Weather Station

Faisal Halabeya & Jonathan Sun

Abstract

Homemade weather station to measure:

Temp, Pressure, Humidity, Wind Speed

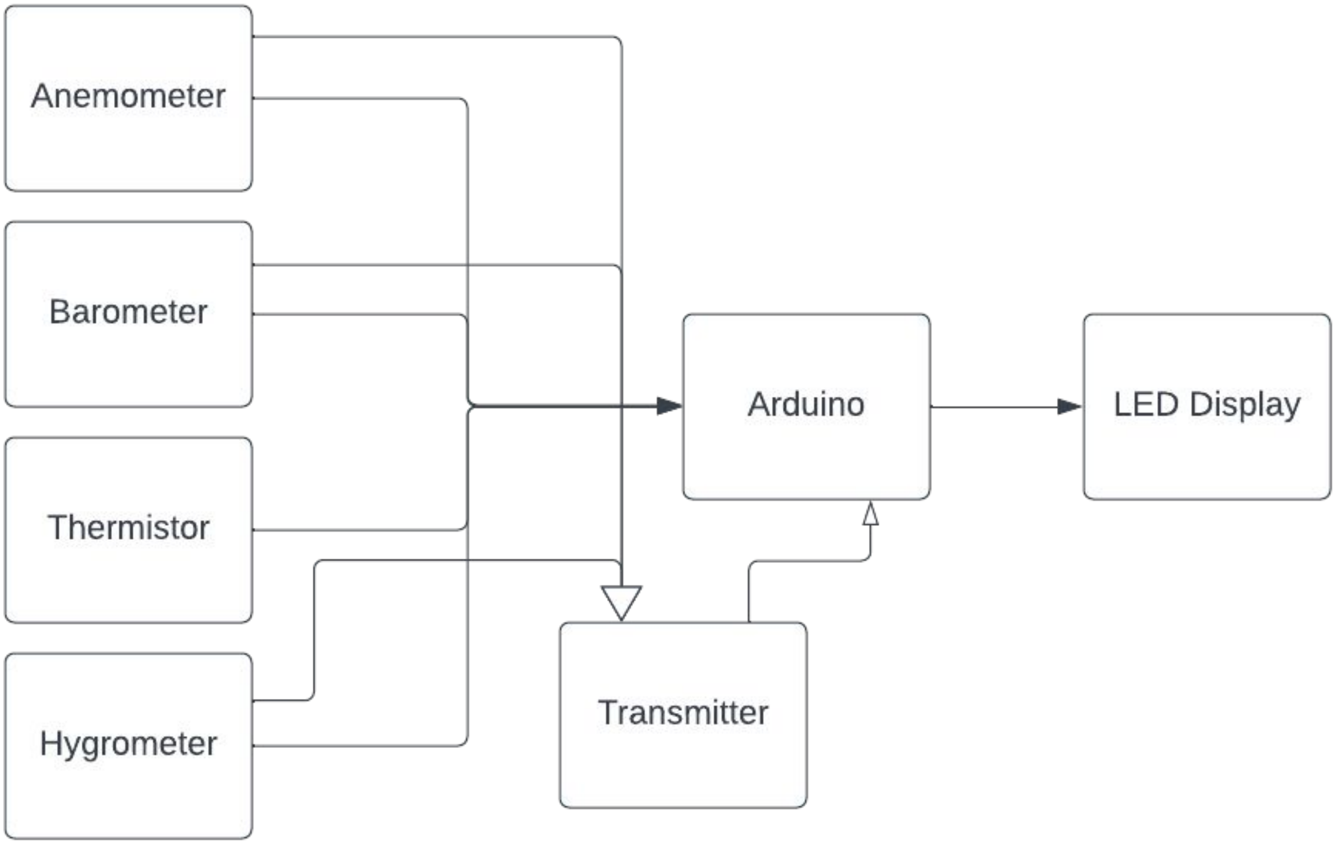
Initial Goal: Display data on LCD screen

Phase 2 Goal: Apply numerical weather prediction (NWP) model to predict weather for next few hours, display on LCD

(Possible Extension: Remote Sensing)

Applications: Control for comfort, health, performance of appliances, plant growth conditions, preservation of art works, etc.

Depending on the location of our weather station we may need to use a transmitter to receive data from our meters.



