CanSat 2022
Critical Design Review (CDR)
Outline
Version 1.1

1064

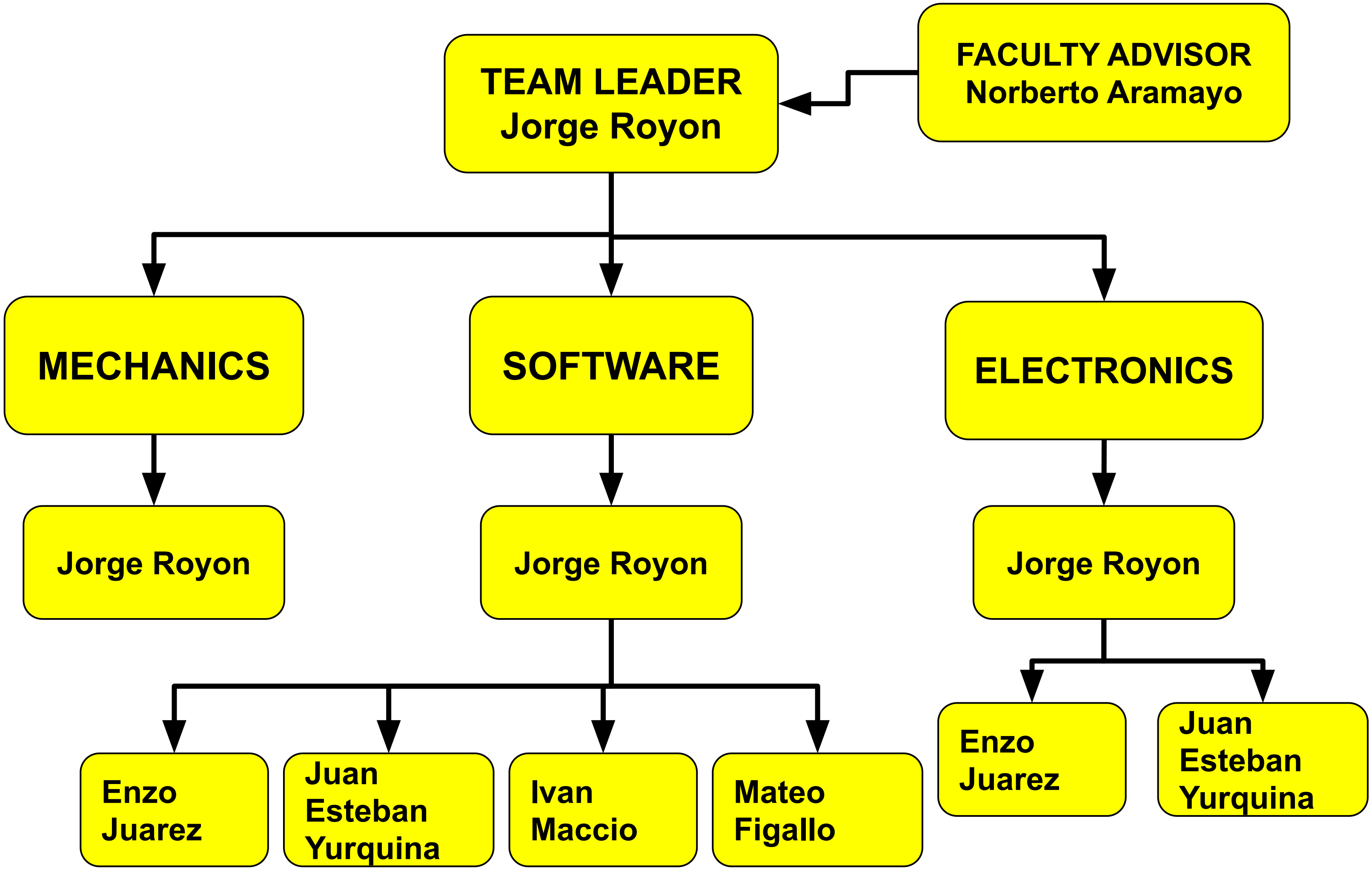
UCASAL

Presentation Outline (1 of 2)

