Building an Arduino Mini Weather Station

Su Wai Tin, Matthew Dylan Bendebel

Computer Science 207

Trevor Tomesh

April 10th, 2020

Table of Contents

Introduction

The project undertaken for this term was the construction of an Arduino-based Mini Weather Station. This project was a way of culminating the CS 207 - Building Interactive Gadgets program which first introduced us to the building and controlling of interactive devices for do-it-yourself projects. Furthermore, this project will moreover serve as a way of culminating our knowledge about sensors: how they function and how they are used.

A weather station is simply a tool that is able to read weather datas such as the air temperature, the relative humidity of the air, the atmospheric pressure, and presence of rain. The units of measurement for the weather datas will be as follows: temperature will be measured in Celsius units, relative humidity of the air will be measured by a percentage ("100-percent represents the air being at its full capacity of moisture, and 0-percent represents air that does not have any moisture in it"), and air pressure will be measured in hectopascal units (hPa). From this data, we will be able to predict the imminent weather forecast by calculating using pre-existing weather forecasting algorithms.

There are a few makers out there of weather stations that are far more accurate and are far more advanced in weather data collection. As an example, these weather stations are able to collect wind speed and wind direction, gather amounts of rainfall, and some can also be integrated as a part of a smart home. Despite these advanced features and added functionalities, these fully-fledged weather stations can cost in the hundreds and further into the thousands depending on their features. Using the Arduino and some sensors, we are able to significantly cut down the costs while still retaining the essential features that make up a weather station.

There have been a few changes in the project since the proposal was submitted. This includes the use of a BME280 instead of the proposed BMP280 sensor along with the DHT-11

sensor. The change of sensors resulted in a cost-savings and was for the betterment of the project. By using the BME280 sensor, we were able to combine the features of the BMP280 sensor and the DHT-11 sensor into one package. Added, the change saved real-estate space on the breadboard allowing for a more compact project. Furthermore, another change was in the LCD screen size, our group could not find the proposed 1.44" screen online and opted to use the more commonly available 1.8" screen. This change allowed for more data to be displayed on the screen since it was overall bigger.

Inspirations

Before we had decided on settling on an idea for our project, we were originally thinking of doing a rain sensor alarm. The rain sensor alarm would basically detect if it was raining outside, and if it were, a sound would be produced that would alarm people that it is indeed raining. However, we wanted to build on the idea of harnessing the weather and reading off weather data. As a result of our further research, we decided on ultimately doing an Arduino-based mini weather station project and adding the rain sensor onto it. This decision was made so that we could further improve ourselves in our knowledge of the building of interactive gadgets and hardware.

Added, by making our own weather stations, it becomes convenient for us to check the weather even if we do not have internet access. We can easily look at the LCD screen and know the weather without looking at the website or going on a mobile phone.

There are two of us in the group. One of us in the group is a beginner in computer science and the other is an intermediate. We chose this project because it is both easy to understand as a

beginner, while still maintaining an attainable challenge for both of us to do. Added, the project was posted on the "**create.arduino.cc**" website which we both found super useful as there is proper documentation for the project and there were also comments from other people who have also tried the project too. Most importantly, the author of the project is accommodating with any questions towards the project build.

Building Process

To start off with the project, we used two breadboards instead of one because we need a decent amount of breadboard real-estate since the components for this project are fairly sizeable. From here, we plot the 1.8" TFT ST7735 LCD display, the BME280 sensor, the raindrops module and raindrops sensor , and one push-button on the breadboard. We connect these components from the breadboard to the Arduino Uno R3 by using jumper wires. In total, there were 22 connections being powered by the Arduino and because of the numerous amounts of connections, it made quite a mess on the breadboard especially if without any proper cable management.

For the rain sensor of this project there are two components, the raindrops sensor itself and the raindrops module board. The raindrops module board has 4 pins, AO (analog output) pin, DO (digital output) pin, GND (ground) pin, and VCC+ (5v power) pin. We will only use 3 pins, these are VCC+, GND, and the AO pin which is connected to analog pin A3. There are two pins from the raindrops module board which we will connect to the raindrops sensor. The raindrop sensor's board surface is coated with numerous lines of nickel. This sensor works on the principle of resistance. The sensor has more resistance when it is dry, and less resistance when it

is wet. The Arduino can know if it is raining or not based on the amount of resistance being detected by the raindrops module board. [1]

For the other weather sensor, we use the BME280 sensor. The BME280 can read air temperature, air pressure, and air humidity. The BME280 has 4 pins which are the SDA (Serial Data) pin, SCL (Serial Clock) pin, GND (ground) pin, and VIN/VCC+ (5v power) pin. The sensor has to be soldered onto the seller's provided pin male header otherwise there will not be a decent soldered connection made and therefore the Arduino cannot read from the sensor. Added, the sensor should not be touching the breadboard otherwise the readings from the sensor will not be accurate.

