

**Angel Fortuny Hellin**

**Sensing and wireless transmission of Conductivity and  
Temperature, in a Reverse Osmosis Column**

**Final Degree Project**

**Directed by Dr. José Luis Ramírez Falo**

**Industrial and Automatic Electronic Engineer Degree**



UNIVERSITAT ROVIRA I VIRGILI

**Tarragona**

**2023**

## **RESUMEN:**

Este proyecto consiste en realizar el estudio de viabilidad del monitoreo de la temperatura y conductividad de aguas tratadas en columnas de ósmosis inversa, así como la viabilidad de transmitir inalámbricamente esos datos desde el interior del recipiente de presión. Para lograrlo, se estudian diversas tecnologías inalámbricas disponibles en el mercado, seleccionando la tecnología BLE ("Bluetooth Low Energy"). A continuación, se llevan a cabo pruebas para evaluar la medición de temperatura y su transmisión mediante "Bluetooth Low Energy", utilizando un dispositivo Android como receptor de la información enviada. Por último, se realizan experimentos para determinar si se puede medir la conductividad y transmitir los datos hacia el exterior de la columna, con el fin de realizar posteriormente el correspondiente análisis.

## **RESUM:**

Aquest projecte consisteix en realitzar l'estudi de viabilitat del monitoratge de la temperatura i conductivitat d'aigües tractades en columnes d'òsmosi inversa, així com la viabilitat de transmetre de manera inalàmbrica aquestes dades des de l'interior del recipient de pressió. Per aconseguir-ho, s'estudien diverses tecnologies inalàmbriques disponibles al mercat, seleccionant la tecnologia BLE (Bluetooth Low Energy). A continuació, es realitzen proves per avaluar la mesura de la temperatura i la seva transmissió mitjançant "Bluetooth Low Energy", utilitzant un dispositiu Android com a receptor de la informació enviada. Finalment, es realitzen experiments per determinar si es pot mesurar la conductivitat i transmetre les dades cap a l'exterior de la columna, amb la finalitat de realitzar posteriorment l'anàlisi corresponent.

## **ABSTRACT:**

This project aims to carry out a feasibility study on the monitoring of temperature and conductivity of treated water in reverse osmosis columns, as well as the feasibility of wirelessly transmitting this data from inside the pressure vessel. To achieve this goal, various wireless technologies available in the market are investigated, with the selection of Bluetooth Low Energy (BLE) technology. Subsequently, tests are conducted to evaluate temperature measurement and its transmission using Bluetooth Low Energy, with an Android device serving as the receiving device for the transmitted information. Finally, experiments are performed to determine if conductivity can be measured and the data transmitted from the column's interior to the outside, for subsequent analysis.

---

# Index

1. Introduction .....	1
1.1 Objectives .....	1
1.2 Scope.....	2
1.3 State of the art .....	2
2. Wireless Technologies.....	3
2.1 BLE vs LoRa vs WiFi .....	3
2.2 Introduction to BLE .....	3
2.2.1 Architecture .....	5
2.2.1.1 Controller Layer .....	5
Physical Layer .....	5
Link Layer .....	5
Available Hardware.....	8
2.2.1.2 Host Layer .....	9
GAPP .....	9
GATT.....	9
2.3 Introduction to LoRa.....	10
2.3.1 Architecture .....	11
2.3.1.1 Physical Layer .....	11
Available Hardware.....	11
2.3.1.2 MAC Layer .....	12
3. Software .....	13
3.1 Arduino IDE .....	13
3.2 APP Inventor .....	14
4. Practical Salinity – Conductivity .....	15
5. Software Description and Flux Diagrams .....	16
5.1 LoRa Sender and LoRa Receiver .....	16
5.2 BLE Sender .....	17
5.3 BLE Receiver.....	18
6. Results.....	20
6.1 Wireless Technology Verification.....	21
6.2 Battery Consumption .....	23

6.3	Temperature.....	26
6.3.1	DEMO Temperature Measure, Laboratory Test .....	26
6.3.2	Sensor Cover Design to test Temperature .....	29
6.3.3	DEMO Temperature Measure, Field Test .....	31
6.3.3.1	Test 1 Battery and Temperature .....	31
6.3.3.2	Temperature and BatteryTest 2 .....	31
6.3.3.3	Temperature and Battery Test 3 .....	32
6.4	Conductivity.....	34
6.4.1	DEMO Salinity Measure, Laboratory Test.....	34
6.4.2	DEMO Salinity Measure, Fiel Test .....	53
6.5	Final Results Inside Pressure Vessel.....	54
7.	Economic Considerations.....	57
8.	Conclusions.....	58
9.	Bibliography .....	59
10.	Annexes .....	61
10.1	Specified parameters .....	61
10.2	Sensor Cover Designs .....	61
10.3	Additional Data .....	65
10.4	Schematics .....	70
10.5	Software BLE Android APP .....	71
10.6	Software BLE Sender Source Code.....	73
10.7	Sender LoRa Source Code.....	82
10.8	Receiver LoRa Source Code.....	85
10.9	User Manual .....	87

## **1. Introduction**

In recent years, the industrial sector has assumed an increasingly significant role in the advancement of mankind. Given its paramount importance, it is imperative to mitigate and prevent potential risks that can prove highly detrimental. Consequently, the establishment of a robust monitoring system capable of assessing the condition of products or compounds employed within industrial plants becomes indispensable.

This research endeavor explores the feasibility of evaluating the quality of water subjected to a desalination process within a Reverse Osmosis Column. To facilitate the transmission of measurements, diverse wireless technologies are proposed and subsequently tested.

The present project has been undertaken as a development initiative for The Company (DuPont), an esteemed multinational chemical company based in the United States. Boasting a workforce of 64,000 employees, The Company stands as one of the foremost chemistry companies globally. The organization primarily focuses on the provision of clean water, enhancement of energy efficiency, and optimization of existing products.

The Company employs a Reverse Osmosis Column as a pivotal component in its water desalination process. This specialized system plays a crucial role in converting saline water, known as the feed, into purified water, referred to as permeated or product water. Through the utilization of reverse osmosis technology, the column facilitates the removal of impurities and salt from the incoming water stream, resulting in the production of desalinated water.

### **1.1 Objectives**

The objective of this project is to measure the conductivity and temperature of the feed and the product in a reverse osmosis filtration column and, afterwards, send these measurements wirelessly to a receiver outside of the column.

Client specifications:

1. Available dimensions.
2. Maximum and minimum salinity parameters in the feed and product.
3. Salty water, flowing through the Reverse Osmosis vessel, creates an electromagnetic shielding that difficults data transmission.
4. Measurements must be sent wirelessly.
5. Battery should last at least 3 years.

## **1.2 Scope**

The Company aims to assess the salinity level of the flowing salty water within the Reverse Osmosis Columns, ensuring that it falls within the specified range. To achieve this, the project primarily focuses on the development of electronic components employed for sensing purposes. A commercial microcontroller is utilized in conjunction with a Thermistor to accurately measure the temperature, while two electrodes are employed to measure the conductivity of the water.

Moreover, the microcontroller is designed to transmit the measured data wirelessly. The receiver of this data is an Android Device, which is responsible for reading and processing the received information. Additionally, the project delves into the chemical aspects associated with the movement of positive and negative charges within the saline water.

Furthermore, various designs of the Sensor Cover have been created using STL files, facilitating the physical protection and integration of the sensing components.

## **1.3 State of the art**

Nowadays, the Conductivity can be measured in different ways such as: 2-Pole conductive Probes, 4-Pole Conductivity Probes and Inductive Conductivity Probes. Different companies offer commercial sensors that measure Conductivity with the different methods mentioned [4] [5] [9]. In order to measure conductivity, this project proposes a measurement with 2 electrodes and a PWM signal in each electrode, each PWM signal is 180 ° phase shifted to the other, and afterwards a comparison with the Company sensors.

The market has different commercial sensors that can be used to measure temperature in this project. Examples of it are: Resistance Temperature Detectors (Pt1000), ThermoCouples, Thermistors. The sensor used on this project to measure temperature is a thermistor, specifically a 100k NTC.

## 2. Wireless Technologies

When choosing a wireless technology for a project, it is necessary to have different considerations like the energy usage, the distance between devices, the implementation cost. That's why different technologies are compared: LoRa, WiFi and BLE.

### 2.1 BLE vs LoRa vs WiFi

The selected technologies are good candidates to provide a solution to the proposed objective. However, in order to compare LoRa with BLE 2 devices from the same manufacturer will be compared meanwhile for WiFi case, another manufacturer has been selected because STM32 didn't have WiFi modules:

For the LoRa case the STM32WL55xx is chosen, for the BLE case the STM32WB55XX is chosen and for the WiFi, an ESP32 is selected:

<b>Technology</b>	<b>BLE</b>	<b>LoRa</b>	<b>WiFi</b>
<i>Range</i>	Maximum 100 meters (open-air)	10-20 km	Maximum 90 meters (open-air)
<i>Consumption in "Ultra-Low-Power"</i>	13 nA	31 nA	10 $\mu$ A
<i>Transmission Consumption</i>	5.2 mA at 0 dBm	15 mA at 10 dBm	240 mA at 19.5 dBm
<i>Antenna dimensions</i>	System on a Chip	6 cm	System on a Chip
<i>Cost</i>	3 \$/u buying a pack of 10 k	4.86 \$/u buying a pack of 10 k	20.95 \$/u buying a pack of 10 k

Observing this comparison, the use of WiFi was rejected because its high consumption and unit prize. Moreover, BLE is prioritized over the use of LoRa because it has a lower consumption and a smaller prize, even though it should be checked and tested under the appropriate conditions to ensure that the protocol works correctly.

### 2.2 Introduction to BLE

BLE (Bluetooth Low Energy) is a technology with specifications for short range wireless communication. It uses of short-range radio signals with minimal power consumption to work for a longer time. It focuses on low battery consumption. It is not as fast as Bluetooth and is not suitable for transferring large amounts of data. Still, it is ideal for transferring small amounts of data with minimum power consumption. Its main purpose is to support IoT (Internet-of-Things) applications. The most suitable applications are when some sensors communicate small amounts of data despite having small batteries.

BLE operates in the 2.4 GHz ISM (Industrial, Scientific, Medical) frequency band and divides this band into 40 channels of 2 MHz each. BLE devices have an advertising mechanism, i.e., they advertise themselves so that other BLE Devices can see them or connect with them.

As previously mentioned, the most qualified or most suitable type of data to be transferred by BLE are:

- Small amount of data.
- Status information, triggered by local events.
- No real time monitoring, no Streaming.

## 2.2.1 Architecture

### 2.2.1.1 Controller Layer

The Controller Layer can be divided into two Layers, the Physical Layer and the Link Layer.

#### Physical Layer

The Physical Layer contains the analogic communications circuitry that translates the digital data received and gives service to the Link Layer.

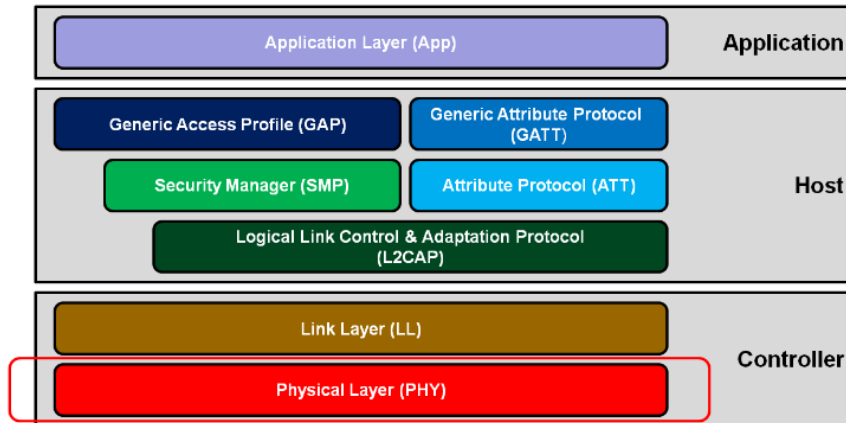


Figure 1 Physical Layer

#### Link Layer

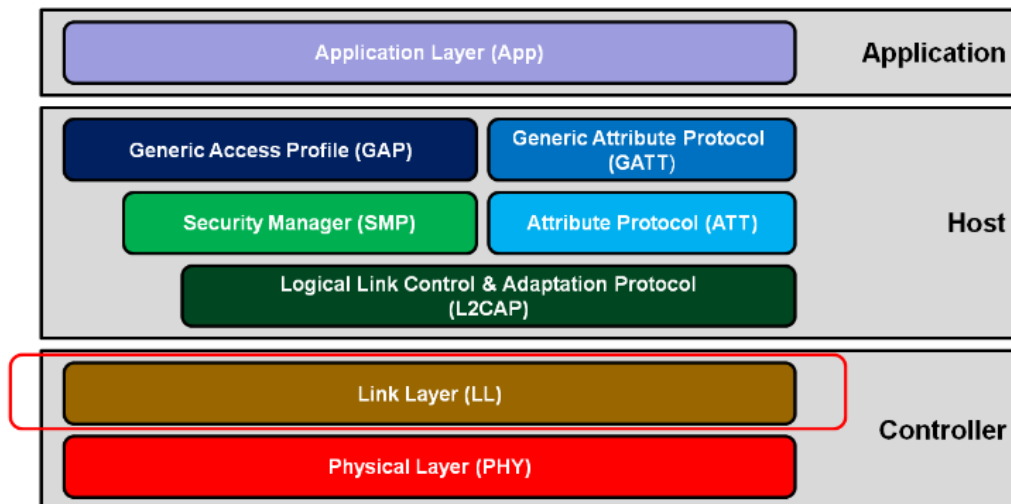


Figure 2 Link Layer

The Link Layer is the responsible of BLE advertisements, scanning and creating and maintaining connections.

There are two types of channels, advertising channels and data channels: channels 37-38-39 are advertising channels and the rest are data channels. Once a device discovers the signal and is paired, both devices are considered connected.



As we can observe, the format is the same for both advertising and data channels, however, the message (payload) is different:

- The messages for the advertising channel contain as it can be the connection to establish with the device: can connect to the device, can not, only in scan mode....

- For the data channel, this message includes a header, the packet and a message integrity check.

**Available Hardware**

Two different devices have been used, the BLE nano Keywish and the BLE Nano Sense 33.

The BLE Nano Keywish includes:

- BLE chip: TI CC2540.
- Micro-USB.
- Input Voltage: USB 5 V, Pin 6-12 V.
- MCU: ATmega328P.
- Bootloader: Newest Arduino 1.8.8.

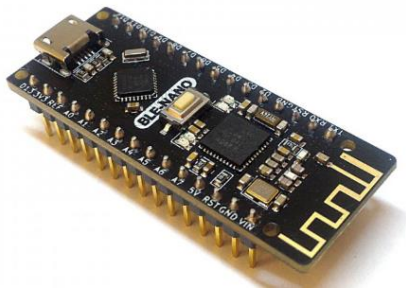


Figure 7 BLE Nano Keywish

The BLE Nano Sense 33 includes:

- NINA B306 Module (Processor, Bluetooth 5 multiprotocol radio, Peripherals).
- MPM3610 (Voltage Regulator).
- APDS-9960 (Digital proximity, Ambient light, RGB and Gesture Sensor).

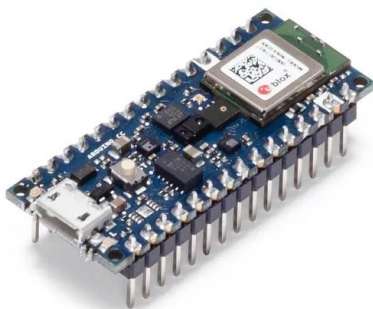


Figure 8 BLE Sense 33

### 2.2.1.2 Host Layer

In the "Host Layer" there's the Generic Access Profile (GAP) And the Generic Access Attribute (GATT).

#### GAPP

The GAP defines the modes of operation, procedures and functions/role. The functions they can perform are:

1. Broadcast: periodically sends frames to advertise.
2. Observer: searches and listens for possible devices advertising.
3. Peripheral: This mode involves bidirectionality, it advertises using advertised packets to connect to.
4. Central: It allows to establish and manage a connection.

#### GATT

1. The GATT establishes how the data will be organized and how they will be exchanged through the Bluetooth Low Energy connection. This is where the services available for a Bluetooth Low Energy device are established with their corresponding UUID and the features that each service has. Each feature also has a UUID, this is usually very similar to its corresponding service changing some value for each feature as you can see in the following example:

```
Service Name: 65504
Service UUID: 0000ffe0-0000-1000-8000-00805f9b34fb
Characteristic Name: SimpleKeyState - Characteristic Short UUID: 0xFFE1 - User Description: Characteristic 1 - Handle: 36 - 0x00000024 - Value: 0D-0A
Characteristic Name: 65506 - Characteristic Short UUID: 0xFFE2 - User Description: Characteristic 2 - Handle: 40 - 0x00000028 - Value: NULL
```

Figure 9 Example of a Service and its Characteristics

As can be seen, the first highlighted value is the service identifier, the other two that are defined as "Characteristic Short UUID" are the modifications to the service identifier for each characteristic. In other words, the Characteristic 1 would have the following UUID: 0000FFE1-0000-1000-8000-00805f9b34fb.

### **2.3 Introduction to LoRa**

LoRa (Long Range) is a wireless technology that uses a type of radio frequency modulation patented by a company called Semtech. It works in ISM bands below GHz. Specifically, these bands are:

Table 1 ISM Frequency Bands

	<b>EUROPA</b>	<b>AMÉRICA</b>	<b>ASIA</b>
<b>BANDA (MHz)</b>	868	915	433

The main characteristics of this technology are:

- High tolerance to interference
- High sensitivity to receive data.
- Its long range that can reach between 10 km and 20 km.
- Low data transfer.

LoRa makes use of the "Spread Spectrum Modulation" technique, which provides secure communications by avoiding interference by sending a signal over a wide frequency bandwidth.

In order to customize a connection, a LoRa device makes use of different transmission parameters:

- Transmit Power (TP)
- Carry Frequency (CF)
- Spread Factor (SF)
- Bandwidth (BW)
- Coding Rate (CR)

LoRa is generally divided into two layers: the Physical layer and the MAC layer.

### 2.3.1 Architecture

#### 2.3.1.1 Physical Layer

The modulation technology used is called Chirp Spread Spectrum (CSS). The so-called Chirp pulses are sent as symbols, which increase and decrease in frequency over time:

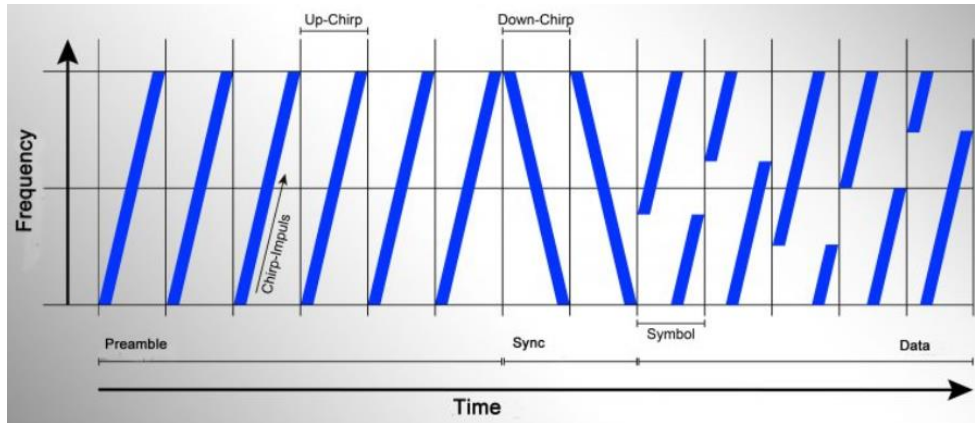


Figure 10

The Data Frame used in LoRa communications is:

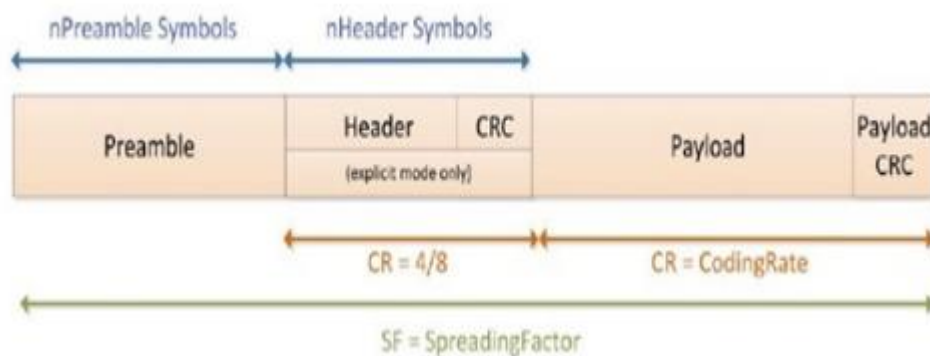


Figure 11 LoRa Data Frame

#### Available Hardware

Hardware used are Heltec ESP32 WiFi LoRa V2.1 devices.

- ESP32 microprocessor (dual-core 32-bit MCU + ULP core).
- LoRa node chip SX1276/SX1278.
- Input voltage: USB 4.7-6 V, 5 V Pin 5-6 V.
- CP2102 USB to serial port chip.
- OLED display 0.96 inch 128\*6.
- Programmable with Arduino programming environment.

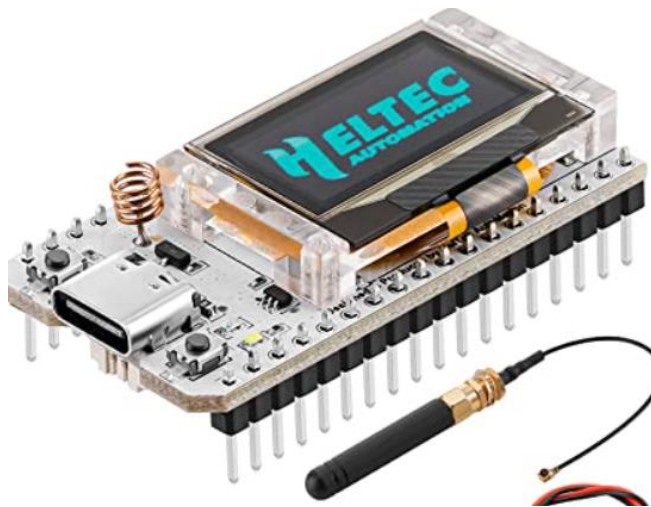


Figure 12 Heltec ESP32 WiFi LoRa V2.1

### **2.3.1.2 MAC Layer**

LoRaWAN is a MAC (Media Access Control) layer protocol and is built on top of LoRa modulation. It is a layer that defines how devices use LoRa, i.e. when they transmit and the frame formats.

A LoRaWAN network consists of different devices (nodes) that communicate with gateways and send data to a network server. Each node must be registered in a network before sending and receiving messages and this procedure is called activation, we can see the two methods:

- Over-The-Air-Activation (OTAA): the most secure and recommended activation method. The devices perform a binding procedure with the network, during which a device address is assigned and security keys are negotiated with the device.
- Activation by personalization (ABP): requires encoding the device address as well as security keys into the device. A disadvantage is that it is not possible to change network providers without manually changing the keys on the device.

### **3. Software**

With the purpose of programming the microcontrollers of the device, the Software used was Arduino IDE. It has been selected because it includes the compiler for the two available programmable microcontrollers (ESP32 from Heltec and ATmega328P for the BLE Nano Keywish).

App Inventor has also been used to create an Android apk to read the values sent by the Sensor Cover.

#### **3.1 Arduino IDE**

Arduino IDE (Integrated Development Environment) is a software application used to write, compile and upload code to Arduino boards, which are open-source microcontroller platforms. Arduino boards are widely used in the field of electronics, robotics and Internet of Things (IoT) projects for prototyping and creating interactive projects.

Developers can write code in the Arduino programming language, C++ with optimised syntax and libraries created especially for the Arduino platform, using the easy-to-use interface provided by the Arduino IDE. The IDE includes a compiler that converts code into machine-readable instructions, a code editor with syntax highlighting and tools for loading the created code onto an Arduino board.

In addition, the Arduino IDE has a built-in library manager that allows programmers to quickly add additional libraries to their projects for added functionality.

Linux, Windows and macOS are just some of the operating systems supported by the free Arduino IDE. It is a popular choice for both novice and experienced developers to create projects using Arduino boards, as it has a sizeable community of developers and users who contribute to the development of libraries, examples and tutorials.



Figure 13

### **3.2 APP Inventor**

App Inventor is a block-based visual programming environment used to create mobile applications (apps) for Android devices. Users, including non-programmers, can create their own mobile applications using this free and open source software without writing conventional code. Users can visually design the user interface and functionality of their application by dragging and dropping blocks representing the various components and functionalities using the graphical user interface (GUI) provided by App Inventor.

With the help of the many built-in components available in App Inventor, users can design a wide range of mobile applications, including games, utilities, educational applications and more. These components include buttons, text boxes, photos, sensors, databases and connectivity options. It also enables custom blocks, which allow users to write their own reusable code snippets to run their applications.

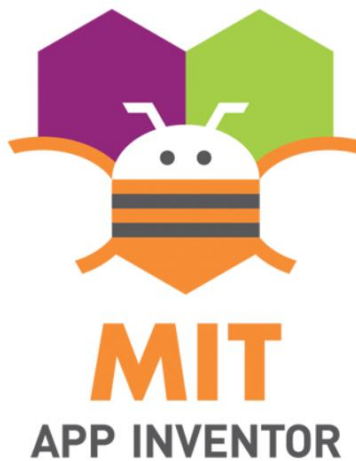


Figure 14

## **4. Practical Salinity – Conductivity**

Salinity is the amount of dissolved salts in water. The conductivity of a solution is the ability of water to conduct an electric current through its dissolved ions. It depends on the concentration of ions in the solution. Both salinity and conductivity are related because the amount of dissolved ions increases the values of both. In this specific case, the higher is the salinity, the more Cl<sup>-</sup> and Na<sup>+</sup> ions there will be, and therefore the higher the conductivity.

The Intergovernmental Oceanographic Commission (IOC) developed the idea of practical salinity to provide a standardized measure of salinity that could be precisely determined using real-world laboratory methods.

Salinity, which is used to describe how salty seawater is overall, is a crucial parameter in oceanography. The term "practical salinity" refers to the total amount of dissolved salts in seawater, which are primarily sodium chloride (common table salt) and other dissolved ions like magnesium, calcium, and potassium.

Practical Salinity is a dimensionless value. The reason of it, is because it is defined as a ratio of two conductivity measurements, both measured at specific values of temperature and pressure as it can be observed in the article [23] . Moreover, the mentioned conductivity measurements can also depend of other factors like de dissolved gases, the presence of other substances...

## 5. Software Description and Flux Diagrams

First it will be focused the LoRa sender and Receiver because they were just used at the beginning of the project. BLE was the final Wireless Technology used.

