A [circuitry diagram](https://github.com/asikorski05/windtunnel_ECE103/blob/main/windtunnel_bb.png) is hosted on Github alongside the program files to easily follow along. To assemble the electronics, begin by setting up a breadboard with a push button. One leg is pulled high, and the other leg is pulled low through a resistor. The grounded side should be connected to digital pin 7 on the Arduino. This button will tare the load cells much like a kitchen scale. Connect 5V and GND from the Arduino to the breadboard’s power rails. The LCD display is optional, but can be added by following [this diagram](https://github.com/asikorski05/windtunnel_ECE103/blob/main/windtunnel_bb.png) in GitHub. A small potentiometer is needed to adjust the display’s contrast, but may be replaced with a resistor if adjustment is not wanted. Next, the load cells are connected to respective HX711 amplifiers. The HX711 outputs are connected to the Arduino’s digital pins 2-5, and the HX711’s 5V and GND wires should be hooked up to the breadboard’s power rails. The drag cell should be digital pins 2 and 3, while the lift cell should be digital pins 4 and 5 as seen in the diagram. Finally, connect the wind sensor’s 5V and GND lines to the breadboard’s power rails. A separate 5 volt power supply is needed for accurate wind speed values and must also be connected to the power rails. The data lines of the wind sensor will be plugged into the Arduino’s analog 0 and 1 pins. In our project, we did not include an LED, but one may be connected to the wind sensor’s OUT pin if desired. When the wind speed increases, the brightness of the LED will increase as a visual representation of the signal.  
  
Download the Arduino scripts from the [GitHub repository](https://github.com/asikorski05/windtunnel_ECE103). Use “CALIBRATE \_LOAD\_CELLS.ino” to calibrate the accuracy of the load cells. The script will walk you through the process, but it boils down to taring the scale with no weight, applying a known weight, reading the output, and dividing that by the known weight. The units of measurement will be whatever the known weight’s units are. Repeat for the other cell and replace the calculated values into the LIFT and “DRAG\_CALIBRATION\_SCALE” variables respectively, found in the top of the “WINDTUNNEL.ino” script.  
  
Next, use “CALIBRATE\_WIND\_SENSOR.ino” for the wind speed calibration. Place a cup over the wind sensor, making sure it does not touch the sides or the surface beneath it. Adjust the variable “zeroWindAdjustment” until the wind sensor reads zero as the wind speed. Remove the cup and it should measure speeds in MPH when you blow on the sensor. Take the “zeroWindAdjustment” value and insert it into the “THERM\_WIND\_CALIBRATION” value found in the “WINDTUNNEL.ino” script.

Finally, change the “units” and “decimalPrecision” variables as desired. These apply to the serial USB data output. Upload your modified “WINDTUNNEL.ino” to the Arduino and notice data displayed on the LCD or through the USB serial monitor in the Arduino SDK. The circuitry and programming is complete!

TROUBLESHOOTING:  
If the display shows a sensor error or if the load cell data seems unrealistic, please check the connections to the HX711 amplifiers. They should be securely connected to the Arduino and the breadboard’s power rails.  
  
If the wind speed seems unrealistic or fluctuates dramatically, please check that a separate 5 volt power supply is connected to the circuit.

If the wind speed shows “nan”, this is normal. It means “not a number” and is essentially a negative value. Please consider recalibrating the wind sensor if this value appears often.