Preliminary Design Report

Team Name:

Jeźdźcy Jarzynowej A.K.A. JJ CanSat Team

Country:

Poland

Table of Contents

1 INTRODUCTION

1.1 Team organisation and roles

Patryk Gałczyński

Project manager, supervisor

Patryk is a third year Computer Science student at the AGH University of Science and Technology. He founded and runs RoboTeam faculty in our high school, where we know him from. He made couple autonomous mobile robots called mini-sumo and line-followers. He works as PaaS layer cloud platform developer in Dreamlab, which means lots of DevOps challenges and writing tons of code in Python (which is his favorite programming language these days).

Tasks:

Patryk is acting as our project manager, and supervisor. That means, he provides us with tools to work efficiently (like trello or git repository), organizes our work by planning meetings, and pushing us to work hard every day. He also is our inexhaustible source of knowledge about electronics, unix systems and software engineering.

Karol Oleszek

Team leader, ground station software

Interests:

Programming:

development of GUI applications (Delphi, C#, C++)

simple games (Delphi, Unity 5)

web solutions (PHP, JavaScript, MySql, HTML5, CSS3) in e-commerce (school's book sale) and web games (simulation of aerospace industry);

Physics:

participated in “International Masterclasses Hands on Particle Physics” workshop organized by CERN;

Economy and social studies:

laureate in Voivodeship Contest of Economy and Social Sciences Competition for Middle School Students, earned two times;

Tasks:

As a leader, Karol binds the group together, distributes tasks and makes sure they are completed on time. He also takes part in programming the base station for our CanSat, documentation and communication with our sponsors.

Jan Kostecki

Electronics, receiver software, construction

Interests:

Robotics:

participated in several robotics competitions,

member of school RoboTeam;

Programming:

programming in C++, SQL, Java, Arduino IDE and NXC;

interested in development of web solutions (PHP, JavaScript, MySql, HTML5, CSS3) in e-commerce (school book sale, together with Karol Oleszek) and web games (simulation of aerospace industry with Karol), attends mobile applications programming course;

Physics:

laureate in Voivodeship Contest of Physics for Middle School Students,

participated in "International Masterclasses Hands on Particle Physics" workshop organized by CERN;

Tasks:

Jan designs electronic and mechanical solutions for our CanSat, such as its wind speed sensor, as well as its radio receiver software.

Bartłomiej Biesiadecki

Construction, electronics

Interests:

Robotics:

took part in Nationwide High School Robotics Contest “µBot” (together with Jan Kostecki),

member of school RoboTeam;

Programming:

programs in C++, SQL,

attends mobile applications programming course (Swift);

Physics:

participated in “International Masterclasses Hands on Particle Physics” workshop organized by CERN;

Economy

took part in Expo FXCUFFS 2016 Investment Congress;

Tasks:

Bartłomiej is mostly involved in developing mechanical solutions, such as the parachute, and the overall process of landing. He also helps Jan Kostecki in development of electronic solutions.

Tymoteusz Chmielecki

On-board software

Interests:

Programming:

programming in C, C#, C++, Python3, HTML5, Java, JavaScript, Arduino IDE;

Aeronautics:

in his free time builds and pilots quadcopters;

Participant of the Lesser Poland's Scholarship Programme for Especially Talented Students for his high grades;

Tasks:

Tymoteusz writes code for onboard electronics and integrates internal solutions with ground station software.

Jan Dziedzic

Physics calculations

Interests:

Programming:

programming in C++, Bash, Python3, Arduino IDE;

laureate in Voivodeship Contest of IT for Middle School Students,

laureate in Theoretical Computer Science Olympics for Middle School Students (individual & team)

Astronomy:

Polish representative on ESO Astronomy Camp 2015,

completed total of 4 weeks of internships in Astronomical Observatory of University of Warsaw,

built his own roll-off roof remote astronomical observatory;

Physics:

laureate in Voivodeship Contest of Physics for Middle School Students,

participated in "International Masterclasses Hands on Particle Physics" workshop organized by CERN;

Participant of the The Polish Children's Fund Programme for his achievements in numerous competitions of IT, Maths and Physics;

Tasks:

Jan takes care of most of the calculations needed for the project, his role will be most important during the process of post-launch data analysis. He is supervising all calculations done in the group. He also prepares mock data to test live data analysis software.

Photo 1, The team

1.2 Mission objectives

1.2.1 Secondary mission

The secondary mission of our CanSat is to investigate landing area and using the gathered data, decide whether the place is suitable for landing of a heavy lander. The mission goal is to simulate a cheap and accurate way of preparing planetary landing of heavier equipment.

We have chosen this job for our device, because it might be used in numerous situations, such as preparation of Mars landing (after adaptation of the probe's parachute) or investigating remote location by dropping the probe from helicopter or plane.

1.2.2 Measurements and tests

Gathering and processing information about conditions and their impact on the landing of a heavy lander in the studied landing area includes:

1. Measuring the speed of wind and analysis of its effect on lander descent,
2. Taking pictures of the surface in order to assess the landing site,
3. GPS tracking of probe's flight,
4. Saving data to onboard storage for further analysis.

1.2.3 Objectives

We will consider CanSat launch successful when the following objectives are completed:

1. Proper transmission of gathered data to the ground station,
2. Online assessment of CanSat’s measurements with the priority of correct determination whether analised site is suitable for heavier lander,
3. CanSat’s safe landing.

1.2.4 Expected results

We expect that competition area will be suitable for heavy lander, but we consider an option that the weather may change the result. If wind is too fast or atmospherical conditions are somehow concerning, we expect that ground analysis will result in negative information about area’s suitability for heavy landing.

2 CANSAT DESCRIPTION

2.1 Mission overview

2.1.1 Overview

Design, manufacture, assembly and testing

CanSat's flight is preceded by long preparations including design, prototyping and numerous tests. Project concept is described in this document, but with further testing some specification may change.

Before takeoff, CanSat is prepared with packing parachute and charging batteries.

2.1.2 Take-off

Before the flight CanSat will stay in standby mode which decreases energy demand, ensuring it can work for three continuous hours. After rocket launch / deployment from the plane, all systems are going to be turned on and the packed parachute will unfold.