### 5.1 LoRa Sender and LoRa Receiver

The Sender has been programmed so that it modifies its Txpower every 4 packet transmissions. In Order to do that there are 2 variables. The variable counter, which is incremented each time a packet is send. When the counter arrives to a value of 4, it restarts to 0 and increases *tandas*<sup>1</sup> variable by 1 (that will modify the Txpower of the next packet). If *tandas* gets to 15 it is restarted to 0.

The Receiver checks if a packet has been received and it saves its size, the RSSI and the strings received and then prints all this data in the display.

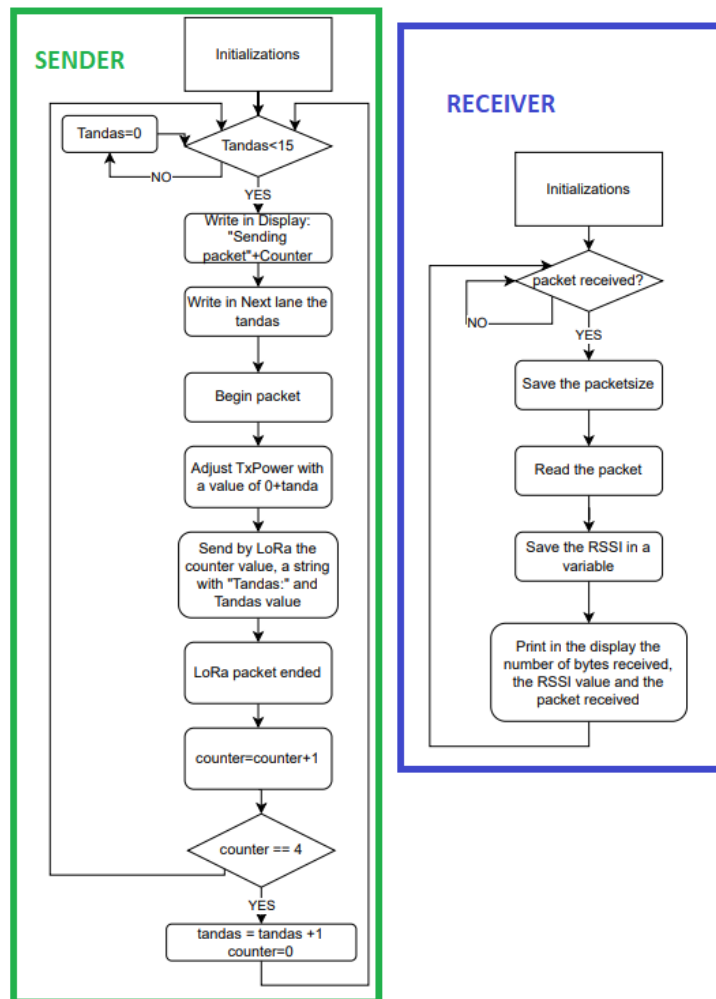


Figure 15 Flux Diagram of the Sender LoRa and Receiver LoRa device.

<sup>1</sup> The variables will be with this style

## 5.2 BLE Sender

The Software code lines are in the Chapter [10.6 Software BLE Sender Source Code](#). The Software is mainly divided into two big parts: the Setup and the Loop.

After the initializations, in the loop, the program starts turning off the PWM0 and turning on the PWM1, there are different functions written to do that, check in the [10.6 Software BLE Sender Source Code](#). Next, the ADC makes its measurements and save the data in a local table. Immediately, both PWM are turned OFF and then all the data is sent with its identifier. Later, the microcontroller sleeps 2 seconds to differentiate the data from the feed and the Product and it repeats the same procedure.

After making the conductivity measurements, it is necessary to adjust de Voltage Reference of the ADC before making the Temperature measurements. When the Temperature measurements are finished, the ADC Voltage Reference is changed to 1.1 V again and the Microcontroller is sent to Sleep the specified time.

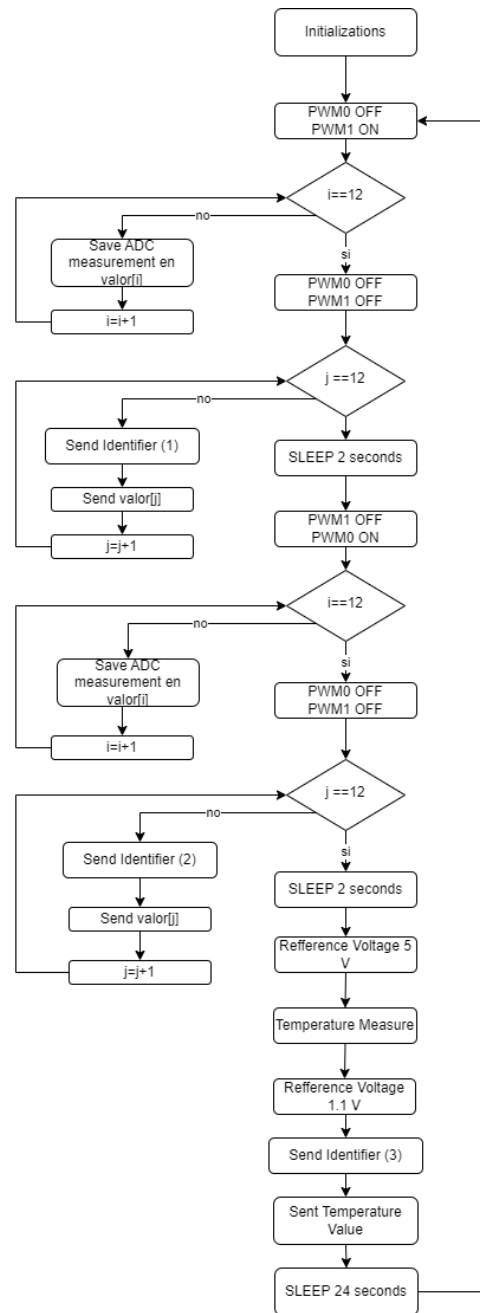


Figure 16 Flux Diagram of Sensor Device

### 5.3 BLE Receiver

The Software Block diagram used in App inventor is in the Chapter [10.5 Software BLE Android APP](#).

It mainly consists of a Screen with different buttons, as it can be observed in [Figure 25 Android App](#).

First, when the button scan is pressed, it starts adding in a List all the found devices until one device is selected. Next, the Service in which the data is being sent needs to be selected.

Once the Service is selected, each time a String is sent to this Service, an Interruption appears, which makes the App send the data to a Google Forms.

In addition, if disconnect button is pressed, the Android device disconnects from the BLE device which was connected.

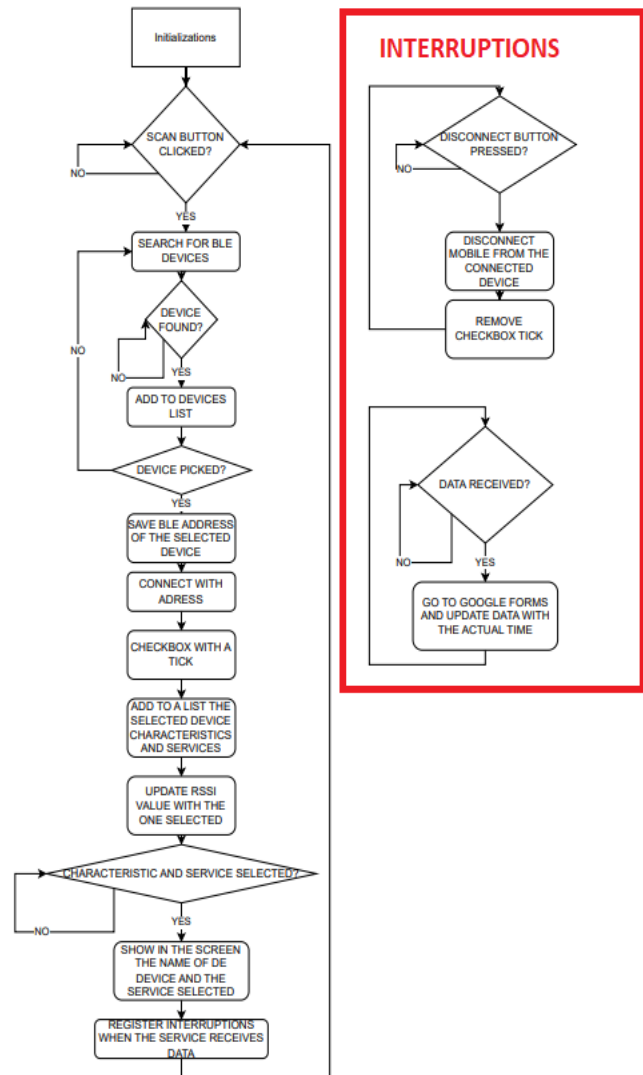


Figure 17 Flux Diagram of the App

The Google Forms needs to be previously created and prepared according to the data that will be received. For instance, in the first test the Android device uploaded in the google forms the name of the device (Figure 18 Example of Data Received Through the app.), the RSSI and the Temperature. The google forms have 3 questions which had a text answer. After creating the Google forms, it is necessary to save the URL of each question so that the data will be updated at each corresponding URL.

Marca temporal	Nombre	Datos	RSSI
20/03/2023 9:07:13	MINOSTEMP	["Temp 2:557"]	-57

Figure 18 Example of Data Received Through the app.

. An example of use of the Google Forms URL is:

[https://docs.google.com/forms/d/10Uk5Gqa3P-4gA\\_BmFGQhbIyWbEUNOvU7a02DRNTgYHs/formResponse?entry.1305724076=ANSWER1&entry.1007060198=ANSWER2](https://docs.google.com/forms/d/10Uk5Gqa3P-4gA_BmFGQhbIyWbEUNOvU7a02DRNTgYHs/formResponse?entry.1305724076=ANSWER1&entry.1007060198=ANSWER2)

Where ANSWER1 and ANSWER2 would be the data send by the mobile APP, so that if the device receives data and wants to save it in the Google Sheets, the Android Device must automatically enter to this URL and replace ANSWER1 and ANSWER2 for the corresponding data read from the sensors. To generate this URL, it is necessary to previously do a preview of the Google Forms URL and make a test answer.

The Android app can be downloaded with this URL:  
<http://ai2.appinventor.mit.edu/b/337p9>

## 6. Results

Five terms must be introduced:

- **Sensor Cover:** which refers to the box in which the device is inserted so that it does not get wet.

- **Endcap:** the interconnector between two RO (Reverse Osmosis) elements into which the Sensor Cover is inserted.

- **Pressure Vessel:** the RO element, where the reverse osmosis process is carried out.

- **Feed:** is the feeder, where the salt water enters to be permeated.

- **Product or Permeate:** is where the water subjected to the Reverse Osmosis process circulates, i.e., the permeated water.

The first steps of the project consisted of designing the Sensor Cover to be introduced in the Endcap. In the first tests, it was decided to use a model made in 3D printing. The material used is polylactic acid, commonly known as PLA. The pressure that this Sensor Cover must withstand is in the 10.1 Specified parameters for water in dynamic form, i.e. this is the osmotic pressure, the minimum operating pressure that causes the dissolved species to separate from the water. Despite, tests can be done with water in static form with the 10.1 Specified parameters. It was decided to carry out the first tests in static and some pressure tests are done to check the integrity of the 3D impression. For this purpose, the dimensions of the endcap are measured and the following calculations are made (considering the worst conditions and working in static):

$$P = \frac{F}{A} \rightarrow 400.000 \text{ Pa} = \frac{F}{1\text{m}^2} \rightarrow 400.000 \text{ Pa} = \frac{F}{10000\text{cm}^2} \rightarrow F = 40 \frac{\text{N}}{\text{cm}^2}$$

Equation 1

To proceed, it was necessary to make use of as much space as possible. Because of the dimensions of the Endcap as shown in [Figure 45 Endcap Dimensions](#), the area of the circular sector is calculated using two spokes of the Endcap, i.e. an angle of 45°.

$$A = \pi \cdot (r1^2 - r2^2) \cdot \frac{\alpha}{360} = \pi \cdot (8.9^2 - 2.45^2) \cdot \frac{45}{360} = 27.37 \text{ cm}^2$$

Equation 2

The Force that is needed to overcome is:

$$F = 40 \frac{\text{N}}{\text{cm}^2} \cdot 27.37 \text{ cm}^2 = \mathbf{1094,8 \text{ N}} \rightarrow 110.65 \text{ kp (110 kg)}$$

Equation 3

The Sensor Cover was designed and printed with a 3D printer by setting the parameters of the thread to be printed (infill) as robust and compressive as possible. The design had to prevent water from entering, which is why the cover was made to fit the base as shown in

Figure 46. After that, 120 kg were placed on top of the Sensor Cover and it was checked that it did not break. In addition, it was flooded in a box with water and left there for 1 hour and it was found that no water entered.

## **6.1 Wireless Technology Verification**

To begin with, a Nano Keywish BLE device was programmed to send a data packet continuously. The device has two microcontrollers, a programmable ATmega328P and a non-programmable CC240. This last device mentioned, acts as a communicator, its only purpose is to transmit to BLE everything it receives by UART from the ATmega328P and sends through UART everything that arrives by BLE. To check that values are read or sent to this device it is necessary to use a BLE application. Initially, the BLE Explorer Bluetooth Computer application is used.

Two Heltec ESP32 devices were programmed using Arduino IDE to communicate with each other using LoRa technology. One device sent the data and the other received it and showed it on the display. In addition, the device sending the data changed its transmission power every 4 packets sent.

The device selected must be able to read the salinity and temperature of both the feed and the permeate. It is for this reason that, in the test, the entire inside of the Sensor Cover needs to be shielded with aluminium foil except for the area closest to the innermost diameter of the Endcap, i.e. where the desalinated water is located. In this way, if communication is possible, it will be possible from both the Feed and the Product.

The battery used is a 3.7 V Alkaline battery. No other battery was available at the time of the first visit, which is why the tests were performed at a voltage below the minimum. This battery voltage was not sufficient to transmit packets, but the Received Signal Strength Indicator (RSSI) could be read.

The first test was done with the BLE device because it is the first choice. Once the BLE device had been placed inside the Sensor Cover, wrapped in cling film and the inside of the Sensor Cover filled with paper, it was placed in its corresponding Endcap, which was placed inside an RO element through which water was circulated, increasing its salinity progressively. The pressure was between the static parameters. The value of the Received Signal Strength Indicator (RSSI) was recorded as the salinity increased. The results obtained by taking values from 0.5 m from the module were the shown in Graphic 26.

It's important to note that the connection with the device was lost at 58 mS/cm, which is when we approached it and re-established the connection. A few seconds later, it was lost again, which is why we proceeded to remove it and when we did so, we found that the electronics were a bit wet. There were no fractures in the Sensor Cover.

It was then decided to do the same experiment without the aluminium foil shielding. Initially we were at 70 mS/cm and the RSSI read was -66 dBm, logically the RSSI decreased with the removal of the shielding. Unfortunately, we were only able to take this measurement because when we increased the salt water flow rate slightly there was a sudden loss of connection. We removed the Sensor Cover and when we opened it, we saw that the inside

was completely flooded and the water had probably short-circuited the electronics. The battery and the device were removed with caution. We did not test the LoRa modules because we did not have any more batteries and because the first option had been successful.

## 6.2 Battery Consumption

The microcontroller that enters to "PowerSave" is the ATmega328P, the CC240 cannot be programmed so it is not possible to put the whole device in "Sleep" mode to save battery. We have proceeded to look at the consumption of the device with one microcontroller awake and the other asleep, we have also proceeded to look at the change. To do this, an  $10.6 \Omega$  resistor was added in series with the battery resistor and this resistor was in series with the input of the device:

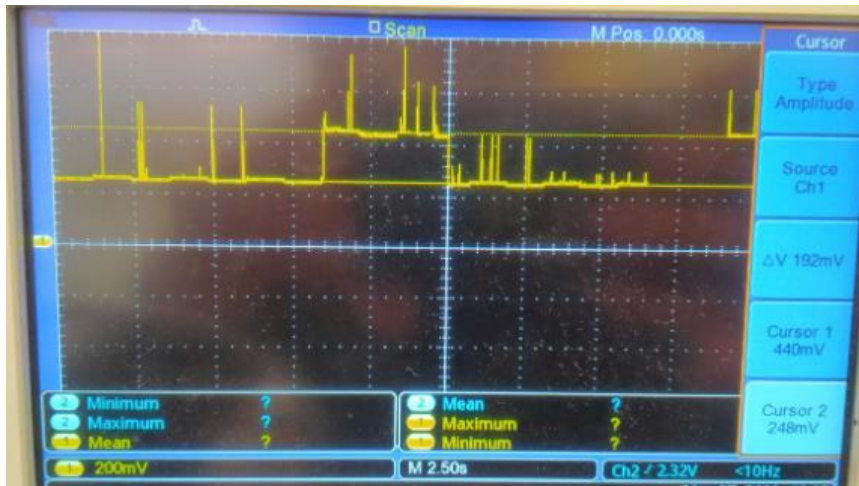


Figure 19 Not connected to any device and 4 seconds awake.

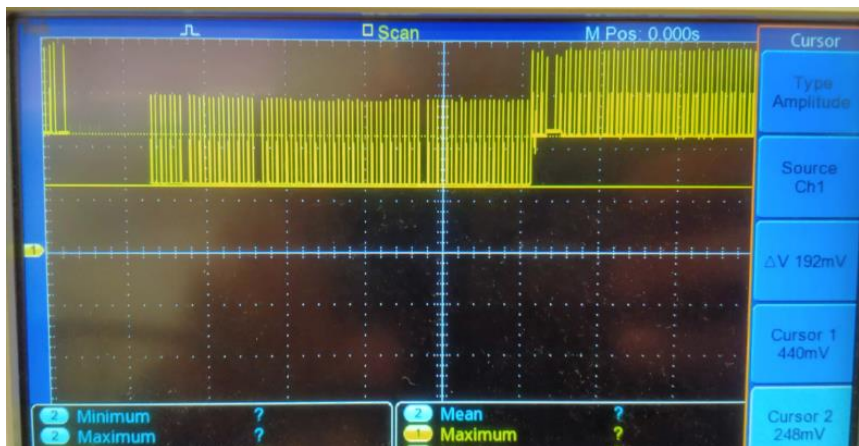


Figure 20 Connected and sending data every 100 mS, 4 seconds awake.

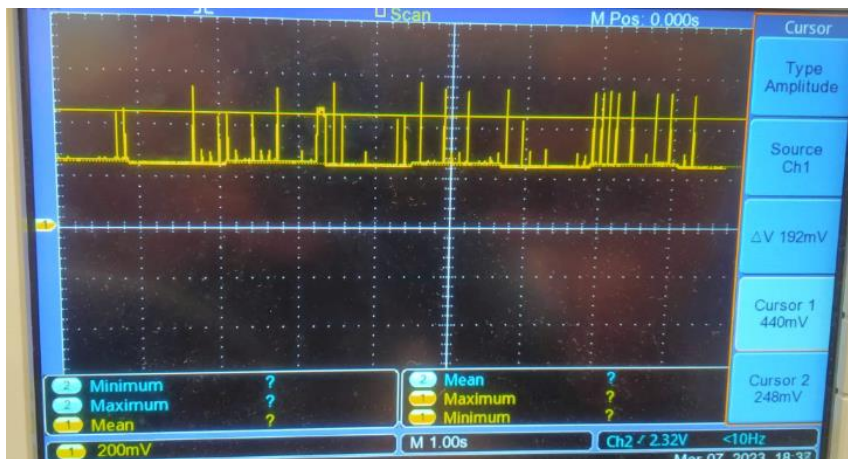


Figure 21 Not connected and awake time of 100 ms. When is awake takes the measure and enters Sleep mode

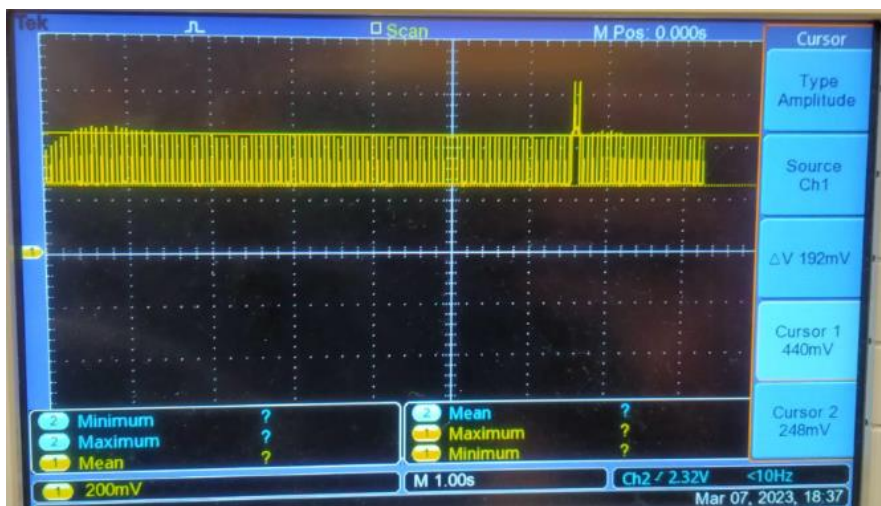


Figure 22 Connected and awake time of 100 ms. When is awake takes the measure and enters Sleep mode

Looking at this previous Figures and knowing that the resistance is  $10.6 \Omega$ . The average power consumption values can be calculated when not transmitting:

$$I = \frac{V}{R}$$

Equation 4

Substituting in the Equation 4 when de device is awake (1) and when is Sleeping(2):

$$1. I = \frac{V}{R} = \frac{0.44 \text{ V}}{10.6 \Omega} = 37.9 \text{ mA}$$

$$2. I = \frac{V}{R} = \frac{0.25 \text{ V}}{10.6 \Omega} = 21.55 \text{ mA}$$

It is important to note that the current peaks when not connected are due to the device being advertising. The other peaks when the device is awake are due to the communication.

With three batteries of 150 mA·h of Capacity and considering 25 mA of average consumption, the batteries will last:

$$T = \frac{450 \text{ mA} \cdot \text{h}}{25 \text{ mA}} = 18 \text{ hours}$$

The desired microcontroller with BLE might have a low power consumption in sleep mode, the proposed one is STM32WB55XX which consumption is:

- 13 nA shutdown mode
- 600 nA Standby mode + RTC + 32 KB RAM
- 2.1 µA Stop mode + RTC + 256 KB RAM
- Radio: Rx 4.5 mA / Tx at 0 dBm 5.2 mA

Figure 24 Sending packets Consumption

Figure 23 Ultra Low Power Mode

Taking into account that the Microcontroller STM32WB55xx might get awake every 3 hours and send the data during approximately 50 milliseconds and go to sleep, a button battery of 150 mA·h would last.

$$T = \frac{150 \text{ mA} \cdot \text{h}}{2.1 \cdot 10^{-3} \text{ mA} + 5.2 \text{ mA} \cdot \frac{50 \text{ ms}}{2 \text{ h} \cdot \frac{3600000 \text{ ms}}{1 \text{ h}}}} \cdot \frac{1 \text{ day}}{24 \text{ h}} \cdot \frac{1 \text{ year}}{365 \text{ days}} = 8.02 \text{ years}$$

Moreover, nowadays there are button batteries of 3 V with capacities of 220 mA·h for 2.38 € (button cell CR2032) or even though, button cell batteries of 620 mA·h (CR2450). It's important to reming that the batteries might be welded in series to have the enough voltage, as it's explained in the 10.9 User Manual.

### 6.3 Temperature

Having seen the results of the communication with BLE, it has been decided to develop a software which puts the Keywish BLE nano device available in "PowerSave" mode (puts the microcontroller in a battery saving mode) and send packets periodically. Subsequently, to check that these packets will be read with the application that has been created using APP Inventor:

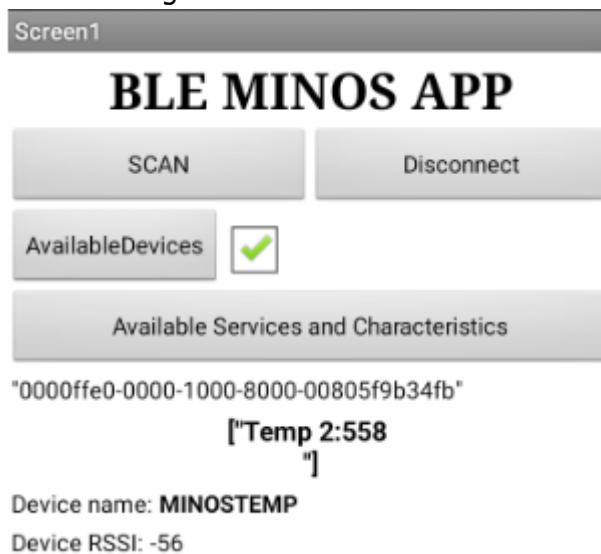


Figure 25 Android App

This application reads the RSSI and the data sent by the connected device. This APP has been programmed to upload the data read and the RSSI into a Google Sheets document, it has been programmed as it follows in [10.5 Software BLE Android APP](#).

#### 6.3.1 DEMO Temperature Measure, Laboratory Test

In order to measure the temperature, a microcontroller will make measures and send the data. The data/packets sent are measures from the analogue-to-digital converter <sup>2</sup>(ADC) provided by the microcontroller. These readings will be values of the voltage drop that a temperature sensor (NTC) will have. It's important to note that the NTC is a non-linear component but we will work between 12 °C -25 °C, so it's behaviour can be approximated as if it was linear. After the NTC changes it's resistance (measure of Temperature explained afterwards, then the CC240 microcontroller is responsible for sending these conversions by BLE and then the microcontroller is put in sleep mode ("PowerSave" mode activated) for the specified time, so that the same procedure can be repeated. The maximum time that can be in "PowerSave" mode is 8 seconds for the processor (ATMega328P) that is available in the BLE nano Keywish device. Despite this small period of time, it can be sleeping the 8 seconds, get awake and instantly go sleep again so if it is necessary a bucle sleeping can be done to sleep the desired time.

---

<sup>2</sup> <https://www.sciencedirect.com/topics/engineering/analog-to-digital-converter>

The ADC (Analog-Digital-Converter) which is available is 10 bits so the voltage value received will be between  $(2^0 - 1)$  and  $(2^{10} - 1 = 1023)$ . The temperature sensor used is an NTC (Negative Temperature Coefficient) thermistor.

