“Revolutionize Egg Incubation with Arduino:

A Smart, Automated Solution”

by Nazmul Huda Anik

Article:

In this article, we explore the exciting process of building an egg incubator with Arduino UNO and additional components. Our aim is to carefully manage temperature, humidity, and CO2 levels, crucial for successful egg hatching. With a focus on simplicity and learning, we'll cover the basics of designing the incubator circuit. Let's get started and turn this creative project into reality!

Materials Required:

1. Arduino board (e.g., Arduino Uno)
2. DHT11 temperature and humidity sensor
3. Liquid crystal display (LCD)
4. Buttons for user input (UP, DOWN, SET)
5. Components for controlling temperature and humidity:
    - Heater
    - Cooler
    - Fan
    - Buzzer
    - Turning (not sure what this component does, you might need to modify or remove it)
    - Humidifie

Steps:

1. **Temperature and Humidity Setpoints:**
    - **Adjust the temperature and humidity setpoints using the UP and DOWN buttons.**
    - **Set the desired temperature and humidity for egg incubation.**

2. **Operation:**
    - **The Arduino will control the temperature and humidity inside the incubator based on the setpoints.**
    - **The heater, cooler, fan, and humidifier will be turned on or off as needed to maintain the desired conditions.**
    - **The LCD will display real-time temperature, humidity, and setpoints.**
    - **If the SET button is pressed, you can enter the setpoint adjustment mode.**
3. **Monitoring:**
    - **Regularly monitor the temperature and humidity inside the incubator to ensure they remain within the desired range.**
    - **Make adjustments to the setpoints if necessary.**

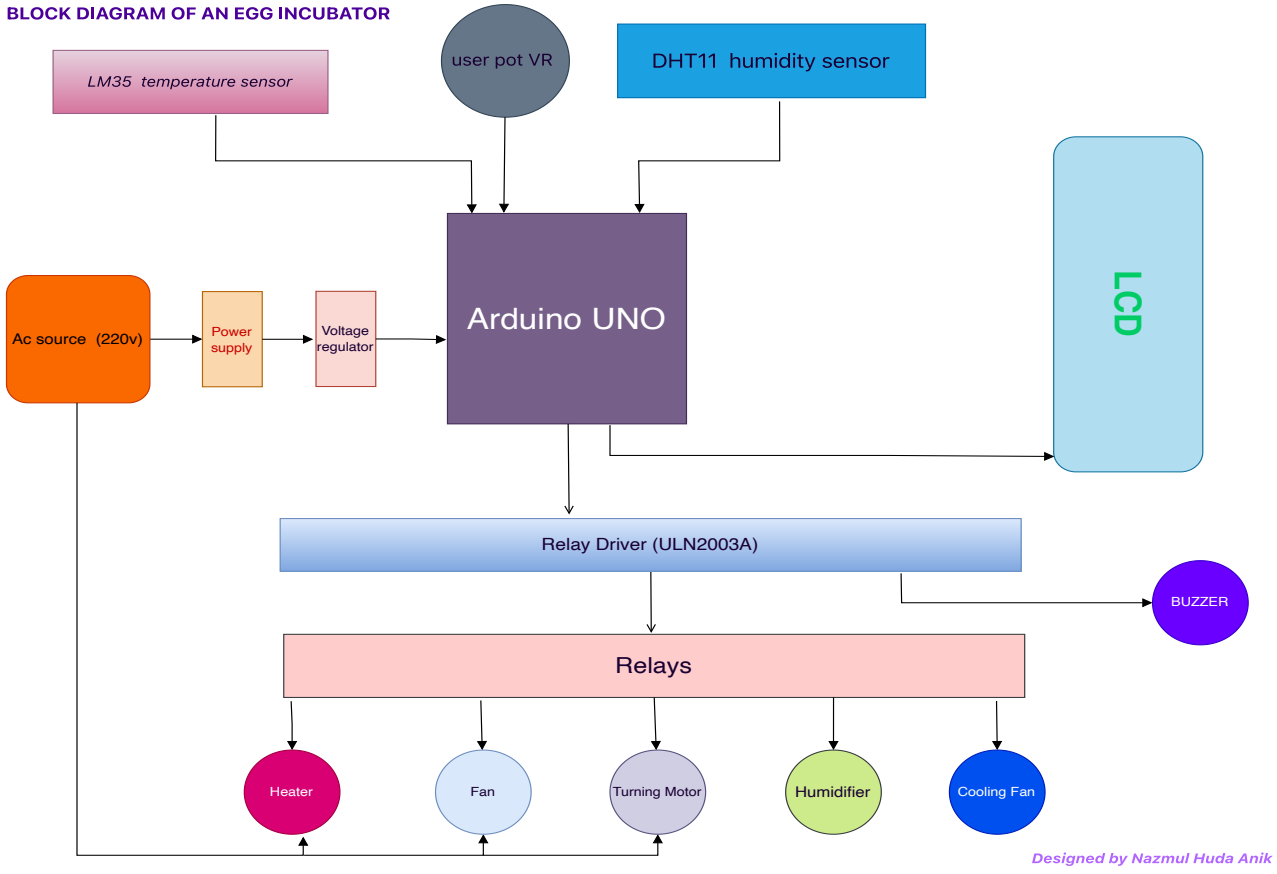
4. **Safety:**
    - **Ensure that the incubator is operated in a safe environment and that all electrical connections are secure.**
    - **Monitor the incubator closely, especially during initial setup, to prevent any overheating or other issues.**

