CPE 301 Final Project Deliverable

Tyler Dishman
Matthew Gaskell

1. Project Overview

The goal of this project was to create a working swamp cooler which uses evaporation to cool off a region using evaporation. Our final project accomplishes this task using the Arduino Mega 2560 as well as various components. Our system properly monitors the status and inputs of various sensors, and controls the state of the system based on these inputs. The result is the working computational components of the swamp cooler.

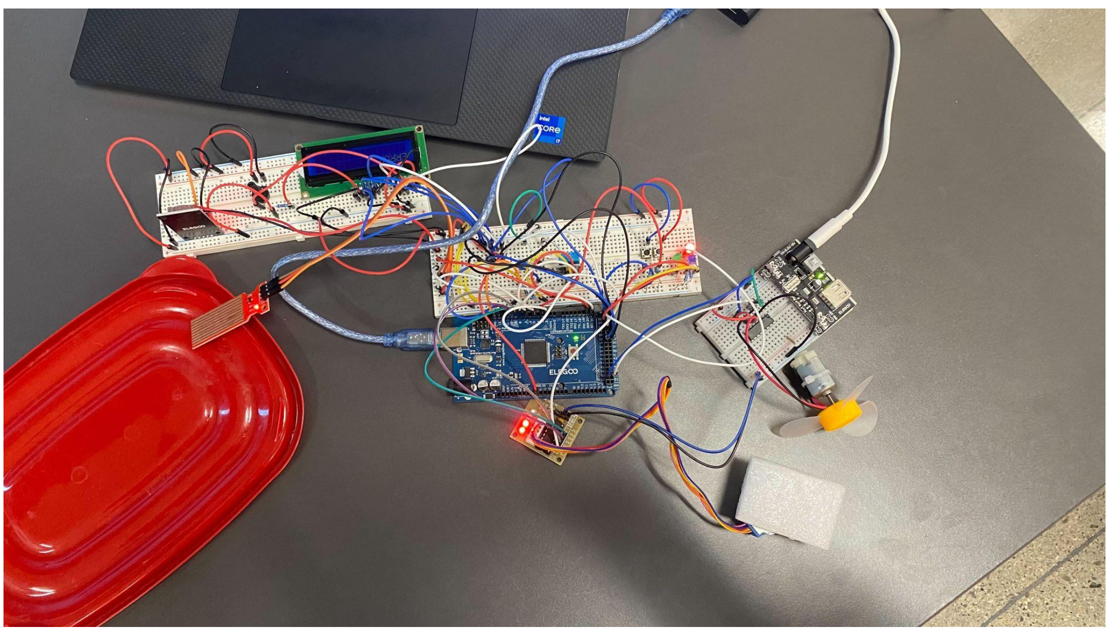
2. Design Overview

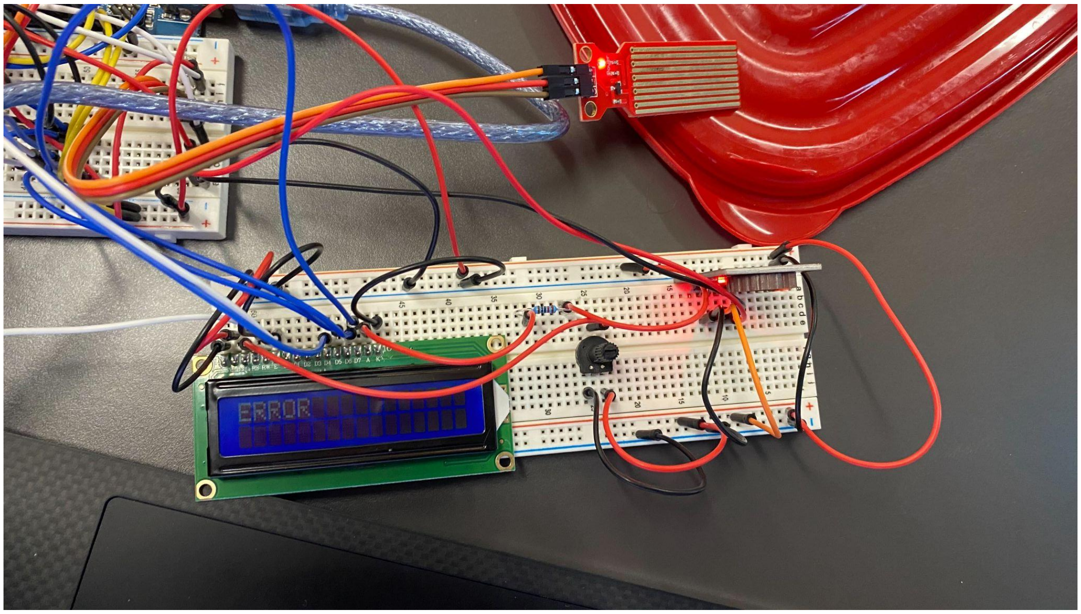
Our design utilized components of the Arduino Mega 2560 kit. We primarily follow the flowchart for state control, so our system allowed vent control in our positions except ERROR. The system has two different sensors. The DHT11 temperature and humidity sensor measures those two values. These values can be viewed on the liquid crystal display. The Arduino libraries were used for both of these apparatuses. Additionally, we added a potentiometer that adjusts the brightness values on this display. A water sensor continuously monitors the water level, using analog-to-digital conversion to take a sample. Our circuit has two motors, one being the stepper motor that is used to open and close the vents. This is controlled by two buttons, each rotating the motor in a different direction. This simulates opening and closing the vent. The stepper motor uses the Arduino stepper library for its commands. The kit motor is used to power the fan, whose blades are attached to the motor. These motors utilize the external 5 volt power supply, instead of taking power from the Arduino. This to protect parts of the circuitry within the Arduino. Two buttons are used for main