Bachelor's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electronic Systems

Ida Frøhoel
Jonas Grønskag Johansen
Julie Klingenberg

Developing an underwater sensing rig

Bachelor's thesis in Electronical Engineering
Supervisor: Rolf Kristian Snilsberg
May 2022

NTNU
Norwegian University of
Science and Technology

Ida Frøhoel
Jonas Grønskag Johansen
Julie Klingenberg

Developing an underwater sensing rig

Bachelor's thesis in Electronical Engineering
Supervisor: Rolf Kristian Snilsberg
May 2022

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electronic Systems

NTNU
Norwegian University of
Science and Technology

Summary

Developing multipurpose Underwater Sensing Rig is the project given by the Department of Engineering Cybernetics. In this project the group had to develop a working rig with four different sensors. The main tasks for this project were creating code for all four sensors, develop code for battery monitoring, and model and 3D print a hardware solution. Data collection and communication was not taken into consideration during the project.

Each of the sensors codes were developed independently, though the structure were the same for all sensors. Two of the sensors used, this being temperature and pressure, were from BlueRobotics, and were manufactured to fit in the watertight enclosure. The other two sensors, dissolved oxygen and conductivity, were from Atlas Scientific. Because of this they needed to be cut, and rehoused to fit in the watertight enclosure. This was done with the help of NTNU workshop, department of electronics.

Because of problems caused by the PSM, and a destroyed ADC, the code for battery monitoring ended up incomplete. This ended up being because the PSM output a voltage of +12V on the ADC instead of the expected maximum of +3.3V. The datasheet for the ADC states that the analog pins can only sustain a voltage up to V_{DD}+0.3V before breaking. This resulted in the ADC breaking, which prompted a halt in development as the group were running out of time and didn't want to risk destroying more of the electronics.

The third main part of the project was to 3D model and print a hardware layout. There were in total ten 3D printed parts which were all printed using the 3D-printer Ultimaker S5, with the plastic PLA.

After the sensorcode was developed a "functionality test" was performed. The test concluded with 3/4 sensors working as expected, but there is need for more research and development on the dissolved oxygen sensor from Atlas Scientific.

The final product is a functioning sensor rig which can read values from three of the sensors, be powered by the battery, and has a 3D modeled hardware solution. The final product is not completely finished, and there are possibilities for further development on both battery monitoring, and the dissolved oxygen sensor.

Sammendrag

Uviklingen av en flerfunksjonell undervanns sensorrigg er en oppgave gitt av institutt for teknisk kybernetikk, som går ut på å lage en fungerende sensorrigg som måler fire ulike sensordataer. For å gjennomføre oppgaven har gruppen måtte designe og printe 3D modellerte deler som de elektriske komponentene kan festes på. Hovedoppgavene i denne oppgaven var å programmere koden til de fire sensorene, lage et batteriovervåkningssystem, og 3D printe- og modellere hardware planløsningen. Datainnsamling og kommunikasjon var ikke tatt hensyn til i dette prosjektet.

Koden til sensorene var utviklet hver for seg, men strukturen var den samme for alle sensorene. To av sensorene, temperatur og trykk, var fra produsenten BlueRobotics, og var laget for å passe i den vanntette beholderen som hører til prosjektet. De to andre sensorene, oppløst oksygen og konduktivitet, var fra produsenten Atlas Scientific. Disse to sensorene måtte bli kuttet for å tre på vanntette kapslinger, og ble også "rehoused" av NTNU verkstedet, avdeling elektronikk.

Koden for batteriovervåkning var uferdig ved prosjektslutt, da det oppstod problemer med å lese av verdiene fra PSM'en, og en ADC ble ødelagt. Dette viste seg å være fordi PSM'en ga en utgangsspenning på +12V, målt med voltmeter, isteden for en verdi lavere enn V_{DD}+0.3V som var oppgitt i databladet. Dette resulterte i at en ADC ble ødelagt, og videre utvikling måtte stoppes, da det var for stor risiko for å ødelegge annen elektronik.

Den tredje hoveddelen var å 3D modellere og printe de utviklede modellene. Det var utviklet totalt ti modeller, og alle ble printet i 3D-printeren Ultimaker S5, med plastikktypen PLA.

Etter at sensorkoden var utviklet ble sensorene tested i en "funksjonalitetstest", da gruppen ikke hadde nøyaktige testvæsker tilgjengelig. Testen ble konkludert med at 3/4 sensorer virker som forventet, og det trengs mer feilsøking og utvikling på sensoren oppløst oksygen, fra Atlas Scientific.

Sluttproduktet er en funksjonell sensorrigg som kan lese verdier fra tre sensorer, få spenningsforsyning fra batteriet, og har en 3D modellert hardware planløsning. Sluttproduktet ble ikke helt ferdig utviklet, og mulige videre utviklinger er ferdigstilling av batteriovervåkningen, og ferdigstilling av sensoren for oppløst oksygen.

Preface

This project report is written in the spring of 2022 for the Department of Engineering Cybernetics at the Norwegian University of Science and Technology. This final project completes the three year study, bachelor in engineering, electronics.

We want to thank our supervisor Rolf Kristian Snilsberg, and our employers Damiano Varagnolo and Behdad Aminian for the support and guidance throughout the semester. We also want to thank Simon Andreas Hagen Hoff, and the NTNU workshop for their assistance.

Contents

List of Figures

List of Tables

Nomenclature/Glossary

ITK	Department of Engineering Cybernetics
ROV	Remotely Operated Vehicle
ROS	Robot Operating System
Micro-controller	A small computer with a programmable interface which can read analog input
OS	Operational System
ADC	Analog-to-digital converter
I2C	Bus for communication
PSM	Power Sense Module
Vertex	A point in a three-dimensional space
Edge	The line between two vertices.
Face	The plane between three or more edges.
Polygon	Flat closed curve, composed of a finite number of straight-line segments.
Mesh	A 3D model composed of a certain number of triangular polynomials.
Solid modeling	3D modeling of solid objects.
Surface modeling	3D modeling of an objects exterior and contours.
CAD (Computer-Aided Design)	A program that is able to digitally create models of real-world products in both 2D and 3D before they are ever manufactured.
STL file	A file type that stores information about 3D models.
Tessellation	A process where one or more geometric shapes are added together without overlapping each other or creating gaps.
Slicer	A program that translates a model into material layers throughout the z-axis, to create a pattern for the printer to follow.
FDM (Fused Deposition Modeling)	A 3D printing method where layers of materials are fused together in a pattern made by the slicer to create an object.

1 Introduction

Sensor technology is an ever evolving field, and several kinds of sensors are commercially available. Most buildings use temperature sensors, and motion sensors for automated lights. There are many kinds of sensors, and their uses are only limited by ones imagination.

The development of artificial sensors kicked off in the mid 1900's. The first sensors developed were piezoresistive accelerometers and pressure sensors, and the number of different kinds of sensors have only increased since [1]. Sensors are used everywhere these days, and most are commercially available. They vary in accuracy and price, and most can be used with a micro-controller.

The availability of the sensors makes them valuable and useful for students and developers, as it gives practical experience.

With global warming and climate change affecting the earth, sensor data is becoming a valuable asset. The earths temperature has been monitored since the early 1900's, and the data helps climate scientists with determining changes on earth [2]. Sensors are also used for monitoring bodies of water. Temperature and how viable the waters are, are common variables to monitor, as it can help determine how the creatures in the waters are affected.

This bachelor project is related to this issue, as the project is the development of an underwater sensing rig. The sensing rig will be equipped with four sensors that can monitor bodies of water. The project is given by IKT, the Department of Engineering Cybernetics, and is a further development of their ROV2.

The goal is to make a fully functioning sensing rig that can operate on its own, or be attached to the ROV2. When the rig is used on it's own, it will be portable and small enough to fit in smaller bodies of water, such as lakes and fish farms. By using the rig in fish farms, it can help the fish farmers with monitoring the fish's environment.

1.1 Thesis Assignment

NTNU

Institutt for elektroniske systemer
Institutt for elkraftteknikk
Institutt for teknisk kybernetikk

Oppgaveforslag bacheloroppgave elektroingeniør (BIELEKTRO) i Trondheim, vårsemester 2022

Navn bedrift: **Institutt for teknisk kybernetikk**	**Kontaktperson: Damiano Varagnolo** **Epost: damiano.varagnolo@ntnu.no** **Telefon/mobil: 48128922**	
Tittel på oppgave: ***Developing multipurpose Underwater Sensing Rig***		
Hvilke studieretninger passer oppgaven for? (kryss av for alle aktuelle retninger; flervalg er mulig):	**Automatisering og robotikk**	x
	Elektronikk og sensorsystemer	x
	Elkraft og bærekraftig energi	
Er oppgaven reservert for noen bestemte studenter? I så fall skriv navnene på studentene til høyre.		
Er dette en lukket oppgave? Dvs. at sluttrapporten ikke kan publiseres senere fordi den inneholder sensitiv informasjon.	[ ] ja [x] nei [ ] ikke enda bestemt	

Kort beskrivelse av oppgaven med problemstilling.

Sensing and collecting the required data is an essential step for any post-processing in research and industry. But the data acquisition can also be challenging due to environmental conditions. On the other hand, the underwater environment is one of the most challenging environments regarding accessibility, natural conditions such as pressure or darkness, communication limitations. Considering all the mentioned facts, developing a sensing rig for the underwater environment can be valuable for many different kinds of research. The target of this project is to complete the development of an underwater sensing rig. The developed rig would be carried by Remotely operated vehicles (ROVs). In this project, the sensing rig should be mounted on BlueRobotics BlueROV and also should be able to announce all the measured sensing values to ROV.

During this project you will learn and get familiar with the following topics and areas:
1- Underwater sensor hardware and sensing condition.
2- Hardware design and development.
3- Robot Operating System (ROS)
4- Developing all the required code for reading the sensor values and publishing it via the ROS.

1.1.1 Criteria and narrowing of project

The goal set by the employer is to make a fully functioning sensing rig, which was split into three main parts. These main parts were the four working sensors, the ability to read and monitor battery health, and have a 3D-printed hardware layout inside the aluminum casing. Additionally, learning to program with ROS2 in the Ubuntu OS was a requirement in this project.

The project will not focus on communication, data analysis, or compatibility with the BlueROV2.

1.1.2 Different prerequisites in regards to 3D modeling and printing

The employer also had several inquiries regarding the 3D modeling and printing, and these are the following. The layout should contain all the electronics on the top side of the main plate, and the barrier strips moved to the end of the main plate. The circle placed at the front of the main plate should attach to the lid of the casing, for easier access to the system. There should also be a railing system on the other side holding the Raspberry Pi in place, in a more modular solution.

Finally, it was desirable that there were used as few different screws as possible for the whole system.

1.2 Achievements

This is a quick summary of the main achievements the group had after completing the project.

The group learned programming in ROS2, and the sensor code for three of the sensors were completed. The group also learned 3D modeling and printing, and the 3D model was successfully redesigned and printed.

1.3 How the report is structured

The three main goals were worked on in parallels throughout the semester, and therefore the report is structured in the same way.

Each chapter contains a main theme, such as method, results, discussion, or conclusion, and each section is split into the three main subsections. This keeps the three main themes separate in the report, while also keeping the report organized, and predictable.

Chapter 1 is the introduction.

Chapter 2 is about the background and theory for the project. The previous work and all of the necessary theory is presented and explained here. The chapter also presents what knowledge the members of the group had before working on the project, and also what the group needed to learn.

Chapter 3 focuses on the development of the software and hardware aspect of the project, which includes code for the sensors and battery, and the testing conducted on the hardware. The chapter explains how the group found relevant information, and the process behind writing and testing the code.

Chapter 4 is about 3D modeling and printing. No one in the group had worked with 3D printing before, so this chapter explains 3D modeling and printing from a beginners perspective. The various methods of 3D modeling and printing are explained, and the chapter elaborates on how each 3D modeled part was made.

The results are presented in chapter 5. This chapter contains the results from various tests done throughout the semester, and the final results from the 3D printing.

Chapter 6 is the discussion. This chapter is a reflection of how the project was completed, and discusses topics such as the preliminary report, what choices were made in regards to hardware and software, 3D modeling, what the results indicate, and future improvements and developments.

The final chapter is the conclusion, and summarizes the main results in the report, and what the group thinks about the completion of the project.

The appendixes are added after the conclusion, in the following order. The first appendix contains screenshots of the publishing code for the sensors. The second appendix has the screenshot of the measurements taken during the functionality test of the sensors. The third appendix is the table with all of the screws. The second to last appendix is a 3D printing procedure that was requested from the employer, and is an informal walkthrough of 3D printing on the Ultimaker S5. The last appendix is the A3 Project Poster, designed by the group.

2 Background and Theory

The project is a further development of a sensing rig compatible with a BlueRobotics BlueROV2 [3], and requires knowledge on previous work.

The bachelor group consists of three electrical engineering students, two with specialization in electronics and sensor systems, and one specialized in automation and robotics. The group was familiar with some of the theory regarding the project, but a lot of new knowledge was needed.

This chapter is split into two main parts. The first part is about the background of the project and the method used to gain the required knowledge. The second part is the theory about the technical aspects, such as software and hardware.

2.1 The previous work

Previously there have been someone else working on the sensing rig, and the group had access to their work. They had a GitHub[4] with the basic code for the sensors, as well as a quick guide on how to use a testing program with ROS2[5].

Parts of the 3D printed layout inside the aluminum casing was also previously made, and the groups task was to improve it. The main plate is inspired by the electronic tray that was developed by BlueRobotics [6], but ended up being redesigned by a former student. Other things that were already made by the former student was a rail, a circle, a semicircle and a spring. All of these components were used as inspiration to the redesigned parts made in this project.

2.2 Gathering information and knowledge

The project required new knowledge about robotics, Linux and 3D modeling. Through previous studies these concepts were introduced, however the group had not worked specifically with this.

2.2.1 Learning Ubuntu

Ubuntu[7] is a Linux based operating system, and is the OS used on the Raspberry Pi 4b.

Because no one in the group had any prior knowledge about ROS2 or Ubuntu, the group needed to gather information. As Ubuntu is a Linux based OS, it uses many of the same commands and has the same structure as regular Linux. Linux is also an open source OS, so many internet forums were used to learn tips and tricks to navigate the OS faster.

ROS2 is not very compatible with Windows OS, so the group decided to switch OS to Ubuntu on two of the group members personal computers, so the group could learn and practice ROS2 without being dependent on the two Raspberry Pi's the group had access to.

2.2.2 Learning ROS2

To help the group with ROS2 and Ubuntu, the employer set up a crash course at the start of the semester, where the basic theory and operations was introduced.

The employer also recommended doing the intro course on ROS2 Galactic, which had more detailed information and follow-along tasks[8].

An important aspect of learning ROS2 for this project would be understanding what nodes are, and how they work. Nodes in ROS2 are able to send and receive data to other nodes via topics, services, actions or parameters.

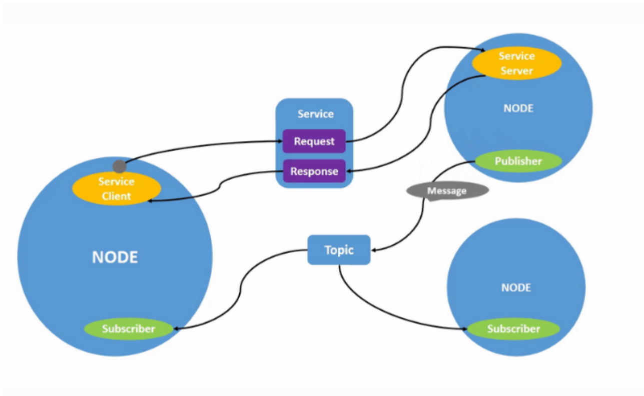


Figure 2.1: Nodes communicating

In this project the group will be using the topics method, where it is crucial to differentiate between a publisher and a subscriber node. A publisher node creates and sends data to a given topic, whilst a subscriber node "subscribes" or reads the data from any given topic. This form of communication between nodes doesn't have to be point-to-point communication, as it can be point-to-many or many-to-many communication.

2.2.3 Learning 3D modeling and printing

3D printing consists of both modeling and printing, which means that there are a number of things to get acquainted with. First it is important to choose a good 3D modeling program. In this project, the Computer-Aided Design (CAD) program SketchUp was used.

Before the designing and modeling started it was important to learn the basics of 3D modeling, which includes concepts such as face, edge and mesh. This was learned by reading various articles on 3D modeling [9]. This also included getting an insight into typical beginner mistakes.

When the basics of the 3D modeling was understood, the next step was to learn SketchUp. Since SketchUp is a popular program for 3D modeling, there is a lot of documentation on how it is used and what it can be used for. The most efficient way of learning is to watch videos [10] for direct explanations and tutorials, but simply trying it out was also a good learning strategy.

SketchUp is only the modeling program, and the printing part is directly tied to the specific printer and corresponding slicer program. The 3D printer in this project is the Ultimaker S5, and uses the slicer program Cura. Cura prepares the model from SketchUp and translates it into a printable model, and this requires various settings of different importance. This program was also learned by reading articles and watching YouTube videos [11] [12].

2.3 Technical theory

To better understand how the project was developed, some background theory is needed. In this section the basic theory of the electrical components are introduced. All of the components were chosen by the employer, and how the electronics should connect with one another was already decided. The task was to make the components work together.

2.3.1 Raspberry Pi

The Raspberry Pi 4b is a microprocessor based mini computer that can be used as a computer, but with the possibilities of a micro controller. This means it can connect with sensors and other electronics directly on its pins [13].

The Raspberry Pi's operating system is flashed (downloaded) to the micro SD card, and this is where all of the data is stored.

Figure 2.2: Raspberry Pi 4b

2.3.2 Sensors

This section is centered around the four different sensors. These being sensors for measuring temperature, pressure, conductivity and dissolved oxygen.

Temperature

The temperature and pressure sensors are both developed by BlueRobotics. The Celsius Fast-Response, ±0.1°C Temperature Sensor (I2C) [14] is sealed from the water and encased in aluminum. The sensor itself is protruding from the bolt, and protected by the cage, and the sensor is the TYSYS01, manufactured by Measurements Specialties. The TSYS01-family provides factory calibrated information, has an operating temperature between -40 to +125°C, and can function to a depth of about 975m.

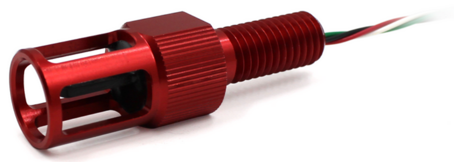


Figure 2.3: BlueRobotics temperature sensor

Pressure

The Bar30 High-Resolution 300m Depth/Pressure Sensor measures with a depth resolution of 2mm. The sensor is the MS5837-30BA and communicates over I2C, and is manufactured by Measurements Specialties. As with the TSYS01-family, it comes pre-calibrated [15].

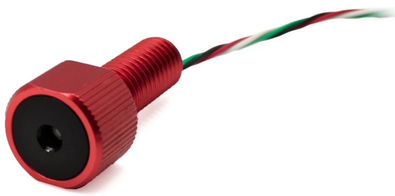


Figure 2.4: BlueRobotics pressure sensor

Oxygen

The sensor used to measure dissolved oxygen is also developed by Atlas Scientific. The dissolved oxygen kit consists of 1x EZO Dissolved Oxygen Circuit, 1x Lab Grade Dissolved Oxygen Probe, 1x 30ml bottle of Electrolyte Solution, 3x 20ml Zero Dissolved Oxygen calibration solution pouches, 1x Electrically isolated EZO Carrier Board, and 1x Syringe.

Figure 2.5: Atlas Scientific dissolved oxygen sensor

The galvanic dissolved oxygen probe from Atlas Scientific consists of a PTFE membrane, an anode soaked in an electrolyte, and a cathode. It works by having oxygen molecules defuse through the membrane at a constant rate. When the molecules cross the membrane they meet the cathode, where they are reduced and produce a small voltage. The produced voltage is directly proportional with the amount of oxygen molecules reducing at the cathode. If there's no oxygen molecules present, the probe will output 0mV. Different probes will output a different voltage in the presence of oxygen, and only when there is no oxygen molecules will all probes read 0mV. One of the downsides to using a galvanic probe is the fact that it will consume a very small amount of oxygen. Hence, to get accurate measurements it is preferred to have little to no water movements.

Conductivity

The Conductivity K1.0 kit from Atlas Scientific were used for measuring the conductivity in this project. This kit includes 1x EZO Conductivity Circuit, 1x Conductivity Probe, 1x Electrically Isolated Carrier Board, and 2x Calibration solutions.

Inside the conductivity probe are two electrodes positioned opposite from each other. When an AC voltage is applied to the electrodes, it causes the cations to move to the negatively charged electrode while the anions move to the positive electrode. A higher level of free electrolytes in the liquid translates to a higher electrical conductivity [16].

Figure 2.6: Atlas Scientific conductivity sensor

2.3.3 Battery

The battery for the sensing rig is a 22.2V, 6C lithium ion battery, from the manufacturer Tattu Gens [17]. It has a discharge rate of 15C, and has AS150 + XT150 Plugs. The sensing rig is not dependent on any specific brand of battery, and the code is meant to be modular to reflect this.

The battery health, or lifespan, is in this project defined as how much power is left on the battery.

Figure 2.7: Tattu 12000mAh 22.2V 15C 6S1P Lipo Smart Battery Pack

2.3.4 Interface electronics

To make the electrical components able to connect with each other, several interface technologies were needed. For easy use of the components, the wires and cables are color coded with red as voltage (or HIGH) and black as ground (or LOW).

I2C Bus Splitter

The BlueRobotics I2C Bus Splitter is a circuit board that allows connection to multiple I2C devices by sharing the same bus. The splitter consists of four header pin holes, four DF13 connectors, and four JST-GH connectors, making it compatible with multiple devices with different headers.

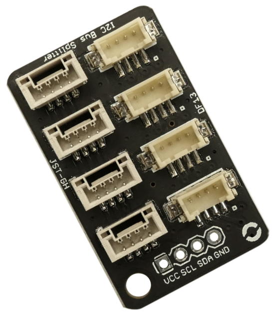


Figure 2.8: BlueRobotics I2C Bus Splitter

I2C is a synchronous communication protocol between a master and a slave. This form of communication only needs two pins to operate, a serial clock (SCL) pin, and serial data (SDA) pin. The master provides the clock, which effectively becomes the clock frequency. I2C is a bidirectional bus meaning that the master can write to the slave, and receive data from the slave. It is a serial bus, meaning the data is clocked or shifted bit by bit. Each "message" is built up the same, and are broken up into frames of data. Each message includes start and stop conditions, an address frame containing the binary address of the slave, read/write bits, ACK/NACK bits between each data frame, and one or multiple data frames containing the transmitted data [18].

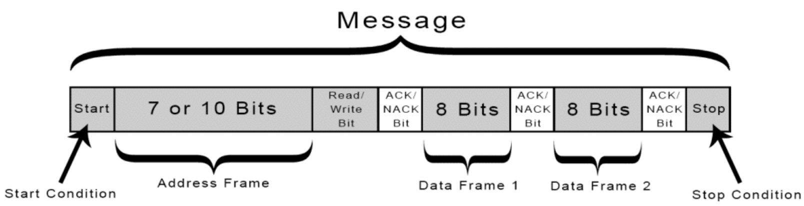


Figure 2.9: I2C message

Power Supply

A converter is needed to connect the battery to the Raspberry Pi, as the Raspberry Pi has a maximum voltage rating of 5V. This Power Supply is by BlueRobotics, and converts the voltage from 22.2V down to 5V. It is connected with jumper cables on the Raspberry Pi, and spades to the barrier terminal strip on the main plate. [19].

Figure 2.10: BlueRobotics Power Supply

Power Sense Module

The Power Sense Module (PSM) is used for measuring the voltage and current levels in the battery, and it transforms the measurements to values an ADC can read [20].

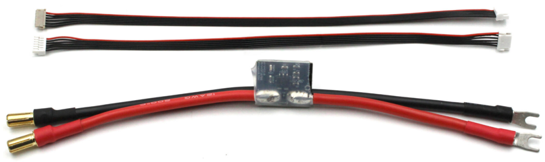


Figure 2.11: BlueRobotics Power Sense Module

ADC

The Analog to Digital Converter converts the readings from the PSM to readable levels. The Adafruit ADS1115 has a 16-bit resolution, and can be configured as four single-ended input channels, or two differential channels. The board uses I2C 7-bit addresses between 0x48-0x4B, which can be selected depending on the state of the ADDR pin [21].

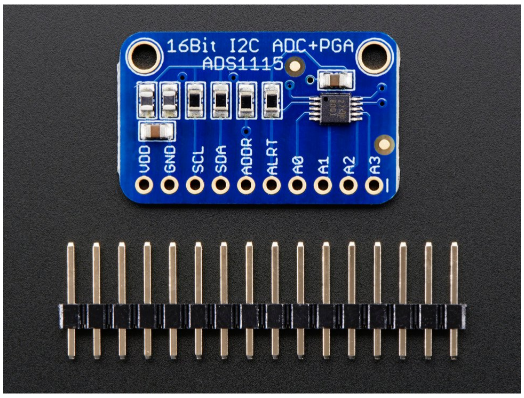


Figure 2.12: Adafruit ADS1115

Barrier strip

The barrier strip is located on the back of the main plate, and has screw terminals that can connect to spades.

2.3.5 3D modeling and printing

3D modeling is a method of producing a digital and mathematical representation of an object or surface in three dimensions [22]. The objects are represented by a collection of points in 3D space, and a single point is called a vertex.

Two vertices with a line between them is called an edge, and with more vertices combined with edges forms a polygon, also called face. The face is the space between the edges. They can be planes or curved shapes, and can be compared to surfaces [9].

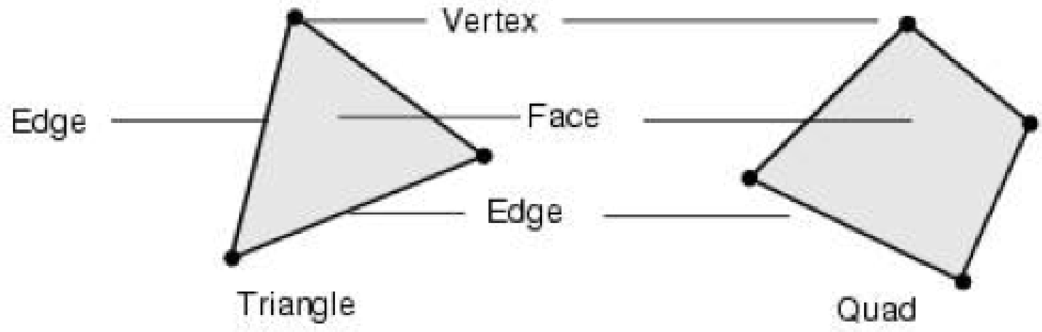


Figure 2.13: Vertex, edge and face. Picture from [23]

An object in 3D space contains a collection of polygons that are connected in their faces, edges and vertices. This collection in called a mesh, and a 3D object consists of one or more meshes. The quality of the mesh is determined by how many polygons the object contains. The more polygons added, the more detailed the object becomes.

There are different methods to use when 3D modeling [24]. Among them are solid modeling, which is a type of modeling that works with three-dimensional shapes. These shapes vary, but they all act like building blocks that can interact with each other. The objective with this type of modeling is to ensure that every surface is geometrically correct. It also consists of topological data of weight, shape, size and connectivity of nodes, edges, and faces [25]. This is a complex way of modeling because the CAD software have to simulate the object from both the outside and inside. Because this type of modeling is geometrically correct and contains topological data, it is suitable for designing products, and 3D printing.

Another method for 3D modeling is surface modeling. This is a type of modeling that have the ability to build out a visual representation of an objects exterior and its contours [26]. The object can be geometrically and physically incorrect with no topological data, meaning that the object can be modified in ways solid models are incapable of. Surface modeling also have the ability to build out faces individually, so control over exact contour and direction is easy. There are other methods for 3D modeling, but they are not relevant for this project.

SketchUp is the 3D modeling program, or CAD program, used in this project [27]. This program uses a combination of solid and surface modeling [28][29]. When making a new object or redesigning an already made object, the surface method is used. The faces, vertices, and edges can be modified without changing the whole object. When the editing of the object is done, the object can be converted into a solid by making the object into a group [30]. This means making all the faces, edges, and vertices one consistent object.

In order to 3D print the 3D model, the file with the model must be exported as a STL file [31]. STL is an abbreviation of the word *stereolithography*. Another name for this type of file is “Standard Triangle Language”. A STL file describes only the surface geometry of an object in three-dimensions, without any representation of common CAD model attributes as texture and color[32]. For the file to store the surface geometry of an object it uses a concept called "tessellation". This is a process in which one or more geometric shapes are added together without overlapping each other or creating gaps. The surface of a 3D object is described by tiny triangles, also called faces, and the information about the faces are stored in a STL file.

To 3D print the object, the STL file must be opened in a slicing software. A slicing software is a program that converts digital 3D models into instructions for the printer [33]. This means the slicer chops up the STL file into many flat layers based on the object. The settings in the slicer program decides the quality of the print. Such settings are for example the layer height, infill percent and support. When slicing a model, all the information is written to a G-code file, which creates a path and specifications for the 3D printer.

As mentioned before, the 3D printer used in this project is the Ultimaker S5, which uses the slicer program Ultimaker Cura. This printer uses a method called fused deposition modeling or FDM. This is a method where a plastic material is being squeezed through a hot end, melting it, and deposing it in layers that are being fused together in a pattern created by the slicer. These layers build up and will eventually form a finished object [34].

