BACHELOR THESIS

Designing a Low-Cost Autonomous Pyranometer

By: Peter van der Burgt - s1838954

Supervisor: Richard Bults
Critical Observer: Hans Scholten

A Creative Technology Graduation Project

Date: 17-07-2020

Abstract

To understand the Urban Heat Island effect and see the impact of the different urban areas around Enschede, a network of low-cost autonomous weather stations is under development.
To further develop this network, a low-cost autonomous pyranometer had to be made, to get further insight on how to accurately measure solar irradiance with low-cost sensors. These questions led to this research with the main research question "*How to Develop a Low-Cost Autonomous Pyranometer?*".

A low-cost autonomous pyranometer was made by designing a low-cost pyranometer that is made autonomous by interfacing it with a low-cost autonomous weather station. The sensor used off-the-shelf modules, a microcontroller and a Digital to Analog Converter to generate a voltage to be read by the low-cost autonomous weather station. The conversion of the output of the light sensors to a voltage that is outputted is done with the help of a calibration function made by a Multiple Linear Regression model. The overall sensor went through four different iterations to get to the end prototype.

The low-cost pyranometer was evaluated with the help of a reference pyranometer, the Davis Instruments Solar Radiation Sensor. The testing setup measured both outputs simultaneously while these pyranometers were outside in the sun. The low-cost pyranometer fulfilled the main requirements, which included accuracy and costs.

A good first step was made in making a low-cost autonomous pyranometer. It was shown that a low-cost pyranometer can be made with low-cost components and the help of machine learning techniques such as Multiple Linear Regression. The exact accuracy was difficult to determine however, due to the inaccuracy of the reference pyranometer.

For further iterations of this low-cost pyranometer, the main points to be tackled are the integration of the sensor with the low-cost autonomous weather station and the usage of a high-end, more accurate reference pyranometer, which will lead to an increased and more certain accuracy.

Acknowledgement

In this document, half a year of work is described by me, however, a lot of people helped get this research as far as it is now. I would like to extend my gratitude to these people here.

First, I would like to thank my supervisor Richard Bults and my critical observer Hans Scholten. They motivated me to look further into the research, as well as find and reach the boundaries of my capabilities during this Graduation project, ultimately ending in a final version I am proud of.

Second, I would like to thank my stakeholders, Wim Timmermans of the ITC department of the University of Twente, as well as Rik Meijer, the contact at the municipality of Enschede. Both gave valuable insights into the project, and the execution of the overall plans as the municipality intends it to be.

Third, I want to show appreciation to Jan-Paul Konijn. He worked simultaneously with me on the design of a low-cost autonomous weather station. We shared ideas, as well as spending a lot of time together brainstorming about both of our projects.

Last, I would like to thank all friends and family who kept my motivation high and inspired me, especially during this time where it was only possible to work from home.

Table of Contents

List of Figures

List of Tables

List of Equations

1. Introduction

In this first chapter of this report, background information is given to understand why this project came to be. Furthermore, the challenges and objectives are explored to properly formulate a Research Question and sub-questions to help answer the research question.

1.1. Background

Global warming has been an increasingly growing discussion amongst every layer of the population. The municipality of Enschede has its concerns regarding global warming and its effects on the urban environment. These concerns mostly envelop the development of so-called Urban Heat Islands (UHI) in the city, see Figure 1 [1]. These Urban Heat Islands are areas of higher temperature because buildings are close together, roads and buildings absorbing heat during a sunny day and radiate heat during the night, and wind not being able to flow as easily as it would have if there was no infrastructure [2], [3].

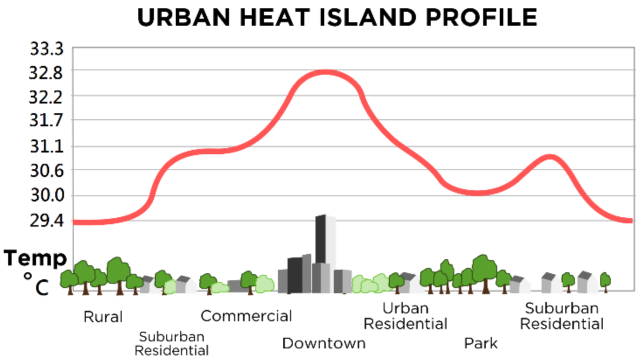


Figure 1: Urban Heat Island – image based on data from NOAA [1].

The effect that a UHI has on the environment and life is substantial. The effects that the municipality of Enschede is most concerned about is so-called heat stress; i.e. an effect of the human body not having the ability to cool down and getting overheated. Increased temperatures in the city can lead to all kinds of heat-related problems within the body [4], [2].

To monitor the influence of the weather and specifically solar irradiance, the municipality is working together with the University of Twente to create a grid of weather stations placed in the city. The creation of this grid of Wireless Sensor Nodes (WSN) is the goal of the research project WHEGS ("Wat Heet Eanske Greune Stad!"). Each weather station contains multiple sensors (temperature, relative humidity, solar irradiance, and wind speed) to monitor weather conditions. Sensor data will be used by the municipality to mitigate the effect of the UHI; for example, by adjusting building regulations to make it as pleasant as possible for individuals.

However, the sensors and other sub-systems of commercially available weather systems currently used are expensive. These include pyranometers, which are sensors specifically designed to measure the solar irradiance at a given location. There are high-end versions such as the ones from Kipp & Zonen [5] and more mid-range versions like the Solar Radiation Sensor and UV Sensor from Davis Instruments [6]. These sensors all measure the amount of solar energy that falls on a square meter per second. But, to do so over a different spectrum of wavelength with different resolution and accuracy.

1.2. Challenges and Objectives

The goal of this Graduation Project is to focus on the development of a low-cost, autonomous pyranometer, a system that can measure solar irradiance in isolated or remote locations. A pyranometer is an instrument that measures solar irradiance. The prospect of why such a system is interesting is to see if with relatively cheap materials a qualitatively good system can be made.

The main challenge of this Graduation Project is to make the system as accurate as the reference sensor system, the Davis Vantage Pro with the Davis Solar Radiation Sensor while using low-cost components and sub-systems. As mentioned in section 1.1, there are already multiple existing solar irradiance sensor systems that measure solar irradiance. However, these are expensive. Besides the sensor system itself, there is also the main controller, which, in the case of the reference system [7] and other cases, is a general-purpose controller to which you can attach multiple sensor systems that all measure different variables. This means that there are multiple angles to approach the aspect of making the system low cost and thus making a more specialized system for measuring just solar irradiance. When such a low-cost system is made, it can be used in a grid to get good coverage of a certain area, in this case, the city of Enschede, without it getting too expensive to fund.

The secondary challenge is for the entire system to be taking these measurements autonomous. This means that the system can measure the solar irradiance for a longer period without being dependent on wired infrastructure e.g. power grid, wired data communication or any supervision. This means that it should have the means to process the measurements and communicate this data wirelessly to a central data point. Furthermore, it should generate its power and use this to power its sub-systems. Besides the fact that it should function on itself, it should also be reliable to function for an extended amount of time with limited to no maintenance or any form of human involvement being necessary.

Furthermore, systems like the Davis system are often closed source, meaning that it is not possible to get data from the system or to extend such a system with self-developed sensor systems. Therefore, it would be better if the made system is open source so people can make it themselves or use it for further research.

Part of this Graduation Project was executed together with Jan-Paul Konijn. The cause of this lies in the overlap regarding the integration of the autonomous expect of both graduation projects. To make sure that the work was not done twice, the supervisors advised cooperation on the execution of the autonomous aspect of both Graduation Projects. This includes wireless communication capabilities of the systems, communication protocol, as well as the battery management system and the time and location data.

1.3. Research Questions

From the challenges and objectives, the main research question can be formulated as follows:

How to Develop a Low-Cost Autonomous Pyranometer?

To correctly answer the main question, sub-questions have been formed.
First, it is important to know what possible ways there are to measure solar irradiance. There are sure to be different ways, but what leads to a good quality of measurements, and what are good ways to get sufficient quality while using low-cost materials and components.

What methods exist for measuring Solar Irradiance by means of a Pyranometer?

When knowing the best way to measure solar irradiance, it is important to know what kinds of sensor systems can be employed to use this technique, thus leaving us with the question:

What type of sensors match these methods for measuring Solar Irradiance?

Last, when sensors have been employed to measure solar irradiance, there must be a way to evaluate the employed sensors. Since the main research questions mainly concern keeping the system low-cost while still being as good as the reference system, the best way to evaluate would be based on costs and the quality of measurements, thus:

How to evaluate a made system, based on costs and accuracy?

1.4. Report Outline

A quick overview of the chapters is given here. The second chapter will describe background research that is performed to better understand the overall subject of this graduation project. This will be done by performing a literature study, as well as looking at State-of-the-Art solutions.

The third chapter will describe the different methods and techniques used in this project. This chapter will discuss methods of how the project will be executed, but also what type of brainstorming methods, interview types and what kinds of software was used to aid in the making of the prototype.

The fourth chapter discusses the ideation phase as set by the Creative Technology Design Process (CTDP) [8]. It discusses ideas generated by the developer as well as preliminary requirements set by Stakeholders or implicated by the usage of the System.

The fifth chapter tackles the specification phase of the CTDP. This is the phase where the ideas generated in the ideation phase. The requirements are also finalised, and flowcharts are made on how the system should function. Furthermore, the different aspects of the system are finalised and are made ready for the next realisation phase.

The sixth chapter describes the realisation phase, where the different iterations of the prototype are described. Not only are connections described but also the different ways these were tested and how these are improved with each iteration.

The seventh chapter discusses the last step of the CTDP, the evaluation phase. Here the different sub-systems of the project are evaluated, including the made sensor, and the autonomous systems. Next to that, the requirements are also evaluated.

The eighth and last chapter is the conclusion. Here the findings are discussed, as well as improvement points. Besides this, future work is presented to be worked on after this project is done. These improvements are things that could increase the overall performance of the project.

2. Background Research

The State of the Art aims to provide the reader with some knowledge about the different aspects that will be discussed further in this report.
To be easily understood, this chapter is divided into a couple of sub-chapters.
The first sub-chapter is about measuring solar irradiance, what are common ways to do this, and how should one go about measuring it.
The second sub-chapter elaborates on the system that will be made. This chapter is further divided into five smaller chapters: a chapter about the sensor part, the data processing, sensor fusion, wireless communication, and the Power Management System.
Lastly, this chapter discusses how the entire system can be calibrated and evaluated.

2.1. Literature Study

This part of the background research chapter talks about the theory of pyranometers and solar irradiance. Furthermore, the difference between solar irradiance and solar radiation will be discussed, how a pyranometer measures solar irradiance.
Furthermore, the sensor system that will be made has four distinct parts. These will also be discussed.
Besides these system parts, there is also a chapter on how one calibrates and evaluates a made system.

2.1.1. Measuring Solar Irradiance

The goal of this research is to make a solar irradiance measuring sensor. For this, some knowledge should be gathered on what exactly solar irradiance is and what types of solar irradiance exist. Secondly, it is good to discover how one goes about measuring solar irradiance.

2.1.1.1. Defining Solar Radiation and Solar Irradiance

The most generic definition for solar radiation is: “Energy radiated from the sun in the form of electromagnetic waves, including visible and ultraviolet light and infrared radiation” [9]. In short, it means that all electromagnetic radiation that is sent out by the sun is referred to as solar radiation.

However, there is another word for solar radiation which means something similar: solar irradiance. solar irradiance, defined as “The amount of electromagnetic energy incident on a surface per unit time per unit area.” [10]. This definition refers to electromagnetic energy, which is the solar radiation, that falls on a surface per a certain amount of time.

As you see, there is a subtle difference. This mostly consists of the fact that solar irradiance includes solar radiation, but not the other way around. Another definition for solar irradiance is: “The amount of solar radiation that falls on a surface per time”. Still, this difference is extremely important to take note of, as to not confuse the two terms in this report.

2.1.1.2. Solar Irradiance Classification

Solar irradiance is mostly classified in two ways. This classification can be important as it could be covering the same range while using other names for it.

One classification, as mentioned by Bilbao *et al*. and Rösemann [11], [12] is the division in ultraviolet light, visible light, and infrared light. The UV has wavelengths from 0.2 - 0.4 µm, the visible spectrum ranges within 0.39 - 0.77 µm and the Infrared portion is divided into two parts, the near IR light, 0.77 - 25 µm and the far IR light, 25 - 1000 µm. This division can be used in this project to narrow the search of different types of sensors, as these sensors are mostly distributed under this division.

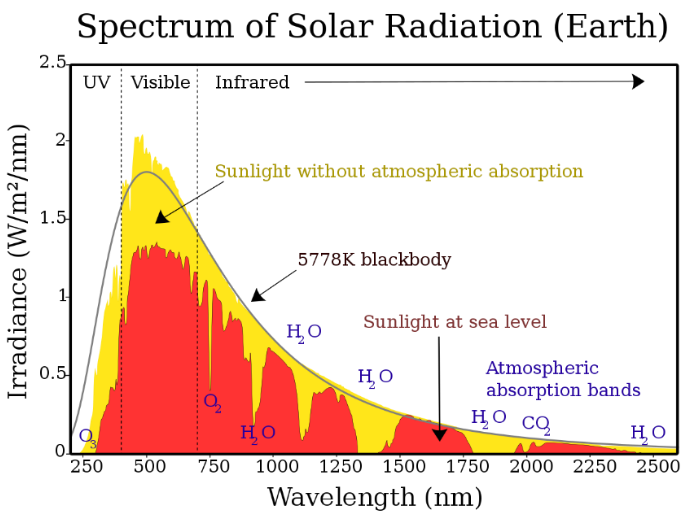


Figure 2: Spectrum of solar radiation above the atmosphere of the earth and at sea level. – image based on data from the American Society for Testing and Materials (ASTM) [14].

A second classification, as mentioned by Solecki *et al.* [2], Bilbao *et al.* [11] and Rösemann [10] and means that solar irradiance is divided into the shortwave and longwave electromagnetic radiation. The shortwave electromagnetic radiation is the most measured irradiance type for weather applications and is set from 300 nm to 3000 nm [2], [13]. The bigger part of this shortwave electromagnetic radiation can be seen in Figure 2 [14]. This second division is less needed than the first one. However, since the range of spectral interest covers the shortwave radiation, it is good to keep this definition as well.

2.1.1.3. Types of Solar Irradiance to measure

Besides the fact that there are divisions based on electromagnetic wavelength in the Solar Irradiance Spectrum, there are also three main variables that you can measure when measuring solar irradiance on earth [13]. These are Global Horizontal Irradiance (GHI), Diffuse Horizontal Irradiance (DHI) and Direct Normal Irradiance (DNI). Djen *et al.* [15] note that GHI is the most commonly measured variable regarding ground-based meteorological stations. Abreu *et al.* [16] state that the DNI plays an important role in the urban heat island effect. Since the GHI also entails the DNI, as GHI is the sum of DHI and DNI, it could be argued that measuring the GHI would be enough for this project as it also encapsulates the DNI. The different types of irradiances can be seen in Figure 3 [17].

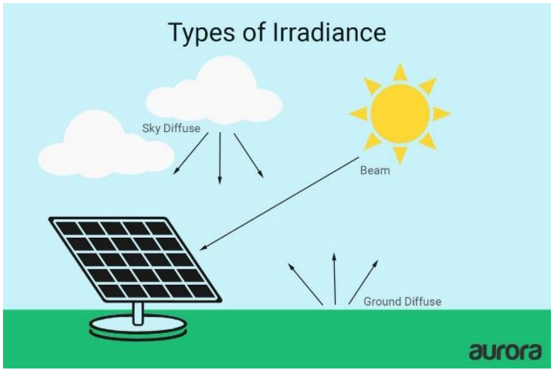


Figure 3: Types of Irradiance. - DNI and DHI are depicted in this image, GHI is the sum of both DNI and DHI. This image is taken from Aurora Solar [17].

2.1.1.4. What Spectral range should be measured?

According to Rösemann [12], the meteorologically significant spectral range is from 300 nm to 3000 nm, which is also known as the shortwave electromagnetic radiation.

This is further supported by Mecherikunnel and Richmond [18], who say that the spectral range from 0.27 µm to 2.6 µm contains 96% of the sun’s energy.

Therefore, it can be said that the spectral range from 300 nm to 3000 nm should be measured.

2.1.1.5. Measuring Solar Irradiance with a Pyranometer

A common instrument used in measuring solar irradiance is the pyranometer.
The most generic definition for a pyranometer is "an instrument that measures solar radiation" [19]–[21]. However, what these sources mostly refer to is the solar irradiance instead of solar radiation.
There are two common ways to measure solar irradiance: utilizing a thermopile or utilizing an optic device [22], [23]. The thermopile is a sensor that converts heat energy into electrical energy. An optic device is a type of sensor that converts electromagnetic waves (light) into electrical energy, examples of this are a solar panel or a photodiode.

2.1.1.6. Solar Irradiance Measurement Protocols

There are some general measurement protocols and operational requirements for the measuring of solar irradiance. These are set in the "Handboek Waarnemingen" [24] by Dutch "Koninklijk Nederlands Meteorologisch Instituut" (KNMI), who got their references from the CIMO-guide [13] by the World Meteorological Organization (WMO).
One of these requirements include rules like a lower limit and an upper limit of solar irradiance that the pyranometer should be able to measure. These values are set at 0 W/m2 and 2000 W/m2 respectively. The measurement resolution should be 1 W/m^2.

Furthermore, the measurements themselves also have a set way on how and what to measure. According to the KNMI guidelines, there should be a measurement every 12 seconds, where the following variables are measured and/or calculated:

- Momentary irradiance,
- Average irradiance over the last minute,
- Average irradiance over the last 10 minutes,
- Maximum irradiance over the last 10 minutes,
- Minimum irradiance over the last 10 minutes and
- Standard deviation of the last 50 momentary irradiance measurements.

Thirdly, some error checks are set by the KNMI. This is when one could know that the sensor could not be functioning correctly, or the sensor itself is obstructed in any way. These error checks include exceeding hourly sum values which are specified per month or returning an hourly sum of zero between specified time frames (p. 7-13) [24].

Lastly, an important measurement condition of the pyranometer is the location. To get the best measurements, pyranometers that measure global radiation should be 1.5 meters above shortly cut grass. Furthermore, there should not be anything in the field of a horizontal view of the sensor for more than 5 degrees, which stretches 200 meters. An overview of an automatic MMS can be seen in Figure 4 [25].

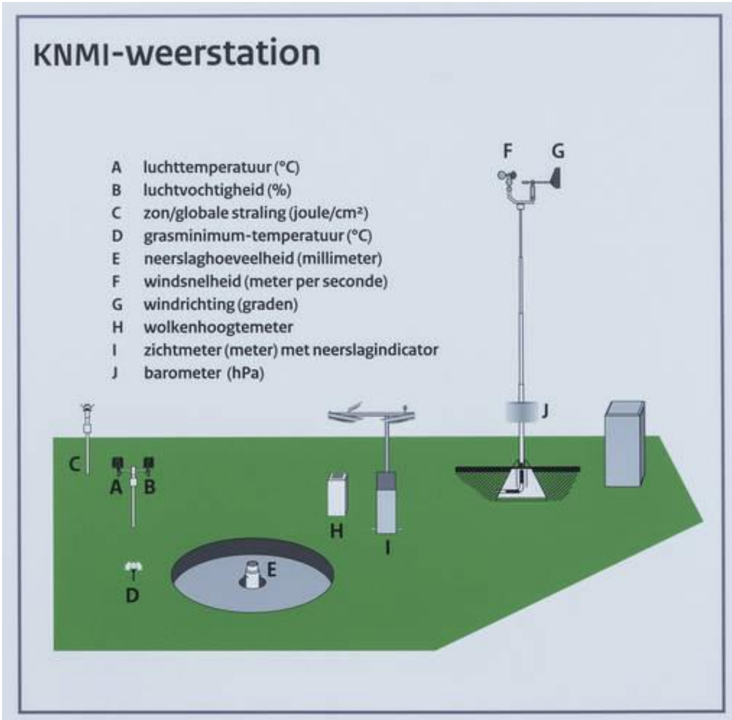


Figure 4: A overview of an Automatic MMS, where the pyranometer is denoted by a C. Retrieved from the website of the KNMI.

2.1.2. Taxonomy of the Pyranometer

To properly structure all types of pyranometers into different categories, and to know what to expect from each category, a classification system has been made.
This classification system is mainly based in two categories: Pyranometers that convert the electrical energy from the electromagnetic spectrum, and pyranometers that convert thermal energy from the electromagnetic spectrum. There are two subcategories, and these are dependent on the type of outputs you see with most pyranometers of a type: Analog and digital output.
The taxonomy as will be used in this thesis can be seen in Figure 5.

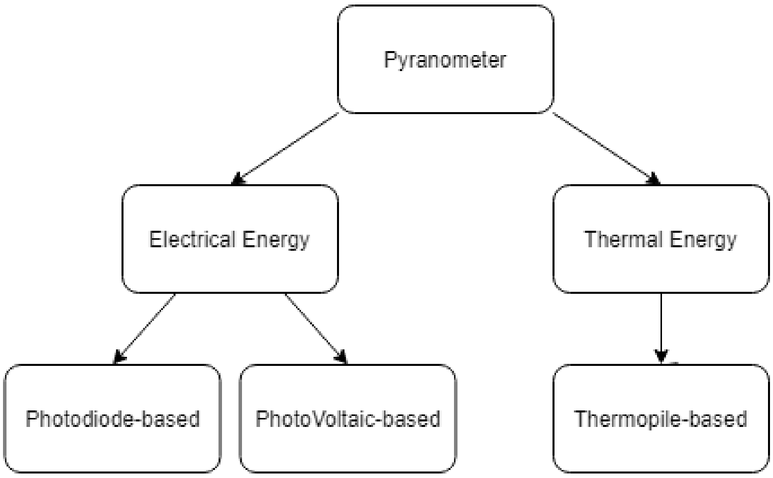


Figure 5: Taxonomy of the pyranometer.

2.1.2.1. Thermal Energy Pyranometers

Thermal Energy Pyranometers are pyranometers that use multiple thermoelectric junctions to generate a few microvolts per W/m^2. This is then proportional to the temperature difference between the thermoelectric junctions. The difference is denoted as one junction is the sensor junction, and the other is a reference junction which is not in direct view of the electromagnetic energy.

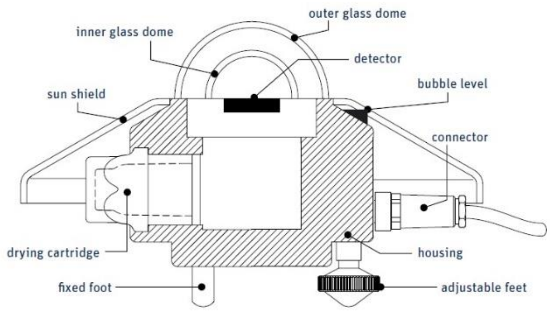


Figure 6: A Thermopile-based Pyranometer – Taken from the CM6 Pyranometer from Kipp & Zonen [28].

2.1.2.1.1. Thermopile-based Pyranometers

The thermopile-based pyranometer often uses two thermopiles to measure a difference in temperature. This difference in temperature can then be converted in the amount of energy that falls on the sensing element of the pyranometer. [26]

These types of pyranometers give an analogue signal out that is then converted into a value for the solar irradiance using a sensitivity value. Often this is done by computers, making it automated, to easily generate measurement values.

A positive aspect of these types of pyranometers is that they uniformly absorb the energy from across the short-wave solar spectrum (285 to 2800 nm). A negative aspect, however, is that they are dependent on how fast the sensor can cool down when there are clouds. This results in a slower response time that electrical energy pyranometers [27]. An overview of a thermopile-based pyranometer can be found in Figure 6 [28].

2.1.2.2. Electrical Energy Pyranometers

Electrical Energy Pyranometers are pyranometers that convert the electromagnetic energy coming from the sun into electricity. They generate a current that passes through a shunt resistor to easily convert the current into a voltage signal. This then results in a sensitivity of about a few microvolts per W/m^2. Often a special type of plastic diffuser is used to generate a cosine response. [26]

A good cosine response means that a 1000 W/m^2 which is perpendicular on the sensor is read as 1000 W/m^2 and when it approaches from a 60-degree angle, it is read as 500 W/m^2. This is important as it would otherwise be hard to compare to the reference system, which does have a cosine response.

2.1.2.2.1. Photovoltaic-based Pyranometers

Photovoltaic-based pyranometers use a photovoltaic cell, an example that is commonly known is a solar panel. This photovoltaic cell is used to measure the amount of energy that falls on the photovoltaic cell. These are often used to check the output of other photovoltaic cells, like a solar power plant. The photovoltaic cell works near short circuit condition and using this a current is generated which can be measured.

2.1.2.2.2. Photodiode-based Pyranometers

A photodiode-based pyranometer uses a photodiode to measure electromagnetic energy [27]. A photodiode is a type of photovoltaic device that is optimised for sensing electromagnetic energy. It is often used with an amplifier to generate a voltage that is proportionate to the current generated by the photodiode. Examples of these are also given by Benghanem [29] and Mukaro [30] since they use silicon solar cell pyranometers in their testing. An example is shown in Figure 7 [31].

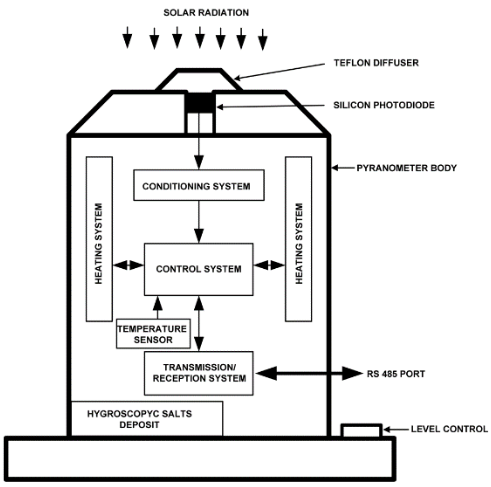


Figure 7: A Photodiode-based Pyranometer – taken from the research of Martinez et al. [31].

2.1.2.3. Electrical and Thermal Energy Pyranometers Comparison

The difference between Electrical and Thermal energy type pyranometers is quite apparent when comparing the output between the two. Electrical Energy Pyranometers are often prone to generate a small error when measuring solar irradiance when there is an overcast sky. This is because they are often calibrated in clear sky conditions. This can be seen in Figure 8.

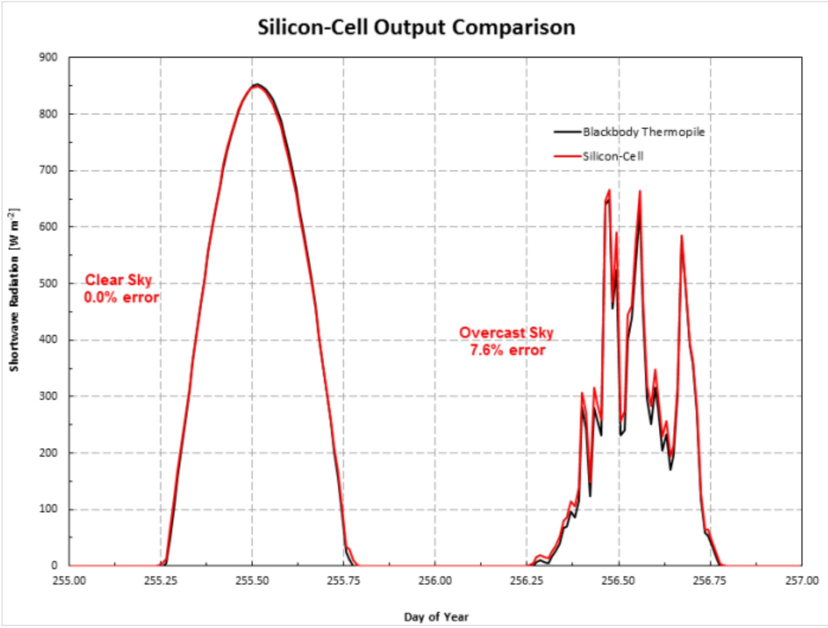


Figure 8: the difference between a thermopile pyranometer and a photodiode pyranometer, when comparing the response of the pyranometers. [26]

Furthermore, the difference between the types of pyranometers is also quite apparent when comparing the spectral response between the types of pyranometers. This difference is shown in Figure 9.

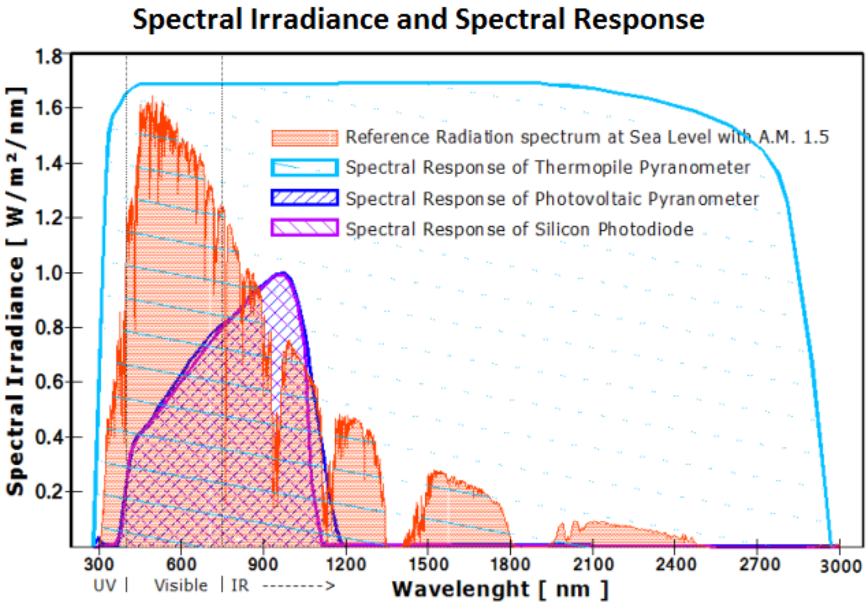


Figure 9: the difference between a thermopile pyranometer and a photodiode pyranometer, when comparing the Spectral response of the pyranometers. [26]

From these results, it is easily concluded that the best type of pyranometer to use would be the thermopile-based pyranometer. However, these pyranometers are very expensive, usually around 1800 to 2000 euros [28], meaning that the budget of 400 euros is easily exceeded.

The photodiode based pyranometer is often a bit cheaper, approximately 200 to 400 euros. This means that the photodiode-based pyranometer is a better option for the sensor that will be used, as it better fits in the budget of 400 euros.

2.1.3. Sensor System

The first sub-system in our sensor system is the sensor, the sub-system that will measure the solar irradiance. The next sub-system is the data processing unit, translating the output of the sensor to an understandable form of data. The third part is the wireless communication that brings the data from the sensor node to a central system. The fourth part is the sub-system that will provide power for all other sub-systems. There is also a chapter about sensor fusion. This is a technique to combine the output of multiple sensors into one, thus resulting in one sensor system that is made of multiple sensor sub-systems.
The sensor part of the system will measure the incoming solar irradiance and will convert it to a value that can be interpreted by the data processor. The sensor system will measure Global Horizontal Irradiance (GHI).

2.1.3.1. Data Processing

When a suitable sensor is made, this sensor will output a signal. This signal needs to be interpreted in some way. This can be done by a data processing unit. This unit should thus have the capabilities of receiving data from the sensor, interpreting it to a value that can be understood, and afterwards either sending it to the wireless communication system.

There are three types of data processing units used in general.
The first option, often when there was not a lot of data processing involved in the system or trying to keep the system low energy, is a microcontroller. An example of this, are the works of Vas *et al.* [32], Fisher *et al.* [33] and Tohsing *et al.* [34]. These so-called microcontrollers are stand-alone minicomputers that can perform a pre-set task. Most of these microcontrollers can be re-programmed. These microcontrollers differ in the amount of computational power they have so it is good to not generalize them too much.

There are also embedded processors. These processing units are often used for more difficult or bigger tasks. These embedded processors often need other components to function. In the works of Guzman *et al.* [35], they are used for processing images, utilizing a more elaborate neural network.

The third option is to just store the raw data over a certain amount of time and then perform post-processing using a computer. This system is called a data logger and is used in systems such as the ones from Tohsing *et al.* [34], Watras *et al.* [36] and Abbate *et al.* [37].

2.1.3.2. Sensor Fusion

Looking at how to measure solar irradiance, section 2.2.1 concluded in the use of a silicon solar cell type pyranometer. However, there is another way to approach this most important part of the system. That is by using multiple sensors and fusing their data into a combined value, thus approaching the to be measured value utilizing the fusion of multiple sensors.

The sensors that would be used exist out of four types of sensors. The first three types being defined by the solar irradiance classifications made in section 2.2.1.3 would make excellent divisions. This results in one sensor measuring the Ultraviolet type solar irradiance, one sensor measuring Visible light type and one sensor measuring Infrared type solar irradiance. There is a fourth option: this is a sensor that spans the bigger part of the Visible light, and the Infrared part of the solar irradiance spectrum. This would result in only using one sensor to cover two parts of the electromagnetic light spectrum.

2.1.3.2.1. Sensors

There are four different types of sensors to investigate: UV Sensors, Visible light Sensors, Infrared Sensors and Sensors that span multiple parts of the chosen spectrum division. When choosing these sensors, it is good to keep the coverage of the electromagnetic spectrum in mind, as it would be good if the spectral responsivity of the sensors connected or overlap.

