FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Low Energy Communication Network For Control And Supervision Of Hydroponic Farms

João Raúl Rodrigues da Silva Lima

WORKING VERSION

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Mário de Sousa

Co-Supervisor: Carlos Resende

July 5, 2015

Resumo

Estima-se que até 2050 a necessidade mundial de cereal cresça 51% em comparação ao valor registado em 2000. Este crescimento pode não ser cumprido devido ao facto da área de cultivo da agricultura tradicional estar continuamente a diminuir por causa da ocupação humana ou pela erosão dos solos devido ao uso de pesticidas e exploração de recursos naturais. Desta forma, uma possível solução para garantir esta previsão é o uso da agricultura hidropônica. No entanto, este tipo de agricultura tem um custo inicial muito elevado o que, muitas das vezes, torna difícil a instalação de um sistema hidropônico.

Com base neste problema, a Fraunhofer AICOS apresenta como ideia um sistema hidropônico, constituído por sensores que comunicam entre si, a baixo custo e de fácil instalação. Surge assim o foco desta tese, propor e desenvolver uma rede de sensores. Nesta proposta é apresentada a arquitetura e a comunicação da rede tal como o hardware, o firmware e o consumo energético de cada elemento da rede.

Abstract

It is estimated that by 2050 the world cereal need will grow 51% compared to the value recorded in 2000. This growth might not be met due to the fact that the cultivation area of the traditional agriculture is continuously decreasing because of human occupation or by soil erosion due to the use of pesticides and exploitation of natural resources. This way, a possible solution to ensure this forecast is the use of hydroponic agriculture. However, this kind of agriculture has a very high initial cost, that often makes it difficult to install a hydroponic system.

Based on this problem, the Fraunhofer AICOS presents the idea of a hydroponic system, constituted by sensors that communicate with each other, at a low cost and of easy installation. Thus arises the focus of this thesis, propose and develop a network of sensors. In this proposal is presented the architecture and the communication of the network such as hardware, firmware and energy consumption of each element of the network.

Agradecimentos

Esta dissertação simboliza a conclusão de cincos anos de esforço, aprendizagem e ambição, como também o início de uma nova etapa. Ao longo destes cinco anos, de uma forma ou de outra muitos contribuíram para minha formação académica e para o meu crescimento pessoal. Uma das pessoas que mais contribuiu foi a Diana Ribas.
A ti quero agradecer todo o apoio, paciência, amizade, amor e motivação que sempre me deste nos momentos difíceis desta tese.
Em seguida quero agradecer aos meus amigos, que conheci na FEUP, em especial ao André Oliveira, Duarte Silva e Pedro Medeiros. Agradecer-vos por todo apoio e amizade que me deram durante a nossa passagem pela FEUP.
Por último quero agradecer ao meu orientador, Engenheiro Mário de Sousa, pela grande ajuda e disponibilidade concedida e também a toda a equipa da Fraunhofer, em particular ao meu co-orientador, Engenheiro Carlos Resende, pelo apoio que foi prestado e pelas condições proporcionadas.

João Raúl Rodrigues da Silva Lima

”Only passions, great passions, can elevate the soul to great things”

Denis Diderot

Contents

List of Figures

List of Tables

Acronyms and Symbols

AES	Advanced Encryption Standard
AFH	Adaptive Frequency Hopping
API	Application Programming Interface
ATT	Attribute Protocol
BLE	Bluetooth Low Energy
CRC	Cyclic Redundancy Check
CSV	Comma-Separated-Values
FAO	Food and Agriculture Organization of the United Nation
FHSS	Frequency-hopping spread spectrum
GAP	General Access Profile
GATT	Generic Attribute Profile
GPRS	General packet radio service
L2CAP	Logical Link Control and Adaptation Protocol
LEACH	Low-Energy Adaptive Clustering Hierarchy
NTF	Nutrient Film Technique
OCR	Oxidation Reduction Potential
SoC	System on a chip
SPI	Serial Peripheral Interface
SPIN	Sensor protocols for information via negotiation
PDU	Protocol Data Unit
P2P	Point-to-Point
TTL	Time To Live
UART	Universal Asynchronous Receiver/Transmiter
USB	Universal Serial Bus
UUID	Universal Unique Identifier
VISA	Virtual Instrument Software Architecture
WiFi	Wireless Fidelity

Chapter 1

Introdution

1.1 Motivation

Traditional agriculture is an ancient practice that man explores for self-consumption and subsistence. Until recently this practice was enough to meet human needs, but with the increase of these needs and with the increase in population and in the purchasing power of emerging people [5] there is overexploitation of this ancestral activity.

According to FAO [5] report, it is estimated that by 2050 the world cereal need will be 2738 million tons, an increase of approximately 51% when compared to the data recorded in 2000 - 1818 million tonnes. The FAO forecast also indicates that the increase in food demand occurs not because the world population is increasing, but because undernourished countries are finally gaining purchasing power and development capacity [5].

However, with the decrease in cultivation area due to human occupation [5] or soil erosion(a result of the use of pesticides and fertilizers by intensive agriculture, water shortage, increased demand for food and biofuel production [5, 6]) traditional agriculture may not be able to answer to the 2050 forecast.

This is particularly true in undernourished countries, as it is the case of the African continent countries, where, according to the FAO organization reports [5, 7] , soil erosion is an increasing problem.

An example of these countries is South Africa, where according to the UN Environment Programme report, 90% of the country is considered arid, semi-arid or sub-humid [8]. A country with this configuration and with a forecast of population growth and rising cost of cereals [5], can lead to a serious problem of malnutrition.

A solution being used to address these, South Africa and worldwide, challenges and meet the future food demands is the hydroponic agriculture.

Hydroponic consists of an agriculture without soil, i.e., the plants are arranged in channelized systems, whose roots are in contact with a solution containing the nutrients necessary for their growth [9].

When compared to traditional agriculture, hydroponic farming is a closed and precise system which allows to prevent water waste, as well as to precisely monitor and control the consumption of nutrients. Additionally, due to the precise control of all the plant environment and its food supplies, hydroponic systems allow to increase the growth rate of the plant, to cultivate it out of the planting season, and to do it without wasting nutrients and water.

Another advantage is that, since the cultivation is done without soil, vertical plantations are possible, i.e., several layers of hydroponic systems can be stack up in the same area. This results that for the same plantation area, hydroponics can produce in larger quantities when compared to conventional farming [10].

With these advantages hydroponic farming is increasingly becoming a reality, which is helping not only to meet the world food demand, but also helping to solve the farming problems of many undernourished countries, as it is the case of South Africa.

Due to its precision characteristic, and in order to take the most of the hydroponic farming, hydroponic systems demand a good monitor and control system. This need arises given the vulnerability of the plants roots, because, being directly exposed to water and nutrients, one minimal change on some of its variables, for example the potential of Hydrogen (pH) of the water, can have a huge impact in all the hydroponic production.

There are already industrial monitor and control hydroponic systems on the market, but they are not developed having in mind undernourished countries, so their advantages are not fitted to these countries needs and are too expensive for their economic reality.

The objective of this thesis is to study the needs of South African hydroponic farm in order to carry out the survey of the requirements necessary to develop a new low-cost, of easy installation and robust hydroponic monitoring system.

1.2 Description

This thesis is incorporated on the HYDROSNET4D project which is integrated on the Fraunhofer ICT4DCC project. The primary goal HYDROSNET4D project is to develop a low cost mechanism for mobile monitoring of hydroponic farms. This includes the development of WSNs, as well as a mobile application to be easily and effectively used by farmers to reduce the time involvement required to monitor a hydroponic culture. The envisaged solution will allow real time monitoring of environmental factors (ambient temperature, ambient humidity, hydroponics' bags' water level, pH, lighting, etc.), as well as provide graphical data, crop statistics and equipment fault warnings. This will contribute for the development made under this Competence Center in what regards hydroponic farm for development countries.

In order to address the problem presented in section 1.1 a set of requirements were already studied and gathered for a hydroponic South African farm, the Olive Tree Farm.

The Olive Tree Farm produces peppers, spring onions and tomatoes with the use of hydroponic agriculture in Port Elizabeth, Eastern Cape, South Africa. It has an area of 6000 m^2, a total of

44 greenhouses, 11 water reservoirs already with an installed hydroponic industrial system for monitoring and controlling part of the plantation.

The choice of this farm is due to a national competition to support the development of infrastructures, in this case the update of the hydroponic system. Its study enabled the characterization and understanding of the functioning of a hydroponic plantation.

Such characterization resulted in the following list of requirements for the new hydroponic system:

- Low power and battery powered: All the elements of the system should be developed with low power considerations in mind and should be supported by a battery in order to account with regular energy cut outs.

- Easy to install, maintain and use: The system main targets are low funded hydroponic farms situated on the rural parts of South Africa with no technical persons on its staff, and where the recruitment of technical teams to maintain and install the system will introduce unwanted costs. Additionally, the farms are always changing their configuration (what is being planted and where) in accordance to the times of the year, so the system needs to be easily modified (uninstall and (re)install) and configured for the harvest of each season.

- Low bandwidth Wireless Fidelity(WiFi)/General packet radio service(GPRS) interface to the external world: The system main targets are low funded hydroponic farms situated on the rural parts of South Africa where GPRS access is not guaranteed and were the WiFi/GPRS signal strength could be weak or null.

- Maximum communication distance of 54 meters: It is necessary to ensure a wireless communication distance of at least 54 meters. This value is obtained from the studies performed in Olive Tree Farm, the biggest greenhouse has 54 meters.

- Minimum of two measurement per zone: The goal is to record the absorption index of each line of nutrients in order to control and optimize the costs of the necessary nutrients added to the solution.

Based on the analysis of the requirements and problems described in section 1.1 and listed above we concluded that the ideal solution should use computing technologies and low energy consumption communication powered by a battery in a wireless distributed architecture. This way it can ensure the requirement of easy installation and system capacity.

In order to perform scalability, using a Gossip communication protocol is an added value, since in addition to ensuring the scalability, this ensures fault tolerance in the system.

A detailed description of this solution is described in chapter 3.

1.3 Objectives

The foundation of this thesis can be organized on its theoretical and the practical objectives.

The theoretical objectives are related with state of the art studies, namely: (1) the study of hydroponic agriculture and their monitoring systems; (2) the study of the requirements of South African hydroponic farms; (3) the study of low power wireless networks and their communication protocols, namely the Epidemic protocols.

The practical objectives are the result of the theoretical analysis, and will culminate in the development of a monitoring solution for hydroponic farms, suitable to be applied on the South African context. Such solution will be based on a low power wireless sensor network, on top of which will be running a epidemic protocol. Additionally, it would be added a gateway to interface the hydroponic monitoring network with the outside world (the Internet).

1.4 Document Structure

The first chapter of this report consists of an introduction to the project that will be developed under the scope of the thesis, where the motivation, description and objectives of the project are described.

The second chapter contains the state of the art on all the topics covered by this thesis: hydroponic farming; Wireless Communication Protocols; and Epidemic Communication Protocols.

The third chapter presents the system specification, which describes the vision of the system to implement based on the technologies and protocols studied in chapter 2.

In the fourth chapter is presented the hardware used under the scope of this thesis, then is explained the firmware developed and applied in the sensing and gathering devices and, finally, the structure of the Software developed to install on the Cloud device.

In the fifth chapter is explained the whole process of measuring the energy consumption of the sensing and gathering devices, as well as, the dimensioning of the batteries to integrate these devices.

Finally, in the sixth chapter, the theme addressed is the conclusion and the enumeration of the fulfilled objectives, followed by a description of the future work.

Chapter 2

State of the Art and Related Work

Firstly, this chapter begins with a brief introduction about what is hydroponics, followed by a description of the techniques most commonly used in this type of culture, and the physical quantities to be measured that most influence in this type of planting. Based on the latter, are presented systems and sensors already on the market.

In the second part, are presented different of wireless low energy technologies. They are compares in various aspects, such as: (1) the type of topology supported by each protocol; (2) their costs; (3) their efficiency; (4) their energy efficiency; (5) their performance; (6) their robustness; (7) their data latency; (8) and their peak consumption.

Finally, it is made an introduction to the concept of epidemic communications where are presented different techniques that could be integrate into a wireless sensor network.

2.1 Introduction to Hydroponic Agriculture

Hydroponics consist of a soilless agriculture in which plants grow in a mineral nutrient solution, as opposed to traditional agriculture, where plants take the nutrients needed from the soil through their roots; a process that requires energy, time and is dependent on the type of soil [9]. On the other hand, in hydroponics, the nutrients are directly provided to the roots of the plants through a mineral nutrient solution. Feeding of the roots can be done by submersion in a solution or by spraying, this way we can distinguish two types of hydroponic cultures: (1) ones where the roots are supported in an inert substrate; (2) and others where the roots are suspended and the feed of the roots is done through its pulverization. These two types of hydroponic cultures can be combined with different irrigation types and growing mediums to create the different hydroponic systems presented in subsection 2.1.1.

As stated above, hydroponic cultures of type (1) need a growing medium that is usually an inert environment known as substrate. There are many different kinds of substrates, each one having its own advantages and disadvantages and appropriate for different growing techniques.

Additionally, these different techniques can have closed or open water circuits, meaning that they can recycle the water used to feed the plants by introducing it again in the hydroponic system (closed circuit), or can simply throw it away when the plants are fed (open circuit) [10]. The closed water circuit has the advantage of being more efficient in the quantity of water and nutrients used, but, since it is closed, it is more vulnerable to the speed at which a disease is spread over the plantation, so it needs constant attention. When an ill plant is spotted it needs to be immediately removed from the system and the tank and the circulation system needs to be clean in order to contain the infection and do not contaminate the other plants.

2.1.1 Types of Hydroponic Growing Systems

There are six main types of hydroponic growing systems: aeroponic, drip, ebb and flow, nutrient film technique (NFT), deep water culture, and wick. Hundreds of variations of these systems can be found, but all the hydroponic techniques are a variation or a combination of those six.

All of the growing systems are carefully thought in order to fulfill all the basic needs of the plants, namely: (1) the amount of nutrients dissolved in the water; (2) the pH level of the water, which influences the capability of the roots to absorb the nutrients in the water; (3) and the amount of air available to the plant roots, which prevents them to get rot (in order to keep a good ventilation of the roots some systems use air stones combined with air pumps, while others use porous growth media).

2.1.1.1 Aeroponics

Aeroponics, represented in 2.1, is a substrateless system in which the roots are suspended in the air on top of a container (reservoir), filled with a mist of the solution containing the nutrients necessary for plant growth. This system periodically provides to the roots a thin nutrient layer by

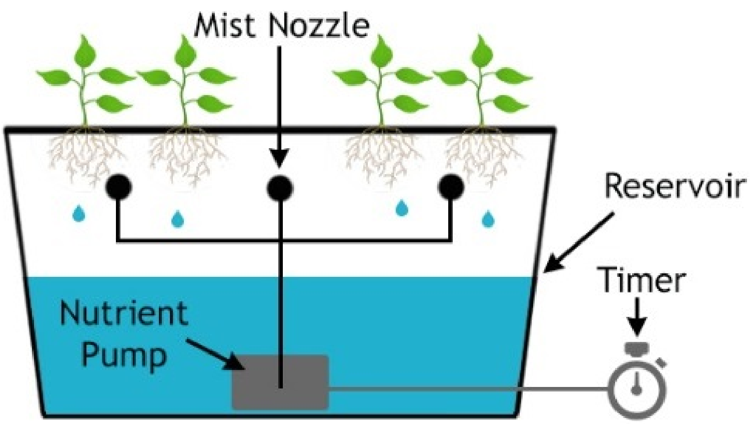

Figure 2.1: Scheme of Aeroponics [1]

activating its pump and pulverizing the roots with the mist of nutrient solution contained in the reservoir. The excess of this irrigation process will fall in the reservoir, so there is no waste of nutrient solution.

Aeroponics advantages are the humidity environment due to the pulverization, the efficient use of water and nutrients, the absence of substrate, and encourages the rapid growth of the plant.

2.1.1.2 Deep water culture

Deep Water Culture (2.2) is a form of hydroponics were the plant roots are permanently immersed in the nutrient solution. This causes a root breathing problem, which is solved by pumping air into the water and guaranteeing that it contains enough oxigen to allow the plant roots to breath. This technique has the advantage of being simple and economic in its implementation.

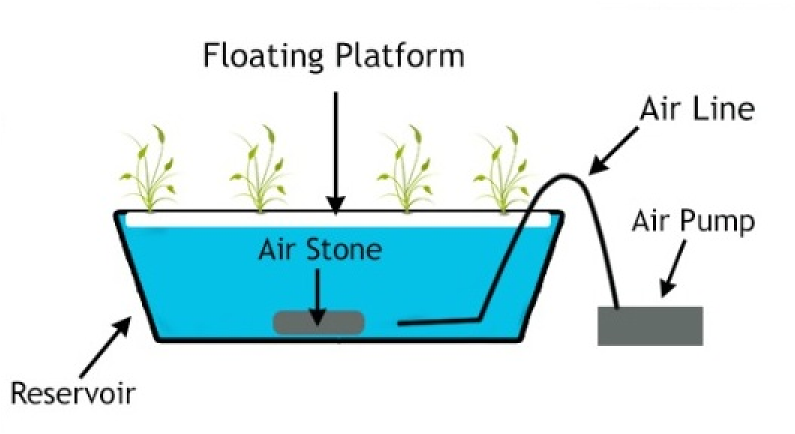

Figure 2.2: Scheme of Deep water culture [1]

2.1.1.3 Drip

In Drip hydroponic system the plants are grown in a tray with substrate and the nutrient solution is prepared in a separate reservoir, as Figure 2.3 shows. This solution is then used to periodically and drop-wise feed the plants. This method allows a precise and economic system on the need for system water. Where the excess water can be collected and reused. Precise because the roots are watered directly dropwise, allowing at the same time their respiration, and economic because there is less evaporation and the nutrients are reused.

