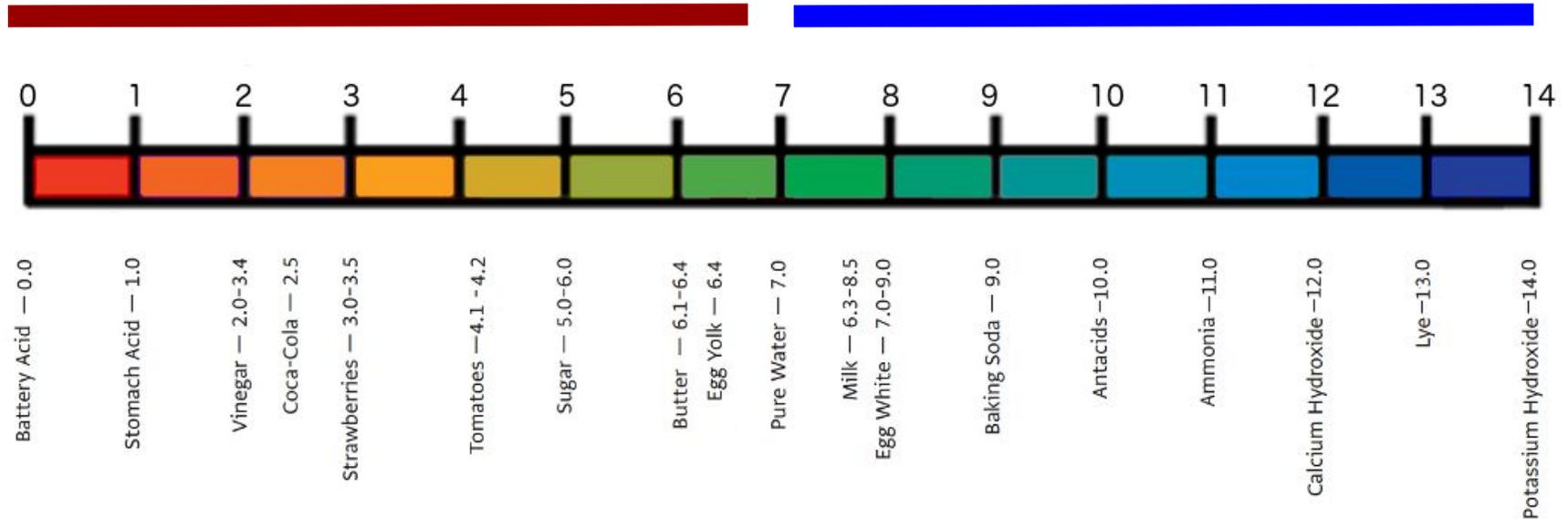


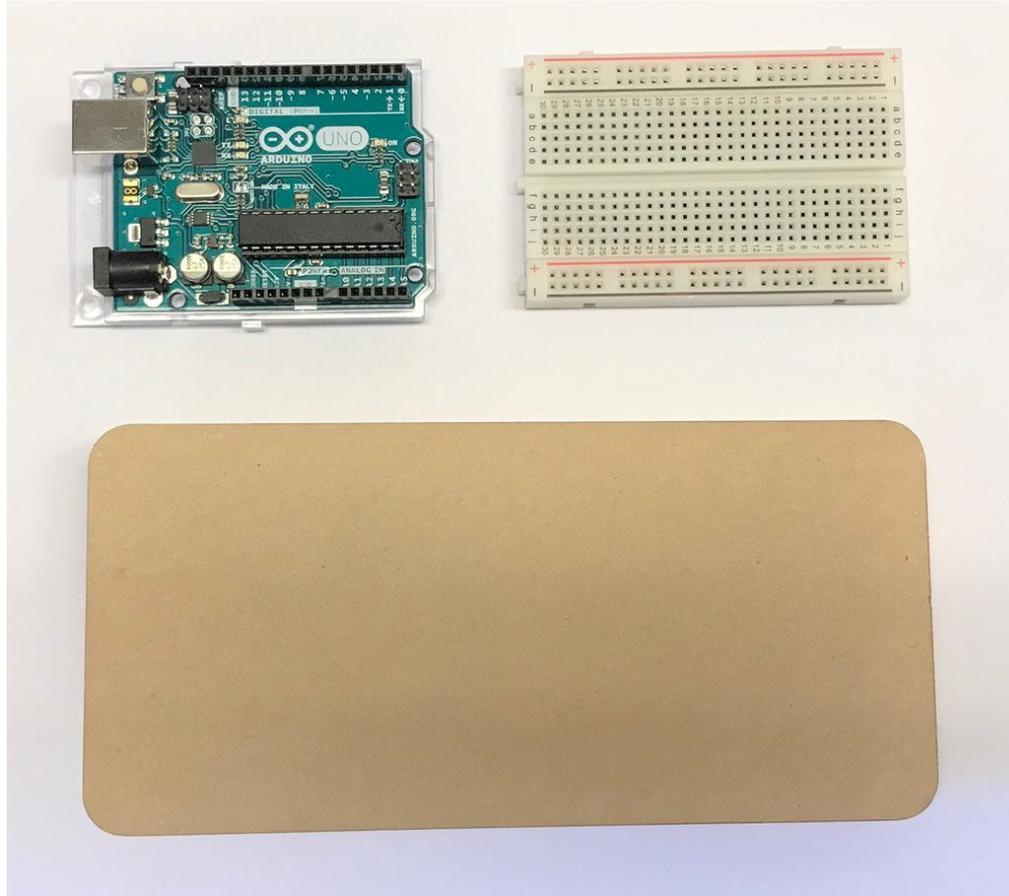
pH Build and Programming

Low

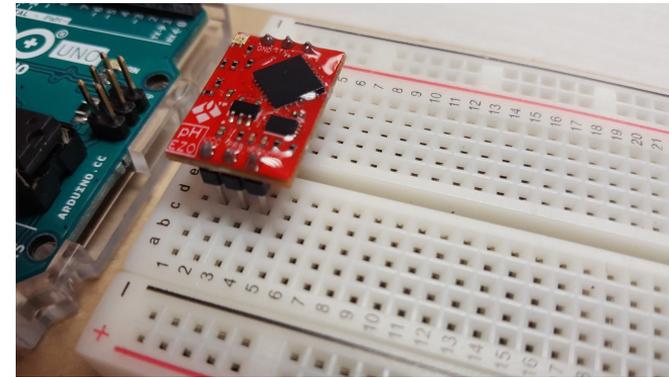
