# Pre-Launch Design Review PLR

**Team: Seraph**

**School Name & City: Premontrei Szent Norbert Highschool, Gödöllő**

**Date: 24/03/2024**

# Table of Contents

# Introduction

## Team members and roles (briefly)

Although most of the tasks were carried out together, each task had a responsible person from the team, who took care of the proper timing and evaluating of the given task, as follows, together with approximate time spent on the task:

- mechanical design, 3D printing: **Fruzsina László** (25 hours)
- recovery system, parachute design, drop tests: **Áron Ferenczy** (15 hours)
- electronics (design, soldering, main board planning): **Máté Magó**, **Pál Tamási** (30 hours)
- programming (sensors, data logging, data storing on SD card): **Hanna Grébel**, **Máté Magó** (40 hours)
- radio communication: **Pál Tamási**, **Máté Magó** (20 hours)
- GPS: **Máté Magó**, **Anna Horel** (5 hours)
- secondary mission (sensors, data analysis): **Anna Horel**, **Pál Tamási** (30 hours)
- outreach, social media: **Hanna Grébel** (10 hours)

**István Seres**, the mentor of the Team, tried to teach the background knowledge needed for creating the CanSat and, for doing and analysing the measurements, for the implementation, etc. Beside that he organised the technical background (3D printing, sensor testing, etc) and the financial background of the project.

## Mission objectives

Our secondary mission has 2 main elements.

Firstly, a $CO_2$ sensor which's data will be useful for seeing the effects of mostly planted areas or even deforested ones, not only near the ground, but in the air too. One of the very important atmospheric parameters from the point of view of environmental protection is the value of the carbon dioxide level. One of our intentions is to numerically verify this with simultaneous $CO_2$ level measurements during the satellite descent.

As carbon dioxide gas is heavier (molar mass is 44 g/mol) than air (average molar mass is 29 g/mol), in the vertical direction its concentration decreases upwards.

We installed an SCD30 precision $CO_2$ sensor to our Cansat, which can determine the $CO_2$ level in every 2 seconds. As it works based on the absorbance of IR radiation in $CO_2$, it uses a special wavelength infrared led and detector, it can measure relatively fast and reliable. The sensor uses an I2C communication port, so its usage is easy. The calibration of the sensor is shown in the Test plan part.

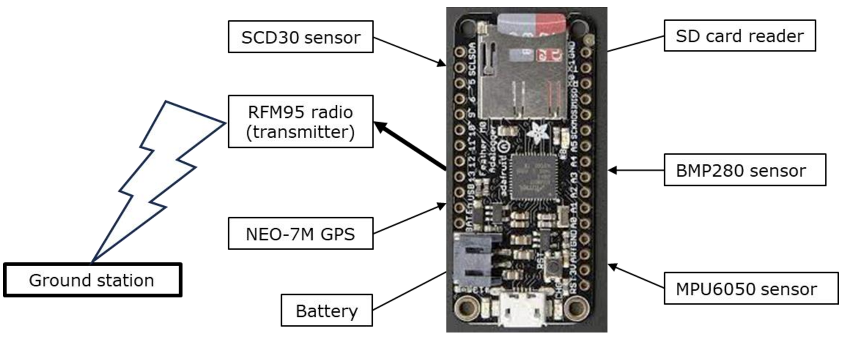
Secondly, our camera which takes place in the bottom of the CanSat and captures a video throughout the landing. This helps us capture multiple pictures, from different altitudes (what we can briefly calculate based on air pressure). When analysing the data after collecting the CanSat we determine the percentile of green areas (trees, bushes, grass), which has an effect on the amount of $CO_2$ level.

The picture analysing is done by an AI, which we made by collecting data from RGB visualizer sites. We set a random RGB colour and determined whether the colour was green or not by eye. We collected 371 samples in total. And after a lot of testing, we calculated that the accuracy score is about 91,9%, but since we deliberately provided the AI with more colours from nature, we expect about 99% accuracy when used in action. The AI, which was trained this way, will be used for the analysis of the pictures made by the Cansat during the flight.

This scientific investigation's goal is to show that the amount of $CO_2$ does correlate to the amount of vegetation on the ground beneath it, and maybe draw the people's attention to stop the deforestation in large quantities and the increasing amount of greenhouse gases in the atmosphere.

# CanSat Description

## Mission outline

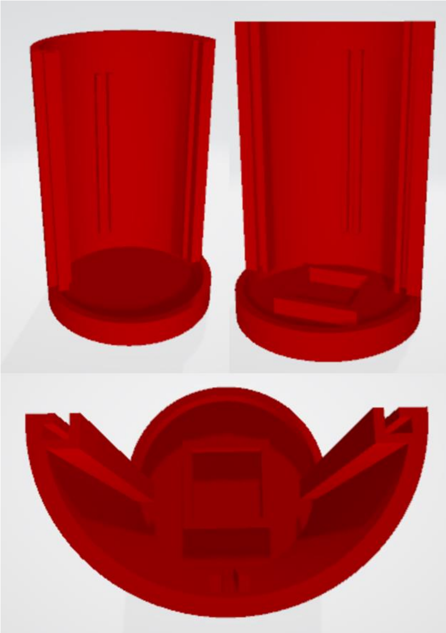



The primary mission element of the CanSat named Seraph is a BMP280 sensor, which measures air pressure and temperature during its flight. The measured data is stored on an SD card and is transmitted to the ground station every second via a radio. The CanSat is equipped with a parachute, which allows it to descend back to the ground from a height of 1 kilometre after being launched with a rocket at a speed of approximately 7 m/s. The device also includes a GPS, which enables tracking its location and helps locate it after landing. The operation of the microcontroller and connected sensors/devices is powered by a single battery that can provide power for at least four hours.

