Steps to Process Raw SLAP Radiometer Data into Geolocated Google Earth Images

Refer to the SLAP Handbook in the same directory for more detailed explanations of this process.

If there are any ancillary scripts not listed that MATLAB requires, they will be in the SLAP Working Directory folder on the SLAPUSB2 Drive.

MATLAB scripts needed

SLAP\_read\_multiple\_radiometer\_data\_files.m

SLAP\_read\_raw\_radiometer\_data.m

importOxTSPostprocessed\_v2.m

CombineAncillaryData.m

m1\_m2\_correcter.m

check\_for\_noise.m

noise\_remover.m

radiometer\_data\_consolidation\_2015\_radar\_test\_flights.m

radiometer\_plotter\_2015\_radar\_test\_flights.m

radiometer\_data\_consolidation\_iPHEx.m

radiometer\_plotter\_iPHEx.m

SLAP\_publisher\_2015.m

Procedure

1. Locate radiometer observations for day of interest
   1. Should be in format like: RADTELEM\_20140505T131000.slapbin

|  |  |
| --- | --- |
| Abbreviation/Number | Meaning |
| RAD | Radiometer |
| TELEM | Telemetry |
| 2014 | Year |
| 05 | Month |
| 05 | Date |
| 13:10:00 | Time in UTC |
| 175 | Orbit Number |
| Slapbin | Binary file |

* 1. Process it with “SLAP\_read\_raw\_radiometer\_data.m”. All data files for that day can be processed with “SLAP\_read\_multiple\_radiometer\_data\_files.m” if they are in the same folder.
  2. Then there are MATLAB workspace files for each ten minute radiometer data set.

1. Perform the m1/m2 correction using “m1\_m2\_correcter.m”
   1. This script creates another MATLAB workspace file with the variables “h2ant” and “v2ant” which are m1/m2 corrected h-pol and v-pol antenna observations, respectively.
2. Next remove the 1.09 Hz noise that is common in the radiometer observations prior to and including the Feb. 22nd 2015 flight. This script then saves new variables called h2new\_v2 and v2new\_v2 without the 1.09 Hz noise.
3. Next run “radiometer\_ data\_consolidation\_2015\_radar\_test\_flights.m” or “radiometer\_data\_consolidation\_iPHEx.m” to get geolocated six degree (in scan angle) averaged h-pol and v-pol pixels based on which data set is being processed. The scripts are commented extensively so only the inputs are discussed here.
   1. The GPS receiver data is in an excel file that must be imported into MATLAB using “importOxTSPostprocessed\_v2.m”.
   2. This produces a file with a format like: OxTSPOSTPROCESSED\_150212\_144640.mat
      1. This MATLAB workspace file contains all of the GPS receiver data for that flight including time tags, altitude, latitude, longitude, pitch, roll, heading, tracking angle, and velocity east, velocity down, and velocity north.
      2. The time tags are in GPS time while the other data (scan angle, radiometer, housekeeping temps) are in UTC time so the GPS time tags must have 16 seconds added to them to convert to UTC time.
   3. The rest of the code is well commented and should be easy to follow.
4. The next script to run is “radiometer\_plotter\_2015\_radar\_test\_flights.m” or “radiometer\_plotter\_iPHEx.m” which generates the Google Earth image of the flight swaths.
   1. The sky and box observations are closely examined to determine a period of stable observation that can be used for calibration. The same m1/m2 correction must be used and the 1.09 Hz noise manually removed.
   2. The housekeeping temps are imported and combined into one file for the entire flight in “CombineAncillaryData.m” which is needed in the plotter script.
   3. The brightness temperature corrections were given by Ed Kim. He has a document that explains the corrections.
   4. There is an optional section that writes all of the accumulated data into tabular format.
   5. There is another test section for conversion of brightness temperatures into soil moisture values using the single channel algorithm which needs more analysis.
   6. The final section writes the geolocated brightness temperature data into a KML file that is readable by Google Earth.
5. Auto report of radiometer antenna observations, scan angle data, GPS geolocation data, and housekeeping temperatures can be autogenerated using “SLAP\_publisher\_2015.m”