

Sound propagation and cosmic radiation measurements with a CanSat satellite

Szabolcs Csabafi, Róbert Dévényi, Mátyás Gerlei

Supervisor: Tibor Herman

October 2022

1 Introduction

CanSat is a competition run by the European Space Agency with the intent of introducing various engineering challenges to students across the globe. The standard CanSat is a device the size (height of 115mm and diameter of 66mm) of a Cola Can, excluding the parachute and GNSS system, with a weight limit in between 300 and 350 grams. The CanSats get launched by a small rocket to an altitude of about 1km, and must reach the ground in two minutes. The CanSat made by each team must perform one or multiple scientific experiments apart from the required ones of temperature, air pressure, and humidity, which must then be presented on various sheets and charts in a web application. The data can either be processed on-board the CanSat or on the computer which receives the raw data. Our two additional experiments will be measuring the power of sound at different altitudes, which will be done with a combination of a speaker and a microphone, and radiation at different altitudes, using a Geiger-Müller tube. The radiation measurements however cannot be done in a free fall, so the parachute system on board will be able to control the speed of descent in order to slow down our CanSat. The system onboard is controlled by a Raspberry Pi Pico with a battery supply. The data will be transferred to another Pi Pico on the ground station, however a backup Micro-SD card slot will be included in order to store all measurements in case the radio system fails.

2 Overview

2.1 Shell

The shell of the CanSat has a size of 115 mm in height and 66 mm in diameter, and only the parachute and an antenna are on its exterior side. We will have it made by a 3D Printer provided by BME and draw it in Onshape, a cloud based CAD software. The 3D Printer will make a light but strong enough shell to endure the falling. It will be split vertically into two parts so that the electronics can be put into it and when done the two parts can be fixed with screws, so the interior can be accessed easily and multiple times. The antenna will be wrapped around it, coming out of a little hole connected to the radio which lies inside of it. The parachute will be on the top of it, and the motor just under it but inside the shell. The exact position of the parts are yet to be decided but probably with the Raspberry Pi Pico in the centre (Figure 1).

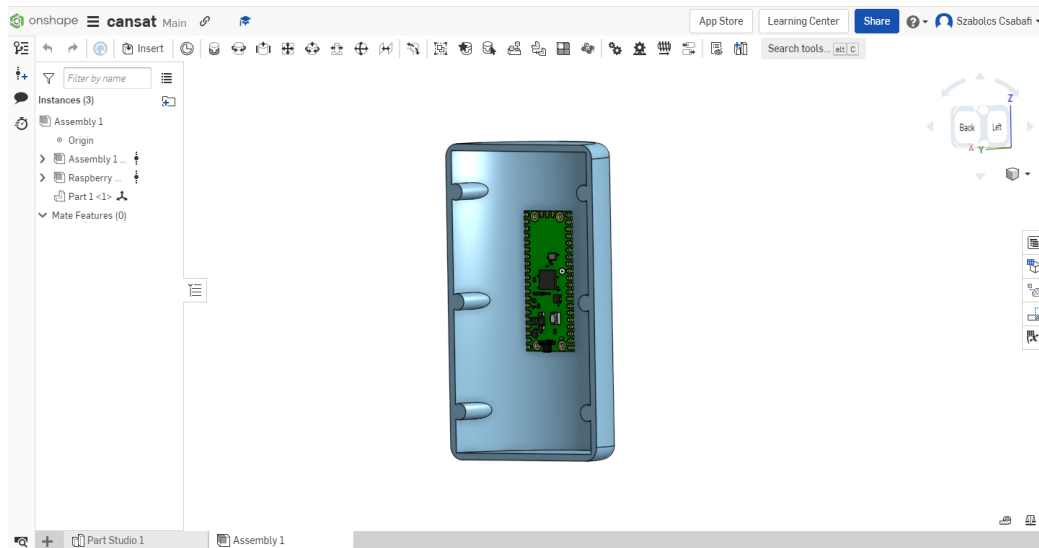


Figure 1: Shell of the CanSat with Raspberry Pi Pico inside of it

2.2 Inside the CanSat

The schematic diagram of the CanSat's hardware can be seen in Figure 2. The heart of the CanSat is the Raspberry Pi Pico microcontroller, whose task is to make the electrical equipments, such as sensors, actuators and other peripherals attached to it work, providing them power and communicating with them to get data. Sensors capable of measuring temperature,