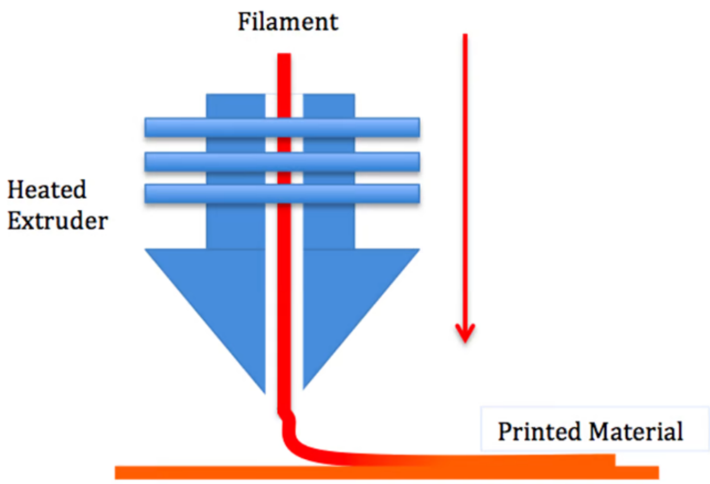


Figure 2.14: FDM. Picture from [35]

The type of plastic used in this project is PLA, which is a popular, cheap, and easy to print material [36].

3 Creating the software and hardware solutions

This chapter focuses on the electronics' hardware and software, with the chapter split into two parts. First the software portion is presented with a walkthrough of the temperature code. And the second part consists of hardware calibration and preparation, and the performed tests.

3.1 Method

The code for the sensors and battery is written in Python[37]. The programming language has been taught in various subjects since the first semester, and everyone in the group has a good understanding of the language.

The most prominent method of programming was by trying and failing. By having a general sense of what the code should do, writing and debugging becomes easier and more intuitive. With the sensors available, the group was able to verify if the code worked with the sensors in real-time, which also sped up the coding process.

3.2 Code for sensors

All of the code used in the project, and how to run said code can be found on Github [38].

There are in total four different sensors that need individual codes. Although the sensors are very different, the code for each of them are akin to one another. This is because they are all publisher nodes and thus share the same structure. To demonstrate how the code is built, the temperature publisher code will serve as an example.

When coding with ROS2, a package must first be created. A *package* is like a container for the code with all its necessities. Packages can be made in either CMake or Python. The amount of files needed will vary depending on the chosen programming language. These necessary files for programming in python are presented in table 3.1:

package.xml	Contains meta information about the package
setup.py	Contains instructions on how to install the package
setup.cfg	Required when a package has executables so that ROS2 can find them
/<package_name>	File directory sharing the same name as the package, used by ROS2 tools to find the package. Directory contains code for the publisher as well as an *__init__.py* file

Table 3.1: Table with the necessary files for programming in python

With the essential files in place, the publisher code can finally be written.

The code shown in appendix **8.1 *Publisher code for Temperature sensor*** is the entire code in charge of setting up the publisher and getting the temperature data. A run through of the code is written below.

```
# This code is responsible for getting the data from the sensor and publishing
    it.
from rclpy.node import Node

# tsys01 needed in order to utilize the BlueRobotics TSYS01 Python Library
    which must be installed
from sensor_thermometer import tsys01
from sensor_interfaces.msg import Thermometer
import time
import re, uuid       # Used for getting the mac address
```

The first step is to import the necessary libraries. By importing *Node* from rclpy, library nodes can be used. The *tsys01* library is for using the sensor. The next statement imports the built-in string message type that the node uses to structure the data that it passes on the topic. *time* library is used for extracting the systems local time. Libraries *uuid* and *re* are used for fetching and formatting the MAC address respectively.

```
class ThermometerDataPublisher(Node):
    # Initializer
    def __init__(self):
        super().__init__('ThermometerDataPublisher')
        self.publisher_ = self.create_publisher(Thermometer, 'thermometer_data',
            10) # Creates a publisher over the topic thermometer_data
        read_period = 2 # Does a reading every 2 seconds
```

```
        self.timer = self.create_timer(read_period,
            self.thermometer_read_and_publish)

        # Assign the function TSYS01() in the tsys01.py to self.sensor
        self.sensor = tsys01.TSYS01()
        if not self.sensor.init():
            # If sensor can not be detected
            print("Sensor could not be initialized")
            exit(1)
```

Next, the *ThermometerDataPublisher* class is created, which inherits from (or is a subclass of) *Node*. By having a class inherit from another class allows for reuse of code, as all the attributes of *Node* class is then available to *ThermometerDataPublisher* without needing to redefine them [39]. Following, *super().__init__* calls the *Node* class' constructor and gives it a node name chosen by the user, which in this case is ThermometerDataPublisher. *create_publisher* declares that the node publishes messages of the type Thermometer which was imported earlier from sensor_interfaces.msg. These messages will then be published over the topic named thermometer_data, with a queue size of 10. Following that, a timer is created with a callback to execute every two seconds. [40]

```
    def thermometer_read_and_publish(self):
        # Custom thermometer message to publish. Can be found in the
            brov2_interfaces.
        msg = Thermometer()

        # Geting the local time
        tim = time.localtime()
        msg.local_time = time.strftime("%H:%M",tim)

        # Getting the mac address of the system:
        msg.mac = ':'.join(re.findall('..','%012x' % uuid.getnode()))

        # Calls the function sensor.read in TSY01 to get the desired values from
            the sensors
        if self.sensor.read():
                msg.temperature_celsius   = self.sensor.temperature()
                                            # Default is degrees C (no arguments)
                msg.temperature_farenheit =
                    self.sensor.temperature(tsys01.UNITS_Farenheit) # Request
                    Farenheit
        else:
                print("Sensor read failed!")
                exit(1)

        # Publishing message and logging data sent over the topic
            /thermometer_data
        self.publisher_.publish(msg)
        self.get_logger().info('Mac: %s T: %0.2f C %0.2f F %s' % (msg.mac,
                                                              msg.temperature_celsius,
```

```
                                    msg.temperature_farenheit,
                                    msg.local_time))
```

In step 3, the function *thermometer_read_and_publish* is declared. Here data is read from the sensor and published to the topic, and also written to the console.

```python
import rclpy
from sensor_thermometer import thermometer_data_publisher_node as node


def main(args=None):
    rclpy.init(args=args)

    # Construct the publisher
    thermometer_data_publisher = node.ThermometerDataPublisher()

    # Reading and publishing data at defined rate (2 seconds)
    rclpy.spin(thermometer_data_publisher)

    # Clean up when script is stopped
    thermometer_data_publisher.destroy_node()
    rclpy.shutdown()


if __name__ == '__main__':
    main()
```

Lastly the main function *thermometer_data_publisher_node_main.py* can be defined, which is in a file by itself. First the rclpy library is initialized. Then the publisher node is created. Lastly, it "spins" the node. "Spinning" a node allows it to keep running and check for events.

All of the packages that are used in ROS2 has to be built. Then the ROS2 installation workspace is sourced in a chosen work terminal. The code can run one by one after finishing, by writing "ros2 run <package_name> <executable_name>". This method is slow, especially in systems where there are multiple nodes we wish to run. Because of this, using the launch file method works well. A launch file executes all nodes and their configurations simultaneously.

3.3 Calibration of the Atlas Scientific sensors

This section describes the methods used for calibrating the two sensors. The sample code used for the calibration can be found on their website [41].

3.3.1 Dissolved oxygen sensor

Just like the conductivity sensor, it does not matter whether the calibration is done in UART or I2C, and this section describes the calibration in UART. First step in the calibration is running the code and letting the probe sit exposed to air. Once readings have been stabilized, write “cal” in the terminal. Step two involves using the zero dissolved oxygen solution pouches. Open the pack and hold the probe in the liquid until readings stabilize, which takes around 0:30-1:30 min, and then write “cal,0” in the terminal. After this there should be readings of 0 when the probe is in the pouch. The sensor needs to be in I2C mode in this project, and the I2C mode is set by writing “I2C,97” in the terminal [42].

3.3.2 Conductivity sensor

Before the sensor can be properly used it needs to be calibrated. The group had two calibration solutions with different values, so a two point calibration for the widest range of accuracy was chosen. Single point calibration is explained in the official datasheet found on their website [41]. Default setting for the EZO conductivity circuit is UART mode, and if the LED on the circuit is cycling between green and cyan it is UART. If the LED is only cyan, then it is in I2C mode. Calibration was done in UART, as it doesn’t matter whether it is done in UART or I2C.

To calibrate, first pour the 12,880μS into a cup, and then put the probe into the solution. Make sure to shake the probe to remove possible trapped air. Once the readings have stabilized write “cal,low,12880” into the terminal. Do the exact same for the 80,000μS solution, but write “cal,high,80000” in the terminal instead. Remember to check the mode after calibration. This project needs I2C mode, and was switched by writing “I2C,100” in the terminal [43].

3.4 Testing the functionality of the sensors

When all of the sensor had their basic code, the functionality was tested. This test was done to ensure that different conditions does in fact affect the sensors, and the sensors react accordingly. Before the testing started the sensors were calibrated as accurately as possible, however the testing mediums were of an unknown value.

3.4.1 Equipment

The test required a container with water, some salt, the four sensors, and the testing program.

3.4.2 Method

The method describes how each sensor was tested individually, and all of the corresponding results are in section **5.1.1 *Testing the four sensors***.

Temperature

The first sensor tested was the temperature. The sensor was tested by holding it in the air, and then holding it in the container with warm water. After registering a change in temperature the sensor was taken out of the water.

During the test the sensor was monitored to make sure the exposed wires stayed dry.

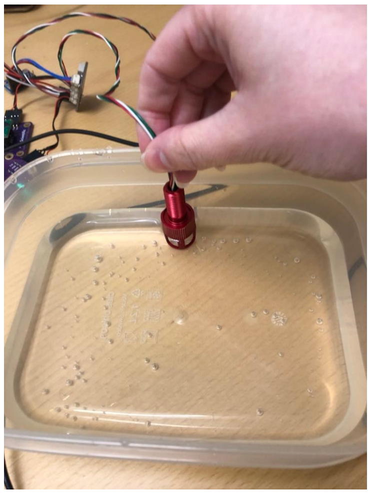


Figure 3.1: Functionality test temperature

Pressure

The pressure sensor was tested by holding it in the air, and then pressing in front of it using thumbs. This was done until the sensor registered a difference in value.

Dissolved oxygen

The test was conducted by placing the sensor in the container with regular, cold tap water. After the sensor registered the change in value, it was taken out of the water.

The sensor was also dipped in warmer water, because temperature affects the value.

Salinity

The conductivity was tested by holding the sensor in the container with warm water, and checking that it read zero μs/cm.

Salt was added to the middle of the container to focus the concentration of salinity. The sensor was then used to swirl the water around and help dissolve the salt. The sensor was then moved to the edge of the container where the concentration of salt was lower, and back to the center. Then the sensor was taken out of the water, and clean water was poured on the sensor to get rid of excess salt.

Figure 3.2: Functionality test dissolved oxygen

3.5 Code for the battery

The battery code is meant to monitor the charge in the battery, and the code uses an equation given by BlueRobotics

3.1 3.2 to convert the voltage readings to readable voltage and current levels.

The battery is connected to the barrier strip through the PSM. The PSM measures the voltage from the battery and converts it down to analog values between 0V and +3.3V. The PSM is connected to the ADC, and the ADC reads the voltage levels. Problems occurred with reading the values, and testing of the voltage over the pins were conducted.

$$V = ADC * (5.0/1024) * 11.0 \tag{3.1}$$

$$I = (ADC * (5.0/1024) - 0.33) * 38.8788 \tag{3.2}$$

3.5.1 Testing the voltage on the PSM

When measuring over the ADC with a voltmeter it read 12V. That is four times as much as the maximum output voltage, because the ADC can only handle a maximum rating of 5.5V. This shorted the first ADC, and new ones had to be soldered.

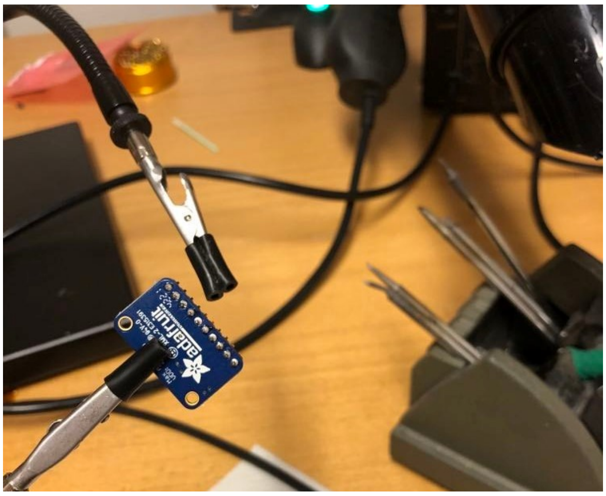


Figure 3.3: Soldering pins on the ADC

3.6 Rehousing the Atlas Scientific sensors

The Atlas Scientific sensors had to be rehoused, because the waterproofing enclosure was to small to go over the SMA housing.

This was done by first measuring an approximate length for the sensor that is outside of the casing , and a length for the inside, enough to reach from the lid to to the circuit boards.

The cable was cut at this length, and the cable was pulled through the BlueRobotics metal enclosure [44], making sure the difference between circumference and diameter was as little as possible, and also making sure it goes on the correct way. The rehousing of the SMA was done with help from the NTNU workshop, department electronics. The result from this is in section **5.1.2 *Rehousing the Atlas Scientific Sensors***.

Potting is a technique for waterproofing components. The BlueRobotics potting kit was used, and was done by the supervisor. The enclosures were clamped to keep the cables and enclosure stable while the epoxy resin was applied. The sensors were then left to dry over the weekend.

3.7 Testing the waterproofing of the cylinder

The purpose of this test is to connect all of the cables and components to the lid on the case, and check for air leakage. This test is a "non-destructive" test, so no electronics is at risk regardless of results.

The first step is to check that the hand pump works as intended. The pump is Mityvac Hand Operated Vacuum Pump [45], and the check was done by following the websites guide.

The rubber stopper was put on the end of the pump and the vacuum was pumped to 15 inHg, and held for 10-15 mins.

3.7.1 Adding the sensors/blanks

As the lid has ten holes on the top, and there was only 7 wires connected, the other holes was filled with blanks. These are just aluminum bolts that seal the unused holes.

First a rubber O-ring was added to the sensor, that acts as a barrier between the bolt and the lid, and it was greased with a dab of silicone grease. Then it was threaded through the hole, and the nut was screwed on with a wrench[46] to fasten the bolt. The placement of the different parts on the lid can be seen in figure 3.4.

3.7.2 Testing the vacuum

After filling all of the holes in the lid, the vacuum test was conducted according to the BlueRobotics guide [47]. The blue top on the pressure valve was taken off, and the hand pump vacuum plug was added. The vacuum was pumped to about 10 inHg, and the value was constantly monitored for changes in the value.

Figure 3.4: Lid with complete set of components

Figure 3.5: The hand pump connected to the valve on the lid

3.8 Wiring

From the battery there are two main cables attached to the PSM. There were three cables from the PSM, and the red cable, power, was connected to the barrier strip on the top of the main plate. The black cable, ground, was connected to the other barrier strip on the opposite side. The last cable going out from the power sensing module was the cable that connected to the ADC. This cable was connected to A0 and A1, but can also be connected to the other A-pins on the ADC. There were also two other cables connected to the barrier strips. These were the cables that were connected to the Power Supply. From the Power Supply there were cables with the converted voltage connected to the Raspberry Pi. These were connected to pins 2, 4 and 6 on the Raspberry Pi. In the Ethernet port on the Raspberry Pi, there was connected a cable going to a connector on the Ethernet switch. From the Raspberry Pi there are five cables to the I2C splitter. The cables were connected in this way: pin 1 on the Raspberry Pi to VCC on the splitter, pin

2 on the Raspberry Pi to SDA on the splitter, pin 3 on the Raspberry Pi to SCL on the splitter, and pin 5 on the Raspberry Pi to GND on the splitter.

In addition to being connected to the Raspberry Pi, the splitter was also connected to all the sensors and the ADC. There was a total of five cables connected to the connectors on the splitter. Among them were the cables to the two isolated carrier boards. The cables from the connectors connects to the five pins, where the red wire was connected to VCC on the isolated carrier boards. The ADC is connected to a connector on the splitter with a cable with four wires. The red wire is connected to VDD on the ADC.

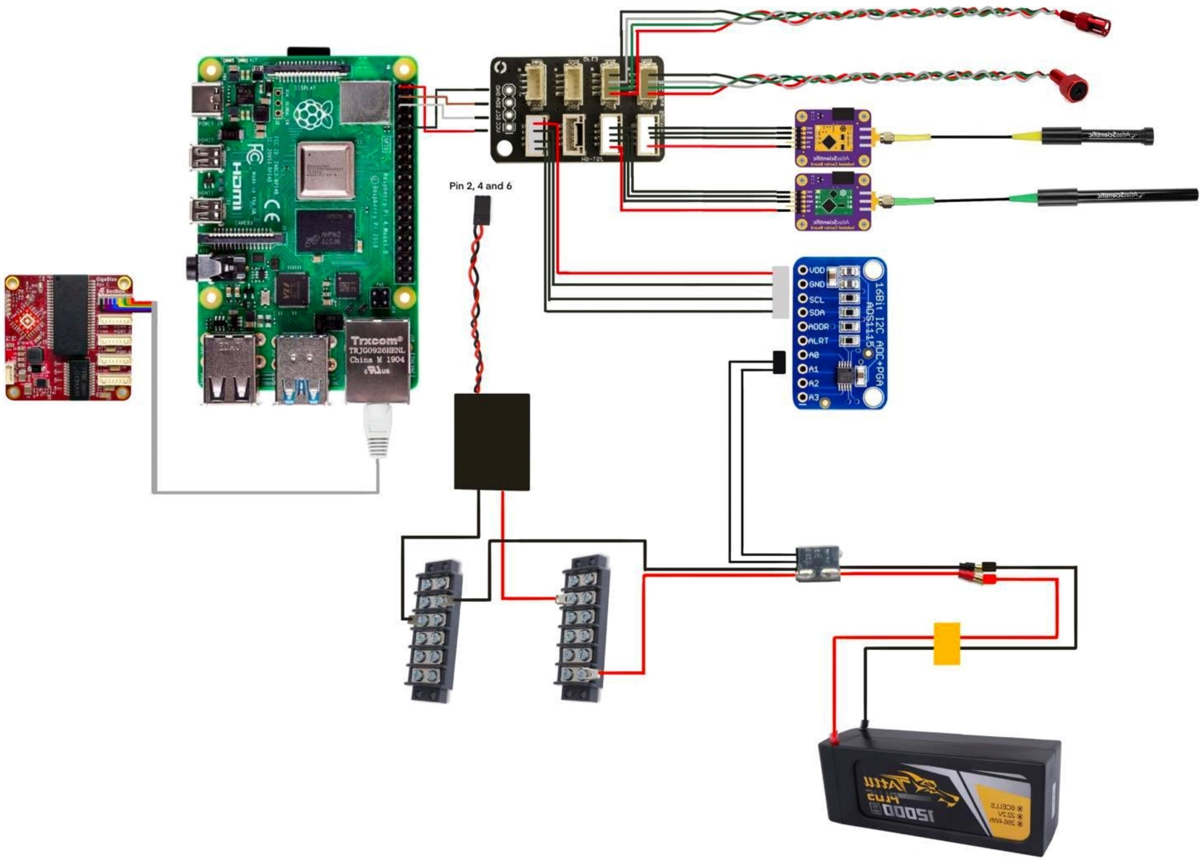


Figure 3.6: Overview of the wiring

4 Creating a 3D modeled hardware layout

The 3D modeling and printing part of the project consists of many separate steps. It starts with the 3D modeling where the object is designed in a CAD program, and then the model is exported to a slicer program to get all the parameters and settings needed for a good quality print.

This chapter will focus on the method for both the modeling and the printing, and there will be a description of every modeled part in the system. These models being the plate holding the Raspberry pi, the spring, the switch stand and splitter stand, semicircle, circle, and the main plate. There will also be a detailed section about the requirements and wishes, as well as how to model in SketchUp and how to choose the right settings in Cura.

4.1 Method

This chapter describes the wishes and requirements that were taken into account, and the various methods used to get the results that are presented in **5.3 *3D printing***.

4.1.1 Designing in SketchUp

The CAD program used in this project is SketchUp, as mentioned in **2.2.3 *Learning 3D modeling and printing***. This section will cover how to start 3D modeling, different tips, and the most used tools in this project.

When starting a new design in SketchUp, it is important to choose the right template. The models in this project are small, and most of the measurements are in millimeter, so the "3D printing" template was selected. After selecting the template, an editor in three dimensions will start, and a movable, transparent box that indicates the area of a 3D printer will automatically be added.

It is recommended to use a computer mouse to design things in SketchUp, because it makes it easier to orientate in the 3D space. Pressing the rotating wheel on the mouse acts as the orbit tool, without having to select it in the program. Rolling the wheel on the mouse will zoom in and out. Several keys on the keyboard also acts as shortcuts to various tools. An overview of all the shortcuts is in figure 4.1.

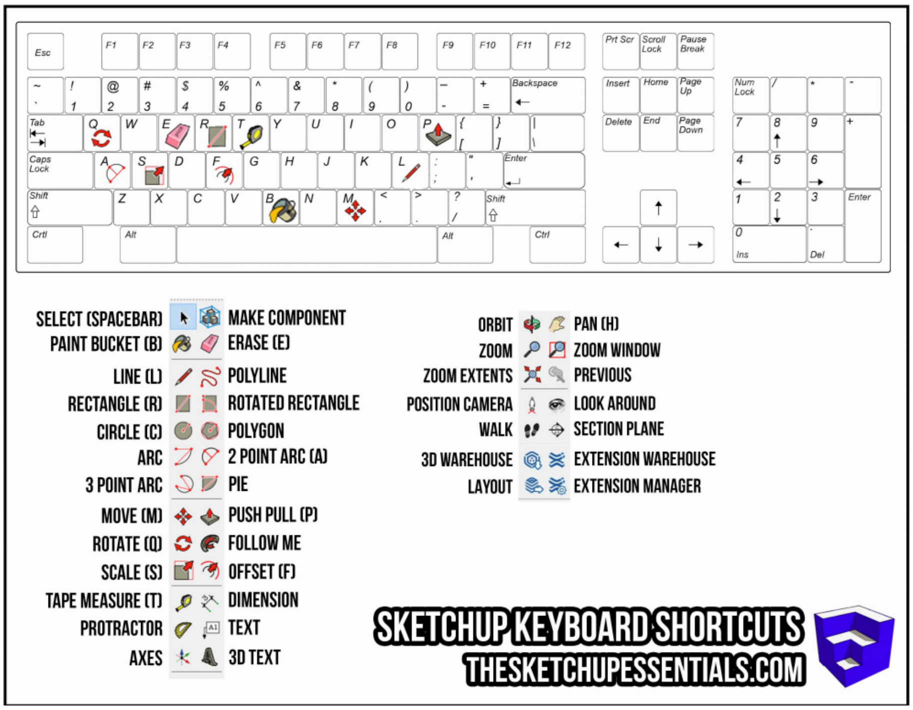


Figure 4.1: Overview of tools and shortcuts to them by The SketchUp Essentials [48]

The various tools in the toolbar above is used for editing in the three dimensional space. The line tool, draws lines which can connect and make a face. The circle tool makes circles, and the rectangle tool makes rectangles, all on one plane. By pressing the arrow keys while making a line, circle or rectangle, the shape can point in a special direction. The up arrow represents the blue axis, the arrow to the left is the green axis and the arrow to the right is the red axis.

When a line makes a face, or if a circle or rectangle is made directly, they will be represented in 2D as a plane. The push/pull tool can be used for transforming the 2D object to 3D. This tool drags the surface up or down, to create for example a cylinder. This tool is also used for making holes in a model, by creating a circle on a models surface and using the tool to push trough the model.

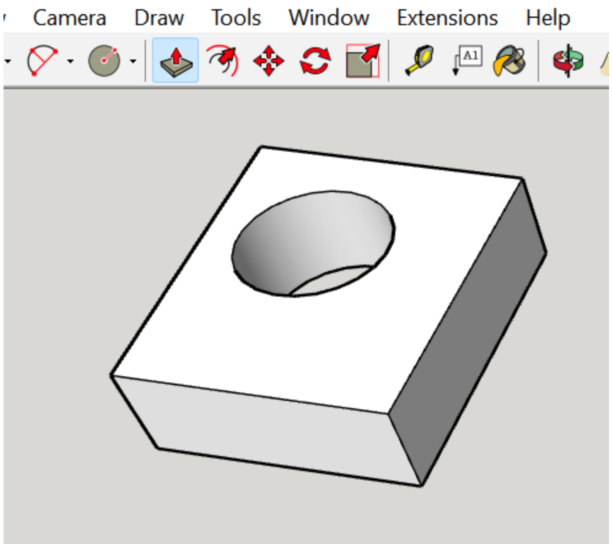


Figure 4.2: Puch/Pull tool to make hole

Another useful tool is the tape measure tool, and is used to find the right placement and dimensions. The tool can both measure distances, and set precise guide lines or guide points. Other useful tools are move tool for moving or rotating parts, eraser tool for removing parts, and offset tool to make for example circles bigger or smaller.

One thing that is important to keep in mind when modeling in SketchUp is to group parts when they are finished designing. The grouping works so that each line, each point and each face becomes one object. This allows for easier maneuvering, and when exporting the model to a STL file, it is less likely to be corrupted. Use the select tool to select the entire model, right-click on the model and press “Make Group”. To edit the model again, the model must exit the group mode. This is done by right-clicking it and pressing “Explode”. If you have a model that consists of many different groups, it may be worthwhile to press “Edit Group” instead of “Explode”.

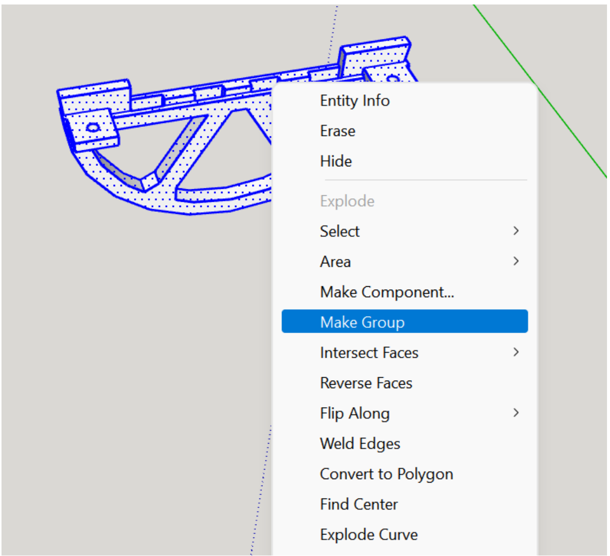


Figure 4.3: Make group

4.1.2 Models of the components

The physical electrical components were also accurately modeled in SketchUp to be used as guides for component placement in the 3D space. The measurements on the modeled components were made by either measuring the circuit boards by hand, or finding sketches online. Before modeling the circuit boards from scratch, attempts were made on finding already made 3D models on the internet.

The sketch of the Raspberry Pi was the simplest to find and use, and the detailed sketch was found on Raspberry Pi's website, as seen in figure 4.4. In the sketch the radius of the screw holes was stated, as well as where each component on the circuit board was located in relation to each other. A 3D model was found in SketchUps' 3D warehouse [49].

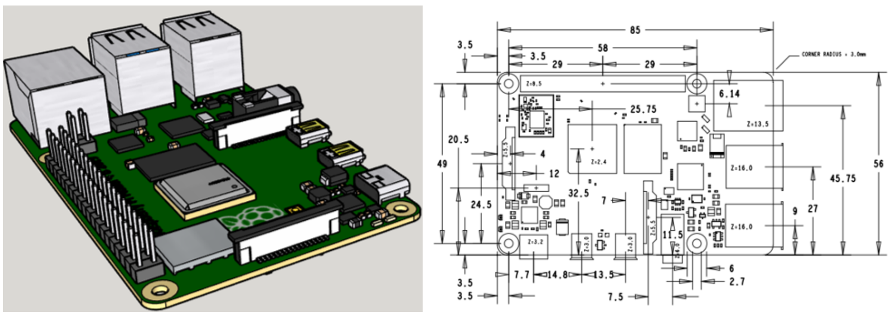


Figure 4.4: Raspberry Pi model and sketch from [50]

The Ethernet switch, on the other hand, had to be modeled because there were no already made models. A sketch of this circuit board was found on the website linked in figure 4.5. The sketch was less accurate than the Raspberry Pi, but it had some good measurements of width and length, as well as height on each side. Manual measurements were done to find the location of the components, and the holes on the circuit board. For safety reasons, a few extra millimeters were added to the model as measurements by hand can be inaccurate. The model was made in SketchUp with the tools mentioned in section **4.1.1 *Designing in SketchUp***. On the upper side of the circuit board, the highest components, and the connectors where included. Also the big component on the underside of the circuit board were included.