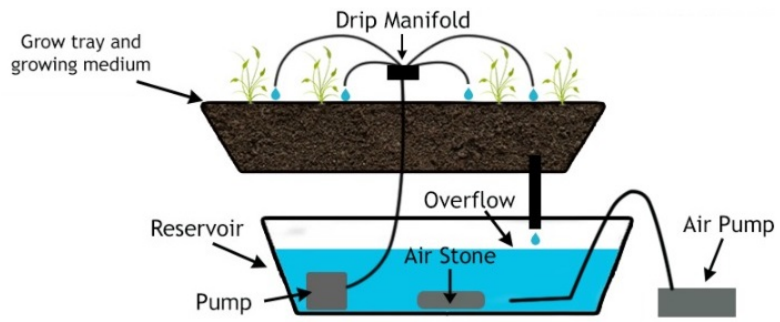

Figure 2.3: Scheme of Drip System [1]

2.1.1.4 Ebb and flow

In Ebb and Flow system, the system structure is very similar to Drip, there is a platform where the plants and the roots are placed and in contact with a substrate. But instead of a dropwise irrigation system is a flood system. That is, the platform where the plants are located is flooded with the solution containing nutrients during a certain time interval. At the end of this time the solution is removed, the systems often use gravity to perform this step. This technique is not very efficient in comparison with other systems in the use of plant nutrients.

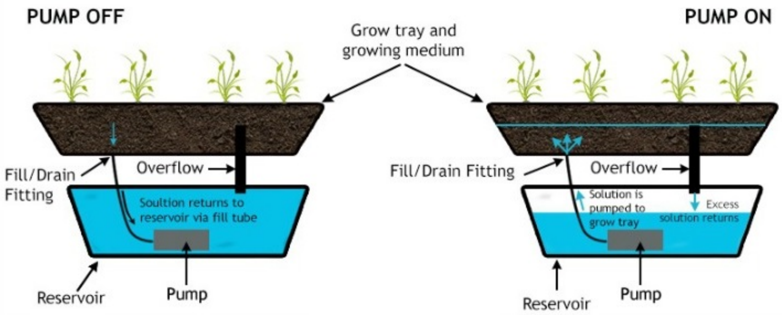

Figure 2.4: Scheme of Ebb and Flow [1]

2.1.1.5 Nutrient Film Technique

NFT technique consists in the direct supply to the plant roots of a constant flow of solution containing nutrients, but not fully flooding the roots so as to allow oxygenation. To better flow and distribution of the solution, the plant-supporting platform has a slight slope. This method is suitable for plants that do not need a lot of support, which are lightweight, fast-growing and need to be taken quickly.

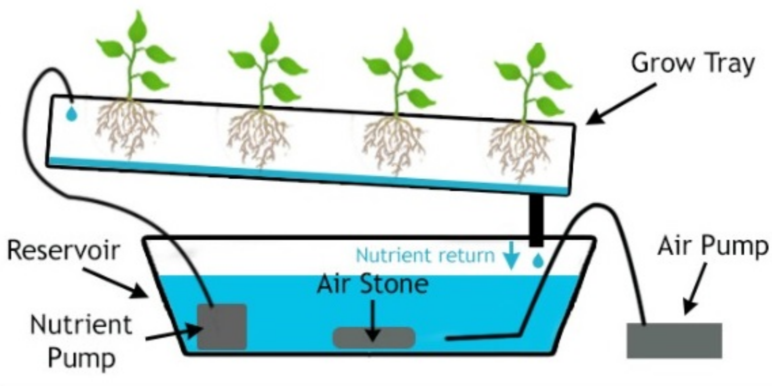

Figure 2.5: Scheme of NTF [1]

2.1.1.6 Wick system

Wick system consists of two platforms, one supporting the plants and another that is the water tank, which are interconnected by one or more pipes with an absorbent material on the inside. Allowing a permanent hydration of the substrate surrounding the roots. This type of system is the most simple and low maintenance but is inefficient compared to the previously described solutions.

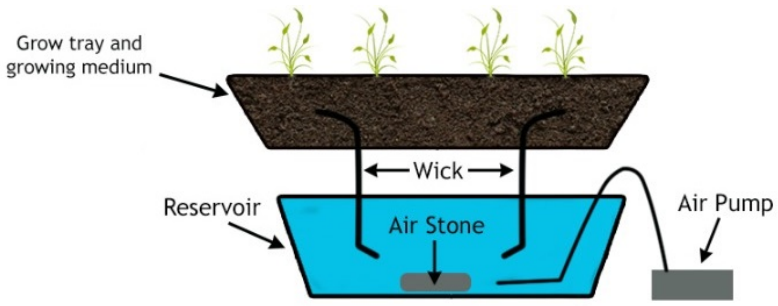

Figure 2.6: Scheme of Wick System [1]

2.1.2 Growing an Hydroponic Farm

In hydroponics, unlike what happens in traditional crop, there isn't a drainage system as exists in the soil to protect the roots of the plants. As the roots of the plant are in direct contact with the solution, it is necessary to control the solution parameters as pH, electrical conductivity (eC) and temperature. These parameters are the solution characteristics that will enable plants to absorb nutrients. The eC is directly proportional to the amount of nutrients dissolved in water, the higher the eC, more nutrients exist in the water and as each plant needs a certain quantity of nutrients is essential to maintain this value controlled, otherwise the lack of nutrients will prevent plant growth and the excess will prevent absorption of water which will lead to dehydration of the plant. The pH level allows the absorption of nutrients being necessary to ensure optimal range, otherwise the plant will not be able to assimilate the nutrients it needs. Temperature influences the efficiency of photosynthesis. At high temperatures, plant growth is higher because there is more oxygen and photosynthesis, whereas for low temperatures there is a slowdown in growth.

There are other measurements which can be made of the irrigation system: the pressure in the tubes, the water level in the reservoirs and the volume of water absorbed for better management of the system.

2.1.2.1 Quantities

The three previously mentioned parameters can change a lot regarding the type of plant that is being considered, for most of the cultures a suitable temperature range is between 18^{o}C to 25^{o}C, however regarding the pH and eC levels the values are a little more restrict as seen in table 2.1 .

For this reason to make a production of more than one type of crop their values need to be similar like lettuce and peas or strawberries and tomatoes.

Table 2.1: Values of pH and eC of vegetables

Plant	pH	eC (mS)
Bean	6.0	2.0 - 4.0
Broccoli	6.0 - 6.5	2.8 - 3.5
Cabbage	6.5 - 7.0	2.5 - 3.0
Cucumber	5.8 - 6.0	1.7 - 2.5
Lettuce	5.5 - 6.5	0.8 - 1.2
Onions	6.5 - 7.0	1.4 - 1.8
Peas	6.0 - 7.0	0.8 - 1.8
Pumpkin	5.5 - 7.5	1.8 - 2.4
Strawberries	5.5 - 6.5	1.6 - 2.2
Tomato	5.5 - 6.5	2.0 -5.0

2.1.2.2 How to measure

- **pH** : The pH concept was introduced by Sørensen in 1909. Where the "p" comes from potenz which means concentration in German and the "H" represents the hydrogen ion (H +). The pH is a logarithmic function whose result varies between 0 and 14. Where 0 indicates that the acidity of the solution is very high and 14 indicates that the solution is too basic or alkaline. As H+ is an ion, it has a certain charge and can be electrically measured as a potential using a pH probe. A pH probe usually has a glass electrode which is composed of a glass bulb with a neutral solution inside and a silver electrode, and a reference electrode which is in a neutral solution of 7 pH. The pH value is calculated from the potential difference between the reference system and the measuring system (glass bulb electrode).

- **eC** : The eC of a solution represents the ease or difficulty of the passage of electric current. This is due to the existence or not of certain ions of compounds in the solution. With this interference of the current flow is possible to determine which are the compounds dissolved in a solution. An eC probe is composed of two electrodes one centimeter apart and as an AC signal is continuously applied the conductivity is measured by the amount of current that passes through the solution between the two electrical conductors. Usually electrical conductivity is measured in mili or microSiemens per centimeter.

2.2 Hydroponic monitoring and control systems

2.2.1 High end commercial Monitor and Control hydroponic systems

A few companies already provide hydroponics systems and control solutions. While some are specialized in irrigation control other are focused in nutrient and water control. Below are presented examples of such companies and their systems.

- Autogrow manufactures a range of equipment for the control of greenhouse climate, irrigation and hydroponic dosing, such as: Multigrow hydroponic eC/pH doser: NutriDose I, IntelliDose eC and pH dosers: MiniDosers. They also make monitoring systems.

- Hanna Instruments manufactures analytical instrumentation. And meters for different purposes. The Hanna Instruments pH&eC meter (HI 98130) is a waterproof pH and eC/TDS meter with temperature compensation [11].

- Hortimax provides low-end and high-end solutions to horticultural processes, from control and climate monitoring-To automate irrigation and nutrition [12].

- PureHydroponics is a company based in New Zealand and supplies all kind of systems for hydroponic farming, from physical structures and growing media to control and measurements systems [13].

- Campbell Scientific provides data acquisition, measurement and control systems for different purposes and not specialized in hydroponic farming although their systems can be customized for that end [14].

2.2.2 Low end commercial hydroponic systems and Sensors

- Libelium [15] manufactures hardware and Application Programming Interfaces(APIs) for wireless sensors networks and in October they launched the Open Garden, a sensor platform that automate control and maintenance tasks in gardens through wireless connection using open source APIs. Open Garden includes a set of sensors such as humidity, light, temperature, or soil moisture to monitor plants for optimal care wherever they are situated. Its actuators can control irrigation, and activate lights and oxygen pumps. A Hydroponics kit is also available including pH and conductivity probes.

- Atlas Scientific [16] offers low-end solution for environmental monitoring. They provide full kits for water monitoring like pH, electrical conductivity, oxidation reduction potential (ORP), dissolved oxygen and temperature.

2.3 Overview Wireless Technologies

This chapter presents the different wireless technologies studied - Bluetooth low-energy, ANT, ANT +, ZigBee and Wi-Fi. During the presentation we will focus on the comparision of: (1) the type of topology supported by each protocol; (2) their costs; (3) their efficiency; (4) their energy efficiency; (5) their performance; (6) their robustness; (7) their data latency; (8) and their peak consumption [4].

2.3.1 Bluetooth Low Energy

Bluetooth Low Energy (BLE) [17, 18, 19, 20] is a low power variation of Bluetooth standard that was created in 1998. The BLE or Bluetooth Smart is managed and developed by Bluetooth SIG. This technology was designed to work with low-power consumption. Comparing with Bluetooth standard, the data protocol was changed to create low-duty-cycle transmissions or a very short transmission burst between long periods. The BLE has sleep modes, this mode allows a Bluetooth Smart product to operate for many years with one coin cell. Nowadays, we can find this technology in many devices like tablets or smartphones.

2.3.2 ANT™

ANT [17] is another low-power wireless technology designed for sensor networks. The implementation of this technology is simple, it is several times used in sensor network like sports, health, home automation and industrial area, although this does not mean that this it widespread. In fact this is a technology not so commonly used by the smartphone and laptop industries.
Like BLE, ANT also uses the very short duty-cycle technique and sleep modes to ensure very low power consumption. It operates for years with a coin cell. ANT+ is an upgrade of ANT. ANT+ makes easy the collection of sensor data with specific implementations.

2.3.3 ZigBee

ZigBee [21, 19, 20] networks are known for taking advantage of simple and low cost electronic, and to be low power device that allow great scalability and expansion, which makes this technology ideal for sensors, controllers, remote monitoring and portable electronic devices. However, since consumer laptops and smartphone do not support this technology it necessary to install a gateway.

2.3.4 Wi-fi

This technology [20, 22] was designed to transfer large data quantatie using high-speed throughput. It is a cheap solution when you want to implement LAN networks, not to mention that this technology is used in houses, universities and all mobile devices such as smartphones, tablets, laptops, etc.

2.4 Network Topologies

It is possible to implement three types of network topologies in wireless sensors:

- **Mesh** — A mesh network consists of a network in which all nodes are linked with one or more connections. So, if a node wants to send a message to a node that is not within reach, should send the message to the nodes between the sender node and the receiver node *push*;

- **Star** — The Star network consists of a network where there is a central node and multiple nodes installed in a star form. If a node wants to communicate with another node, the sender node must send the message to the central node, and this directs the message to the receiver node;

- **Point-to-Point** — In this mode, a one-to-one connection exists, where only two nodes are connected.

With this brief presentation of the most typical network topologies, we can make a connection between wireless technologies section 2.3 and network topologies.

As a first analysis we have the BLE technology. Where the three types of topologies are supported. An observation to make at this technology is that it allows Broadcast message, which can prove an advantage when we need to make, rapidly, the dissemination of information.

Then, the ANT (+) technology supports a communication according to the three types of topologies previously presented. Compared to the BLE, the ANT(+) allows Broadcast message, but the receiver node need to send the message back to the transmitter node. This can be a disadvantage if there are many nodes in the system, because it can cause limitations on the signal bandwidth.

The Zigbee supports the three types of topologies presented. Compared to previous technologies, the Zigbee does not allow Broadcast communication.

Finally, the Wi-Fi has the same specification of Zigbee, does not support the Mesh topology and does not send message in Broadcast.

2.5 Power efficiency

The energetic efficiency is extremely important when you want to develop mobile devices that have a high capacity. One of the most important factors in this thesis is the energy efficiency of wireless devices. As described in section 1.3, it is extremely important the development of devices with low energy consumption so that it is possible to guarantee its functioning through a battery. So, it is necessary to know the energy costs associated with each technology. This topic will present the power efficiency for different technologies. The following assessment is made power consumption per bit:

2.5.1 ANT

In ANT each packet has 20 bytes of useful payload and consumes 61 μA at 3 V.

- A byte consists of 8 bits, therefore 32 x 8 = 256 bits/second

- Power = VI = 3 V x 61 μA = 0.183 mW

- Power per bit = 0.183 mW / 256 bits = 0.71 μW/bit

2.5.2 Bluetooth low energy

In BLE, Each packet has 20 bytes of useful payload and consumes 49 μA at 3 V.

- Power consumption = 49 μA x 3 V = 0.147 mW
- Bytes per second = 20 x (1 second/500 ms) x 3 channels = 120 bytes/second
- Bits per second = 120 bytes/second x 8 = 960 bits/second
- Power per bit = 0.147 mW/960 = 0.153 μW/bit

2.5.3 Wi-Fi

Wi-Fi consumes approximately 116 mA at 1.8 V when transmitting a 40 Mbps User Datagram Protocol (UDP) payload. Unfortunately, current consumption does not reduce when throughput is reduced in a Wi-Fi chipset.

- Power = 116 mA x 1.8 V = 0.210 W
- Power per bit = 0.210/40,000,000 = 0.00525 μW/bit

2.5.4 Zigbee

A Zigbee device consumes 0.035706 W when transferring 24 bytes of data.

- Bits per second = 24 x 8 = 192 bits
- Power per bit = 0.035706/192 = 185.9 μW/bit

It is possible to see from the calculations that the Wi-Fi technology is the most power efficient technology and it is ideal to large data transference. Unfortunately, its peak current consumption is far beyond the capabilities of a coin cell and would need to be provided with a large battery. Based on this, it is possible to conclude that the technologies with low power consumption and that are able to be powered by a coin cell are BLE, ZigBee and ANT.

2.6 Performance

Table 2.2 presents and compares different wireless technologies for different parameters of performance. These parameters are (1) Range, (2) Throughput, (3) Latency and (4) Power Consumption.

In Wireless technology, maximum range means the maximum distance that a signal can reach on ideal conditions. Therefore, under the scope of this thesis, if we want to ensure the ideal condition of transmission, the technology to choose should fulfill the requirement described in section 1.2, which consists of wireless technology having a minimum range of 54 meters. According to information presented in table 2.2, all technologies meet this requirement. The second parameter to present is the throughput. Throughput of a wireless network can be measured in two ways:

- On air signaling rate, which is often quoted on packaging (for example, Wi-Fi at 54 Mbps).
- Measuring how quickly useful payload data can be transferred, *which is the more useful method*.

Then, the latency of a wireless system can be defined by a user action sent to a receiving device. It is necessary to mention that ANT and Wi-Fi have low latencies but they require that the receiving device is continuously listening and, therefore, they use considerable energy.