control of our circuit, the start/stop button and the reset button. The different states of the system are modeled by the LEDs, with yellow meaning disabled, green meaning idle, blue meaning running, and red for error. In the disabled state, nothing is enabled, monitored, or switched on, except for the start/stop button. Once that is pressed, the system goes to idle. From this state, the machine prints out the temperature and the humidity on the liquid crystal display, giving updates every minute. The buttons that control the vent are enabled, allowing it to be opened and closed at any point. The fan is turned off, and the water sensor is reporting adequate water levels. Once the temperature sensor has reported values that fall above our predetermined threshold, the circuit switches into the running state, where the fan turns on, and begins cooling down the system. This continues until the temperature values recedes back below the threshold, in which case the state goes back to idle. If at any point in the running or the idle state the water sensor reports levels that are too low, the circuit is kicked to the error state. In this state, the fan is off, the vents are unable to be opened or closed, and the display shows an error message. This can be resolved by pressing the reset button, which sends the circuit back to idle, if sufficient water levels have been achieved. Every state change is sent to the computer and displayed on the serial monitor using UART. This is accompanied by the time, which is monitored and taken from the DS1307RTC, or the real-time clock. This only has to be initialized once, and now everytime our circuit is powered up it will have the time. Our input and output use register values, with written binary values to control their uses.