Parts

Acquired/Present in eLab:

- Arduino UNO
- Thermistor
- Breadboard, Resistors, & Wires
- Barometer (Adafruit BMP280 I2C or SPI): $9.95 (Link)
- Thermometer and Hygrometer (DHT20 - AHT20 Pin Module - I2C): $4.50 (Link)
- Anemometer
- LCD With 4 Lines of Text (20X4): $17.95 (Link)

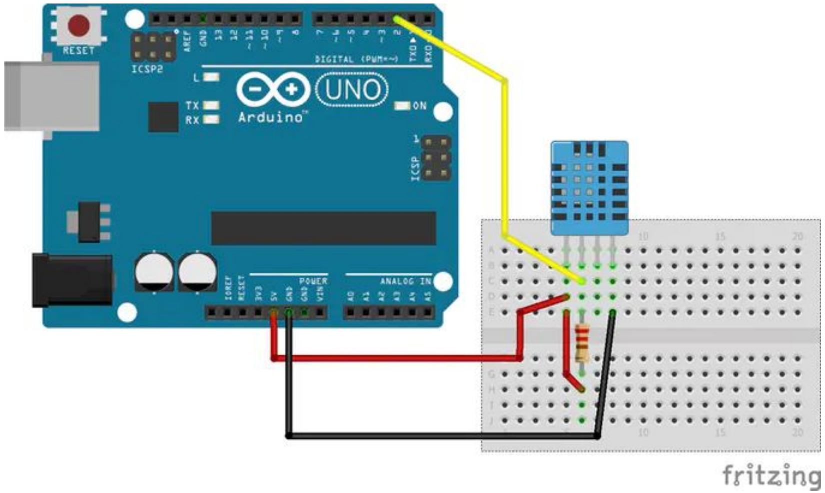
Grand Total: $32.40

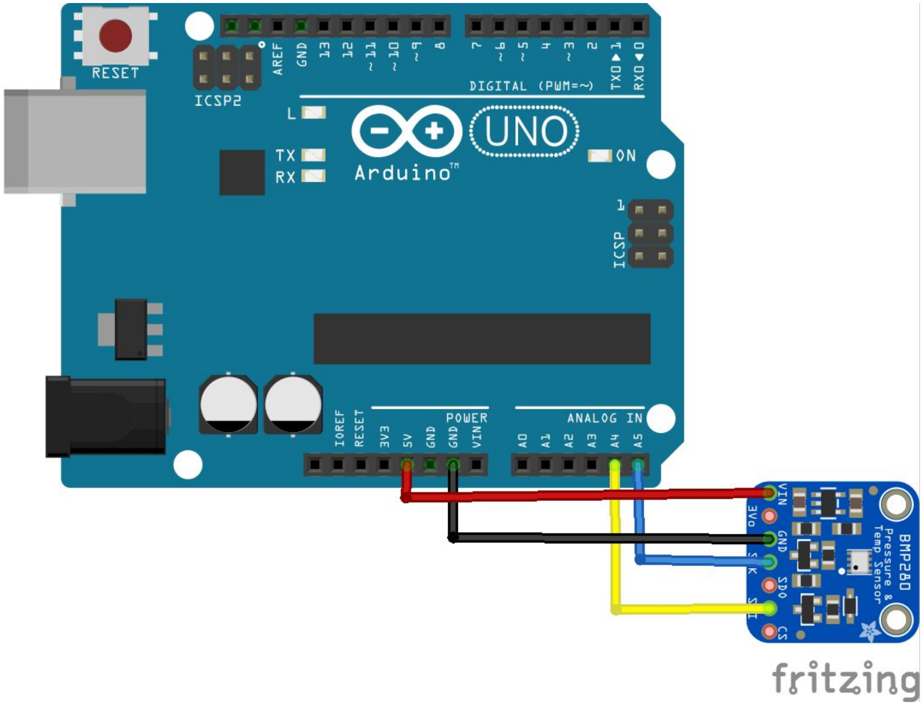
Updates for 4/20

This week we will be:

- Wiring up sensors up to make sure they work as intended
- Drafting code for weather station
- Producing a schematic diagram for how all components fit together

```
Pseudo-Code (loop):

delay(10000); // will take data every 10 seconds

analogRead(sensors); // read temp, pressure, etc.

digitalWrite(LCD); //display current weather
conditions

int num_predict = 0; // we will use numbers to
                       // represent different possible
                       // weather predictions

num_predict = // here we aim to run a NWP algorithm

if num_predict == 1 {     // e.g. 1 is for a sunny
forecast
      digitalWrite(LCD, “Sunny”)       // display
forecast
      }
```


RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX0▶1
RX0◀0
DIGITAL (PWM=~)
L
TX
RX
UNO
Arduino™
ON
ICSP
POWER
IOREF
RESET
3V3
5V
GND
GND
VIN
ANALOG IN
A0
A1
A2
A3
A4
A5
fritzing


RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX0▶1
RX0◀0
DIGITAL (PWM=~)
L
TX
RX
UNO
Arduino™
ON
1
ICSP
POWER
IOREF
RESET
3V3
5V
GND
GND
VIN
ANALOG IN
A0
A1
A2
A3
A4
A5
VIN
3Vo
GND
SCK
SDO
SDI
CS
BMP280
Pressure &
Temp Sensor
fritzing

(Skip) Abstract

We will create a homemade weather station to measure air pressure, temperature, humidity and wind speed. We will then compile data collected by the barometer, thermometer, hygrometer, and anemometer into the Arduino. Our immediate goal is to use the data to give an accurate representation of current weather conditions. The data will be processed and displayed on a simple and user-friendly LED interface.

If time permits, we will apply a numerical weather prediction (NWP) model to predict the weather for the next few hours based on data collected. Prediction will be displayed on LCD.

Our weather station will have indoor practicality since thermometer and hygrometer data can be used to display ambient indoor conditions, allowing users to control for comfort, health and maintenance of indoor appliances. When conditions are too dry or too moist they can directly affect comfort, energy consumption, and bacteria and fungi growth.

Updates for 4/11

Our initial goals for the week were pretty simple, we wanted to make sure we had all of our parts ordered and then also make sure that we were looking at how to start coding our project.

Devices obtained

Last week we were able to confirm with Situ that the lab has a compatible thermistor and hygrometer that we will be able to use. In further note we have also supplied Situ with a LCD display and barometer we would like to acquire for the project as well. Also as advised by Situ, the lab will also be acquiring us an anemometer that is compatible with our arduino. This means all of the parts for our project will be arriving soon.

Final Thoughts

We have also begun looking at code and writing starter code for our project but we don’t have anything substantial to show the class so far. We can expect to show more in the coming weeks and show how we expect our entire project to function.

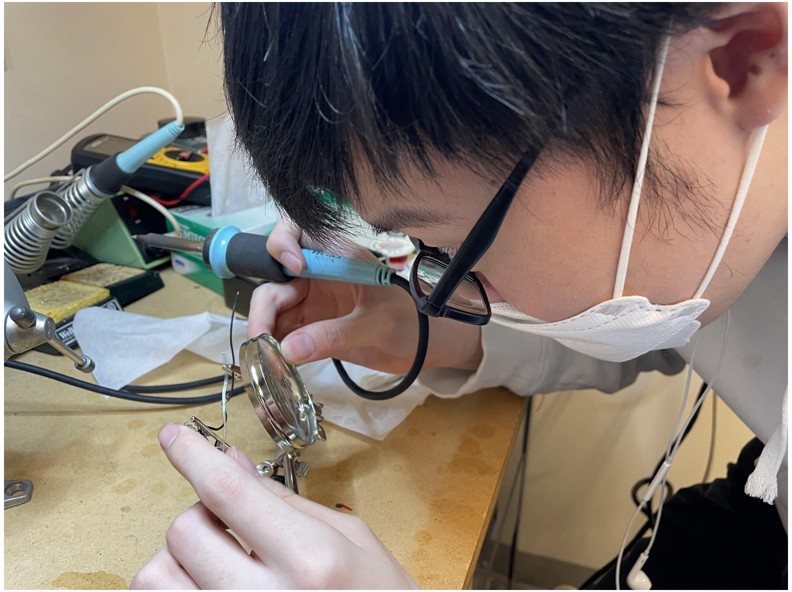
04/24/2022 Update

We managed to get all of the parts working individually, we haven’t tested combining all of the code into one yet as the Arduino only has 2 slots of the I2C bus required for the barometer and the LCD displays, ports A4 & A5. From the research we did, we should be able to wire both devices to the same two pins without any difficulties, but we haven’t tested it yet,

