ Potentiometer

L293D DC motor control IC

DHT11 Temperature and Humidity Sensor

1x Push Button

5x 330 Ω Resistors

Fan Blade and 3-6 V DC motor

Water Level Detection Sensor Module

Servo Motor SG90

Power Supply Module (External Supply)

Mega 2560 Arduino

“Barrel Jack” Power Supply Cable

USB Power Supply Cable

Operation Specifications

Idle = TEMP < 70° F, WATER > 200*

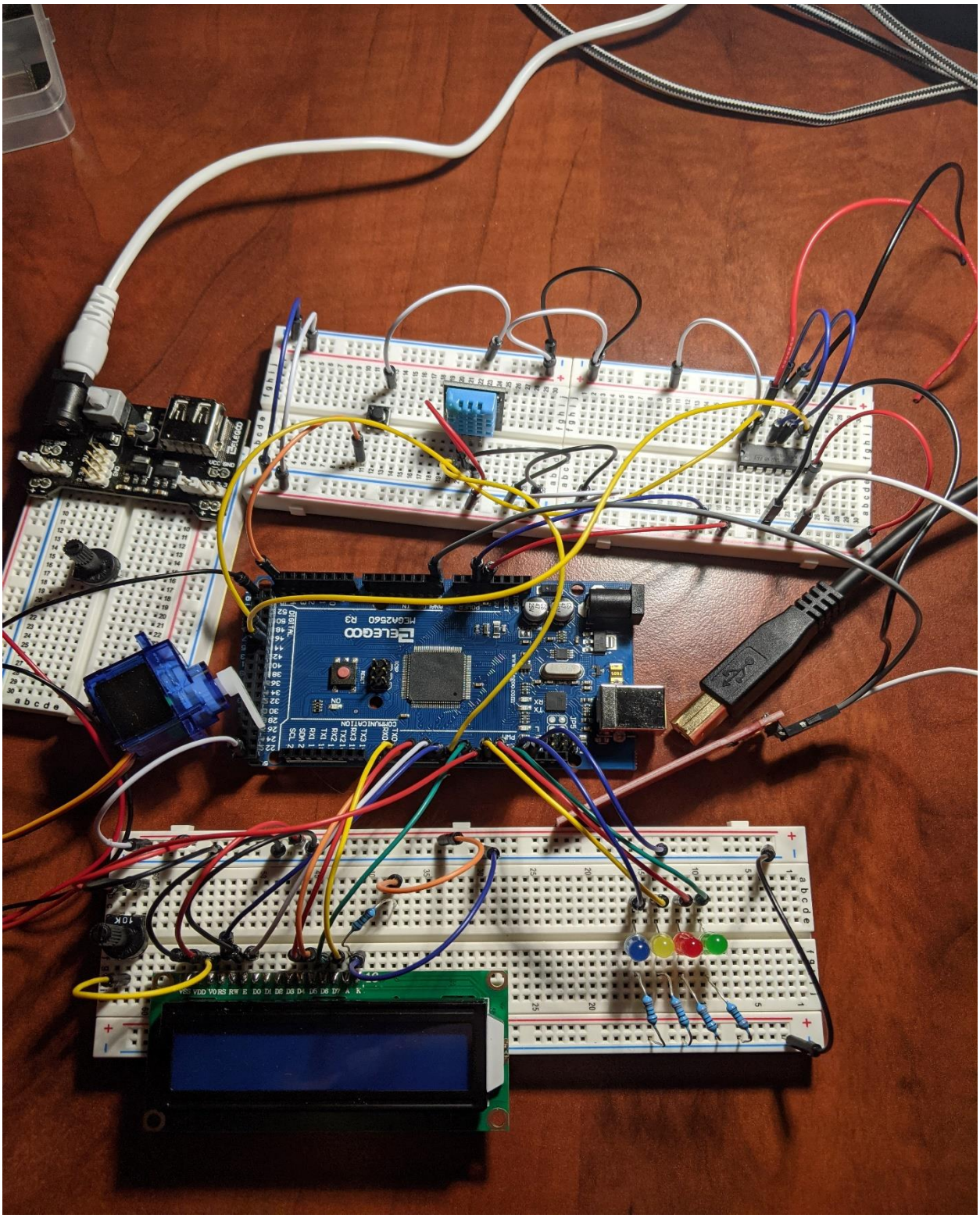
Error = WATER < 200*

