RSI Technical Note 050

Rockland Data Logger (RDL/DL-3) Integration Guide

v1.1

Brett Prairie
JP Juteau

2020-06-23

Rockland Scientific International Inc.
520 Dupplin Rd
Victoria, BC, CANADA, V8Z 1C1
www.rocklandscientific.com

Contents

List of Figures

List of Tables

1 Overview

The Rockland Data Logger (RDL), or ”DL-3” as it is known internally, is a third-generation data logger, designed by Rockland Scientific. The RDL replaces Rockland’s second-generation data logger (powered by the Persistor CF2 computer).

The RDL is powered by a Cortex-A8 1GHz, SoC that has 512 MB of RAM and uses a Linux Kernel (4.19) for the operating system. The OS build is customized for the data logger’s hardware. The OS is able to run at up to 1 GHz speed and utilize up to 512 MB of RAM.

1.1 Power Consumption

The Cortex-A8 processor is responsible for the majority of the power consumption on the data logger, drawing approximately 0.6 W of power during data acquisition.

The power consumption of instruments with the RDL is expected to be 0.35 W [1] higher than the CF2-based system. The estimated power consumption for a MicroRider with a CF2 and RDL are listed in Table 1:

Instrument Configuration	CF2	RDL
MicroRider (standalone)	1.00 W	1.35 W
MicroRider (with EM Sensor)	1.60 W	1.95 W

Table 1: Estimated power consumption

1.2 Powering On

To turn an instrument on, one must provide the instrument with a 9-18 VDC source of power as well as provide an ON signal to the instrument.

There are three ways to send an ON signal to an instrument:

1. Applying a 1 mA current to the ON signal pins.
2. Shorting the connection between the ON signal pins.
3. Issuing a system command (Section 3.1).

The first two methods require the instrument’s bulkhead connector pins to be wired to the appropriate internal ON signal input. In addition, the bulkhead connector pins vary depending on the instrument model. Consult with Rockland Scientific for the specific wiring required for your system.

[1] Future software and firmware development have been identified that could result in the RDL drawing an average of 0.15 W. This would mean that a MicroRider-1000 with RDL as a whole would draw 0.9 W during DAQ. Rockland does not have a timeline for this release.

2 Interfacing with the data logger

2.1 Establishing Console Access

Console access can be established with the data logger by connecting to an RS-232C port or by connecting to USB (Table 2).

Serial Port	External Connection	Baud Rate
RS-232C #1	Console access	115200
RS-232C #2	Console access with DAQ pass-through[2]	115200
USB0	Console access	N/A

Table 2: Serial connections

Notes:

- The boot sequence could take up to a minute.
- The system automatically logs in and goes directly to a default command prompt.
- The command prompt could change in future releases. The provided command prompt is the default command prompt provided by the embedded Linux OS. Scripts that look for the command prompt should be written accordingly.
- Data acquisition will start automatically if enabled on boot (Section 3.2) and the instrument is powered by a 9-18 VDC source.
- When DAQ starts and stops, interruptions to serial communication can occur when connecting via USB.
- If connecting via USB, any baud rate is acceptable (it can even be undefined for maximum speed).

[2] Data Acquisition launches in the foreground when communicating through this port (Section 3.3). The same behaviour exists on CF2 systems.

2.2 Connecting to USB

If one wishes to transfer files to/from a PC, simply connect your PC to the data logger's Micro-USB connector. Alternatively, if supplied with an assembled MicroRider, connect the MicroRider to your PC using the supplied deck cable.

Once a USB connnection has been established to the data logger, the "user storage" location (where data and configuration files are located - see Section 4.1), will be available to the PC via USB Mass Storage (Section 4.2).

To prevent corruption of the filesystem, if data acquisition running, access to the user storage via USB Mass Storage will be restricted to "read only".