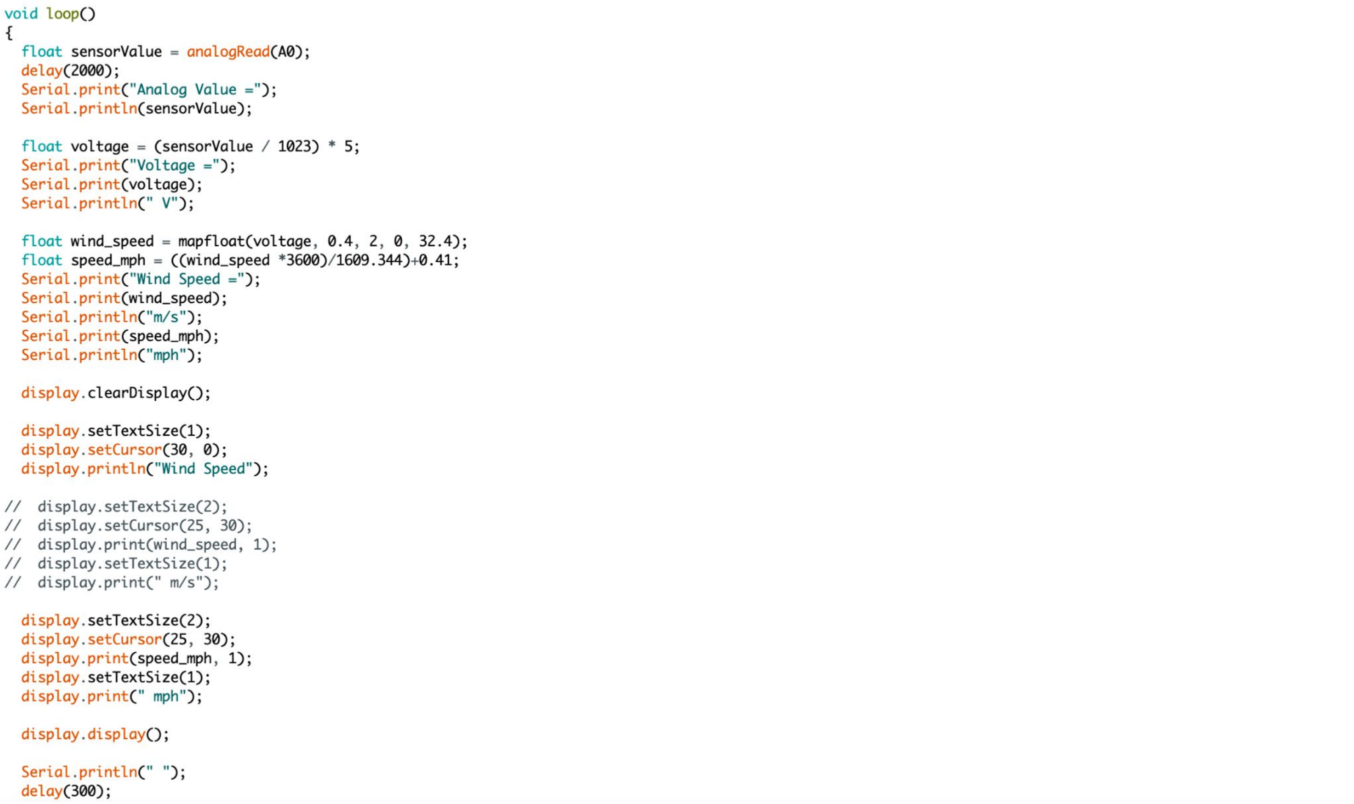
Soldering the Anemometer

One of the highlights of this project was when Faisal and I were taught how to solder the wires for the anemometer.

[Insert photo of Faisal here]