- One of the most critical things in the whole mission is sudden great forces that can ruin things at launch or ejection. It is hard to simulate these conditions, so that is why it is critical. Another critical point is the SD card. It can easily slip out of its case, so the data saving stops immediately, and we cannot use the data later for evaluation.
- Also, the folding of the parachute can be a great challenge because if we fold it incorrectly the CanSat will hit the ground after ejection at a huge speed, not to talk about the accidents it can cause.
- The camera used for our secondary mission has a separate SD card, which is stably fixed in the device. Thanks to this, it will most likely not slip out of it under the influence of huge forces, which is why we can analyse the photos taken after CanSat's return in accordance with the mission with artificial intelligence.

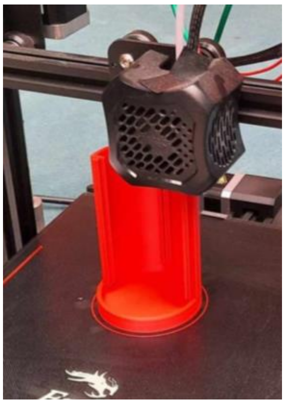
The details of the CanSat can be read below.

## Mechanical/structural design



The casing was designed in Autodesk Fusion360, a 3D design program. It consists of 2 half-cylinders with rails on their sides for precise fitting, allowing the two casings to slide together, making the assembling and adjusting process easy.

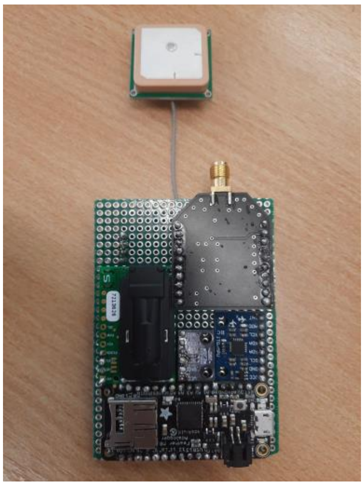
The rims at the bottom and top also aid in this. At the bottom of one of the cylinders a circular hole, 1 centimetre in diameter will be drilled where the camera lens is inserted, so we can take pictures while descending. The camera is located in a 2.34x2.34cm square case at the bottom of the CanSat with a height of 0.58 centimetre. The camera will be securely fixed in its appropriate place with adhesive to prevent it from moving or being damaged by external forces.

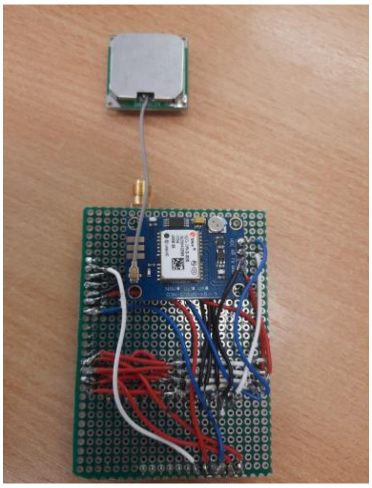


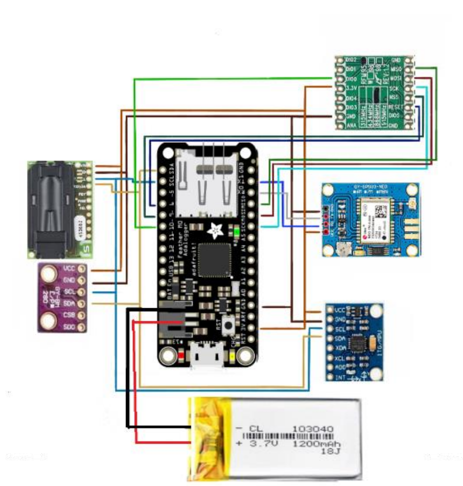
Inside the cylinder, there is a durable structure where the main board will be glued. The thickness of the cylinder, base, and cover is 3 mm, which proved to be thick enough during test drops.

The first version of the case was made out of white PLA, but the final version will be a vibrant red PLA. This will greatly facilitate locating the CanSat after it lands.

The microcontroller used for the mission and the sensors are soldered onto a circuit board. Soldering the connections ensures that the wires are fixed in place, minimising the possibility of them touching or breaking apart during the flight. Thanks to soldering, a stable and reliable connection is guaranteed between the microcontroller and the sensors, which is essential for the successful execution of the mission.







