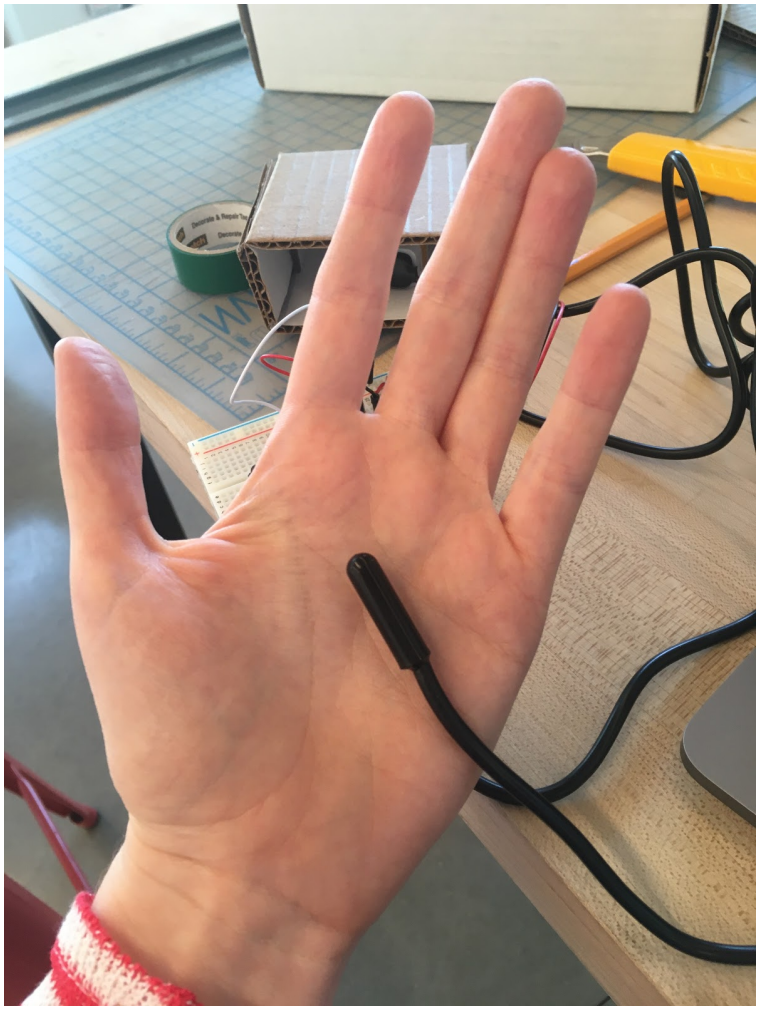
Leanna’s Trials and Tribulations with the Cow Thermometer

