**Embry – Riddle Ionospheric Scintillation Algorithm (EISA)**

**PARSING.PY CODE**

The purpose of this code is to convert .GPS binary files into readable .csv files. To change the settings of the code, go to settings.csv.

*Only parse data in intervals within the same year. (E.g. January 1, 2018 – December 31, 2018,* ***NOT*** *November 10, 2018 – February 15, 2019).*

*If the code does not find the binary file even though the directory is correct in the Settings.csv file, rewrite the directory (MAKE SURE THERE ARE NO BLANK SPACES AFTER ANY WORDS IN EACH CELL – e.g. “Desktop” NOT “Desktop “).*

**CSV FILES - PARSING.PY CODE**

For the graphing.py code to run properly, **two csv files must be placed in the same folder as the code: GPSCALENDAR.csv and Settings.csv.** GPSCALENDAR.csv does NOT require any modifications by the user before running the code. However, it is necessary for the code to run without any issues, so do not delete it by any means. The settings.csv must follow the requirements specified as follows:

***SETTINGS.csv***

1. In row 2, insert the **path-directory to the folder where the Binary files** are saved.

The first cell of this row must include the drive name (e.g. “C:\” on Windows or “\” on Linux). See snapshot on page 8 to see an example.

1. In row 4, insert the **path-directory to the folder where the parsing.py** code is located.

The first cell of this row must include the drive name (e.g. “C:\” on Windows or “\” on Linux). See snapshot on page 8 to see an example.

1. In row 6, insert 1 to parse only REDUCED data, 2 for RAW data, and 3 for BOTH.
2. In row 8, insert the **path-directory to the folder where you want the csv files** to be saved.

The first cell of this row must include the drive name (e.g. “C:\” on Windows or “\” on Linux).

See snapshot on page 8 to see an example.

1. Novatel GPStation-6 collects data from three different constellation systems: Global Positioning System (GPS), GLONASS, and GALILEO. These constellations were assigned a variable for code processing purposes as follows:

GPS: **G**

GLONASS: **R**

GALILEO: **E**

In row 10, column 1, input the variable corresponding to the constellation (e.g. G to create csv files for GPS). If you want to generate csv files for several constellations use column 2 and 3 to add the other variables.

1. **PRN number**: The Pseudo Random Number (PRN) code acts as an identifier for each satellite (e.g. GPS PRN 1 represents satellite 1 from the GPS system). In row 12, column 1, insert the PRN number. The code will combine the constellation selected in row 10 and the selected PRN number (e.g. G1=GPS PRN 1, R5=GLONASS PRN 5, E7=GALILEO PRN 7, etc.) In row 12, the user can add as many PRN's as they want in the following cells (column 2, 3…). If the user wants to generate csv files for ALL PRN numbers within a constellation, they must insert “**T**” ONLY in the first column.
2. Row 15: For Parseraw, if you want to parse data between a specific time range insert 1 in column 1. Otherwise, insert 0. If column 1 is set to 1, insert the start time in the second cell and the end time in the third cell. These time must be in GPS TOW (Seconds). When the binary file contains more than 1 week worth of data. In the fourth cell indicate the start GPS week, and in the fifth cell the end GPS week. DO NOT USE THIS OPTION IF YOU ARE NOT ENTIRELY SURE HOW TO SET IT UP.
3. Row 17: Insert the receiver name in column 1. The receiver name can be determined by looking at the names of the binary files. (E.g. in file 1992\_2\_00\_RX1.GPS, the receiver name is RX1). Make sure to insert the correct name for the receiver, otherwise the code will not find the file.
4. Row 19: Insert the year (E.g. 2018). Only parse data in intervals within the same year. (E.g. January 1, 2018 – December 31, 2018, NOT November 10, 2018 – February 15, 2019).
5. Row 21: The code will parse data within a specified date range (E.g. January 1 – April 2). Insert the initial date in row 21. Insert the month in column1 and the day in column 2.
6. Row 23: Insert the final date in row 23. Insert the month in column1 and the day in column 2.

**GRAPHING.PY CODE**

The purpose of this code is to plot and save ionospheric scintillation and TEC graphs. The directories can be changed by modifying the paths.csv file. The settings can be changed by modifying the graphsettings.csv file.

The code generates plots for multiple variables (TEC, Azimuth, Elevation, etc.) with respect to time (i.e. the x-axis will always be time in UT).