Contents	Presenter	Slides
Team Organization	Jorge Royon	4
Acronyms	Jorge Royon	5
Systems Overviews	Jorge Royon	6-14
Sensor Subsystem Design	Enzo Juarez Juan Esteban Yurquina	15-27
Descent Control Design	Jorge Royon	28-51
Mechanical Subsystem Design	Jorge Royon	52-86
Communication and Data Handling (CDH) Subsystem Design	Jorge Royon Enzo Juarez Juan Esteban Yurquina	87-106

Presentation Outline (2 of 2)

Team Organization



Acronyms

Acronyms	Analysis
CDR	Critical design review
Hz	Frequency
DCS	Descent Control System
FRR	Flight Readiness Review
GCS	Ground Control Station
CG	Center of gravity
LCO	Launch Control Officer
RPM	Revolution per minute

System Overview

Jorge Royon

Mission Summary (1 of 2)

MAIN OBJECTIVES

1. **Design a Cansat that shall consist of a container and a payload. The Cansat shall be launched to an altitude ranging from 670 meters to 725 meters above the launch site and deployed near apogee (peak altitude).**

2. **The Cansat must survive the forces incurred at launch and deployment.**

3. **The Cansat shall descend using a parachute at a rate of 15 m/s.**

4. **At 400 meters, the Cansat shall deploy a larger parachute to reduce the descent rate to 5 m/s.**

5. **At 300 meters, the Cansat shall release a tethered payload to a distance of 10 meters in 20 seconds.**

6. **During that time, the payload shall maintain the orientation of a video camera pointing in the south direction.**

7. **The video camera shall be pointed 45 degrees downward to assure terrain is in the video.**