pressure and moisture at the same time, a Geiger-Müller tube, sensing radiation incident on it, and a electet microphone measuring the power of sound, with a speaker close to it, are all on the board. The Pi Pico will be powered by a Lithium-ion battery. In order to make connection with the ground station a radio will be placed on the CanSat, thus real-time data can be obtained and if necessary, commands can be given to the Raspberry Pi Pico and via it anything connected to it. A Micro-SD card also got its place so that if anything happens to the radio connections between the CanSat and the ground station, data would not be lost but stored on it. To be able to evaluate the measurements in respect to the altitude it is required to have the CanSat's coordinates tracked, so a GNSS receiver is necessary to have inside the CanSat. We would like to spend the most time possible at higher altitude because that is where the gained measurements differ the most than at ground level. For that purpose, the parachutes are immediately let loose open, but because there is a given time limit within it the CanSat has to reach the ground, a DC-motor capable of rotating in both direction is connected to the parachute strings. With it we can alter how open the parachute is, thus influence the CanSat falling speed, so after a while the strings of the parachute will be pulled in to reduce the area of the parachute and increase acceleration. Status showing LED's are also in the system to have feedback that everything is working well.

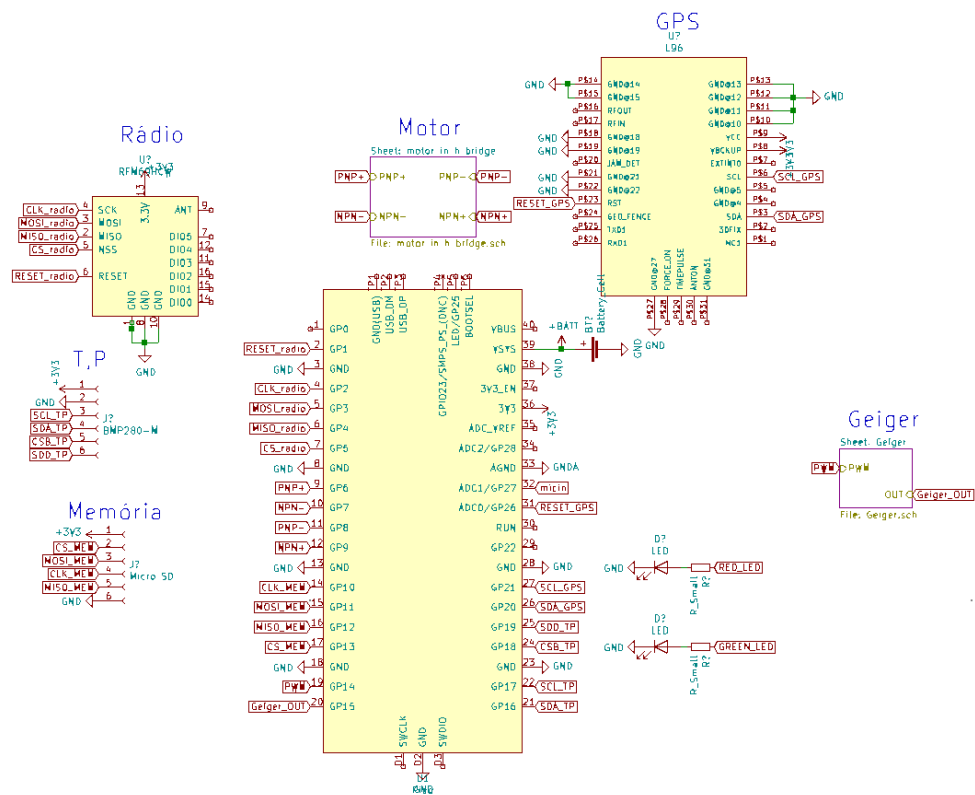


Figure 2: CanSat hardware

3 Experiment 1: Measuring speaker volume with varying altitude

3.1 Theory

Our first experiment in the CanSat competition will be investigating how the volume of a sound varies with altitude. At altitudes 400m-1km, the atmosphere contains less air. Since sound waves are nothing else but the vibration of air as the intermediate medium, there must be a relationship between the quantity of air available and sound volume.

3.2 Preliminary tests

Our experiment will be tested first on ground and only after that will we bring it up to higher altitudes. We will measure the air pressure in an almost perfectly isolated box and generate sounds with our speaker, which will be recorded by our microphone. We will pull the air out of the box with a pump

and investigate the relationship between air pressure and sound volume.

