WEATHER INFORMATION MANAGEMENT FOR EAST AFRICA USING SUITABLE INFORMATION COMMUNICATION TECHNOLOGY

AUTOMATIC WEATHER STATION

WIMEA-ICT AWS

User Manual

Version 3.0

February, 2021

**Table of Contents**

[LIST OF FIGURES ii](#_Toc65010790)

[LIST OF TABLES iii](#_Toc65010791)

[1 INTRODUCTION 1](#_Toc65010792)

[1.1 Purpose 1](#_Toc65010793)

[1.2 Overview 1](#_Toc65010794)

[2 GETTING STARTED 3](#_Toc65010795)

[2.1 AWS Requirement 3](#_Toc65010796)

[2.1.1 Software 3](#_Toc65010797)

[2.1.2 Hardware 3](#_Toc65010798)

[2.1.3 Sensors 4](#_Toc65010799)

[2.2 Installing and Configuring the Software on Windows Computers 5](#_Toc65010800)

[2.2.1 Configuring Putty 5](#_Toc65010801)

[2.2.2 Configuring AVRDUDESS 6](#_Toc65010802)

[2.2.3 Configuring Particle CLI 8](#_Toc65010803)

[2.3 Installing and Configuring the Software on Linux Computers 9](#_Toc65010804)

[2.3.1 Install & Configure AVR tool chains 9](#_Toc65010805)

[2.3.2 Install & Configure Putty 10](#_Toc65010806)

[2.3.3 Install and Configure Particle CLI 11](#_Toc65010807)

[3 USING WIMEA-ICT AWS 12](#_Toc65010808)

[3.1 Configuring the Sensor nodes: Transmitting node 12](#_Toc65010809)

[3.2 Configuring the Sensor nodes: The Sink node 13](#_Toc65010810)

[3.3 Configuring and setting up the Low Power Gateway 14](#_Toc65010811)

[4 TROUBLESHOOTING & SUPPORT 16](#_Toc65010812)

[4.1 Error (Physical or Error Messages) 16](#_Toc65010813)

[4.2 Special Considerations 16](#_Toc65010814)

[5 APPENDICES 17](#_Toc65010815)

[5.1 Appendix A: Acronyms 17](#_Toc65010816)

[5.2 Appendix B: Glossary 17](#_Toc65010817)

# LIST OF FIGURES

[Figure 2.1: Temperature and Wind sensors 4](#_Toc65010889)

[Figure 2.2: Soil Sensors and rain gauge 4](#_Toc65010890)

[Figure 2.3: Solar Panels used at the WIMEA-ICT AWS 5](#_Toc65010891)

[Figure 2.4: Configuring Putty client 6](#_Toc65010892)

[Figure 2.5: Configuring AVRDUDESS for flashing the firmware to the mote 7](#_Toc65010893)

[Figure 2.6: Setting up particle account and particle electron device 8](#_Toc65010894)

[Figure 2.7: Result of running avrdude command on the terminal 9](#_Toc65010895)

[Figure 2.8: Result of running command dmesg | grep tty for checking the USB serial port 10](#_Toc65010896)

[Figure 2.9: Installing Putty terminal client on Linux 11](#_Toc65010897)

[Figure 3.1: The result of running the help command, h, on putty terminal 12](#_Toc65010898)

[Figure 3.2: Running the different essential commands on putty 13](#_Toc65010899)

[Figure 3.3: Configuring the sink node 14](#_Toc65010900)

# LIST OF TABLES

[Table 1: List of different acronyms in the manual 17](#_Toc65010901)

[Table 2: List of definition of different terms 17](#_Toc65010902)

# INTRODUCTION

The WIMEA-ICT AWS was developed under one of the WIMEA-ICT project Research Component, RC3. This Research Component aimed at assisting the Meteorological services in increasing the number of Automatic Weather Station (AWS) in Uganda, South Sudan and Tanzania by designing a robust AWS.

