# DEVELOPING A SENSOR NETWORK FOR REAL TIME TEMPERATURE MONITORING IN ENSCHEDE



Gemeente 🔀 Enschede UNIVERSITY OF TWENTE.

This report presents the reader the methods and the results of the research on creating a sensor network around the city of Enschede

This assignment is commissioned by H.J. Scholten and R.G.A. Bults as part of the cooperation between the municipality of Enschede and the University of Twente. The project has been done by:

*Tom Onderwater (s1473832)* 

This research is done in order to complete the bachelor of Creative Technology at the University of Twente.

February 2018

Examination committee:

Supervisor University of Twente: Hans J. Scholten

Critical Observer: Richard. G.A. Bults

# Acknowledgements

This report is the result of months of hard work and cooperation between many people. First of all I would like to thank both my critical observer and GP Supervisor Richard Bults and Hans Scholten for the opportunity to work on this assignment and for the great feedback given by them. Hans has helped me with the electronics and to get the communications working. He has been a great help in getting the project done.

Richard has been a great inspiration and has been full of ideas, thanks to him the sensor nodes have come out working as well as they are right now. He has been full of support and was always approachable for questions and feedback.

Next I would like to thank Hendrikjan Teekens and Rik Meijer, the contacts at the municipality, Hendrikjan has supported the project greatly and provided valuable information like contacts throughout the project. Rik has helped with the deployment of the sensor nodes and in the end placed them all around the city.

Finally I would like to thank Yoann Latzer, my project partner, he has worked on visualization part of the the project and I feel like it has been a cooperation project all along. No decisions were made without first consulting Yoann and or making him aware of any changes. He has helped with all the presentations and all the preparations for the project.

# Table of Contents

Acknowledgements	
Table of Contents	
Abstract	7
Chapter 1 - Introduction	8
1.1 The Internet of Things	8
1.2 LoRaWan	8
1.3 The Things Network	9
1.4 Rapid prototyping	9
1.5 Urban Heat Island Effect	9
Effects:	10
Chapter 2 - Statement of Research Questions	11
Chapter 3 - State of the Art on temperature nodes deployment	12
3.1 Heat measurement projects	12
3.1.1 Belgrade and Pancevo, in cooperation with Sony Ericsson:	12
3.1.2 DIMAP-FactorLink, detection of forest fires	13
3.1.3 Array of Things	13
3.1.4 Natural urban ventilation optimization	14
3.1.5 Urban Microclimates study London	14
3.2 Temperature measurement technologies	15
3.2.1 Radiation Shielding	15
3.2.2 Types of temperature sensors	15
3.2.3 Digital home weather stations	16
3.3 Communication Protocols	16
3.3.1 LoRaWAN	16
3.3.2 IEEE 802.15.4	16
3.3.3 Bluetooth	16
3.3.4 WiFi	16
Chapter 4 - Methods and Techniques	17
4.1 Design process:	17
4.1.1 Testing methods:	18
4.1.2 Iterations:	18
4.1.3 Multipart design	19
4.2 Requirements and Challenges	19

4.2.1 Requirement: Sensors need to be placed outside of a controlled env	vironment 19
4.2.2 Requirement: The temperature has to be measured accurately.	20
4.2.3 Requirement: Sensors need to communicate with server	21
4.2.4 Requirement: Sensors need to stay operational for at least one year	. 22
4.2.5 Requirement: Sensors need to be be deployed in publicly accessible	e locations. 23
Chapter 5 - Ideation	24
5.1 Housing	24
5.1.1 State of the Art findings:	24
Chapter 6 - Specification	26
6.1 First generation design method	26
6.2 Design V-1	27
6.2.1 Design sketches	27
6.2.2 3D model	27
6.2.3 Hardware	28
6.2.4 Software	29
6.3 Development planning Iterations	30
6.3.1 Development method for the first sensor:	30
6.3.2 Development method further sensors:	30
6.3.3 Example of Sensor deployment:	30
6.3.4 Full feature list:	31
6.4 Preliminary tests:	32
6.4.1 GPS test	32
6.4.2 TTN connection test	32
6.4.3 Temperature measurement test:	33
Chapter 7 - Realisation	33
7.1 Development first prototype	33
7.1.1 Test Conditions	34
7.1.2 Test Requirements:	34
7.1.3 Test Results:	35
7.1.4 Proposed Changes based on test results	36
7.2 Development second prototype	37
7.2.1 Testing of second prototype	38
7.2.2 Test Conditions:	38
7.2.3 Test Requirements:	38
7.2.4 Test Results:	39
7.2.5 Changes made:	40
7.3 Development third prototype	40
7.3.1 Deployment	40

7.3.2 Locations	41
7.3.3 Test Requirements:	41
7.3.4 Results:	42
7.4 Development final prototype	43
7.4.1 Testing of new charging circuit:	44
7.4.2 Results	44
7.4.3 Final prototype deployment	45
Chapter 8 - Evaluation	46
8.1.1 Time frame	46
8.1.2 Developing the sensor	46
8.1.3 Data accuracy	46
8.1.4 Deployment	46
8.1.5 Long term	47
8.1.6 Project Goals:	47
Chapter 9 - Conclusion	48
Chapter 10 - Future work	49
References	51

# Abstract

To gain better insight in the urban heat island effect and for policy reasons a temperature network has been developed around the city of Enschede that should remain operational for at least one year. This network consists out of six temperature probes placed strategically around the city.

To develop these probes an iterative design cycle has been used in conjunction with rapid prototyping. Generations were built in in quick succession and put to the test in conditions representing real operation circumstances. A number of requirements and challenges were set up and tests were created so operation of a single sensor node could be confirmed.

A waterproof housing was designed to house the electronics and a heat shield to protect the sensor. The sensor nodes are built around the Sodaq ONE development board which incorporates both LoRa, a long range communication protocol and GPS whilst allowing multiple different sensor types to be connected to it. TTN was used as LoRa platform since it allowed multiple connections to be made and it provided good coverage of Enschede. To measure the temperature a digital temperature sensor has been used; the DS18B20 which has been calibrated to get temperature readings that are as accurate as possible. The sensor nodes are able to stay operational for at least one year by having a solar panel incorporated in the design which is able to produce charge at medium light conditions.

With the use of the iterative design method all needed challenges could be tested for and the long term operation of the sensor node is better known than would otherwise be possible due to the numerous tests conducted. The housing went through 2 major iterations which were enough to guarantee the waterproofness of the design. The design of the whole project was with future development in mind, allowing the project to go a number of different routes in the future.

# Chapter 1 - Introduction

Due to increasing urbanization and global warming, more attention is focused on the environmental impact of cities. Enschede is striving to become more of a smart city. Which can be defined as being: "A smart sustainable city is an innovative city that uses information and communication technologies (ICTs) and other means to improve quality of life, efficiency of urban operation and services, and competitiveness, while ensuring that it meets the needs of present and future generations with respect to economic, social and environmental aspects."[1]

Enschede currently has a number of projects running which are meant to improve the living conditions of its residents. One of these projects is the Ensketon project which is meant to manage the rain water distribution in the city. Based on the success of this project the city has decided to go further on the key environmental issues threatening cities in the Netherlands. Now attention is also paid to temperature mapping in relation to heat stress. In the last 30 years more than half of the heat waves have taken place since the beginning of measurements in 1901 [2], and they are increasing in number. The goal of this research project is to develop a temperature measuring system for the city of Enschede. To create this system sensor nodes will be used to measure the temperature. Multiple sensor nodes are needed to get insight in the temperature of the city of Enschede.

To create a network of sensors that spans the city of Enschede a lot of different kinds of technologies are necessary. In this introduction some of the technologies and terms referenced in this document will be explained.

### 1.1 The Internet of Things

The internet of things (IoT) is a term to describe the connection between devices and the internet. IoT was kickstarted by the decreasing costs and size of integrated circuits and uses various wireless communication protocols [3]. The IoT can be found anywhere, in home environments or placed around cities, nature or in vehicles. When the internet of things is deployed in an environment at home it falls under the category of domotics, examples of this are the Google Home and Amazon's Alexa. The system that is developed in this report is a city wide application of the internet of things aimed at gaining more insight. The described IoT application's target is enabling Enschede to improve its status as a smart city.

#### 1.2 LoRaWan

LoRaWAN is a low power Wide Area Network (LPWAN) specification intended for wireless battery operated things in a regional, national or global networks. [4]. LoRa is supported by the LoRa Alliance. LoRaWAN allows for secure bi-directional communication over 433, 868 and 915 MHz The LoRa protocol specifies that small packets of data are sent at a time. The communication is done from IoT device to gateway to endpoint and does not allow point-to-point communication [4]. For the current project point-to-point communication is not needed. LoRa is an open protocol and there are multiple infrastructures that provide gateways which use the LoRa specification.

#### 1.3 The Things Network

The Things Network (TTN) is an open application of LoRaWAN in which users place their own gateways and allow other uses to connect to these. TTN allows for a 1% duty cycle per device to ensure fair usage of the network. This means that devices can either send a larger packet with a longer interval or smaller packets with a shorter interval. After sending a packet the device will wait at least 100 times its transmission length before sending a new packet [5].

### 1.4 Rapid prototyping

Rapid prototyping is the creation of a (near) production quality product in significantly less time than it would take to produce it using conventional production means [6]. This production method allows for quick iteration cycles and better understanding of the behaviour of prototypes. The Rapid Prototyping method used in this project is 3D printing using an FDM (Fused Deposition Modeling) machine [7].

#### 1.5 Urban Heat Island Effect

An Heat Island is an area which is measurably warmer than the surrounding areas. This is mostly caused by urbanization, however it can also happen in nature. The temperature difference between an urban heat island and its surrounding environment is greatest at night when heat gets trapped by the lack of green as can be seen in figure 1 [8]:



Figure 1: Solar Radiation trapping in cities

Forests have a way of naturally regulating the temperature by evaporating water, a process where heat is lost [8]. Heat islands differ in severity between cities and also within a city as can bee seen in figure 2 [9].



Figure 2: Heat spread in Urban Environment during a typical day-night cycle

Areas that are more urbanized are prone to suffer more from the heat island effect than less urbanized ones. Factors that contribute to the heat island effect are [8, 38]:

-Low wind speeds

-Wide streets and low buildings

-Anthropogenic heat pollution

-Low concentration of vegetation

#### Effects:

#### 1.5.1 Health Problems

Higher than average temperatures can pose problems for risk groups containing the elderly, children, sick people, people with obesity, people with cardiac disease, and people with black skin color [10]. The higher than average night temperatures can also be problematic for the entire population when it comes to nightrest. Sleeping less than is required can reduce the efficiency at work and decrease physical health [11].