2.1.3.2.1.1. UV sensors

UV sensors measure the part of the chosen spectrum with the smallest wavelength. These types of wavelengths are also one of the most harmful types of wavelengths. Multiple low-cost sensor options return the value of the measured UV light.
The modules that can be used are almost all based around a sensor made by Vishay, as it is either the VEML6070 [38] or the VEML6075 [39]. These sensors are often chosen as they include a spectral range that can measure the biggest part of the UV spectrum and have a pretty good cosine angle

Some examples of modules that use these sensors are sensors from SparkFun, Adafruit, and Grove. These modules all include an I2C communication method, as this type of communication is already included in the Vishay sensor.

2.1.3.2.1.2. Visible Light sensors

There are multiple options to measure the incoming visible light. Some options use sensors from Vishay, such as the VEML7700 [40] and the VEML6030 [41], but some modules use the TSL2591 sensor [42] made by AMS. All these sensors measure the amount of Lux incoming on the sensor and output this digitally via I2C communication.
There is also the option to use a Light Dependent Resistor (LDR) [43] or a 5V solar panel, and then measure the resistance and current respectively. These methods require more calibration, especially the latter, as this is also dependent on temperature, and thus using a temperature sensor is also required.

Examples of modules using Vishay sensors are modules from Adafruit [44] and SparkFun [45]. The LDR and 5V solar panels are readily available, as well as temperature sensors that need to be used together with the solar panel.

2.1.3.2.1.3. Infrared sensors

There are not as many options for measuring the Infrared light as there are for the other types of light. The most used method is the use of an IR phototransistor or an IR photodiode. These components can be used in a voltage divider setup, or with the help of an Operational Amplifier and passive components. Both methods would use the ADC of a microcontroller to be measured properly, thus the resolution of the ADC also sets the resolution of the sensor.

2.1.3.2.1.4. Multiple Range sensors

Sensors that fall under this category can measure multiple parts of the spectral range. For instance, the SI1145 [46] sensor can be used to measure all types of light. However, this sensor does require calibration and thus requires too much work for the types of measurements that will be made in this Graduation Project.
Sensors that can be used to measure the Visible light and Infrared light are the BPX43 [47] and the BPX43-3. This sensor is also used by Tohsing *et al.* in their low-cost pyranometer [34].

2.1.3.2.2. Data fusion Techniques

When using multiple sensors as described in section 2.2.3.1, it is needed to combine all of the outputs of the sensors into one value. This is called data fusion.

The use of data fusion for a measured amount of solar irradiance on earth has not been done very often, except for Gschwind and Wald [48]. They combined two datasets coming from two different satellites to calculate the average solar irradiance for multiple cities. Their research has shown that more complicated models such as affine transforms and quantile mapping performed the best.

However, there is the chance that in this project a microcontroller will be used, on which working with more complicated models is not possible. Furthermore, there is also the option of using machine learning to get the required output. However, this requires quite an extended algorithm that will use a fair bit of processing power.

2.1.3.3. Wireless Communications

With a suitable autonomous system also comes a way to communicate the gathered data wirelessly to a central point, or an interconnected network of machines. There are lots of ways to do this and a lot of different protocols.

Some of these ways include the following low-cost options; the use of a GSM/GPRS module that uses the 3G or 4G capabilities to send data over the GSM network, used by Zhang *et al.* [49]. Another low-cost option for wireless communication is Bluetooth [50], [51]. This does, however, have a limited range. Another option, used by related Creative Technology Projects before, is the LoRaWAN [52], [53]. A not as often chosen option is IEEE 802.15.4, a protocol frequently used by Zigbee products to create wireless networks [51]. Lastly, there is the option to use Wi-Fi, as in the city of Enschede, there is quite a broad network. Wi-Fi has been used before to work in wireless sensor node networks [54], [55]. An overview of the different communication technologies as well as important parameters can be found in Table 1. For the low-cost autonomous pyranometer, it would be best to have a quite large range of communication, as well as an energy-friendly communication technology since the system cannot be powered with a cabled power source. This leaves the option for LoRaWAN communication technology.

Communication method	Range	Typical data Rate	Energy Friendly
Bluetooth	≈ 10 m	2 Mbps	BLE
Wi-Fi	≈ 50 m	>100Mbps	No
IEEE 802.15.4	≈ 10 m	250 kbps	Yes
LoRaWAN	> 10km	<50kbbps	Yes
GSM networks	> 10km	>100Mbps	No

Table 1: Comparison of multiple communication methods.

2.1.3.4. Power Management System

To make the entire system function autonomously, the sensor system should have the capability to generate energy on its own. An often-chosen solution to this problem is the use of solar panels and a battery to store the energy generated by the solar panel for when the solar panel does not generate any more power. This solution is probably chosen because of the cost efficiency, and the ease of installing such a sub-system. Besides this, the weather stations are all outside, enabling them to harvest solar energy. These systems will be further explained in Chapter 2.2.

2.1.4. System Calibration

One of the most important parts of making a sensor system is to calibrate the sensors properly. This can be done by making a possible calibration equation to use with your sensor.

Mukaro *et al.* [30] worked with a calibration function in the form of a second-order polynomial with a proportional term *a,* and a quadratic term *b* to calculate the GHI with the recorded data value from their pyranometer. The coefficients *a* and *b* were retrieved with the use of a second, commercially available pyranometer. This was done by measuring the output of the low-cost pyranometer and the available pyranometer, and then mapping a trendline to these data points. Furthermore, the data that came from the sensor was first amplified by a low power amplifier to get the signal to a voltage that could be sampled by the microcontroller. The measurements are taken by averaging 20 consecutive readings.

A similar way to hone a low-cost pyranometer was applied by Tohsing *et al.* [34]. They also used a commercially available pyranometer to make a curve that sets the GHI as measured by the commercially available pyranometer against the voltage gotten from the low-cost pyranometer. They used four layers of Teflon sheet to reduce the solar irradiance for the used sensor, as it is sensitive to low solar irradiance levels. They mention that according to the ISO 9847 standard, the calibration method is to get the output voltage of a field pyranometer and the global irradiance from a reference pyranometer to calculate a Sensitivity S, which is then equal to the voltage divided by the global irradiance. Thus, a calibration curve was calculated which had a linear trend line. However, the phototransistor used as a low-cost pyranometer was misaligned at the beginning of the experiment. For the calibration of the system, it is important to make sure of the fact that the test setup is correct.

2.1.5. System Evaluation

Finally, it is good to evaluate the resolution and accuracy of your sensor, if it has a linear response and how well it compares against already existing systems. By evaluating the system, it can be seen if the required accuracy is reached. For the system, it would be good to evaluate the energy usage, and the range of communication as well. Next to that, the accuracy should be reviewed, just like the overall costs of the prototype. In this section, the accuracy will be discussed.

After tweaking their calibration function Mukaro *et al.* [30], used the standard deviation on their calibration coefficients to see how precise these were. One was quite precise, the other however not as much. This was mitigated by Mukaro *et al.* by the fact that it does not contribute much to the overall calibration function. When the average coefficients were used, they achieved a Root Mean Square Error (RMSE) of about 13 W/m^2, where the RMSE is a measure for the accuracy of the sensor. This shows quite some promise for making a low-cost pyranometer which is quite accurate.

Tohsing *et al.* did have a problem in the lower regions of the spectral range accuracy of their pyranometer. They say this was due to a misalignment of the phototransistor in the box. This was the reason that the accuracy was not very good. The overall RMSE of the sensors was 15.5%. This was explained by the fact that the sensors have a different field of view and a different spectral range.

In the research of Kim [50], the measure of accuracy was determined by a coefficient of determination. This is a measure of how well the estimation line fits the actual line, this is quite similar to the use of an RMSE.

2.2. State of the Art Solutions

This part of the background research chapter talks about the practice of pyranometers and measuring solar irradiance. This includes already build pyranometers and Meteorological Measurement Systems (MMS) where other ways of sensing solar irradiance were applied. It could also include how to make a MMS base structure, which needs power harvesting, data processing and communication technology.

2.2.1. Low-cost Thermal-Energy Pyranometer

Thermal-energy Pyranometers have been made by researchers like Hafid *et al.*[56].
Hafid *et al.* made a thermal energy pyranometer inspired by the Kipp & Zonen pyranometer, but the thermopile sensor was switched with a Peltier module. The overall design of the Kipp & Zonen pyranometer was kept. The pyranometer used a microcontroller to interface with the pyranometer and a computer over USB. The pyranometer can be seen in Figure 10. The pyranometer could measure a spectral response of about 300 to 3000nm. The overall accuracy response of the pyranometer seemed to be good from about 400- 1000 W/m^2, but not much was said for the lower ranges of solar irradiance.

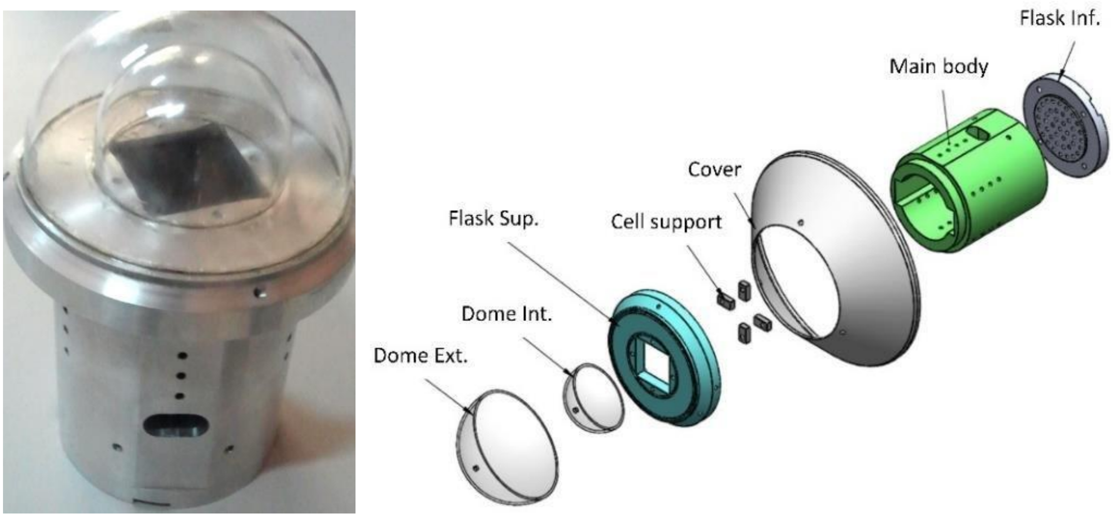


Figure 10: The Peltier module pyranometer made by Hafid et al. [56].

2.2.2. Low-cost Electrical-Energy Pyranometer

An example of an already made low-cost pyranometer is the developed pyranometer by Tohsing *et al.* [34]. This pyranometer is made with the use of a Silicon cell-based phototransistor: the BPX43-4. This sensor has a spectral response that lies between 450 nm and 1100 nm. The data processing is done by an Arduino Pro mini ATmega328P microcontroller. This microcontroller passed the gathered data to a micro SD card that was used as a data logger. This meant that there was no wireless communication, and the data was retrieved afterwards. The timestamps of the measurements were gained with the use of a real-time clock. The entire system was powered with a normal power supply, meaning that the system was not autonomous. The pyranometer can be seen in Figure 11.

(a) (b)

Figure 11: The BPX43-4 Phototransistor (a) and the made pyranometer (b) from Tohsing et al. [34].

2.2.3. Meteorological Measurement Systems

Meteorological Measurement Systems (MMS) are not new. A lot already has been made, all having different goals and results.

Devaraju *et al.*[57] have made a microcontroller-based weather monitoring system. They are using commercially available sensors, made by Davis Instruments, and are interfacing them with a microcontroller. They are using the reference system's Solar Radiation Sensor and other sensors to monitor the weather. The pyranometer is interfaced by using buffering amplifiers. These make sure that there is no loss of signal due to the sensor not being able to generate enough current [58]. They make use of a PIC16F887 microcontroller. The wireless connectivity is employed using an XBee-Pro module which uses the IEEE 802.15.4 standards. There is nothing mentioned about the power supply of the entire systems. The weather monitoring station can be seen in Figure 12.

Figure 12: Weather Monitoring Station from Devaraju et.al. [57]

Besides this system, there is also the SenseBox [59], which was made by the German Institute for Geoinformatics. They focus on making an autonomous MMS which can be used by everyone. Their project consists out of the main circuit board, to which different sensors can be connected. The overall system can be programmed with visual programming languages such as Blockly. They also use low-cost sensors to get a relatively cheap MMS, so that it is accessible for everyone. The system can be seen in Figure 13.

Figure 13: SenseBox low-cost MMS [59].

2.2.4. MMS by Creative Technologists

Other weather stations that have already been made are the stations made by former Creative Technology Students Tom Onderwater [52], Laura Kester [53], David Vrijenhoek [60] and Max Pijnappel [54]. The weather station of Pijnappel can be seen in Figure 14.
These are all versions and works that are built on top of each other, from the oldest to the newest system.

The first three weather stations did not measure the solar irradiance, only the latest Weather Station used a pyranometer to measure the solar irradiance. This sensor was the Davis Instruments Solar Radiation Sensor. This sensor was chosen because it fit within the budget and it fit the given requirements.

Figure 14: Weather Monitoring Station from Max Pijnappel [54].

The microcontroller that was used, was the SODAQ ONE [61] for the first three systems. The last system used an ESP32 [62]. This change was made due to the better availability of the ESP32 and the lower costs. There was also the fact that the ESP32 can store data in the flash memory, which means that when power is lost, measurements are not.

The SODAQ ONE has LoRa capabilities and for the ESP32 based weather station, an additional LoRa module was integrated to communicate data wirelessly. The reason why all of the weather stations use LoRaWAN is that it has a large range, so getting good coverage is relatively easy. Furthermore, LoRaWAN is an energy-efficient option.

The power management system that was chosen for the first one, was to use a power bank to have an easy solution with a high-power output. The rest of the weather stations used a solar panel with a battery and a charger circuit. In the latest Weather Station, a voltage divider was used to keep track of the battery voltage.

2.3. Conclusion

Multiple conclusions can be drawn from the research done in this chapter, but also from the practical examples that already have been made.

First, the difference between solar radiation and solar irradiance is important to take note of. The difference is subtle, but solar radiation is defined as the electromagnetic energy that is radiated by the sun. The solar irradiance is the solar radiation that reaches the earth and falls on a square meter per second. This is also where the unit (W/m^2) comes from. In this thesis, this quantity and unit to denote the solar irradiance will be used.

Secondly, the taxonomy of a Low-Cost Autonomous pyranometer was set, dividing pyranometers in two main categories: Electrical-energy and thermal-energy based pyranometers. Where the latter one is more expensive, but more accurate.

The sensor part, where sensor fusion is also an option, gives two ways to approach the measuring of solar irradiance. The first option uses a silicon cell-based pyranometer. These pyranometers give medium accuracy, medium-range spectral coverage and are relatively cheap, enabling them to use it in the to be made pyranometer. The second option uses sensor fusion to come to the desired accuracy, resolution and spectral range. This means that multiple sensors will be used, combining their data, to get one combined output for the solar irradiance.

The output of the sensor part will be sent to the data processing part of the system which will most likely be a microcontroller. This is because microcontrollers are cheap, and when choosing the correct one, it will result in quite a powerful data processing ability, while keeping energy consumption low.

The processed data will then be sent wirelessly, over a chosen medium. In combination with the fact that for the data processing a microcontroller will be used, a microcontroller can be chosen which can communicate wirelessly. Examples of these microcontrollers are the ESP32 and the SODAQ ONE. These have Wi-Fi and Bluetooth, and LoRaWAN capabilities respectively.

The entire system will need to be powered autonomously, meaning that it cannot be connected to the power grid. An often chosen and relatively cheap option is using a solar panel, battery, and battery charger circuit to provide this power to the system. This will result in the harvesting of energy and using this energy to autonomously operate the system.

After choosing the hardware for the entire system, the sensor or sensors that are going to be used need to be calibrated. This can be done by employing a reference pyranometer and measuring the output of the sensor or sensors whilst also measuring the output of the reference pyranometer. When using this technique, a calibration function can be formulated to map the output of the sensor or sensors to a solar irradiance value.

When an output value is calculated by the system when using a calibration curve, it can be evaluated with the use of a reference pyranometer. The level of accuracy can be expressed in a Root Mean Square Error value or a normalized Root Mean Square Error value. This will then give us a value that can be compared to the accuracy of the reference system.

Finally, there are already systems that look quite like the system that will be made in this research. These seem to contribute to the thought that a low-cost pyranometer can be made, with sufficient accuracy. Furthermore, a microcontroller seems eligible to be used as a data processing system and sending the information wirelessly.

3. Method and Techniques

This chapter will discuss the methods and techniques that are used during this graduation project. It will include an overview of the Creative Technology Design Method by Mader and Eggink [8], the method for the identification of different stakeholders, and the approach for analysing the different requirements that are set by stakeholders. Besides that, different ways of interviewing will be reviewed, and how functional architecture diagrams can be created. Lastly, the tools and testing procedures will be discussed and the evaluation methods of prototypes that are made.

3.1. Creative Technology Design method

The Creative Technology Design Method consists of four main phases: ideation, specification, realisation, and evaluation. One goes through these phases one at a time with a defined set of results coming out and going into each phase, but with the possibility to iterate and go back a step to further improve the outcome of a previous phase. The model is depicted in Figure 15.

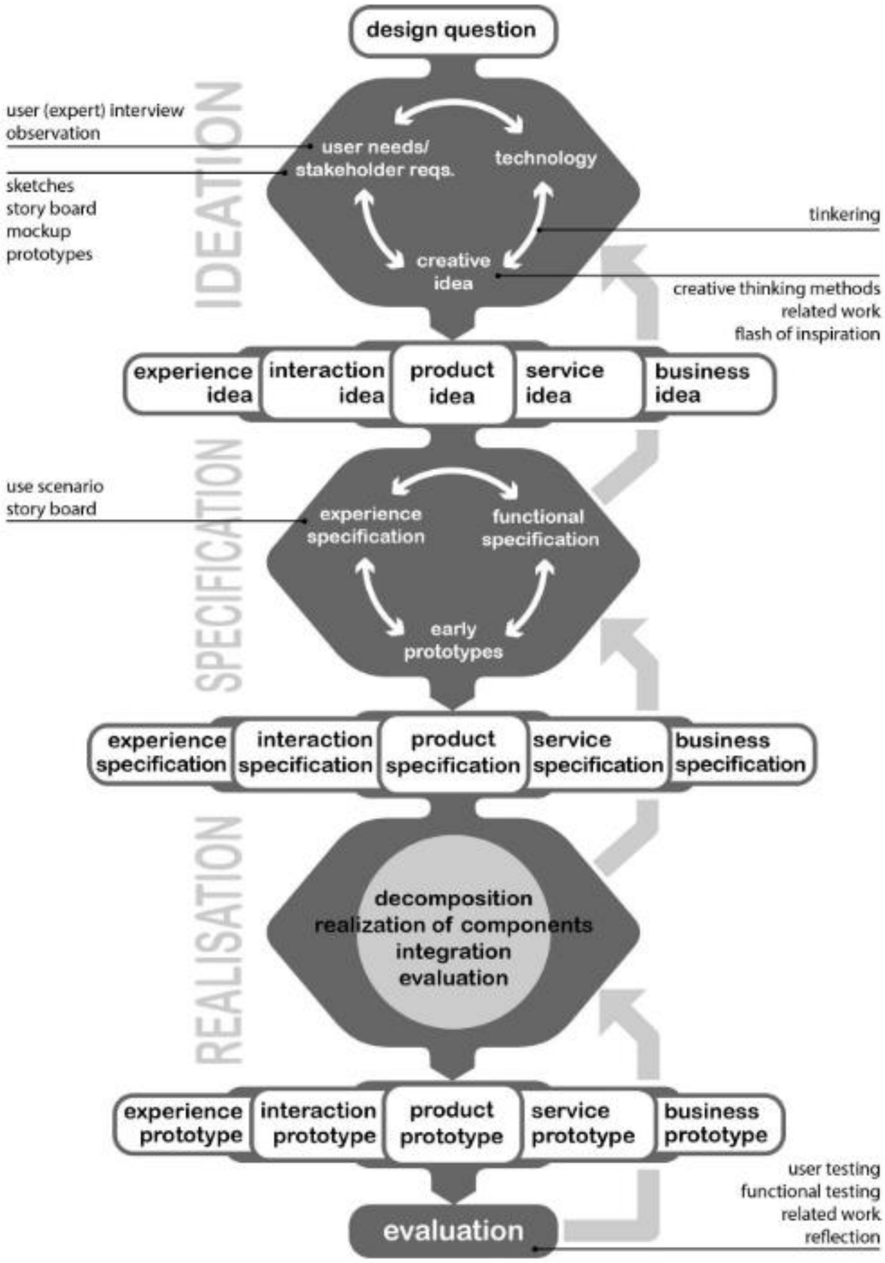


Figure 15: The Creative Technology Design Method as described by Mader and Eggink [8].

3.1.1. Ideation

The Ideation phase of the Creative Technology Design Method starts with a design question. This question is related to making a prototype of a product or system. In the ideation phase, the designer looks at the stakeholder's requirements and needs of the user to make a list of the required characteristics of the system. Furthermore, the designer thinks of creative ideas by using techniques like brainstorming and mind-maps, looking at related work and flashes of inspiration. The designer keeps these ideas in mind, just like the requirements and needs of stakeholders. When starting to tinker with some existing technology to come up with possible ideas an idea is created: the product idea is the desired outcome in this Graduation Project.

The ideation phase will be implemented in the following way in this Graduation Project. First, the Stakeholders will be identified and interviewed (see Chapter 3.3.), to get a better understanding of the general context of the project and requests and requirements that should be implemented in the project. These requests and requirements will be formulated as the preliminary requirements.

Secondly, the Environmental Factors will be investigated to find out what types of weather the system should be resilient to. These include factors such as the precipitation and humidity as these could impact the measurements of solar irradiance taken. It is important to take these factors into account as they could introduce new requirements to be aware of when making a prototype.

Furthermore, ideas will be explored for every sub-system of the low-cost autonomous pyranometer. These include sensors, microcontroller, wireless communication, power management and internal communication and wiring. This will be done by investigating the possibilities that came out of the State of the Art more and using brainstorming techniques, such as a Mind-Map, to generate multiple concept ideas to approach the low-cost autonomous pyranometer.

Then, some concepts will be generated on possible ways to make a low-cost autonomous pyranometer and these will then be evaluated against the preliminary requirements that were set during this phase.

From these ideas then comes a specific concept that will be taken into the Specification phase to form a final concept.

3.1.2. Specification

With the product idea that was generated in the ideation phase, the designer now enters the specification phase. Here the designer starts thinking about making prototypes to further explore the product idea and evaluate these prototypes based on the requirements that are set in the ideation phase and sharpened during the specification phase, employing a feedback loop to make another prototype. Any feedback that comes from an earlier prototype will be attempted to improve upon in the next prototype. The making of these prototypes results in a specification of what the product should entail and features that it should have: a product specification.

The specification phase in this thesis enlightens the sensors and autonomy enabling systems used in this project. This means that sensors that were picked in the first instance, are evaluated through a test to see which sensor gives the most reliable output. This is then used to sharpen the requirements and further improve upon making a prototype. This test is also used to pick a method of calibrating the sensors against a reference pyranometer to have a reliable output.

Next to the sensor sub-system, there are also other sub-systems. For the microcontroller, this means that it should be tested to be able to read and process the data in the most reliable, fast and energy-efficient way. Besides this, the microcontroller should possibly do some on-site processing to be able to stick to the requirements for the communication protocol.

Furthermore, the power requirements are to be specified. Next to that the sensors will be investigated and how much energy these consume. Besides, the power delivery system that was ideated in the ideation phase will be reviewed.

Then there is also the shielding of the sensors. The different opportunities to shield the sensors from the weather should be explored and presented. This would consist of the transmission of the different types of lights to be measured by the sensor to measure the solar irradiance.

To further investigate the different functions and interactions in the system, it is good to make a functional architecture diagram. This gives a schematic overview of the subsystems and which subsystems interact with each other in which way. This also results in better handling of the system complexity.

This phase will result in an overview of the requirements, as discussed with stakeholders, that the system should be able to achieve.

3.1.3. Realisation

When a product specification is created, the designer knows the specifications of the envisioned systems and can thus enter the realisation phase. Decomposing the system results in the components that can be chosen to be able to achieve the product specification. Then all the components are gathered, and the product should be assembled which requires the integration of every component into one working system. This phase will thus result in a functional product prototype.

The realisation phase of this graduation project focuses on the building and iterative developing of low-cost autonomous pyranometer prototypes. This is done by choosing components based on the specifications made in the specification phase and integrating them into the to be made system. These components are tested and then reviewed to see if they need to be replaced in the next iteration of the prototype. Whenever parts of the system are replaced, these need to be tested again. This is done until it converges into a working low-cost autonomous pyranometer.

3.1.4. Evaluation

After making the product prototype, it needs to be evaluated. This means that the prototype will be tested using a chosen method. Out of these tests come results which can be used to evaluate the made product. Out of these results, conclusions can be drawn by comparing the results against the requirements that are agreed with the stakeholders. These conclusions may then result in recommendations for future work.

The evaluation phase for the low-cost autonomous pyranometer consists out of two main parts. The evaluation of the accuracy of measured solar irradiance with a reference system, as well as the overall costs of the system. The overall accuracy is evaluated through a test, which compares the low-cost autonomous pyranometer data against a reference pyranometer.
The costs of the system are compared to a similar system to see if the costs are in proportion with the quality. These evaluations then result in quantified data to be evaluated and used to draw conclusions.

3.2. Stakeholders Identification and Analysis

The identification of the stakeholders is one of the most important parts of the ideation phase. Stakeholders are “any group or individual who can affect or is affected by the achievement of one’s objectives” [63]. This is because they put money in the project, or they will continue with the production of the project or product after it is finished. If one knows that they are designing a product with the influence of certain stakeholders, one can act on the influences and interests of these stakeholders.

Sharp *et al.* [63] divide stakeholders into two main groups which are then further divided. These two groups are the Baseline Stakeholders and the network of stakeholders around the Baseline Stakeholders.

The Baseline Stakeholders are further divided into four groups: Users, developers, legislators, and decision-makers. The users are the people, groups or companies that interact with the system directly and those that use the information and results that come from it.

Developers are stakeholders in the Requirements Identification and Analysis process. However, they do not have the same influence as the other type of stakeholders on the final requirements themselves, and thus the final system.

Legislators are professional bodies, government agencies, safety executives and such who may come forth with guidelines that the product should follow for a save/functional operation and development of the final system.

Decision-makers are often part of the users and developers. These parties relate to the system under development, often including managing the developers or being a financial controller of the developer or users.

The network around the Baseline Stakeholders gives three more types of stakeholders: the supplier stakeholder, the client stakeholder, and the satellite stakeholder. The supplier stakeholder mostly supplies information or resources to the baseline stakeholders, while the client stakeholders do the opposite and get information or resources from the baseline stakeholders. The satellite stakeholders’ interaction with the baseline stakeholders varies, however, it does not have much impact on the baseline stakeholders’ actions. In this thesis, the focus will be more on the baseline stakeholders for the requirements of the system.

The analysis of stakeholders can be done in different ways, but in this project, the power versus interest matrix is used, as first discussed by Mendelow [64]. However, the matrix of Mendelow is slightly adapted by mindtools.com [65] to get Figure 16. It could be beneficial to accentuate the internal relations between the different stakeholders, as this allows us to see how the communication works between different stakeholders, and who needs to be informed directly and who gets their information from another stakeholder.

Figure 16: Power vs Interest matrix – Adapted by mindtools.com from Mendelow, A.L. (1981) [64]

These connections are not as important as the four quadrants, as indicated in Figure 16. These quadrants tell what to do with the stakeholders, either keeping them satisfied, managing them closely, keeping them informed or just monitoring them.

To be able to properly set requirements, the stakeholders will be categorised with the use of the power vs interest matrix.

3.3. Interview Types

As discussed in section 3.1.1., interviews will be used to get a better understanding of the context and to get requirements from the stakeholders. There are five main types of interview techniques to investigate [66], which all require different levels of skill from the interviewer and interviewed, and will result in either quantitative data or qualitative data:

Structured Interviews: In these interviews, questions are created in advance in an interview guide, with little room for deviation of these questions, with not many open-ended questions. The interviewer does not insert their opinion in this type of interview. These types of interviews are best used when the interviewee understands the topic of the interview exceptionally well so that every question can be answered concise and short, resulting in qualitative data.

Semi-Structured Interviews: Here the questions are also created in advance in a guide, but there is room to follow topical deviation of the questions, which may be appropriate to gain knowledge. These types of interviews are best used when there is only one chance to interview someone. These types of interviews allow being prepared beforehand, while still having the freedom to express one's view on certain topics and gather qualitative data.

Focus Groups: A type of semi-structured interview, but then done with a group. Moderated by a group leader, these focus groups aim to explore the knowledge of a group of people, often consisting of 6-10 people. Furthermore, often multiple interviews are taken with different groups to explore the knowledge of a targeted group and gather quantitative data.

Unstructured Interviews: This interview includes a plan in the mind of the interviewer but not any written notes, like an interview guide. This makes questions very open-ended and leaves little control over the interviewee' responses. This means that the interviewer should be knowledgeable about the topic, to make sure to steer the interview in the right direction. This type of interview does require that there are multiple interviews, to gather the qualitative data.

Informal Interviewing: The previous interview all were formal interviews. With this type, the interviewer does not use any type of guide, but the interviewer recalls conversations with informants to gather knowledge from the interviewed. This type of interview is often practised when there is not much literature and is used to uncover new topics of interest that may have been overlooked by previous research. This type of interview is often done when an interviewer is practising fieldwork while observing the situation of interest. This interview results in quantitative data.

In this project, there are not a lot of chances to interview stakeholders and clients, so it would be good to choose the Semi-Structured Interview, as this is the best when you only have one opportunity to interview an expert.

3.4. Brainstorm techniques

There are different ways to visually brainstorm, to quickly generate ideas. In this section some of these visual brainstorm techniques. These techniques are retrieved from Creately [67], where multiple are discussed. The two most promising ones will be discussed further. These are mind-maps and flowcharts. Both techniques use the connection of different ideas to create a kind of hierarchy.
A mind-map is focussed towards idea generation and thinking more in-depth and into details as opposed to the flowchart. This would make it more suitable for the ideation phase.
A flowchart focusses more on the steps and path towards a goal to show a process. This would make it more suitable for describing the steps the system would take in the software, thus making it more usable for the specification phase.

3.5. Requirement Identification and Analysis

The requirements that are set for this product will be analysed to gain a good overview of all the requirements and the importance of them. To analyse and properly structure these requirements, the MoSCoW model will be employed, as will be discussed in the next section. It is also good to further divide the requirements in an overview of Functional and Non-Functional Requirements.

3.5.1. Requirement Analysis

The MoSCoW method uses a more understandable way of categorisation requirements than just setting a High, Medium, and Low priority. Instead, the categories are as follows [68]:

- Must have: The requirements within the category "Must have" are requirements that are critical to a product or are requirements that should be achieved. When the product does not have these requirements, it would mean the delivery of the system has failed.
- Should have: These requirements are important but not necessary for the system to work. Often these requirements are just as important as the ones in the "Must have" category, but it could wait until a next iteration to be implemented.
- Could have: The "Could have" requirements are desirable to have but not necessary. They could improve the user experience or satisfaction.
- Won't have: Requirements that are labelled as "Won't have" are requirements that do not need to be fulfilled for this iteration. However, these will be put forward to implement another time.

These categories contribute to an overview of all the requirements being prioritised. This will be done for the preliminary requirements and the final requirements. These requirements that are set will be agreed upon with the stakeholders to make sure they are in the correct category.

3.5.2. Functional and Non-Functional Requirements

After prioritising the requirements, they can be further divided into functional and non-functional requirements [69]. Functional requirements are requirements that specify functions that the system or sub-system must be able to perform. Often these requirements can be measured. An example of this is: The system should be able to measure Solar Irradiance, with an accuracy of 5 W/ m^2. Non-functional requirements are requirements that describe how the system must perform these tasks, or how it should look. An example of this is: Make a system that costs less than 400 euros.

3.6. Functional Architecture diagrams

To properly determine the interaction between different sections of the hardware in the system, a Functional Architecture diagram can be made. This can be done for the hardware, as well as the software. To make such a diagram, the method proposed by Brinkkemper and Pachidi [70] will be followed.

The method proposed by Brinkkemper and Pachidi consists of five steps:

1. **Determine the scope**: This step is important to know with what kind of environment the system will have contact.
2. **Define request-feedback flows**: after determining the outside connections, the interactions between the outside environment and the system are defined.
3. **Model the operational module flow**: This step determines the inner workings of the product. Here the modules and interactions between these modules in the system are defined.
4. **Add control and monitoring modules**: Here modules are added that are capable of interaction by a user. This makes sure that when it is needed the user can add input, or see the output of the system
5. **Specify external to/from internal interactions**: This step looks back on step 2, as the connections between the system and outside systems are evaluated and reconsidered. Here it is possible to discover new connections that were not found in step 2.

3.7. Tools

This section will determine the types of tools that were used in making the different prototypes during the realisation phase.