2.1.3 Flight

CanSat is about to descend with velocity about 9 m/s (no faster than 11 m/s). During the flight, numerous measurements are conducted (wind speed, atmospheric pressure, air temperature, acceleration) and sent back to the ground station, where they are analyzed and results are being displayed and published online. GPS position is tracked, altitude above the surface calculated and photos of terrain are taken.

2.1.4 Landing

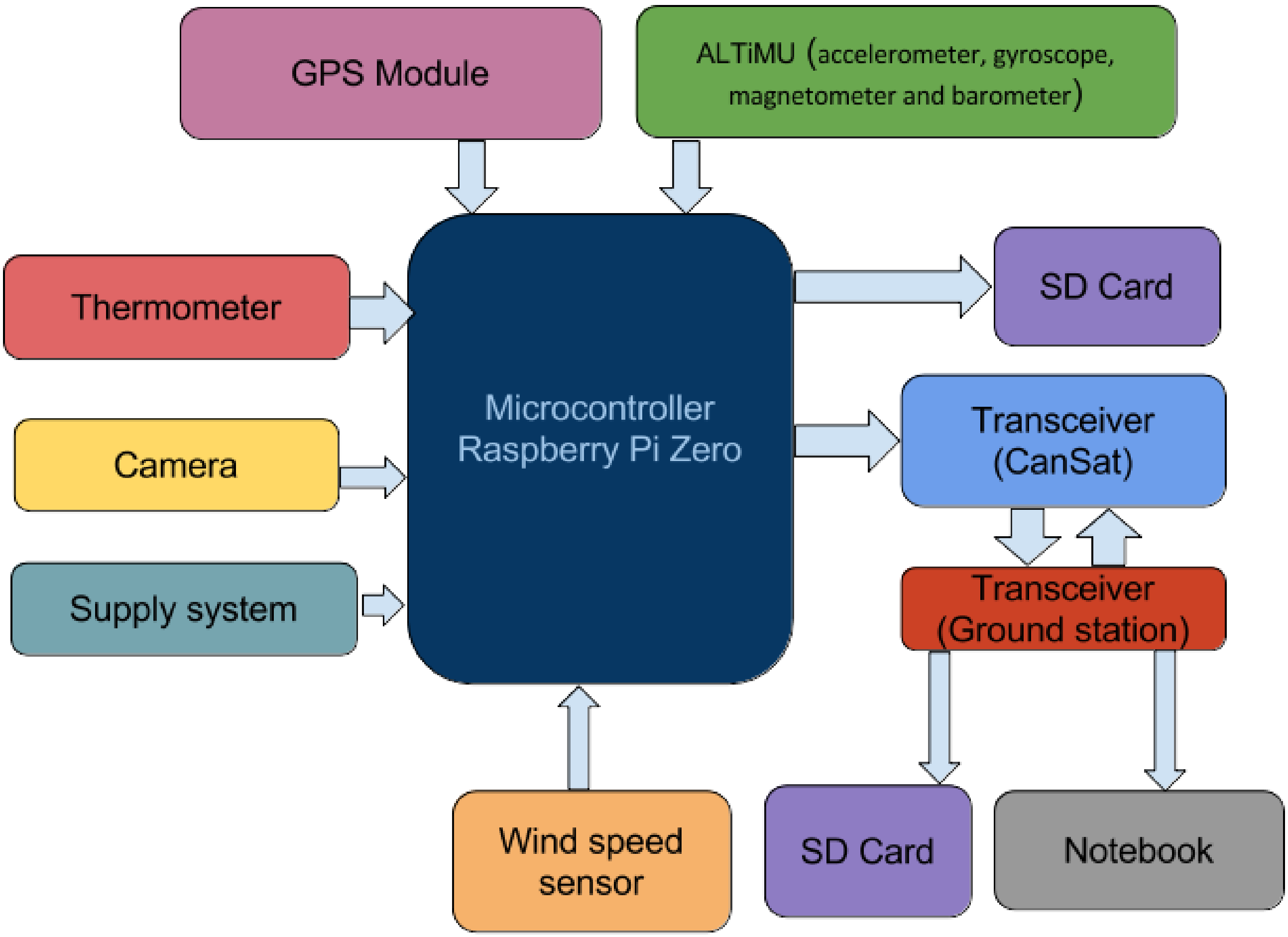
Although CanSat is descending with a parachute, special mechanical construction is being applied to ensure that it will withstand touch-down with velocity exceeding desired one in case of partial parachute malfunction.

Landing system is still being developed, due to future tests this conception might be changed.

2.1.5 Key elements

CanSat electronics includes various sensors, camera, GPS with additional antenna module, radio transmitter, it's antenna and mechanical elements - parachute, wind speed sensor and casing.

2.1.6 Block diagram of electronic components



2.2 Diagram 1, electronic componentsMechanical/ structural design

2.2.1 Ground Station

Ground Station consists of two main elements: the receiver and a computer (described below):

Receiver - based on an Arduino module - includes transceiver (to receive data), YAGI antenna (to amplify signal), microSD module (to save received data) and barometer (for reference while computing altitude from atmospheric pressure). Received data is going to be saved on microSD (backup purposes) card and send to the computer via USB or COM port.

2.2.2 Mechanical construction

All electronic elements will be soldered to the PCB (batteries holder too), perhaps we will use multiple boards. Empty space inside will be filled with styrofoam balls for additional cussion.

Casing will be probably made of carbon fiber or other high-tech material.

Component	Weight [grams]
Supply system	~100 (47.5g each battery)
Casing	~20
Raspberry Pi Zero	9
GPS	3
ALTiMU	1
Camera	3
DC-DC converter	1
Transceiver	5
GPS Antenna	20

Table 1, components weight

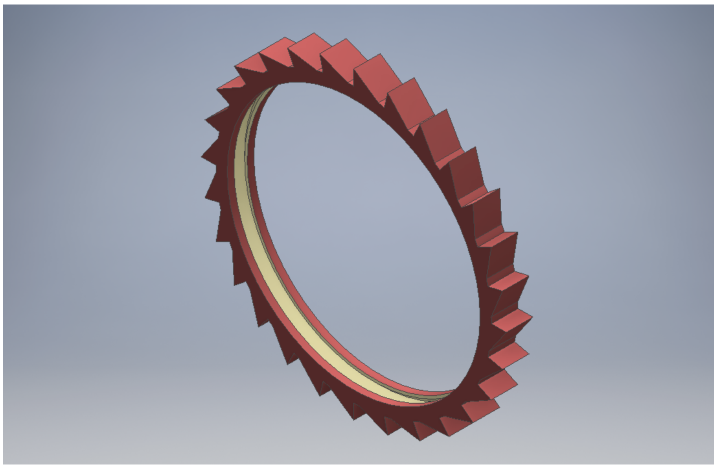
Weight of our CanSat might increase in the future, but it won't exceed the limit of 350g.