#### 1.5.2 Infrastructure Problems

The high temperatures which can occur in heat islands can be a risk to existing infrastructure. High temperatures can cause asphalt to become soft, forming potholes and rutting [12]. And vegetation in cities to dry out potentially worsening the urban heat island effect.

# Chapter 2 - Statement of Research Questions

For this graduation project the goal is to gather data so that the heat island effect can be measured in the city of Enschede. To answer how such a system can be created the following research question will need to be answered:

# How can a temperature data gathering system be developed which monitors selected locations in the city of Enschede in real time?

The challenges that measuring real time data across selected parts of the city will pose is to be divided in a set of sub questions that can be answered separately:

First of all it is necessary to know how a sensor node can be developed which can deal with the circumstances that are present outside of working lab conditions. For this the following sub question will need to be answered:

# Which challenges are present in developing a sensor node operating in an outside environment?

Secondly for measuring the heat island effect it is required that the temperature is measured with a certain amount of accuracy. The following sub question will need to be answered for this:

# How can a sensor be created that measures temperature as accurately as needed in most situations?

Thirdly it is important that the temperature data is sent to a place which is accessible for the municipality and the people involved in the project. For this the following sub question has been formulated:

# How can a system be developed that can connect wirelessly to a central database from anywhere in the city of Enschede?

Finally the sensor node will need to stay operational for a significant amount of time. The duration of the project is still unknown so some way to keep the sensor node powered has to be found. The following sub question will have to be answered:

# How can a sensor node be created that operates for at least one year without access to power sources?

# Chapter 3 - State of the Art on temperature nodes deployment

The deployment of a temperature sensor network is not a new concept and has been done in a number of variants. In order to learn how other people have deployed their temperature sensors and to gain insight in the design of remote temperature sensors a study has been done into projects that used heat monitoring in different insightful ways.

## 3.1 Heat measurement projects

#### 3.1.1 Belgrade and Pancevo, in cooperation with Sony Ericsson:

This project has been done to measure the temperature and air quality along the bus routes [15]. By introducing mobile sensors a larger area can be measured compared to having a single stationary sensor. The sensors are installed on the top of 65 city busses and send their data to a web application visible to the public. On top of environmental factors, traffic data is also collected and sent to the web application enabling users to choose their route based on these factors.



Figure 3: Waspmotes on busses

### 3.1.2 DIMAP-FactorLink, detection of forest fires

The project by DIMAP-FactorLink [13] was meant as a forest fire detection method, the project entailed 90 Waspmotes which sent 4 parameters(Temperature, Humidity, CO and CO2) every 5 minutes to a central server. The communication protocol made use of ZigBees. Working on the IEEE 802.15.4 standard for wireless personal area networks [15]. ZigBees operate at 868, 902-928 MHZ and 2.4GHz. The range of these devices varies per model, but can reach 3.2 km when operating at 2.4Ghz [16]. And theoretically as much as 105 KM when operating at lower frequencies [16].

The goal of the project was to reduce the chance of a forest fire occurring and to give insight in a forest fire which was already in progress.



Figure 4: Forest Fire Prevention

#### 3.1.3 Array of Things

The Array of Things is a project by the University of Chicago [17] which will eventually consist of 500 sensor nodes monitoring a large number of parameters. The goal of the project is to provide a large data set which app developers can use to create apps to better people's life. An example of such an application is creating routes for asthma patients along paths with better air quality. The whole project is posted on github and all the data that is gathered is accessible to anyone.



Figure 5: Array of Things Sensor Node

### 3.1.4 Natural urban ventilation optimization

This project was conducted by the University of Thessaloniki [18] in order to to monitor the heat island effect and the canyon effect. With temperature sensors and air flow sensors mounted on cars insight is gained on the temperature and airflow above the city's streets. The natural draft of air is modeled in this project to get a better understanding on how buildings will need to be constructed in the future to allow for better natural airflow to reduce the temperature in residents homes.



#### 3.1.5 Urban Microclimates study London

Research was done on the effects of building design and orientation on the microclimate of London [19]. Temperatures were measured for both the air and the ground on street level in highly dense urban areas in London. The goal of the project was to determine how the landscape of the city influenced temperature and how large temperature spikes could be prevented in the future.

## 3.2 Temperature measurement technologies

#### 3.2.1 Radiation Shielding

Most temperature sensors use a radiation shield to keep the sensor itself protected from the influence of light. This radiation shield is designed in such a way that air can flow freely through it [20].



*Figure 7: Air flow through radiation shield* 

#### 3.2.2 Types of temperature sensors

There are a number of methods to measure temperature, first they are spread in three different groups, invasive, semi-invasive and non-invasive [23]. These represent how much the medium that measures temperature comes in contact with the medium to be measured. Each of these types have different measurement methods and price points, the cheapest methods being the invasive end and the most expensive being the non-invasive end. Since the medium that is supposed to be measured is extremely large the invasive one is prefered.

#### List of invasive temperature measurement systems:

- Liquid based temperature measurement, examples of these are mercury based thermometers [21, 23], these sensors are often analog but also exist in a digital form
- Gas based [21, 23], an example of these is a temperature sensor working with carbon nanotubes as the cathode and anode. By using different gases it is possible to detect temperature [20, 23].
- Thermal expansion of solids, amorphous materials, e.g. glass and metals have an equal change in all directions when they change in temperature [10, 23]. This can be used to measure temperature.
- Thermoelectric devices [23]
- Semiconductors [23]
- Diode thermometers [23]
- Fiber optic probes [23]

- Capacitance thermometers [23]
- Noise thermometry [23]
- Quartz thermometers [23]
- Paramagnetic and nuclear magnetic resonance thermometry [23]

#### 3.2.3 Digital home weather stations

Digital home weather stations often consist out of two parts, a part that is placed inside providing the information and a part that is placed outside, logging the temperature and humidity. These sensors are not shielded and must be placed in the shade for accurate measurements. The data is transmitted by a low power RF signal on 433MHz [24].

#### 3.3 Communication Protocols

#### 3.3.1 LoRaWAN

As described earlier LoRa is a modulation format that is generated by dedicated transceiver chips [4, 26]. It is able to send and receive data from a series of gateways they are connected to. The theoretical range of LoRa is only limited by the power of the transceiver. But it actuality is limited due to regulations. In Europe 868MHz is used to communicate. LoRa has a number of infrastructure suppliers in the Netherlands including KPN [28] and TTN [27]. It is both easy to use and easy to communicate with and is specialized in low power consumption.

#### 3.3.2 IEEE 802.15.4

IEEE 802.15.4 [25, 26] is a standard which specifies the physical layer and media access control for low-rate wireless personal area networks (LR-WPANs). It has multiple applications including the ZigBee, ISA100.11a, WirelessHART and MiWi. It can also be used with 6LoWPAN and internet protocols to build an embedded internet. This protocol has a large set of applications and comes in almost any range.

#### 3.3.3 Bluetooth

Bluetooth operates at 2.4GHz, and is capable of maintaining a relatively high bitrate of 1Mbs [26]. The 2.4GHz frequency puts it in the ISM (industrial, Scientific and Medical) band shared with a lot of other protocols like WiFi and IEEE 802.15.4. This makes the amount of noise on the channel great and reduces the effective range to a maximum of 150 meters.

#### 3.3.4 WiFi

WiFi operates at both 2.4GHz and 5.8GHz [26], with the latter having less noise on the channel. The higher frequency reduces the penetration power of the signal reducing the effective range. WiFi has the highest bitrate compared to other IoT protocols.

# Chapter 4 - Methods and Techniques

### 4.1 Design process:

To create a system which can be placed outside and continue operating for a given amount of time a number of challenges need to be overcome. When operating outside of a controlled environment the continuously changing weather will be detrimental for any system. The temperature will go outside of the range in which most consumer grade electronics are meant to function and the presence of rain will force the design in a certain direction.

Since the system that will be developed is meant to stay operational for at least one year some way to keep it powered must be devised.

Any system that operates outside of a controlled environment will come in contact with a number of expected and unexpected situations. Some situations can be prepared for and can be tested for. There will however also be situations that could not have been foreseen. This is why the iterative design method will be used in the creation of the sensor node. This way it is possible to test for real world situations and change the design accordingly.



Figure 8: Iterative design process model [29]

The iterative design process is chosen because of the time constraints that are put up for the assignment. In order to ensure that the final system that will be produced is functional, each iteration should test a number of categories. Even the first iteration of a prototype should be

made in such a way that it could theoretically function as a working prototype (if everything is designed correctly) this will most likely not be the case, but every mistake that is made and recognised is one that can be removed in the next iteration in the prototype. The goal of the iterative design process is not to create a perfect prototype from the start but to be able to identify as many challenges as possible per iteration. By doing so each version can be evaluated and the successful features can be passed on to the next iteration whereas any challenge that was not overcome, can be improved upon. When faced with situations that are continuously changing experience is the best method to improve.

Iteration	Research goal
1	Proof of concept, feasibility test in real world situation
2	Improvements made on first iteration, All required features present
3	Fully functional prototype attempt
4 and later	Potential bug fixing

Table 1: Goals per iteration

### 4.1.1 Testing methods:

To test whether the current generation of a prototype is operational the challenges that the prototype will face will have to be identified beforehand. Since it is most likely that a physical prototype will be made in the form of a sensor node the challenges will need to be defined for such a system. Some of the challenges can be determined by brainstorming and some can only be identified by creating a working system.

Tests will be performed by placing the sensor node in a situation that most closely resembles the actual operational situation. Or if possible by placing the sensor node in the actual operation environment. During a test the data coming out of the sensor node will be monitored. Based on this an appropriate plan of action will be determined for the next generation. After each test the prototype will be analyzed and possible improvements will be placed in a table.

### 4.1.2 Iterations:

Rapid prototyping will be used to create a new iteration quickly. The most likely candidate for quick production will be the use of 3D printers since earlier projects have shown it to be very capable of producing waterproof shells. A 3D model will be created for each prototype and then tested. Mistakes made in the model will be fixed within the same 3D model. If a larger redesign is needed because of fundamental mistakes in a 3D model this will be the prefered plan of action.

#### 4.1.3 Multipart design

By using a design consisting out of multiple parts two advantages are gained and one disadvantage arises: Each iteration of the prototype will be able to be more complex than would be possible with a single part design and issues with a part of the design can be resolved more easily. In exchange for this the usage of multiple parts is capable of compromising the waterproofness of the design. This can be solved by allowing larger tolerances in the design and the usage of proper finishing tools and materials.

## 4.2 Requirements and Challenges

Certain requirements will need to be set up to produce the sensor nodes. These requirements are mostly based on the research questions. Each of them has a number of challenges that most likely will have to be faced. For each of these challenges a potential testing method is explained. Some of these tests will actually be performed, and some will not be performed based on the needs of the project cycle.