3.3 Method

We will place a speaker and a microphone at a fixed distance from each other, inside a tube to protect isolate these components from outside noises. We will generate sine waves of different frequency from 100Hz to 100kHz in a specified order, periodically. The signal will be measured by the microphone, from which we will compute the power of the signal (sound intensity) and its energy relative to the ground value. The sound from the speaker will be generated using PWM (pulse with modulation). The low pass filter in the schematic attenuates the high frequency components of the PWM signal, leaving the much lower desired signal intact. The following code generates the required sine wave:

```
from machine import Pin, PWM
import time, math
from time import sleep
#from matplotlib import pyplot as plt
pwm0 = PWM(Pin(25))      # create PWM object from a pin
pwm0.duty_u16(0)
u = []
def sin_values(k,fs):
    f = 1000
    for i in range(0, k+1):
        value = math.sin(2*math.pi*f*i/fs)
        adjusted_val = (value+1)*65535/2
        u.append(adjusted_val)
sin_values(2000,80000)
plt.plot(u);plt.show()
```

Figure 3: Python code generating sine wave later used as signal

The output of the code is the following sine wave:

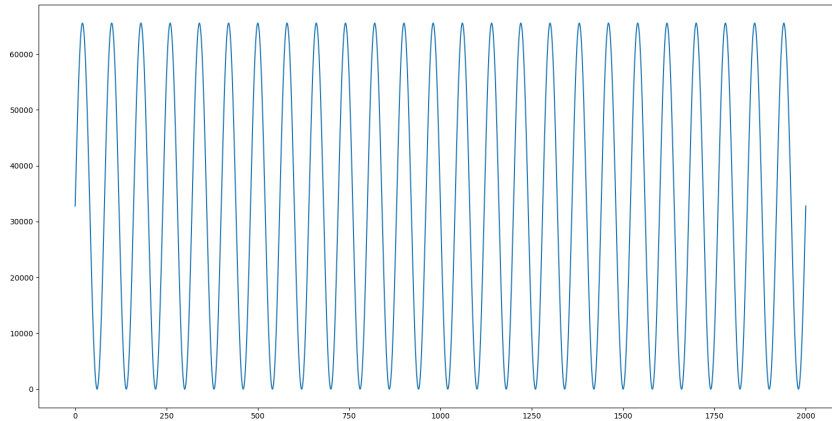


Figure 4: Sampled sine wave signal generated for the PWM

As it can be seen from the code and the graph above, the maximum value of the sine waves is scaled from 0 up to 65535. This is because the duty cycle parameter of the PWM allows to enter values of this range.

3.4 Electronics

The supply voltage provided by the Pi-Pico microcontroller is 3.3V. In order to operate the speaker and the microphone properly, we need to increase the input signal to generate a larger output signal. This can be done by constructing two amplifier circuits connected separately to the speaker and the microphone, with the common power source of 3.3V. Our amplifier circuits contain non-inverting operational amplifiers. The speaker signal's amplifier has unity gain, in order to have the power supplied from the Lithium-ion battery instead of the Pi Pico's GPIO Pin. The microphone amplifier can amplify the signal by a factor of about 100. The amplifier circuit we designed in KiCad are in Figures 5, 6 and 7.

