renewable energy innovation
electrical solutions for renewable energy systems

RE-LOGGER Installation & User Guide

Table of contents

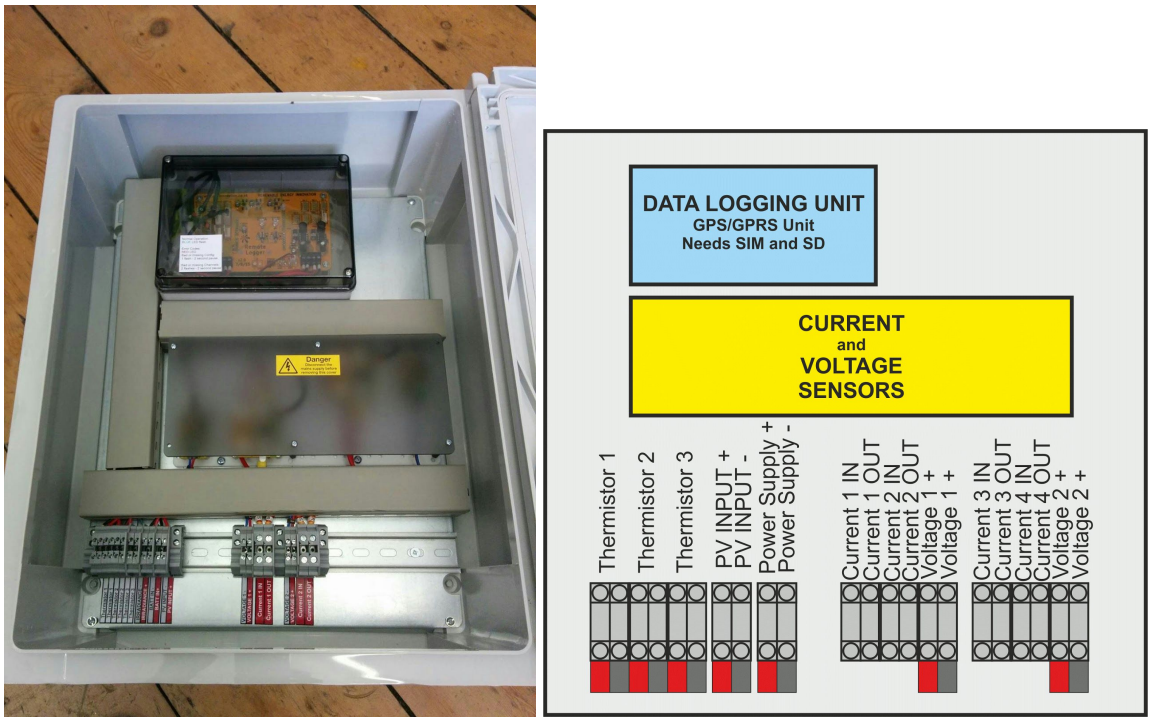
Design overview

This document contains the user guide, installation instructions and a more detailed overview for the RE-LOGGER, a stand-alone datalogger for renewable energy systems.

This work has been commissioned by CREST, Loughborough University, UK for remote monitoring of stand-alone power supply systems, generally remote, off-grid, solar PV systems.

The RE-LOGGER unit is based upon the LinkIt ONE, an open-source IoT base unit with GSM/GPRS connectivity, GPS and SD card slots.

Overview Diagram



This is a general overview of the the unit. The exact number of Current and Voltage sensors depends upon the application.

Specifications

This unit has the following specifications:

- Input supply voltage range: 7-60VDC
- Input current : 500mA (max)
- Internal battery: 2200mAh 3.7V LiPo
- Input PV supply voltage: 7-40V DC
- ADC range: 15 bits
- ADC sensitivity: 0.1mV
- Data Sample rate: 5Hz (averaged over user specified time)
- SD Card size: Up to 32GB

Temperature sensor:

- Temperature range: -10C to 100C
- Temperature accuracy: +/- 1C

Current sensor large:

- Current range: 0-50A DC
- Current accuracy: 50mA

Current sensor small:

- Current range: 0-5A DC
- Current accuracy: 5mA

Voltage sensor low

- Voltage range: 0-100V DC
- Voltage accuracy: 50mV

Voltage sensor high

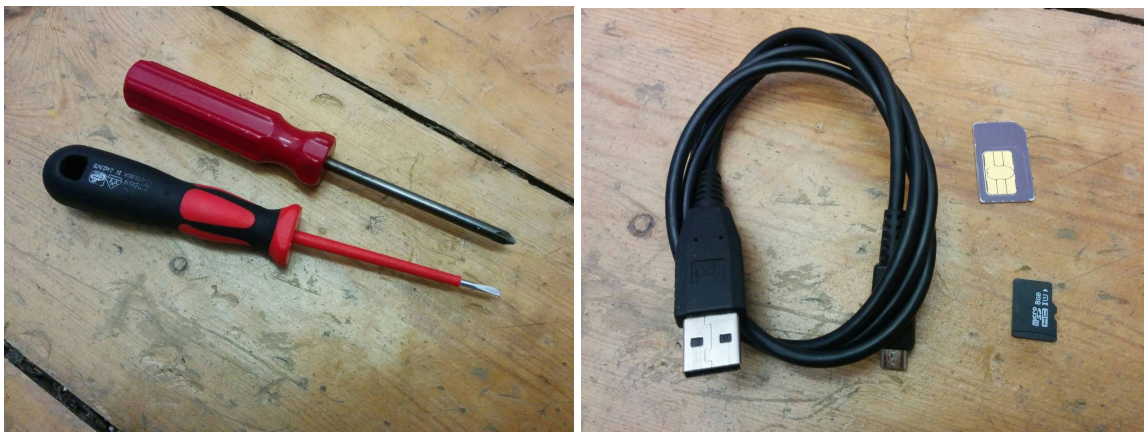
- Voltage range: 0-300V DC
- Voltage accuracy: 500mV