3.7.1. Arduino IDE

The Arduino IDE is an open-source software to write code for microcontrollers [71]. It is based on other open-source software like Processing. It is mainly used to program C++ for Arduino Boards. However, it is possible to add other types of microcontrollers to the Arduino IDE and use this platform to program them. This is also possible with the TTGO LoRa32 SC1276 OLED microcontroller that is used in this version of the low-cost autonomous pyranometer.

3.7.2. EasyEDA

To be able to document the electrical connections and wirings EasyEDA is used. EasyEDA is an open-source platform to document electronics projects. It is capable of visualizing schematics as well as order Printed Circuit Boards from these schematics [72].

3.7.3. Microsoft Excel

To be able to perform a regression algorithm, Microsoft Excel will be used. This is to determine a calibration function by way of linear regression. This is done by using the Analyser Toolpak add-on that is available [73].

3.8. Testing Procedures

In this section, the different testing procedures that will be used to examine the functionalities of the system will be discussed. The first section will discuss the basic tests of functionality for the sensors that were gathered as potential candidates and to compare them to the reference system.

The second section will discuss the testing of the first prototype. Here the functional requirements of the system will be tested to see if they are feasible or not. This will result in testing the accuracy, resolution, and precision of the used sensors. But also, things as the power generation of the system and the communication technology.

3.8.1. Sensor Testing

To make sure that the first selection of sensors was adequate, it is good to test the different sensors before they are used in a prototype. This makes sure that there are no unexpected problems with the use of different sensors. For example, that the sensor is maxed out and it caps even though theoretically, it should be possible.

To achieve a good set of readings, this test will be performed in the Volkspark as it has quite an open unobstructed area with a grass field, which conforms the rules and regulations set by the KNMI [24]. Furthermore, the testing bed will be levelled to make sure that all sensors are level and give an accurate reading. Besides this, the sensors should be as unobstructed as possible during the tests to make sure that they give an accurate reading.

The data gathered in this test can be used to decide on with a calibration method for the sensors to be able to measure the solar irradiance. This calibration method can then be used in the next prototypes and be tested. The calibration methods that will be used is a type of regression algorithm implemented by Microsoft Excel.

3.8.2. Prototype Testing

To make sure everything functions as designed, it is good to test the functional requirements with the first prototype, as these are often the easily measured basic requirements that should be passed. This testing will consist of two main steps: Testing the sensor itself and testing the system around the sensor.

3.8.2.1. Sensor Testing and Calibrating

For the testing of the solar irradiance sensor, a couple of things should be kept in mind.
When gathering data, it would be best if every sensor has as much unobstructed view as possible. Besides the unobstructed view, the sensors should be level with the ground, to make sure they can be compared to the reference sensor. A tripod or another type of stable platform is thus preferred as this makes sure the sensors stay level. The reference sensor should be installed beside the low-cost sensors that are implemented in the system to make sure it gives a reading that can be used to calibrate the low-cost sensors.

To make sure that the sensors are well-calibrated, there should be enough data points to be able to calibrate the low-cost sensors. These data points should be diverse as to make sure that it works for both sunny as well as cloudy days, and not have a bias towards one, like most electrical energy pyranometers (see Chapter 2.1.2.3.). This means that it would also be good to have a test during the night, as to see what values are returned then.

Furthermore, it would be good to test different materials to cover the low-cost sensors, to shield them from the weather. As it is also interesting to measure UV light, a shielding that is transmissive for UV light is needed. Multiple types of these materials could be tested to make sure the best one is used. The covering of the sensors with a cover will result in less exposed sensors as all transparent casings absorb some of the solar irradiance.

When a good amount of data points has been gathered, the Microsoft Excel Analysis Toolpak can be used. From the analysis tools, the regression analysis is used to formulate an equation where the output data of the low-cost sensors are the input for the X range and the output of the reference sensor as input for the Y range. The coefficients that come out of these tests, can subsequently be used for a calibration function, which can then be tested.

The most important features when looking at Multiple Linear Regression (MLR) are R squared, the standard error of the regression, the significance of the test and the p-value for each variable [74]. R squared is also known as the Coefficient of Determination. This shows how much of the points fall on the regression line. If it was 0.80, it means that 80% of the points are on the regression line, meaning that the closer to 1 this number is, the better the regression fit. The standard error gives the average distance from the data points to the line [75]. The lower this is, the better. This is not the standard deviation known in classic statistics. The Significance F shows if the test had significant enough test entries, the closer to zero the better. And the P-value gives the p-value for the hypothesis test, this should be below 0.05.

This method will be used for the calibration of the solar irradiance sensor to be made. Besides the reference system used in this thesis, of course, others can be used to calibrate low-cost sensors.

3.8.2.2. System Testing

Besides the solar irradiance sensor, the other sensor that needs to be checked is the GPS location sensor. This sensor can be checked using locations of which the coordinates are known. The resolution can also be retrieved, by using the datasheet of the sensor, but this also depends on the resolution that can be transmitted by the communication method.

Next to the sensors, there is also power management. It would be good to monitor the battery level during such a test. The battery level should not drop during the test. This can be measured, to determine if the battery had dropped or not.

Lastly, the communication can be evaluated. The data packets that are retrieved versus the amount sent can be checked and the 1% duty cycle rule should be kept in check. This can be done by calculating a percentage of packets received and transmitted and by calculating the number of bytes sent every day and comparing this to how much bytes are allowed.

4. Ideation

This chapter is used for the gathering of requirements that stakeholders and users have for the low-cost autonomous pyranometer. Different stakeholders will be interviewed to gather necessary information and finally generate some preliminary requirements.

4.1. Stakeholders

With the use of the method described by Sharp *et al.* [63], identification and analysis of the stakeholders have been done, and the results of this are shown in Table 2.

Stakeholder	Contact Person	Category
University of Twente	Richard Bults Hans Scholten Wim Timmermans	Decision-Maker User
Municipality of Enschede	Rik Meijer	Legislator User
Developer	Peter van der Burgt	Developer
KNMI	N.A.	Supplier Stakeholder of University of Twente Legislator
The Things Network	N.A.	Supplier Stakeholder of Developer Legislator
Residents of Enschede	N.A.	Client Stakeholders of the Municipality of Enschede

Table 2: Identifying the stakeholders and analysing their category according to Sharp et al. [62].

4.1.1. University of Twente

The main stakeholders are often the stakeholders that have given the assignment. In this project, these stakeholders are Richard Bults and Hans Scholten from the University of Twente. Together with Wim Timmermans from the ITC department of the University of Twente, they work on a network of wireless sensor node systems with the Municipality of Enschede. This research led to the need for a low-cost autonomous pyranometer.

4.1.1.1. Richard Bults and Hans Scholten (Creative Technology)

To gain some more insight into the project, like the background of the project, expectations for the system and requirements for the system, a semi-structured interview has been conducted (Appendix A). Richard and Hans mentioned that because of the collective price of the Davis UV Sensor and Davis Solar Radiation Sensor, the need for a low-cost high-quality solution had risen, which uses off the shelf components.

According to Richard and Hans, the main aspects that should be considered with the Low-Cost Autonomous Pyranometer, are the quality of data and the cost of the entire system. To ensure good quality of data, multiple things should be considered. Not only the sensors should be of good quality, but also the location of the sensor is important, even though the system would most certainly be stationary. This is not only because a location stamp is a well-added variable for quality, but the time at which the measurement is taken is even more important. The timestamp makes sure that the data can be gathered at correct times. The quality of measurements also takes into account the rules and regulations for measurement frequency as set by the Royal Netherlands Meteorological Institute [24].

To be able to compare the made system, the system should be validated against the Davis Instruments Vantage Pro 2 system.

The cost of the system in total, should not exceed 400 euros. This means that compromises may have to be made for functionality against the cost of the system, looking at what is a must-have requirement versus what would or should be good to have.

Next to that, there is the operation of the system. As mentioned before the system should be able to function autonomously, meaning that it should be able to operate without an external power source. With this comes power management, for example only allowing power to go to sub-systems that are in use. This also means that when choosing sub-systems, energy consumption should be considered besides the costs.

It could also be an option to look into graceful degradation, meaning that some systems will not be in use, during periods where they do not contribute to gathering data, however in the case of the low-cost autonomous pyranometer this will not be the case.

Lastly, Richard and Hans mentioned the dependability of the system. This includes the fact that the system should be able to function for three months at least. With this comes the availability of the system, for example, if the system supports 24/7 functionality and does not need any maintenance. It would be good to look more into the dependability parameters. Vrijenhoek [60] researched the dependability of a system based on the energy-awareness and the availability of the sensor system. These are two parameters that Richard Bults and Hans Scholten specifically mentioned. It would be good to take these into account when designing the low-cost autonomous pyranometer. Next to the continuous functionality of the system, the dependability of the system also includes the quality of the gathered data. This quality of data is to be ensured by the fact that time and location data should be gathered to extend this quality of data.

4.1.1.2. Wim Timmermans (ITC),

Wim Timmermans is a researcher at the Faculty of Geo-Information Science and Earth Observation (ITC) department, specialising in making models to simulate meteorological phenomena. Together with Richard Bults, Hans Scholten and the Municipality of Enschede, he is working on making a network of Meteorological Measurement Stations. There have been two interviews with Wim Timmermans to make sure that all requirements were set properly.

The main point that could be inherited from the interview with Wim Timmermans, is that the most important part of designing the system is to know in what context you are making the system. For example, if the system is to be used for the generation of data for mathematical models, on which warnings are based, it needs to be more precise than when it is used for informing residents of Enschede of the temperature.

Figure 17: Provisional placement of the low-cost MMS shown with yellow circles. – received from Wim Timmermans

Also, the dependability of the system was very important for Wim Timmermans. The system should be able to be online 24/7. The foremost point here is the power supply for the system. There should be sufficient power, to make sure the system functions under all conditions. The system itself should be able to function for a long time or be cheap enough to easily replace again. Meaning amongst other things, that if the system was to not function properly, it should be able to tell so itself, as to make sure that the system can then be replaced.

Furthermore, the placement of the system, as seen in Figure 17, can be important for how the mounting point should be designed. Wim Timmermans mentioned that the systems will be placed on different types of poles around the city. Next to that, the system will not be placed 1.5

meters above the ground, as mentioned in KNMI guidelines, since systems could be vandalized or stolen. However, they will all be placed at the same height to be able to compare variables. This will be about three to six meters high Solar Irradiance is the only variable that is not affected by height; thus, this should not interfere with measurements.

The quality of measurements depends, according to Wim, not as much on GPS location, as he would measure those himself. However, this would depend on the time stamp and the data itself. The data itself is mostly affected by the resolution of the system. This should be about 10 W/m^2 for low-cost applications according to Wim. Besides this, the number of packets sent should be recorded as well, as this allows the user of the data to determine if the data is representable to use.

In the second interview, the frequency of measurements and the frequency of data communication is mostly discussed. For the frequency of the measurements, Wim mentioned that it would be good to try to stick to the guidelines of the KNMI. This is because the values measured in the city will be compared to the values measured outside the city.

This would mean that every 12 seconds, a measurement is taken (SAMPLE measurement [24]). However, Wim said that the average of the measurements is the most interesting to him. The minimum, maximum and standard deviation would be good to have, however, Wim would rather have the SAMPLE measurements [24] in that case. To achieve this, a small data logger could be used to store all SAMPLE measurements. Besides this, Wim mentions that there is a possibility there will be a system in place, such that students will check every sensor node about every 2 weeks. This could then mean that the system could offload all taken sample measurements.

Furthermore, the communication frequency is discussed. Wim mentioned that it would be fine to have a bulk of measurements send at once, meaning that 150 SAMPLE measurements are sent at once. However, this should be with a maximum time interval of half an hour.

4.1.2. Municipality of Enschede

The second stakeholder is the municipality of Enschede, as they will house the system, but will also use it to gather data, and possibly change rules and regulations based on this data. The goal of the municipality is to gain insight into the severity of the UHI effect and find where it has the most impact. The nodes will be deployed in Enschede, meaning that the municipality of Enschede is a big stakeholder, as they can influence or impact the distribution both positively and negatively.
The system and its deployment should abide by any rules and regulations that already exist.
The rules and regulations they might adjust with the acquired data from the sensor will be mostly regulations on how buildings are built in the city of Enschede, to keep the UHI effect to a minimum.

In an interview with Rik Meijer, a policy advisor for Climate adaptation at the municipality, he mentioned that the municipality was mainly interested in mapping the Urban Heat Island effect. Especially when an area has been adapted, as it would be good to see the effects of the adaptation on the Urban Heat Island.

There are also plans to investigate designing a web application so that the inhabitants of Enschede can see the influence of the UHI on their neighbourhood.
It would also be possible to use this platform to incentivise the inhabitants to work on their own house and garden, to make sure that the UHI effect is minimalised.

Furthermore, the possible vandalization of an MMS (Meteorological Measurement System) has been discussed. To combat this, the municipality aims to install the systems high enough of the ground so that someone with a stick should not be able to reach it.
Another solution would be to install these systems on private grounds, in the backyard of an inhabitant of Enschede. However, this could pose a problem for the data gathering as this could be a problem with the GDPR law, as the location of the system would be important.

There was also a small discussion about how the systems should be installed. The most important thing is that it should not be as heavy as to damage the lampposts, and in general the system should not damage the lamppost.

4.1.3. Residents of Enschede

Lesser noticeable stakeholders are the residents of Enschede. They could gain an interest in the project, as it could lead to different urban architecture. Furthermore, they could encounter the low-cost autonomous pyranometer while outside, meaning that the system should be robust and not easily be broken.

However, above all, they are client stakeholders of the Municipality of Enschede. This is because they get information from the municipality, but also rules and regulations that they should follow, based on the information the municipality could gather with the pyranometer in place.

4.1.4. The Things Network

Although this may not seem like a stakeholder, The Things Network (TTN) do supply other stakeholders with important information, which leads to requirements that will be set for the Low-cost Autonomous Pyranometer. This mostly has to do with the limitations set by this service for LoRaWAN communication, which means that the system should not exceed the requirements mentioned in the fair use policy. According to the LoRaWAN use policy, a node should have a 1% duty cycle for transmitting data [76] and according to the fair use policy of TTN, they do not allow more than 30 seconds of uptime during one day [77].

4.1.5. Royal Netherlands Meteorological Institute (KNMI)

Although the KNMI may not seem like a stakeholder, they do supply other stakeholders with important information, which leads to requirements that will be set for the low-cost autonomous pyranometer. This has to do with the measurement frequency and with the communication frequency of the system. For the representation of the solar irradiance, the KNMI uses multiple units. For most of the measurements the W/m^2, and for the hourly and daily sum the unit J/m^2 is used. As can be found in the next sections the W/m^2 is converted to J/m^2 by multiplying it with the time in seconds. For an hourly average, this results in multiplication with 3600s.

4.1.5.1. Resolution, Accuracy and Precision

As mentioned in the KNMI Manual for Observations [24], the global irradiance is measured from a spectral range of about 305 to 2800nm. This is done so with a range from 0 to 2000 W/m^2 to ensure sensing the irradiance correctly. The resolution mentioned should be 1 W/m^2. About the accuracy, only the net total radiation is mentioned, so this does not directly correlate to the global irradiance. However, this is approximately 5% when the daily sum is more than 8 MJ/m^2, and approximately 0.4 MJ/m^2 when the daily sum is less than 8 MJ/m^2.

4.1.5.2. Measurement Frequency

According to the KNMI Manual for Observations [24], every 12th second the following things should be measured:

- SAMPLE: the momentary solar irradiance (W/m^2), measured over a "couple" of seconds
- MINUUT: average solar irradiance over the last minute (W/m^2), calculated with the use of the last 5 SAMPLE measurements.
- 10GEM: average solar irradiance over the last ten minutes (W/m^2), calculated with the use of the last 50 SAMPLE measurements.
- MAX: maximum solar irradiance over the last ten minutes (W/m^2), calculated with the use of the last 50 SAMPLE measurements.
- MIN: minimum solar irradiance over the last ten minutes (W/m^2), calculated with the use of the last 50 SAMPLE measurements.
- STD: standard deviation of the solar irradiance (W/m^2), calculated with the use of the last 50 SAMPLE measurements.

The measurements that span ten minutes (10GEM, MAX, MIN) are calculated between the period of five minutes before the point in time, and five minutes after. Figure 18 shows a visual representation of how these measurements are measured and calculated.

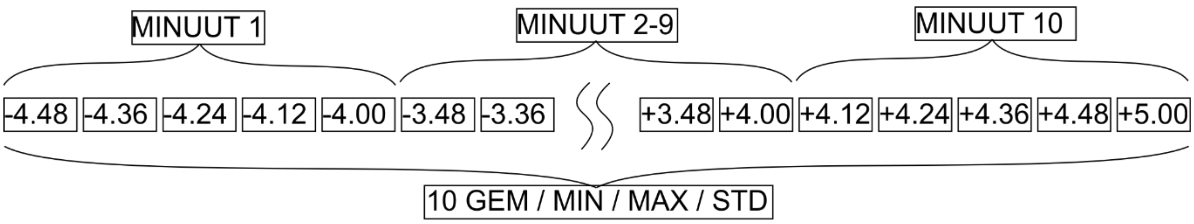


Figure 18: A visual representation of the different measurements taken.

These measurements are stored separately in storage that handles the "ten minutes" values, these thus include 10GEM, MAX, MIN and STD.

Every hour, ten minutes before the hour, the following is calculated:

- An hourly average is calculated with the previous 300 SAMPLE measurements (W/m^2).
- An hourly sum calculated by multiplying the hourly average with 3600 seconds (J/m^2).

Furthermore, a daily sum is calculated, by adding the hourly-sum values of an entire day (J/m^2).

4.1.5.3. Communication Frequency

Not much is known about the communication frequency. However, the KNMI website states that every "couple" of seconds a measurement is taken, which is then sent towards the KNMI. An employee reviews data on quality and alters this data where needed. Every 10 minutes the data gathered on the website is updated.

4.1.6. Power vs Interest Matrix

This chapter is to reflect on the different stakeholders and analysing them by using the power vs interest matrix. The matrix, depicted in Figure 19, is used to divide the stakeholders into four quadrants, resulting in a suggested way of handling these stakeholders.

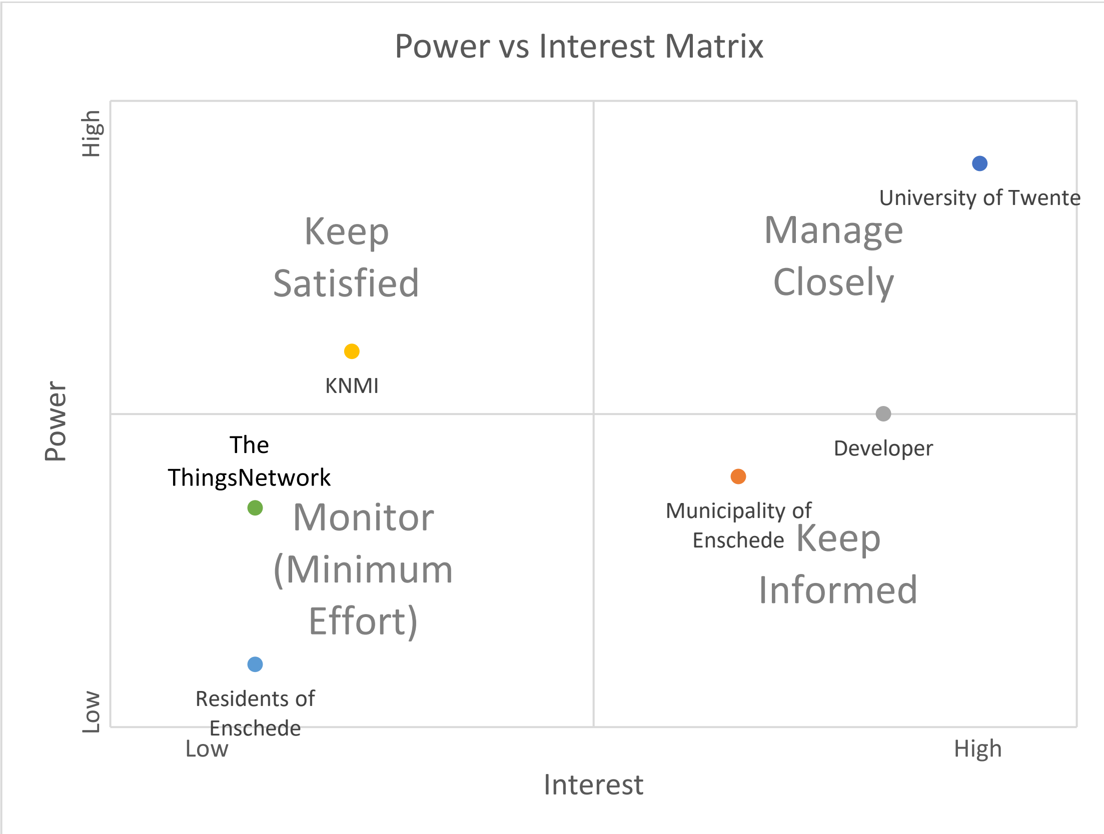


Figure 19: The Power vs Interest Matrix adapted by Mindtools.com from Mendelow [63], [64].

4.2. Environmental Factors

The low-cost autonomous pyranometer will be deployed outside in the city of Enschede. This means that the system should be able to handle the weather that occurs in Enschede. This means that temperature, solar irradiance, humidity, precipitation, and wind speed will be investigated.

4.2.1. Weather Variables

These average, minimum and maximum values are taken from the dataset from the KNMI (2019) [78]. The following numbers and measurements were taken during the entirety of the year 2019, meaning from the 1st of January 2019 to the 31st of December 2019.

The climate of Enschede is often described as warm and moderate. This means that there are no real extremities. The lowest measured temperature in Enschede in 2019 is -10.1°C and the maximum temperature measure is 40.2°C. These are the extreme cases, but it may be better to look at the average temperatures. The lowest average daily temperature is -5.2°C and the highest average daily temperature is 30.5°C.

The solar irradiance that has been measured by the KNMI is also interesting for this Graduation Project. This may show the practical boundaries for measuring solar irradiance. The lowest amount of solar irradiance that has been measured is 47 J/cm^2 and the highest amount is 3055 J/cm^2. These measurements are the daily sum and should not be confused with measurements taken in W/m^2. Chapter 4.1.5.2. shows how a daily sum is calculated.

The humidity that has been measured is also important. This could influence which sensors will be used in the system. The minimum daily average humidity measured is 37%, the highest daily average humidity is 99%. The highest measured humidity is 100%, the lowest is 17%.

The measured precipitation could influence how well the sealing of the enclosure should be investigated. However, it should be superfluous to say that the system should not sustain any water damage. The minimum daily precipitation (i.e. rain, snow, hail) measured is less than 0.05mm, the highest daily precipitation 43.8mm.

The windspeed that has been measured is also important, as high wind speeds could knock the system of a mounting point. This may influence the manner on how the system is attached, as it should not be knocked over, and should stay level. The minimum daily average windspeed measured is 0.7m/s, the highest daily average wind speed is 8.9 m/s. The highest measured hourly wind speed is 12m/s, the lowest is 0 m/s. The maximum wind gust is 23 m/s.

4.2.2. Climate measuring in Urban Areas

The World Meteorological Organization also published a document about measuring representative observation at urban sites [79]. It mentions that for the measurement of solar irradiance, urban cites are mostly avoided, due to the aerosol and gaseous pollutants. Furthermore, it is mentioned that the measurement of solar irradiance is important to calculate more sophisticated measures.

For the placement of pyranometers, it is mentioned that the fundamental needs are for the sensor to be level, free of vibration and free of any obstruction above the sensor, both fixed features, like buildings, as non-fixed features, such as smoke clouds. This often results in the roof of a taller building being used, as it meets most of these requirements.

For gathering measurements, it is explicitly stated, that rooftop sites should be avoided, except for measuring solar irradiance. Furthermore, the type of surface where the sensor is placed should be representative of the terrain where the measurement is taken.

4.3. Ideation Low-Cost Autonomous Pyranometer

To create a low-cost autonomous pyranometer, it is first good to explore all different aspects to be designed to make such a system. In Chapter 2.1.2. there is already a start made in defining the sub-systems of the pyranometer. However, this does not include the underlying interactions between the sub-systems. To further explore the different systems and possible ways to perform their tasks, a mind-map has been made, which can be seen in Figure 20.

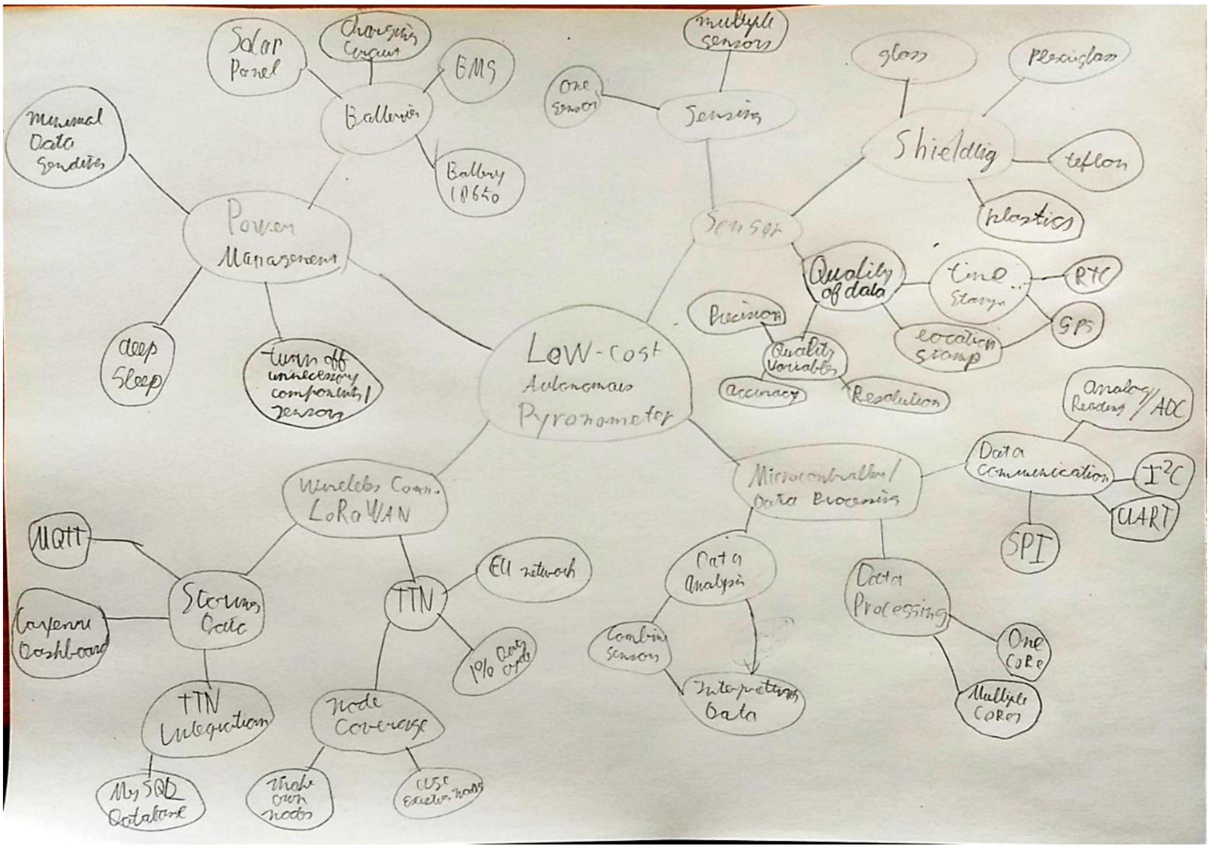


Figure 20: Mind Map of the different sub-systems and possible solutions.

4.3.1. Sensor

From the state of the art and literature research came some ideas on how to measure solar irradiance utilizing cheaper components. However, besides the fact that these multiple ways of measurements will be explored, the possibilities of shielding the sensors for rain should also be investigated, whilst still be able to measure the solar irradiance. Furthermore, the power consumption of the sensor should be explored, and ideas should be generated to keep this to a minimum.

4.3.1.1. Sensing Solar Irradiance

There are two main ways to measure solar irradiance with a low-cost sensor.
The first is to have one sensor, which spans a broader range of wavelengths. For example, one sensor that spans the visible light and the infrared light. This often includes a type of phototransistor or photoconductor such as in the research of Tohsing [34].

The second option is to use multiple sensors, and by way of sensor fusing, combining the data to measure the solar irradiance. This means that a sensor for every type of light can be used, in the case of the system to be made, ultraviolet, visible light and infrared, after which the data of all are combined to get an overall solar irradiance.

4.3.1.2. Pre-Processing and Combining Sensor Data

When the solar irradiance sensor sub-system would exist out of multiple sensors, techniques should be employed to properly combine the data coming from the different sensors. Factors include the sensitivity of the sensor, the spectral range, and the resolution.

Firstly, there should also be a correction for the Spectral Response to the different types of light, as the entirety of the spectral response should be measured. However, most of the sensor modules that could be used, use a library that already accounts for this spectral responsivity.

Furthermore, another signal conditioning aspect to look at is the overlapping spectral responsivity and negate one of the overlapping parts to form a continuous spectral response. This is done so none of the wavelength ranges are counted twice.

Then, the combining of data streams can be done in multiple ways. There is a way of adding weighted variables like done by Amer *et al* [80]. Here the sum of the different sensor outputs is used, where the sensor outputs are normalized with a factor. This can then be compared to the actual generate a function that can predict the outcomes.

Besides these possibilities, there is also the option to create a neural network to try to tailor incoming data from the sensors to an output of solar irradiance in W/m^2. However, this does include the usage of a reference system, to learn the required outputs. It would be best to let the software create a function that can be used to calculate the solar irradiance.

There is also the option to directly get the measured data and use them as inputs to a calibration function that then calculates the amount of solar irradiance. This means that an existing pyranometer should be used as a reference system to create a calibration function. The existing pyranometer could be a Kipp & Zonen CNR1 being used by Wim Timmermans or the reference system sensor, the Davis Solar Radiation Sensor. These types of techniques are things like linear regression, multiple regression and using them to calculate a trendline. A relatively easy way to implement this is with the use of Microsoft Excel.

Whatever type of sensor fusion is used, it would be best to create a final calibration function, to make sure that the pyranometer system itself does not consume too much power by needing to calculate a multi polynomial equation. To be able to test this function, it would be best to have a type of controlled experiment setup, to only regulate the solar irradiance.

4.3.1.3. Shielding the Sensor

The sensor or sensors that are used for measuring solar irradiance will need to be shielded against precipitation and wind, meaning that they need to be covered. However, when covering them the solar irradiance that would normally fall upon the sensors will be altered. This has to do with the transmission properties of the material that will be used when covering the sensor.
This means that for proper measuring of the ultraviolet, visible, and infrared light, a material needs to be chosen that does not affect the transmission of the wavelengths of interest. If the material does influence the transmission of certain wavelengths, then it should be accounted for by the processing of the data.

The transmissive properties of a sensor shielding material can also be used as an advantage like done in the research by Tohsing [34]. The sensor that was used by Tohsing was too sensitive and became saturated at high intensities radiated by the sun, and this resulted at clipping. Teflon was used as a filter to reduce the amount of radiation intensity, so that the sensor could be used again, without any clipping occurring. This did mean that lower intensities were more difficult to read.

An example is given by Vishay while designing the VEML6070 is ACRYLITE OP-4 sheets [81], which can already pass UVA light. This material allows light from UVA onwards to pass through. However, with the thickness of the material comes a drop off in light transmission.

4.3.1.4. *Quality of Measurements*

To ensure the quality of measurements taken by the sensor system, multiple things require extra attention. One of these was mentioned by one of the stakeholders: having accurate location and time data. This can be done with the use of a GPS. From the GPS module, both coordinates of the system and the time when the coordinates have been gotten will be retrieved, which can be used to accurately time the measurements.

Besides the location and time of the measurements, there are a couple of variables that every sensor has that contribute to the quality of the sensor: precision, accuracy, and resolution. The accuracy has been set utilizing a reference system, which the low-cost pyranometer should match or outperform. This accuracy has been set on 90 W/m^2. The resolution of the reference system is stated to be 1 W/m^2, which is also a requirement set by the KNMI. Nothing has been stated with regards to the precision in the documentation of the reference system.

4.3.2. Microcontroller

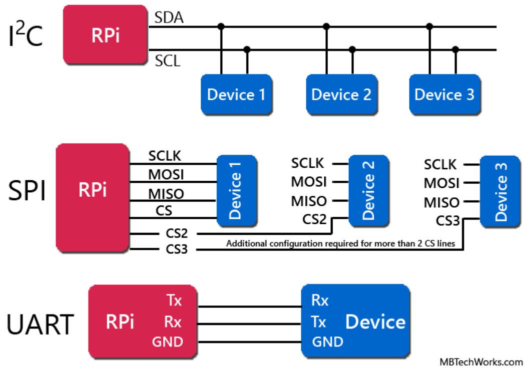


Figure 21: Types of communication – taken from mbtechworks.com [81].

There are three main parts that the microcontroller should be able to do, which are data sampling, data processing and data analysis.

The data sampling takes care of the transmission from the sensors to the microcontroller, and from the microcontroller to the wireless communication technique. This can be done in a variety of ways including the ADC, I2C, UART and SPI. These are shown in Figure 21 [82].

