**APPENDIX ZZZ PROCESSING THE TEXT FILE OUTPUT FROM THE LOGGER**

The raw text file produced by the EC-GPS logger (“LOG\_DATA.TXT” or “LOG\_MMDD.TXT”) contains the following parameters printed to each row: year, month, day, hour\_UTC, minute, second, lat, lon, altitude, satellites, gps\_fix, gps\_fix\_qual, batt, and GS3\_data. Each of these parameters gives information about the status of the sensor, GPS, or logger. Before the data can be used or plotted on a map, it must be translated from the logger output to understandable values. This can be done with functions in a program such as Microsoft Excel or with the Shiny Apps described below. A description of each parameter and how it needs to be processed is included in Table 1.

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Processing Required** |
| year | Date and time retrieved from the GPS | Time is in Universal Time Coordinated “UTC” and needs to be shifted to local time zone; additionally if there is a poor satellite connection these values may be obviously incorrect. |
| month |
| day |
| hour\_UTC |
| minute |
| second |
| lat | GPS Latitude (degrees minutes direction) | Correct formatting to decimal degrees; ie 7654.321N = (76+54.321/60)°, and 12345.678W = -(123+45.678/60)° |
| lon | GPS Longitude (degrees minutes direction) |
| altitude | GPS Altitude (meters) | -- |
| satellites | Number of satellites being tracked | -- |
| gps\_fix | Integer indicating GPS fix (0 = No GPS fix, 1 = GPS fixed) | -- |
| gps\_fix\_qual | Integer indicating the quality of the GPS fix (0 = invalid, 1 = SPS fix, 2 = DGPS fix, 3 = PPS fix, 4 = RTK fix, 5 = RTK Float, 6 = estimated location, 7 = Manual input mode, 8 = Simulation mode) | -- |
| batt | The voltage in the battery | To convert to volts multiply by 2\*3.3/1024 |
| GS3\_data | Sensor address + Dielectirc + Temperature(°C) + EC(uS/cm) | Handle null values and separate out the Temperature and Electrical Conductivity. |

*Table 1. Parameters logged by the EC-GPS Logger*

**Using an R Script of Shiny App to process the raw EC-GPS Logger Data**

1. **Introduction**

This program was developed to process data from the EC-GPS logger using a Shiny App in the programming language R. There are three versions that can be accessed online or downloaded and run locally. See Table 2 for a comparison of the features.

The program principally does four things:

1. It reads the text file (‘LOG\_MMDD.TXT’ or “LOG\_DATA.TXT”) produced by the EC-GPS logger
2. It processes and formats the date and time information from the satellite, converts the GPS locations to decimal degrees, and plots the data on a map
3. It processes and formats the data from the sensor
4. It allows the user to specify the local timezone, filter by date or electrical conductivity (EC), and to download the processed data into a .csv format which can be read in Microsoft Excel, or another text editor.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Online Version** | **Local R: GoogleMaps** | **Local R: Offline** |
| Google Maps interface | Yes | Yes | Yes |
| Interactive data details on map | Yes | Yes | No |
| Internet required | Yes | Yes | No |
| Programing familiarity required | No | Some | Some |
| R installation required on computer | No | Yes | Yes |
| R Package dependencies | None | ‘shiny', 'readr', 'DT', 'RJSONIO', 'dplyr', 'tidyr' | ‘shiny', 'readr', 'DT', 'dplyr', 'tidyr', ‘sp’, ‘broom’, ‘ggplot2’ |

*Table 2. Comparison of EC-GPS data processing versions*

1. **If using the Online Version**
2. Go to shiny app location <https://decolonizingwater.shinyapps.io/EC-GPS-Logger/>
3. Upload desired data log file and manipulate (can use included file 'LOG\_MMDD.TXT' to test)

a. set time zone

b. filter by date (if multiple files have been copied together)

c. filter by EC value (check box and sliders)

d. plot to map: "Plot Data"

Note 1: data needs to be replotted after filters are changed

Note 2: clicking on data points on the map displays the details associated

e. Download Processed Data to .csv file that can be opened in Microsoft Excel or other text editor

f. Additionally Data can be sorted and searched using the Search bar and controls on the data table

1. **If using one of the Local R methods**

1) Make sure that R is installed on your computer (R downloads with RGUi by default; RStudio is optional but a bit nicer); these programs were built with R version 3.5.0 (2018-04-23)

R can be installed from here: <https://cran.rstudio.com/>

RStudio can be installed from here: <https://www.rstudio.com/products/rstudio/download/#download>

Good instructions about setting up R and RStudio can be found here: <http://stat545.com/block000_r-rstudio-install.html>

2) Make sure that all needed R packages are installed and up to date

a) Needed packages can be installed with the following code in R or RStudio (if asked do install dependencies): install.packages(c('shiny','readr','DT','RJSONIO','dplyr','tidyr','sp','broom','ggplot2'))

b) If some of the packages were previously installed they can be updated with the following: update.packages(c('shiny','readr','DT','RJSONIO','dplyr','tidyr','sp','broom','ggplot2')) Note: alternatively all packages can be updated with: update.packages(ask = FALSE, checkBuilt = TRUE)

3) Ensure that the needed files are in the same folder; either

a) **GoogleMaps**: 'ui.R' file, 'server.R' file, and 'www' folder (containing 'map.html'), or

b) **Offline**: 'ui.R' file, 'server.R' file, and 'GADM\_2.8\_CAN\_adm1.rds' file

4) Run the selected Shiny app (GoogleMaps or Offline) in RGUi or RStudio

a) **RGui**: run ‘shiny::runApp('YOUR CONTAINING FOLDER LOCATION')’

containing folder location should be something like

'C:/Users/userName/Documents/EC-GPS/GoogleMaps', or

'C:/Users/userName/Documents/EC-GPS/Offline'

b) **RStudio:** Two options (1) open ui.R or server.R and click "Run App" button, or

(2) in Console run 'shiny::runApp()' or 'shiny::runApp('YOUR CONTAINING FOLDER LOCATION')';

containing folder location should be something like

'C:/Users/userName/Documents/EC-GPS/GoogleMaps', or

'C:/Users/userName/Documents/EC-GPS/Offline'

Note: for running the "GoogleMaps" Shiny App in RStudio

(depending on RStudio settings) you may need to click the "Open in Browser" button

5) Upload desired data log file (i.e. LOG\_MMDD.TXT) and manipulate

a. set time zone

b. filter by date (if multiple files have been copied together)

c. filter by EC value (check box and sliders)

d. plot to map

i) for GoogleMaps: "Plot Data"

Note1: data needs to be replotted after filters are changed

Note2: clicking on data points on the map displays the details associated

ii) for Offline: (1) choose provinces to plot

select from the drop-down menu,

provinces can be removed by using the delete/backspace key in the dilog box

(2) Change the map Zoom

"region" shows the provincial extent

"data" shows just the data extent

e. Download Processed Data to .csv file that can be opened in Microsoft Excel or other text editor

f. Additionally Data can be sorted and searched using the Search bar and controls on the data table