Parts Included

- The main logger unit, pre-wired with sensors
- Solar PV module
- Thermistor temperature sensors x 2

Additional Equipment Required

- A high quality SD card with a capacity 8Gb or greater.
- An active SIM card with data contract and GPRS reception
- A flat head 3mm screwdriver
- A flat head 5mm screwdriver



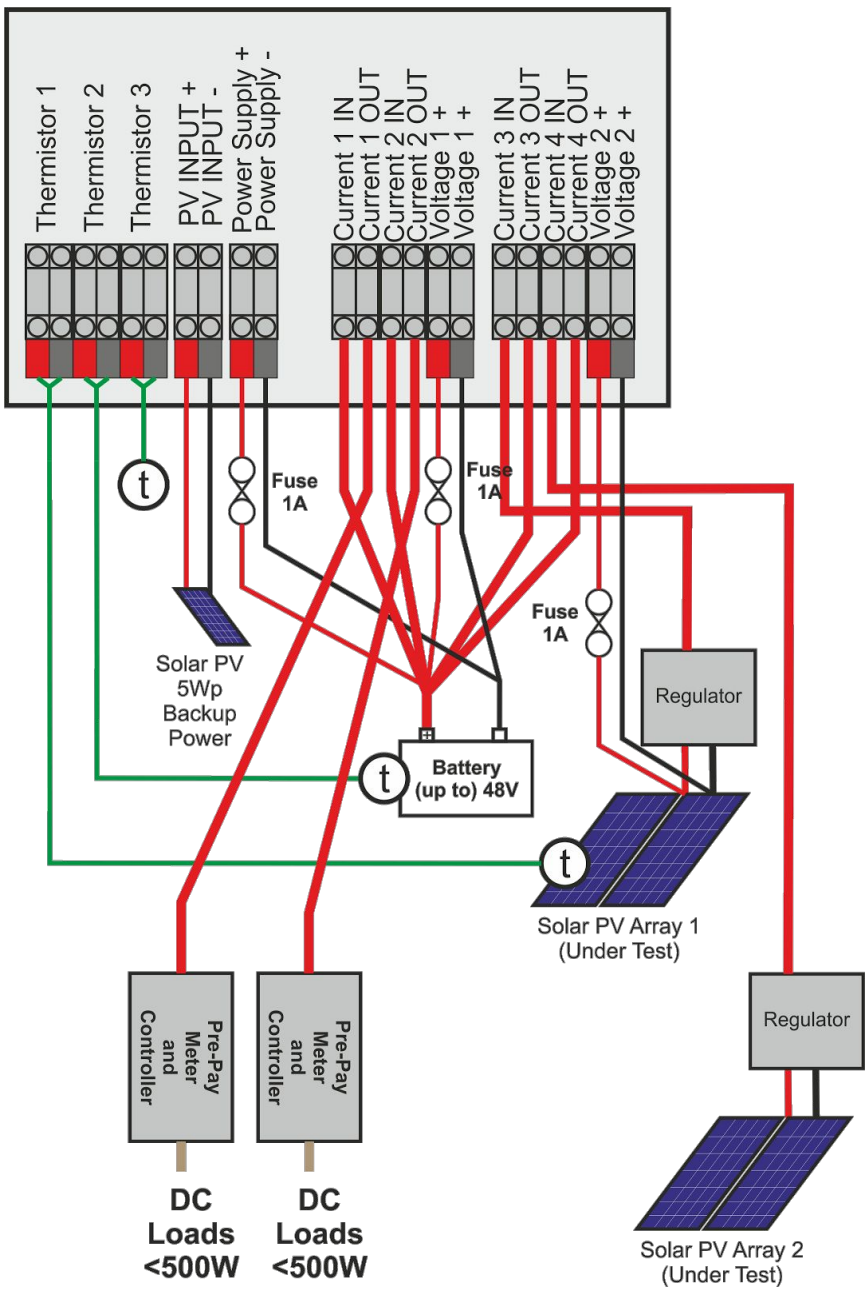
Tools and additional parts required.

Installation guide

Warning: Electrical knowledge is assumed here – if in doubt consult a qualified engineer.
Warning: Batteries and high voltages are dangerous – Stay Safe.

Wiring

Here is a wiring diagram for wiring inputs/output to the unit.



Cable glands

The cables will need to be routed through the base of the plastic enclosure. Drilling holes to fit close-fitting cable glands is the best way to do this.

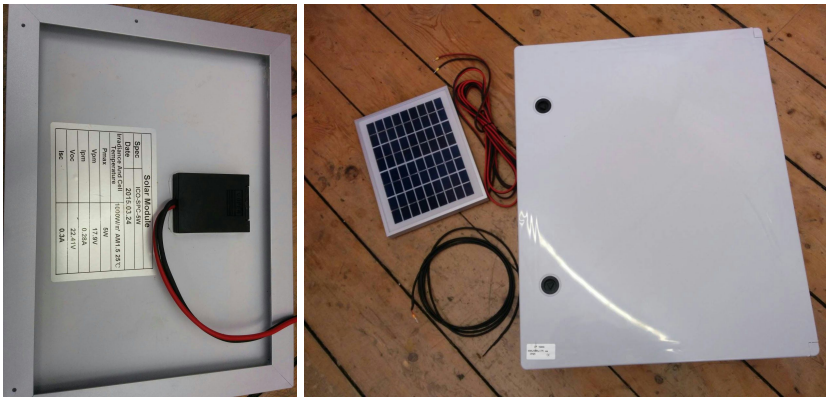
Wiring Connections

Battery connections

The Batt In Terminals are for connecting the unit to the main battery to power the unit. This should have an in-line fuse rated at 1A to protect the unit.