## Electrical design

For a microcontroller we chose the Adafruit Feather M0 Adalogger, as it is, as the name indicates, small and light while also having an SD card reader which makes it simpler to write on it and ensures the safety of the SD card. Its code can also be edited in Arduino IDE which is definitely a positive.

Considering the primary mission, we chose the BMP280 sensor, as it is quite accurate, according to our tests. It was also a positive, that it solves all the different tasks of the primary mission alone.

For the secondary mission we measure acceleration with the MPU6050 and the CO2 levels in the atmosphere with the SCD 30 sensor. Both were tested.

For the green area rate measurement during the flight a mini HD camera is used, with independent power (built in battery and SD card), its type SQ11, its picture can be seen below:

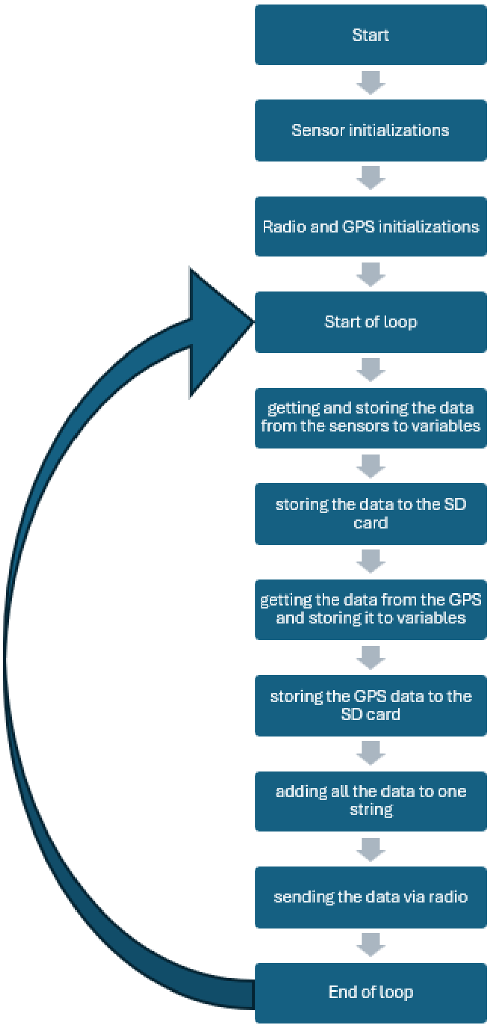

To make retrieving the CanSat easier we also put a Neo 7m sensor in our system. The Neo 7m proved to be almost accurate to the meter, so we have no doubt that we can find the CanSat post landing.
Our data rate of downlink is once every second.

The next table indicates the

| Sensor | Current usage |
|---|---|
| BMP280 | 0,1 mA |
| MPU6050 | 3,9 mA |
| SCD30 | 8,5 mA |
| NEO 7M | 17 mA |
| RFM 95 | 87 mA |
| Adafruit Feather main board | 15 mA |
| SUM: | 131,5mA |

Our battery is a PKCELL LP 503562 1200 mAh, with an estimated 130 mA consumption. It is estimated to last about 10 hours.

## Software design




- The estimated time of the fall will be around 140 seconds, however we will be storing the data from the start of the rocket until we find it, which should be around 4 hours. During this time we will send approximately 1*60*240 radio signals (1 every second), and write the same amount of data onto the SD card. The amount of data, which will be stored is approximately 5760 Kilobytes (50/125 * 240 * 60 = the number of characters per signal / the number of characters per kilobyte * the amount of repetition), which is 5,625 Megabytes. This data can easily be stored using an SD card with 8 Gigabytes of storage space.

- For the editing of the code we are using the Arduino IDE software, which is easy to operate and has all the libraries we need for the sensors.

## Recovery system

The recovery system consists of a parachute attached to our CanSat with a carabiner and ropes. This is made possible by the bolts and holes on the CanSat's body.

Our parachute is roughly the shape of a flat octagon, from now on, I will refer to it as a circle for the sake of simplicity. The radius of it is 15,8 centimetres. We calculated with a pessimistic 1,1 drag coefficient to leave room for error which could come from the differentiating atmospheric pressure.

The parachute also has a hole in the middle to make it more stable and more adjustable. The parachute has to be able to withstand 50N force, so we hung 9 litres of water on it.

The proper size of the parachute was checked by control measurement for the descent speed of the Cansat. This section is introduced in the 3.2.3 Test plan section.

## Ground support equipment

As a ground station, we want to use a laptop and an Adafruit Feather M0 microcontroller connected to the laptop. The same type of radio chip (RFM95W) is connected to the microcontroller (it is the same as the type of radio chip on the satellite). In this way, the data measured on the satellite are sent to the laptop via the radio and the microcontroller, where they are saved and further processed.

At the ground station, the microcontroller really only acts as a data transfer unit between the radio chip and the laptop. The data acquisition and processing itself takes place with the help of programs running on the computer. An obvious and convenient solution is to use the Microsoft Excel program, whose Data Streamer utility allows the microcontroller connected to the laptop to read the measurement data received from the microcontroller (so from the radio) directly. These can even be stored in an Excel table, or even a direct visualisation/graphic creation can be implemented from data. Of course, these data can also be saved directly to the computer for later data processing.