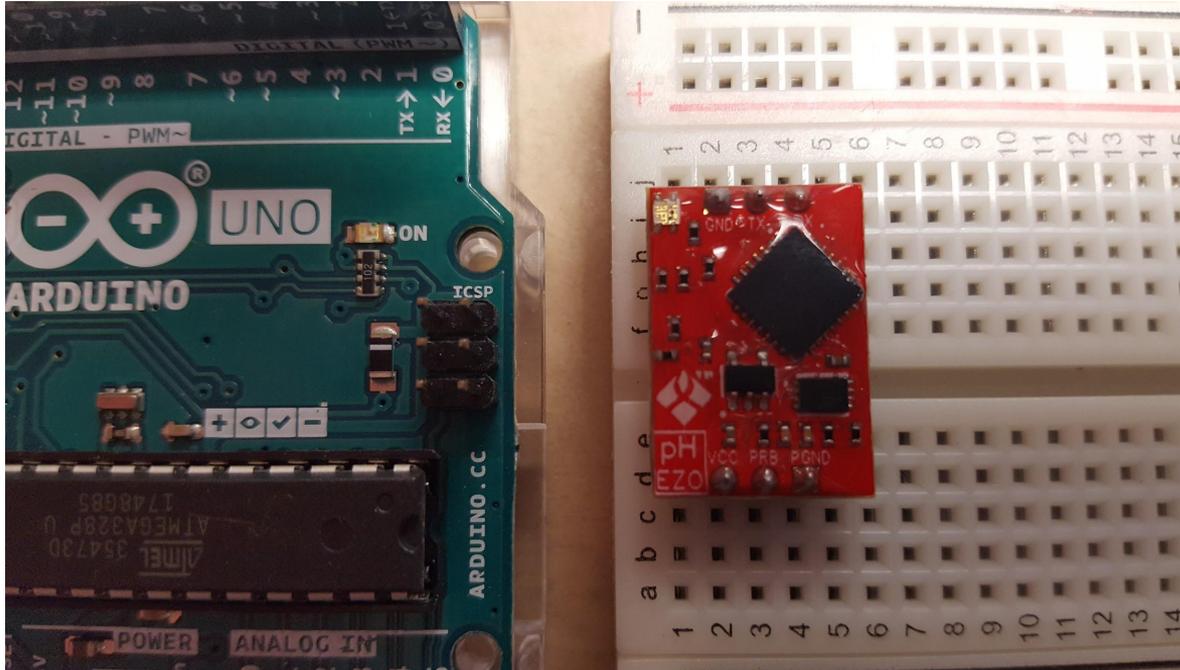
High



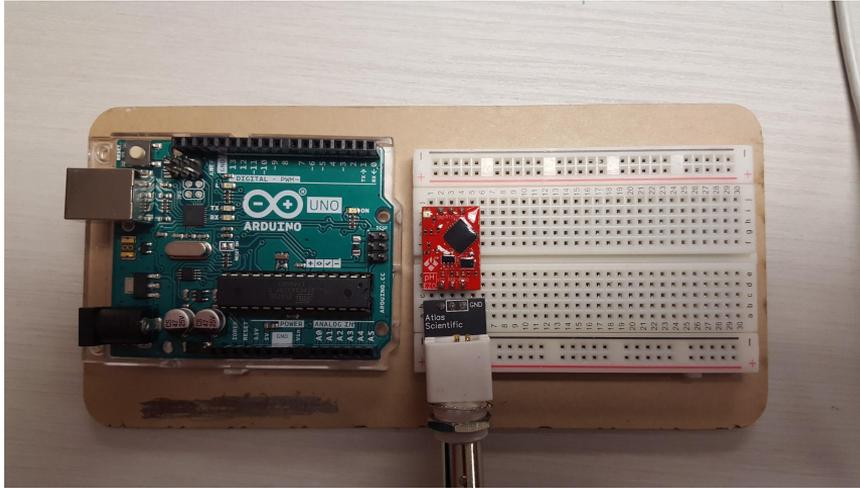
Start with the breadboard, acrylic sheet, and Arduino Uno.