2.2.3 The wind speed sensor

During the landing, information about speed of the wind might be useful. In the Internet, there are many wind sensors to buy, but their mechanical structure doesn't fit our project - they are too big and don't withstand big g-forces. Therefore we decided to develop our own.

Our sensor is going to be a "ring" driven by the wind. Inside of the CanSat, optical encoder will collect data about its rotation. The central computer (Raspberry Pi Zero), based on this information, is going to estimate the wind speed.

This idea is still in a heavy development, we have yet created only the first prototype (screenshot from Autodesk Inventor Studio) and we printed it on the 3D printer (photo below).



Picture 1, wind speed sensor concept

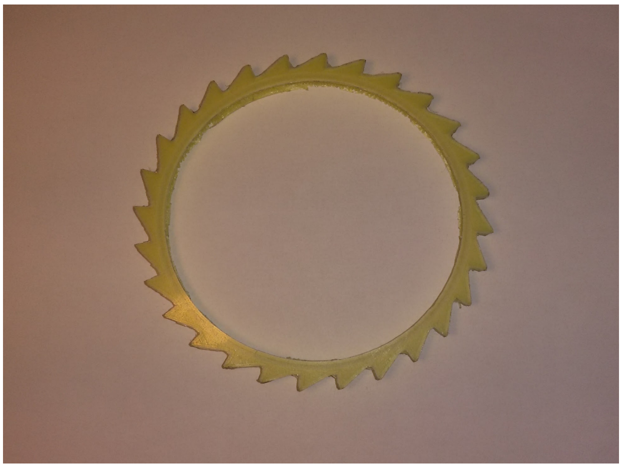


Photo 2, wind speed sensor 3d printed prototype

Our next step is to find the best shape of the "ring" and the method of fixing it to the bearing.

As the method of measuring "ring" angular velocity we have opt for an optical encoder. We consider it the best solution because it greatly decreases need to use other mechanical elements. Inside, the "ring" will be painted in black and white stripes consecutively (change of the reflection factor means rotation by known angle).

2.3 Electrical design

2.3.1 Electronic elements

The following list includes electronics elements which are going to be inside of our CanSat.
(Detailed parts names/numbers with links can be found in Attachment 1 at the end of this report)

- Raspberry Pi Zero - main computer, collects data from sensors, passes them to the transceiver and save on the microSD card,
- Thermometer - measuring the temperature,
- Optical encoder - measuring the angular velocity of the wind sensor,
- Pololu AltIMU - includes accelerometer, gyroscope, magnetometer and barometer. Acceleration measurement will be used in the algorithm of noise reduction from the wind speed sensor,
- GPS module with additional antenna - to locate CanSat after landing and pinpoint place of the heavy landing, as well as measure horizontal speed of the probe,
- Raspberry Pi Zero Camera - we are planning to take pictures of the terrain from big height, camera is going to be fixed on the bottom of the CanSat,
- Li-Ion batteries - power supply system,
- DC-DC converter 3.3V - part of the supply system, Raspberry Pi Zero and others electronic elements require 3.3V to operate,
- Power conditioning system - system including resistors, capacitors and diodes to make sure the current is stable,
- Transceiver - to send data to the ground station.

We are also planning to include a charging module extracted from a powerbank - it is capable of charging 2 cells at the same time.
As PCB design may vary greatly with even a slightest change in used hardware we are going to put PCB layouts and schematics in further reports, as we establish what exact hardware is going to be used.

2.3.2 Battery & power supply

After long research of the best power supply we decided to use Li-Ion cells.
We noticed that the same type of batteries is popular in powerbanks. First we wanted to use just the whole system of powerbank (cells and electronics), but now we have figured that better solution will be to use separate Li-Ion cells.
Batteries used in our design are four type 18650 Panasonic NCR-18650B Li-Ion cells with nominal capacity of no less than 3250mAh each and nominal voltage of 3.6V. One cell should be sufficient, but extreme conditions may affect battery capacity, therefore we are taking two onboard.
Our estimated maximum flow current has been rated at no more than 1A and we've decided to use components powered with 3.3V DC. Therefore our choice was to use two step-up/step-down converters each rated at no less than 750mA of output current in a pessimistic case of batteries hitting their minimum voltage which is rated at 2.5V. Following table represents estimated power usage in a condition of maximum performance.

Component	Amperage rating [mA]
Raspberry Pi Zero v1.3	250
GPS module	90
Raspberry Pi Camera Module v2	250
Radio communication module	120
Wind speed sensor (optical encoder)	12
ALTiMU	6
Thermometer	<1
TOTAL	~730