Lastly, the peak power consumption is a critical situation when designing long-life low-power mobile devices. In the context of this thesis, it is extremely important the analysis of this parameter, since one of the objectives is to develop devices with a long battery life.

Table 2.2: Performance of Wireless Technologies [4]

Peformance	ZigBee	BLE	ANT	Wi-Fi
Range (m)	100	100	30	150
Throughput (Kbps)	128	305	20	6000
Latency (ms)	20	2.5	0	1.5
Peak Power Consumption (mA)	30	12.5	17	116

2.7 Epidemic Communication Protocols

The Gossip protocol or Epidemic Protocol is used in distributed systems. Initially proposed to implement in database replication systems [23], fault detection [24], publisher-subscriber communication models and application level reliable broadcast. This success is due to the robustness and simplicity of the protocol and the high degree of reliability combined with a great potential for scalability and fault tolerance [25]. The operating principle is very similar to an epidemic disease such as flu, a person with flu will infect another with whom he had contact, this in turn will spread the disease to other agents. Thinking in applications in distributed systems, a node chooses another with which exchange information, wich in turn, disseminates information to its neighbors.

The interaction between nodes can take several strategies [24, 25, 26]:

- Eager push approach: the node must retransmit a message as soon as it arrives for the first time;
- Pull approach: the node 1 chooses the node 2 and extracts its updates;
- Push approach: the node 1 chooses the node 2 and sends him its updates;
- Hybrid approach: the node 1 and the node 2 exchange updates.

Another aspect necessary to define the Gossiping epidemic algorithm is the spread technique to be used.

2.7.1 Dissemination techniques

The techniques can be [27, 26, 28]: **Direct Mailing**, **Anti-entropy** and **Rumor Mongering**. In **Direct Mailing** technique when the node has a new information it sends it to all its neighbors in broadcast. It is a simple method but it generates a lot of traffic on the network. In the **Anti-entropy** technique a node randomly chooses another to exchange data, this in turn chooses another node to exchange data again. Eventually all the nodes will be updated, because the probability that a node does not receive data will tend to zero. Finally, in the **Rumor Mongering** technique when a node generates a new update, this becomes a “hot rumor”. This rumor is sent as many times as possible for the neighbors of the sending node. When the entire network knows the rumor, the algorithm stops to spread the rumor.

2.7.1.1 Applications with gossip protocol

Trickle [29, 30, 28] is a code update algorithm of a task designed to apply in wireless sensor networks based on **Rumor Mongering** technique of the Gossip protocol. When a node has a new update it disseminates in broadcast information until all have received the update. In [30] is shown that with this simple mechanism, **Trickle**, you can quickly disseminate an update in multi-hop with low overhead with network congestion control. This can be done through the probabilistic transmission mechanism and the counter-based.

The probabilistic transmission [31] is the probability of transmission of a node. If the probability is 1 the node sends a broadcast message, otherwise the message is sent to the number of nodes corresponding to the probability of transmission. In the counter-based mechanism, when a node wants to transmit a message, first listens to the medium, if it is busy, the node to transmit will wait for the release of the medium.

As each message is associated with a count of the number of times the message was retransmitted, when a node to transmit collects a message it is up to him to check if the message counter is lower than the collected message counter, if there is this condition the message is retransmitted.

The operation of the [32] algorithm describes that when a node detects an event, it sends a consensus to the neighborhood and waits for a quorum. If the neighboring nodes confirm the same event for the node that sparked the consensus, it reports the event for the sink.

Directed diffusion is an exemple of datacentric routing. The data sink identifies a set of attributes, packs them into a message and sends it throughout the network. The message is sent in broadcast. Each receiving node records the interest and define a gradient, the state indicating the next hop direction for other nodes to send the message. When this message arrives at a data producer, data are being forwarded to the sink along established gradients. Each node compares the interest packaged in the message with the measured data, considering that a node contains a sensor. It is also compared if its location corresponds to the location from the interest message. In response to the sending of an interest message, the protocol uses the memorization of the path so the requested data return to the sink.

Another data dissemination technique is the Sensor protocols for information via negotiation, **SPIN** [29]. The operation of this protocol consists of using data descriptors, called metadata. This metadata transfer, with size smaller than the real data, eliminates sending redudant data for the entire network. Thus, the network overload is reduced both in terms of bandwidth and energy consumption. With this technique is possible to solve three existing problems in classic Gossip Protocol: Implosion, Overlap and Resources Blindness.

LEACH [33] described the low-energy adaptive clustering hierarchy (**LEACH**) protocol.

When a node decides to become a cluster-head, this sends a message to the neighboring nodes reporting that decision. With admission of the new cluster-head, the neighboring node sends it a status recognition message. With the recognition of all nodes within range of cluter-head, this gives each node a time slot for updating data, which subsequently sends to the sink. The selection of cluster-heads is repeated periodically, so there is a balance in the energy consumption of the multiple nodes.

If a node does not receive any type of message before the timeout, it restarts the protocol with the decision to become cluster-head.

In [34] suggests a mobile collector to collect data from nodes. The operating idea is, that the collector in order to collect the data has to travel between nodes. The nodes data are stored in cluster-heads, and are only those that report the data. The coordination of communication between cluster-heads and the collector can be done in three different ways. Round-robin: neatly and for a certain period of time is requested from each cluster-head the stored data. Rate-based: the number of visits to each cluster-head is defined by the amount of data to share. Min-movement: neatly from each cluster-head is requested the data for a certain period of time proportional to the amount of data to be obtained.

2.8 Summary

In the first place, have been described the advantages, applications and different techniques of the Hydroponic Agriculture. Following is an explanation of the measurement techniques of the pH and the eC of the aqueous solution feeding a hydroponic culture.

The chapter continues with a presentation of monitoring and control systems for hydroponic cultures. First are presented hydroponic systems existing on the market, for large-scale applications. Secondly are presented sensors for reading of physical quantities pH and eC of the water.

Finally, was made an introduction of what are Epidemic Communication Protocols, where were described the different data dissemination techniques such as (1) Direct Mailing, (2) Anti-entropy and (3) Rumor Mongering. As conclusion of this explanation, were described several examples of developed algorithms that apply Epidemic Communication Protocols.

Chapter 3

System Specification

This chapter begins with an introduction about the BLE technology, which is used on the system. In this introduction, is explained what is this technology and how it works. Then is explained the Softdevice, a pre-compilation of the BLE stack developed by Nordic Semiconductor for use in the BLE devices that it manufactures. Lastly, is described the system created to address the objectives presented on section 1.3 with the conclusions from chapter 2. More precisely, it is presented the system architecture along with the type of communication protocol and communication technology to use, the elements and layers of the communication network, as well as the hierarchical relationship between the elements.

3.1 Adopted Technologies and Techniques

As described in section 1.3, one of the objectives of this thesis is the development of a sensor network with low energy consumption and high capacity. To achieve it, it is very important to choose a technology with high performance and low energy.

Based on these requirements and in the in section 2.3, the technology best suited to this relationship is BLE.The choice of this technology is based on the evaluation made in section 2.4, section 2.5 and section 2.6.

In section 2.6, the BLE technology has proven to have the best performance at a low energetic consumption (section 2.5). Other advantages of this technology is the support of the two types of topologies (section 2.4) and to allow the sending of broadcast message. The latter is a great advantage when is intended to implement Epidemic Communication Protocols.

In addition to the technology characteristics be aligned with our requirements, this shown to have a promising performance. This technology is increasingly used in the industry of smartphones, tablets and laptops, thus opening the possibility of creating numerous solutions that incorporate a BLE modem to communicate with this type of devices, offering the user an easy and quick integration with BLE applications. This way, within the scope of this thesis, it is of total interest a quick and easy availability of data obtained from sensors without the need for additional hardware. That is, from a smartphone it is possible to easily access data obtained by a certain BLE device. Another advantage is that, with the BLE, with the appropriate amendments, it is possible to communicate with devices of Bluetooth Classic, making it possible to open more the range of devices to communicate with.

Another aspect that is related to the capacity of a device is the way that the elements of a network communicate among themselves, i.e., the communication protocol implemented. As described in section 2.7, an example is the epidemic communication protocols. This type of protocol shows to have great characteristics for the developing of a WSN, since it provides robustness and reliability with low power consumption.
In this thesis, the technique implemented in such protocols is the Direct Mailing. This choice is due to the fact that there is no synchronization between the system elements, otherwise it would be necessary to have message exchanges for synchronizing all the network elements which could take a long time, in case the system have large dimensions, as well as the increase in data traffic and the energy expenditure associated with the transmission of the synchronization messages that would be reflected in decreasing capacity of battery installed in the network element.

3.1.1 Bluetooth Low Energy Technology

Bluetooth Low Energy is a technique that appeared in version 4.0 of Bluetooth, in 2010. This technique allows very low levels of energy consumption on devices that do not need to transmit large volumes of data, and may have an energy expenditure of only 10% compared to classical Bluetooth [35], making it possible to feed them with batteries such as button-cell CR2032 due to the fact of energy consumptions with peaks of 15 mA, but with an average of only 1 uA [36].
Due to its striking characteristic being the energy economy, a BLE device remains in sleep mode during most of the time, coming out of this mode just to make connections that last only milliseconds. Connections that in BLE are the most optimized possible according to the redefinition of the client-server architecture, asymmetrical design, discovery of BLE devices, among others [36].
Despite inheriting characteristics of classic Bluetooth, a BLE device is not compatible and does not work the same way as the traditional version. Thus, a device that only implements the BLE is called single-mode or "smart" and only makes connection with a similar, this being the most common case in sensors and smartphones [37]. However, the implementation of both dual-mode or "smart ready" architectures is possible in the same device, through the implementation of a protocol that mediates between classic Bluetooth and the BLE. As a disadvantage, this implementation does not usually have the same energy gain of the first [37].

3.1.1.1 Architecture

The BLE architecture is divided into host and controller (Figure 3.1). The controller is capable of allowing the host to sleep for long periods and only wakes up when it is necessary to do some action.

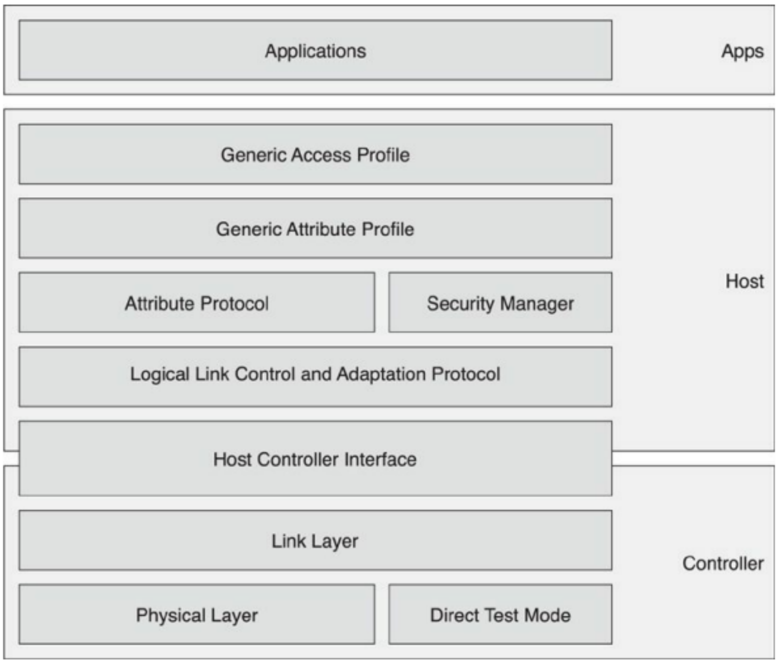

Figure 3.1: The Bluetooth Architecture [2]

3.1.1.2 Physical Layer

One of the procedures that BLE inherits from the classic version is the physical layer, that use Frequency-hopping spread spectrum (FHSS). FHSS is a method of transmitting radio signals which consists of constantly changing the carrier through various frequency channels using a pseudorandom sequence known to both transmitter and receiver.

In the classic version there are 79 channels of 1 MHz while the BLE works with 40 channels of 2 MHz. In these 40 channels there are two types: advertising channels and data channels. The advertising channels occupy 3 of the 40 channels and are used to discover other devices, establish a connection and broadcast transmission. The second type, data channels, are used for bi-directional communication between connected devices [38].

In Figure 3.2, the green color symbolizes advertising channels and blue symbolizes data channels.

However, to use the free spectrum of 2.4GHz, the BLE runs the risk of colliding with a Wi-Fi/802.11 signal which is operating in a given frequency. As such, on the data channels channels is used Adaptive Frequency Hopping(AFH) mechanism. This process causes the available frequency map to be readapted, so that the channels already occupied are excluded from the availability list. In addition, the AFH make master and slave use the same channel, in order to prevent that one uses a good channel and the other uses a bad channel, which could cause a series of retransmissions.

The modulation used is the Gaussian Frequency-Shift Keying, with index between 0.45 and 0.55, greater than 0.35 of Bluetooth. This arrangement allows lower energy consumption, longer range and greater robustness (less susceptible to interference). The BLE has a data rate (data rate)

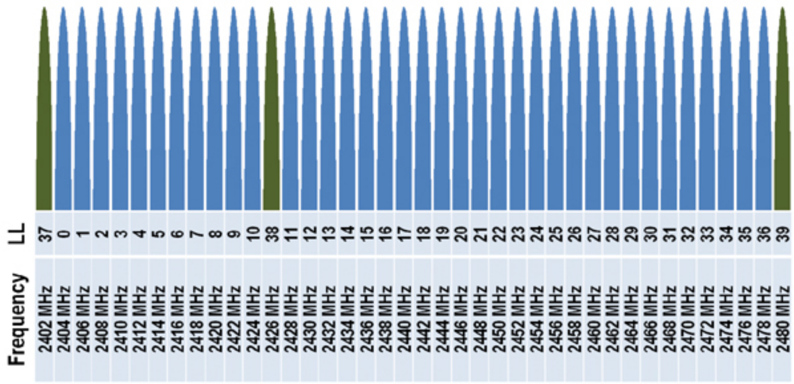

Figure 3.2: Frequency spectrum of the BLE [2]

of 1Mbps (BT 1 - 3Mbps); body of data (payload data) with size <0.3Mbps (BT 0.7 - 2.1 Mbps); signal range of 50m (BT 10 - 100m).

3.1.1.3 Link Layer

This layer is responsible for monitoring the device connection status. Each link operates in five states, which are: standby, advertising, scanning, and initiating connection.

Figure 3.3 shows the relationship between these states. Only one of these states is active at a given instant.

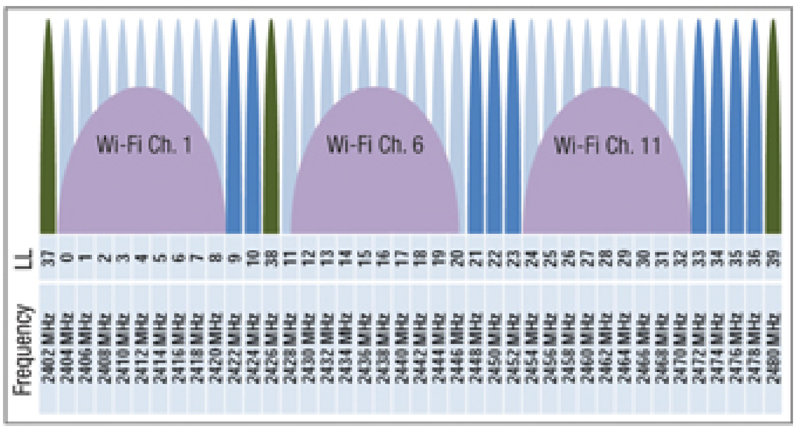

Figure 3.3: Adaptive Frequency Hopping [2]

In Standby state, the device does not receive or send any package and can be accessed from any other state. In Advertising state, the advertiser device sends packets to each advertising channel and can respond to requests from other devices. A device in this state can be accessed through the standby state. In Scanning state, the scanner device listens to advertising packets transmissions and may request additional information to other devices. In Initiating state: the initiator device, unlike the scanner, intends to connect to an advertiser. By locating one, it responds requesting the start of a connection. The link layer to operate in the state Connection, indicates that the connection is formed between the initiator and the advertiser, both enter in this state and communicate through data channels. Any state may be accessed by the standby state.

Formed a connection, now each of the devices takes on a role: master (initiator) and slave (advertiser). The model follows the piconet network, the master element is responsible for managing multiple connections with slaves, these in turn can only be connected to one master.

The slave, by default, in order to save energy, stays in sleep mode. The master, through Time Division Multiple Access scheme, where packets are sent at predetermined times and intervals, is also responsible for waking up the slave in order to fulfill the transmissions. In addition, it is part of its responsabilities the management of FHSS algorithm, the connection supervision and the updates of the map of available channels.

After this, the connection between elements of a network and the sending of information is carried out by two kinds of event: advertising and connection. In advertising event, the advertiser device uses all three channels sequentially (10ms interval) to send an advertising package. Scanners receive these packages without the intention of connecting and can make more information requests; the advertiser can then respond to the same channel. It can also stop the event when it wants and recommence. The advertising interval corresponds to the time spent between two advertiser events, and goes from 20ms to 10.24s, with multiples of 0.625ms.