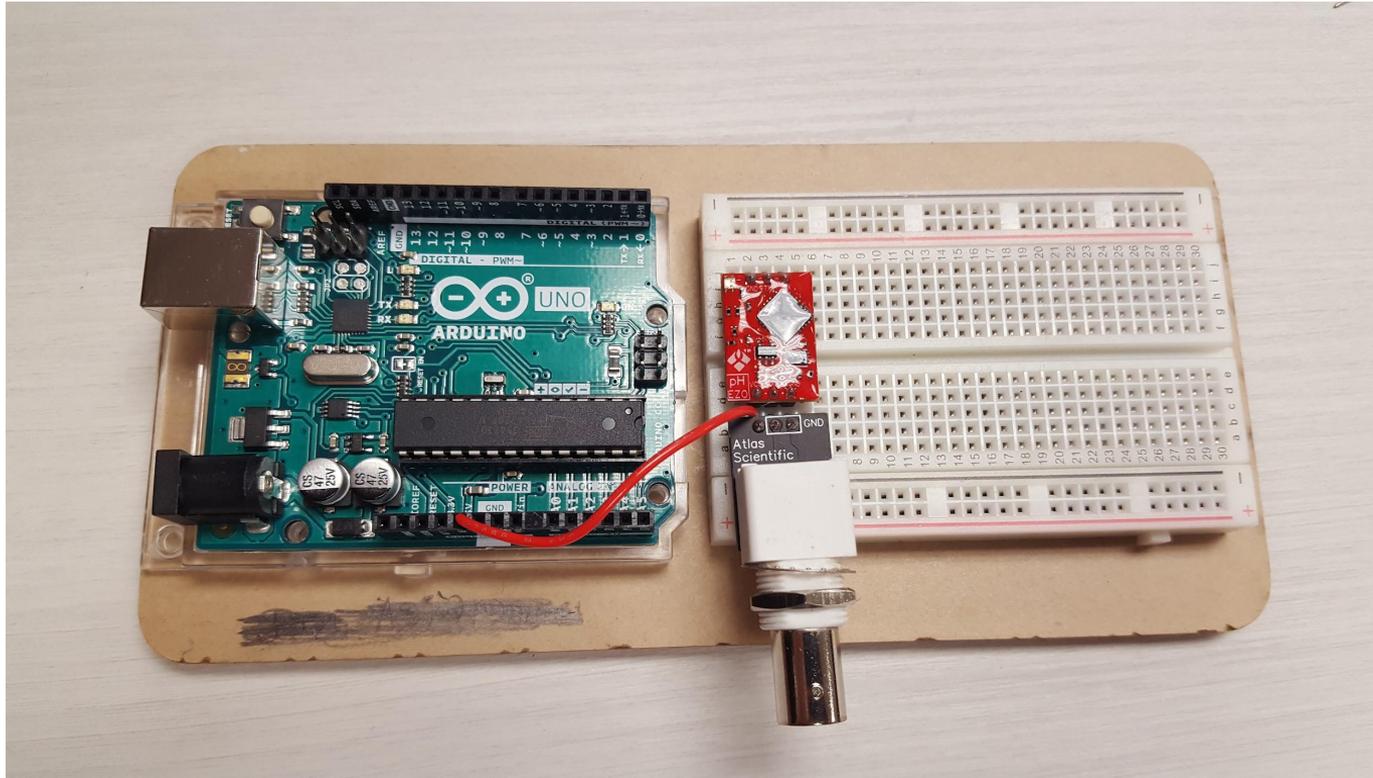
Insert Chip 1 onto the breadboard. You could do that in different ways, but to make things easy use the following: Insert the bottom three pins (with the "pH EZO" label) into 2, 3, 4 on row d. Insert the top three pins into 2,3,4 on row i.



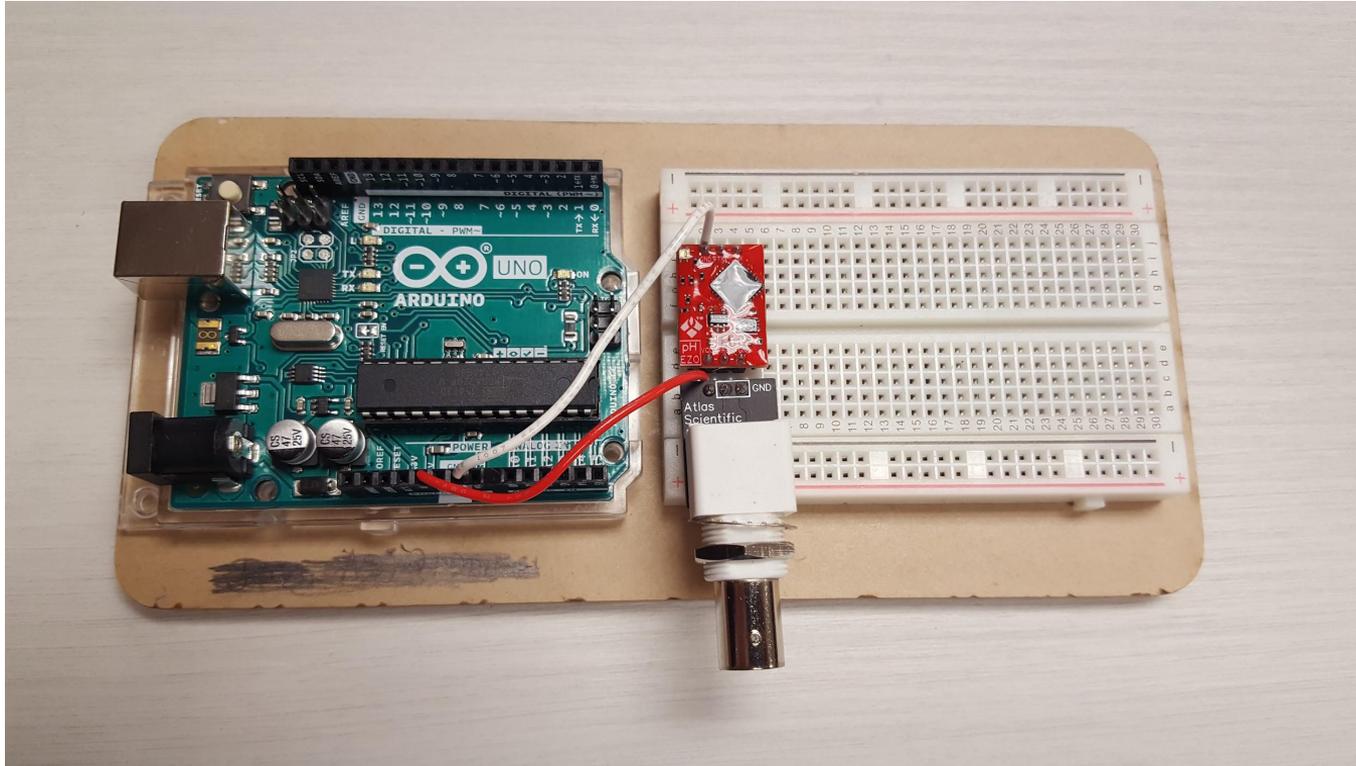
Insert Chip 2 directly below Chip 1, leaving room for a row of wires between the two chips. Check to make sure the pins from both chips line up vertically on the breadboard. That is to say, put the three pins into 2, 3, and 4 on row b.