***For RAW (High-Rate) data to be plotted, it is necessary to have REDTEC (Low-rate) files within the same folder.***

In order for this code to run, matplotlib must be installed on the computer. Refer to the following webpage to see instructions on how to install matplotlib:

<https://matplotlib.org/users/installing.html>

On Windows, run the following commands on the command window:

* python –m pip install –U pip
* pip install matplotlib

On Linux:

* Installation steps vary depending on the OS (e.g. Ubuntu). See the link above to see a more detailed explanation on the installation process for every OS.

**CSV FILES - GRAPHING.PY CODE**

For the graphing.py code to run properly, **two csv files must be placed in the same folder as the code: Paths.csv and Graphsettings.csv.** These files must follow the requirements specified as follows:

***PATHS.csv***

The paths.csv file has **two** input sections. For the code to work, fill out ENTIRELY and CORRECTLY all of the following sections:

1. In row 2, insert the **path-directory to the folder where the CSVfiles** are saved. This folder MUST have folders inside named after each date (i.e. YYYYMMDD or 20170820).

The first cell of this row must include the drive name (e.g. “C:\” on Windows or “\” on Linux). See snapshot on page 8 to see an example.

1. In row 4, insert the **path-directory to the folder where the graphs** will be saved after running the code.

The first cell of this row must include the drive name (e.g. “C:\” on Windows or “\” on Linux). See snapshot on page 8 to see an example.

***GRAPHSETTINGS.csv***

The graphingsettings.csv file has \_\_ input sections. These sections have different purposes and need to be filled out accordingly depending on what the user wants to do.

1. The first and most important input is the **file type**. Novatel receiver generates data in the following logs:

**REDOBS** 60-second measurements of Amplitude and Phase Scintillation. In the first case (S4 - Amplitude scintillation), values are automatically detrended by the GPStation-6 monitor either by using a 6th-order Butterworth filter or using the average measurement of the 60-second log. The receiver outputs both "S4-raw" values as well as "s4-corrected" values after ambient-noise correction. Regarding Phase (sigma-phi) scintillation, a 6th-order Butterworth filter is also used by the receiver to detrend the scintillation measurements. The data is logged at intervals of 1 second, 3 seconds, 10 seconds, 30 seconds and 60 seconds, for a total of 5 measurements in a span of one minute.

**REDTEC**  TEC measurements at 15 second intervals. 4 measurements are logged in a minute along with delta TEC values (change in TEC every 15 seconds). The delta TEC values are computed by the monitor based upon carrier phase measurements differences between frequencies.

**IsmRawTEC** 1 second TEC measurements for the different valid frequency combinations for each satellite. The data logged by the GPStation-6 is unfiltered data and requires post-processing to be analyzed. The post-processing of raw TEC values is known as TEC detrending, and involves a low-pass = Butterworth filter which is embedded into the graphing.py.

**IsmRawOBS** Data logged by the GPStation-6 GPS receiver at a rate of 50 Hz, for a total of 3000 measurements per minute. This data is unfiltered and gives RAW ADR and power measurements.

**IsmDetOBS** The GPStation-6 monitor is also capable of filtering this data and outputs an additional log with detrended High-Rate Ionospheric Scintillation data. This correction takes into account external factors such as satellite motion.

Each log outputs the data in different CSV files. For this reason, the user must select the file type (i.e. the log type before proceeding) based on the parameters established above.

E.G: If you want to graph data for low-rate TEC, select REDTEC. For high-rate TEC, select IsmRawTEC.

In row 2, column 1, insert the prefix. There are three VALID prefixes: **RED, IsmRaw, IsmDet**. Make sure to write the prefix exactly as above (i.e. RED, not red).

In row 2, column 2, insert the second part of the file name. There are two possible options: **TEC, OBS.** TEC will output Total Electron Content data, while OBS will output Ionospheric Scintillation data.

Depending on the user selection for file type in row 2, the code will show a different menu to the user. The following are the options for each file (log) type (TIME VS. \_\_\_\_):

|  |  |
| --- | --- |
| **REDOBS** | Azimuth, Elevation, CNo, Lock Time, CMC avg, CMC std, S4, S4 Cor, 1secsigma, 3secsigma, 10secsigma, 30secsigma, 60secsigma |
| **REDTEC** | Azimuth, Elevation, SecSig Lock, SecSig CNo, TEC15, TECRate15, TEC30, TECRate30, TEC45, TECRate45, TECTOW, TECRateTOW |
| **IsmRawTEC** | TEC, TECdot |
| **IsmRawOBS** | ADR, Power |
| **IsmDetOBS** | ADR, Power |

A detailed explanation of each of the listed options can be found in the Novatel GPStation-6 User Manual **(Chapter 5)** in the following link: <https://myerauedu-my.sharepoint.com/:b:/g/personal/gachancj_my_erau_edu/EXISy3awwxVBu_jrnDz6ijIBpzR3KoQAiqlHCf0aav4SOg?e=VL2FO3>

1. **Elevation threshold**.