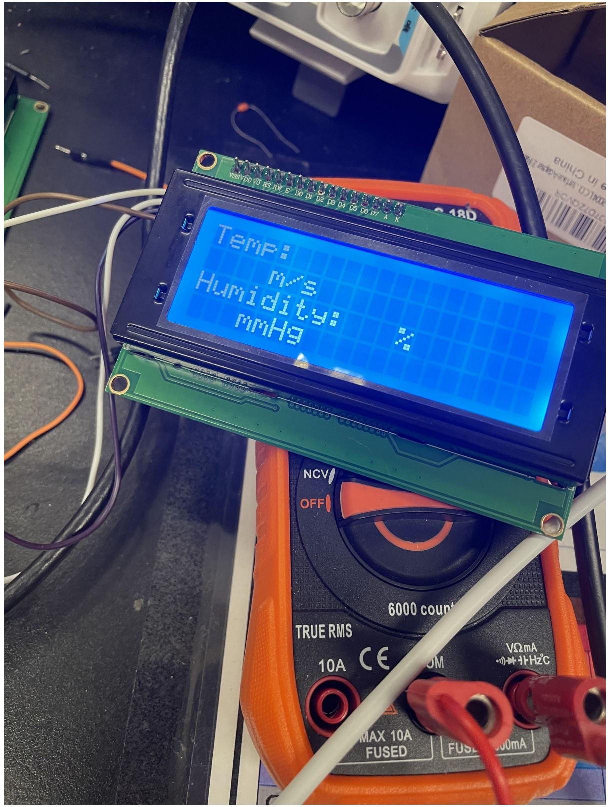


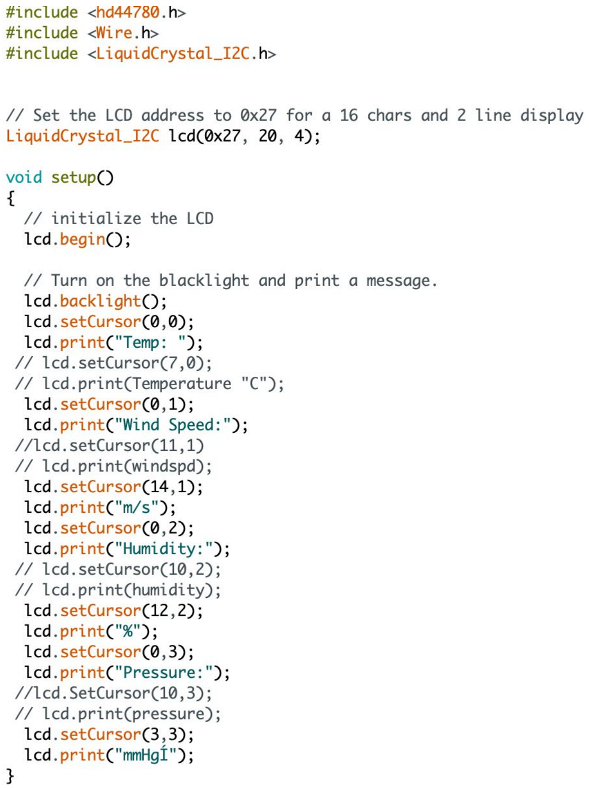
Code: Anemometer



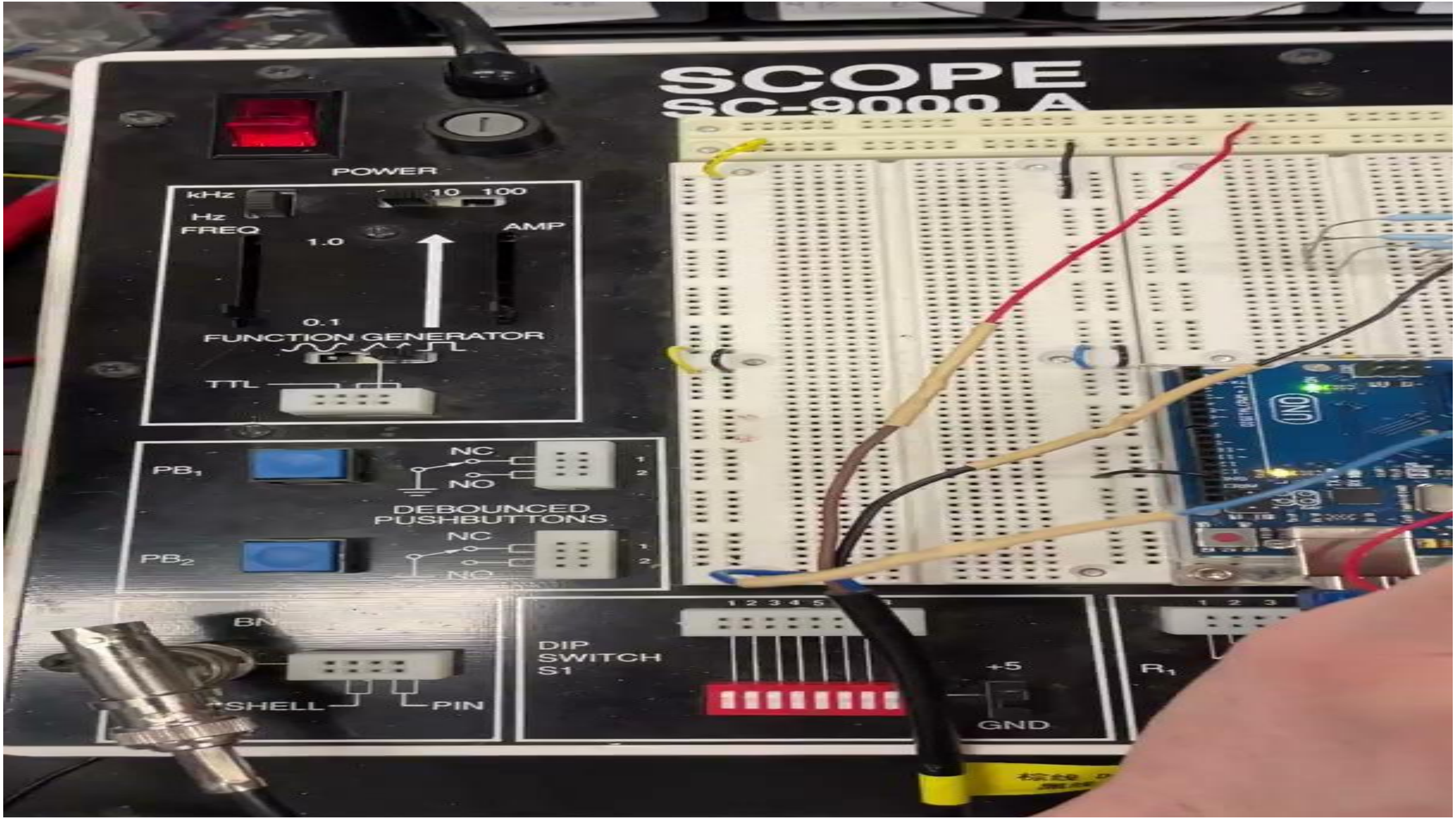
```
void loop()
{
  float sensorValue = analogRead(A0);
  delay(2000);
  Serial.print("Analog Value =");
  Serial.println(sensorValue);

  float voltage = (sensorValue / 1023) * 5;
  Serial.print("Voltage =");
  Serial.print(voltage);
  Serial.println(" V");

  float wind_speed = mapfloat(voltage, 0.4, 2, 0, 32.4);
  float speed_mph = ((wind_speed *3600)/1609.344)+0.41;
  Serial.print("Wind Speed =");
  Serial.print(wind_speed);
  Serial.println("m/s");
  Serial.print(speed_mph);
  Serial.println("mph");

  display.clearDisplay();

  display.setTextSize(1);
  display.setCursor(30, 0);
  display.println("Wind Speed");

//  display.setTextSize(2);
//  display.setCursor(25, 30);
//  display.print(wind_speed, 1);
//  display.setTextSize(1);
//  display.print(" m/s");

  display.setTextSize(2);
  display.setCursor(25, 30);
  display.print(speed_mph, 1);
  display.setTextSize(1);
  display.print(" mph");