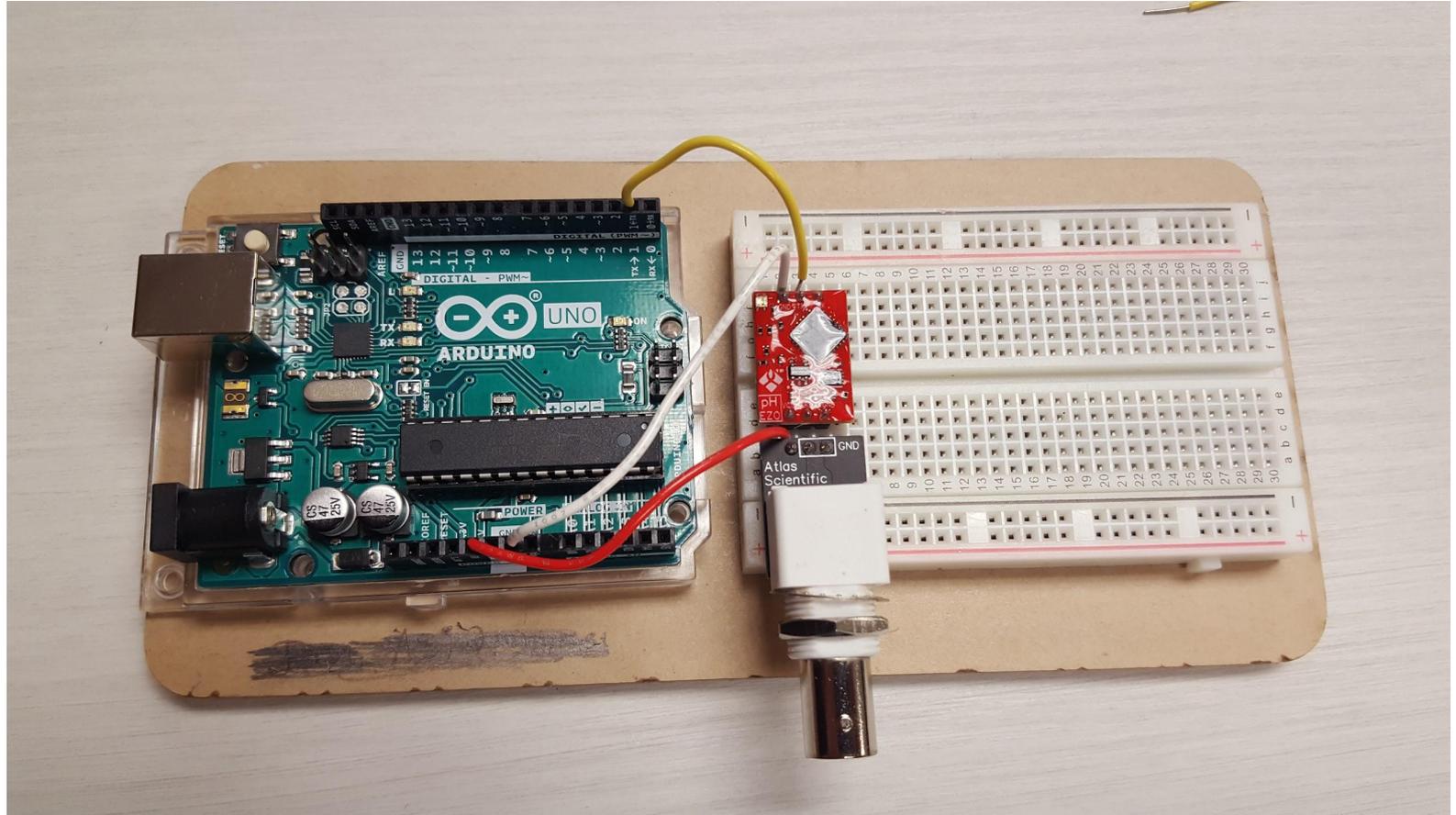
Cut some wire, strip the insulation off of each end, and connect the 5V on the Arduino to the Vcc on the breadboard (column 2, row c).



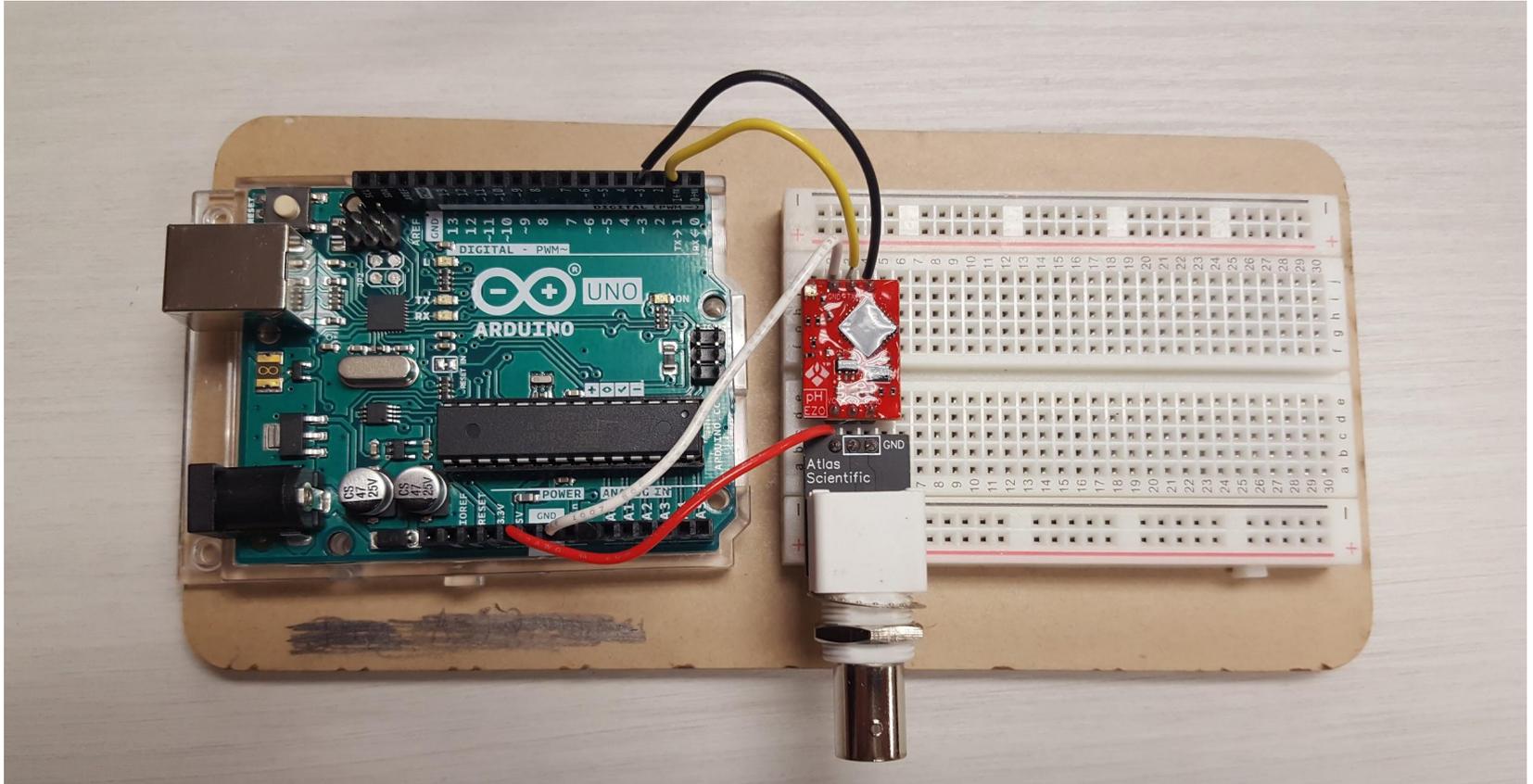
Cut some wire, strip the insulation off of each end, and connect the ground (GND input) on the Arduino to the ground on the breadboard (column 2, row j).