3 Data Acquisition

Before data acquisition is possible:

1. The instrument is supplied by a 9–18 VDC power source and has been turned ON (Section 1.2).
2. A valid configuration file (`setup.cfg`) is present in the root directory of the user storage (Section 4.1).

3.1 ODAS Commands

ODAS stands for Oceanographic Data Acquisition System and this acronym is used in the commands for controlling data acquisition on the data logger. Table 3 contains a list of available commands for configuring the system and controlling data acquisition.

Command	Action
`odas_disable`	Disable DAQ on boot
`odas_enable`	Enable DAQ on boot
`odas_help`	Provides a list of available ODAS commands and their descriptions
`odas_p050_on`	Supplies an ON signal to the instrument
`odas_p050_off`	Supplies an OFF signal to the instrument
`odas_start`	Starts DAQ. Interfacing with DAQ is only possible from the RS-232C #2 port.
`odas_stop`	Stops DAQ. Only used when not connected to the RS-232C #2 port.

Table 3: A list of ODAS commands

3.2 Disabling/Enabling Automatic Data Acquisition on Boot

Instruments can be configured to launch data acquisition on boot. However, for instruments that require data acquisition to be controlled by a host, this might not be a desirable feature. To change the boot behaviour of the ODAS service, use the `odas_enable` / `odas_disable` commands (Table 3).

3.3 Starting/Stopping Data Acquisition

Data acquisition can be started in two ways:

1. typing `odas_start`
2. or by enabling automatic DAQ on boot.

Data acquisition can be stopped in two ways:

1. typing `q`
2. typing `odas_stop` (only used when not connected to the RS-232C #2 port).

3.4 Channel Statistics

Channel statistics[3] can be requested during data acquisition for helping determine if the instrument and its sensors are functioning normally. For some users, this can be vital for making operational decisions.

Channel statistics can be requested by:

- typing `s`

and the resultant output is streamed from the serial connection shown in Figure 1).

[3] for all channels that are included in the address matrix section within the instrument's configuration file: `setup.cfg`.

Querying the channel statistics will provide the user with the following information:

- The time in UTC for when the data was recorded
- The data file for the record
- The record's number within the data file.
- Channel number and channel name (ID)
- Minimum value recorded (MIN) for each channel
- Maximum value recorded (MAX) for each channel
- Channel average recorded (AVG) within a record
- Pseudo-standard deviation recorded (STD) for each channel
- The number of bad buffers (ERR) for each channel

Rockland recommends that a platform query channel statistics in the middle of profiling (as opposed to when it is on the surface). In addition, please ensure that any buoyancy engines or other vibration-inducing motors/actuators be off when channel statistics are obtained.

When channel statistics are requested, statistics for the most recently observed record are provided. The DAQ system buffers up to two minutes worth of records before providing them to the CPU. As such, when requesting channel statistics, the returned statistics could be up to two minutes old. In addition, requesting channel statistics multiple times without delay will result in statistics for the same record being returned.

Channel Statistics

UTC Time: Wed May 20 20:07:44 2020

File: Dat_1030.p, Record Number: 62

ID		MIN	MAX	AVG	STD	ERR
0	Gnd	-2	-1	-1.3906	0.48789	0
3	AN4	-3	0	-1.2969	0.69999	0
13	AN3	-3	0	-1.0781	0.6201	0
14	AN2	-3	0	-1.5	0.66144	0
15	AN1	-3	0	-1.2969	0.72197	0
1	Ax	-11	2	-4.8242	3.1617	0
2	Ay	-10	4	-3.084	3.219	0
4	T1	-19	-16	-17.594	0.70086	0
5	T1_dT1	-40	-5	-17.414	3.8901	0
6	T2	19	21	20.234	0.65532	0
7	T2_dT2	9	29	19.82	3.3329	0
8	sh1	-25	17	-4.3281	12.572	0
9	sh2	-13	9	-2.1602	6.357	0
10	P	654	657	654.86	0.65829	0
11	P_dP	653	674	657.44	5.7984	0
12	PV	-580	-577	-578.41	0.60515	0
32	V_Bat	23432	23463	23441	7.5392	0
33	Relea	23430	23453	23440	6.8613	0
40	Incl_Y	-7	-7	-7	0	0
41	Incl_X	-121	-121	-121	0	0
42	Incl_T	1287	1287	1287	0	0