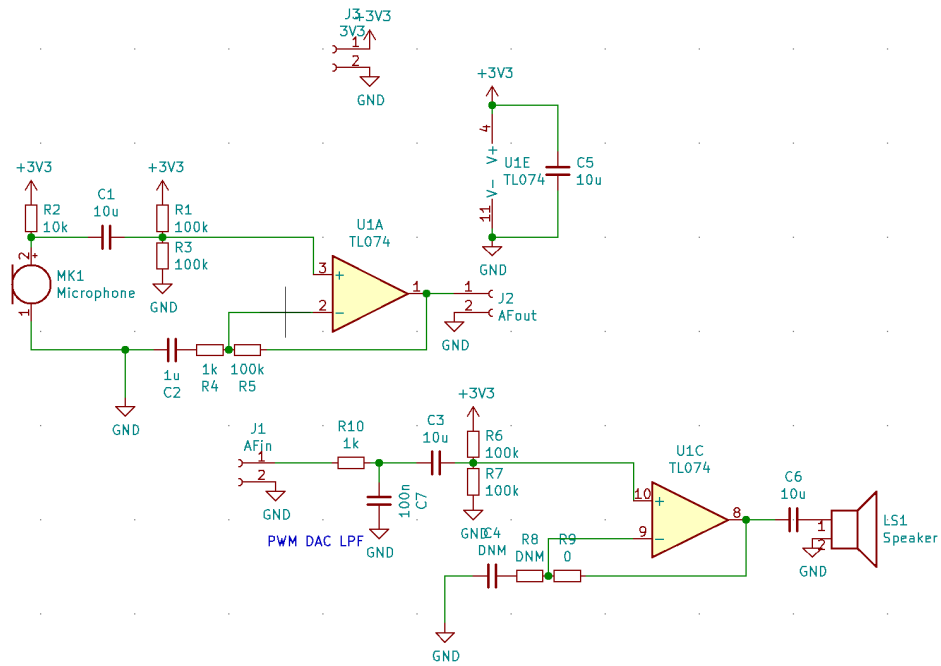


Figure 5: Amplifier circuit

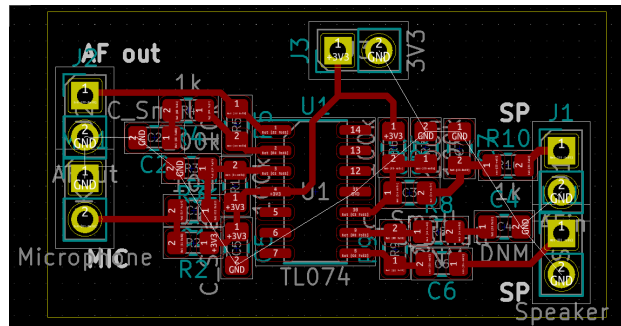


Figure 6: Amplifier circuit layout

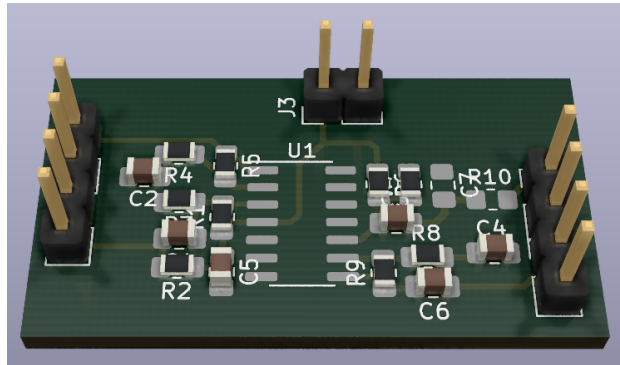


Figure 7: Amplifier circuit 3D rendered picture

4 Experiment 2: measuring radiation at higher altitudes

4.1 Theory

Our second experiment in the CanSat competition will be focusing on how the radiation level varies, if it varies with altitude. Our speculation is that it does, due to the fact that it is more exposed to Sun rays, radiation coming from space.

To test it we will send up a Geiger-Müller counter inside the CanSat to an altitude 1km up.

4.2 Geiger-Müller counter

The Geiger-Müller counter is an electronic instrument used for detecting and measuring ionizing radiation. Its main part is the so-called Geiger-Müller tube, an ionization chamber, where the radiation can be detected. Mostly tubular, but some types are “pancake” shaped, filled with inert gas and a high voltage is applied to it. Depending on what gas is in the system, it can interact with alpha particles, beta particles, gamma rays, x-rays and neutrons. To make it work a circuit has to be made which can provide high voltage, 400-1000 voltage, the manufacturer provides the information exactly how much. The final part is a counter, which can add up the electrical signals coming from the tube.[2]

4.3 Operation

Initially no current flows through the tube, since inert gases are very good insulators. When a high energy particle enters the tube it ionizes the gas, makes it briefly a conductor and to enhance this, so it can be more easily detected, it is amplified by two phenomena. Photoelectric effect and Townsend discharge. A DC current is applied to the tube and the thin wire going through it, in such way that the wire is in the role of an anode and the wall of the tube is the cathode. When the gas is ionized, so electrons and ions are freed and photons are made, photons hitting the cathode causes emission of electrons from it (photoelectric effect), and the freed electrons, ions are accelerated towards the cathode due to the electric field, collide with gas molecules, consequently freeing additional electrons.[4] These processes would sustain themselves so they need to be stopped in order to be able to detect another particle or ray. For that there are two solutions. One is that the voltage applied to the tube is lowered. Its disadvantage that it takes a relatively long time, which would be wasteful. The other is that to the inert gas in the tube other halogen, polyatomic gases are added so they absorbed a part of the “avalanche”. Even with this solution there is a little time out when no other particle can be detected and it lasts for around 10^{-5} seconds. At max it can count 100000 particles, rays incident on it in a second [3].

4.4 Limitations

- The output pulse, the signal from the tube is always the same magnitude, no distinction can be made between radiation types.
- The aforementioned time out or dead time, which happens when detecting one incident radiation, denies the possibility to detect another particle incident on it, during it.