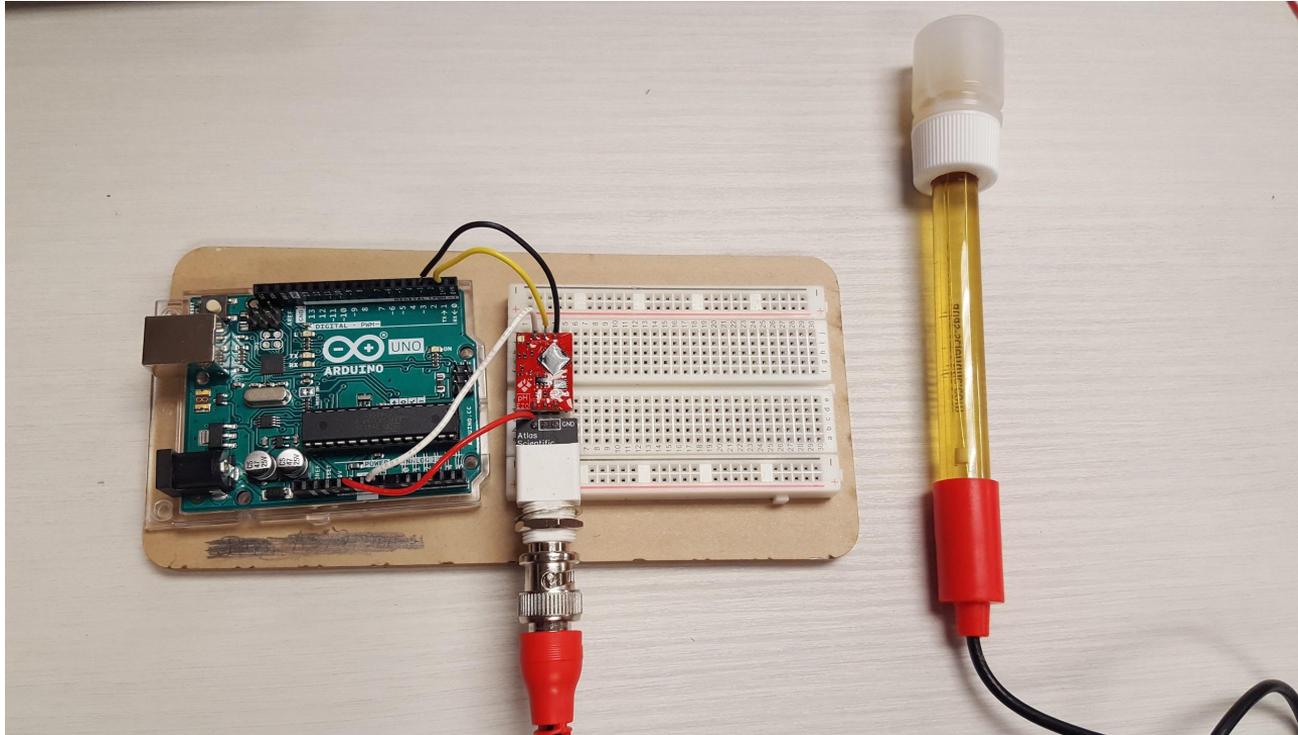
Cut some wire, strip the insulation off of each end, and connect input 2 on the Arduino to the Tx on the breadboard (column 3, row j)



Cut some wire, strip the insulation off of each end, and connect input 3 on the Arduino to the Rx on the breadboard (column 4, row j).