#### 4.2.1 Requirement: Sensors need to be placed outside of a controlled environment

Since the sensor nodes will have to operate outside, the operating environment cannot be controlled. For the project it is important that the sensor nodes are kept operational at all times. Below the challenges that each sensor node might face are described and how this will be tested for.

#### Challenge: (Splash) waterproofness.

The sensor nodes will be placed in a location in which large amounts of water can come in contact with the outer shell. To ensure that the inside components stay clear of the water the sensor node has to be proven to be at least splash waterproof.

#### Test:

The sensor node should be tested for waterproofness, this will be done by pouring water over the housing with a piece of paper towel inside, any leaks should be visible.

#### Challenge: Freezing

The sensor node will be placed outside during the winter and will most likely face temperatures below the freezing point. Lithium batteries are normally not meant to be charged when operating in subzero temperatures and discharge temperatures are limited to -20 °C [31]. To ensure that the sensor node batteries will have a lifespan for at least a few years this should be taken into account.

#### Test:

The sensor node will be placed outside with a cheap battery. If the sensor nodes stops operating at temperatures below 0 degrees the system is not frostproof.

#### Challenge: Heat stress

The sensor node will also face temperatures which can lead up to 40 °C in certain areas. [37] Direct sunlight might also greatly increase the temperature on the sensor node. *Test:* 

To test the resistance to heat an oven will be used which will be set at 50 °C. If the sensor node stays operational it will at least have proven to be somewhat temperature resistant. It is impossible to know what sort of temperatures the sensor node will face in reality but 50 °C can be seen as a reasonable worst case scenario.

#### 4.2.2 Requirement: The temperature has to be measured accurately.

The goal of the project is to provide temperature measurements around the city so that the heat island effect can be visualized. To do so accurate measurements are necessary.

*Challenge: Temperature measurements have to be accurate without radiation sources.* In controlled situations the sensor node should be able to accurately measure the temperature of the surrounding air.

Test:

The first test will include the accuracy of the temperature probe. The probe will be placed in freezing and boiling water in order to calibrate with both the freezing temperature and boiling temperature of water.

*Challenge: Temperature measurements have to be accurate with radiation sources present.* In situations where the sensor node will be exposed to sunlight the temperature still has to be measured accurately.

Test:

After the temperature probe has been calibrated it needs to be able to retain its accuracy when faced with radiation sources. A light with a light bulb will act as radiation source for this test. The temperature should not spike when it is turned on for a while.

#### Challenge: The measurements have to be consistent with each other.

Temperature sensors are known to drift around whilst the actual air temperature does not change.

#### Test:

The temperature will be measured repeatedly and checked with a calibrated temperature sensor. A maximum drift of +- 0.25 °C is permitted.

#### 4.2.3 Requirement: Sensors need to communicate with server

When the data has been gathered it needs to be sent to a server so that it can be used for other applications. Direct communication with the server is important since it should be a live temperature measurement system. The most obvious way to send the data is to do this wirelessly due to the distance between the sensor nodes and the server.

# *Challenge: The range has to be sufficient for each sensor node to be able to cover most of the city.*

To cover most of the city, each sensor node should have sufficient range to be able to communicate with the TTN gateways. LoRa is meant for long range communication but still has a finite amount of range.

#### Test:

The range has to be tested by having the sensor node communicate with a far away access point.

#### Challenge: Limited network coverage.

TTN has a limited amount of gateways. The locations of the gateways will need to be identified and a planning should be made where the sensor nodes should be placed.

#### Test:

The network coverage will have to be mapped.

#### Challenge: TTN has a limited bandwidth.

TTN only allows for a limited amount of network usage. The maximum allowed data sending duty cycle is 1% as defined by the TTN fair usage policy [27].

#### Test:

The refresh rate and and the data packages will have to be tested in such a way that they will conform with the TTN's policies.

#### 4.2.4 Requirement: Sensors need to stay operational for at least one year.

To keep the sensor nodes operational design choices will need to be made. To keep the sensor nodes easy to deploy batteries will be used to power the system. When choosing to use batteries some challenges will be present to keep them supplying power for a sufficient amount of time.

#### *Challenge: The sensor will have to be powered by batteries.*

The batteries in the sensor nodes have limited capacity. The sensor node will need to be able to function on batteries.

#### Test:

The sensor node will need to be tested whilst operating on battery power by having it run standalone.

#### *Challenge: The batteries should be able to recharge on solar power.*

To extend the lifetime of the project some way of recharging the sensor nodes' batteries should be devised. One easy solution is to add a solar panel to each node. *Test:* 

The voltages of solar panels are inconsistent [30]. A test is needed to prove the viability of charging the sensor node on solar power. This test will include different light intensities to measure the charge rate of the sensor node.

#### Challenge: Power usage may exceed charging rate

Solar panels only deliver a very small wattage. Especially during winter times periods when a very limited amount of sunlight are common. Coupling this with the decreased capacity of batteries in cold temperatures [31] it is possible that the sensor nodes will lose more charge than they gain from the solar panels.

Test:

The charge rate will have to be measured as well as the current draw. A net positive current flow is required.

#### Challenge: Batteries lose capacity in low temperatures

The sensors will be deployed during the winter and will face temperatures below the freezing point of water. It is essential that the sensor nodes stay operational in these conditions. *Test:* 

A test will need to be performed when it is freezing.

#### 4.2.5 Requirement: Sensors need to be be deployed in publicly accessible locations.

#### Challenge: The sensors should be easy to mount

It should be possible for a person who is not familiar with the project to deploy the sensor node without the need for any specialized tools.

#### Test:

The sensor node should be handed out to random people and they should be instructed to mount it in a location they deem appropriate.

#### *Challenge: The sensor should be able to resist physical contact.*

The sensor node will be deployed outside and people might be prone to touching the sensor node. Heavy weather might be able to throw debris on the sensor which might also damage it. *Test:* 

The sensor node should face a drop test of the typical deployment height. (3m)

#### Challenge: The sensor node could be stolen

The sensor node will be placed within range of the public and might get stolen. To improve the lifetime of the project the sensor nodes should reduce the incentive for people to take it with them.

#### Test:

It is hard to test whether or not people feel inclined to steal the sensor node, precautions will be taken on each sensor node. The only way to test for theft is to deploy the sensor nodes.

# Chapter 5 - Ideation

The development of the sensor node can be split in three parts; The development of a housing capable of withstanding uncontrolled environments for electronics, The development of software capable of gathering the data and relaying this efficiently to the central server and the development of the electronics which form the sensor node. To maintain a high development speed iterations of each component will be tested and redesigned until a design which works sufficiently is created.

## 5.1 Housing

Based on the challenges that will be faced a sensor node design can be created. The paper prototype (mockup) stage will be skipped since this would not be a functional prototype as would be required for use case tests performed outside of a controlled environment.

#### 5.1.1 State of the Art findings:

By looking at other air temperature probes a basic starting point can be defined for the sensor node.

#### Shielding:

To accurately measure air temperature it is essential that the temperature sensor is in direct contact with the surrounding air but not exposed to sunlight. This is why a radiation shield has to be designed



Figure 9: typical air flow of radiation housing.

It is important to keep track of the air flow. Hot air rises so it should be able to leave the radiation shield. If hot air is unable to freely leave the shielding it will build up in the top of the shielding and the measured temperature will be higher than the actual air temperature. As shown on figure 9 the hot air is able to leave the radiation shielding by the pressure from the cold air flowing in the shielding. As long as the lowest point of the top shielding ring is higher than the location of the temperature sensor no build up of hot air will be measured since this

would have already left the radiation shielding before being able to reach such a low point in the top ring of shielding.

#### Colour:

The radiation shielding should be coloured white as white reflects the greatest amount of light in the visible as well as the invisible spectrum of all available colours [33]. The body of the sensor node could be any colour since it is less critical that this reflects light. It is however good practice to make the entire sensor node white since the electronics inside could overheat else the materials used could lose strength.

The initial sensor nodes will be 3D printed out of PLA which has a low temperature tolerance for a plastic. PLA becomes soft when exposed to temperatures ranging from 70 - 80 °C [32]If it stays exposed to temperatures like this it will start to deform compromising the integrity of the housing.

#### Water proofing:

The sensor node will be placed outside meaning that the electronics must stay dry. The design should be made in such a way that no water is able to enter the housing. Parts that cannot be waterproofed in the design by form will be closed by using hot glue.

#### Attachment:

The sensor node should be placed across the city. A straightforward attachment point is to use the light poles that are placed everywhere in the city. An easy way to attach the sensor node is by using a strap going through a slot in the housing which can be wrapped around a light pole.

# Chapter 6 - Specification

## 6.1 First generation design method

For the first round of the iterative design method the first four phases of the waterfall design method are used. Each step after the requirements will be shortened to keep time for multiple iterations. The maintenance step will not be included in the iterative design method until a final iteration is reached.

For each iteration a design step will be quickly followed by an implementation phase, which will include a number of tests. After these tests have been conducted there will be a verification phase in which the functional behaviour of the current prototype will be analyzed. This will then loop back to a new design for a next iteration if deemed necessary. Each cycle in the iterative design process will keep the same requirements since these do not change over the course of the project. Although unlikely, it is possible that changes in the requirements are needed. When this happens, the requirements will be changed.



Figure 10: Shortened waterfall model

## 6.2 Design V-1

#### 6.2.1 Design sketches

Based on the design criteria sketches have been made showing how the sensor node should look like roughly.



*Figure 11, 12: Rough design sketches* 

#### 6.2.2 3D model

With the sketches, a 3D model can be created (figure 13) since changes can still be made in CAD it gives a much better overview how different parts will operate together.



Figure 13: 3D design of version 1

#### 6.2.3 Hardware

The heart of the system will be formed by the Sodaq ONE development board, (Appendix A - full specification list), this board is compatible with the Arduino IDE and is capable using the LoRa protocol to communicate with a centralized server. LoRa was chosen because of the extensive coverage in the Netherlands and it's ease of use.

The temperature sensor used is a standard analog temperature sensor in a TO220 package. This will be upgraded in the future.

To design the electronics a breadboard and the Sodaq One development board will be used to test how all components would interact with each other. The electronics design of the first prototype will be kept simple because of 2 reasons:

-The parts necessary to create the final prototype were not present yet -The first iteration is a proof of concept test and not meant as a full system test

To power the system a power bank will be used since this provides a cheap and easy solution with a high power output.