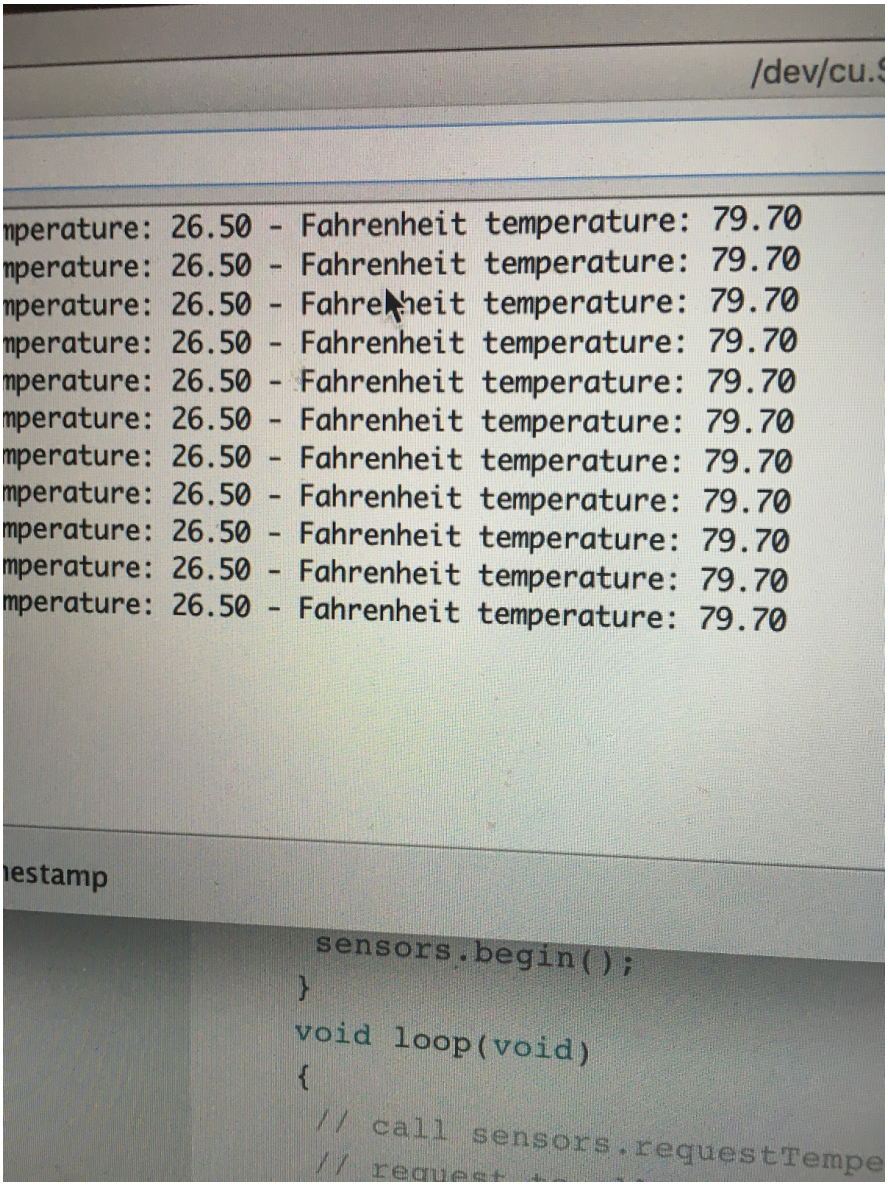
Leanna Mulvihill
Lpm36
Interactive Device Design
Wendy Ju
Fall 2019

The cow thermometer is basically a re-creation of SmaXtec Classic Bolus with just a temperature sensor and a human(Leanna) instead of a cow. Using a human instead of a cow means that a bolus is not possible and this needs to be wearable instead. I was imagining that the arduino and the sensor would be inside the bolus and communicating with the raspberry pi over bluetooth, but plans changed.