Table 2, power consumption estimation

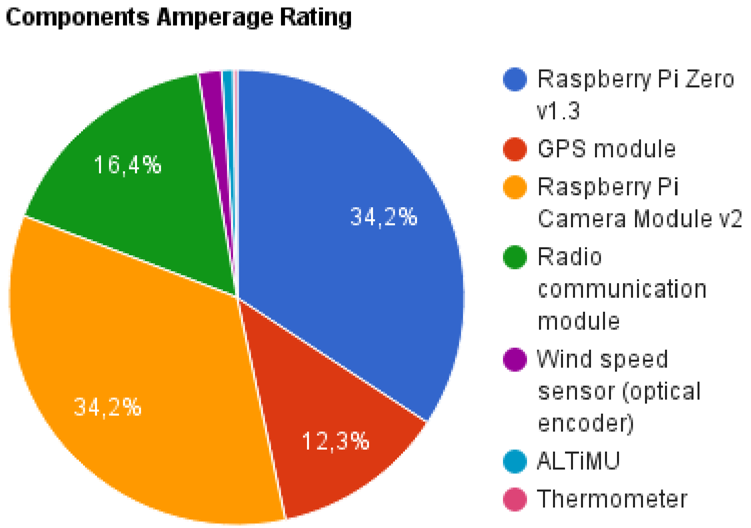
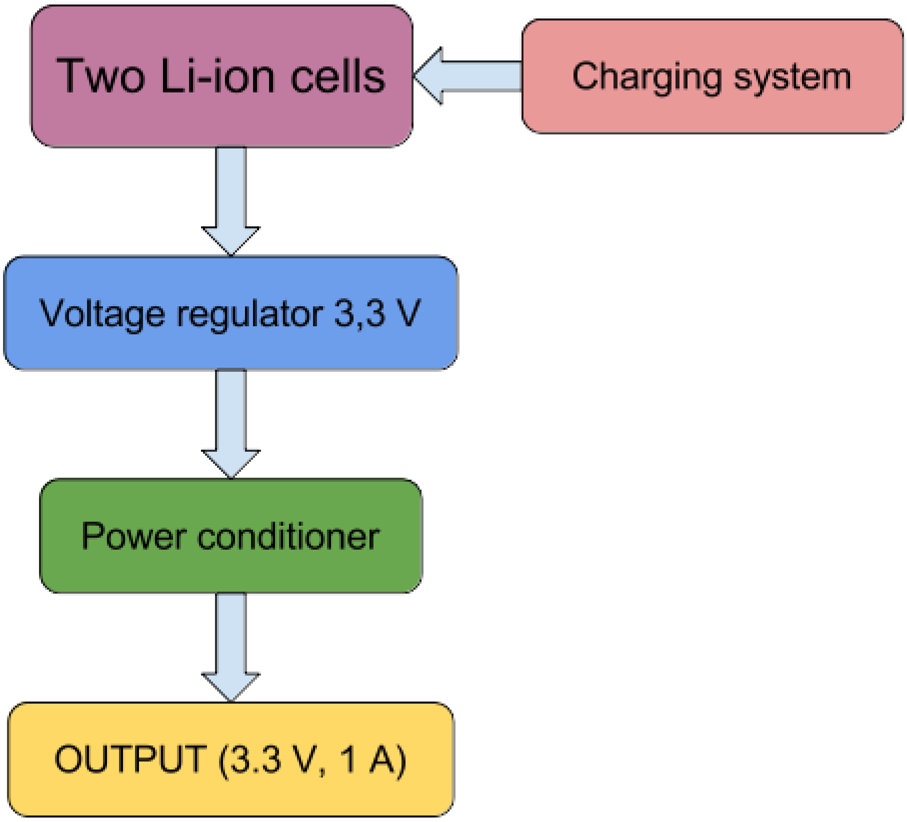


Chart 1, components power consumption percentage share

All estimates are based on official component manufacturer specification with exclusion of Raspberry Pi Zero computer module which power rating we have measured ourselves as data provided by Raspberry were based on measurements with both USB accessories and HDMI port powered. During the test we've figured that Raspberry Pi had a greatest load while decoding compressed video with passing the image over X enabled SSH, this resulted in the abovementioned (table) value.

Supply system



2.3.3 Diagram 2, power supply diagramRadio communication system

Due to high equipment availability, suitable wavelength legal limitations and contest rules we have decided to use 433MHz band for communication with our CanSat. It's designated as LPD433 which stands for Low Power Device 433MHz and is meant to be used for amateur radio applications requiring relatively long range. This frequency corresponds with 70 cm wavelength therefore allowing usage of 70 cm, 35 cm and 17 cm long antennae which are optimal considering CanSat dimensions. Our experimental idea is to embed a 70 cm long antenna in a parachute with a corresponding diameter. As we gather CanSat parts we will begin testing of different antennae shapes and wires used to produce it in order to estimate an optimal solution both in terms of

range and construction, considering flexibility and least interference with parachute itself. A quarter wavelength antenna will also be a part of our communication set.
On the receiving end we plan to use a commercial grade CDMA-10/TNC/15 antenna with the same radio module as in our CanSat. Modules on transmitting and receiving end will be controlled with on-board Raspberry Pi and ground based Arduino.
Despite hardware capability for now we don't plan to maintain nor even attempt a 2-way communication. Further testing will indicate whether CanSat antennas will allow receiving of signal with reasonable quality and at reasonable range.

2.4 Software design

2.4.1 CanSat software (based on RaspberryPi Zero)

CanSat software (based on RaspberryPi Zero)