Operation = TEMP > 70° F, WATER > 200*

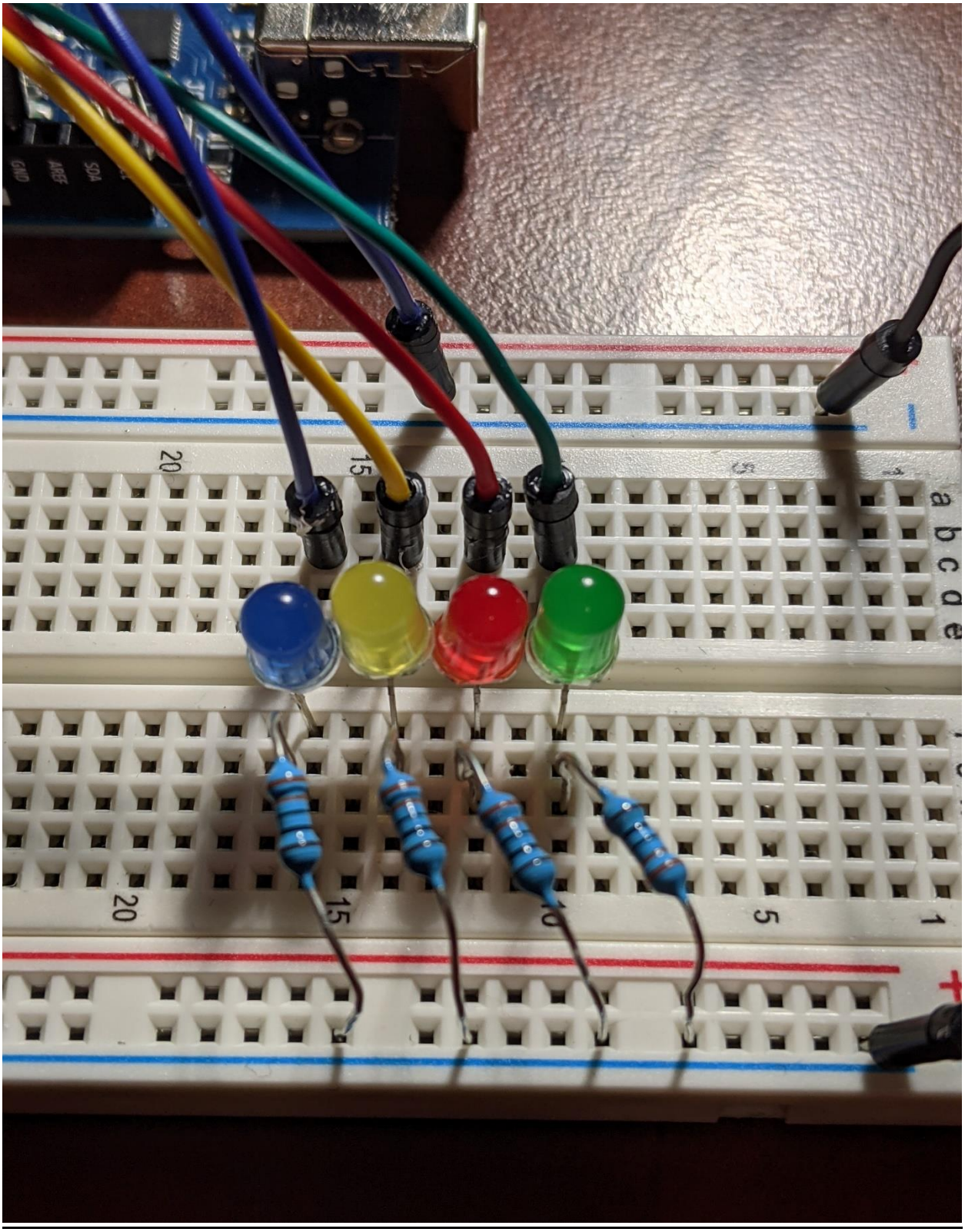
*Water Sensor Level

Swamp Cooler System

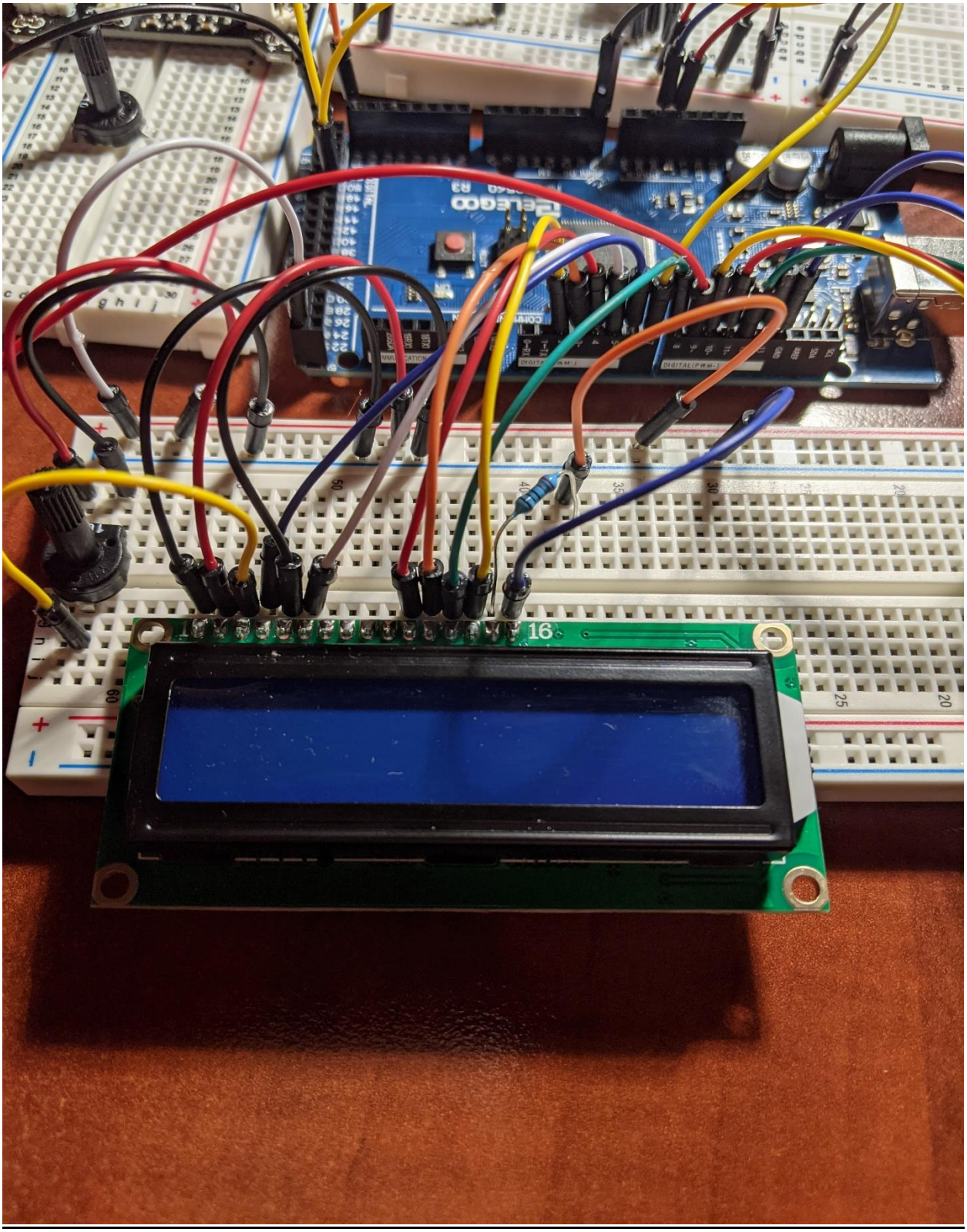
1.System



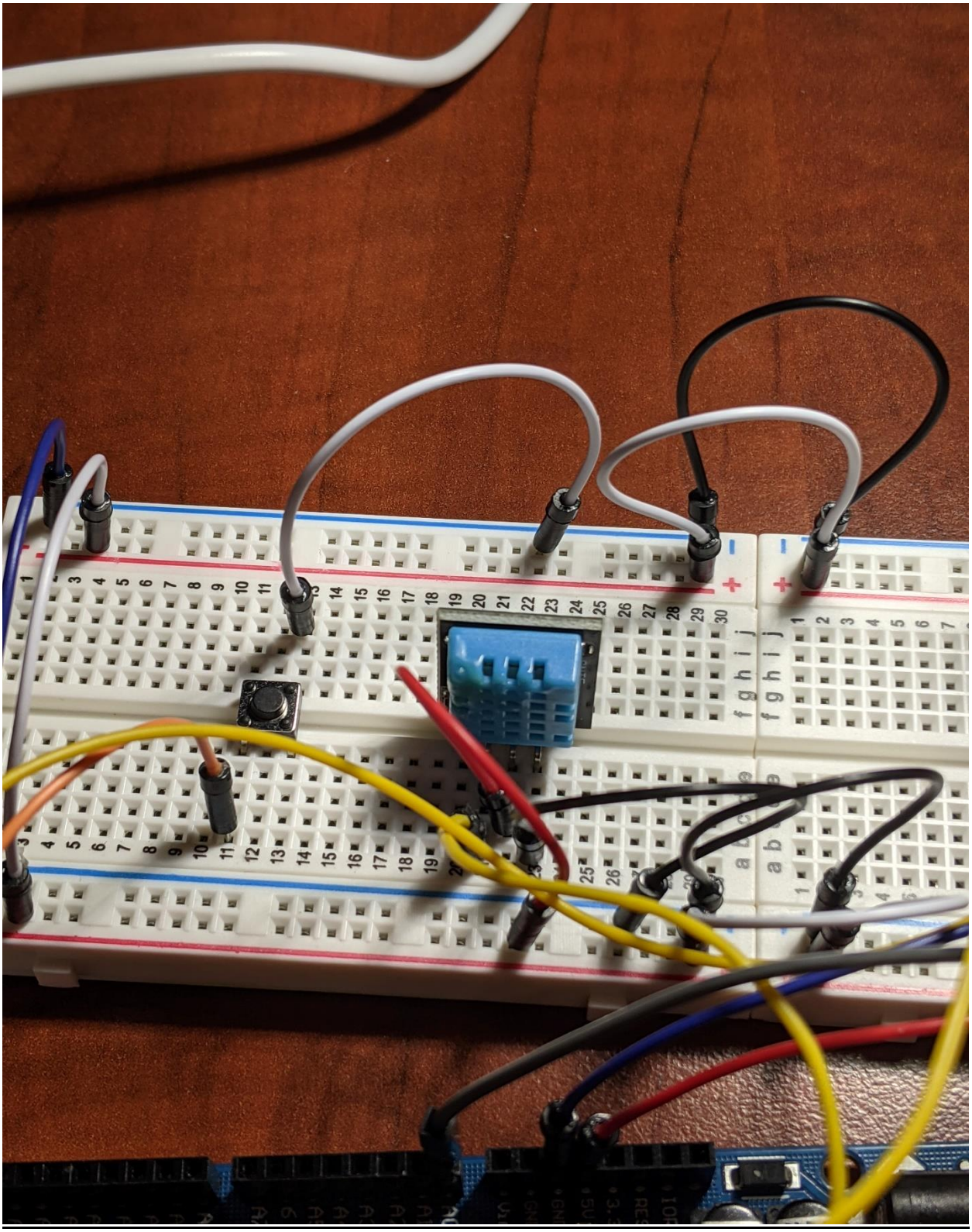