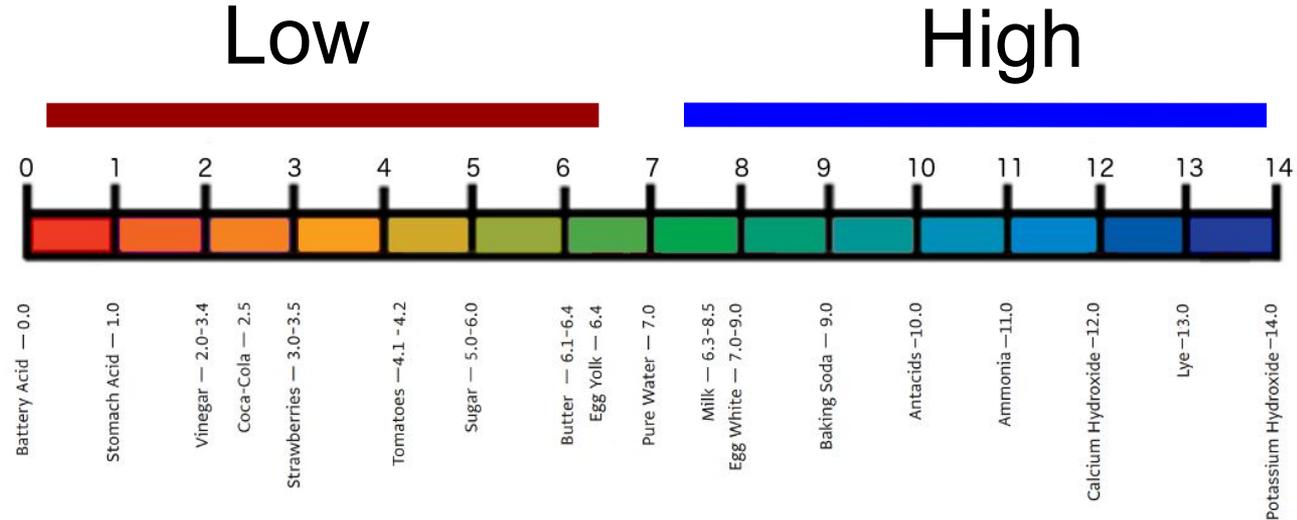
Connect the pH probe to Chip 2 on the breadboard with the BNC connector. Align the pin, push the collar on and twist.



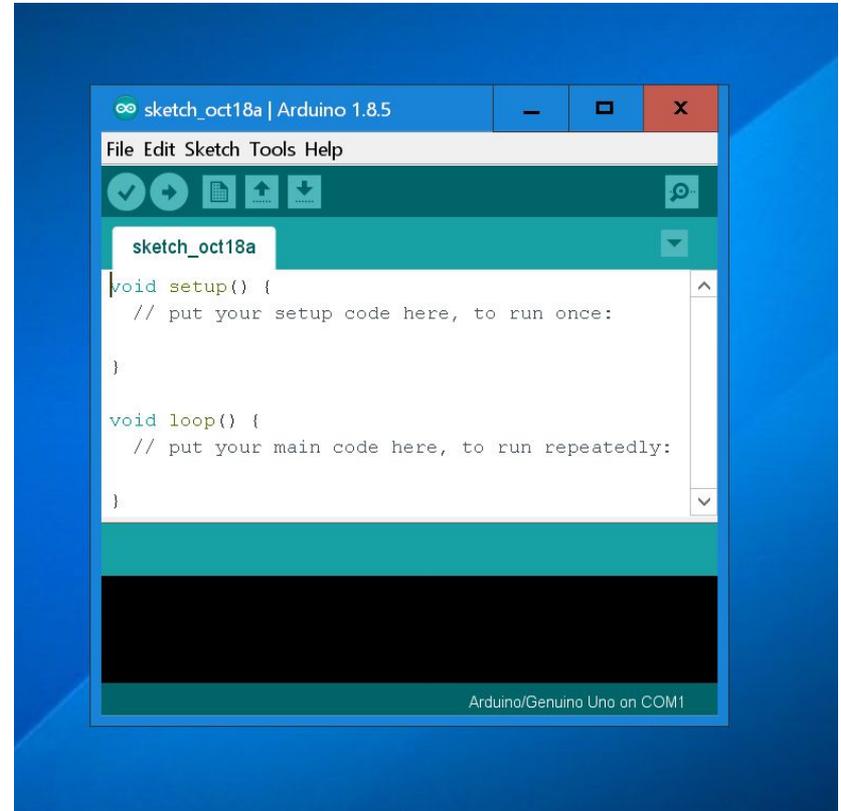
pH Probe Calibration

Materials List:

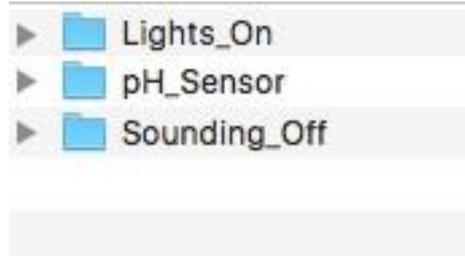
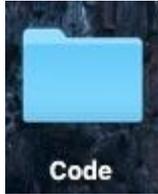
- pH probe and Arduino setup
- Windows PC (Laptop)
- 4 paper cups
- **pH 4.00 (Low)** calibration Solution
- **pH 7.00 (Mid)** calibration Solution
- **pH 10.00 (High)** calibration Solution
- squirt bottle with water
- probe storage solution



From the Windows Desktop, double click on the Arduino IDE icon and the Arduino IDE (Integrated Development Environment) will start.



In the the Arduino IDE click “File” and then “Open”. Open “Code” folder on the Desktop. Open the “pH_Sensor” folder and click “pH_Sensor” file.