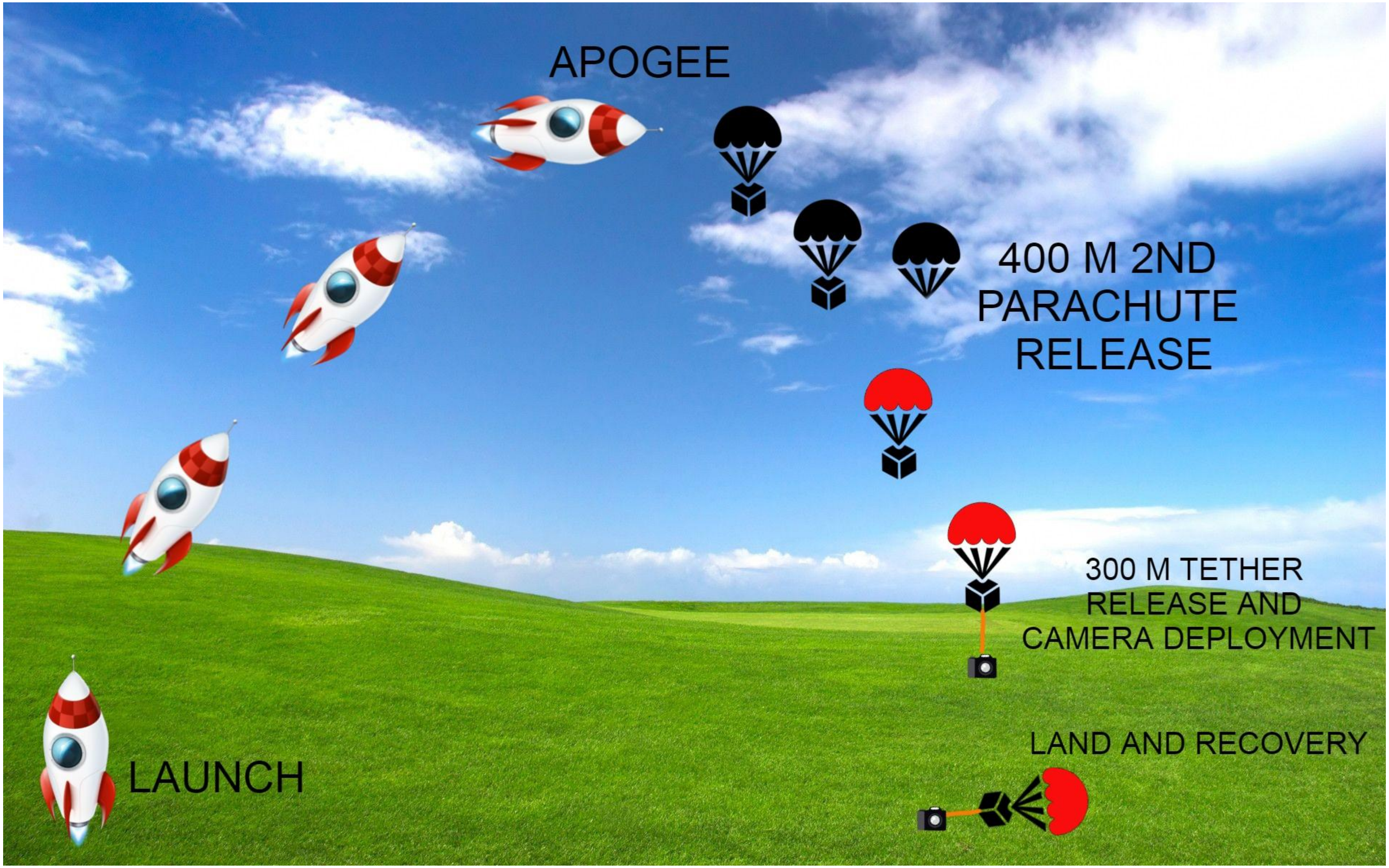
BONUS OBJECTIVES

- **As the container is releasing the payload, the container shall contain a video camera and start recording to show the descent of the payload. All videos are to be recorded and recovered when the Cansat is recovered from the field.**

Summary of Changes Since PDR

- **We haven’t changes to this section since the submission of the PDR.**

System Concept of Operations (CONOPS)



System Concept of Operations (CONOPS)

1 → 2 → 3

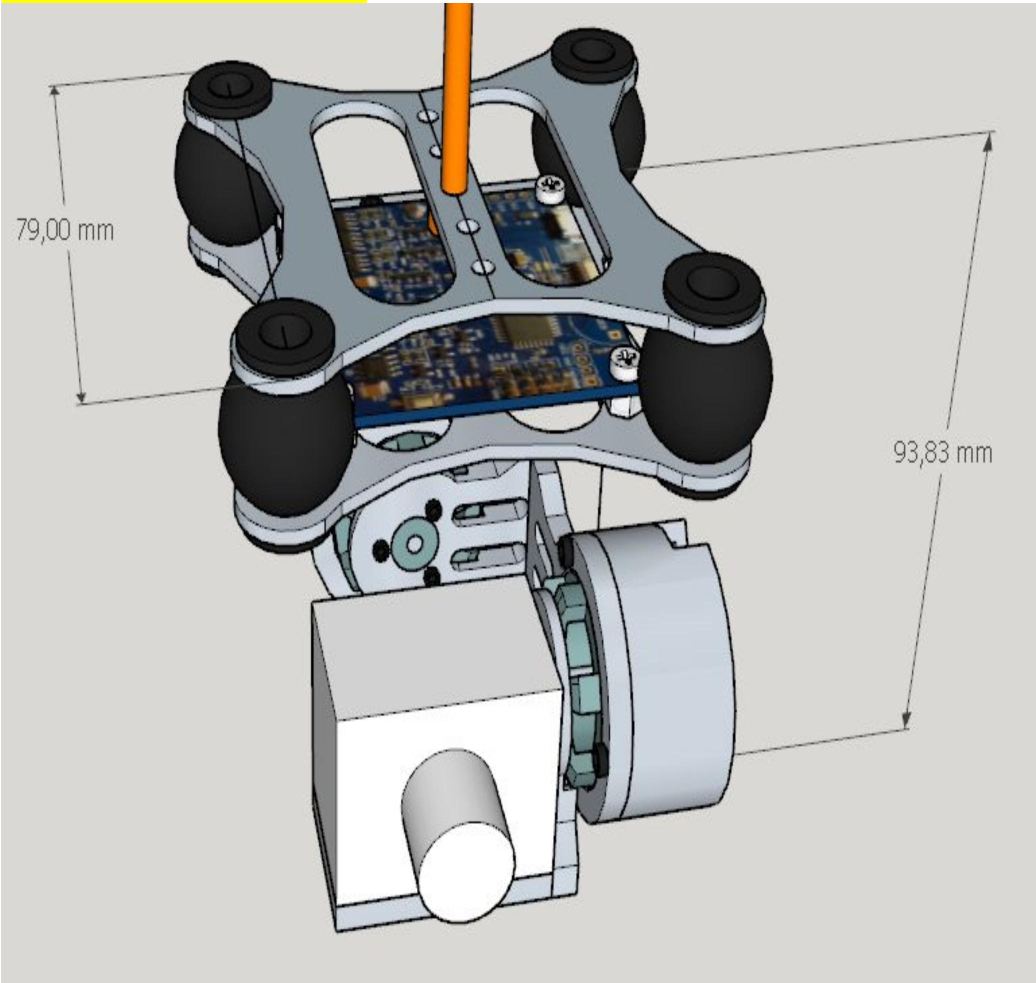
1
- Competition area arrival
- Team briefing
- Checklist of materials
- Deploy Ground Control Station
- Last check of fully integrated CanSat
- Placement of CanSat into the rocket payload section