Note: The Ground for the Solar PV input and the Battery input is common.

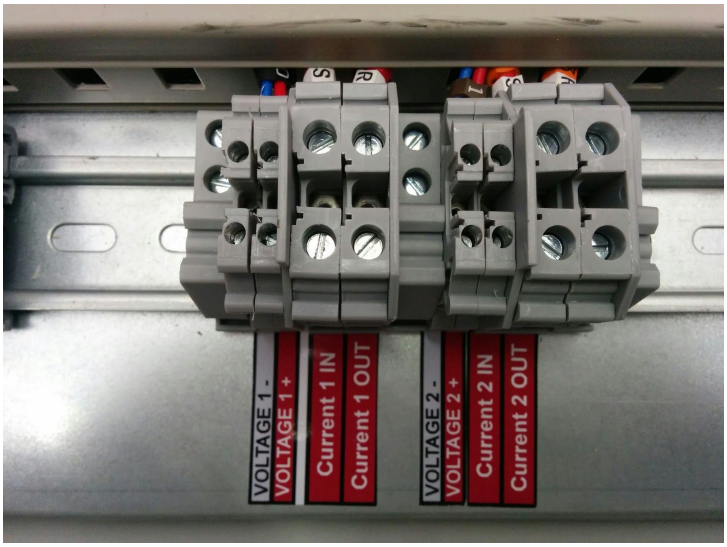
Solar PV Input



A 5Wp solar PV module provides an alternative source of power for the logger. This is included with the logger.
Please wire this to the terminals marked PV INPUT.

Note: The Ground for the Solar PV input and the Battery input is common.

The 5Wp solar PV must be mounted to receive enough solar irradiance to keep the unit powered. This could be on the roof of the power unit or at an angle on the side of the power house. Ensure this is pointing towards the sun (eg South in the Northern hemisphere, North in the Southern hemisphere) and at an angle of around the latitude of the location from horizontal (eg UK: 52 degrees, Philippines: 10 degrees).



Voltage Measurements

Voltage measurements are taken using an isolated instrumentation amplifier. This means it is isolated from any other voltage being monitored.
Voltage measurements should be wired to the Voltage +/- terminals. A 1A in-line fuse would be a sensible precaution, in case of any short circuits.

Current Measurements

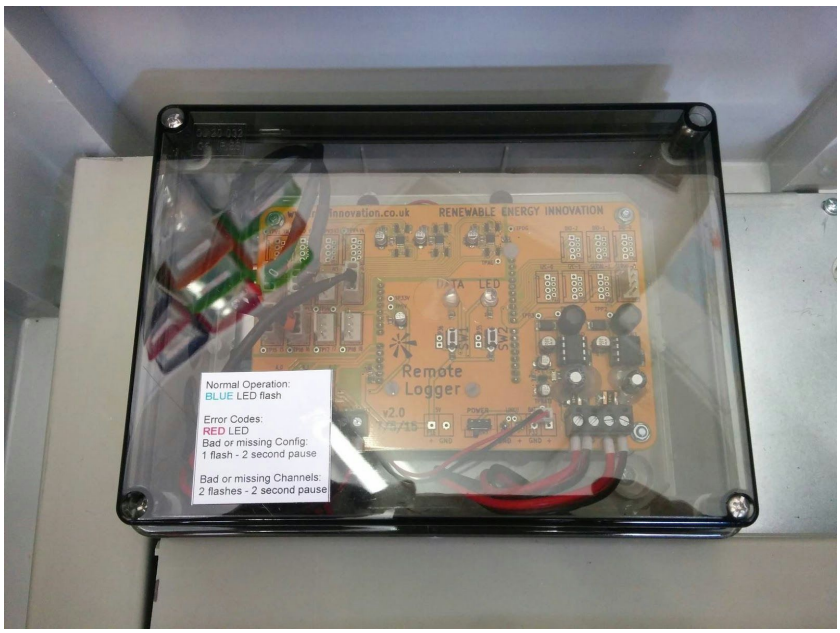
Current measurements are taken using an hall-effect current sensor.
This is isolated from any other reading. The full current will flow through the sensor, so the current-carrying cables must pass through this datalogger unit.
The cables must be sized according to the current they will carry.
Note: A rule-of-thumb is that 1mm^2 will carry around 5A, so 4mm^2 is good for up to 20A.

User Guide

Once installed and set-up there should be very little user interaction with this unit. Checking its correct function every week is a good check, although there are maintenance text messages that will be sent.

User Controls

There are three main user controls: An On/Off switch, a **RED** LED (marked 'LED') and a **BLUE** LED (marked 'Data').



The On/Off switch controls the power. The LEDs will flash when there is power.

The LEDs have the following functions:

LED	Function	Error?
BLUE flash	Unit ON and Working	NO
1 RED flash, 2 second pause	No Settings Config File	YES - check settings config file on SD card
2 RED flashes, 2 second pause	No Channels Config File	YES - check channels config file on SD card

Hardware Overview

Note: The hardware has been pre-wired and tested. Please install as shown in the Installation section. This overview is for information only.