This sensor is a resistive sensor that decreases its resistance as a function of temperature. The relationship is as follows:

$$R(T_{NTC}) = R_{298} \cdot e^{\left(B \cdot \left(\frac{1}{T_{NTC}} - \frac{1}{T_{298}}\right)\right)}$$

Equation 5

Where the specified temperature is in °K and B can be approximated by the data given by the manufacturer as we can see in [Figure 26](#):

°C	Ohms
2.78	302466
3.89	285206
5.00	269035
6.11	253877
7.22	239664
8.33	226331
9.44	213819
10.56	201971
11.67	190946
12.78	180588
13.89	170853
15.00	161700
16.11	153092
17.22	144992
18.33	137367
19.44	130189
20.56	123368
21.67	117000
22.78	110998
23.89	105338
25.00	100000

Figure 26

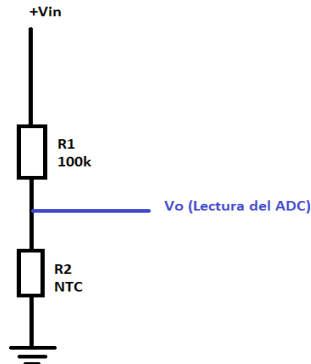
With these values we can find an approximate value of B, knowing the temperature range in which we find ourselves. In this way B is:

$$B = \frac{\ln(R1/R2)}{\frac{1}{T1} - \frac{1}{T2}}$$

Equation 6

And substituting values between the range of 5° and 25° we see that we can approximate B to a value of 4130 °K. Now, we must find the value of the Resistance at the measured temperature and once this Resistance value is obtained, the Temperature value can be found.

To find the value of the resistor we will make a voltage divider assuming R1 as a 100 kΩ resistor that is put in series to the thermistor R2, Vin a known voltage input and Vout the value read by the analogue-to-digital converter:



$$V_0 = \frac{R_2}{R_1 + R_2} V_i \rightarrow R_2 = \frac{V_0}{V_i} \cdot (R_1 + R_2) \rightarrow R_2 = \frac{\left(\frac{V_0}{V_i} \cdot R_1\right)}{1 - \frac{V_0}{V_i}}$$

Equation 7

Figure 27

Once the resistance value for each temperature is known, the measured temperature value can be obtained as follows:

$$\begin{aligned} R(T_{NTC}) &= R_{298} \cdot e^{\left(B \cdot \left(\frac{1}{T_{NTC}} - \frac{1}{T_{298}}\right)\right)} \rightarrow \frac{R(T_{NTC})}{R_{298}} = e^{\left(B \cdot \left(\frac{1}{T_{NTC}} - \frac{1}{T_{298}}\right)\right)} \rightarrow \ln\left(\frac{R(T_{NTC})}{R_{298}}\right) \\ &= \left(B \cdot \left(\frac{1}{T_{NTC}} - \frac{1}{T_{298}}\right)\right) \rightarrow \frac{\ln\left(\frac{R(T_{NTC})}{R_{298}}\right)}{B} = \left(\frac{1}{T_{NTC}} - \frac{1}{T_{298}}\right) \rightarrow \\ &\rightarrow T_{NTC} = \left(\frac{\ln\left(\frac{R(T_{NTC})}{R_{298}}\right)}{B} + \frac{1}{T_{298}}\right)^{-1} \end{aligned}$$

As we know the constants of our thermistor and substituting we obtain:

$$T_{NTC} = \left(\frac{\ln\left(\frac{R(T_{NTC})}{100000}\right)}{4130} + \frac{1}{298}\right)^{-1}$$

Transformed to °C is:

$$T_{NTC} = \left(\frac{\ln\left(\frac{R(T_{NTC})}{100000}\right)}{4130} + \frac{1}{298}\right)^{-1} - 273.15$$

In this way we obtain the temperature values as a function of the resistance of the NTC, which we can obtain thanks to the previous voltage divider, substituting we obtain:

$$T_{NTC} = \left( \frac{\ln \left( \frac{\left( \frac{V_o}{V_i} \cdot R_1 \right)}{100000 \cdot \left( 1 - \frac{V_o}{V_i} \right)} \right)}{4130} + \frac{1}{298} \right)^{-1} - 273.15$$

Equation 8

Where:

- $V_o$  is the ADC measure.
- $V_{in}$  is the voltage given to both resistances.
- $R_1$  is the resistance in series to the one of  $100k\Omega$ .

Once the theoretical framework was known, tests were carried out in the laboratory to verify correct operation. To do this, the  $R_{100k}$  was connected to the output of a pin of the BLE nano Keywish which gave us 5V, the NTC was placed in series and then to ground and the measurements were taken, the circuit is as shown in Illustration 8. Firstly, the NTC was left at the ambient temperature we were at, which was  $21.1^\circ$  and then it was placed in water-ice, estimated at  $0^\circ$ . The results obtained are:

Table 2

ADC value	ADC (V)	ADC value	Thermistor Resistance ( $\Omega$ )	Temperature ( $^\circ\text{C}$ )
563	2.751710655		122024.1304	20.71356633
790	3.861192571		338038.6266	0.848370035

Therefore, the use of the NTC is verified with a small error. This error occurs because the NTC isn't lineal. It is observed that the error is bigger at  $0^\circ\text{C}$ . In our project, we the temperature conditions will be between  $12^\circ\text{C}$  and  $25^\circ\text{C}$ , because of that we can approximate this no-lineal behaviour as if it was lineal.

### 6.3.2 Sensor Cover Design to test Temperature

At the same time, since water got inside the sensor cover at the last tests, a new sensor cover was designed. It was decided to put an O-ring (rubber rubber) around the Sensor Cover and to close the cover with screws. This way, when the screws are tightened, the rubber is squeezed and as it is elastic, it is squashed on the adjacent sides so that it should prevent

water from entering:

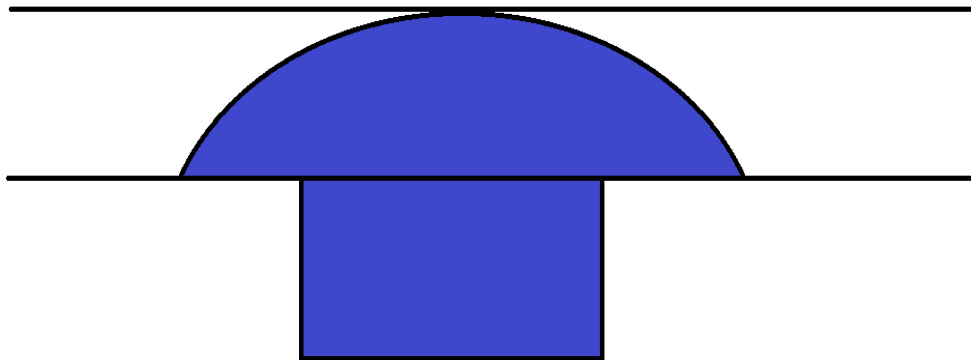


Figure 28

The proposed design was [Figure 47](#). The design was then 3D printed and the 1mm diameter rubber was added, along with the appropriate grafts to screw the lid on. It was placed in a container of water and swirled to see if any water could get in.

Unfortunately, it entered the device at the intermediate areas between screw and the screw. In view of these results, a new design was made (not the final design) with more grafts and screws, but of smaller sizes so as not to influence the space needed for the electronics and battery as is shown in [Figure 49](#).

### 6.3.3 DEMO Temperature Measure, Field Test

#### 6.3.3.1 Test 1 Battery and Temperature

After performing the DEMO tests in the laboratory, the results were checked with the device inside the RO element. To carry out the tests, two programmed BLE nano Keywish BLEs were used. One of those devices sent the voltage received by the battery and the other sent the current temperature data. These were inserted into two different Sensor Covers and added to an Endcap that had 2 cut spokes as we can see in [Figure 50](#):

Inside each Sensor Cover, there were also the corresponding batteries. These were 2 batteries soldered in series that were of 3.3 V and 190 mA·h capacity. This corresponds to 6.6V batteries with a capacity of 190 mA·h, and it had to be done because a battery had to be improvised as the ordered A544-4LR44-6V did not arrive on time. Assuming a consumption of 30 mA (it was at 21.5 mA asleep and 37.9 mA awake) the battery would last a little more than 6 hour →  $30 \text{ mA} \cdot 6 \text{ h} = 180 \text{ mA}\cdot\text{h}$ . Even so, the voltage provided is decreasing and considering that it must be supplied between 6 and 12 V in about 3 hours we would already be below the minimum supply voltage and it would not be able to send the data correctly. For this reason, it was decided to close the Sensor Cover with the batteries in the same The Company facilities and then the Endcap with both Sensor Covers was placed inside the RO element. The results obtained were as follows:

1. On the one hand, in the case of the device that checked the battery, we could not get any results because when the water was introduced, we lost the connection to the battery.
2. On the other hand, with the sensor measuring the temperature we got results:

Table 3

Conductivity (mS/cm)	RSSI (dBm)	Temperature (°C)
1.85	-51	21.04
1.85	-51	17.23
1.85	-51	16.16
1.85	-51	15.34
1.85	-51	17.73
2.4	-51	18.31

The water temperature was 15 °C. After making a few measurements, we set out to add more salt to the tank. Unfortunately, we also lost the connection then, and were not able to take any more measurements. The Sensor Covers were carefully removed and when we opened them we saw that they were both full of water. We cautiously removed the batteries and dried everything out and fortunately after drying both devices could be recovered.

#### 6.3.3.2 Temperature and Battery Test 2

It was decided to repeat the test sealing the Sensor Cover in the The Company's installations. This was done due to the short life of the batteries. The temperature measuring

device was placed inside the Sensor Cover with its corresponding battery and sealed. While the same procedure was to be repeated for the other device, the connection to the first sealed device was lost, it was later discovered that this was due to a bad contact with the battery and unfortunately the tests could not be carried out with this device.

The second device could be sealed without a problem, was introduced into the Pressure Vessel and it was decided to start with very salty water this time, 70 mS/cm when the standard was 50 mS/cm. Initially when the water was circulated through the element, it was not able to establish a connection with the device, the conductivity was lowered to see if the high salinity was the reason, but it was never detected. When the Pressure Vessel was opened to remove the Sensor Cover, the endcap was stuck inside as we can see in [Figure 51](#) Endcap Stuck inside the Pressure Vessel.

We opened the sealed Sensor Cover and found that a small amount of water had entered, it possibly created a short circuit, but after cautiously removing the battery and drying the device. It was still working.

### **6.3.3.3 Temperature and Battery Test 3**

After the test was completed, a technician from the Mechanical Engineering department was contacted. The technician explained that, at high pressures, a 3D printed design is not a barrier to fluid flow through it. Because these are simply layers of filament on top of layers of filament, water passes through these ABS 3D printed polymers. It was then decided to do a CNC mechanised Sensor Cover to repeat the tests performed:

Once this CNC mechanised was achieved, two pairs of two A544-4LR44-6V batteries were soldered in parallel for longer life, these were 190 mA-h.

To carry out the tests inside the Pressure Vessel, two Sensor Covers were taken: the CNC mechanised and the one 3D printed. The device that measured temperature was introduced Inside the CNC mechanised and the one that sent the battery level and the RSSI of the transmission received was inside the 3D printed one. In addition, the 3D printed Sensor Cover was packaged inside two vacuum-sealed plastic bags.

Both Sensor Covers were placed inside the Pressure Vessel and the results obtained were as follows in [Figure 52](#) **Mechanised** CNC Sensor Cover.

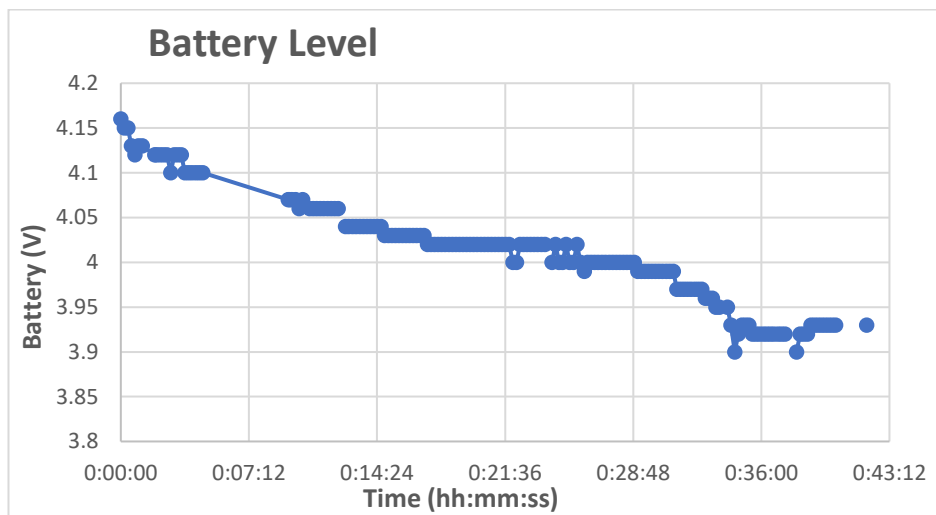
For the sensor which measured Temperature, the results obtained by our Sensor and by The Company's sensor is show in [Graphic 24](#) Shows the Temperature comparisons between the inside and the outside of the Sensor Cover.

[Graphic 24](#) Shows the Temperature comparisons between the inside and the outside of the Sensor Cover. shows with the blue series indicates the temperature being read by the NTC inside the Sensor Cover. The orange series is the external sensor used by the company to read the temperature.

Initially, after inserting the Sensor Cover inside the element, 25 minutes were spent recirculating at a pressure of 0.5 bar and waiting for the temperature read inside the Sensor Cover to stabilise and stationary. At this point, the pressure was increased to 2 bar and the water we intentionally increased the water temperature. The orange series indicates the actual

water temperature, and the blue series indicates the read water temperature. As expected, due to heat transfer, the temperature inside the Sensor Cover increased as well, with a time difference of approximately 5 minutes. After 40 minutes, the pressure was increased to 5 bar and after 53 minutes the water was stopped heating. As we can observe, temperature of both series have been decreasing. Initially, both series had a small temperature difference, but then a step is appreciated in both series can be seen. This was because instead of letting the water cool down, new water has been added at a new temperature of 11° and this is the reason why there has been a decreasing step in the temperature.

Regarding to the sensor that measured the battery and the Received Signal Strength Indicator (RSSI). Initially it was possible to measure both values, but as soon as the pressure went to 5 bar, water started to enter the Sensor Cover and the signal was lost a few minutes later:



Graphic 1 Battery measures

## 6.4 Conductivity

The main point of the salinity tests is to differentiate feed water from the product water. In order to do achieve it, Conductance will be the main focus of the experiment and therefore the Resistance (the resistance is the inverse of the Conductance  $R = 1/G$ ). It is going to be measured the measured the voltage differential in the water and transform this value into conductance.

### 6.4.1 DEMO Salinity Measure, Laboratory Test

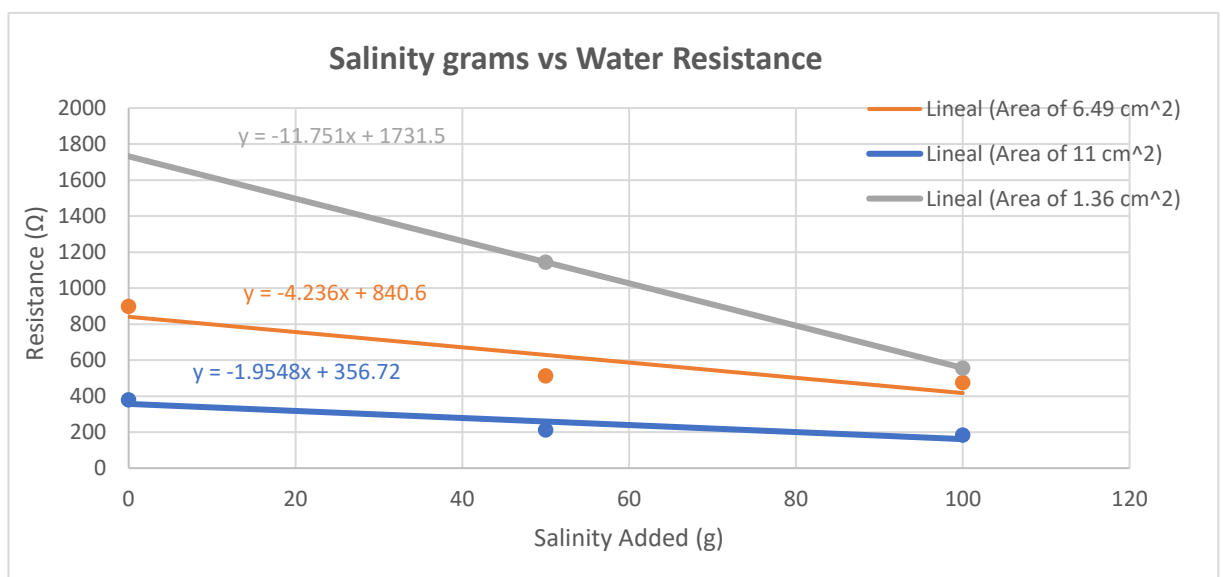
We have been working with two electrodes and testing whether it is possible to check whether the salinity is within the established limits by using these two electrodes. It is worth mentioning that the input signal for the tests must be an alternating signal, because in this way the positive and negative ions of the salt water move from one electrode to the other and do not remain deposited on the electrode surfaces. Initially, the tests should be able to differentiate between the two specified salinity thresholds. It should be specified that salt water has a capacitive behaviour, i.e. in addition of being resistive, it is capacitive, although we are focusing on the resistive part of this.

It is proposed to carry out tests for different electrode sections and, using the voltage divider, to find out the different resistance values:

$$V_o = \frac{R_A}{R_A + R_B} V_i \rightarrow R_A = \frac{V_o}{V_i} \cdot (R_A + R_B) \rightarrow R_A = \frac{\left(\frac{V_o}{V_i} \cdot R_B\right)}{1 - \frac{V_o}{V_i}}$$

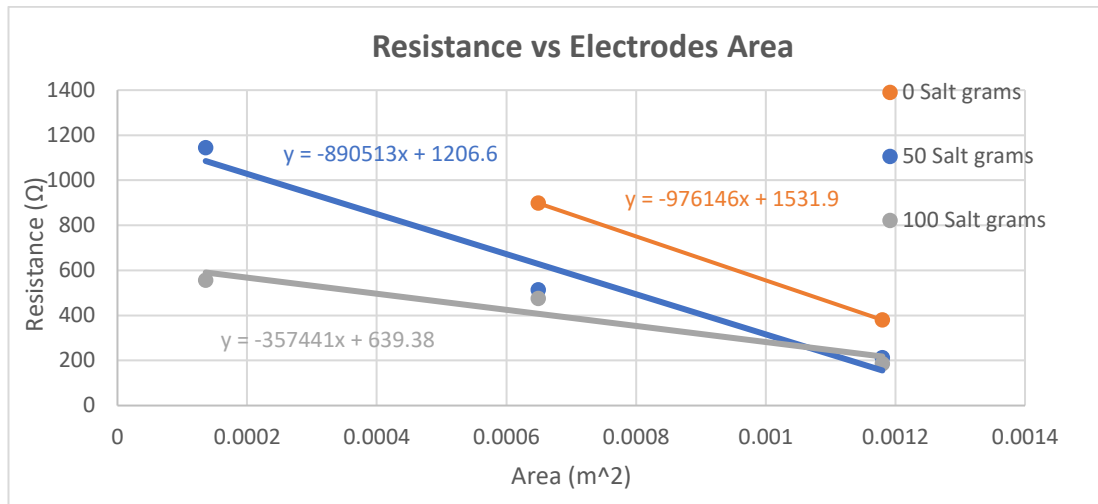
Equation 9

Where  $R_B$  is the  $R_{bias}$  introduced before the electrolyte and  $R_A$  la the corresponding water resistance. For the first tests a  $R_B$  of  $220 \Omega$  was introduced, afterwards it will be discussed which value of  $R_B$  is better to select. In addition, frequencies of 150 Hz were used and the amount of water used in the tests was 220 ml. The results were as follows:



Graphic 2 Resistance vs Grams with 3 different Areas.

In the Graphic 2 Resistance vs Grams with 3 different Areas, it is observed that the bigger is the area, the lower the resistance. It is worth noting that the first resistance data for the smallest section (data in grey) is a value of 14300 Ω, this is the reason why the value is not added because the rest of the values would not be visible. This resistance is so big because a non-saline water with such a small area acts as a much bigger resistance value. The following graph shows the values as a function of the electrode cross-section:



Graphic 3

We can see in both graphs that the higher the amount of solute, the lower the resistance and therefore the more conductive the water is. As mentioned above, the resistance depends on the area of the electrolyte used as follows:

$$R = \rho \cdot \frac{L}{A}$$

Equation 10

Where A is the area of the electrodes used, L the length, ρ the conductivity value.

Once we have seen that there is a relationship between the selected section, the amount of solute and the conductivity, we proceed to carry out different tests to select the resistance values (depending on the section) that correspond to the different salinity limits given.

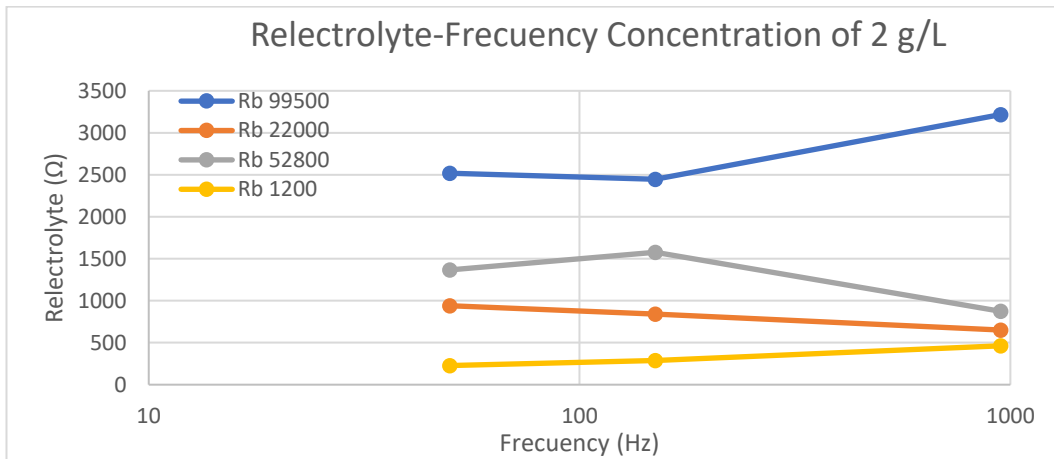
Two different samples are prepared which contain the parameters established by The Company, they are mainly created with dissolutions and dilutions:

- For the Feed it should be between 0.05 g/L and 2 g/L with a standard value of 0.5 g/L. The samples prepared are for working with the Feed: One with distilled water and salt with a ratio of 2 g of salt per litre of distilled water (maximum case for the Feed) and the other sample with 0.5 g of salt per litre of distilled water (standard case for the Feed). In order to make these correspondences, 5-litre bottles of distilled water were

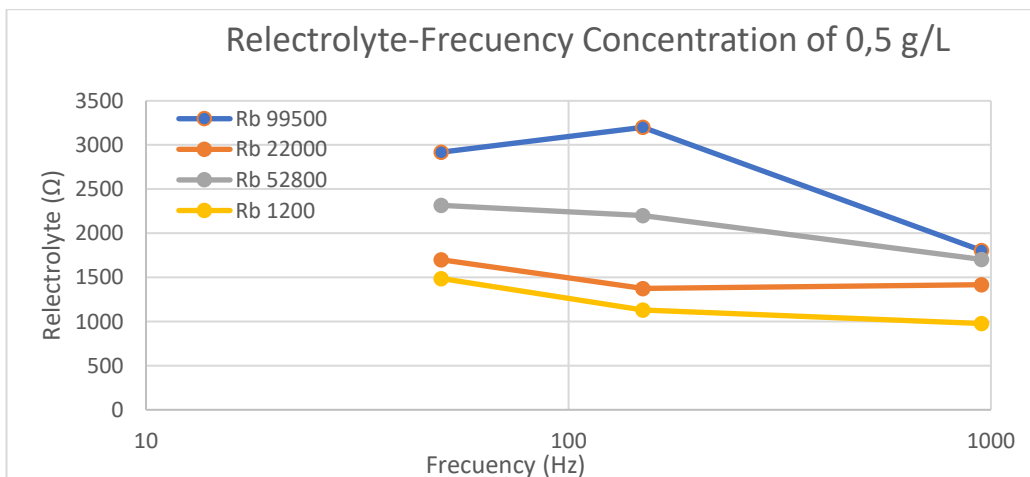
taken and taken to the laboratory. Using a scale, the corresponding amounts of table salt were added: 10 g for the first sample and 2.5 g for the second sample. For the minimum case of the feed, 10 % water with a previously prepared concentration of 0.5 g/L is diluted in pure distilled water.

- For the permeate it should be between 0.005 g/L and 0.2 g/L with a standard value of 0.02 g/L. In order to create these permeate samples, dilutions are made from those already created, i.e. for the maximum case of 0.2 g/L permeate a dilution will be made where 10 % will be water of 2 g/L previously created and 90 % will be distilled water. Focusing on the standard value of 0.02 g/L, the same procedure shall be repeated, but with 1 % of previously created 2 g/L water and 99 % distilled water. For the minimum case of permeate, 1 % water 0.5 g/L previously created and 99 % distilled water will be used.

Afterwards, other tests are carried out with different resistors in series with the electrolyte and at different frequencies. The area of the electrodes in contact with the electrolyte initially used was a simple bare wire on each side. The results obtained were as follows:



Graphic 4 Mole Plate Electrolyte resistance with Rb of 1 kΩ at frequency of 3000 Hz.



Graphic 5 Mole Plate Electrolyte resistance with  $R_b$  of 1 k $\Omega$  at frequency of 3000 Hz.

Since the electrolyte resistance values are all close to a few k $\Omega$ , the  $R_b$  to be selected will be close to these values. This was decided because when making the voltage divider we wanted to have a similar resistance rather than have a resistance 10 or 30 times greater and lose precision and resolution in the measurement.

The input signal selected from the function generator was a square signal of 2.5 V amplitude (5 V peak-to-peak) and this was the signal seen from the oscilloscope. In contrast, the input signal measured once the leads were inserted into the water changed. It was then assumed that the positive and negative charges in the water were deposited on the wires and so the input measurement was affected:

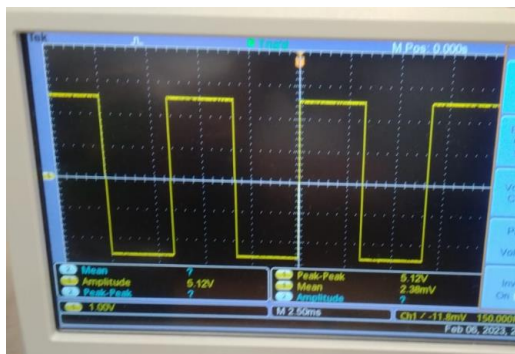


Figure 29 Input Signal outside the water

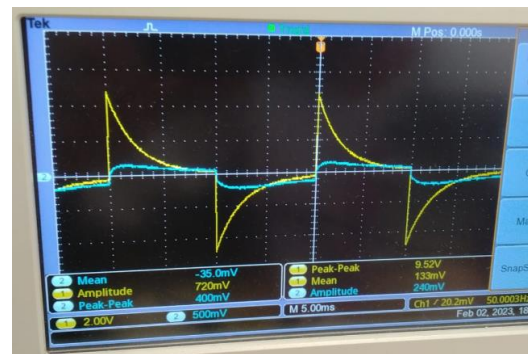


Figure 30 Input Signal Measured inside the water

In addition, we came up with the idea that the measurement should be made with 3 electrodes. The input electrode, the output electrode and the measurement electrode. This is because positive and negative ions are deposited on the input and output electrodes and this affects the measurement. For this reason, an intermediate measuring electrode should be used.