In row 4, column 1, insert a number (either a float or an integer) between 0 and 90. This number represents the elevation threshold. Every value from the data gathered under this elevation will be cut off before making a plot. Make sure that the other cells in this row remain EMPTY.

E.g. If the threshold is 35, the code will remove every value under 35 degrees from the excel file.

1. **Constellation type.**

Novatel GPStation-6 collects data from three different constellation systems: Global Positioning System (GPS), GLONASS, and GALILEO. These constellations were assigned a variable for code processing purposes as follows:

GPS: **G**

GLONASS: **R**

GALILEO: **E**

In row 6, column 1, input the variable corresponding to the constellation (e.g. G to plot graphs for GPS). If you want to generate plots for several constellations use column 2 and 3 to add the other variables.

1. **PRN number**.

The Pseudo Random Number (PRN) code acts as an identifier for each satellite (e.g. GPS PRN 1 represents satellite 1 from the GPS system). In row 8, column 1, insert the PRN number. The code will combine the constellation selected in row 6 and the selected PRN number (e.g. G1=GPS PRN 1, R5=GLONASS PRN 5, E7=GALILEO PRN 7, etc.) In row 8, the user can add as many PRN's as they want in the following cells (column 2, 3…). If the user wants to generate plots for ALL PRN numbers within a constellation, they must insert “**T**” ONLY in the first column.

1. **Date**

In rows 10, 12, and 14 insert the month, day and year respectively. Insert values ONLY in the first column of each row. These values will be used to open the respective date folder. This folder needs to be located in the directory specified in the PATHS.csv file**.**

1. **Summary Plot**

This option lets the user create plots that show data from multiple PRNs. In row 16, column 1, the user must input either “0” or “1”. “0” is the default value and represents NO summary plots. When “0” is selected, the code generates individual plots for every PRN. If “1” is selected, the code will generate a summary plot with data from multiple PRNs. Row 16, column 2, is an optional input and represents the shift value. This option becomes useful when doing summary plots with stack plotting for Ionospheric Scintillation data (S4 or sigma-phi). The value represents the offset between the data of each PRN in the summary plot. This option will ONLY work if column 1=”1” (i.e. if the user is generating a summary plot).

1. **Night Subtraction = Normalizing**

This method is **ONLY** used for Low-Rate TEC (REDTEC) summary plots. It calculates the minimum TEC value out of all the selected PRNs and sets that vale as the zero-reference value. All the other points in the data are then shifted with respect to this reference value. This process is called night-subtraction, as the lowest TEC value normally occurs during the night period. For this code, we use the term “normalizing” to refer to the night-subtraction process.

In row 18, column 1, insert 1 to normalize the low-rate TEC (REDTEC) data.

1. **Axis limits**

In rows 20 and 21, the user can specify the range of the x-axis and y-axis. The code will only plot the values within this range. E.G. If the user inputs 10 and 20 as the x-axis limits, the code will generate a plot showing an x-axis between 10 and 20.

For both rows. If you want to insert some limits, set the value in the first column to “1”. Then select the start value of the range in column 2, and the final range value in column 3.

1. **TEC detrending**

This part of the code was written based on an existent MATLAB code provided by Professor Kshitija Deshpande from Embry-Riddle Aeronautical University (ERAU). This method uses statistics to get rid of depletions in the High-rate TEC which may affect the correct reading of the data. These irregularities in the data may be a product of multi-path, satellite motion, among other external factors. If you are plotting high-rate TEC, it is highly recommended to use this option.

To activate this option, insert “1” in row 23, column 1.

1. **PRN number and legend**

The following two options are mostly used when producing a summary plot:

Insert “1” in row 25, column 1, to show the PRN numbers next to each line within the plot.

Insert “1” in row 27, column 1, to show a legend next to the plot.

1. **Vertical line across the graph**

This option is used in very specific cases, when the user wants to mark a specific time within the graph. E.G. During the 2017 Great American Eclipse, this option was used to mark when the maximum eclipse occurred. The code plots a vertical line perpendicular to the x-axis passing exactly through the time that the user selected.

To activate this option, insert “1” in row 29, column 1. Also, insert the x-axis value in column 2 (e.g. inserting “12” in column 2 will generate a vertical line across the plot perpendicular to the x-axis crossing through 12UT).