4.5 Electronics

We need to make the required 400 V from the 4.2 V provided by the battery. We use a circuit called a boost converter. This concept is based on the phenomenon that when in a coil the current disappears, for a short time high voltage will appear across it. To make this high voltage somewhat constant we will use the PWM function of the Raspberry Pi Pico. With it we can turn the transistor on and off, thus provide current pulses to the coil very fast. When there is no current the transistor is off, but the high voltage current has to go somewhere so a diode will help out and lead it towards a

capacitor. The capacitor's voltage can not change as fast as it will in the coil so it will keep it high and constant for the tube (Figure 8).

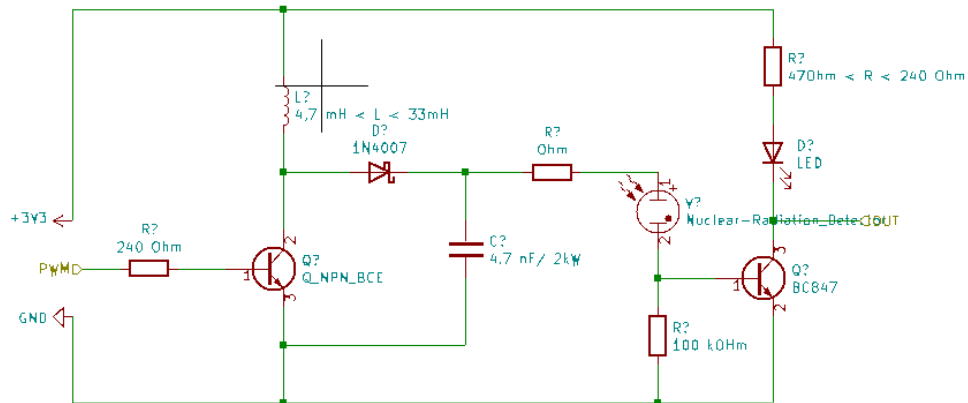


Figure 8: Geiger-Müller tube in circuit with a boost converter

5 Microcontroller

The Raspberry Pi Pico is a microcontroller board. It is cheap, has good availability, high quality and flexible usage with many built-in functions. It is a single-sided 51×21 mm 1 mm thick PCB with a micro-USB port overhanging the top edge and dual castellated/through-hole pins around the remaining edge. Altogether there are 40 pins on the sides. Of the 40, 30 are multi-function General Purpose IO, for short GPIO, including various digital peripherals:

- 2 x UART
- 2 × I^2C (SDA, SCL)
- 2 × SPI (TX, RX, SCK, CS_n)
- 16 × PWM channels
- 1 × Timer with 4 alarms
- 1 × Real Time Counter
- 4 × ADC

It has 2 MB flash memory. A Micro-USB B port can be found on it for powering, transmitting data, programming, debugging. There is an on-board LED on it. It is capable of generating 3.3 V from a wide range of input voltages (1.8 to 5.5V), this allows significant flexibility in powering the unit from various sources such as a Lithium-ion cell. Pi Pico has a Bootsel button on it, and by pushing it down bootloader mode can be activated, which allows programming firmware over the USB port. [1]

We used Micropython, which is a Python 3 implementation for microcontrollers and small embedded systems, for programming in a Linux operation system. We used an Integrated Development Environment called Thonny.

5.1 Features

Pi Pico mainly uses two kinds of communication protocol between its modules, either using SPI or I^2C . In both cases there is a master-slave relationship where Pi Pico always gets to be the master, giving the clock signal and initiating the communication. It can also communicate with another device having UART, since it has it, too. UART is a protocol integrated on a microcontroller, which is not a master-slave communication interface. The Pico is able to generate Pulse Width Modulation (PWM), which is a way to