2
- Launch CanSat.
- Container deploy.
- Payload deploy.
- Record telemetry.

3
- Recovery of Payload and Container
- Analysis of sampled data
- Preparation of the PFR
- PFR presentation and delivery

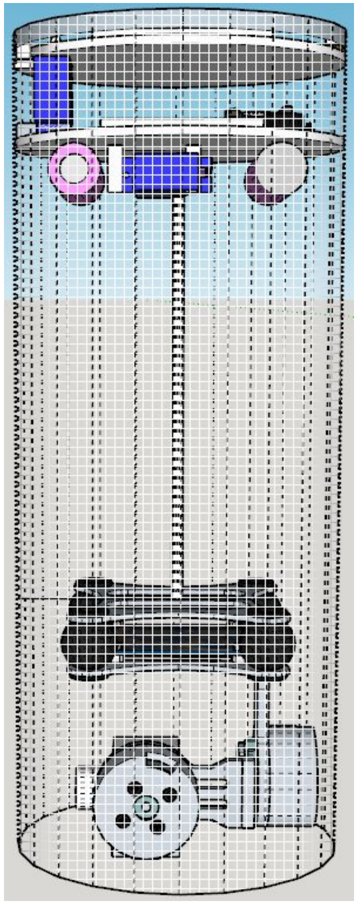
PAYLOAD

The payload consist in a 3 axis stabilization gimbal that keeps the camera at the required angle and has a bay for all the electronics