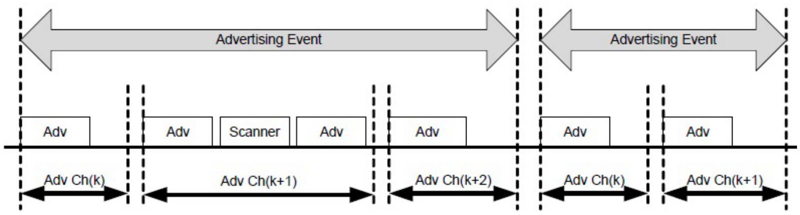

Figure 3.4: Advertising Event [2]

In connection event, the initiator device is listening to advertisers in advertising channels. When it finds a device ready to connect, it sends a connection request message to the advertiser, which, accepting, generates a point-to-point (P2P) connection between the two devices. The initiator becomes master and can break the connection with the slave (before advertiser) at any moment. The connection interval follows the same advertisment interval logic, corresponding to multiples of 1.25ms between 7.5ms - 4.0s.

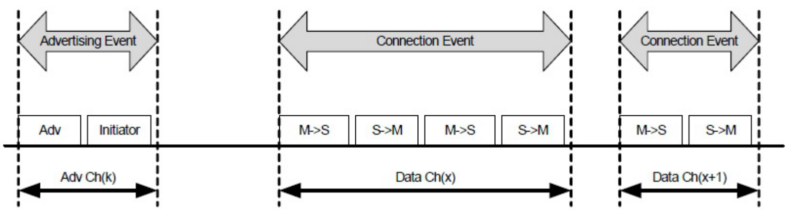

Figure 3.5: Connection Event [2]

3.1.1.4 Host-Controller Interface (HCI)

HCI layer provides the means of communication between the controller section (Controller) and the host section (Host) through interfaces such as an API or hardware interfaces such as Universal Asynchronous Receiver/Transmiter (UART), Serial Peripheral Interface (SPI) or Universal Serial Bus (USB).

3.1.1.5 Logical Link Control and Adaptation Protocol (L2CAP)

The L2CAP is responsible for encapsulation of messages to the upper layers, allowing a logic data communication end-to-end between BLE devices. The higher layer protocols provide data structures that fit the maximum size of the L2CAP payload, which is 23 bytes.

3.1.1.6 Security Manager Protocol

Defines link methods of devices of a network, providing functions to the other layers so that the exchange of information is done in a safe manner. This protocol defines the methods for pairing and distribution of keys and functions for other stack layers to connect safely and to exchange data with other devices. Data exchange is performed only when the connection is authenticated, that is, when there is a pairing between two BLE devices.

When this process occurs the keys are established either to encrypt as to authenticate the connection [39]. The most commom case consists of a peripheral device request a key, called "passkey", to the central device to complete the pairing process. After the central device sends the key correctly, the two devices exchange security keys to encrypt and authenticate the connection.

However, as a central device and a peripheral device can be connected and disconnected, repeatedly over time, the pairing process may not be efficient in energy and temporal aspects. For that, BLE has a safety function, called "bonding", which enables the two devices, in the pairing process, provide each other a set of long term security keys. This function allows two devices quickly restore the encryption and authentication after re-connect without having to go through the pairing process again.

The link layer supports encryption and authentication using algorithms of ciphers for blocks. When it is applied encryption and authentication to a connection, 4 bytes are added to the channel payload. These 4 bytes are called Message Integrity Check. It is also possible to transmit authenticated data through an unencrypted connection. In this case, a 12-byte signature is placed after the payload in the Attribute Protocol(ATT) layer. The signature is made by applying an algorithm that uses 128-bit Advanced Encryption Standard(AES) as block ciphers.

3.1.1.7 Attribute Protocol

This protocol allows a given device to expose certain components of its data to other devices, called attributes. Attributes are data structures that store information provided by Generic Attribute Profile(GATT) layer subsubsection 3.1.1.8. An attribute consists of 4 elements: 16 bits to the handle that defines solely one attribute; a Universal Unique Identifier(UUID) of 16 or 32 bits that defines the type of attribute; its description; and finally, its value of variable size. In the context of ATT, a device that exposes its attributes is referred to as the server and the neighbor device is the client. This designation is independent of the designation previously seen in the link layer.

Through the implementation of the ATT, one master device can be both a client and a server, the same happens to the device that takes the slave paper or a device that can be client and server simultaneously. The client can access the server attributes by sending requests, and the server, in order to ensure efficiency, can send to customer two types of unsolicited messages containing attributes, which are: notifications, where the server sends the data to customer and it does not confirm receipt; or indications, that require confirmation by the customer.

3.1.1.8 Generic Attribute Profile

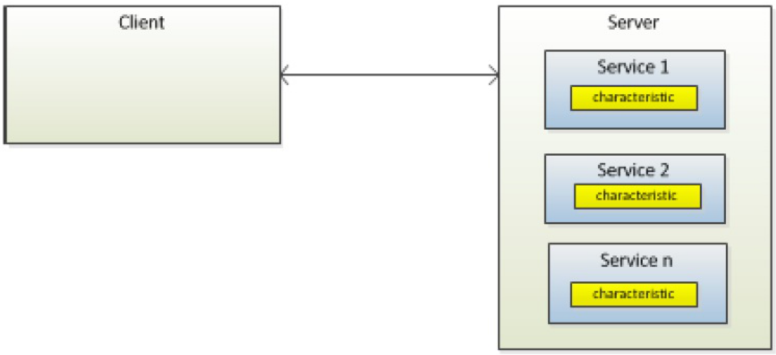

Figure 3.6: GATT Architecture [2]

The GATT layer defines a framework for use of procedures of the ATT layer for service discovery and specify the structure of the profiles. All data communications that occur between devices in a BLE connection are processed through the procedures in this layer. All data that are being used in a service are called "characteristics" which are in turn a set of data which includes values and certain properties. This data is stored in attributes. Hierarchically we can say that a profile can have multiple services and each service can have several characteristics. The definition of services and characteristics is organized in the form of attributes mapped in a table, called the attribute table, stored in the GATT Server. When necessary a GATT client accesses these attributes through read and write operations, depending on the permissions implemented.

3.1.1.9 General Access Profile

The GAP layer defines the direct interface with the applications and profiles as well as the management of the specification of the roles of the devices, the connection and security of communication. Altogether there are four roles that operate exclusively: broadcaster, observer, peripheral or central.

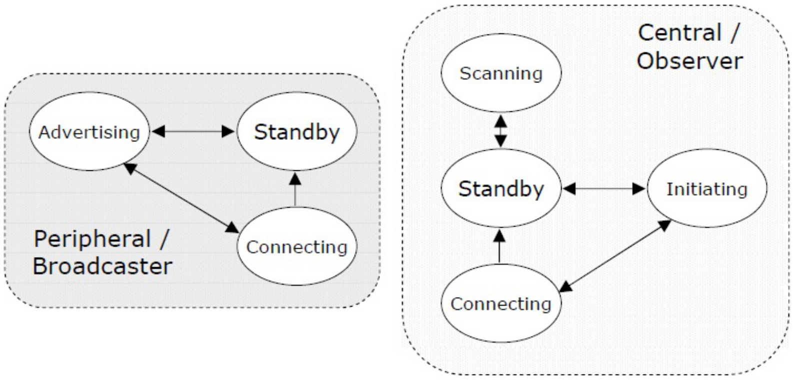

Figure 3.7: GAP Layer [2]

A device in the role of broadcaster transmits information in broadcast through advertising channels, without being able to connect with other devices. A device that is observer is the complement of the broadcaster, i.e., listens to the advertisements, not having also the ability to connect. A central device searches for ads and starts connections, operates as master and can manage multiple connections. Finally, a peripheral device is an advertiser that has the competence to connect and operate as a slave on a simple connection, i.e, is connected to only one master.

In a typical BLE system, the peripheral device advertises that it is a device ready to establish a connection, to inform the central devices.
These advertisements contain the device unique address and may contain additional data such as the device name. The central device, after receiving the advertisement, sends to the peripheral device a "Scan request" to which this responds with a "Scan response". This is the devices discovery process, to the extent that the central device becomes to know that the device, with which it has communicated, is a peripheral station and may thus establish a connection. The central device then sends a request for connection establishment (Link Request) to the peripheral device, and it responds with a "Link Response".

3.1.2 Softdevice

Softdevice is the commercial implementation of a precompiled library of the host layer and the controller used in the NORDIC SEMICONDUCTOR [40] development boards nRF51822. The purpose of this pre-compilation is to provide a flexible manipulation of the Generic Attribute Protocol (GATT), Generic Access Profile (GAP), and Logical Link Control and Adaptation Protocol (L2CAP) explained in subsection 3.1.1, for application development for Bluetooth® Smart devices.

This flexibility is made through the use of an API, which is based on the System-on-a-chip(SoC) Library and Softdevice Manager. This way there is an abstraction in controlling the controller element, HCI and advanced properties of the Host element, such as the definition of the operating rules of a BLE device.

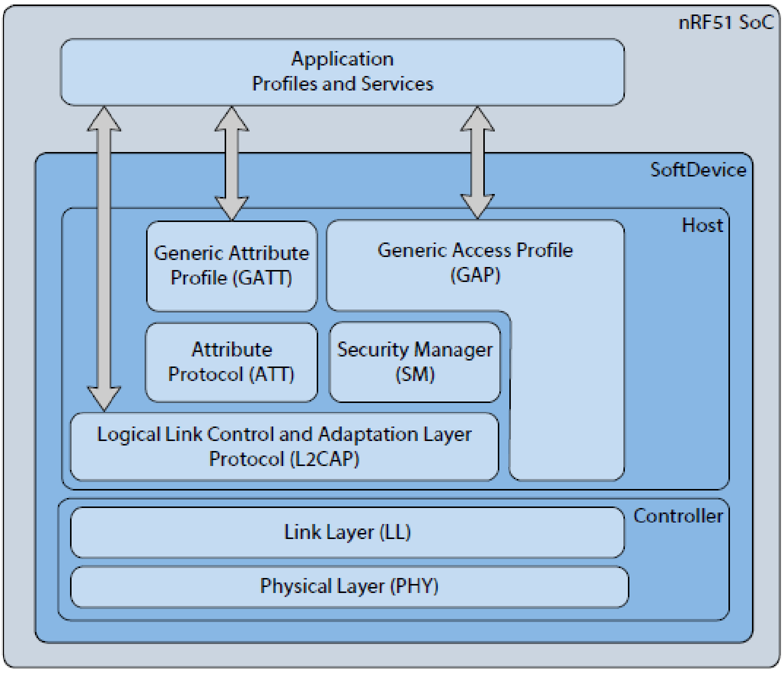

Figure 3.8: SoftDevice stack architecture [3]

Considering this there are three types of softdevice, S110, S120 and S130, that differ between each other on the configured operating rule, that may be peripheral, broadcaster, center and observer.

In the pre-compilation S110 are compiled the peripheral and/or broadcaster operating rules. While in the S120 the pre-compilation consists of the central and/or observer operating rules. Finally, the Softdevice S130 is already a pre-compilation that supports all the BLE standart operating rules, i.e., peripheral and/or broadcaster and/or central and/or observer.

In the context of this thesis, based on the description of the system of the section 3.2, the S110 and S130 softdevice are used, because it is necessary to ensure a peripheral behavior in the sensing device, and a multirule behavior central, observer and broadcaster in the gathering device.

3.2 System Architecture

In order to address the requirements presented on section 1.3 this thesis proposes to develop a distributed wireless communication architecture, broken down by hierarchical layers(figure 3.10). This hierarchy is composed by three distinct layers: (1) sensing layer; (2) gathering layer; and (3) cloud layer. In Sensing Layer, the different devices perform the periodic reading of the physical and electrical quantities of the water necessary for a hydroponic planting and report to the upper layer, Gathering Layer, when one of its elements request. In the Gathering layer, the devices of this layer collect the data obtained in the sensing layer and sends these in broadcast to other devices of the same layer and of the upper layer, the Cloud Layer. The Cloud Layer, contains a device that reads the messages whose senders are the devices of the Gathering layer

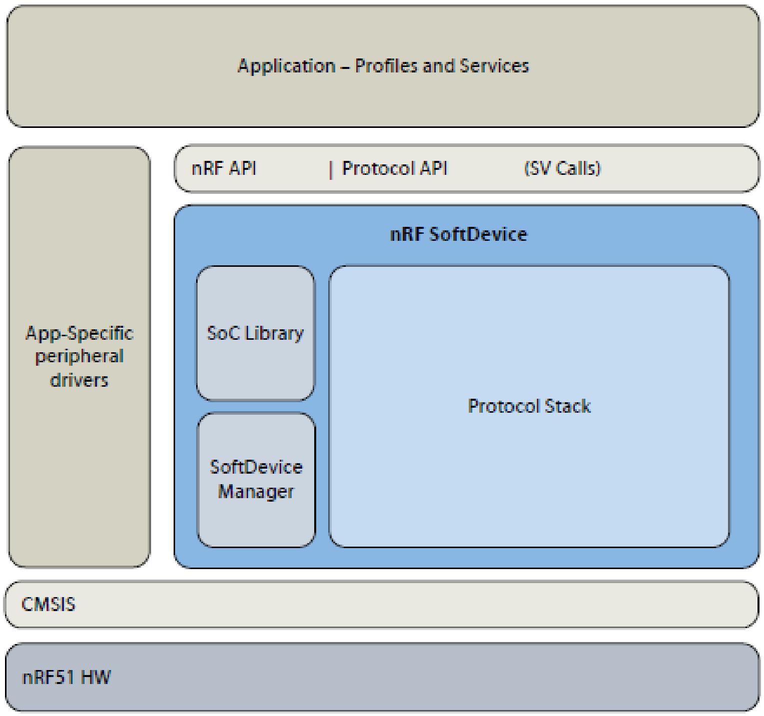

Figure 3.9: System on Chip application with the SoftDevice [3]

This composition resulted from the need to ensure a system that is reliable, scalable and has low power consumption. A reliable system is guaranteed due to the cooperation model implemented in the layers. This statement arises from the possibility to integrate two Gathering Devices in the same range of a set of sensing devices. Case one of the Gathering devices fails, the other will ensure the communication with the set of sensing devices, thus having a redundant system. Among elements of the Sensing layer and Gathering is used the model of **master-slave** cooperation in which the Gathering is the master. This way, the pairing of the master with the slave ensures the exchange of information.

The other cooperation model between Gathering devices is **publisher-subscriber**, which, by sending the data in broadcast, leads to a duplication of data, thus, with the proper control of the message sending is possible, in a simple way, to ensure the delivery of data to a cloud device.

These type of cooperation models leads to a minimal energy consumption. With the master-slave cooperation model, the sensing device communicates only when the gathering device initiates the communication. This behavior contributes to the minimum spending of the capacity of the sensing device, since this is powered from a battery. As each element of the gathering layer can be aggregated with various sensing devices and the data from de gathering device are sent in broadcast, is obtained an easily scalable system.

3.2.1 Sensing Device

The sensing layer is the lowest layer of the system, where each node periodically reads the physical quantities to monitor and sends them to the upper layer. Each node of this layer should be composed by the sensors, a battery, a microcontroller and a BLE modem .

The sensors incorporated into this element are responsible for reading the pH value, eC and the water temperature as well as the air temperature and humidity and atmospheric pressure.

These values are obtained at a given sampling frequency and packaged in an advertising packet to send to the element of the upper layer, Gathering device.

This way, by reading and periodic sending of data it is possible to minimize the energy expenditure of the device, being possible to ensure the capacity of this through a battery.

The need of the structure of this element comes in response to the requirements described in section 1.3, so it is possible to ensure a low power consumption device, of easy installation and maintenance with a range of communication maximum of 100m. The installation of this device is made inside a greenhouse with access to the conduits where pass the solution to be monitored.

3.2.2 Gathering Device

The middle layer of the system is the Gathering layer, and each of its nodes is responsible to gather all the data sent from the lower layer, treat them and periodically send it to the Cloud layer.

The proposed local of installation of these devices would be the top of a greenhouse. Thus, the gathering device communicates with the sensing devices inside the greenhouse in which it is installed and forwards the data obtained to another gathering device installed in another greenhouse. Sending data between greenhouses is done through an advertising packet which is successively sent from Gathering device to Gathering Device until reaching the Cloud device.

However, the successive communications that a gathering layer is subject, results in a high energy expenditure. In response to this problem and based on the requirements described in section 1.3 it is proposed that the nodes of this layer should be composed by a battery, a photovoltaic panel, a BLE modem and a microcontroller.

This way, the installation of the gathering device on top of a greenhouse makes it possible to charge the device's battery through a photovoltaic panel, having as guarantee an capacity of the device for extended periods of time.

3.2.3 Cloud Device

Finally the Cloud layer, the highest layer level of the system, plays the role of gateway for the system allowing it to communicate with the outside world, i.e., the cloud layer receives the data sent from the gathering layer and sends it to a cloud service, allowing to remotely control the hydroponic farm. Each node of this system should be composed by a BLE modem to interact with the Gathering layer, and a GSM and WiFi module to communicate with the cloud service according to the communication services available at each installation. Additionally, it could be useful to add to the gathering layer a graphical interface to allow the user to use this nodes to locally consult the state of the farm.