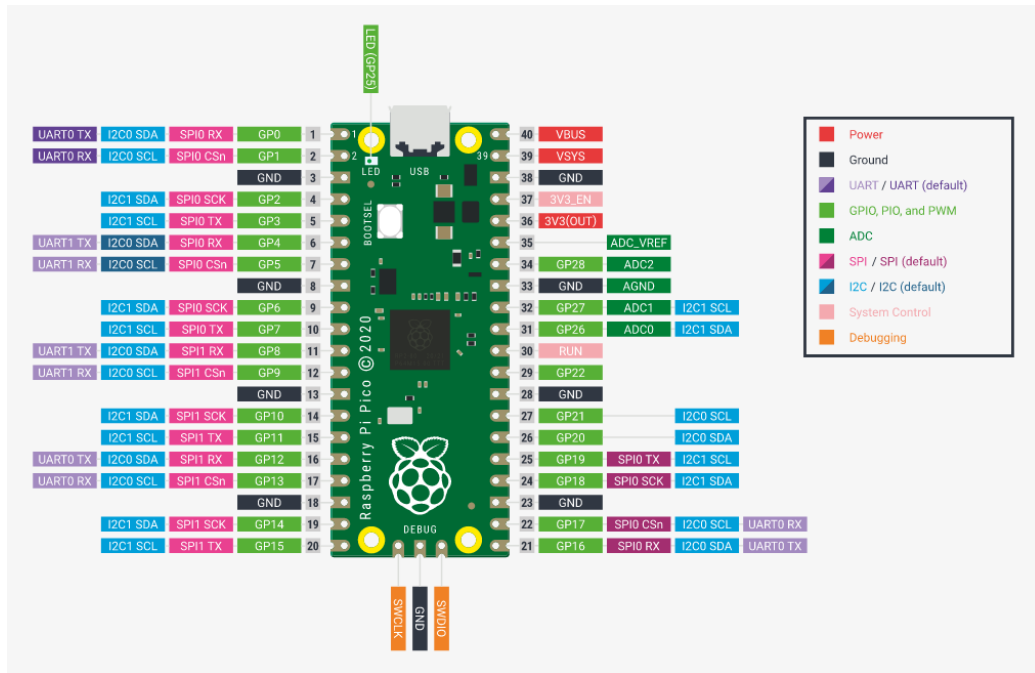


Figure 9: The pinout of the Raspberry Pi Pico

get an artificial analog output on a digital pin. It achieves this by rapidly toggling the pin from low to high. Two parameters can be changed in it, one is the frequency of the toggling the other is duty cycle, which defines how long should the pin stay in high. It also has analogue-to-digital converter (ADC), which can encode an analogue voltage to digital numbers.

6 Power System

The power supply to the CanSat is a single cell Lithium-ion battery. The Pico has a quite robust power scheme, which can provide fixed 3.3 V output voltage that delivers at least 300 mA [1].

7 Parachute system

Since a Geiger-Müller tube measures average clicks per minute, in order to best get average results for radiation with respect to height the Can sat needs to be able to slow down and speed up at different altitudes. To do this, a small dc motor will be able to retract and lengthen a string attached to the center of the parachute (Figure 10), this way it will be able to control the

resistance offered by the parachute and thus increase or decrease its velocity. This is also important due to the fact that there is a time limit to how fast the CanSat must reach the ground after reaching max altitude.

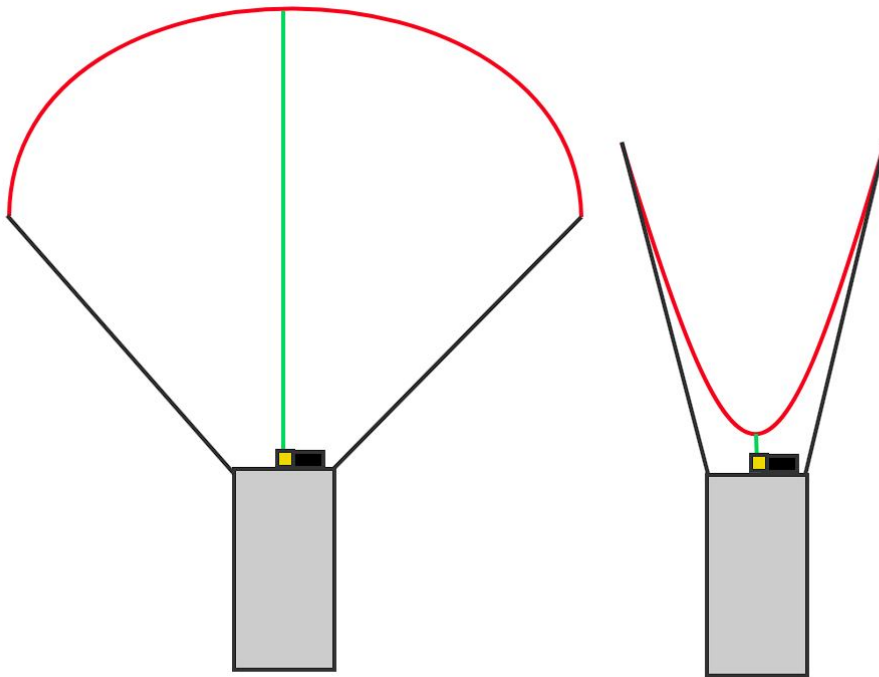


Figure 10: Approximate diagram of the cross section of the parachute including the the parachute control string (green) at maximum resistance (right, fully extended) and at minimum resistance (left, fully pulled in)

7.1 Motor

A small 26x10x12 mm DC motor is used. It works on 3V and can exert 68 mNm torque. To be able to rotate it in both direction we need to change the polarity of the voltage applied to it. We can do this by making a H-bridge.