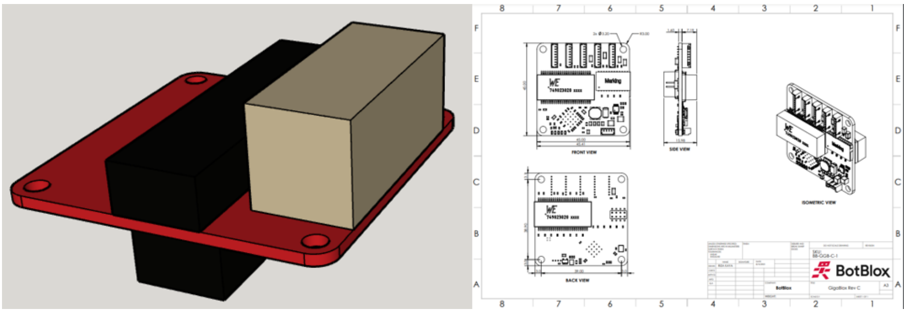


Figure 4.5: Ethernet switch model and sketch from [51]

There was also a sketch for the two isolated carrier boards, but no model. The sketch showed length, width, location of components, and screw holes. The only thing the sketch lacked was the height and placement of another small circuit board located on top of the carrier board. This was measured manually. The measurements was used to make the model of the circuit board in SketchUp, and the model includes the cable on one side, and the pins on the other.

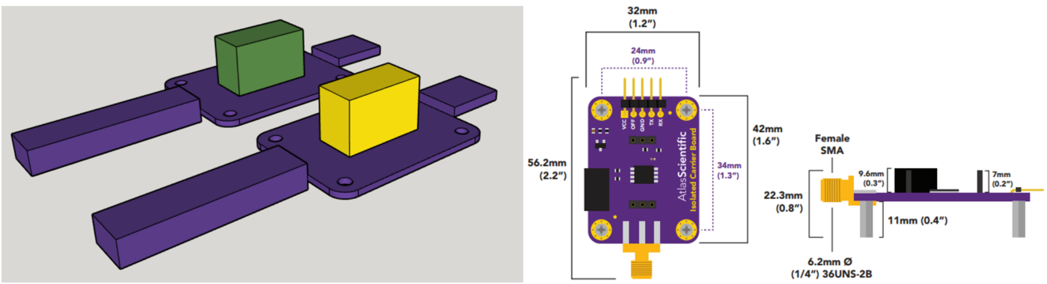


Figure 4.6: Both models of the isolated carrier boards, and the sketch from [52]

There was no sketch of the I2C bus splitter nor model. This meant that almost all measures had to be taken manually. The only dimensions stated on the BlueRobotics website were length and width. Otherwise, all other measurements such as height, location of components, and screw holes had to be taken manually. The model made in SketchUp considered the connectors and the rigid cables. The pins at the end of the circuit board would protrude on the back as well, so this was also modeled in the model.

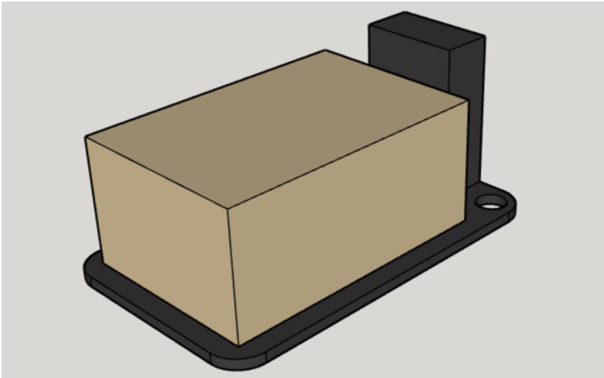


Figure 4.7: The I2C bus splitter model

The ADC ADS1115 had a sketch drawn in inches, and thus had to be converted to millimeters. The drawing stated several important dimensions such as length, width, and distance between the centers in each screw hole. But there was no diameter of the screw holes. This had to be measured by hand. The model consisted of the plate itself, the pins on both sides and the two holes.

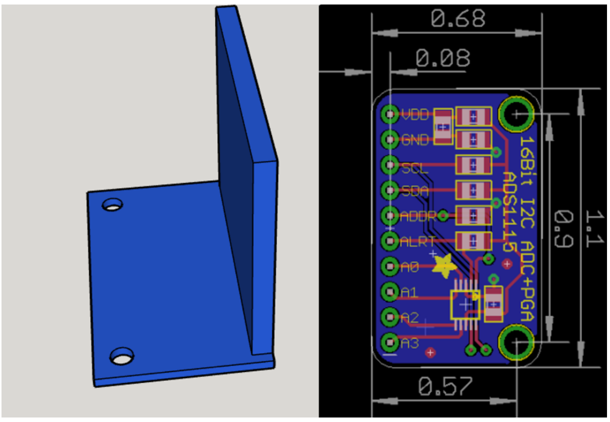


Figure 4.8: ADC model and sketch from [53]

Finally, a model of the barrier strips was made. A sketch with similar design and dimensions were found, and used to make the model of the barrier strip. The barrier strips used in the project also had some jumpers connected, and these were included in the model.

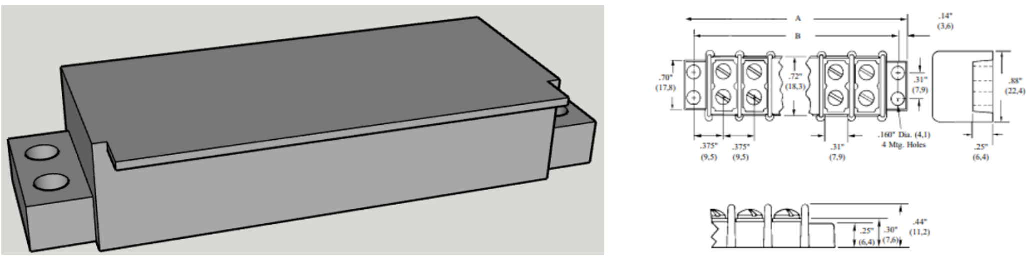


Figure 4.9: The barrier strip model and sketch form [54]

All measurements on all models were double-checked both with measurements from existing sketches, and manual measurements before they were used to design parts and deciding placement. All models were also colored as the corresponding components for easy identification.

4.1.3 Finding right screws

Finding the right screws that fit the different circuit boards was important. It was requested to use as few different screws as possible, and this had to be considered when the models were designed.

To find the right screws, an overview was made of which screw dimensions that fit in each circuit board. This was done by physically testing different screws on the different circuit boards. In addition to the circuit boards, there were other parts that had to be fastened with screws. These included racks, circle, semicircle and some plates for the Raspberry Pi.

When the diameters were chosen, the screw length also had to be found. Again, it was important to choose as many identical screws as possible. Location and depths of the holes, combined with screws already chosen, and whether they should be threaded up or not, were factored in when deciding.

An overview of which screws are used, and where they are located in the system are detailed in the appendix **8.3 *Table of screws***. This table was made after all the models were finished. Every hole using screws was given an individual number. These numbers were then marked on the models, and later placed in the table. The table describes which screws fits in which hole, and the length of the screw. There is also an overview of whether the holes should be threaded up, and which modeled parts the screws hold together.

4.1.4 Settings in Ultimaker Cura

In the slicer, Ultimaker Cura, there is a lot of different things to to be aware of, such as the settings [55]. In this section there will be a description about the most important settings, how to set these settings, and how to navigate trough the program.

When starting the program one will see the printers surface shown in the middle, and on the left side of the screen and at the top there is two menus. Before uploading a model, the right printer must be selected. The selection of printers are located far left in the top menu. For this project the printer Ultimaker S5 is used and therefore selected. In the middle of the menu one can choose filament material and print core. The filament used is PLA, and the print core is set to AA 0.4.

To the far right is the printer settings, which are auto-filled by the program when starting a new project. The settings that often get changed here is the infill density and infill pattern. When designing a model in SketchUp or other CAD programs, the model will have a outer wall and empty space inside, and the slicer program will fill these spaces with infill. The infill pattern can also be chosen in the settings. The pattern used in this project is triangles, and the infill percent, which decides how dense the model will be, were chosen depending on the model.

Other settings are Build Plate Adhesion and Support. The Build Plate Adhesion makes a thin plate at the start of the print. This is to prevent the model from moving during printing. For this project the Build Plate Adhesion type, brim type, was used. There are also settings for support. Support is a thin structures made of plastic, so the printer is able to create overhang. Supports was used on every print made in this project. Both the Build Plate Adhesion and Support setting, have their own detailed settings, but they are often set to default.

On the far left of the screen is another bar. Here the tools for changing size, moving the model, and other things are located. It is also here the change in infill of a chosen part of the model is. This is done by first using a "Support Blocker" located at the bottom of the menu on the left. The blocker is then placed in the desired place on the model by moving and resizing it. Moving and resizing can be found at the top of this menu. When the block covers the desired area, it can change its settings by selecting the block and then pressing "Per Model Setting" in the menu. Here, the "Modify settings for overlaps" is selected, and "Settings" pressed. Inside the settings, click "Infill Density" and then close the window. A box should have appeared at the bottom of the menu under "modify settings for overlaps" with the name "Infill Density". This is then set to the desired infill value [56].

4.2 Main plate

The main plate is the plate that holds all the circuit boards, barrier strips, and the Raspberry Pi. The plate had two previous design made by a former student. From now on the front of the plate will refer to where the rail is located, while the back is where all cables exit the system. Referring to the top means the side of the plate with all the components, while the bottom is the underside.

The first version was printed, and had a width of 3 mm. At one end there was placed a rail system with a thickness of 7 millimeters, and therefore protruded on the underside. The plate also had five holes for cables. The barrier strips were placed on the side edges on the top and bottom of the plate. There were also hooks on each side of the plate so that cables could easily be threaded through.

The other version of the plate made by the former student was not printed, only modeled in SketchUp. This model was very similar to the first, only that the thickness of the plate was slightly changed. It went from being 3 millimeters to 11 millimeters in the back, and 8 in the front. Another change was that the rail was separated from the model. Otherwise, all holes and hooks were in the same place.

The third version is the one used and made in this project. It is based on the previous models, in terms of length and width, as well as the location of some holes, but some changes were made. The second version was imported to SketchUp, and changes were done directly on this model. First, the thickness was changed to be 6 mm across the entire plate. Second, the two large holes at the back were moved a few millimeters forward. The three other holes that were placed on the old model remained in the same place on the new one. Third, the rail was integrated into this model so that both the plate and the rail became part of the main plate.

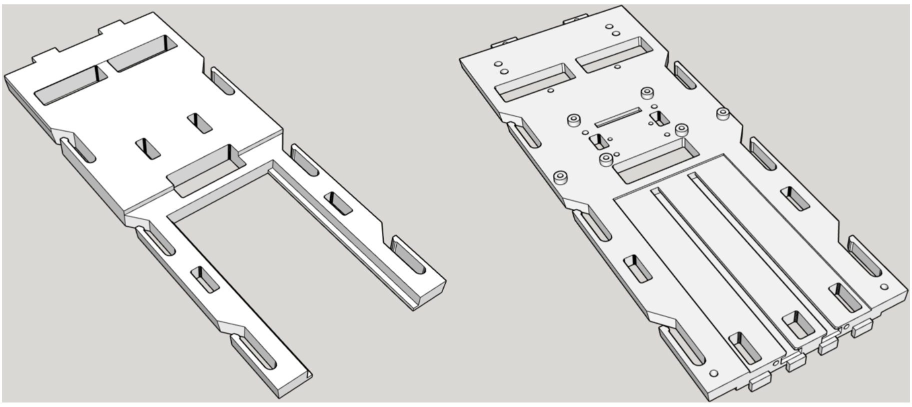


Figure 4.10: Comparison of the two last versions of the main plate. The one on the left is before and the one on the right is after

The cylinders on the main plate are made by first making circles, which are then pushed down with the push/pull tool. Then the offset tool is used to make a circle around the hole, which is pulled upwards again by using the push /pull tool. The holes on the main plate are made by using the circle tool, and then using the push/pull. For one of the components, there was made a rectangle with a depth of 2 mm by using the rectangle and push/pull tools.

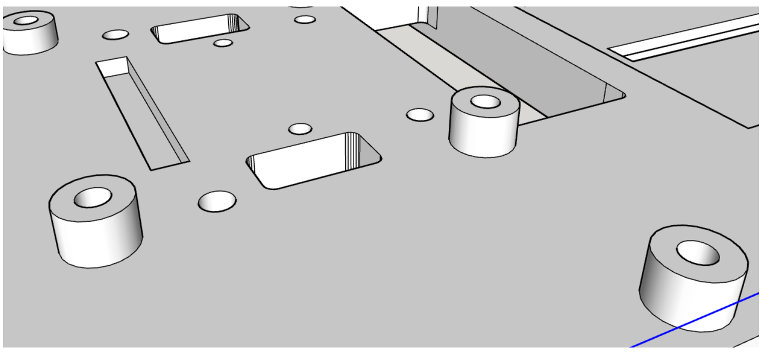


Figure 4.11: Cylinders and holes on the main plate

4.2.1 Rail

The rail itself on the latest version is quite similar to the previous model, but some changes were made. The opening in the middle was turned into a hole, and matches the other holes. Changes were also made to the four pieces that protruded on the front. These were extended downwards to be able to be attached to the semicircle. The width of the rail was also reduced as it was too large in the previous model. It was changed to fit perfectly in the opening on the main plate.

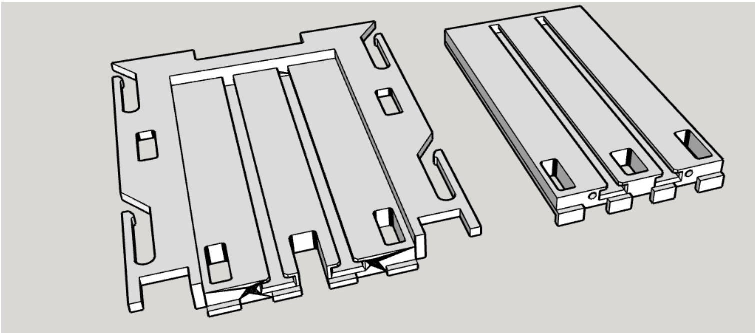


Figure 4.12: The rail before and after

4.2.2 Placement

Various circuit boards, barrier strips, rails, other plates, and stands were to be placed on the main plate. Therefore, it was important to meticulously plan the location of the various parts. It was also important to think about how the various circuit boards were connected when placing the parts. In the beginning, there was a desire to have all circuit boards on one side of the main plate, which was eventually changed. All the modeled components placements are shown in section **4.1.2 *Models the components***, with correct size and location of the holes. Every part was placed on the main plate and moved around to find the best positioning.

To be able to place all the circuit boards in the most efficient place the whole space had to be used, also the space above the main plate. There were made two different stands for two of the circuit boards. This was the splitter and the switch. The stands made the circuit boards raised above other circuit boards. Once the placement was set, holes had to be made to secure the various parts on the main plate. This had to be done by making an extra model in 2D. Here, the outline of each circuit board, stand and barrier strip was placed just above the main plate as a shadow, to see the correct placement of holes and other parts.

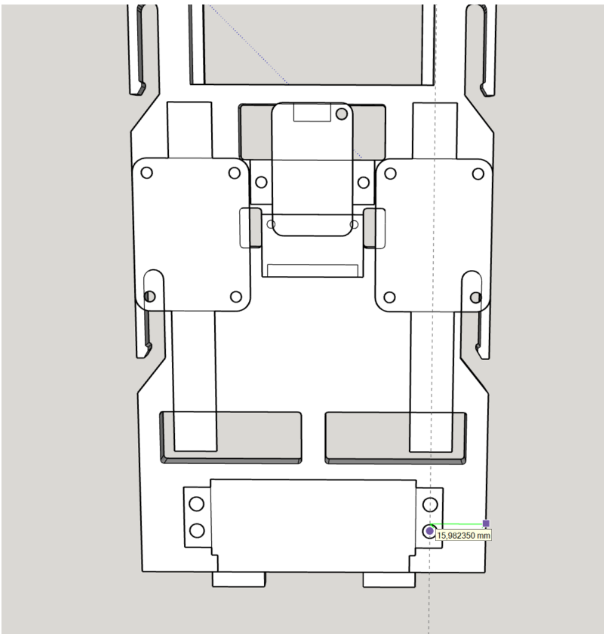


Figure 4.13: 2D model of placement

This model was then used to find distances that could be used to place the holes correctly on the main plate. To find the placement of the holes, the tool called tape measurement tool was used in SketchUp. This tool was used to measure how many millimeters it was from different starting points, such as the edge of the plate or the edge of a hole, to the center of the desired hole. This was done both horizontally and vertically to find the exact location of the center of the hole. This was first done on the 2D model where the holes already were correctly placed. Then almost the same thing was done on the main plate that needed to be modified. Here, the same tool was used to measure the same amount of millimeters as it was for the 2D model form the two starting points. This made two guidelines and where the lines met was the center of the circle. The tool circle was used to create a circle with the diameter that fit. This was done for all the parts that were to be placed on the main plate.

To be sure that all placements and sizes were correct, it was important to double check these. In addition, it was wise to put everything in SketchUp to check that everything is as planned.

4.3 Raspi-plate and spring/trackplate

As mentioned earlier, it was requested that the Raspberry Pi could be attached to a separate plate, called raspi-plate. This plate should be able to attach to either the spring modeled by a former student, or the plate with just tracks, which can be slid on and off the rail on the main plate. The plate that was going to be attached to the spring or trackplate had to be designed from scratch as this part did not exist. It was chosen to make a plate with walls that could be pushed on the spring. This was done by measuring the surface of the spring, and then using these measurements to find where the walls should stand. On the walls, a total of four holes were placed to screw the plate to the spring. Once this was done, there had to be a way to be able to attach the Raspberry Pi to the plate. Four cylinders were made exactly where the screw holes on the Raspberry Pi were located. This is to raise the Raspberry Pi so that the components on the underside did not hit the plate. In the cylinders there are holes that go through the whole plate, so that the Raspberry Pi could be fastened with screws that are fastened with a nut on the underside of the plate.

The spring also had a few changes. The changes were adding holes on the sides of the plate at the top of the spring, to attach the raspi-plate with screws. Changes were also made to the surface. Because the raspi-plate is going to be attached to the top of the spring, space must be made for the nuts that are under this plate. This was done by making alignments in the plate on the spring. The trackplate consists of the same plate as for the spring but without the spring and with tracks right beneath.

4.4 Switch and splitter stand

To accommodate all the parts, the switch and splitter had to have their own stand to be raised above the other circuit boards. This had to be modeled in SketchUp, and the models made of the switch and splitter were used as a size reference.

The stand for the switch had two versions. The first version consisted of a plate at the bottom to attach to the main plate. In the middle of the plate was a pillar that held up another plate where the switch was attached. The switch was fastened with screws in high-rise cylinders that raised the switch a few millimeters above the plate. In the middle of the plate, a part was also removed to make room for the components on the underside of the switch. When it became known that not all components could be placed on one same side of the plate, the switch was placed on the bottom side. The design of the stand then changed. There was still a plate at the bottom, but it was made larger, and the switch was fastened to four cylinders attached to the plate.

The splitter stand is somewhat similar to the first version of the switch stand. It consists of a plate at the bottom where the screws are screwed through the main plate. In the middle of the plate is a pillar that holds another plate up. The plate at the top consists of a frame and has a hole and a rectangle. The hole is to screw the splitter in place and the rectangle takes into account that the pins on the splitter will pass through on the underside of the circuit board.

4.5 Circle and semicircle

A circle was attached to the rear end of the main plate. This circle was part of the parts that were already designed by a former student. The first version made in this project consisted of the old circle, but with two holes on each outside to attach it to the main plate. In addition to this, there were made two rectangular holes in the middle where part of the main plate was to be placed. Here, vertical holes were also made through to fasten screws. In version two, four holes were added to the circle itself to attach to the lid of the container. This was done by uploading the model of the lid to SketchUp in the same file as the circle. Then the circle was aligned with the lid, to see if the size was correct. After this the circle was moved some distance in front of the lid, and some guidelines from the holes were made. These guidelines then crossed the circle where the holes had to be made. The holes were made by the circle and push/pull tools.

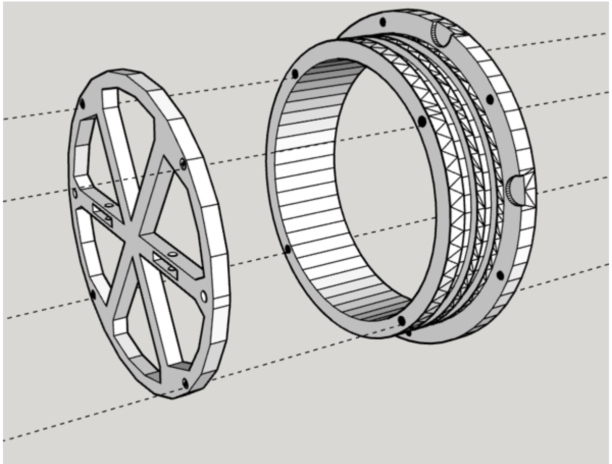


Figure 4.14: How the second version of the circle was made

The holes on each outside were also slightly altered. Space was made so that the head of the screws did not protrude from the model.

A semicircle was to be placed on the other side of the main plate. This semicircle was also designed before, but here quite a few changes were made. Tracks were made at the top of the semicircle so that it could be attached to the the pins that go down from the rail. In addition, small plates were added on the outer sides which were to lie next to the main plate and plates which extended horizontally from the semicircle which was to lie below the main plate. Holes were made in these horizontally plates to fasten screws through the semicircle and the main plate.

4.6 The Printing

When all the parts were modeled in SketchUp, it was time to 3D print them. This was done by first converting the models to a STL file, which is done by clicking export in the file menu in the left corner in the program. Then select Steriolithography Mesh.

This file was then opened in Cura, where settings were selected. All models have 40% infill with support and adhesion activated. The exceptions were the circle and the semicircle. The semicircle had an infill percent of 80, while the circle had an infill percent of 100. Most models have some places where there is more infill than in general. These were places around holes, and thin areas.

Once the settings were set and the model was ready to print, the model was sliced. This was done by pressing the slice button in the lower right corner. Here one would see how long the print would take, as well as how much material would be used. If preview is clicked, one can see exactly the path the printer will take to print the object.

When one have gone over the model one last time and everything looks good, the sliced model can be saved to a USB stick. The USB stick is then inserted into the printer. The printer has a touch screen where one can find the right model and press “Start print”, to start the printer. For a more detailed description of how the settings are set and how the printing is done, see the 3D printing procedure in appendix **8.4 *Procedure 3D printing*** .

There are a total of ten 3D modeled parts in this project, and they have to be printed in two rounds. The first round consists only of the main plate, and the second prints the rest.

4.6.1 After the print

When the print is complete, all supports must be removed. This is done by using pliers and a razor blade. Some places are worse to remove the support plastic than others. This is for example in the rail, or in horizontal holes. Here one can use a flat screwdriver to pick it out. When all the prints were finished, the threading of the holes began. This was done with two different threaded pins. The threaded pins were attached to a handle, to make it easier to thread up by hand. A table of which holes should bee threaded up with different sizes is found in the appendix **8.3 *Table of screws***.

4.7 Every part together

When every part is printed, every support part is removed and the holes are threaded up, there is time to assemble all the parts. The process starts by attaching the switch by screwing the screws from the top of the main plate. When the screws were almost through the main plate, the switch stand was placed over the screws so that the stand attached to the main plate when the screw continued down. Then the switch was fasten with nuts on top of the stand.

Next was to attach the isolated carrier boards to their slots on the top of the main plate. These were the six cylinders on the plate. After isolated carrier boards, the ADC was fastened in the middle between these. Finally, the splitter was attached to the splitter stand, and the splitter stand was attached to the main plate directly above the ADC.

The semicircle was fastened to the front of the main plate by sliding the pins in the tracks made in the semicircle. Screws was then fastened on the underside of the main plate through the plate on the semicircle.

Then the circle was attached to the main plate by placing the tabs on the back of the main plate in the rectangular holes on the circle. Screws were screwed into the holes, both those that go vertically and those that go horizontally. On the front of the main plate, the locks were attached to the two holes located there.

The Raspberry pi was attached to the raspi-plate, which again was attached to either the spring or the trackplate based on whether the heat sink was used or not. The spring or trackplate was then pushed onto the main plate.

See the attached document ***8.4 Procedure 3D printing*** for more details on how the components are attached to the main plate and where they are to be placed.

5 Results

This chapter contains the results from the conducted tests, the rehousing and potting, and the results from the 3D printing.

5.1 Functionality test of the sensors

These are the results from testing the functionality of the sensors. The results are presented as graphs, and the values are gathered from appendix ***8.2 Data from functionality test***.

Temperature

The temperature sensor reacted as expected. In the lukewarm water the sensor registered a maximum at 38°C, and about 22°C in the air.

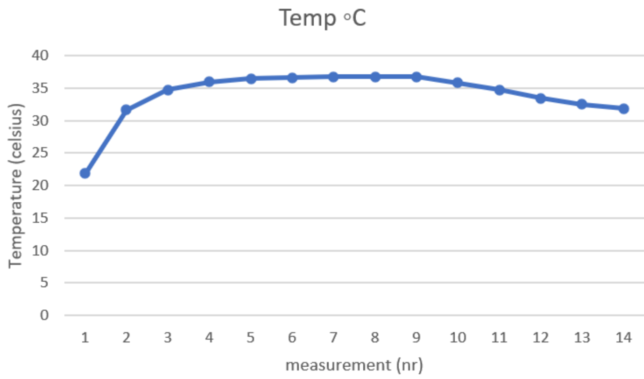


Figure 5.1: Graph of measured temperature

Pressure

The sensor was tested by pressing on the sensor, and measured the pressure in both mBar and Psi simultaneously. The results are shown in figures 5.2 and 5.3.

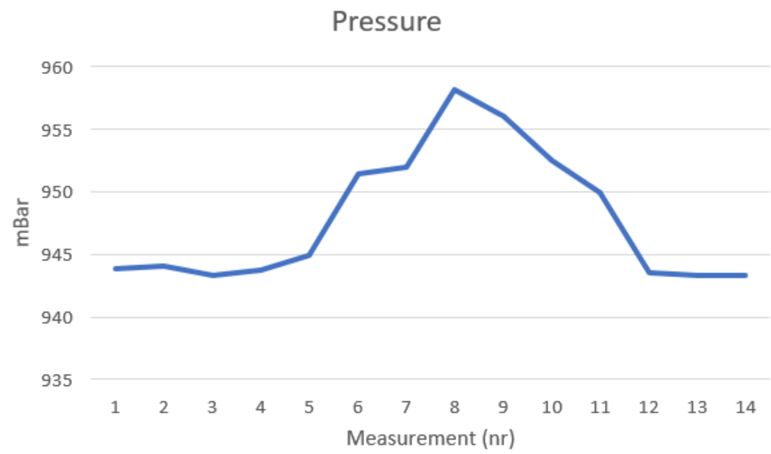


Figure 5.2: Results conductivity measured in mBar

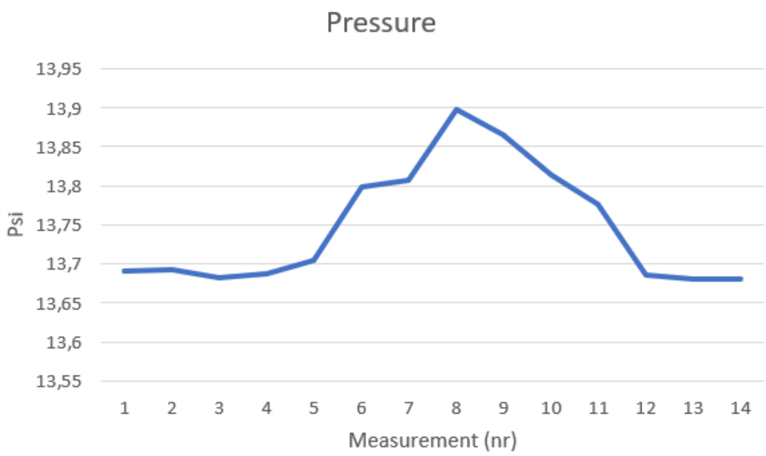


Figure 5.3: Results conductivity measured in Psi

Oxygen

The dissolved oxygen was measured in both cold and warm water, and the results can be seen in 5.4 and 5.5.