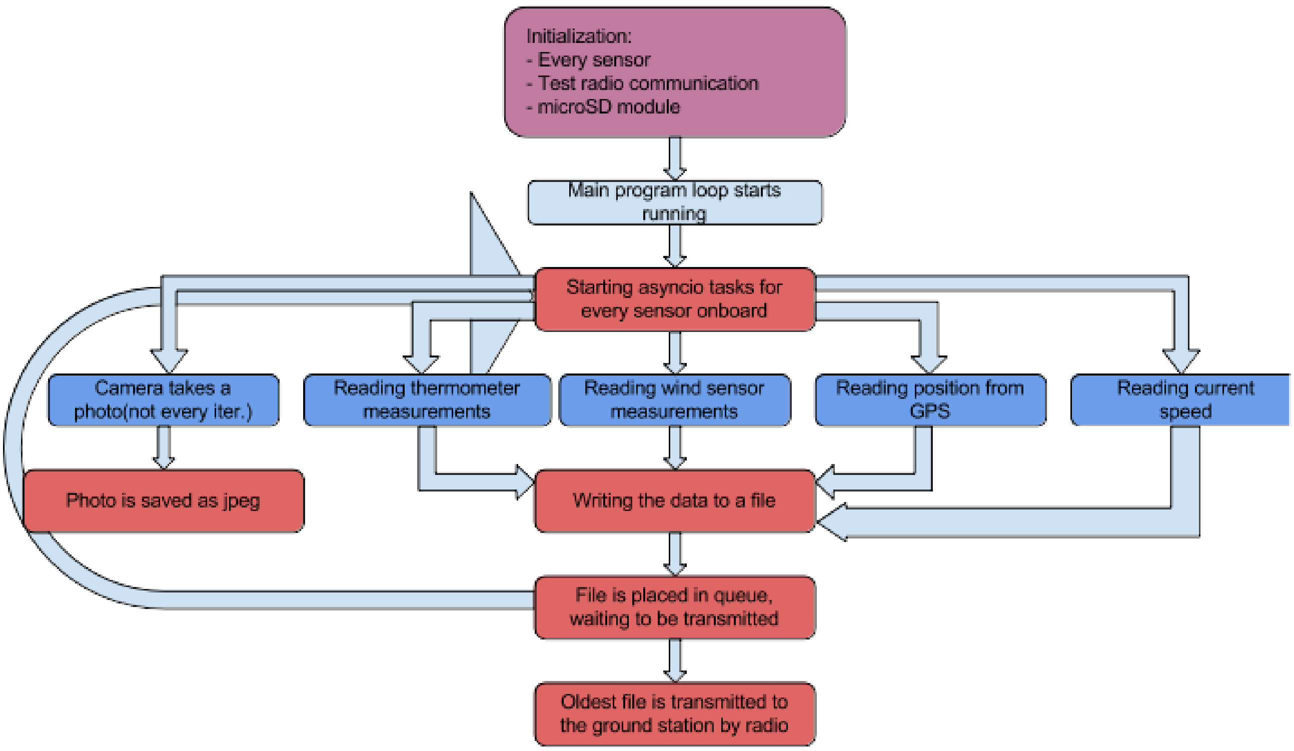


Diagram 3, CanSat software flow diagramThe program on the CanSat will be written in Python3 and is going to be focused around a main loop which logs data from the sensors. Its meant to be as simple as possible, which reduces the risk of bugs and crashes. Inside the loop, there are 5 asyncio tasks starting, one for each of the sensors outputs. Thanks to the properties of the asyncio library it allows us to wait for everything to be finished, and doesn't compromise the rest of the data if one sensor malfunctions. After that our CanSat will be programmed to gather environmental details, flight parameters and occasionally a photo of the land beneath. All data gathered will be stored on an onboard SD Card. After the parachute deployment the Cansat will begin transmitting already accumulated data and begin live parameters transmission. We aim to achieve a data resolution of 1 second. Further testing is required to choose an optimal picture properties and time interval at which they will be taken. Following landing, our CanSat will begin a loop transmission of all parameters gathered throughout the flight to ensure missing packets will be retransmitted. Further collection of onboard storage may optimistically turn out to be required only to acquire photos taken during flight. Micro SD card proves to be a reliable method of storage of both onboard code and gathered data as contemporary devices of this type are resistant to fluctuating temperature, water and even mild radiation. Their small size practically does not attribute to CanSat mass nor limit space available inside. Also a maximum capacity of many hundreds of Gigabytes is certainly above our requirements.

As no reliable testing has been done to time, we don't know how data transmission rate will change throughout the flight. We considered implementing a TCP/IP stack but we don't know if it would work for the same reason. As we acquire necessary parts and assemble them, intensive testing will be performed in order to clarify that.

All avionics and environmental data gathered have a simple form of an integer which takes 64 bits per number, together with a timestamp we expect about 0.5KB of data acquisition per second (excluding photos). Such excess may even allow continuous retransmission of legacy data. Connection stability testing will certainly answer whether such solution will prove useful.

Camera used in our design is a dedicated Raspberry Pi Camera module version 2 which uses an 8Mpx sensor. Assuming 8 bits per each of three channels an expected weight of each photo is 2MB, thanks to JPEG compression. Such a number is not a challenge for our onboard storage (provided a decent interval of about 10 seconds) but certainly exceeds our prognosed radio capabilities. Happily we've decided to put a Linux-driven computer onboard and an attempt to highly compress and transmit an image is certainly within our technological capabilities. Mainly data rate tests will decide whether we will attempt to do so.

2.4.2 Used IDE's and languages

Ground station:

Language: JavaScript, HTML5, CSS

Node.js 6.5.0, Chromium 53.0.2785.113 or higher, and Electron 1.4.5.

Receiver:

Language: C++

IDE: Arduino IDE

CanSat (Raspberry Pi Zero):

Language: Python3

2.5 Recovery system

At the beginning of planning the manners in which CanSat will be recovered, two methods were taken into account:

- parachute (necessary)
- airbag (optional)

Parachute as an indispensable and main element of recovery system was analyzed in several versions and combinations. It is able to reduce the speed of the CanSat efficiently and stabilize the descend simultaneously. The most important test was one, which was meant to determine the optimal shape of the parachute canopy (pic. 2.5.1). More informations about tests and their results are in section 3.3 "Describe tests" .

Airbag was an option that we had to reject because of many factors, these are the most important ones:

- There are no airbags widely available that we can afford to use (due to our budget, maximum weight and space inside the CanSat);
- It's not possible to build adequate airbag without an adequately supplied laboratory;
- In our CanSat we are going to use a camera module, which is located underneath the Can, therefore it poses a serious limitation in terms of placing such an airbag.

The way in which we want to fix the parachute to the CanSat is to connect top loop of the core with loop of a paracord by a climbing carabiner (strength 23 kN).

Assuming a height of 3000m and speed of $7\frac{m}{s}$ to $11\frac{m}{s}$, maximum fall time is about 7 minutes and minimum time is about 4 minutes 30 seconds.

Photo 3, handmade parachutes

2.6 Ground support Equipment

2.6.1 Ground station

By this term we consider a notebook equipped with receiving module and software for analyzing data gathered and transmitted by our CanSat.