Hardware Photo

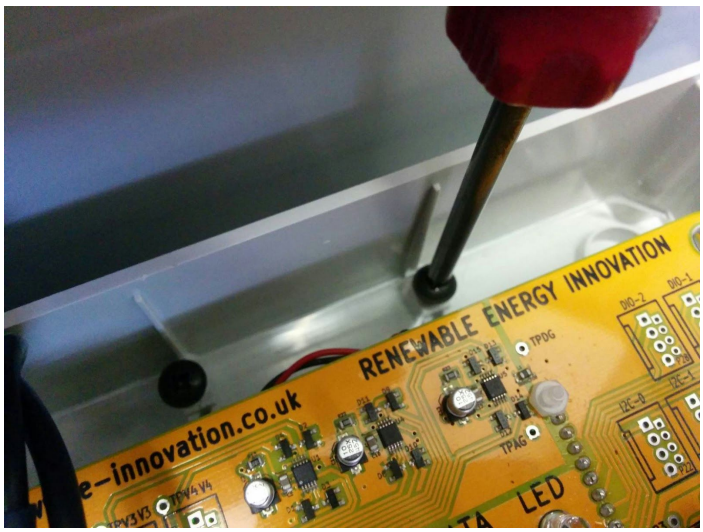
RELOGGER unit

The RELOGGER unit is based upon the LinkIt ONE IoT base unit:
http://www.seeedstudio.com/wiki/LinkIt_ONE
This contains the SD card, GPRS module, WiFi module (not implemented), Bluetooth LBE (not implemented) and GPS module .
This plugs in to a bespoke PCB which contains 3 x analog to digital converters (the ADS1115, http://www.ti.com/lit/ds/symlink/ads1115.pdf), the input power supply unit and the PV input voltage regulator.

SIM Card

Any standard SIM card can be used in this unit, although the exact function of the SIM card can only be tested ‘in the field’.
You will need the data provider's: APN (access point name), Username and Password. These are available from the data provider.

Installation of SIM card



The SIM card is slotted into the back of the RE-LOGGER unit.
Remove the front of the brown enclosure using the four screws on each corner.
Remove the whole logger unit using the four self-tapping screws on the baseplate.

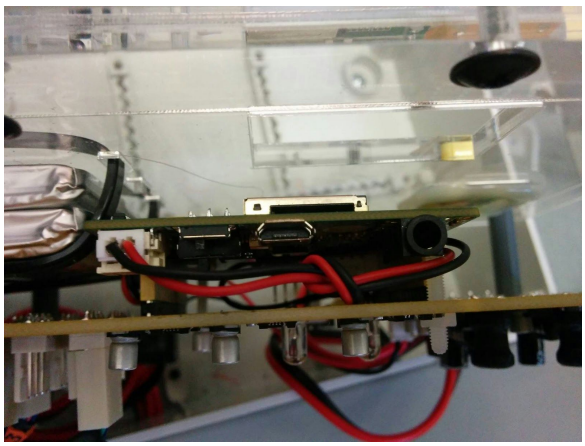
Carefully move the baseplate to get access to the back of this where there is a cut-out.
Slot in the SIM card to the SIM card holder and re-fix the baseplate.
Replace and fasten the brown enclosure cover.

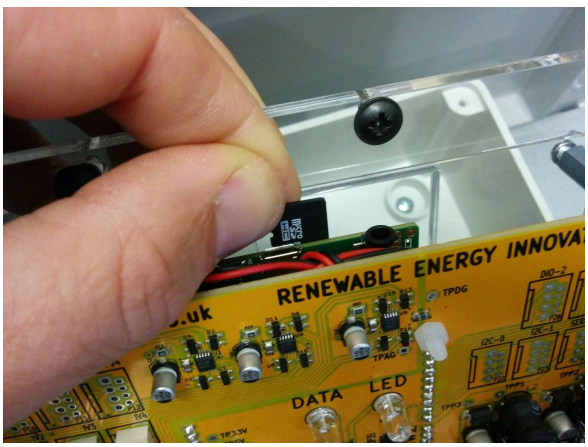
SD card

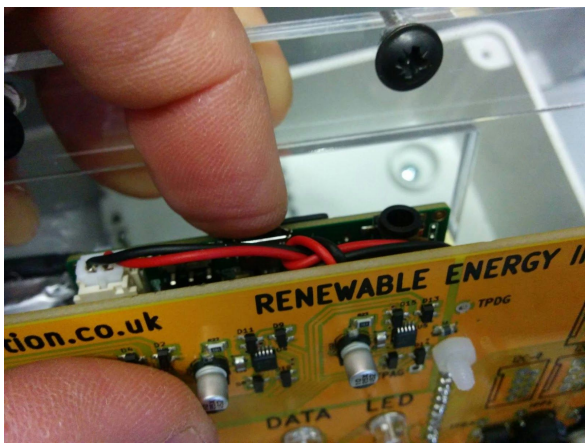
The SD card is installed next to the SIM card.

Installation of SD card

The SD card is slotted into the side of the RE-LOGGER unit.
Remove the front of the brown enclosure using the four screws on each corner.
Remove the whole logger unit using the four self-tapping screws on the baseplate.
Carefully move the baseplate to get access to the back of this where there is a cut-out.
Slot in the SD card into the side at 90 degrees from the SIM card holder.







Re-fix the baseplate.
Replace and fasten the brown enclosure cover.

