**Raw to L0 Processing in MATLAB**

This sequence of instructions converts all downloaded BOEM lander data into MATLAB files for further processing. Before following the steps here, make sure:

1. All data is either downloaded and stored in a distinctly named **BOEM\_Dep[12345678]** folder with individual **station ID** folders on the computer that will run the following codes. OR, the computer is on the network and the **ONEDRIVE BOEM Data** folder is found.
2. That an empty folder called **L0** is created within each **station ID** folder.
3. Make sure that the latest **Functions** toolbox is downloaded from GitHub and that includes the **RDADCP** and **RSKTools** toolboxes.

Steps to process raw to L0:

Open MATLAB and create a new .m file that will serve as the master code to run raw to L0proccessing. See the code, **BOEM\_Dep1\_FPSC0\_L0.m** as an example.

1. Change the addpath(…) line to correctly identify where the Functions toolbox is downloaded.
2. For each station, the following variables should be created and edited each time:

*depnum = xx; %the deployment number*

*StationID = ‘FPSxx’; % Station ID to process*

*basepath = ‘xx/xx/xx’; %parent path to the deployment folder*

**Instrument path names** are potentially unique to each deployment and station depending on how the data are exported and saved in **step A** above. But enter as many as you need. Eg., :

*fname\_c6 = ‘FPSC\_C6.csv’;*

**Start and End times** are unique to each station. All times should be in GMT. Consult deployment/recovery logs

1. This ends the typical (no QC Flagging) processing. This section 3 then runs an individual load code that is instrument dependent

SBE37 CT

* Two possible MATLAB functions are called to process SBE37 raw .asc files. **SBE37\_load\_nopressure.m** for instruments that do not have a pressure sensor. **SBE37\_load.m** for instruments that do have a pressure sensor. An input switch of 1 = if sound speed is calculated or 0 = if sound speed is not is added). See the 5th column in the .asc file to see if values of 1500 are there for sound speed.
* Fname\_sbe\_full is the filename that serves as input.
* Output a structure saved as a .mat file into the L0processing folder.

RBR SoloT, Duet and Tridente

* Three types of RBR instruments are possibly used. Each one has their own .m file, use each as needed: **RBRsolot\_load.m, RBRduet\_load.m, RBRtri\_load.m**
* Fname\_rbr[sdt]\_full is the filename that serves as input.
* Output a structure saved as a .mat file into the L0processing folder.

RDIWH 1200 KHz ADCP

* The MATLAB function **RDIWH\_load.m** runs with deploypath as the input, instead of the filename. The function uses the raw spectrum files (individual output from WAVESMON), wave log9.txt wave bulk statistics, and current data \*.PD0 files. If waves were not processed, the function skips that part.
* A .mat file with the instruments SN will be placed in the L0processing folder.

1. As the **Run the codes** section progresses, each instrument will have basic plots appear. One to pay attention to is the **RDIWH\_load.m** code, as it produces a plot of the beam amplitude at each vertical level in the water column. Take note of an average surface bin (high intensity), to be used in L1 processing.
2. After all load and save functions above are complete, move to **L0\_to\_L1\_BOEMprocessingInstructions.docx**

**APPENDIX:**

If not currently in your path, download and put the RDADCP toolbox in your path for MATLAB to pull from when running RDIWH\_load.m function. Here’s a link to the website to download the toolbox: <https://www.eoas.ubc.ca/~rich/>

The RSKtools toolbox is needed to process the “.rsk” file. It can be downloaded at this link: <https://rbr-global.com/support/matlab-tools/> . The toolbox needs to be setup in your path when running the RBRtri\_load.m function in MATLAB.