

# **Weather Monitoring Project**

**Jordan Alcira, Casey Giordano, Alis Craig**

**CDA3631 Embedded Operating Systems**

**Hoan Ngo**

## **Introduction**

For our final project, we selected the task of creating a weather monitoring system. Hypothetically, the system would be able to detect warmth and light in its environment and be able to identify what weather pattern matched the data. When the trigger button would be pushed, the input would be received through a breadboard setup linked to a Nucleo-F446RE board, with code being run through STM32CubeIDE on a laptop. The weather conditions would then display on an LCD attached to a breadboard.

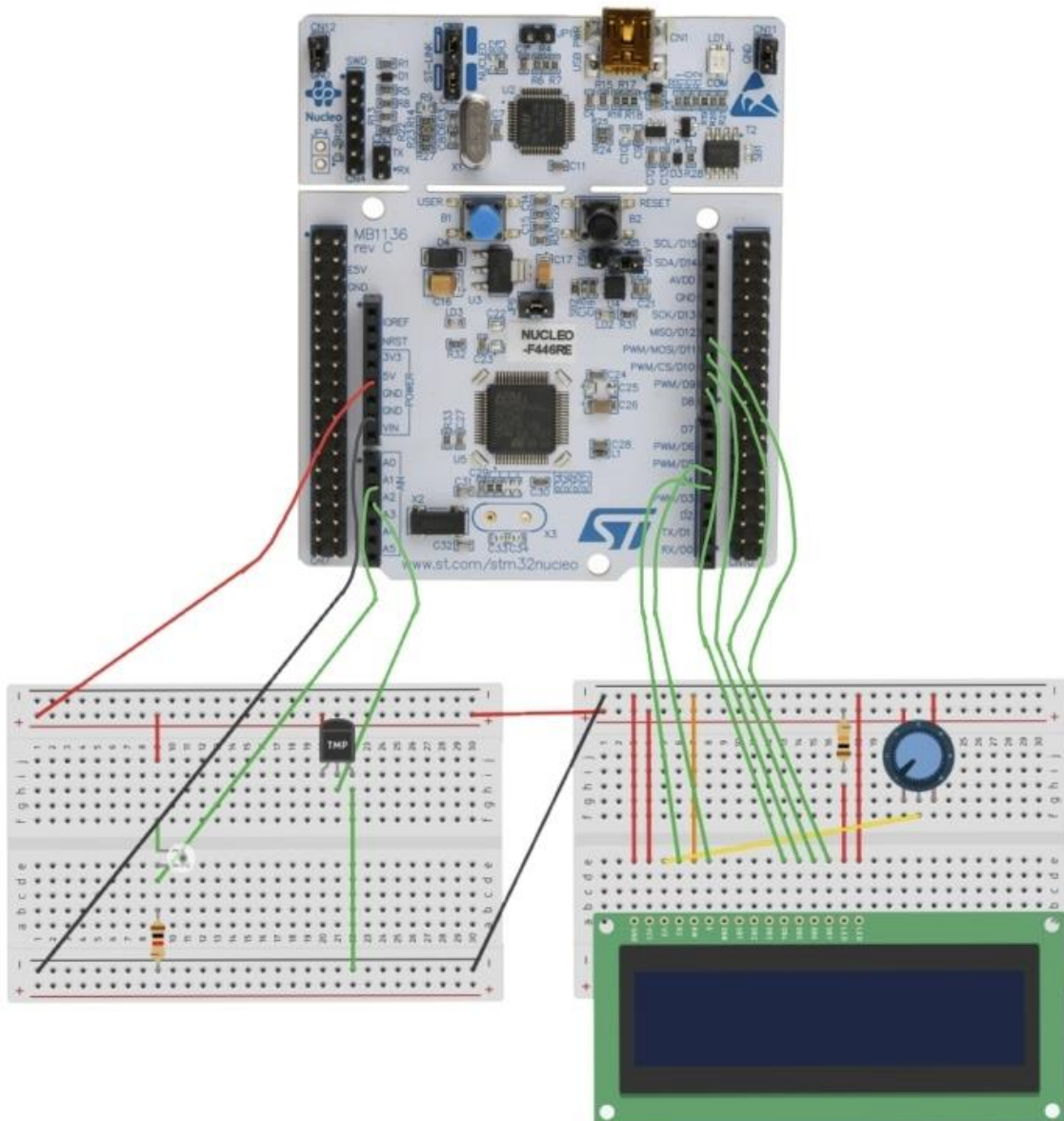
## **Materials**

- 1 LCD Screen 16x2 Characters
- 1  $1\text{ k}\pm 5\%$  (J) Resistor
- 1  $10\text{ k}\pm 5\%$  (J) Resistor
- 1 Nucleo-F446RE board
- USBC to USB3.0 Nucleo board adapter cable
- 1 Laptop with USB 3.0 slot
- 2 400-hole breadboards
- 1 Potentiometer
- 1 Phototransistor
- 1 TMP36 (Temperature Sensor)
- 22 Wires

## **Methods**

We used two different resistors in order to not burn out the heat sensor, photo-resistor, and potentiometer. In order to achieve the desired results, we combined the use of the photo resistor and heat sensor. The photo resistor was placed to detect the light (or absence of light) from the sun and would use this measurement to determine the current cloud coverage. To account for readings at night, the code uses the current time to decide whether the change in light is from lack of daylight. The temperature sensor was to help detect the current temperature outside. To get both the values for the photo-resistor and temperature sensor, the voltage value was measured using ADC1 in STM32. The voltage values were then translated into readable data. The results of both would then show up on the LCD display after a push of the (built-in) button in the Nucleo board. The purpose of the potentiometer was to change the brightness of the characters on the LCD display.

## Wiring Diagram





## **Results and Discussion**

Pressing the blue button on the Nucleo board causes the board to actively read the photo resistor and heat sensor and then display the findings on the LCD display. First, the light levels will cause an output of “Cloud Coverage: ” and either “Overcast,” “Cloudy,” “Clear Skies,” or “Night.” After a couple of moments, the temperature will display on the LCD. The text then clears. The potentiometer may be twisted to change the brightness of the LCD display.