The radio frequency is in the range of 868 MHz, but it can be fine tuned by the software of the Cansat, as the following:

Bejövő adatok innen: Soros USB-eszköz

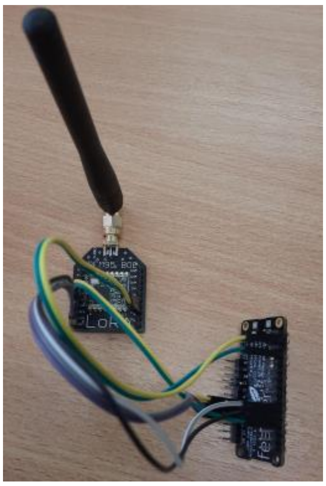
Ebben a szakaszban fognak megjelenni a jelenlegi adatforrásból érkező a

Jelenlegi adatok

| TIME | myTime | Temperature | Pressure | Altitude |
|---|---|---|---|---|
| 15:55:31,90 | 255723 | 27,37 | 100271,8 | 135,68 |

Előzményadatok

| TIME | myTime | Temperature | Pressure | Altitude |
|---|---|---|---|---|
| 15:55:31,90 | 255723 | 27,37 | 100271,8 | 135,68 |
| 15:55:30,90 | 254716 | 27,36 | 100259,93 | 136,67 |
| 15:55:29,89 | 253711 | 27,44 | 100268,67 | 135,94 |
| 15:55:28,89 | 252704 | 27,49 | 100268,46 | 135,96 |
| 15:55:27,88 | 251698 | 27,51 | 100268,43 | 135,96 |



```
#include <SPI.h>
#include <RH_RF95.h>

#define RF95_FREQ 868.0
```

The frequency provided by the jury can be set even on the site if it is necessary.

# Project Planning

## Time schedule of the CanSat preparation

| | 2023.11.15 | 2023.11.29 | 2023.12.13 | 2023.12.27 | 2024.01.10 | 2024.01.24 | 2024.02.07 | 2024.02.21 | 2024.03.20 | 2024.03.24 | 2024.03.25 | 2024.04.01 | 2024.04.04 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Summarising the secondary mission ideas, PDR | | | | | | | | | | | | | |
| Primary mission, data storage on SD card | | | | | | | | | | | | | |
| Secondary mission sensor determination and testing, | | | | | | | | | | | | | |
| 3D planning and printing the Cansat body | | | | | | | | | | | | | |
| Recovery system, test flights, final parachute | | | | | | | | | | | | | |
| Radio communication (RFM69HCW) | | | | | | | | | | | | | |
| GPS positioning | | | | | | | | | | | | | |
| Secondary mission measurements | | | | | | | | | | | | | |
| Ground station development | | | | | | | | | | | | | |
| Radio communication (RFM95W) | | | | | | | | | | | | | |
| Testing of the whole system | | | | | | | | | | | | | |
| Critical Design Report | | | | | | | | | | | | | |
| Further developments and test flights | | | | | | | | | | | | | |
| Pre-Launch Report | | | | | | | | | | | | | |
| Outreach (school presentations, social media) | | | | | | | | | | | | | |

Light green: Working on the task

Dark green: Deadline

## Resource estimation

### Budget

| **Items** | **Price** |
|---|---|
| Adafruit Feather Adalogger M0 | 23€ |
| BMP280 | 1,15€ |
| MPU6050 | 3,2€ |
| RFM95 radio | 18,5€ |
| NEO-7M GPS | 10,25€ |
| SCD30 | 34,6€ |
| 3D printing, tin for soldering, PLA | 10€ |
| Battery (Lipo, 3,7 V, 1200 mAh | 10€ |
| cables, SD card | 6€ |
| material for parachute, carabiner and cabels | 10€ |
| total: | 126,7€ ~ 130 Euro |

### External support

Our two sponsors were our school, Premontrei Highschool and the Hungarian University of Agriculture and Life Sciences, which is located nearby our school. Our school provided budgetary and non budgetary support as well, the budgetary support made it possible to buy the components of our CanSat, while the non-budgetary support consisted of providing a venue for some of our test drops and help to acquire the softwares which are mandatory for the 3D printing. Parallel to this the university offered us a workshop, where we could use their 3D printing machine, test various sensors with the equipment available there, and were able to solder, which is a must when dealing with sensors and microcomputers. We would like to mention Tamás Bálint, the alumni student of our school by name as well, who helped us a lot with the radio communication.