The WIMEA-ICT AWS is composed of different meteorological sensors structured under three sets of nodes, one gateway for establishing the server. The nodes are named according to the position and/or the heights of the sensors above the ground. The nodes that make up the WIMEA-ICT AWS are; the 2m node that has sensors for reading the relative humidity and the temperature values, the 10m have sensors that read the wind speed and the wind direction values and the solar insolation. The ground node contains sensors that read the soil temperature values, soil moisture, pressure and the rainfall and/or precipitation values. The gateway is a combination of the sink node that receives the weather data reports broadcasted by the three nodes, the particle electron with the GSM module for the internet connection to the server. At the gateway, the RTC module attaches the timestamp to the report received by the sink and the SD card temporarily stores the sensor data report before its uploaded to the server.

## Purpose

The WIMEA-ICT AWS user manual is intended to offered basic and simplest guide to the users of the AWS. Thus, this document shall give directives on how to use and/or configure the WIMEA-ICT AWS. This user manual illustrates the actual components of the AWS including the nodes and the gateway, the hardware and software in use, the sensors and how to assemble.

## Overview

The user manual is structured in about four sections including but not limited to the introduction that presents the purpose of the document, the overview of the document and the acronyms. The next section, getting started, presents the WIMEA-ICT AWS requirements including the software, the hardware, the sensors and other requirements for using and operating the AWS, the installation and configuration processes involving the software. The next section, using the WIMEA-ICT AWS talks about assembling and configuring the nodes and the gateway. The next section, Troubleshooting presents the possible challenges and errors that the AWS might run into as well as a work around on how to solve the issues. Last section is the Appendix that presents the acronyms, glossary and references to the supporting documents.

# GETTING STARTED

## AWS Requirement

For effective maintenance and use of the WIMEA-ICT AWS, the following requirements are essential and should be available.

### Software

The WIMEA-ICT application hex file with a *.hex* file extension, this is firmware that runs the AWS sensor nodes. For the gateway, the bin file with a .bin file extension which is the firmware that runs the gateway’s particle electron.

Putty client terminal program for monitoring the performance of the different nodes and checking for errors as well as performing the configuration of the sensor nodes.

For windows users, AVRDUDESS will be essential for flashing (uploading) the firmware hex file to the different motes. Linux users can install the AVR tool chains that will perform the exact function of flashing the firmware.

### Hardware

Model S2 v2.4 RSS2 motes which is the sensor board for attaching the different sensors used at the WIMEA-ICT AWS. Depending on the type of sensor attached on this PCB board and the firmware uploaded on to it, it makes up one of the nodes.

Particle electron contains a GSM module and its used at the WIMEA-ICT AWS low power gateway for internet connection as well as uploading the sensor data to the central server.

DS3231 RTC module provides the real day time so as to keep track of the different sensor data reports received.

SD card module together with the memory card attached to it offers a provision for temporary storage of the reports before its uploaded to the server.

Sim card attached on the particle electron. This with the help of the GSM module on the electron provides cellular network connection. Thus, the whole gateway can efficiently be able to connect to the internet.

### Sensors

The different sensors used at the WIMEA-ICT AWS includes the vegetronix soil moisture sensor, soil temperature sensor, the pressure sensor, the rain gauge, the sht25 temperature and humidity sensor, the inspeed vortex anemometer and wind vane as well as the solar insolation sensor.

Figure 2.1: Temperature and Wind sensors



Figure 2.2: Soil Sensors and rain gauge

Other essential materials required for the WIMEA-ICT AWS operation includes the solar panels in the ratings of 2watts, 1watt and 0.5watt, the cylindrical lithium ion 3.7V batteries, the FTDI USB-TTL cable and a computer having all the necessary software installed.



Figure 2.3: Solar Panels used at the WIMEA-ICT AWS

## Installing and Configuring the Software on Windows Computers

Download and install the necessary software.

