



UNIVERSITEIT•STELLENBOSCH•UNIVERSITY  
jou kennisvennoot • your knowledge partner

# **Remote data logging - A comprehensive solution**

Pieter Andries van Wyk  
22533788

Report submitted in partial fulfilment of the requirements of the module Project (E) 448 for the degree Baccalaureus in Engineering in the Department of Electrical and Electronic Engineering at Stellenbosch University.

Supervisor: Dr W. Smit

October 2022

# **Acknowledgements**

I dedicate this report to my mother and father. Both of whom have given me so much for which I will always be grateful. I want to thank JP, Muller and Gunter for all the laughs and your companionship throughout the duration of my thesis. To my granddad Andries, thank you for all the coffee and all the spelling errors you found throughout this report. And finally, thank you to my supervisor Dr Smit for your guidance and help.



UNIVERSITEIT•STELLENBOSCH•UNIVERSITY  
jou kennisvennoot • your knowledge partner

## Plagiaatverklaring / Plagiarism Declaration

1. Plagiaat is die oorneem en gebruik van die idees, materiaal en ander intellektuele eiendom van ander persone asof dit jou eie werk is.

*Plagiarism is the use of ideas, material and other intellectual property of another's work and to present it as my own.*

2. Ek erken dat die pleeg van plagiaat 'n strafbare oortreding is aangesien dit 'n vorm van diefstal is.

*I agree that plagiarism is a punishable offence because it constitutes theft.*

3. Ek verstaan ook dat direkte vertalings plagiaat is.

*I also understand that direct translations are plagiarism.*

4. Dienooreenkomsdig is alle aanhalings en bydraes vanuit enige bron (ingesluit die internet) volledig verwys (erken). Ek erken dat die woordelikse aanhaal van teks sonder aanhalingstekens (selfs al word die bron volledig erken) plagiaat is.

*Accordingly all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism*

5. Ek verklaar dat die werk in hierdie skryfstuk vervat, behalwe waar anders aangedui, my eie oorspronklike werk is en dat ek dit nie vantevore in die geheel of gedeeltelik ingehandig het vir bepunting in hierdie module/werkstuk of 'n ander module/werkstuk nie.

*I declare that the work contained in this assignment, except where otherwise stated, is my original work and that I have not previously (in its entirety or in part) submitted it for grading in this module/assignment or another module/assignment.*

22533788	
Studentenommer / Student number	Handtekening / Signature
PA van Wyk	30-10-2022
Voorletters en van / Initials and surname	Datum / Date

# **Abstract**

## **English**

This report covers the design and implementation of a remote data logging system. This system allows for two-way communication between a base station and a remote station. Data logged through the system is saved to an online database and visualised through a custom web application. Loadshedding compatibility is implemented in this system, allowing for data to be logged during periods of internet unavailability. This report contains the system-, electronic-, network- and software design required to create this solution. The final system is fully operational and can be used within farming or weather tracking operations. The remote data logging solution proposed with this report was built with modularity in mind and can be extended to various fields of operation.

## **Afrikaans**

Hierdie verslag behels die ontwerp en implementering van 'n afstand gebaseerde data aantekenings sisteem. Hierdie sisteem maak voorsiening vir tweerigting kommunikasie tussen 'n tuis- en afstand geleë stasie. Data wat aangeteken word deur die sisteem word opgeneem in 'n webgebaseerde databasis en daarna gevisualiseer deur 'n self-gemaakte webtoepassing. Daar word vir beurtkrag voorsiening gemaak deur hierdie sisteem. Data word aangeteken selfs gedurende tydperke van internet onderbrekings en krag afwesigheid. Hierdie verslag bevat die benodigde sisteem-, elektroniese-, netwerk- en sagteware ontwerp om hierdie voorgestelde oplossing te implementeer. Die finale sisteem is ten volle operasioneel en kan benut word deur landbou en weervoorspelling ondernemings. Die sisteem wat voorgestel word met hierdie verslag was ontwikkel met modulariteit in gedagte en kan uitgebrei word vir verskeie werksveldे.

# Contents

<b>Declaration</b>	ii
<b>Abstract</b>	iii
<b>List of Figures</b>	vii
<b>List of Tables</b>	ix
<b>Nomenclature</b>	x
<b>1. Introduction</b>	1
1.1. Background . . . . .	1
1.2. Problem statement . . . . .	1
1.3. Objective . . . . .	1
1.4. Summary of work . . . . .	2
1.5. Scope . . . . .	2
<b>2. Literature Review</b>	3
2.1. Related work . . . . .	3
2.1.1. Objectives . . . . .	3
2.1.2. Methods . . . . .	3
2.1.3. Results . . . . .	3
2.1.4. Remaining challenges . . . . .	3
2.2. Physical phenomena and measurement . . . . .	4
2.2.1. Standard Test Conditions . . . . .	4
2.2.2. UV . . . . .	4
2.2.3. Temperature . . . . .	4
2.2.4. Humidity . . . . .	5
2.2.5. Antenna gain . . . . .	5
2.3. Hardware . . . . .	6
2.3.1. Solar photo voltaic cells and solar modules . . . . .	6
2.3.2. Solar charge controller . . . . .	6
2.3.3. Power over Ethernet . . . . .	7
2.4. Network . . . . .	8
2.4.1. Communication protocols . . . . .	8

2.4.2. Routers and bridges . . . . .	8
2.4.3. Bridges . . . . .	9
2.5. Databases and web-design . . . . .	9
2.5.1. Database models . . . . .	9
2.5.2. Web frameworks . . . . .	10
<b>3. System Design</b>	<b>11</b>
3.1. System diagram . . . . .	11
3.2. Remote station . . . . .	11
3.3. Base station . . . . .	12
3.4. Online components . . . . .	12
3.4.1. Database . . . . .	12
3.4.2. Online dashboard . . . . .	13
3.5. Metrics . . . . .	13
3.5.1. Remote station . . . . .	13
3.5.2. Base station . . . . .	14
<b>4. Detailed design</b>	<b>15</b>
4.1. Remote station hardware design . . . . .	15
4.1.1. Power . . . . .	15
4.1.2. Undervoltage protection . . . . .	17
4.1.3. DC-DC converters . . . . .	22
4.1.4. Fuses . . . . .	23
4.1.5. Power over Ethernet injector . . . . .	24
4.1.6. Sensors . . . . .	24
4.1.7. Raspberry Pi 4b - remote station . . . . .	27
4.2. Network configuration . . . . .	28
4.2.1. Utilities . . . . .	29
4.3. Software design . . . . .	29
4.3.1. Database . . . . .	29
4.3.2. Online dashboard . . . . .	30
4.3.3. Server - base station . . . . .	31
4.3.4. Client - remote station . . . . .	32
4.3.5. Software dependencies and libraries . . . . .	32
4.4. Implementation . . . . .	33
<b>5. Results</b>	<b>34</b>
5.1. Hardware measurements . . . . .	34
5.1.1. PV-module . . . . .	34
5.1.2. Undervoltage protection . . . . .	34

5.1.3. Voltage rails . . . . .	36
5.1.4. Power measurements . . . . .	37
5.2. Network communication . . . . .	38
5.2.1. Device detection . . . . .	38
5.2.2. Remote access and transfer . . . . .	38
5.3. Remote station range . . . . .	39
5.4. Online capacity . . . . .	39
5.4.1. Database . . . . .	39
5.4.2. Online dashboard . . . . .	39
5.5. Offline capacity . . . . .	39
<b>6. Summary and Conclusion</b>	<b>40</b>
6.1. Summary . . . . .	40
6.2. Application . . . . .	40
6.3. Future work . . . . .	40
<b>Bibliography</b>	<b>42</b>
<b>A. Project Planning Schedule</b>	<b>45</b>
<b>B. Outcome Compliance</b>	<b>46</b>
<b>C. Network configuration - AirOS</b>	<b>48</b>
<b>D. Components</b>	<b>50</b>
<b>E. Hardware</b>	<b>52</b>
<b>F. Database</b>	<b>53</b>
<b>G. Dashboard</b>	<b>54</b>
<b>H. Future work</b>	<b>57</b>

# List of Figures

2.1.	High-level hardware schematic of the LTR390 UV sensor. . . . .	4
2.2.	Antenna gain . . . . .	5
2.3.	Graph indicating current-voltage curves of a SLP005-12 PV module at different temperatures [1]. The maximum power points are indicated in red. . . . .	6
2.4.	Graph indicating the charging characteristics of a LiFePo4 battery [2] . . . . .	7
2.5.	HTTP request network transfer . . . . .	8
2.6.	Graph database model. . . . .	10
2.7.	Relational database schema. . . . .	10
3.1.	System diagram . . . . .	11
4.1.	Hardware diagram . . . . .	15
4.2.	Battery and diode . . . . .	18
4.3.	Batter voltage clipper and scaler schematic . . . . .	18
4.4.	Schmitt trigger schematic . . . . .	20
4.5.	High-side switch . . . . .	21
4.6.	Undervoltage protection schematic . . . . .	22
4.7.	Buck converter and diode . . . . .	23
4.8.	Boost converter . . . . .	23
4.9.	POE injector schematic . . . . .	24
4.10.	DHT22 temperature and humidity sensor . . . . .	25
4.11.	Anemometer flowchart . . . . .	25
4.12.	Anemometer flowchart . . . . .	26
4.13.	Raspberry Pi 4b connectivity diagram. . . . .	27
4.14.	Network hardware layout . . . . .	28
4.15.	Server flowchart . . . . .	31
4.16.	Client flowchart . . . . .	32
4.17.	System photograph . . . . .	33
5.1.	Batter voltage clipper and scaler output graph . . . . .	35
5.2.	Schmitt trigger output graph . . . . .	35
5.3.	Schmitt trigger output graph, battery voltage included . . . . .	36
5.4.	High-side switch output graph . . . . .	36
A.1.	Estimated timeline, Gantt chart . . . . .	45

A.2. Estimated timeline . . . . .	45
C.1. Sensor dashboard - viewed on cellphone . . . . .	48
C.2. Sensor dashboard - viewed on cellphone . . . . .	49
E.1. System photograph . . . . .	52
F.1. Online database visualisation . . . . .	53
G.1. Website mock-up . . . . .	54
G.2. Sensor dashboard - viewed on cellphone . . . . .	55
G.3. Sensor dashboard - viewed on desktop . . . . .	56
H.1. Future remote station . . . . .	57

# List of Tables

4.1.	Comparison of batteries for stand-alone PV systems' design. [3] . . . . .	16
4.2.	Power consumption of various hardware components in the remote station.	16
4.3.	Power saving on Raspberry Pi by disabling features. [4] . . . . .	17
4.4.	SLP005-12R PV-module rated values and properties [1] . . . . .	17
4.5.	Voltage points to consider with undervoltage protection design . . . . .	18
4.6.	Resistor value selection . . . . .	19
4.7.	Resistor value selection . . . . .	21
4.8.	Boost converter capacitor and inductor values . . . . .	23
4.9.	Fuse selection and placement . . . . .	23
4.10.	LTR-390 pinout . . . . .	26
4.11.	Raspberry Pi 4b connections . . . . .	27
4.12.	AirOS configuration, AirGrid M5 - base station . . . . .	28
4.13.	AirOS configuration, Bullet M5 - remote station . . . . .	29
4.14.	Network dependencies and description . . . . .	29
4.15.	Online database table fields . . . . .	30
4.16.	Software dependencies and description . . . . .	32
5.1.	SLP005-12R PV-module measured values . . . . .	34
5.2.	Buck converter voltage measurements . . . . .	37
5.3.	Boost converter voltage measurements . . . . .	37
5.4.	Remote station power measurements . . . . .	37
5.5.	Online database uploading times. Internet connection - download: 56.99Mbps, upload: 29.71Mbps. . . . .	39
B.1.	Outcome compliance table . . . . .	46
B.2.	Outcome compliance table . . . . .	47
D.1.	Table indicating components used. . . . .	50
D.2.	Table indicating resistors used. . . . .	51

# Nomenclature

## Variables and functions

$V_{OC}$	Open circuit voltage
$I_{SC}$	Short circuit current
$V_{min\_terminal}$	Minimum battery voltage pre diode
$V_{max\_terminal}$	Maximum battery voltage pre diode
$V_F$	Diode forward voltage
$V_{open\_switch}$	Hysteresis switch open point
$V_{close\_switch}$	Hysteresis switch close point
$V_{in+}$	Operational amplifier non-inverting input
$V_{in-}$	Operational amplifier inverting input
$V_{ref\_rail\_1}$	Reference voltage, source 5V buck converter

**Acronyms and abbreviations**

AP	Access Point
ALS	Ambient light sensor
CC-CV	Constant current, constant voltage charging
CSV	Comma-separated values
DHCP	Dynamic Host Configuration Protocol
DOD	Depth of discharge
FTP	File transfer protocol
GND	Ground
GPIO	General purpose input/ output
HTTP	Hypertext transfer protocol
HTML	Hypertext markup language
I2C	Inter-integrated circuit
IC	Integrated circuit
IoT	Internet of things
ISP	Internet service provider
LiFePo4	Lithium iron phosphate; specifically batteries
M5	Ubiquiti products; indication of operation over 5Ghz electromagnetic spectrum
MAC	Media Access Control
MQTT	Message Queue Telemetry Transport
NAT	Network address translation
NC	No connection pin
PoE	Power-over-ethernet
PNG	Portable network graphics
PV	Photovoltaic
PWM	Pulse width modulated
RBerry	Raspberry Pi; specifically 4b model
SCC	Solar charge controller
SCL, SDA	Serial clock, Serial data - I2c communication
SSID	Service set identifier
STC	Standard testing conditions
SQL	Structured query language
TCP	Transmission control protocol
URL	Uniform resource locator
UVS	Ultra-violet light sensor

# **Chapter 1**

## **Introduction**

### **1.1. Background**

Effective use of analytics and data gathering has become a crucial field across all sectors. This increased control of operation and nuanced data collection has poured over to smaller and privately owned enterprises. Specifically within agricultural and wildlife sectors, new remote data collection solutions are required to provide operators with the information necessary to make decisions and build data driven models for economic utilization to compete within their competitive markets. South Africa is facing nation wide power outages in the form of loadshedding, this reduces the data collection capacity of local enterprises if they are grid-power based. New off-grid solutions are required to combat the effects of these power outages.

### **1.2. Problem statement**

Data-driven and IoT solutions are focused on helping enterprises with remote operations close the supply demand gap [5]. Such solutions can help ensure profitability and moderate extraneous carbon emissions associated with traditional motorised or combustion-based collection methods. Operators want immediate access to their live-updating data from anywhere with the capacity to control and update the collection system remotely. Bulk, accessible data storage independent from proprietary monitoring solutions is imperative to ensure the integrity of the operation [6] and to safeguard information from competitors. An affordable, modular solution is required with scalability in mind. Operators require a system that can be integrated and expanded with pre-existing sensors and operational hardware. Such a system would have to be completely independent of existing communication infrastructure and grid electricity.

### **1.3. Objective**

To propose a solution to the remote data collection problem whilst incorporating data storage, visualisation and accessibility. The solution must be loadshedding-robust and

should have the modular capacity to extend to further operations if needed. The system should operate independently from the power grid and have sufficient range to warrant its cost. Furthermore, the system must be controllable from an online interface with the ability to update or upgrade the data-collection software remotely. The solution proposed with this report will gather weather data through an array of sensors including an anemometer, UV-, wind- and humidity sensor.

## 1.4. Summary of work

This report investigates various remote data transmission technologies, covers the network design required for data logging via TCP sockets and the hardware design of a remote data-collection station. Furthermore, this report will expand on the options for data storage and online access to data collected in the past, this extends into web-design and the creation of a custom deployed web dashboard for displaying data collected - accessible anywhere on the planet. The system designed within this report is robust against power-outages and will continue with its collection of data logged remotely during loadshedding, the locally cached data will be bulk uploaded automatically to an online database when an internet connection is re-established. Power calculations and battery management techniques for off-grid operation of the remote station will be covered and implemented within the final product. The final system will be built and tested to ensure full operation, further edge-cases will be tested and physical measurements of the hardware will be tracked to ensure the longevity of the system. Finally, the real-world application and feasibility of the system will be discussed along with recommended adaptations and faults within the current system.

## 1.5. Scope

This report can be divided into several fields of design: network creation, electronic design and web-design. With regards to wireless network creation, this report focuses purely on the networking aspect of communication between devices and not the antenna design or signal modulation required for transmission. Data is transmitted wirelessly using off-the-shelf products configured for this solution's specific requirements. A custom, published website is created to visualise the data gathered through this system. This in turn is done by utilising pre-existing web frameworks - see Section 2.5.2. The custom dashboard created along with the online database will be deployed to the internet using a service called Heroku [7], making it available globally to anyone with an active internet connection. Finally, hardware design is completed by using low-level components - resistors, capacitors and operational amplifiers.

# **Chapter 2**

## **Literature Review**

### **2.1. Related work**

#### **2.1.1. Objectives**

The paper “Smart Community Monitoring System” [8] seeks to create *Smart Communities* through the implementation of IoT networks. This paper implements cyber-physical systems that try to improve community safety, home security, healthcare quality, and emergency response abilities.

#### **2.1.2. Methods**

The creation of a Wi-Fi network through which several smart homes communicate using the MQTT protocol. Each smart home is fitted with a range of sensors constantly logging data and security conditions from each household. This data is sent to a ThingSpeak IoT platform to be visualised through charts and graphs.

#### **2.1.3. Results**

Successful integration of separate disconnected houses and monitoring of household conditions. The data is easily viewed via the ThingSpeak platform.