Conductivity

Conductivity is the amount of salt in the water, and the results from the test are shown in figure 5.6

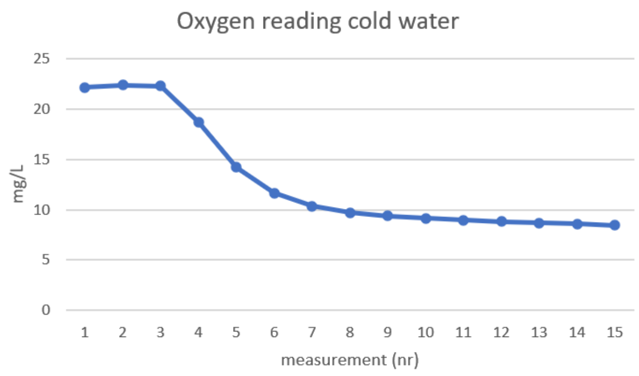


Figure 5.4: Oxygen measured in cold water

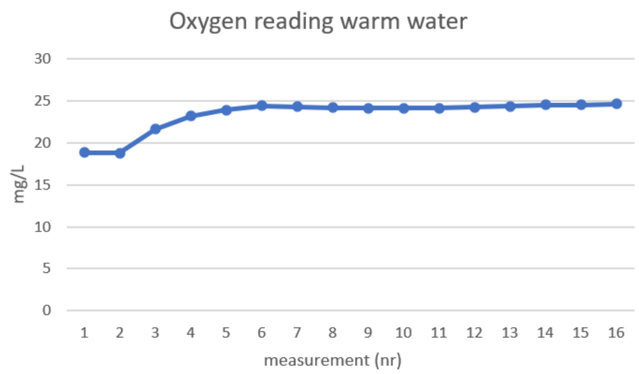


Figure 5.5: Oxygen measured in warm water

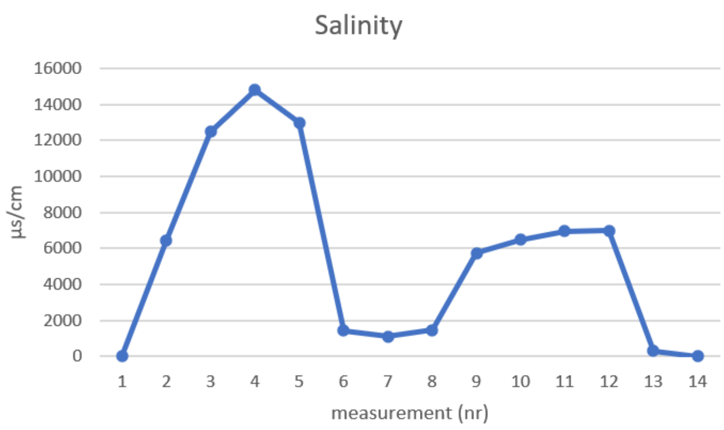


Figure 5.6: Conductivity measured in lukewarm water

5.1.1 Rehousing and potting the Atlas sensors

The sensors were successfully rehoused by the NTNU Workshop, and figure 5.7 show the end result.

Figure 5.7: The result from rehousing and potting on the conductivity sensor

5.2 Testing the waterproofing of the cylinder

The first check of the vacuum pump with the rubber stopper showed no signs of leakage, and the pump was even left for 30 minutes.

For the test with the casing, the vacuum pump was pumped to about 10 inHg, and left to sit. After observing it for less than 30 seconds, the value was down to 8 inHg.

After taking off the lid and checking all of the components in the lid, and greasing around the lid, the test was conducted again. The pump was again pumped up to a vacuum of 10 inHg, and the value dropped to nine after about a minute.

5.3 3D printing

Another part finished was the 3D printing of the hardware layout.

The hardware layout is depending on the 3D models holding the electronics in place. This subsection gives an overview of the final designs of each piece. It will also be given a overview of the screws used, and the placement of each component.

5.3.1 Screw dimensions

Figure 5.8 shows the placement of the screws used in the project. Each hole is numbered and have a corresponding screw, as seen in appendix **8.3 *Table of screws***. The holes located on the spring are also located in the same place on the track plate.

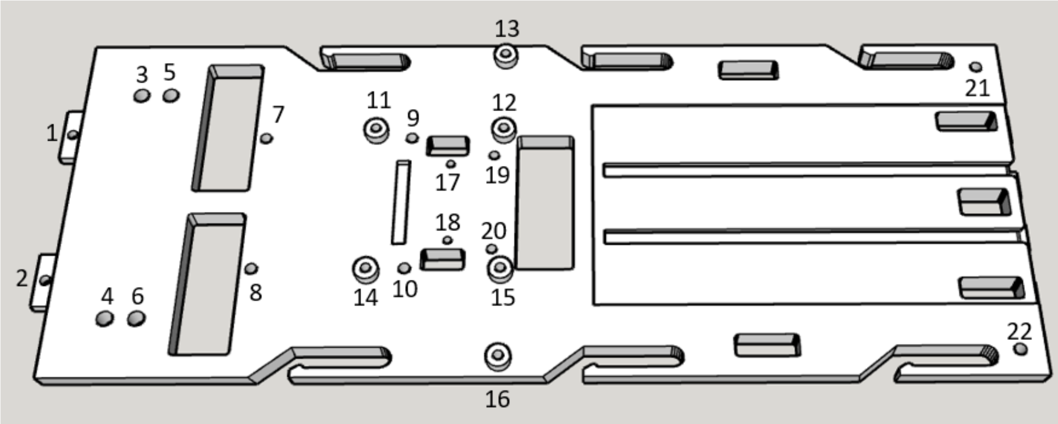


Figure 5.8: Overview of holes in the main plate

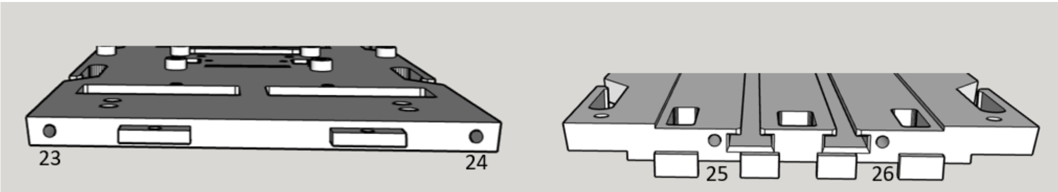


Figure 5.9: Overview of holes on each end of the main plate

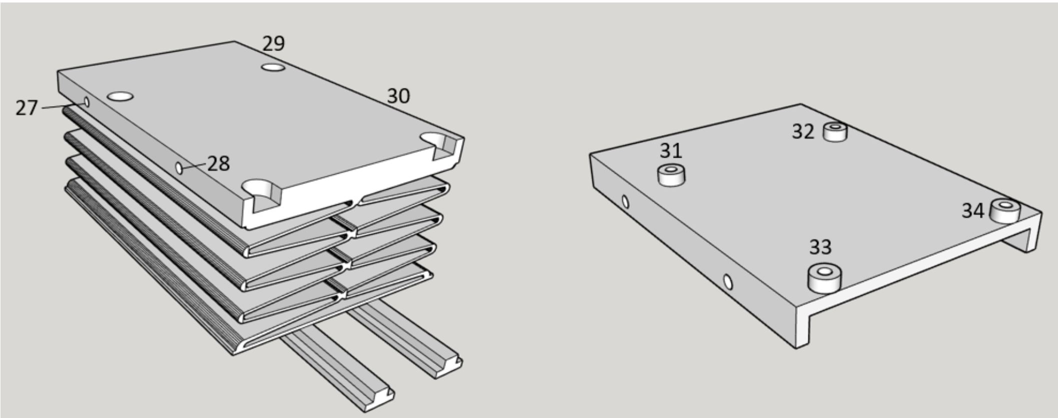


Figure 5.10: Overview of the holes on the spring/trackplate, and raspi-plate

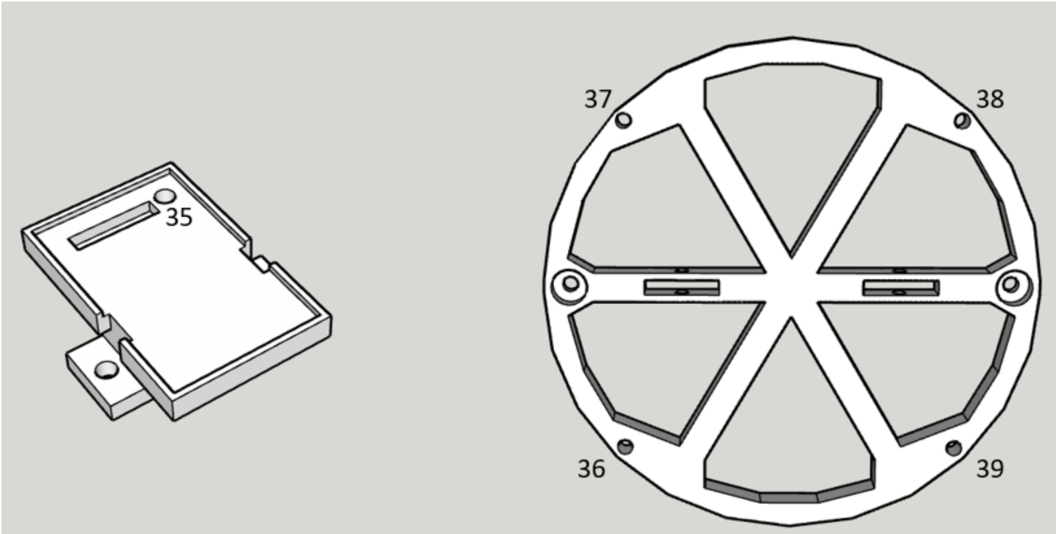


Figure 5.11: Overview of the holes in the splitter stand and circle

The table describes the various holes in different parts of the system. The numbered holes are listed in the left column. Next to this is the column describing the dimensions of the screw, and next to this again is the length of the screw in millimeters. The "Threaded" column shows whether the holes should be threaded up or not. In the last column there is an overview of which two parts are screwed together.

5.3.2 Main Plate

The main plate is 256,2 mm long and 98,6 mm wide and has a total of 26 screw holes that holds all the parts in place. Figure 5.12 shows the model in SketchUp on the left and the actual print on the right. The picture was taken when the main plate was still in the printer, and has all of the support attached.

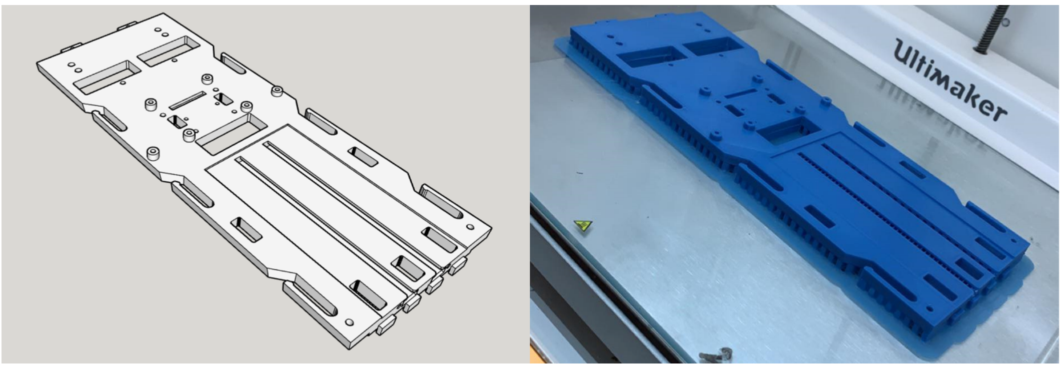


Figure 5.12: Result of main plate

5.3.3 Placement

The two barrier strips are placed on opposite side at the back of of the main plate. They are fastened to the main plate with four screws that goes through both barriers strips and are secured with nuts. The two isolated carrier boards are located a little further up on the plate on each side, and are raised up from the plate with hollow cylinders. Between the isolated carrier boards are the ADC and the splitter located. The ADC are placed directly on the plate, and the splitter stands above the ADC on a stand that is also attached directly to the plate. The switch is located on the underside of the plate, right next to the two large holes. The holes for the switch go completely through the plate and the screws are fastened with nuts.

5.3.4 Raspi-plate

The raspi-plate has a length of 87.8 mm and a width of 66.4 mm. It has a total of 8 holes to both attach the Raspberry Pi, and attach the plate to the other spring- or track plate. figure 5.13 below shows the model and the print.

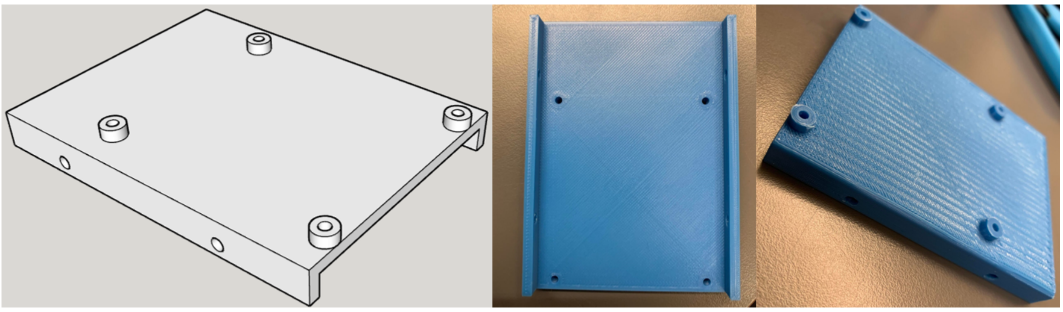


Figure 5.13: Result of the raspi-plate

5.3.5 Spring

The spring is as long as the raspi-plate, but with a width of 59.8mm. The tracks are 108 mm, just as long as the rails on the main plate. There are four holes on the spring plate that allows attachment to the Raspberry Pi plate. There are also four carvings for the nuts, that keeps the two plates flush.

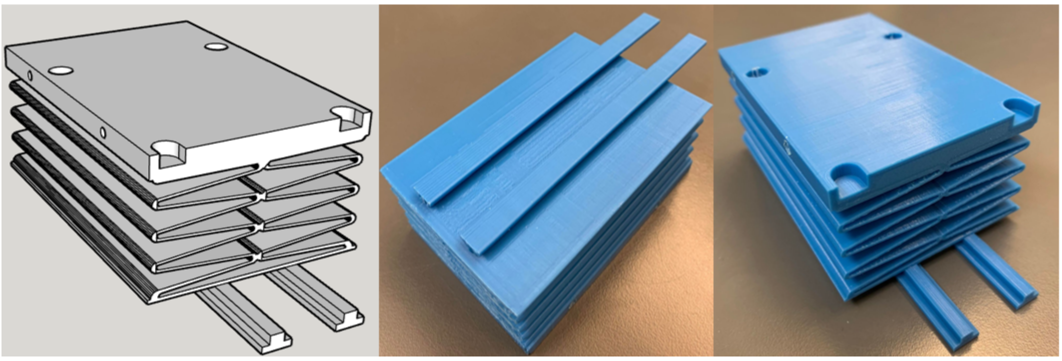


Figure 5.14: Result of the spring

5.3.6 Trackplate

The track plate is exactly the same plate and tracks as the spring, but without the spring.

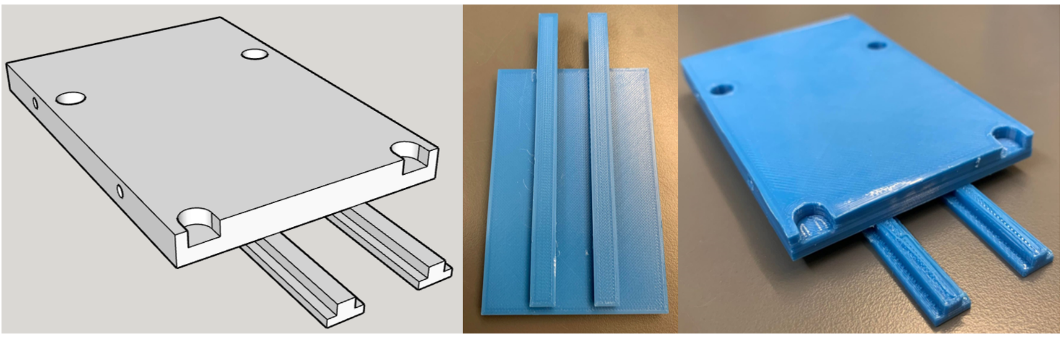


Figure 5.15: Result of the trackplate

5.3.7 Switch stand

The switch stand has the same length and width as the switch itself, 45 mm both ways. The columns have a height of 15 mm above the main plate, and there are four screw holding the switch in place.

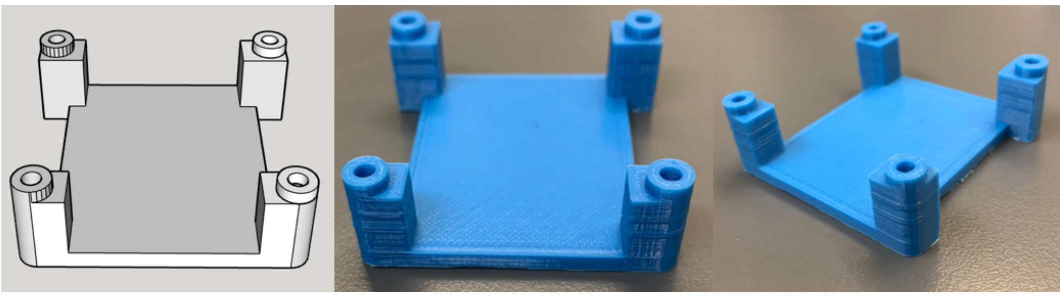


Figure 5.16: Result of the switch stand

5.3.8 Splitter stand

The splitter stand consists of a raised frame. It is 7 mm above the main plate, and thus 6 mm above the ADC. There are three screw holes on the stand in total, two down in the main plate and one that holds the splitter firmly to the stand. The result is shown below in figure 5.17.

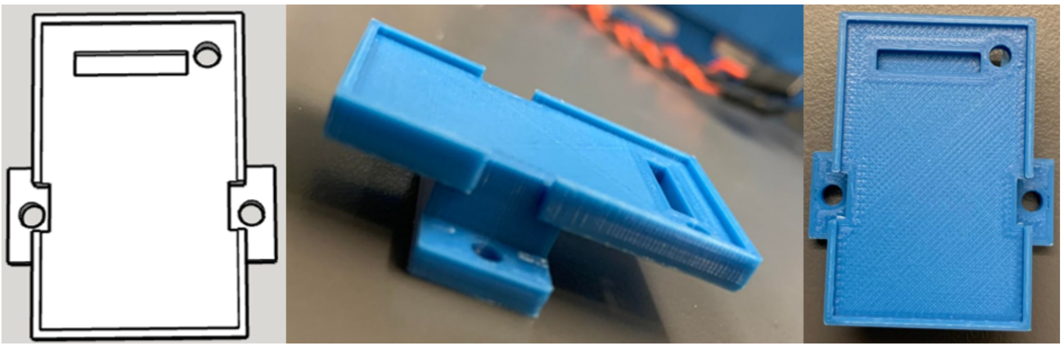


Figure 5.17: Result of the splitter stand

5.3.9 Semicircle

The semicircle consists of a circle with columns, as well as tracks for attachment to the main plate. It also has two screw holes to hold it to the main plate. See figure 5.18.

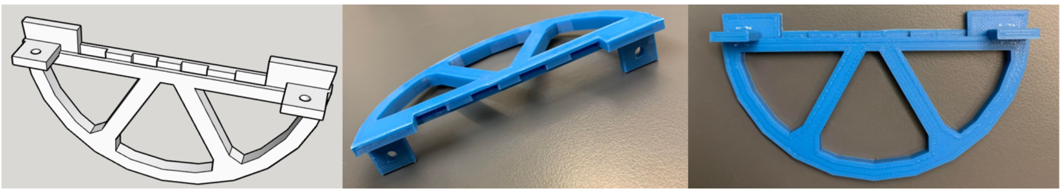


Figure 5.18: Result of the semicircle

5.3.10 Circle

The last version of the circle was not printed, due to the printer being busy. Therefore figure 5.19 only shows the finished model in SketchUp. The circle is slightly smaller than 100 mm in diameter, and has a total of 8 holes.

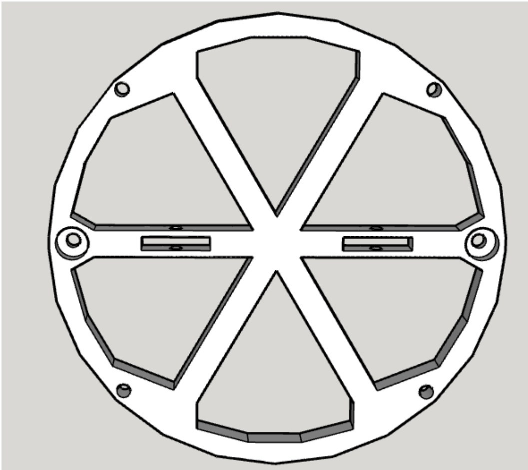


Figure 5.19: The model of the circle in SketchUp

5.4 The whole system together

Figure 5.20 show the complete system both as a model and as a finished product, with all parts assembled. The fit inside of the container was also verified by doing a physical test.

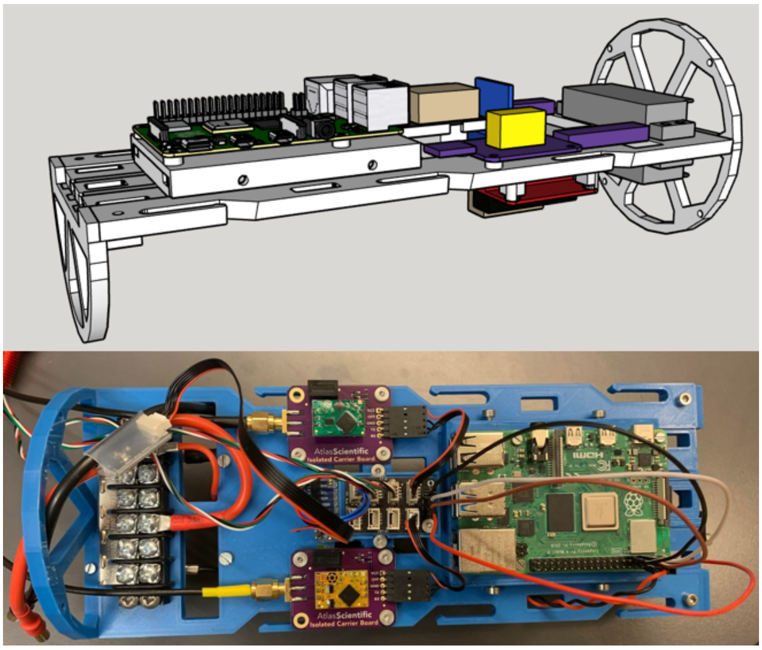


Figure 5.20: The result of the whole system together

6 Discussion

This chapter discusses the preliminary report, the choices made for developing the sensor and battery code, the 3D models, the results from testing, and the final assembly of the rig. Each main part of the project, the sensors, the battery, and 3D modeling also contains a subsection about troubleshooting done to fix problems that arose. The group has decided to include it as it has valuable information for those who will continue working on the project, and also prevent them from having the same problems we did.

6.1 The preliminary report

This section about the preliminary report is important to include, because that report served as the basis for the entire project.

A time management plan, a so called Gantt-diagram, was made in the preliminary report as a guide for the rate of progress during the semester. The group tried to follow the Gantt-diagram as it kept track of milestones and remaining tasks, but some deviations and changes were made.

The project was also split into work packages with corresponding milestones. The work packages and the Gantt-diagram were used together, as the work packages had more detailed milestones, and the Gantt-diagram showed how much time was left on each task.

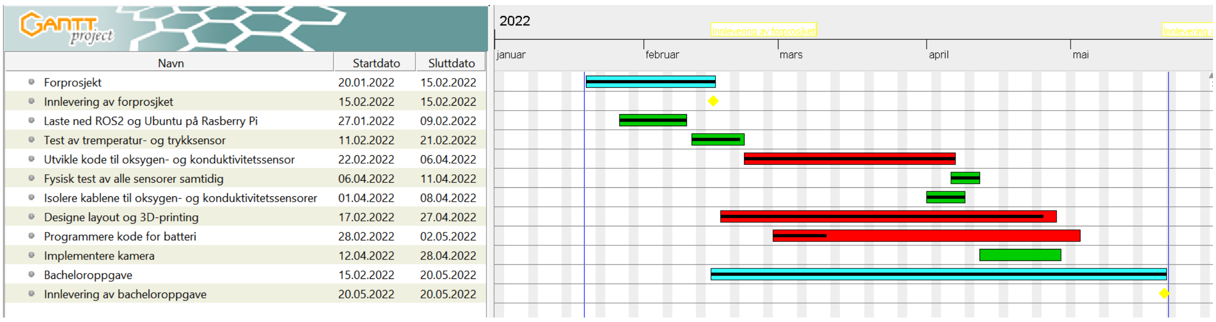


Figure 6.1: Gantt diagram

6.2 Electronics

This section is about all of the choices made in regards to the electronics. This includes discussions about troubleshooting and results.

6.2.1 Raspberry pi Image flashing

At the beginning of the project when the group was learning Ubuntu and ROS, an issue occurred with the image. The image with previous work wouldn't flash onto the SD card. This was because the image was sent over Microsoft Teams, and something went wrong in the upload. So the downloaded image was corrupted.

***Solution* 1**

Because the image we were supposed to use had issues, it was decided to use one Raspberry Pi and SD card that had downloaded Ubuntu from scratch. This worked until it was realized that the Ubuntu OS had the wrong version, which wouldn't work with ROS2.

The version needed was only a Server version, and the group decided to add desktop on top of the server version. It works, but the screen is a little slow and freezes sometimes.

***Solution* 2**

The other Raspberry Pi got the correct image flashed directly from one of the supervisors computers. This ensured that the image was working, and no error could occur in the transmission from the PC to the SD card. The SD card with the correct image was the one primarily used throughout the semester.

6.2.2 Sensors

Because there had been people already working on this project, the electrical components were already chosen, and that included the four sensors. The BlueRobotics sensors were factory calibrated, and did not need any extra calibration.

The Atlas Scientific sensors on the other hand needed to be calibrated by hand as described in chapter **3.3 *Calibration of the Atlas Scientific sensors***. These calibrations required the sensors to sit in solutions with known values. Because we had two solutions with known values for the conductivity sensor, we opted with the two point calibration. In doing this the odds of getting more precise readings increased as we are covering a wider spectrum of possible values. Though the known values for the calibration solutions were only at 12,880- and 80,000μS/cm, the probe had an accurate reading from 5-200,000μS/cm. Seeing how big the difference is between the highest calibration solution and maximum accurate reading from the probe, we are not sure how accurate the readings will get with higher values.

Although there were only one pack of solution for dissolved oxygen, the sensor still allows for a two point calibration. Since one of the "solutions" used for this was air, there is some uncertainty, and an estimate is used for this value.

6.2.3 ADC

An issue occurred with the ADC used for monitoring the battery life. This happened after soldering the pins in place, and it was most likely because of a short somewhere on the board.

It was attempted to clean the board with isopropanol in case any impurities from the soldering were affecting the board. This didn't change anything, and we got a new ADC to replace the shorted.

Although the ADC was now visible in the terminal in the Raspberry Pi, the values were not as expected. This led to an additional problem regarding the ADC and PSM. After testing with a voltmeter over the pins on the ADC, it was discovered that the voltage going from the PSM to the ADC was not the expected 3.3V, but about 12V. After testing the ADC with a voltage divider it became clear that the analog pins A0-A2 did not work as they did not measure any changes in voltage values.

The analog pins A0-A2 didn't work anymore, but fortunately A3 did. The A3 pin ended up measuring the expected values from this voltage divider test.

The ADS1115 datasheet[57] states that the absolute maximum ratings for the analog pins is -0,3 to V_{DD} + 0,3. Because the device is powered by a maximum voltage of 3,3 from the raspberry, it makes sense that some of the pins do not work due to over voltage.

This problem was discovered late into the project, and the battery code would not be finished within the due date. Although the group will not have it ready, a description of how we wanted to solve the task is written in **6.2.5 *Battery***.

6.2.4 I2C Bus Splitter

When it was time to test all of the sensors together, and start finalizing the code for the battery, the I2C splitter broke. Two of the DF13 connectors came off because of too much force used when trying to remove the headers from the connectors, and thus rendered the board useless. When testing the board with the remaining connectors, it still wouldn't work.

During testing to see if the Raspberry Pi could connect with the board at all, the jumper cable also melted. This was solved by finding a new female-female cable, and no cables attached to the I2C board has melted since.

***Solution* 1**

The first attempt to fix the I2C was by soldering on the two connectors. It didn't work because there was to much solder on the pins on one of the connectors, which created a bridge on the pins and shorted the connector. The other connector had an incomplete soldering, because of the small space and a too large tip on the soldering iron.

***Solution* 2**

The second attempt at fixing the I2C was by re-soldering the two connectors. The pins were removed by using a soldering iron and copper to pull the solder off the pins. After the connectors were removed, all excess solder was removed, and they were soldered on again. This time the soldering iron had a pointier tip which helped with precision. Soldering flux was also used to help ease the soldering.

After the connectors were soldered on they were tested, and were working again. However the connectors have a weaker bond to the PCB, and it is not recommended to switch around on the cables, as that pulls on the solder. If the cables need to be removed from the connectors, use a flat screwdriver on the back of the header to pry the header out of the connector.

6.2.5 Battery

The third main part of the project was developing code for battery health monitoring. The group wanted to be able to read the voltage and current ratings in real time, and develop an approximate timer for the battery power, if we had enough time.

The plan was to make an approximation of how the battery discharges, by using the ADC to log data. The battery would be charged to max capacity, and left on until the battery was reaching minimum charge levels. These data points would be made into a graph, which we could use to develop an approximation of how much time there was left of battery life.

6.2.6 Power Sense Module

However there were problems with the BluRobotics Power Sense Module. The PSM was supposed to have an output voltage of +3.3V, but by measuring the voltage on the analog pins of the ADC, it measured 12V. The ADC's max voltage rating was 5.5V, and the high voltage most likely fried the pins on the broken ADC.

BlueRobotics costumer service was contacted to help solve the issue, however the time difference resulted in us not getting a response fast enough to fix the problem before handing in the report. The issue with the PSM could be from our end, because of wrong wiring or wrong interpretation of the technical information. Or it could be from their end as fault on the hardware or misleading technical details.

Their first response was to double check the voltage, which the group did. The group responded with a confirmation that we had double checked, and also added our wiring in case that impacted the results. At the final week of the project, we had not gotten any other response from costumer service. Because we were depended on help we decided to stop working on the battery and rather leave it unfinished, instead of taking the risk in case it affects the other electronics connected with the PSM and ADC.