#### 6.2.4 Software

Before writing the software the operation of the sensor nodes has to be understood. This is why a process flow diagram is created to visualize the operation of the first sensor node. The Sodaq ONE will make use of the TTN network since this is open source and isn't limited by a maximum amount of sensors for a single application [27, 28]. TTN also has a good coverage in the city of Enschede. (Appendix B)



Figure 14: Software process flow diagram

## 6.3 Development planning Iterations

#### 6.3.1 Development method for the first sensor:

Before building the first sensor node, it is important to get an overview of what should be tested and when a test should happen in the iteration cycle. When setting up the requirements potential testing methods for different challenges have been explored. These testing methods are prone to changes based on the necessities of the project. During the first iteration the key challenges should be tackled, these are:

- Waterproofing
- Battery testing
- Temperature reading
- LoRa connectivity (TTN connection)

When these have all been successfully tested, a working prototype has been created. It is important to note that when these have all been found in working order, the design process is not done. Improvements are necessary in almost all of these categories. Having a waterproof housing does not imply that it is the ideal housing, it might have limited air flow, use a lot of material to produce, be difficult to mount or anything else which would hinder the project. The same goes for the temperature sensing or the batteries. The temperature might be inaccurate or the battery might not be able to function for over 1 year. The key function of a first prototype is to be a platform on which improvements can be made.

#### 6.3.2 Development method further sensors:

After the first sensor node has been produced it can be assumed that a lot has been learned about creating a temperature sensing system. The second version, and any version thereafter should improve based on the conclusions drawn from testing earlier versions, and add new functionalities which have not been tested in earlier versions. An ideal situation would be one in which the second version would introduce all sensors and features that the final version would need to have to function for the project. These would not necessarily function properly, but they will be tested.

#### 6.3.3 Example of Sensor deployment:

In order to decide which features are required, a scenario is necessary to describe how the sensor nodes will be deployed. It is equally possible that the sensor nodes will be deployed by the person responsible for the project as it would be that the municipality is responsible for deployment. Regardless of the party deploying the sensor nodes, they should work accurately. For it to work accurately, not only temperature is important, but also position. This is why the sensor nodes should be fitted with a GPS sensor. If GPS is used, it is possible to deploy the node anywhere outside. It will also be possible to relocate the sensor node and to have it update it's position automatically.

The second feature that enables easy deployment is an easy mounting mechanism. The most logical option would be to mount the sensor nodes to pre-existing light poles or trees. This way, the sensor nodes can be placed on a reasonable height and they will not damage any city owned property.

A third feature would be radiation shielding, when placing a sensor node, it should give accurate measurements regardless if it was placed in the sun, or in the shade. A place in the shadow would give better temperature readings, but limit the usability of the fourth feature: Power generation. Batteries only have a limited amount of charge, it is very likely that it is needed to charge them over time. The addition of a solar panel will greatly increase the lifespan of the project.

#### Scenario:

The sensor node is brought to its deployment location. It is not exactly known where it can be placed. A suitable light pole with a height of more than 4 meters is needed. When a suitable light pole is found, the researcher climbs into the light pole with the use of a ladder. When the required height is reached, the sensor node is attached to the pole using a strap. All this time the sensor has been turned on. The researcher climbs down the ladder and drives away. The sensor node has been placed with the solar panel facing south, and the batteries start charging. When it is night, the sensor node tries to get a fix on its position using the GPS sensor that is built in. The fix is successful and the sensor node is now transmitting its accurate location as well as its temperature to a remote server.

6.3.4 Full feature list:

- Temperature sensor
- Battery powered
- Waterproofing
- LoRa connectivity
- GPS
- Radiation shielding
- Mounting point
- Power generation

## 6.4 Preliminary tests:

Before going on with the construction of the first sensor node, it is important to test some subsystems. If any of these subsystems is not working correctly, it should be found out as soon as possible since it will take a while to order new components online. These systems include the GPS, the Lora chip and the temperature sensor.

#### 6.4.1 GPS test

The first test was to try to get a fix on the position of the sensor node. A test version of the software was written which only tried to get a fix on the current location for as long as needed. The tests were performed in an inside environment as well as in an outside environment. The tests were left running for 15 minutes to obtain a fix that was as accurate as possible. *Results:* 

The outside test failed to obtain a fix. This was most likely due to the location of the test at the back entrance of the Zilverling, a 5 story building at the University of Twente. Only 30% of the sky is unobstructed by the surrounding buildings so it can be imagined that this was the main factor why the test failed.

The inside test obtained a fix within 12 minutes, this test was performed on the balcony in the SmartXp. The accuracy of the GPS fix was however not within the specified 5 meters minimum of GPS [34]. This is most likely also due to the small LoS (Line of Sight) to the GPS satellites or bouncing of the signal. GPS requires satellites to be spread apart over the sky to be able to reach its maximum accuracy.

#### Findings:

To test the GPS properly during the first iteration an environment has to be selected that has a mostly unobstructed view of the sky. The roof of the Zilverling could be a place where such a test could be performed.

#### 6.4.2 TTN connection test

The TTN connection test was meant to test the bandwidth and the connection protocol with TTN. OTAA activation was selected since this meant higher security and a simpler deployment of the sensor nodes. It did require the DevUID's and Application Keys to be hardcoded in each separate sensor but this was not seen as a big problem compared to the advantages OTAA offered.

Results:

Using OTAA activation and a 16 character message the maximum rate at which data could be sent was approximately 1 message per 30 seconds. The connection was easily established and no problems were foreseen for the first systems test.

#### Findings:

The same method as used in the small system test will be incorporated in the first systems test. The protocol will be left unaltered, the data rate will be changed to 1 message per minute.

#### 6.4.3 Temperature measurement test:

A LM35DT temperature sensor was used for the first temperature test. This temperature sensor gives an analog reading and uses the development board's onboard ADC. This meant that the maximum resolution was hindered by the 11 bit ADC range. *Results:* 

The temperature sensor gave a reasonable indication of the on board temperature. The maximum resolution was 0.5  $^\circ\text{C}.$ 

#### Findings:

To get a better resolution a switch to a digital temperature sensor is needed. This way the resolution is no longer hindered by the 11 bit ADC.

# Chapter 7 - Realisation

With the information retrieved during the specification phase it is possible to create a first working version. By testing some critical parts before constructing a sensor node knowledge has been gained on the behaviour of LoRa, GPS and the temperature sensor.

## 7.1 Development first prototype

The first iteration was constructed using a 3d printer. The original color was green but it was painted white to better reflect light and to provide some protection for the plastic materials used.



Figure 14: First sensor node

## 7.1.1 Test Conditions

The first iteration was placed on the roof of the Zilverling to get a more direct line of sight to the GPS satellites since preliminary tests showed that getting a fix with GPS could take a long time when an open sky is not available. During the testing period freezing temperatures, rain and snow were present.



Figure 15: Sensor node deployed on roof of Zilverling

#### 7.1.2 Test Requirements:

#### GPS:

For a successful test a fix has to be made within 5 minutes.

#### Temperature:

For a successful test the temperature needs to be recorded for the duration of the test.

#### Housing:

For a successful test the inside components need to stay dry for the duration of the test

#### Power:

For a successful test the sensor node needs to stay powered for the duration of the test.

#### LoRa connectivity:

For a successful test connection with the TTN network needs to be established and data needs to be transmitted.

#### 7.1.3 Test Results:

#### GPS:

The sensor node was capable of getting a fix on its current location. The accuracy of this fix stayed within 50 meters, which was comparable with earlier tests.

#### Temperature:

The temperature sensor stopped working soon after the test started displaying a value of -127 °C. Later review showed that this was because a pin of the TO220 package came disconnected. Before the temperature sensor stopped working, the temperature readings were noisy, often fluctuating more than 1 degree above or below a median reading. The temperature sensor used was an analog one and thus limited by the accuracy of the ADC on board of the development board. During the specification phase it was determined that a digital temperature sensor would be required.

#### Housing:

After retrieval of the sensor node signs of condensation were found inside. This is most likely due to the total enclosure of the sensor node. No air circulation is allowed in the housing causing water vapor to settle on the slightly warmer electronics. No components were broken in this test.

No water or mud had entered the device showing that the enclosure was working sufficiently in this regard.

#### Power:

The sensor node stopped transmitting data after 14 hours of usage. The full test lasted 2 days. It is believed that a power bank is a non ideal solution for keeping the system powered due to the inefficiency of the 5V regulator located inside.

#### LoRa connectivity:

For 14 hours the sensor node transmitted the data correctly to the TTN service. No frames were dropped in the time of usage.

#### 7.1.4 Proposed Changes based on test results

#### GPS:

The time at which the GPS obtained a fix was reduced to under a minute. A 60 second cut off will be implemented to reduce power draw by the GPS module. Also a timer will be introduced which will only allow a GPS fix to occur at a set time interval

#### Temperature:

The robustness of the temperature sensor was inadequate, a temperature sensor with a more solid build quality will be used in future iterations. The most likely candidate is the DS18B20 digital temperature sensor. This is often used in applications working outside due to its waterproof casing.

#### Housing:

The condensation found in the housing showed that complete isolation from the outside is unwanted. Future designs will have to take care of free air flow without compromising the water proofness.

The design of the housing was found bulky and inefficient. A more modular design that is smaller will be created for future tests.

#### Power:

The usage of a power bank is not ideal. For future designs a standard LiPo battery will be used. Current will only be drawn when necessary.

A way to generate power will be researched for future applications.

#### LoRa connectivity:

The current method of connecting to the TTN network was found to be adequate for future designs.
## 7.2 Development second prototype

During the testing of the first prototype, several possible improvements were found. The second version has a radically different design compared to the first version.



Figure 16: Change overview V-2

The only part that was kept the same was the radiation shielding. The shielding functioned without issues and allowed air to flow freely. Only the legs connecting them to each other were made more solid since these had a tendency to break when snapped together. Below are the rest of the changes made on the second prototype.

### List of changes:

- Thinner housing for reduced material usage
- Slots for additional modules for future upgrades
- Breathing hole for improved air flow
- External connection for solar panel or other sensors
- Solar panel for power generation
- Improved mounting mechanism for improved stability
- New logo for recognition
- Improved bottom cap design

### 7.2.1 Testing of second prototype

Testing the second iteration of the sensor node is made more difficult since it requires a longer testing period to prove the battery capacity, the sturdiness of the sensor and the way it reacts to more hostile environments.

### 7.2.2 Test Conditions:

The tests were performed in mid December during a colder than average week with temperatures regularly going below 0°C. The sensor was placed outside and tested for durability as well as temperature accuracy. This test lasted for 4 days in which the sensor is required to keep broadcasting with an update rate of one measurement per minute.



Figure 17: deployment V-2

### 7.2.3 Test Requirements:

### GPS:

A successful test would require that the sensor node would update it's position each 255 cycles (each cycle taking one minute) meaning that the sensor node would update its GPS position every 4 hours and 15 minutes.