First, there is the option to use the ADC (Analog to Digital Converter) means that the output voltages of the sensors are converted to a digital signal that the microcontroller understands. The voltages will be divided into steps that are predetermined by the resolution of the ADC. This determines the size of the steps as well. Smaller steps lead to a higher resolution possible for reading the output of the sensor. To keep the quality of the data as good as possible, the resolution of the ADC should be equal to or better than the resolution of the sensor output. Furthermore, often the ADC is not calibrated in a linear function, so this should be corrected as well.

The second option for data sampling is I2C. This is a type of communication which uses a central bus to which all sub-systems could be connected. An advantage of this is that you don't need to take the resolution of the communication technique into account, resulting in precise measurements.

The third option is the use of SPI, a way of data sampling with multiple devices, like I2C. This does require more pins to be used of the microcontroller than I2C. I2C and PSI have the same benefits, meaning that no extra calibration is needed, like when using the ADC of the microcontroller.

The last option is UART. However, this way of data sampling is limited to only one device, so this way of communication is not of much interest to this research, as probably multiple sensors will be used.

4.3.3. Wireless Communication

From the Literature Research, it could be concluded that the combination of the microcontroller and wireless communication technique is a good way of making the system easier. Possible communication types, according to the Literature Review and State of the Art are Wi-Fi, IEEE 802.15.4, GSM/GPRS and LoRaWAN. The first two options are not feasible, as they require a lot of energy, with a small communication range. GSM/GPRS is a more viable option. However, for this technique, a subscription-based sim card needs to be used.
In the end, wireless communication was set in the preliminary requirements by one of the stakeholders to be LoRaWAN (Long Range Wide Area Network).
This type of communication is certainly possible in combination with a microcontroller, which already alleviates some of the wiring. LoRaWAN enables one to send small amounts of data over long distances with relatively low power consumption.

This type of communication does come with its limitations. The most widely used network is hosted by The Things Network (TTN). This provider does have additional restrictions besides the 1% duty cycle LoRaWAN gives. This means that you cannot have more than 30 seconds of airtime per day, and there are only a certain number of gateways to which the sensor node could connect to. However, for measuring solar irradiance in Enschede, there is enough gateway coverage. If this was not the case, there is also the possibility to make a gateway yourself to connect to the TTN network. This gateway is then used as a way for the sensor node to communicate with the TTN network.

Then there is also the matter of storing the data in a database. There is the option from TTN, which can store the data up to seven days for free. Besides this, there is also the possibility to use an online database, such as hosting a database on your computer, which then receives the data via a data-parser, such as Node-Red [83]. This can then be combined so that the values can be saved in a database such as InfluxDB [84]. This way you can easily obtain a database and use the option to graphically represent the data with Grafana [85]. There was an easier way by using a Cayenne integration in TTN.

4.3.4. Power Management

Fourthly, in the system, there is the main component of power management. This sub-system will make sure that all other sub-systems are provided with power. With this subsystem, there are two main things to take note off.

First, there is the supplying of energy. This can be done by simply attaching the system to a power socket; however, this does not make the system autonomous. As seen in the State of the Art and Literature Research, a lot of already existing solutions use a solar panel which charges batteries in its turn. Since the system will be out in the open, this would be a good way to generate electricity. It would be good to have the ability to monitor the charge level of the batteries. Furthermore, these batteries would need some type of protection circuit to make sure they do not overcharge.

Secondly, there is the preservation of energy. Meaning that sub-systems that require power, try to use as less as possible. This can be done by turning parts of the systems off when not in use, or simply by having an energy-efficient sub-system. Ways of turning sub-systems off are by the means of deep sleep in the microcontroller, or by using transistors to digitally turn off the sub-systems. This can be done with sub-systems that are not always in use, such as the sensor or the wireless communication sub-system.

4.3.5. Casing

Lastly, there is the casing for the low-cost autonomous pyranometer. This is an important aspect of the system as the sensors should be shielded against the environmental factors, but they should still be able to measure the incoming solar irradiance. This means that a transparent casing should be made which does not block the wavelengths that the sensors will measure.

4.4. Preliminary Requirements

The preliminary requirements are given by different stakeholders. These requirements mostly came out of the interviews. These requirements will later be prioritised with the MoSCoW model [68], and further divided into Functional and Non-Functional Requirements in the specification chapter. The preliminary requirements can be found in Table 3.

Preliminary Requirements	Source
The System Must	
Measure Solar Irradiance with an accuracy of at least 90 W/m^2	Richard Bults and Hans Scholten
Use GPS to gather location data and time data	Richard Bults and Hans Scholten
Use LoRaWAN communication technology	Richard Bults and Hans Scholten
Harvest energy and use this to power its sub-systems continuously	Richard Bults and Hans Scholten
Be able to check it’s functioning and data gathering.	Wim Timmermans
Send an average value for the solar irradiance every half hour	Wim Timmermans
Send LoRaWAN packages with a duty cycle of less than 1% and less than 30 seconds total per day	The Things Network
The System Should	
Take measurements every 12 seconds	Wim Timmermans, KNMI
Have the Solar Irradiance Sensors under the direct view of the air	KNMI
Be able to withstand temperatures from -10.1°C to 40.2°C	Environmental factors
Be able to withstand wind speeds up to 76 kph	Environmental factors
Be able to withstand precipitation up to 50mm per day	Environmental factors
Have the photodiode-based sensors shielded from precipitation	Environmental factors
Not damage the mounting place in any way	Municipality of Enschede
The System Could	
Be able to send a message when it is not functioning properly	Wim Timmermans
Save all measured values for a limited amount of time	Wim Timmermans
Have a way to offload all saved values.	Wim Timmermans
Measure Solar Irradiance with an accuracy getting close to 10 W/m^2	Wim Timmermans

Table 3: Preliminary Requirements for the systems, as taken from the interviews and the Ideation phase.

4.5. Conclusion

In this chapter, the stakeholders that influence the design and workings of the system have been identified and analysed. Furthermore, they were interviewed to get an overview of the different requirements that the system should meet. These initial requirements are classified with the help of the MoSCoW system. There were also additional requirements set by the extremities of weather variables that influence the working of the system.

Besides this, the possibilities within the design space were explored. These ideas were presented to the stakeholders which lead to the sensor consisting out of multiple sensors that each has its sensitivity in the electromagnetic light spectrum. These sensor outputs are then combined by way of a calibration function, which can be made by comparing the output of the low-cost sensors to the reference system.

The different sub-systems to the low-cost autonomous pyranometer to make it autonomous are the microcontroller, which will have an integrated way of communication. This communication will be LoRaWAN, as it has a very long reach, with little power consumption.
To power the system, a battery will be used that is charged with the help of a solar panel.

5. Specification

In the specification chapter, the requirements are tightened into a final set of requirements to work with when making functional prototypes in the realisation phase. The preliminary requirements that were gathered in the ideation chapter are elaborated more and analysed. To tackle each one, every sub-system of the low-cost autonomous pyranometer is discussed.

5.1. Final Requirements

The final requirements are concluded by the interviews with the different stakeholders. Furthermore, other requirements were encountered and specified during the ideation. These requirements are prioritised with the MoSCoW model, and further divided into Functional and Non-Functional Requirements, as found in Table 4 and 5.

Functional Requirements	Source
The System Must	
Measure Solar Irradiance with an accuracy of at least 90 W/m^2	Richard Bults and Hans Scholten
Send an average value for the solar irradiance at least every half hour	Wim Timmermans
Send LoRaWAN packages with a duty cycle of less than 1% and less than 30 seconds total per day	The Things Network
Take measurements every 12 seconds	Wim Timmermans, KNMI
The System Should	
Stick to the measurement frequency that is set by the KNMI	Wim Timmermans
Have a covering of UV-transmitting glass	Ideation
The System Could	
Be able to withstand temperatures from -10.1°C to 40.2°C	Environmental factors
Be able to withstand wind speeds up to 76 kph	Environmental factors
Be able to withstand precipitation up to 50mm per day	Environmental factors
Measure Solar Irradiance with an accuracy getting close to 10 W/m^2	Wim Timmermans
The System Won't	

Table 4: Final Functional Requirements for the systems, as taken from the interviews and the Ideation phase.

Non-Functional Requirements	Source
The System Must	
Harvest enough energy during the day and use this to power its systems during the day and night	Richard Bults and Hans Scholten
Be able to save Timestamp, GPS and measurement data to a database	Richard Bults and Hans Scholten
Use GPS to gather location data with a resolution of 1 meter and time data to 1 second	Richard Bults and Hans Scholten
Use LoRaWAN communication technology	Richard Bults and Hans Scholten
The System Should	
Be able to check it's functioning and data gathering.	Wim Timmermans
Be easily placeable	Ideation
Be placed in a place with a sky view	Ideation
Have the photodiode-based sensors shielded from precipitation	Environmental factors
Be able to send a message when it is not functioning properly	Wim Timmermans
Have a LoRaWAN availability of at least 33%	Wim Timmermans
Have the Solar Irradiance Sensor under the direct view of the air	KNMI
Use standards	Richard Bults and Hans Scholten, Ideation
Not damage the mounting place in any way	Municipality of Enschede
The System Could	
Save all measured values for a limited amount of time	Wim Timmermans
Have a way to offload all saved values.	Wim Timmermans
Be able to keep track of the battery level	Ideation
Have a colour with a high albedo value	Ideation
The System Won't	

Table 5: Final Non-Functional Requirements for the systems, as taken from the interviews and the Ideation phase.

5.2. Solar Irradiance Measurements

To properly measure the different types of solar irradiance, multiple sensors can be used.
There should be a sensor to cover the bigger part of every type of light. This means that there could be one sensor for UV light, and one to measure both visible and infrared light.
For every type of sensor, it would be best if it has a digital interface, as this makes sure that the system is not depended on the accuracy of the analogue to digital converter of the Arduino.

Adafruit makes quite a few good Visible light sensors, which also measure quite a significant portion of the Infrared light. Furthermore, there are also quite a good number of UV light sensors made by Adafruit, which need little interfacing. These sensors could be used to capture the incoming light across a spectral response from 240 to 375 and from about 360 to 1100 nm.

5.2.1. Sensor Testing

To know which types of sensors would give the best correlation between the reference system that will be used for calibration and the low-cost sensor module, a test has been executed.
Here the sensors were tested in an as close as possible real environment.
The sensors tested are SI1145 (UV, Visible Light, Infrared), GUVA-S12SD, TSL2591, BH1750 and the ML8511.

Figure 22 shows the different responses of the sensors, plotted against the Davis Solar Radiation sensor. From the output of the figures, it could be seen that the BH1750 did not have the range to cover the entirety of the range of the reference system. Furthermore, the TSL2591 sensor showed a linear behaviour up until about 850 W/m^2, where it started to linearly dip down again, this is shown on the next page in Figure 22. These sensors were subsequently removed from the viable sensors to use for the sensor. The other sensors were tested further, with the use of machine learning techniques, to be used as a low-cost pyranometer.

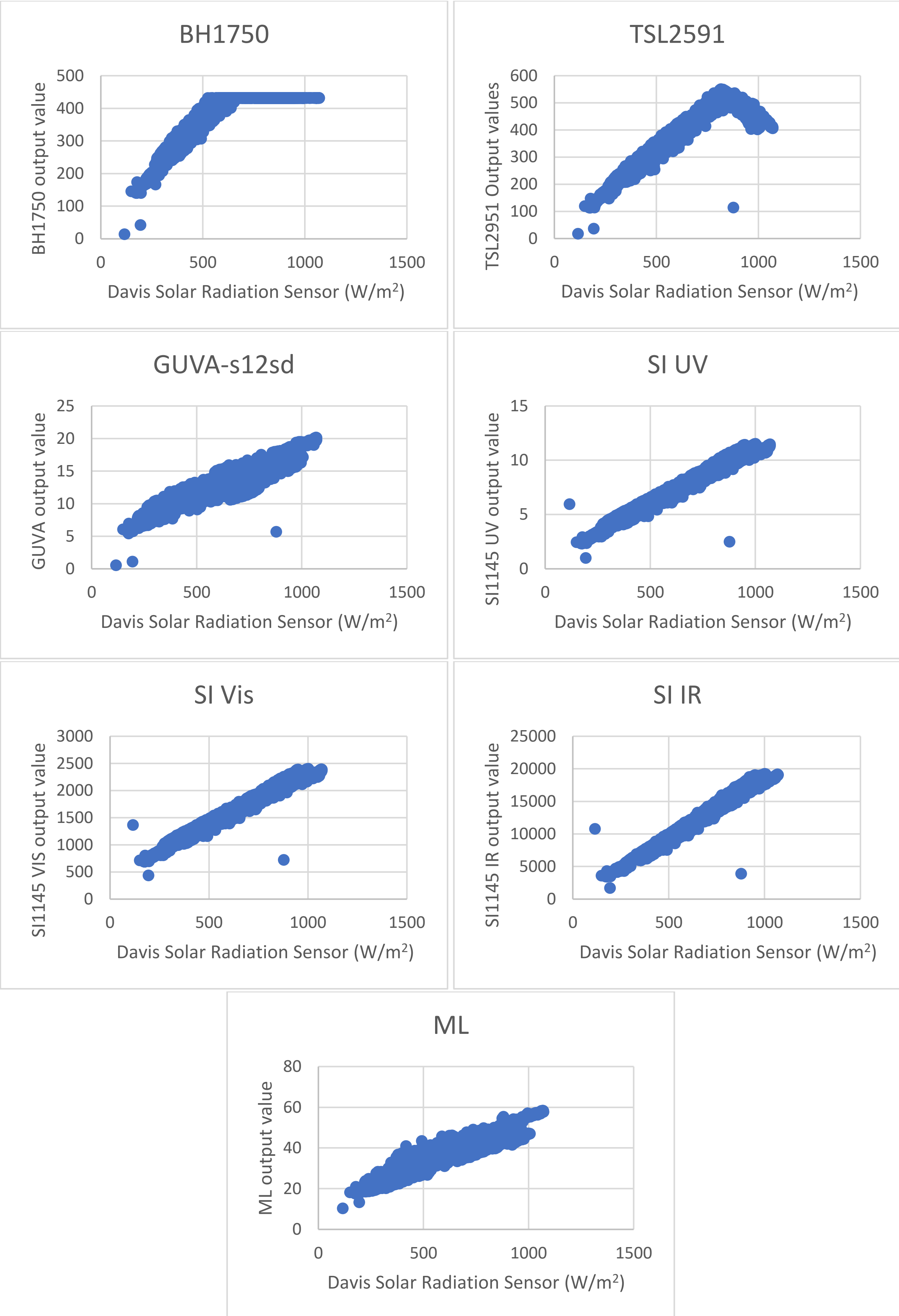


Figure 22: the output of the different sensors mapped against the output of the reference sensor.

5.3. Data Processing

The gathered data from the sensors will need to be processed to calculate the solar irradiance. Furthermore, the solar irradiance then also comes in a different variety of measurements, as given by the KNMI handbook.

5.3.1. Converting Sensor Data to Solar Irradiance

Concluding from the discovered ways to combine sensor data in Chapter 4.3.1.2., the best way to go about converting sensor data to solar irradiance is using a regression function. This is because this can be done beforehand, which means that the microcontroller can more easily calculate the solar irradiance when using the inputs of certain sensors, whilst still being low-power.

5.3.2. Calculating KNMI measurements

When the SAMPLE solar irradiance measurement has been retrieved, it is possible to calculate the other KNMI measurements as can be found in Chapter 4.1.5.2. To be able to send measurements over LoRaWAN to a database, it would be best to send the SAMPLE measurements [24] over LoRaWAN and use post-processing to calculate the other variables. These SAMPLE measurements are best to send, as it enables the fact that one can decide at a later stage what to do with each of the data points. However, it could also be beneficial to calculate averages. If, for example, SAMPLE measurements would get lost, the average cannot be calculated. Thus, it could be an advantage to calculate the averages on the system and sending these averages, instead of the SAMPLE measurements.

When the system should be able to have accurate data every 30 minutes, i.e. not real-time data as well, it is possible to calculate some values afterwards employing post-processing. This could, for example, mean that the SAMPLE and MINUTE measurements are done on the chip, but that the “ten minutes” measurements and hourly and daily measurements are done with the use of post-processing.

However, it would be best to only send the SAMPLE measurements, and later decide what to do with the data, as this enables users of the system to change the output afterwards more easily. This means that the system should send out SAMPLE measurements with the highest frequency possible.

5.4. LoRaWAN communication

In this chapter, the wireless communication protocol will be discussed. The pyranometer will be a sub-system of the Automatic Weather Station (AWS) designed by Jan-Paul Konijn, and as such the data communication of the pyranometer will be integrated with the AWS data communication. Therefore, this section describes the entire data communication of the AWS.
As ideated in the ideation the LoRa communication technique will be used. LoRaWAN offers a low power solution for sending small amounts of data over long distances. There are different networks with different specifications all over the world. The specifications [76] given in Europe are:

- The airtime of a given LoRaWAN node is limited to a duty cycle of 1%.
- The LoRaWAN communication works on a frequency of 863 to 870MHz, the common name for this is EU868.

For the infrastructure of the LoRaWAN a community-driven platform will be used: The Things Network (TTN). TTN is a community-driven provider that offers gateways to which a LoRaWAN node can be connected and as such data can be sent from the node to the gateway. From this gateway, the data is sent over the internet to the online service.
For this, TTN as a provider handles a set of rules and regulations to ensure fair usage. The most important being the following:

- Every LoRaWAN node can send messages for a total of 30 seconds per day.
- The frequency band used by the TTN is the EU868.1-869.525MHz.

As can be seen in the restrictions posed by the LoRaWAN communication technology and the TTN regulations, it is important to calculate the amount of data that can be sent. This will be done in the next section.

5.4.1. The Things Network Timing Calculations

To help calculate the overall time that a LoRaWAN node is sending messages, the TTN offers a service called the TTN LoRaWAN airtime calculator [86]. Here, one can see that when working with LoRaWAN, four different parameters influence the transmission time:

- Payload in the number of bytes. For the EU868 network, there is a maximum payload size of 51 bytes
- Spreading Factor, see Figure 23 [87]. The value ranges for the spreading factor range from SF7 to SF12. A higher Spreading Factor provides the gateway more opportunity to sample the signal power, increasing the sensitivity, thus the likelihood of the data being received. However, a higher spreading factor takes a longer time to send and has a higher power consumption. Thus, using a low spreading factor would be best for the battery life of the system, as well as the amount of data that can be sent.

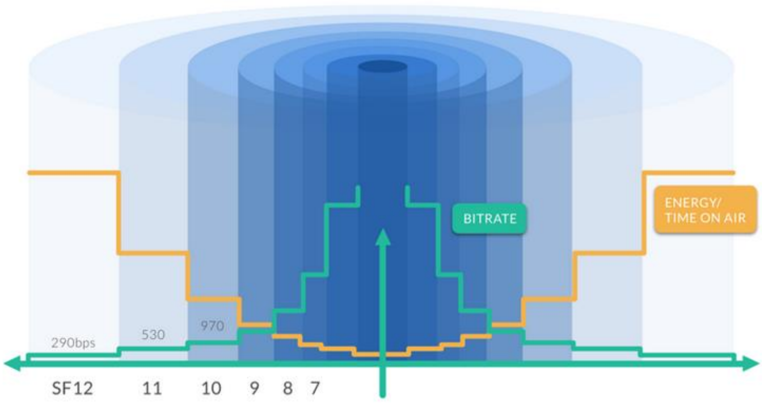


Figure 23: Spreading Factors – taken from TTN [85].

- Region. The region indicates the main frequency being used. For this GP, this will be EU868. The frequency deviates from 868 to 870 MHz
- Bandwidth. The possible bandwidth for EU868 is 125 and 250 kHz [88]. A higher bandwidth results in the possibility to send more data. However, the microcontroller library that is in use, LMIC, does not allow to change the bandwidth, therefore, the 125kHz bandwidth is a given.

The first calculation to be made is the overall restrictions given by the LoRaWAN 1% duty cycle.

$$T_{Off\ Subband} = \frac{T_{On\ Air}}{DutyCyle} - T_{On\ Air}$$

Equation 1: Calculating the Time off the sub-band

For calculating the off time of a sub-band, Equation 1 is provided. The equation provides the amount of time that a device is not allowed to send messages for a sub-band after sending on that sub-band. The airtime of the node per message will be calculated with the help of the TTN LoRaWAN airtime calculator [86]. Filling in the airtime calculator with the appropriate parameters (51 bytes, SF7, EU868, and 125 kHz), gives that the entire message takes 118ms to send.
Using this 118ms together with the 1% duty cycle and filling in Equation 1, the off-time should be 11.7s. This would mean that about every 12 seconds a message could be sent on that sub-band.

Besides the restrictions given by the LoRaWAN communication technology, the TTN also has the restriction of maximum airtime of 30 seconds per node per day.
When using this restriction, the 30 seconds can be divided by the earlier found 118ms to find that 254 messages can be sent.
As the regulation given by the TTN is more restrictive, this will be the regulation that will be worked with.

With the knowledge that 254 messages can be sent, each consisting out 51 bytes, it can be concluded that 12954 bytes can be sent per day, resulting in 103632 bits per day as every byte contains 8 bits.
This can also be used that about every 6 minutes a message can be sent. However, together with the stakeholders, it was decided that every 10 minutes a message would be sent, containing the ten 1-minute averages of every variable.
The exact timing calculations can be found in Appendix C.

5.4.2. Payload Structure

As the maximum payload that is possible for the communication technology is 51 bytes, the type of data that should be sent must be limited to the essential data. The data will need to include location data, time of measurement and the measurement data itself. For the location data, the GPS latitude and longitude are most important, and the height does not matter as much. As mentioned by Wim Timmermans in his interview, the height does not influence the solar irradiance measurement much. There are two ways to go about sending the data, using an existing standard method, or writing the payload structure.

5.4.2.1. Cayenne Payload Structure

The first option for sending messages is the standard Cayenne Low Power Payload (Cayenne LPP). This means that one can use existing libraries to send data over LoRaWAN.

The advantage of using Cayenne LPP is that it is a set way of communication, resulting in the fact that others are easily able to read the messages. Also, the TTN console provides an existing decoder for Cayenne making it easy to use on the server-side as no custom decoder is required. Next to that, identifier bytes are used to identify the different variables that can be sent with Cayenne LPP. These variables can be seen in Table 6.

Type	LPP	Hex	Data Size	Data Resolution per bit
Digital Input	0	0	1	1
Digital Output	1	1	1	1
Analog Input	2	2	2	0.01 Signed
Analog Output	3	3	2	0.01 Signed
Illuminance Sensor	101	65	2	1 Lux Unsigned MSB
Presence Sensor	102	66	1	1
Temperature Sensor	103	67	2	0.1 °C Signed MSB
Humidity Sensor	104	68	1	0.5 % Unsigned
Accelerometer	113	71	6	0.001 G Signed MSB per axis
Barometer	115	73	2	0.1 hPa Unsigned MSB
Gyrometer	134	86	6	0.01 °/s Signed MSB per axis
GPS Location	136	88	9	Latitude: 0.0001 ° Signed MSB
				Longitude: 0.0001 ° Signed MSB
				Altitude: 0.01 meter Signed MSB

Table 6: Exploring the Cayenne LPP variables. Taken from Cayenne Docs.

This is also where the disadvantages of Cayenne LPP are. Every variable that is sent, has two extra bytes, one being an identifier, one being a channel. Lastly, not every byte is fully used, leaving some bits empty or unused. An example of this is, that the Temperature uses 2 full bytes, meaning that it has a range of 0 to 65535. With a 0.1 °C resolution this means that a range of about 0 °C to 6553.6 °C can be implemented. This is only done because using one byte gives a range of 0 - 255, thus 0 °C to 25.5 °C is not enough. As can be seen this results in a lot of extra range which is not needed.

5.4.2.2. Unique Payload structure

The second option for sending messages over LoRaWAN is designing the payload structure. The benefit of this is that bytes can be fully utilised by the system, meaning that higher measurement frequencies can be reached. The full potential of the bytes can be used when they are divided into loose bits, and the bits are assigned to variables. A disadvantage is that such a payload structure is not widely known, and thus requires to described very clearly.

To maximise the measurement frequency, the designing of a unique payload structure will be chosen. First and foremost, the different types of data that should be sent are identified. For the entire Meteorological Measurement Station as designed by Jan-Paul Konijn, these are wind speed, temperature, humidity, and solar irradiance.

In Table 7, these variables are further explored to find out how many bits should be used, to cover the range and resolution of the measurements when sending the data of these variables.

Variable	Range and Resolution	Start Measurement	Start Measurement Mapping	Delta Measurement	Delta Measurement Mapping
Wind speed	Range: 0 – 50m/s Resolution: 0.1m/s	9 bits (0 – 511)	0: 0m/s 500: 50.0m/s	9 bits (0 – 511)	0: 0m/s 500: 50.0m/s
Temperature	Range: -30 – 60 °C Resolution: 0.1 °C	10 bits (0 – 1023)	0: -30.0°C 900: 60.0°C	6 bits (0 – 63)	0 – 62: -3.1°C – +3.1°C 63: error
Humidity	Range: 0 – 100% Resolution: 1%	7 bits (0 – 127)	0: 0% 100: 100%	4 bits (0 – 15)	0 – 14: -7% – +7% 15: error
Solar Irradiance	Range: 0 – 1800 W/m^2 Resolution: 1 W/m^2	11 bits (0 – 2047)	0: 0 W/m^2 1800: 1800 W/m^2	11 bits (0 – 2047)	0: 0 W/m^2 1800: 1800 W/m^2

Table 7: The variables to be sent. Here the different mappings that will be used, can be seen.

As can be seen in Table 7, the delta compression is not used for every variable. This is because the wind speed, as well as the solar irradiance, can differ a lot in a small amount of time. This does not occur as much for the temperature and humidity, and thus delta compression is used for these variables. As can be seen in Table 7, the number of bits used for temperature and humidity is decreased respectively with 4 and 3 bits, which may not seem like much, but when used for 9 measurements, sums to a lot.

In Table 8, these variables are further explored to find out how many bits should be used, to cover the range and resolution of the measurements when sending the data of these variables. With the data structure as described in Table 8, five SAMPLE measurements are combined into one measurement every 60 seconds. When there are ten such measurements, they will be sent over LoRaWAN to TTN. This is done with the following data structure, presented in Table 8. This payload consists of 51 bytes, thus in 408 bits to be divided.

Variable	Sub-Variable	Number of bits
Time Stamp (Unix Time)		32 bits
GPS	Longitude (0.00001°) (around 1 m [89])	24 bits
	Latitude (0.00001°) (around 1 m [89])	24 bits
System logging	0-255 system codes	8 bits
Start Measurement	Windspeed	9 bits
	Temperature	10 bits
	Humidity	7 bits
	Solar Irradiance	11 bits
Delta Measurement (x 9)	Windspeed	9 bits
	Temperature	6 bits
	Humidity	4 bits
	Solar Irradiance	11 bits
Total		395 bits out of a possible 408

Table 8: Payload structure of one message. There are some bits left, these will be put at the end of the message.

5.5. Power Usage and Generating

To properly power the entire AWS of which the solar irradiance sensor is a sub-system, it would be good to have an overview of the power requirements of the system. In this chapter, the power delivery and the power consumption are brought under attention.

5.5.1. Power Delivery

The power will be supplied by a battery that is charged with a solar panel. These batteries will most likely be a type of 18650 battery which have a high maximum discharge current (a minimum of about 2.5A) and have a discharge voltage of about 4.2 to 2.7V dependent on the type of battery used.

There is, however, quite a difference in the charging current for these batteries. An often-chosen circuit to charge 18650 batteries is the TP4056 charging circuit which can charge a 18650 battery with maximum 1A. It should thus be possible to charge the battery with up to 1A of current.

The solar panel that would be used should, for optimal usage, have a 5W output, as this would then lead to optimal usage of the TP4056 charging circuit to charge the batteries. This 5W would, however, be a maximum rating and in real life will not be as ideal. It would probably result in lower average power delivery.

5.5.2. Power Consumption

There are multiple parts in the system that need the power to function. In general, these include the microcontroller, the GPS module and the different light sensing modules that are used as solar irradiance sensor.

The microcontroller that will be used is a TTGO ESP32 Lora with OLED microcontroller. This is because this microcontroller has a built-in LoRaWAN communications module and is quite inexpensive as compared to other LoRaWAN microcontrollers. Depending on which devices are used and which are not, i.e. the OLED is not needed, when awake the microcontroller uses about 50mA. When using LoRa about 80mA and when in sleep it uses about 10mA [90]. Then there are the MOSFETs, such as the BS170, in the off state it uses only a maximum of 10nA [91]. These MOSFETs will be used to turn different sub-systems off and on, depending on whether they are needed or not. The GPS module which will be used is the NEO-6m. The absolute maximum power consumption of this GPS module is 67mA [92].

Then there are the solar irradiance sensor modules. The viable sensors are BH1750, ML8511, SI1145 and the TSL2591. The BH1750 has a power dissipation of 260mW [93]8, resulting in about 52mA. The ML8511 consumes about 500uA [94]. The SI1145 module uses about 5.5 mA when actively measuring and 0.5mA when in standby mode [46]. The TSL2591 module uses about 0.4mA when actively sensing and 5uA when in power-down mode [42].

To get a good overview of the power consumption of the entire node, it is good to measure the current draw when the node is in use. However, an overview of the components can be found in Table 9.

Component	Active power consumption (3.3V)	Inactive power consumption (3.3V)
TTGO ESP32 LoRa	80mA (LoRa) 50mA (no LoRa)	10mA
MOSFETs	30mA	10nA
NEO-6m	67mA	11mA
BH1750	79mA	1uA
ML8511	500uA	1uA
SI1145	5.5mA	0.5mA
TSL2591	0.4mA	5uA

Table 9: Power consumption of the different components that will be used in the overall system.

5.6. Shielding the sensors

The sensors that probably will be used, as they were the most accurate according to the first prototype, are the SI1145 [46] and the ML8511 [94]. Table 10 shows all types of sensors that are used by these sensor modules and their respective spectral response. Here it can be seen that the SI1145 does not have a UV sensor, but it approximates its value on the visible light sensor and infrared sensor [95].

Sensor module	Sensor	Spectral response
SI1145	Visible light sensor	400nm – 800nm (centred on 530nm)
	Infrared light sensor	550nm – 1000nm (centred on 800nm)
ML8511	UV sensor	280nm - 440nm (centred on 370 nm)

Table 10: the spectral response of the to be used sensor modules.

This means that the cover that is used, should be able to transmit wavelengths from 280nm to 1000nm. Another alternative is to use different types of cover for every sensor. The advantage of using this would be that the covers could be optimised for every separate sensor.

A good option for a window that the sensors could be covered by is given by Vishay in their “designing the VEML6070” report [81]. Here a material ACRYLITE OP-4 is mentioned which has good transmissive properties from about 280nm. This is also shown in the document by CYRO industries which is the producer of this material [96]. This type of acrylic material is often referred to as Ultraviolet Acrylic. The transmissive properties can be seen in Figure 24 [96].

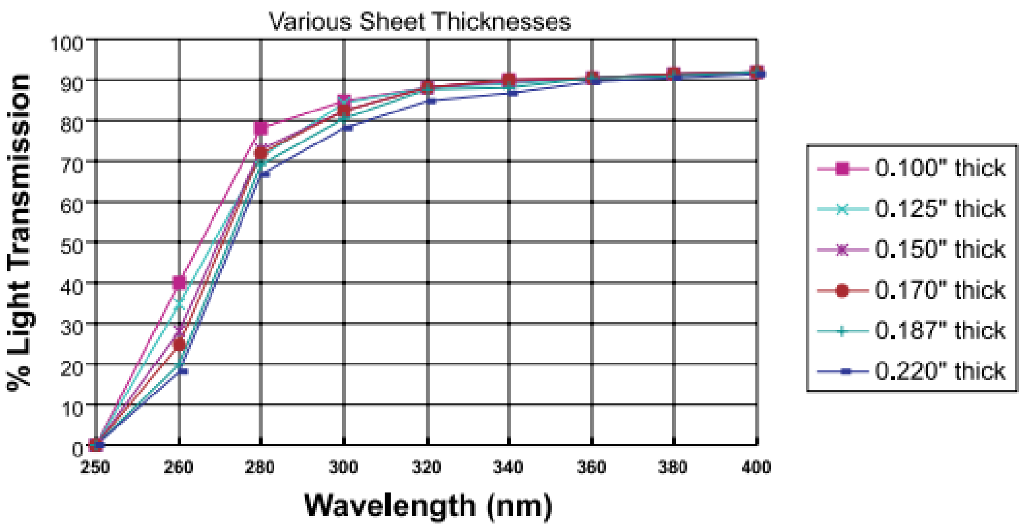


Figure 24: The transmissive properties of ACRYLITE OP-4 [94].

Besides these types of acrylic material, there is also Silica-based glass. This is what the domes on thermopile-based pyranometers are made from. These are often special UV fused Silica materials. This Spectral responsivity can also be seen in Figure 25. These types of materials are more often found under the term UV-transmitting glass.