2.System Control LEDs



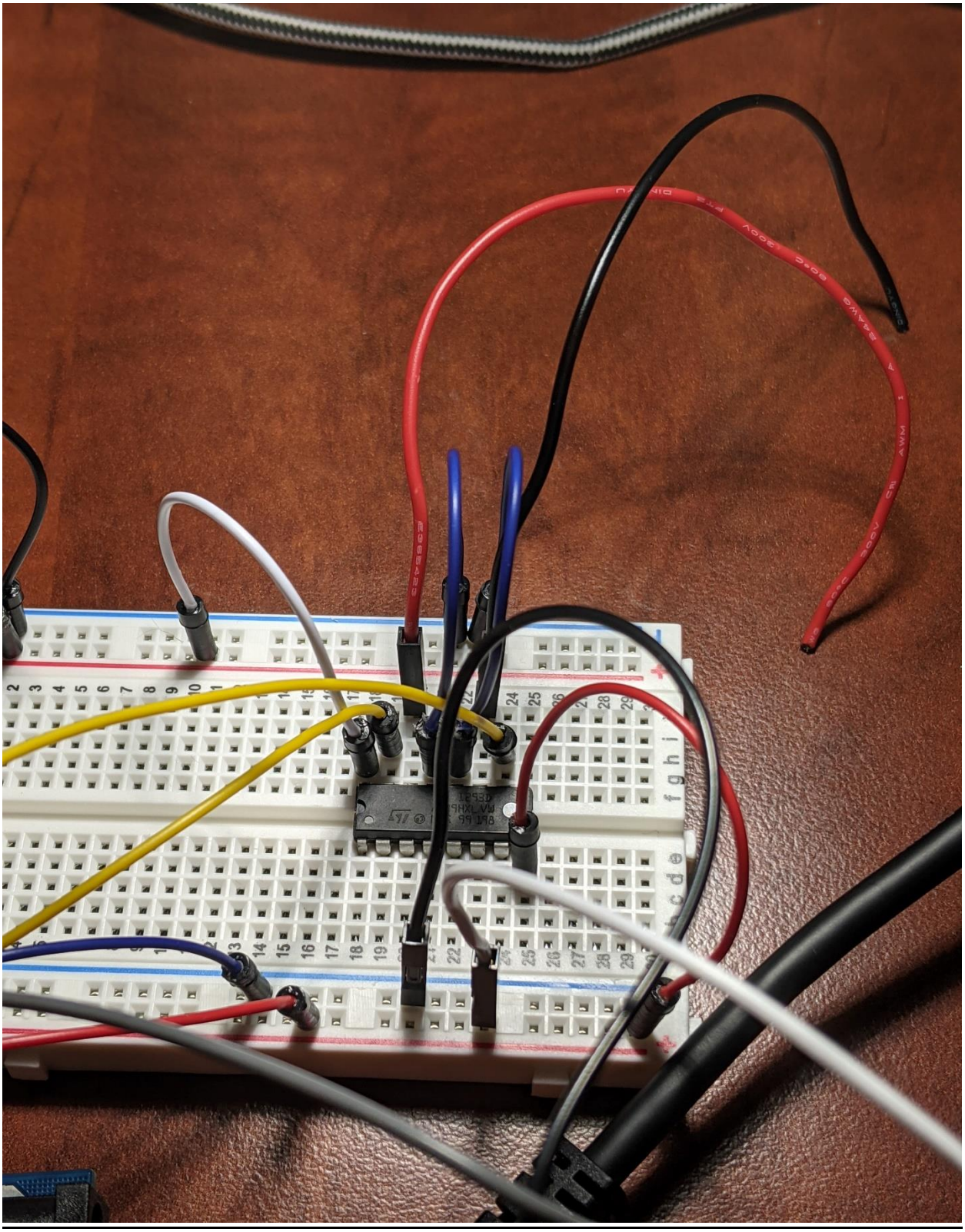
3.System LCD



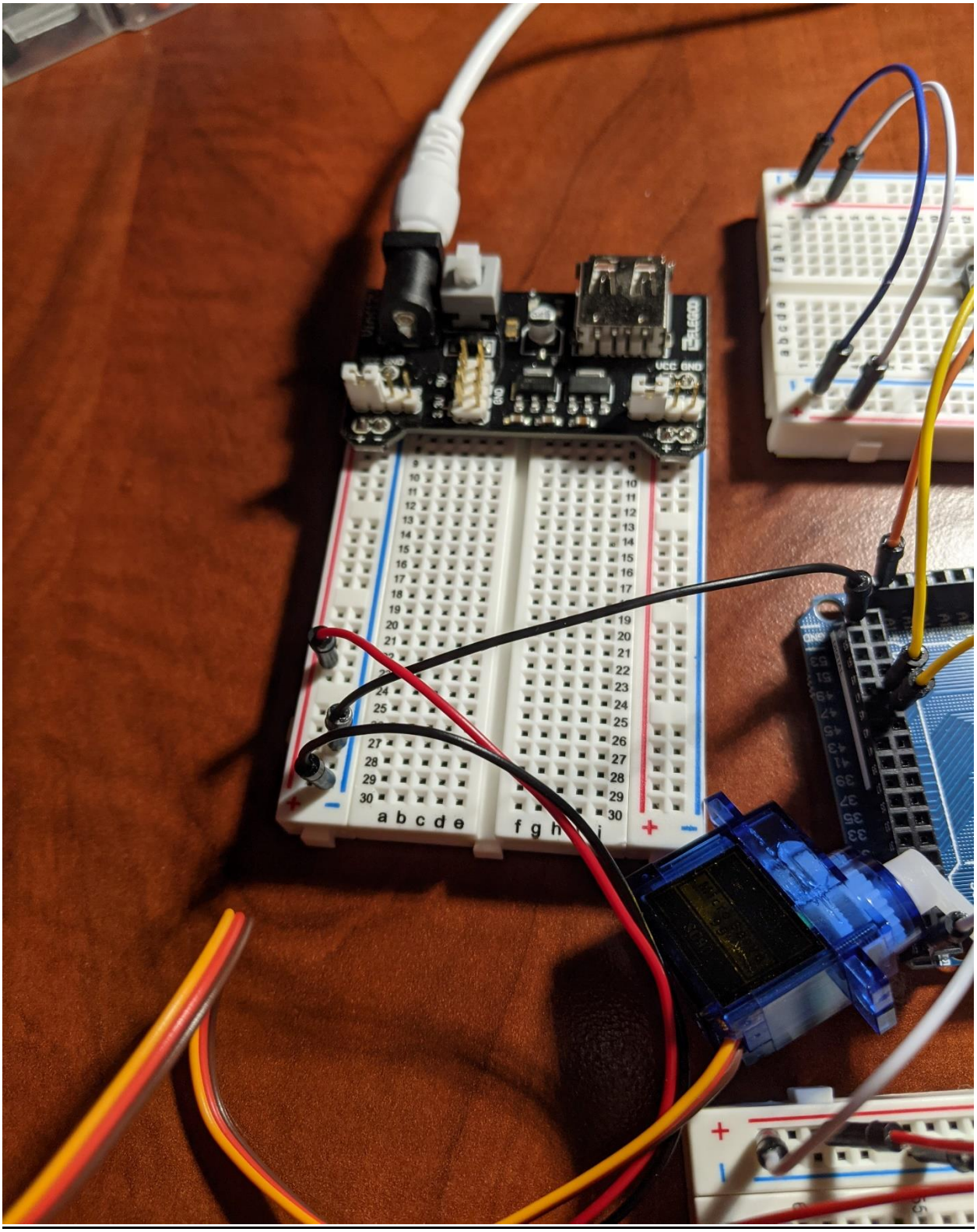
4.Temp Sensor & ‘Standby’ button



5. Fan output w/ Motor Control IC*



6. Servo w/ External Power Supply



1. The test system. All components visible.
2. System control LEDs, all LEDs each in series w/ 330Ω resistor.
3. System LCD w/ 10kΩ potentiometer (used for contrast with LCD).
4. Temp sensor w/ V, GND, and Sensor output.
5. The fan component was lost as shown by the red and black output from the motor control IC. After testing with a complete system, the fan output functioned as intended.
6. External power supply module for servo.