7.1.1 H-bridge

A H-bridge consists of 4 switches. In our case these 4 switches are going to be transistors, 2 PNP transistors at the top, on the supply voltage side, and 2 NPN transistors at the bottom, on the ground voltage side. When the base of the transistor of the top left and bottom right is forward-biased by

some current, they start to allow current flowing through them thus making a positive current. If we do the same but with the other transistors, we will have a negative current. To prevent the transistors from damage when changing the polarity of the voltage across the motor, we placed 4 diodes parallel with the transistors. The motor will be in the centre of the H-bridge.

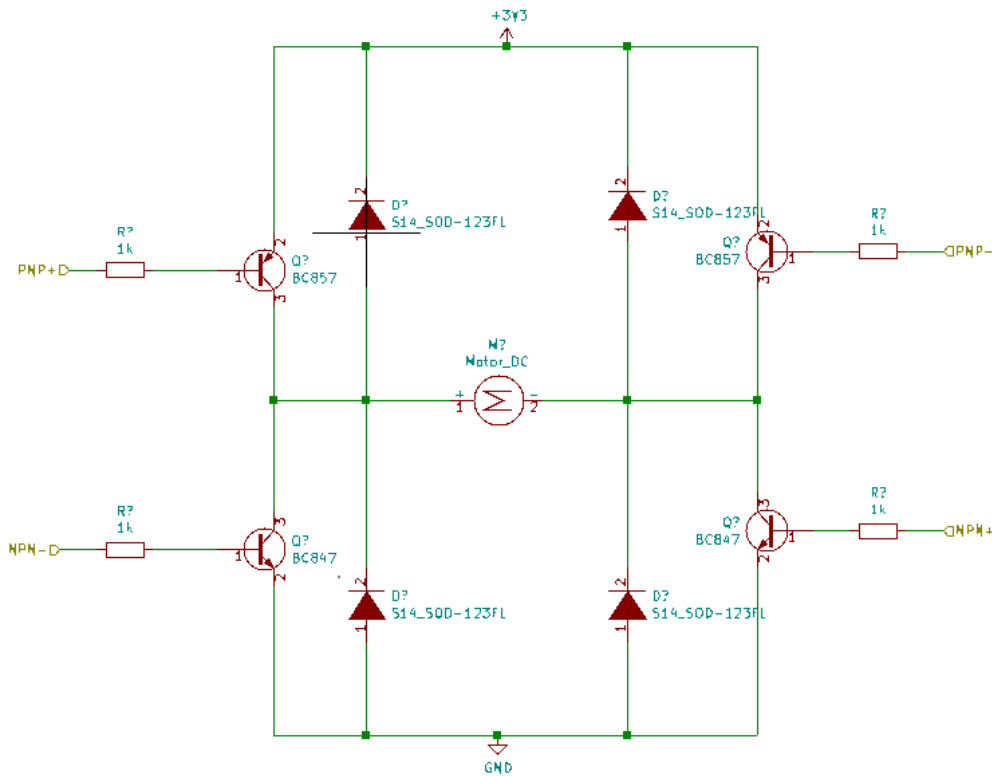


Figure 11: H-bridge with motor in the centre

8 Micro-SD Card Slot

The Micro-SD card module used will serve as a backup to store all the data collected during the flight. The data will be stored in simple text files, from which they can be processed into CSV format. Since the volume of measurements taken over time is not that great, practically any Micro-SD card can function as storage, and even the simplest Micro-SD module can transfer at high enough rates.



Figure 12: Micro-SD Module

9 Communication System

9.1 On-Board Radio

In the CanSat a RFM69HCW will be placed. It is a transceiver (able to do both, transmit and receive messages) module capable of communicating over a wide frequency including 315, 433, 868 and 915MHz ISM (Industry Scientific and Medical) bands. We will be using the 868MHz frequency band. It is a low power, high-performance transceiver, which can work down to -120dBm received signal power at 12kbps and transmit 20dBm (100mW). The module communicates over SPI with the microcontroller. The antenna attached to it will be wrapped around the exterior of the CanSat [5].