6.3 3D modeling and printing

There were many things to consider when designing the layout of the electronics inside the casing. This section about 3D modeling and printing discusses the choices made to achieve the wanted results. The choices were more comprehensive than the group first realized, and were about everything from the screws and nuts, modeling in SketchUp, settings in Cura, and the placement of all of the parts on the main plate. Therefore we have decided to add a lot of information about the 3D modeling and printing in this report.

6.3.1 CAD program selection

A CAD program had to be selected before the project started. Since there were no experience of 3D modeling in the group, the choice was based on recommendations on the internet and from former students. As mentioned earlier, some of the 3D models was already designed by the former student. These models were designed in the program SketchUp[58]. With that in mind, along with several articles recommending SketchUp for beginners, it was this program that was being pursued.

6.3.2 Model of all components

As mentioned in the method, each circuit board was modeled into SketchUp using sketches and datasheets found online. This was done to make it easier to check out different dimensions of the different parts, and easier to check if the modeled and physical holes would match up.

Each circuit board consists of a plate with different components on it. When the models were made, only the most important parts, and parts that could get in the way were modeled. This was to easily see if the circuit boards could be placed where they were intended without being in the way of anything else, or too tall for inside of the container.

In addition to important components, parts of cables and pins were also modeled. The cables on, among other things, the isolated carrier boards were rigid and therefore could not be bent as much. This had to be taken into account when placing the circuit broads on the main plate. On the other side of these circuit boards were the pins that connect with the I2C board. These went horizontally out of the circuit board and would use some space behind, and were therefore also included in the model. On the other circuit boards such as ADC and splitter, many cables were to be attached from the top of the circuit boards. This was also modeled by manually measuring how high the cables would go, and this was to see if the height would be a potential issue.

6.3.3 Screws and threaded up holes

Since there was a desire to use as many identical screws as possible, the placement and dimensions of the screws had to be well planned. To find this, notes on the diameters were taken for each circuit board. Most of them had a diameter of 3 mm and could use M3 screws, except the ADC and Raspberry Pi which had a diameter of 2.7 mm, so M2.5 screws had to be used. In addition, M4 screws had to be used on the barrier strips.

After the first print was made, it was discovered that the print makes the screw holes smaller than in SketchUp. To solve this another print was made, and the dimensions of the holes were changed by adding 0.2mm to the diameter of the holes. The M3 screws have a corresponding 3.2mm in diameter hole, and the M2.5 have 2.7mm. This applied to holes that would not be threaded.

The holes that were threaded up needed holes smaller than the screw to be able to make threads. That means that the holes in SketchUp were made smaller on purpose. Holes that would use M3 screws got a diameter of 2,6mm in SketchUp, while M2,5 screws got 2,15 mm diameters. These values were taken from figure 6.2 that have an overview of how small the holes have to be for different threads [59].

Gjengetapper M		Ø –mm
M 1	0,25	0,75
M 1,1	0,25	0,85
M 1,2	0,25	0,95
M 1,4	0,3	1,1
M 1,6	0,35	1,25
M (1,7)	0,35	1,3
M 1,8	0,35	1,45
M 2	0,4	1,6
M 2,2	0,45	1,75
M (2,3)	0,4	1,9
M 2,5	0,45	2,05
M (2,6)	0,45	2,1
M 3	0,5	2,5

Figure 6.2: Table of different threading pins

It was decided that the ADC should have screws of M2.5 with a length of 6 mm. This is because the ADC is located directly on the main board, which is a total of 6 mm, and the ADC is 1 mm. The screw is 6 mm so it doesn't poke through the main plate, but also gets as much fastening as possible.

For the isolated carrier broads and the splitter stand, it was decided to use an M3 screw of 8 mm. The same argument for deciding the screws for the ADC is applied here. To prevent the screw from going through the main board, it had to be smaller than 9 mm, and 8 mm was chosen for the splitter board. The 8mm M3 screw was also chosen for the isolated carrier board, to have as many identical screws as possible, although a longer screw could be used. The switch was secured by 28mm M3 screws.

For the barrier strips, the M4 screws were used. These are the only M4 screws in the system, so the length did not have to match other screws. The screws are 24 mm long.

To attach the semicircle to the main plate, screws of M3 and a length of 16 mm were used, as it was a decent length when nuts were used. This screw length suited several places, so the lock and the spring/track plate also got these screws.

The horizontal screws on the circle was also of type M3 with a length of 16 mm, to use as many identical screws as possible. The vertical screws holding the circle were M2.5 of 10 mm. Screws of 2.5 mm in diameter had to be used because the pins were not large enough for 3 mm screws. The length of 10 mm was chosen, because this length was used on the raspi-plate as well.

There should also be screws on the circle that attach the circle to the lid. The lid had four holes for screws, and the type were M3. The length was 5 mm, and because the circle is 5 mm, the screws had to be at least 10 mm.

6.3.4 Main plate

It was decided to use the previously made main plate as inspiration, because it was well designed, and had the correct dimensions of width and length, however a few changes were made to better fit the electronics on the plate. As the casing is cylindrical, and the rig will move around in bodies of water, it was important to fasten the electronics so they don't float around inside the casing, risking damage.

The first version of the plate were 3 mm thick, which the group determined was too thin, as it weakened the integrity of the plate and risked breaking. A thickness of 11 mm was considered, but we decided against it as the plate needed to occupy as little of the space as possible. In the end a thickness of 6 mm was chosen.

6.3.5 Placement

As mentioned in the method, it was important to plan where each of the individual components would fit on the main plate. The placement was decided by holding the physical components on the main board, and checking that all of the cables had enough reach. All but one component was also placed on the topside of the main plate to save space on the bottom for future developments.

The barrier strips were placed at the back on opposite sides of the main plate to take up as little space as possible. In front of them again, two rectangular holes are placed for threading cables between the topside and the bottom. The barrier strips were placed opposite each other to save space on the cable, and keep the exposed metal on the barrier strip away from the other electronics as a precaution.

The switch was placed on a stand on the opposite side of the main plate to reduce the number of cables on the same side. With too many cables, it can quickly become cluttered.

The two isolated carrier boards were originally supposed to sit closer to the barrier strip, however they were moved further back so the metal on the cables does not touch the barrier strip. This was also done as a precaution. Additionally, the cables on the isolated carrier boards were hard to bend, and had to be placed with a small clearing to other hard parts. There was a concern that the pins would get in the way of the raspi-plate, but the circuit boards were placed far out on the edge of the main plate, so there was no problem. These decisions were made by looking at the model of the circuit broads in SketchUp, and it shows how important they were for the placement of the different parts on the main plate.

The ADC is located between the two isolated carrier boards. On the main plate, a rectangle was made to make space for the pins on the underside of the circuit board.

The splitter had to be within range of almost all of the circuit boards. This is because most circuit boards in the system have cables that go to the I2C splitter. The fact that most cables go through the splitter means that it has to be placed with access to the all of the cables as desired. For this reason it was placed right between the isolated carrier boards, and over the ADC. The elevation also saved space by using the airspace above the main plate.

6.3.6 The remaining parts

On the main plate there is a rail embedded that a spring or trackplate, with a mounted Raspberry Pi, can slide on. The spring consisted of the track at the bottom that interlocks with the main plate. The spring is flexible and sturdy, and holds the Raspberry Pi. The spring will only be used if the heat sink is designed. The heat sink will press the spring together so it fits in the casing. The trackplate is the same as the spring, only without the flexible spring. This plate was made to be used if the heat sink was not ready for use.

To mount the Raspberry Pi to the different models, a separate plate holding only the Raspberry Pi was designed. The plate has four raised cylinders with holes that keeps the Raspberry Pi from touching the plastic directly. The Raspberry Pi is also held in place with screws and nuts.

Both the spring and trackplate have carved out holes for the nuts on the raspi-plate, to keep the two plastic plates flush against each other.

Separate stands for the Switch and splitter had to be made to make all of the components fit on the main plate.

The switch had bulky components on both sides of the circuit board, which had to be accounted for when designing the placing of the component. It was decided to make a four legged stand that holds the plate securely in the corners. The legs were designed to be slightly longer than the biggest component to allow more air circulation, because some of the components can get hot. The switch was also moved to the underside of the plate, away from the rest of the electrical components.

The I2C splitter circuit only had one hole in one of the corners, which made securing it to the main plate more challenging. A frame was made for the splitter that it fits nicely in. This stops the splitter from moving around, while also being secure. The frame is attached to a stand that towers the splitter above the ADC, making it more space efficient.

The circle design was kept as close to the original as possible, however changes to the locking mechanism and some adaptations had to be made so it fit on the new main plate. The circle is stabilized using rods, which also allows the cables and sensors to easily pass through, as the full circle is the one closest to the lid of the container. In the middle of the circle there is a band across the diameter that interlocks with the main plate. The locking mechanism is also reinforced with two screws. As an extra security measure there are two extra screws on the ends of the diameter that screws the circle to the plate.

When the main plate would be placed and pulled out of the casing in the middle of the semester, the circle would be used as a handle to pull on. Therefore it was important to secure it tightly and make it sturdy.

Towards the end of the project it was requested to make the full circle able to attach to the lid of the casing. This meant making holes along the circumference that matched up with the holes on the lid. In theory this makes more sense for the user, as it makes it so the user only has to open the lid and pull it out to extract the main plate. This is for easier access to change setup or components on the main plate. However the circle was pretty thin and by adding holes it could weaken the circle. Therefore the circle was modeled again, and the thickness of the circle was increased. The extra material was added by drawing a line from the columns and a bit further out on the circle. When the circle was attached to the lid, the screws holding the circle on the main plate was in the way. This was solved by adding space for the screw heads so the circle lays flush with the lid.

The semicircle ended up with a pretty new design. Tracks were added to the middle of the diameter of the semicircle to match the ones on the main plate. There was also added shelves on both sides that holds the main plate in place. With this interlocking system, only the two screws in the two holes are necessary.

When the semicircles was designed, we had to consider that the print often makes the print slightly larger than the model. Between 0.5mm and 1mm was therefore removed so the interlocking system would work as intended. This worked well on the printed pieces.

6.3.7 Printer settings

When it came to printing the parts, several different choices were made, and they all influenced the final result in varying degrees. The many different settings the group wanted were made in the program Cura, which is the program that prepares the print.

It was decided that the infill percent would in general be 40%, if no other choices were made. Cura originally recommended 20%, however the group chose to go with 40, to be sure that the model was strong enough, as the prints would be subjected to high pressure, and a lot of wear.

"Support" is a setting that adds excess print material to support the printing of the model, when the model has overhang. The support was turned on for all of the prints. The setting "Build plate adhesion" was turned on during printing, because it helps the plastic stay in place during the printing. Without this setting the prints risk loosening and falling over during printing.

The main plate was printed with the upside up. It was also considered to print it upside down to see if it would save time and material, but it was the other way that came out best. This is partly because there is less overhang with the upside up, and thus less support to remove. This print has the general settings as described above, but has several different places where the infill percentage varies. This is at the ends of the two long sides as there are thin parts and holes that need reinforcement in the form of infill. This means that the infill percentage was set to 100. This also applies to the very back and front of the model, because of holes and thin parts that need reinforcement. But also a part in the middle of the main plate had to have a high infill percent, because there were many holes that were threaded, and the plastic must be completely drawn to make threads.

The circle has an infill percentage of 100 as it must withstand a lot of force, considering it will be used to pull the whole system out of the casing. The semicircle has an infill percentage of 80. It has a higher infill than in general since it has to withstand quite a lot, but not 100%. This is because it does not need to withstand as much as the circle, but it still needs to be strong. At the same time, it is unnecessary to use an infill percentage of 100 when it is not needed, as more materials are being used and in turn takes longer time to print.

The splitter stand uses the general settings. This is because there is no need for extra strength as it is quite small and no holes are threaded up. The same goes for the locks. For the switch stand, on the other hand, the pillars have 100% infill. This is because they are quite long and can risk breakage with too little infill.

As for the raspi-plate and the trackplate, both are printed upside down. This way of printing takes less time, while it at the same time creates an easier way to remove the support structure. There are some places on the boards that have a different infill percentage than in general, like 100%. On the raspi-plate, this is on both sides as there are thin walls that can break. The trackplate also have another infill percent on the edges and the tracks. The tracks can easily break and should therefore have a lot of infill. The edges also need a lot of infill when there are holes with threading placed.

The spring was printed with the tracks pointing upwards. This is to make the print as simple as possible, without having to make any support. The infill percentage was the same as for the trackplate, with 100 on the ends and tracks. Printing this took almost as long time as the main plate. This is because it is detailed and quite tall.

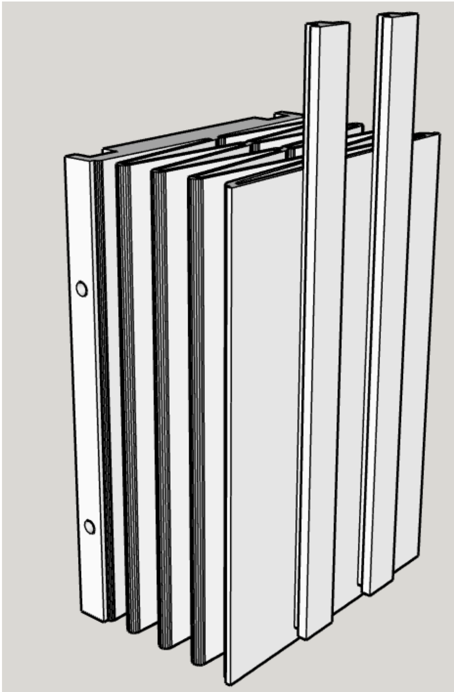


Figure 6.3: The direction the splitter is printed

6.3.8 Small changes made along the way

When using a 3D printer, not everything goes according to plan all the time. Many changes were made along the way when it was physically tested with other parts and elements. This is usually because one overlook things in SketchUp, forget to double check, or do not take into account that the print will be a little bigger than you think due to the printing method.

The size of the half circle is one of the things that was changed along the way. When the first version was printed and mounted on the main plate, a physical test was conducted. The main plate with the half circle were put in the casing to check the fit. It was discovered that the size of the half circle was a little short, so there was too much wiggle room between the edge and the parts. Therefore, the half circle had to be printed again.

After the first print of the splitter stand, it turned out that one could not screw down screws in the main plate because the frame was too wide. Thus, an attempt was made to reprint it with parts of the frame removed exactly where the screws were to pass. The new print worked well.

It was also noticed that the tracks on the spring and trackplate were too short, so the spring slipped back and forth if you tilted the main plate. This had to be fixed, because it could cause cables to be pulled loose. Therefore, the tracks were made longer on both sides of the plate so they fit perfectly in the rails.

Another thing that changed along the way was the height of three cylinders and the depth of two other holes on the main plate. When all the circuit boards and parts were tried out on the main plate, it turned out that one of the isolated carrier boards had pins on the underside of the circuit board that came into contact with a metal screw. Attempts of cutting off some parts of the pins were made. This helped a little, but there was still the possibility of contact. Therefore, the cylinders with holes were raised by 1 mm where this circuit board were placed. This means that the next time the system is printed, this will not be a problem. But the version that was printed during the project also had to work. A solution to these problem was to place kapton tape over the screw in question. Kapton tape is a type of electrical tape that can act as an insulator.

6.3.9 Problems with the lock

After the main plate was printed and most of the support plastic was removed, all the holes were threaded up. When it came to threading the holes that hold the screws for the locks at the front of the main plate, things went wrong. In one hole there was still too much support plastic, and when an attempt was made to thread it up, this plastic was pushed too far inwards. This led to the hole becoming too large when trying to rethread the hole. This mean that the M3 screw was too small for the hole.

***Solution* 1**

One solution was to opt for just one lock. It seemed quite stable and worked well on its own.

***Solution* 2**

The second solution was to print a new piece that was twice as long and could cover both outputs from the tracks. The only problem with this was that it had to be unscrewed every time you had to push on or off the spring or the trackplate. But this solution is more secure than the first solution.

6.4 Results from testing, and hardware solutions

The results from the conducted tests show that the sensing rig mostly works as intended, however some of the tests could have been more precise or redone. This section will cover these insecurities in the tests, and also touch on possible future developments.

6.4.1 The functionality test of the sensors

The results from the functionality test indicate how well the code and sensors works. The test shows that each sensor works in the sense that they read values, but not how accurate. The data was logged every two seconds, which isn't accurate enough to see how detailed and accurate each sensor is. With the temperature sensor, the degrees changes very drastically, so this uncertainty might have been avoided if the measurements had shorter time intervals.

The two second interval also affected the accuracy of the other sensors.

Temperature

Although the test done on the temperature sensor was a pretty simple one, it was enough to confirm that the sensor works as intended. As it's been written earlier the sensor was simply exposed to the air for only a couple of seconds before being dipped in the lukewarm water. As seen from the graph in results, it measures a temperature of 22°C when in air, and about 37°C when submerged.

Pressure

Pressure is another measurement that is hard to properly measure without good equipment. Seeing as the group only had a small container it would make no difference whether we submerged the sensor or not. Instead, one of the members proceeded to grip around it and apply pressure that way. What we ended up with was a small change in value, but to what one would expect from somebody just applying mechanical pressure.

Dissolved oxygen

As we do not have any proper equipment to check the exact measurements of dissolved oxygen, we decided to check the difference between dissolved oxygen in warm and cold water. First we filled the container with warm water and let it sit. To get accurate measurements the sensor laid still in the container. Once the measurements stabilized, the warm water was swapped with cold water.

Prior to performing the experiment, the group had researched how temperature affects the dissolved oxygen values. Our research concluded that cold water holds greater levels of dissolved oxygen than warm water[60]. This can be attributed to the fact that molecules move faster in warm water than in cold, allowing the oxygen to escape faster from the warmer water. With this in mind, the group expected a higher value of dissolved oxygen in cold water, but as seen in **5.1. *Functionality test of the sensors*** this was not the case.

Instead of the expected result, the group ended up with the exact opposite. We suspected that the results were because of faulty code, which prompted a check of the library code used for the dissolved oxygen sensor. After going through it, we found nothing in particular that could have caused these results. After that, we ran the sample code for the Atlas Scientific sensors that could be found on the website [61]. When running the sample code we ended up with the same values as we did with our own code.

This result being both good and bad, as we know that there isn't anything "wrong" with our code, but we don't know what caused the readings to be so different from our previous research.

Conductivity

The conductivity sensor is the sensor with the most varying graph. The sensor was first placed in a tub of water, and then the salt was added. The sensor was also used as a "wand" to swirl the salt around, and this caused the high concentration of salt in the beginning. Then it was moved to the corner with the lower concentration of salt, and back. At the end the sensor was lifted up of the tub, and clean water was poured on the sensor for it to read null, and to clean the sensor.

If the reader doesn't know the sensor was moved around in the tub, these results might look less accurate. Perhaps another test should have been conducted to relieve some of the confusion, with the sensor laying still, and something else used to swirl the salt.

6.4.2 Rehousing and potting of the Atlas Scientific sensors

The two Atlas Scientific sensors had to be rehoused so we could add the waterproofing enclosure around the cable. The rehousing was done with help from the NTNU workshop. The group followed the instructions from the BlueRobics website [62], where it was instructed to remove the insulation where the enclosure would be placed. Therefore we asked the workshop to also remove this piece.

This was something we should not have done, as the supervisor didn't want epoxy on the exposed coax cable. This was fixed by simply potting the enclosure above the exposed wire, and taping the exposed wire with kapton tape.

The potting itself was also done with the help from the supervisor. The potting of the sensors were indirectly tested in the vacuum test on the container.

6.4.3 Vacuum test on the container

At the end of the project, the casing was vacuum tested. The cables, sensors and bolts were added to the lid, and the lid was pushed onto the casing, without any other electronics inside. This test was to check if the casing was airtight, and ready for a field test.

The result show that the vacuum drops too fast, so there is a leak somewhere. It was made an attempt to fasten the components on the lid tighter, and this slowed the leak a little bit, but not enough for the test to be satisfactory.

The air could be leaking from one of the ten components on the lid, including the two potted Atlas Scientific sensors, or from the gap between the lid and cylinder.

This means that the waterproofing of the case is not complete by the end of the project, and is something that will be worked on after this report is handed in.

6.5 Future developments

The rig is designed for taking future developments into account. The electronics in this project is placed on one side of the main plate, with the exception of one component, to keep space available for future technologies.

There was a suggestion from the employer to implement a camera to the sensing rig, so this is a possible add-on for future developers.

6.5.1 Battery health monitoring

The development of battery health monitoring was one of the main tasks in this project, however this was underdeveloped and not finalized. The missing pieces to finalize the battery, is to find out how the PSM is supposed to work/be connected to the rest of the system so the output on the PSM is a maximum voltage of 3.3V.

In chapter **6.2.5 *Battery*** we discussed how we wanted to implement the battery monitoring, and this method could be used by the next persons working on the sensing rig.

6.5.2 Dissolved oxygen sensor

The readings from the dissolved oxygen sensor didn't correspond with the expected results, and we couldn't find out why, so this could be a task for future development.

6.6 What would we do differently?

If we were to do the project again, we would start with the development of the battery monitoring much sooner. We started after we finished developing the codes for the sensors, but we should maybe have started in parallel.

We should have also added logging to the sensor codes, so we have all of the data saved in better formats than a screenshot. This would have reduced the time used on the follow-up work, because we wouldn't have to log each data value manually in Microsoft Excel.

We would also have done tests with mediums of known values, to accurately find out how calibrated and reliable the sensors are.

7 Conclusion

The goals for the bachelor project was to create a fully functioning sensing rig equipped with four sensors, have battery health monitoring, and have the electronics inside the case mounted on a 3D printed layout.

Throughout the semester the milestones were completed as the finish line came closer. The first part finished was the code for the four sensors. All of the sensors but the dissolved oxygen sensor reads data as predicted, and the results from the functionality test show that the other sensors works as expected.

The 3D modeled hardware layout was also redesigned and completed, so all of the electrical components are mounted when they are inside the case, to prevent breakage.

The third part of the project, the battery health monitoring was the only underdeveloped part. The battery works as a power source for the electronics, but not the monitoring of the battery lifespan. This area, and a troubleshooting of the dissolved oxygen sensor, can be suggestions as future improvements.

What we achieved throughout this project is a good starting point for the next persons working on the sensing rig, because the rig has it's basic functionality in place.

8 Appendix

8.1 Publisher code for Temperature sensor

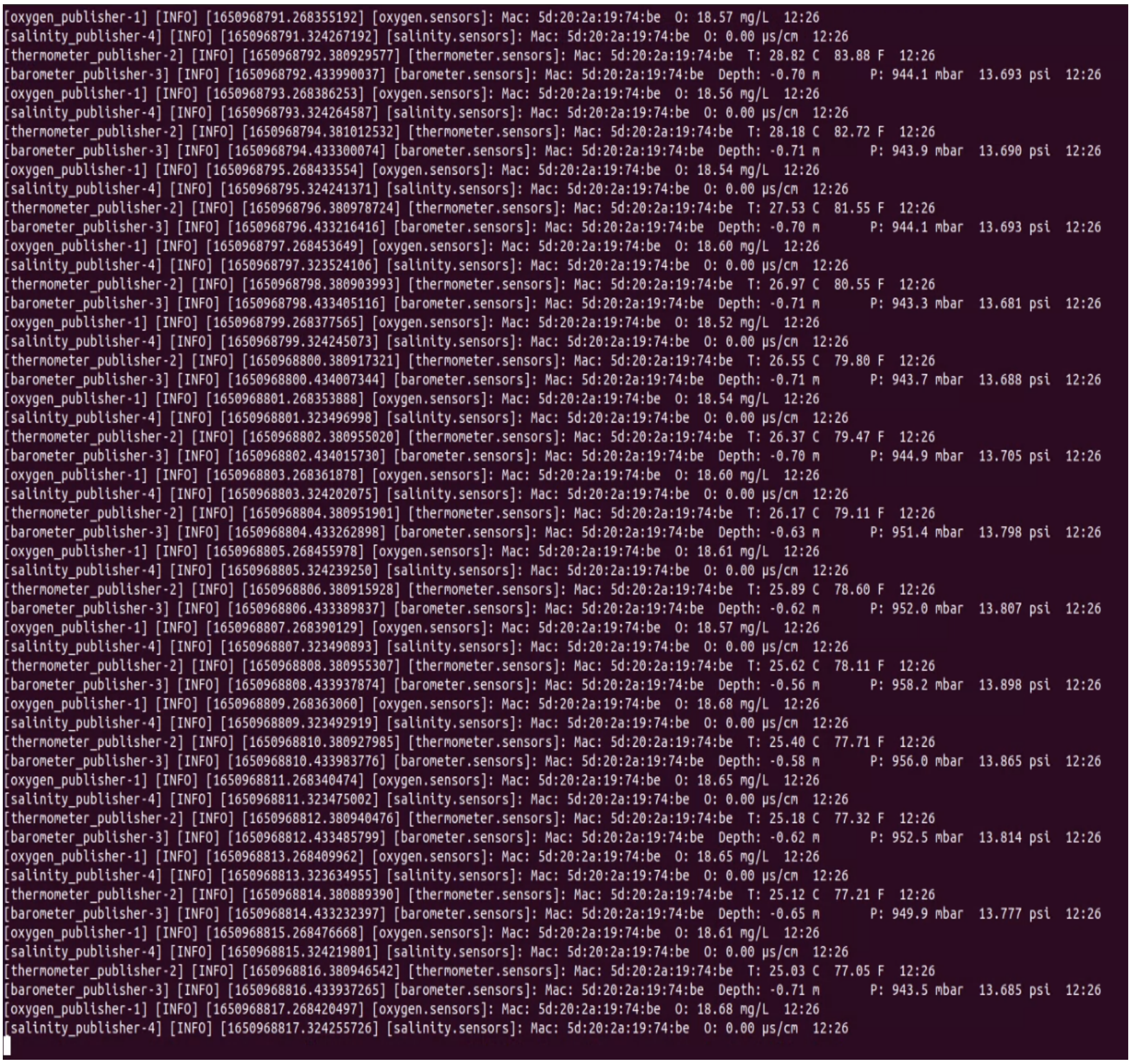
```python
# This code is responsible for getting the data from the sensor and publishing it.
from rclpy.node import Node

# tsys01 needed in order to utilize the BlueRobotics TSYS01 Python Library which must be installed
from sensor_thermometer import tsys01
from sensor_interfaces.msg import Thermometer
import time
import  re, uuid        # Used for getting the mac address

class ThermometerDataPublisher(Node):
    # Initializer
    def __init__(self):
        super().__init__('ThermometerDataPublisher')
        self.publisher_ = self.create_publisher(Thermometer, 'thermometer_data', 10)    # Creates a publisher over the topic thermometer_data
        read_period = 2  # Does a reading every 2 seconds
        self.timer = self.create_timer(read_period, self.thermometer_read_and_publish)

        # Assign the function TSYS01() in the tsys01.py to self.sensor
        self.sensor = tsys01.TSYS01()
        if not self.sensor.init():
            # If sensor can not be detected
            print("Sensor could not be initialized")
            exit(1)

    def thermometer_read_and_publish(self):
        # Custom thermometer message to publish. Can be found in the brov2_interfaces.
        msg = Thermometer()

        # Geting the local time
        tim = time.localtime()
        msg.local_time =  time.strftime("%H:%M",tim)

        # Getting the mac address of the system:
        msg.mac = ':'.join(re.findall('..','%012x' % uuid.getnode()))

        # Calls the function sensor.read in TSY01 to get the desired values from the sensors
        if self.sensor.read():
                msg.temperature_celsius     = self.sensor.temperature()                         # Default is degrees C (no arguments)
                msg.temperature_farenheit   = self.sensor.temperature(tsys01.UNITS_Farenheit)   # Request Farenheit
        else:
                print("Sensor read failed!")
                exit(1)

        # Publishing message and logging data sent over the topic /thermometer_data
        self.publisher_.publish(msg)
        self.get_logger().info('Mac: %s  T: %0.2f C  %0.2f F  %s' % (msg.mac,
                                                          msg.temperature_celsius,
                                                          msg.temperature_farenheit,
                                                          msg.local_time))
```

