Stolan Belikove
CPE 301.1001
December 14, 2023

# Final Project (Swamp Cooler)

This report will detail how an Arduino Mega 2560 was used to help construct a “swamp cooler” (evaporating cooling system). Parts used include a water sensor, humidity sensor, LCD display, four buttons, a fan motor, a L293D IC, and four LED lights. The system measures the current humidity and air temperature (outputting the results to an LCD screen), and will activate as necessary based on these readings.

Despite effort on the author’s part, a functional system was not created. The author had significant issues with creating a working circuit. Though the components were functional (aside from the LCD as detailed below), the system was non-responsive despite input from the user. The author also encountered an issue wherein the ELEGOO power supply module was found to generate significant heat, and a noticeable smell of smoke when left powered on for too long. This made debugging the circuit difficult. Though the power supply appeared to be grounded properly, this may suggest some sort of wiring mistake on the part of the author.

The author also had issues with the LCD module, as mentioned in a previous lab. While the LCD would turn on, no readings were shown on the screen. The author made an attempt to write working code despite these limitations. An explanation of the circuit design and how the system is programmed to operate will also  be provided.

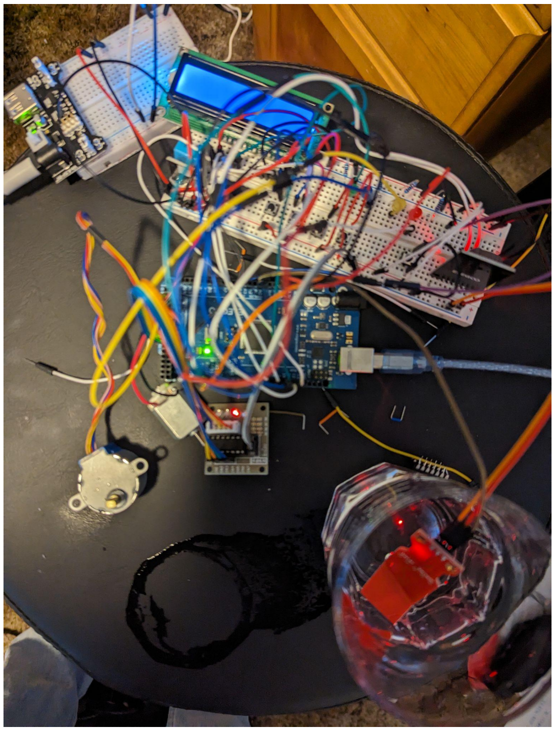
The ambient temperature and humidity are read using the DHT.h library and sent to the appropriate pin. The readTemperature() and readHumidity() functions were used for this. Based on the read temperature, the system should change states to and from “idle” and “running”.

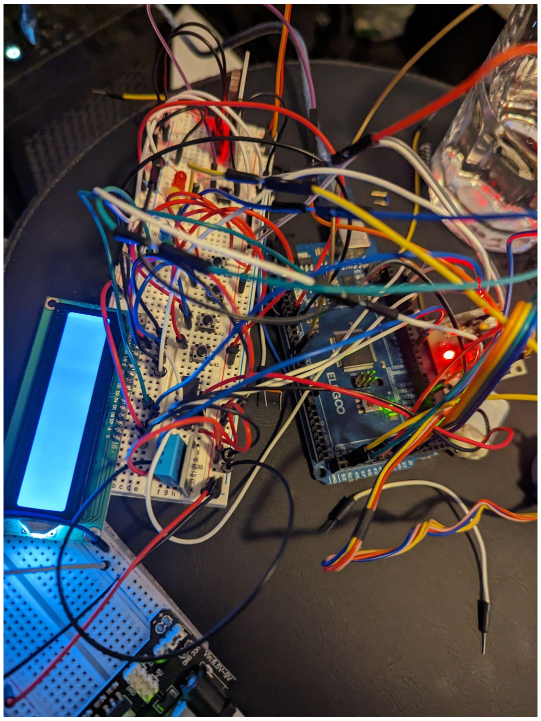
A water level sensor is used to ensure the system does not activate when no water is present. A cup or any container can be used to store the water, and by submerging the water sensor, the water level can be read with the adc_read() function.

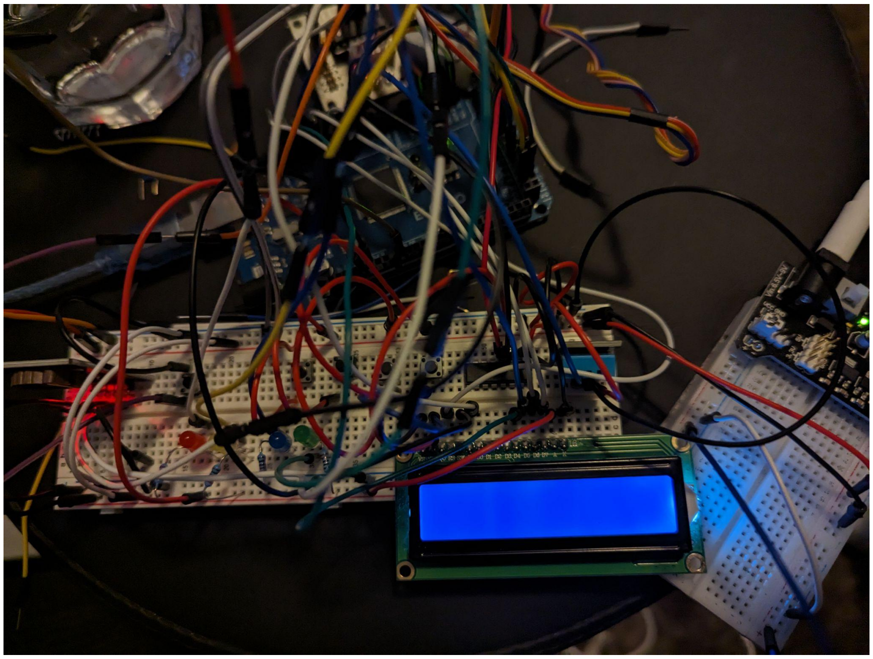
Four buttons on the breadboard handle system operations: an On/Off button, a Reset button, and two stepper buttons to control up and down.