Once the file is open we need to select the "Port" on the computer that should be used to receive data from the pH probe and Arduino. Click on the Tools menu at the top of the IDE, then click on "Port". The computer should show a list of ports with one of them indicating that the Arduino is attached. In the picture, this computer found the Arduino attached to the COM3 communications port. Click on the correct COM port for your computer and a checkmark will show that it is selected. Then press the arrow button to upload the code.

```
pin_server
//This code has written to be easy to understand.
//Modify this code as you see fit.
//This code will output data to the Arduino serial monitor.
//You connect into the Arduino serial monitor to control the pH circuits.
//This code was written in the Arduino 1.8.5 IDE
//An Arduino Uno was used to test this code.
//This code was last tested 6/2017

#include <SoftwareSerial.h> //we have to include the SoftwareSerial library, or else we can't use it
#define rx 2 //define what pin rx is going to be
#define tx 3 //define what pin tx is going to be

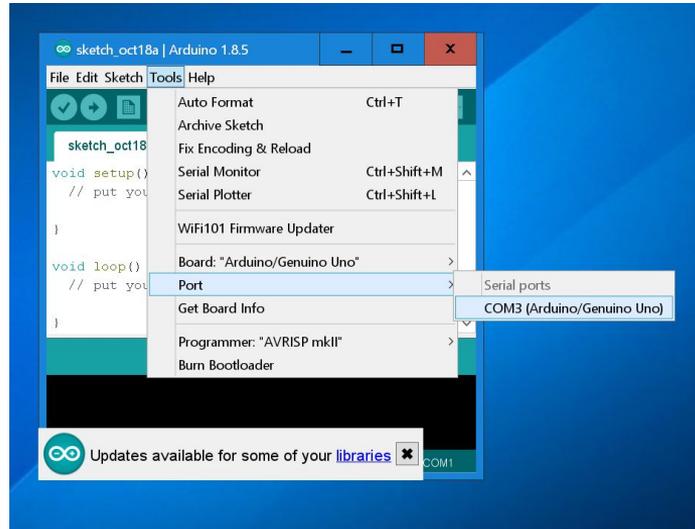
SoftwareSerial mySerial(rx, tx); //define how the soft serial port is going to work

String incoming = ""; //a string to hold incoming data from the PC
boolean transmitting = 0; //a string to hold the data from the Atlas Scientific product
boolean mySerial_complete = false; //have we received all the data from the PC
boolean sender_string_complete = false; //have we received all the data from the Atlas Scientific product
float pH; //used to hold a floating point number that is the pH

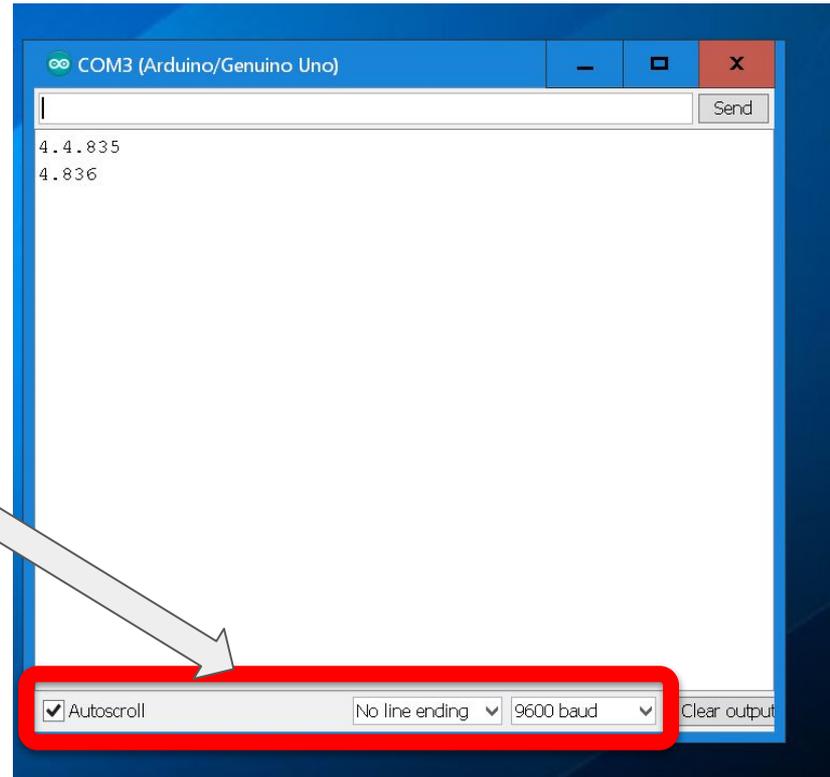
void setup() { //set up the hardware
  Serial.begin(9600); //set baud rate for the hardware serial port_B to 9600
  mySerial.begin(9600); //set baud rate for the software serial port to 9600
  pinMode(rx,INPUT_PULLUP); //set aside some bytes for receiving data from the PC
  pinMode(tx,OUTPUT); //set aside some bytes for receiving data from Atlas Scientific product
}

void serialEvent() { //if the hardware serial port_B receives a char
  if(Serial.available() > 0) { //read the string until we see a <del>
    incoming = Serial.readStringUntil(<del>); //see the flag used to tell if we have received a completed string from the PC
  }
}

void loop() { //here we go...
  if (Serial.available() > 0) { //if a string from the PC has been received in its entirety
    incoming = Serial.readString(); //read that string to the Atlas Scientific product.
    mySerial.print(incoming); //add <del> to the end of the string
    transmitting = 1; //clear the string
    Serial.flush(); //reset the flag used to tell if we have received a completed string from the PC
  }
  if (mySerial.available() > 0) { //if a string from the PC has been received in its entirety
    sender_string = mySerial.readString(); //read that string to the Atlas Scientific product.
    mySerial.print(sender_string); //add <del> to the end of the string
    sender_string_complete = false; //clear the string
  }
}
```



Once the code has uploaded successfully, click on the Serial monitor (the magnifying glass icon in the upper right of the IDE) and the serial monitor window will open. You will see pH values start to show up in the box, this is an indication that your pH probe is functioning properly and sending information to the Arduino. If you do not see values showing up in the box, double check that the COM port that shows "Arduino" has a checkmark next to it from step 2.



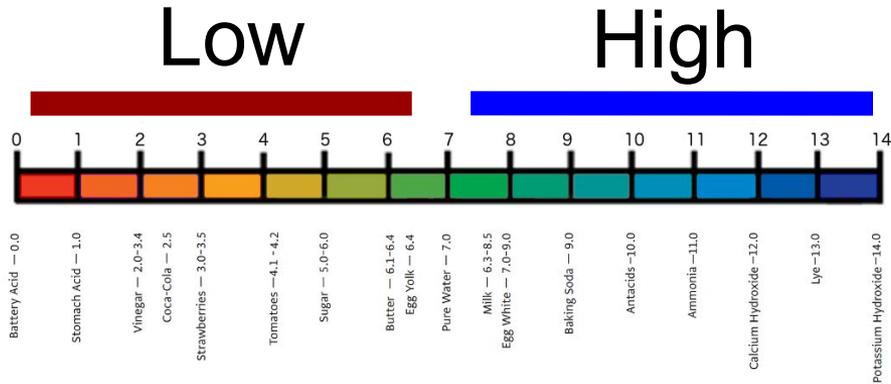
Make sure that the drop down menus match what is highlighted here.