Figure 1: Channel statistics displayed while running ODAS in the foreground

4 File Transfer

4.1 User Storage

The user storage is contained on a memory module board (P115) connected to the RDL/DL-3 (P113). The contents of the user storage include their configuration file, log files and data files (all within the root directory). It is also where Rockland System Update files (.RSU) should be placed when users need to update the software/firmware on the data logger. When accessing the user storage via a serial connection with the data logger, the user storage is located in the `/home/debian/data` folder.

4.2 USB mass storage

You may wish to establish a connection to the user storage using your PC's file explorer (i.e. Windows Explorer or MacOS Finder), via a USB connection, to transfer files between your PC and the instrument:

1. Ensure the instrument has been turned off. Rockland recommends waiting for at least one minute.
2. Connect the Deck cable to your computer. Note: To access the user storage via USB mass storage, the instrument must only be powered via USB (Section 2.2).
3. After roughly a minute, a new mass storage device will appear on your computer.
4. The mass storage device should look like a new USB thumb drive connected to your computer.

Transferring files between your PC and the memory module can be done using standard methods (i.e. copy/paste or drag and drop). Before disconnecting USB cable from your computer, ensure that the memory module is properly ejected from your PC.

4.3 X/Y/Z Modem

The data logger supports X/Y/Z Model for file transfers. When the instrument is powered from the 9-18 VDC, the data is stored in the `/home/debian/data` folder.

5 Applying System Updates

The data logger has the capacity to be updated with *"*.rsu"* (Rockland Scientific Updater) files. In order to update the system, copy the *.rsu* file to the memory module and reboot the instrument.

6 Troubleshooting

There are several sources of feedback that are provided by an instrument, some which can be valuable for troubleshooting the system.

6.1 Internal LED indicator (located on the data logger)

There are three green LEDs located in the corner of the board near the serial instrument bus cable. They are labelled D17, D19, D20.

- D17 – heartbeat indicator. This will flash regularly in a heartbeat fashion to show that the system is alive and responsive
- D19 – CPU activity indicator. This LED will light up proportional to the CPU usage. Solid green, means the CPU is active at 100
- D20 – Memory module activity and ODAS active indicator. This LED serves two purposes. It will be solid on when ODAS data acquisition is running and otherwise indicate the memory module activity. Note, when transferring data to and from the memory module the LED is more likely to blink occasionally than be solid ON.

6.2 External LED indicator (if equipped with an instrument)

It is possible to equip an instrument with an external LED for providing feedback to a user as to whether the instrument is booting up, starting data acquisition or whether an error has occurred with the instrument. Table 4 highlights the behaviour of the external LED.

Flashing Pattern	Meaning
"Slow" blink[4]	The instrument is booting up, or the instrument has finished booting up, but data acquisition has not yet started (or did not start)
constant ON	Data acquisition has started. When data acquisition starts, the LED should remain continuously ON for 64 seconds.
"Fast" blink[5]	An error has occurred. Consult the log file for information on the error that has occurred (or contact Rockland)