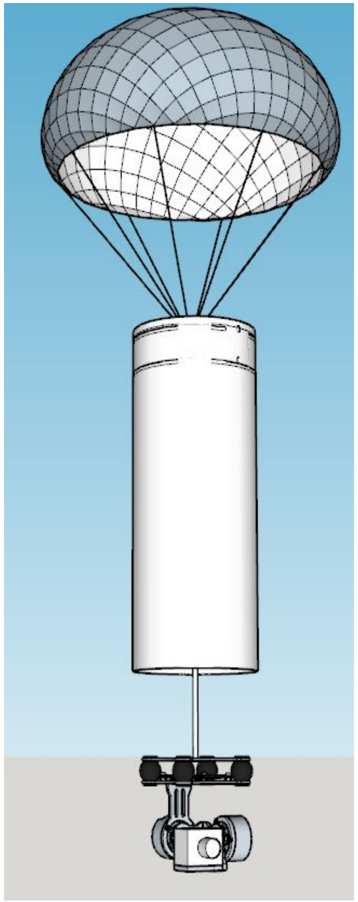


Payload Physical Layout

STOWED STATE



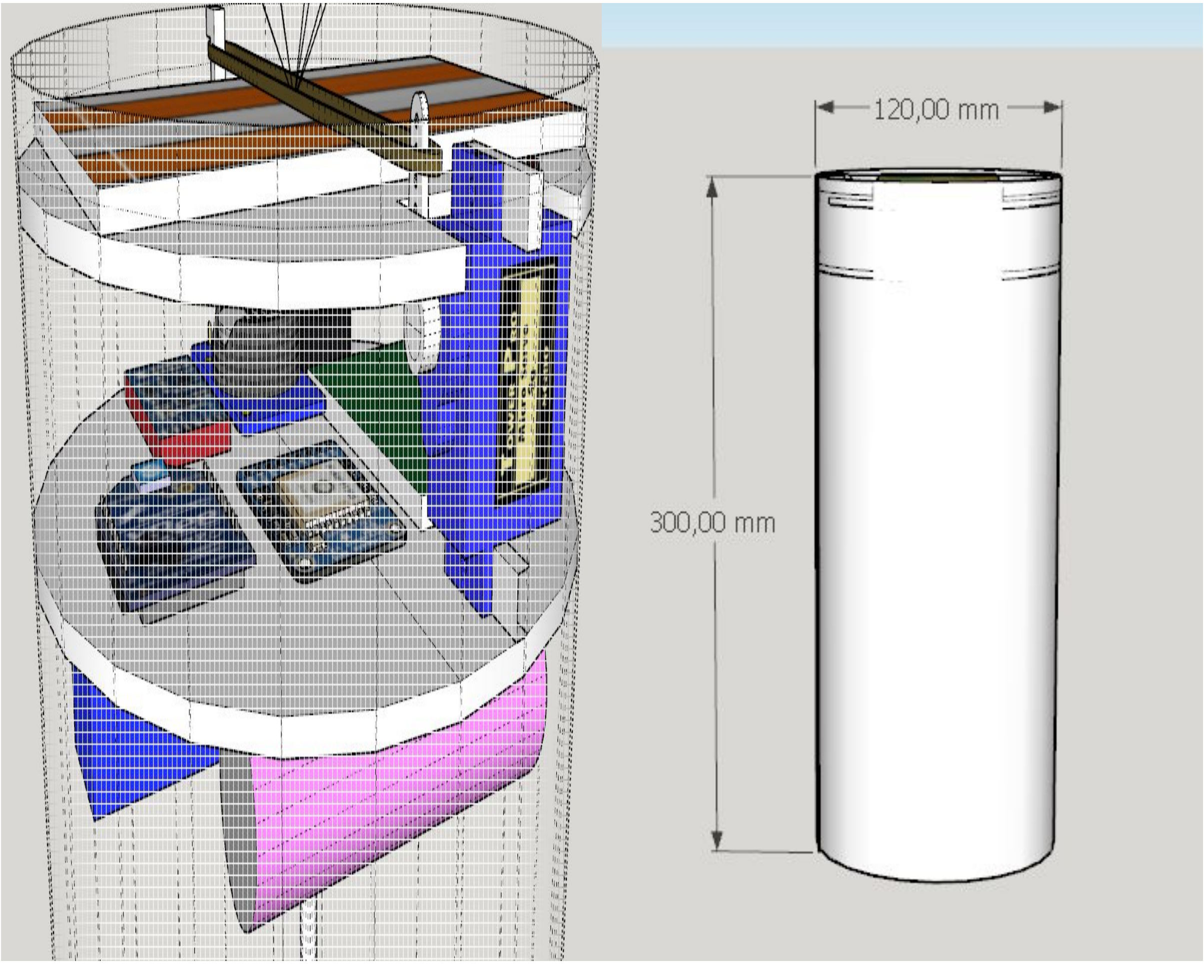
DEPLOYED STATE



Payload Physical Layout

CONTAINER

The container consist on two bays, one to hold the parachutes and the other one to hold the electronics, both are hold to the container with wood screw



Launch Vehicle Compatibility

- ★Rocket payload section dimensions based on Mission Guide:
  - ○Height: 310 mm
  - ○Diameter: 125 mm
- ★CanSat dimensions:
  - ○Height: 300 mm
  - ○Diameter: 120 mm
- ★Payload dimensions:
  - ○Diameter: 100mm
  - ○Height: 90 mm
- ★First Parachute dimensions:
  - ○Diameter: 190 mm
  - ○Spill Hole Diameter: 42.56 mm
- ★Second Parachute dimensions:
  - ○Diameter: 571 mm
  - ○Spill Hole Diameter: 63.86 mm