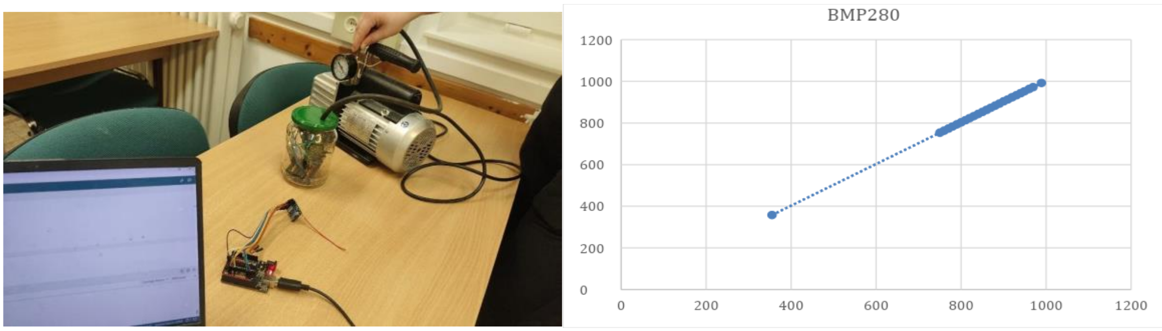
# Testing

## ***Primary mission tests***

First the primary mission sensors were tested. For the primary mission a BMP280 sensor was used, but we were not sure on its exact working, so a calibration measurement was done with a professional air pressure and thermometer unit.

*Air pressure calibration measurement*

The sensor to be verified and the reference measuring device (Greisinger GTD 1100) were placed in a closed space (in a sealed mason jar), then with the help of a vacuum pump we began to slowly reduce the pressure and stopped at certain intervals and recorded the values.

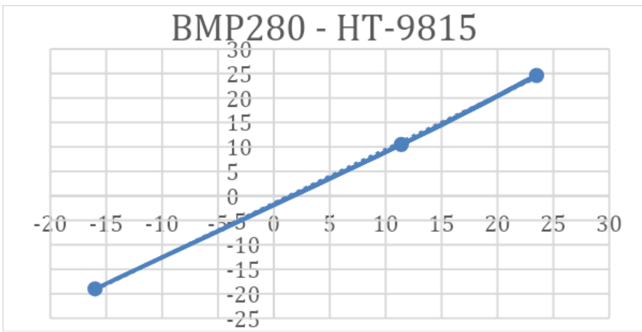



Photos of the pressure sensor calibration and the results

It is very clear from the graph that the pressure sensor calibrated in this way can measure the old pressure value with sufficient accuracy.

*Temperature calibration measurement*

The BMP280 sensor was tested for the temperature measurement accuracy, as well. In this case just a few temperature values were checked from -20°C till 40°C. The -20°C and the 5°C were set by a deep freezer and a fridge. The measured values agreed with the values of the reference thermometer.



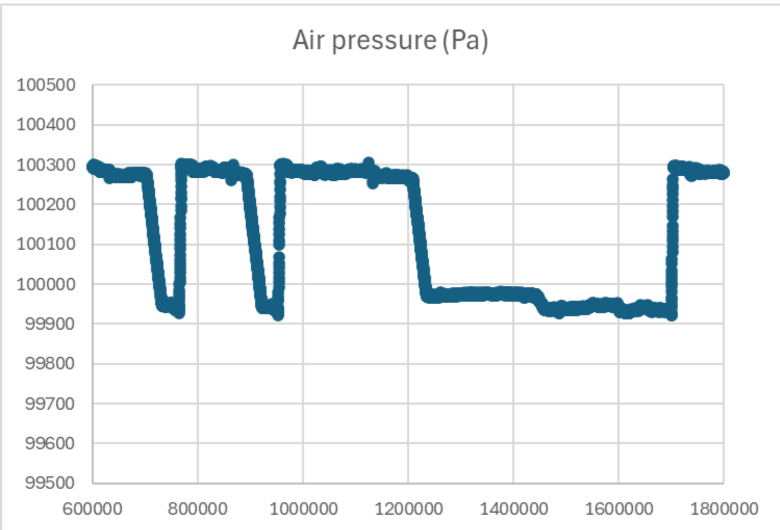
*Test drops*

For testing the sinking speed of the CanSat drop tests were carried out. The first tests were in the Aula of the school (2 floors), in the Aula of the University (4 floors), and finally in a 10 storey high block of flats in Gödöllő Palotakert. During the test drops the timestamps (time data) and the sensor values were recorded to an SD card (that time the radio broadcast was not operating yet). For the sinking speed measurement, the change of the air pressure was analysed - actually the height values calculated from the pressure values. The sinking speed of the Cansat was get as

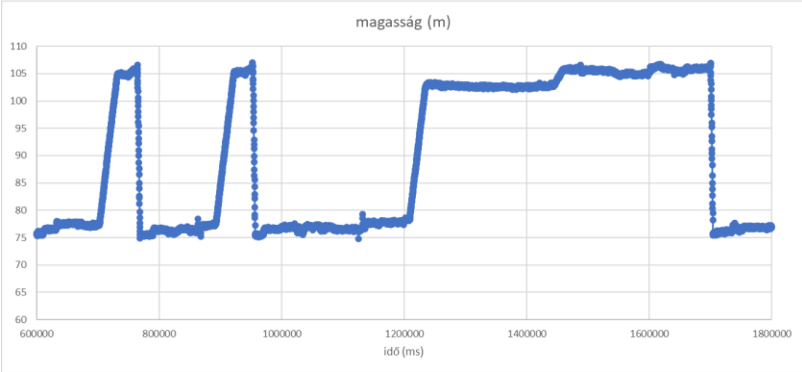
$$v = \frac{h_2 - h_1}{t_2 - t_1}$$

during the drop test h2 is less than h1 (the Cansat is moving down), so the sign of the velocity is negative, but we use its absolute value.