1. Putty: <https://the.earth.li/~sgtatham/putty/latest/w64/putty-64bit-0.74-installer.msi>
2. AVRDUDESS: <https://blog.zakkemble.net/download/AVRDUDESS-2.13-setup.exe>
3. Particle CLI: <https://binaries.particle.io/cli/installer/windows/ParticleCLISetup.exe>

Downloading and running the Particle CLI installer will install the necessary device drivers for use and other utilities.

### Configuring Putty

1. Open Putty. (Make sure FTDI USB-TTL cable in attached to one of the USB ports on your computer)

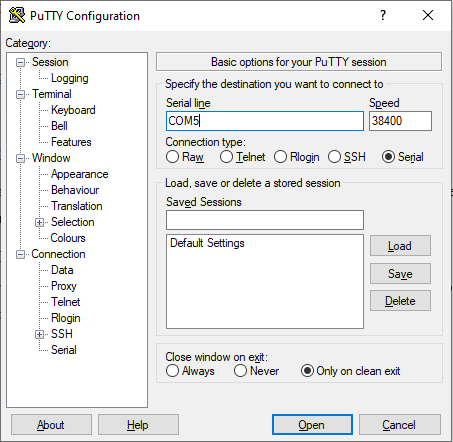


Figure 2.4: Configuring Putty client

1. Under Basic options for your PuTTY session, on Serial line, enter the COM (communication) port number assigned to the FTDI cable. Check this by opening Device Manager – under Ports (COM & LPT).
2. On the Speed box, enter 38400 Baud Rate as the speed.
3. Under Connection Type, select Serial
4. Click Open

### Configuring AVRDUDESS

1. Open AVRDUDESS (Ensure that the FDTI cable is connected to your computer)

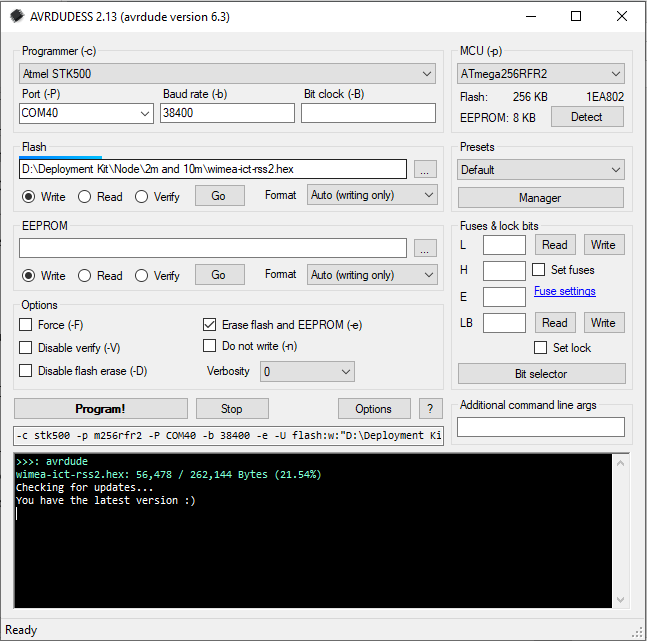


Figure 2.5: Configuring AVRDUDESS for flashing the firmware to the mote

1. Under Programmer, on the drop-down list, locate and select *Atmel STK500*
2. Under Port, select the COM port number assigned to the FTDI cable from the Device Manager
3. For Baud rate, enter 3840 and leave the box for Bit clock empty
4. Under Flash, browser your directory to where the hex file is located. Select the hex file and click open. Make sure the option for Write is selected
5. Under EEPROM, leave the box empty. But make sure the option for Write is selected under that box.
6. Under Options, check the box for Erase flash and EEPROM.
7. Press the Restart button on the mote connected to the FTDI cable, and while the yellow light is still blinking, click on the **Program!** button on the AVRDUDESS. The yellow light on the mote will be blinking while the firmware is being flashed on the mote.