1. Select this temperature sensor Waterproof (DS18B20).

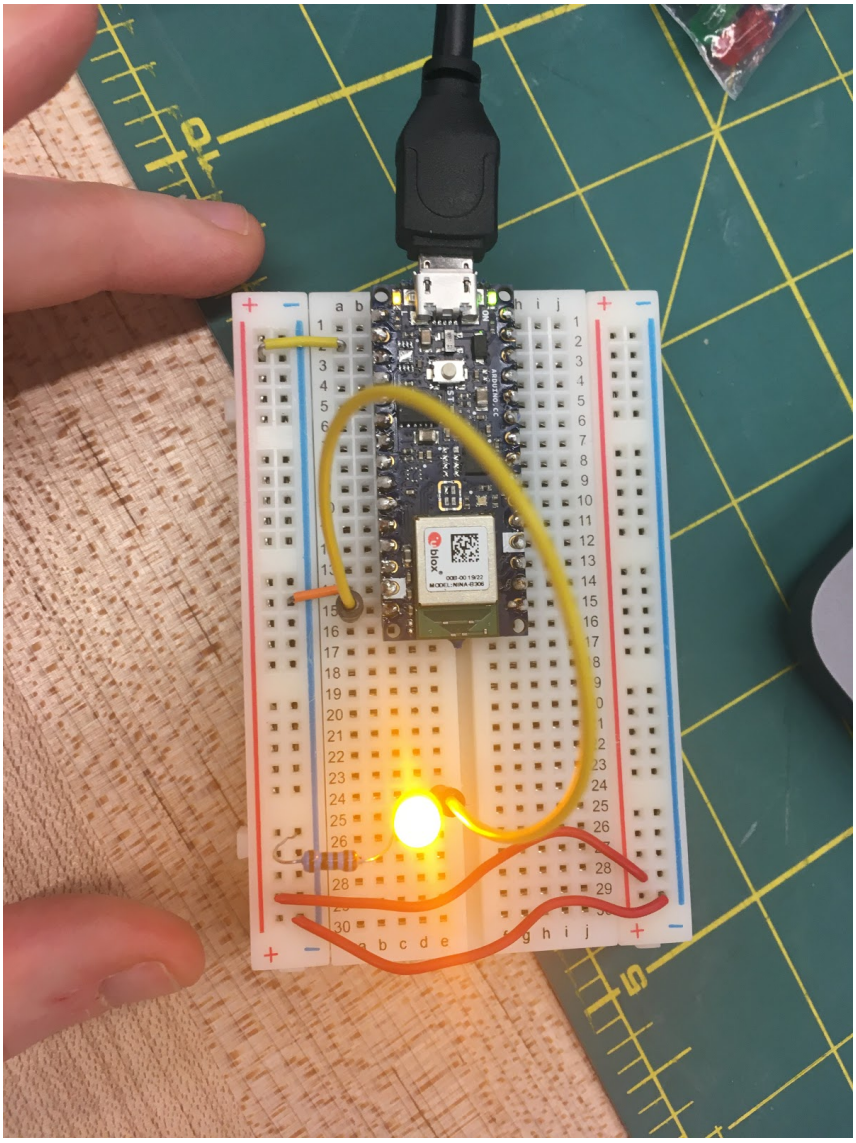


   a. This is the best sensor for taking body temperature rather than ambient temperature because it has a long cable, is waterproof and is sensitive enough to detect within 0.5 degrees fahrenheit.
   b. It reads out the temperature on the serial monitor from the arduino. This tutorial was helpful.



   c. Need to use both the One Wire Library and the Dallas Temperature Library.
   d. Fry the arduino metro mini with a battery mishap. Womp.
2. Buy the arduino nano 33 BLE and try to make it work with the temperature sensor.
   a. After much trouble shooting, the sensor is not compatible with the arduino nano. These are the tests I ran

i. Fade LED works - the arduino nano board works.



ii. AnalogReadSerial works with an FSR - the serial monitor works with the arduino nano board.

iii. Parasite mode for powering the temperature sensor did not work on the arduino nano board.

iv. You still need a 4.7k resistor according to the forums. The resistor does not change for 5v vs. 3.3v boards.

v. The ds18b20 should work down to 3.0v.

vi. The ds18b20 works with the metro mini board still.

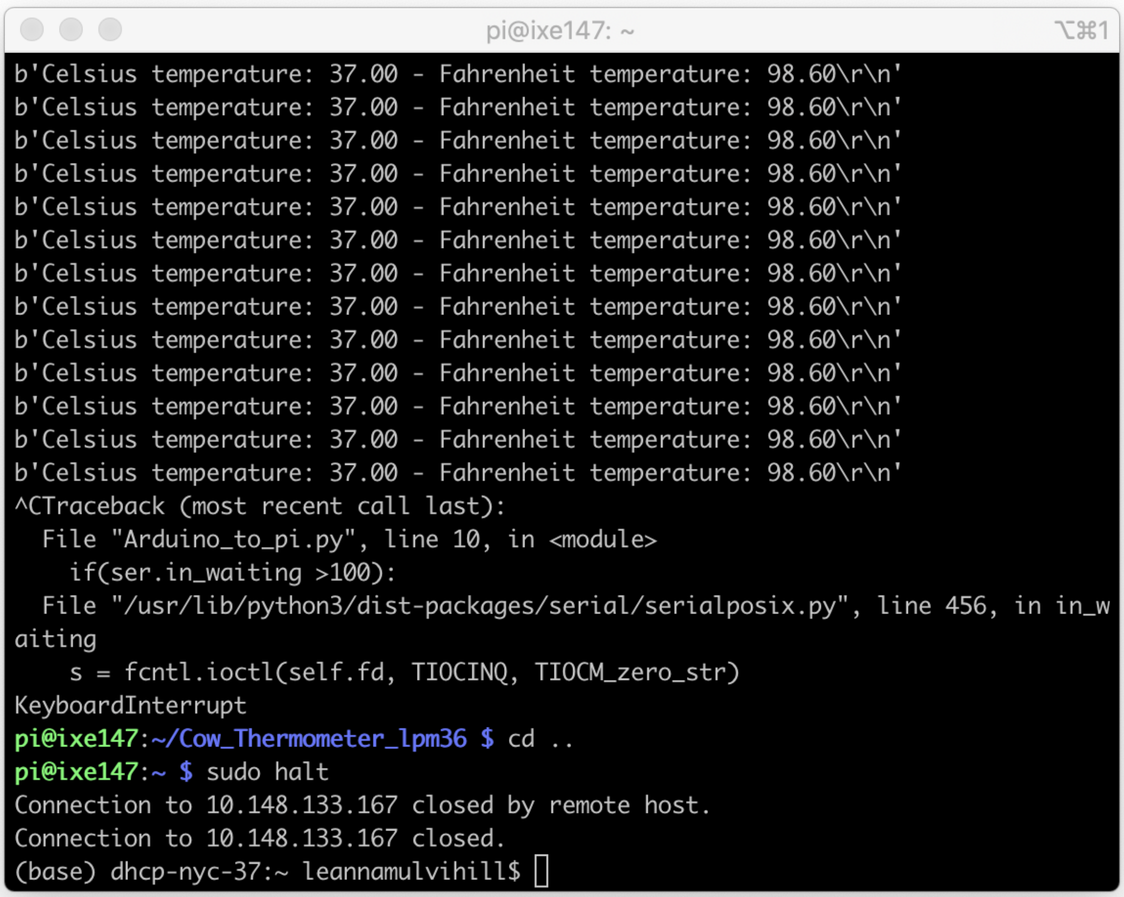
vii. Try the Dallas Temperature Single example - did not work on the arduino nano.