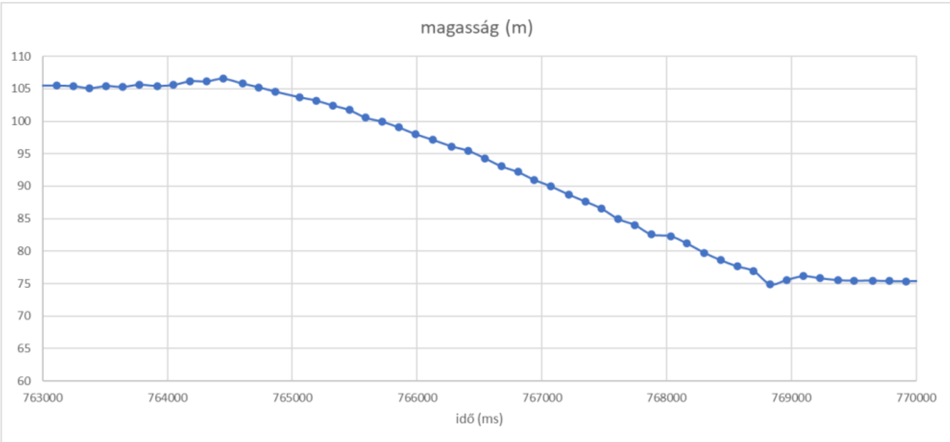
The pressure change as the function of the time (in 0,1 s) during a typical drop test for the 30 m drop is as follows (the horizontal axe is the time in ms):




From this the height change is:



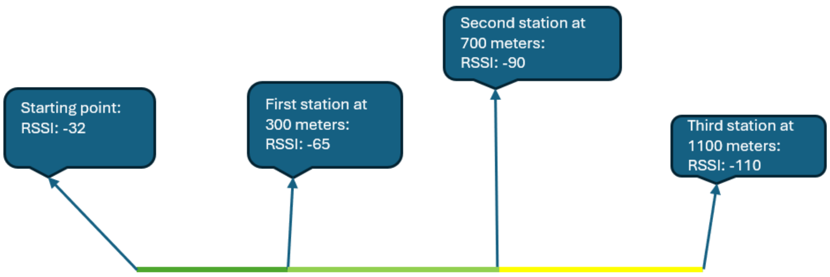

for a typical drop period the height - time function is as follows:




From this function the descend speed is -6,97 m/s ~ -7 m/s.

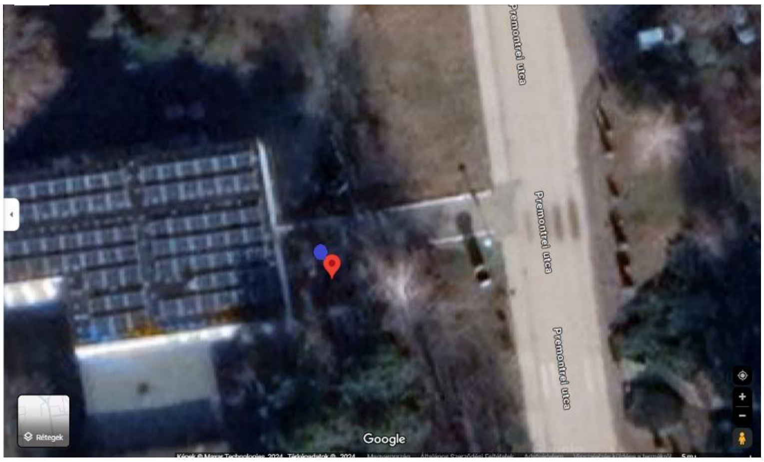
*Radio tests:*

For the radio we are using a RFM95W LoRa radio chip. In the sensor testing we managed to receive data from more that a kilometre away with an RSSI number of -110 dB. Which is good for us, because the radio signal sensitivity is -149 dB, and we will only be approximately a kilometre away from the Cansat.




*GPS tests:*

We are using a Neo 7M GPS, which, proven by our tests, is accurate to the metre, even when we are inside a building.




Red: The coordinates, which the GPS measured

Blue: The position, where we were standing

***Secondary mission tests***

Several tests were performed for the secondary mission, as well, among them calibration of the sensors was done:

