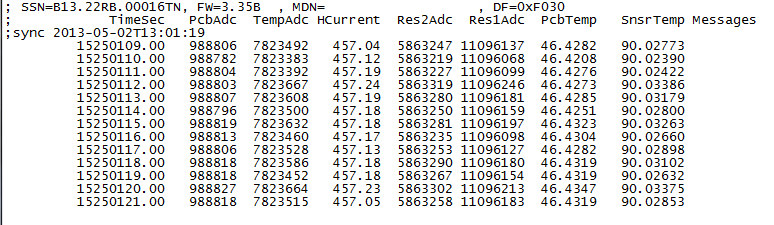
# Software Developer Project

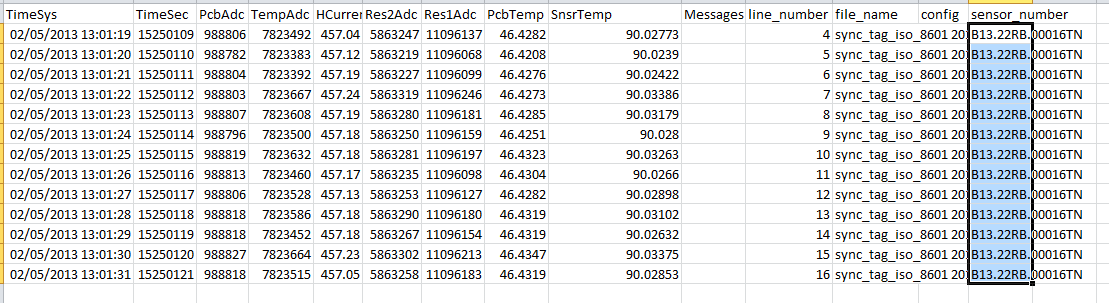
1. Section I
   1. Using a scripted programming language, extract the data (parse) in the data capture file called “sync\_tag\_iso\_8601 20141110-173211” and place in an Excel File with the data separated by columns including headers.
   2. The screenshots below show the input data and the resulting output of your parser. Be sure to include the additional columns shown in the spreadsheet below, notably a column titled “TimeSys.” This column should be a date/time in the format of MM/DD/YYYY HH:MM:SS, calculated based on the “Sync Tag” (YYYY-DD-MM(T)HH:MM:SS). Each increment of the TimeSec data from the capture file is assumed to be a 1 second increment in time from the Sync Tag. If a Sync Tag is not present in the capture file data, use the date in the filename and assume a start time of 08:00 (am).
   3. Using the same script, perform the same exercise with the filename “B15.09R.10060TN1X 12L-3077-3914 20150611 14.7 PsiA VT1”.
   4. (Bonus) Expand the functionality of the parser to properly parse the file “B11.03LT.10079TN1X VT1 150729”. In this file there are hundreds of lines of configuration data that does not need to be extracted. Additionally, the data column header changes several times throughout with occasional interruptions.
   5. Load the software written for this project to <https://github.com/smimatt/devtest>

Sensor



Sync Tag

Columns



1. Section II
   1. Using Django, create browser based display of data where the capture files are displayed by sensor number.
   2. When the sensor number is clicked, it presents the parsed data.
   3. (Bonus) Add functionality to plot the data as a function of real date/time within the browser window.