viii. Try the One Wire temperature example on the arduino nano - no more addresses error.

ix. oneWireSearch example - no reading on the serial monitor at all for the arduino nano.

x. Dallas Simple example - "Error: Could not read temperature data" on the arduino nano.

3. Because bluetooth is not an option for the temperature sensor, I moved forward and hooked the arduino up to the pi.
    a. This tutorial was helpful for serial communication between the arduino and the pi. I was worried I would have to write my own server code in javascript and couldn't find any examples to work with, but this solved my whole problem. Now the temperature readings are spit out in the terminal on my laptop over wifi from the pi.

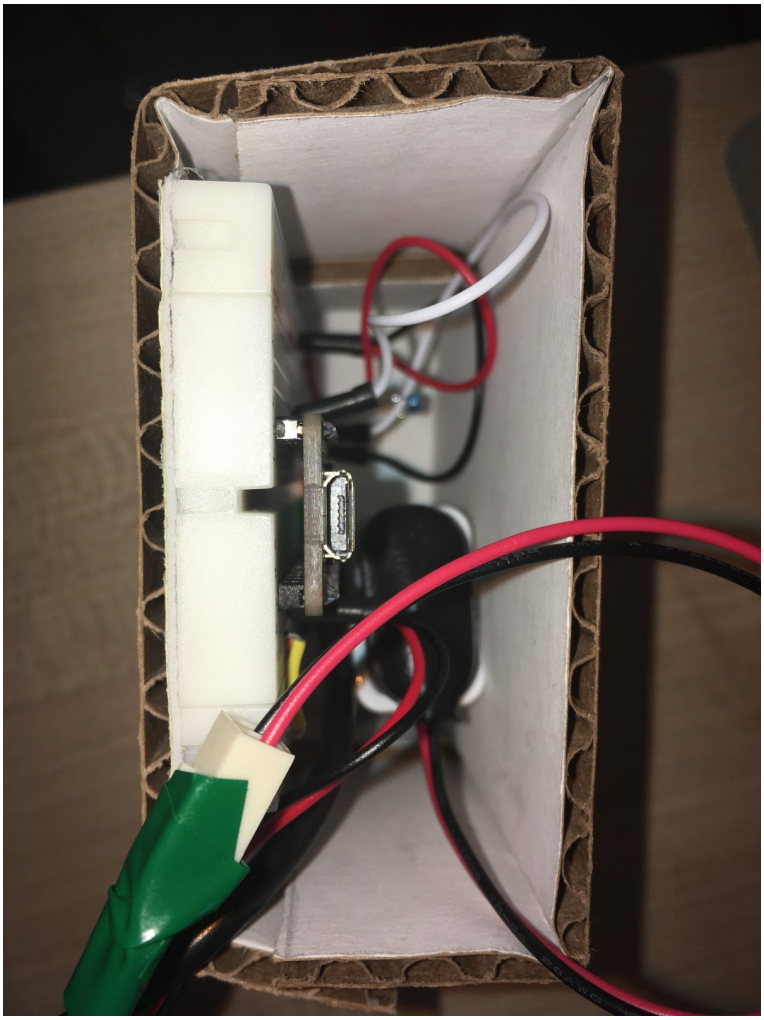


```
pi@ixe147: ~                                                        ⌥⌘1
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
b'Celsius temperature: 37.00 - Fahrenheit temperature: 98.60\r\n'
^CTraceback (most recent call last):
  File "Arduino_to_pi.py", line 10, in <module>
    if(ser.in_waiting >100):
  File "/usr/lib/python3/dist-packages/serial/serialposix.py", line 456, in in_w
aiting
    s = fcntl.ioctl(self.fd, TIOCINQ, TIOCM_zero_str)
KeyboardInterrupt
pi@ixe147:~/Cow_Thermometer_lpm36 $ cd ..
pi@ixe147:~ $ sudo halt
Connection to 10.148.133.167 closed by remote host.
Connection to 10.148.133.167 closed.
(base) dhcp-nyc-37:~ leannamulvihill$ 
```