### Temperature:

A successful test requires that the sensor node gives no large abbreviations on its own measured temperature, meaning that the temperature readings will not move much around a median value. Changes larger than a full degree with no changed circumstances meant that the temperature sensor used was too inaccurate to be used in the application that was intended. Accuracy to actual air temperature was not part of this test. *LoRa connectivity:* 

For this test to be successful the system should adhere to the TTN rules of using no more than a 1% duty cycle (thus no packages being rejected by the network). Also at least 95% of the packages should arrive on the TTN server.

### Power:

The sensor node should stay operational for the entirety of the 4 days it is deployed. *Housing:* 

The housing of the sensor node should stay in one piece to be regarded functional. Also no water and or mud should be able to enter either of the electronics compartments of the sensor node. Based on results from the previous test the presence of condensation will also be checked for.

### 7.2.4 Test Results:

### GPS:

The GPS showed mixed results in this test. Some fixes showed great accuracy of the GPS sensor, some even showing as little as a 1 meter deviation of the actual location of the sensor node. Some fixes however were less accurate showing a distance of more than 100 meters of the actual location of the sensor node. This was due to the sensor node not having enough satellites to fix its location accurately, but enough to allow for a fix to happen. 4 satellites is the minimum for a fix but is not very accurate when taking a single reading [34].

### Temperature:

During the entirety of the test the temperature readings stayed fairly consistent with no jumps larger than 0.25 degrees between readings. These jumps were measured in the night since then the possibility of sunlight falling on the sensor, allowing greater jumps to be possible, can be ruled out. The temperature test is considered to be a success.

### LoRa connectivity:

The sensor node kept a stable connection with LoRa for the duration of the test. No packages were denied because of the possible over usage of the TTN bandwidth. *Power:* 

The sensor node kept power for the duration of the test. The charge of the battery did however lower during the duration of the test. The battery started with a voltage of 4.1 V and ended with a voltage of 4.02 V. Whether this was due to the low amount of sunlight or the state of the battery was unclear.

### Housing:

The sensor node was faced with snow, hail, rain and subzero temperatures but stayed intact. No water came in the solar panel housing as well as the main body. Also no signs of condensation were recorded when the sensor node was retrieved revealing that the addition of the air sharing hole was successful.

### 7.2.5 Changes made:

GPS:

The GPS location was unstable, there were moments in which the GPS fix was very accurate and moments in which it was more than 100 meters away from the actual location. A new protocol was written to get a fix from a satellite, now a minimum number of 6 satellites is required.

### Temperature:

The temperature readings where much less noisy than before, it is however still possible that they are not entirely accurate, the next iteration will have the temperature sensors calibrated. *TTN:* 

A 1 minute time interval is very rapid and approaches the 1% limit of TTN with the current message bandwidth. If more sensors will be added in the future the time interval should be lowered. A 2 minute time interval is selected for both battery preservation and less pressure on the TTN network.

## 7.3 Development third prototype

The third version was the first version that was mass produced, almost no changes were made to the design of the housing. Some holes were added for air circulation in the solar panel. Additionally the arrangement of the electronics in the solar panel was changed: hot glue was used to fix the electronics instead of electrical tape which should be more sturdy.

The most notable changes were made in the firmware. The new iteration used a sleep function to make the sensor node go in low power mode when it was not in use. Every two minutes the sensor node would wake up, get a temperature reading and transmit this to the remote server. Also the GPS function was changed to make it more accurate. When there are less than 6 satellites, the GPS fix will not be deemed accurate enough and the sensor node will attempt to get a new fix in the next cycle. This will happen for a maximum of 10 attempts. After that it will wait 255 cycles before it will attempt to create a new fix.

### 7.3.1 Deployment

Deployment was done by contacting people around the University and private companies asking if they were willing to house a sensor. These people / companies were selected based on their respective locations. The idea behind the sensor is that the sensor can be deployed by someone who does not know how the sensor operates or how it should be attached.

People given the sensor were only told to take the sensor home and place it outside. A strap was given to these people so they could attach it to a pole if this was present. The goal was to keep the sensor nodes in the field over the Christmas holiday, which lasts for 2 weeks.

### 7.3.2 Locations

In the end 4 sensors were deployed around Enschede. For privacy reasons the exact addresses will not be given in this document but approximate locations:

- Helmerhoek Noord
- Het Volkspark
- Wittebreuksweg
- G.J. van Heekstraat



Figure 18: Deployment locations

### 7.3.3 Test Requirements:

### GPS:

The new GPS protocol should ensure that there will be no more wrong GPS locations. All sensor nodes should transmit their exact location in all situations.

### Temperature:

A new method was written to get a temperature reading, each sensor has been calibrated with the first sensor, which in its turn was calibrated with boiling water and freezing water. To limit the amount of noise during a temperature measurement, 10 readings are taken in the span of 2 seconds and the average of these readings is stored as the current temperature. It should be visible that there is less noise on the temperature sensor by comparing consecutive temperature readings.

### LoRa connectivity:

Since the sensors are located all around the city, the coverage of the TTN network can be tested at this point. If successful, each sensor should be capable of sending data. *Power:* 

The sensor nodes will be outside for a much longer period in this test, they will stay outside for at least 2 weeks. if they stay operational during the entire test this will be regarded as a success.

### Housing:

The design of the housings has been proven to be waterproof. Since this is the first time the sensor nodes are mass produced, the individual quality of each housing will be tested in this long term test.

### Theft:

This is the first time the sensor nodes will be placed in public areas. The sensor nodes will be secured using zip-ties, if all goes well, none of the sensor nodes will be destroyed or stolen during the testing period.

7.3.4 Results:

GPS:

The GPS location was stable in this iteration, during the period in which the sensor nodes were regularly checked none of them displayed a false or inaccurate location *Temperature:* 

The temperature was more stable than in previous iterations. The accuracy is still unknown since due to the heat island effect higher and lower temperatures are expected to occur. None of the temperature readings were deemed illogical.

LoRa connectivity:

All the sensor nodes were able to transmit data. Some sensor nodes still managed to lose packages but they always resumed sending data later.

Power:

The sensor nodes stopped operating after 12 days, indicating that the batteries did not charge correctly. The actual cause will be investigated later.

Housing:

None of the housings showed signs of wear or leaking. This will be regarded as the final design for the housing.

Theft:

None of the sensor nodes were stolen. No changes in deployment will be made.

# 7.4 Development final prototype

The third iteration showed that most systems were working as required with the exception of one single feature: the solar panel. The sensor nodes stopped transmitting data after 12 days of deployment which is significantly less than a full year. This is why it can be concluded that the charging circuit did not operate as was hoped. Further research showed that the charging current on the TP4056 module operates typically with a charging current of 1A [35]. Since the charging current is formed like a bell curve and the typical current from the solar panel used is around 150 - 200 mAh [35]. The defined charging current should be drastically lowered to increase the efficiency.

arrent oc
IBAT
(mA)
130
250
300
400
580
690
780
900
1000

# **Rprog Current Setting**

Figure 19: Rprog (R3) current settings

A value of around 8K should be optimized for a charging current between 150 - 200 mAh, a reasonable current for a 1W solar panel while placed in the sun. The Rprog resistor on the TP4056 modules will be replaced with one of these in each of the sensor nodes.

### 7.4.1 Testing of new charging circuit:

Before implementing the new charging set up in each of the sensor nodes, its functionality must be proven. That is why a testing setup is required. The testing setup for measuring the ability of the new charging circuit to charge the battery using solar power is as follows:

A single solar panel will be opened and the voltage on the battery will be measured at every single moment. The setup will be placed under a 150W lightsource which closely resembles the light output of the sun on a sunny day. The voltage will be measured with the light off before turning the light on for a full hour and then measured afterwards, also with the lights turned off. If the voltage of the battery has increased in this period the new charging circuit is operational. If the voltage is the same or lower, the charging circuit does not work and another solution must be found.





Figure 20, 21: Battery voltage before and after 1 hour charging interval

### 7.4.2 Results

During the 1 hour testing period, the voltage over the battery went from 4.026 V to 4.041 V. This is only a small jump in charge, but it does show that the new charging circuit is capable of charging the batteries of the sensor nodes.

### 7.4.3 Final prototype deployment

After the completion of the new charging circuits the sensor nodes were ready to be deployed around the city. The municipality had offered to be responsible for the deployment of the sensor nodes. The sensor nodes were deployed by Rik Meijer, one of the contacts at the municipality



Figure 22: Deployment locations sensor nodes

In figure 22, the final locations of the sensor nodes can be found. The sensor node placed on university grounds has not been deployed by the municipality and is meant as test sensor for further iterations.

# Chapter 8 - Evaluation

Currently the sensor network is operational and sending data. As of writing, the sensor node network has been non stop sending data for 6 weeks, this is longer than any of the previous iterations. The sensors have worked fine in the cold weather and the batteries have proven to be able to cope with the cold weather.

### 8.1.1 Time frame

For the development of the sensor node a time frame was used (Appendix C) This time frame was kept strictly to make sure that the sensor network was ready in time and that the municipality was aware when timelines were met. The use of a timeframe with hard deadlines was a positive influence on completing versions on the set dates but they brought a lot of unnecessary last minute stress for some iterations of the sensor nodes due to the production time being longer than expected.

### 8.1.2 Developing the sensor

By starting quickly with the development of the sensor node, it was possible to get a good insight on how the sensor nodes would operate in the end. Due to the relative simplicity of a single sensor node, it was possible to make a lot of iterations which helped with identifying problems in an early stage. By having a lot of iterations it is possible to catch problems in an early stage and to engineer solutions for these. Some problems still occurred that could have been foreseen, for instance the problem with the charging circuit could have been foreseen if there would have been more research done into the charging mechanics of the TP4056. Thanks to the long term test the problem was identified and after performing research it was fixed for future generations.

### 8.1.3 Data accuracy

The sensor nodes have been calibrated on a single other sensor node. This calibration was done by measuring the temperature of known substances like boiling water and then reading the measurement. The other sensor were calibrated at room temperature until they would give the same reading as the first sensor node. This was done due to time constraints because the production of the sensor nodes took longer than expected and they had to be deployed on a specific day. The data seems to be fairly accurate yet it might be possible that some readings are off. Sensor node 1 is currently running on an older version of the firmware which does not have the calibration offset implemented in it yet. This sensor node should be re calibrated in the future.

### 8.1.4 Deployment

The sensor nodes have now been deployed by the municipality, this has helped the conflict of interest between the different departments in the municipality. Because the sensor nodes

were designed to be deployed on light poles, which are public property, permission was needed from the department that was responsible for the streets. In the start these did not agree with placing the sensor nodes on light poles.