Radio receiver

Software

Ground station

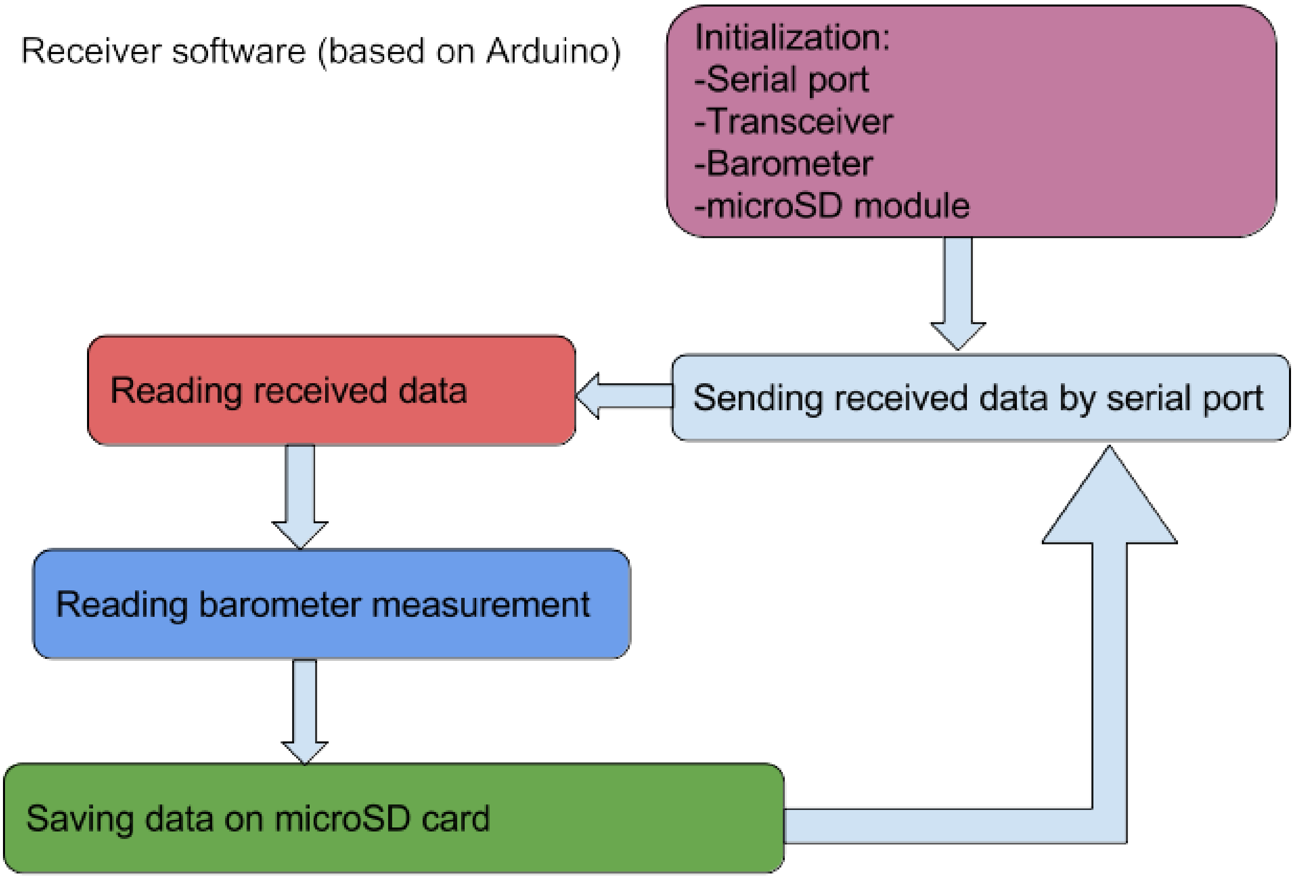
In order to receive data from the CanSat and send them to a computer we are going to use following components:

- Arduino Uno - main computer, processes received data, sends them to the computer and saves them on a microSD card,
- Transceiver - collects data send by CanSat and sends them to the Arduino,
- MicroSD module & microSD card - to keep received data as a low-level backup and for debugging purposes,
- YAGI antenna - to amplify the signal sent by the CanSat,
- USB cable - to communicate between the Arduino and a computer,
- Barometer - measurements might be useful to compare them with data received from the CanSat in order to determine relative height with high accuracy.

Connection with a computer will use Arduino’s built-in USB port.

Ground station software will consist of two main parts:

- Software in the radio receiver device,
- Desktop application for analysis and display of gathered data.


Receiver software (based on Arduino)
Initialization:
-Serial port
-Transceiver
-Barometer
-microSD module
Sending received data by serial port
Reading received data
Reading barometer measurement
Saving data on microSD card