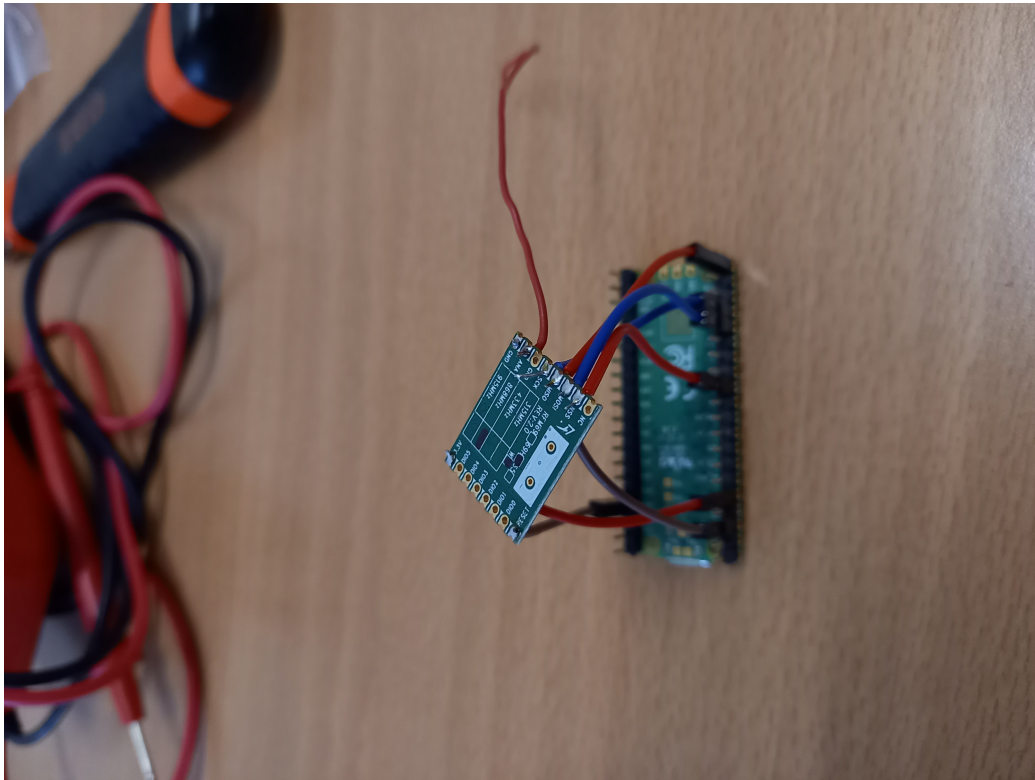


Figure 13: Radio connected to Raspberry Pi Pico

9.2 Coding

We used a library made for a similar radio on github, which with a little alteration was ready to use. Firstly, we needed to setup the SPI communication line, defining which pins were connected, and then we can change

parameters like frequency, deviation, bit-rate or some registers. With using simple modules like transmit, send, receive we can make a communication line.

Here is an example of code from the sender side when we made a connection between RFM69HCW radio and sent messages:

```
1 from adafruit_rfm69 import RFM69
2 from time import sleep
3 from machine import SPI, Pin
4 led = Pin(25, Pin.OUT)
5 radio = RFM69(spi=0, cs=22, reset=2, sck=18, miso=16, mosi=19, frequency=867.95)
6 radio.frequency_deviation = 25000
7 radio.bitrate = 100000
8 counter=0
9 while True:
10     sleep(1)
11     print(radio.send(bytes("message number {}".format(counter), "UTF-8"),
12                       destination=0x55,
13                       node=0x56,
14                       identifier=0x57,
15                       flags=0x58))
16
17     led.toggle()
18     counter+=1
19
```

Figure 14: Example code for radio

References

- [1] Rapsberry Pi Pico datasheet <https://rptl.io/pico-datasheet> Accessed November 1, 2022
- [2] Geiger counter wikipedia https://en.wikipedia.org/wiki/Geiger_counter Accessed November 1, 2022
- [3] Geiger-Müller-számláló <https://hu.wikipedia.org/wiki/Geiger%E2%80%93sz%C3%A1ml%C3%A1l%C3%B3> Accessed November 1, 2022
- [4] Townsend discharge https://en.wikipedia.org/wiki/Townsend_discharge Accessed November 1, 2022
- [5] RFM69HCW datasheet <https://cdn.sparkfun.com/datasheets/Wireless/General/RFM69HCW-V1.1.pdf> Accessed November 1, 2022