The interaction between these three layers should be done using a communication protocol tolerant to node failure, that enables easy scalability with the least possible intervention and which is oriented to a high energy efficiency. One example of such protocol is Gossip(section 2.7). By using such a protocol as the bases of communication between nodes and layers, we will be able to comply with all the requirements needed for these communications.

With this approach the communication between nodes in the same layer, Gathering layer, and between nodes from different layers, Gathering/Cloud layers, will be made in a epidemic form, guaranteeing robustness to node failure, easy scalability and network modification.

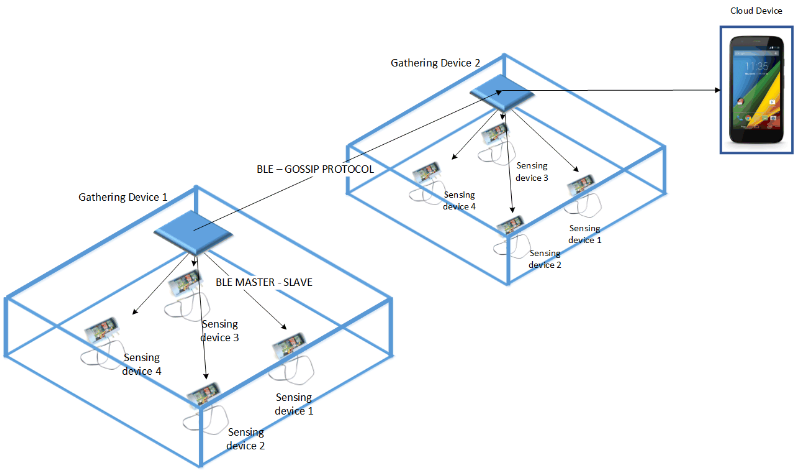

Figure 3.10: System Architecture

3.3 Framework

In a BLE communication are used two types of packets, Advertising packet and Data packet, wherein the base structure of these two is defined in the Link Layer. As illustrated in figure 3.11 the basis of these packages is composed by 1 byte Preamble 4 bytes of Access Address, 2 to 39 bytes of Protocol Data Unit(PDU) and 3 bytes of Cyclic Redundancy Check(CRC).

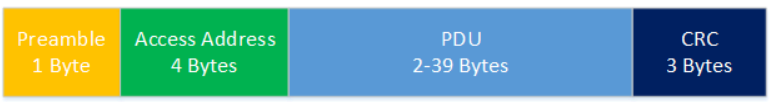

Figure 3.11: Base Structure of packet

Within the PDU field, the PDU Header defines the type of package to send, advertising or data. This difference consists of the configuration of the transmission channels on which the packet is sent. In PDU Payload are attached L2CAP signals and data.

These two types of packages are used to communicate between the different elements of the proposed system. In the first layer of the system, periodically a sensing device wakes up from sleep mode and sends an advertising package for a gathering device. This package is composed of the sensing device name, for example "SENSOR". This way, when the Gathering device receives an advertising with the name of "SENSOR" it knows that is communicating with a sensing device and initiates the pairing process. This process ensures that only one gathering device contains the information of a sensing device. This way, is avoided the duplication of the same data in different Gathering devices, the unnecessary increase in data traffic on the network and the energy expenditure increase of the sensing device to transmit to all the gathering devices that request data relating to a sampling period.

Established the pairing, the gathering device reads the attributes of the characteristic of BLE service defined in GATT server with use of data packet.

The contents of the data packet is made up of the quantities obtained from reading the device sensing sensor.

After reading, the gathering device terminates the connection and waits for a new advertising packet, in turn, the sensing device with the end of the connection goes into sleep mode until the next instant transmission.

With the attributes read, the gathering device packs the information obtained in an advertising packet and sends it to the network. The gathering devices that receive the message relay the same in an advertising packet to other nodes until the message with useful information reaches the cloud device, as explained in section 3.2.

3.4 Summary

Firstly, the BLE technology was presented. In this presentation are explained the general aspects of how the BLE stack works and how it is structured. Then is explained the Softdevice, how this tool works and the importance of it for the execution of this thesis. The different versions of this allow the manipulation of the BLE stack for the different operating rules.

In third place, it was presented a system architecture, which is composed of three layers hierarchically distributed: (1) Sensing layer, (2) Gathering layer and (3) Cloud layer.

In the sensing layer is made the collection of the physical data of the water so that later are collected by elements of the Gathering layer. These elements send data among each other until reaching an element of the Cloud layer. This solution shows that the system can be (1) reliable, (2) scalable and has (3) low power consumption.

- Reliable: in a communication between gathering device and sensing device and gathering device and cloud device is guaranteed the exchange of data;
- Scalable: to a gathering device are associated various sensing devices;
- Low power consumption: the devices communicate in the smallest amount of time possible.

Finally, it was presented the structure of a BLE package, where are described the differences between an advertising package and data. It is based on these differences that we have two types of model of cooperation between elements of the different layers. For a master-slave cooperation model are used data packets, due to the fact that there is a pairing between a sensing and a gathering device. While for a model of publisher-subscriber cooperation, the gathering device uses advertising packages to send data in broadcast to other similar elements.

Chapter 4

System Implementation

This chapter begins with a presentation of the hardware used under the scope of this thesis, thereafter the explanation of the firmware developed and applied for the sensing device and gathering device. Finally, it is presented the application structure developed to incorporate the cloud device.

4.1 Hardware

4.1.1 Sensing Device

As described in subsection 3.2.1, a Sensing Device is composed of a set of sensors, a battery, a microcontroller and a BLE modem. The sensors used under the scope of this thesis are the sensors Atlas Scientific EZO pH circuit and Ezo Ec circuit described in subsection 2.2.2. The choice of these sensors is due to the fact that they communicate by I2C and, compared to the remaining sensors, these show a better quality / price ratio.

The battery used in this project consists of a lithium battery with capacity of 2070 mAh at 3,7V. The choice of this battery is based on the energy needs of the device, which are presented in chapter 5.

The microcontroller to use is the nRF51822[40]. This microcontroller was proposed by Fraunhofer Portugal, since they already have experience with this microcontroller and already have the material in its possession.

The nRF51822[40] was ideally developed for the support of Bluetooth Low Energy technology, i.e., the data transmission operation with a frequency of 2.4GHz with low power consumption. The SoC consists of a 32-bit ARM® Cortex ™ M0 CPU with 256kB/128kB flash + 32kB/16kB RAM, 3 x 16/24-bit timers with counter mode, a RTC and an incorporated transceiver that supports both BLE technology as the stack Nordic Gazell 2.4 GHz protocol with a maximum baud rate of 2Mbps. The nRF51822 requires a power of 1.8-3.6V, a direct 1.8V mode and a on chip DC-DC buck converter giving a supply range of 2.1-3.6V, in which it can be adjusted in accordance with the needs of the microcontroller current and of the radio transceiver.
The following table shows the current peaks for the different Transmition Power:

Current peaks	Transmition Power (dBm)	Observations
6.3	-4	Using DC-DC
8	0	Using DC-DC
11.8	4	Using DC-DC
10.5	0	No DC-DC

The following table lists the current consumption according to the different operating modes of the system:

Current consumption (μA)	System Mode	Observations
0.6	OFF	Using DC-DC
1.2	OFF	8KB RAM retention
2.6	ON	All peripherals in idle mode

For the development of this device is used a development kit from Nordic whose model is the Smart Beacon Kit.

4.1.2 Gathering Device

As described in subsection 3.2.2, a Gathering Device is composed of a solar photovoltaic system, a battery, a microcontroller and a BLE modem.

The control and the BLE communication of the device is made by SoC nRF51822 already described in subsection 4.1.1.

The battery used in this project consists of a lithium battery with an autonomy of 335mAh at 3,7V. The choice of this battery is based on the energy needs of the device, which are presented in chapter 5.

Lastly, the device is coupled to a solar photovoltaic system that in the context of this thesis will be presented the respective dimensioning of the photovoltaic system according to the energy expenditures studied in chapter 5.

The purpose of using a photovoltaic panel is to ensure the continuous functioning of the Gathering device. As described in subsection 3.2.2, this device must be able to communicate with both the sensing devices and the others Gathering devices. This constant communication has considerable energetic losses which makes it impossible to maintain the autonomy of the Gathering device just with a battery in comparison with the case of the sensing device. Another aspect is the access to the electrical distribution network, as this is limited (section 1.2) and suffers cuts it is not feasible to power the device from this source. Thus arises as a solution the use of a photovoltaic panel. This solution is based on the low cost and easy installation of a photovoltaic system and the solar disposition that South Africa presents. According to the data provided by the Department of Energy of South Africa [41], on average the country has 2500 hours of sun per year, wich, an average very similar to Portugal [42], which makes it, according to the portuguese experience, a viable solution as an energetic source.

For the development of this device is used a development kit from Nordic whose model is the PCA20008 [40].

4.1.3 Cloud Device

Based on the composition described in subsection 3.2.3, the Cloud device is composed of a battery, a microcontroller, a BLE, GSM and WiFi modem. Based on these requirements we chose to use a smartphone as Cloud device. The use of a smartphone ensures reliable integration of different communication modules as well as the optimization and the good performance of the different processes that the cloud device can turn out to execute at a low cost. Another advantage in the use of a smartphone is that it already has an operating system integrated that makes the scheduling of tasks. Nowadays this operating system can be Android, iOS or Windows Mobile.

In the context of this thesis, will be used a smartphone with the operating system Android 4.4, more precisely, the model of Smartphone used under the scope of this thesis will be the MOTO G 1st Generation.

4.2 Firmware

In this chapter is addressed the software architecture applied to the different elements of the network. Initially is explained the firmware developed for sensing and gathering devices. This explanation consists of how the code is structured, the functions necessary for a communication and the respective data exchange.

4.2.1 Sensing Device

Based on what was mentioned in section 3.2, the basic functioning of Sensing device consists of three distinct processes: (1) send the advertising packet, (2) connection and sending data and (3) sleep mode. As illustrated in the flowchart of the Figure 4.1, these operations are powered only by a single battery, so it is necessary to ensure maximum optimization of the functioning time of the Sensing Device.

A first approach to this optimization is the temporal definition of the active mode and the sleep mode of the sensing device.

Every 15 minutes, is invoked the active mode of the sensing device, where it sends an advertising packet and waits during a period of time for a connection for the send of data. As the active mode lasts approximately 10 ms, for the remaining time interval, the sensing device enters in the sleep mode.

The choice of 15 minutes period for invoking the active mode is based on a consensus between Fraunhofer and the Olive Tree Farm.

The remaining optimization is done by the BLE technology, as mentioned in subsection 3.1.1. In the case of the hardware used, nrf51822, this optimization and management of the BLE stack is made by the Softdevice S110 (subsection 3.1.2), wich is integrated in the developed firmware.

As illustrated in the class diagram, of the figure 4.2, the integration is made based on three subclasses (1) basic_initialize, (2) start_advertising and (3) initialize_bluetooth_stack.

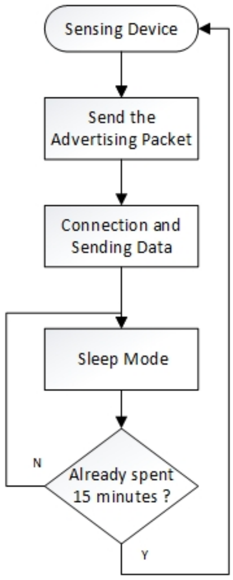

Figure 4.1: Flowchart of Sensing Device

The basic_initialize subclass is constituted by the function (1) timer_init, (2) gpiote_init, (3) ble_stack_init and (4) device_manager_init.

The timer_init function consists of the initialization of the timer module and the start of the timer of the meas_timeout_handler application. After connecting with a Gathering device, this application consists of measuring and send the data obtained from the sensors. The first information obtained are the two least significant bytes of the MAC address, which posteriorly, with the measured data will be stored in an array of the uint16_t type. This array of the uint16_t type is constituted as a whole by 11 positions, where the data are organized in the following order: (1) Mac address of the sensing device, (2) value of the pH of the water, two positions for the (3) value of the electroconductivity, two more positions for (4) the atmospheric temperature, (5) value of the temperature of the water, (6) two positions for the value of the atmospheric pressure, and (7) two positions for the value of the humidity.

The objective of sending the last two bytes of the MAC address is to identify the origin of the message, i.e., associate the sensing device to the information that it sends. This way it is possible to analyze the data obtained in a farm in terms of the sensor.

The gpiote_init function starts the GPIOTE module, that is, starts the use of the microcontroller's I/O pins.

The ble_stack_init function aims the initialization of the events pointer, the activation of the BLE stack and the initialization of the SoftDevice handler module. Only then will it be made the management of events in a BLE communication.

Lastly, it is in the device_manager_init function that the communication security parameters are defined, as for example, the size of the encryption key, the Timeout for Pairing Request or Security Request, among others.

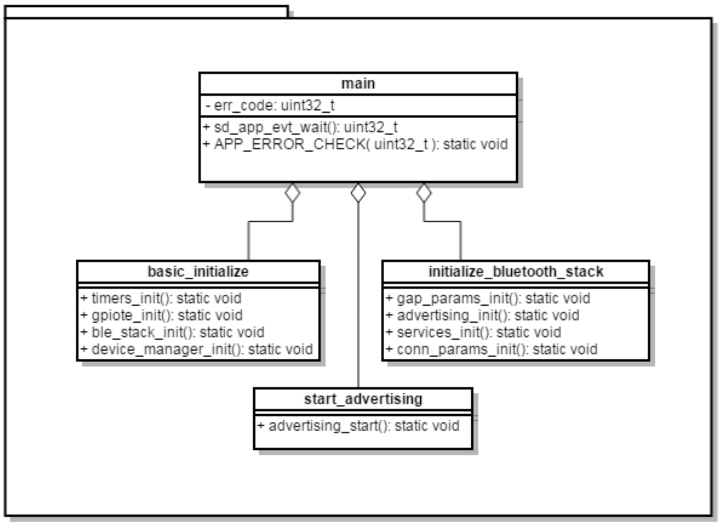

Figure 4.2: Class diagram of Sensing Device

In the following subclass, initialize_bluetooth_stack, are defined all the communication parameters and services necessary for the functioning of a BLE communication. This is constituted by the function (1) gap_params_init(), (2) advertising_init(), (3) services_init() and (4) conn_params_init().

The gap_params_init function sets up all the necessary GAP (Generic Access Profile) parameters of the device including the device name, appearance, and the preferred connection parameters.

In the case of the advertising_init function this encodes the required advertising data and passes it to the stack. As explained in subsection 3.1.1, an advertising packet transports different types of information, such as for example, the RSSI level, the MAC-Address and the name and information of the type of services that the device supports. For this it is necessary to make proper configuration of parameters of the advertinsing packet. Being the most important the attribution of the UUID of the service and the inclusion of the device name in the packet. Another aspect in the configuration of an advertising packet is the temporal component and the direction of the communication. In this case, it was defined an advertising packet with a timout value and a sending period with the possibility of, in the communication, any device send scan requests and connect requests.

As for services_init function, this initialize the service that will be used by the application. As explained in subsection 3.1.1, each service is identified by an UUID, which in turn can be equal to the value of the standard services defined by the Bluetooth SIG organization. In the case of our application, it is used a known UUID, more specifically the Heart Monitor Rate service. Before the definition of the current UUID of the application, was defined a value of identification of the non-standard service, but this was not recognized when the Gathering Device was performing the discovery process of the GATT server services of the sensing device (subsection 4.2.2). In order to overcome this problem it was decided to use an UUID known by the BLE stack, thus ensuring the communication between the two devices. To the UUID of the service is associated the service module, where this is constituted by different features and service functions. As illustrated in class diagram of the figure 4.3, the service consists of three structures and three subclasses.

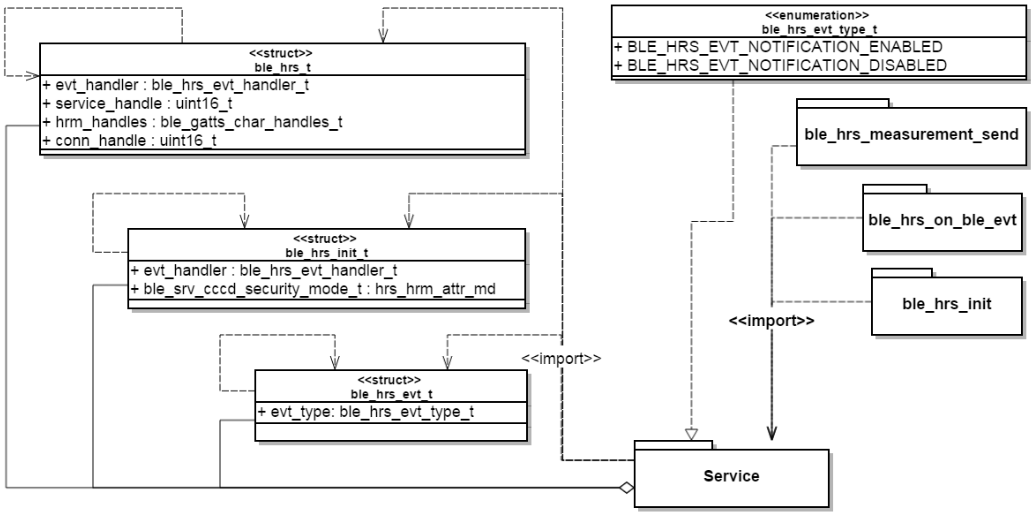

Figure 4.3: Class diagram of Service Module of Sensing Device

The respective structures have as name (1) ble_hrs_t,(2) ble_hrs_init_t e (3) ble_hrs_evt_t, and the three subclasses have the names (1) ble_hrs_init, (2) ble_hrs_measurement_send e (3) ble_hrs_on_ble_evt.