1. **Vertical TEC**

The low-rate TEC values collected by the receiver are computed in terms of Slant TEC. These values must be converted to vertical TEC to account for the effects of the thickness of the atmosphere. Activate this option to show more accurate values for Low-Rate TEC. To activate it, insert “1” in row 31, column 1. **Vertical TEC conversion WILL ONLY WORK if the data is being normalized (i.e. row 18, column 1 is set to “1”), AND for reduced (REDTEC) data (i.e. first do night-subtraction, then convert to vertical TEC).**

1. **Only one graph per PRN.**

Sometimes, TEC and scintillation values may be computed using multiple combinations of signals for a single PRN. For this reason, the data sometimes shows duplicates for every PRN in the summary plots. If this is the case, and the user wants to get rid of the duplicates, this option can be activated. Insert “1” in row 33, column 1.

1. **File format – Plots**

In row 35, column 1, insert the type of file you want to generate. “.png” will save plots in .png format, “.pdf” will generate plots in .pdf format, etc.

1. **Font Size**

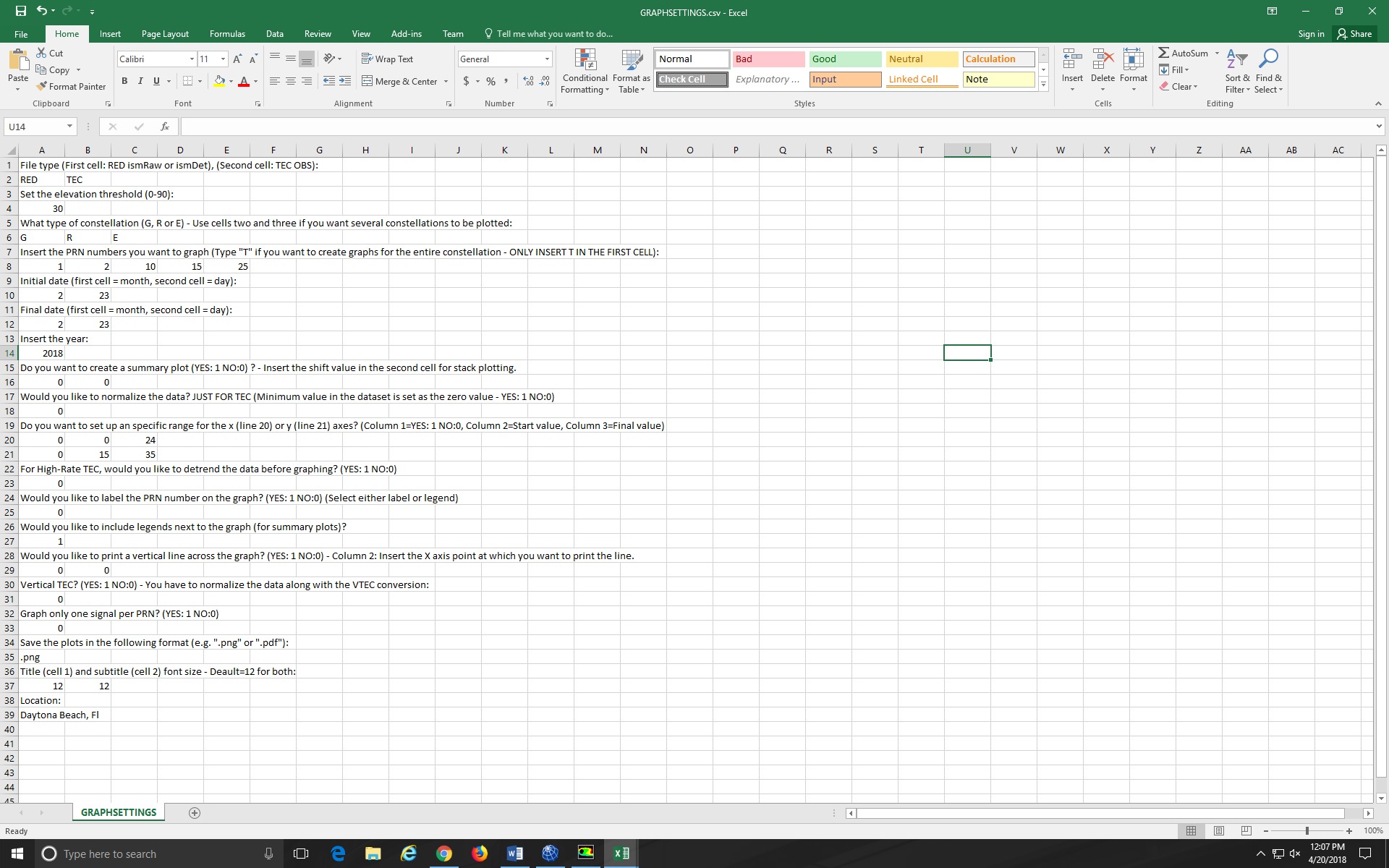
Row 37: If you want to change the font size for the title and subtitle of the plots, change column 1 for the title’s font size, and column 2 for the subtitle’s font size. Default values are “12” for both of them.

