Resources used:
https://stackoverflow.com/questions/35952254/how-to-append-to-a-new-row-when-writing-to-csv-file
https://stackoverflow.com/questions/12277864/python-clear-csv-file#
https://stackoverflow.com/questions/29725932/deleting-rows-with-python-in-a-csv-file

Monday, May 20

At first, we were able to use the FTDI chip driver to directly connect our raspberry pi to Ryan's computer. This does not use the HDMI output so it uses far less power than before. Installation of the software was relatively trivial. We created test files so that we could manipulate the existing python code as described below.

Today, wrote out the python code that takes the relevant information (temperature, humidity, date and time etc.), and appends it to two CSV files. The two CSV files are titled "buffer.csv" and "storage.csv." The purpose of the storage file is to permanently store all data collected. The purpose of the buffer file is to hold a small amount of information before cellular transmission. Once there is enough data for a cellular transmission, then the CSV file is converted to a JSON file and then transmitted. In the code, we were told to use 4 as a reference. Then, we clear the buffer CSV file so it is empty.

At the end of class, Raj checked out the python code that we had going for this aspect of the project. We decided to slightly alter the way that the transmission works. First, we decided to read the CSV file as few times as possible to prevent damage. Second, we decided to avoid if statements because those can often cause problems in coding. Ultimately, the solution has to do with reading the CSV file into a numerical python array on a cron job. Then, divide the array into sections such that we do not exceed the maximum cellular transmission limit. Then, have read the output of the cellular transmission. Once we have worked through the array, we append any parts of the array back to the CSV file that failed to send. There are many ways to transmit this data, we just need to make sure we are doing it safely and efficiently.

Ryan began work on the GitHub page. We reviewed what we have been working on for the last few weeks and we are starting to think about how we can best document the progress we have made. Raj emphasized that proper documentation of this work online would be extremely helpful for others who are interested in power consumption of a raspberry pi, autonomous weather stations or other applications of the charge controller. Once we finish the python scripts, the remained of our time will be spent on the GitHub page so others can complete the same project or use our own findings to help with their own DIY projects.