The ble_hrs_t structure contains various status information for the service. This structure will have to be supplied by the application. It will be initialized by this function, and will later be used to identify this particular service instance. The different status of information are:

- evt_handler : Event handler to be called for handling events in the Service.
- service_handle : Handle of Service (as provided by the BLE stack).
- hrm_handles : Handles related to the Measurement characteristic.
- conn_handle : Handle of the current connection (as provided by the BLE stack, is BLE_CONN_HANDLE_INVALID if not in a connection).

The ble_hrs_init_t structure contains all options and data needed for initialization of the service, where the objective of each parameter consists of the following:

- evt_handler: Event handler to be called for handling events in the Service.
- hrs_hrm_attr_md : Initial security level for heart rate service measurement attribute

Lastly, the ble_hrs_evt_t structure has as information the type of service event:

- BLE_HRS_EVT_NOTIFICATION_ENABLED : measurement value notification enabled event.

- BLE_HRS_EVT_NOTIFICATION_DISABLED : measurement value notification disabled event.

Regarding to the functions, the ble_hrs_init function is responsible for the initialization of the Service. More precisely the attribution of the UUID, of the characteristic and of the values of the ble_hrs_init_t structure to the ble_hrs_t structure.

The ble_hrs_measurement_send function sends the value of all sensors if connected and notifying. To this function is passed as parameter an array of the uint16_t type, that contains the read values of the different sensors of the system. Already inside the function this array is converted into an array of the uint8_t type. The uint16_t type information is divided into two bytes, where the most significant byte always remains on the left of the least significant byte in the new array. After this conversion, is then updated the attribute of the characteristic with the data stored in the new array type.

The ble_hrs_on_ble_evt function is used for handling the Application's BLE Stack events. This function handles all events from the BLE stack of interest to the Service. In this case our service handles three types of events:

- BLE_GAP_EVT_CONNECTED : indicates that the device is connected.
- BLE_GAP_EVT_DISCONNECTED : indicates that the device is disconnected.
- BLE_GATTS_EVT_WRITE : when is wanted the writing on the characteristic of the service.

Although the UUID is of a known service, the service module was developed specifically for the sensing device application. This results in a different structure than the typical Heart Rate Monitor Service standard, of the Bluetooh SIG.

Returning to the initialize_bluetooth_stack subclass, the conn_params_init function aims to initialize the Connection Parameters Negotiation module of BLE. In this function are made the negotiation attributes, such as the number of attempts before giving up the negotiation, the time between each call to sd_ble_gap_conn_param_update after the first time, the time from initiating event (connect or start of notification) to first time sd_ble_gap_conn_param_update is called and the pointer to the connection parameters desired by the application.

Regarding to the start_advertising subclass this contains a function whose name is advertising_start. This function begins the sending of the advertising packet according to the parameters defined in the advertising_init function.

4.2.2 Gathering Device

As described in section 3.2, the Gathering device has two functioning modes, where each of them corresponds to the operating rules of the BLE technology, central/observer and peripheral/broadcaster. The alteration of rules in real-time is due to the fact that the Softdevice S130 (subsection 3.1.2) consist of a pre-compilation of the two functioning modes of the BLE stack.

Only this way the alternation between the two rules is guaranteed. Each operating rule of the Gathering Device is constituted by distinct processes. In case the Gathering Device work as Central, it performs four processes: (1) listening to (scan) Advertising Packets, (2) verify if the sender of the Advertising packet is valid, (3) connection to the sender of the Advertising Packet in the case of this being a Sensing Device and (4) disconnect after the reception of data from a connection with a Sensing Device. While in the peripheral rule, the Gathering Device has a single process: (1) Send the data stored in an Advertising Packet.

As illustrated in the flowchart of figure Figure 4.4, these two operating rules and the respective processes are executed by the functions (1) do_work_central and (2) do_work_peripheral in an infinite loop. This way, it is decreased the probability of the loss of data. In comparison with the sensing device, the gathering device does not have the process of sleep mode, in order for this to be always listening for advertising packages.

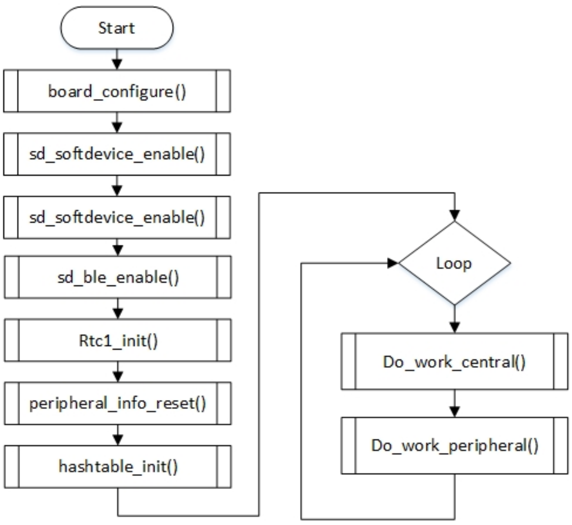

Figure 4.4: Flowchart of Gathering Device

The remaining functions in the figure Figure 4.4, (3) board_configure, (4) sd_softdevice_enable, (5) sd_ble_enable, (6) rtc1_init, (7) peripheral_info_reset and (8) hashtable_init aim the following:

- board_configure: hardware configuration as LEDs and the boot parameters of the UART for purposes of Debug of the Gathering Device functioning;
- sd_softdevice_enable: initialization of the softdevice;
- sd_ble_enable: initialization of the BLE stack functioning;
- rtc1_init: initialization of the RTC1 timer of the nrF51822 to be used by the softdevice;
- peripheral_info_reset: reset of the connection handle of the peripheral;
- hashtable_init: allocation and initialization to 0 of a hashtable with twenty rows and four columns.

After this initialization, the Gathering device is able to work and alternate between the two operating rules, (1) central and (2) peripheral.

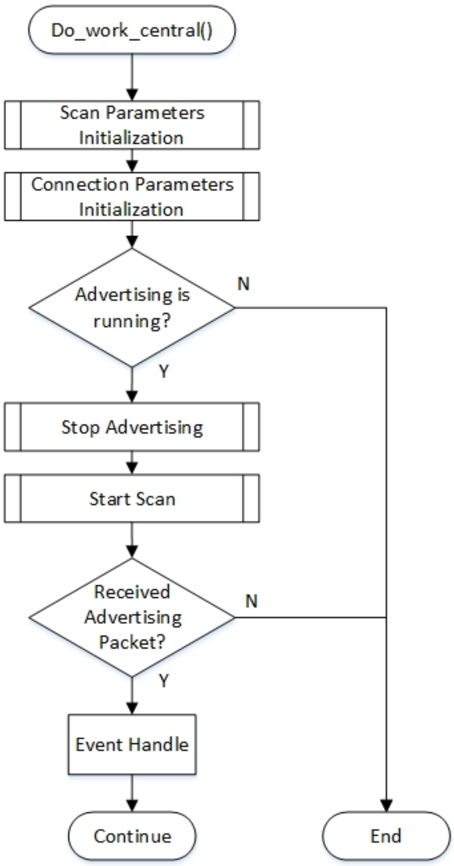

Figure 4.5: Flowchart of Do_work_central funtion

As illustrated in the flowchart of the Figure 4.5, when it works as central, this initiates the scan parameters and connection of the communication. The scan parameters are intended to declare the duration and the period in which is made the advertising packet scan. In turn, the connection parameters are intended to determine the minimum and maximum duration of a communication between a peripheral and a central. After this initialization, the Gathering device is able to work in central rule.

After this, is verified if the Gathering device is sending some advertising packet, i.e., if it is working as a peripheral. In case of confirmation of this condition, is executed the sd_ble_gap_adv_stop function, which aims to end the advertisement. With the end of the functioning as peripheral is possible to initiate the functioning of the Gathering device as central. As the scan and connection parameters have already been initialized it is possible, with the sd_ble_gap_scan_start function, to initiate the scan of the advertising packets from peripheral devices, i.e., sensing devices. After the beginning of the scan is expected a BLE event of the BLE_GAP_EVT_ADV_REPORT type during a period of time, timeout. In case this event is received, will be read the PDU content (section 3.3) of the advertising packet and the number of bytes that constitute it. With this data, will be analyzed if the advertising packet is originating from a (1) sensing device, (2) gathering device or (3) another device.

In the first analysis of the advertising packet is verified if the bytes contained in the PDU field correspond to the bytes of the name "SENSOR", in case is confirmed this condition is activated a flag, g_entity, used in the do_work_central function (Figure 4.6), otherwise is verified if it is originating from a gathering device. This verification is made based on the number of bytes of the PDU field and in the verification of the byte 4(Figure 4.8) of the PDU field.

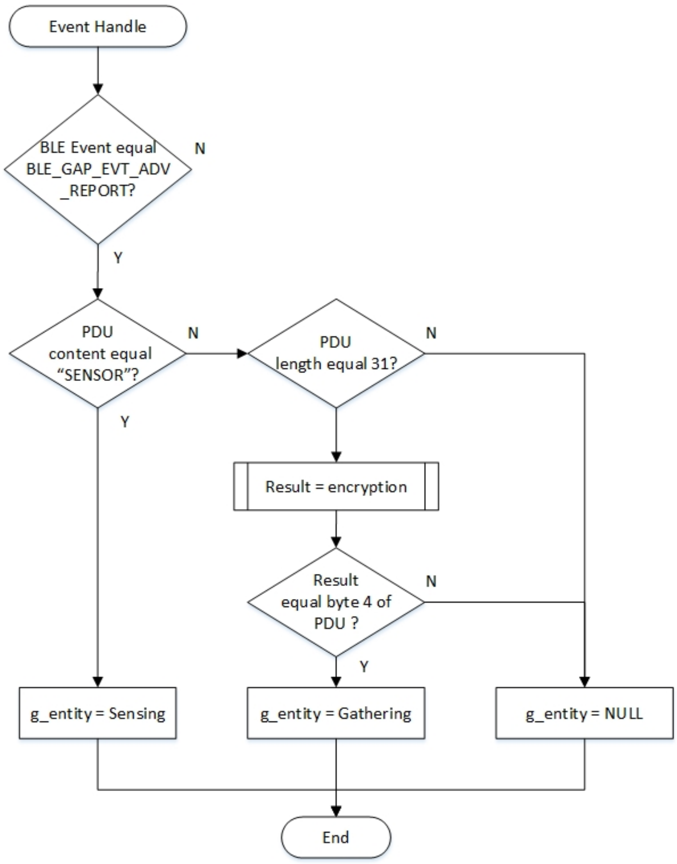

Figure 4.6: Flowchart of event_handle function

In this case, a message which the **sender is a Gathering device**, the PDU field should have a number of bytes equal to 31 and the byte 4 should be equal to the result of the encryption function developed in the context of this thesis. This function arises from the need to authenticate if the message is originating from a Gathering device or not. As illustrated in the flowchart of the Figure 4.7, this uses as parameters the two least significant bytes of the MAC-Address of the sensing and gathering devices. The execution of this function consists of *xor* operations with these four bytes and a predefined key in which the result is stored or compared to the byte 4 of the advertising packet of a Gathering Device.

The encrypt function is used in two contexts:

- In the first case, is when there is reception of an advertising packet from a Gathering device.

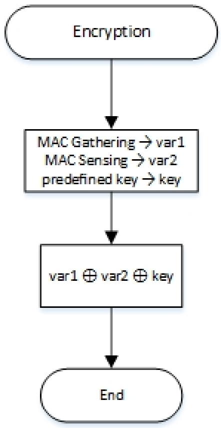

Figure 4.7: Flowchart of encrypt function

The objective is to compare the result of this with the byte 4 originating from the advertising packet.

- The second case, is when there is the packaging of the advertising packet of the Gathering device when this works as peripheral.

In case is verified that the **sender of the message is a Gathering device** is disabled the g_entity flag.

The state of the g_entity flag determines the type of procedure that the Gathering device will follow. If the value of this flag corresponds to the logical value that identifies that the message is **originating from a sensing device**, the connection process is initiated.The connect function is responsible for the beginning of connection of the gathering device with the sensing device, sender of the advertising message.

Within the connect function, this process is done using the sd_ble_gap_connect function, in what is passed as a parameter the MAC-Address of the sender of the advertising packet and the connection parameters previously defined at the beginning of the do_work_central function.

In case the connection is successful, the event_handle function processes the event of the BLE_GAP_EVT_CONNECTED type. In this event is made known that the Gathering device is paired.

After the pairing between Sensing and Gathering devices is started the discovery function.As mentioned in subsection 3.1.1, the sensing device contains a GATT server that is constituted by a service with characteristics. In order to make the reading of these server, is necessary to know the descriptor of the service and of each feature. This process is done with updating of various parameters existing in the softdevice and in the BLE stack.

With the pairing and the discovery of the service concluded, the Gathering device waits during a certain time the reading data of the sensing device. When the Gathering device receives the data, it stores in a buffer and disconnects from the sensing device to ensure the minimum possible time of communication between these two elements. This action, as described in subsection 4.2.1 and section 3.2, aims to minimize the energy expenditure in communications of the sensing device.

With the data stored is activated a flag whose purpose is to indicate that the received information is new. When the Gathering device functions as peripheral this flag distinguishes if the Gathering device has something new to send or not. With this control is avoided the successive resend of the same information, which could cause the congestion of the wireless network, as it is also optimized the energy expenditure, due to the non-transmission of repeated data.

Returning to the g_entity flag, in case this has the logical value that indicates that the origin of the message is a gathering device, is executed a function in do_work_peripheral, whose name is sink_handle.

This function arises from the need to control duplication of messages originating from the sending of packets in broadcast of a Gathering device. As explained previously, when the gathering device contains a new information from a sensing device, this sends to the network in broadcast. In turn, the neighboring gathering devices will receive this message and send in broadcast. In case these two neighboring gathering devices contain the same information, they will send the same data to each other in an infinite loop. In order to combat this resend of data in loop between two Gathering devices is used the function of message duplication control sink_handle.

The first condition to check is if the received message is not a reflection of the network, i.e., if the Mac-address address, in the PDU field (Figure 3.11, Figure 4.8), matches the address of the Gathering device itself. This is done from a get_mac_address function that returns the two least significant bytes of the device's Mac-address. The second condition is to check if the value of the TTL, byte 4 of the PDU field, is not equal to zero.

In case it is confirmed that the message is not a reflection of the network is passed to the next stage. At this stage, with the bytes of the Mac-address of the Gathering device and the message counter, CNT, originating from the advertising packet is formed a key.

The intention of integrating the CNT byte in the formation of the key is due to the fact that this variable in conjunction with the value of the Mac-address generate unique keys. Since each time that a Gathering device sends a new message with data of the sensing device, is sent in the message the increment of one unit of the CNT.

This key is the index of position in a hashtable. A hashtable consists of a quick way to allocate and return data stored in a given position. The index value of the position has as origin a hash of a certain value/key. In comparison with a vector, the hashtable, does not need to go through all the positions until it finds the data that is intended. In this case, simply insert a key and this is converted into a hashtable index. The use of this solution is due to the rapidity that the data are allocated or returned. As the gathering device is a device that works between the two rules of the BLE, it is necessary to ensure that it moves between these as quickly as possible. For this reason, the application of a hashtable to store messages duplication control data demonstrates to be an added value. Under the scope of this thesis, for the application of the hashtable was used an already developed and validated library. In this contains 20 positions with a 4 byte key, where the last byte consists of '\0' to form a string. In each memory position of the hashtable are saved the remaining bytes of the advertising packet, starting from the byte 5.

That said, is verified if the data of a position associated to a key are null. If so, this means that the information originating from the advertising packet is a new information to be stored. For this it is necessary to verify if the table is already full or not. This control is done through a variable called cnt_table. At this is incremented one unit for each new data entry. In case the cnt_table is not equal to 20, value of the hashtable size, the data are inserted into the hashtable and the value of the key is stored in an array whose name is table. This array is used to create the logic of a FIFO waiting queue. Thus when the cnt_table is equal to 20, which indicates that the table is already full, the control of knowing which is the oldest information is the responsibility of the array table. That is, with the return of the key stored in the last memory position of the array table, the data stored in the hashtable are eliminated, releasing a position for a new entry. Meanwhile, in the array table, an operation is performed in which there is a shift of one unit of the memory positions of the array according to the FIFO logic.