Table 4: External LED patterns

6.3 Logfile.txt

The data acquisition software produces at least two files – a data file (e.g. dat_0001.p) and a log file. The log file contains a record of the events that occurred during data acquisition. The log file is named logfile.txt. Example events from a log file are shown in Figure 2:

```
5ea7d706 11 0000 Undefined  --- 2020-04-28 07:11:02.033 - "power applied"
5ea7d747 10 0000 Undefined  --- 2020-04-28 07:12:07.035 - "start signal received"
5ea7d747 0b 0001 Dat_0662.p --- 2020-04-28 07:12:07.061 - "starting data file"
5ea7d79d 0d 0001 Dat_0662.p --- 2020-04-28 07:13:33.897 - "bad buffer"
5ea7d7a0 07 0056 Dat_0662.p --- 2020-04-28 07:13:36.102 - "manual stop"
5eb49fb9 10 0000 Undefined  --- 2020-05-07 23:54:33.082 - "start signal received"
5eb49fb9 0b 0001 Dat_0919.p --- 2020-05-07 23:54:33.112 - "starting data file"
5eb4a00f 0d 0001 Dat_0919.p --- 2020-05-07 23:55:59.614 - "bad buffer"
5eb4a1e5 04 0200 Dat_0919.p --- 2020-05-08 00:03:49.502 - "power is bad" supply
voltage at 4.75714 V
```

Figure 2: Example of a log file generated by the data acquisition software

[5] 1 second ON and 1 second OFF
[5] 4 successive "short" blinks (0.25 second ON/OFF)

6.3.1 Log File Format

Each line in a log file describes a single event. The first part of each line is in a machine-readable form to support automated scripts. The second part of each line is in human-readable form.

Events are recorded using the following format (with each field defined in Table 5):

```
EVENT = MACHINE_FORMAT --- HUMAN_FORMAT
MACHINE_FORMAT = time event_type record# data_file_name
HUMAN_FORMAT = human_time - ``event_desc"
```

Field	Description	Example
`time`	Time of the event, as given by the UNIX "time" function. Value is in hexadecimal format and always uses the first 8 characters.	`5eb4a1e5`
`event_type`	Type of event identified by a two-digit hexadecimal number. See Table 7 (in section 9.3.2) for a list of event types.	`04`
`record #`	The record number during which the event occurred. Formatted as a 4-character hexadecimal value.	`0200`
`data_file_name`	Name of the data file. If a data file has not been created, the value "unknown" is used. This entry always has a width of 10 characters.	`Dat_0919.p`
`human_time`	Time of the event.	`2020-05-08 00:03:49.502`
`event_desc`	A description for the event type.	``power is bad"
`additional_info`	Optional extra information for the event	`Supply voltage at 4.7514 V`

Table 5: Description of log file entries

6.3.2 Event Types

The event types that are possible to be generated with a DL-3 equipped instrument are show in Table 6.

Code	Event Name	Description
01	pressure release	Release triggered by the FPGA because the pressure exceeded max pressure.
02	fall rate release	Release triggered by the FPGA because the fall rate was less than 0.2 dbar/s for 4 consecutive seconds.
04	power is bad	The power-good signal from the power-supply board (PS) is disabled.
06	switch is off	The power ON/OFF switch was set to OFF.
07	manual stop	Acquisition was stopped manually by an operator.
0b	starting data file	Data acquisition started. The event time corresponds with the creation of a new data file.
0c	acquisition stop	Data acquisition stopped.
0d	bad buffer	Record contains a bad buffer.
10	start signal	A start signal was received. This occurs when ODAS5IR is first launched but before data acquisition starts.
11	power applied	Instrument switched on and power applied.
12	data file trimmed	This event corresponds with power being shut off before data acquisition has stopped. The last record in the data file is removed because it contains invalid data.
13	Release due to low voltage	Release triggered because the input voltage fell below the minimum valid voltage specified in the configuration file.
14	Release triggered by software	Release triggered by the CPU instead of the FPGA. An example is a release generated when data acquisition exits.

Table 6: Possible event types with a DL-3 equipped instrument

6.4 Error Log

When the data acquisition software exits with an error code, an error log is copied to the memory module and time stamped (error_log_YYYY-MM-DD_HH-MM-SS). The error log consists of a series of verbose statements from the different scripts (last 200 logged messaged of the system).

By reading the final entries of the error log, the user should be able to identify the source of the error with their data acquisition. The file is not presented in a format that is friendly to being parsed by an integrator.