Component Specifications

LCD: https://components101.com/16x2-lcd-pinout-datasheet

Water Level Sensor: https://curtocircuito.com.br/datasheet/sensor/nivel_de_agua_analogico.pdf

Sg90 Servo Motor: https://components101.com/servo-motor-basics-pinout-datasheet

Power Supply Module: http://eeshop.unl.edu/pdf/mb-v2%20datasheet.pdf

DH11 Temperature and Humidity Sensor: https://components101.com/dht11-temperature-sensor

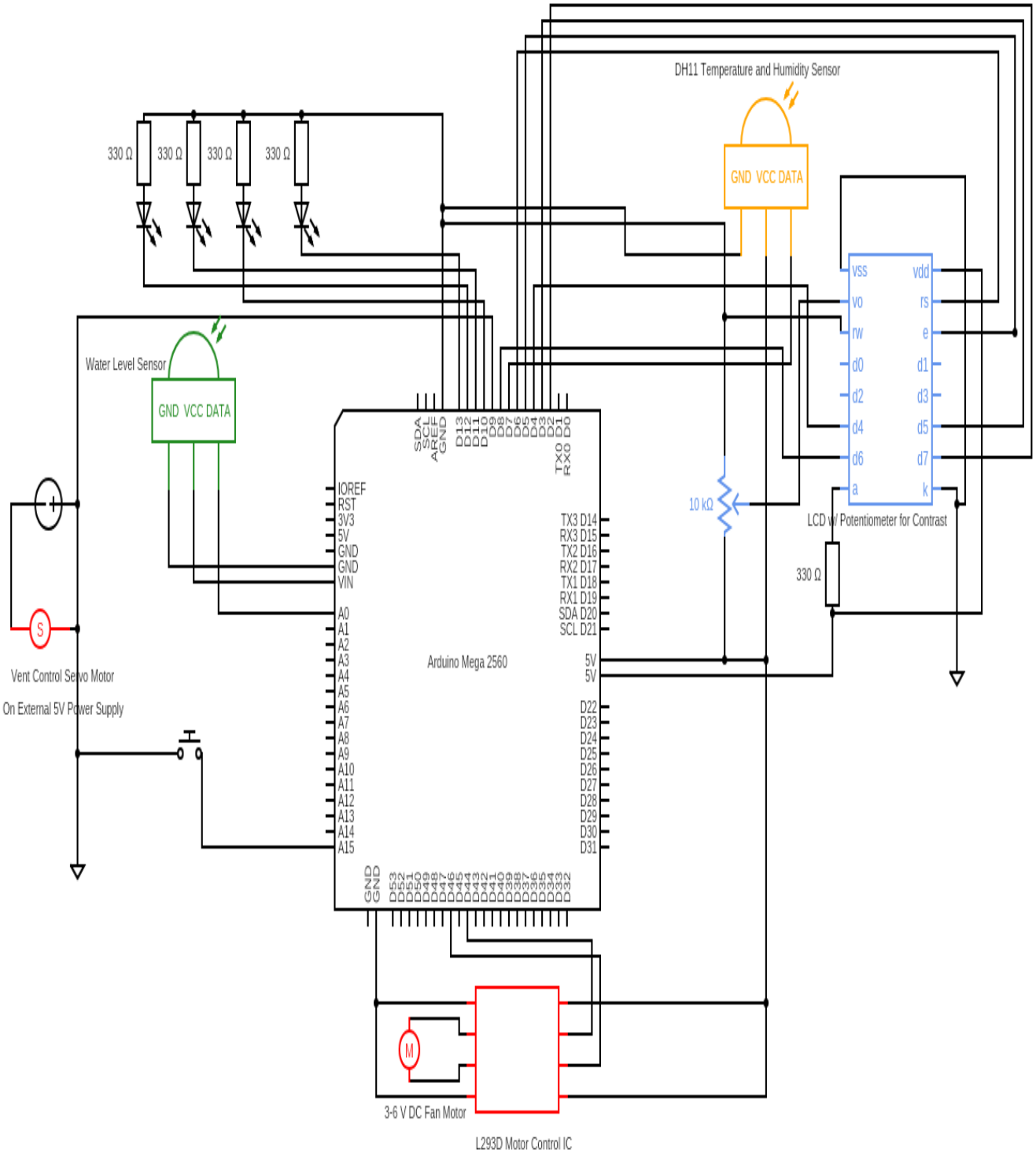
L293D Motor Control IC: https://www.arduino.cc/documents/datasheets/H-bridge_motor_driver.PDF

3-6 V DC Motor: https://bc-robotics.com/tutorials/controlling-dc-motor-arduino/

Mega 2560: https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf

Github

https://github.com/hkfrtyseven/CPE301_Swampcooler.git



Schematic of the system