Early readings show the heat island effect happening the way it was predicted. The locations are therefore probably sufficient. Some sensor nodes have difficulties getting their messages to the gateways, this can be due to the distance or due to the amount of traffic on the gateway. On average 50% of the packages are lost for sensor node 2 which is located in Enschede south east.

### 8.1.5 Long term

There are no reasons to predict that the sensor network will stop operating in the foreseeable future. In the future the housing might give up since a biodegradable plastic was used. When it is exposed to water and oxygen it will begin to dissolve slowly. PLA will degrade over a period of 3 years approximately.

### 8.1.6 Project Goals:

The goals of the project were all fulfilled, it was possible to create a sensor network that measured the temperature on different locations all over Enschede and to retrieve the data in real time.

# Chapter 9 - Conclusion

After 4 iteration cycles and the research done beforehand 6 sensor nodes were supplied to the municipality. With all of the sensor nodes now deployed and operating it is possible to answer the research question. In order to do that first each subquestion will be answered before answering the main research question.

# Which challenges are present in developing a sensor node operating in an outside environment?

Deploying a sensor node outside brings a lot of challenges, the first challenge is the location. A sensor node must be placed somewhere and it is not possible to place a sensor node wherever it is deemed necessary. There are rules and regulations which must first be overcome. The second challenge is placing the sensor node, it should be placed at a sufficient height and placed sturdy enough to not glide down. Lastly the weather plays a big role in the design of the sensor node. Proper precautions must be taken to protect against rain, snow and the sun.

# How can a sensor be created that measures temperature as accurately as needed in most situations?

To measure temperature accurately there are 3 main things that must be taken care of, accuracy, consistency and resolution. Accuracy can be gained by calibrating the sensor node and shielding it from the sun. Materials play a big role in the shielding process, a material is needed that does not retain heat well. PLA is a good choice for this. Consistency is more difficult to achieve, but can be achieved by having a large mass on the temperature sensor or by using smart algorithms. The DS18B20 has a significant amount of metal surrounding the temperature sensor which reduces the noise. This is coupled with an algorithm which takes multiple measurements and takes the average of these readings. Resolution is down to the temperature sensor itself, depending on the temperature range, a sensor node with a 12 bit resolution is prefered.

# How can a system be developed that can connect wirelessly to a central database from anywhere in the city of Enschede?

There are several technologies available that would allow a sensor node to communicate wirelessly with a centralized database from anywhere in Enschede. LoRa is the easiest solution and allows for good battery life. In terms of network TTN is the most scalable since it does not have a maximum amount of sensor nodes that can be on the network at a single time.

# How can a sensor node be created that operates for at least one year without access to power sources?

LoRa devices are meant to stay deployed in the field for years on a single battery but the findings in this research project showed that this does hold for the sensor nodes currently deployed due to the higher current draw of the GPS module and the rate at which data is transmitted. A solar panel allows the sensor nodes to stay operational for an indefinite time.

Together these findings give an answer to the main research question of the project: How can a temperature data gathering system be developed which monitors selected locations in the city of Enschede in real time?

The use of the iterative design model was a good way of producing a sensor node capable of operating outside of a controlled environment. In the end many versions were needed to produce a sensor node that operated sufficiently and will continue to operate for the foreseeable future. To create these versions the rapid prototyping method was invaluable. The sensor nodes measure air temperature relatively accurately and can be placed anywhere in the city. The use of LoRa and TTN was a good choice for the communication between the sensor nodes and the server. TTN has a limited uplink time, but is more than sufficient for the current generation of sensor nodes and the sensor nodes created in the foreseeable future.

The longevity of the project is still too be proven, currently the sensor nodes have been operational for 6 weeks and shown no signs of fatigue. There are still improvements to be made on the network but this is something that should be done in future iteration cycles.

# Chapter 10 - Future work

The sensor nodes are currently operational but there is a lot that can be done to improve on the platform since they were built as a proof of concept. The sensor node only contains a single sensor and has room for a lot more. The current firmware is operational but a lot of optimization can be done. Below are things that might appear in the future:

### • Adaptive update rates

The sensor nodes are currently sending data every 5 minutes, this is a fixed rate and does not change, it is therefore possible that some interesting data is not picked up. The main power consumption of power is done when the sensor node is sending data. It is possible that the sensor node wakes up more often and gathers data. If there are no changes in the data, it does not send anything. If the temperature is moving a lot however (due to cloudy weather with some sun) it might be relevant to send data more often. That is why in a future generation an adaptive update rate would be beneficial, both for the completeness of the project and for the power management of the sensor nodes.

### • More sensors

Temperature data is interesting, but there are a lot more interesting factors in city life if you want to measure the quality of life of the citizens. Light sensors are interesting to see when the sun is shining and how this relates with temperature data. CO2 sensors are also interesting to see how traffic influences the quality of air. A particulate matter sensor is capable of seeing how many small particles are floating in the air. There are a lot of sensors that can be added to the platform which would make the project a lot more valuable.

### • Stronger housing

Currently the housing is made out of a biodegradable plastic called PLA. This is an material that is easy to 3D print, and is bad at transferring heat, something a housing for a sensor node should be bad at. But the biodegradability and it's lower temperature resistance might make it a bad plastic for a sensor that is placed outside on the long term. In the future a new body is

required made out of a stronger material. This material will be harder to print and thus lengthen the production time, but since most of the development of the housing has been done already, this is not a bad thing.

### • Incorporation of Ensketon data

Currently there are two IoT networks from CreaTe that are meant to gather data in and around Enschede. The Ensketon project measures rainfall, it is possible that there is a link between the rate at which water drains away and the temperature over the last few days. Just like there is a link between temperature and rainfall. It might be worthwhile to couple the two datasets.

### • OTA firmware updates

TTN is currently in the works on making OTA (Over The Air) firmware updates a possibility [36] it is currently not possible yet for users of the network, but the plan is to release it in the future. Currently the nodes have to be collected or someone has to visit them to update the code running on them. An OTA update would benefit the security of the system and make it easier to make improvements.

# References

[1] R. Lea, Smart Cities: An overview of the technology Trends Driving Smart Cities, IEEE, 2017, pp. 2

[2] https://www.knmi.nl/nederland-nu/klimatologie/lijsten/hittegolven

[3] Alexey Vinel et all., Enabling wireless communication and networking technologies for the internet of things, IEEE, 2016

[4] A. Augustin et all., A Study of LoRa: Long Range & Low Power Networks for the Internet of Things, Sensors, 2016

[5] https://www.thethingsnetwork.org/docs/lorawan/duty-cycle.html

[6] C. K. Chua, K. F. Leong, C. S. Lim, Rapid Prototyping: Principles and Applications, Volume 1, World Scientific, 2003, CH1, pp.12-17

[7] <u>https://www.livescience.com/39810-fused-deposition-modeling.html</u>

[8] E. Sharifi, S. Lehmann, Correlation analysis of surface temperature of rooftops, streetscapes and urban heat island effect: Case study of central Sydney, Journal of Urban and Environmental engineering, 2015, pp. 3-11

[9] S. Craham, Life support, The political ecology of urban air, City, Newcastle University, 2015

[10] R. Basu, B. D. Ostro, A multicounty Analysis Identifying the Populations Vulnerable to Mortality Associated with High Ambient Temperature in California, American Journal of Epidemiology, 2008, Volume 168, pp. 632-637

[11] Why lack of sleep is bad for your health, <u>https://www.nhs.uk/Livewell/tiredness-and-fatigue/Pages/lack-of-sleep-health-risks.aspx</u>

[12] Climate Impacts on Transportation: https://19january2017snapshot.epa.gov/climate-impacts/climate-impacts-transportation\_.html

[13] Detecting Forest Fires using Wireless Sensor Networks with Waspmote, 2010 http://www.libelium.com/wireless\_sensor\_networks\_to\_detec\_forest\_fires/

[14] ZigBee Wireless Technology Architecture and Applications

[15] Smart City project in Serbia to monitor Environmental Parameters by Public Transportation with Waspmote 2012 <u>http://www.libelium.com/smart\_city\_environmental\_parameters\_public\_transportation\_waspm</u> ote/

[16] Digi XBee Family Features Comparison Chart, 2017 https://www.digi.com/pdf/chart\_xbee\_rf\_features.pdf [17] Chicago becomes first city to launch Array of Things, 2016 <u>https://news.uchicago.edu/article/2016/08/29/chicago-becomes-first-city-launch-array-things</u>

[18] Z.T. Ai, C.M. Mak, From street canyon microclimate to indoor environmental quality in naturally ventilated urban buildings: Issues and possibilities for improvement, Building and Environment, 2015, Volume 94, pp. 489-503

[19] M. Shahrestani et all., A field study of urban microclimates in London, Renewable Energy, 2015, Volume 73, pp. 3-9

[20] Radiation shield air flow: http://www.trutrack.com/intech/sol\_rad\_shield.html

[21] Zhang, Y., et all, High performance gas sensors with temperature measurement, Scientific reports, 2013

[22] E. Ramsden, introduction to temperature sensing, 2010. http://www.edscave.com/docs/ch2\_mechanical\_temperature\_sensors.pdf

[23] P.R.N. Childs, J.R. Greenwood, C.A. Long, Review of temperature measurement, Review of Scientific Instruments, 2000, Volume 71

[24] Reverse engineering weather station RF signals with an RTL-SDR, 2018 <u>https://www.rtl-sdr.com/tag/weather-station/</u>

[25] E. Callaway et all., Home networking with IEEE 802.15.4: a developing standard for low-rate wireless personal area networks, IEEE Communications Magazine, 2002, Volume 40, pp. 70-77

[26] 11 Internet of Things (IoT) Protocols You Need to Know About, 2015 <u>https://www.rs-online.com/designspark/eleven-internet-of-things-iot-protocols-you-need-to-know-about</u>

[27] <u>https://www.thethingsnetwork.org/country/the-netherlands/</u>

[28] https://www.kpn.com/zakelijk/grootzakelijk/internet-of-things/lora-netwerk.htm

[29] Iterative design process:

https://commons.wikimedia.org/wiki/File:Spiral\_model\_(Boehm,\_1988).svg

[30] L. Gao et all., Parallel-Connected Solar PV System to Address Partial and Rapidly Fluctuating Shadow Conditions, IEEE Transactions on Industrial Electronics, 2009, Volume 56, pp. 1548 - 1556

[31] Charging at high and low temperatures <u>http://batteryuniversity.com/learn/article/charging\_at\_high\_and\_low\_temperatures</u>

[32] How to succeed when printing in pla <u>https://www.matterhackers.com/articles/how-to-succeed-when-printing-in-pla</u>

#### [33] Albedo: Lighter Colors Reflect, Darker Ones Absorb, Heat, 2014

https://www.yaleclimateconnections.org/2014/12/albedo-lighter-colors-reflect-darker-ones-ab sorb-heat/

[34] GPS Accuracy

https://www.gps.gov/systems/gps/performance/accuracy/

[35] <u>http://www.rogerclark.net/lipo-battery-charging-from-a-1w-5v-solar-panel/</u>

#### [36] OTA firmware updates TTN

https://www.thethingsnetwork.org/article/firmware-updates-over-low-power-wide-area-netwo rks

[37] Tegen de hitte, groen en de opwarming van de stad, a case study for the municipality of Tilburg, 2013

https://ruimtelijkeadaptatie.nl/publish/pages/116505/tegen-hitte-groen-en-de-opwarming-van -de-stad-lr.pdf.