We added a reset push-button to our weather station. Its purpose is so that we do not have to rely on the Arduino Uno's built-in reset button on the board. This makes it convenient for resetting the weather station when there is an error or if we want an update on the imminent weather forecast.

As for the sketch code, we had to add many libraries and were required to use different library functions so that we could communicate with both the board and the sensor. Added, we also had to tweak the forecast algorithm we used, the Zambretti Forecaster, to fit with our code. As a result, we had to write numerous amounts of code, especially when writing code to print to the LCD screen. Furthermore, the use of numerous libraries made our code use up a lot of memory.

Designing Process

There was not much of any design process or principle that we did to the mini weather station aside from making sure that the components were not touching each other so that there would not be any mishaps such as an electrical short or a misreading from the sensors. We were not able to print an enclosure as proposed to fit our weather station due to recent events making 3D-printing services in the library unavailable.

If we were able to 3D-print one, we would have gladly gone that route but since we were not able to, we will have to make do with what is available. However, we still wished we were able to 3D-print an enclosure because doing so would fit the completed mini weather station into a presentable and compact package.

User Manual for Mini Weather Station

The initial setup fro the Arduino-based mini weather station can be a challenge for some due to the numerous connections made, but there are helpful breadboard and circuit schematics made using Fritzing provided on the Github repository which can be found in Appendix C. Despite this, after its initial setup, using the weather station is extremely easy to use and read, and requires no set up besides placing it in its extended location. Using the mini weather station, most people will surely find that it is simple and convenient.

The first step would be to download the code from the Github repository and upload it onto an Arduino Uno. The Arduino Uno supplies 5V of power to the breadboard rails which will then be connected from and used to power the LCD display, sensor, and raindrops module board.

After that, you have to start set up your hardware as shown on the circuit schematic and breadboard diagram on Appendix A and Appendix B. You have to connect the wires correctly with your Arduino, breadboard, push button and sensor or else it will not function as intended. Everything should work as intended especially if the BME280 sensor pin connections are soldered to the provided pin male headers as it does not come pre-soldered.

As for the raindrops module board, it might take some adjustments or turns on the potentiometer on the board to adjust to your desired threshold. By turning the potentiometer thus setting the threshold, you adjust the amount of water that should be placed on the raindrops sensor board for the reading to trigger a yes or no.

We suggest setting the mini weather station on a wall just as if it was a house thermostat. By doing so, people will not have to look down on the screen for them to read the weather readings and forecast. You can also place this right outside an outdoor door. However, we would advise on not using this under extreme temperatures such as too cold or too hot weather conditions as this might damage the components and the Arduino itself.

Milestones

We can confidently say the project is a success because we made a mini weather station that could read the air temperature, relative humidity of the air, the atmospheric pressure, and use these data points to predict a weather forecast all the while displaying this to an LCD screen. The main goal was achieved and so was the extended goal which was to predict the weather forecast. It was a challenge making the code but we pulled through.

We had some milestones not reached on time. One milestone did not reach was the making of a 3D-printed custom made enclosure for the mini weather station. This was because we needed a 3D printer for the enclosure and the only accessible one was from the library. Because of the suspension of library services, we decided to opt out on this part of the proposal.

Added, another thing that did not really align with our milestones in terms of dates was the milestone for getting the supplies and components for building the project. This was a result of shipping time delays because of the current events around the world. And we also had a mishap in buying the correct components for our project.

I am sure this project is useful for future use beyond being a requirement for CS 207. We can improve with the design by fitting it to a larger LCD screen size and maybe enclose the weather station it into waterproof enclosure so that it will be at least weather. We can build upon the original project and make it useful while keeping the costs low.

Set Backs and Failures

However, the only thing that we cannot make for our project was the 3D-printed custom made enclosure for the mini weather station. This was because we needed a 3D printer for the enclosure and the only accessible one was from the library. Because of the suspension of library services, we decided to opt out on this part of the proposal. As a result, we just made the mini weather station simple and usable despite the lack of an enclosure.

Another failure we had encountered was when we ordered the wrong raindrop sensor kit. Instead of ordering the kit with the raindrop module board and sensor, we accidentally ordered the raindrop sensor kit with a relay. In the end, the relay was ultimately unusable and we had to

reorder one again from Amazon. Shipping time delays were extended and so we had to wait a week before the raindrop sensor kit arrived to our door.