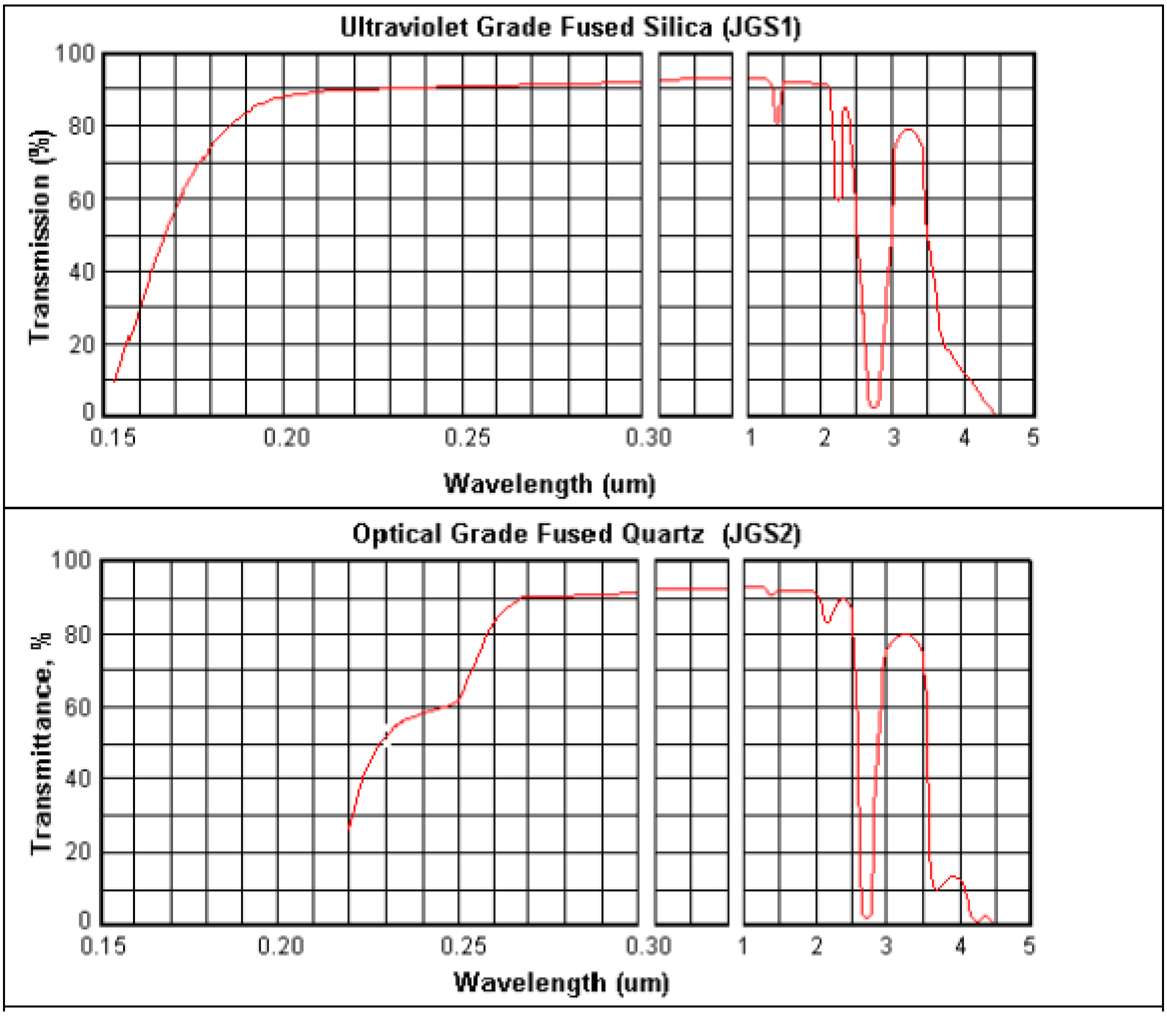


Figure 25: The transmissive properties of 4mm thick silica-based glass.

5.7. System Architecture

This section describes the low-cost autonomous pyranometer and its sub-systems.
To get a good overview of the different functionalities and the different connections between sub-systems, a functional architecture diagram can be made. This is done for both the hardware and the software of this system.

5.7.1. Hardware Architecture

These systems are broken down in components to be easily distinguished. The system hardware architecture can be seen in Figure 26. This diagram shows the hardware of the autonomous low-cost pyranometer and its sub-systems. Here one can also see the flow on energy and data between the different sub-systems.

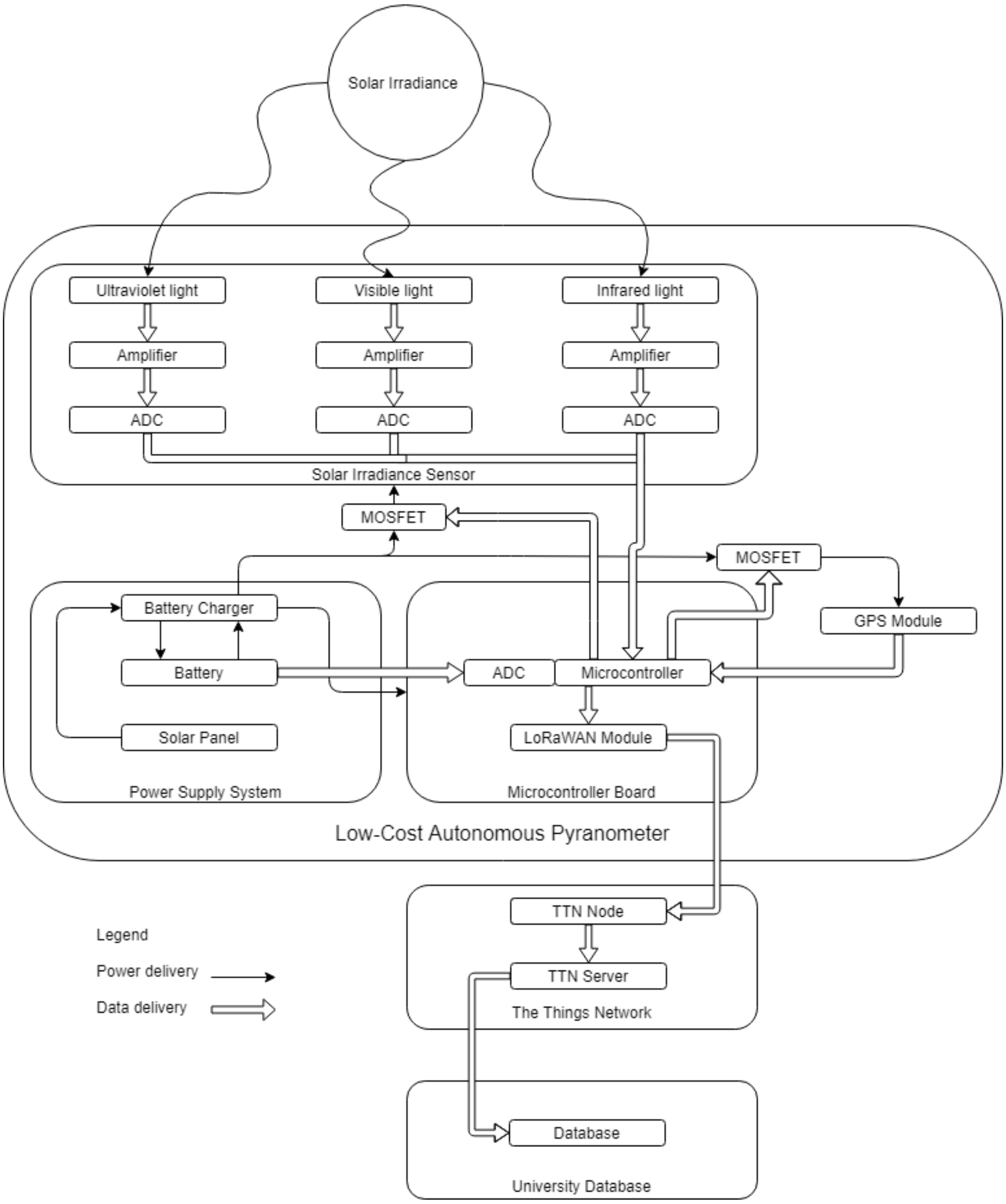


Figure 26: The hardware of the low-cost autonomous pyranometer and its sub-systems.

5.7.2. Software Architecture

These systems are broken down in components to be easily distinguished. The system hardware architecture can be seen in Figure 27. This diagram shows the flow of the software that the microcontroller will follow.

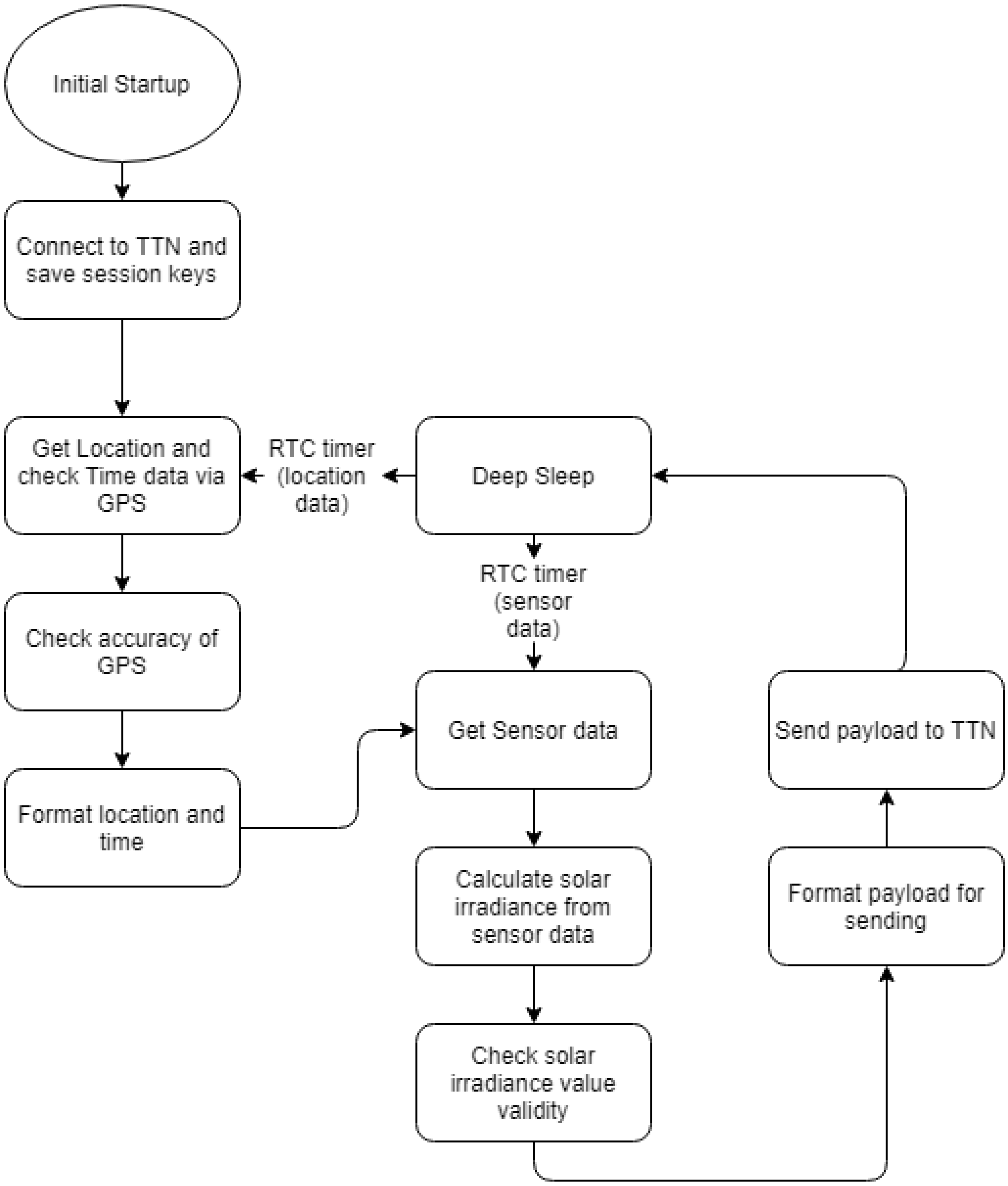


Figure 27: The software diagram of the low-cost autonomous pyranometer.

6. Realisation

In this chapter, the realisation phase of this project will be discussed. This phase consists out of 4 iterations of the prototype, where the focus of the prototype is not only on the chosen hardware but also on the software and the Multiple Linear Regression algorithm. The functional evaluation of each component is given to portray what the next prototype will focus on, but the overall evaluation is given in Chapter 7., the evaluation phase.

6.1. First Prototype

The first prototype was used for the selection of the different light sensors and pick out the ones that have a sufficient range and the best response. A picture of the setup can be seen in Figure 28.

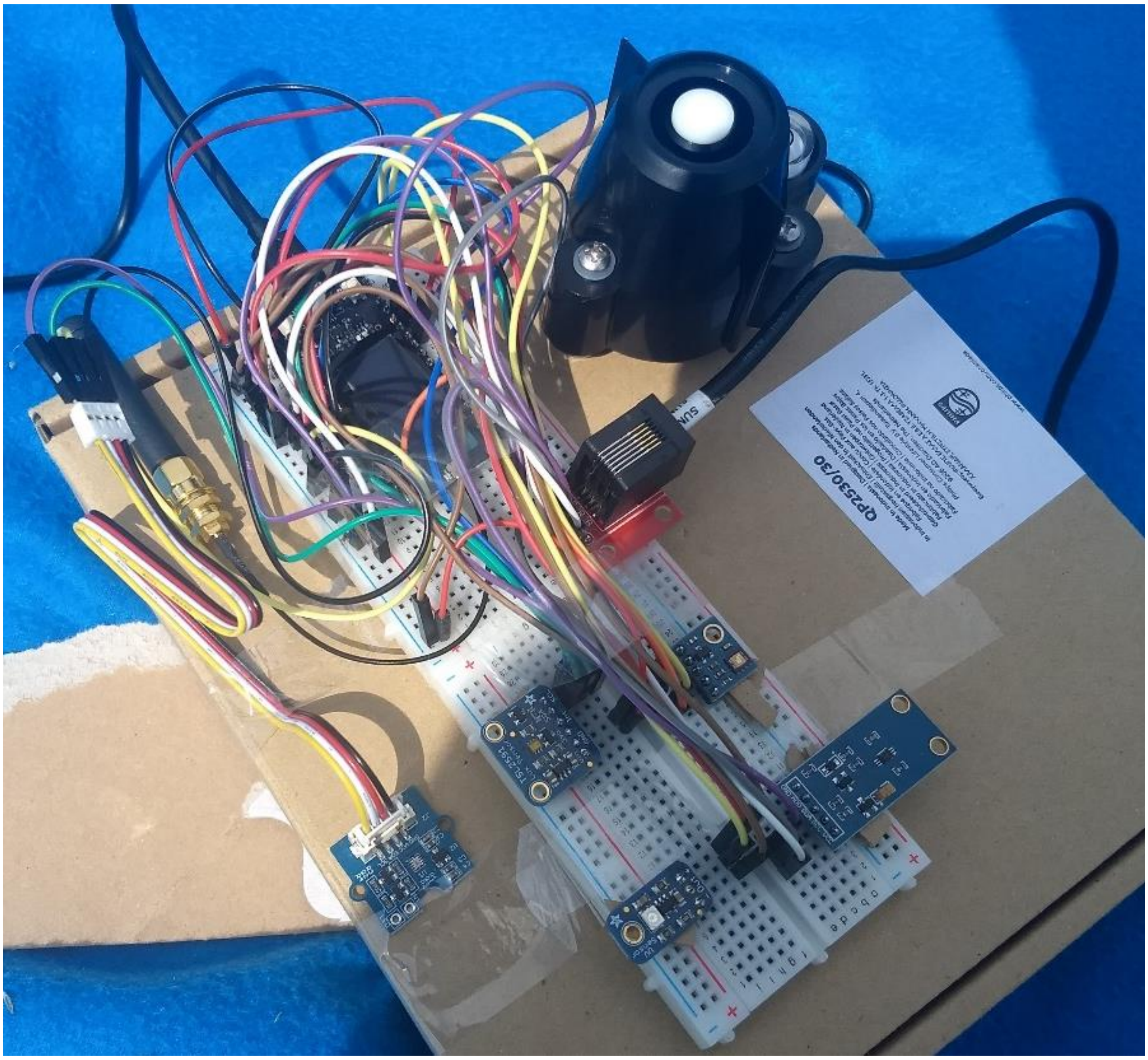


Figure 28: The setup of the first prototype. All available low-cost sensors were attached for testing.

6.1.1. Microcontroller

The microcontroller used for the first test is the TTGO LoRa32 as this is a microcontroller with LoRa capabilities on the board itself.
This microcontroller was also chosen to do adequate testing with the different types of communication interfaces on the chip itself to make sure they are functioning properly, for all sensors. These communication interfaces are things like I2C and the ADC of the ESP32.

6.1.2. Sensor interfacing

The different sensors were interfaced to the microcontroller via a breadboard and jumper wires. This may not be the most stable connection, however, for the first prototype and the selection of different sensors this sufficed. The full schematic is shown in Figure 29.

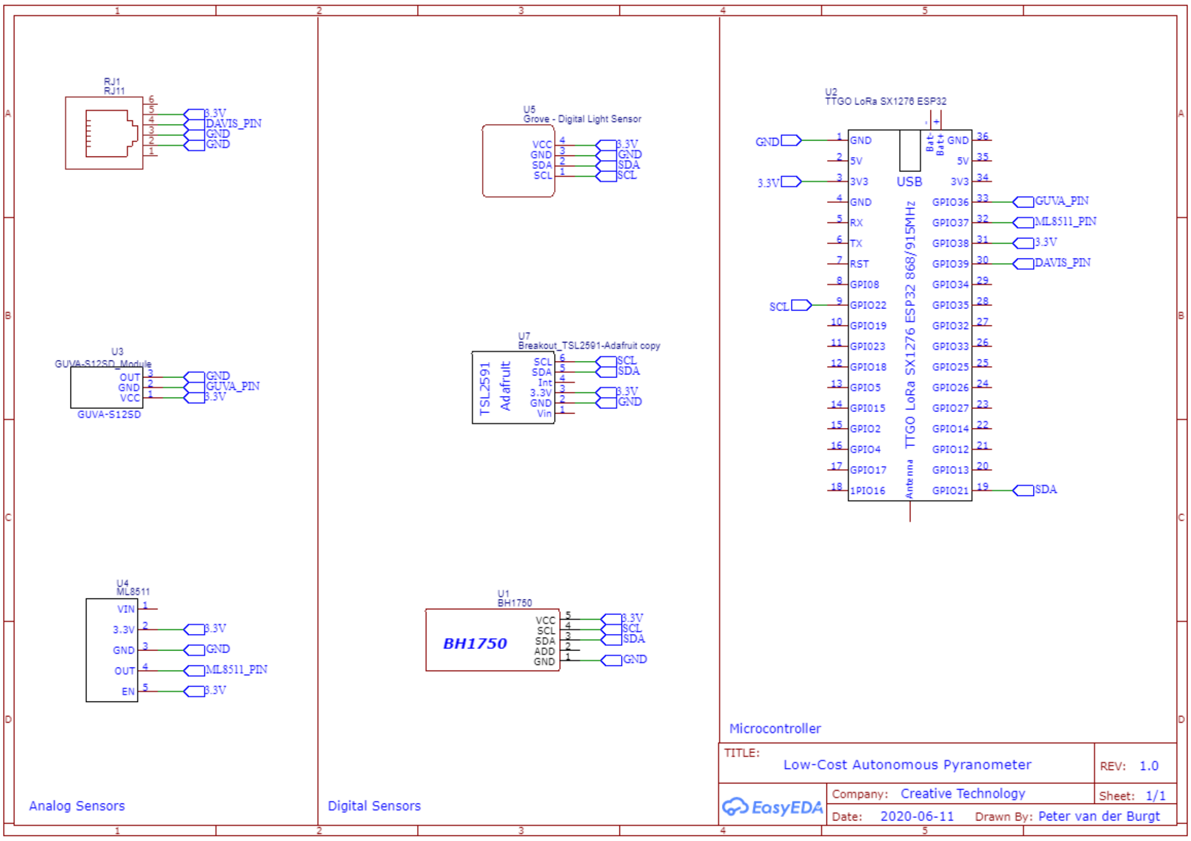


Figure 29: The full schematic of the first prototype.

6.1.2.1. BH1750

The BH1750 light sensor works with I2C so subsequently the I2C pins of the microcontroller were used to interface with this sensor. Because of this, the BH1750 library could be used to read the Lux levels that the sensor measures more easily. This can then be converted in solar irradiance W/m^2 by multiplying the levels in Lux with 0.0079 [97].

6.1.2.2. GUVA-s12sd

The GUVA-s12sd sensor has an analogue voltage output. This can be read by an analogue input pin on the microcontroller. In the software, this voltage can be converted to solar irradiance. This is done by first calculating the voltage and converting that to the current in the light diode in μA as stated in the datasheet. This can then be converted to W/m^2.

6.1.2.3. GY-ML8511

The GY-ML8511 sensor has an analogue voltage output. This can be read by an analogue input pin on the microcontroller. In the software, this voltage can be converted to solar irradiance. This can be done by mapping the voltage with a function as stated in the datasheet by the microcontroller, to finally get a solar irradiance value in W/m^2.

6.1.2.4. Grove SI1145

The Grove SI1145 sensor uses I2C to connect to the microcontroller and can thus be added to the I2C bus. Using the SI1145 library written for Arduino, one can easily measure the UV, visible and infrared light with this sensor. This is respectively returned as a value for the UV index and in the amount of Lux.

6.1.2.5. Adafruit TSL2591

The Adafruit TSL2591 sensor uses I2C to connect to the microcontroller and can thus be added to the I2C bus. The Adafruit library can be used to retrieve the amount of light in Lux. This can subsequently be converted to solar irradiance by multiplying the Lux value with 0.0079.

6.1.2.6. Davis Solar Radiation Sensor

The Davis Solar Radiation Sensor has an analogue voltage output and can thus be read by the microcontroller by using the onboard ADC. The voltage that is read can be converted to the solar irradiance by dividing with 1.67 according to the datasheet [98].

6.1.3. Testing

For this prototype, it was important to look at the performance of the individual sensors, as well as explore the possibilities of the Multiple Linear Regression (MLR) algorithm as implemented by the Excel Analysis Toolpak. As described in Chapter 5.2.1., the BH1750 and the TSL2591 did not have the right range or a linear response when exposed to sunlight and were thus ruled out for further testing.

To test the different combinations of sensor values for the MLR algorithm, the SI1145, GUVA-s12sd and ML8511 were explored further. The results are shown in Table 11.

Sensor	R^2 /Adjusted R^2	Standard Error	Significance F	P-value
SI1145 UV	0.996994	37.35138	0	0
SI1145 Visible	0.991609	62.40672	0	0
SI1145 Infrared	0.998124	29.50717	0	0
GUVA-s12sd	0.981610	92.31353	0	0
ML8511	0.978501	99.89048	0	0
SI1145 (UV, Vis, IR)	0.997857	29.42221	0	UV: 0.734374 Vis: 0.073601 IR: 1.4E-216
SI1145 (UV, Vis, IR) And GUVA-s12sd	0.998788	20.82036	0	UV: 7.6E-197 Vis: 3.3E-106 IR: 0 GUVA: 0
SI1145 (UV, Vis, IR) And ML8511	0.999118	16.74118	0	UV: 9.16E-14 Vis: 0 IR: 0 ML: 0

Table 11: The sensors tested with the regression algorithm.

In Table 11, the most important outputs, as mentioned in Chapter 3.8.2.1., of the regression algorithm are shown. These tests were taken from 14:06:24 to 16:06:30 for a total of two hours. The weather was relatively nice, as there was no rain, and there were quite clear skies, with sometimes clouds.
The overall measurements by the reference pyranometer and the Last entry of Table 11 can be seen in Figure 30.

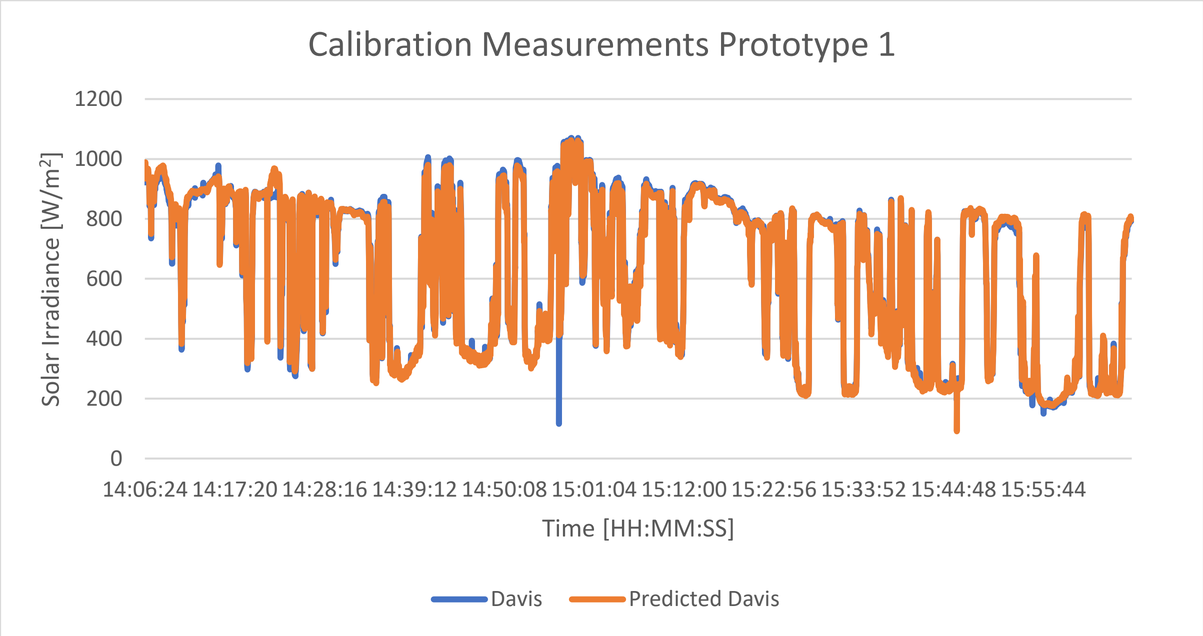


Figure 30: the calibration measurements taken for the first prototype.

Taking a measurement every 2 seconds resulted in a total amount of 3604 measurements. These measurements were taken every 2 seconds, to get a high amount of data points, as more data points would work better to train the Multiple Linear Regression Model. Furthermore, the 2 seconds were chosen since it would give a representable sample frequency, as within a small time the solar irradiance can differ within a short time due to clouds covering the sun.

As can be seen from Table 11, the setup where all three variables from the SI1145 are used, in combination with the ML8511 performs the best. This means that the SI1145 and the ML8511 seem to be the best combination of sensors to use for the low-cost autonomous pyranometer.

6.2. Second Prototype

The second prototype that was made, used the knowledge gained from the first prototype. This meant that the chosen sensors were included and the way of calibrating the sensors was used. Although, due to a change in the way of interfacing and connecting the sensors, this was not fully tested. However, knowledge can still be gained by the fact that the electrical circuitry of this prototype was functional, and this way of interfacing and connections can still be used. The prototype can be seen in Figure 31.

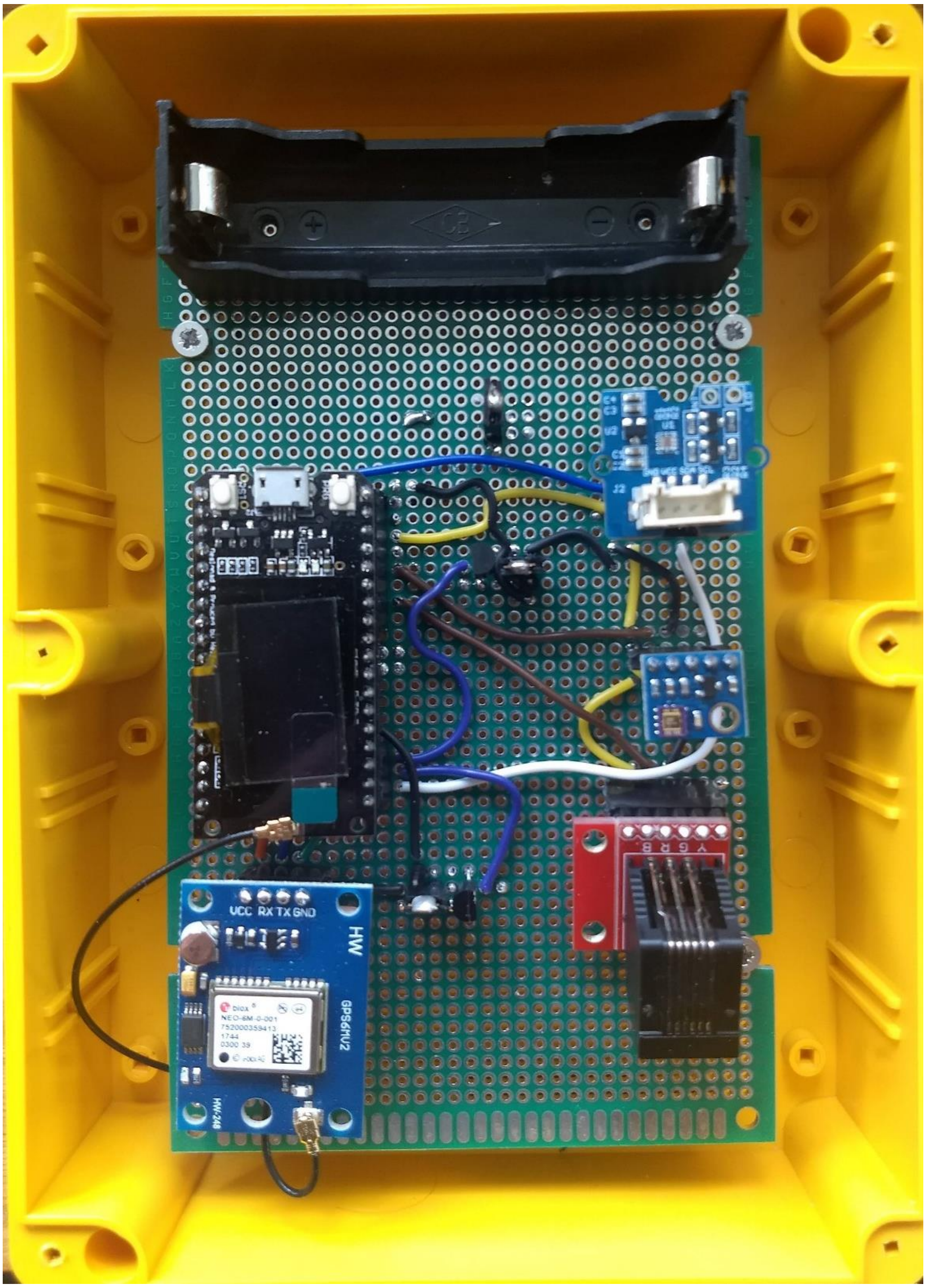


Figure 31: The setup of the second prototype. The Solar Irradiance sensors and the GPS sensor can be seen here, besides the LoRaWAN microcontroller. The battery was not inserted.

6.2.1. Microcontroller

In the second prototype, the microcontroller stayed the same as it was in the first prototype, so this did not lead to any unexpected changes. It functioned well in the first prototype, and thus the microcontroller was used again. However, in this stage of the realisation phase, the microcontroller USB connector started to falter. It was possible to program the microcontroller and use the serial monitor to check on the microcontroller. This meant that the faltering USB connector did not have any influence on the testing of components.

6.2.2. Sensor interfacing

The light sensors that were continued for the second prototype were the SI1145 and the ML8511. The connections of these used sensors did not change with regards to the first prototype, as they functioned well. No changes were thus made in the electrical circuit.
Besides the low-cost light sensors, the Davis Solar Radiation Sensor was also integrated into the new prototype to use as a reference sensor.
Next to the light sensors, a new sensor was added to the system. A GPS sensor is used to get location and time data.
Apart from that, MOSFETs were added for switching the sub-systems on and off.
The full schematic can be seen in Figure 32.

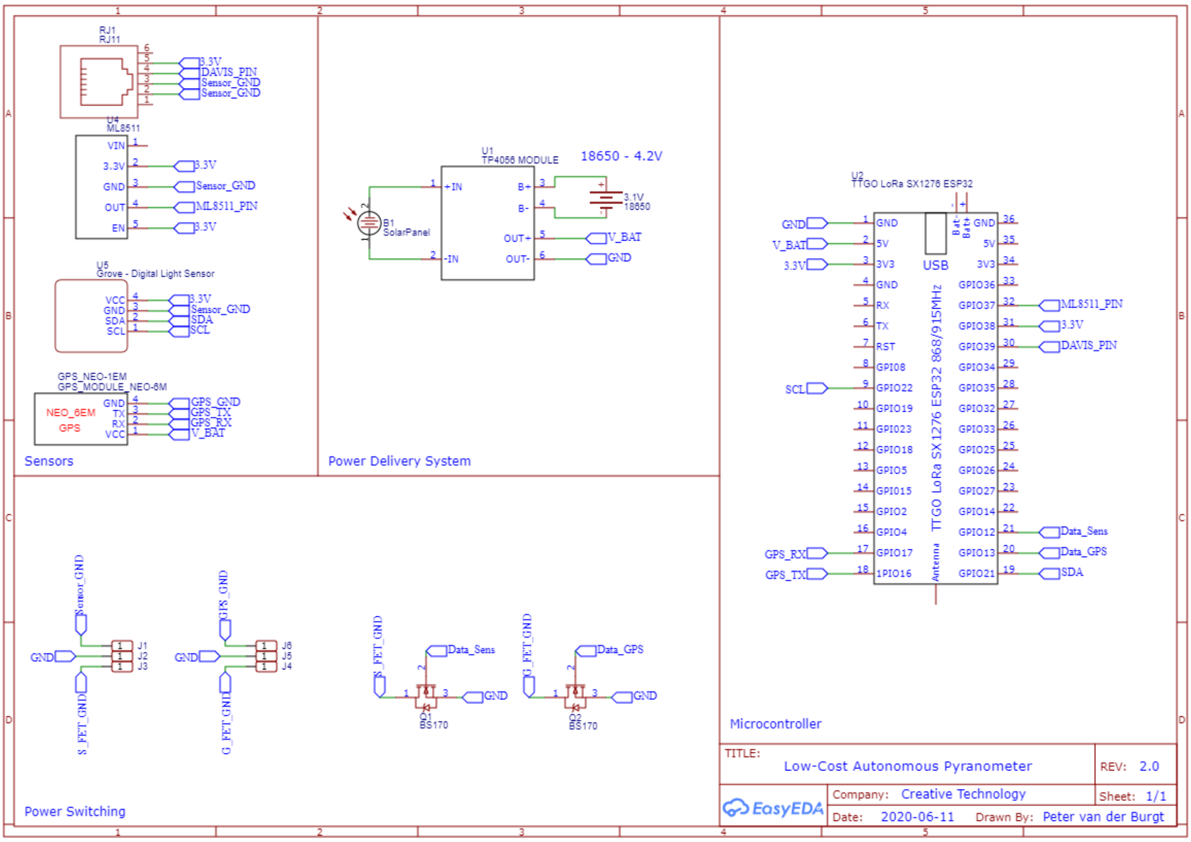


Figure 32: The full schematic of the second prototype.