[38] R.A. Memon, D.Y.C. Leung, L. Chunho, A review on the generation, determination and mitigation of Urban Heat Island, University of Hong Kong, Journal of Environmental Sciences, 2007, Volume 20, pp. 120-128

# Appendix

# Appendix A: Sodaq ONE board and pinout

Microcontroller	ATSAMD21G18, 32-Bit ARM Cortex M0+
Compatibility	Arduino M0 Compatible
Size	40 x 25 mm
Operating Voltage	3.3V
I/O Pins	14, All can be used for digital and 12 log with PWM, UART, SPI and TWI (I2C)
Analog Output Pin	10-bit DAC
External Interrupts	Available on all pins
DC Current per I/O pin	7 mA
Flash Memory	256 KB
SRAM	32KB
EEPROM	Up to 16KB by emulation
Clock Speed	48 MHz
Debug	Serial Wire Interface
Power	5V USB power and/or 3.7 LiPo battery
Charging	Solar charge controller, up to 500mA charge current
LED	RGB LED
LoRa Microchip	RN2483 Module
GPS	uBlox EVA 8M
Accelerometer	LIS3DE

# LoRaONE external pin descriptions



Appendix B: TTN Coverage of Enschede:



### Appendix C: Project Planning and proposal: MEASURING TEMPERATURE IN ENSCHEDE

A Creative Technology bachelor graduation project by Tom Onderwater and Yoann M.E. Latzer

### Goal of the project

The project's goal is a feasibility study to supply a data visualization (e.g.: map) showing the different temperatures around the city of Enschede. This visualization will have some points showing real temperatures and may provide a map with calculated temperatures based on a pre existing RIVM model. The final goal is to evaluate the feasibility of a model that shows the temperatures in different areas as **accurately as possible**. The final use case scenario of the model will be left to future projects and the municipality of Enschede.

### Term of the project

The current project will be developed between September 2017 and February 2018. The data measurements will start latest in January 2018.

## Planning of the project

Sensor Development	Visualization Development	Milestones	Time (in week number)
Planning of project	Planning of project		46
Design first prototype sensor	Design back and front end		47
Building first prototype sensor	Build back end prototype	Tom: First working Sensor node on UT campus Yoann: First working back end (server) prototype	48
Multiple version 1 sensor nodes developed (at least 3)	Version 1 back end with real data from sensor nodes developed	<b>Tom &amp; Yoann:</b> Working network with (minimum) 3 nodes	49
Developing and designing version 2 (mass producible version with solar panel)	Design prediction model	<b>Yoann:</b> Front end prediction map prototype	50

Building version 2	Implement prediction map on front end interface	<b>Tom:</b> Deployment of sensor nodes throughout city	51
Brainstorm on possible applications Get visualization requirements from the municipality		Presentation for municipality discussing use cases	
Holiday			52
Holiday			1
Development of municipality interface			2
Evaluation of the municipality interface			3
Graduation project report			4
		<b>Tom &amp; Yoann:</b> Graduation project presentation (Enschede is invited)	5
		Tom & Yoann: A0 project poster	6

### Cost of the project

For the feasibility phase, most costs will be covered by the University of Twente.

# Suggestions for use of the sensors

#### Datavisualization

From the data generated by our sensor network, different visualizations are proposed. The main one consist of a map which displays the current temperature levels, the second one is a forecast of the entire city temperature levels: Finally additional visualisations will be developed depending on the final use (bar charts, heat calendar, ...).



Main visualization: display measured temperature levels



Second visualization: display the projected temperature levels

As we can see on the two visualizations presented above, the second one provides information about the entire city with the same amount of sensors. It can be done using data from RIVM for example (see the heat map available in the "Location of the Sensors section"), which would be combined with our data in order to have an accurate projection.



Visualization interface: display an overview of the network on a webpage

In the end the system is delivered with a visualization interface which is based on the data produced by our sensor network. There are different way of using it, which can be split into two categories: internal use for Enschede's administration and external use for the citizens.

#### Internal use

- Web interface: visualize the current temperature of the city and check the status of each sensor (on/off, last measurements, ...). This interface also allows to look at the temperature levels over time. Graphs and raw data can be exported (for example to pass on to another service)
- Temperature level alerts: a certain service of the municipality receives an alert message when a certain level of temperature is reached

#### **External use**

- Web interface: same as the internal one, possibly with less functionalities
- Mobile application: a mobile version of the web interface
- Prevention campaign: It is possible to generate a poster or street marketing campaign from the data gathered (to give direction for hot locations to avoid

### Locations of the Sensors



- 1. Volkspark (park) w51
- 2. South east (downtown) w51
- 3. van Heek Plein (commercial) w51
- 4. Campus (Rural) w51
- 5. Enschede South (suburban) w51
- 6. Twekkelerveld (commercial) w51

Source: RIVM

Sensors will have to be spread out across the city to get a broad overview over the different temperature zones. This is necessary to create a temperature map that is as accurately as possible. The current list of proposed locations is still susceptible to small changes if the situation requires a different sensor position. Help of the municipality is needed to get precise locations and permission to place the sensors.

### Conclusion

The University will provide a feasibility test of a temperature measurement system for the City of Enschede, real world uses will be discussed after the feasibility tests have been performed in early 2018.

### Appendix D: Design cycles





# Appendix E: Assembly guide:

Components needed:	Amount
<i>3D printed:</i>	
Top_disc	1x
Disc	2x
Bottom_disc	1x
Housing	1x
Сар	1x
Inside	1x
Bottom_panel	1x
Top_panel	1x
Screws:	
M5x25	1x
M4x10	2x
M2.5x8 (wood screws)	4x
Nuts:	
M5	1x
M4	2x
Electronics:	
Sodaq One	1x
DS18B20	1x
TP4056	1x
8K resistor	1x
4.7K resistor	1x
LiPo battery 1200mAh	1x
GPS antenna	1x
Solar Panel 1W	1x
Female jumper cables	бх

### **Tools needed**

Phillips screwdriver	1x
Soldering iron	1x
Solder	1x
Super glue	1x
Hot glue	1x
Electrical tape	1x
Stanley knife	1x
Contact glue	

### Setup:

- 1. Begin with removing the brim from the 3D printed parts.
- 2. Cut the DS18B20 temperature sensor so that the cable attached is 10 cm long.
- 3. Keep the cut cable, this will be needed later.
- 4. Strip the individual cables from the DS18B20 so that the copper is exposed
- 5. Insert the DS18B20 into the bottom disc part like shown:



6. Solder a 4.7K resistor between the data cable and the VCC cable coming out of the DS18B20 like shown:



- 7. Cut 3 female headers and solder these to the open connections.
- 8. Make sure the temperature sensor is waterproof and secured by surrounding it with hot glue:



- 9. Attach the GPS sensor and the LoRa antenna to the Sodaq One by attaching them to the labeled antenna connections.
- 10. Glue the Sodaq One onto the inside (3d printed part), and the GPS sensor too, they should fit in the holes in only one way, make sure the Sodaq's USB connection faces downwards.





11. Slide the inside into the housing, use the rail closest to the back (away from the logo)

12. Attach the female pins from the DS18B20 to pin D11 (data), 3.3V out (VCC) and GND (GND) respectively on the Sodaq One.



13. Attach the housing to the bottom disc, **cover the entire top of the housing with superglue** for a water tight seal (IMPORTANT!!!):



14. Attach the rest of the discs to the bottom disc like shown it is possible that some sanding or drilling is needed to widen the holes:



- 15. It is possible to glue the discs onto each other, do so if necessary.
- 16. Get the solar panel and remove the plastic covering it.
- 17. Place the solar panel in the indent in the top solar panel.





18. Glue the solar panel in place with a couple of dabs of hot glue



19. Glue the TP4056 in place



20. Solder the red and black connection from the solar panel to the positive and negative terminals on the TP4056.

21. Add a lot of hot glue to the solar panel and wait for it to cool of until it becomes solid (wait 5 minutes)



22. Add a little bit of new hot glue and attach the battery on it. *Explanation: LiPo batteries* don't handle heat well, we add the hot glue as an insulating layer, to remove it from the heat from the solar panel in the summer. We can't glue it directly to the solar panel with hot glue however, this would heat the LiPo battery too much. This is why we only use 2 small dabs, the LiPo battery is capable of handling this little amount of heat.



23. Wait for it to cool a little bit and attach the battery and solder it to the B+ and B- pin on the TP4056.



24. Take the leftover piece of cable from the DS18B20 sensor and place it in the hole in the solar panel and solder the red and black cable to the output of the TP4056.



25. Glue the cable in place so it does not tear the TP4056 from the solar panel.



26. Find R3 and remove it from the TP4056



27. Solder the 8K resistor between the two exposed pieces of metal. It is good practice to use something which can hold the resistor firmly in place.


- 28. Charge the battery, using the USB port
- 29. Close the solar panel housing

73



30. Add M2.5 wood screws to secure it in place:



- 31. Insert hot glue in the holes above the screws so that water can't enter.
- 32. Fill the gaps around the cable coming outside the panel to ensure that no water can enter.
- 33. Attach the solar panel to the main housing using the M5 bolt and nut



34. Pull the cable into the housing and solder 2 female connectors to the Sodaq one's 3.7V li-Ion + pin and the GND pin.



- 35. Fill the gaps between the cable and the hole in the housing.
- 36. Upload the Sodaq\_One\_DEV\_EUI software and get the Dev EUI from the sodaq one from the serial monitor.
- 37. Fill in the DEV EUI in the DEV EUI field in Ensketemp.ino
- 38. Upload the code to the Sodaq One.
- 39. Add a jumper between the external switch pin and the secondary 3.7V Li-Ion battery+ (5th pin from up) *This is needed for the Sodaq One to turn on.*



40. Push the M4 Screws in the holes below the housing. *You can use a soldering iron to push them in, heating the screws will momentarily melt the plastic and allow the screws to go in smoothly.* 



- 41. Screw the cap onto the housing
- 42. Glue the delrin cover on the solar panel. Use contact glue to make it waterproof.