Moreover, it was very uncomfortable to take the measurements by putting the cables in the water continuously, so a mole plate was used, a more manageable surface that could be placed in the water and did not even need to be held. This plate was also a better simulated the case to be used outside the demo.

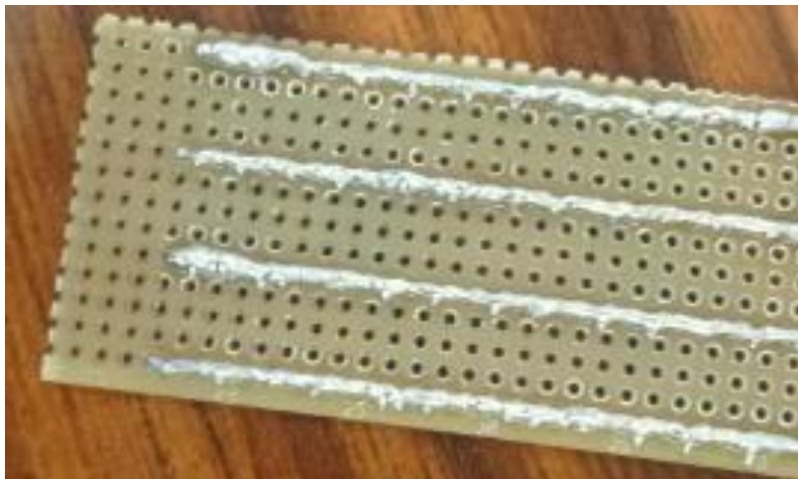


Figure 31 Mole Plate with 4 tinned rows (electrodes)

For the following tests, it was decided that the square signal would be provided by the BLE nano Keywish. In order to simulate the alternating signal, a square PWM signal from 0 to 5 V with a Duty Cycle of 50% and a frequency of 3000 Hz was generated through one pin of the device. The frequency has been increased to 3000 Hz because the higher the frequency, the faster the positive and negative ions change from electrode to electrode.

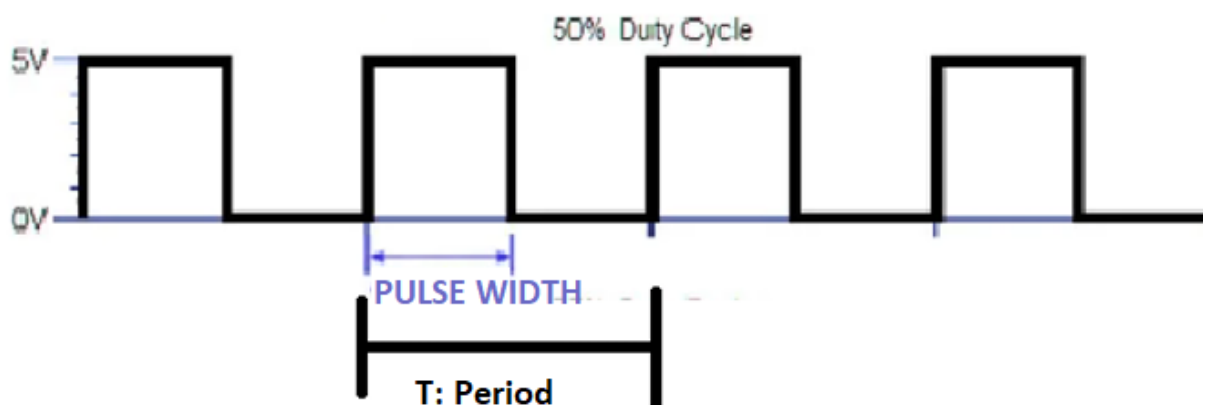


Figure 32 PWM Signal Example Further information in (Science Direct Pulse Width Modulation, n.d.)

To simulate the positive and negative voltage variation, the reference signal is not the GND of the microcontroller, it is a 2.5 V reference signal. In this way, the water referenced to 2.5 V "watches" a signal 2.5 V above (5 V of the input signal) and 2.5 V below (0 V of the input signal), thus preventing the positive and negative charges from being deposited on the electrodes. In this way, the ions are moved continuously.

To create the reference signal, a voltage divisor is created with 150  $\Omega$  resistors. The values chosen for these resistors had to be as low as possible so that the water would not inject current into them, but they had to have the minimum value of not exceeding the maximum current of one pin of the microphone, which in this case was 40 mA:

$$5 \text{ V} = 40 \text{ mA} \cdot R \rightarrow R = 125 \Omega$$

The two resistors should add up to at least 125  $\Omega$ , so two resistors in series of 150  $\Omega$ , i.e. 300  $\Omega$ , are selected to ensure that this limit is not reached.

The circuit used is as follows:

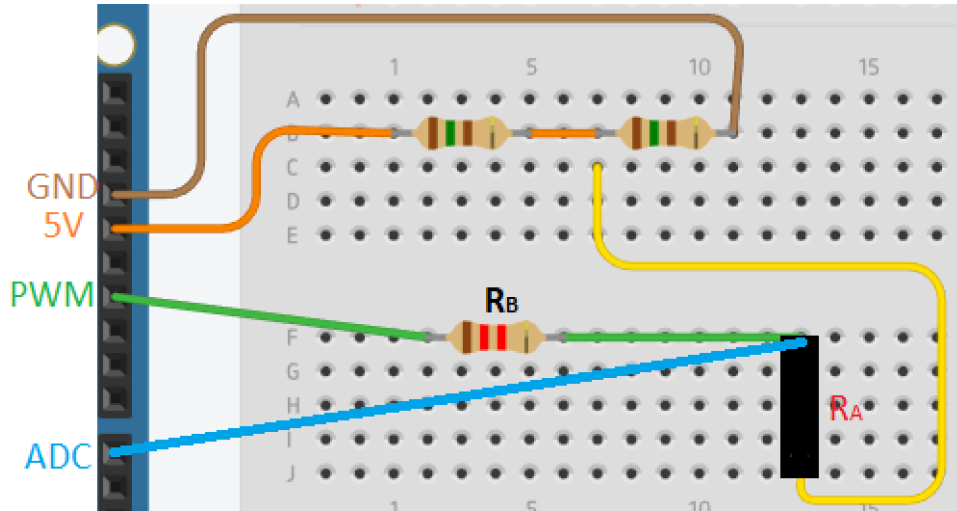
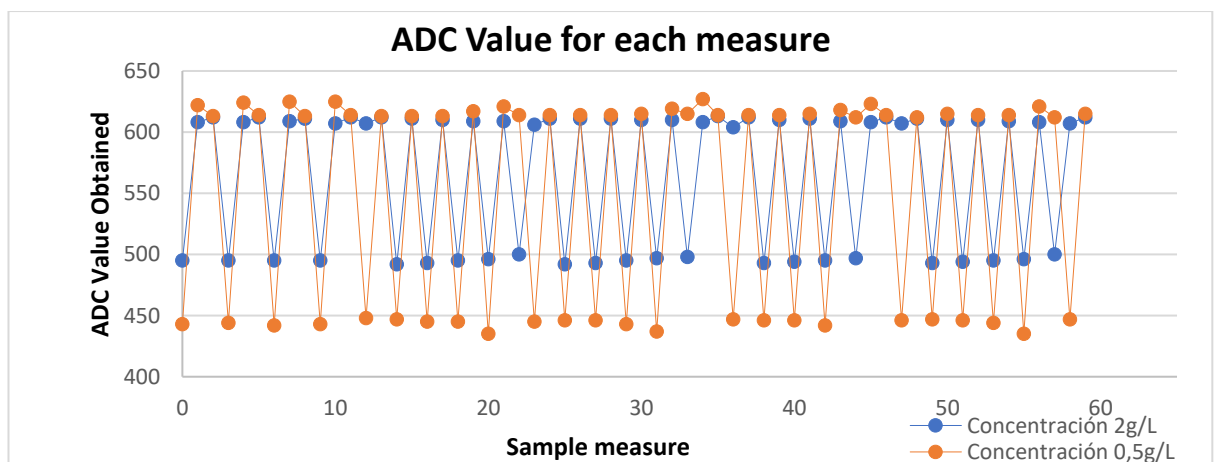


Figure 33 Used Circuit to check Salinity

The electrolyte used, i.e. the  $R_A$  used, is the one shown in [Figure 31](#) Mole Plate with 4 tinned rows (electrodes) Focusing on the rows of the mole plate, the first row of tin gives the square signal 0-5V, the last two are referenced to +2.5 V in order to have a voltage drop from +2.5 V to -2.5 V. The second row of tin is where the measurement is made, 4 rows had been prepared to make the reading between the second and third, but the test microcontroller available always reads with respect to its GND. Depending on how conductive (salty) the water is, the value of the measurement will be higher or lower. The next tests carried out, measured the voltage difference between the upper end of the electrolyte and the GND of the microcontroller from the same microcontroller. This was done for the two samples with different concentrations and the values obtained are as follows:

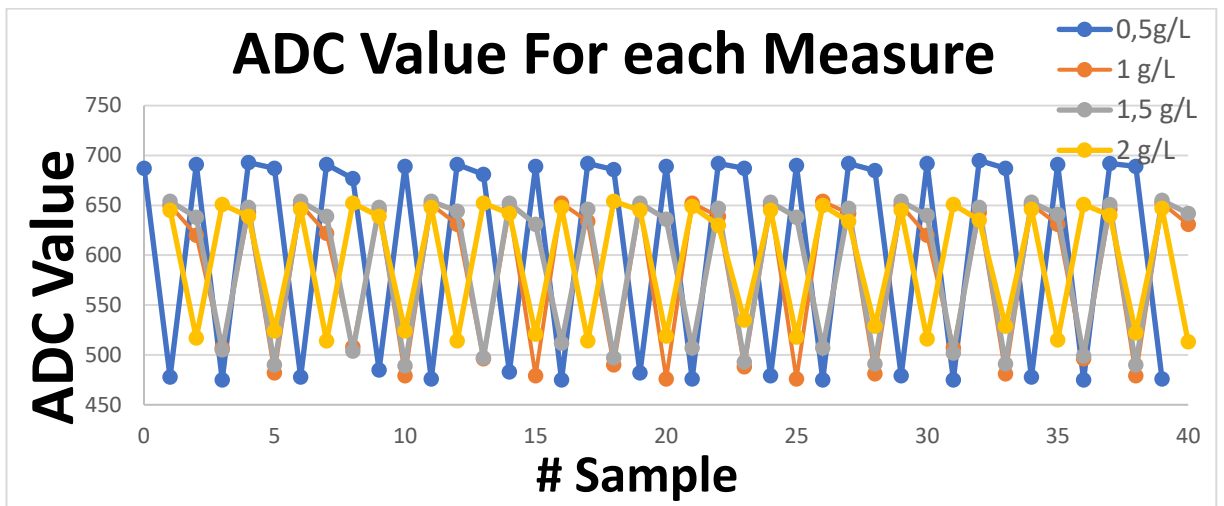


Graphic 6 Mole Plate Electrolyte resistance with  $R_B$  of 1 k $\Omega$  at frequency of 3000 Hz.

Looking at [Graphic 6](#) Mole Plate Electrolyte resistance with  $R_B$  of 1 k $\Omega$  at frequency of 3000 Hz. above, a noticeable difference in the two concentrations can be observed for the

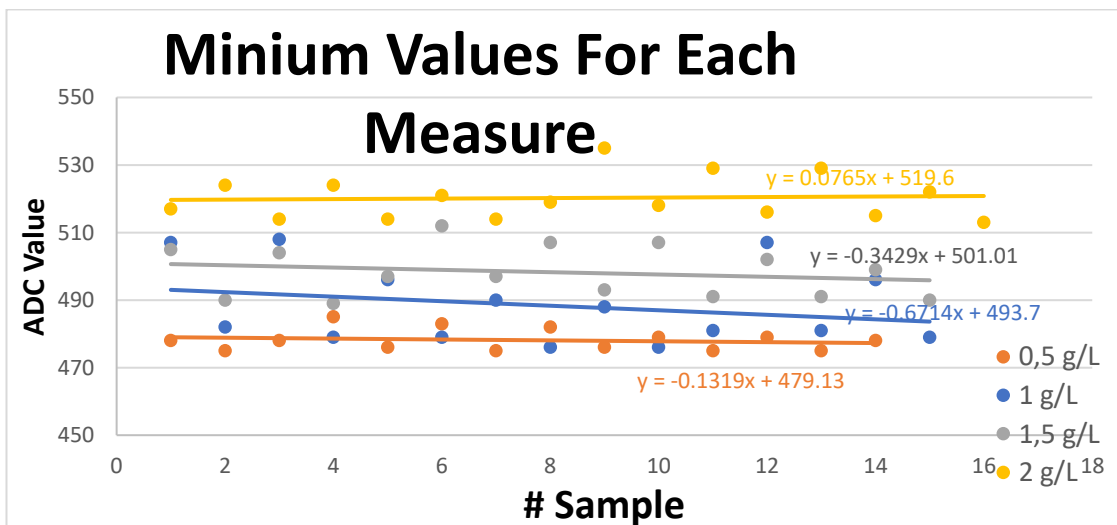
moments when the PWM is at 0 V. This way we can differentiate whether we are above the allowed value or not. In addition, four measuring vessels have been used to see what the trend of the concentration change from 0.5 g/L to 2 g/L is. Two more samples have been created which are 1 g/L and 1.5 g/L.

The 1 g/L sample was made by adding 50% of water without salinity and 50% with 2 g/L water in a millilitre measuring cup. To make the 1.5 g/L sample, 75 % of 2 g/L salt water and 25 % of non-saline water were added to a container. The results obtained with this test were as follows:



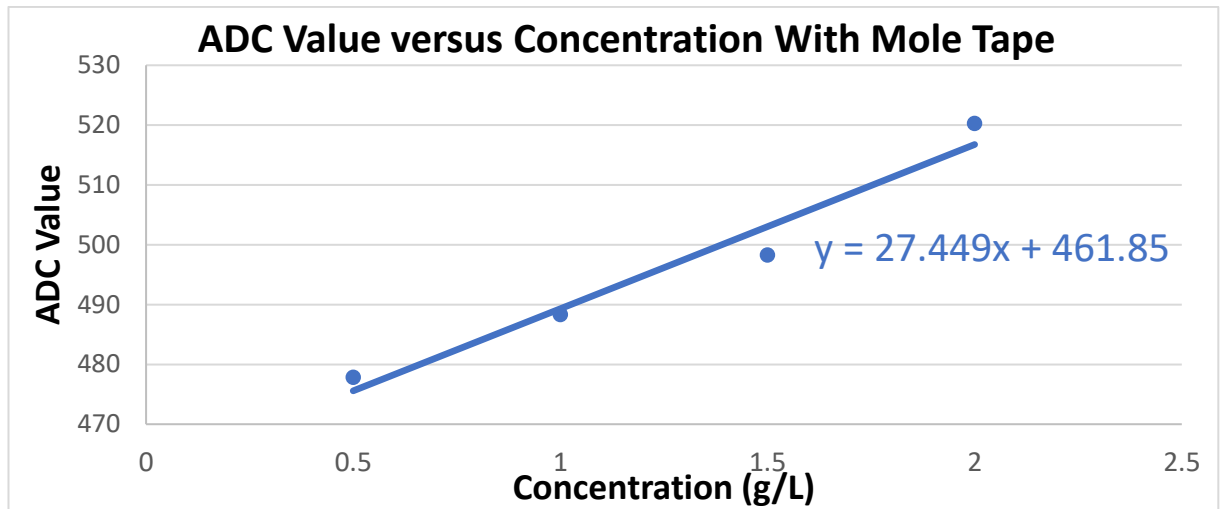
Graphic 7 Mole Plate Electrolyte resistance with Rb of 1 kΩ at frequency of 3000 Hz.

To differentiate between the different salinities, we focused on the value when the PWM signal was at low level, since there was at least a 40-LSBs difference between the lower and higher readings:



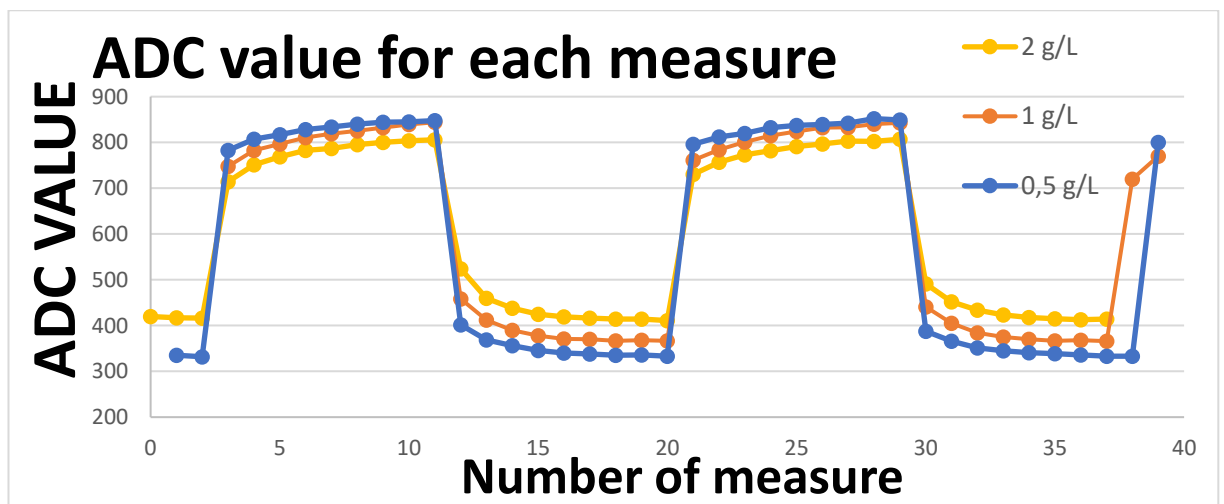
Graphic 8 Selecting Low Values Mole Plate Electrolyte resistance with Rb of 1 kΩ at frequency of 3000 Hz.

Looking at the results of the minimum values, these minimum values were averaged and a calibration line of reading as a function of salinity was created for this case of this electrolyte, the mole plate:



Graphic 9 Mole Plate Electrolyte resistance with Rb of 1 kΩ at frequency of 3000 Hz. Averages of all the measures for each concentration.

It should be noted that there were moments in when two high PWM levels or even 3 high PWM levels were found in a row. This was due to the fact that between conversion and conversion the data was sent, and this took up a time gap that could cause the PWM to go all the way down and return to the high level. To solve this, the code was modified to store the data in a table and then send it out and also the frequency was lowered from 3000 Hz to 500 Hz so that more readings could be taken at high and low level:



Graphic 10 Mole Plate Electrolyte resistance with Rb of 1 kΩ at frequency of 500 Hz.

As previously mentioned, there was a greater difference when we were at low PWM signal level, so it was decided to measure at low level. Also, knowing this, we could also try to increase the resolution by using the ADC's internal Reference Voltage of 1.1 V, as shown in Figure 34 Voltage reference. To do this, we it's necessary to adjust the R<sub>b</sub> value so that the minimum values are less than 225 (1.1 V-1023/5 V).

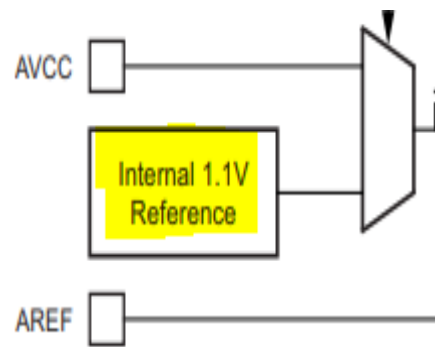


Figure 34 Voltage reference

Care should be taken when lowering the R<sub>b</sub> value as this may cause the maximum 40 mA of a pin to be exceeded.

DC current per I/O pin		40.0		mA
------------------------	--	------	--	----

Figure 35 Obtained from the Datasheet of the ATmega328P.

The circuit used is the one shown in [Figure 36](#). The two cases will be analysed, for when the PWM signal is at 0 V and for when it is at 5 V. For both cases the voltage drop is 2.5 V but the currents will go in the opposite direction, because in the first case it will be 5-2.5 V = 2.5 V while in the second case it will be 0-2.5 V = -2.5 V. Knowing this, we proceed to analyse the current flowing:

$$\frac{2.5}{R_b + R_{elec}} = I$$

Assuming the worst case where the water is extremely conductive, i.e. R<sub>elec</sub> is 0 considering the maximum 40 mA, the minimum R<sub>b</sub> should be 62.5 Ω.

Knowing that the water, in our case, will never reach this point and that at least in all cases we have seen that it was approximately always above 300 Ω. An R<sub>b</sub> resistor of up to 310 Ω could be used. Even so, we will try with an R<sub>b</sub> resistance of 510 Ω in order not to get close to this maximum value.

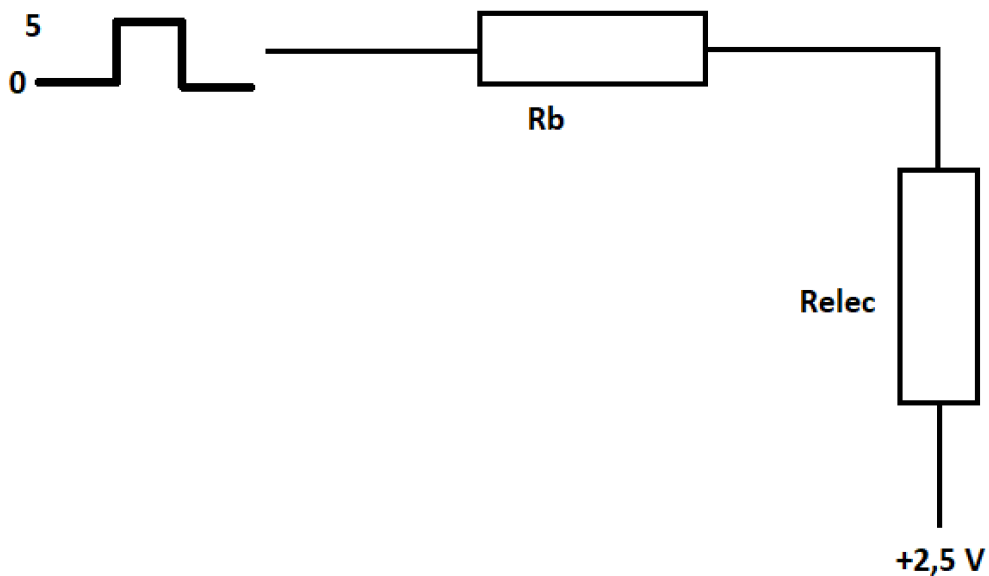
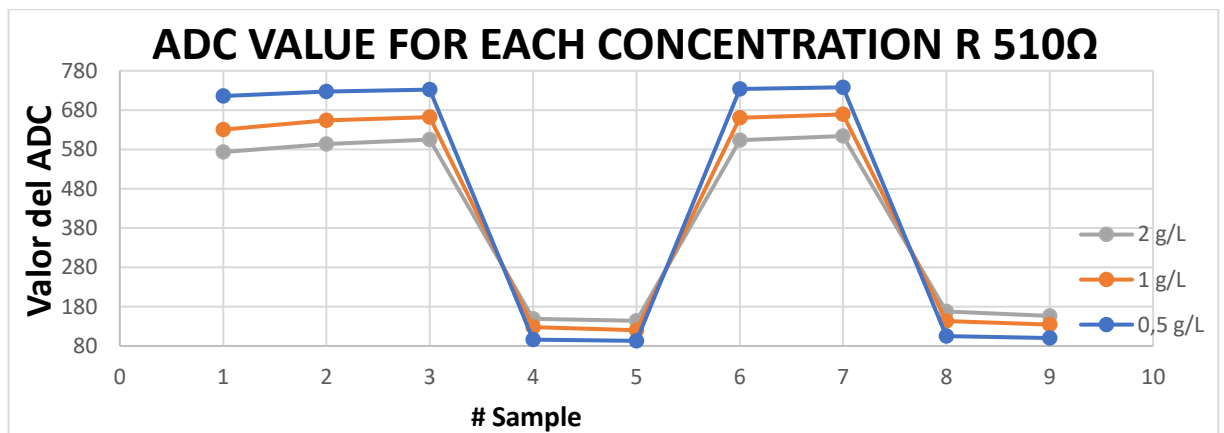


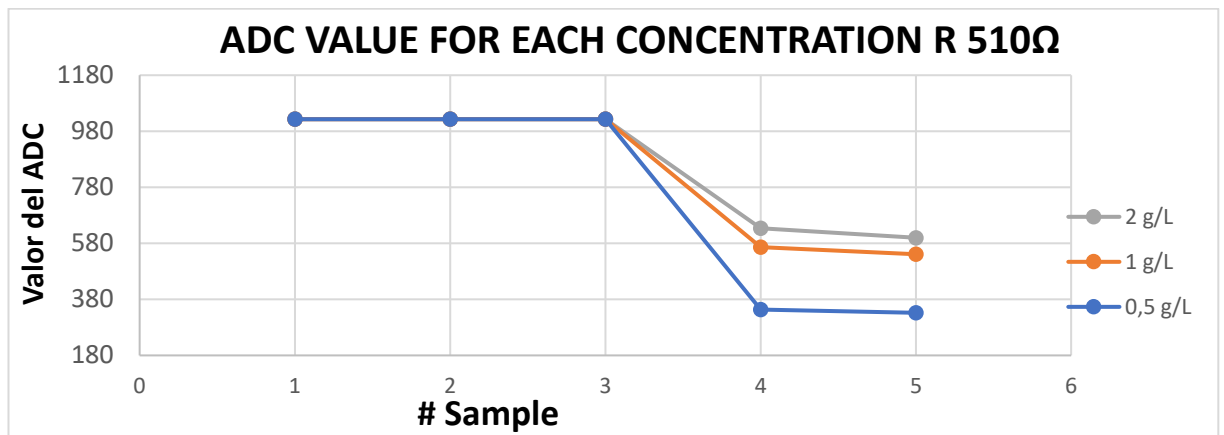
Figure 36 Circuit used

Knowing the resistance  $R_b$  selected, it is also decided to try to measure the reading on the input electrode, and check if in this way they can be differentiated and parameterise the behaviour, in this way we would save having to take 3 electrodes out of the Sensor Cover to make the measurements as we would measure inside the first one. The  $R_b$  used is  $510\ \Omega$ , the frequency is set to 1000 Hz because 2-3 values for each level (high and low) will be enough:



Graphic 11 Mole Plate  $R_b$  of  $510\ \Omega$  with an Input of 1000 Hz of Frequency reference of 1.1 V.

We have seen that with this resistance in the low levels of the PWM we are below 225 LSBs, we will make use of this resistor and will use 1.1 V as a reference voltage for the ADC. In this way at high level the ADC will read 1023, but as we will simply focus on the low levels, from these we will be able to differentiate which salinity we are at.