(Autoscroll is checked, 'No line ending' and '9600 baud' are selected)

(a baud is an SI unit for bits per second)

Prepare calibration solutions for testing the probe:

1. Safety first! Put on your safety glasses and lab coat.
2. Arrange 3 paper cups in a row on your desk.
3. Open the calibration solution for **pH 4.00 (low)** and pour some of the liquid into the first paper cup.
4. Open the calibration solution for **pH 7.00 (mid)** and pour some of the liquid into the second paper cup.
5. Open the calibration solution for **pH 10.00 (high)** and pour some of the liquid into the third paper cup.
6. Insert the end of the pH probe into the first paper cup (pH 4.00) and watch the readings in the serial monitor window on your computer. The readings should start to move toward 4.00. However, they probably will not be exactly 4.00. We need to tell the Arduino that the current amount of Hydrogen Ions it is sensing is equal to a pH of 4.00. Type the following into the box at the top of the serial monitor, then press enter:

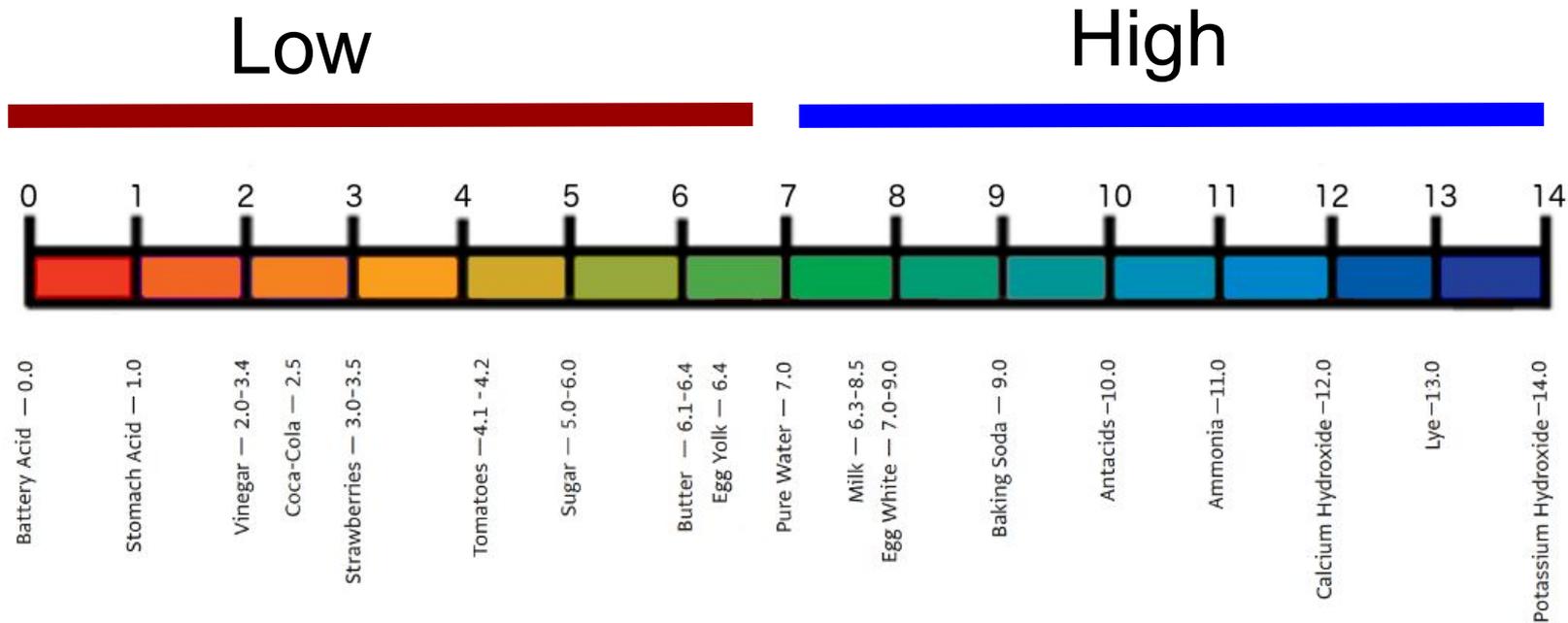


Type in Cal,low,4.000



Rinse the end of the probe with water from the squirt bottle into the 4th paper cup and repeat the process in (f) for the 10.00 pH calibration solution and the 7.00 pH calibration solution, rinsing before each liquid. To tell the Arduino that the current amount of Hydrogen Ions and Hydroxide Ions it is reading is equal to 10.00 pH, enter this into the top box of the serial monitor, then hit return: `Cal,high,10.000` To tell the Arduino that the current amount of Hydrogen Ions and Hydroxide Ions it is reading is equal to 7.00 pH, enter this into the top box of the serial monitor, then hit return: `Cal,mid,7.000`

Congratulations, your pH probe is now calibrated! However, look at the serial monitor and compare the read outs to the pH scale below. What problem or bug do you see? Write answer here then go onto the next page:



“Debugging” as “Computational Thinking”

Computational thinking - “Using special thinking patterns and processes to pose and solve problems or prepare program for computation” (Krauss and Protsman 2017, 173)

Debugging - “To track down and correct errors” in code, technology, or life more generally (Krauss and Protsman 2017, 173)

Krauss, Jane & Protsman, Kiki. (2017). *Computational Thinking {and Coding} for Every Student*. Thousand Oaks: Crowin.

Your job is to debug the code:

- 1) Go into the Arduino IDE where you will find two columns of information. To the left is the actual “code” and to the right is “documentation”, and explanation of what the code does.

```
sensorstring.reserve(30);
```

```
//set aside some bytes for receiving data from Atlas Scientific product
```

Code

Documentation

- 2) Look over the code, identify the bug, and fix the bug by typing in the correct information.