    b. The delays on the arduino code and the pi code had to be adjusted because the pi was reading temperatures faster than the arduino was providing them. This resulted in the terminal alternating showing accurate readings with completely garbage readings. Now it works fine!

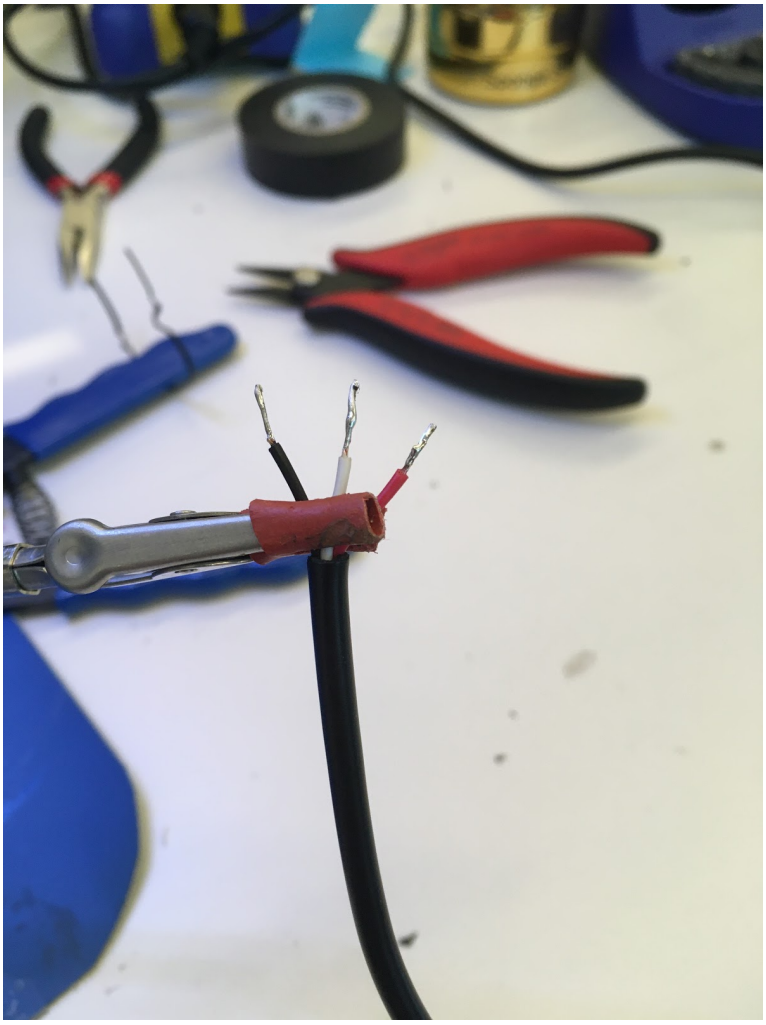
4. Mechanical Parts

a. I made a cardboard box for the arduino and breadboard.