#### **2.1.4. Remaining challenges**

The solution proposed in the report listed above leaves much to be desired. This system is limited to the standard Wi-Fi range of the modules used. For a long-distance solution a different approach will be required. This system has no offline capacity. Should a power or internet outage was to occur sensor data recorded won’t be logged by the system. Full website customisability is not available due to a pre-made ThingSpeak platform being used. This also compromises potentially private data to the public.

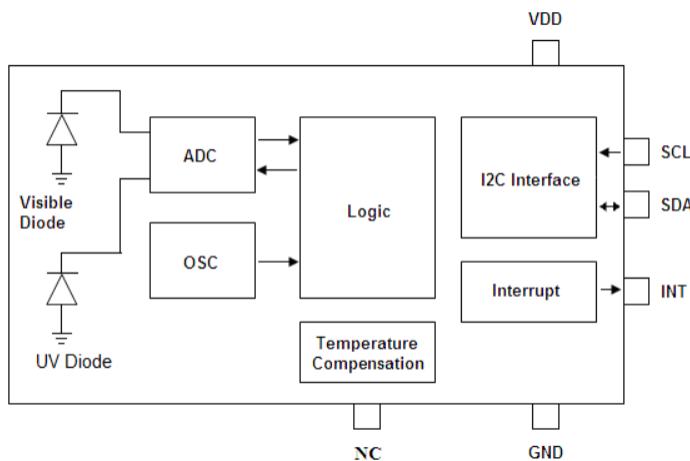
## 2.2. Physical phenomena and measurement

### 2.2.1. Standard Test Conditions

- **Cell Temperature:** Temperature of individual cells within the PV module; can vary from the ambient temperature [25°C]
- **Irradiance:** Light intensity [1000W/m<sup>2</sup>]
- **Air Mass:** Coefficient of air quality - quantifying clarity and viscosity [1.5]

### 2.2.2. UV

The LTR390 UV sensor utilised within this report is a photodiode-type UV sensor. This sensor can measure both ambient light (ALS) and ultra-violet light (UVS) through measuring illuminance. When light strikes either the ALS or UVS diode the internal electrons are energised, resulting in a current flow [9]. This current is measured through an onboard current sensor on the LTR390 after which the resulting signal is passed through an analog-to-digital converter. These digital values of the measurements are passed to a microcontroller through the I<sup>2</sup>C communication protocol. The hardware diagram of the LTR390 is shown at Figure 2.1.



**Figure 2.1:** A diagram indicating the high-level hardware design of the LTR390 UV sensor. [10]

### 2.2.3. Temperature

The temperature sensor utilises a negative temperature coefficient thermistor to determine the ambient temperature [11]. A decrease in the ambient temperature causes a reduction in the resistance of this element, therefore an increase in current. A current sensing circuit is used to determine the change in current and accordingly the change in temperature can

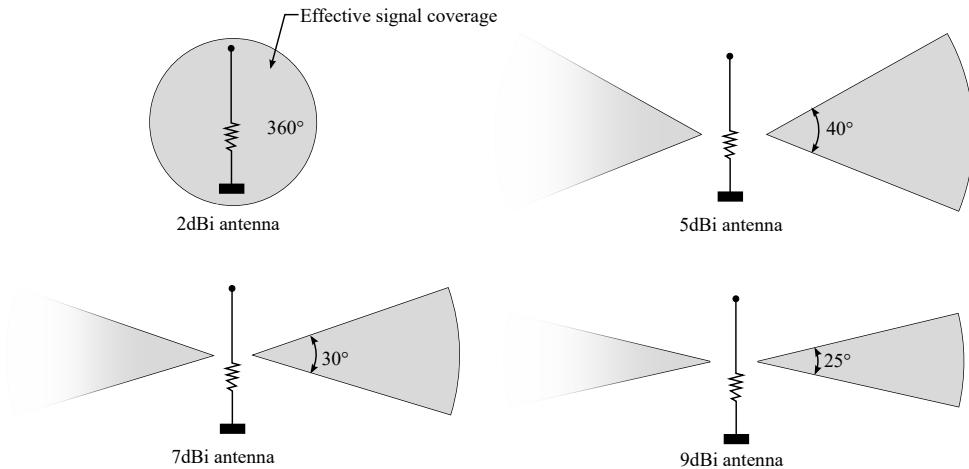
be detected. After processing the signal from the thermistor, the signal can be digitised through an analog-to-digital converter.

### 2.2.4. Humidity

A DHT22 sensor will be used for its humidity measurement capabilities. Measurements are achieved through a capacitive humidity sensing element. A hygroscopic dielectric material is placed between two electrodes to form a capacitor. As water adsorption within the capacitor increases so will the capacitance. The signal from the capacitor is processed by an onboard IC and converted to a digital signal through an analog-to-digital converter.

### 2.2.5. Antenna gain

Antennas increase signals not through amplification but through the redistribution of the signal in a specific direction [12]. The increase in a signal from an antenna is called the antenna gain [dBi]. Therefore, to increase the effective range of the antenna one must increase the antenna gain, at expense of range in other directions. To determine the gain required for an antenna, one must consider the application of the antenna; consider range and directivity of use. A high-level illustration of the effects of antenna gain is shown in Figure 2.2.

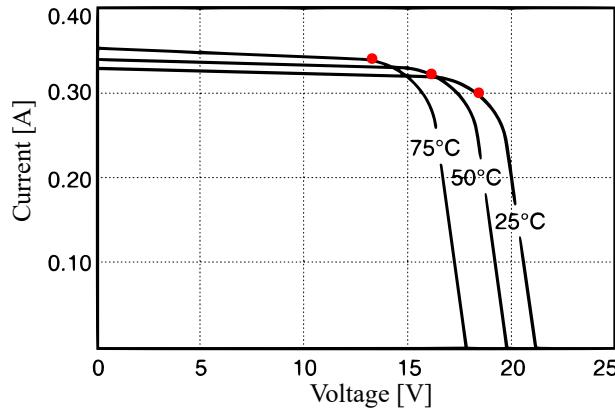


**Figure 2.2:** Figure indicating effect of antenna gain on signal propagation.

## 2.3. Hardware

### 2.3.1. Solar photo voltaic cells and solar modules

A PV Module is an array of photo-voltaic cells assembled into a single module. A photo-voltaic cell is a semiconductor material that absorbs photons from a light source, which in turn allows for the flow of electrons through the material therefore resulting in a DC current flow. Polycrystalline PV modules have a conversion efficiency of around 13-16% [13]. Figure 2.3 describes the ratio between the current provided by the PV module and the voltage across its terminals.



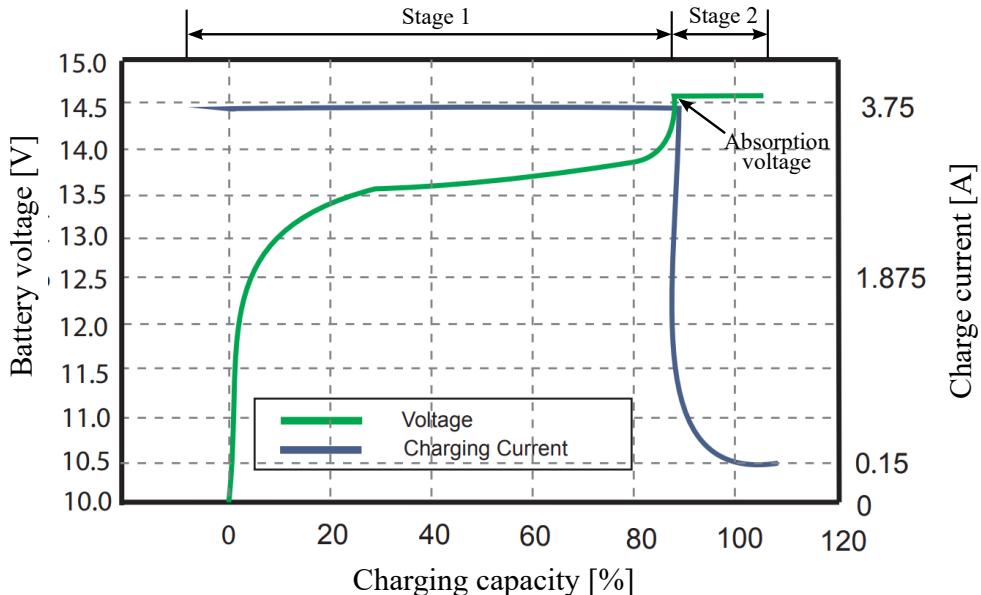
**Figure 2.3:** Graph indicating current-voltage curves of a SLP005-12 PV module at different temperatures [1]. The maximum power points are indicated in red.

- $I_{SC}$ : The current flowing in the PV module when the voltage over the terminals is zero.
- $V_{OC}$ : The voltage over the module terminals when in open circuit.
- $MPP$ : The point on the curve where the ratio between the current and voltage will give the highest power,  $P = I \cdot V$ . This point is found at the “knee” or sharp curve of the graph.

### 2.3.2. Solar charge controller

LiFePo4 batteries can be charged with either a 1-stage profile (constant current charge, CC) or through a 2-stage profile (constant current, constant voltage charge, CC-CV) [14]. During the first stage the battery charging device will provide power at a voltage just above the present voltage of the battery at a constant current, this stage continues until the battery reaches its absorption voltage. However, when the battery reaches the second stage, also known as the absorption stage, the battery charging device should hold the absorption voltage until the charge current reduces to termination current. See Figure

2.4 for a LiFePo4 battery charging graph. CC charging will stop charging the battery at around 97% while CC-CV charging can charge the battery up to 100%. This report will focus on 2-stage PWM solar charge controllers. PWM solar charge controllers act as switches between a PV module and a battery. During stage 1, the switch is ON allowing current to flow from the PV module to the battery. At stage 2 (the absorption stage), the solar charge controller generates a PWM signal to control the switch. This PWM signal, effectively switching between ON and OFF, results in a stable voltage output at the absorption voltage of the battery.



**Figure 2.4:** Graph indicating the charging characteristics of a LiFePo4 battery [2]

### 2.3.3. Power over Ethernet

The Power over Ethernet (PoE) facility, which is standardized under IEEE 802.3af, was developed to simplify and reduce the cost of network planning, cabling and installation [15]. The equipment is powered directly over the data line i.e. ethernet cable such as CAT5. In eight-wire ethernet networks such as the one designed within this report, PoE uses spare-pair power transmission. PoE injectors are devices that take in a data ethernet line and output a PoE line. Some devices require active PoE injectors, through which device voltage requirements are communicated between the device and the injector before power is sourced from the injector. However, this report focuses on passive PoE injectors where no serial communication between the powered device and the injector is required.

## 2.4. Network

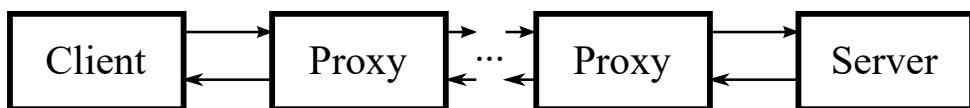
### 2.4.1. Communication protocols

#### TCP

Transmission Control Protocol is a communications protocol focused on error-checked delivery of messages between devices on a network. Through a series of acknowledgements TCP can detect lost or duplicated packets, network congestion or out-of-order delivery of packets and request the re-transmission of data. If unsuccessful the target device will be notified of the failed delivery.

#### HTTP

Hypertext Transfer Protocol is client-server based communication protocol. This communication protocol is always initiated through the client, traditionally a web browser, before being handled by the server. The server will accept the request and provide an answer to the client, called the response. Information sent between the client and the server may pass through various proxy servers, depending on the network configuration. The information sent through HTTP requests is not limited to hypertext documents but can include object structures, text or audio-visual material. After receiving a response from the server, the client - a web browser in this case - will parse the response and process the contained information. This can include the layout of websites through HTML and CSS, extra commands through JavaScript or arbitrary serial data [16].



**Figure 2.5:** A diagram indicating the relay of information through a HTTP request. Arbitrary amounts of proxy servers can be involved in this process.

### 2.4.2. Routers and bridges

#### Routers

Routers are hardware devices that connect two separate networks, usually a local network and a wide area network. This done by forwarding data packets to their requested IP addresses and ensuring connectivity between all devices on the local network to an external network – such as the internet. Modern Wi-Fi routers contain both a router and a modem, modems being devices that modulate and demodulate signals from your ISP. The Wi-Fi router is also responsible for allocating a unique IP address to all devices on the local network. This can be done dynamically through DHCP or devices can be given static IP

addresses. The solution proposed in this report will allocate static IP addresses to devices on the network.

### 2.4.3. Bridges

A network bridge serves two main functions – creating connections between two separate networks or dividing a single network into two. For this report network bridges will be utilised to combine two networks, so that all devices can communicate with each other. The network bridge logs the MAC address of all communicating devices on the network into a forwarding table. When a message is sent to a device from another network, the bridge will forward the message to the requested device by referring to its MAC address logged on the forward table. [17].

## 2.5. Databases and web-design

### 2.5.1. Database models

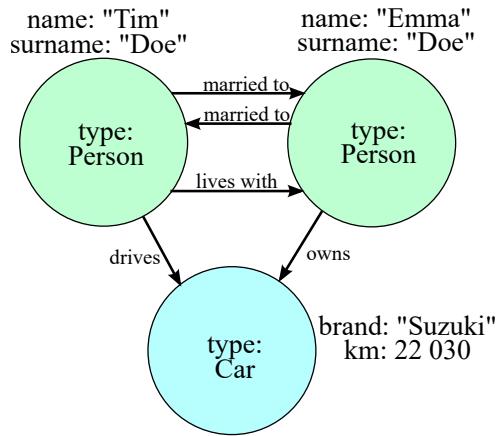
A database is an organized collection of structured information [18]. Besides providing storage for data, modern database solutions allow for manipulation and retrieval of data locally or through the cloud. Several different models for database storage exist. Two popular types are discussed in Section 2.5.1 and 2.5.1.

#### Graph database

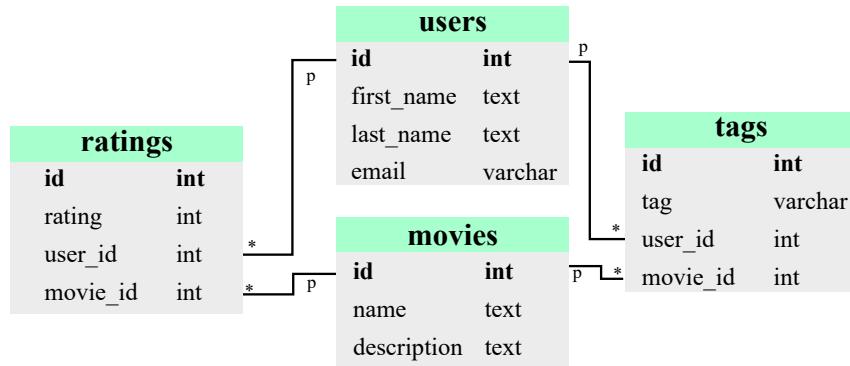
A data-model that stores information in the form of nodes and relations. Within this model data elements are stored as node with relationships to other nodes in the database. This system allows for effective traversing through highly relationship driven data quickly. Graph databases excel in navigating deep hierarchies, discovering inter-relationships and finding hidden relationships between obscure items. [19]

#### Relational database

A database model that allows for data holding tables to be related to other tables using common attributes. Within relational databases there are tables that store information, represented as columns (attributes) and rows (records). All records within a relational database table contain an unique identifier known as its primary-key. Relationships to another table are created through a record's foreign-key; a reference to a record on another table's primary key [20]. Data insertions, deletion and selection can be executed through SQL - Structured Query Language.



**Figure 2.6:** In the figure above nodes are indicated in light blue and light green. Nodes have a type and key-value pairs associated with them. Relationships between nodes are indicated with the arrows.



**Figure 2.7:** The relation schema above indicates user ratings given to movies and the associated tags of each movie. This schema holds four tables each containing keys for their respective fields, the primary-keys are indicated in bold. Relations between tables are indicated with lines, with (\*) representing the table's foreign keys and (p) indicating the primary key of the related table.

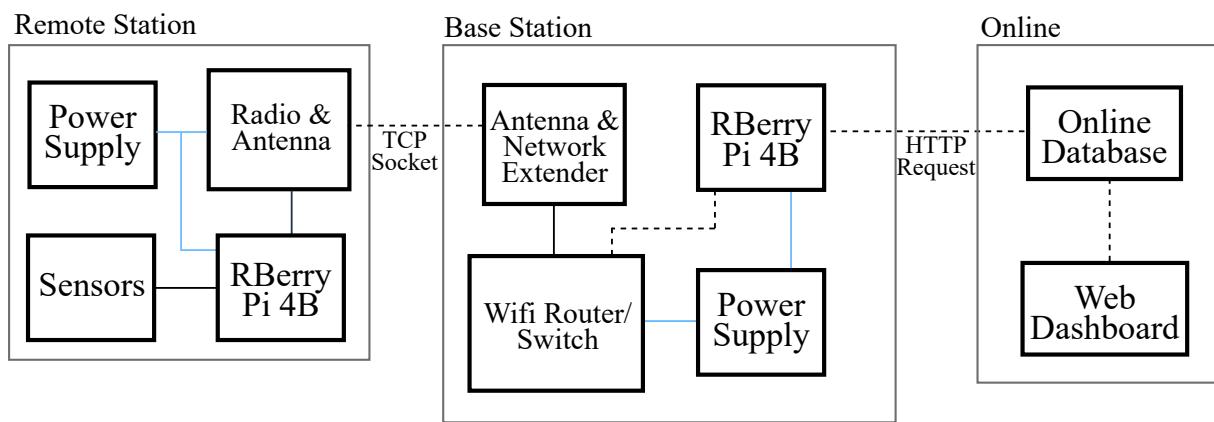
## 2.5.2. Web frameworks

Web frameworks are a set of resources utilised by software developers for the creation of web applications. Web frameworks oversee the more repetitive and low-level processes required for building a web application [21]. These features include, but are not limited to, web caching, authorisation and URL mapping. Web frameworks drastically improve the reliability and safety of web applications.