Launch Vehicle Compatibility

There's enough space for the Container freely slide in the rocket frame

Sensor Subsystem Design

Juan Esteban Yurquina
Juárez Enzo

Container Section

Sensor	Model	Purpose
Air Pressure	BMP280	Measure air pressure
GPS Sensor	Adafruit Ultimate GPS Sensor	Get GPS position
Voltage Sensor	Arduino ADC pin	Measure voltage
Altitude	BMP280	Detect altitude
Air Temperature	BMP280	Measure air, temperature & pressure
Bonus Camera	Adafruit MINI SPY CAMERA	Record video

Sensor Subsystem Overview

Payload Section

Sensor	Model	Purpose
Air Pressure	BMP280	Measure air, temperature & pressure
Air Temperature	BMP280	Measure air, temperature & pressure
Rotation Sensor	MPU9250	Measure rotation
Voltage Sensor	Arduino ADC pin	Measure voltage
Gyroscope	MPU9255	Calculate the orientation.
Magnetometer	MPU9255	Measure magnetic fields in order to orientate the camera.
Camera	Adafruit MINI SPY CAMERA	Record video

Sensor Changes Since PDR

- **We haven't changes to this section since the submission of the PDR.**

Container Air Pressure Sensor Summary

MODEL	INTERFACE	RANGE	ACCURACY	SIZE	MASS	COST
Waveshare 10 DOF IMU	I2c	300 hPa / 1100 hPa	± 1 hPa	31.2 mm x 17 mm x 3 mm	2gr	$17.99

Waveshare 10DOF IMU with bmp280

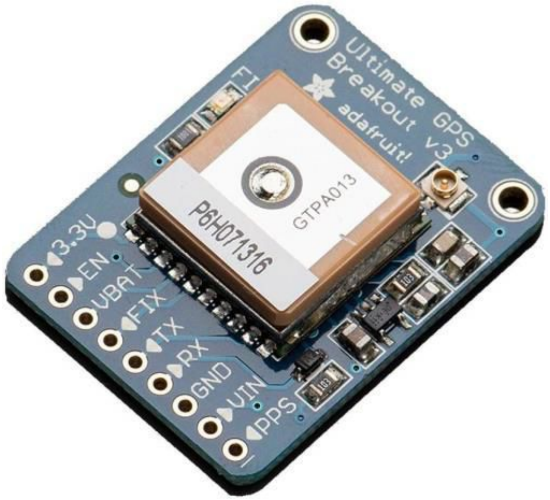
- **Considering 1 hPa, high measurement accuracy.**
- **Wide range for healthy measurement.**
- **Easy to access and useful library.**
- **One small board that save cost for the price of three sensors.**
- **Small size.**

Container GPS Sensor Summary

MODEL	INTERFACE	ACCURACY	SIZE	MASS	COST
Adafruit Ultimate GPS	UART	± 1.8 m	25 mm x 35 mm x 6.5 mm	8,5g	$39.95

Adafruit Ultimate GPS

- Considering its affordable cost, satisfies high position accuracy as 1.8 meter.
- Antenna of Adafruit Ultimate GPS is onboard, so it is useful in terms of volume.
- According to other projects experiences , Adafruit Ultimate GPS works stable.

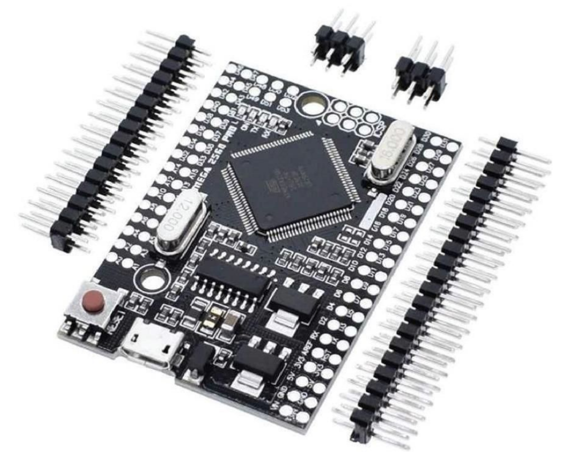


Container Voltage Sensor Summary

MODEL	ACCURACY	SIZE	MASS	COST
Arduino analog pin	±0.0049v	108 mm x 54 mm	34,9g	$ 0

Arduino analog pin

- **Considering it comes with the controlling electronics and high precision there's no need to invest in other sensor**