Battery backup

There is a LiPo 3.7V 2200mAh backup battery to run the logger when input power is lost.
If this battery is low on charge then a text will be sent to the maintenance SMS phone

numbers.

GPS sensor

The GPS sensor is used to update the Real Time Clock. This could take 10-15mins to synchronise. The GPS sensor will not work underground or in a very thick-walled building.

Future Feature (not yet implemented):
Every two hours the location of the unit will be checked. If this has differed from the previous location by more than 50m then a text will be sent to the SMS phone numbers.

Current Sensors

These are separate units based upon the ACS758 Allegro Hall Effect current sensors:
http://www.allegromicro.com/~/media/Files/Datasheets/ACS758-Datasheet.ashx
The 50A uni-directional version is used here: ACS758-X050U.
This is supplied with a regulated 5V supply voltage. The output is 0.6V + 60mV/A.

Voltage Sensors

There are two voltage ranges to measure:
Low: 0-100V DC
High: 0-300V DC
There are two different sensors for these ranges:

Voltage Sensor (Low)

The voltage sensors are fully isolated from the other voltages.
An isolation amplifier is used to provide up to 1000V isolation between the 300V signal and the ADC.
This is performed using an ACPL-C870:
http://www.avagotech.com/docs/AV02-3563EN

A potential divider of 100k and 1k is used to give the input signal, giving an input range of 0-100V DC.

Voltage Sensor (High)

The voltage sensors are fully isolated from the other voltages.
An isolation amplifier is used to provide up to 1000V isolation between the 300V signal and the ADC.
This is performed using an ACPL-C870:
http://www.avagotech.com/docs/AV02-3563EN

A potential divider of 300k and 1k is used to give the input signal, giving an input range of 0-300V DC.

Temperature Sensors

A thermistor is used as part of a potential divider circuit with an accurate and stable reference voltage.

- Low cost.
- Good accuracy (within 1-2C).

A 10K thermistor is used to record the temperature. The thermistor is put as half of a potential divider, with a 10k precision resistor and a precision voltage regulator to supply the potential divider. The thermistor output must have the correct interpolation applied so that the temperature reading is accurate.
There are three inputs for temperature sensors. The actual thermistor sensors must have a datasheet with a K value, as this is required to calculate the temperature.
The thermistor must be thermally bonded to the unit/device under test.
The thermistors supplied with this unit are:

10K NTC Thermistor
Beta Value: 3988
Farnell Part Number: 1299930
EPCOS Part: B57863S0103F040
Datasheet: http://www.farnell.com/datasheets/1899659.pdf

Power supply

The logger is powered by a 2200mAh 3.7V LiPo battery.
This is kept charged through a DC-DC converter power supply from the battery connection or via a solar PV backup.

DC-DC converter

This converts the input voltage of 6-60V DC, down to the 5V used by the datalogger. If possible, this is connected to the battery bank input via a 1A 20mm x 5mm in-line fuse.

Solar PV backup

There is space for a solar PV input to keep the battery topped up. A 5W PV module is provided to perform that function.
This input can be up to 40V DC input. It uses a DC-DC converter to provide the 5V recharge voltage. It does not maximum power point track.

Embedded Software Overview

The embedded software has been written in c programming language. It is designed for upload to the LinkIt ONE unit via the Arduino IDE.

Settings Configuration file

A datalogger application has several configurable options. This document details those options and how the configuration file is written. The configuration file is a small file on the root of the SD card.

Filename & Location

The configuration file **MUST** be stored on the root of the SD card under the name "Datalogger.settings.conf".

File Structure

The configuration file has one setting per line.

Each line has a setting name, an equals sign, and a setting value, e.g.:

THINGSPEAK_URL=agile-headland-8076.herokuapp.com

Setting names are all uppercase. Words in the name are separated with underscores.

Setting values are either strings or numbers. All numbers are read as signed integers.

Spaces on either side of the equals sign are ignored. For example:

THINGSPEAK_URL=agile-headland-8076.herokuapp.com

and

THINGSPEAK_URL = agile-headland-8076.herokuapp.com

will be treated identically.

Comments

Lines starting with a # are treated as comments. The entirety of the line will be ignored.

Comments at the end of a line are not supported. e.g.:

GPRS_APN=giffgaff.com # Some comment here

is **NOT** valid and would result in the comment becoming part of the GPRS_APN setting.

GPRS Settings

There are three settings for GPRS connections:

Setting Name	Notes
GPRS_APN	The APN of the data provider
GPRS_USERNAME	The username for the data provider.
GPRS_PASSWORD	The password (sometimes not required).

These settings are readily available online for various data providers.

Data Storage Services

The datalogger can upload data to remote online data storage services.

Due to differences in how each service works, the settings for one service may differ from another.

Currently the only supported service is thingspeak. Other services may be added in the future.

Setting Name	Notes
THINGSPEAK_URL	The URL of the thingspeak service
THINGSPEAK_API_KEY	The write API KEY for the thingspeak channel.

Telephone Numbers

The datalogger can send text messages to inform users of its status.

There are two sets of telephone numbers.
General numbers are for basic status information.
Maintenance numbers are for “alarm” type messages such as low battery warning.

Up to four numbers may be specified for each set. Messages will be sent to all numbers in the set.

Setting Name	Notes
GENERAL_PHONE_NUMBER_1 GENERAL_PHONE_NUMBER_2 GENERAL_PHONE_NUMBER_3 GENERAL_PHONE_NUMBER_4	Up to 4 general numbers.
MAINTENANCE_PHONE_NUMBER_1 MAINTENANCE_PHONE_NUMBER_2 MAINTENANCE_PHONE_NUMBER_3 MAINTENANCE_PHONE_NUMBER_4	Up to 4 maintenance numbers.