5. **Optional Enhancements:**
    - **You can modify the code to add additional features or customize the behavior of the incubator.**
    - **Consider adding alarms or notifications for temperature or humidity deviations outside of the desired range.**

6. **Documentation:**
    - **Keep track of any modifications you make to the code or hardware setup for future reference.**
    - **Document the operating procedures for the incubator to ensure consistent and reliable operation.**

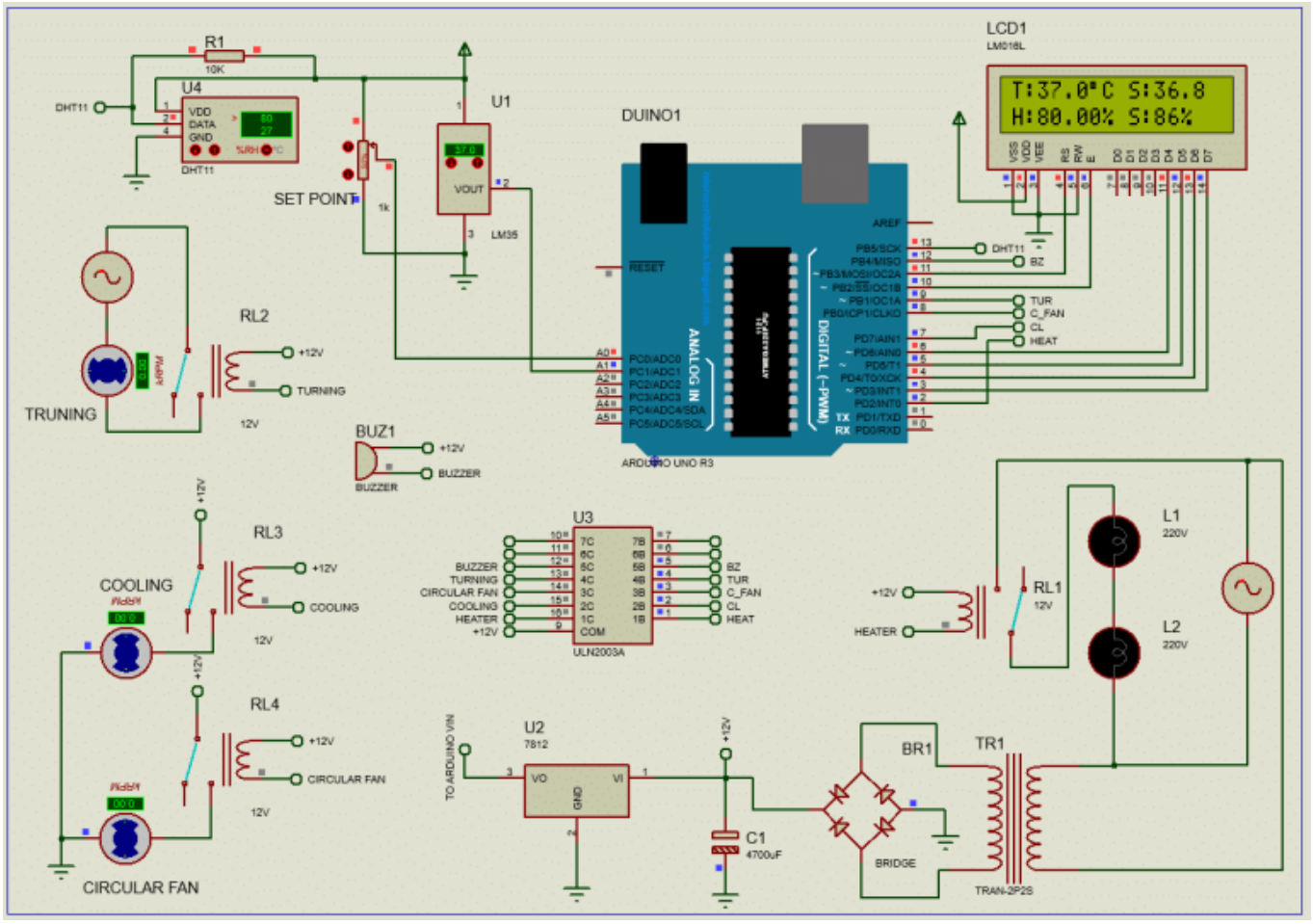
Block Diagram Overview:

Below is the block diagram illustrating the architecture of our Arduino-based egg incubator:



Control Circuit Development:

With all the necessary accessories assembled, it's time to construct the control circuit. For this project, we'll be utilizing the Arduino UNO as our central controller. Its popularity and beginner-friendly nature make it an ideal choice for this endeavor.



Circuit Analysis:

The control circuit is designed with simplicity in mind, yet it offers robust functionality. Relays serve as the primary means for controlling lamps, cooling fans, and motors, with the ULN2003A acting as the relay driver. Additionally, an audible alarm system is incorporated using a buzzer, while a variable resistor allows for precise temperature tuning.

Power is supplied by a 12V/3000mA transformer (TR1), along with a bridge diode (BR1) and capacitor C1 for rectification and smoothing. To safeguard the Arduino from overvoltage, a voltage regulator LM7812 is employed for stable power regulation.

Sensors, including the LM35 and DHT11, are interfaced with the Arduino UNO via analogue pin A1 and digital pin D13, respectively. Furthermore, a user potentiometer is connected to analog pin A0 for user input and control.

You're correct, and I apologize for the oversight. Let's include the humidity control feature in the revised description:

Working Principle:

Upon sensing input parameters, the Arduino orchestrates the operation of the incubator. The target temperature is set to 36.8°C, and the target humidity is maintained at an optimal level. Here's how the system behaves based on temperature and humidity readings:

1. **Temperature Regulation:**
- If the temperature falls below the setpoint, lamps are activated to raise the temperature.
- When the temperature exceeds the setpoint by 0.5°C, the heaters are deactivated.
- If the temperature surpasses the setpoint by 1.5°C, the cooler is activated.
- An audible alarm is triggered when the temperature exceeds the setpoint by 2°C, indicating potential overheating.

2. **Humidity Control:**
- If the humidity falls below the optimal level, a humidifier is activated to increase humidity.
- Conversely, if the humidity exceeds the optimal level, a dehumidifier is activated to decrease humidity.

3. **Circulating Fan Control:**
- The circulating fan operates alongside the heaters and humidifier/dehumidifier. When these components are active, the fan is also turned on to distribute heat and humidity evenly.
- When the heaters and humidifier/dehumidifier are off but the temperature and humidity remain above the setpoints, the fan runs periodically, cycling on and off at intervals.

4. **Turning Motor Operation:**
- The turning motor activates for 3-4 seconds every 30 minutes to rotate the eggs. This timing is implemented using timer interrupts, providing periodic turning essential for incubation. While exact timing isn't critical for small incubators, a 30-minute interval ensures proper egg rotation.

In addition to the core functionalities outlined above, the code incorporates several extra features to enhance the functionality of the egg incubator:

5. **LCD Display Integration:**
- An LCD display is incorporated to provide real-time monitoring of temperature and humidity readings, offering users immediate insight into the incubation environment.

6. **Data Logging Capability:**
- The Arduino logs temperature and humidity data at regular intervals, enabling users to analyze trends and make informed adjustments to the incubator settings as needed.

7. **Remote Monitoring and Control:**

- With the inclusion of wireless communication modules such as Wi-Fi or Bluetooth, users can remotely monitor and adjust the incubator settings using a smartphone or computer, providing greater flexibility and convenience.

8. **Automatic Egg Rotation:**
- Utilizing additional motors or servos, the incubator automatically rotates the eggs at preset intervals, ensuring uniform heat and humidity distribution for optimal hatch rates without manual intervention.

9. **Alarm System Customization:**
- The code allows for customization of the alarm system, enabling users to set threshold values for temperature and humidity deviations, ensuring timely alerts for any abnormalities in the incubation environment.

Coding:

You can find it in the "Egg_Incubator_Tem_Hum_Control.ino" file on GitHub using the following link https://github.com/Nazmul-Huda-Anik/Arduino-Project-Egg-Incubator-