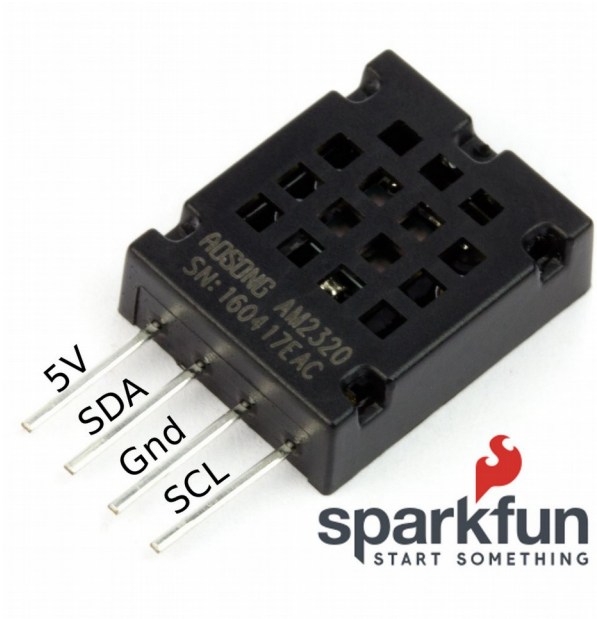
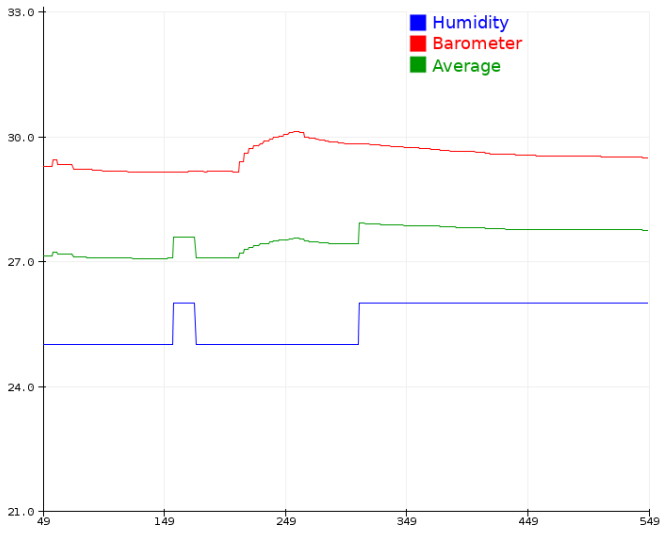
Now that we've learned how to deal with our barometer and humidity sensors on their own, let's take a deeper look at them.

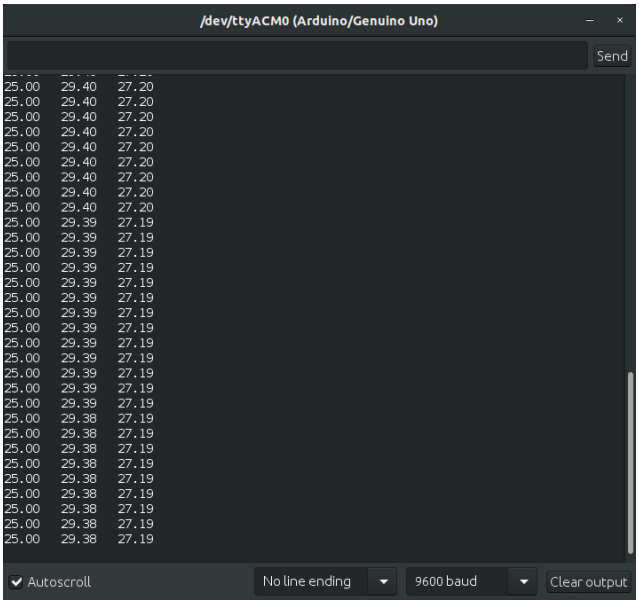
A barometric altimeter will use air pressure with context of air temperature to calculate a resulting altitude assuming a local ground level pressure offset. Humidity sensors also often have temperature sensors inside to be used in combination with humidity to calculate "heat index" which is how the humidity affects how air temperature feels to skin. For example, a certain temperature with very dry air would feel cooler to skin because moisture would evaporate from the skin, further cooling the skin. However, at the same temperature with highly humid air would feel warmer to a person's skin because, with already high amounts of moisture in the air, skin moisture would be less prone to evaporate. Instead, moisture would be more prone to condensing from the air onto the skin, which then absorbs the air moisture thermal energy.



Our barometer and humidity sensors have internal temperature sensors for this compensation for humidity and air pressure, however they're usually less sensitive and accurate compared to a dedicated temperature sensor. In this example, we will look at how to access the temperature data from these sensors, and compare the sensitivity.

Reconstruct the humidity and barometer circuits and upload the temperature program to the arduino and open the serial monitor or plotter. Since both devices use I^2C to communicate with the host arduino, the sensor SDA and SCL pins are connected to the same pins on the arduino.





These two screen captures are results from running the temperature program. The temperature program loads both the barometer and humidity sensor’s libraries and uses their respective temperature functions to read temperature from the sensors in Celsius. Notice the humidity sensor’s temperature reading has a square step shape to it just the same as the barometer’s internal temperature sensor, except the steps are much smaller in the case of the temperature sensor. An average reading is included since an average reading can be useful in a payload reading temperature from many sources all at once. As can be seen in the graph, the barometer and humidity sensor’s internal temperature sensors were stimulated individually by simply touching the casing of one while being careful to not touch the other. So a heat event from either device is seen in the average. This also means that errors from one device can propagate to the average and decrease the quality of data being recorded. Recording both individual signals and their average gets the best of both worlds, but the averaging can also be done in post processing of the data if program size needs to be optimized, but this increases post processing workload. So this is the trade off.