Exact phone number format will depend on country of use. Known formats:

Country	Format
UK	e.g. 074531234332 (no spaces or brackets)

International numbers must be prefixed with the country code and have leading zeros removed. For example, the UK number in the above table would be +4474531234332.

Upload Settings

Raw data is read 5 times per second.

This data is averaged, stored in RAM and uploaded. Available configuration settings are:

Setting Name	Notes
UPLOAD_AVERAGING_INTERVAL_SECS	Time to average data for upload over (in seconds)
DATA_UPLOAD_INTERVAL_SECS	Time between uploads.

For example settings of

UPLOAD_AVERAGING_INTERVAL_SECS = 30
DATA_UPLOAD_INTERVAL_SECS = 180

would result in a single average of 150 readings being taken and stored in RAM every 30 seconds. These readings would be uploaded every 180 seconds. This means that 6 averages would be uploaded every 180 seconds.

Local Storage Settings

Raw data is read 5 times per second.

This data is averaged, stored in RAM and stored to SD card. Available configuration settings are:

Setting Name	Notes
STORAGE_AVERAGING_INTERVAL_SECS	Time to average data for storage over (in seconds)
DATA_STORAGE_INTERVAL_SECS	Time between stores.

For example settings of:

STORAGE_AVERAGING_INTERVAL_SECS = 5
DATA_STORAGE_INTERVAL_SECS = 60

would result in a single average of 25 readings being taken and stored in RAM every 5 seconds. These readings would be stored to SD card every 60 seconds. This means that 12 averages would be stored every 60 seconds.

Note: data storage and data upload are **entirely independent functions**. They both use the same raw data, but the uploading and storage are not dependent on each other to function.

Battery Level Warning

The battery level is checked every hour unless this is overridden in settings. If the battery level is equal to or less than the level set, the maintenance phones numbers are sent a warning message.

Setting Name	Notes
BATTERY_WARN_INTERVAL_MINUTES	Number of minutes between checking the battery (defaults to 60)

BATTERY_WARN_LEVEL	Battery warning level in percent (not including % sign). e.g to warn at 80% or below use BATTERY_WARN_LEVEL = 80

Advanced/Debug Settings

Because fully developed datalogger applications are complex and can only be debugged using the serial output, there are settings to configure the serial output to allow for some level of control.

Setting Name	Notes
DEBUG_MODULES	A comma-separated list of modules to enable debugging output for. Currently supported modules are: LocalStorage Upload GPS Batt More modules are likely to be added in the future.
DEBUG_FIELDS	A comma separated list of the fields to include in once-per-second debug output of data. The fields MUST be zero-padded to two characters. e.g. DEBUG_FIELDS = 01, 02, 03, 05, 07, 08 will turn on debugging for fields 1, 2, 3, 5, 7 and 8.

Example Datalogger.settings.conf

```
# This is the global settings file for the datalogger.
# Any lines that start with a '#' are comments.
# A comment must be on its own line.

# Each setting is a setting name, followed by an equals sign, followed by the setting value.
# The setting name is case-sensitive, e.g. GPRS_APN is a value name, but gprs_apn is not.

# There can be whitespace between the '=' sign and the name/value.
# e.g. GPRS_APN=giffgaff.com and GPRS_APN = giffgaff.com are equally valid.
```


```
# GPRS settings
GPRS_APN=everywhere
GPRS_USERNAME=eesecure
GPRS_PASSWORD=secure

# Thingspeak settings
THINGSPEAK_URL=agile-headland-8076.herokuapp.com
THINGSPEAK_API_KEY=************************

# Data settings
STORAGE_AVERAGING_INTERVAL_SECS = 1
UPLOAD_AVERAGING_INTERVAL_SECS = 30
DATA_STORAGE_INTERVAL_SECS = 60
DATA_UPLOAD_INTERVAL_SECS = 30

# Battery level settings
BATTERY_WARN_LEVEL = 95

# SMS Numbers
MAINTENANCE_PHONE_NUMBER_1 =0123456789

# Debugging settings
DEBUG_MODULES=GPS, Batt, LocalStorage, Upload
ENABLE_DATA_DEBUG=1
```

Channels Configuration file

A datalogger application logs data for several *channels*. In order for the software to convert raw data readings (ADC reads, pulse counts etc.) to physical values (voltage, temperature etc.), each channel must be configured. This configuration is done by writing a channels configuration file.

Filename & Location

The configuration file MUST be stored on the root of the SD card under the name "Datalogger.channels.conf".

File Structure

The configuration file has one setting per line.

Each line has a channel number, a channel setting name, an equals sign, and a subsetting value. The channel number and setting name are separated by a dot, e.g.:

Channel1.Type = Voltage

Channel numbers may be in the form *ChannelX* or *ChX*, where X is a valid channel number.

Setting values are either strings or numbers. All numbers are read as floating point.

Spaces on either side of the equals sign are ignored. For example:

Channel1.mvPerBit = 0.125

and

Channel1.mvPerBit=0.125

will be treated identically.

Comments

Lines starting with a # are treated as comments. The entirety of the line will be ignored. Comments at the end of a line are not supported. e.g.:

Channel3.mvperamp = 60 # Some comment here

is **NOT** valid and would result in a failed setup of channel 3.

Channel Setup

The first configuration line for a channel **MUST** be the Type setting.