Graphic 12 Mole Plate ADC with a reference of 1.1 V.

Looking at these results, we have a 200 LSBs difference between 0.5 g/L and 1 g/L and an 80 LSBs difference between 1 g/L and 2 g/L. They can be differentiated correctly, and it has been seen after different measurements that they give very similar values. For this reason, it was decided that measurements could be taken at two electrodes, the reference electrode and the one in series with the Rb, taking the readings on the inside of the latter.

With the circuit of the [Figure 36 Circuit](#) used, we can differentiate the different salinities and by using the 510 Ω resistor Rb, the ADC can be referenced to 1.1 V for better resolution. Despite of this, we are making use of 2 resistors of 150 Ω to make this divider and current is flowing through an extra pin of the microcontroller to make this divider, current that is divided and a part of it circulates through the electrolyte. This is why, it is proposed to make use of two PWM signals 180° out of phase. One PWM signal will enter through the Rb as previously done and the other will be replaced by the 2.5 V reference signal used. In this way, the water ions will continue to switch from electrode to electrode because they will continue to see a voltage difference between them.

Our microcontroller has a register where, by modifying the appropriate bits, an inverted PWM signal can be output:

1	0	Clear OC1A/OC1B on compare match, set OC1A/OC1B at BOTTOM (non-inverting mode)
1	1	Set OC1A/OC1B on compare match, clear OC1A/OC1B at BOTTOM (inverting mode)

Figure 37 Datasheet Screenshot

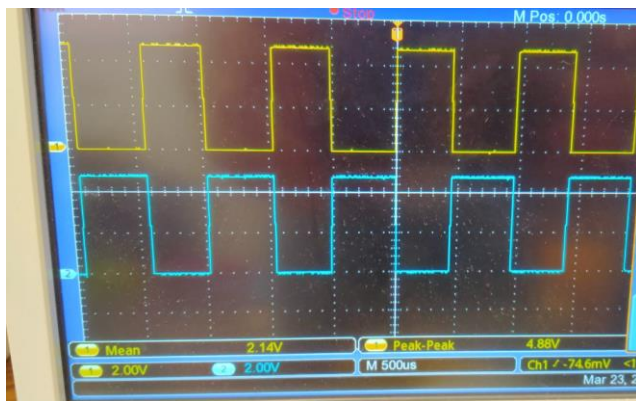


Figure 38 PWM Signals provided by the microcontroller 180 ° misaligned.

Having the two PWM signals available, it was decided to compare the results obtained for the two cases (referenced to 2.5 V and with the two PWM signals) to see whether it is possible to use this double PWM signal or whether it is necessary to stick with the voltage divider.

To start, the behaviour of the new circuit shown in [Figure 39](#) Circuit is studied when the R<sub>b</sub> values are changed:

$$V_o = PWM1 \cdot \frac{R_{elec}}{R_b + R_{elec}}$$

Equation 11

$$V_o = PWM2 \cdot \frac{R_b}{R_b + R_{elec}}$$

Equation 12

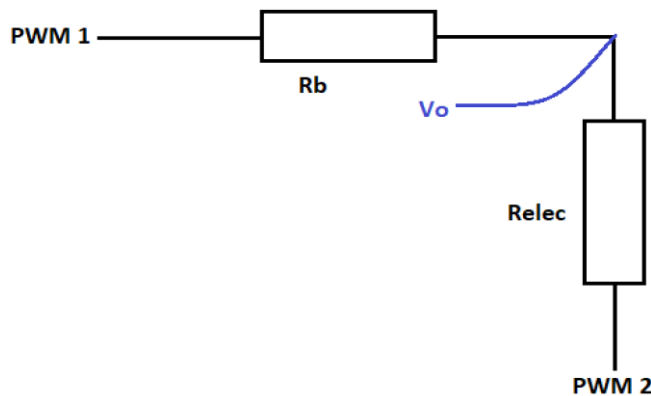


Figure 39 Circuit proposed with 2 PWM.

In case of the [Equation 11](#), lowering the R<sub>b</sub> will make the output closer to 5 V when the PWM is at high level, and by the time it is at low level it will be at 0 for sure because of what we multiply by 0. If we raise the R<sub>b</sub> for the first case, the V<sub>o</sub> will go down if we are at the high level of the PWM.

In case of the [Equation 12](#), lowering the R<sub>b</sub> will make the output closer to 0 when the PWM is high and closer to 0 when the PWM is low. If we raise the R<sub>b</sub> value, V<sub>o</sub> will be closer and closer to 5 V when the PWM 2 signal is at high level.

Before selecting the  $R_b$  value, let's understand how the measures experiment will work. Supposing that the  $R_b$  value is set, the variable value is the  $R_{elec}$ . If the conductance of the electrolyte is very low (low salinity), the  $R_{elec}$  value will be very big; in this case the  $V_o$  read when PWM1 is at 5 V will be very low near 0 V and the  $V_o$  read when PWM2 is at 5 V will be near 5 V. Adding salinity which implies increasing the conductance, will make that the  $R_{elec}$  value decreases. This means that the  $V_o$  read when PWM1 is at 5V will start increasing and the  $V_o$  read when PWM2 is at 5 V will start decreasing. If we continuously decrease the  $R_{elec}$  value, there will be a point in which both values for PWM1 and PWM2 will be the same and then the  $V_o$  from PWM1 at High will have a smaller value than  $V_o$  from PWM1 at High. If we arrive to a very conductive electrolyte, the  $V_o$  when PWM1 is at 5 V will be near 5 V and the  $V_o$  read when PWM2 is at 5 V will be near 0 V. In the Figures below we see the example with two  $R_b$ : the lower  $R_b$  of 2200  $\Omega$  and 50000 $\Omega$ .

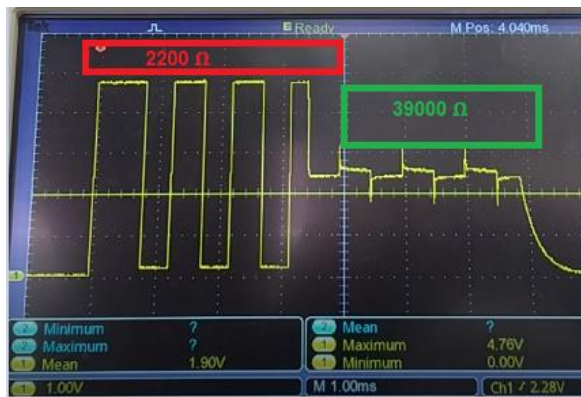


Figure 40

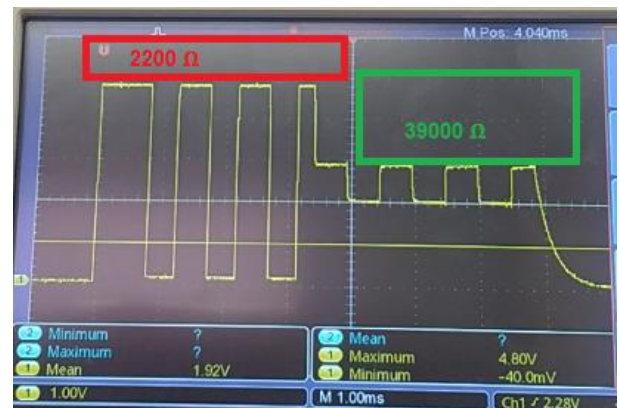


Figure 41

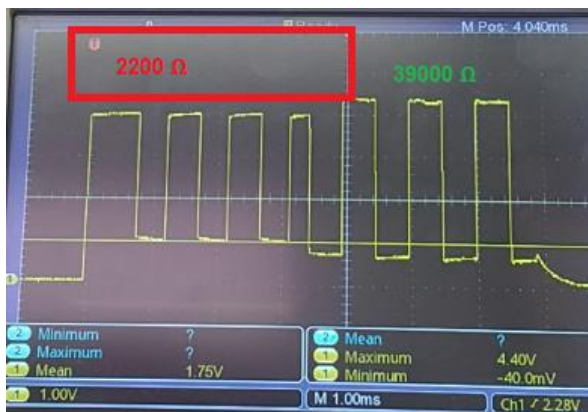


Figure 42

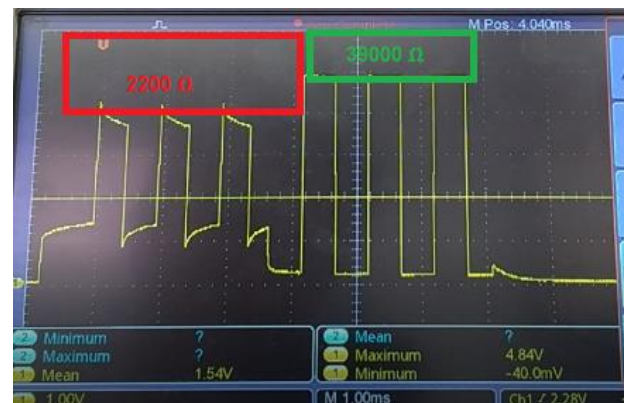
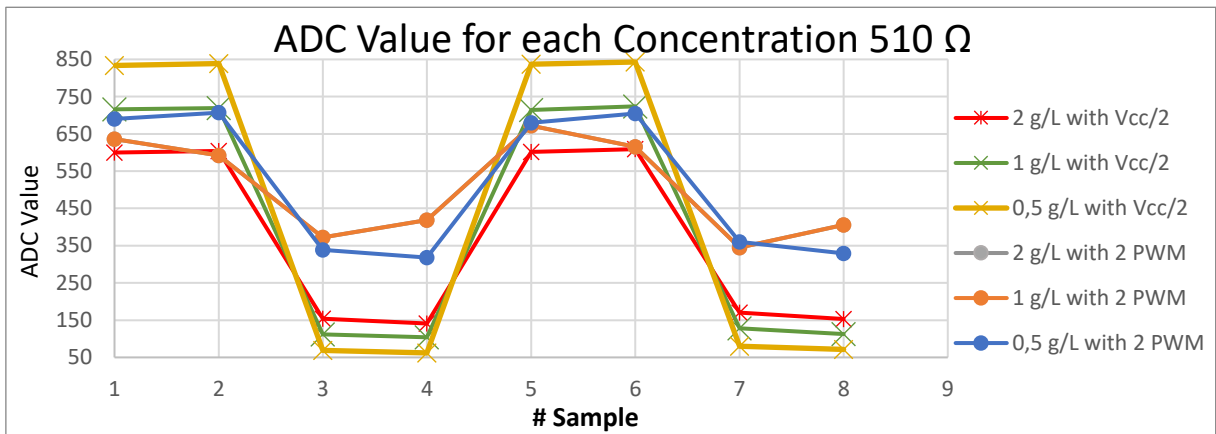


Figure 43

Initially here, with the value of the 39000  $\Omega$  we are at the point of having the same  $V_o$  when both PWMs are at high. With the 2200  $\Omega$  we are with  $V_o$  from the Equation 11 at higher value and  $V_o$  from the Equation 12 at lower value. In the [Figure 41](#) salt has been added and  $R_{elec}$  value has decreased, in this case the  $V_o$  from Equation 12 has increased meanwhile  $V_o$  from [Equation 11](#) decreased. The same happens with the  $R_b$  with the red colour, case the  $V_o$  from [Equation 12](#) has increased meanwhile  $V_o$  from Equation 11 decreased. In the [Figure 42](#) we can observe in the red  $R_b$  how the  $V_o$  values from each PWM swapped its positions and now the  $V_o$  from the Equation 11 has a lower value than the one in the Equation 12.

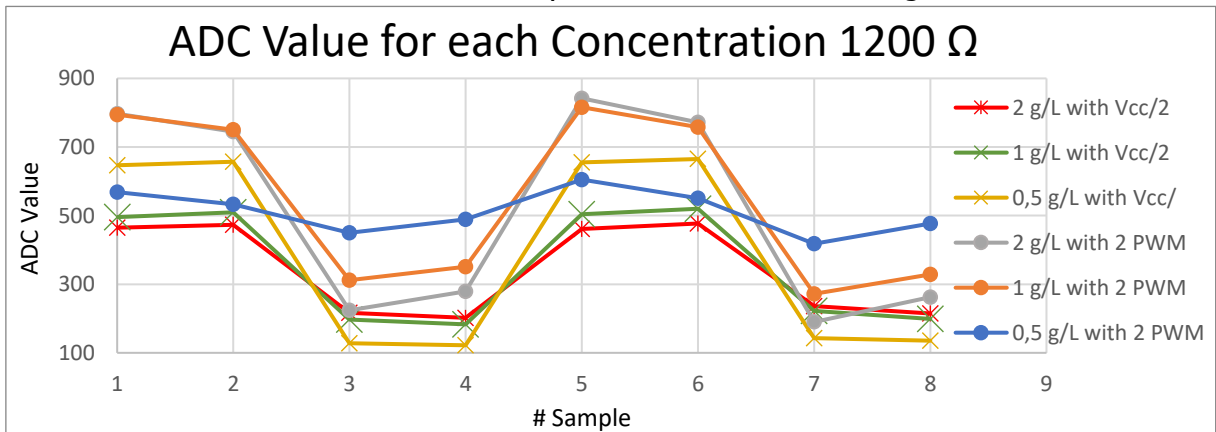
With these considerations in mind, tests are made with different values of  $R_b$  to see which value is the most qualified. It should be remembered that the  $R_b$  value cannot be lowered below  $300\ \Omega$  so as not to have excessive current through that pin, and even then it will not go below  $500\ \Omega$ . It should also be pointed out that the higher the value of  $R_b$ , the better it will be for consumption because the current that will circulate will be lower.

Next, the comparisons of different resistor values are shown for the cases where 2 PWMs are used, one inverted with respect to the other [Figure 39](#) Circuit and the case of the  $V_{cc}/2$  reference [Figure 36](#) Circuit used, the ADC reference is 5V.



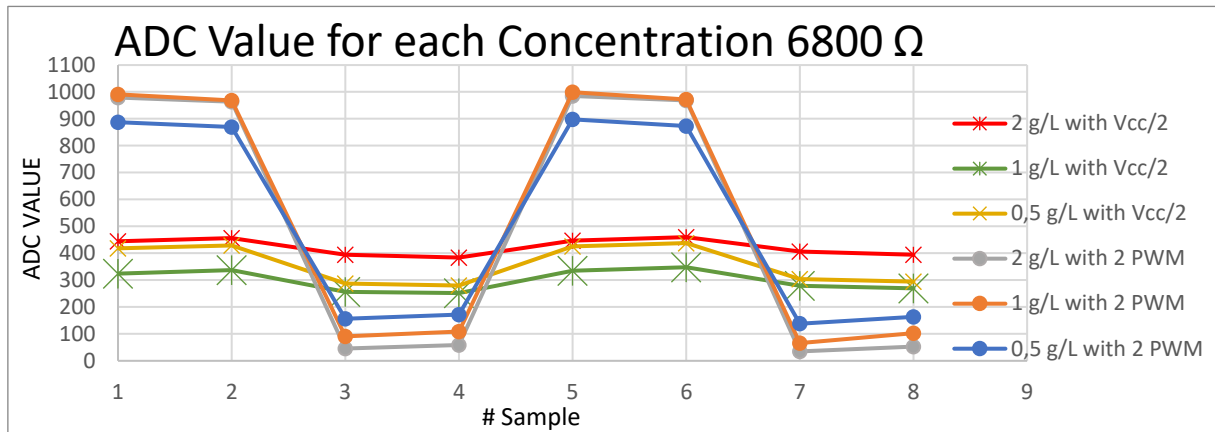
Graphic 13 Using Mole Plate,  $R_b$  of 510 with ADC reference of 5 V

In the [Graphic 13](#) Using Mole Plate,  $R_b$  of 510 with ADC reference of 5 V it can be seen how the different salinities can be better distinguished when the  $V_{cc}/2$  reference is used, whereas with the two PWM signals it would be difficult to differentiate the salinities because there are times when they are not distinguishable at all.



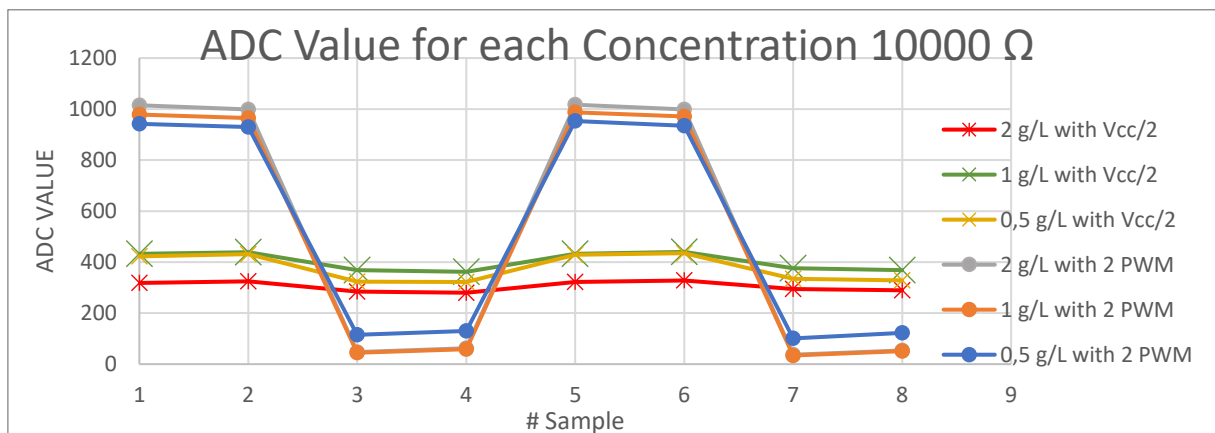
Graphic 14 Using Mole Plate,  $1200\ \Omega$   $R_b$  case with ADC reference 5 V.

With the  $R_b$  of  $1200\ \Omega$ , the 3 salinities are still better differentiated using the  $V_{cc}/2$  reference than using 2 PWM signals. Even so, it is worth noting that for the 2 g/L and 1 g/L salinities, differences of up to 80 LSBs are already beginning to appear.



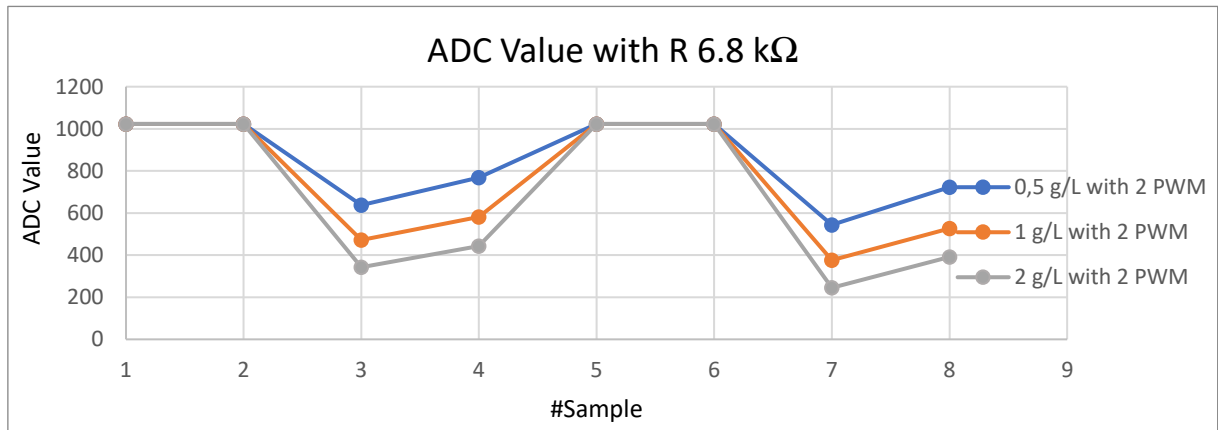
Graphic 15 Using Mole Plate, Rb case of 6800 Ω with ADC reference 5 V.

Using the Rb of 6800 Ω, the salinities do not differ using the Vcc/2 reference. If we change PWM 2 to Vcc/2, and put a resistor Rb much bigger than Relec, the value will be around Vcc/2. By looking at the signals obtained using the two PWMs, the three salinities can be differentiated by looking at the area where PWM1 is at high level and PWM2 is at low level. In addition, these low-level values are below 225 LSBs so you can make use of the 1.1 V ADC reference to gain resolution.



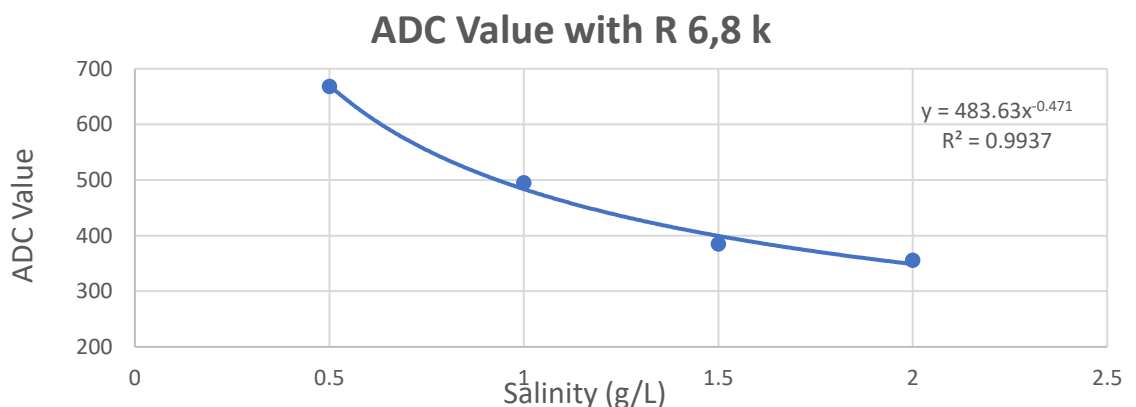
Graphic 16 Using Mole Plate, Case of 10,000 Ω Rb with ADC reference 5 V.

In the Graphic 16 Using Mole Plate, Case of 10,000 Ω Rb with ADC reference 5 V. we find ourselves in a similar case to the previous one. Despite of that, in this case, for low levels, waters with salinities of 2 g/L and 1 g/L have very close values, which is why it was decided to use the Rb of 6800 Ω.



Graphic 17 Using Mole Plate, Case of Rb 6800 Ω with ADC reference 1.1 V.

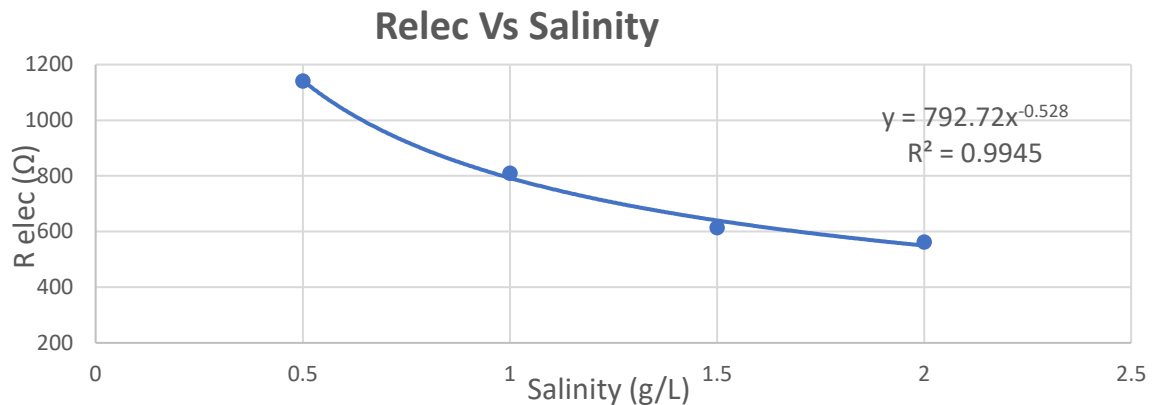
Based on these results, measurements are made for 4 different salinities and averaging the first four measurements of low values in order to be able to see the behaviour of the salinity. Samples of 0.5 g/L, 1 g/L, 1.5 g/L and 2 g/L will be used:



Graphic 18 Using the Mole Plate and focusing on the Feed Values 6800 Ω.

With the results of Graphic 18 Using the Mole Plate and focusing on the Feed Values, it can be seen that these are differentiable salinities and the behaviour of the conductivity measurement is parabolic with a small error. It should be specified that all readings higher than 700 will be salinities lower than 0.5 g/L. The following graph shows the resistive behaviour in this case:

For the permeate, the same structure will be used, modifying the corresponding Rb value for some values that the permeate should give, it is necessary to work with lower concentrations: between 5 mg/L and 200 mg/L with a standard value of 20 mg/L. Once the samples have been prepared as specified after Equation 10, the Rb values to be selected are much higher values because the salinity will be much lower. Approximately, and using Graphic 19, calculate how much the Resistance of the electrolyte will be approximately for the 0.2 g/L and 0.02 g/L cases. The Graphic 19 has been obtained transforming the ADC value received into Voltage and then doing a Voltage divider with the Rb of 6800 Ω.



Graphic 19 Relec value vs Salinity for Rb of 6800.

