**Documentation for RemoteLogger Library**

# Documentation Wishlist

User-facing documentation:

* Instructions on full setup of Arduino IDE for the Feather Adalogger
* Instructions on installing the RemoteLogger library from GitHub
* Instructions on downloading your data from the database website (if you’re using Alex’s database)
* Instructions on installing the libraries you need for the RL library
* Instructions on custom-designing your own sketches with the RL library for sensors that are not supported
  + User-facing documentation of each public function (arguments, return, description)
* Introduction to the project
* License for use (open source)
* Update wiring diagrams and materials lists
* List of sensors supported by the library and links to them
* Instructions on field setup and testing
* Instructions on recommended bench testing
* Acknowledgements

Documentation for source code:

* Description of the library source code
  + Scope and design choices (what was included/left out)
  + Likely changes/directions of improvement and things that are necessary to keep
  + Limitations and things to look out for
  + Dependencies between functions/variables
* List of letters used for different measured parameters in the database and what they correspond to

# User-Facing Docs – OUTLINE

1. Introduction to the project
   1. ~~What is the project for? What are its benefits?~~
   2. Acknowledgements
   3. Licensing – how can you use this product?
2. Getting started
   1. ~~How to download, install, and set up the Arduino IDE for use with the Feather Adalogger~~
   2. How to download and install the RemoteLogger library
   3. ~~How to install necessary libraries for the RL library to work~~
   4. ~~How to access examples from RL library, compile, and upload to the Feather Adalogger~~
   5. Troubleshooting during setup
3. Building a datalogger
   1. Supported sensors
   2. Wiring diagrams and materials lists
   3. Build instructions
   4. Recommended bench tests
   5. Field deployment and testing
   6. Accessing your data (if you’re on Alex’s database)
   7. Ongoing maintenance/troubleshooting
4. Library functions
5. Beyond the library – designing your own network
   1. Writing sketches with sensors not supported by the library
   2. Setting up Iridium RockBlock system for your own modems
   3. Overriding library functions (e.g. writing your own send\_msg for a different satellite/cellular/wifi modem)
   4. Setting up your own database

# Source Code Docs – OUTLINE

1. Scope
   1. What was included
   2. What was not included – left to the user
      1. How to help the users figure out how to do their part
   3. Why things were included or left out
2. Limitations and improvements
   1. Limitations on the design by hardware/external systems (e.g. Iridium’s send timeout determining how much data is too much to try to send)
   2. Limitations from message structure on scalability
3. Dependencies
   1. Dependencies between functions/variables
4. Interface with database
   1. Message structure to be processed properly by database
   2. Letters that are mapped to particular messages