# Chapter 3

## System Design

### 3.1. System diagram



**Figure 3.1:** The high-level system diagram of the solution proposed within this report. Subsystems include a remote station, base station and the online components. Power lines are indicated in light blue, with data communication indicated in black. Wireless communication is indicated with dashed lines.

### 3.2. Remote station

The remote station subsystem is a moveable, electrically independent hardware system that will transmit and receive data to and from the base station through the wireless network. This station has a PV module and LiFePo4 battery for power, a PWM solar charge controller for charging the battery and a high-side switch designed with hysteresis in mind to ensure undervoltage protection. Current protection is provided to both the battery and the rest of the components through a range of inline fuses. The voltage rails required by the various components within the system is provided through a DC-DC boost and buck converter. The remote station centres around the Raspberry Pi 4b, a powerful small computer with a full operating system. The Raspberry Pi provides GPIO pins for external sensor measurements and a networking interface through an ethernet port, RJ45. For the solution proposed in the report the remote station will act as a basic weather station, including an anemometer, UV-sensor, temperature-sensor and humidity-sensor. The sensor

sampling interval is configurable by the system operator. This system was designed with modularity in mind, therefore adding solenoids, activators, cameras or any other devices that require remote activation will be trivial. Wireless communication with the base station is dependent on a radio-antenna combination, specifically an omni-directional Ubiquiti Bullet M5 radio and antenna pair. This radio will be configured as a bridge between the base station network and the remote station. The radio receives power through a PoE connection - this system includes a DC-PoE injector that will feed the required 15V to the radio.

### 3.3. Base station

The base station acts as the heart of the network containing the network switch and router. For this project a household Wi-Fi router was used to act as both the switch and router to the internet. The wireless range of the network is extended through another radio-antenna combination, specifically the mono-directional Ubiquiti AirGrid M5 radio. A Raspberry Pi 4b is used to process incoming messages from the remote station and upload them either to an online database or a local storage solution independent of internet availability. The solution proposed in this paper strives to be loadshedding resistant therefore requiring another battery configuration similar to that of the remote station at the base station. Otherwise, a household inverter and battery configuration can provide power to the system during times of loadshedding. After internet connectivity has been restored at the base station the locally stored sensor data will be bulk uploaded to the online database, ensuring that there is no lack of sensor data during periods of internet absence.

## 3.4. Online components

### 3.4.1. Database

The system proposed requires live access to sensor data logged previously from anywhere globally. Due to this requirement storing data locally will not suffice. An online database is created to store the sensor data on the cloud and allowing global access to the data logged in the system through HTTP requests. Through this method an online dashboard can access and make changes to the database if necessary. For the case of a weather station such as implemented with this report, each data entry point will contain the following – timestamp of data capture, wind speed, temperature, humidity and the UV-index. The database solution must be able to handle bulk insertions from the base station after long periods without internet access. Bulk insertions are to be completed without crashing or significantly slowing down the database solution. To ensure longevity and scalability of the system, the database solution is expandable in the future. Various database solutions

and their implementation are discussed in Chapter 4.

### 3.4.2. Online dashboard

To view live data from the remote station a graphical user interface is required. For great cross-device compatibility an online dashboard is developed displaying graphs and tables of previously logged data. This web application is deployed to a server which allows global access to the dashboard by visiting the specified URL. The interface is designed to be compatible irrespective of device orientation, resolution or aspect ratio. The web application retrieves information from the online database through HTTP requests and formats this information to be displayed to the operator. A mock-up for the application is shown in Figure G.1.

## 3.5. Metrics

### 3.5.1. Remote station

#### Battery charge

To ensure that the battery stops charging at full capacity and to increase its longevity we must be certain that the SCC significantly reduces current flow to the battery at the required voltage. To test whether the current is reduced as the battery reaches maximum voltage, a current sensor will be placed between the SCC and battery.. During ideal circumstances the current should reduce in accordance with the battery's specification [2].

#### Undervoltage protection

To increase the lifespan of the battery it is required to prevent the battery from discharging completely. It is important to implement hysteresis within an undervoltage protection circuit to prevent sporadic battery switching due to noise. To ensure that the undervoltage protection solution works as intended SPICE and lab power supply simulations will be conducted.

#### Stable rail power

Devices such as the Raspberry Pi and the radio on the remote station must receive power from the LiFePo4 battery onboard. These devices require different voltage supplies than that of the battery, therefore a DC-DC voltage conversion is required. Due to the sensitivity of these devices to voltage spikes and noise, the DC-DC voltage converters will have to be tested under various supply voltage and current conditions to ensure that their output

is suitable. A voltage sweep with a lab power supply will be provided to the converters. Their output will be monitored with voltage meters.

### 3.5.2. Base station

#### Network communication

Data from the remote station travels through a TCP socket through two radio-antenna pairs, through the Wi-Fi router and finally the base station. To create this network one must be aware of all the devices connected. For this purpose a command line tool, NMAP, will be used to scan the network for devices. The network created will allow for remote access to the Raspberry Pi on the remote station through SSH. This in turn will allow for changes and updates to the software to be made on the remote station from the base station. File transfer can also be performed through piping and FTP. Tools for this include NCAT, see Table 4.14.

The effective range of the system will also be tested. For this purpose the line of sight between the two antennas must be ensured. The range provided through this network ranges over kilometers. Online mapping utilities will be used to find the exact distance between the remote- and base station.

#### Online capacity

The system will be left running for four days to ensure stability. Throughout this period the sensor sampling interval of the remote station will be set to five minutes, therefore 1152 records will be inserted to the online database. The online dashboard will be visited daily and refreshed to display new data on the web application. The dashboard will be accessed from multiple devices with different screen resolutions and aspect ratios to ensure legibility.

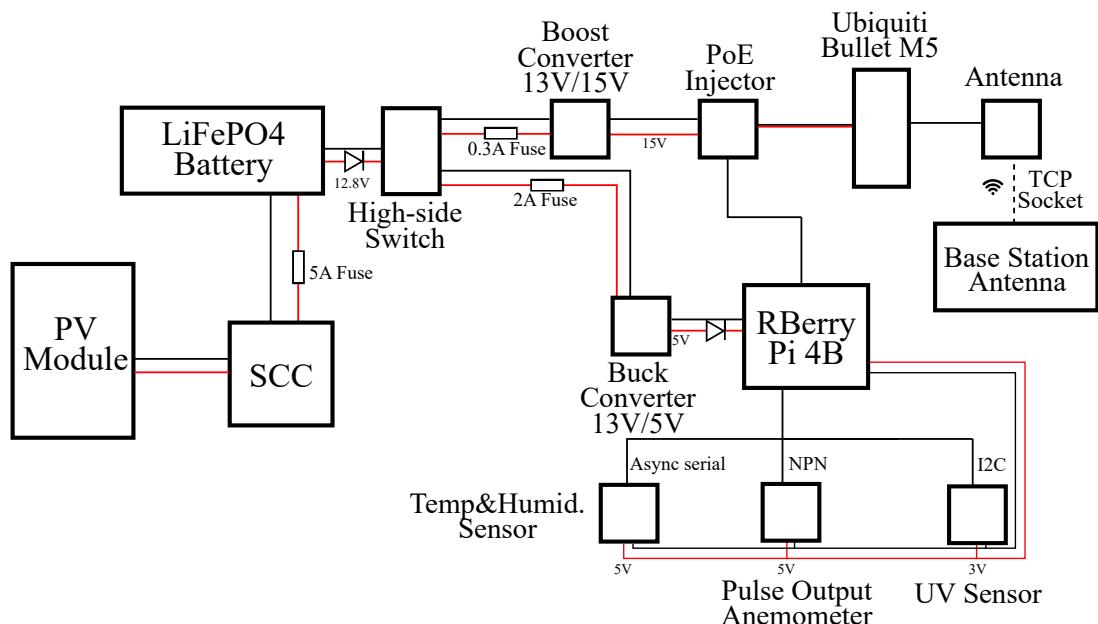
#### Offline capacity

To imitate the loss of internet access due to loadshedding, the RJ45 cable providing the internet connection to the Wi-Fi router will be removed during the operation of the system. Following the disruption of internet access in the system, the offline storage solution will be monitored. Sensor data from the remote station will still be sent irrespective of the internet connectivity within the system. After an internet connection has been re-established, the bulk data uploading capabilities of the system will be tested. To test the uploading capacity 1152 offline data records will be inserted into the online database. This is equivalent to four days of data logging at a sample interval of five minutes. A script is written to determine the elapsed time from when the insertion starts and ends.

# Chapter 4

## Detailed design

### 4.1. Remote station hardware design



**Figure 4.1:** Hardware diagram for the remote station. Power lines are indicated in red. Wireless communication is indicated with dashed lines.

#### 4.1.1. Power

##### Battery selection

The remote station is a stand-alone solar photovoltaic system, therefore requiring a battery for energy storage. To select a suitable battery for the project various types were compared in terms of lifespan, efficiency and cost. In Table 4.1 the characteristics of different variants of lead-acid batteries, lithium-ion batteries and lithium-iron-phosphate are compared. It is found that LiFePo4 batteries exceeds the others in terms of lifespan and usable capacity, both of which being crucial characteristics for stand-alone photovoltaic systems. Therefore a LiFePo4 battery is used within the remote station.

**Table 4.1:** Comparison of batteries for stand-alone PV systems' design. [3]

Battery type	Lead acid; Flooded	Lead acid; AGM	Lead acid; GEL	Li-Mg/Co	LiFePo4
Nominal cell voltage	2.12	2.12	2.12	3.7	3.3
Lifespan [cycles]	300	300	300	300	2000
Energy efficiency [%]	80	80	80	90	> 90
Usable capacity [%]	70	70	70	90	> 90
Self-discharge rate [% per month]	3	3	3	1	1
Typical cost [R/kWh]	1.81	2.71	2.71	5.43	5.43
Environmental friendliness; relative	Low	Low	Medium	High	High

**Table 4.2:** Power consumption of various hardware components in the remote station.

Power consumption [W]	$P_{max}$	$P_{nominal}$
Raspberry Pi 4b [22]	5.1	3.1
Ubiquiti Bullet M5 [23]	6.0	0.6
Buck converter - 85% eff.	0.765	0.465
Boost converter - 85% eff. [24]	0.9	0.09
LTR390 [10]	$3.3 \times 10^{-3}$	$3.3 \times 10^{-6}$
DHT22 [25]	$3.3 \times 10^{-3}$	$0.132 \times 10^{-3}$
Anemometer	$0.167 \times 10^{-3}$	$0.167 \times 10^{-3}$

From Table 4.2 we find that the remote station will have nominal power draw of  $P_{nominal} = 4.256W$ . This amount will be further reduced by optimising the power usage of the Raspberry Pi 4b in Section 4.1.1. The battery selected for the remote station is 12.8V, 7.5Ah LiFePo4 battery from RS Components [2]. This battery has an energy capacity of 96Wh. Before optimising the Raspberry Pi 4b for power efficiency. The prior power calculations are shown in Equation 4.1. DOD refers to the battery's depth of discharge.

$$E_{usable} = E_{rated} \cdot DOD \quad (4.1)$$

$$E_{usable} = 96Wh \cdot 0.9 = 86.4Wh \quad (4.2)$$

### Power reduction

To extend the battery life of the LiFePo4 battery, the power usage of the Raspberry Pi can be reduced by making some configurations. The Raspberry Pi in the remote station will only be accessed remotely via SSH. Therefore all onboard peripheral interfaces can be disabled to reduce the current draw of the Raspberry Pi.

**Table 4.3:** Power saving on Raspberry Pi by disabling features. [4]

Feature	Current draw [mA]
Disable HDMI	25
Disable LED x4	20
Disable Wi-Fi radio & Bluetooth	75

Disabling the features represented in Table 4.3 will reduce the power usage of the Raspberry Pi by 0.6W. This brings the device's total nominal consumption down to 2.5W. The estimated total nominal power consumption of the remote station is now estimated to be  $E_{total} = 3.656W$ .

### PV module and solar charge controller

Assuming an effective window of operation of 5h/day for the PV module [26], the total energy generation is calculated as  $E_{generated} = 5W \cdot 5hours = 25Wh$ . From a single 5W PV module the system will be not sustainably operate in a stand-alone configuration. Another 5W PV module is to be connected to ensure stable operation of the remote station.

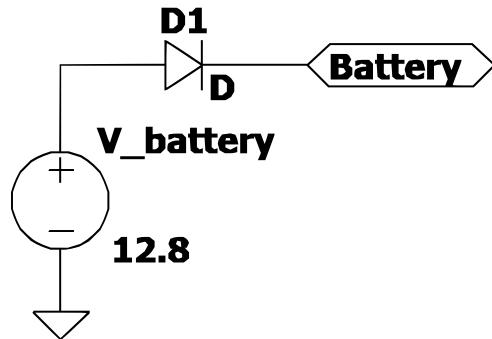
**Table 4.4:** SLP005-12R PV-module rated values and properties [1]

Feature	Rated value
Number of cells	36
$V_{OC}$	21.6V
$I_{SC}$	0.34A
$P_{OUT}$ (rated under STC)	5.00W

### 4.1.2. Undervoltage protection

#### Overview

This section of the detailed design chapter focusses on providing undervoltage protection to the 12.8V LiFePo4 battery. Between the load of the battery and its terminals a high-side switch is placed that can be toggled to either allow or block current flow from the battery's terminals. The design described in this section incorporates hysteresis within the toggling of the high-side switch, turning off at 11.7V (90% discharged) and turning back on at 12.0V (83% discharged). The voltage range of the battery terminals stretches from 14.7V (when charging via the SCC) down to 10.5 ( $> 98\%$  discharged). A diode,  $V_F = 0.5V$ , is placed inline between the high-side switch and the battery terminal to prevent current backflow. The subsystems of this design include a high-side switch, battery voltage clipper-scaler and a Schmitt trigger.

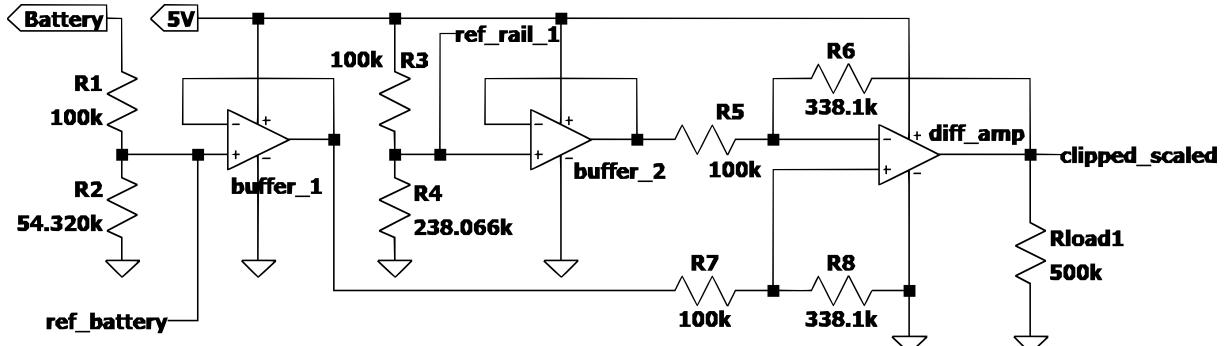


**Figure 4.2:** Hardware schematic for LiFePo4 battery and diode.

**Table 4.5:** Voltage points to consider with undervoltage protection design

Voltage point	Symbol	Voltage [V]	Reason
Minimum battery voltage pre diode	$V_{min\_terminal}$	10.5	Battery at 98% depletion
Maximum battery voltage pre diode	$V_{max\_terminal}$	14.7	Maximum charge voltage from SCC
Diode forward voltage	$V_F$	0.5	From datasheet
Minimum battery voltage after diode	$V_{min}$	10	$V_{min\_terminal} - V_F$
Maximum battery voltage after diode	$V_{max}$	14.2	$V_{max\_terminal} - V_F$
Switch open point	$V_{open\_switch}$	11.7	Battery at 90% depletion
Switch close point	$V_{close\_switch}$	12.0	Battery at 83% depletion

### Battery voltage clipping and scaling



**Figure 4.3:** Hardware schematic for battery voltage clipper and scaler.

The undervoltage protection circuit should cut current flow from the battery at 11.7V and allow current flow again when the battery terminals reach a voltage of 12V. However, all the operational amplifiers used with this design are rated for operation below 5V. When scaling the battery terminals down to 5V through a simple voltage divider, the voltage difference between the two hysteresis points 11.7V and 12V becomes very small, 0.005V. This small voltage difference is very susceptible to noise and will reduce the effectiveness of the undervoltage protection circuit. To mitigate this issue a voltage clipper and scaling circuit is designed, this circuit will clip and scale the 10V-14.2V battery voltage range to 0V-5V allowing for the effective implementation of hysteresis. First the maximum voltage

of 14.2V is scaled to 5V through a voltage divider.

$$5 = 14.2 \cdot \frac{R_2}{R_1 + R_2} \quad (4.3)$$

$$\implies \frac{R_2}{R_1 + R_2} = 0.352 \quad (4.4)$$

Next the reference voltage reference  $V_{ref\_battery}$  is calculated using the resistor ratio from Equation 4.3.  $V_{ref\_battery}$  is used to calculate the resistor ratio required to determine  $V_{ref\_rail\_1}$ .  $V_{ref\_rail\_1}$ 's voltage source is a the 5V output from a buck converter implemented in Section 4.1.3. Both  $V_{ref\_battery}$  and  $V_{ref\_rail\_1}$  are fed into voltage buffers 1 and 2 respectively to maintain constant output irrespective of feedback loops.

$$V_{ref\_rail\_1} = V_{ref\_battery} = 10 \cdot \frac{R_2}{R_1 + R_2} = 3.521V \quad (4.5)$$

$$\implies \frac{R_4}{R_3 + R_4} = 0.7042 \quad (4.6)$$

Now a differential amplifier is implemented that delivers an output  $V_{clipped\_scaled}$  ranging from 0V-5V. The output from buffer 1 feeds into  $V_{in+}$  while buffer 2 feeds into  $V_{in-}$  of the differential amplifier. The differential amplifier equation shown in Equation 4.7 is used to find the resistor ratio  $\frac{R_6}{R_5}$ . We know that within differential amplifiers  $\frac{R_6}{R_5} = \frac{R_8}{R_7}$ . Resistor values are selected using the calculated ratios - see Table 4.6. Large resistance values are chosen where applicable to reduce current draw. The output from the comparator is fed into a logic inverter, the resultant signal controls the high-side switch.