6.2.2.1. Neo-6m GPS sensor

The Neo-6m GPS sensor was used to retrieve accurate location and time data. It can be interfaced via UART to the RX and TX pins of the microcontroller. With the help of the TinyGPS++ library, the incoming data stream can be parsed from the NMEA format into time and location data to increase the quality of measurements.

6.2.2.2. Power management
For power management, MOSFETs were added to be able to turn off and on the different systems when it's needed. These were attached to digital GPIO pins of the microcontroller, which thus was able to turn the different sub-systems off and on.
There is also a configuration of jumpers set up to bypass the MOSFETs and directly pull the system to ground, turning the system directly on.

6.2.3. Casing
The casing for the second prototype was an ABS plastic casing with a clear top. This clear top made it possible for the light to be measured inside of the casing by the light sensors.
Furthermore, the casing has an IP value of 65, which should make it dust-tight and protect it against water projected from a nozzle [99]. This makes it suitable for outdoor usage.

6.2.4. Testing
Functional tests were done for this prototype to see if all features worked properly.
Two main things were tested: The LoRaWAN connection and reliability to send data to The Things Network (TTN), and the different sensors that are used in this prototype.

6.2.4.1. Sending Data to TTN
The LoRaWAN connection to send data to TTN has also been tested. There were some interesting observations on the reliability of sending the data.
For instance, the data would be received better when the system was higher up and outside. Furthermore, the line of sight also seems to be important. This means that there should be no big obstructions in the way of the low-cost autonomous pyranometer and the gateway of the LoRaWAN network.

6.2.4.2. Sensors
The low-cost light sensors behaved as they should and worked as expected. It became apparent that the Davis Solar Radiation sensor did work, but the ADC of the ESP32 did cause a rather large discrepancy in the measured voltage and the actual voltage. This was further confirmed by the research of Max Pijnappel [54], and others [100], [101]. As can be found here, the ESP32’s Analog to Digital Converter suffers quite a bit from non-linear behaviour. This does not seem to come from certain circumstances but seems to be a known error that can be fixed by using so-called error correction functions and lookup tables.

The GPS sensor worked but did take about 30 seconds to get a fix on time and another 200 seconds to get a fix on the location. However, when the GPS sensor was on for a longer time and was disconnected for a short time, the on-board battery on the module ensured the data was saved. This resulted in both time and location fix within 3 seconds after booted again. This is caused by the fact that the GPS can be switched off via software, where the GPS sensor keeps the RF part working and switches of the embedded processor, resulting in quick time and location fix. This does, however, consume small amounts of power and is thus less efficient than using a MOSFET to shut the GPS sensor off.

6.3. Third prototype

After the construction of the second prototype, it was decided that the solar irradiance sensor should be able to interface the system being made by Jan-Paul Konijn. This means that this new prototype should emulate a connection just like the sensor that he is using. This would make it easily interchangeable and so-called plug and play. The sensor that Konijn uses in his system is the same as the reference sensor, the Davis Solar Radiation sensor. The third prototype can be seen in Figure 33.

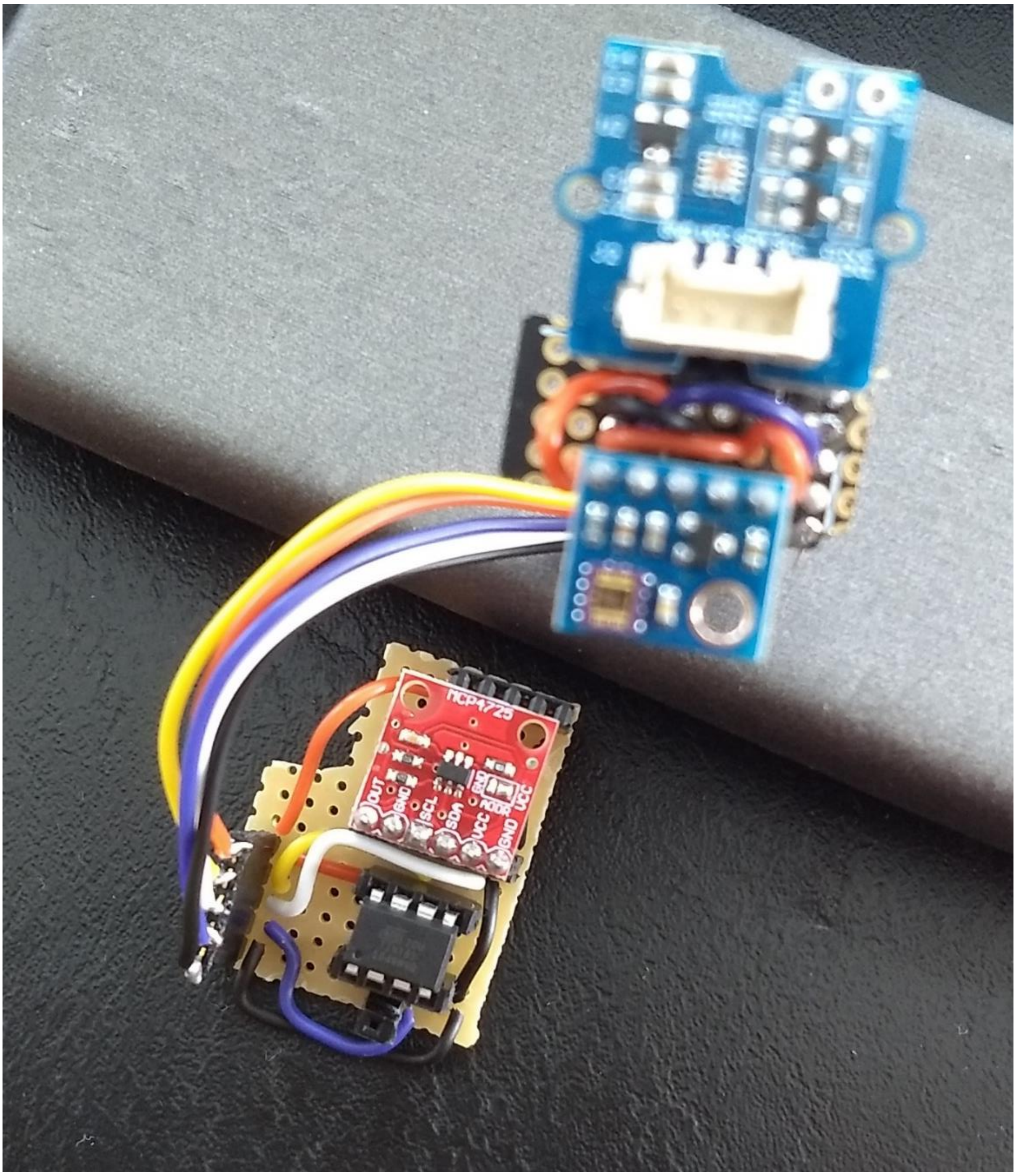


Figure 33: The setup of the Third prototype. The Solar Irradiance sensors can be seen here on top, on the bottom stand the Attiny85 microcontroller and the MCP4725 DAC.

6.3.1. Microcontroller

A new microcontroller has been chosen to be able to do the processing of the sensors and convert that to a voltage to be outputted. This is done best when a low-energy microcontroller is chosen that can function on 3.3V, as this is the same voltage as the reference sensor. For this task, an Attiny85 [102] has been chosen. This microcontroller can be programmed with an Arduino UNO and also uses an Arduino UNO to use the Serial Monitor functions. The microcontroller was switched from a TTGO LoRa32 OLED to an Attiny85, because of the difference in power consumption, as the Attiny85 only uses about 5mA when working, compared to the 50 to 80mA of the TTGO Lora32 OLED.

6.3.2. Sensor interfacing

The solar irradiance sensors that were used and tested in previous prototypes were continued in this prototype as these sensors function the best. Other sensors, like the Davis Solar Radiation sensor and the GPS sensor, were not needed anymore, as these are incorporated in the system that Konijn is making. However, when this low-cost pyranometer is tested, another system will be made to read the reference sensor when calibrating the system, and when evaluating the system.

Just like the GPS sensor, the power management capabilities are also integrated into Konijn's system and are thus not needed in the design of the sensor.
The full schematic of the third prototype can be seen in Figure 34.

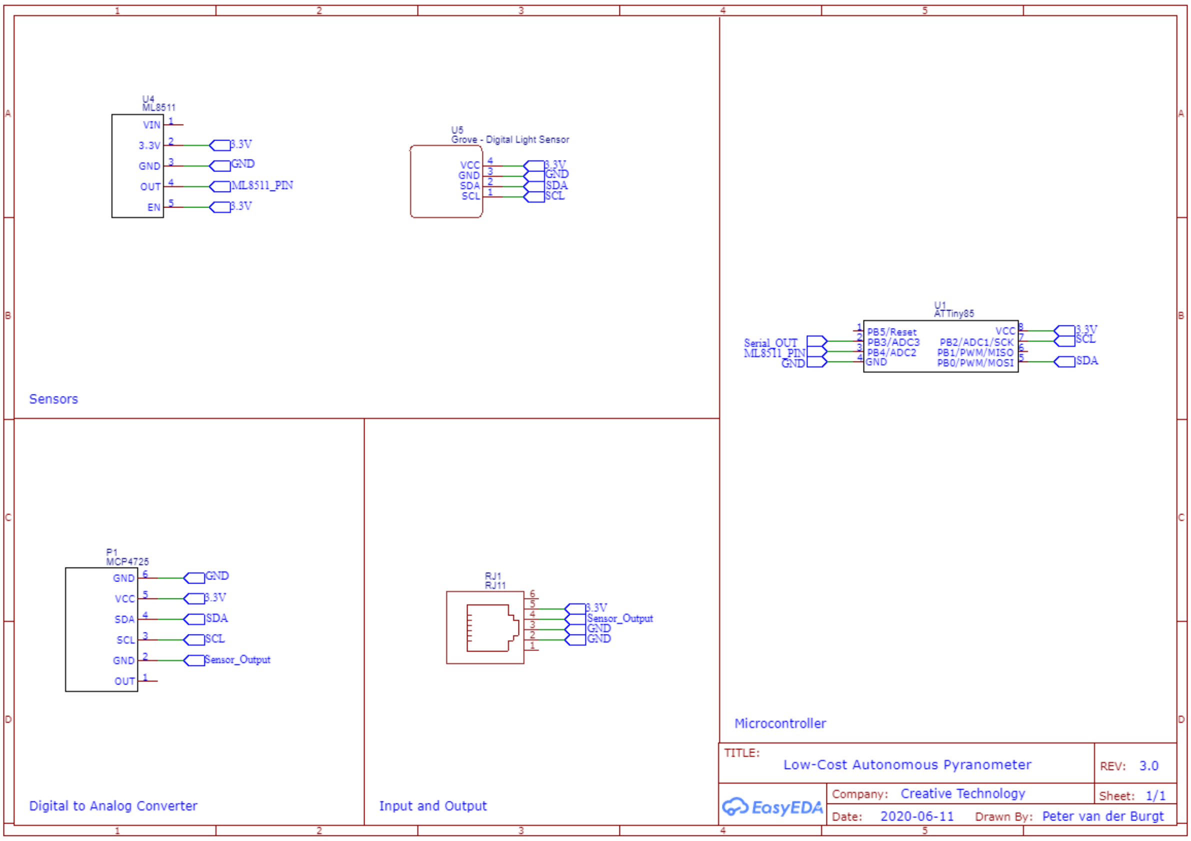


Figure 34: The full schematic of the third prototype.

6.3.3. LoRaWAN Decoder

As this low-cost pyranometer was meant to interface with Konijn's MMS, the communication from the MMS to the TTN also needed to be addressed. The unique payload structure designed for this MMS as described in Chapter 5.4.2.2., also needs a decoder to parse the payload back into the data. This decoder is implemented in the console of TTN and is written in JavaScript.

6.3.4. Casing

This prototype did not have a casing yet, as during the testing it became apparent that the sensors did not function properly.

6.3.5. Testing

When starting the functional tests, it was noted that one of the sensor variable's values seemed off. This was the Infrared light value given by the SI1145. This was quite odd since this is an I2C sensor and the Visible light and UV light values that were retrieved from the SI1145 seemed correct. After some testing and changing wires as well as measuring voltages, it seemed that the problem was with the Attiny85's special I2C library. However, there was not enough time to find the problem within the library and thus the next iteration was made.

6.4. Fourth prototype

After the construction of the third prototype, it was noted that the microcontroller was the only system component that caused the issues and not the sensor, so this microcontroller was changed. Furthermore, the entire prototype was placed in a waterproof case to protect it against the environment.
The fourth and final prototype can be seen in Figure 35.

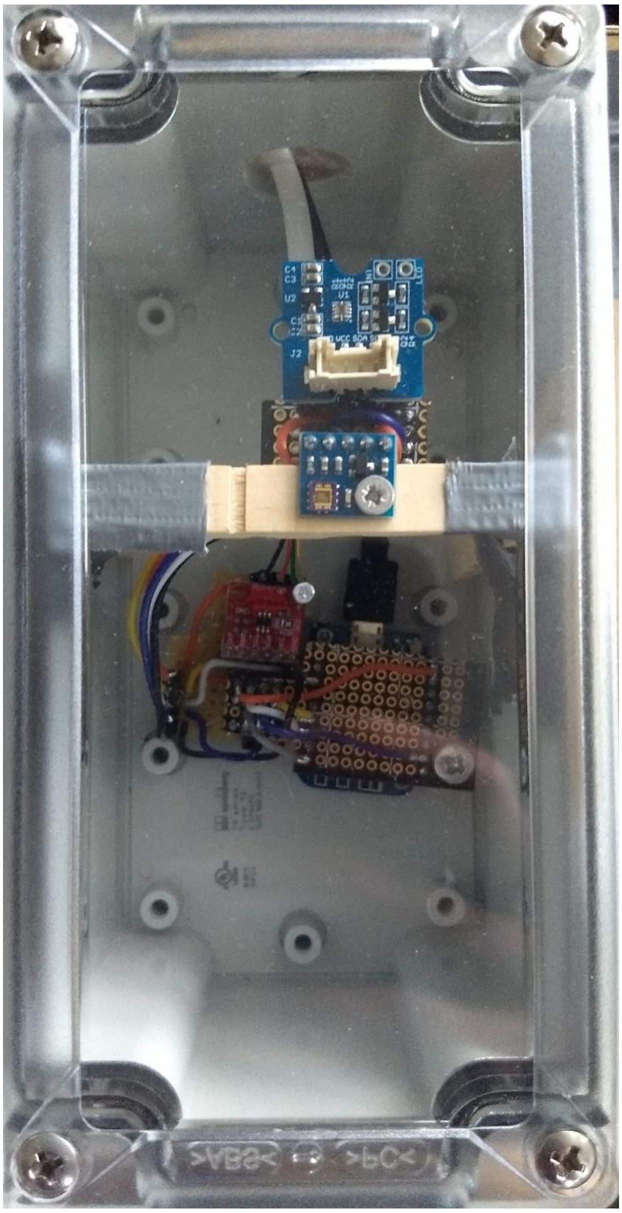


Figure 35: The setup of the final prototype. Both Solar Irradiance sensors can be seen at the top. The red PCB is the DAC, and the Wemos D1 Mini is on the right of that.

6.4.1. Microcontroller

The microcontroller to replace the Attiny85 is the Wemos D1 Mini, which uses an ESP8266 [103]. This is because this microcontroller functions on 3.3V but still had all the needed connections. This microcontroller was also available at the time of realising this prototype, unlike other microcontrollers.

6.4.2. Sensor interfacing

The sensor interfacing did not change since this worked correctly in the third prototype. The new schematic can be seen in Figure 36. The only thing that is changed from the third prototype in this schematic, is the microcontroller and the connections to this microcontroller.

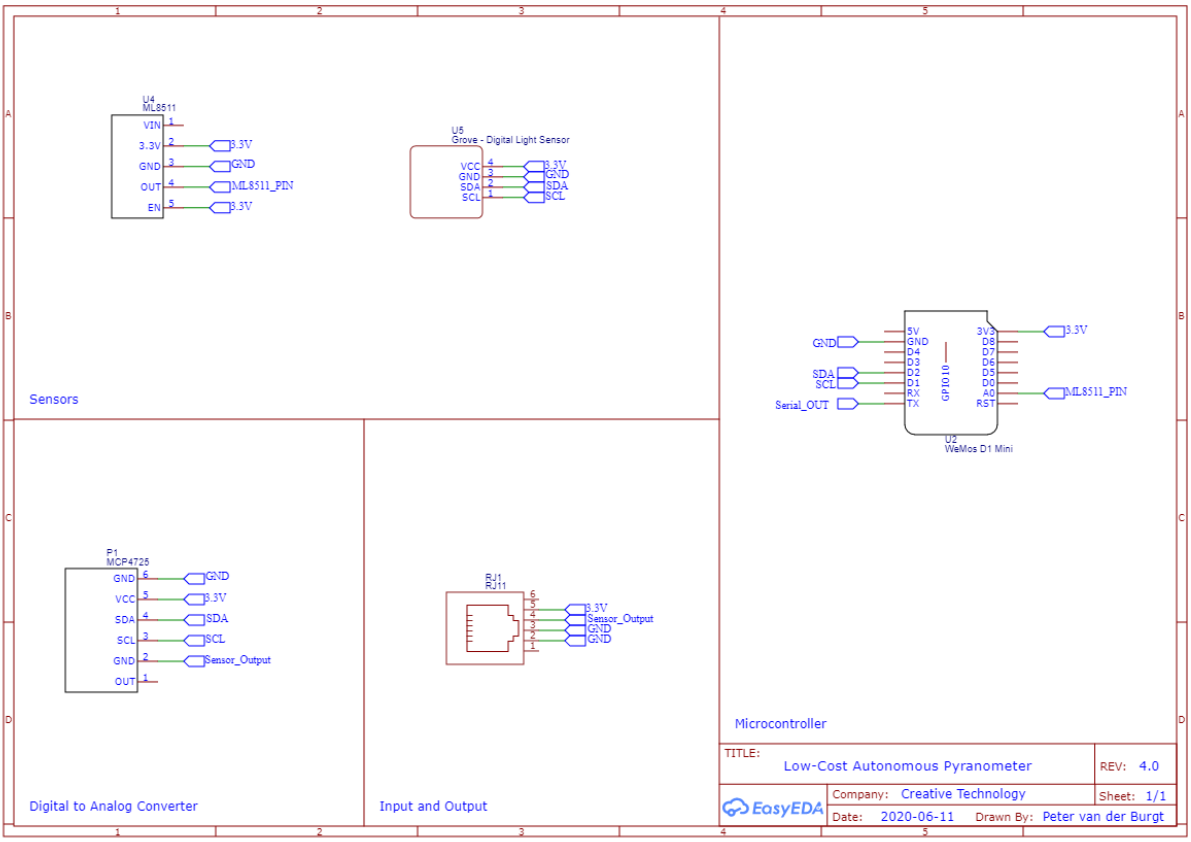


Figure 36: The full schematic of the fourth prototype.

6.4.3. Casing

The casing used for this prototype is a SenseBox Casing [104]. This casing was chosen due to unavailability of UV transmissive glass and plexiglass. The main reason why this casing was chosen, was because this casing is made to house the SenseBox Microcontroller and Sensors. These sensors are the TSL45315 [105] and the VEML6070 [38]. These sensors have a spectral response comparable to the sensors, which is why this casing was a suitable choice at this moment. This casing could, however, be improved by using a non-transparent covering for the sensors except for small windows covered by special transparent covers for the individual sensors. This will be further discussed in the evaluation and future work.

6.4.4. Calibration Setup

To be able to properly calibrate and later test this sensor, a proper Analog to Digital Converter (ADC) needs to be chosen. To be able to measure the voltages accurate, an ADC with high enough resolution and linear response needs to be chosen. This chosen ADC is the ADS1115 [106]. It is a 16-bit low power ADC and it will be used to measure the output of the reference sensor, and during testing also the output of the low-cost sensor. The 16-bit ADC means that a total of 65336 steps are possible to measure, which meant that every step was 0.125mV.
The overall conversion used for the Davis Sensor can be seen in Equation 2 [98].

$$Solar\ Irradiance\ \left[\frac{W}{m^2}\right] = MeasuredOutput\ \ [mV]\ /\ 1.67\ \left[mV\ /\ \frac{W}{m^2}\right]$$

Equation 2: Calculating the solar irradiance from the voltage level for the Davis Solar Radiation sensor.

As the resolution of the Davis Solar Radiation sensor is 1 W/m^2 [98], the minimum voltage that should be measured is 1.67 mV. As the resolution of the ADC is 0.125mV, this is low enough to detect these voltages, thus meaning that the ADS1115 can be used to properly measure the output of the pyranometer. For the calibration phase, a USB connection to the Wemos D1 Mini and a laptop has been used to gather the sensor data.

To gather a representative amount of data, there were three different measurement sessions of which the data was used for calibrations, one during the morning and early afternoon (10:04 – 14:04 on the 19th of June 2020), one during the late afternoon (16:56 – 17:57 on the 18th of June 2020), and one during the evening (19:55 – 22:26 on the 19th of June 2020). This is to make sure that there is enough variety of data that the Multiple Linear Regression (MLR) does not over learn on a certain data range, i.e. meaning that it would be better at measuring a higher range of Solar Irradiance, than a lower range of Solar Irradiance.
The schematic of the calibration and test setup can be found in Figure 37.

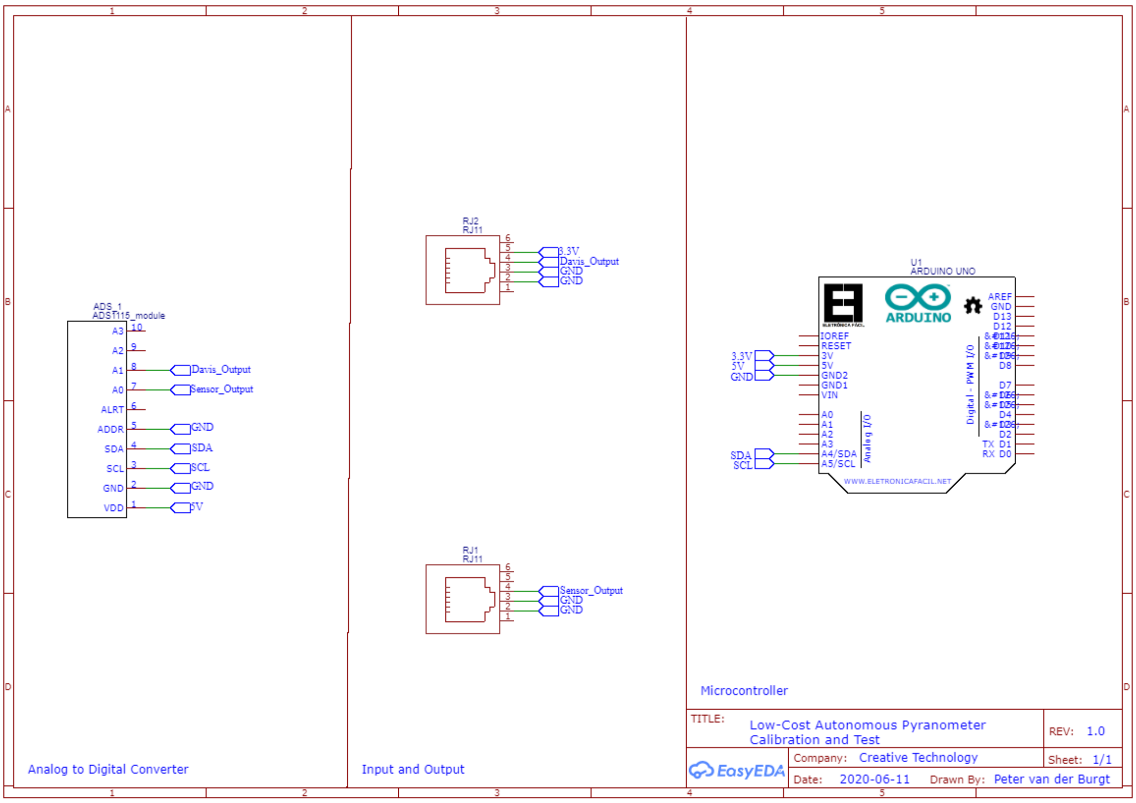


Figure 37: The full schematic of the Calibration and Test setup.

6.4.5. Calibration Results

After calibration measurements, which can be found in Appendix D, an MLR model was made of which the results can be found in Table 12.

Sensor	Adjusted R^2	Standard Error	Significance F	P-value	Coefficients
Low-Cost Pyranometer	0.994888	23.93375	0		
Intercept (offset)				3.5E-07	-397.597
SI1145 Ultraviolet Light				7.24E-05	-224.739
SI1145 Visible Light				3.4E-06	1.410168
SI1145 Infrared Light				1.1E-251	0.036012
ML8511				9.04E-24	3.626188

Table 12: The MLR model of the final prototype.

As described in Chapter 3.8.2.1., important values can be seen in the table. The table shows that the Significance F is 0, and all P-values are below 0.05. These values show that it was a statistically correct test. Furthermore, the Adjusted R^2 shows that almost 99.5% of the predicted points are on the regression line. This means that this regression model seems to be an accurate fit.
The overall RMSE value for all measurements is 23.93 W/m^2. This means that there is an accuracy of about 23.93 W/m^2 on the values given by the MLR model with the low-cost sensor output.
An example of this fitting can be seen in Figure 38 the other figures can be found in Appendix D.

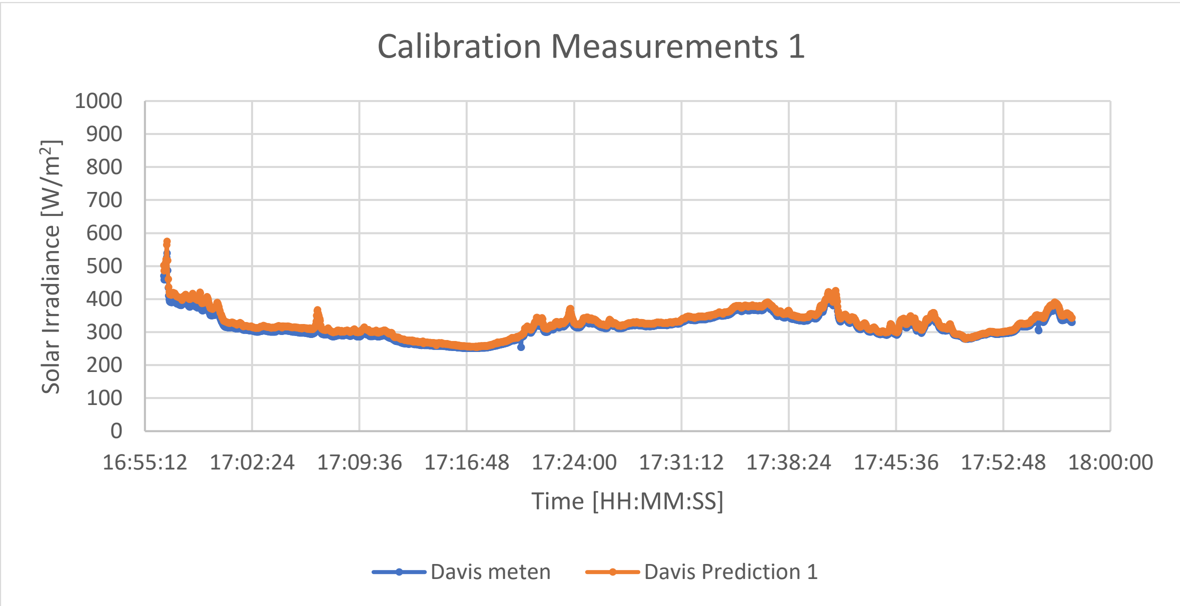


Figure 38: One of the calibration measurements taken for the solar irradiance sensor.

Next to that, the different coefficients that make up the equation to predict the Solar Irradiance are also given. The overall prediction equation can be seen in Equation 3.

$$Solar\ Irradiance\ [W/m^2] = X_{SI\ UV} * -224.739 + X_{SI\ VIS} * 1.410168 + X_{SI\ IR} * 0.036012 + X_{ML} * 3.626188 - 397.597$$

Equation 3: Calculating the Solar Irradiance with the low-cost sensor outputs

6.4.6. Testing

During the testing of the prototype, it became apparent that the RJ11 cable that is used for connecting the sensor to Konijn's MMS did not function. When measuring voltages it appears there is a large voltage drop over the cable, which results in the Wemos D1 Mini not being able to function, as it is lower than the 3.0V the ESP8266 requires [103].

This was solved by powering the Wemos D1 Mini directly over USB and measuring the output voltages with the ADS1115.

7. Evaluation

The evaluation chapter is the last chapter of this report that focusses on the Creative Technology Design method. In this chapter, the evaluations and tests of the system and prototypes are discussed. The performance of the solar irradiance sensor will also be evaluated, after which the requirements are evaluated.

7.1. Test Setup

The reference sensor and the low-cost sensors were placed level alongside each other on a camera tripod. The camera tripod was chosen to be able to easily level the sensors with the ground, to make sure representative measurements were taken. The test setup can be seen in Figure 39.

Figure 39: The test setup. Here you can see both pyranometers alongside each other.

The reference sensor is used as the calibrated sensor and subsequently, the output is measured to act as a baseline for the Low-Cost Pyranometer to attain.
The different analogue values that come out of both systems are monitored with the use of an Arduino and an ADC. This is also described as the test setup in Chapter 6.4.5..

7.2. Evaluation of Low-Cost Autonomous Pyranometer

The final Low-Cost Autonomous Pyranometer should be evaluated on its performance. This is done in three sections, the first focusing on the total costs of the low-cost autonomous pyranometer, the second on the solar irradiance measurement performance, and then the third focussing on the autonomous system itself, which was developed in cooperation with Jan-Paul Konijn.

7.2.1. Costs of the Solar Irradiance Sensor

Besides the different accuracy aspects to review, there is also the cost aspect to review.
A total overview of the costs can be found in Appendix B. The costs for the low-cost sensor itself is €56.60. These costs can be lowered by, for example, making a 3D printed casing or looking at different options for buying electronics. This is because the casing used now is relatively expensive and not ideal and optimised for the sensor. The electronics were bought from a Dutch retailer and there are cheaper options on the market.

7.2.2. Solar Irradiance Sensor

The solar irradiance sensor was tested as described in Chapter 7.1. and is further analysed in this section. The extended tables with all gathered data can be retrieved when requested.

The gathered data contained three different values for solar irradiance. One was the solar irradiance as measured by the Davis Solar Radiation Sensor. The second data point was the solar irradiance as measured by the output voltage of the low-cost solar irradiance sensor. And the last one was the solar irradiance as measured by the low-cost sensor which was outputted over the serial monitor. All three can be seen in Figure 40, a bigger version can be seen in Appendix E. The weather was quite similar to the weather during calibration, as it was sunny, with a sporadic cloud. Measurements were taken at 2 seconds intervals from 14:30:34 to 16:30:32 on the 20th of June 2020, for a total of 3600 measurement points.