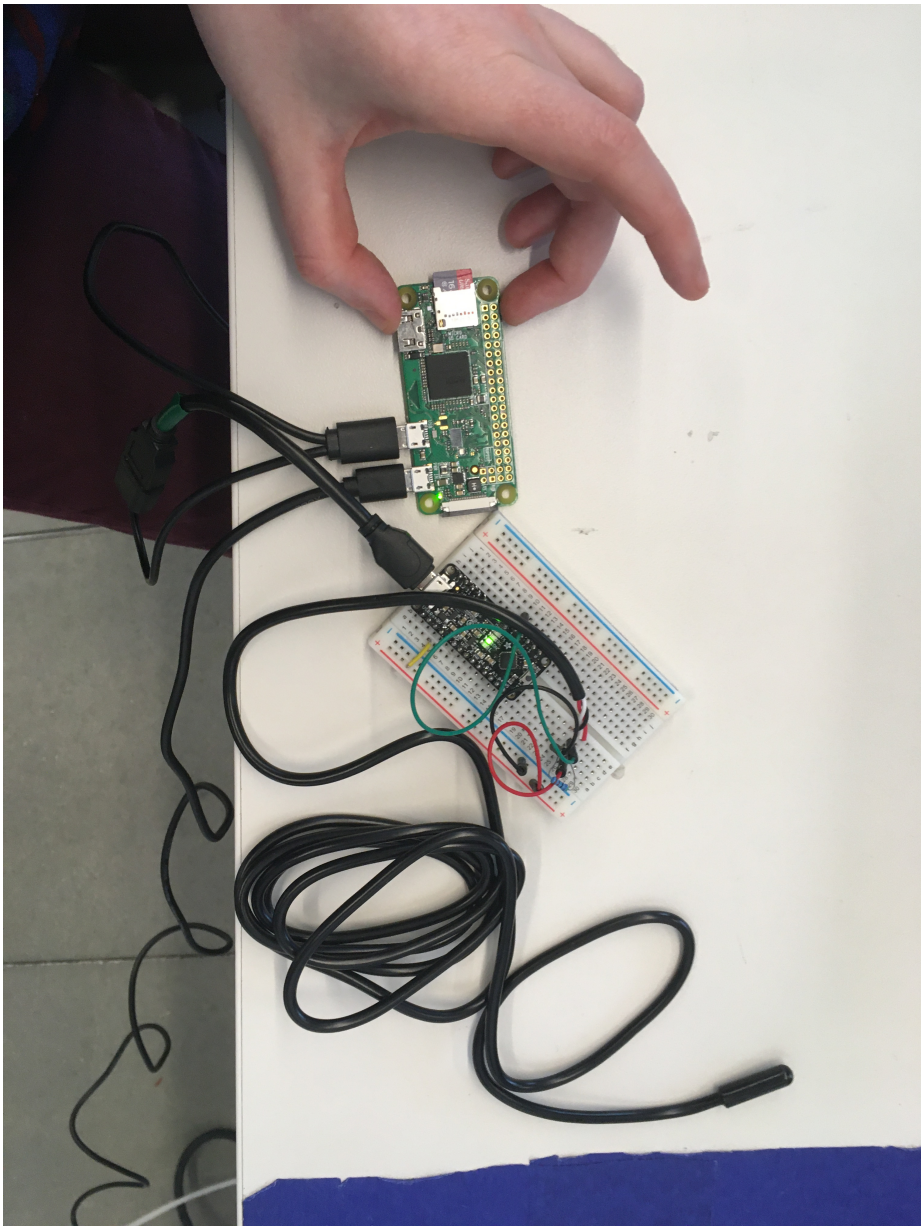
b. The wires from the temperature sensor kept popping out of the breadboard because they had very little of the wire itself exposed and it was uneven. I trimmed the whole cable, stripped each of the three wires to expose more metal

and coated the tips in solder. It worked great.



c. The arduino and the pi fit in a fanny pack to be able to wear the temperature in my armpit for the best readings. I can get it right up to 98.6 degrees fahrenheit.

5. Pi Zero. To get even closer to what the real bolus would be like I borrowed a Pi Zero from Wendy and made that work with the temperature sensor. I had to find the mac address and the appropriate cables to hook up the pi to a monitor which was an adventure.



a. The [smaXtec classic bolus] has dimensions of 105mm×35 mm (length×diameter). We could approach this size if we had a pcb made for this prototype.