  display.display();

  Serial.println(" ");
  delay(300);
```

Working the LCD





```
#include <hd44780.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>


// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 20, 4);

void setup()
{
  // initialize the LCD
  lcd.begin();

  // Turn on the blacklight and print a message.
  lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print("Temp: ");
 // lcd.setCursor(7,0);
 // lcd.print(Temperature "C");
  lcd.setCursor(0,1);
  lcd.print("Wind Speed:");
 //lcd.setCursor(11,1)
 // lcd.print(windspd);
  lcd.setCursor(14,1);
  lcd.print("m/s");
  lcd.setCursor(0,2);
  lcd.print("Humidity:");
 // lcd.setCursor(10,2);
 // lcd.print(humidity);
  lcd.setCursor(12,2);
  lcd.print("%");
  lcd.setCursor(0,3);
  lcd.print("Pressure:");
 //lcd.SetCursor(10,3);
 // lcd.print(pressure);
  lcd.setCursor(3,3);
  lcd.print("mmHgÍ");
}
```


SCOPE
SC-9000 A
POWER
kHz
Hz
10
100
FREQ
1.0
0.1
AMP
FUNCTION GENERATOR
TTL
PB1
NC
NO
1
2
DEBOUNCED
PUSHBUTTONS
PB2
NC
NO
1
2
BN
SHELL
PIN
DIP
SWITCH
S1
1 2 3 4 5
+5
GND
R1
1 2 3
UNO