Payload Air Pressure Sensor Summary

MODEL	INTERFACE	RANGE	ACCURACY	SIZE	MASS	COST
Waveshare 10 DOF IMU	I2c	300 hPa / 1100 hPa	± 1 hPa	31.2 mm x 17 mm x 3 mm	2gr	$17.99

Waveshare 10DOF IMU with bmp280

- **Considering 1 hPa, high measurement accuracy.**
- **Wide range for healthy measurement.**
- **Easy to access and useful library.**
- **One small board that save cost for the price of three sensors.**
- **Small size.**

Payload Air Temperature Sensor Summary

MODEL	INTERFACE	RANGE	ACCURACY	SIZE	MASS	COST
Waveshare 10 DOF IMU with bmp280	I2c	-40 °C/85 °C	± 1°C	31.2 mm x 17 mm x 3 mm	2gr	$17.99

Waveshare 10DOF IMU with bmp280

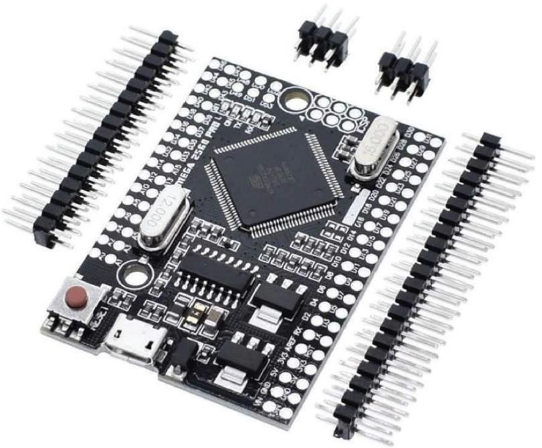
- **Considering 1°C, high measurement accuracy.**
- **Wide range for healthy measurement.**
- **Easy to access and useful library.**
- **One small board that save cost for the price of three sensors.**
- **Small size.**

Payload Voltage Sensor Summary

MODEL	ACCURACY	SIZE	MASS	COST
Arduino analog pin	±0.0049v	108 mm x 54 mm	34,9g	$ 0

Arduino analog pin

- **Considering it comes with the controlling electronics and high precision there's no need to invest in other sensor**



Payload Rotation Control Sensor Summary

MODEL	INTERFACE	RANGE	ACCURACY	SIZE	MASS	COST
Waveshare 10 DOF IMU	I2c	2000 °/s	± 1 hPa	31.2 mm x 17 mm x 3 mm	2gr	$17.99

Waveshare 10DOF IMU with bmp280

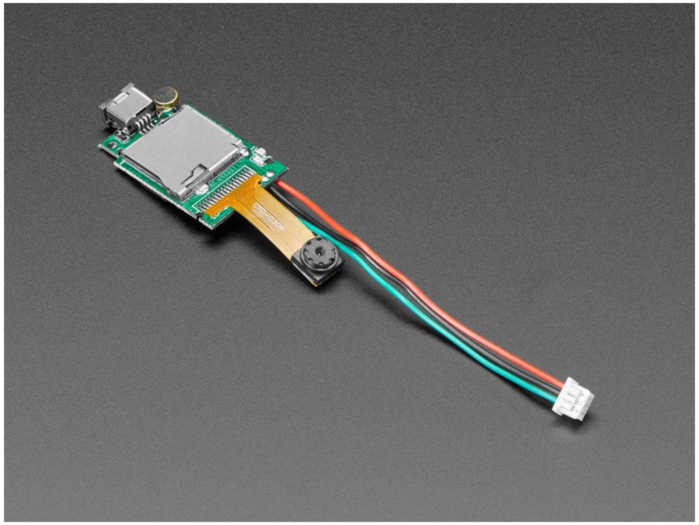
- Considering ±3% , high measurement accuracy.
- Wide range for healthy measurement.
- Easy to access and useful library.
- One small board that save cost for the price of three sensors.
- Small size.