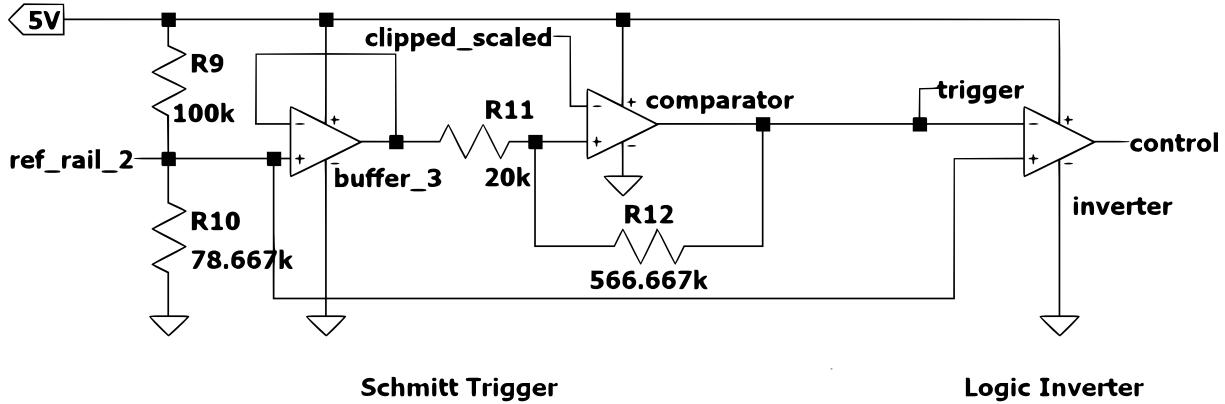
$$V_{clipped\_scaled} = \frac{R_6}{R_5} \cdot (V_{in+} - V_{in-}) \quad (4.7)$$

$$5 = \frac{R_6}{R_5} \cdot (5 - 3.521) \quad (4.8)$$

$$\implies \frac{R_6}{R_5} = \frac{R_8}{R_7} = 3.381 \quad (4.9)$$

**Table 4.6:** Resistor value selection

Resistor	Resistance [ $\Omega$ ]
$R_1, R_3, R_5, R_7$	100k
$R_2$	54.32k
$R_4$	238.07k
$R_6, R_8$	338.1k

**Schmitt trigger****Figure 4.4:** Hardware schematic for Schmitt trigger and logic inverter.

This section focuses on the implantation of a Schmitt trigger to introduce hysteresis into the undervoltage protection high-side switch. The Schmitt trigger consists of a operational amplifier with a positive feedback loop, therefore acting as a comparator. In Section 4.1.2  $V_{min}$  and  $V_{max}$  were clipped and scaled to 0V-5V. After  $V_{min}$  was clipped to 0V the remaining range is scaled by a factor of 1.19. Using this scaling factor we find  $V_{open\_switch\_scaled}$  and  $V_{close\_switch\_scaled}$  in Equation 4.10 and 4.11.

$$V_{open\_switch\_scaled} = (V_{open\_switch} - V_{min}) \cdot 1.19 = 2.034V \quad (4.10)$$

$$V_{close\_switch\_scaled} = (V_{close\_switch} - V_{min}) \cdot 1.19 = 2.390V \quad (4.11)$$

We see that the difference in voltage between the two hysteresis points is now 0.357V, a significant improvement from the 0.005V from using only a voltage divider on  $V_{min}$  and  $V_{max}$ . To determine the reference voltage for the Schmitt trigger  $V_{ref\_rail\_2}$ , we summate and divide  $V_{open\_switch\_scaled}$  and  $V_{close\_switch\_scaled}$ . The reference voltage is passed through a voltage buffer for stability from the feedback loop on the comparator operational amplifier. A voltage divider is used to calculate resistors for the reference voltage  $V_{ref\_rail\_2}$ , shown in Equation 4.12. Finally  $R_{11}$  is added between the voltage buffer and the comparator to prevent current backflow.

$$V_{ref\_rail\_2} = \frac{V_{open\_switch\_scaled} + V_{close\_switch\_scaled}}{2} = 2.2015 \quad (4.12)$$

$$\Rightarrow V_{ref\_rail\_2} = V_{rail} \cdot \frac{R_{10}}{R_9 + R_{10}} \quad (4.13)$$

$$\Rightarrow \frac{R_{10}}{R_9 + R_{10}} = 0.4403 \quad (4.14)$$

Finally we find the feedback resistor  $R_{12}$ :

$$\frac{R_{12}}{R_9} \approx \frac{V_{open\_switch\_scaled}}{V_{open\_switch\_scaled} + V_{close\_switch\_scaled}} = 5.667 \quad (4.15)$$

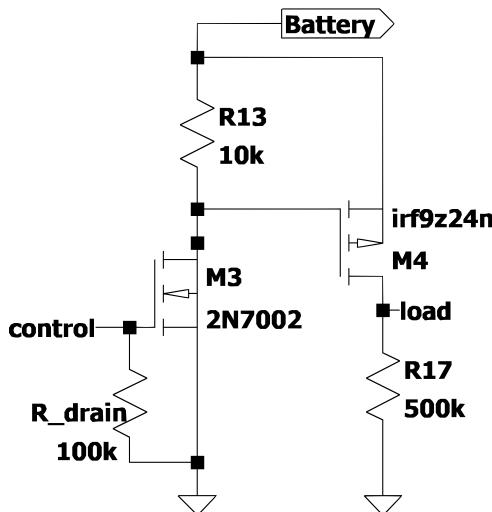
$$\implies \frac{R_{12}}{R_9} \approx 5.667 \quad (4.16)$$

**Table 4.7:** Resistor value selection

Resistor	Resistance [ $\Omega$ ]
$R_9$	100k
$R_{10}$	78.67k
$R_{11}$	20k
$R_{12}$	566, 67k

### High-side switch

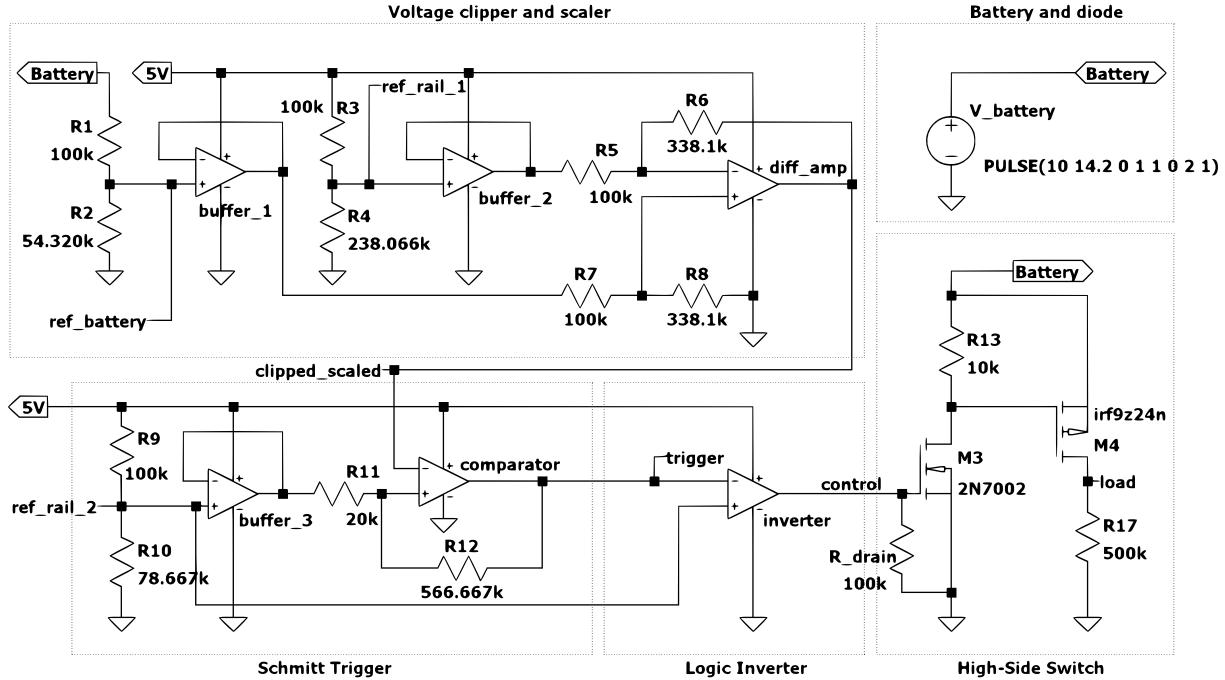
The high-side switch consists of a PMOS and NMOS transistor as seen in Figure 4.5. The NMOS serving as the intermediary between the control voltage and the battery current controlling PMOS transistor. When 5V is applied to the gate of the NMOS the threshold voltage  $V_{Th}$  is reached allowing current to flow from the drain to the source (ground in this case). This in turn creates a difference in voltage between the gate and source of the PMOS transistor, exceeding  $V_{Th}$  of the PMOS and allowing current from the battery to flow from the source to the drain of the transistor. The load circuit is connected to the drain of the PMOS. A resistor  $R_{drain}$  is placed at the gate of the NMOS to drain current coming from the control circuit once its voltage reduces to 0V, this improves the transient response of the high-side switch.



**Figure 4.5:** Hardware schematic for high-side switch.

## Full schematic

The full undervoltage protection schematic is shown in Figure 4.6. The SPICE results from this design is shown in the Sections 5.1.2 - 5.1.2.



**Figure 4.6:** Hardware schematic for undervoltage protection.

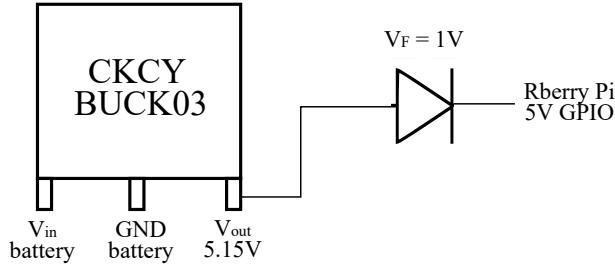
### 4.1.3. DC-DC converters

#### Buck converter

To provide the Raspberry Pi 4b with the 5 volts required for operation an unbranded *Model: CKCY BUCK03* buck regulator is used. The regulator receives an input voltage between 4.5V-16V and gives an output of 5.2V. The Raspberry Pi has a maximum input voltage of 5.2V, therefore a diode with a forward voltage  $V_F = 1V$  is used between the output voltage of the regulator and the Raspberry Pi. The regulator's output voltage was measured under various conditions to define its true properties, see Section 5.1.3. A maximum output voltage of 5.150V was found, therefore the Raspberry will always operate within safe margins.

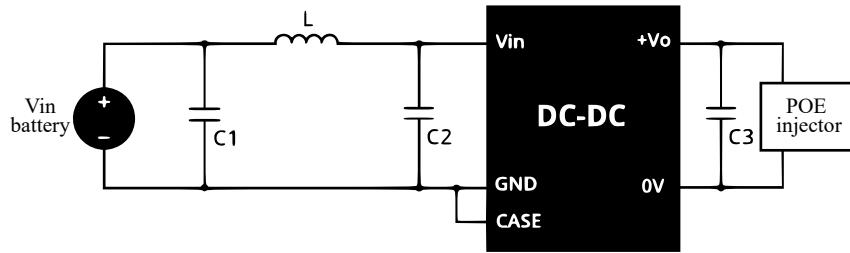
#### Boost converter

To provide the required 15V that feeds into the POE injector that will in turn power the Ubiquiti Bullet M5 radio a PYBJ6-D12-S15 boost converter is used. The Bullet M5 radio requires a differential 15V source. This is obtained through the output reference 0V and reference 15V pins from the boost converter. The hardware configuration for the



**Figure 4.7:** Hardware schematic for 12.8V/5.2V buck converter with  $V_F = 1V$  diode.

PYBJ6-D12-S15 boost converter is shown in Figure 4.8. This configuration is obtained from its datasheet [24].



**Figure 4.8:** Hardware schematic for 12.8V/15V boost converter values are shown in Table 4.8.

**Table 4.8:** Boost converter capacitor and inductor values

Component	Value
$C_1, C_2$	$4.7\mu F$
$C_3$	$10\mu F$
$L$	$4.7\mu H$

#### 4.1.4. Fuses

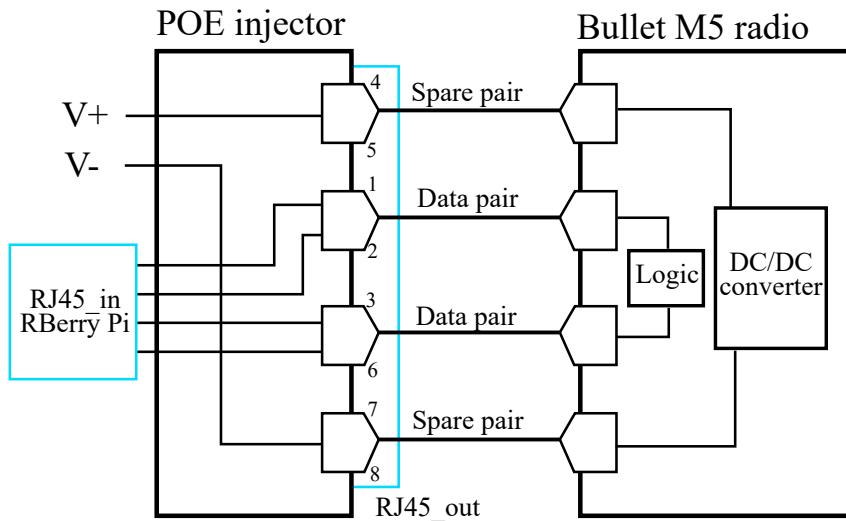
To ensure overcurrent protection within the remote station a selection of inline fuses is utilised to protect components from excessive current. The fuse ratings and placement adjacent nodes are listed in Table 4.9.

**Table 4.9:** Fuse selection and placement

Fuse	Current rating [A]	Expected max current [A]	Adjacent nodes
Blade fuse	5A	0.3	SCC - LiFePo4 positive terminal
Blade fuse	2A	1.02	PMOS drain - $V_{in}$ buck converter
Resettable fuse	0.5	0.4	PMOS drain - $V_{in}$ boost converter

### 4.1.5. Power over Ethernet injector

The DC-DC POE injector combines the data communication via an RJ45 cable from the Raspberry Pi and the 15V differential voltage from the boost converter to an output RJ45 cable feeding into the Bullet M5 radio. The injector designed below utilises a spare-pair power feed configuration to transfer power through the spare pairs 4/5 (+) and 7/8 (-). Connectors 1/2 and 3/4 are used for data communication between the devices. The hardware schematic of the POE injector is shown in Figure 4.9.

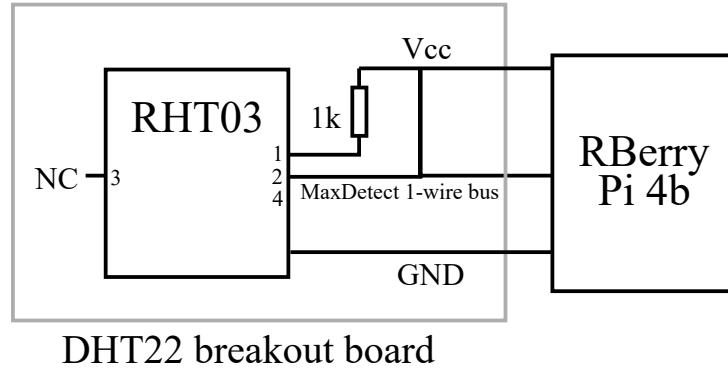


**Figure 4.9:** Hardware schematic for DC-DC POE injector.

### 4.1.6. Sensors

#### Temperature and humidity

Both the temperature and humidity measurements reach the Raspberry Pi from a DHT22 sensor breakout board. The breakout board centres around an RHT03 unit from which measurements are taken. The breakout board schematic is shown in Figure 4.10. Communication between the DHT22 and the Raspberry Pi is achieved through a non-standard single line communication protocol - *MaxDetect 1-wire bus*. This communication protocol is interpreted through a library created by Adafruit - adafruit-circuitpython-dht [27]. To read the measurements class object is created initialising the sensor, from which the temperature and humidity readings can be accessed. It is important to note that the values can only be read once every two seconds.