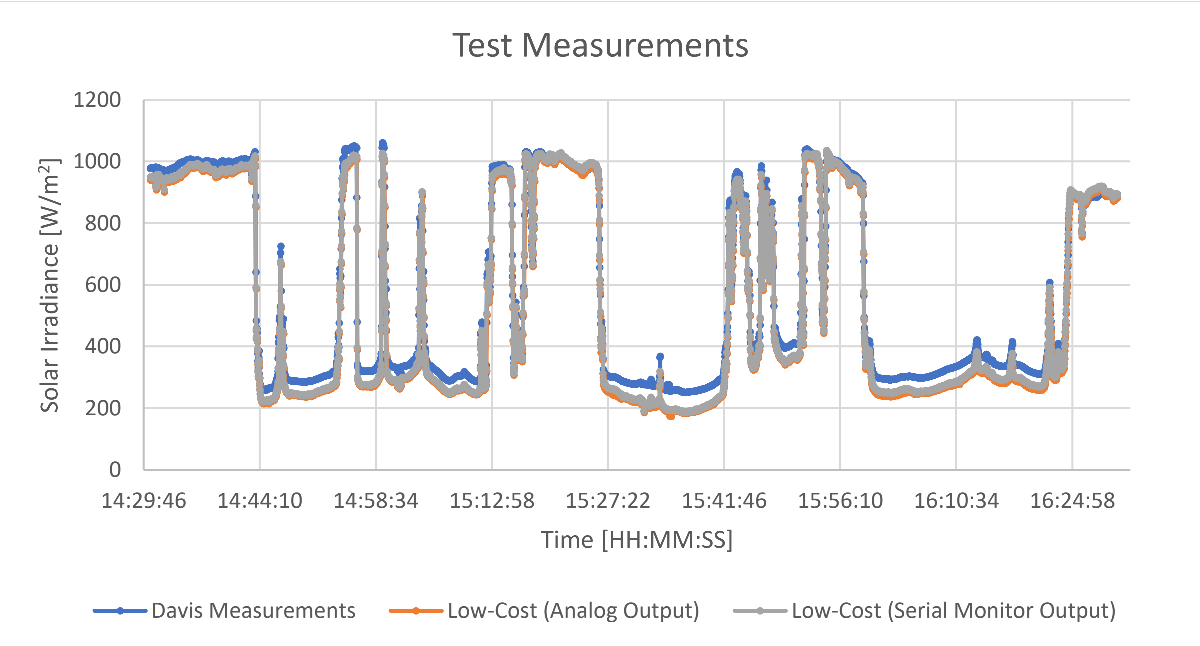


Figure 40: The final test measurements as measured from the solar irradiance sensors.

As can be seen in Figure 40, there is an offset of about 50 W/m^2, when the solar irradiance is at a value of about 200 to 400 W/m^2. This can be caused by the fact that the Multiple Linear Regression did not have enough data to learn these values of solar irradiance. It can also be caused by the fact that there were too many data for the other ranges of solar irradiance. To figure this out properly, it would be best to test these solar irradiance sensors in a controlled environment, where the solar irradiance can be tested easily, without any other variables influencing these measurements. In such a controlled environment it is easier to have a certain number of variable points for every solar irradiance intensity.

The overall RMSE value of the test measurements, when comparing the measured low-cost pyranometer voltages versus the measured Davis pyranometer voltages was 46.37 W/m^2. As can be seen by this value, there was no significant difference between the output as calculated by the low-cost sensor as compared to the measured value from the low-cost sensor, this can also be seen in Figure 40 when comparing the Davis Measurements with the Analog Output of the Low-Cost sensor.

Lastly, the plug and play integration with Konijn's system as initially intended did not work. After some testing and measuring, it was concluded that the supply voltage that was provided to power the system via the cable was too low. This was not due to a 'normal' voltage drop as the output voltages that were measured were still correct, as can be seen in the figure above. Thus, it was concluded that the power usage by the sensor as is, is too high for the use of the current cable.

7.2.3. Autonomous System

This part of the report was created together with Jan-Paul Konijn and the data has been obtained in collaboration with him. This section will describe the evaluation of the autonomous part of the system, consisting of the data communication over the LoRa network. This is since the low-cost sensor that was made, should be integrated into the system of Jan-Paul Konijn.

The MMS is located in the area of Alkmaar North-Holland. The map in Figure 41 depicts the gateways surrounding this area. The measurements for the sending of data over the LoRaWAN network were done over the course of four days, starting at the 19th of June 2020 00:00 until 23rd of June 2020 00:00.

Figure 41: The location of the MMS and Gateways around Alkmaar.

During this time, a total of 576 packets were successfully transmitted and out of this, 316 packets were successfully received. This means that approximately 55% of all packets were received successfully.

Reasons for this packet loss can be attributed to multiple factors. These factors all have to do with the reliability of the connection of the LoRaWAN network. This can also be seen in the metadata of the packets that were sent later, as the Received Signal Strength Indicator (RSSI) value for these packets varied around the -120dB mark, indicating that the connection was very weak.

Possible factors for this are the antenna and the distance between the node and the gateway. The antenna could be improved to increase the range that it could cover. As of now, it is a relatively short omnidirectional antenna. This means for example that the antenna could be designed to reach further over the horizontal plane, and less further in the vertical plane, as well as being longer. When this is used, it should be made sure however that the antenna is properly mounted, as this could influence the RSSI negatively when not done properly.

Another possible solution is to decrease the distance between the gateway and the node. This test was done in the surrounding area of Alkmaar, and as such it would be better to also test in Enschede, as Enschede already has a denser network of LoRaWAN gateways. If this would not be enough, it could be chosen to increase the number of gateways in Enschede.

7.3. Evaluation of Requirements

This section of the evaluation is centred around the evaluation of the requirements that were set in the specification phase of this graduation project. The functional requirements can be found in Table 13, and the non-functional requirements can be found in Table 14.

Functional Requirements	Requirements met?
The System Must	
Measure Solar Irradiance with an accuracy of at least 90 W/m^2	Yes, with an accuracy of 46.37 W/m^2
Send an average value for the solar irradiance at least every half hour	Yes, average values being sent every 10 minutes
Send LoRaWAN packages with a duty cycle of less than 1% and less than 30 seconds total per day	Yes
Take measurements every 12 seconds	Yes, but averaged over 3 measurements
The System Should	
Stick to the measurement frequency that is set by the KNMI	Yes
Have a covering of UV-transmitting glass	Not Tested
The System Could	
Be able to withstand temperatures from -10.1°C to 40.2°C	Not Tested
Be able to withstand wind speeds up to 76 kph	Not Tested
Be able to withstand precipitation up to 50mm per day	Not Tested
Measure Solar Irradiance with an accuracy getting close to 10 W/m^2	No, see above, accuracy of 46.37 W/m^2
The System Won't	

Table 13: Final Functional Requirements for the systems, evaluated.

The covering of the sensors was not properly tested or retrieved from datasheets so it cannot be said for certain that this requirement was met. This cover was used since it is made for housing low-cost photodiode sensors like the ones used in this project.
Furthermore, the different environmental factors were not tested either and thus cannot be concluded in the evaluation.
Next to that, the requirements for taking measurements every 12 seconds was met. But these values are not directly sent. They are averaged over three measurements and this measurement is sent over the LoRa network every 10 minutes, together with 9 other measurements.

Non-Functional Requirements	Requirements Met?
The System Must	
Harvest enough energy during the day and use this to power its systems during the day and night	Yes, using a solar panel
Be able to save Timestamp, GPS and measurement data to a database	Yes, using InfluxDB
Use GPS to gather location data with a resolution of 1 meter and time data to 1 second	Yes, using the Neo-6m
Use LoRaWAN communication technology	Yes, using SX1276 LoRaWAN chip
The System Should	
Be able to check its functioning and data gathering.	Not implemented
Be easily placeable	Not implemented
Be placed in a place with a sky view	N.A.
Have the photodiode-based sensors shielded from precipitation	Yes, with an IP value of 66
Be able to send a message when it is not functioning properly	Possibility for Error bits, however not implemented
Have a LoRaWAN availability of at least 33%	Yes, about 55% of packets were received
Have the Solar Irradiance Sensor under the direct view of the air	N.A.
Use standards	No, i.e. own payload made
Not damage the mounting place in any way	Not implemented
The System Could	
Save all measured values for a limited amount of time	Not implemented
Have a way to offload all saved values.	Not implemented
Be able to keep track of the battery level	Not implemented
Have a colour with a high albedo value	Yes, using light grey
The System Won't	

Table 14: Final Non-Functional Requirements for the systems, evaluated.

Not all non-functional requirements were met. Some of these are not implemented, others need more work. For example, the system uses System logging bits, however, these bits do not have an assigned function yet, just like the fact that checking of functionality and data gathering has not been implemented yet.
The bare minimum requirement of receiving 10 1-minute averages out of the thirty every half hour has been achieved, with receiving a average of 55% of the transmitted packages, this number would increase further when increasing the RSSI value.
The IP value of 66 of the casing means that the container is dust-tight and can handle a powerful jet of water making it a suitable outdoors casing. Furthermore, the saving of measured values requirement was removed since saving the data locally can cause errors to sneak in the system. Besides this, the albedo value of the colour used now is already quite high, but this can be further improved by using a white coloured casing.
For the requirements followed with “Not Applicable”, these are requirements that should be upheld whenever the system is placed, and thus cannot be implemented in the system itself.

8. Conclusion

This final chapter in this report will discuss the findings from this graduation project and will answer the given sub-questions and ultimately the research question as stated in Chapter 1.3. of this report. Next to that, it will discuss possible opportunities to further explore on, when this research will be continued. These are often opportunities and challenges that were out of the scope of this project, or other subjects, for which there was no more time.

8.1. Discussion

This first section will discuss the main research question, by answering the sub-questions first, and will be followed by some general discussions about the different sub-systems of interests within the system and the tests performed.

The first sub-question to be answered is: “*What methods exist for measuring Solar Irradiance by means of a Pyranometer?”.* Literature showed, that two types of existing methods for measuring solar irradiance are the thermal-energy pyranometers and the electrical-energy pyranometers.

Thermal-energy pyranometers are more expensive than the electrical-energy pyranometers. On the other hand, thermal-energy pyranometers do have greater accuracy as opposed to the electrical-energy pyranometers. As the goal was to make a low-cost autonomous pyranometer, electrical energy pyranometers were further investigated.

Electrical-energy pyranometers are divided in photovoltaic-based pyranometers and photodiode-based pyranometers. Both rely on the photo-electric effect, however, photodiodes are more specialised in sensing, as opposed to the energy harvesting specialised photovoltaic systems, like solar panels. Photodiode-based pyranometers are preferred to measure the solar irradiance.

The second sub-question that is formulated is: *“What type of sensors match these methods for measuring Solar Irradiance?”*. The research proved that it would be best to have different photodiode-based sensors, each having a different spectral responsivity, either belonging to ultraviolet, visible, or infrared light to measure the respective light to measure solar irradiance. Multiple sensors were tested and out of these sensors, the combination of the SI1145 sensor and the ML8511 sensor worked the best. The SI1145 sensor contains a visible and infrared light sensor, while the ML8511 contains a UVA and UVB sensor. Another observation for the choice of sensors can be made from how they output the measured value. If the sensor uses a digital communication technology such as I2C, there is less loss of accuracy due to voltage drops as opposed to analogue communication technology. With the analogue communication, the microcontroller needs to interpret the analogue value again, which will result in the accuracy of the sensor also depending on the accuracy of the ADC of the microcontroller as well as the resolution. Furthermore, the voltage drops over a length of wire, further decreasing accuracy. In the low-cost autonomous pyranometer that was made, one of the modules, the SI1145, uses a digital I2C connection. The other, the ML8511, uses an analogue voltage connection.

The best choice would be to have three different sensors, one for every type of light, that uses digital communication. In this graduation project, the infrared and visible light sensors were on one module that uses digital communication, and the other sensor which has an ultraviolet light sensor uses analogue communication. This meant that there were three different sensors that were placed on two modules.

The third sub-question to answer is: *"How to evaluate a made system, based on costs and accuracy?".*
In the state of the art it was discovered that when using reference sensors to calibrate the low-cost sensor, the accuracy was often determined by using the Root Mean Square Error (RMSE) value [30], [34], [50].
In the ideation chapter, when researching sensor fusion with the help of multiple linear regression, it was discovered that for this also the root mean square error value is used, thus leading to using the RMSE value to calculate the accuracy of the low-cost pyranometer. The accuracy of the low-cost pyranometer was calculated to be 46.37 W/m^2, which is about twice as accurate as compared to the 90 W/m^2 of the reference pyranometer. However, these measurements were based and trained on a reference pyranometer with an accuracy of 90 W/m^2. This would mean that the actual accuracy would be worse than the 46.37 W/m^2 and thus it is advised to train the developed pyranometer with a high-end pyranometer such as the ones from Kipp & Zonen. This accuracy would be worse because if all measurements were 45 W/m^2 off, than the overall accuracy would be about 90 W/m^2. Next to the accuracy of the pyranometer, there was also the costs aspect. The final low-cost pyranometer costs €56.60. This is quite the bit lower than the €400 set in the requirements, however, this was for the entire system, including other sensors. It would be better to compare it to the retail price of the reference pyranometer, which is €185.00. This is approximately three times more expensive than the low-cost pyranometer.

The answers to these three sub-questions help in answering the main research question: *"How to Develop a Low-Cost Autonomous Pyranometer?".* To make the sub-system, a low-cost pyranometer to be interfaced with an autonomous weather station, the right sensors are needed, which turned out to be photodiode-based light sensors, across the ultraviolet, visible and near-infrared light. These were found to be the SI1145 sensor and the ML8511 sensor in combination with the multiple linear regression machine learning technique. This system was then evaluated on accuracy and costs, to be about twice as accurate as compared to the reference sensor and about three times cheaper.

8.2. Future work

After this graduation project is concluded, there are some interesting points to work on further to improve the low-cost autonomous pyranometer. This section will describe the steps that could be taken to improve the sensor itself, as well as the sub-systems that make the system autonomous.

8.2.1. Solar Irradiance Sensor

For the solar irradiance sensor to function better, multiple improvements can be made in future works and iterations of the sensor.

First, the plug and play problems, concerning the integration with Konijn's system should be solved. As discussed in Chapter 7.2.1., these problems seem to originate by the fact that the wire that is used cannot handle the power that is required to power the sensor. This can be fixed by using a thicker cable, which could then handle more watts. Perhaps a better option is to change the microcontroller that is used within the sensor. This is because the Wemos D1 mini [103] was used since it was at hand, and there was not enough time to fix the problems that occurred when using the Attiny85 [102]. A good solution would be to try fixing these problems. However, another option would be to use a different low power 3.3V compatible microcontroller, such as an Arduino Pro mini [107]. This microcontroller would most likely not have the same issues as the Attiny85, as it uses the same library as the microcontroller used now.

Besides the plug and play integration, another important aspect also needs further attention: the casing and especially the transparent cover. The cover that was used now was an ABS plastic-type covering used by the SenseBox [104] system. It was thus deducted that this would be sufficient for the sensors. However, other covers like the UV-transmissive glass as described in Chapter 5.6. were ordered. Due to Covid-19 related issues, these did not arrive in time to be tested in this graduation project, but the sensor could benefit from these types of cover. This is especially because the transmissive properties will be known as opposed to the cover that was used now, of which this was not known. The general transmission of the cover could be measured by utilising a setup where one pyranometer would be placed under the cover, and another pyranometer without cover. However, this does not give a wavelength specific graph, as this would make it easier to specify what kinds of influence the cover has on the ultraviolet, visible and infrared light that is measured. Other covering should thus be tested, where the best practice would be to have multiple identical solar irradiance sensors, all with different types of covers, and see how well they perform. For the enclosure itself, it would be good to look into the options of separating the sensors from the control electronics like the microcontroller and the DAC. This is so that the electronics do not undergo a 'greenhouse' effect as it does now. The solar irradiance gets 'trapped' inside of the casing. It would be better to design a kind of weather hut with a transparent roof, so the sensors can be easily cooled, where the control electronics are placed in a closed box. Furthermore, it would be good to decrease the albedo value on the inside of the measuring compartment of the sensor, as this decreases the amount of internally reflected light within the box, as this influences the measurements. This would also make sure that the internal temperature stays within acceptable boundaries.

Lastly, the calibration of the sensor with Multiple Linear Regression (MLR) could be improved. As was shown in Chapter 7.2.1., the fit of the calibration could be improved. First and foremost, the overall accuracy of the sensor could be increased with the usage of a high-end reference pyranometer. This is because the reference pyranometer used in this research can be quite inaccurate and the accuracy of the low-cost sensor is reliant on the accuracy of this reference pyranometer. This means that using a high-end, accurate pyranometer would increase the overall accuracy of the low-cost solar irradiance sensor as well.

Besides using a more accurate reference sensor, it would also be best to gather more data to calibrate the sensor. Machine learning techniques such as MLR rely on the amount of data, but also the quality of data to return a good calibration curve. It would be good to have a continuous measurement of the low-cost sensor and the reference sensor, as this would increase the variety of data, thus making it more suitable for MLR.

The solar irradiance sensor itself was made and tested in an iterative process. This process was not foolproof and did have some errors. First, the testing bed where the low-cost sensor and the reference sensor was mounted to was not as stiff as it was hoped to be. This made it more difficult to keep the sensors level during the entire testing period. Next to the testing bed, the mounting point to the tripod itself was also enabling the bed to be easier tipped.

Furthermore, the test setup as described in Chapter 6.4.4., was made with the use of a breadboard. This may have caused instability in connection due to the quality of the breadboard. To minimize this, solid core wires were used for important connections instead of breadboard wires. However, before measurements were taken the connections were inspected and the moving around of certain components did seem to influence the measurements to a small degree.

The variety of data would also be increased by measuring during different seasons, and overall measuring during a variety of circumstances. This could be fixed by having a controlled environment, where the sensor could be tested without any other outside interference to maximize the quality of data.

Another solution to increase the accuracy of the solar irradiance sensor, is to create a controlled environment where the amount of solar irradiance can be set. This allows for accurate calibration, as well as the fact that there could be a set number of measurements for every solar irradiance intensity, thus decreasing the influence this have on the calibration with Multiple Linear Regression.

8.2.2. Autonomous System

With regards to the reliability of the communication, from the low-cost pyranometer to the autonomous weather station, the following can be improved. It would be good to look into the making the data communication from the pyranometer to the weather station a digital type of communication, as the analog voltage communication that is used now is impacted by the properties of the cables. This could decrease the overall accuracy of the low-cost pyranometer.

With regards to the reliability of the communication, from the autonomous system to the LoRa gateway, the following conclusions can be drawn. The positioning seems to be one of the main factors for data packet loss and there are a few options to handle this. First, the distance between the low-cost autonomous pyranometer and the LoRa gateway can be minimized. An alternative is to increase the density of the LoRa gateways in the area. As the low-cost autonomous pyranometer is to be deployed in the city of Enschede this is a manageable task to do. Another benefit of increasing the number of gateways is redundancy. This would mean that the data packets sent by the system are received by multiple gateways, which increases the overall reliability of the LoRa infrastructure.

Besides increasing the number of gateways, LoRa communication subsystems can also be improved. This can be done by testing other LoRa chips to be used in the low-cost autonomous pyranometer or by changing the gateways of the LoRaWAN. Since most gateways are already in place, it would be best to try other LoRa chips to be used by the system. It would then also be good to look into the possibility to perform calculations of the solar irradiance on the main microcontroller, instead of using a smaller separate microcontroller on the sensor sub-system.

Another improvement can be made in the payload structure. As of now, it can be easily adjusted to increase or decrease the communication frequency. It now sends GPS data every time data is sent, while this is not necessary. The low-cost autonomous pyranometer should be stationary, thus it is not needed to send the GPS data every 10 minutes. This increases the number of bits that can be used for weather measurements, thus increasing the measurement frequency, or decrease the number of bits sent, thus conserving energy. Another option would be to increase the resolution of measurements. Besides these options, this free space could be used to send statistical data such as the standard deviation between measurements that were used to calculate the average.

Bibliography

[1] TheNewPhobia, "Urban heat island." [Online]. Available: https://commons.wikimedia.org/wiki/File:Urban_heat_island.svg. [Accessed: 04-Jul-2020].

[2] W. D. Solecki *et al.*, "Mitigation of the heat island effect in urban New Jersey," *Environ. Hazards*, 2005.

[3] T. R. Oke, "The energetic basis of the urban heat island," *Q. J. R. Meteorol. Soc.*, vol. 108, no. 455, pp. 1–24, Jan. 1982.

[4] R. S. Kovats and S. Hajat, "Heat Stress and Public Health: A Critical Review," *Annu. Rev. Public Health*, vol. 29, no. 1, pp. 41–55, Apr. 2008.

[5] Kipp & Zonen, "CMP11 secondary standard pyranometer." [Online]. Available: https://www.kippzonen.com/Product/13/CMP11-Pyranometer#.XoRl4YgzaUk. [Accessed: 01-Apr-2020].

[6] Davis Instruments, "Wireless Vantage Pro2™ Plus including UV & Solar Radiation Sensors." [Online]. Available: https://www.davisinstruments.com/product/wireless-vantage-pro2-plus-including-uv-solar-radiation-sensors/. [Accessed: 01-Apr-2020].

[7] Davis Instruments, "Vantage Pro2," 2015. [Online]. Available: https://www.davisinstruments.com/solution/vantage-pro2/. [Accessed: 01-Apr-2020].

[8] A. Mader and W. Eggink, "A design process for Creative Technology," *Proc. 16th Int. Conf. Eng. Prod. Des. Educ. Des. Educ. Hum. Technol. Relations, E PDE 2014*, pp. 568–573, 2014.

[9] Dictionary.com, "Solar radiation." [Online]. Available: https://www.dictionary.com/browse/solar-radiation#. [Accessed: 03-Apr-2020].

[10] G. Stickler, "Solar Radiation and the Earth System," *Educ. Br.*, pp. 3–5, 2011.

[11] J. Bilbao, R. Román, C. Yousif, A. Pérez-Burgos, D. Mateos, and A. de Miguel, "Global, diffuse, beam and ultraviolet solar irradiance recorded in Malta and atmospheric component influences under cloudless skies," *Sol. Energy*, 2015.

[12] R. Rösemann, *Solar Radiation Measurement: From Sensor to Application in Meteorology and Environment ; an Overview of the State of the Art ; UV - Visible Light - Infrared*, 2nd ed. Kipp & Zonen, Delft, 2004.

[13] World Meteorological Organization, *WMO-No. 8 - Guide to meteorological instruments and methods of observation*. 2012.

[14] Nick84, "Solar spectrum ita." [Online]. Available: https://commons.wikimedia.org/wiki/File:Solar_spectrum_ita.svg. [Accessed: 04-Jul-2020].

[15] E. F. M. Abreu, P. Canhoto, and M. J. Costa, "Prediction of diffuse horizontal irradiance using a new climate zone model," *Renew. Sustain. Energy Rev.*, vol. 110, pp. 28–42, Aug. 2019.

[16] C. S. Djen, Z. Jingchun, and W. Lin, "Solar radiation and surface temperature in Shanghai City and their relation to urban heat island intensity," *Atmos. Environ.*, 1994.

[17] Aurora Solar, "Types of irradiance." [Online]. Available: https://twitter.com/aurorasolarinc/status/1073641311369084930. [Accessed: 04-Jul-2020].

[18] A. Mecherikunnel and J. Richmond, "Spectral distribution of solar radiation," *NASA STI/Recon Tech. …*, 1980.

[19] Dictionary.com, “Solarimeter.” [Online]. Available: https://www.dictionary.com/browse/solarimeter. [Accessed: 03-Apr-2020].

[20] R. K. Yin, “Exothermic definition and meaning | Collins English Dictionary,” 2010. [Online]. Available: https://www.collinsdictionary.com/dictionary/english/pyranometer. [Accessed: 03-Apr-2020].

[21] Encyclopedia.com, “pyranometer.” [Online]. Available: https://www.encyclopedia.com/reference/encyclopedias-almanacs-transcripts-and-maps/pyranometer. [Accessed: 03-Apr-2020].

[22] Hukseflux, “What Is A Pyranometer ?,” 2018. [Online]. Available: https://www.azosensors.com/article.aspx?ArticleID=248. [Accessed: 03-Apr-2020].

[23] Kipp & Zonen, “The Working Principle of a Thermopile Pyranometer,” 2016. [Online]. Available: http://www.kippzonen.com/News/572/The-Working-Principle-of-a-Thermopile-Pyranometer. [Accessed: 03-Apr-2020].

[24] Koninklijk Nederlands Meteorologisch Instituut, “Handboek waarnemingen Hoofdstuk 7. Straling,” in *Handboek Waarnemingen*, pp. 7-1-7–21.

[25] KNMI, “Automatische weerstations.” [Online]. Available: https://www.knmi.nl/kennis-en-datacentrum/uitleg/automatische-weerstations. [Accessed: 04-Jul-2020].

[26] A. Hinckley, “Pyranometers: What You Need to Know,” *Campbell*, 2017. [Online]. Available: https://www.campbellsci.com/blog/pyranometers-need-to-know. [Accessed: 21-May-2020].

[27] M. P. Thekaekara, “Solar radiation measurement: Techniques and instrumentation,” *Sol. Energy*, vol. 18, no. 4, pp. 309–325, 1976.

[28] Kipp & Zonen, “CMP6 Pyranometer,” 2018. [Online]. Available: https://www.omniinstruments.co.uk/cmp6-pyranometer.html. [Accessed: 04-Jul-2020].

[29] M. Benghanem, “A low cost wireless data acquisition system for weather station monitoring,” *Renew. Energy*, 2010.

[30] R. Mukaro, X. F. Carelse, and L. Olumekor, “First performance analysis of a silicon-cell microcontroller-based solar radiation monitoring system,” *Sol. Energy*, 1998.

[31] M. A. Martínez, J. M. Andújar, and J. M. Enrique, “A new and inexpensive pyranometer for the visible spectral range,” *Sensors (Switzerland)*, vol. 9, no. 6, pp. 4615–4634, Jun. 2009.

[32] Á. Vas, Á. Fazekas, G. Nagy, and L. Tóth, “Distributed Sensor Network for meteorological observations and numerical weather Prediction Calculations,” *Carpathian J. Electron. Comput. Eng.*, vol. 6, no. 1, pp. 56–63, 2013.

[33] D. K. Fisher and H. Kebede, “A low-cost microcontroller-based system to monitor crop temperature and water status,” *Comput. Electron. Agric.*, vol. 74, no. 1, pp. 168–173, Oct. 2010.

[34] K. Tohsing *et al.*, “A development of a low-cost pyranometer for measuring broadband solar radiation,” *J. Phys. Conf. Ser.*, vol. 1380, p. 012045, Nov. 2019.

[35] P. Guzmán *et al.*, “Low-cost sensor to detect overtaking based on optical flow,” *Mach. Vis. Appl.*, vol. 25, pp. 699–711, 2014.

[36] C. J. Watras *et al.*, “Evaluation of wireless sensor networks (WSNs) for remote wetland monitoring: Design and initial results,” *Environ. Monit. Assess.*, vol. 186, no. 2, pp. 919–934,

2014.

[37] S. Abbate, M. Avvenuti, L. Carturan, and D. Cesarini, “Deploying a communicating automatic weather station on an alpine glacier,” in *Procedia Computer Science*, 2013, vol. 19, pp. 1190–1195.

[38] V. Semiconductors, “VEML6070 UVA and UVB Light Sensor with I2C Interface.” p. 11, 2016.

[39] V. Semiconductors, “VEML6075 UVA and UVB Light Sensor with I2C Interface.” p. 11, 2016.

[40] V. Semiconductors, “VEML7700 High Accuracy Ambient Light Sensor With I2C Interface,” *Current*, no. July. pp. 1–11, 2005.

[41] V. Semiconductors, “VEML6030 High Accuracy Ambient Light Sensor With I 2 C Interface.” .

[42] ams AG, “TSL2591 Light-to-Digital Converter.” .

[43] Unknown, “VT900 PhotoConductive Cell.” .

[44] Adafruit, “Overview | Adafruit VEML6070 UV Sensor Breakout | Adafruit Learning System.” .

[45] Sparkfun, “SparkFun gator:UV - micro:bit Accessory Board - SEN-15273 - SparkFun Electronics.” .

[46] S. Labs, “Si1145/46/47 P ROXIMITY/ UV/AMBIENT LIGHT SENSOR IC WITH I 2 C INTERFACE Features Applications Description.” 2013.

[47] V. Semiconductors, “BPX43 Datasheet.” .

[48] B. Gschwind and L. Wald, “Testing several methods for the fusion of the two HelioClim-1 and -3 data sets of daily downwelling solar radiation at surface and creating a new consistent 1985-2016 data set,” pp. 1–24, 2017.

[49] D. Zhang, B. Heery, M. O’Neil, S. Little, N. E. O’Connor, and F. Regan, “A Low-Cost Smart Sensor Network for Catchment Monitoring,” *Sensors*, vol. 19, no. 10, p. 2278, May 2019.

[50] Y. Kim, J. D. Jabro, and R. G. Evans, “Wireless lysimeters for real-time online soil water monitoring,” *Irrig. Sci.*, vol. 29, no. 5, pp. 423–430, Sep. 2011.

[51] J. Lee, Z. Zhong, B. Du, S. Gutesa, K. Kim, and J. A. Reif, “Low-Cost and Energy-Saving Wireless Sensor Network for Real-Time Urban Mobility Monitoring System,” 2015.

[52] T. Onderwater, “Developing a sensor network for real time temperature monitoring in Enschede,” 2018.

[53] L. Kester, “Climate measurements in public spaces,” 2018.

[54] M. Pijnappel, “Low-cost Autonomous Temperature Measurements System.”

[55] J. H. Seong and D. H. Seo, “Environment Adaptive Localization Method Using Wi-Fi and Bluetooth Low Energy,” *Wirel. Pers. Commun.*, vol. 99, no. 2, pp. 765–778, Mar. 2018.

[56] A. A. HAFID, K. MEDDAH, M. ATTARI, and Y. REMRAM, “A Thermopile Based Pyranometer for Large Spectrum Sunlight Measurement,” *Int. Conf. Embed. Syst. Telecommun. Instrum.*, 2014.

[57] J. T. Devaraju, K. R. Suhas, H. K. Mohana, and V. A. Patil, “Wireless Portable Microcontroller based Weather Monitoring Station,” *Meas. J. Int. Meas. Confed.*, 2015.

[58] IPSON, “Opamp basics – Technology for Art and Education.” [Online]. Available: http://www.ipson.nl/opamp-circuit-examples/. [Accessed: 10-Apr-2020].

[59] senseBox, “senseBox | Die Kiste mit Sinn.” [Online]. Available: https://sensebox.de/en/. [Accessed: 06-Jul-2020].

[60] D. Vrijenhoek, “Improving the dependability of the temperature build-up sensor system in the city of Enschede A Creative Technology Graduation Project,” 2019.

[61] SODAQ, “SODAQ One Specifications.” .

[62] Espressif System, “ESP32 Series Datasheet,” *Espressif Systems*. pp. 1–61, 2019.

[63] H. Sharp, A. Finkelstein, and G. Galal, “Stakeholder identification in the requirements engineering process,” in *Proceedings. Tenth International Workshop on Database and Expert Systems Applications. DEXA 99*, 1999, pp. 387–391.

[64] A. L. Mendelow, “The Impact of the Stakeholder Concept,” 1981.

[65] MindTools.com, “Stakeholder Analysis - Project Management Skills.” [Online]. Available: https://www.mindtools.com/pages/article/newPPM_07.htm. [Accessed: 30-Apr-2020].

[66] D. Cohen and B. Crabtree, “Qualitative Research Guidelines Project,” 2006. [Online]. Available: http://www.qualres.org/HomeInte-3595.html. [Accessed: 29-Apr-2020].

[67] Creately, “7 Visual Brainstorming Techniques to Generate Ideas Super Efficiently.” [Online]. Available: https://creately.com/blog/diagrams/visual-brainstorming-techniques/. [Accessed: 03-Jul-2020].

[68] IIBA, *MoSCoW Analysis*, 2nd ed., vol. 2. International Institute of Business Analysis, 2009.

[69] Guru99, “Functional Requirements vs Non Functional Requirements: Key Differences,” 2019. [Online]. Available: https://www.guru99.com/functional-vs-non-functional-requirements.html. [Accessed: 22-May-2020].

[70] S. Brinkkemper and S. Pachidi, “Functional architecture modeling for the software product industry,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6285 LNCS, 2010, pp. 198–213.

[71] Arduino, “Arduino IDE.” [Online]. Available: https://www.arduino.cc/en/Main/Software. [Accessed: 22-May-2020].

[72] EasyEDA, “EasyEDA - Online printplaat ontwerp & simulator voor elektronische schakelingen.” [Online]. Available: https://easyeda.com/. [Accessed: 04-Jul-2020].

[73] Microsoft Excel 2016, “Microsoft Excel 2016 Spreadsheet Software, Excel Free Trial,” 2018. [Online]. Available: https://www.microsoft.com/nl-nl/microsoft-365/excel. [Accessed: 23-May-2020].

[74] www.statisticshowto.datasciencecentral.com (Stephanie), “Excel Regression Analysis Output Explained - Statistics How To,” 2014. [Online]. Available: https://www.statisticshowto.com/excel-regression-analysis-output-explained/. [Accessed: 23-May-2020].

[75] J. Frost, “Standard Error of the Regression,” 2019. [Online]. Available: https://statisticsbyjim.com/glossary/standard-error-regression/. [Accessed: 23-May-2020].

[76] LoRa Alliance, “LoRa Specification,” 2015.

[77] ArjanvanB, “Limitations: data rate, packet size, 30 seconds uplink and 10 messages downlink per day Fair Access Policy [guidelines] - End Devices (Nodes) - The Things Network,” 2020. [Online]. Available: https://www.thethingsnetwork.org/forum/t/limitations-data-rate-packet-

size-30-seconds-uplink-and-10-messages-downlink-per-day-fair-access-policy-guidelines/1300. [Accessed: 02-Jul-2020].

[78] KNMI, “KNMI - Daggegevens van het weer in Nederland,” *projects.knmi.nl*, 2019. [Online]. Available: http://projects.knmi.nl/klimatologie/daggegevens/selectie.cgi. [Accessed: 24-Apr-2020].