Originally, we intended to use multiple photoresistors, but we ended up using only one. This worked out pretty well. Additionally, we were having garbage numbers appear on the LCD for a while and searched for bugs in the code, but eventually we realized it was a wiring issue, which we then fixed. Adjusting the wiring resolved the issue, and from then on, everything worked perfectly.

## **Conclusion**

We learned much throughout this project. For starters, we found that a potentiometer should be used alongside an LCD, or the characters will only show up as bright squares. The potentiometer allows the contrast/brightness to be adjusted. We also, through experimentation, learned how to properly wire and use a heat sensor and a photo-resistor for the first time during this course. All in all, this project allowed us an opportunity to discover how to use several useful components that will benefit us in future courses. For future projects, we would like to delve further into using task priorities to improve the resource management of the project.

## **Links**

<https://github.com/alciraj2022/weather-monitoring-system>

## **Acknowledgement**

Casey Giordano contributed by collaborating on the code of the project as well as troubleshooting the physical makeup of the breadboard setup. Jordan Alcira drafted/streamlined code and troubleshooted bugs, as well as toggled aspects of the physical layout. Alis Craig researched components of the project and reviewed the physical components’ layout and created the digital wiring diagram.

## **References**

- [1] K. Sahil, "Arduino Tutorial: LCD Display," *Arduino Project Hub*, 2020. [Online]. Available: <https://projecthub.arduino.cc/khushisahil36/arduino-tutorial-lcd-display-b8285a>
- [2] Instructables, "How to use a photoresistor or photocell – Arduino Tutorial," *Instructables Circuits*, 2015. [Online]. Available: <https://www.instructables.com/How-to-use-a-photoresistor-or-photocell-Arduino-Tu/>
- [3] Arduino, "The Temperature Sensor," *Arduino Sensor Kit Documentation*, 2020. [Online]. Available: <https://sensorkit.arduino.cc/sensorkit/module/lessons/lesson/08-the-temperature-sensor>
- [4] R. Santos, "Electronics Basics: How a Potentiometer Works," *Random Nerd Tutorials*, 2019. [Online]. Available: <https://randomnerdtutorials.com/electronics-basics-how-a-potentiometer-works/>
- [5] Arrow Electronics, "Resistor Color Code," *Arrow Electronics Articles*, 2021. [Online]. Available: <https://www.arrow.com/en/research-and-events/articles/resistor-color-code>