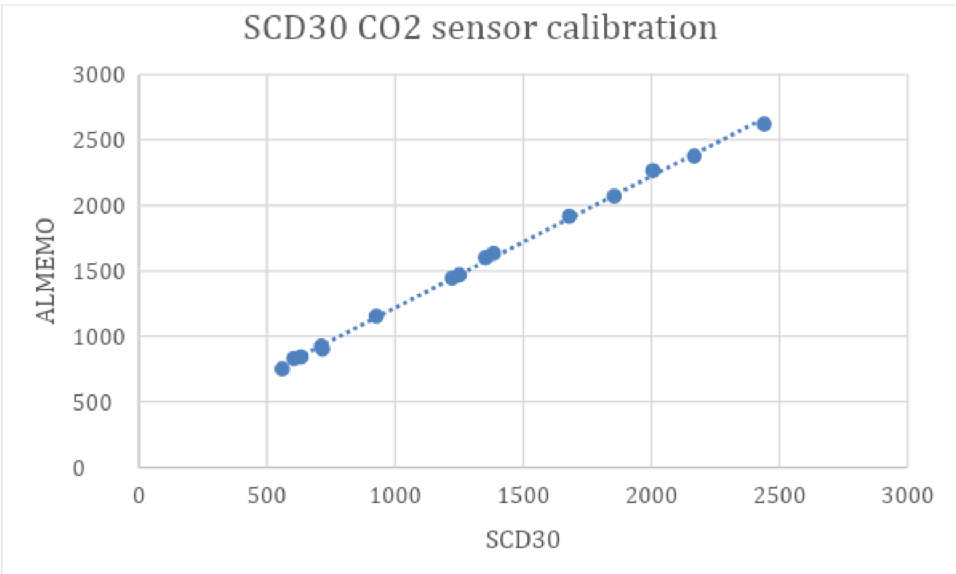
*$CO_2$ sensor (SCD30) calibration*

Several $CO_2$ sensors were tested, e.g. the Adafruit CSS811 air quality sensor, which provides eCO2 (equivalent calculated carbon-dioxide) concentration within a range of 400 to 8192 parts per million (ppm) through I2C interface, but it was not repeatable, and in the lower ranges (below 600 ppm, which will be interesting for us during the flight) it was not measuring correctly.

So we changed for a bit more expensive, but more reliable sensor, for Sensirion SCD30, which can measure the $CO_2$ content with the accuracy of 30 ppm in the range of 400 – 10000 ppm. For the of the sensor a professional unit was borrowed from the Environmental Engineering laboratory of the Hungarian University of Agricultural and Life Sciences. The measuring unit was an ALMEMO 2590 data logger with and authenticated Alhborn $CO_2$ sensor (0 – 10 000 ppm)


Calibration of the SCD30 $CO_2$ sensor

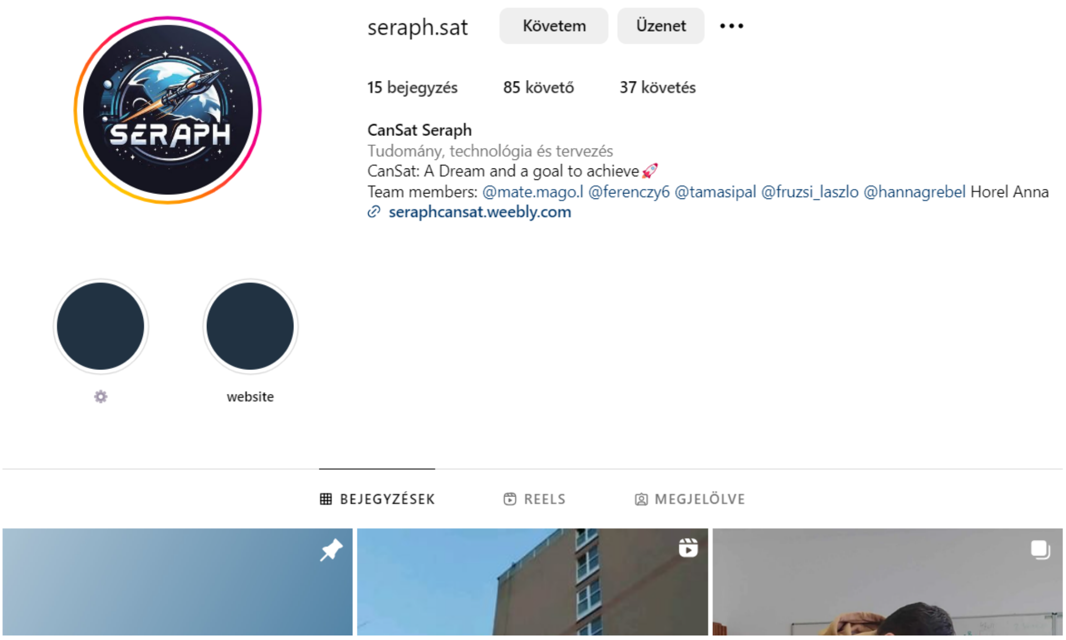
During the calibration measurement, we placed the sensor to be verified and the verifying sensor in a closed space (inside an aquarium), then gradually changed the $CO_2$ level (by breathing) and measured the signal of the two sensors. The result of the beekeeping is shown in the graph below:



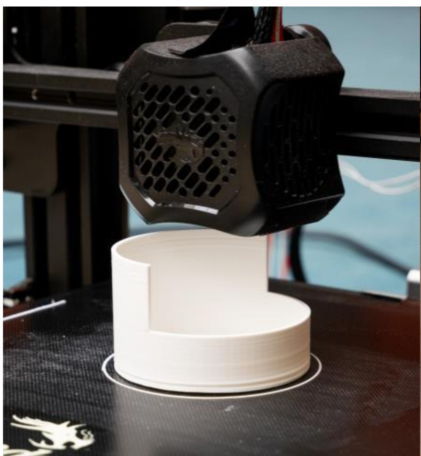

From the linear fit, it can be seen, that the sensitivity of the sensor is perfectly fit to the professional one, but there is an offset of 213 ppm, which can be controlled by the measurement software. So we can conclude that the $CO_2$ sensor can be measured with this sensor properly.