The Rb is sought for the lower salinity case as it will be more restrictive:

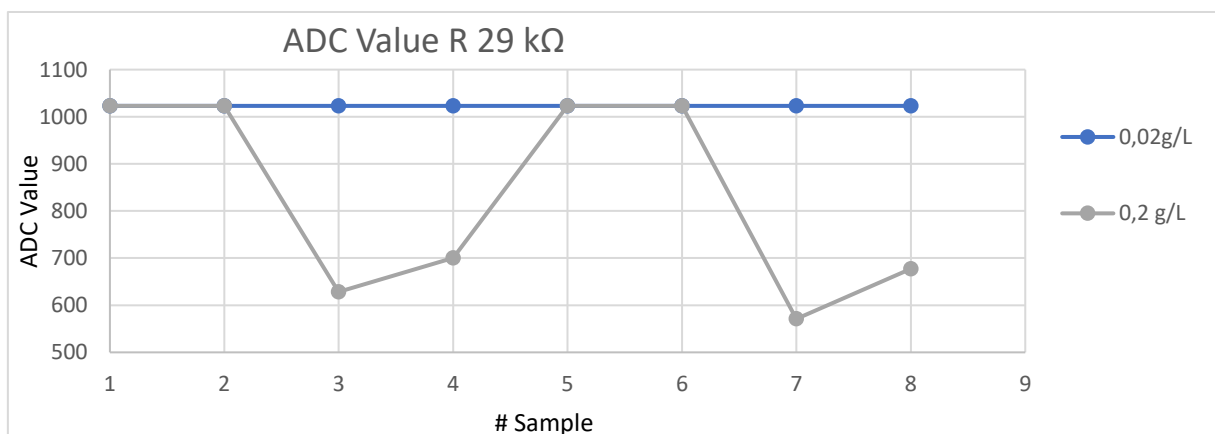
$$\text{Relec} = 792.72 \cdot 0,02^{-0.528} = 6258.26 \Omega$$

Equation 13

To calculate Rb, an approximate Vo value of 0.25 V or 233 LSBs be considered (this is chosen because it must be a value between 0 and 1.1 V and close to 0 but leaving some margin). Isolating from Equation 9 Rb we obtain:

$$R_b = \text{Relec} \cdot \left(1 - \frac{0.25 \text{ V}}{5 \text{ V}}\right) \cdot \frac{5 \text{ V}}{0.25 \text{ V}} = 118906 \Omega$$

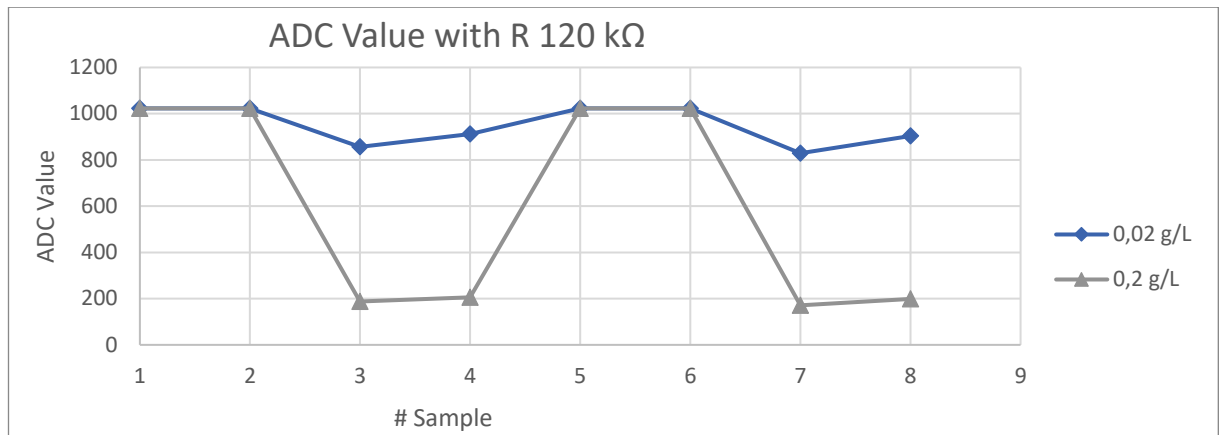
Tests are carried out with similar Rb and the Feed procedure is repeated with the new samples and using the double PWM configuration. The calculated value is approximately 120 kΩ but different tests with different Rb have been carried out:



Graphic 20 Permeate case with mole plate and Rb 29 kΩ.

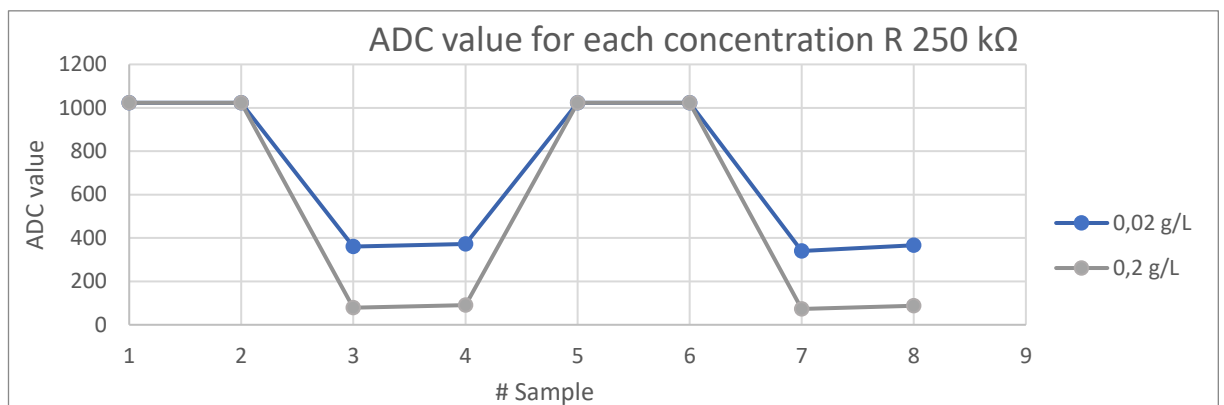
With this 29 kΩ Rb value, it is observed that for salinity of 0.02 g/L the ADC reading is always at 1023 at both low and high levels. This means that the voltage is higher than 1.1 V in all cases. This is why the Rb is increased until a value below 1023 can be read at low level. The Rb value was increased to 100 kΩ and the value was still a constant 1023. Then 120 kΩ

was tested and a value below 1023 could be read at low level and the two concentrations could be differentiated:



Graphic 21 Permeate case with mole plate and Rb 120 kΩ.

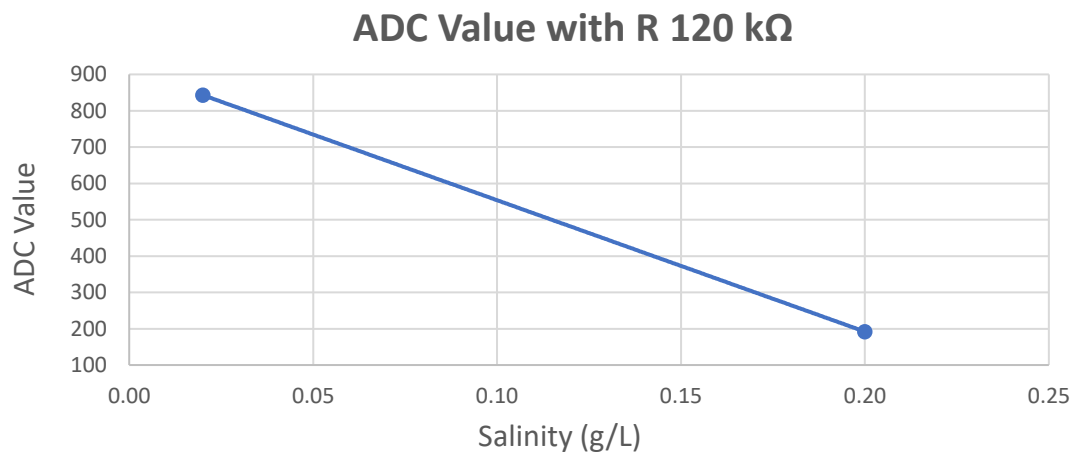
Next, the Rb got increased up to 250 kΩ, but the difference of LSBs between one salinity and the other is considerably smaller so the resolution is smaller:



Graphic 22 Permeate case with mole plate and Rb 250 kΩ.

As we can observe the difference between both salinities is from 300 LSB with RB of 250 kΩ, meanwhile with the Rb of 120 kΩ the difference is 700 LSBs

Based on these results with 120 kΩ, measurements are made for 2 different salinities and averaging the first four measurements of low values in order to be able to see the behaviour of the salinity. Samples of 0.02 g/L, 0.2 g/L will be used:



Graphic 23 Using the Mole Plate and focusing on the Feed Values 120.000  $\Omega$ .

### **6.4.2 DEMO Salinity Measure, Fiel Test**

Once the final CNC mechanised Sensor Cover with its electrodes was finished, we had to check which was the most suitable Rb to add in series with it as we can see in [Figure 53](#) Final Sensor Cover with 2 electrodes..

We decided to test with the same electrodes the conductance of the feed and the product, that's why we had to select two Rb (one per case) and connect it properly. It was also important to take into account that we might configure the microcontroller pins connected to the electrodes as outputs or inputs when it's necessary, i.e. defining as outputs when the PWM is on and as inputs when the PWM is off.

After repeating the last tests of [DEMO Salinity Measure, Laboratory](#) with different Rb and with the salinities specified by The Company. With these new electrodes, it was decided that the most suitable series resistance for the feed was the 1200  $\Omega$  (measured 1205  $\Omega$ ) for the feed case and 39000  $\Omega$  (measured 38500  $\Omega$ ) for the product. Once this was decided, the prototype was ready to measure conductivity in the pressure vessel.

## **6.5 Final Results Inside Pressure Vessel**

The final Test had the purpose of measuring Feed Conductance, Product conductance and Temperature, the circuit used is [Figure 54](#) Final Schematic Used. After measuring the conductances with the Sensor Cover, the main point was to make a calibration using the conductivity measure of The Company sensors as references. This should be possible if the company conductimeters were data-logging their measurements. Unfortunately, this wasn't possible because The Company's sensors weren't saving the data measured.

The batteries used were 3 A544-4LR44-6V, soldered in parallel so that the battery could last longer. The experiment lasted 1 hour and 35 minutes, the first 20 minutes the water was product water, then we started to increase the salinity to feed parameters water and when we had an hour of experiment we added no-salty-water to decrease the conductance. The results of temperature obtained in the experiment are shown in [Graphic 27](#) Temperature Measured By The Sensor Cover.

We started at high temperature as the device was stored at high ambient temperature. Then, when introduced in the pressure vessel and the experiment started, temperature decreased because the water starting flowing and caused a cooling effect. Afterwards, the temperature increased while the water was recirculating. At the end, we added the new fresh water and, therefore, temperature decreased again ([Graphic 27](#) Temperature measured by the Sensor Cover).

Focusing on the conductivity-conductance relation, it was not possible to create this calibration because The Company Conductimeters weren't connected with The Company system and they weren't saving the measurements done. However, some measurements were manually noted as a reference, so that some comparisons could be done. It's important to point that all the data received from the Sensor should be interpreted and post-processed.

During the first 20 minutes the sensor could follow the results of the product, that were [Graphic 28](#) Product Measures in Final Test.

Moreover, during the Feed Measurement period, the Sensor measures were [Graphic 31](#) Feed Values measured.

There's a Graphic that shows the intermediate step from low salinities to high salinities, this is the [Graphic 30](#) Feed Values measured with the change of PWM high at minute 30. It is important to note that The Company's sensor measured Conductivity meanwhile our sensors were measuring a Voltage which can be afterwards transformed into conductance.

In order to understand [Graphic 28](#) Product Measures in Final Test, and [Graphic 31](#) Feed Values measured., it is necessary to remind the schematic circuit from [Figure 39](#) Circuit proposed with 2 PWM., the equations [Equation 11](#) and [Equation 12](#) and the brief explanation there is after this equations with the help of [Figure 40](#), [Figure 41](#), [Figure 42](#) and [Figure 43](#). At first, in the [Graphic 28](#) Product Measures in Final Test we can observe that the ADC value in the permeated increased when the salinity increased, that was because initially, the Rb had a lower value than Relec and this meant that we were initially measuring with 5 V at PWM1 we are at very high values (5 V – 1023 LSBs) and when measuring with PWM2 at 5 V, we were

at low values (values below 1023 read by the ADC which is referenced at 1.1 V), this all implied that we were initially with the Equation 12.

An important comment in order to have a better understanding is that increasing the salinity alludes to decreasing the Relec value, checking the Equation 12, this means that  $V_o$  value increases. Meanwhile  $V_o$  value from Equation 12 increases, the  $V_o$  value from Equation 11 decreases because the Relec multiplying in the numerator decreases. If salinity gets increased slowly, there will be a point in which both  $V_o$  values from both equations will be the same and then  $V_o$  value from Equation 11 will continue decreasing and  $V_o$  value from Equation 12 will continue increasing. Once the point has reached here, the numbers read by the ADC will be the values from Equation 11.

However, this couldn't be appreciated in this Graphics because we added too much salt the first time and it suddenly passed from the case of the Equation 12 to the change of the Equation 11. Fortunately, we were able to appreciate it with the Feed measurements.

Focusing in the Feed measures, in Graphic 30 Feed Values measured with the change of PWM high at minute 30. at start, the value  $V_o$  received is from the Equation 11 and because of that conductivity increase, the value read increases. However, when another step of salt is done in the pressure vessel at the minute 30, we observe that the ADC value started to decrease constantly each time the conductivity increased. That's because with this step at minute 30 the value read by the ADC wasn't the value of the Equation 12 and was the  $V_o$  value of the Equation 11.

Next, values from the ADC will be interpreted as conductances, to do it first we need to interpret which is the resistance of the electrolyte for each ADC value, to do it we will isolate Relec in Equation 11 and Equation 12. Let's remind that product Relec will be obtained from Equation 12 and feed Relec resistances will be obtained from Equation 11 in this experiment:

$$R_{elec} = \frac{\left(\frac{V_o}{V_i} \cdot R_B\right)}{1 - \frac{V_o}{V_i}} = \frac{\left(\frac{ADC \cdot \frac{1.1 V}{1023} \cdot R_B\right)}{1 - \frac{ADC \cdot \frac{1.1 V}{1023}}$$

Equation 14 Feed Relec in this test

$$R_{elec} = \frac{\left(\left(1 - \frac{V_o}{V_i}\right) \cdot R_B\right)}{\frac{V_o}{V_i}} = \frac{\left(\left(1 - \frac{ADC \cdot \frac{1.1 V}{1023}\right) \cdot R_B\right)}{\frac{ADC \cdot \frac{1.1 V}{1023}}$$

Equation 15 Product Relec in this test

Where  $R_B$  is 1205  $\Omega$  for the Feed and 38500  $\Omega$  for the Product. Once the Relec is known, the Conductance is:

$$\sigma = \frac{1}{R_{elec}}$$

The final Graphics obtained transforming this ADC value into Conductance are [Graphic 29](#) Product Values measure in Conductance. and [Graphic 32](#) Feed Values measure in Conductance..

In this both Graphics it observed how the sensor cover conductance's measurements have a similar behaviour to the conductivity measurements of The Company's conductimeter. The values are different because with our Sensor Cover we read conductance ( $\mu\text{S}$ ) meanwhile with their conductimeter they read conductivity ( $\mu\text{S}/\text{cm}$ ). The difference is that the conductimeter value given is dependent value of its electrodes surface area. Our value is not dependent of that surface area. In order to know the conductivity value, the section of the electrodes and the path the electricity went through as we can observe in the next equation and the next Figure:

$$\frac{1}{G} = \frac{1}{\sigma} \cdot \frac{l}{A} \rightarrow \sigma = G \cdot \frac{l}{A}$$

Where G is the Conductance, l the length (supposing a prismatic Geometry as in Figure 44 Conductor Extrated from Wikipedia and A the surface Area).

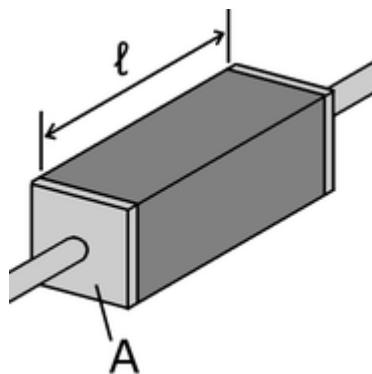


Figure 44 Conductor Extrated from Wikipedia

In our case, we can't be certain of the path that the electricity has carried out through the water, because of that we maintain as our final value the conductance read.

To conclude, we also got the RSSI value every time a packet was sent, and the Graphic done with the known RSSI and the Conductivity of The Company's Conductometers is [Graphic 33](#) RSSI versus Conductivity in final test. Once the salty water enters to the pressure vessel, the RSSI decreases form -60 dBm to -80 dBm. However, the RSSI value is around -70 dBm and -85 dBm during all the experiment no matter how much salt is added.As it was supposed, once the salty water is circulating through the Pressure Vessel, the data sent by the device doesn't pass through the salty water, it passes through the Endcap that is made of plastic.

## 7. Economic Considerations

The total prize of the material for this final prototype is:

Description	Unity measure	Unitary Price	N. Units	Price
<b>Materials</b>				
BLE Nano Keywish	u	13.60	1.00	13.60 €
Resistor 1200 Ω of 1/4 W	u	0.20	1.00	0.20 €
Resistor39000 Ω of 1/4 W	u	0.20	1.00	0.20 €
Prototyping board (Mole Tape) 27x27 cm	1	1.69	1.00	1.69 €
Heat Shrink tube ø 3 mm	m	3.00	0.50	1.50 €
Conductive Cable 1.5 mm 500 V	m	0.92	0.50	0.46 €
A544-4LR44 6V Battery of 140 mAh	u	3.42	3.00	10.26 €
PVC Sensor Cover	u	1.00	250.00	250.00 €
<b>Total Cost:</b>				<b>277.91 €</b>

It is necessary to consider that this is the prize of the final prototype Sensor Cover used. However, other materials were bought and tested but weren't implemented in this final prototype. such as button batteries, BLE nano Arduino devices, PLA 3D filament.

## **8. Conclusions**

The main objective of this project was to demonstrate the feasibility of measuring temperature and conductivity within a Reverse Osmosis column, and subsequently transmitting the collected data outside the pressure vessel for retrieval by a receiver.

To begin, the most suitable wireless technology was carefully selected. Through testing conducted within The Company's premises, it was determined that Bluetooth Low Energy (BLE) emerged as the successful choice. Considering the test results and the priority placed on BLE, this wireless technology was ultimately selected.

Moving forward, temperature measurement was carried out using a 100k NTC (Negative Temperature Coefficient) sensor, which exhibited a tolerance of 1 °C. Laboratory tests were performed to measure temperature, wherein the temperature readings were captured by a mobile device equipped with the project's dedicated application. These values were found to align with the readings from the room's temperature sensor. Subsequently, a prototype was subjected to test in The Company's building, enclosed within a 3D-printed design. Initially, temperature measurement was successfully accomplished; however, after a few minutes, water inadvertently infiltrated the device. Consequently, a PVC Sensor Cover was created to prevent water ingress into the device.

Regarding conductivity measurement, laboratory experiments involved the use of two electrodes (screws were employed in this case) to assess conductance values. Notably, the salinity of the feed water and product water varied for each test. In The Company's building, the device was introduced into the PVC Sensor Cover and subsequently placed in the pressure vessel. The experiment successfully distinguished between product water and feed water, while also allowing for the adjustment of conductance values as salinity levels were incrementally increased and decreased. It is important to note that The Company possessed conductivity sensors, which provided conductivity values independent of electrode area. The intention was to compare these values with our conductance values, which were dependent on electrode area, thereby establishing a calibration between the conductance and conductivity. However, the conductimeters used by The Company did not have the data-logging connected, rendering it impossible to create the desired conductivity-conductance relationship using our sensor's surface area.

## **9. Bibliography**

- [1] Arduino. (n.d.). *Arduino*. Retrieved from <https://www.arduino.cc/>
- [2] EATON Powering Business Worldwide. (n.d.). *NRBG Glass sealed radial lead NTC thermistor*.
- [3] EE POWER. (2023, May). *EE POWER NTC Thermistor*. Retrieved from <https://eepower.com/resistor-guide/resistor-types/ntc-thermistor/>
- [4] Endress+Hauser. (2023, May). *Conductivity Sensors and transmitters*. Retrieved from <https://www.endress.com/en/field-instruments-overview/liquid-analysis-product-overview/conductivity-sensors-transmitters>
- [5] Farnell. (n.d.). *LFS1K0.1107.2IB.070-6.S Conductivity Sensor*.
- [6] (n.d.). *Folleto Informativo Conductividad Eléctrica/Salinidad*. Folleto Informativo 3.1.3.0.
- [7] Frondiest. (n.d.). Retrieved from <https://www.fondriest.com/environmental-measurements/parameters/water-quality/conductivity-salinity-tds/>
- [8] H14. (n.d.). *100k Thermistor Output Table*.
- [9] HAMILTON COMPANY. (2023, May). *Conductivity Probes and Conductivity Sensors*. Retrieved from [https://www.hamiltoncompany.com/process-analytics/sensors/conductivity?utm\\_source=bing&utm\\_medium=ppc&utm\\_campaign=parameter&utm\\_term=conductivity&utm\\_content=generic&utm\\_term=conductivity%20measurement&utm\\_campaign=Process+Analytics+%7C+Search+%7C+Sensor](https://www.hamiltoncompany.com/process-analytics/sensors/conductivity?utm_source=bing&utm_medium=ppc&utm_campaign=parameter&utm_term=conductivity&utm_content=generic&utm_term=conductivity%20measurement&utm_campaign=Process+Analytics+%7C+Search+%7C+Sensor)
- [10] Internet of Things. (2023, April). *BluetoothLEIntro*. Retrieved from <https://iot.appinventor.mit.edu/#/bluetoothle/bluetoothleintro>
- [11] Mathur, D. (24/04/2018). *Conductivity: Water Quality Assesment*. INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT).
- [12] Microchip Developer. (n.d.). *ATMega328P Datasheet*.
- [13] Microchip. (n.d.). *Measure VCC/Battery Voltage Without Using I/O Pin on tinyAVR and megaAVR. : AN2447*.
- [14] Microchip. (n.d.). *Microchip Developer BLE*. Retrieved from <https://microchipdeveloper.com/wireless:ble-phy-layer>
- [15] MIT App Inventor. (n.d.). *MIT App Inventor*.
- [16] MIT AppInventor. (2023, April). *Connectivity*. Retrieved from <http://ai2.appinventor.mit.edu/reference/components/connectivity.html#Web>
- [17] Moko Smart. (n.d.). Retrieved from <https://www.mokosmart.com/es/lora-frequency/>
- [18] Pedro M. Ramos, M. I. (2006). *A Four Terminal Water Quality Monitoring Conductivity Sensor*. Lisboa, Portugal.

**Sensing and wireless transmission of  
Conductivity and Temperature, in a  
Reverse Osmosis Column**

- [19] Science Direct Pulse Width Modulation. (n.d.). *Science Direct*. Retrieved from <https://www.sciencedirect.com/topics/engineering/pulse-width-modulation>
- [20] Science Direct. (n.d.). *Science Direct Analog-to-Digital-Converter*. Retrieved from <https://www.sciencedirect.com/topics/engineering/analog-to-digital-converter>
- [21] Sudhakar M. Rao, T. T. (2007). *Role of direction of salt migration on the swelling behaviour of compacted clays*. Department of Civil Engineering, Indian Institute of Science, Bangalore 560012, India.
- [22] The Things Network. (n.d.). Retrieved from <https://www.thethingsnetwork.org/docs/lorawan/what-is-lorawan/>
- [23] Unesco. (1983). *Algorithms for computation of fundamental properties of seawater*. Unesco technical papers in marine science 44.
- [24] Wikipedia. (2023, May). *Electrical resistivity and conductivity*. Retrieved from [https://en.wikipedia.org/wiki/Electrical\\_resistivity\\_and\\_conductivity](https://en.wikipedia.org/wiki/Electrical_resistivity_and_conductivity)
- [25] Yacoob, H. Y. (Desember 2011). *A New system for Measuring Electrical Conductivity of Water as a Function of Admittance*.
- [26] Youtube - Pedro F. García Martín. (n.d.). *APP INVENTOR ESCRIBIR EN GOOGLE SHEETS*. Retrieved from <https://www.youtube.com/watch?v=DAC01jeenPY>

## 10. Annexes

### 10.1 Specified parameters

1. Pressure inside the Pressure Vessel 15 bar with Dynamic water.
2. 0.5 – 4 Bar inside the Pressure Vessel for Static Water
3. Dimensions of endcap

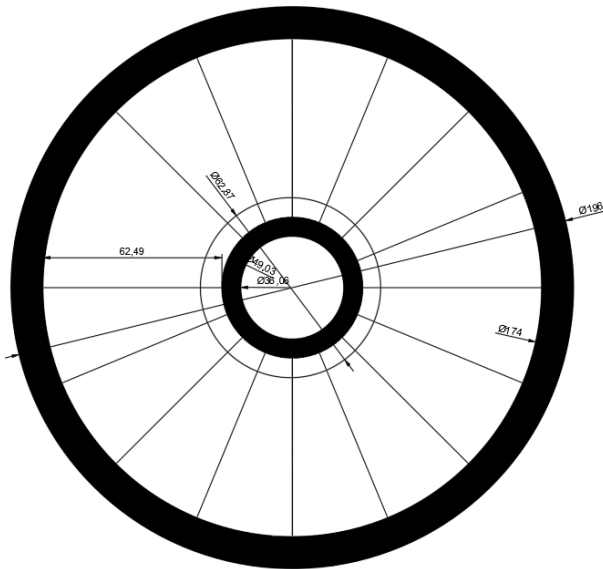


Figure 45 Endcap Dimensions

### 10.2 Sensor Cover Designs

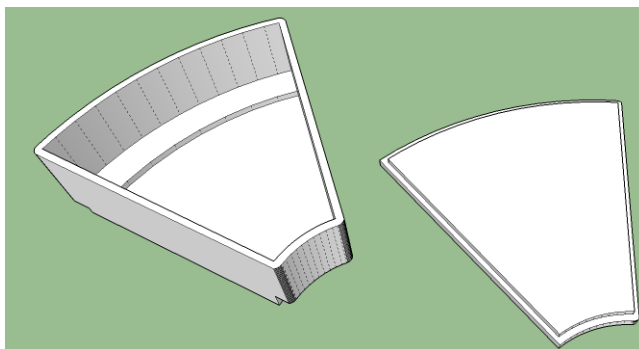


Figure 46

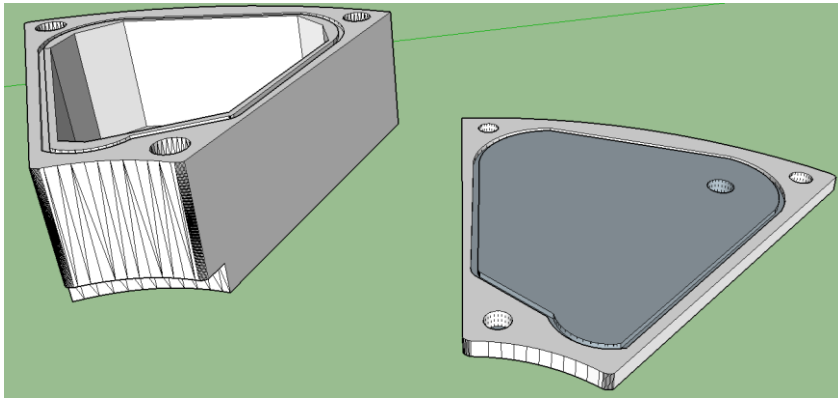


Figure 47



Figure 48

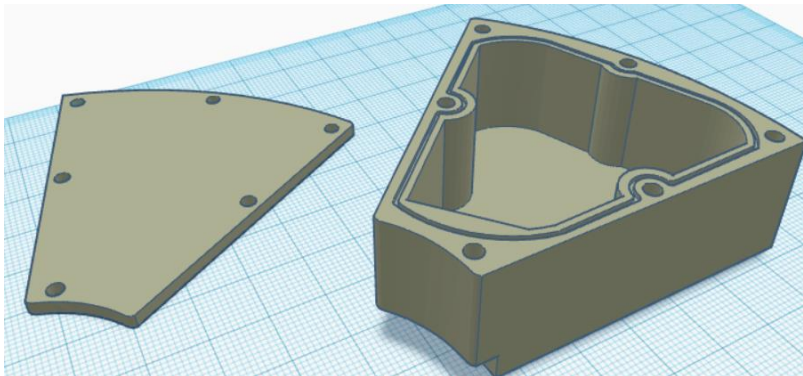


Figure 49



Figure 50 Endcap with two cut spokes and two Sensor Cover.