### Configuring Particle CLI

1. On the command line, run the command *particle setup* which will show you a screen like in figure xx below. Either create a new account or login if you already have an account. (Ensure that the particle electron is connected to one of the USB ports on your computer). This will setup the particle device connected, in this case the electron device.

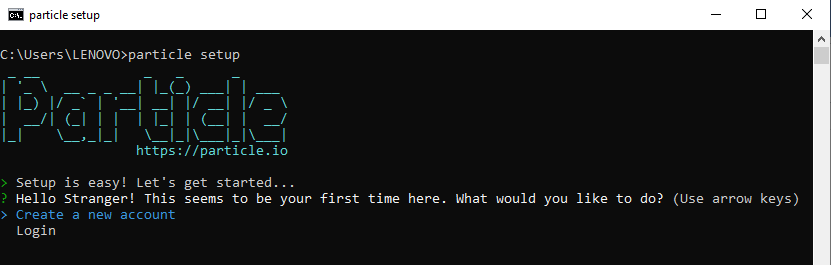


Figure 2.6: Setting up particle account and particle electron device

1. Put the electron device in the DFU (Device Firmware Update) mode. To put the electron in the DFU mode, press and hold down both the reset and the node button. First release the reset button, then, wait until the RGB led is blinking yellow, then, release the mode button. Now your electron is in DFU mode waiting for a new firmware to be flashed to it.
2. To flash the firmware (.bin file) to the electron, run this command while the electron is still in DFU mode

*particle flash --usb (filename).bin*.

For instance, if the bin file is called *electron.bin*, the command would be

*particle flash --usb electron.bin*

1. On success, the electron should restart.

## Installing and Configuring the Software on Linux Computers

### Install & Configure AVR tool chains

1. To install the AVR tool chains, open your terminal and run the following command. The packages such as gcc-avr, binutils-avr, gdb-avr, avr-libc and avrdude will be installed on running the command below. For confirmation purposes on whether the installation was successful, type *avrdude* and press enter.

*sudo apt install gcc-avr binutils-avr gdb-avr avr-libc avrdude*

1. On successful installation, running the command *avrdude* on the terminal should display something similar to that in figure xx below.