**Figure 4.10:** Hardware schematic DHT22 breakout board.

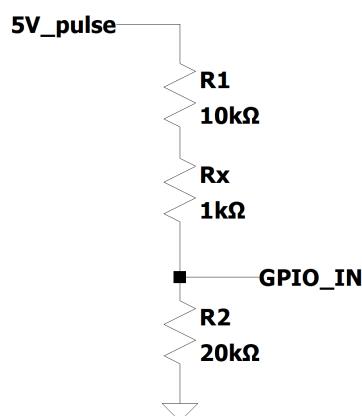
### Anemometer

To obtain the wind speed at the remote station a 5V NPN pulsed output anemometer is used. From the datasheet we find that the anemometer has a resolution of  $0.0875m/s$ . This means that the wind speed increases  $0.0875m/s$  for every pulse detected per second. This can be converted to a km/h resolution as follows;  $1m/s = 3.6km/h$  therefore  $resolution_{km/h} = 0.0875 \cdot 3.6 = 0.315km/h$ . The necessary 5V and ground will be provided to the anemometer through a 5V power and ground pin from the Raspberry Pi respectively. The pulsed output from the anemometer ranges between 0V-5V. Therefore a voltage divider is used to protect the input pins on the Raspberry Pi – only rated for 3.3V. See Equation 4.17.

$$V_{GPIO} = V_{5Vpulse} \cdot \frac{R_2}{R_1 + R_2} \quad (4.17)$$

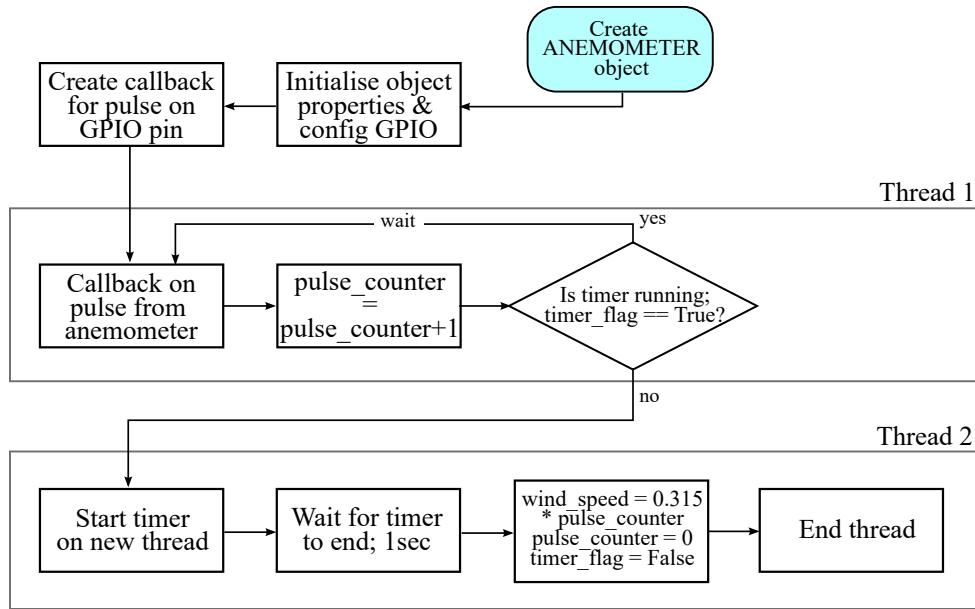
$$\text{Choose } R_1 = 10k\Omega \quad (4.18)$$

$$\implies R_2 = 20k\Omega \quad (4.19)$$



**Figure 4.11:** Voltage divider to scale 5V pulsed output from anemometer to 3.3V for GPIO pin.  $R_x = 1k\Omega$  for safety.

The current wind speed must be randomly accessible by the remote client station. On the firmware side, an object class is created to hold the current wind speed and keep it updated irrespective of the operation of the rest of the remote system. On the creation of the object the required GPIO pins are configured, and operation starts. To ensure that the timer starts exactly at the start of a pulse multi-threading is utilised to ensure accuracy within the measurements. Multi-threading will prevent a timer from starting between two pulses from the anemometer and therefore yielding inaccurate results.



**Figure 4.12:** Flowchart showing the software operation of the anemometer. Through this system, the current wind speed [km/h] can be accessed any time.

## UV sensor

UV measurements are taken from LTR-390UV-01 optical sensor. Sensor measurements are sent to the Raspberry Pi through I2C communication. This device is powered through the 3.3V pin on the Raspberry Pi. The pinout of the LTR-390 is shown in Table 4.10.

**Table 4.10:** LTR-390 pinout

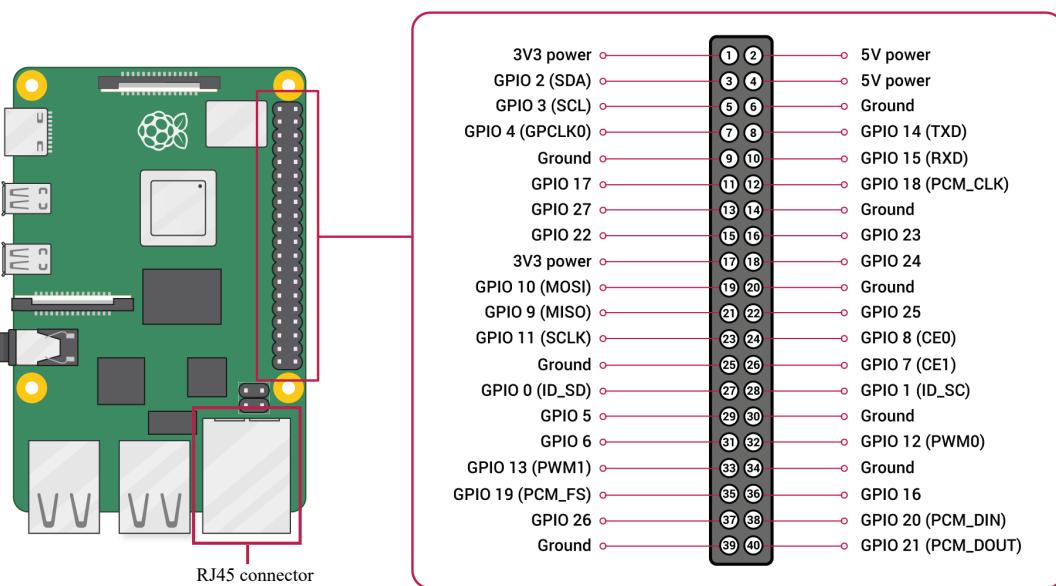
Pin	Symbol	Description
1	$V_{CC}$	Power supply voltage
2	NC	No connection pin
3	GND	Ground
4	SCL	I2C serial clock
5	INT	Level interrupt pin
6	SDA	I2C serial data

### 4.1.7. Raspberry Pi 4b - remote station

This section focusses on the connectivity of the Raspberry Pi at the remote station. Figure 4.11 lists the connection interface of the Raspberry Pi and what devices are connected. It is important to note that the Raspberry Pi receives power via one of its 5V pins and not via the onboard USB-C connector. The network created at the base station is extended to the remote station's Raspberry Pi through its RJ45 connector. The list of pin numbers is shown on Figure 4.13.

**Table 4.11:** Raspberry Pi 4b connections

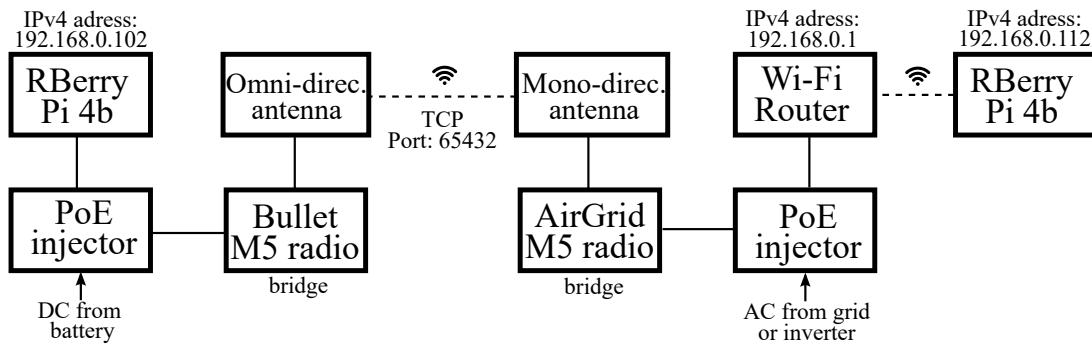
Raspberry interface	Connected device	Type	Protocol/ Voltage
Pin 1	DHT22 and LTR-390	$V_{CC\_out}$	3.3V
Pin 2	Buck converter	$V_{CC\_in}$	5V
Pin 3	LTR-390	I2C communication	SDA
Pin 4	Anemometer divider	$V_{CC\_out}$	5V
Pin 5	LTR-390	I2C communication	SCL
Pin 6	DHT22, LTR-390 and anemometer	GND	0V
Pin 7	LTR-390	Interrupt communication	INT
Pin 11	DHT22	Communication	MaxDetect 1-wire bus
Pin 34	Buck converter	GND	0V
Pin 36	Anemometer	NPN pulsed output	3.3V
RJ45 connector	POE injector to radio	Communication	Ethernet



**Figure 4.13:** Diagram indicating available connection interfaces to the Raspberry Pi from other components and devices.

## 4.2. Network configuration

The network created through the solution proposed with this report centres around a typical home Wi-Fi router. Specifically the TP-Link Archer 50 was used. However any home router can be used. This Wi-Fi range is extended through a Uniquiti AirGrid M5 radio and antenna configuration located at the base station. At the remote station a Ubiquiti Bullet M5 radio intercepts the network and extends it to the remote Raspberry Pi 4b. By default Ubiquiti products can be configured by connecting them to a computer via Ethernet cable and visiting the locally hosted web application dashboard. This application is known as AirOS. Table 4.12 and Table 4.13 shows the necessary configuration to be performed on the two radio devices to create this network. This configuration is done through the AirOS dashboard application. Screenshots are shown in Figure C.1.



**Figure 4.14:** Diagram indicating communication flow between remote- and base station.

**Table 4.12:** AirOS configuration, AirGrid M5 - base station

Setting name	Selection	Reason
Network mode	Bridge	Bridges base- and remote station networks
Wireless mode	Access point	Projects network communication from router wirelessly
SSID	pieter_airgrid_M5	Name discoverable by other Wi-Fi devices
Security	WPA2-AES	Typical security protocol, requires password creation
AirMax	Disabled	Standard protocols to be used; TCP
Channel/Frequency	40 / 5200 MHz	Standard frequency channel of AirGrid M5
Channel width	20 MHz	Smaller channel for faster communication
Bridge IP address	Static	Simplifies interaction
IP address	192.168.10.20	Device IP, not on network - only for configuration

**Table 4.13:** AirOS configuration, Bullet M5 - remote station

Setting name	Selection	Reason
Network mode	Bridge	Bridges base- and remote station networks
Wireless mode	Station	Enables bi-directional NAT
Connect to SSID	pieter_airgrid_M5	Connect to AirGrid radio
Lock to AP MAC	00:27:22:1A:87:08	Ensures connection to AirGrid if SSID changes
Security	WPA2-AES	Typical security protocol, requires password creation
AirMax	Disabled	Standard protocols to be used; TCP
Channel/Frequency	40 / 5200 MHz	Same frequency of AirGrid radio
Channel width	20 MHz	Smaller channel for faster communication
Bridge IP address	Static	Simplifies interaction
IP address	192.168.10.20	Device IP, not on network - only for configuration

After configuring the devices, they will automatically connect to each other if within range. The Wi-Fi router automatically assigns IP addresses to all devices visible on the network, 192.168.0.x in this case. The two radio devices are configured in *bridge* mode. Therefore they will not be visible to the other devices on the network. Communication will appear to propagate seamlessly across the wireless network, that means no manual port forwarding will have to be implemented for communication between the base- and remote station.

## 4.2.1. Utilities

This sections lists several software utilities used to create and manage devices on the network.

**Table 4.14:** Network dependencies and description

Utility	Description
NMAP	Device discovery on network
NCAT	Messaging between devices on network and file piping
Socket	Creation of TCP socket for network communication
SSH	Access to remote Raspberry Pi through network

## 4.3. Software design

### 4.3.1. Database

All data logged through this system is independent of any other data points, this negates the requirement designing a database solution with relational tables or nodal relationships. It was decided that a Postgresql [28] relational database would be most suitable for data storage. The schema for this database solution is a single table with the fields shown in

Table 4.15. This schema is linked to an online database through a service called Heroku. Heroku provides servers to host the database for free. This online database can then be accessed through HTTP requests containing SQL commands. Through the SQL commands data can be viewed, inserted or deleted from anywhere with internet access.

**Table 4.15:** Online database table fields

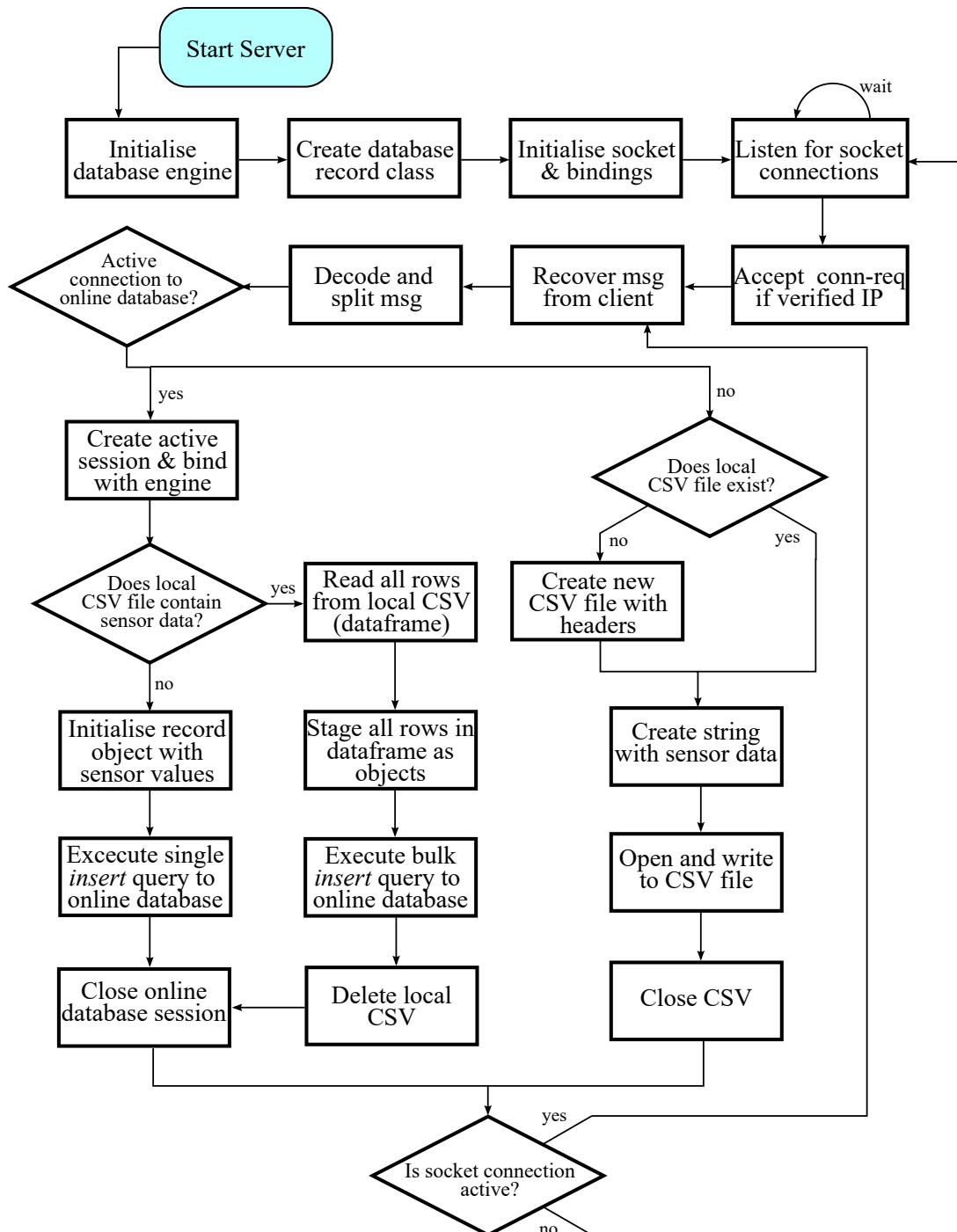
Field	Description
id	[Number] Unique id for every entry
wind	[Real] Wind speed in km/h
temperature	[Real] Temperature in degrees Celcius
humidity	[Real] Humidity in percentage points
uv	[Number] Integer value on UV index scale

### 4.3.2. Online dashboard

To visualise the data logged through the remote system a dashboard web application is developed. The website was built with cross platform compatibility in mind, therefore should scale according to the screen size and resolution of the viewing device. The website contains four graphs for the four measured phenomena respectively. The website also displays a data table listing all data entries in the relational database. On start-up, the website makes a HTTP request to the online database to fetch all data entries. These entries are loaded into a Pandas dataframe for efficient manipulation. The datapoints within the dataframe is displayed on the y-axis of a graph, with the x-axis displaying the measurement timestamp. These graphs can be scaled, panned, zoomed or downloaded as PNG images by the viewer. The websites will refresh its dataset once every 5 minutes to obtain the latest data logged via the hardware system. This web application was deployed via Heroku and can be visited through the URL <http://pieter-skripsie.herokuapp.com/>. All dependencies of the web application are listed in Table 4.16.