Figure 51 Endcap Stuck inside the Pressure Vessel

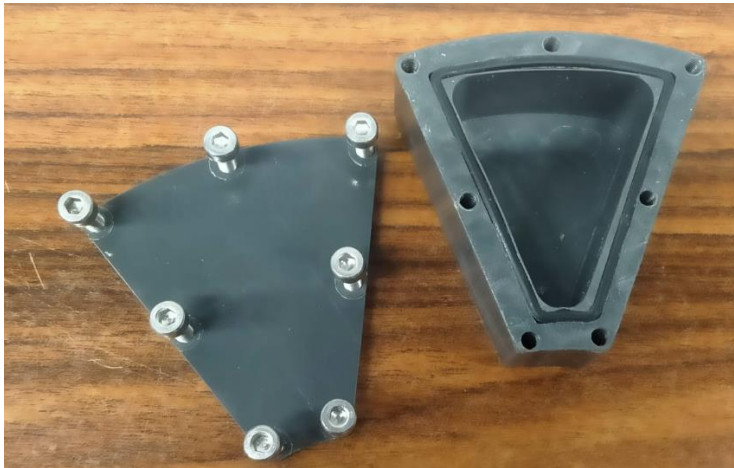
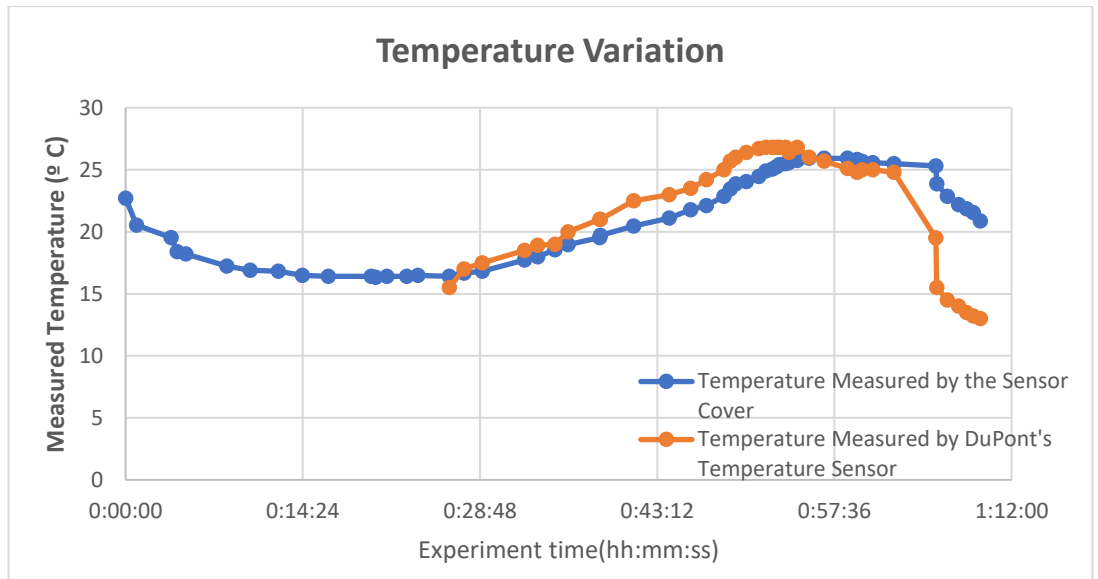


Figure 52 Mechanised CNC Sensor Cover

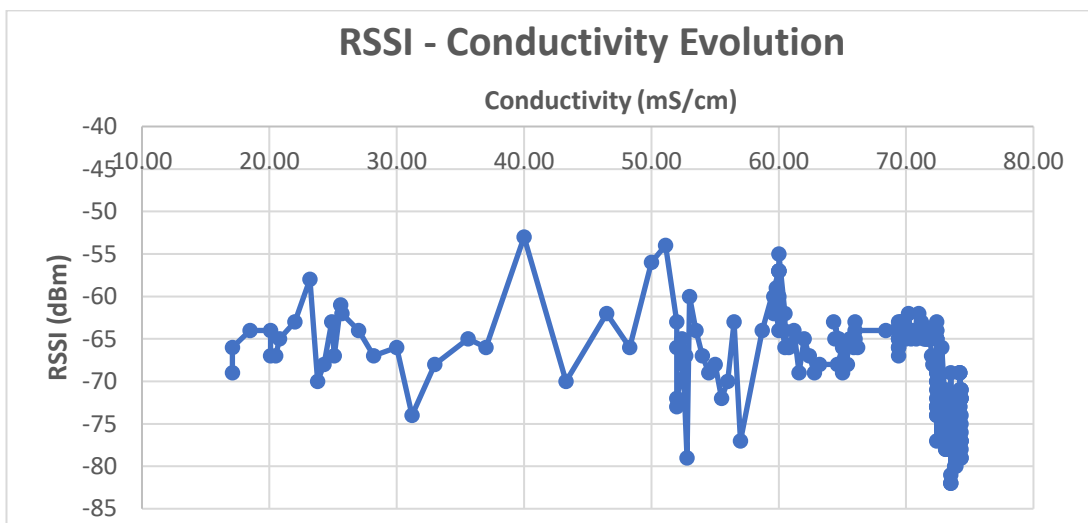


Figure 53 Final Sensor Cover with 2 electrodes.

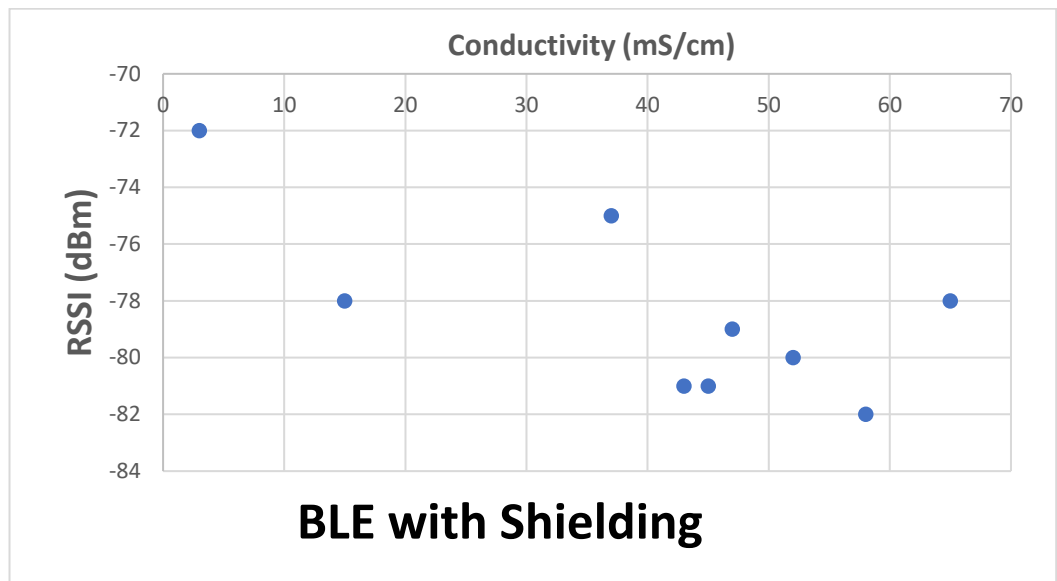
**10.3 Additional Data**



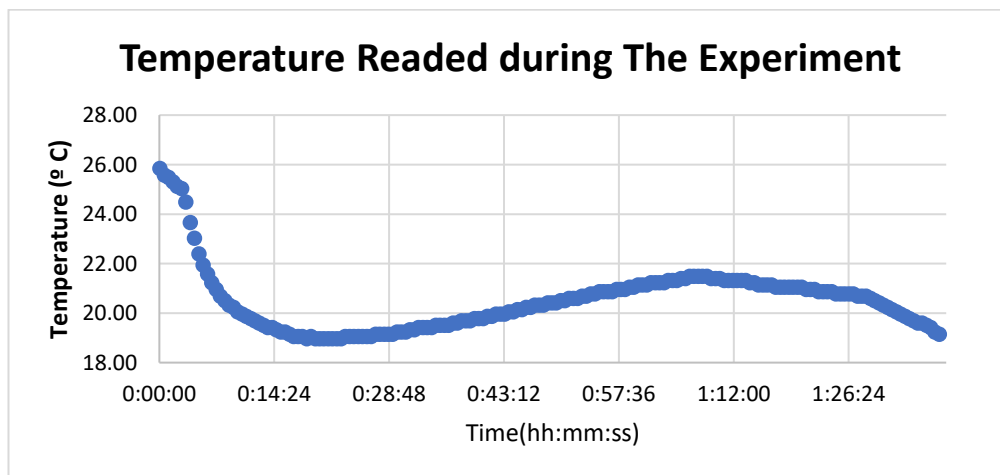
Graphic 24 Shows the Temperature comparisons between the inside and the outside of the Sensor Cover.



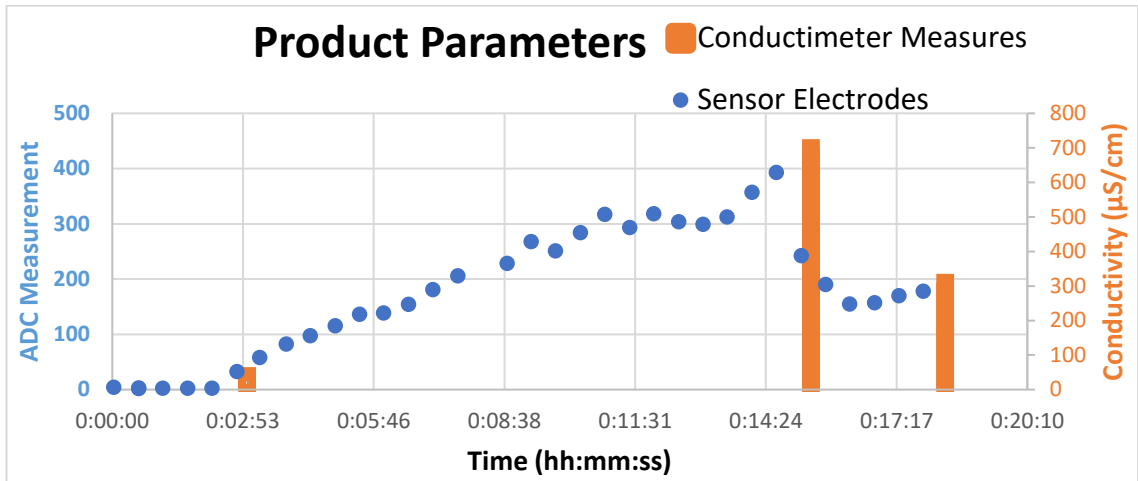
Graphic 25 RSSI versus the Conductivity read by The Company's Conductimeter when sensing only Temperature.



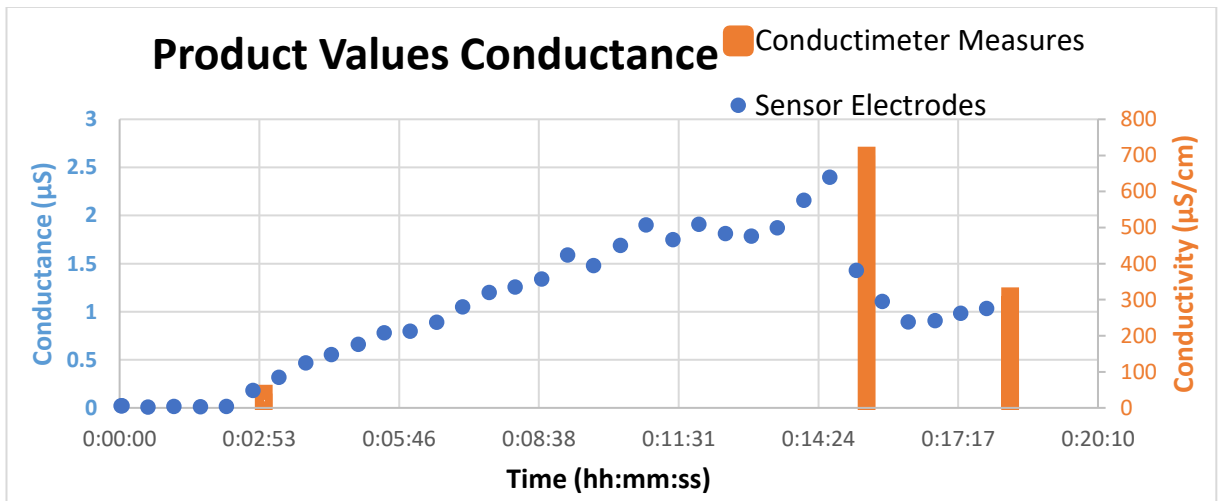
Graphic 26 Testing technology BLE with Shielding



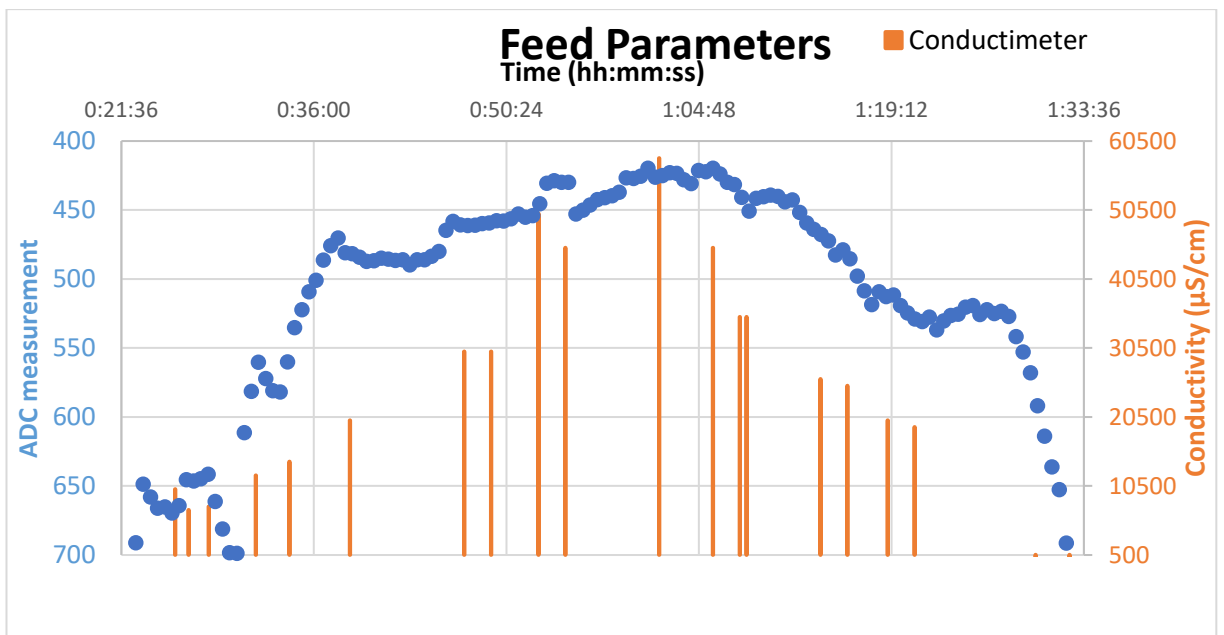
Graphic 27 Temperature measured by the Sensor Cover in Final test



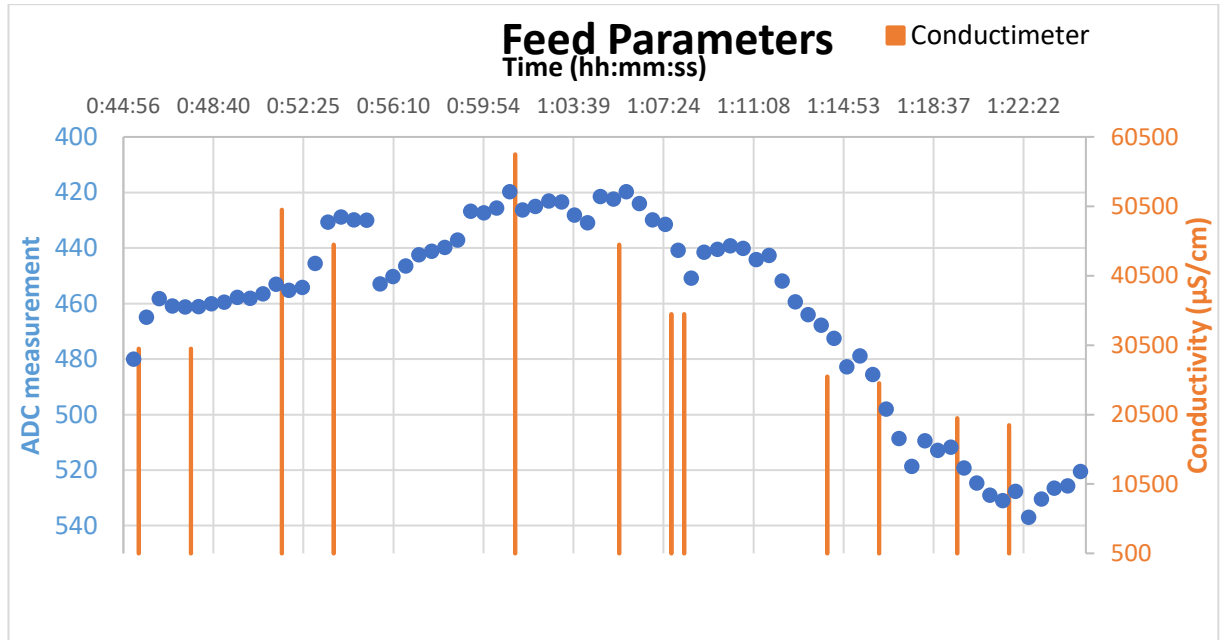
Graphic 28 Product Measures in Final Test



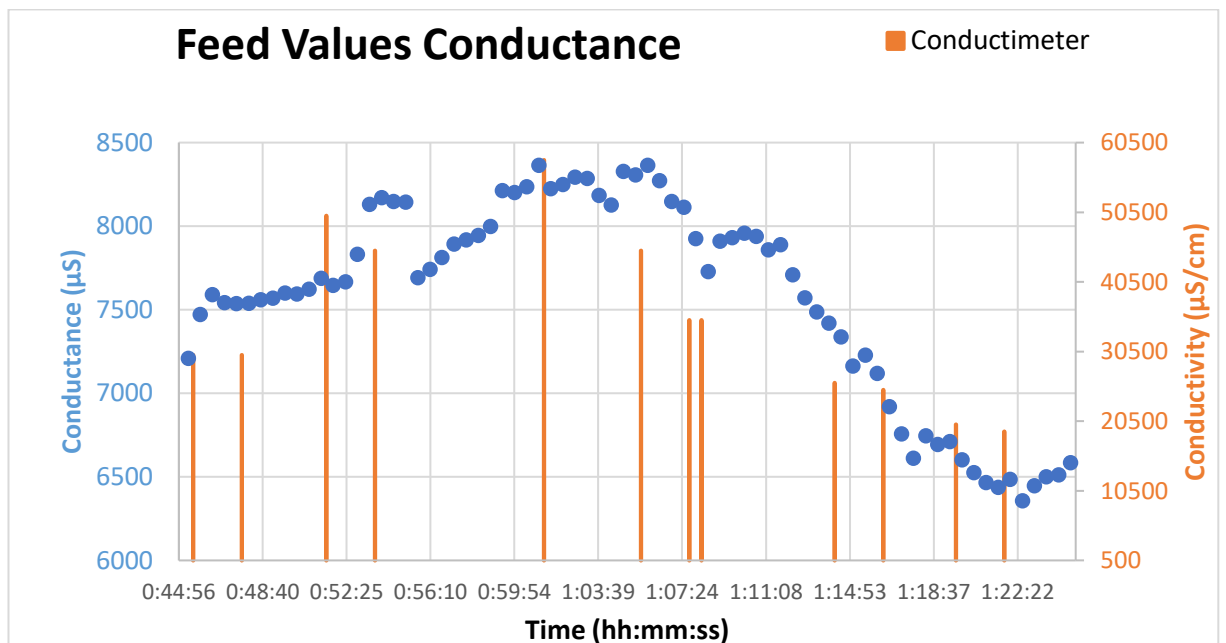
Graphic 29 Product Values measure in Conductance.



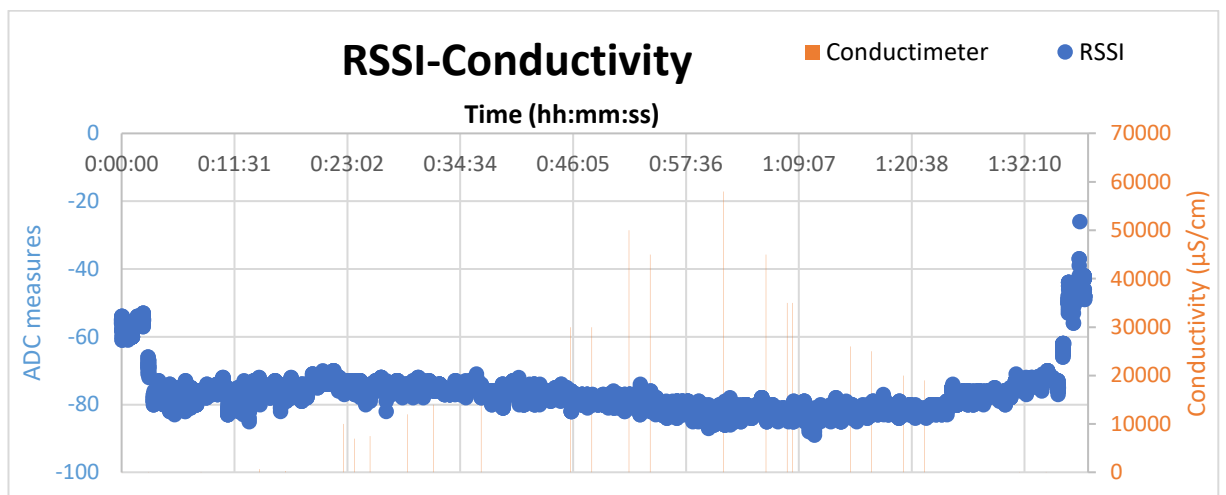
Graphic 30 Feed Values measured with the change of PWM high at minute 30.



Graphic 31 Feed Values measured.



Graphic 32 Feed Values measure in Conductance.



Graphic 33 RSSI versus Conductivity in final test

### 10.4 Schematics

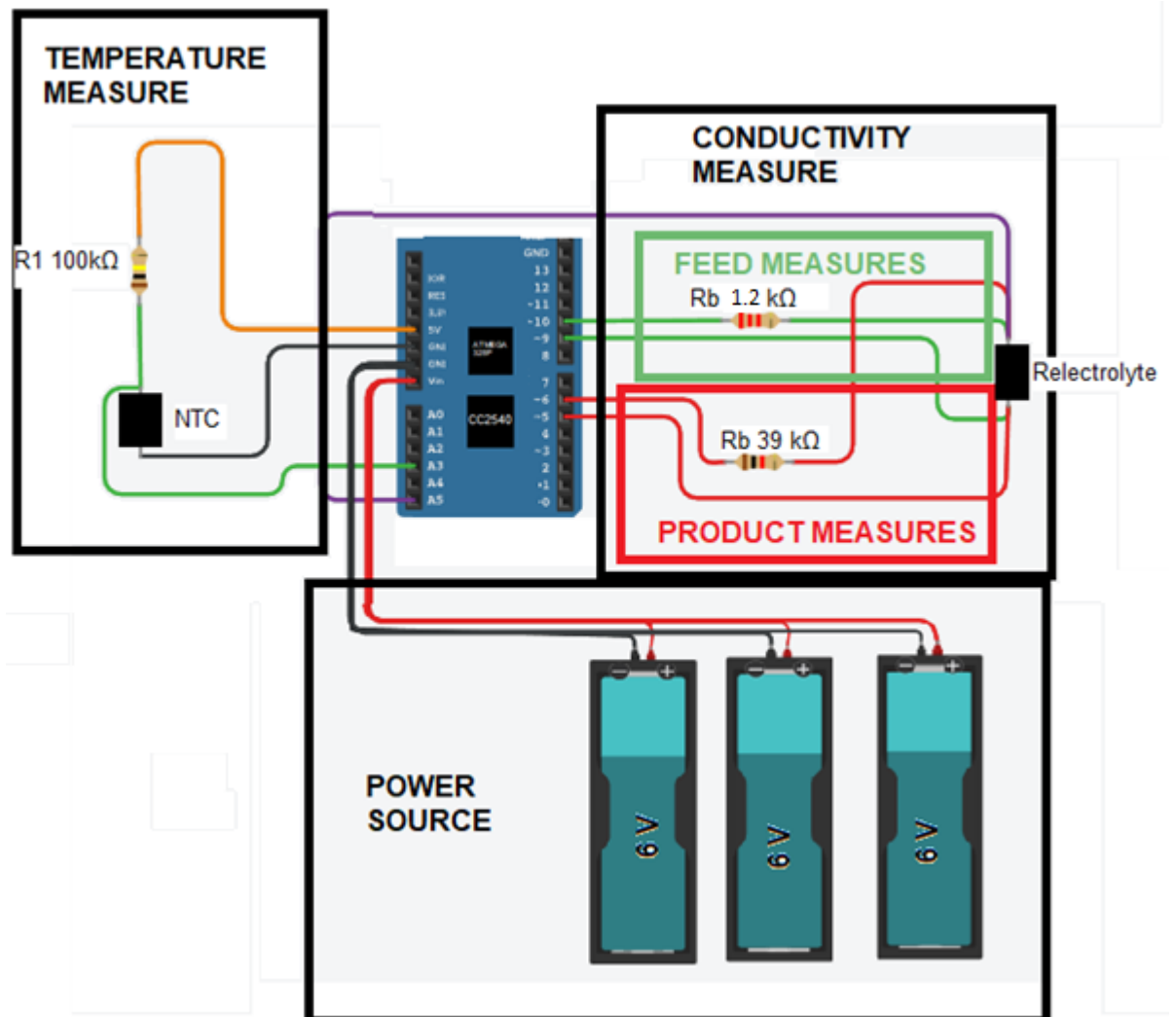


Figure 54 Final Schematic Used

### 10.5 Software BLE Android APP

```
initialize global device to  
initialize global service to  
initialize global name to  
  
when SCAN .Click  
do  
  call BluetoothLE2 .StartScanning  
  
when BluetoothLE2 .DeviceFound  
do  
  set ListPicker1 . ElementsFromString to BluetoothLE2 . DeviceList  
  
when ListPicker1 .AfterPicking  
do  
  set global device to segment text ListPicker1 . Selection  
  start 1  
  length 17  
  call BluetoothLE2 .ConnectWithAddress  
  address get global device  
  set ListPicker2 . ElementsFromString to BluetoothLE2 . DeviceCharacteristics  
  set RSSI . Text to call BluetoothLE2 .FoundDeviceRssi  
  index ListPicker1 . SelectionIndex  
  
when BluetoothLE2 .Connected  
do  
  set CheckBox1 . Checked to true  
  
when Disconnect .Click  
do  
  call BluetoothLE2 .DisconnectWithAddress  
  address get global device  
  set CheckBox1 . Checked to false
```

```

when ListPicker2 . AfterPicking
do
  if CheckBox1 . Checked
  then
    set global service to segment text ListPicker2 . Selection
    start 2
    length 38
    set Label4 . Text to BluetoothLE2 . ConnectedDeviceName
    set global name to BluetoothLE2 . ConnectedDeviceName
    set Clock1 . TimerEnabled to true
    set Label3 . Text to get global service
    call BluetoothLE2 . ReadConnectedRssi
    call BluetoothLE2 . RegisterForStrings
      serviceUuid " 0000ffe0-0000-1000-8000-00805f9b34fb "
      characteristicUuid " 0000ffe1-0000-1000-8000-00805f9b34fb "
      utf16 false
  else
    set Label3 . Text to " "
  