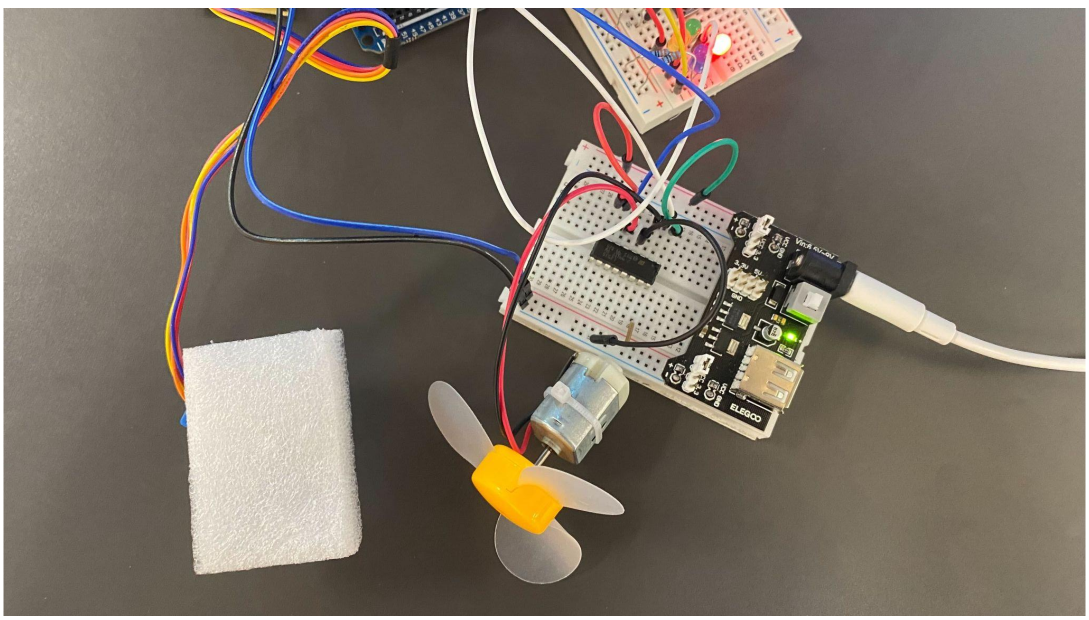
3. *System Constraints*

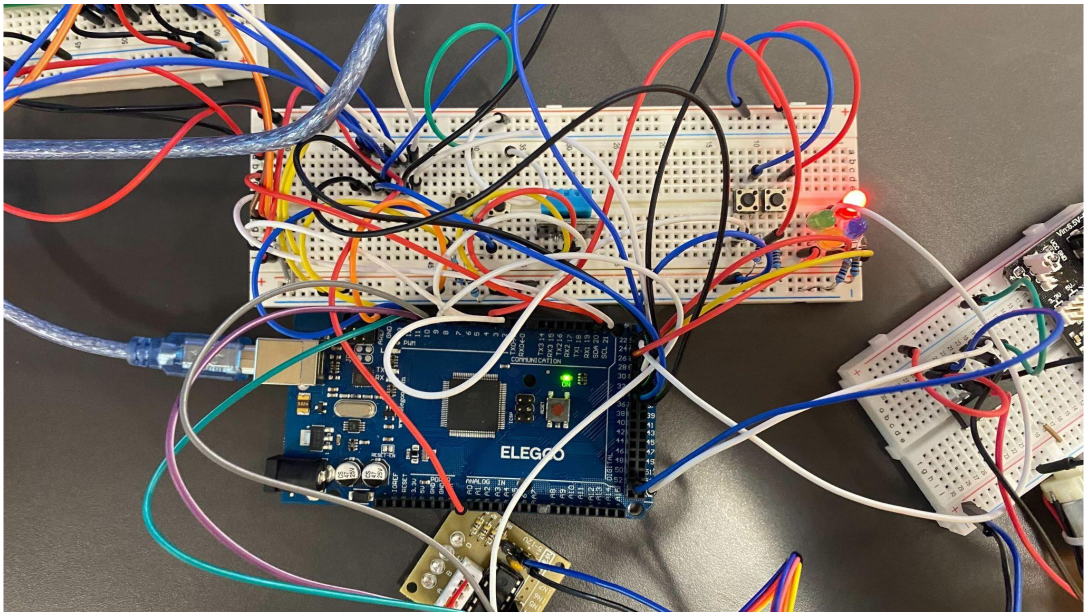
a. System temperature constraints:
   i. 0-50 °C (Limited by DHT11 operating temperature range)
b. System power requirements:
   i. Idle: 72.5 mA @ 5V DC
   ii. Running: 130.0 mA @ 5V DC
   iii. Full load: 260.0 mA @ 5V DC
c. System humidity constraints:
   i. 20-80% (Limited by DHT11 operating humidity range)