8.2 Data from functionality test

8.2.1 Temperature sensor data

```
[thermometer_publisher-2] [INFO] [1650968758.440352745] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 21.83 C  71.30 F  12:25
[barometer_publisher-3] [INFO] [1650968758.492107577] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 944.0 mbar  13.691 psi  12:25
[oxygen_publisher-1] [INFO] [1650968759.325211688] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.78 mg/L  12:25
[salinity_publisher-4] [INFO] [1650968759.382982512] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:25
[thermometer_publisher-2] [INFO] [1650968760.377367056] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 31.64 C  88.95 F  12:26
[barometer_publisher-3] [INFO] [1650968760.430345185] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 943.9 mbar  13.690 psi  12:26
[oxygen_publisher-1] [INFO] [1650968761.264755771] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.76 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968761.320607385] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968762.380926058] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 34.76 C  94.57 F  12:26
[barometer_publisher-3] [INFO] [1650968762.433243531] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.70 m      P: 944.1 mbar  13.694 psi  12:26
[oxygen_publisher-1] [INFO] [1650968763.268336624] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.75 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968763.323429156] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968764.380998757] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 35.96 C  96.73 F  12:26
[barometer_publisher-3] [INFO] [1650968764.433647238] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.70 m      P: 944.1 mbar  13.694 psi  12:26
[oxygen_publisher-1] [INFO] [1650968765.268420526] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.73 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968765.324260053] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968766.380958924] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 36.44 C  97.59 F  12:26
[barometer_publisher-3] [INFO] [1650968766.434029449] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 944.0 mbar  13.691 psi  12:26
[oxygen_publisher-1] [INFO] [1650968767.268432727] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.71 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968767.324269588] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968768.380926003] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 36.62 C  97.92 F  12:26
[barometer_publisher-3] [INFO] [1650968768.433309150] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 943.8 mbar  13.688 psi  12:26
[oxygen_publisher-1] [INFO] [1650968769.268364212] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.70 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968769.324458376] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968770.380922793] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 36.73 C  98.11 F  12:26
[barometer_publisher-3] [INFO] [1650968770.433995212] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 943.7 mbar  13.688 psi  12:26
[oxygen_publisher-1] [INFO] [1650968771.268410539] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.65 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968771.324266776] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968772.380975199] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 36.78 C  98.21 F  12:26
[barometer_publisher-3] [INFO] [1650968772.433409166] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.70 m      P: 944.5 mbar  13.699 psi  12:26
[oxygen_publisher-1] [INFO] [1650968773.268491480] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.72 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968773.324327515] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968774.380894829] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 36.78 C  98.21 F  12:26
[barometer_publisher-3] [INFO] [1650968774.433214352] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 943.7 mbar  13.687 psi  12:26
[oxygen_publisher-1] [INFO] [1650968775.268474033] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.64 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968775.324219402] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968776.380905055] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 35.79 C  96.42 F  12:26
[barometer_publisher-3] [INFO] [1650968776.434071999] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 943.8 mbar  13.689 psi  12:26
[oxygen_publisher-1] [INFO] [1650968777.268351607] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.65 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968777.324235610] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968778.380859398] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 34.77 C  94.59 F  12:26
[barometer_publisher-3] [INFO] [1650968778.433984955] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.70 m      P: 944.2 mbar  13.694 psi  12:26
[oxygen_publisher-1] [INFO] [1650968779.268316985] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.62 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968779.324247139] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968780.380968821] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 33.42 C  92.15 F  12:26
[barometer_publisher-3] [INFO] [1650968780.433405426] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 943.7 mbar  13.687 psi  12:26
[oxygen_publisher-1] [INFO] [1650968781.268451053] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.63 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968781.324266208] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968782.380986726] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 32.54 C  90.56 F  12:26
[barometer_publisher-3] [INFO] [1650968782.433961340] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.70 m      P: 944.6 mbar  13.700 psi  12:26
[oxygen_publisher-1] [INFO] [1650968783.268391231] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.54 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968783.324280852] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968784.380958076] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 31.85 C  89.33 F  12:26
[barometer_publisher-3] [INFO] [1650968784.433235238] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m      P: 943.6 mbar  13.686 psi  12:26
```

8.2.2 Pressure sensor data

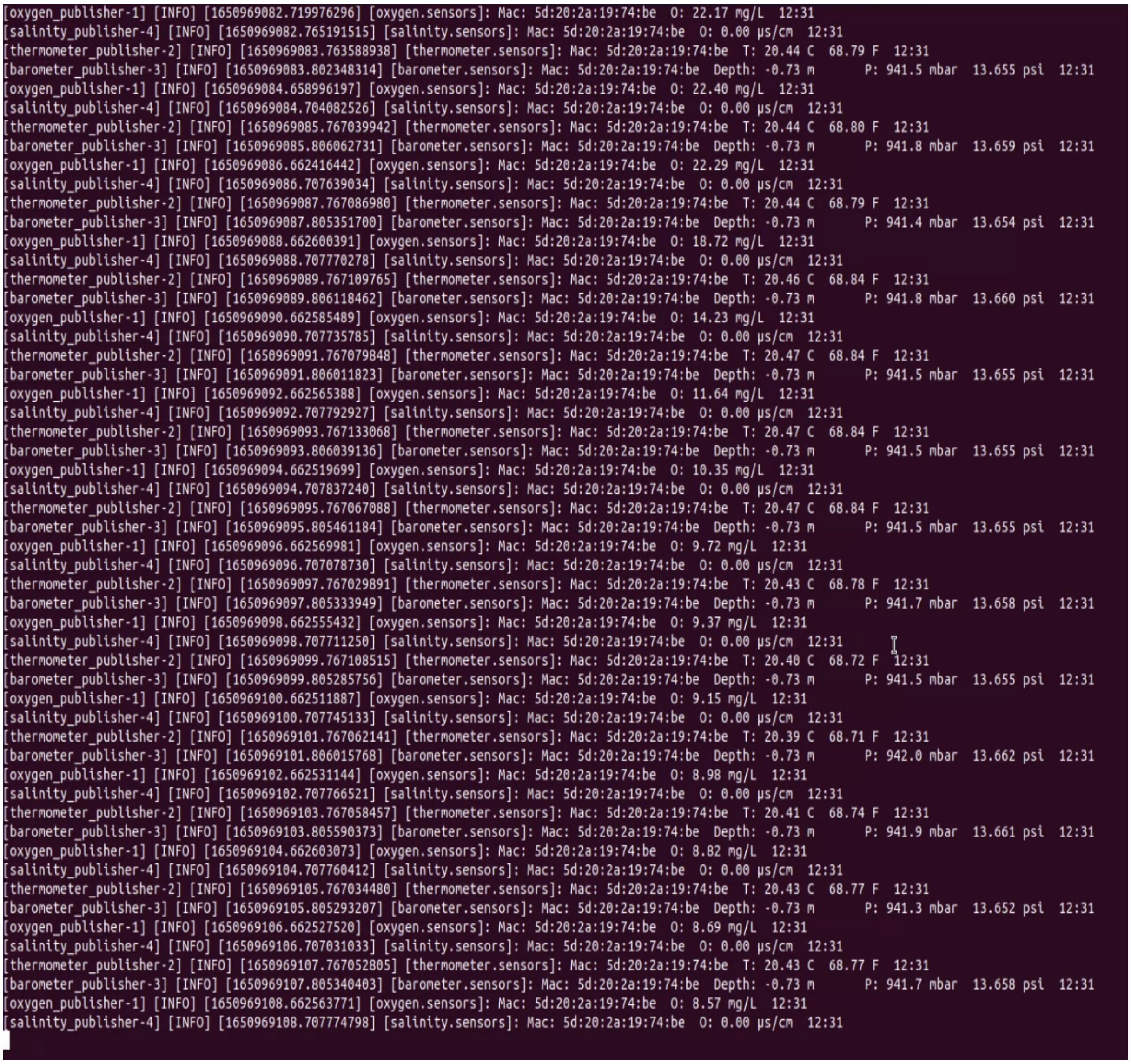


```
[oxygen_publisher-1] [INFO] [1650968791.268355192] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.57 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968791.324267192] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968792.380929577] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 28.82 C  83.88 F  12:26
[barometer_publisher-3] [INFO] [1650968792.433990037] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.70 m       P: 944.1 mbar  13.693 psi  12:26
[oxygen_publisher-1] [INFO] [1650968793.268386253] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.56 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968793.324264587] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968794.381012532] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 28.18 C  82.72 F  12:26
[barometer_publisher-3] [INFO] [1650968794.433300074] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.9 mbar  13.690 psi  12:26
[oxygen_publisher-1] [INFO] [1650968795.268433554] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.54 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968795.324241371] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968796.380978724] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 27.53 C  81.55 F  12:26
[barometer_publisher-3] [INFO] [1650968796.433216416] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.70 m       P: 944.1 mbar  13.693 psi  12:26
[oxygen_publisher-1] [INFO] [1650968797.268453649] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.60 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968797.323524106] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968798.380903993] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 26.97 C  80.55 F  12:26
[barometer_publisher-3] [INFO] [1650968798.433405116] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.3 mbar  13.681 psi  12:26
[oxygen_publisher-1] [INFO] [1650968799.268377565] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.52 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968799.324245073] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968800.380917321] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 26.55 C  79.80 F  12:26
[barometer_publisher-3] [INFO] [1650968800.434007344] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.7 mbar  13.688 psi  12:26
[oxygen_publisher-1] [INFO] [1650968801.268353888] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.54 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968801.323496998] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968802.380955020] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 26.37 C  79.47 F  12:26
[barometer_publisher-3] [INFO] [1650968802.434015730] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.70 m       P: 944.9 mbar  13.705 psi  12:26
[oxygen_publisher-1] [INFO] [1650968803.268361878] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.60 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968803.324202075] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968804.380951901] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 26.17 C  79.11 F  12:26
[barometer_publisher-3] [INFO] [1650968804.433262898] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.63 m       P: 951.4 mbar  13.798 psi  12:26
[oxygen_publisher-1] [INFO] [1650968805.268455978] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.61 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968805.324239250] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968806.380915928] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.89 C  78.60 F  12:26
[barometer_publisher-3] [INFO] [1650968806.433389837] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.62 m       P: 952.0 mbar  13.807 psi  12:26
[oxygen_publisher-1] [INFO] [1650968807.268390129] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.57 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968807.323490893] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968808.380955307] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.62 C  78.11 F  12:26
[barometer_publisher-3] [INFO] [1650968808.433937874] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.56 m       P: 958.2 mbar  13.898 psi  12:26
[oxygen_publisher-1] [INFO] [1650968809.268363060] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.68 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968809.323492919] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968810.380927985] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.40 C  77.71 F  12:26
[barometer_publisher-3] [INFO] [1650968810.433983776] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.58 m       P: 956.0 mbar  13.865 psi  12:26
[oxygen_publisher-1] [INFO] [1650968811.268340474] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.65 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968811.323475002] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968812.380940476] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.18 C  77.32 F  12:26
[barometer_publisher-3] [INFO] [1650968812.433485799] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.62 m       P: 952.5 mbar  13.814 psi  12:26
[oxygen_publisher-1] [INFO] [1650968813.268409962] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.65 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968813.323634955] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968814.380889390] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.12 C  77.21 F  12:26
[barometer_publisher-3] [INFO] [1650968814.433232397] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.65 m       P: 949.9 mbar  13.777 psi  12:26
[oxygen_publisher-1] [INFO] [1650968815.268476668] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.61 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968815.324219801] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968816.380946542] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.03 C  77.05 F  12:26
[barometer_publisher-3] [INFO] [1650968816.433937265] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.5 mbar  13.685 psi  12:26
[oxygen_publisher-1] [INFO] [1650968817.268420497] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.68 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968817.324255726] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
```

8.2.3 Dissolved oxygen in warm water data

```
[oxygen_publisher-1] [INFO] [1650968809.268363060] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.68 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968809.323492919] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968810.380927985] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.40 C  77.71 F  12:26
[barometer_publisher-3] [INFO] [1650968810.433983776] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.58 m       P: 956.0 mbar  13.865 psi  12:26
[oxygen_publisher-1] [INFO] [1650968811.268340474] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.65 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968811.323475002] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968812.380940476] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.18 C  77.32 F  12:26
[barometer_publisher-3] [INFO] [1650968812.433485799] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.62 m       P: 952.5 mbar  13.814 psi  12:26
[oxygen_publisher-1] [INFO] [1650968813.268409962] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.65 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968813.323634955] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968814.380889390] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.12 C  77.21 F  12:26
[barometer_publisher-3] [INFO] [1650968814.433232397] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.65 m       P: 949.9 mbar  13.777 psi  12:26
[oxygen_publisher-1] [INFO] [1650968815.268476668] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.61 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968815.324219801] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968816.380946542] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 25.03 C  77.05 F  12:26
[barometer_publisher-3] [INFO] [1650968816.433937265] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.5 mbar  13.685 psi  12:26
[oxygen_publisher-1] [INFO] [1650968817.268420497] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.68 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968817.324255726] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968818.380984432] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 24.86 C  76.75 F  12:26
[barometer_publisher-3] [INFO] [1650968818.434020213] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.3 mbar  13.681 psi  12:26
[oxygen_publisher-1] [INFO] [1650968819.268332438] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.65 mg/L  12:26
[salinity_publisher-4] [INFO] [1650968819.323470066] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:26
[thermometer_publisher-2] [INFO] [1650968820.380964230] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 24.56 C  76.20 F  12:27
[barometer_publisher-3] [INFO] [1650968820.433634859] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.3 mbar  13.681 psi  12:27
[oxygen_publisher-1] [INFO] [1650968821.268403490] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.68 mg/L  12:27
[salinity_publisher-4] [INFO] [1650968821.323644659] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:27
[thermometer_publisher-2] [INFO] [1650968822.380919431] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 24.32 C  75.77 F  12:27
[barometer_publisher-3] [INFO] [1650968822.433931234] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.2 mbar  13.680 psi  12:27
[oxygen_publisher-1] [INFO] [1650968823.268385668] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.65 mg/L  12:27
[salinity_publisher-4] [INFO] [1650968823.324235866] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:27
[thermometer_publisher-2] [INFO] [1650968824.380916703] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 24.08 C  75.34 F  12:27
[barometer_publisher-3] [INFO] [1650968824.433104069] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.6 mbar  13.685 psi  12:27
[oxygen_publisher-1] [INFO] [1650968825.268423156] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.51 mg/L  12:27
[salinity_publisher-4] [INFO] [1650968825.323687996] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:27
[thermometer_publisher-2] [INFO] [1650968826.380898118] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 23.87 C  74.97 F  12:27
[barometer_publisher-3] [INFO] [1650968826.433977259] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.3 mbar  13.681 psi  12:27
[oxygen_publisher-1] [INFO] [1650968827.268398029] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.85 mg/L  12:27
[salinity_publisher-4] [INFO] [1650968827.324412567] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:27
[thermometer_publisher-2] [INFO] [1650968828.380979736] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 23.75 C  74.76 F  12:27
[barometer_publisher-3] [INFO] [1650968828.433246329] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.8 mbar  13.689 psi  12:27
[oxygen_publisher-1] [INFO] [1650968829.268415010] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.81 mg/L  12:27
[salinity_publisher-4] [INFO] [1650968829.323531610] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:27
[thermometer_publisher-2] [INFO] [1650968830.380911793] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 23.73 C  74.71 F  12:27
[barometer_publisher-3] [INFO] [1650968830.433280649] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.4 mbar  13.683 psi  12:27
[oxygen_publisher-1] [INFO] [1650968831.268362803] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 21.61 mg/L  12:27
[salinity_publisher-4] [INFO] [1650968831.324200361] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:27
[thermometer_publisher-2] [INFO] [1650968832.380906070] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 23.68 C  74.63 F  12:27
[barometer_publisher-3] [INFO] [1650968832.433974605] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.72 m       P: 943.1 mbar  13.678 psi  12:27
[oxygen_publisher-1] [INFO] [1650968833.268357684] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 23.20 mg/L  12:27
[salinity_publisher-4] [INFO] [1650968833.324300986] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:27
[thermometer_publisher-2] [INFO] [1650968834.380921600] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 23.64 C  74.55 F  12:27
[barometer_publisher-3] [INFO] [1650968834.433251144] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.5 mbar  13.685 psi  12:27
[oxygen_publisher-1] [INFO] [1650968835.268381133] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 23.89 mg/L  12:27
[salinity_publisher-4] [INFO] [1650968835.324274233] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:27
```

8.2.4 Dissolved oxygen in cold water data



```
[oxygen_publisher-1] [INFO] [1650969082.719976296] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 22.17 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969082.765191515] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969083.763588938] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.44 C  68.79 F  12:31
[barometer_publisher-3] [INFO] [1650969083.802348314] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.5 mbar  13.655 psi  12:31
[oxygen_publisher-1] [INFO] [1650969084.658996197] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 22.40 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969084.704082526] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969085.767039942] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.44 C  68.80 F  12:31
[barometer_publisher-3] [INFO] [1650969085.806062731] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.8 mbar  13.659 psi  12:31
[oxygen_publisher-1] [INFO] [1650969086.662416442] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 22.29 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969086.707639034] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969087.767086980] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.44 C  68.79 F  12:31
[barometer_publisher-3] [INFO] [1650969087.805351700] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.4 mbar  13.654 psi  12:31
[oxygen_publisher-1] [INFO] [1650969088.662600391] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 18.72 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969088.707770278] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969089.767109765] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.46 C  68.84 F  12:31
[barometer_publisher-3] [INFO] [1650969089.806118462] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.8 mbar  13.660 psi  12:31
[oxygen_publisher-1] [INFO] [1650969090.662585489] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 14.23 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969090.707735785] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969091.767079848] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.47 C  68.84 F  12:31
[barometer_publisher-3] [INFO] [1650969091.806011823] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.5 mbar  13.655 psi  12:31
[oxygen_publisher-1] [INFO] [1650969092.662565388] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 11.64 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969092.707792927] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969093.767133068] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.47 C  68.84 F  12:31
[barometer_publisher-3] [INFO] [1650969093.806039136] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.5 mbar  13.655 psi  12:31
[oxygen_publisher-1] [INFO] [1650969094.662519699] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 10.35 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969094.707837240] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969095.767067088] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.47 C  68.84 F  12:31
[barometer_publisher-3] [INFO] [1650969095.805461184] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.5 mbar  13.655 psi  12:31
[oxygen_publisher-1] [INFO] [1650969096.662569981] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 9.72 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969096.707078730] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969097.767029891] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.43 C  68.78 F  12:31
[barometer_publisher-3] [INFO] [1650969097.805333949] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.7 mbar  13.658 psi  12:31
[oxygen_publisher-1] [INFO] [1650969098.662555432] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 9.37 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969098.707711250] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969099.767108515] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.40 C  68.72 F  12:31
[barometer_publisher-3] [INFO] [1650969099.805285756] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.5 mbar  13.655 psi  12:31
[oxygen_publisher-1] [INFO] [1650969100.662511887] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 9.15 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969100.707745133] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969101.767062141] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.39 C  68.71 F  12:31
[barometer_publisher-3] [INFO] [1650969101.806015768] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 942.0 mbar  13.662 psi  12:31
[oxygen_publisher-1] [INFO] [1650969102.662531144] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 8.98 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969102.707766521] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969103.767058457] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.41 C  68.74 F  12:31
[barometer_publisher-3] [INFO] [1650969103.805590373] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.9 mbar  13.661 psi  12:31
[oxygen_publisher-1] [INFO] [1650969104.662603073] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 8.82 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969104.707760412] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969105.767034480] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.43 C  68.77 F  12:31
[barometer_publisher-3] [INFO] [1650969105.805293207] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.3 mbar  13.652 psi  12:31
[oxygen_publisher-1] [INFO] [1650969106.662527520] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 8.69 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969106.707031033] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
[thermometer_publisher-2] [INFO] [1650969107.767052805] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 20.43 C  68.77 F  12:31
[barometer_publisher-3] [INFO] [1650969107.805340403] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.73 m       P: 941.7 mbar  13.658 psi  12:31
[oxygen_publisher-1] [INFO] [1650969108.662563771] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 8.57 mg/L  12:31
[salinity_publisher-4] [INFO] [1650969108.707774798] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:31
```

8.2.5 Conductivity sensor data

```
[thermometer_publisher-2] [INFO] [1650968884.380915105] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.30 C  72.14 F  12:28
[barometer_publisher-3] [INFO] [1650968884.433356360] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.3 mbar  13.681 psi  12:28
[oxygen_publisher-1] [INFO] [1650968885.268452873] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.56 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968885.323509506] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968886.380861854] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.28 C  72.10 F  12:28
[barometer_publisher-3] [INFO] [1650968886.433361891] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.4 mbar  13.683 psi  12:28
[oxygen_publisher-1] [INFO] [1650968887.268364077] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.60 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968887.324191483] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 0.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968888.380894059] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.29 C  72.12 F  12:28
[barometer_publisher-3] [INFO] [1650968888.433965828] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.5 mbar  13.685 psi  12:28
[oxygen_publisher-1] [INFO] [1650968889.268339380] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.63 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968889.324211807] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 6456.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968890.380917322] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.30 C  72.14 F  12:28
[barometer_publisher-3] [INFO] [1650968890.433939148] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.72 m       P: 942.9 mbar  13.676 psi  12:28
[oxygen_publisher-1] [INFO] [1650968891.268296635] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.68 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968891.323436218] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 12480.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968892.380983151] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.30 C  72.13 F  12:28
[barometer_publisher-3] [INFO] [1650968892.433357598] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.1 mbar  13.679 psi  12:28
[oxygen_publisher-1] [INFO] [1650968893.268377455] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.76 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968893.323507966] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 14800.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968894.380911188] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.22 C  72.00 F  12:28
[barometer_publisher-3] [INFO] [1650968894.433971388] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.72 m       P: 942.9 mbar  13.676 psi  12:28
[oxygen_publisher-1] [INFO] [1650968895.268406038] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.77 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968895.324206228] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 12990.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968896.380867822] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.15 C  71.86 F  12:28
[barometer_publisher-3] [INFO] [1650968896.433541332] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.4 mbar  13.683 psi  12:28
[oxygen_publisher-1] [INFO] [1650968897.268355330] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.79 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968897.324209707] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 1423.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968898.380874168] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.06 C  71.71 F  12:28
[barometer_publisher-3] [INFO] [1650968898.433959038] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.72 m       P: 943.0 mbar  13.678 psi  12:28
[oxygen_publisher-1] [INFO] [1650968899.268455316] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.76 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968899.323506516] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 1095.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968900.380930854] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.03 C  71.66 F  12:28
[barometer_publisher-3] [INFO] [1650968900.433556330] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.72 m       P: 943.0 mbar  13.677 psi  12:28
[oxygen_publisher-1] [INFO] [1650968901.268354119] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.76 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968901.323688844] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 1442.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968902.380889026] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.03 C  71.65 F  12:28
[barometer_publisher-3] [INFO] [1650968902.433914991] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.72 m       P: 942.8 mbar  13.674 psi  12:28
[oxygen_publisher-1] [INFO] [1650968903.268280854] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.78 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968903.324178997] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 5749.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968904.380888981] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.03 C  71.66 F  12:28
[barometer_publisher-3] [INFO] [1650968904.433927410] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.72 m       P: 943.0 mbar  13.678 psi  12:28
[oxygen_publisher-1] [INFO] [1650968905.268375078] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.72 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968905.323613008] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 6490.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968906.380904644] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.06 C  71.70 F  12:28
[barometer_publisher-3] [INFO] [1650968906.433948779] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.1 mbar  13.679 psi  12:28
[oxygen_publisher-1] [INFO] [1650968907.268406621] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.70 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968907.324200247] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 6965.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968908.380894460] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.07 C  71.73 F  12:28
[barometer_publisher-3] [INFO] [1650968908.433318661] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.2 mbar  13.679 psi  12:28
[oxygen_publisher-1] [INFO] [1650968909.268370313] [oxygen.sensors]: Mac: 5d:20:2a:19:74:be  O: 25.68 mg/L  12:28
[salinity_publisher-4] [INFO] [1650968909.323513874] [salinity.sensors]: Mac: 5d:20:2a:19:74:be  O: 6995.00 µs/cm  12:28
[thermometer_publisher-2] [INFO] [1650968910.380911316] [thermometer.sensors]: Mac: 5d:20:2a:19:74:be  T: 22.06 C  71.71 F  12:28
[barometer_publisher-3] [INFO] [1650968910.433128255] [barometer.sensors]: Mac: 5d:20:2a:19:74:be  Depth: -0.71 m       P: 943.6 mbar  13.686 psi  12:28
```

8.3 Table of screws

Hole number	Dimension	Length(mm)	Threaded	Screws between
1	M2,5	10	No	Main plate + circle
2	M2,5	10	No	Main plate + circle
3	M4	24	No	Main plate + barrier strip
4	M4	24	No	Main plate + barrier strip
5	M4	24	No	Main plate + barrier strip
6	M4	24	No	Main plate + barrier strip
7	M3	28	Yes	Main plate + switch stand
8	M3	28	Yes	Main plate + switch stand
9	M3	28	Yes	Main plate + switch stand
10	M3	28	Yes	Main plate + switch stand
11	M3	8	Yes	Main plate + isolated carrier board
12	M3	8	Yes	Main plate + isolated carrier board
13	M3	8	Yes	Main plate + isolated carrier board

14	M3	8	Yes	Main plate + isolated carrier board
15	M3	8	Yes	Main plate + isolated carrier board
16	M3	8	Yes	Main plate + isolated carrier board
17	M2,5	6	Yes	Main plate + ADC
18	M2,5	6	Yes	Main plate + ADC
19	M3	8	Yes	Main plate + splitter stand
20	M3	8	Yes	Main plate + splitter stand
21	M3	16	No	Main plate + half circle
22	M3	16	No	Main plate + half circle
23	M3	16	Yes	Main plate + circle
24	M3	16	Yes	Main plate + circle
25	M3	16	Yes	Main plate + lock
26	M3	16	Yes	Main plate + lock
27	M3	16	Yes	Spring/trackplate + raspi-plate
28	M3	16	Yes	Spring/trackplate + raspi-plate
29	M3	16	Yes	Spring/trackplate + raspi-plate
30	M3	16	Yes	Spring/trackplate + raspi-plate
31	M2,5	10	No	Spring/trackplate + raspi-plate
32	M2,5	10	No	Spring/trackplate + raspi-plate
33	M2,5	10	No	Spring/trackplate + raspi-plate
34	M2,5	10	No	Spring/trackplate + raspi-plate
35	M3	8	No	Splitter stand + splitter
36	M3	10	No	Circle + lid
37	M3	10	No	Circle + lid
38	M3	10	No	Circle + lid
39	M3	10	No	Circle + lid

8.4 Procedure 3D printing

Procedure 3D printing

Written by: Julie Klingenberg. Spring 2022

Introduction

This document describes the 3D printing procedure done in the bachelor thesis “underwater sensing rig”. The document describes which programs were used, and useful tools for making the process easier. It also gives an overview of screws and threading tools, files and punctual procedure on the printing and attachment.

A brief description of the project

The goal of the project was to create a fully functioning sensing rig equipped with four sensors, battery health monitoring, and the electronics inside the case is mounted on a 3D printed layout. The 3D printed layout consisted of a main plate, where all the different parts were to be fastened. The layout also consisted of a circle, and a semicircle attached at the ends of the main plate. Two different stands were made for two of the circuit boards to save space on the main plate. In addition to all this, there were also made a plate for attaching the Raspberry Pi. This plate was fastened to another plate, either a spring or just a plate with tracks. These plates could slide on or of the rail on the main plate. The spring would be used if the Raspberry Pi had a heat sink, or else the trackplate would be used.

Programs

It is mainly one program needed to print the parts, the Ultimaker Cura. Cura is a slicer program and prepares settings for an already modelled object to be printed. The slicer is a program that translates the model into something the printer understands. In this case, it is the path the printer must take to create an object. There is a lot of settings for this program, and most of them don’t need to be changed. Also, a lot of the settings are already set by the program in the recommended settings. But often we want to change some of settings such as infill percentage, support, adhesion, and other things. The size, orientation, and location can be changed in this program. One can also choose places on the model that should have other settings than the rest of the model. This is described in the procedure as some places need to have infill percent of 100 percent.

Even if you don’t need it, you may also want to have the program SketchUp. This is the program the models are made in.