43. Attach the strap to the sensor node and place it outside.









## **Appendix F: Source Code**

#include "Arduino.h"
#include <Sodaq\_RN2483.h>
#include <RTCZero.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Sodaq\_UBlox\_GPS.h>

//Real time clock library RTCZero zerortc;

//temperature sensor is attached to pin 11 OneWire oneWire(11); DallasTemperature sensors (&oneWire);

//LoRa uses Serial1 #define loraSerial Serial1

//Define the maximum number of tries the GPS module can take to get a fix: #define MAXTRIES 5 #define DEBUG false

//change these to fit your LoRa module

uint8\_t AppEUI[8] = { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x00, 0x85, 0x5B }; //node 1 //uint8\_t DevEUI[8] = REDACTED //uint8\_t AppKey[16] = REDACTED

//node 2 //uint8\_t DevEUI[8] = REDACTED //uint8\_t AppKey[16] = REDACTED

//node 3
uint8\_t DevEUI[8] = REDACTED
uint8\_t AppKey[16] = REDACTED

//node 4
//uint8\_t DevEUI[8] = REDACTED
//uint8\_t AppKey[16] = REDACTED

//node 5
//uint8\_t DevEUI[8] = REDACTED
//uint8\_t AppKey[16] = REDACTED

//NODE 6 //uint8\_t DevEUI[8] = REDACTED //uint8\_t AppKey[16] = REDACTED

//No defines used, could be settable:

#define TEMPOFFSET 0.0f

byte sleepHours, sleepMinutes, sleepSeconds;

```
void setup() {
 //define how +long the LoRa module should normally sleep between packets
 sleepHours = 0;
 sleepMinutes = 5;
 sleepSeconds = 0;
 // inspired by...
 // http://forum.sodaq.com/t/sodaq-one-power-consumption/346/7
 pinMode(ENABLE PIN IO, OUTPUT);
 digitalWrite(ENABLE_PIN_IO, HIGH);
 pinMode(GPS ENABLE, OUTPUT);
 digitalWrite(GPS_ENABLE, LOW);
 //set a default GPS location
 setGPS();
 // RGB LED:
 initalizeLEDs();
 lightBlueLED();
 //setSensors();
 delay(10000);
 setupLoRa();
 zerortc.begin(); // Set up clocks and such
 resetAlarm(); // Set alarm
 zerortc.attachInterrupt(alarmMatch); // Set up a handler for the alarm
 delay(100);
 lightNoLED();
}
void loop() {
 if (GPSAvailable()) {
  //try to get a fix
  tryGPS();
 }
 String packet = "";
// startSensors();
 sensors.begin();
 packet += "t";
 packet += tempToString();
 packet += getGPS();
 sendPacket(packet);
 incrementGPSCounter(); //add one to the GPS (will reroll after 255 and update the GPS location)
 //Do nothing after this:
// endSensors(); //unplug the temperature Sensor
 resetAlarm(); //reset a alarm with your defined time
 initSleep(); //initialize the neccesary sleep settings
 //if (DEBUG) {
  lightNoLED(); //turn off all LED's (if not done yet)
 //}
 //Call deep sleep
 _WFI(); //Wait for Interrupt of the RTC timer which keeps running
```

```
delay(200);
}
byte counter = 0;
boolean firstStartUp = true;
//set maximum movement distance in KM:
float MAXDIST = 1 / 111.0;
#define MINIMUMSATELITES 6
void incrementGPSCounter() {
 counter ++;
}
void decrementGPSCounter(const byte amount) {
 counter -= amount;
}
byte returnGPSCounter() {
 return counter;
}
boolean GPSAvailable() {
return (returnGPSCounter() == 0);
}
String lat, lon;
String lastLat, lastLon;
byte tries = 0;
byte numSatellites = 0;
String getGPS() {
 String out = "L" + lat + "l" + lon;
 if (DEBUG) {
  out += "S";
  out += String(numSatellites);
 }
 return out;
}
void setGPS() {
lat = "52.2396"; //fill in with the sensor location found
 lon = "6.8570"; //fill in with the sensor location found
}
void tryGPS() {
 //turn on the GPS
 digitalWrite(GPS_ENABLE, HIGH);
 sodaq_gps.init(GPS_ENABLE);
 uint32 t timeout = 90L * 1000; //normal 90 second timeout
 boolean succesFullGPS;
 if (sodaq gps.scan(false, timeout)) {
  numSatellites = byte(sodaq_gps.getNumberOfSatellites());
  if (numSatellites >= MINIMUMSATELITES) {
   lat = String(sodaq_gps.getLat(), 4);
```

```
lon = String(sodaq_gps.getLon(), 4);
   //lat = tempLat;
   //lon = tempLon;
   succesFullGPS = true;
   tries = 0;
   if (DEBUG) {
    lightBlueLED();
   }
  } else {
   if (tries < MAXTRIES) {
    decrementGPSCounter(1);
    tries++;
   } else {
    //stop trying and wait for the next cycle to try again
    tries = 0;
   }
   if (DEBUG) {
    lightRedLED();
   }
   succesFullGPS = false;
  }
 } else {
  if (DEBUG) {
  lightRedLED();
  }
  succesFullGPS = false;
  if (tries < MAXTRIES) {
   //try again
   decrementGPSCounter(1);
   tries ++;
  } else {
   tries = 0;
  }
 }
 //turn off the GPS
 digitalWrite(GPS_ENABLE, LOW);
 if (DEBUG) {
  delay(200);
  lightNoLED();
}
}
float dist(float first, float second) {
 return abs(first - second);
}
void lightRedLED() {
digitalWrite(LED_RED, LOW);
digitalWrite(LED_GREEN, HIGH);
digitalWrite(LED_BLUE, HIGH);
}
void lightGreenLED() {
digitalWrite(LED RED, HIGH);
digitalWrite(LED_GREEN, LOW);
digitalWrite(LED_BLUE, HIGH);
}
```

```
81
```

```
void lightBlueLED() {
digitalWrite(LED_RED, HIGH);
digitalWrite(LED_GREEN, HIGH);
digitalWrite(LED_BLUE, LOW);
}
void lightNoLED() {
digitalWrite(LED_RED, HIGH);
digitalWrite(LED_GREEN, HIGH);
digitalWrite(LED_BLUE, HIGH);
}
void initalizeLEDs()
{
pinMode(LED_RED, OUTPUT);
pinMode(LED_GREEN, OUTPUT);
pinMode(LED_BLUE, OUTPUT);
}
void setupLoRaOTAA() {
 if (LoRaBee.initOTA(loraSerial, DevEUI, AppEUI, AppKey, false))
 {
  // SerialUSB.println("Communication to LoRaBEE successful.");
  if (DEBUG) {
   lightGreenLED();
  }
 }
 else
 {
  if (DEBUG) {
   lightRedLED();
  }
  // SerialUSB.println("OTAA Setup failed!");
 }
}
void setupLoRa() {
 loraSerial.begin(LoRaBee.getDefaultBaudRate());
 setupLoRaOTAA();
 LoRaBee.setSpreadingFactor(9);
 delay(500);
 if (DEBUG) {
  lightNoLED();
}
}
void sendPacket(String packet) {
 switch (LoRaBee.send(1, (uint8_t*)packet.c_str(), packet.length()))
 {
  case NoError:
   // debugSerial.println("Successful transmission.");
   if (DEBUG) {
    lightGreenLED();
   }
   break;
  case NoResponse:
   // debugSerial.println("There was no response from the device.");
```

```
if (DEBUG) {
    lightRedLED();
   }
   setupLoRa();
   break;
  case Timeout:
   // debugSerial.println("Connection timed-out. Check your serial connection to the device! Sleeping for 20sec.");
   delayMicroseconds(2000000);
   if (DEBUG) {
    lightRedLED();
   }
   break;
  case PayloadSizeError:
   // debugSerial.println("The size of the payload is greater than allowed. Transmission failed!");
   break;
  case InternalError:
   if (DEBUG) {
    lightRedLED();
   }
   // SerialUSB.println("Oh No! This shouldn't happen. Something is really wrong! Try restarting the device!\r\nThe
network connection will reset.");
   setupLoRa();
   break;
  case Busy:
   if (DEBUG) {
    lightBlueLED();
    delay(200);
    lightNoLED();
    delay(200);
    lightBlueLED();
   }
       debugSerial.println("The device is busy. Sleeping for 10 extra seconds.");
   11
   delayMicroseconds(1000000);
   break;
  case NetworkFatalError:
   // debugSerial.println("There is a non-recoverable error with the network connection. You should re-connect.\r\nThe
network connection will reset.");
   setupLoRa();
   break;
  case NotConnected:
   if (DEBUG) {
    lightRedLED();
   }
   // debugSerial.println("The device is not connected to the network. Please connect to the network before attempting
to send data.\r\nThe network connection will reset.");
   setupLoRa();
   break;
  case NoAcknowledgment:
   if (DEBUG) {
   lightRedLED();
   }
   // debugSerial.println("There was no acknowledgment sent back!");
   // When you this message you are probaly out of range of the network.
   break;
  default:
   break;
}
```

```
83
```

```
}
void initSleep() {
// tell the LoRa-Module to sleep!
delay(200);
// Set the sleep mode
SCB->SCR |= SCB_SCR_SLEEPDEEP_Msk;
}
// do nothing just wake up
void alarmMatch(void){}
void resetAlarm(void) {
// hours, minutes, seconds
zerortc.setTime(0, 0, 0);
// day, month, year
zerortc.setDate(1, 1, 1);
// alarmHours, alarmMinutes, alarmSeconds
zerortc.setAlarmTime(sleepHours, sleepMinutes, sleepSeconds);
zerortc.enableAlarm(zerortc.MATCH_HHMMSS);
}
String tempToString() {
//Temperature Code
//sensors.begin();
sensors.requestTemperatures();
float temp = avgTemp();//sensors.getTempCByIndex(0);
temp += TEMPOFFSET;
//sensors.end();
 return (String) temp;
}
void setSensors() {
 pinMode(10, OUTPUT);
 pinMode(9, OUTPUT);
 digitalWrite(10, LOW);
 digitalWrite(9, LOW);
}
void startSensors() {
 digitalWrite(10, HIGH);
}
void endSensors() {
 digitalWrite(10, LOW);
}
float avgTemp() {
 float samples = 10;
 float sum = 0;
 for (int i = 0; i < samples; i++) {
  sum += sensors.getTempCByIndex(0);
  delay(10);
 }
 return sum /= samples;
}
```