[79] T. R. Oke, “Initial guidance to obtain representative meteorological observations at urban sites,” *World Meteorol. Organ.*, no. 81, p. 51, 2004.

[80] G. M. Amer, W. Mohamed, and H. T. Dorrah, “A New Multi-Sensor Fusion Strategy for Optimal Spatial Design and Operation of Renewable Energy Grids,” no. December, 2014.

[81] R. Schaar, “Designing the VEML6070 UV Light Sensor Into Applications.”

[82] MBTechWorks, “I2C, SPI, UART Data Communications on the Raspberry Pi,” 2018. [Online]. Available: https://www.mbtechworks.com/hardware/raspberry-pi-UART-SPI-I2C.html. [Accessed: 04-Jul-2020].

[83] Node-red, “Node-RED.” [Online]. Available: https://nodered.org/. [Accessed: 02-Jul-2020].

[84] InfluxDB, “InfluxDB: Purpose-Built Open Source Time Series Database,” 2019. [Online]. Available: https://www.influxdata.com/. [Accessed: 06-Jul-2020].

[85] Grafana, “Grafana: The open observability plataform,” 2019. [Online]. Available: https://grafana.com/. [Accessed: 02-Jul-2020].

[86] The Things Network, “LoRaWAN airtime calculator,” *The things network*, 2020. [Online]. Available: https://www.thethingsnetwork.org/airtime-calculator. [Accessed: 13-May-2020].

[87] Qoitech, “How Spreading Factor affects LoRaWAN device battery life,” 2019. [Online]. Available: https://www.thethingsnetwork.org/article/how-spreading-factor-affects-lorawan-device-battery-life. [Accessed: 02-Jul-2020].

[88] TTN, “Modulation & Data Rate | The Things Network,” *Www.Thethingsnetwork.Org*. [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/modulation-data-rate.html. [Accessed: 02-Jul-2020].

[89] Wikipedia, “Decimal degrees - Wikipedia,” *Wikipedia*, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Decimal_degrees. [Accessed: 02-Jul-2020].

[90] Stack Overflow, “Low-power ESP32 + LoRa with deep sleep mode (TTGO LoRa32 OLED Board V1.0).” [Online]. Available: https://stackoverflow.com/questions/56548700/low-power-esp32-lora-with-deep-sleep-mode-ttgo-lora32-oled-board-v1-0. [Accessed: 11-Jun-2020].

[91] ON Semiconductor, “BS170 - Small Signal MOSFET 500 mA, 60 Volts.” .

[92] u-blox, “NEO-6 datasheet.” 2011.

[93] ROHM semiconductor, “Digital 16bit Serial Output Type Ambient Light Sensor IC,” no. 11046. p. 21, 2011.

[94] Lapis semiconductor, “ML8511.” p. 8, 2013.

[95] ADAFRUIT INDUSTRIES, “SI1145 Digital UV Index / IR / Visible Light Sensor.” .

[96] Cyro, “Acrylite OP-4.” .

[97] Stack Exchange, “Lux and W/m2 relationship,” 2014. [Online]. Available: https://physics.stackexchange.com/questions/135618/rm-lux-and-w-m2-relationship.

[Accessed: 01-Jul-2020].

[98] Davis Intruments, “Solar radiation sensor; Vantage Pro2 Accessories.” 2013.

[99] O. H. Greiner, “IP degrees of protection.” .

[100] Josmunpav, “Inconsistent values when using ‘analogRead()’ · Issue #92 · espressif/arduino-esp32.” [Online]. Available: https://github.com/espressif/arduino-esp32/issues/92. [Accessed: 02-Jul-2020].

[101] F. Koyanagi, “Do You Know About ESP32 ADC Adjustment? : 29 Steps - Instructables.” [Online]. Available: https://www.instructables.com/id/Do-You-Know-About-ESP32-ADC-Adjustment/. [Accessed: 02-Jul-2020].

[102] Atmel, “Atmel 8-bit AVR Microcontroller with 2/4/8K Bytes In-System Programmable Flash ATtiny25/V / ATtiny45/V / ATtiny85/V,” *Datasheet*. 2014.

[103] Espressif, “ESP8266EX Datasheet,” *Espressif Systems Datasheet*. pp. 1–31, 2015.

[104] senseBox, “SenseBox Outdoor-Gehäuse.” .

[105] ams AG, “TSL4531 Digital Ambient Light Sensor.” .

[106] Texas Instruments, “Ultra-Small, Low-Power, 16-Bit Analog-to-Digital Converter with Internal Reference.” 2009.

[107] I. Experiments, “Arduino Pro Mini : Power consumption.” pp. 1–11, 2018.

Appendices

Appendix A: Interviews

Interview Richard Bults and Hans Scholten

1. **What has been the main motivation to switch from a Davis system to a low-cost system (both pyranometer and MMS)?**
    1. *MMS: contact HJ Teekens two or three years ago. What can we do with the rainwater problem, smart rain buffer, further from a creathon? Soon the idea arose to look wider next to the rainfall. Then look at Enschede's climate adaptation. So also, an insight into the temperature structure → Heat island.*
    2. *Tom was the first to develop a complete system to be placed at a location in the city. Independence came from a solar panel, communication tool and 3D printing to experiment housing, to measure temperature. Yoann Latzer then made the data presentable. (initially a proof of concept)*
    3. *When that turned out to be going well, the plan continued, from then on with Wim Timmermans, by Laura Kester and Adam Bako. Laura's system went further than that of Tom Onderwater but added humidity and wind speed. She also looked at solar radiation through Wim Timmermans.*
    4. *That system gave its own weather station that was relatively cheap. Good feeling about communication technology, just like the skills of CreaTers. From there, a big step up. Then there was a plan for an own MMS. Together with Wim Timmermans, WHEGS was born to start a research project to get 80 WSN in Enschede.*
    5. *In the end, it was decided to buy SenseBox due to financial considerations. The quality left it to new plans since it was actually below level.*
    6. *Finally, it was Max's turn, he just did not get far enough, so JP was added*
    7. *Hans and Richard also saw that the UV and Solar Radiation sensor together is 500 euros. This resulted in Peter's assignment.*
    8. *There is a solid interest in the municipality of Enschede, as they want MMS to provide insight into the temperature in the city. Wim is interested in the data of the systems, and Hans and Richard are interested in making low-cost MMS.*
    9. *It is, of course, also great for CreaTe to use students with the latest technology, and it provides a nice graduation assignment.*
    10. *Ultimately, the system can, of course, also provide economic aspects with insight for the municipality of Enschede. It can ensure that you can search for warmer places in the city.*

2. **What are the most important aspects for you that the (Low-Cost Autonomous Pyranometer) and the (MMS) can / have?**
    1. *How would you prioritise this? / which would drop out first*
    2. *Cost price*
    3. *Quality*
    4. *The two things that are mainly looked at can buy an expensive system for 800 euros and for everything, even double that. It is now the challenge to keep it as low-cost as possible.*
    5. *Within Hans his department there is also research into communication, and a LoRa plays a major role, which gives you more contact and data that you obtain from typing Lora, such as triangular measurements*
    6. *Educational aspect, creativity works better than obstruction. It is more fun to look at solutions creatively instead of putting down a lot of money.*

3. **Is it important that the system can determine its own location, or is it sufficient to keep track of the years of identification system?**
    1. *Not necessary to determine locations, because they are stationary systems. On the other hand, GPS can be very useful, for example, a kind of theft alarm. More importantly, if the system is moved, the measurements may become invalid. Furthermore, the GPS can determine a very accurate time. Almost at the microsecond level and determine right measuring moments. Furthermore, GPS is useful for when you want to measure mobile. (equip buses or means of transport)*
    2. *Stationary measuring does not require a GPS*
    3. *Richard: Having accurate location information during a measurement is a must-have requirement. For example, the mobile measurement of patients, patients who are mobile, who have data collection, and which is sent. Is the quality of the data good enough to let a machine decide? Yes, it is true that the MMS is stationary, and you also want it to hang in a place that belongs to the data. And so, this is also part of the quality of the data. [questions to Wim Timmermans and municipality] bring GPS, although it takes a lot of energy and takes a lot of money. Time stamping can be important. So, if you are wondering if you can take GPS, you have to think creatively about how you want to do this. This has to do with the Data quality. Do this especially if the option is available.*

1. **What would be the best data frequency? According to the KNMI, there should be a measurement every 12 seconds, but what do you think the minimum usable data frequency would be?**
    1. *This is determined by the KNMI manual, measuring according to the standardised way. KNMI does not measure per 12 seconds, but sometimes measures once per minute and then determines the average. Determine a kind of hierarchy of temperature. For example, the average per hour, which is then divided by minutes and the like. It is clear to Hans that we must measure according to KNMI's standards. Richard agrees.*
    2. *These are often fed from the wall-socket, but this is again seen as a challenge. One way of saving energy is to turn things on when measuring (relay transistors, etc.). Follow here the manual as well as measuring in an energy-saving way.*
    3. *Richard agrees and adds the difference between a sensor and sensor system or the MMS. There is a possible difference between the measuring frequency of a system than a measuring frequency of a sensor system. The MMS transmits measured values to the server once in a while, but that one meeting is based on possibly several measurements from one sensor. What the MMS transmits is different from the sensor.*
    4. *GPS is necessary to divide the day into hours, and that into minutes, and in some minutes, you need to take more readings than others.*

1. **Is there a preferred communication method for you?**
    1. *As far as Richard is concerned, we note that we use LoRaWAN. Looking at the scatter radius, MMS aren't always close to private homes. Things like Wi-Fi are also dropped here.*

1. **It still happens that the data sent does not arrive properly. This is of course not what we want, but what is a maximum number of unpacked packets, for example in percentages?**

   1. *Interesting question. If you make a choice for your data communication infrastructure, it can also mean how reliable, how big and how fast your data is. When we say that we are going to use LoRaWAN, TTN only says that uplink is needed. Here the provider chooses not to choose bidirectionally. This would block the entire network. If we are only committed to uploading data, you are not sure that your data has been uploaded correctly. Because you have no feedback. You can solve this by placing a fine-meshed network. If it is finer, then it makes less difference that there are nodes over a large area. (what is acceptable in terms of fine to coarse-meshed network. Ask Wim Timmermans about data loss)*
   2. *A certain amount of redundancy is needed to ensure that if one node is not communicating properly, another must accommodate it.*
   3. *(Possibly less important for us, because this network is determined by Wim Timmermans)*
   4. *Take a good look to ensure that there are still measurements for each type of measurement.*
   5. *Example, 20 systems, 10 for green 10 for buildings. Falling out by half is okay, as long as it's a bit with both groups and not all 10 of a group.*
   6. *A bit out of the scope of the individual system.*
   7. *It depends on the reason for the failure.*
   8. *When looking at reliability is. One of the most important aspects is that the system must remain powered, so enough battery capacity. Can the system be operational 24/7, or would it be 50% enough, but when or not? Take a good look at those dependability parameters and analyse, which we have to address in our system design.*
   9. *We can do less about LoRa that breaks down, but if there is not enough power. If we have less energy like winter, we can send data once a day instead of once an hour.*
   10. *Possibly graceful degradation and how is it communicated. So, there is also a kind of notification so that the recipient knows how the system functions (For example, use a byte in your package.)*

1. **What do you see as the minimum resolution/accuracy/precision of the sensors?**
   1. *Wim Timmermans does have an opinion about this. Max's report does state this. Here are things stated like the accuracy of sensors.*
   2. *If we know the accuracy that Wim wants to see, we have to take this into account and also say something about the resolution.*
2. **In the previous projects, what are the measurement qualities that you want to see improved?**
   1. *The quality of the system must be considered. For example, there was a very good digital sensor that tom used, but the new ones that were bought suddenly all broke.*
   2. *General reliability and component reliability. You don't want to go there for maintenance all the time.*
   3. *The first systems were a kind of sniffing phase.*
   4. *Now it is important not only that it works, but also how well and how long it works, that the system continues to work.*
3. **I heard that you bought Senseboxes in collaboration with the municipality. Why did you choose this system? What are the advantages? And what are the disadvantages?**
   1. *It was decided to choose SenseBox because the three participants in the project are in the project for their own reason.*
   2. *Ultimately, there was pressure from the subsidy provider that a system had to be created, so that SenseBox was determined to be used.*

3. *Then it came from Richard and Hans that they make an MMS for less than 400 euros that has just as good data as the Davis vantage pro 2 system. That whole thing costs about 1200-1500 euros.*
4. *This also resulted in requirements that have been given to measure all sensors.*

1. **How do you see the placement process of an MMS? What should be the steps for a user to get the system up and running?**
    1. *Wim Timmermans is now discussing with the municipality where all systems should be located in order to obtain good quality data. So, nothing is sure yet.*
    2. *Placement of systems are not necessarily compliant with standard MMSs, but it is compared to the situation with concrete buildings and brick buildings and the like [ask Wim Timmermans]*
2. **Should the data be publicly accessible? And what about making/calibrating the system?**
    1. *If such a system is correct and works, it is also great that everyone can make and use such a system with a limited system.*
    2. *The data is outside the scope but should actually be publicly available.*
    3. *However, this would also mean that it should be clear how a data unit is created.*

Interview Wim Timmermans

1. **It still happens that the data sent does not arrive properly. This is of course not what we want, but what is a maximum number of not received packets, for example in percentages?**
    1. *Not immediately a number, except of course one hundred percent that arrives. In the event that things don't arrive, then one should know how much is received, then you know if it's good data.*
    2. *Especially the time of the observation is important. The temperature fluctuates more during the day than at night. At night it is less bad if it breaks down.*
    3. *That depends on the data, and it varies a lot. Those criteria are not necessarily fixed either.*
    4. *It also depends on the weather pattern, sometimes it is very fixed, and sometimes it changes at once.*
    5. *A percentage of how much data was retrieved and the time of that data is the best.*
    6. *It mainly depends on who uses the data. For a model that produces warnings, or to residents of Enschede.*
    7. *Mesh network, how much is needed and how good the data must be. (given a "data loss" of 25 or 10 percent, how would he place the sensor systems in the city Enschede. If there are four systems and 1 doesn't work, what does the quality of the data do?)*
    8. *What are the parameters/sensors that you need 24/7, and which can possibly fail, for example, to save power? (graceful degradation)*
2. **You are talking about a sensor network of 80 units. Can you give us an indication of where these may be placed?**
    1. *Can send a map where it will be placed globally. Exact has not yet been determined.*
    2. *A little bit of everything, a square, a park. Radiation applies less too, but the other variable, you want to measure at the same level because otherwise it cannot be compared.*
    3. *Measure on the street, in a clearing, near a house.*
    4. *Scientifically, you have to measure it at a meter and a half, but that becomes difficult because then they are stolen.*
3. ***Does it have to have a mounting system/slot?***
    1. *In principle, it is thought to attach to poles. Existing poles especially, because that is the universal solution imaginable.*
    2. *Possibly make something that fits on different types of poles, lampposts are somewhat inconvenient because of the light. In principle, a fixed time (13 minutes before sunset and 13 minutes after sunrise) preferably as vandal-proof as possible, and as little attention as possible. As small as possible*
4. **With regard to previous studies, what were things that should receive some extra attention from us?**
    1. *The reliability, how long does it last. Especially the power supply is very important. Actually, he just needs the MMS to be in the air 24/7.*
5. **If you purchase sensor nodes, what are the most important quality requirements that a system must meet?**
    1. *What do you see as the minimum resolution/accuracy/precision of the sensors?*
    2. *See previous studies*
6. **Which parameters are most important for your research? And which least?**

    1. *Temperature and radiation, but wind speed is also important.*

7. **If you install an autonomous measuring instrument, how long should it be able to operate alone without intervention?**
    1. *About 50-60 years*
    2. *At least a year, because you want to be able to measure all seasons and pick up all influences.*
    3. *For Wim's research, there must have been something until the end of time.*
    4. *For climate you can only measure in 30 years, so for climate change, you need 60.*
    5. *It is mainly in the context where you place it.*
    6. *If he does not last long, he should also be able to indicate that he needs to be replaced.*
8. **What should be the accuracy of the location?**
    1. *For Wim, it is not important what GPS value the station indicates, Wim will measure it itself, so that does not matter that much.*
    2. *Wim thinks that monitoring a sensor's location does not add much value*
    3. *Not necessarily a requirement.*
9. **I had heard from Richard that it might be possible to get data from a calibrated Pyranometer from the ITC, is this still correct?**
    1. *Just send an email. On average, he sends values for half an hour. He is not fully in accordance with the KNMI measures. It is not always well-calibrated.*
    2. *Just put it on the mail for the dates of yesterday.*
10. **In a previous interview with an old CreaTe student, you had noticed that a solar radiation resolution of 10 W / m 2 should be sufficient, do you still agree?**
    1. *This is a number that Wim hopes is feasible.*
11. **A pyranometer is generally placed a meter and a half above-cut grass, how do you envisage this in the municipality of Enschede?**
    1. *The diffuse radiation is less of reflection from buildings but still from clouds. The requirements set by KNMI are designed for the perfect weather stations, which must meet the requirements.*
    2. *This is simply not possible within the city, so the requirements should be as close to acceptable levels as possible.*
    3. *This is a lot of consideration and sees how good the measurements are going to be.*
12. ***Also, look a bit at measuring in cities. There are probably manuals for that.***
    1. *WMO has initial guide to obtain meteorological observations at urban sites.*
13. ***Further ideas about the SenseBox***
    1. *There is now an idea for making those sensors. Alfred is mainly working on a 3D model. It is mainly a matter of waiting for several factors before it comes to hang again.*

Second Interview Wim Timmermans

1. **Pre-questions**
    1. *Conversation municipality Enschede. Lampposts: fix rubber or something. Otherwise, the paint of the lampposts and the like will be damaged*
    2. *Wim is also going into town with Hendrik Jan Teekens to scout for places for the measuring systems.*
    3. *What about the technical measurement, according to the expert?*
2. **What is the desired measurement frequency required for all variables (also looking at LoRa restrictions)?**
    1. *The minimum measurement frequency is mainly about the scientific measurements being taken primarily as the scientists find it interesting.*
    2. *This ensures that the rules are certainly not always followed.*
    3. *The systems in Enschede are used to compare with the systems of the KNMI, so try to get as close as possible.*
    4. *Wim also understands that it causes problems with the transmission of data.*
    5. *The minimum resolution would, in principle go to half-hourly average. Based on how many measurements. These samples can then come from the manual.*
    6. *Basic storage of half an hour would be nice. If you then assume that you then calculate everything on the node, and send that, then that thunders away. Especially the average is interesting. Minimum, maximum, and standard deviation are okay, but then Wim prefers the raw data.*
    7. *The data forwarded would be nice if it can be checked.*
    8. *There will be a system that students will check the entire system every 2 weeks.*
    9. *A GPS determination can be more interesting for things like APP measurements on a mobile.*
    10. *Closer to the surface is the most interesting variation.*
    11. *They turn off 13 minutes before sunrise and turn on 13 minutes after sunset. So, lampposts can be a problem, this makes it interesting.*
    12. *Optionally, you can use an SD card for data logging of samples. Whether you let it be overwritten or stopped.*
    13. *Where does your timestamp belong to? This is the beginning, middle or end of the measuring interval, the most common is the end.*
    14. *For Wim, it is also a fundamental discussion since it is also part of the type of measurement. Also, because the system is now placed, and it is still a manageable amount. Also, because a certain accuracy has to be achieved. You also record how you do the measurement, see whether you measure at a wall or at a park or not.*
3. **What is the desired communication frequency (how bad should it be in real-time?)**
    1. *Solar radiation*
    2. *Temperature*
    3. *Humidity*
    4. *Barometric pressure*
    5. *Wind speed*
    6. *Wind direction*

Interview Rik Meijer

1. **What is the end goal that the Municipality of Enschede wants to achieve with these sensor nodes?**
    1. *That is not quite clear yet. There is a need for flat insight, how it divides with the heat in Enschede and you can visualise something of a relationship about certain establishments and what you measure about it. Think of new construction situations but also adjusting things*
    2. *If you have a web application, people can also see where it is warmer and cooler. What does the municipality think is important, it must be colder near a nursing home?*
    3. *Giving people insight. Heat affects people, so maybe people will also adjust things to suit their own lot.*
    4. *It would be great to be able to show people what is going on and what people can do about it themselves.*
2. **What is your experience with vandalism of this type of project?**
    1. *By changing the things that are hanging now, they ensure that you cannot reach them with a long stick from the ground. This is already handy with a four-meter-high lamp post.*
    2. *Of the five hanging, not one is broken yet.*
    3. *Possibly hanging at people's house, this must be discussed. It is necessary to measure in public places, so it is a difficult subject.*
3. **What measuring instruments does the municipality of Enschede currently use and what are the purposes?**
    1. *What did you run into when using these measurement systems?*
    2. *A number of systems have already been installed, which mainly measure temperature, as a first pilot on how to make something like this (Tom / Laura).*
    3. *Those insights have not yet yielded much for the municipality, especially because additional things must also be measured such as humidity and wind speed and direction.*

4. **How important is the design of the device? And what are the requirements of the municipality with regard to design?**
    1. *Not much agreed upon. The models shown were white. No idea if that is the easiest. Colour is useful to match the environment. When placed on a light pole, the orange straps of today are clumsy. They stand out.*
    2. *In a green environment perhaps in a suitable colour.*
    3. *Here, colour is mainly related to the places where they are placed*
5. **Is the data generated used for policymaking and if so, which?**
    1. *Example building regulations., Greening of the city*
6. **Should the data be publicly accessible to not only people from the municipality but also residents of Enschede?**
    1. *Yes, but it depends on where they are placed. If they are placed on private land, it is inconvenient for privacy.*
    2. *Is it information that can be made available just like that, or that it gives a problem with the GDPR?*
    3. *It is important to consider what information can and cannot be made available.*
    4. *Subsidised from municipal money but getting enough information.*
7. **For the placement of technologies, there are laws that prevent that. How does that legally work?**
    1. *No idea, Rik thinks it is not so bad because there are often no people to trace. This shouldn't really be possible with this technology.*

2. *If it is all made then, it should be doable.*
3. *There are now in the centre of Enschede cameras.*

8. **Have you previously applied a sensor network or sensors in the city, if so? Which?**
    1. *The smart water buffer, heat build-up what we do. These are the two things that are currently running with the UT. Also, a pilot project to build an extra city stream in the Elferinksweg.*
    2. *Which communication methods work to get private individuals working.*

9. **How do you foresee Enschede in the future, which technologies will you use?**
    1. *Colleagues are working on a smart city concept*
    2. *All transport movements due to corona are made transparent. So, flows of traffic.*
    3. *For example, smartphones are also used to see how many people can be where. No idea what it will be used for and where it can be used.*
    4. *Smartphones that measure whether roads need maintenance. Quality of measurements then becomes a number of measurements.*

Appendix B: Solar Irradiance Sensor Costs

Product	Price per	Amount	Total
Micro USB Cable	€2,50	1	€2,50
Wires	€1,00	1	€1,00
Prototype board	€2,50	1	€2,50
MCP4725	€3,50	1	€3,50
Headers (Fe)male	€0,70	1	€0,70
Wemos D1 mini	€6,50	1	€6,50
RJ11 Cable	€2,50	1	€2,50
Casing	€17,50	1	€17,50
Grove Sunlight sensor (SI1145)	€11,95	1	€11,95
GY-ML8511 UV light sensor	€7,95	1	€7,95
Total			€56,60

Appendix C: TTN Timing Calculations

Spreadings factor SF 7
Bandwidth 125 kHz
Region EU 868
Payload size 51 Bytes

Air-time 118 ms

fair-use-policy = 30 s

30000 ms / 118 ms = 254 messages

51 bytes x 254 = 12 954 Bytes / 24 h ≈ 539.75 bytes / hour

254 messages / 24 h = 10.58 Messages / hour

~~0.18 messages / hour~~ 0.18 messages / minute

approx 1 message / 6 minutes

Timestamp + Location + Temperature + humidity + wind speed + Solar Irradiance

— 9 bytes 2 bytes 1 byte 2 bytes 2 bytes

16 bytes

No timestamp in cayenne so message size will be 16 bytes ~~+~~ 2 bytes per parameter = 26 bytes without timestamp

Dataset temp = { 7.2, $\underbrace{6.2, 5.2, 3.5, 2.7, 2.3, 2.1, 2.2, 2.0}_{\text{Based on delta measurem}}$ }

N = 10
1: initial
9: Δ measurements

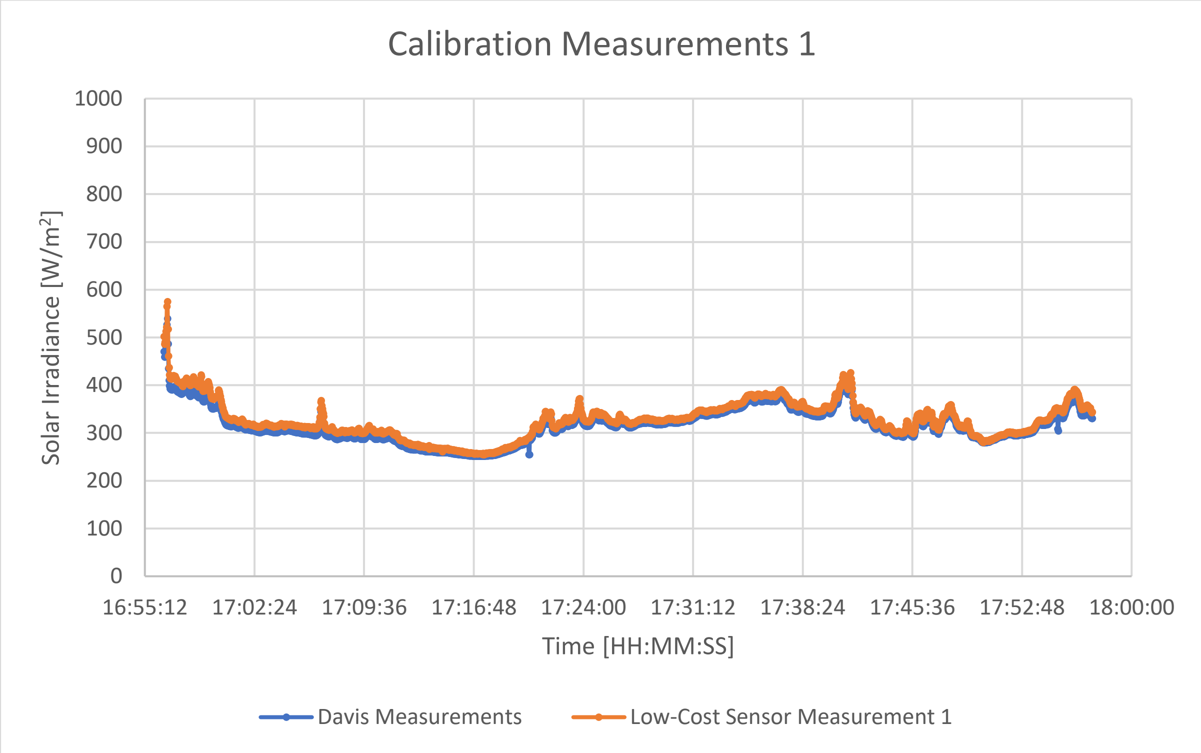
initial measurement

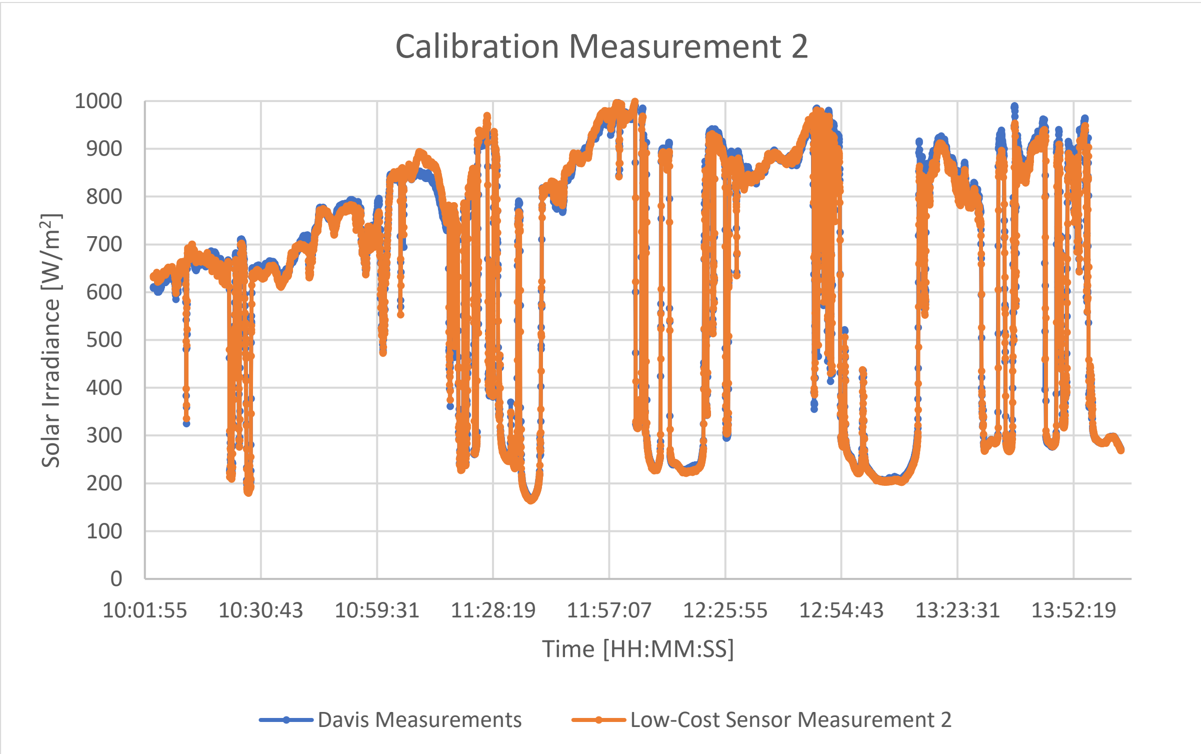
Δ measurement = Measurement n+1

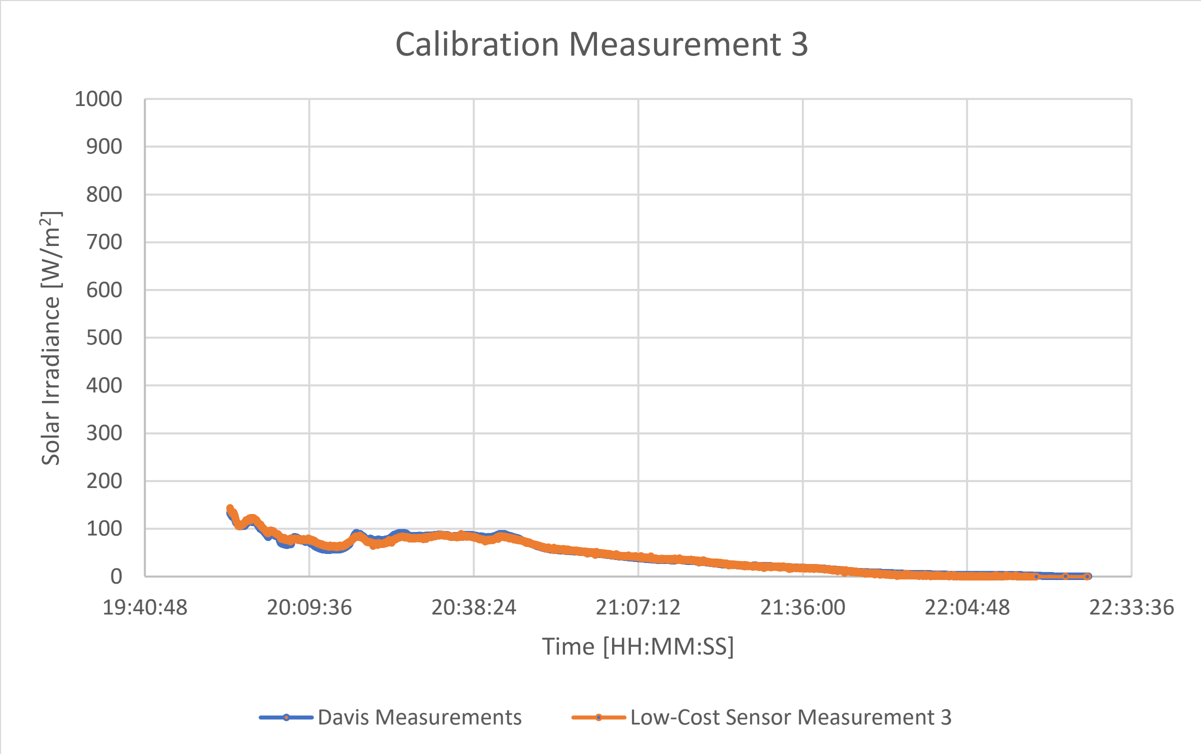
Δ measurement

Example structure { 7.2, -1, -1, -1.7, -0.8, -0.4, -0.2, 0.1, 0.2 } ← 1 1/ ser

Appendix D: Calibration Measurements







Calibration Measurement 3
Solar Irradiance [W/m²]
1000
900
800
700
600
500
400
300
200
100
0
19:40:48
20:09:36
20:38:24
21:07:12
21:36:00
22:04:48
22:33:36
Time [HH:MM:SS]
Davis Measurements
Low-Cost Sensor Measurement 3

Appendix E: Test Measurements