We also had some problems during the project due to current events. Our project is supposed to be a group project but since we cannot meet in person to discuss about the project and we cannot build the project together, this hindered the time it took for our project from start-to-finish and also delayed some of our milestones. Added, there were also some set back when it comes to communication. We cannot communicate very well especially during the building of the project due to different schedules in online learning. Despite the setbacks the group met, we still tried our best, endured, and finished our project.

Group Contribution

Matthew's main contribution to the project is making the sketch code and researching about the hardware and software of the required components. Matthew researched the component's different pin functions and their corresponding component's library functions. Matthew also researched the weather forecasting algorithm, the Zambretti Forecaster, that would be deemed appropriate for the limitations of the project. Matthew also made the presentation video that goes along with the project.

Su Wai's contribution to this project is the writing of the project write-up and of the making of the Github Repository. Ultimately, Su Wai is responsible for the documentation of this project. Added, Su Wai was responsible for researching about the function and background of the rain sensor detector.

Conclusion

The Arduino-based Mini Weather Station was a fun and educational project. It thought us so much about collecting data about the weather and about what goes into weather forecasting. We learned how a new sensor that was not tackled in the class worked, the BME280/BMP280 sensor. Added, we learned to use different libraries and library functions. Most importantly, we learned to do teamwork to complete the project.

Despite the setbacks and failures caused by current events, we were able to push through and endure and finish the project. What is a project if there are no setbacks?

The CS 207: Building Interactive Gadgets program of the University of Regina is a class that inspires people to start hacking, to start being “clever”. In the class description from the Computer Science department, CS 207 is meant to introduce people into do-it-yourself projects and it has exactly achieved its purpose to us. It has taught us to be critical thinkers in doing these projects and coming up with other cool products, and to also work effectively with other people. To work smart, and not necessarily hard.

References

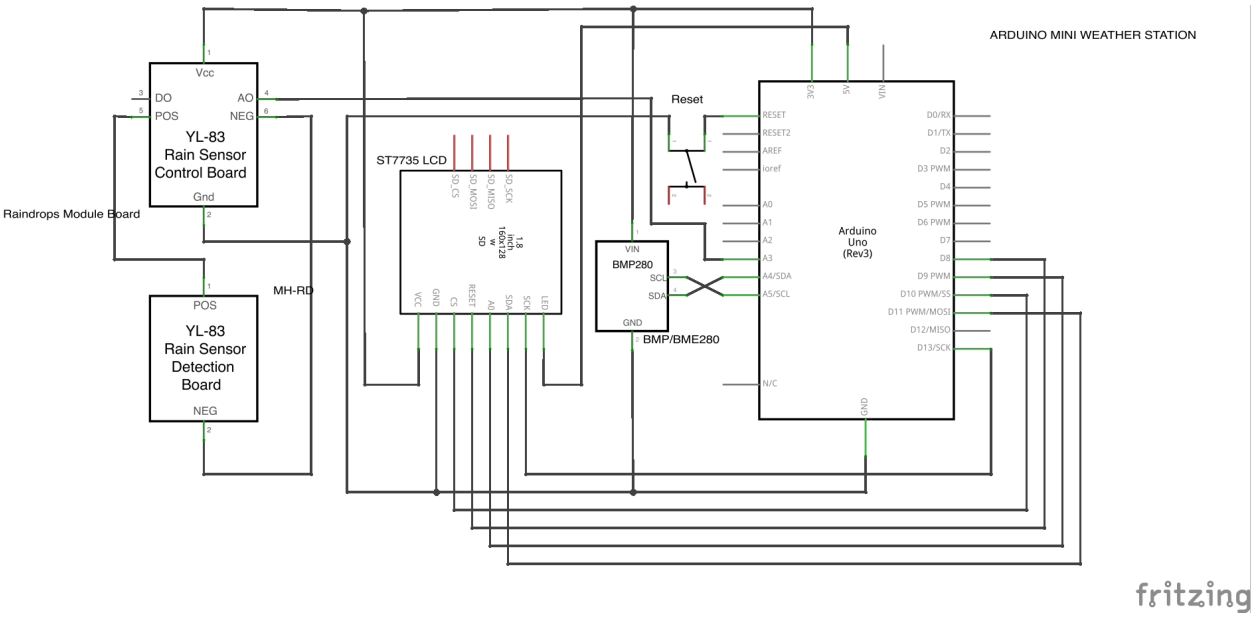
[1] Components101.com. 2019. *Rain Drop Sensor Module Pinout, Datasheet & How To Use It In A Circuit*. [online] Available at:
<https://components101.com/sensors/rain-drop-sensor-module>
[Accessed 5 April 2020].