```

```

when BluetoothLE2 . RssiChanged
  rssi
do
  call BluetoothLE2 . ReadConnectedRssi
  
```

```

when BluetoothLE2 . StringsReceived
  serviceUuid characteristicUuid stringValues
do
  if CheckBox1 . Checked
  then
    set Web1 . Url to join
      " https://docs.google.com/forms/d/10Uk5Gqa3P-4gA_B... "
      "?entry.1305724076="
      get stringValues
      "&entry.1007060198="
      BluetoothLE2 . ConnectedDeviceRssi
    call Web1 . Get
    set Data . Text to get stringValues
  else
    set Data . Text to " "
  
```

## **10.6 Software BLE Sender Source Code**

```
#include <LowPower.h>

#define freq 1000

unsigned int cond_val[40];
unsigned int temp_val;
unsigned int i;
unsigned int sum = 0;
float battery = 0;

void setup() {

    Serial.begin(9600);
    Serial.println("Type AT commands!");
    pinMode (A3, INPUT);

    //mode fast PWM1
    TCCR1A &=~ (1<<WGM10);
    TCCR1A |= (1<<WGM11);
    TCCR1B |= (1<<WGM12);
    TCCR1B |= (1<<WGM13);

    //Define el TOP
    ICR1 = (16000000/1024/freq) - 1;

    //Output active Clear OC1A/OC1B on compare match, set OC1A/OC1B atBOTTOM (non-
    inverting mode)
    TCCR1A &=~ (1<<COM1A0);
    TCCR1A |= (1<<COM1A1);
```

```
//Output active Set OC1A/OC1B on compare match, clear OC1A/OC1B at BOTTOM  
(inverting mode)
```

```
TCCR1A |= (1<<COM1B0);
```

```
TCCR1A |= (1<<COM1B1);
```

```
//define the time at HIGH
```

```
OCR1A = (((16000000/1024/freq) - 1)*50)/100;
```

```
//define the time at HIGH the inverted signal
```

```
OCR1B = (((16000000/1024/freq) - 1)*50)/100;
```

```
// Mode Fast PWM0
```

```
TCCR0B &= ~(1<<WGM02);
```

```
TCCR0A |= (1<<WGM01);
```

```
TCCR0A |= (1<<WGM00);
```

```
// Prescalador 64
```

```
TCCR0B &= ~(1<<CS02);
```

```
TCCR0B |= (1<<CS01);
```

```
TCCR0B |= (1<<CS00);
```

```
//Output active Clear OC1A/OC1B on compare match, set OC1A/OC1B atBOTTOM (non-  
inverting mode)
```

```
TCCR0A &= ~(1<<COM0A0);
```

```
TCCR0A |= (1<<COM0A1);
```

```
//Output active Set OC1A/OC1B on compare match, clear OC1A/OC1B at BOTTOM  
(inverting mode)
```

```
TCCR0A |= (1<<COM0B0);
```

```
TCCR0A |= (1<<COM0B1);
```

```
//define the time at HIGH (duty cycle of 50%)
```

```
OCR0A = 127;
```

```
//define the time at HIGH the inverted signal (duty cycle of 50%)
OCR0B = 127;

//Adjust to the right
ADMUX &=~ (1<<ADLAR);

//Adjust reference Voltage to 1.1 V
ADMUX |= (1<<REFS0);
ADMUX |= (1<<REFS1);
//Adjust reference Voltage to 5V
//ADMUX |= (1<<REFS0);
//ADMUX &=~ (1<<REFS1);

//Adjust the frequency divisor at 128: 16.000.000/128=125 kHz
ADCSRA = ADCSRA | 0x07; //hacemos la frecuencia del adc a 125kHz
}

void PWM0_on()
{
    TCNT0 = 0x0000;

    //Pin configuration //we set PD5 and PD6 that are OC0A and OC0B as OUTPUTS
    DDRD |= (1<<DDD6); //pin D6
    DDRD |= (1<<DDD5); //pin D5

    //Output active Clear OC1A/OC1B on compare match, set OC0A/OC0B atBOTTOM (non-
    inverting mode)
    TCCR0A &=~ (1<<COM0A0);
    TCCR0A |= (1<<COM0A1);
```

```
//Output active Set OC1A/OC1B on compare match, clear OC0A/OC0B at BOTTOM  
(inverting mode)
```

```
TCCR0A |= (1<<COM0B0);  
TCCR0A |= (1<<COM0B1);  
}
```

```
void PWM0_off()
```

```
{  
  //Pin configuration //we set PD5 and PD6 that are OC0A and OC0B as INPUTS  
  DDRD &=~ (1<<DDD6); //pin D6  
  DDRD &=~ (1<<DDD5); //pin D5
```

```
//Output active Clear OC0A/OC1B on compare match, set OC0A/OC1B atBOTTOM (non-  
inverting mode)
```

```
TCCR0A &=~ (1<<COM0A0);  
TCCR0A &=~ (1<<COM0A1);
```

```
//Output active Set OC0A/OC1B on compare match, clear OC1A/OC1B at BOTTOM  
(inverting mode)
```

```
TCCR0A &=~ (1<<COM0B0);  
TCCR0A &=~ (1<<COM0B1);  
}
```

```
void PWM1_on()
```

```
{  
  //Pin configuration //we set PB1 and PB2 that are OC1A and OC1B as OUTPUTS  
  DDRB |= (1<<DDB1); //pin D9  
  DDRB |= (1<<DDB2); //pin D10  
  TCNT1 = 0x0000;  
  //N = 1024 q es el preescaler  
  TCCR1B |= (1<<CS10);  
  TCCR1B &=~ (1<<CS11);  
  TCCR1B |= (1<<CS12);
```

```
//Output active Clear OC1A/OC1B on compare match, set OC1A/OC1B atBOTTOM  
(non-inverting mode)
```

```
TCCR1A &=~ (1<<COM1A0);
```

```
TCCR1A |= (1<<COM1A1);
```

```
//Output active Set OC1A/OC1B on compare match, clear OC1A/OC1B at BOTTOM  
(inverting mode)
```

```
TCCR1A |= (1<<COM1B0);
```

```
TCCR1A |= (1<<COM1B1);
```

```
}
```

```
void PWM1_off()
```

```
{
```

```
//Pin configuration //we set PB1 and PB2 that are OC1A and OC1B as INPUTS
```

```
DDRB &=~ (1<<DDB1); //pin D9
```

```
DDRB &=~ (1<<DDB2); //pin D10
```

```
//Clock setting T1clock = 0 Hz
```

```
TCCR1B &=~ (1<<CS10);
```

```
TCCR1B &=~ (1<<CS11);
```

```
TCCR1B &=~ (1<<CS12);
```

```
//Output active Clear OC1A/OC1B on compare match, set OC1A/OC1B atBOTTOM  
(non-inverting mode)
```

```
TCCR1A &=~ (1<<COM1A0);
```

```
TCCR1A &=~ (1<<COM1A1);
```

```
//Output active Set OC1A/OC1B on compare match, clear OC1A/OC1B at BOTTOM  
(inverting mode)
```

```
TCCR1A &=~ (1<<COM1B0);
```

```
TCCR1A &=~ (1<<COM1B1);
```

```
}
```

```
int ADC_GetData(int canal)
{
    // Selection of the ADC channel //
    ADMUX &=~ 0x0F;
    ADMUX |= canal;

    // Turn on ADC
    ADCSRA |= (1<<ADEN);
    delayMicroseconds(10); // Esperamos a que caliente

    // Send the sample
    ADCSRA |= (1<<ADSC);

    // Wait until it samples reading the flag
    while( !(ADCSRA & (1<<ADIF)) );
    ADCSRA |= (1<<ADIF); // Restartflag

    // Turn OFF ADC
    ADCSRA &=~ (1<<ADEN);

    return ADC;
}

float getBatteryVolts2() {

    const float InternalReferenceVoltage = 1.1; // <- as measured (or 1v1 by default)

    // turn ADC on
    ADCSRA = bit (ADEN);

    // Prescaler of 128
    ADCSRA |= bit (ADPS0) | bit (ADPS1) | bit (ADPS2);
```

```
// MUX3 MUX2 MUX1 MUX0 --> 1110 1.1V (VBG) - Selects channel 14, bandgap
voltage, to measure
ADMUX = bit (REFS0) | bit (MUX3) | bit (MUX2) | bit (MUX1);

// let it stabilize
delay (10);

// start a conversion
bitSet (ADCSRA, ADSC);

// Wait for the conversion to finish
while (bit_is_set(ADCSRA, ADSC))
{
;
}

// Float normally reduces precion but works OK here. Add 0.5 for rounding not
truncating.
float results = InternalReferenceVoltage / float (ADC + 0.5) * 1024.0;
return results;
}

void loop() {
  PWM0_off();
  PWM1_on();
  for (int i =0; i<16; i++)
  {
    cond_val[i] = ADC_GetData(5);
  }
  PWM1_off();
  PWM0_off();
}
```

```
for (int i =0; i<12; i++)
{
  delay (40);
  Serial.print (1);
  Serial.print (" ");
  Serial.println (cond_val[i]);
}
delay(200);
LowPower.powerDown(SLEEP_2S, ADC_OFF, BOD_OFF);
PWM1_off();
PWM0_on();
for (int i =0; i<16; i++)
{
  cond_val[i] = ADC_GetData(5);
}
PWM1_off();
PWM0_off();
for (int i =0; i<12; i++)
{
  delay (40);
  Serial.print (2);
  Serial.print (" ");
  Serial.println (cond_val[i]);
}
PWM1_off();
PWM0_off();
delay(200);
LowPower.powerDown(SLEEP_2S, ADC_OFF, BOD_OFF);
//Adjust reference Voltage to 5 V
ADMUX |= (1<<REFS0);
ADMUX &=~ (1<<REFS1);
temp_val = analogRead(A3);
```

```
//Adjust refference Voltage to 1.1 V
ADMUX |= (1<<REFS0);
ADMUX |= (1<<REFS1);
Serial.print(3);
Serial.print(" ");
Serial.println(temp_val);
delay (200);
LowPower.powerDown(SLEEP_8S, ADC_OFF, BOD_OFF);
LowPower.powerDown(SLEEP_8S, ADC_OFF, BOD_OFF);
LowPower.powerDown(SLEEP_8S, ADC_OFF, BOD_OFF);
}
```

## **10.7 Sender LoRa Source Code**

```
#include "heltec.h"
#include "images.h"

#define BAND 868E6 //you can set band here directly,e.g. 868E6,915E6
unsigned int counter = 0;
unsigned int tandas = 0;
char btchar = 0;

String rssi = "RSSI --";
String packSize = "--";
String packet;

void logo()
{
  Heltec.display->clear();
  Heltec.display->drawXbm(0,5, WiFi_Logo_width, WiFi_Logo_height, WiFi_Logo_bits);
  Heltec.display->display();
}

void setup() {

  Serial.begin(28800);
  //WIFI Kit series V1 not support Vext control
  Heltec.begin(true /*DisplayEnable Enable*/, true /*Heltec.LoRa Disable*/, true
/*Serial Enable*/, true /*PABOOST Enable*/, BAND /*long BAND*/);

  Heltec.display->init(); //initialise display module
  Heltec.display->setFont(ArialMT_Plain_10); //adjust display letter
  logo(); //function that will show the display
  delay(1500);
  Heltec.display->clear(); //clear screen
```

```
    Heltec.display->drawString (0,0, "Heltec.Lora Initial Success!");
    Heltec.display->display();
    delay(1000);
}

void loop()
{

    if (tandas < 15)
    {
        Heltec.display->clear();
        Heltec.display->setTextAlignment(TEXT_ALIGN_LEFT);
        Heltec.display->setFont(ArialMT_Plain_10);
        Heltec.display->drawString (0,0, "Sending Packet:");
        Heltec.display->drawString (90,0, String(counter));
        Heltec.display->drawString (0,15, "Tandas:"+String(tandas));
        Heltec.display->display();
        // send packet
        LoRa.beginPacket();
        //LoRa.setTxPower(txPower,RFOUT_pin); //adjust power transmission values
        // txPower -- 0 ~ 20
        //   RFOUT_pin could be RF_PACONFIG_PASELECT_PABOOST or
RF_PACONFIG_PASELECT_RFO
        //   - RF_PACONFIG_PASELECT_PABOOST -- LoRa single output via PABOOST,
maximum output 20dBm
        //   - RF_PACONFIG_PASELECT_RFO -- LoRa single output via RFO_HF / RFO_LF,
maximum output 14dBm
        LoRa.setTxPower(0+tandas,RF_PACONFIG_PASELECT_PABOOST);
        LoRa.print(counter);
        LoRa.print("Tanda:");
        LoRa.print(tandas);
        LoRa.endPacket();
```

```
counter++;  
delay(1000);  
//Serial.println("Espera 1");           // wait for a second  
//digitalWrite(25, LOW); // turn the LED off by making the voltage LOW  
delay(1000);  
  
if (counter == 4){  
    tandas++;  
    counter=0;  
}  
} else {tandas =0;}  
}
```

## **10.8 Receiver LoRa Source Code**

```
#include "heltec.h"
#include "images.h"

#define BAND 868E6 //you can set band here directly,e.g. 868E6,915E6
unsigned int counter = 0;

String rssi = "RSSI --";
float potencia;
String packSize = "--";
String packet;

void logo()
{
  Heltec.display->clear();
  Heltec.display->drawXbm(0,5, WiFi_Logo_width, WiFi_Logo_height, WiFi_Logo_bits);
  Heltec.display->display();
}

void LoRaData(){
  Heltec.display->clear();
  Heltec.display->setTextAlignment(TEXT_ALIGN_LEFT);
  Heltec.display->setFont(ArialMT_Plain_10);
  Heltec.display->drawString(0, 15, "Recibidos:"+packSize+"bytes");
  Heltec.display->drawString(0, 0,"RSSI" + rssi);
  Heltec.display->drawString(0, 25, "Paquete:"+packet);
  Heltec.display->display();
}

void cbk (int packetSize){
  packet="";
```

```
    packSize = String (packetSize, DEC);
    for (int i = 0; i < packetSize; i++)
    {
        packet += (char) LoRa.read();
    }
    rssi = String(LoRa.packetRssi(), DEC);
    Serial.print(potencia);
    LoRaData();
}
void setup() {

    Heltec.begin(true /*DisplayEnable Enable*/, true /*Heltec.LoRa Disable*/, true /*Serial
Enable*/, true /*PABOOST Enable*/, BAND /*long BAND*/);
    Heltec.display->init(); //initialise display module
    Heltec.display->setFont(ArialMT_Plain_10); //adjust display letter
    logo(); //function that will show the display
    delay(1500);
    Heltec.display->clear(); //clear screen
    Heltec.display->drawString (0,15, "Waiting values");
    Heltec.display->display();
    delay(1000);
}

void loop() {

    int packetSize = LoRa.parsePacket(); //check if a packet has been received, it returns
the number of bytes of the packet received, if no packet returns 0
    if (packetSize) {cbk (packetSize);} //if it hasn't returned 0, go to the function: read the
packet and print it
    delay (10);
}
```

## 10.9 User Manual

In order to know how to connect or create the power source, it is necessary to weld with a parallel connexion the three batteries. Welding in parallel means that the positive pole of each battery needs to be in conductive contact with the other positives pole of the other batteries, it is needed to repeat the same procedure with the negative poles of the batteries. Making that battery-parallel, the capacities of the batteries (150 mA·h) will be together, and the total capacity will be 450 mA·h. Increasing the total capacity means that that the batteries would last longer. In this case the batteries have 6 V, the Kit we used for the nano BLE Keywish needed 6-12 V.

Once the power source is connected and the program is in the microcontroller, the data can be readed thanks to the APP created with APP inventor. The mobile reads the data and sends it to a Google Forms that is linked with a Google Sheets, after it the data can be copied with the timeline from the Google Sheets to a tool to analyse the data (suggestion: excel). The Sensor is programmed, as we can see in Chapter 10.6 Software BLE Sender Source Code, to sent 12 measurements of Conductance in each case (with the different Rb) and 1 measurement of temperature. Afterwards in our excel we do an average of the low values read from the different Rb values.

The URL to download the APP is: <http://ai2.appinventor.mit.edu/b/337p9>

ADC values are send through BLE. The Android app read these values and save them in a google sheet. After that a post-mortem data analisys may be carried out with the prefered tool of the user.

This post-mortem analysis consists of converting the values into the correct measure. If it has been sent the identifier and is a 3, the value sent after this identifier was Temperature, let's see it with an example:

Marca temporal	Datos	RSSI
19/04/2023 11:15:42	["3 "]	-58
19/04/2023 11:15:42	["545"]	-60

In order to convert this "545" ADC value into Temperature we need to use this Equation:

$$T_{NTC} = \left( \frac{\ln \left( \frac{\left( \frac{ADC \cdot 5/1023}{V_i} \cdot R_1 \right)}{100000 \cdot \left( 1 - \frac{ADC \cdot 5/1023}{V_i} \right)} \right)}{4130} + \frac{1}{298} \right)^{-1} - 273.15$$

If we substitute our know values, assuming Vi as 5 V.

$$T_{NTC} = \left( \frac{\ln \left( \frac{\left( \frac{545 \cdot 5/1023}{5} \cdot 100000 \right)}{100000 \cdot \left( 1 - \frac{545 \cdot 5/1023}{5} \right)} \right)}{4130} + \frac{1}{298} \right)^{-1} - 273.15 = 22,2 \text{ } ^\circ\text{C}$$

In order to understand the data read by the product or the feed, it is necessary to remind the schematic circuit from [Figure 39](#) Circuit proposed with 2 PWM. and the Equations 11 and 12

$$V_o = PWM1 \cdot \frac{R_{elec}}{R_b + R_{elec}}$$

Equation 16

$$V_o = PWM2 \cdot \frac{R_b}{R_b + R_{elec}}$$

Equation 17

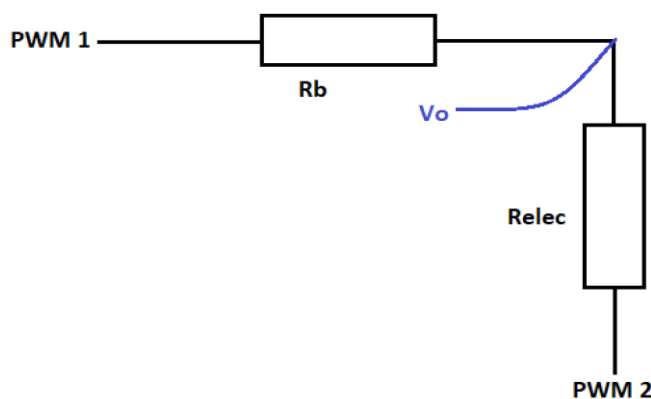


Figure 55 Circuit proposed with 2 PWM.

In case of the [Equation 11](#), lowering the  $R_b$  will make the output closer to 5 V when the PWM is at high level, and by the time it is at low level it will be at 0 for sure because of what we multiply by 0. If we raise the  $R_b$  for the first case, the  $V_o$  will go down if we are at the high level of the PWM.

In case of the [Equation 12](#), lowering the  $R_b$  will make the output closer to 0 when the PWM is high and closer to 5 V when the PWM is low. If we raise the  $R_b$  value,  $V_o$  will be closer and closer to 5 V when the PWM 2 signal is at high level.

Before selecting the  $R_b$  value, let's understand how the measures experiment will work. Supposing that the  $R_b$  value is set, the variable value is the  $R_{elec}$ . If the conductance of the electrolyte is very low (low salinity), the  $R_{elec}$  value will be very big; in this case the  $V_o$  read when PWM1 is at 5 V will be very low near 0 V and the  $V_o$  read when PWM2 is at 5 V will be near 5 V. Adding salinity which implies increasing the conductance, will make that the  $R_{elec}$  value decreases. This means that the  $V_o$  read when PWM1 is at 5V will start increasing and the  $V_o$  read when PWM2 is at 5 V will start decreasing. If we continuously decrease the  $R_{elec}$  value, there will be a point in which both values for PWM1 and PWM2 will be the same and

then the  $V_o$  from PWM1 at High will have a smaller value than  $V_o$  from PWM1 at High. If we arrive to a very conductive electrolyte, the  $V_o$  when PWM1 is at 5 V will be near 5 V and the  $V_o$  read when PWM2 is at 5 V will be near 0 V. In the Figures below we see the example with two  $R_b$ : the lower  $R_b$  of 2200  $\Omega$  and 50000 $\Omega$ .

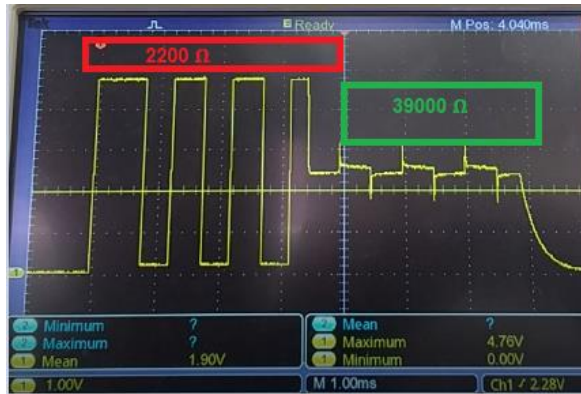


Figure 56

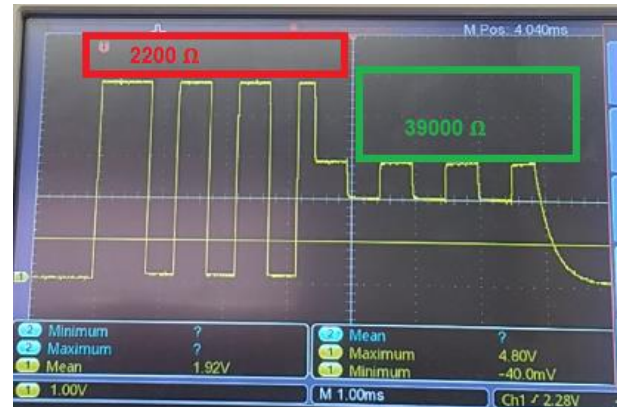


Figure 57

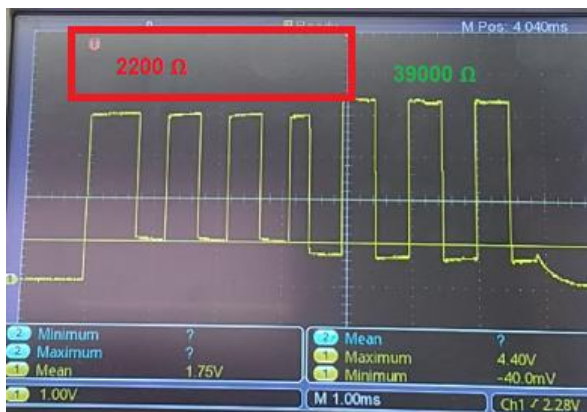


Figure 58

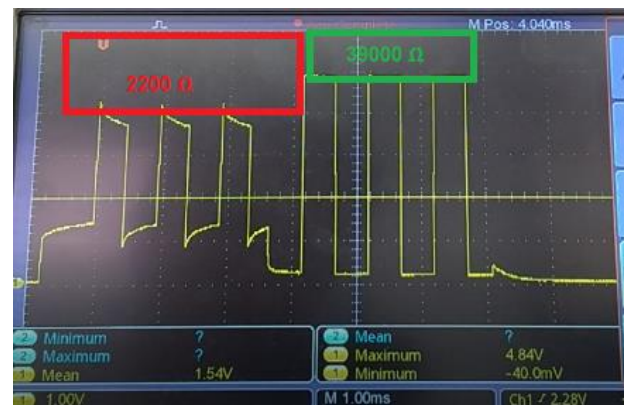


Figure 59

Initially here, with the value of the 39000  $\Omega$  we are at the point of having the same  $V_o$  when both PWMs are at high. With the 2200  $\Omega$  we are with  $V_o$  from the Equation 11 at higher value and  $V_o$  from the Equation 12 at lower value. In the [Figure 41](#) salt has been added and Relec value has decreased, in this case the  $V_o$  from Equation 12 has increased meanwhile  $V_o$  from [Equation 11](#) decreased. The same happens with the  $R_b$  with the red colour, case the  $V_o$  from [Equation 12](#) has increased meanwhile  $V_o$  from Equation 11 decreased. In the [Figure 42](#) we can observe in the red  $R_b$  how the  $V_o$  values from each PWM swapped its positions and now the  $V_o$  from the Equation 11 has a lower value than the one in the Equation 12.

Now that there has been a brief introduction, at first, if the salinity is very low, the ADC value increases when the salinity increases, that is because initially, the  $R_b$  has a lower value than Relec and this means that we are initially measuring with 5 V at PWM1 we are at very high values (5 V – 1023 LSBs) and when measuring with PWM2 at 5 V, we are at low values (values below 1023 read by the ADC which is referenced at 1.1 V), this all implies that we are initially with the [Equation 12](#).

An important comment in order to have a better understanding is that increasing the salinity alludes to decreasing the Relec value, checking the Equation 12, this means that Vo value increases. Meanwhile Vo value from Equation 12 increases, the Vo value from Equation 11 decreases because the Relec multiplying in the numerator decreases. If salinity gets increased slowly, there will be a point in which both Vo values from both equations will be the same and then Vo value from Equation 11 will continue decreasing and Vo value from Equation 12 will continue increasing. Once the point has reached here, the numbers read by the ADC will be the values from Equation 11.

The equations to calculate the conductance values are:

$$R_{elec} = \frac{\left(\frac{V_o}{V_i} \cdot R_B\right)}{1 - \frac{V_o}{V_i}}$$

$$= \frac{\left(\frac{ADC \cdot \frac{1.1 V}{1023} \cdot R_B\right)}{1 - \frac{ADC \cdot \frac{1.1 V}{1023}}$$

Equation 18 Feed Relec

$$R_{elec} = \frac{\left(\left(1 - \frac{V_o}{V_i}\right) \cdot R_B\right)}{\frac{V_o}{V_i}}$$

$$= \frac{\left(\left(1 - \frac{ADC \cdot \frac{1.1 V}{1023}\right) \cdot R_B\right)}{\frac{ADC \cdot \frac{1.1 V}{1023}}$$

Equation 19 Product Relec