

Swamp Cooler

Leslie Becerra, Kj Moreno, Khoa Minh Do

University of Nevada Reno

CPE 301.1104

1 Design Overview

This project focuses on building an Arduino based embedded control system for a swamp cooler. It uses the Arduino Mega2560 to manage a set of sensors and actuators that work together to maintain proper cooling and report system status in real time. The system includes a start button triggered by an interrupt service routine (ISR), a stepper motor paired with a potentiometer to adjust the vent direction, a DC motor to power the fan, a DHT11 sensor to monitor temperature and humidity, a water level sensor to detect low water conditions, and an RTC module to log events like state changes and motor activity. LEDs are used to indicate the system's current state (Disabled, Idle, Running, or Error), and an LCD screen displays live sensor readings and updates.

The software for this project is written in C and runs on the Arduino IDE, with minimal use of high-level libraries to emphasize low level register-based programming. The system operates in a loop, constantly checking its current state using a `checkState()` function. This function is only called when the system is not disabled. It reads sensor inputs to determine whether the swamp cooler should be in Idle, Running, or Error. Depending on the conditions, such as water level and temperature, it transitions to the correct state. Each state is handled using a switch-case structure, which controls actions like activating the fan, displaying sensor values, or responding to user input. The overall design keeps the system responsive and accurate while maintaining efficient operation using real-time data and hardware control.

1.1 Components Used

- **Arduino Mega2560** – This microcontroller serves as the brain of the system, integrating all sensors, actuators, and control logic for the swamp cooler.
- **4 LEDs (Green, Blue, Red, Yellow)** – These LEDs indicate the current state of the swamp cooler. Each color corresponds to a different state: GREEN for IDLE, BLUE for RUNNING, RED for ERROR, and YELLOW for DISABLED.
- **6mm Pushbuttons (Start, Stop, Reset)** – These buttons allow the user to control the system. The Start button uses an interrupt to enable the system, the Stop button disables it, and the Reset button clears the ERROR state once water is restored.
- **LCD1602 Display** – Displays the current temperature and humidity values every 60 seconds. In case of an error (such as low water), it displays an appropriate message.
- **DS1307 RTC Module** – This Real-Time Clock module provides accurate timekeeping for the system. It timestamps every state transition and motor event, outputting the data to the Serial monitor.
- **DHT11 Temperature and Humidity Sensor** – Provides real-time readings of the surrounding environment, which are used to control when the fan should turn on or off.
- **Water Level Sensor** – Detects whether the water reservoir is at a sufficient level. If water is too low, the system transitions to the ERROR state.

- **3–6V DC Motor** – Represents the main cooling fan in the swamp cooler. It activates when the temperature rises above a certain threshold and turns off when the temperature stabilizes.
- **L293D Motor Driver IC** – Powers and controls the DC fan motor. Since speed control is not required, the system uses the enable pin to turn the fan on or off.
- **10k Ω Potentiometer** – Allows user control of the vent direction by adjusting the angle of the stepper motor. The analog input is mapped to the motor's step range for full rotation control.
- **28BYJ-48 Stepper Motor** – Simulates the vent fan angle control. Its position changes based on the potentiometer reading.
- **ULN2003A Stepper Motor Driver Module** – Used to drive and power the stepper motor safely and efficiently.
- **Power Supply Module** – Provides external power to the DC motor and stepper motor, preventing electrical strain on the Arduino board.