## Outreach programme

We believe that outreach is crucial, so we tried to advertise our project any chance we got. First and foremost, we created a logo for our team to make our instagram page more professional. Then obviously the Instagram page, where we made sure to share all the advancements in the project, whether in the form of a post or a story. One of our short videos even got a bit of traction and got 1000 views.

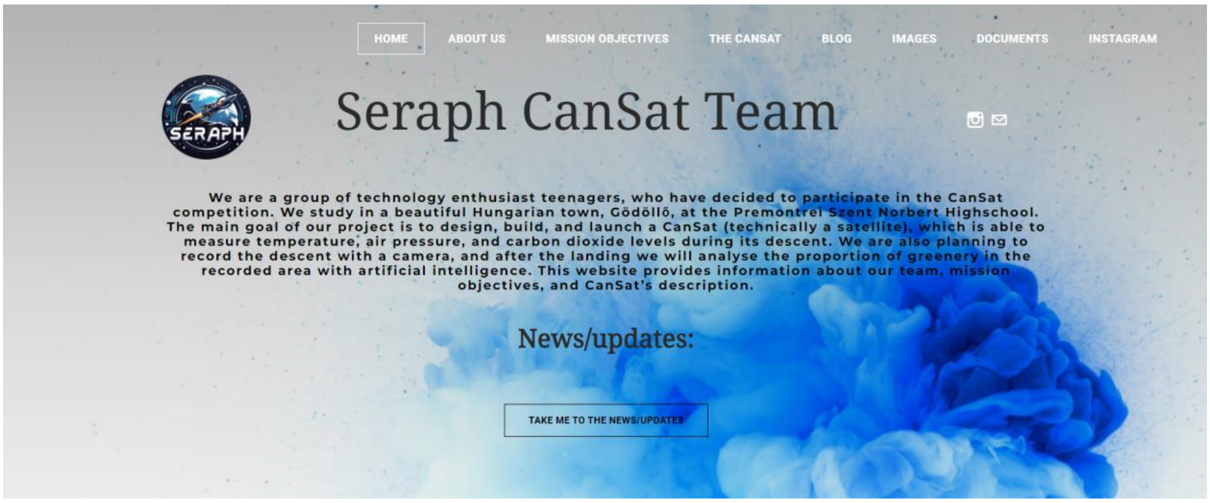



We were also active outside of the cyber world, for example we organised a program in our school for everyone from 5th to 12th grade in order to popularize the project. For example, we placed our 3D printer in the middle of the main hall of our school. According to our calculations about 300-400 of our peers walked by it, therefore the talk spread quickly: "The schools CanSat team has their 3D printer exhibited, let's take a look". For the ones who were more interested we held a presentation which introduced our project with a short explanation of what the project is about and how our CanSat works. Following that the participants had the chance to take a look at the software which made the planning of the components of the body possible.



We also designed a T-shirt for our team:


Besides all that we also made a website, so that those who are more interested can take a look at the project and can have a deeper understanding of our work.




Link to our website: https://seraphcansat.weebly.com/

Link to our Instagram: https://www.instagram.com/seraph.sat/

## Requirements

| Characteristics | Value (units) |
|---|---|
| Height of the CanSat | 110 mm |
| Mass of the CanSat | 305 g |
| Diameter of the CanSat | 66 mm |
| Additional length of external elements (along axial dimension) | 57,5 cm |
| Flight time scheduled | 140 s |
| Calculated descent rate | 7 m/s |
| Radio frequency used | 868 MHz |
| Power consumption | 433 mW |
| Total cost | 130 Euro |

## Power budget

| Device | Voltage (V) | Current (mA) | Power (mW) |
|---|---|---|---|
| Adafruit Feather M0 adalogger | 3,7 V | 15 mA | 50 mW |
| Radio module (RFM95) | 3,3 V | 87 mA | 288 mW |
| Camera | 5 V | independent power supply | |
| Pressure and Temperature sensor BMP280 | 3,3 V | 0,1 mA | 4 mW |
| acceleration sensor MPU6050 | 3,3 V | 3,9 mA | 13 mW |
| Neo 7M GPS | 3,3 V | 17 mA | 56 mW |
| SCD30 | 3,3 V | 8,5 mA | 28 mW |
| **Total power (sum of all)** | **3,3/3,7** | **123 mA** | **433 mW (peak)** |

As the Cansat is supplied with a 3,7 V 1200 mAh LiPo accumulator, the unit can be powered for about 10 hours.

Please include at the end the following statement:

*On behalf of the team, I confirm that our CanSat complies with all the requirements established for the Hungarian CanSat Competition in the official Guidelines,*

*Gödöllő, 24/03/2024*

dr Seres István

*István Seres*
*Team mentor*