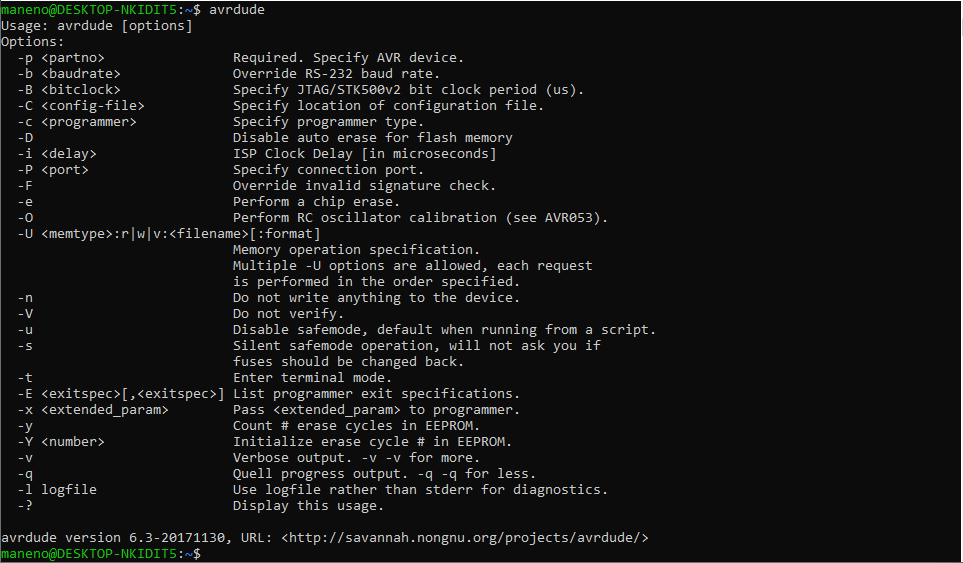


Figure 2.7: Result of running avrdude command on the terminal

1. To flash a firmware onto the mote, run the following commands. Make sure the FTDI cable is attached to the USB port with a mote connected to it. Run the command *dmesg | grep tty* and check for /dev/ttyUSB0. The result of that command should display something similar to that in figure xx below. The numbers vary every time you plug in the FTDI cable, thus check for one with *connection.* Update the /dev/ttyUSB0 on the command according to that displayed after running the *dmesg* command

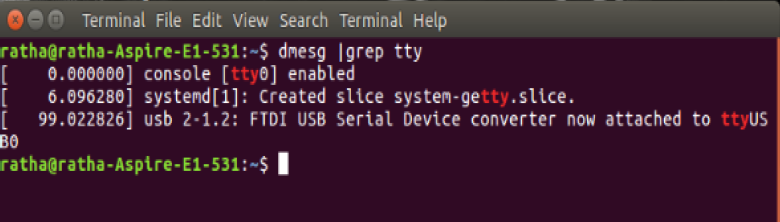


Figure 2.8: Result of running command dmesg | grep tty for checking the USB serial port

1. Flash (download) the firmware to the mote by running this command below

*avrdude -c stk500v2 -P /dev/ttyUSB(number) -p m256rfr2 -b 38400 -e –U flash:w:(filename).hex*

1. For instance, if the filename is wimea-ict-rss2.hex and the USB port number assigned to the FTDI is 0, then, the command would be

*avrdude -c stk500v2 -P /dev/ttyUSB0 -p m256rfr2 -b 38400 -e –U flash:w:wimea-ict-rss2.hex*

1. Before hitting enter on the keyboard to run the command, press the restart button on the mote and while the yellow light on the mote is still blinking, hit the enter key so as to run the command (flash the firmware).

### Install & Configure Putty

To install putty, open your terminal and run this command, *sudo apt install putty*. After a successful installation, run the command *sudo putty* to open the putty. Configure putty COM port, baud rate speed and connection type as in section above and click on open to start the putty serial terminal.

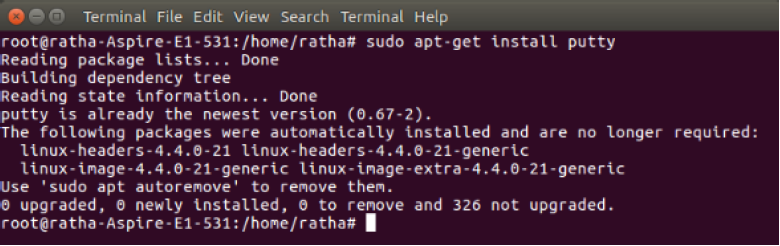


Figure 2.9: Installing Putty terminal client on Linux

### Install and Configure Particle CLI

1. To install Particle CLI on Linux or Mac, open your terminal and run the following command.

*bash <( curl -sL https://particle.io/install-cli )*

1. After the CLI has installed successfully, running the command Particle on the terminal should show you something similar to what is on figure xx above.
2. Follow the instructions in section 2.2.3 above to setup your particle device as well as the account and also flashing the firmware.

# USING WIMEA-ICT AWS

## Configuring the Sensor nodes: Transmitting node

The transmitting node makes up the three nodes of the WIMEA-ICT AWS, that is, the 2m node, the 10m node and the ground node. To configure these nodes, flash the *wimea-ict-rss2.hex* firmware to the motes by following the instruction steps in section above.

Connect the FTDI cable (with the mote attached to it) to the computer. Depending on the platform you are using (Windows or Linux), open Putty as described in section 2.2.1 above. This will open the putty terminal for monitoring the activities and performance of that particular node.