The four LEDs are as follows:, green for “idle” mode, yellow for “disabled”, blue for “active” mode, and red for “error” or “interrupted state”.

The loop function handles setting the appropriate state and controlling the step motor. Readings are gathered every minute (if in the appropriate state). When a state changes, it is logged and output by sending characters directly to the UART.







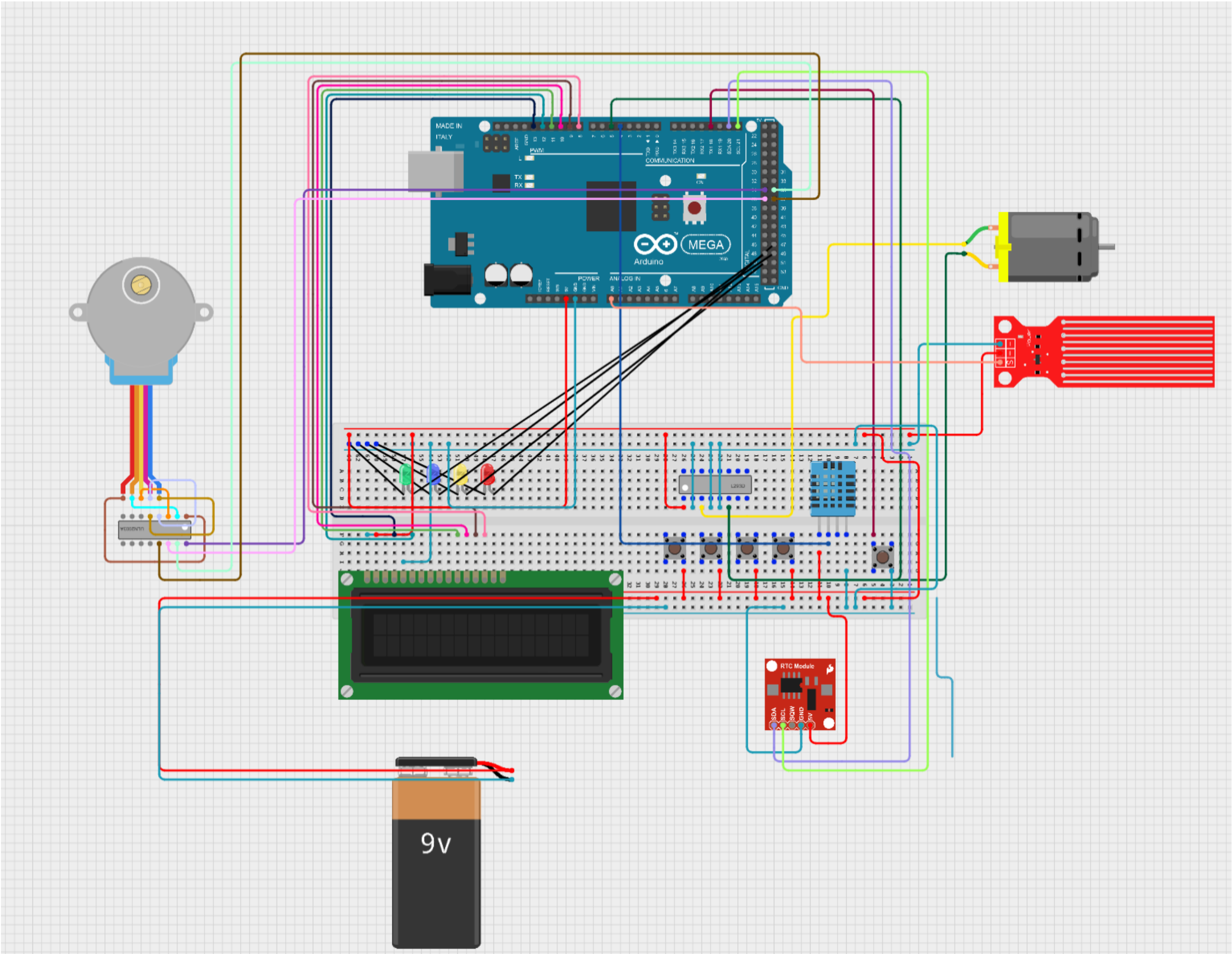
```
struct SensorReadings {
```

Message (Enter to send message to 'Arduino Mega or Mega 2560' on 'COM3')

New Line

9600 baud

Schematic



Note: the breadboard power supply module, and motor driver module which are included in the author's arduino kit, were not available in the software "Cirkit Studio" at the time of writing. A battery (bottom left) and the  relevant IC attached to the driver module (left, under the motor) were substituted instead.

Github Repository

https://github.com/Stolan00/CPE-301-Final-Project

Specification Sheets

**Clock**
https://www.analog.com/media/en/technical-documentation/data-sheets/ds1307.pdf

**DHT11**

https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf

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