4. *Pictures*




ERROR

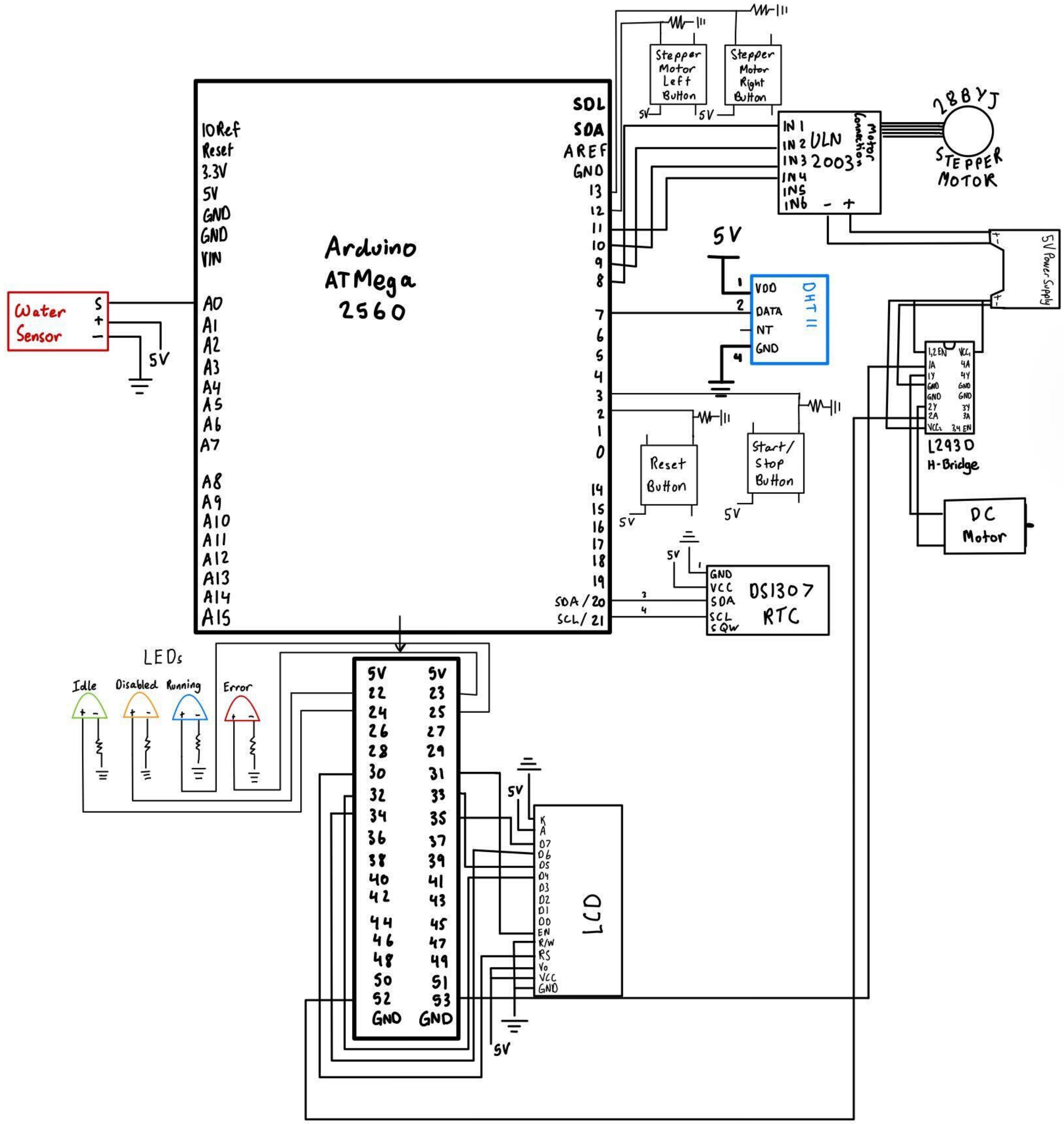

ELEGOO



5. *Link to Operational Video*

a. Circuit Operation Video
b. Additional Circuit Explanation Video

6. Technical Drawing/Complete Schematic



7. Relevant Specification Sheets/Resources Used

a. ATmega2560 Datasheet
b. Arduino Mega 2560 Pinout
c. DHT11
d. L293D (H-bridge) Datasheet and DC Motor
e. ULN2003 Driver and 28BYJ-48 Stepper Motor
f. DS1307RTC

g. Liquid Crystal Display
h. 5V Power Supply
i. Water Sensor

8. ***Github Repository***
    a. https://github.com/Dishman-Tyler-1102/CPE301Final