For example, if channel 5 is a temperature channel, the first setup line for channel 5 must be:

Channel5.Type = Temperature_C

Any other configuration for channel 5 will result in an error.

Other settings for a channel can be added in any order.

All settings for a channel must be present for a channel to be configured correctly.

Data Types

Valid settings for channel types are:

Type Name
Voltage
Current
Temperature_C

It is expected that additional datatypes will be added in the future.

Voltage Channel Settings

A voltage channel is configured assuming that the voltage to be read is first passed through a potential divider to reduce the voltage to a low level for an ADC input.

The ADC resolution and potential divider need to be configured.

Setting Name	Notes
mvPerBit	The number of millivolts that a 1-bit change of ADC value represents.
R1	The value of the resistance between the measured voltage and the ADC input.
R2	The value of the resistance between the ADC input and 0V.
Note: If no potential divider is present, set R1 = 0 and R2 = 1.	

e.g. a voltage to be measured on channel 6 is passed through a potential divider with 100K and 10K resistors. The ADC is 10-bit with a 5V reference. The required setting lines are:

```
Channel6.Type = Voltage
Channel6.mvPerBit = 4.88
Channel6.R1 = 100000
Channel6.R2 = 10000
```

Current Channel Settings

A current channel is configured assuming that the current to be read is converted to a voltage and read by an ADC.

The ADC resolution, conversion ratio (millivolts per amp) and conversion offset (converter output at zero current) need to be configured.

Setting Name	**Notes**
mvPerBit	The number of millivolts that a 1-bit change of ADC value represents.
offset	The convertor output in millivolts when no current is passing through it.
mvPerAmp	The change in converter output in millivolts when the current changes by 1A.
Note: Even if offset is zero, it must still be set in the configuration file.	

e.g. a current to be measured on channel 3 is passed through a converter with a 0.1V per amp conversion ratio and an offset of 2.5V at 0A. The ADC reading this value is 12-bit with a 3.3V reference. The required setting lines are:

```
Channel3.Type = Current
Channel3.mvPerBit = 0.806
Channel3.offset = 2500
Channel3.mvPerAmp = 100
```

Temperature Channel Settings

A temperature channel is configured assuming that a thermistor is the sensing element that is part of a potential divider with a fixed input voltage.

The ADC and thermistor parameters need to be configured.

Setting Name	Notes
R25	The thermistor resistance at 25C.
B	The thermistor B value
otherr	The value of the fixed resistor in the potential divider
highside	Set to 1 if the thermistor is on the “high” side of the potential divider, 0 if on “low” side.
maxADC	The ADC value that would be read if the ADC input were connected to the top of the potential divider.

e.g. a temperature to be measured on channel 11 is read with a thermistor of B value 3300K and resistance at 25C of 10K. The fixed resistor in the potential divider is also 10K. The potential divider is connected to a 3.3V source. The ADC is 12-bit and its reference voltage is 5V. The thermistor is on the “low” side of the divider. The required setting lines are:

```
Channel11.Type = Temperature_C
Channel11.R25 = 10000
Channel11.B = 3300
Channel11.otherr = 10000
Channel11.highside = 0
Channel11.maxADC = 2702
```

Example Datalogger.Channels.conf

```
Channel1.Type = VOLTAGE
Channel1.mvPerBit = 0.125
Channel1.R1 = 200000
Channel1.R2 = 10000

Channel2.Type = VOLTAGE
Channel2.mvPerBit = 0.125
Channel2.R1 = 200000
Channel2.R2 = 10000
```

Channel3.Type = VOLTAGE
Channel3.mvPerBit = 0.125
Channel3.R1 = 200000
Channel3.R2 = 10000

Channel4.Type = VOLTAGE
Channel4.mvPerBit = 0.125
Channel4.R1 = 200000
Channel4.R2 = 10000

Channel5.Type = CURRENT
Channel5.mvPerBit = 0.125
Channel5.offset = 59
Channel5.mvPerAmp = 594

Channel6.Type = CURRENT
Channel6.mvPerBit = 0.125
Channel6.offset = 59
Channel6.mvPerAmp = 592

Channel7.Type = CURRENT
Channel7.mvPerBit = 0.125
Channel7.offset = 59
Channel7.mvPerAmp = 591

Channel8.Type = CURRENT
Channel8.mvPerBit = 0.125
Channel8.offset = 60
Channel8.mvPerAmp = 596

Channel9.Type = CURRENT
Channel9.mvPerBit = 0.125
Channel9.offset = 59
Channel9.mvPerAmp = 592

Channel10.Type = CURRENT
Channel10.mvPerBit = 0.125
Channel10.offset = 59
Channel10.mvPerAmp = 592