With the data entry made, is copied the PDU content of the advertising packet received to an array of bytes, re_send, where the byte 4 of this new array is decremented by one unit. Are also activated two flags, flag_type_data and its_new. The its_new, previously explained, serves to provide an indication that the information is new to the gathering device. While the flag_type_data indicates that the received message is to resend. This information is useful in the operating mode of the gathering device as peripheral.

This mode is executed by the do_work_peripheral functiononly begins the process of advertisement in case this is not already in progress and in case there is a new information to send based on the logical value of the its_new flag.

If these two conditions are met is executed the advertise function, having as input parameter the flag_type_data flag. The status of this flag indicates if the message to send is a message to resend, i.e., a message whose origin is of another gathering device and that have to be diffused. Otherwise, the message to send is a new message in the system with the information about the state of the sensing device sensors. So for sending these two types of message is required the construction of the PDU field of the advertising packet.

This field contains a maximum limit of 31 bytes available for the insertion of data in a free form. In the case of sensing device, this form is used to send the device name, "SENSOR", while in the case of the Gathering device, as illustrated in figure 4.8, is used to send useful data of the system, such as the (1) total number of bytes of the PDU field to read, (2) MAC-Address of the Gathering device, (3) CNT, (4) TTL, (5) encrypted value, (6) MAC-Address of the Sensing device, (7) pH value, (8) eC value, (9) air temperature, (10) water temperature, (11) atmospheric pressure value and (12) value of humidity.

This composition is organized as follows:

- Number of bytes : byte with the total number of bytes of the PDU field to read, in this case is the value 31.

- MAC-Gathering : two bytes where are stored the two least significant bytes of the MAC-Address of the Gathering device. The proposal to associate the Mac-address of the gathering

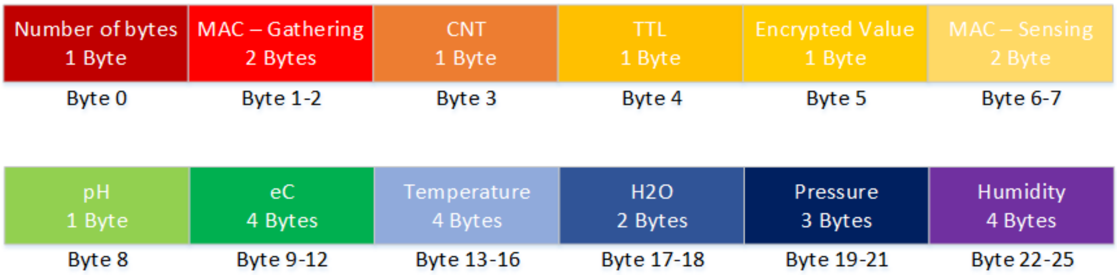

Figure 4.8: Advertising Packet with useful data from Gathering device

device to the data of the sensing device, allows to know precisely where the data come from. That is, with the Mac-address I know which is the greenhouse where I should search for the sensing device.

- CNT : message counter. This counter consists of a byte in which is incremented one unit each time the gathering device sends a new information, i.e., when there is data originated from a Gathering-Sensing devices communication. With this counter it is possible to rapidly control the duplication of messages.

- TTL : byte to which is decremented one unit each time a message is sent back from Gathering to Gathering device. When this reaches zero, the message is eliminated. This avoids that a message travels, eternally, on the network.

- Encrypted value : byte whose value is the result of an encryption operation of xor operations between a key and the two bytes of the MAC-Address of the Gathering and Sensing devices.

- MAC-Sensing : two bytes in which are stored the two least significant bytes of the MAC-Address of the Sensing device.

- pH : byte that stores the pH value read by the Sensing device.

- eC : four bytes that store the eC value read by the Sensing device.

- Temperature : four bytes that store the atmospheric temperature value read by Sensing device.

- H2O : two bytes that store the value of the water temperature read by the Sensing device.

- Pressure : three bytes that store the atmospheric pressure value read by the Sensing device.

- Humidity : four bytes that store the humidity value read by the Sensing device.

So in the building of the advertise packet, in case the message being a new message to send, it is made packaging of: the increment of the CNT, the MAC-Address of the Gathering device (only two bytes), the initialization of the TTL byte and the data originating from the sensing device. In the case of the resend of a message, as the TTL byte has already been decremented in sink_handle this does not need to perform a packaging.

That said, simply execute the send of the advertising packet, a process already described in subsection 4.2.1.

4.3 Software

This subchapter explains the code structure of the API developed in Java for the app to be installed in the Cloud Device.

4.3.1 Cloud Device

As mentioned in section 3.2, the system is constituted by a device whose purpose is to be a gateway between the BLE technology and Wi-Fi/GPRS. The best way to implement this gateway , as described in section 4.1, is using a smartphone on which an application responsible for forwarding data derived from the BLE modem to the Wi-Fi/GPRS modem is running.

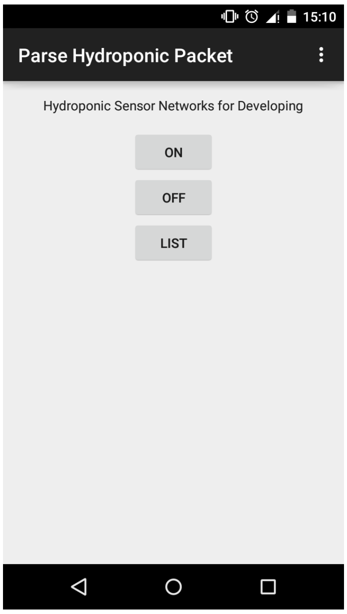

Figure 4.9: App Panel Control

Thus arises a need to develop an API that deals with the functioning status of the BLE modem and with the data obtained from the BLE communications. This results in an API with three functions (1) EnableBleModem(), (2) ParseHydroponicPacket(ScanResult result) and (3) DisableBleModem():

- EnableBleModem(): As indicated by the function name, this is responsible for connecting the BLE modem incorporated in the smartphone.

- ParseHydroponicPacket(ScanResult result): The function main objective is to realize the parse of the data packet obtained from the gathering device, where this returns the data sent by the BLE communication.
- EnableBleModem(): This function is responsible for disconnecting the BLE modem incorporated in the smartphone.

In order to exemplify the use of the projected API was developed an application in JAVA for Android that demonstrates the functioning of the different functions with the aid of Debug messages in the display.

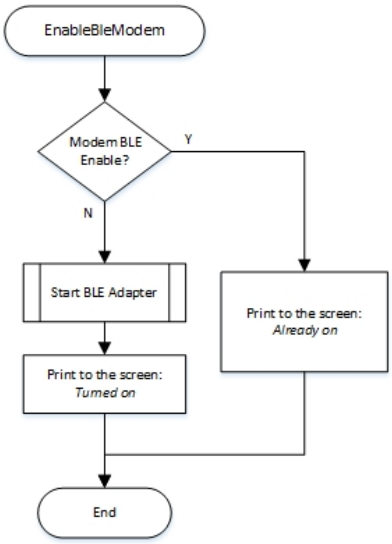

Figure 4.10: Flowchart of EnableBleModem Function

The developed application runs on a single Activity [43], and contains three buttons (1) ON, (2) OFF and (3) LIST, as illustrated in the Figure 4.9.

The **first button y**invokes the EnableBleModem function of the API. As illustrated in the flowchart of the Figure 4.10, in case the BLE modem is not active, it is turned on and a notification is printed to the display reporting the On state of the modem. Otherwise is reported that the BLE modem is already functioning.

The **second button** gives order to the execution of the DisableBleModem function, which consists of turn off the BLE modem and report this state to the display.

The **third button** initiates the receipt of BLE packages. When the smartphone receives a BLE package this is processed by the ParseHydroponicPacket function. Inside the function, as illustrated in the Figure 4.11, is verified if the message is comprised of 31 bytes. In case of confirmation, the next step is to validate if the received message is from a Gathering Device.

As mentioned in subsection 4.2.2, it is used an encryption function whose aim is to create a form of differentiation and authentication of the message origin.

Applying this function in the developed application, wherein are passed as parameters the bytes encapsulated in the message that store the addresses of the devices subsection 4.2.2, the

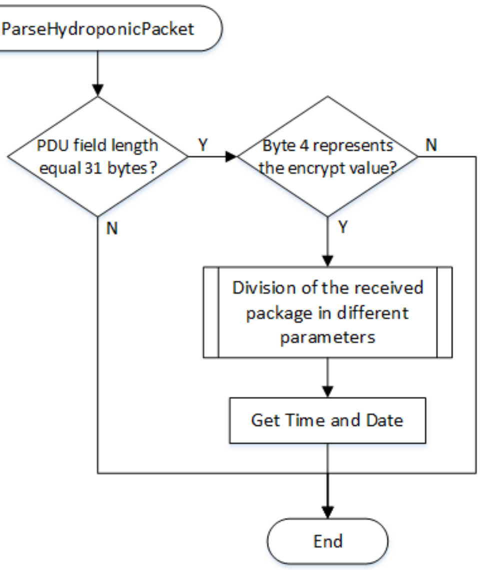

Figure 4.11: Flowchart of ParseHydroponicPacket function

result of this should be the same that is stored in the byte 4. In case of confirmation of this condition, the received package is divided in different parameters that compose it, (1) message counter, (2) pH, (3) eC (4) water temperature, (5) temperature and (6) humidity of the air (7) atmospheric pressure, (8)(9) MAC Address of the sensing and gathering devices. To these data are also added the (10) time and the (11) date of receipt on which the smartphone has received the data. The objective of associating a date and time to the data received from an Advertising packet consists in differentiate whether a package is or not duplicated as well as the validity of the information.

As explained in section 3.2 and subsection 4.2.2, the data of each sensor are sent, in broadcast, from Gathering device to Gathering device until reaching the Cloud device. And sometimes, sending messages in broadcast can lead to duplication of data, therefore, the creation of control strategies and elimination of duplicate messages is needed. One way of doing this is, as explained in subsection 4.2.2, through a TTL byte. The other way is the analysis of the temporal difference, so, in the Cloud device, by associating the date and time of receipt of a message is possible to check with pre-defined time basis if the message is or not duplicated. This proposal of control safeguards the situation when there is overflow of the counter of the message sent by the Gathering device. Without this, when the cloud device received a message and checked that the data received were the same that it had already stored, it does not knew if the information received was the most recent sensor state, if there was overflow of the counter of the message, or if it was a replication of the stored information. So with the storage of the information with the date and time of receipt is possible to verify if the information received is or not a replication through the observation of the temporal difference. This temporal difference is defined by the size of the network.

4.4 Summary

Firstly, the hardware was presented and was explained the reason of its choice for the composition of each system element. Then, was explained the firmware developed for the Sensing and Gathering devices. In this explanation is described how the code is organized, the implementation of a service in the sensing device, the implementation of the framework adopted in the communications in broadcast of the Gathering device and the message duplication control systems.

Finally, it is explained the API developed for android that performs the parse of the frames originating from the Gathering devices.

Chapter 5

Testing and Analysis

In this chapter, is initially explained, in section 5.1, the process and the debug tools of the communication protocol developed. Then, in section 5.2, is explained the process of reading the energy consumption of the sensing and gathering devices. These readings are used in the choice of the battery that is most suitable for the different devices and in the projection of a Solar Photovoltaic System to integrate the Gathering Device. This choice will be described in subsection 5.3.2.

5.1 Testing the Communication Protocol

In this subchapter is explained the process performed to validate the implemented communication protocol. For this were used two methods:

- The first consists in the use of the USART port existing in the development boards. With a computer connected to these ports, is made a sniff of the various messages, wherein this contain indication of the state of the different phases of the device functioning. This method was often used to validate the functioning of the Gathering and Sensing devices, the hashtable and the pairing between Gathering and Sensing devices.

- The second method consists in the use of an Application (app) developed and provided by NORDIC SEMICONDUCTOR to the operating system Android 4.4, whose name is "*nRF Master Control Panel (BLE)*". In the context of this thesis, this app is used to validate the functioning of the different phases of communication of the sensing device and gathering device. This means that, with the app, it is possible to monitor the name "SENSOR" that travels in the advertising packet, make a connection, validate the reading of the GATT Server and disconnect from the sensing device.In turn, in the case of the Gathering device, by using the app is possible to validate the sending of the advertising packet with the data from a sensing device communication. In order to prove the functioning of the resend of advertising packet between gathering devices, have been used two gathering devices and a smartphone with the app "*nRF Master Control Panel (BLE)*." Thus, it was possible to verify the decrement of one unit in the TTL Byte and that there was no change of the packet format from the Gathering device that sent it.

5.2 Energy consumption

One of the objectives of this thesis is the development of devices with autonomy, that is, independents of the distribution network of electric energy. A way to achieve this objective is the coupling of a Lithium battery to the device. This requires proper conditioning of the battery capacity, wherein this process is done by reading the circuit current consumption.

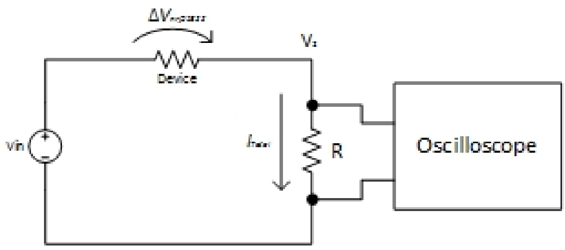

Figure 5.1: Scheme to current measurement

As illustrated in the figure 5.1, one of the ways to measure the consumptions of current is to install a resistance (R) in series after the circuit with a tip of oscilloscope connected to its terminals. In this case, it is ensured that the circuit to be measured and the measuring device share the same Ground.

As a resistance is a current limiter is necessary to correctly dimension its value. This dimensioning is based on the maximum value of current that can pass through the circuit, in the case of the sensing device the maximum value expected of the current is approximately 20 mA. This value comes from the consultation of the datasheet [44, 45] of the sensors installed on the sensing device, being this the value of current necessary for the operation of each sensor.

Based on this forecast, the calculating of the resistance to be installed is done in the following way:

$$
\begin{aligned}
V_2 &= V_{in} - \Delta V_{nrf51822} \\
&= 0.20V \\
V_2 &= I_{Total} \times R \\
R &= \frac{V_2}{I_{Total}} \\
R &= 10\Omega
\end{aligned}
$$

Getting a resistance (R) value of 10 Ω.

5.3 Experimental Results

After being made the explanation of the technique used for the measurement of the energy consumption (section 5.2) of a device is necessary to develop a calculation method of the average value of the energy consumption read.

The first phase of this method is immigrate of the analog world to the digital world. This process is made based on the use of a Python library and a characteristic of the oscilloscope used in this experiment. This characteristic is the Virtual Instrument Software Architecture protocol (VISA). This protocol allows communication between a PC and an oscilloscope using a script. In this case, the script developed was in Python, where is used the PyVisa library. The basic principle of this script consists in the request of 2500 samples to the oscilloscope, the return of this is saved in a file of the comma-separated-values (csv) type. The second phase of the method is the calculation of the average value of current based on the csv type file generated. For that, was used the MatLab software, wherein was developed a script that from the data stored of the csv file generated calculates the average value of the current consumption, the maximum and minimum current peak and the duration of the event. After that, with the application of the technique described in section 5.2

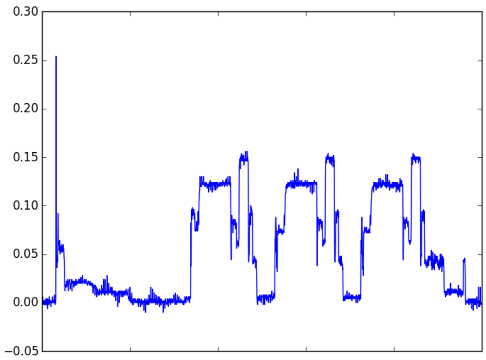

Figure 5.2: Wave form of Advertise Process

and the calculation method of the current consumption explained is possible the dimensioning of the battery of each device.

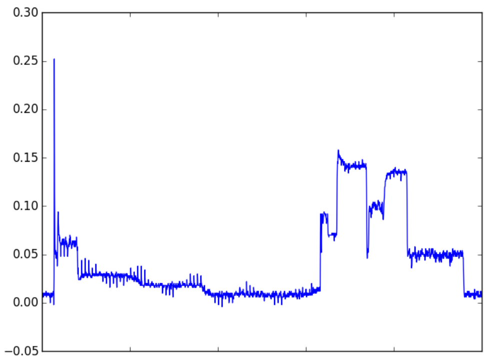

Figure 5.3: Wave form of Connect Process

5.3.1 Sensing Device

This section presents the results of the energy consumption of the different phases of communication and tasks that a sensing device is subject (1) Advertising, (2) Connect and (3) Measure.

This experience has two phases, the first consists in measuring the average consumption of the different stages of communication. The purpose of these readings is to compare them with the manufacturer readings, in order to check the wave form during a communication of the sensing device. In the second phase, is presented the average consumption of the sensing device with all the tasks proposed for the sensing device. Since the sensing device consists of a PCA20008 development kit and it was not been integrated the sensors as described in section 4.1, it was not measured the measuring task. But through the information of the energy consumption and the time of duration of the measure task is possible to simulate the consumption. This information arises from the board developed by Fraunhofer, which contains the incorporated sensors.