### 4.3.3. Server - base station

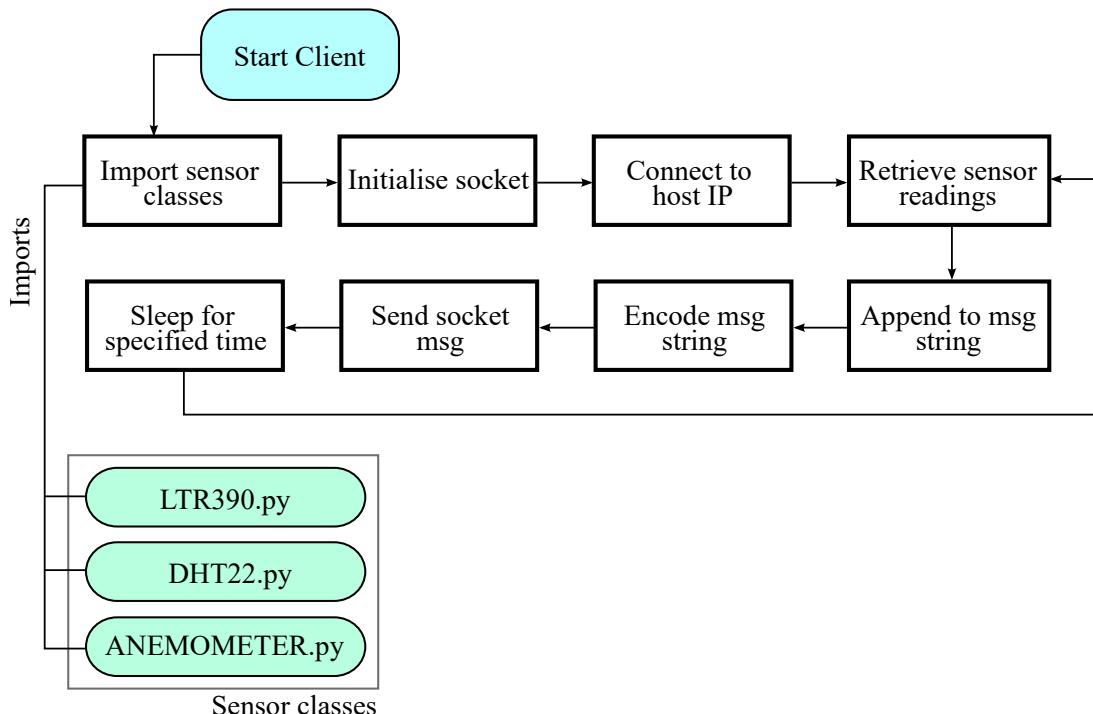
The flow diagram shown in Figure 4.15 shows how data is received, processed and stored at the base station. All messages received from the remote station are UTF-8 encoded. It is important to note that if no internet connection is available all logged data will be stored locally on a CSV file. After the internet connection is restored the data is bulk uploaded onto the online database. Software dependencies are listed in Table 4.16.



**Figure 4.15:** Flowchart indicating the operation of the server program running at the base station.

#### 4.3.4. Client - remote station

The flow diagram shown in Figure 4.16 shows the data measurement, processing and communication to the base station via the network. This is executed on the remote station by the Raspberry Pi. Three classes for the separate sensors were written. Objects are created and requested for measurements. Messages are UTF-8 encoded before being sent to the base station through the TCP socket. The measurement frequency is set to five minutes, but can be increased or reduced based on operator preference.



**Figure 4.16:** Flowchart indicating the operation of the client program running at the remote station.

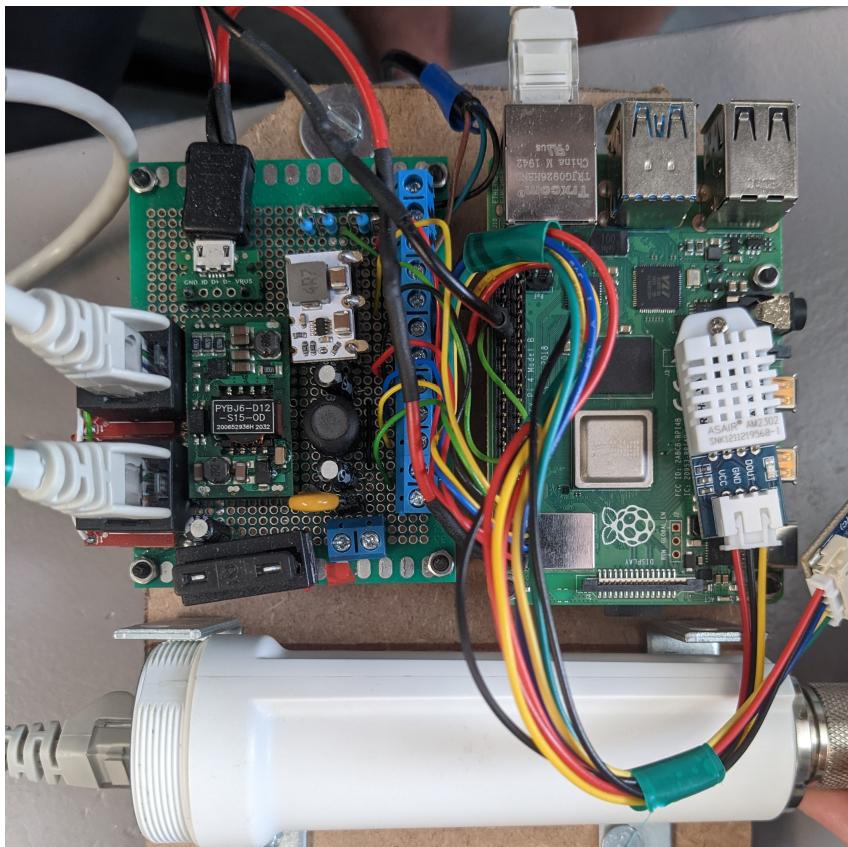
#### 4.3.5. Software dependencies and libraries

**Table 4.16:** Software dependencies and description

Dependency	Description
Flask	Web framework
Plotly	Graph creation library
Dash	UI library
SQLalchemy	Communication between web application and online database
Pandas	Fast bulk data handling library
Heroku	Deployment service
adafruit-circuitpython-dht	MaxDetect 1-wire bus communication library
Socket repository	<a href="https://github.com/pietervwyk/socket">https://github.com/pietervwyk/socket</a>
Dashboard repository	<a href="https://github.com/pietervwyk/sensor_dashboard">https://github.com/pietervwyk/sensor_dashboard</a>

## 4.4. Implementation

The circuit designed for the remote station was constructed on a 4cm x 6cm protoboard. All sensors connect to the protoboard through an array of screw-terminals, which in turn are connected to the Raspberry Pi's pins through a female header and wire connectors. All resistor, capacitor and inductors were mounted through-hole and soldered. A Micro-USB connector is installed to provide the Raspberry Pi power to its 5V pins from the 5V rail. Both the boost and buck converter are mounted into female headers for easy access and removal. The Raspberry Pi, LiFePo4 battery, SCC and the protoboard are mounted on a piece of compressed wood with metal brackets. All sensors and the radio are connected to the protoboard and housed withing a plastic holder. See Figure 4.17 and Figure E.1.



**Figure 4.17:** Photograph of protoboard, radio and Raspberry Pi configuration.

# Chapter 5

## Results

### 5.1. Hardware measurements

#### 5.1.1. PV-module

To test the true properties of the PV-module suggested in this report, its open circuit voltage and short circuit current was measured under various conditions. These measurements are listed in Table 5.1.

**Table 5.1:** SLP005-12R PV-module measured values

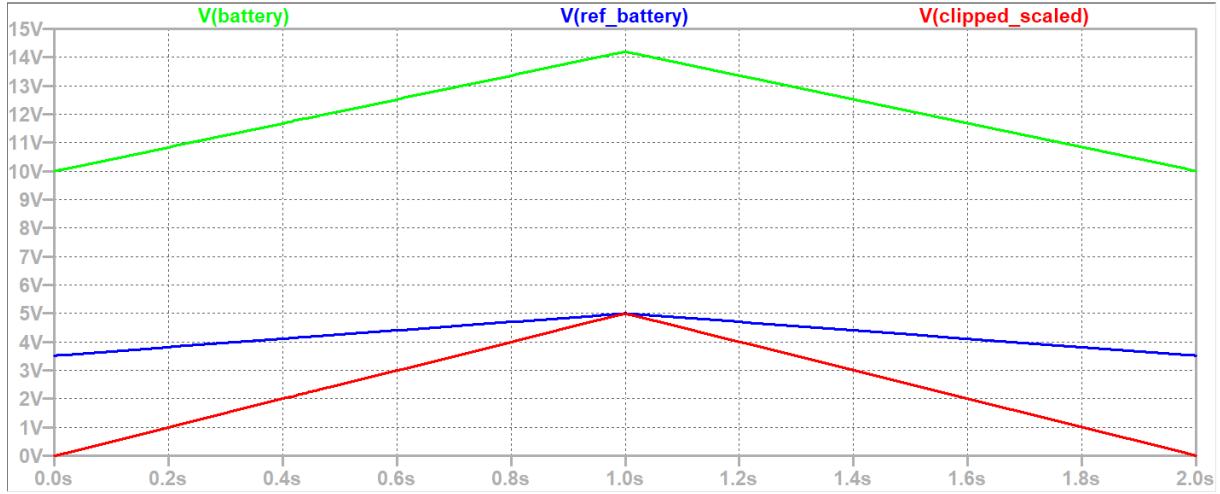
Condition	$I_{SC}$ [mA]	$V_{OC}$ [V]
Dark (covered)	0.006	0.65
Upside down – light on back	1.7	13.06
Ambient light – indoors	0.7	10.93
Oblique sunlight	120	20.6
Perpendicular sunlight	170	22.7

From these measurements we see that the PV-module is performing below specification - at a maximum power output of only 3.86W. The PV-module that was measured should still be well within it specified performance because it has seen very little use in the field. From these measurements one can conclude that for the remote system to function as a stand-alone solar powered station at least three of these PV-modules would have to be used - see power usage in Section 5.1.4.

#### 5.1.2. Undervoltage protection

To test the undervoltage protection solution designed in this report, voltage measurements on the various subcircuits were taken. To perform the measurements, SPICE simulations are performed on the design. In this case a triangular pulse is provided from the simulated battery. This pulse rises from 10V to 14.2V (the calculated maximum voltage range that will reach the circuit) with a rise time of 1s and a fall time of 1s. The full undervoltage protection circuit can be seen at Figure 4.6.

### Battery voltage clippper and scaler



**Figure 5.1:** Output graph from SPICE simulation on battery voltage clipper and scaler circuit.

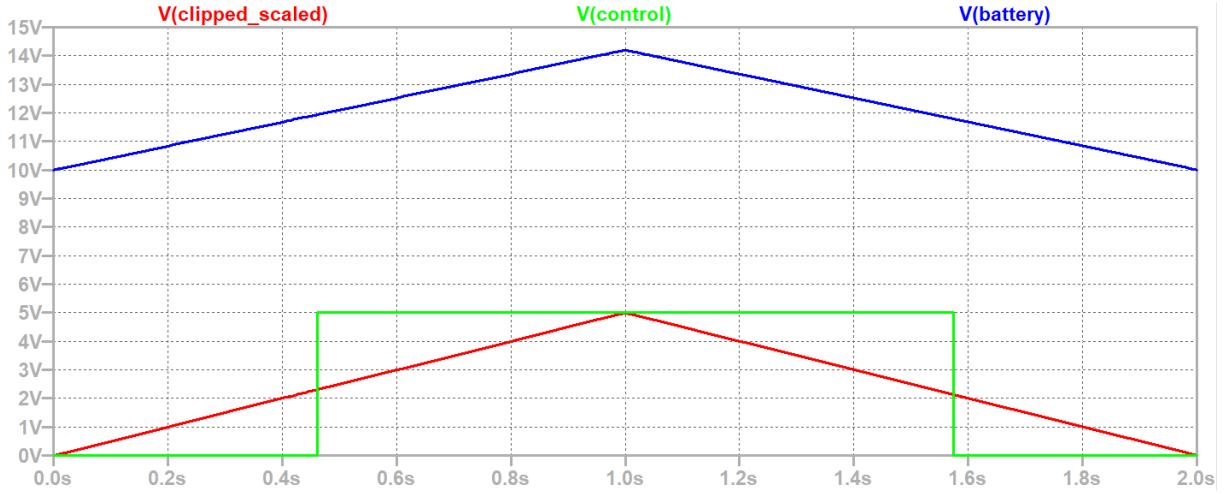
We find from Figure 5.1 that the battery voltage clipper and scaler circuit designed in Section 4.1.2 is working as intended. The input battery voltage, pulsed from 10V to 14.2V, is scaled to the required 0V to 5V. In this case the output voltage is  $V_{clipped\_scaled}$ . The battery voltage after being scaled by a voltage divider is shown as  $V_{ref\_battery}$ .

### Schmitt trigger

From Figure 5.2 we find that the Schmitt trigger designed in Section 4.1.2 is operational. From the graph we see that hysteresis is working as intended: the high-side switch turning OFF at  $V_{clipped\_scaled} = 2.12V$  and turning back ON when  $V_{clipped\_scaled} = 2.30V$ . This implies OFF at the battery voltage after the diode,  $V_{battery} = 11.75V$  and ON at  $V_{battery} = 11.95V$  as we can see in Figure 5.3.



**Figure 5.2:** Output graph from SPICE simulation on schmitt trigger circuit.



**Figure 5.3:** Output graph from SPICE simulation on Schmitt trigger, battery voltage included.

### High-side switch

Figure 5.4 shows operation of the high-side switch designed in Section 4.1.2, working as intended. With the control signal  $V_{control} = 5V$  the switch closes allowing current from the battery to flow to the rest of the circuit, shown as  $V_{load}$ . When the control voltage drops to 0V, current from the battery is interrupted.



**Figure 5.4:** Output graph from SPICE simulation on high-side switch.

### 5.1.3. Voltage rails

This section lists voltage measurements from both the DC-DC boost- and buck converters utilised within the electronic design of the proposed system. The output from these converters feed into critical hardware components, therefore stability is imperative. The output voltage was measured at different input currents and voltages to ensure operation within the required margins for the devices.

## Buck converter

**Table 5.2:** Buck converter voltage measurements

Input voltage [V]	Input current [A]			
	0.45	0.69	1.02	1.91
12	5.153	5.153	5.146	5.147
12.5	5.151	5.150	5.145	5.145
13	5.150	5.148	5.145	5.144

In Table 5.2 we see that irrespective of the input conditions the output voltage of the buck converter remains relatively constant, at around 5.150V. It is important to note the output voltage is on average 0.5V lower than the 5.2V that its datasheet specifies.

## Boost converter

**Table 5.3:** Boost converter voltage measurements

Input voltage [V]	Input current [mA]		
	101	208	315
12	15.102	15.102	15.098
12.5	15.099	5.097	15.098
13	15.097	15.096	15.096

### 5.1.4. Power measurements

**Table 5.4:** Remote station power measurements

Device	Input voltage [V]	Input current [mA]	Power usage [W]
Raspberry Pi 4b	5.151	232	1.20
Ubiquiti Bullet M5	5.102	124	0.632.65
Total system	12.605	439	5.53

The total power usage of the system is 5.53W, which is larger than expected. At the battery's nominal capacity of 7.5Ah we determine that this system will be operational for 17.08 hours on a single charge. In order to create a stand-alone remote station using the actual power capacity of the PV-modules used, three modules would be required.

## 5.2. Network communication

### 5.2.1. Device detection

Once the network is operational, an IP address scan is performed using a utility called NMAP. Through this scan all devices and their IP addresses on the network are found. From the command line snippet shown in Listing 5.1 we see that this scan finds three devices on the network: the base station Raspberry Pi, the remote station Raspberry Pi and finally the Wi-Fi router. This scan indicates the network is operating as intended. It is important that the two radios configured in the system are not detected by the scan. This is due to them being configured as network bridges and therefore only forwarding data they receive.

**Listing 5.1:** IP address scan using NMAP - executed on base station

```
base@raspberrypi:~ $ sudo nmap -sn 192.168.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2022-10-29 01:29 SAST
Nmap scan report for 192.168.0.1
Host is up (0.0020s latency).
MAC Address: 90:9A:4A:E9:5A:54 (Tp-link Technologies)
Nmap scan report for 192.168.0.112
Host is up (0.014s latency).
MAC Address: 00:15:6D:5A:FA:BC (Ubiquiti Networks)
Nmap scan report for 192.168.0.102
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 2.21 seconds
```

### 5.2.2. Remote access and transfer

After the IP addresses of the devices have been obtained in Section 5.2.1 remote access to the remote station can be achieved through network tools such as SSH or NCAT. SSH allows for access to the remote station's shell from the base station using a terminal - see Listing 5.2. For file transfer between the two stations NCAT is used [29]. Therefore, changes to the software on the remote station can be made from the base station.

**Listing 5.2:** Access to remote station shell - executed on base station

```
base@raspberrypi:~ $ ssh remote@192.168.0.112
remote@192.168.0.112's password: ****
```

## 5.3. Remote station range

The remote station was transported 2.41km away from the base station. The system successfully initialised the required TCP socket. Communication between the remote- and base station is successful.

## 5.4. Online capacity

### 5.4.1. Database

The Postgresql database was successfully created and deployed online. A database visualisation client pgAdmin is used to visualise the database tables, see Figure F.1. Measurements were taken to determine the upload speed of individual data entries from the system designed. These upload speeds are plotted in Table 5.5. From these results we can determine the minimum sample speed of the remote station in order to prevent message overflow. Thereby the fastest sampling speed of the system is determined to be 3 seconds. The system was left running at a sampling interval of five minutes for four consecutive days, logging 1152 data entries. No problems were detected.