To check the list of available commands, type *h* on the putty terminal and press the Enter key on the keyboard. The result should be similar to figure 3.1 below.

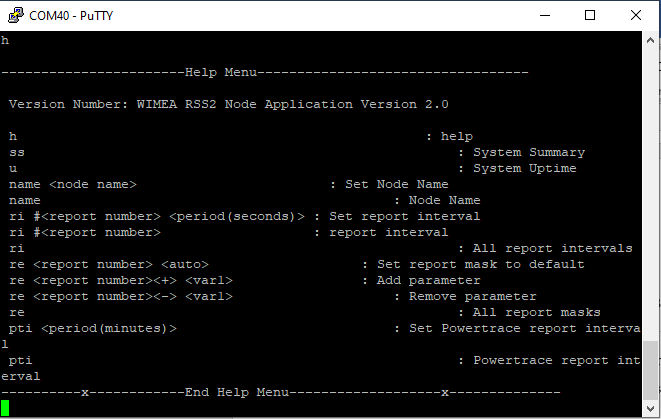


Figure 3.1: The result of running the help command, h, on putty terminal

Change the name of node by running the command, *name <new node name>* on putty terminal. To check the current node name, run the command, *name* on putty terminal

Change the report interval by running the command, *ri #0 10* This will change the reporting interval of that node to 10 seconds. Run the command, *ri* to check the current reporting interval.

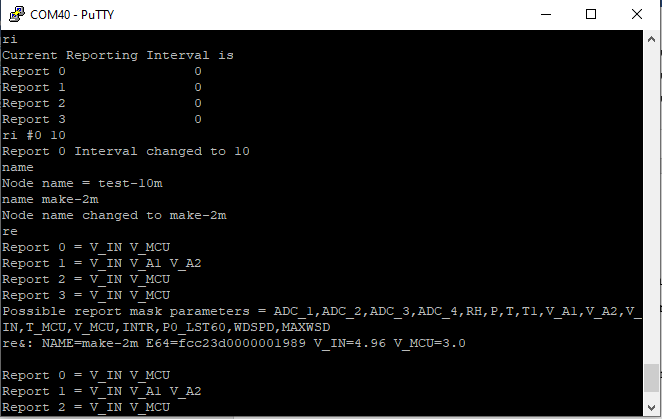


Figure 3.2: Running the different essential commands on putty

To add and/or remove a parameter from the report mask, run the command *re 0+ RH T* or re 0- RH T which will add and remove the parameters RH (Relative Humidity & Temperature) respectively.

You can run any other command by following the syntax of that specific command. For help, run *h* on the putty terminal to get the list of all commands and how to run them. Figure xx shows the result of running such a command.

## Configuring the Sensor nodes: The Sink node

Connect the FTDI cable with the sink node mote to the computer. If the sink node firmware is not yet flashed to the mote, flash the *gen3sink-15s.hex* firmware to the mote by following the instruction set in section above.

Open putty terminal as described in section above. The sink node will be receiving the reports from the different transmitting nodes and displaying it on putty terminal.

Change the sink node name by running the command, *sinkname <new sink node name>* on putty terminal. The naming convention of the sink node name is *stationName-sink* and the station is shortened to only three letters. Iganga station should have a name like igg, Kamuli station should have kml, Bukwo station should have bkw and others.

Thus, to change the sink node name of Entebbe station, the command should be *sinkname ebb-sink* This will change the sink node name for Entebbe station to *ebb-sink*. To check the current sink node name, run the command, *sinkname* on putty terminal.