1. **Location**

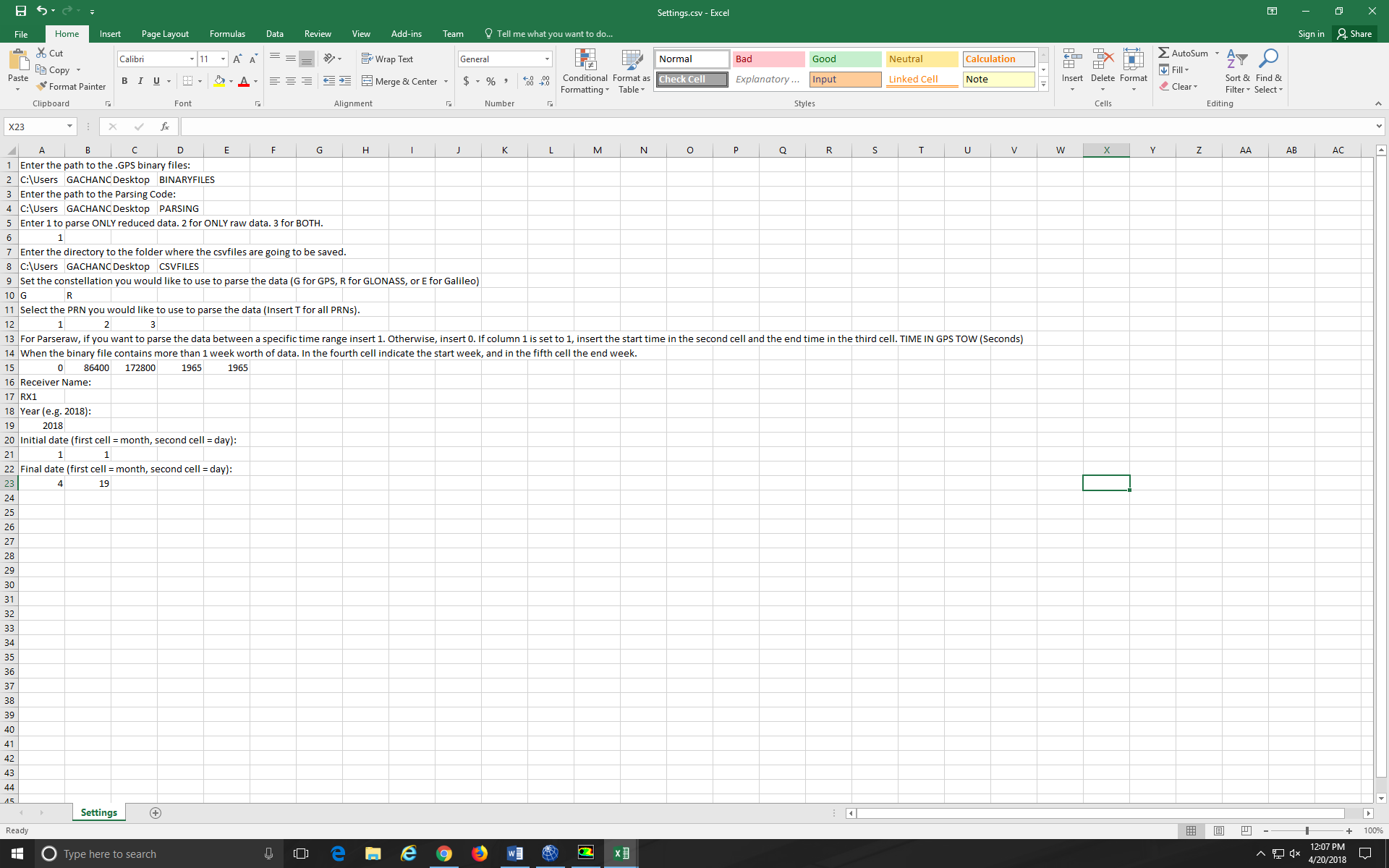
Insert the location where the receiver was placed when it collected the data. This location will be included in the title of the plot. Insert the location in row 39, column 1.

**SNAPSHOTS**

**GRAPHSETTINGS.CSV**



**SETTINGS.CSV**



**PATHS.CSV**