[2] Bakshi, D., 2020. *Mini Weather Station Using Arduino Nano*. [online] Arduino Project Hub. Available at:
<https://create.arduino.cc/projecthub/dnbakshi07/mini-weather-station-using-arduino-nano-b211fe?fbclid=IwAR1R-0RmYrLAWF0fvP6udtu6RO1YdiKmWeMhqSdZYjTSqjjDTWrpLVpetC8>
[Accessed 5 April 2020].

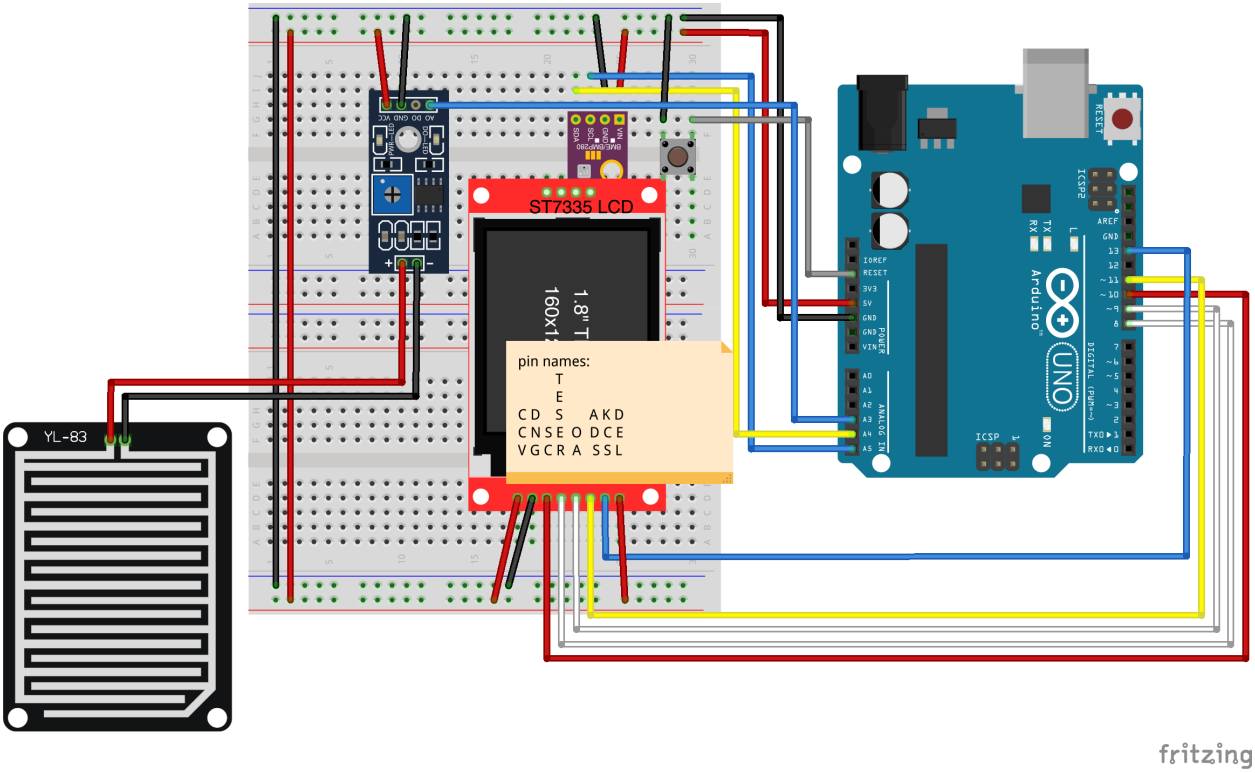
[3] Kumar, M., 2020. Rain Detector Using Arduino And Raindrop Sensor. [online] Circuitdigest.com. Available at:
<https://circuitdigest.com/microcontroller-projects/rain-detector-using-arduino>
[Accessed 6 April 2020].

[4] Mengazi, T. M. T., & Rada, C., 2020. *Weather Forecast Based on Pressure, Temperature and Humidity Only (for implementation in Arduino)*. [online] Available at: <https://earthscience.stackexchange.com/questions/16366/weather-forecast-based-on-pressure-temperature-and-humidity-only-for-implement>
[Accessed 18 March 2020]

Appendix A: Mini Weather Station Circuit Schematic



Appendix B: Mini Weather Station Breadboard



Appendix C: Sketch code

The code that have been used in this project can be found in Github Repository located at

https://github.com/mbendebel/CS207MiniWeatherStation

Appendix D: Final Product