### 5.4.2. Online dashboard

The online dashboard was successfully deployed and can be visited by anyone. Visit [pieter-skripsie.herokuapp.com](http://pieter-skripsie.herokuapp.com) to view the website. Data visualisation is clear and the website scales according to the visiting device's screen size.

## 5.5. Offline capacity

Following the manual disruption of an active internet connection to the system, the system automatically switches over into offline mode. All data logged through the system is stored locally on the base station. Once an active internet connection is restored to the system, the bulk uploading of data logged offline is uploaded successfully. The uploading time of several bulk batches are listed in Table 5.5.

**Table 5.5:** Online database uploading times. Internet connection - download: 56.99Mbps, upload: 29.71Mbps.

Number of entries	Time [s]
1	1,48
1152	6.85
4608	10.79

# **Chapter 6**

## **Summary and Conclusion**

### **6.1. Summary**

This report seeks to provide an end-to-end solution to remote data logging. A comprehensive modular solution is designed and implemented, centring around a base station that can be implemented at a household or research facility which communicates with a stand-alone solar powered remote station. The solution provided with this report allows for two-way communication between the base- and remote stations, allowing for remote access to sensor data and the ability to adapt the software on the remote station from the comfort of your house. On the remote station, battery longevity is ensured through a custom undervoltage protection circuit while at the base station loadshedding compatibility is implemented. Information logged through the system is globally accessible through a self-made website, accessible on any device.

### **6.2. Application**

Through slight alterations and implementation of the solution designed in this report, farmers, researchers or weather trackers can gather information and control operation of their facilities remotely. This system can provide the required infrastructure to create data-driven models which in turn can increase operation efficiency and profits. The remote data logging system designed is very customisable to the end users' requirements allowing for niche use cases, irrespective of the operators' field of work.

### **6.3. Future work**

Future work for this system includes adapting it to a specific application. An example of this could be operational control on a game-lodge or farm range. The remote station can be adapted to serve as the central communication unit to several wirelessly connected sensor nodes. Some cases will require sensors at different spots within a specific area, this data can be sent to the remote station through short-distance wireless communications whereafter the remote station can forward the data to the base station. Figure H.1 shows

an example of a possible use case. There is room for customisability within the online dashboard. Future dashboard improvements can include integration into a pre-existing website, access control and historic measurement records. Finally, a suitable weatherproof housing will have to be created for the remote station. This housing should allow for the antenna and anemometer to be mounted several meters from the ground on a pole or column.

# Bibliography

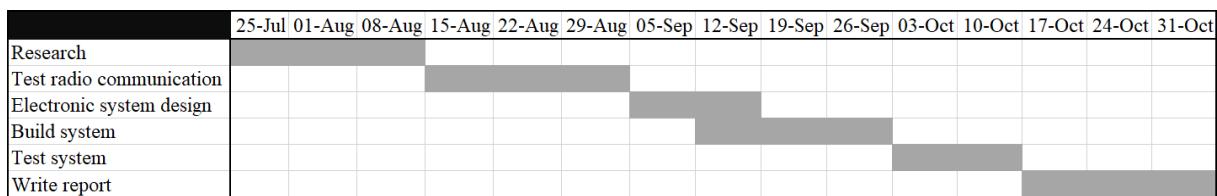
- [1] *High Efficiency Multicrystalline PV Module*, Solarland, 2 2016, pV module.
- [2] *LITHIUM IRON PHOSPHATE (LIFEPO4) BATTERY*, RS PRO, 7 2018, battery specification.
- [3] X. Wang, P. Adelmann, and T. Reindl, “Use of lifepo4 batteries in stand-alone solar system,” *Energy Procedia*, vol. 25, p. 135–140, 2012.
- [4] J. Geerling, “Raspberry pi - conserve power and reduce draw,” Nov 2015. [Online]. Available: <https://www.jeffgeerling.com/blogs/jeff-geerling/raspberry-pi-zero-conserve-energy>
- [5] OpenIndustry, “Iot transforming the future of agriculture: Iot solutions world congress,” Jul 2019. [Online]. Available: <https://www.iotsworldcongress.com/iot-transforming-the-future-of-agriculture/>
- [6] V. Hyseni, “Why is data protection important?” Nov 2021. [Online]. Available: <https://pecb.com/article/why-is-data-protection-important>
- [7] D. Heroku, “Heroku documentation,” Jul 2022. [Online]. Available: <https://devcenter.heroku.com/categories/reference>
- [8] D. Nettikadan and S. Raj, “International journal of applied science and engineering research,” *Smart Community Monitoring System using Thingspeak IoT Platform*, vol. 13, no. 17, p. 13403–13405, 2018.
- [9] D. Instruments, “What is a uv sensor?” 2018. [Online]. Available: <https://www.davisinstruments.com/pages/what-is-a-uv-sensor>
- [10] *Optical Sensor Product Data Sheet*, Lite On, 6 2015, uV sensor.
- [11] A. Lavaaa, “Temperature and humidity sensors: Ultimate guide,” Oct 2021. [Online]. Available: <https://www.linchip.com/blog/temperature-and-humidity-sensors-an-ultimate/>
- [12] I. Inhand, “What is antenna gain,” Nov 2019. [Online]. Available: <https://www.inhandnetworks.com/article/277.html>

- [13] GeoTherm, “Polycrystalline solar cells vs monocrystalline: Which is better?” [Online]. Available: <https://www.altestore.com/diy-solar-resources/electrical-characteristics-of-solar-panels-pv-modules/>
- [14] *RELiON LITHIUM IRON PHOSPHATE (LiFePO<sub>4</sub>) BATTERIES*, reli3ion, 1 2020, sCC operation.
- [15] B. Hirschman, “Power over ethernet ieee 802.3af,” 2011.
- [16] M. Contributors, “An overview of http,” Sep 2022. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>
- [17] V. Harvey, “What is a network bridge?” Oct 2022. [Online]. Available: <https://www.easytechjunkie.com/what-is-a-network-bridge.htm>
- [18] Oracle, “What is a database?” [Online]. Available: <https://www.oracle.com/za/database/what-is-database/>
- [19] Neo4j, “What is a graph database? - developer guides.” [Online]. Available: <https://neo4j.com/developer/graph-database/>
- [20] G. Cloud, “What is a relational database (rdbms)?” [Online]. Available: <https://cloud.google.com/learn/what-is-a-relational-database>
- [21] N. Greene, “What is a web framework?” Nov 2017. [Online]. Available: <https://evolve.ie/q-and-a/what-is-a-web-framework/>
- [22] *Raspberry Pi 4 Model B - Datasheet*, Raspberry Pi (Trading) Ltd, 1 2019, raspberry Pi 4b computer.
- [23] *Zero-Variable Wireless Infrastructure Deployment - Datasheet*, Ubiquiti, 1 2011, bullet M5 Radio.
- [24] *PYBJ6 DC-DC CONVERTER - Datasheet*, CUI inc., 7 2021, boost converter.
- [25] *Digital relative humidity and temperature sensor RHT03*, MaxDetect, 2 2014, temperature and humidity sensor.
- [26] L. Solar basics, “Solarbasics,” Apr 2020. [Online]. Available: <https://www.ecotrades.co.za/solar-info/>
- [27] L. Ada, “Dht humidity sensing on raspberry pi or beaglebone black with gdocs logging,” Oct 2022. [Online]. Available: <https://learn.adafruit.com/dht-humidity-sensing-on-raspberry-pi-with-gdocs-logging/python-setup>
- [28] T. P. G. D. Group, *Documentation PostgreSQL 10.3*, 2018.

[29] *File Transfer*, nmap, 1 2020.

# Appendix A

## Project Planning Schedule



**Figure A.1:** Estimated timeline, Gantt chart.

Date	Work
25-Jul-22	Research
01-Aug-22	Research
08-Aug-22	Research
15-Aug-22	Test radio communication
22-Aug-22	Test radio communication
29-Aug-22	Test radio communication
05-Sep-22	Create website
12-Sep-22	Electronic system design
19-Sep-22	Electronic system design
26-Sep-22	Build system
03-Oct-22	Build system
10-Oct-22	Test system
17-Oct-22	Write report
24-Oct-22	Write report
31-Oct-22	Write report

**Figure A.2:** Estimated timeline.

# Appendix B

## Outcome Compliance

**Table B.1:** Outcome compliance table

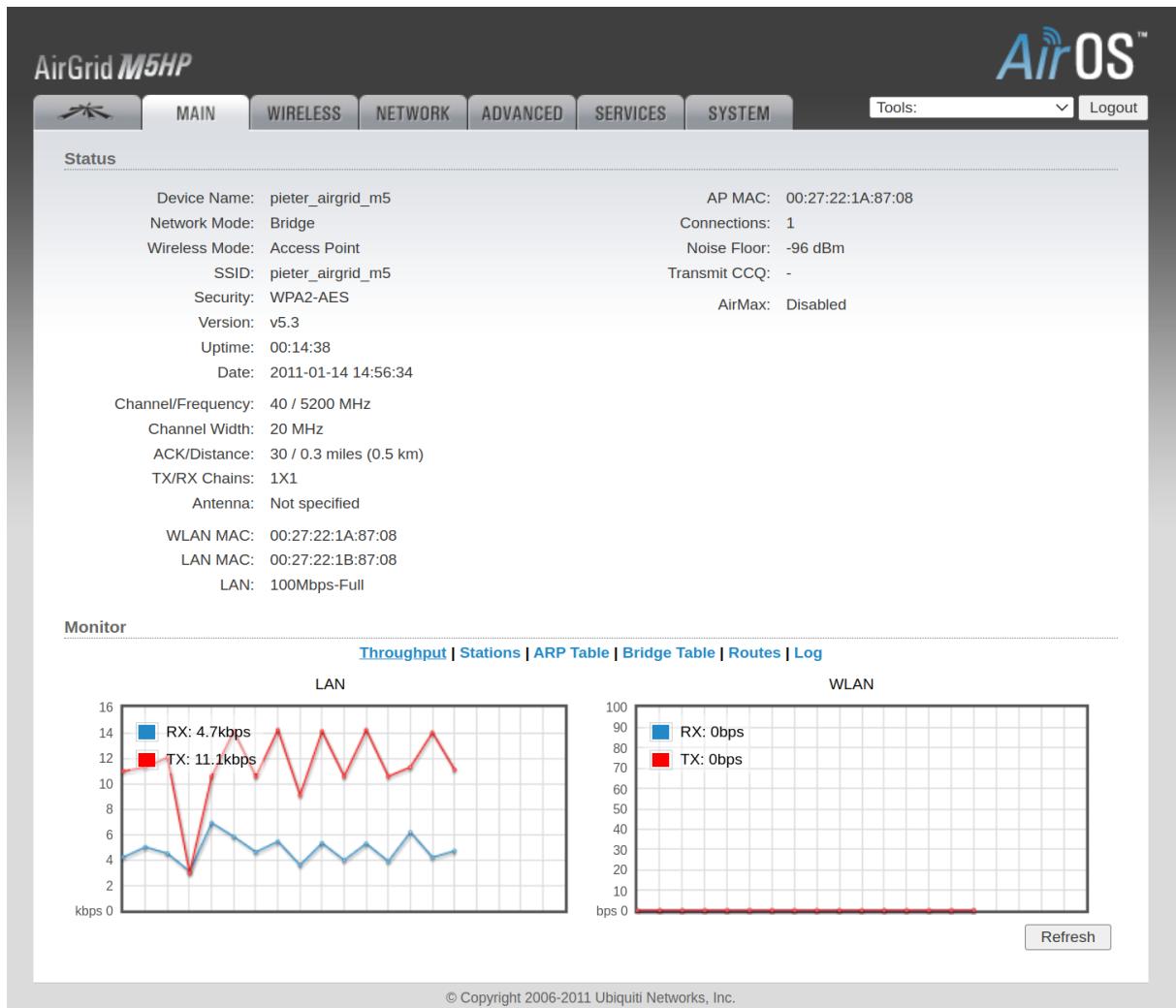
Outcome	Description of compliance
<b>GA 1.</b> Problem solving	The problem presented with this thesis was stated very broadly – wirelessly transmit data from a remote location to a base station. To tackle this problem various subsystems will have to be designed and implemented to present an elegant and working solution. The final solution to this problem required a unique combination of various electronic and electrical engineering fields. Present solutions to this problem are currently in the hands of private companies and not available to the public. This allows for a truly independent design and implementation for this thesis.
<b>GA 2.</b> Application of scientific and engineering knowledge	This thesis required a solution that combines both electronic design, low-level software development, high level database and web design, networking and physical system design for the end-product. Integration of these fields of engineering will required a non-obvious, independent approach that implemented techniques and fields of studies learnt throughout my time at university. This included extended use of mathematics, engineering design and realisable conceptualisation.
<b>GA 3.</b> Engineering Design	The problem presented with this thesis required a complex engineering solution which allowed for creativity within its design. Many of the techniques such as the software development both low and high level as well as the electronic design will be invaluable skills when entering the work force. I specialised in informatics in my final year, with many of these skills having been applicable to the development of my final design.
<b>GA 4.</b> Investigations, experiments and data analysis	Various investigative and analytical tests have been conducted throughout the thesis. These include finding the real-world properties of the PV module, the true range and directional nuance of the two radios, various database comparisons and an in-depth analysis on different solar charge controller options.

**Table B.2:** Outcome compliance table

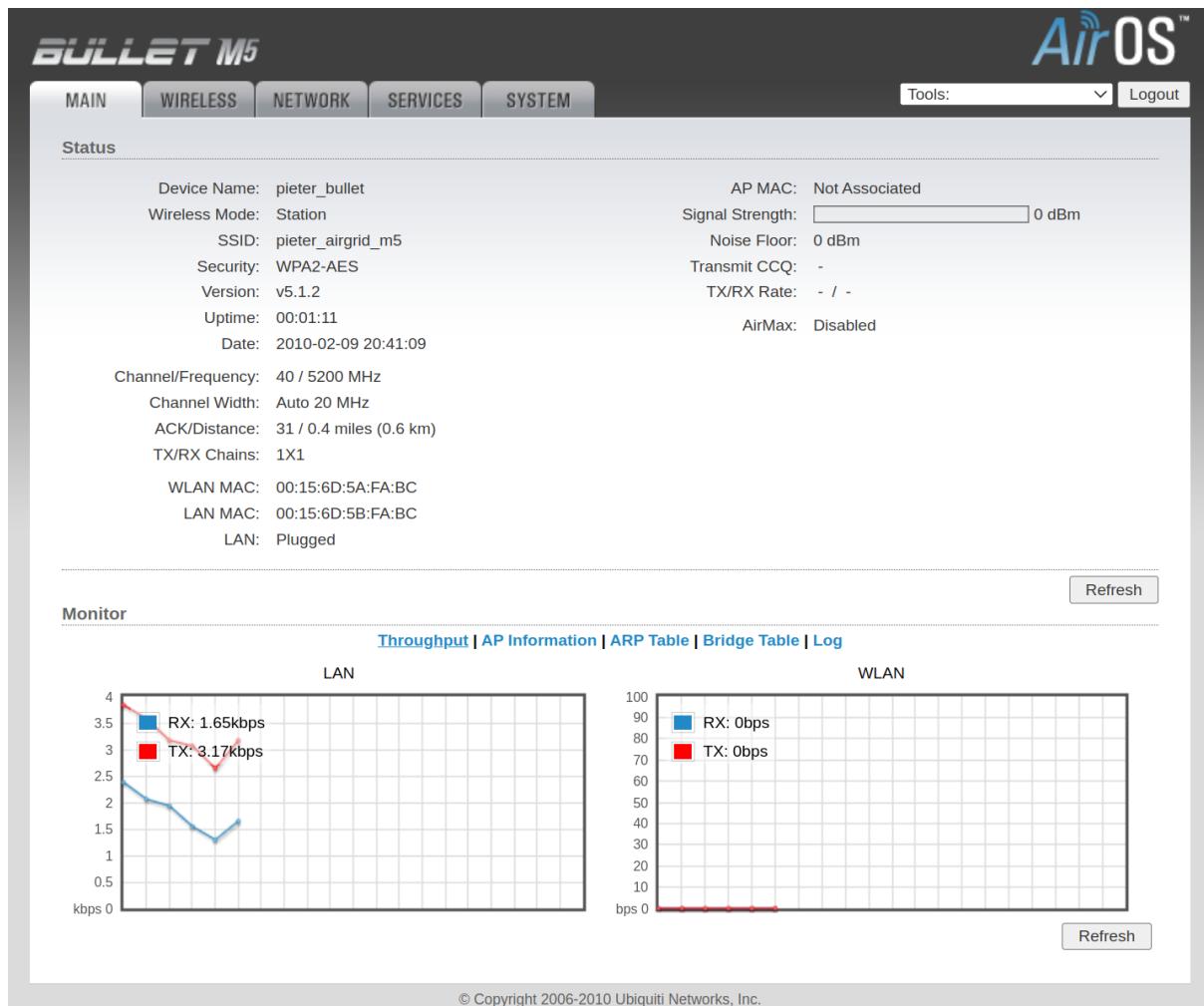
Outcome	Description of compliance
<b>GA 5.</b> Engineering methods, skills and tools, including Information Technology	This thesis included engineering design principles and analysis techniques. Components from many different manufacturers were used which includes the utilisation of separate datasheets and design recommendations. On the software side, an understanding of low- and high-level development was needed. Industry standard tools such as LT-Spice, Relational databases and Heroku have been used. This thesis required the utilisation of Linux, Postgres, Flask and many terminal based command line tools. For the electronic design aspect of the thesis standard lab equipment such as oscilloscopes and bench power supplies were used for testing, this demonstrates the prototyping abilities required to finish the thesis.
<b>GA 6.</b> Professional and technical communication	Extensive care was taken to format this report in an easy to read and intuitive way, this relates to the phrasing of sentences and explanations of equations. Graphs and figures within the report were self-created using vector graphics manipulation software.
<b>GA 8.</b> Individual work	This project was completed end-to-end by me. All resources which I required to design this system are cited in the bibliography.
<b>GA 9.</b> Independent learning ability	The real-world application of the system was considered to determine the system requirements. All testing and design were performed individually, and no advisor-mandated schedule requirements were made. No external parties were consulted for help throughout the development of the solution.