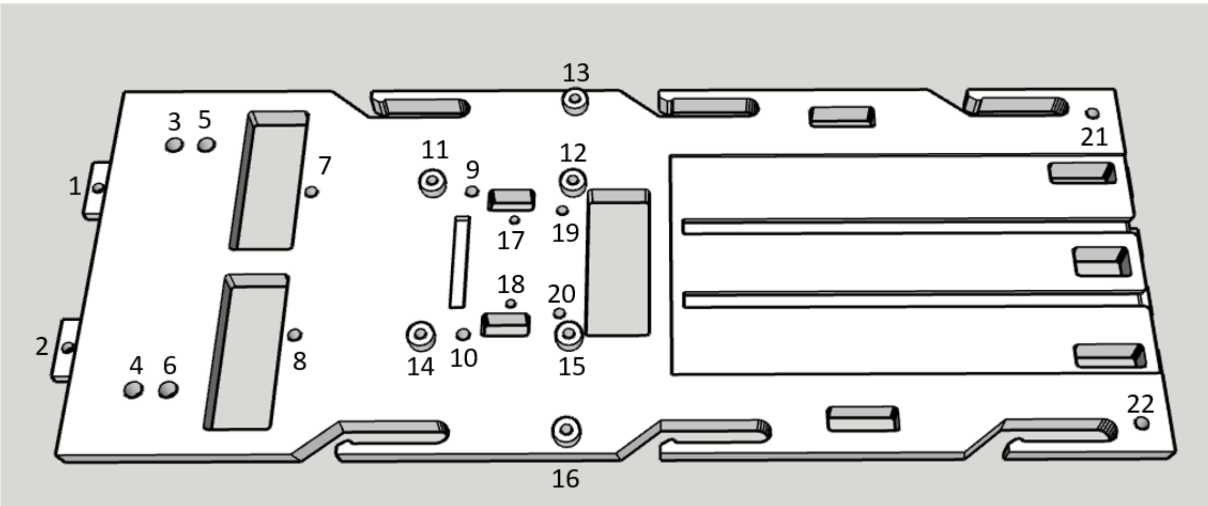
Tools

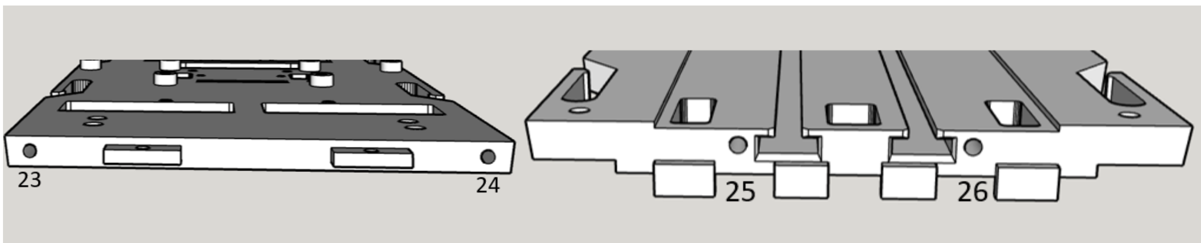
After the print is finished there will be support structures of plastic on different places on the object. These must be removed, and tools are often needed for this. One also needs screws and threading pins for the holes. The recommended tools are:

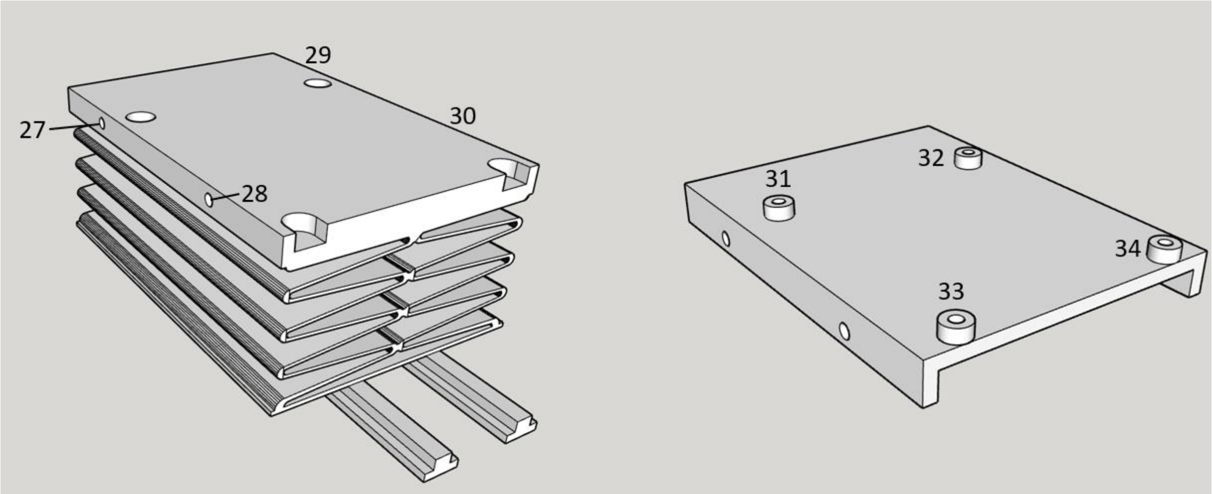
- Pair of pliers
- Razor blade or knife
- Flat head screwdriver
- Screws of different dimensions (they are described in a table)
- Nuts
- Threaded pins (M2,5 and M3)
- Screwdriver set (recommends iFixit Mako Driver Kit)

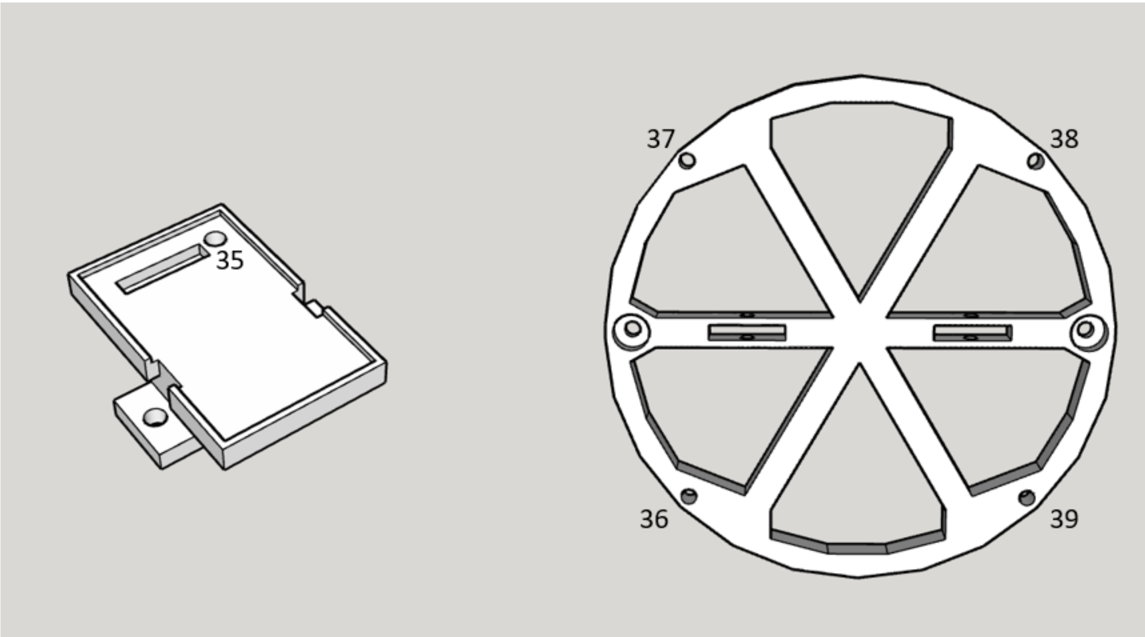
Overview

To get an overview of all screws and holes, a picture of all holes on the main plate, the spring/trackplate, the raspi-plate, the splitter stand, and circle were made. Each hole has been assigned a number. The table below the pictures describes the types of screws to be used, as well as their length, whether the holes are to be threaded up, and which part they belong to.









Hole number	Dimension	Length(mm)	Threaded	Screws between
1	M2,5	10	No	Main plate + circle
2	M2,5	10	No	Main plate + circle
3	M4	24	No	Main plate + barrier strip
4	M4	24	No	Main plate + barrier strip
5	M4	24	No	Main plate + barrier strip
6	M4	24	No	Main plate + barrier strip
7	M3	28	Yes	Main plate + switch stand

8	M3	28	Yes	Main plate + switch stand
9	M3	28	Yes	Main plate + switch stand
10	M3	28	Yes	Main plate + switch stand
11	M3	8	Yes	Main plate + isolated carrier board
12	M3	8	Yes	Main plate + isolated carrier board
13	M3	8	Yes	Main plate + isolated carrier board
14	M3	8	Yes	Main plate + isolated carrier board
15	M3	8	Yes	Main plate + isolated carrier board
16	M3	8	Yes	Main plate + isolated carrier board
17	M2,5	6	Yes	Main plate + ADC
18	M2,5	6	Yes	Main plate + ADC
19	M3	8	Yes	Main plate + splitter stand
20	M3	8	Yes	Main plate + splitter stand
21	M3	16	No	Main plate + half circle
22	M3	16	No	Main plate + half circle
23	M3	16	Yes	Main plate + circle
24	M3	16	Yes	Main plate + circle
25	M3	16	Yes	Main plate + lock
26	M3	16	Yes	Main plate + lock
27	M3	16	Yes	Spring/trackplate + raspi-plate
28	M3	16	Yes	Spring/trackplate + raspi-plate

29	M3	16	Yes	Spring/trackplate + raspi-plate
30	M3	16	Yes	Spring/trackplate + raspi-plate
31	M2,5	10	No	Spring/trackplate + raspi-plate
32	M2,5	10	No	Spring/trackplate + raspi-plate
33	M2,5	10	No	Spring/trackplate + raspi-plate
34	M2,5	10	No	Spring/trackplate + raspi-plate
35	M3	8	No	Splitter stand + splitter
36	M3	10	No	Circle + lid
37	M3	10	No	Circle + lid
38	M3	10	No	Circle + lid
39	M3	10	No	Circle + lid

Files

To print the parts, one must find the right models. In the zip files there are two folders. One of them is the SketchUp file that can be edited, while the other one is the STL files for direct print. In the SketchUp folder we find the models both alone and together as they are printed. There are two main files, the two files that will be printed. One of them are the main plate alone, while the other is the remaining parts put together. This is because it is more efficient to print many parts at the same time. In these files all the models are oriented in a way that uses the least support, material, and time.

The other folder named "Cura files" has the STL file that is used in Cura. By opening these files, you will directly enter Cura, and can start correcting the settings. These files contain the same as the SketchUp files with multiple objects, just exported to STL files for a direct print in Cura.

Tips

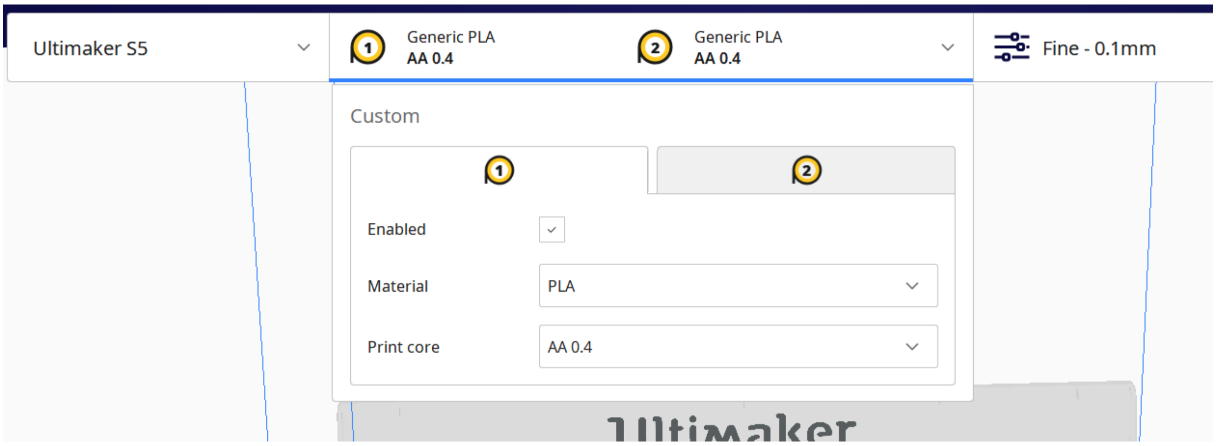
Use a computer mouse to easily orient in Cura. The wheel on the computer mouse functions as zoom in and zoom out, but you can also change position by clicking on the wheel and drag. The left button acts as an orbit tool, so you can see the model from several angles.

Some of the printed holes that were not meant to be threaded up may be too small when printing, and it may be too difficult to screw the screws through the hole. One can either just screw the screw directly into the hole by using force or choose to thread up the hole instead.

Procedure

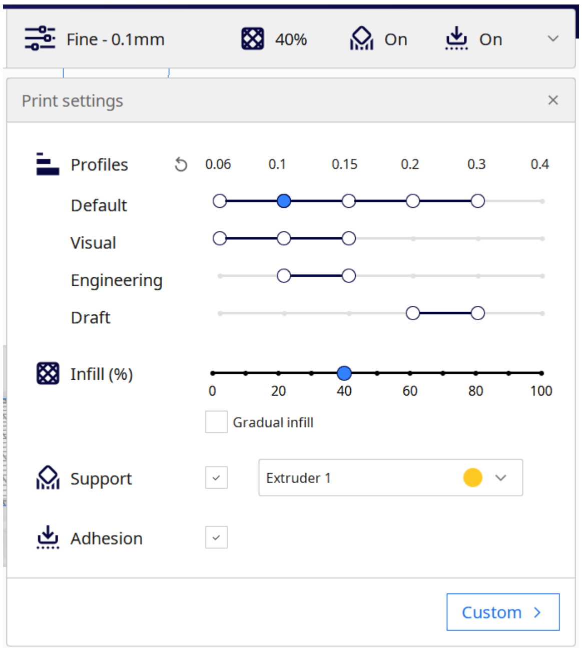
Mainplate_print procedure

1. Start the Ultimaker Cura program
2. Choose the right printer up in the left corner of the screen. This is done by choosing "add printer" and then "Add a non-networked printer". Here is a list of all the printers available. Choose "Ultimaker S5" and click "Add". Note that there is a video on how to do this.
3. Check if the middle part of the top menu says this:



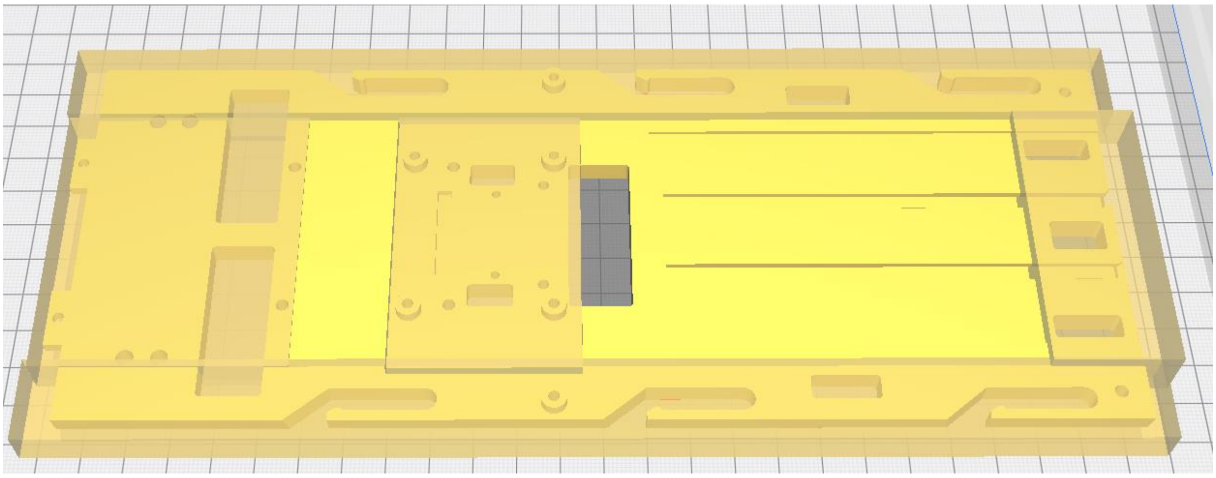
There is only going to be used one nozzle, so ignore the second.

4. Now the model can be uploaded. This is done by clicking the folder symbol far left in the upper corner. Select the right file, in this case the file with name "Mainplate_print".
5. Now its time to select the settings. Even though there are recommended settings, they might have to be changed to achieve the desired result. For this print the infill percent is set to 40%, and both "support" and "build plate adhesion" is turned on. The print settings should look something like this:



See video "Starting Ultimaker Cura" for a demonstration on nr 1-5.

6. To make the model strong enough, it is possible to change the infill percentage to 100 in different places. This is often around holes, and thin areas. This is done by first using a “Support Blocker” located at the bottom of the menu on the left. The blocker is then placed in the desired place on the model by moving and resizing the block. Moving and resizing can be found at the top of this menu. When resizing it is important to make sure that “Snap Scaling” and “Uniform Scaling” is turned off. Make five blocks and place them as shown under:



   When the block covers the desired area, it can change its infill settings by selecting the block and then pressing "Per Model Settings" in the menu. Here, select «Modify settings for overlaps», and press "Select settings". Inside the settings, click «Infill Density» and then press “Close”. A box should have appeared at the bottom of the menu under "Modify settings for overlaps" with the name "Infill Density". Change this to 100 percent.
   See video “Add different infill percent” for a demonstration on nr 6.
7. Slice the model by clicking slice. Here it’s shown how long the print will take and how much material will be used. Se video “Slicing”.
8. Click “Preview” to see how the model will be printed.
   NB! Here it is very important to check the preview carefully. There may be problems with the model. Check that the layers are similar to the model in SketchUp. If there are places on the model in Cura that do not match, try reuploading the model. If the same error occurs again, go into SketchUp, and create extra lines over the existing lines with line tool where the error occurred. Then re-export the file with the same name. Do not close Cura, as the change will occur almost automatically in Cura. A window will appear asking if you want to upload the new version. Do this. Then all previous settings will be the same. Slice again and check preview again. Do this until the model is exactly the same as in SketchUp.
9. When the model is sliced and checked, the file has to be saved on an USB stick. This is done by plugging in the USB stick to the computer, and then clicking the blue button down in the right corner “Save to removable disk”.
10. Before printing it’s important to see if the printer is ready. Check if there is enough material. The material holder is located on the back of the printer. Also check that there is no previous print running.

11. Plug the USB pin in the 3D printer. Wait till the button on the screen says “Select from USB”, and then click it.

Choose the right file, with name “Mainplate_print”. Check that the information on the printer corresponds to what is written in Cura on the computer. Click “Start print” to start the print.

12. Wait till the printing starts (around 4 minutes after pressing start) before leaving, to be sure everything is working as it should. The printer uses some time to heat up the material in the nozzle and the printing bed. It also runs a calibration at the start.
13. When the print is done, the object has to be removed from the print bed. Be careful as the build plate is made of glass. Click “Confirm removal” on the printer when the object is removed.

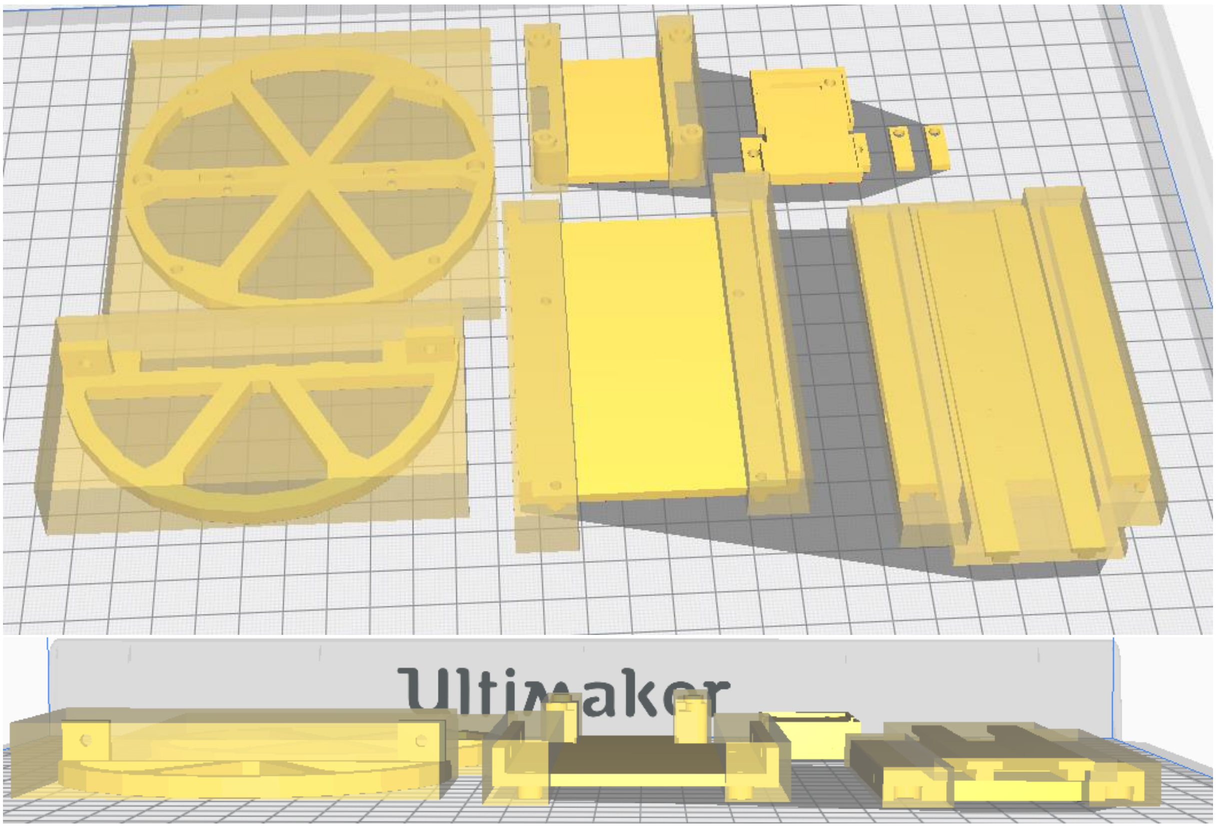
14. Remove the support plastic by using the tools listed over. It can be challenging to remove the support some places, like in the rail and in holes. Be sure to remove all off the support, especially in horizontal holes that are to be threaded up. If not all the support is removed, the hole will be too short and wide when trying to thread up.

15. Thread up the holes marked with “Yes” in the table. When threading, it is important to occasionally turn back a little so that the loose plastic can be pulled out. Then screw in two rounds, and a half back. continue like this until it becomes heavy to turn around. Then the end is reached. Unscrew the threaded pin completely and brush off the plastic that comes with it. Do the same thing again in the same hole to make sure that all the loose plastic is removed.

    On careful when threading holes 17-20. It is easy to go too far, and out on the other side of the plate. Therefore, you may want to keep an eye on the back of the plate when threading these holes

Remaining_parts procedure

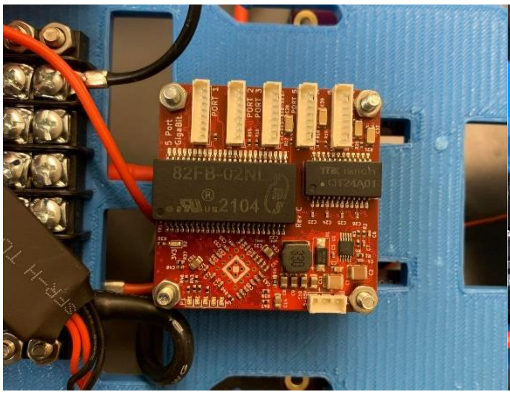
1. Do step 1-3 in the Main_plate procedure.
2. Upload the model with name Remaining_parts by clicking the folder symbol in up in the left corner in Cura.
3. Set the basic settings to the same as in the Main_plate procedure.
4. Make 9 boxes. Resize and place them as shown in the pictures under. To make these boxes go to step 6 in the Main_plate procedure, or watch the video “The other parts” on how to do this.

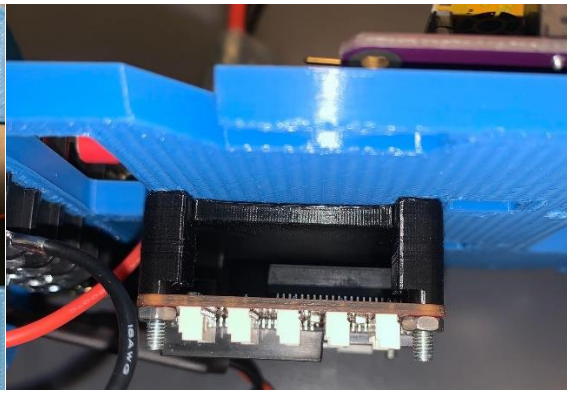
    

    Change the infill percent to 80 on the semicircle. Then change the infill percent to 100 for the remaining boxes.
5. Do step 7-13 in the Main_plate proceduse, but instead of choosing “Main_plate”, choose “Remaining_parts” on the printer. See video “Slicing and preview”.

6. Remove all the support of every part. Use the different tools listed over. Sometimes the edges can be uneven, so use the razor blade to smooth them out. Be careful to not cut yourself.
7. When you are sure that all the support plastic has been removed in the holes on the plate with rails, thread up the holes marked with “yes” in the table.

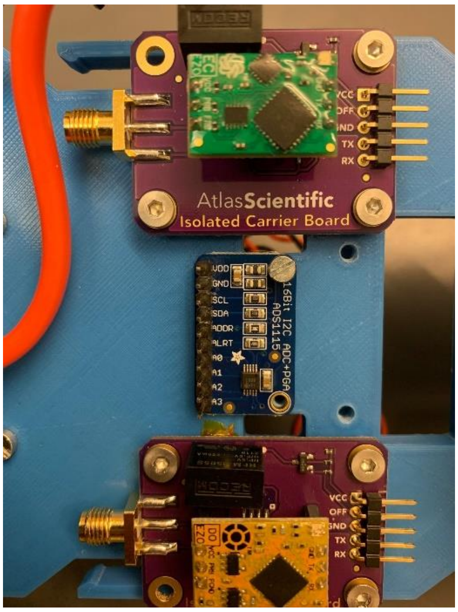
Attaching everything

1. Start by attaching the switch. This is done by screwing the M3x28mm screws in holes 7-10 from the top of the main plate and down through the switch stand. When the screws are fully screwed in, put the switch in place on top of the stand. Secure the switch by screwing nuts on the screws on top of the switch. It should look like this:


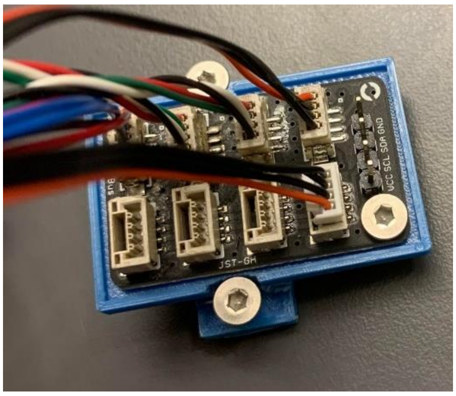

2. Next is fastening the two insulated carrier boards. Place them on top of the small cylinders, hole 11-16, on the main plate and secure them with M3x8mm screws. The picture on the right shows the first print of the plate where one of the isolated carrier boards touched the screw with pins under the circuit board. This was fixed by using isolated tape, called Kapton tape. The issue is fixed on the file used in this procedure, by elevating the cylinders with 2 mm. If the pins still touch the screw, or is too close to the screw, the pins can be slightly cut. Because of this you can ignore the yellow tape attached to the screw in the pictures.

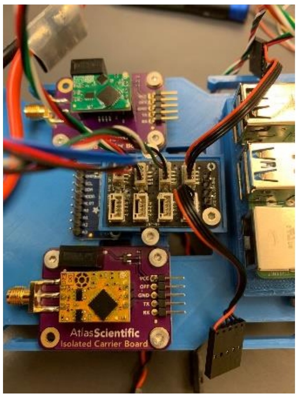


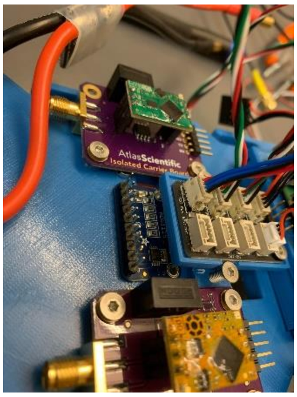
3. Attach the ADC to the main plate between the two insulated carrier boards. Use M2.5x6mm screws in hole 17 and 18.



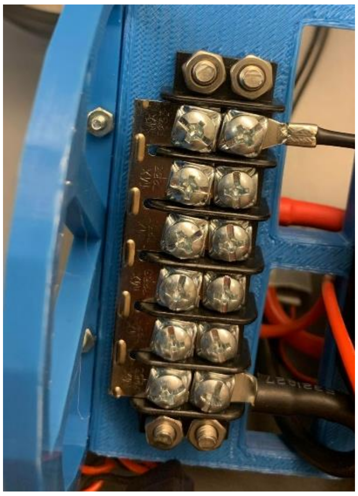
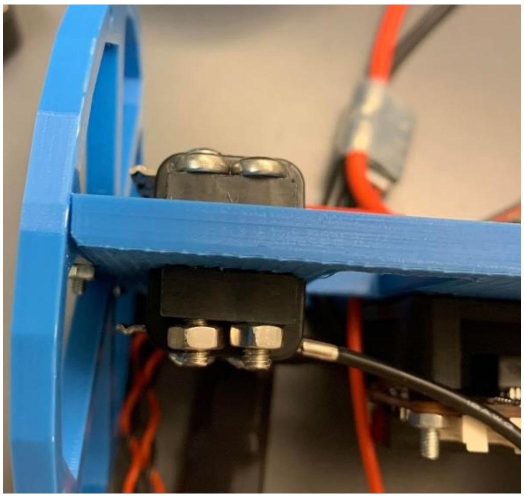
4. Fasten the splitter to the splitter stand by screwing a M3x8mm screw through the splitter and the stand, hole 35. Fasten the screw with a nut. Then attach the stand to the main plate directly in front of the ADC with the longest part of the stand pointing backwards and above the ADC, hole 19 and 20.



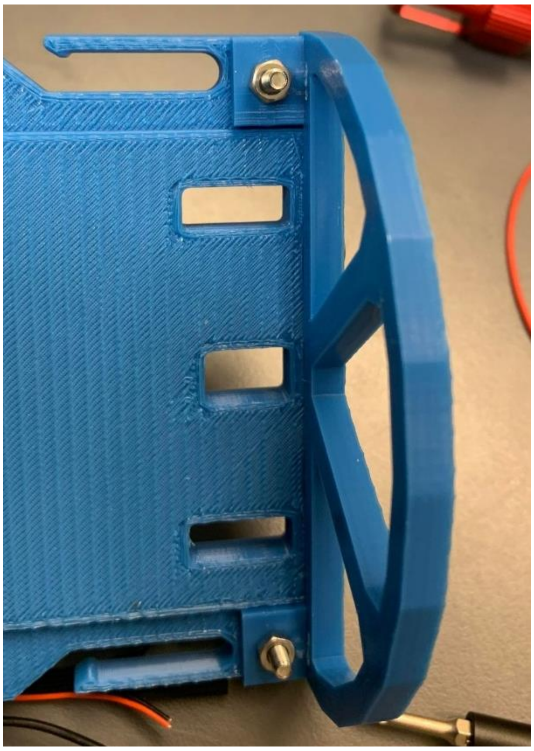




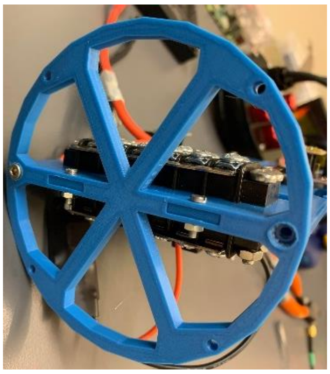
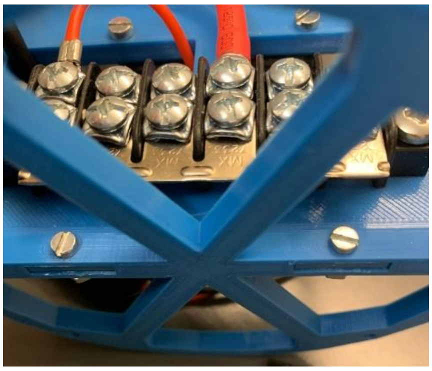
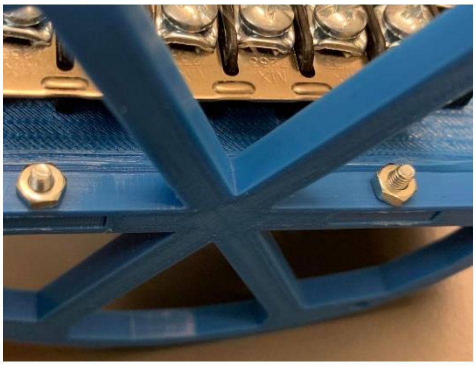
5. Place the two barrier strips at the back of the main plate on opposite sides. The jumpers should point towards the circle. Use a M4x24mm to fasten the two barrier strips to the plate, by screwing first trough the first barrier strip, then the main plate and then the last barrier strip, in hole 3-6. Secure the screw with nuts.