For help menu, run the command, ??, on the putty terminal

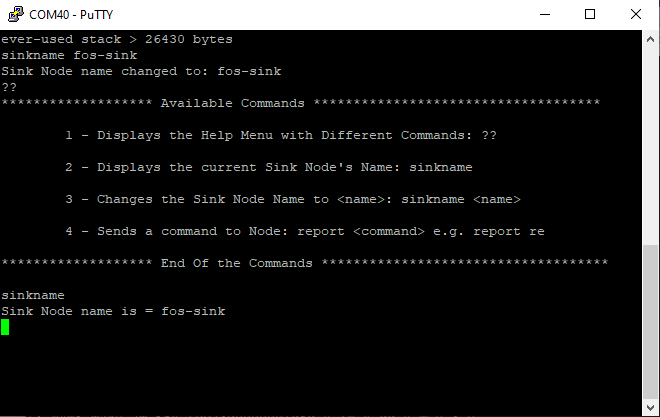


Figure 3.3: Configuring the sink node

## Configuring and setting up the Low Power Gateway

Flash the electron.bin firmware to the Particle Electron by following the instructions in section 2.2.3 above.

Configure the *config* text file by changing the APN type to internet, SERVER to server name you are using (for instance wimea.mak.ac.ug), UPLOADCOUNT to 250, DEVICE – this should have a name like *stationName-elect*. For instance, for Jinja station, it should be DEVICE:jja-elec

Copy the config.txt file and sensor.dat file to the SD card

On the WIMEA-ICT PCB board, fix the electron with its antenna and the sim card, the DS3231 RTC module, the SD card module with the memory card, and the sink node to their respective position. Connect the CON\_ADC1 power pin to pin D1 on the electron. Join the electron’s RX (receive) pin to the sink node’s TX (transmit) pin

Power up the board by connection the battery to the battery header pins on the board. Toggle the tactile switch on the board to power up the whole gateway component.

Monitor the sink node and the electron to ensure that the sink node is receiving the report from the transmitting nodes by blinking the red led and also waking the electron (the electron RGB led should blink white). If this does not happen, check the connections of all the components.

Some time later, possibly in the interval of about 90 minutes, the electron will try to establish a connection to the server in order to upload the reports that were temporary being stored in sensor.dat file in the memory card. The electron’s RGB led will be blinking green for some seconds during this process and then blue towards the end of the upload.

Connect the 2watts solar panel also to ensure that the battery is recharged during the day.

# TROUBLESHOOTING & SUPPORT

## Error (Physical or Error Messages)

Sometimes the node’s report will only contain the node’s name and Energy Consumption. This implies that the battery is below the recommended voltage and thus the sensor readings can not be taken and as a result no parameter shall be transmitted but only Energy Consumption.

When the node is constantly restarting by blinking the yellow led, the best explanation to this would be a corrupted firmware. That is, the firmware is no longer running as it’s supposed to. If it’s happening at the sink node, check whether the DS3231 RTC module is working, the module might not be communicating with the sink node.

Electron not uploading or node powering up. This can be caused by the low power supply at the gateway or the gateway being fully dead. Thus, try to replace the battery with a fully charged battery and check whether the whole gateway starts working, else change the electron with the spare that you should be having. Also, ensure that the solar panel is working perfectly and supplying at least 6.6V of power to the board.

## Special Considerations

Before anything, the best trouble shooting step is to connect the node to the computer using FTDI and monitoring its performance on putty terminal. Changing the firmware or re-flashing the same firmware can be another important work around to solve the errors that may come up. If everything fails, replace the whole mote with the one that is working.

# APPENDICES

## Appendix A: Acronyms

|  |  |
| --- | --- |
| **Acronym** | **Literal Translation** |
| AWS | Automatic Weather Station |
| RTC | Real Time Clock |
| WIMEA-ICT | Weather Information Management for East Africa using Information Communication Technology |
|  |  |

Table 1: List of different acronyms in the manual

## Appendix B: Glossary

|  |  |  |
| --- | --- | --- |
| **Term** | **Acronym** | **Definition** |
| Hex file | .hex | The compiled application WIMEA-ICT file in hexadecimal format |
| Bin file | .bin | The compiled electron application in binary format |
|  |  |  |
|  |  |  |

Table 2: List of definition of different terms