# Appendix C

## Network configuration - AirOS



**Figure C.1:** AirOS configuration of AirGrid M5. Note that WLAN is showing no connection due to the device not being operational when configured on a separate Windows PC.



**Figure C.2:** AirOS configuration of Bullet M5. Note that WLAN is showing no connection due to the device not being operational when configured on a separate Windows PC.

# Appendix D

## Components

**Table D.1:** Table indicating components used.

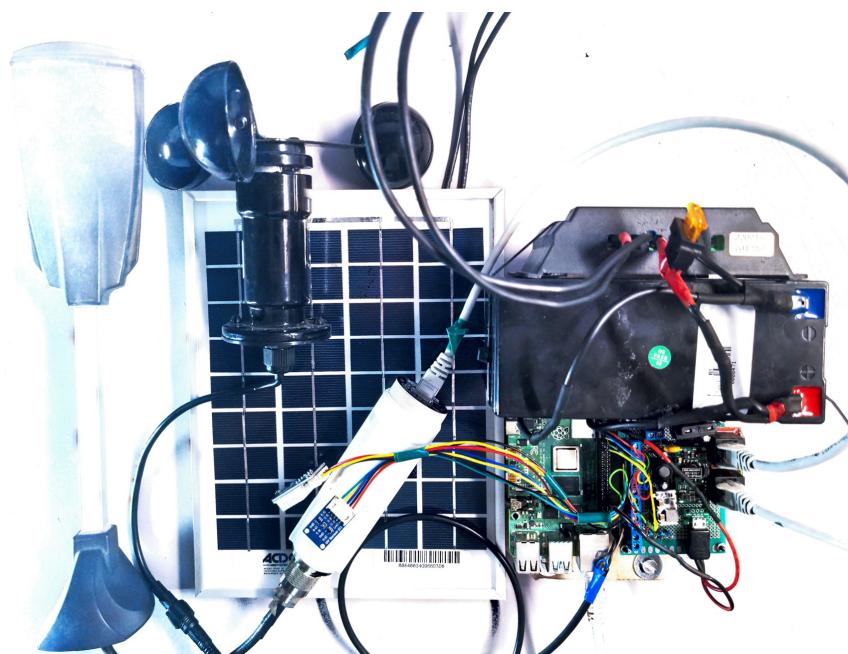
Component type	Product number or value	Number of units
LiFePo4 battery	RSAML9131	1
SCC	STCC10	1
Radio and antenna	Ubiquiti Bullet M5	1
Radio and antenna	Ubiquiti AirGrid M5	1
Computer	Raspberry Pi 4b	2
Buck converter	CKCY BUCK03	1
Boost converter	PYBJ6-D12-S15	1
UV sensor	LTR-390	1
Anemometer	-	1
Temperature and humidity sensor	DHT22	1
RJ45 female connector	-	2
RJ45 breakout board	-	2
4cm x 6cm Protoboard	-	1
Fuse	Blade 5A	1
Fuse	Blade 2A	1
Fuse	Resettable 0.5A	1
Screw terminal	-	7
Operational amplifier	LTC6079	6
Diode	SB560	2
Female header	20 port	2
Micro-USB connector	-	1
Micro-USB port	-	1
Ethernet cable	CAT5 or higher	4
Capacitor	$4.7\mu F$	2
Capacitor	$10\mu F$	1
Inductor	$4.7\mu H$	1

**Table D.2:** Table indicating resistors used.

Component type	Product number or value	Number of units
Resistor	$1k\Omega$	1
Resistor	$5k\Omega$	1
Resistor	$10k\Omega$	2
Resistor	$20k\Omega$	3
Resistor	$47k\Omega$	1
Resistor	$82k\Omega$	1
Resistor	$100k\Omega$	6
Resistor	$220k\Omega$	1
Resistor	$330k\Omega$	2
Resistor	$560k\Omega$	1

# Appendix E

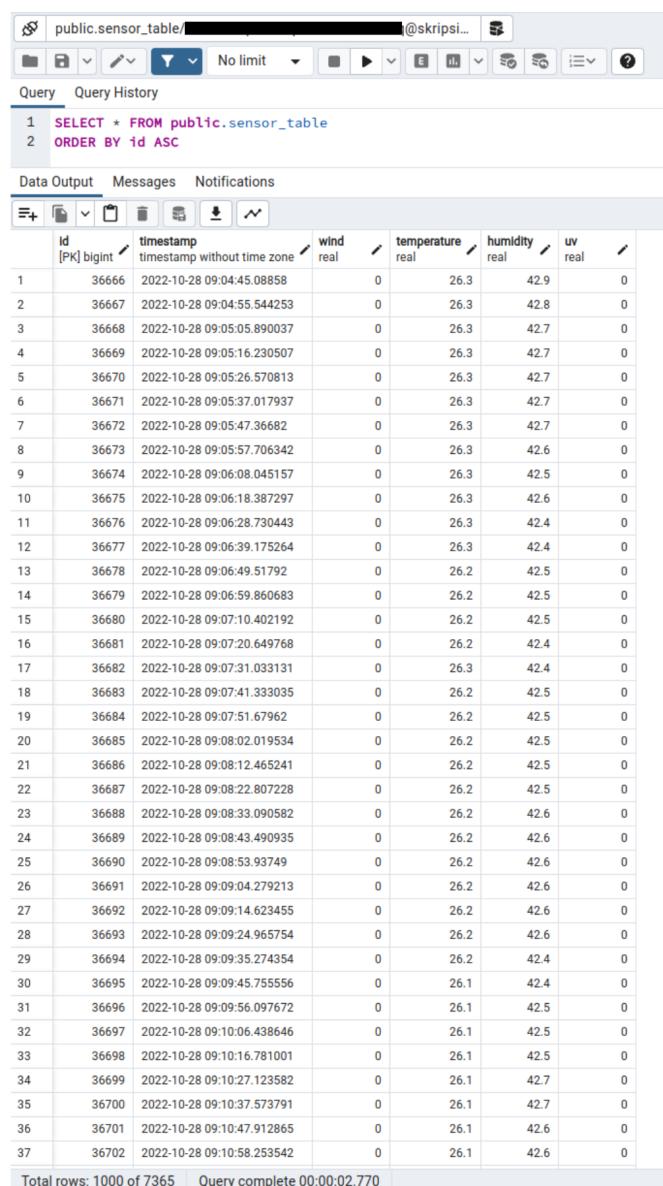
## Hardware



**Figure E.1:** Photograph of all hardware on the remote station.

# Appendix F

## Database



The screenshot shows the pgAdmin client interface with a query results table. The query executed is:

```
1 SELECT * FROM public.sensor_table
2 ORDER BY id ASC
```

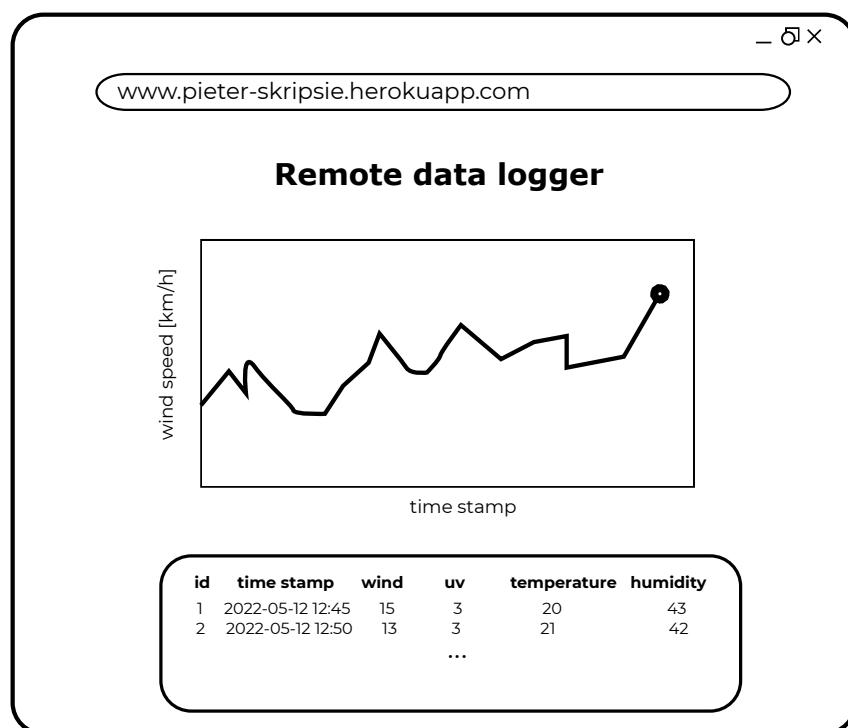
The table has the following columns: Id [PK] bigint, timestamp timestamp without time zone, wind real, temperature real, humidity real, and uv real. The data shows 1000 rows of sensor data from October 28, 2022.

	Id [PK] bigint	timestamp	wind	temperature	humidity	uv
1	36666	2022-10-28 09:04:45.08858	0	26.3	42.9	0
2	36667	2022-10-28 09:04:55.544253	0	26.3	42.8	0
3	36668	2022-10-28 09:05:05.890037	0	26.3	42.7	0
4	36669	2022-10-28 09:05:16.230507	0	26.3	42.7	0
5	36670	2022-10-28 09:05:26.570813	0	26.3	42.7	0
6	36671	2022-10-28 09:05:37.017937	0	26.3	42.7	0
7	36672	2022-10-28 09:05:47.36682	0	26.3	42.7	0
8	36673	2022-10-28 09:05:57.706342	0	26.3	42.6	0
9	36674	2022-10-28 09:06:08.045157	0	26.3	42.5	0
10	36675	2022-10-28 09:06:18.387297	0	26.3	42.6	0
11	36676	2022-10-28 09:06:28.730443	0	26.3	42.4	0
12	36677	2022-10-28 09:06:39.175264	0	26.3	42.4	0
13	36678	2022-10-28 09:06:49.51792	0	26.2	42.5	0
14	36679	2022-10-28 09:06:59.860683	0	26.2	42.5	0
15	36680	2022-10-28 09:07:10.402192	0	26.2	42.5	0
16	36681	2022-10-28 09:07:20.649768	0	26.2	42.4	0
17	36682	2022-10-28 09:07:31.033131	0	26.3	42.4	0
18	36683	2022-10-28 09:07:41.333035	0	26.2	42.5	0
19	36684	2022-10-28 09:07:51.67962	0	26.2	42.5	0
20	36685	2022-10-28 09:08:02.019534	0	26.2	42.5	0
21	36686	2022-10-28 09:08:12.465241	0	26.2	42.5	0
22	36687	2022-10-28 09:08:22.807228	0	26.2	42.5	0
23	36688	2022-10-28 09:08:33.090582	0	26.2	42.6	0
24	36689	2022-10-28 09:08:43.490935	0	26.2	42.6	0
25	36690	2022-10-28 09:08:53.93749	0	26.2	42.6	0
26	36691	2022-10-28 09:09:04.279213	0	26.2	42.6	0
27	36692	2022-10-28 09:09:14.623455	0	26.2	42.6	0
28	36693	2022-10-28 09:09:24.965754	0	26.2	42.6	0
29	36694	2022-10-28 09:09:35.274354	0	26.2	42.4	0
30	36695	2022-10-28 09:09:45.755556	0	26.1	42.4	0
31	36696	2022-10-28 09:09:56.097672	0	26.1	42.5	0
32	36697	2022-10-28 09:10:06.438646	0	26.1	42.5	0
33	36698	2022-10-28 09:10:16.781001	0	26.1	42.5	0
34	36699	2022-10-28 09:10:27.123582	0	26.1	42.7	0
35	36700	2022-10-28 09:10:37.573791	0	26.1	42.7	0
36	36701	2022-10-28 09:10:47.912865	0	26.1	42.6	0
37	36702	2022-10-28 09:10:58.253542	0	26.1	42.6	0

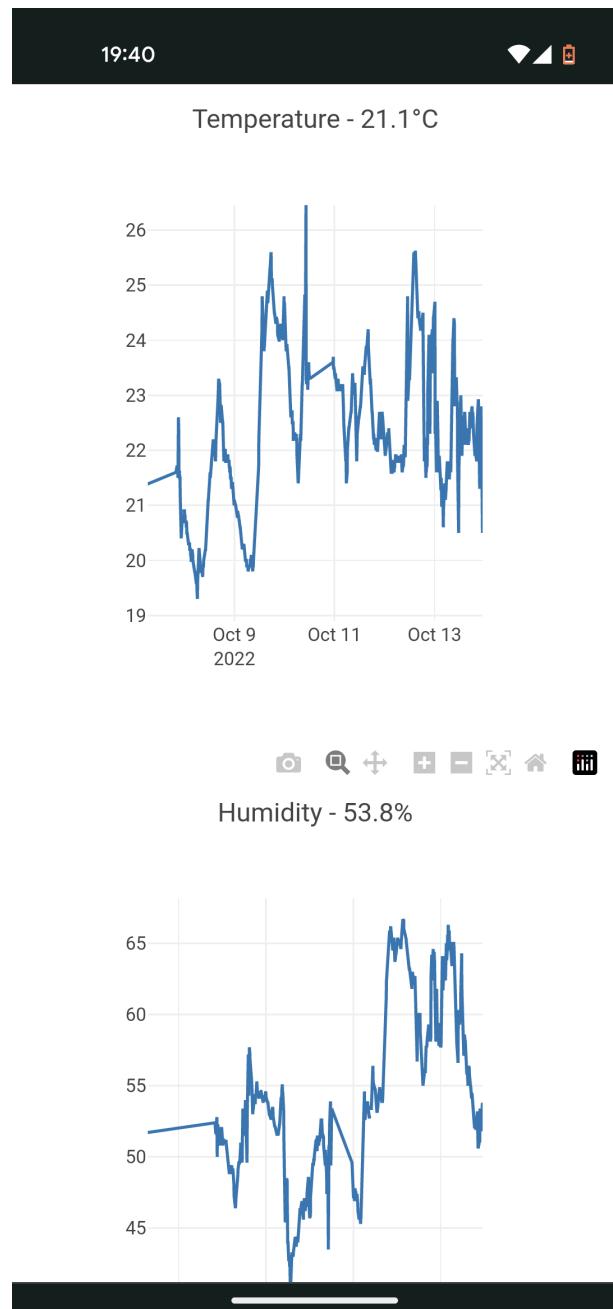
**Figure F.1:** Online database visualisation through pgAdmin client.

# Appendix G

## Dashboard



**Figure G.1:** Web application interface mock-up. Interface includes graphs for all sensor metrics and a data table to display all records stored within the database.



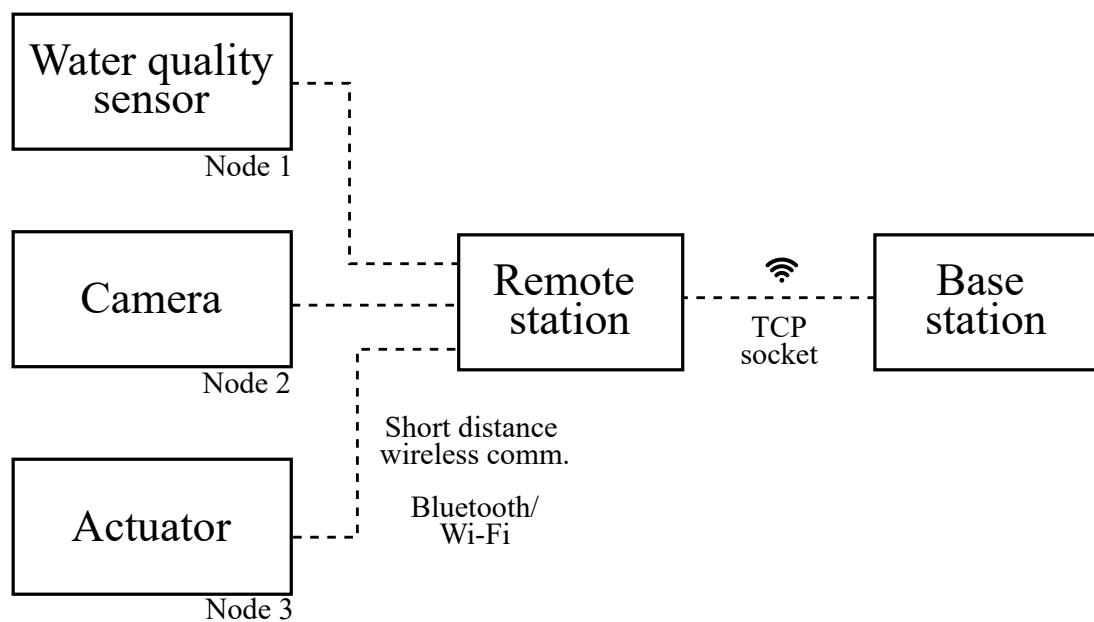
**Figure G.2:** Screen shot of sensor dashboard from a cellphone. The screenshot shows that the website is accessible and scales correctly to the specific device.



**Figure G.3:** Screen shot of sensor dashboard from a desktop. The screenshot shows that the website is accessible and scales correctly to the specific device.

# Appendix H

## Future work



**Figure H.1:** Wireless nodal extension of remote station - future work.