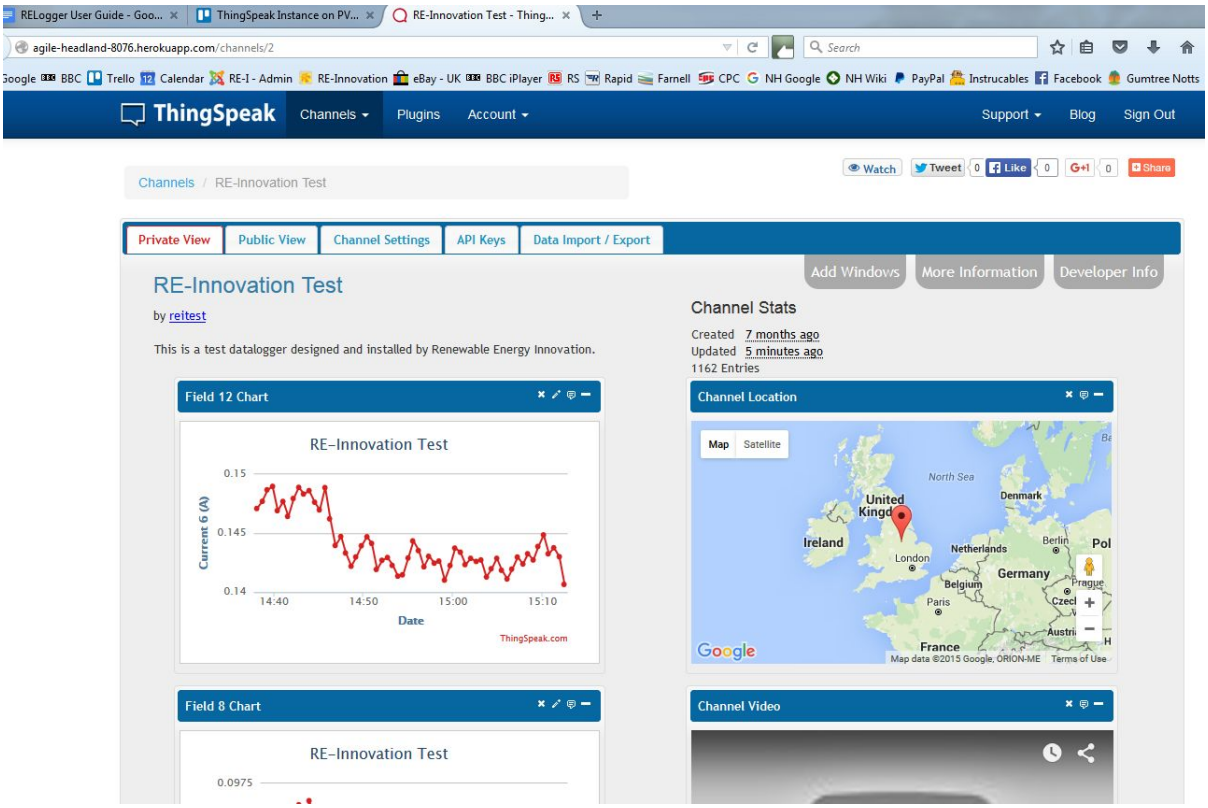
Channel11.Type = CURRENT
Channel11.mvPerBit = 0.125

```
Channel11.offset = 60
Channel11.mvPerAmp = 600

Channel12.Type = CURRENT
Channel12.mvPerBit = 0.125
Channel12.offset = 60
Channel12.mvPerAmp = 600

# Thermistor channels
# ADC goes to 1023 at 5V, thermistor is attached to 3.3V rail,
# So maximum ADC read will be 1023 * 3.3 / 5 = 675

Channel13.Type=Temperature_C
Channel13.otherr = 10000.0
Channel13.B = 4400.0
Channel13.R25 = 10000.0
Channel13.maxadc = 675
Channel13.highside = 0

Channel14.Type=Temperature_C
Channel14.otherr = 10000.0
Channel14.B = 4400.0
Channel14.R25 = 100000.0
Channel14.maxadc = 675
Channel14.highside = 0

Channel15.Type=Temperature_C
Channel15.otherr = 10000.0
Channel15.B = 4400.0
Channel15.R25 = 100000.0
Channel15.maxadc = 675
Channel15.highside = 0
```

Data Monitoring

Data is monitored via a web interface. This uses a bespoke version of ThingSpeak:
https://thingspeak.com/

This shows the data on a number of graphs.
Data can be downloaded directly from this site.



The download process for the data is as follows:

- Go to the ThingSpeak Channel.
- Sign In with your log-in details.
- Click on ‘Data Import/Export’
- Then Click on ‘Download’ under the Export heading.
- This will download all the data from that as a .csv file.
- Other file formats can be chosen.

The software can be written to handle a number of different web interfaces.
This will be implemented more fully in future versions.

RE-Innovation Test Unit

A test unit has been installed for monitoring its performance. The data is available to view here:
http://agile-headland-8076.herokuapp.com/channels/2

CREST PROJECT

Five Dataloggers have been supplied to CREST. The data for these units is available here:

CREST Test unit
http://agile-headland-8076.herokuapp.com/channels/7

KENYA 1
http://agile-headland-8076.herokuapp.com/channels/8

KENYA 2
http://agile-headland-8076.herokuapp.com/channels/9

BANGLADESH 1
http://agile-headland-8076.herokuapp.com/channels/10

BANGLADESH 2
http://agile-headland-8076.herokuapp.com/channels/11

Open Source Files

This project is open source and the file are hosted on GITHUB here:
https://github.com/re-innovation/

License

Company Information

Renewable Energy Innovation.
info@re-innovation.co.uk
www.re-innovation.co.uk
Hopkinson Gallery
21 Station Street
Nottingham
NG7 6PD
UK

Please email info@re-innovation.co.uk with any questions or comments.

More technical information can be found via www.re-innovation.co.uk.

Returns

We want you to be happy with your product. If you have any problem at all, then please get in touch and we will try to resolve the issue or issue a refund. Please email us before returning any item.

End of Life

If you no longer require your product, please return it to us for safe and environmentally friendly disposal.
We would like to ensure that our products are used for as long as possible and will attempt to reuse or recycle as many components as possible.
We are attempting to have a low environmental impact and try to **reduce** any waste streams, **reuse** as much packaging material as possible and **recycle** as much as we can.