Table 5.1: Average values of the power consumption of each task of the Sensing device without measure task.

Task	Interval (ms)	Average Value(mA)
Advertising	4.998	5.453
Connect	2.499	3.469
Sleep	4.998	1.075

Thus, will be presented two values for the device capacity, the first with the expense associated to the measure task and the other without the energy expenditure of the measure task.

Table 5.2: Average values of the power consumption of each task of the Sensing device with measure task.

Task	Interval (ms)	Average Value(mA)
Advertising	4.998	5.453
Connect	2.499	3.469
Measure	3000	20.000
Sleep	4.998	1.075

In a first approach was made the analysis of the energy expenditure associated to the sending of an advertise packet. Through the mounting described in section 5.2 is obtained the wave form represented in Figure 5.2.

By means of an integral calculation of the wave area, this results in the current average value of 5.453 mA.

Table 5.3: Average value of current cunsumption of sensing device task

Capacity (mAh)	Hours	Days
335	294	12
2070	1819	75
5000	4394	183

The next step was the analysis of the connect process, following the same methodology of measurement of the advertise, was obtained the wave form represented in Figure 5.3.

Again, with an integral calculation of the wave area, the result of the current average value is 3.469 mA.

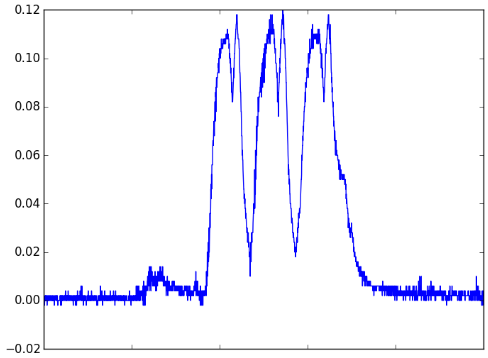

Figure 5.4: Wave form of Advertise Process

In the second phase of the experiment, is determined the total expenditure of the circuit. For this, with the current averages obtained during the first phase of this experiment and with the temporal study of a communication cycle, i.e., perform the advertise, the connect and the sleep mode, is obtained 1.075 mA as result of the circuit total expense.

This result arises from the weighted average of the different tasks that the sensing device performs. As enumerated in the Table 5.1, during approximately 12ms the sensing device performs all the phases of communication. In the remaining 15 minutes, it enters in sleep mode.

In turn, with the measure task, the readjustment of the weighted average is required. Based on the values presented in the Table 5.2, the result of the circuit total expenditure is 1.138 mA.

The inclusion of the measure task results in a 5.87% increasing of the average energy expenditure of the sensing device. Based on this value, we can conclude that the inclusion of the measure task is not very significant in the overall consumption of the sensing device.

With this, is possible to perform the dimensioning of the battery to integrate in the sensing device. The following Table 5.3 presents, for different battery capacities, its duration in hours and days in accordance with the average expenditure of the sensing device:

5.3.2 Gathering Device

This experience follows the same measurement method described in subsection 5.3.1. In the first phase, is measured the average consumption of the two communication phases to which the Gathering device is subject, the phases (1) advertise and (2) connect. In the second phase, is presented the average consumption of the gathering and the dimensioning of the photovoltaic panel to ensure the capacity of the gathering device.

Table 5.4: Average values of the power consumption of each task of the Gathering device.

Task	Interval (ms)	Average Value(mA)
Advertising	4.998	9.400
Connect	2.499	2.300
Sleep	2.592	57.540

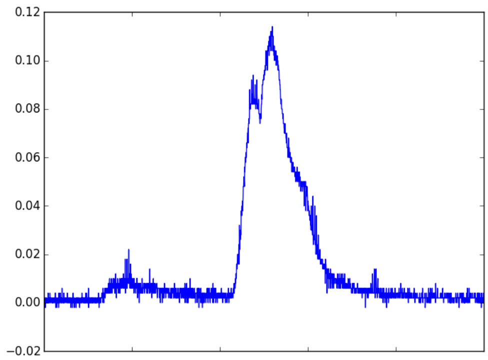

Figure 5.5: Wave form of Connect Process

By means of an integral calculation of the wave area of the Figure 5.4, the result of the current average value, of an event of advertise, is 9.4 mA.

Table 5.5: Average value of current cunsumption of Gathering device task

Capacity (mAh)	Hours	Days
335	16	-
2070	103	4
5000	249	10

Following the same measuring methodology of the average value of the advertise event current, the average value of the connect event Figure 5.5 is 2.3 mA.

With this, it is possible to determine the total expenditure of the circuit. Following the same analysis made in subsection 5.3.1, with the weighted average of the Table 5.4 values, the result of the average value of the circuit current is 20.0091763307 mA.

With this average value, it is possible to dimension the battery to use in the Gathering device. The Table 5.5 represents, to different batteries capacities, the capacity in hours and days based on the average expenditure of the Gathering device current.

Based on what was described in section 4.1 this device does not require an autonomy limit, since the battery charging is done with a photovoltaic panel. So the choice of battery must be based on the maximum period of time where there is no sun. Considering the worst case, the absence of solar light during 14 hours, which value represents the longest night of the year in the Southern Hemisphere, the battery 3.7 V with a capacity of 335 mAh is sufficient to maintain the capacity of the Gathering device during this period.

The choice of the solar panel is made by the generating capacity in Ah, so as the consumption of the Gathering device is 20.009 mAh, it should be chosen a photovoltaic panel with a power of production of at least 77.7mWh.

So a photovoltaic panel with a generating capacity of 800mWh, panel which is widely available for sale in the market, is more than sufficient to ensure the battery charging of the Gathering device.

In relation to the remaining components that integrate a photovoltaic system, these will not be explained, since they are not a theme of this thesis.

5.4 Summary

In the first place, the technique used to measure the current of each device was presented. Followed by a presentation of the energy expenses of each communication phase and task, as well as the average expenditure of the current of the sensing and gathering devices. This average expense allows the following explanation, the dimensioning of the batteries to integrate in each device.

Ultimately, it is explained the dimensioning of the photovoltaic panel to integrate in the gathering device.

Chapter 6

Conclusions and Future Work

6.1 Satisfaction of Objectives

Based on all that has been presented in this thesis it is necessary to evaluate the fulfillment of requirements and objectives described in section 1.2 and in section 1.3.

One of the first requirements to be fulfilled was related to the creation of a **low energy consumption system**. For this, we opted for the choice of **BLE technology**. Based on the specifications described section 2.7, this technology proved to be the one with the best performance for the theme of this thesis, the creation of a system of wireless sensors, of low power and low cost. In addition to the performance, the choice of this technology is also reflected in the maximum range requirement of 54 meters section 1.2, in this case, the BLE technology has a **maximum range of 100 meters**.

With the choice of technology is possible to define the network structure. In order to fulfill the requirement of creating an **easy installation and maintenance** system is proposed in section 3.2 a **network structure with three layers**, (1) Sensing Layer, (2) Gathering Layer and (3) Cloud Layer, that communicate with each other with specific communication protocols. This conjuncture is more detailed in section 4.1 and section 4.2, where it is revealed that in fact the system is easy to install, since the sensing device consists of a mobile system, powered by a battery, the gathering device consists of a simple board powered by a photovoltaic panel and installed on top of a greenhouse and the Cloud device consists of a smartphone with BLE interface.

That said, in relation to the last requirement described in section 1.2, at a minimum, there must be two measurements per zone, i.e., at a minimum two sensing devices must communicate with a Gathering device. Unfortunately, due to reasons of time, was not possible to perform the test with multiple sensing devices to a single Gathering device. However, the communication between a sensing device and a Gathering device was proved as proposed in section 3.3.

In conclusion, the developed prototype system complies with all of the requirements imposed at the beginning of this thesis, but also proves the concept that it is possible to implement a network with mesh typology based on the use of BLE technology.

6.2 Future Work

As previously mentioned, the developed system is just a prototype, to become a viable and robust system is necessary to make improvements. These improvements can go through the integration of new software blocks or hardware components.

So, as future work, it is necessary to integrate the sensing device firmware on the hardware developed by Fraunhofer Portugal. So far the Fraunhofer Portugal has developed all the sensing device hardware component, as described in subsection 4.1.1.

After this integration, it is proposed as future work to ascertain the functioning of the network. For this it is necessary to install different sets of sensing devices for distinct gathering devices and a Cloud device within range of at least one Gathering device.

With this mounting of the network prototype I propose the analysis of the functioning and performance of the network, i.e., if all communications of the Gathering and Sensing devices are made, if there is packet losses, how long it takes the message to reach the cloud device to the number of nodes existing on the network, study the relation of the initialization number of the TTL byte according to the number of nodes existing on the network as described in subsection 4.2.2 and the analysis of the energy consumption of each device.

After validation of the network functioning, is required the construction of the Gathering device Hardware component as described in subsection 4.1.2, so, as future work is proposed the development of hardware and the integration of firmware in this device. Lastly, for the Cloud device, is necessary to create an app that uses the API developed in this thesis to obtain the data originating from the Gathering/Sensing devices communication to send them by GPRS to a Cloud service.

After the fulfillment of these three points of improvement of the system and validation and study of the network behavior, is necessary to validate again the functioning of this, for that, with the new hardware and the firmware improvement of each device, is proposed to mount the network architecture and to perform the tests realized in the first evaluation. This way, is possible to compare if the new firmware and hardware have better performance in relation to their prototype versions.

References

[1] Health Smart Living. Popular hydroponic systems. http://healthsmartliving.com/hydroponics/popular-hydroponic-systems/.

[2] Bluetooth Developer Portal. URL: https://developer.bluetooth.org/DevelopmentResources/Pages/Bluetooth-Smart-Optimizations.aspx.

[3] Nordic Semiconductor. S110 nRF51 Bluetooth® low energy Peripheral SoftDevice SoftDevice Specification v2.0. 2014.

[4] Phill Smith. Comparisons between Low Power Wireless Technologies. 2011.

[5] Piero Conforti. *Looking ahead in worLd food and agricuLture: Perspectives to 2050*. 2011. URL: http://www.fao.org/docrep/014/i2280e/i2280e.pdf.

[6] David Pimentel and Anne Wilson. World Population, Agriculture and Malnutrition. *World Watch Magazine*, (October):22–25, 2004. URL: http://www.worldwatch.org/system/files/EP175D.pdf.

[7] FAO Corporate. Land and environmental degradation and desertification in africa... http://www.fao.org/docrep/x5318e/x5318e02.htm.

[8] South Africa Environmental Monitoring Group, Cape Town. Desertification and land degradation in south africa. http://www.botany.uwc.ac.za/inforeep/land1.htm.

[9] Martha Stone. *How To Hydroponics: A Beginner's and Intermediate's In Depth Guide To Hydroponics*. 2013.

[10] Howard M Resh and M Howard. *Hydroponic Food Production: A Definitive Guidebook for the Advanced Home Gardener and the Commercial Hydroponic Grower, Sixth Edition*. 2002. URL: http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Hydroponic+Food+Production#1$\delimiter"026E30F$nhttp://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/093123199X.

[11] HANNA instrumens. HI98130 pH/Conductivity/TDS Tester. URL: http://shop.hannainst.com/hi98130-ph-ec-tds-tester.html?ProdCode=HI%252098130&id=002003.

[12] HortiMaX. HortiMaX Products . URL: http://www.hortimax.com/4/en/products.html.

[13] Pure Hydroponics. URL: http://purehydroponics.com/.

[14] campbellsci. URL: http://www.campbellsci.com/.

[15] Libelium. URL: http://www.libelium.com/libelium-launches-hydroponics-and-aquaponics-

[16] atlas scientific. URL: http://www.atlas-scientific.com/kits.html?

[17] Lou Frenzel. What's the difference between bluetooth low energy and ant?, Novembro 2012. http://electronicdesign.com/mobile/what-s-difference-between-bluetooth-low-energy-and-ant#3.

[18] Bluetooth SIG. The low energy technology behind bluetooth smart. http://www.bluetooth.com/Pages/low-energy-tech-info.aspx.

[19] Artem Dementyev, Steve Hodges, Stuart Taylor, and Joshua Smith. Power consumption analysis of Bluetooth Low Energy, ZigBee and ANT sensor nodes in a cyclic sleep scenario. *2013 IEEE International Wireless Symposium, IWS 2013*, pages 2–5, 2013.

[20] Jin-shyan Lee, Yu-wei Su, and Chung-chou Shen. A Comparative Study of Wireless Protocols : Bluetooth, UWB, Zigbee, and Wi-Fi. pages 46–51, 2007.

[21] Karel Povalac and Peter Ligertwood. Low Power Considerations for ZigBee Applications Operated by Coin Cell Batteries. (August):1–12, 2012.

[22] Vladimir Bychkovsky, Bret Hull, Allen Miu, Hari Balakrishnan, and Samuel Madden. A Measurement Study of Vehicular Internet Access Using In Situ Wi-Fi Networks. *MobiCom '06 Proceedings of the 12th annual international conference on Mobile computing and networking*, pages 50–61, 2006.

[23] Renata Lopes Rosa. Análise da Associação dos Protocolos de Roteamento AODV e DSR com o Algoritmo Gossip, Sistema de Quorum e com um Novo Algoritmo de Economia de Energia, PWSave. page 143, 2009.

[24] João Leitão. Gossip-based broadcast protocols. (May), 2007. URL: http://docs.di.fc.ul.pt/handle/10455/3076.

[25] Ali Saidi and Mojdeh Mohtashemi. Minimum-cost first-push-then-pull gossip algorithm. *IEEE Wireless Communications and Networking Conference, WCNC*, pages 2554–2559, 2012.

[26] Alan Demers, Dan Greene, Carl Houser, Wes Irish, John Larson, Scott Shenker, Howard Sturgis, Dan Swinehart, and Doug Terry. Epidemic algorithms for replicated database maintenance. *ACM SIGOPS Operating Systems Review*, 22:8–32, 1988.

[27] Ken Birman. The promise, and limitations, of gossip protocols. *ACM SIGOPS Operating Systems Review*, 41:8, 2007.

[28] Mert Akdere, Cemal Çağatay Bilgin, Ozan Gerdaneri, Ibrahim Korpeoglu, Özgür Ulusoy, and Uğur Çetintemel. A comparison of epidemic algorithms in wireless sensor networks. *Computer Communications*, 29:2450–2457, 2006.

[29] Wendi Rabiner Heinzelman, Joanna Kulik, and Hari Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking - MobiCom '99*, pages 174–185, 1999. URL: http://portal.acm.org/citation.cfm?doid=313451.313529.

[30] Philip Levis, Neil Patel, David Culler, and Scott Shenker. Trickle: A self-regulating algorithm for code propagation and maintenance in wireless sensor networks. *Proceedings of the First Symposium on Networked Systems Design and Implementation*, pages 15–28, 2004. URL: http://portal.acm.org/citation.cfm?id=1251177.

[31] Yu Chee Tseng, Sze Y. Ni, Yuh Shyan Chen, and Jang Ping Sheu. The broadcast storm problem in a mobile ad hoc network. *Wireless Networks*, 8:153–167, 2002.

[32] M. Kumar, L. Schwiebert, and M. Brockmeyer. Efficient data aggregation middleware for wireless sensor networks. *2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE Cat. No.04EX975)*, pages 579–581. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1392214.

[33] Wendi Rabiner Heinzelman, W.R. Heinzelman, Anantha Chandrakasan, Hari Balakrishnan, a. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, 00(c):3005–3014, 2000. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=926982.

[34] Yuldi Tirta, Zhiyuan Li, Yung Hsiang Lu, and Saurabh Bagchi. Efficient collection of sensor data in remote fields using mobile collectors. In *Proceedings - International Conference on Computer Communications and Networks, ICCCN*, pages 515–519, 2004.

[35] Bluetooth Low Energy : a smart choice for connected health devices. (8):2014, 2014.

[36] Robin Heydon. *Bluetooth Low Energy: The Developer's Handbook*.

[37] LitePoint Corporation. Bluetooth® Low Energy. 2012.

[38] Carles Gomez, Joaquim Oller, and Josep Paradells. Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology. pages 1–20, 2012.

[39] António José Freitas. Bluetooth Low Energy para monitorização da postura no ciclismo. pages 1–162, 2014.

[40] NORDIC SEMICONDUCTOR. URL: http://www.nordicsemi.com/.

[41] Department of Energy of South Africa. URL: http://www.energy.gov.za/files/esources/renewables/r_solar.html.

[42] Energias Renováveis. URL: http://www.energiasrenovaveis.com/DetalheConceitos.asp?ID_conteudo=47&ID_area=8&ID_sub_area=27.

[43] Android Developers. URL: http://developer.android.com/guide/components/fundamentals.html.

[44] Atlas Scientific. pH EZO datasheet. URL: http://www.atlas-scientific.com/_files/_datasheets/_circuit/pH_EZO_datasheet.pdf?

[45] Atlas Scientific. eC EZO datasheet. URL: http://www.atlas-scientific.com/_files/_datasheets/_circuit/EC_EZO_Datasheet.pdf?