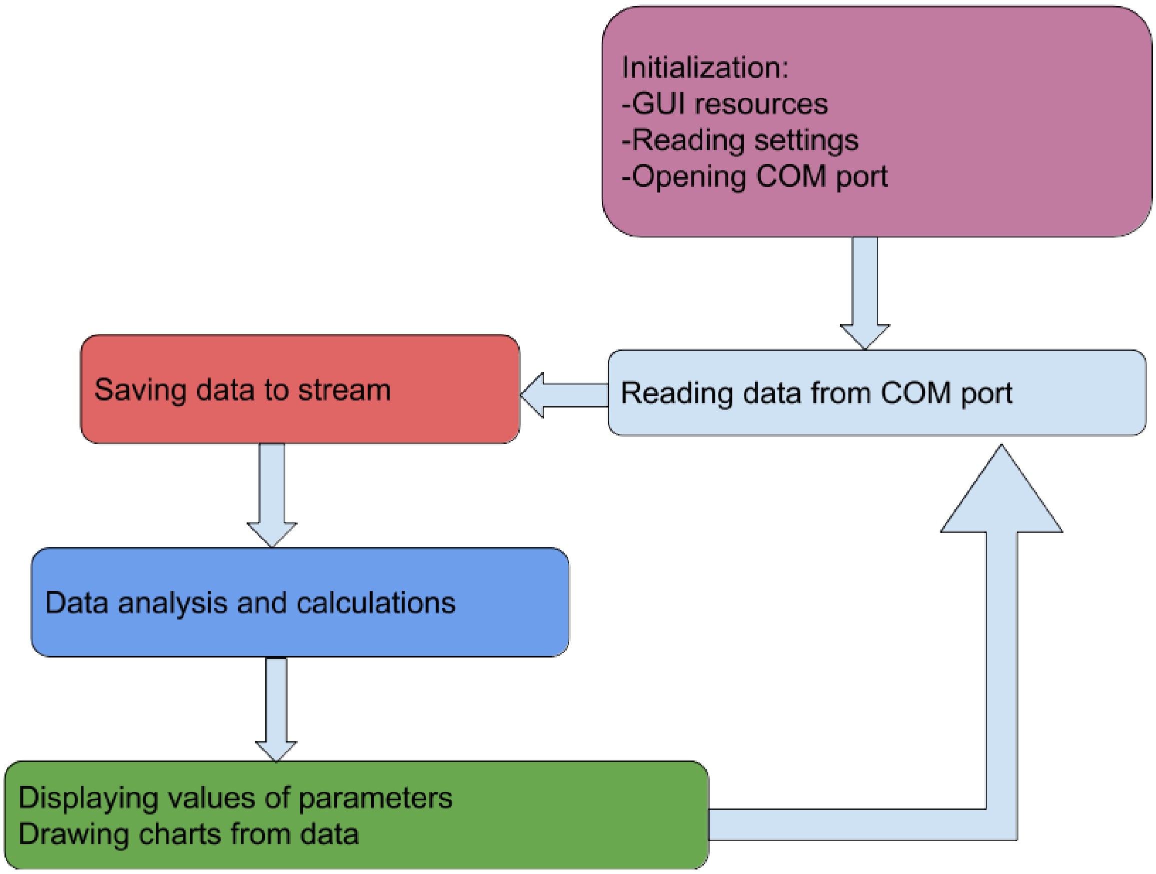


Diagram 4, radio receiver algorithm

Diagram 5, Desktop application algorithm

Initialization

After launching, the application will prepare all structures for handling received data, load last configuration, prepare charts library and it will ask user to choose used COM port.

Reading data from COM port

Data sent by the radio receiver to the COM port via USB cable will be read by the application. Library or component used for this purpose will be mentioned in a further design document.

Saving data to stream

Received data will be saved to stream and proper data structures. After CanSat's landing the file with received data will be created.

Data analysis and calculations

Primary objective of online radio transmission is to acquire measured temperature and pressure and compute CanSat's altitude above the ground.

Our secondary objective requirement is to analyze in real-time measurements from additional sensors. They will be displayed on the charts alongside with basic parameters. Application will process and show conditions for heavy lander's landing.

Displaying values of parameters and drawing data charts

Although standard controls will be used to display parameters values, drawing charts will require usage of a custom library, which will be included in a further design document.

Application's main loop will be repeated from this point.

3 PROJECT PLANNING

Task list of our project can be accessed using the following link.

https://trello.com/b/DM7W659L

3.1 Time schedule of the CanSat preparation

Our schedule is built around 4 key dates:

1. Preliminary Design Review (PDR) 6 November 2016
2. Critical Design Review (CDR) 15 January 2017
3. Final Design Review (FDR) 12 March 2017
4. Competition 13-16 April 2017

One week before all of them we have got planned critical team meetings which are meant to assess work progress.

3.1.1 Software

We plan to start work on both onboard and ground solutions as soon as we have applicable devices delivered for testing. By the date of CDR we plan to have working software prototypes. Missing features will be filled and bugs will be fixed during FDR preparations. Time directly before competition will be spent on polishing prepared solutions.

3.1.2 Hardware

Although the part list and hardware concepts are delivered within PDR, they will be bought and gathered within few days after PDR. Mechanical design is to be completed before CDR.

3.1.3 Tests

We are going to test every part of project before take-off, in the closest possible date after finishing its design. Considering that separate tests will not provide us assurance on the reliability of project solutions, we plan to execute numerous system integration tests.

November:

1. Research meant to determine impact of the shape of the parachute on the descend parameters,
2. Radio connection tests,
3. Dropping a similar weight to the one of a CanSat (with a parachute) from a tall structure - meant to observe miscellaneous factors we have not yet taken into consideration.

December:

1. Study of the CanSat's behavior during descent at a specific rate,
2. Comparison of structural materials,
3. Major sensor test (testing wind speed sensor and other sensors),
4. Reliability tests of power supply system aimed to determine parts factual power consumption and battery capacity.

January:

1. Base station mock data analysis,
2. System integration test - testing radio transmission during descent and online base station analysis,

3.2 Resource estimation

3.2.1 Budget

Probe	
Raspberry Pi Zero	72,60 zł
Thermometer	23,90 zł
Optical encoder	29,90 zł
Pololu AltIMU-10 v5	89,90 zł
GPS	197,00 zł
GPS antenna	57,00 zł
Raspberry Pi Camera module	135,00 zł
Li-Ion batteries	103,60 zł
3.3V step-up/step-down module	47,80 zł
Passive elements (filtration)	10,00 zł
Radio module	50,37 zł
TOTAL	817,07 zł

Table 3, CanSat budget

Ground station	
Arduino Uno Rev3	95,00 zł
Radio module	50,37 zł
Yagi antenna	91,00 zł
MicroSD module	14,90 zł
USB cable	4,19 zł
TOTAL	**255,46 zł**

Table 4, Ground station budget

Recovery system	130,00 zł
Production support budget	550,00 zł
Outreach budget	350,00 zł
1 Euro to PLN	4,312 PLN
Total	**2 102,53 zł**
Total [Euro]	**€487,60**