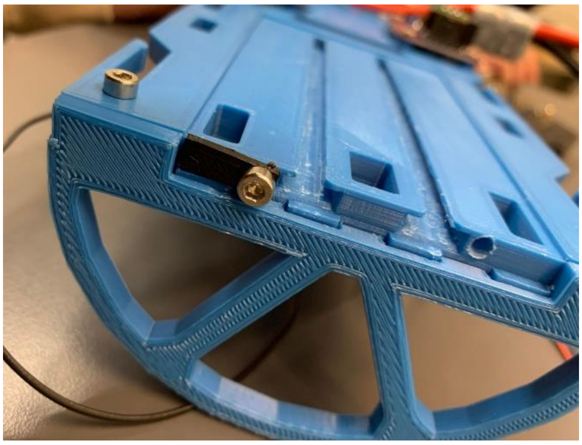
6. Place the semicircle in the tracks at the front of the main plate. You have to use some force to get it in place. Fasten the semicircle to the main plate by M3x16mm screws in hole 21 and 22.



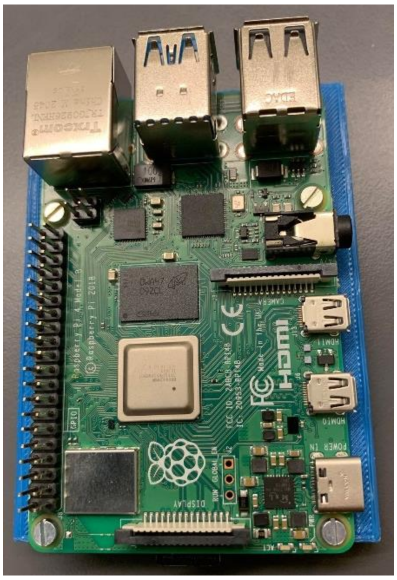

7. Slide the circle onto the pins on the other side of the main plane. Fasten the screws trough the pins, hole 1 and 2, by M2,5x10mm screws. Then fasten two screws in the two holes on each side, 23 and 24, by M3x16mm screws. The first version of the print had a defect circle, therefore there are only one screw on the pictures. On the version attached in this procedure this is no problem.

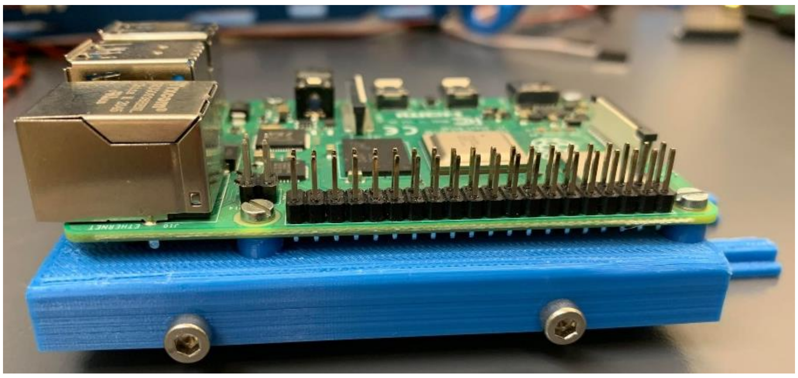
8. Screw the two locks to the front side of the main plate, in hole 25 and 26. There were also a problem here. Due to the fact that not all the plastic inside the hole was removed, the hole became too large when threading it up. Therefore, it is important to make sure that all the plastic is removed.



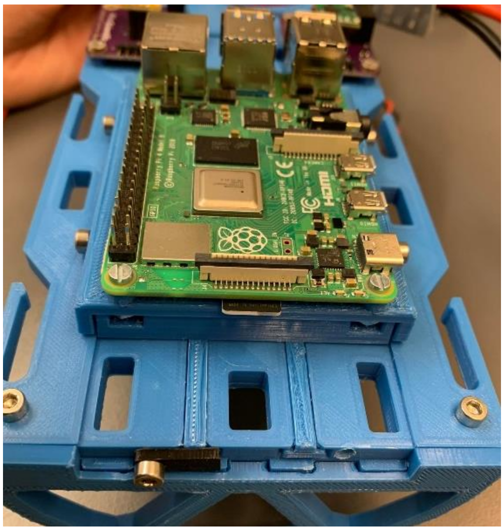
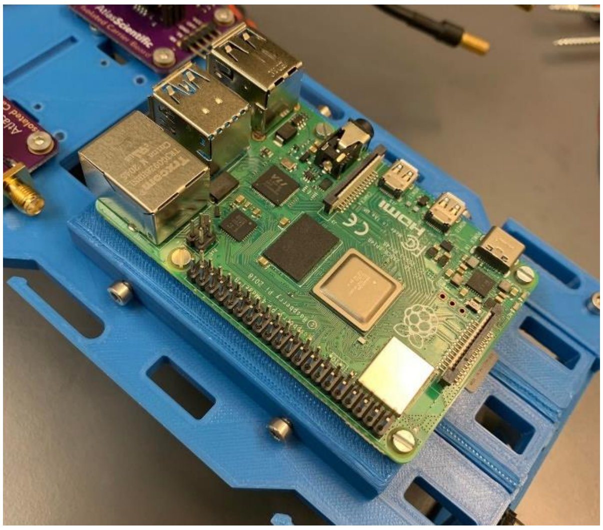
9. Fasten the Raspberry Pi to the Raspi-plate, by using M2,5x10mm, in holes 31-34. Use nuts to secure on the underside.



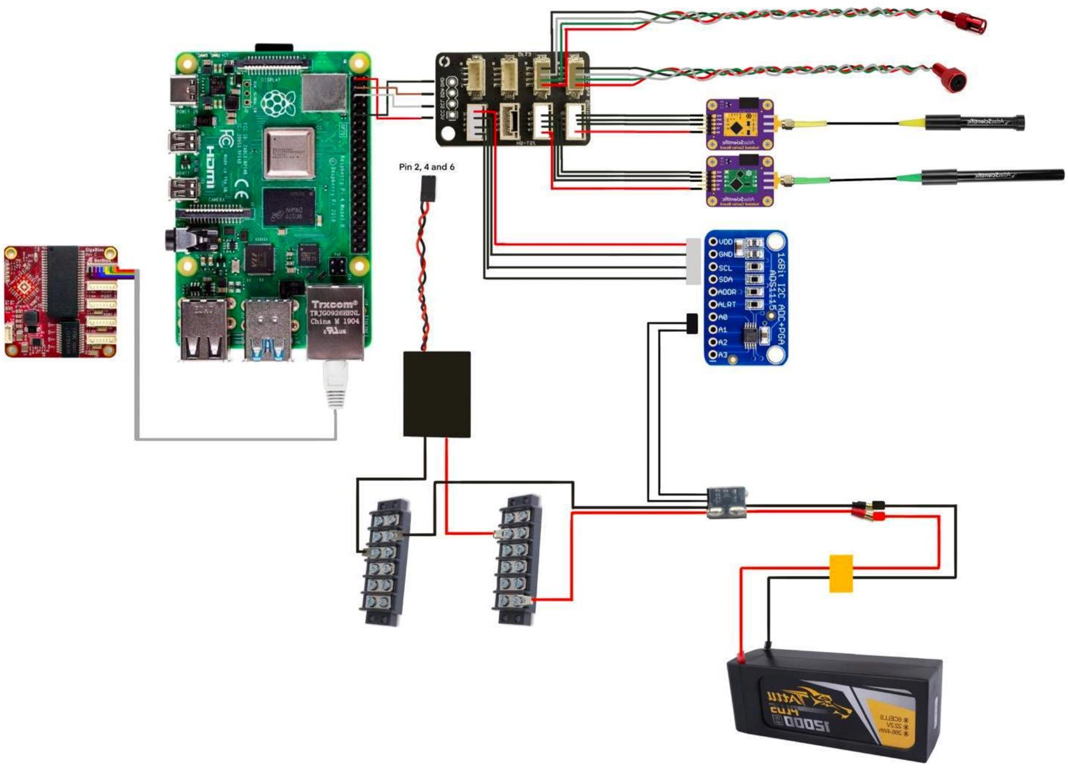

10. Place the raspi-plate over the trackplate (or spring if the heat sink is used), by aligning the holes with the nuts. Then screw the raspi-plate to the trackplate with M3x16mm screws, two on each long side.

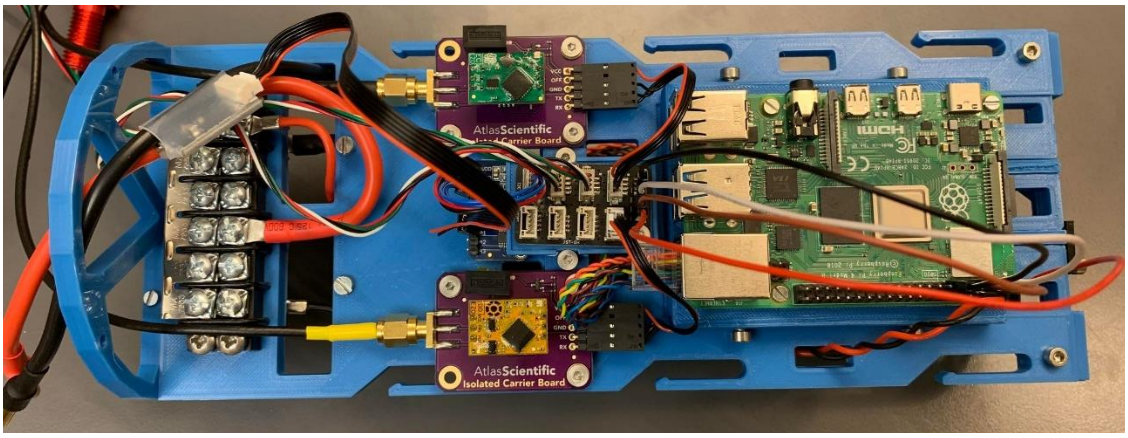


11. Slide the trackplate on the rail on the main plate. Turn the locks to lock the plate.

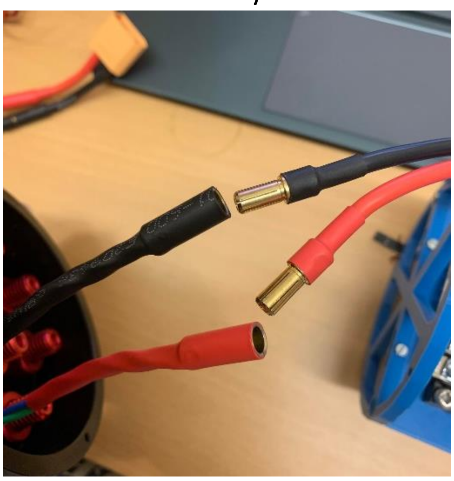
 

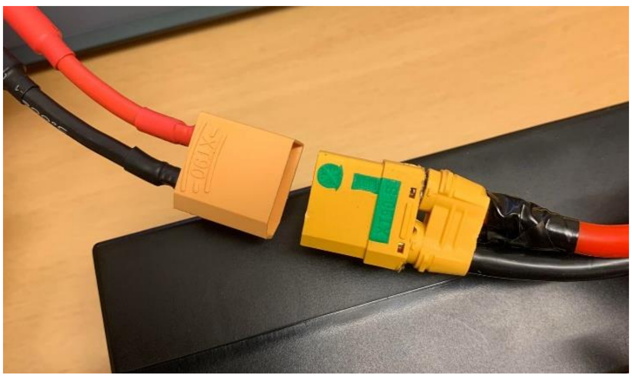
12. Start connecting all cables. They must be connected as shown in the overview picture. Connect the red cable, power, to the barrier strip on the top of the main plate. Then connect the black cable, ground, to the other barrier strip on the opposite side. The last cable going out from the power sensing module is the cable that connects to the ADC. Connect this to the pins A0 and A1. There are also two other cables that are connected to the barrier strips. These are the cables that are connected to the "power supply". Fasten the read cable to the barrier strip on the top and the black one to the barrier strip on the opposite side. Then connect the cables coming from the power supply to the Raspberry pi to pins 2, 4 and 6. In the ethernet port on the raspberry pi, connect a cable going to a connector on the ethernet switch. From the Raspberry pi there also goes cables to the splitter. Connect the cables in this way: pin 1 on the Raspberry pi to VCC on the splitter, pin 2 on the Raspberry pi to SDA on the splitter, pin 3 on the Raspberry pi to SCL on the splitter, and pin 5 on the Raspberry pi to GND on the splitter. In addition to being connected to the Raspberry pi, the splitter is also connected to all the sensors and the ADC. Start connecting cables going from the splitter to the two isolated carrier boards. The red wire in the cable is connected to VCC on the isolated carrier boards. At last, connect cables from the splitter to the ADC. The red wire goes to VDD on the ADC.





13. Connect the battery cables to the battery as shown in the pictures under:





8.5 Project poster

(Next page)

NTNU | Department of Engineering Cybernetics

SENSING RIG

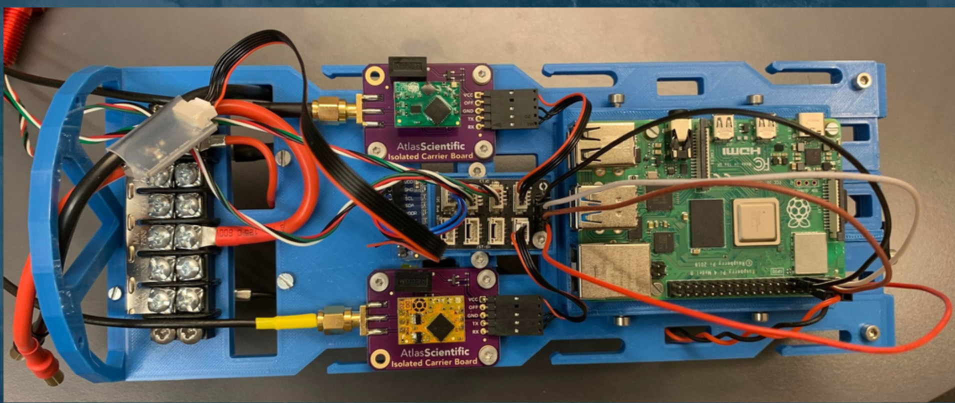
Bachelorgroup E2224: Ida Frøhoel, Jonas Grønskag Johansen and Julie Klingenberg

NTNU, Trondheim | 20.05.2022

BACKGROUND

The project was developing an underwater sensing rig for the IKT, Department of Engineering Cybernetics, that can be used for surveillance of bodies of water. The rig can be attached to the bigger BlueRobotics BlueROV2, or be used as a stand alone unit. The sensing rig can for example be used in fish farms to measure the living condition of the fish, or in smaller lakes and rivers.

The task was to make a fully functional sensing rig equipped with four sensors, battery health monitoring, and the electronics inside the case mounted on a 3D printed layout.



SENSORS

The rig consists of four sensors that measured temperature, pressure, dissolved oxygen and conductivity. The sensors would be connected to a Raspberry Pi 4b through an I2C bus splitter. The code would be created using ROS2, in the operating system Ubuntu from Linux.

BATTERY

The entire system is powered by a 6S 22,2 V lithium ion battery. A corresponding battery code would be implemented to monitor battery health, and the voltage levels across the cells. The code would be created using ROS2, and the values would be sent to the Raspberry Pi through an Analog to Digital Converter.

3D PRINTING

All electronics would be placed inside the cylindrical casing, and it was therefore necessary to make and print a main plate that the other components were attached to. The program used for the modeling was SketchUp, while the program used for the actual printing was Ultimaker Cura.

RESULTS

The project resulted in a rig with sensors that measured values using the Raspberry Pi and a working code. 3 out of 4 sensors worked as intended, while one needed further developments.

There were some problems with the battery along the way, which resulted in an underdeveloped battery health monitoring code.

The hardware layout consisted of a total of 10 3D printed parts. The main part being the main plate on which all the circuit boards were placed. A circle and a semicircle were also designed, and some stands were designed to raise some of the circuit boards.

CONCLUTION

Based on the results, it was concluded that the group achieved a lot through the project.

Although not all parts of the project were completely finished, the result achieved is a good start for further developments in the future.

References

[1] A. Sensors. “Sensor history.” (), [Online]. Available: https://allsensors.com/engineering-resources/sensor-history (visited on 05/19/2022).

[2] J. Merzdorf. “New studies increase confidence in nasa’s measure of earth’s temperature.” (), [Online]. Available: https://climate.nasa.gov/news/2876/new-studies-increase-confidence-in-nasas-measure-of-earths-temperature/ (visited on 05/19/2022).

[3] BlueRobotics. “Bluerov2.” (), [Online]. Available: https://bluerobotics.com/store/rov/bluerov2/ (visited on 05/11/2022).

[4] NathanLec. “Sensors_drivers_brov2.” (), [Online]. Available: https://github.com/NathanLec/Sensors_drivers_BROV2 (visited on 05/19/2022).

[5] Galactic. “Ros 2 documentation.” (), [Online]. Available: https://docs.ros.org/en/galactic/index.html (visited on 05/11/2022).

[6] BlueRobotics. “Electronics tray.” (), [Online]. Available: https://bluerobotics.com/store/watertight-enclosures/4-series/wte4-etray-r1/ (visited on 05/11/2022).

[7] Wikipedia. “Ubuntu (operativsystem).” (), [Online]. Available: https://no.wikipedia.org/wiki/Ubuntu_(operativsystem) (visited on 05/11/2022).

[8] R. Galactic. “Tutorials.” (), [Online]. Available: https://docs.ros.org/en/galactic/Tutorials.html (visited on 05/19/2022).

[9] S. Ekaran. “3d modeling: The basics - easy beginner’s guide.” (), [Online]. Available: https://all3dp.com/2/3d-modeling-basics-simply-explained/ (visited on 04/25/2022).

[10] TheSketchUpEssentials. “Sketchup beginners start here! (desktop version).” (), [Online]. Available: https://www.youtube.com/watch?v=UEcyuvQyVm0&list=PLEQT0qjXe6zgmMMP6ZzoUDG_fdjcRmCQE (visited on 05/11/2022).

[11] 3. Now. “The ultimate beginner’s guide to cura slicer!” (), [Online]. Available: https://www.youtube.com/watch?v=l_wDwySm2YQ (visited on 05/11/2022).

[12] C. Prints. “Ultimaker cura tutorial and basics (beginner’s turtorial).” (), [Online]. Available: https://www.youtube.com/watch?v=NsGOoi3677M (visited on 05/11/2022).

[13] R. Pi. “Raspberry pi 4.” (), [Online]. Available: https://www.raspberrypi.com/products/raspberry-pi-4-model-b/ (visited on 05/19/2022).

[14] BlueRobotics. “Celsius fast-response, ±0.1°c temperature sensor (i2c).” (), [Online]. Available: https://bluerobotics.com/store/sensors-sonars-cameras/sensors/celsius-sensor-r1/ (visited on 05/19/2022).

[15] ——, “Bar30 high-resolution 300m depth/pressure sensor.” (), [Online]. Available: https://bluerobotics.com/store/sensors-sonars-cameras/sensors/bar30-sensor-r1/ (visited on 05/19/2022).

[16] A. Scientific. “Conductivity probe k 1.0.” (), [Online]. Available: https://files.atlas-scientific.com/EC_K_1.0_probe.pdf (visited on 05/13/2022).

[17] T. Gens. "Tattu 12000mah 22.2v 15c 6s1p lipo smart battery pack with as150 + xt150 plug (new version)." (), [Online]. Available: https://www.genstattu.com/tattu-plus-15c-12000mah-6s1p-as150-xt150-plug-lipo-battery.html (visited on 05/18/2022).

[18] S. Campbell. "Basics of the i2c communication protocol." (), [Online]. Available: https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/ (visited on 05/19/2022).

[19] BlueRobotics. "5v 6a power supply." (), [Online]. Available: https://bluerobotics.com/store/comm-control-power/elec-packages/bec-5v6a-r1/ (visited on 05/19/2022).

[20] ——, "Power sense module." (), [Online]. Available: https://bluerobotics.com/store/comm-control-power/elec-packages/psm-asm-r2-rp/ (visited on 05/19/2022).

[21] Adafruit. "Ads1115 16-bit adc - 4 channel with programmable gain amplifier." (), [Online]. Available: https://www.adafruit.com/product/1085 (visited on 05/19/2022).

[22] Wikipedia. "3d modeling." (), [Online]. Available: https://en.wikipedia.org/wiki/3D_modeling (visited on 04/25/2022).

[23] Ji, HyunKyung, *Polygon modeling*, [Accessed 2022-05-20], 2016. [Online]. Available: https://jiharu.github.io/datt2500/week04.html.

[24] L. Ortiz. "The types of 3d modeling - simply explained." (), [Online]. Available: https://all3dp.com/2/types-of-3d-modeling/ (visited on 04/25/2022).

[25] Spatial. "Solid modeling." (), [Online]. Available: https://www.spatial.com/resources/glossary/what-is-solid-modeling (visited on 04/25/2022).

[26] ——, "Surface modeling." (), [Online]. Available: https://www.spatial.com/resources/glossary/what-is-surface-modeling (visited on 04/25/2022).

[27] SketchUp. "Help center." (), [Online]. Available: https://help.sketchup.com/en (visited on 05/20/2022).

[28] SeniorCare2Share. "Quick answer: Will a hollow object 3d print solid from sketchup." (), [Online]. Available: https://www.seniorcareto.com/will-a-hollow-object-3d-print-solid-from-sketchup/.

[29] Sketchup. "Sketchup is [not] a surface modeler." (), [Online]. Available: https://forums.sketchup.com/t/sketchup-is-not-a-surface-modeler/23592 (visited on 04/25/2022).

[30] S. Community. "Why is this not a solid?" (), [Online]. Available: https://forums.sketchup.com/t/why-is-this-not-a-solid/146053 (visited on 04/25/2022).

[31] Wikipedia. "Stl (file format)." (), [Online]. Available: https://en.wikipedia.org/wiki/STL_(file_format) (visited on 04/25/2022).

[32] AlleDP. "The stl file format simply explained." (), [Online]. Available: https://all3dp.com/1/stl-file-format-3d-printing/ (visited on 05/20/2022).

[33] L. Carolo. "What is a 3d slicer? - simply explained." (), [Online]. Available: https://all3dp.com/2/what-is-a-3d-slicer-simply-explained/ (visited on 05/20/2022).

[34] E. Grames. "What is fdm 3d printing? - simply explained." (), [Online]. Available: https://all3dp.com/2/fused-deposition-modeling-fdm-3d-printing-simply-explained/ (visited on 04/25/2022).

[35] Grames, Emmett, *Fdm 3d printing - simply explaned*, [Accessed: 2022-05-20], 2020. [Online]. Available: https://all3dp.com/2/fused-deposition-modeling-fdm-3d-%20printing-simply-explained/.

[36] SIMPLIFY3D. "Pla." (), [Online]. Available: https://www.simplify3d.com/support/materials-guide/pla/ (visited on 05/20/2022).

[37] Wikipedia. "Python (programming language)." (), [Online]. Available: https://en.wikipedia.org/wiki/Python_(programming_language) (visited on 05/11/2022).

[38] JoJoHTM. "Sensors_drivers_brov2." (), [Online]. Available: https://github.com/JoJoHTM/Sensors_drivers_BROV2 (visited on 05/19/2022).

[39] Programiz. "Python inheritance." (), [Online]. Available: https://www.programiz.com/python-programming/inheritance.

[40] R. Galactic. "Writing a simple publisher and subscriber (python)." (), [Online]. Available: https://docs.ros.org/en/galactic/Tutorials/Writing-A-Simple-Py-Publisher-And-Subscriber.html#write-the-subscriber-node (visited on 05/16/2022).

[41] A. Scientific. "Conductivity k 1.0 kit." (), [Online]. Available: https://atlas-scientific.com/kits/conductivity-k-1-0-kit/ (visited on 05/19/2022).

[42] ——, "Ezo-do embedded dissolved oxygen circuit." (), [Online]. Available: https://files.atlas-scientific.com/DO_EZO_Datasheet.pdf (visited on 05/13/2022).

[43] ——, "Ezo-ec embedded conductivity circuit." (), [Online]. Available: https://files.atlas-scientific.com/EC_EZO_Datasheet.pdf (visited on 05/13/2022).

[44] BlueRobotics. "Potted cable penetrator." (), [Online]. Available: https://bluerobotics.com/store/cables-connectors/penetrators/penetrator-vp/ (visited on 05/18/2022).

[45] ——, "Mityvac hand operated vacuum pump." (), [Online]. Available: https://bluerobotics.com/store/watertight-enclosures/enclosure-tools-supplies/vacuum-pump-kit-r2-rp/ (visited on 05/18/2022).

[46] ——, "Bulkhead wrench (r1 style)." (), [Online]. Available: https://bluerobotics.com/store/cables-connectors/tools/tool-penetrator-wrench-r1-rp/ (visited on 05/18/2022).

[47] ——, "Using the vacuum plug and hand pump." (), [Online]. Available: https://bluerobotics.com/learn/using-the-vacuum-test-plug/ (visited on 05/18/2022).

[48] The SketchUp Essentials, *Atomic force microscopy*, [Accessed 2022-05-20], Unknown. [Online]. Available: http://www.thesketchupessentials.com/beginner#LB-tytYDc3YxXWGUygu4oJkmT.

[49] SketchUp. "3d warehouse." (), [Online]. Available: https://3dwarehouse.sketchup.com/ (visited on 05/11/2022).

[50] Raspberrypi, *Raspberry pi 4 mechanical drawing*, [Accessed 2022-05-20], Unknown. [Online]. Available: https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-mechanical-drawing.pdf.

[51] BotBlox, *Raspberry pi 4 mechanical drawing*, [Accessed 2022-05-20], 2021. [Online]. Available: https://cdn.shopify.com/s/files/1/0577/8705/6295/files/GigaBlox_Rev_C.pdf?v=1628628577.

[52] Atlas Scientific, *Eclectically isolated ezo carrier board gen 2*, [Accessed 2022-05-20], Unknown. [Online]. Available: https://files.atlas-scientific.com/electrically-isolated-ezo-carrier-board.pd.

[53] Earl, Bill, *Adafruiy 4-channel adc breakouts*, [Accessed 2022-05-20], 2021. [Online]. Available: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-4-channel-adc-breakouts.pdf.

[54] Molex Industrial Division, *.375" double row low profile barrier terminal strip*, [Accessed 2022-05-20], Unknown. [Online]. Available: https://www.datasheetarchive.com/pdf/download.php?id=f2cd14a6037e91011b6191b5f3cc07c43f69ef%5C&type=P%5C&term=beau%5C%2520terminal%5C%2520block%5C%2520series%5C%252077.

[55] All3Dp. "Cura tutorial: Mater cura slicer settings." (), [Online]. Available: https://all3dp.com/1/cura-tutorial-software-slicer-cura-3d/ (visited on 05/20/2022).

[56] CHEP. "3 cura slicer setting tricks for beginners." (), [Online]. Available: https://www.youtube.com/watch?v=su_m5zV9rvA (visited on 05/11/2022).

[57] T. Instrumens. "Ads111x ultra-small, low-power, i2c-compatible, 860-sps, 16-bit adcs with internal reference, oscillator, and programmable comparator." (), [Online]. Available: https://www.ti.com/document-viewer/ADS1115/datasheet/specifications-sbas4446665#SBAS4446665 (visited on 05/20/2022).

[58] Wikipedia. "Sketchup." (), [Online]. Available: https://no.wikipedia.org/wiki/SketchUp (visited on 05/11/2022).

[59] TOPPTOOLS. "Hulldiamater (innside-tapper)." (), [Online]. Available: https://www.topptools.no/web/side/16 (visited on 05/11/2022).

[60] F. Environmental. "Dissolved oxygen." (), [Online]. Available: https://www.fondriest.com/environmental-measurements/parameters/water-quality/dissolved-oxygen/ (visited on 05/18/2022).

[61] A. Scientific. "Dissolved oxygen kit." (), [Online]. Available: https://atlas-scientific.com/kits/dissolved-oxygen-kit/ (visited on 05/19/2022).

[62] BlueRobotics. "Potting a cable penetrator." (), [Online]. Available: https://bluerobotics.com/learn/potting-a-cable-penetrator/ (visited on 05/18/2022).

NTNU

Norwegian University of
Science and Technology