Table 5, Total budget

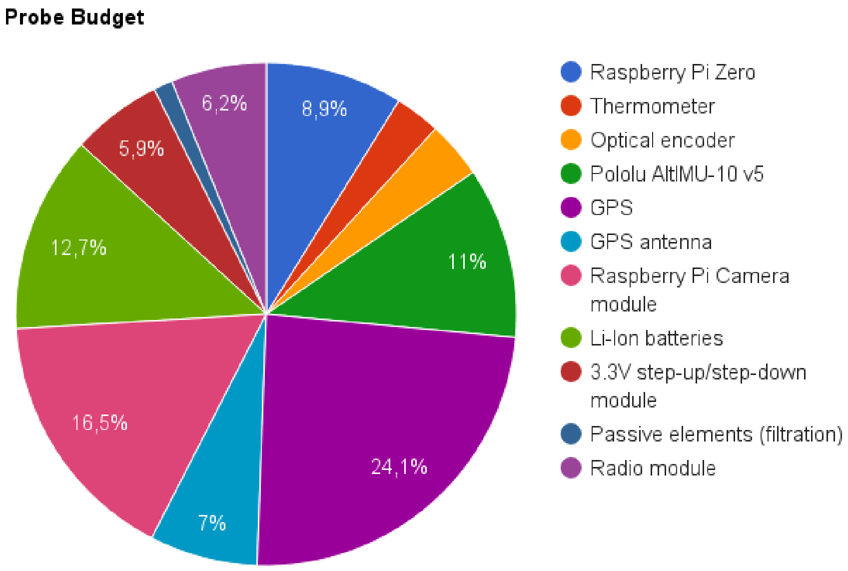


Chart 2, CanSat components price percentage share

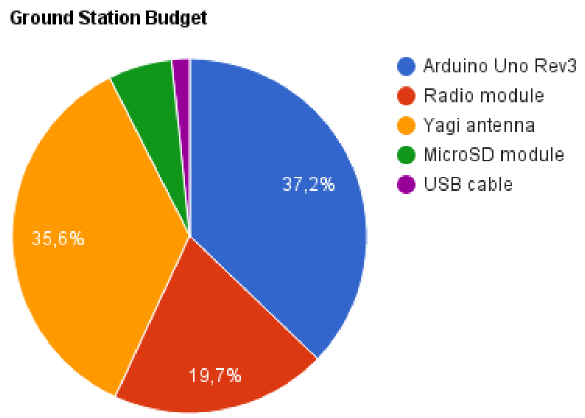


Chart 3, Ground station components price percentage share

Production support and outreach budget are positions that include all nonfinancial support from other organisations (access to conference room, etc.) and money spent on promoting our team’s activity.

3.2.2 External support

- We have successfully started cooperation with Delphi Poland S.A. which provides us with financial support in exchange for promoting Delphi with our marketing activities.
- DreamLab provides us with access to their conference room, where our meetings take place.

3.3 Test plan

Test schedule is described in the Section 3.1 of this document.

3.3.1 Parachute tests

Parachute test took place at Skałki Twardowskiego - limestone cliffs that form a group of hills located in the southern part of Kraków. Parachutes were made of an airtight material and paracords.
During tests we examined few issues:

- stability of flight trajectory,
- dependence the canopy shape on fall time,
- speed of falling object with parachute attached.

Parachute tests were conducted with three different shapes of canopy that were released from 30 meters above ground:

Data	Dome-shaped canopy	Dome-shaped canopy with a central hole	Square-shaped canopy
Fall time	16.9s	6.7s	15.1s
Stability	medium	perfect	medium
Flight trajectory	spiral	perfectly vertical	linear
Landing precision	medium	perfect	none
Other comments	-	high falling speed	soft but unpre-dictable flightpath

Table 6, parachutes comparison

Given the above data, we decided to focus on a dome-shaped canopy with a hole because of its flight stability, which is key for our CanSat mission. Further tests will be conducted in order to determine both optimal size of the canopy and the hole, as one used during testing appears to be of excessive.

3.3.2 Transmission tests

Transmission tests will be carried out in upcoming time. At the moment our estimates are based on the manufacturer's specification.

4 OUTREACH PROGRAMME

CanSat Team DevBlog

https://jjcansat.wordpress.com/

JJ CanSat Facebook profile

https://www.facebook.com/JJCanSatTeam/

In this field we will be cooperating with our sponsors. We will surely discuss the possibility of presentations in our school with the headmaster.

We created two profiles, one on Facebook.com, wrote in Polish and one on WordPress.com in English. Our Facebook profile is gradually gaining popularity. At the moment of writing this report, we have more than 200 followers on our Facebook page and we are reaching more than twice of that with our posts.

We are planning to make a promotional video for our project, which would serve as an advertisement of our actions if we succeed in contacting and cooperating with any large media.

We consider contacting with a local radio station (Radio Kraków) in order to arrange a radio interview about our work. The local newspaper, Gazeta Krakowska, is also under consideration of our outreach team.

Done outreach actions
Making Logo
Creating Facebook profile
Creating DevBlog
Making background image for our online profiles
Activity on Facebook

Table 7, done outreach actions

Planned outreach actions
Promotional video
Competition for our CanSat team name
Radio interview
Newspaper interview
Further activity on our profiles
Presentations in our school

Table 8, planned outreach actions

Picture 2, JJ CanSat logo

Picture 3, JJ CanSat background image

5 REQUIREMENTS

Characteristics	Figure
Height of the CanSat	113mm
Mass of the CanSat	345g (extra weight included)
Diameter of the CanSat	65mm
Length of the recovery system	780mm
Flight time scheduled	120s
Calculated descent rate	9 m/s
Radio frequency used	433.8MHz
Power consumption	0.75A (3.3V)/2.5W
Total cost	€487,60

Table 9, requirements

On behalf of the team I confirm that our CanSat complies with all the requirements established for the European CanSat competition in the official Guidelines, APPENDIX 1 of this document.

Signature, place and date

Patryk Gałczyński, Kraków 05.11.2016

6 ATTACHMENTS

Attachment 1 - Parts #1

APPENDIX 1 THE CANSAT REQUIREMENTS

1. All the components of the CanSat must fit inside a standard soda can (115 mm height and 66 mm diameter), with the exception of the parachute, radio antennas and GPS antennas, which can be mounted externally (on the top or bottom of the can, not on the sides). NOTE: the rocket payload area has 4.5cm of space available per CanSat, along the can's axial dimension (i.e. height), which must accommodate all external elements including: parachute, parachute attachment hardware, and any antennas.
2. The antennas, transducers and other elements of the CanSat cannot extend beyond the can's diameter until it has left the launch vehicle.
3. The maximum mass of the CanSat is limited to 350 g.
4. Explosives, detonators, pyrotechnics, and flammable or dangerous materials are strictly forbidden. All materials used must be safe for the personnel, the equipment and the environment. Material Safety Data Sheets (MSDS) will be requested in case of doubt.
5. The CanSat must be powered by a battery and/or solar panels. It must be possible for the systems to be switched on for three continuous hours.
6. The battery must be easily accessible, in case it has to be replaced or recharged in the field.
7. The CanSat should have a recovery system, such as a parachute, which is able to be reused after launch. It is recommended to use bright coloured fabric, which will facilitate recovery of the CanSat after landing.
8. The parachute connection must be able to withstand up to 1000N of force. The strength of the parachute must be tested, to give confidence that the system will operate nominally.
9. The flight time is limited to 120 sec.
10. The descent rate must be between 8 m/s and 11m/s.
11. The CanSat must be able to withstand an acceleration of up to 20g.
12. The total budget of the CanSat should not exceed €500.

If your team has any problems fulfilling some of these terms, please contact the competition organisers for advice. Failure to meet the requirements will be taken into consideration by the jury but will not necessarily prevent the CanSat from being launched.