Camera Summary

MODEL	INTERFACE	RESOLUTION	SIZE	MASS	COST
Adafruit MINI SPY CAMERA	USB/ANALOG	640x480	28.5 mm x 17mm x 4.2mm	2.8g	$ 12.50

Adafruit camera

- **Considering its size that is easy to fit in a slip tray to ease rotation.**
- **Easy interface with arduino, only one signal wire to start and stop recording.**

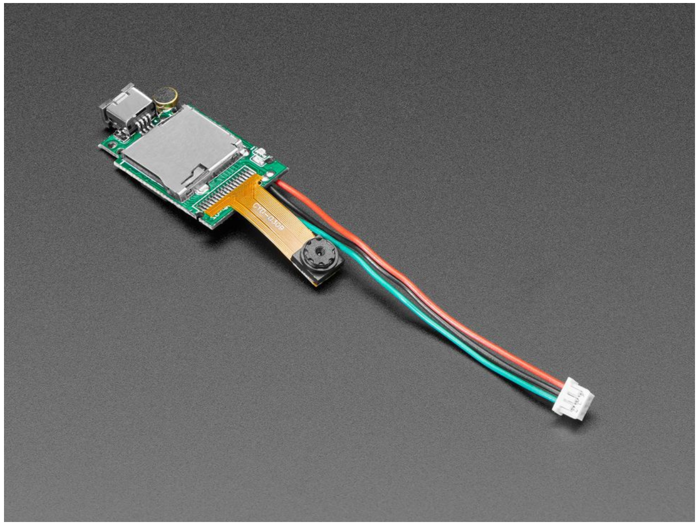


Bonus Camera Summary

MODEL	INTERFACE	RESOLUTION	SIZE	MASS	COST
Adafruit MINI SPY CAMERA	USB/ANALOG	640x480	28.5 mm x 17mm x 4.2mm	2.8g	$ 12.50

Adafruit camera

- Considering its size that is easy to fit in a slip tray to ease rotation.
- Easy interface with arduino, only one signal wire to start and stop recording.



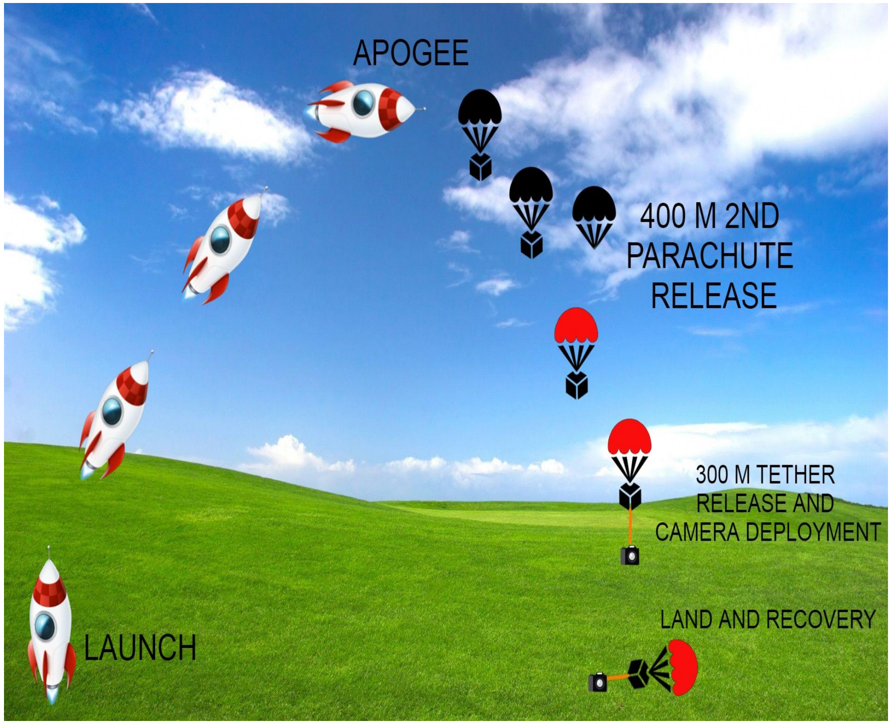
Descent Control Design

Jorge Royon

Container Descent Control System

The first parachute has 19 cm diameter to reach 15 m/s. It has been made from plastic and It has 2.28 cm radius spill hole to stabilize container during descent

The second parachute has 57 cm diameter to reach 5 m/s. It has been made from plastic and It has 6.38 cm radius spill hole to stabilize container during descent

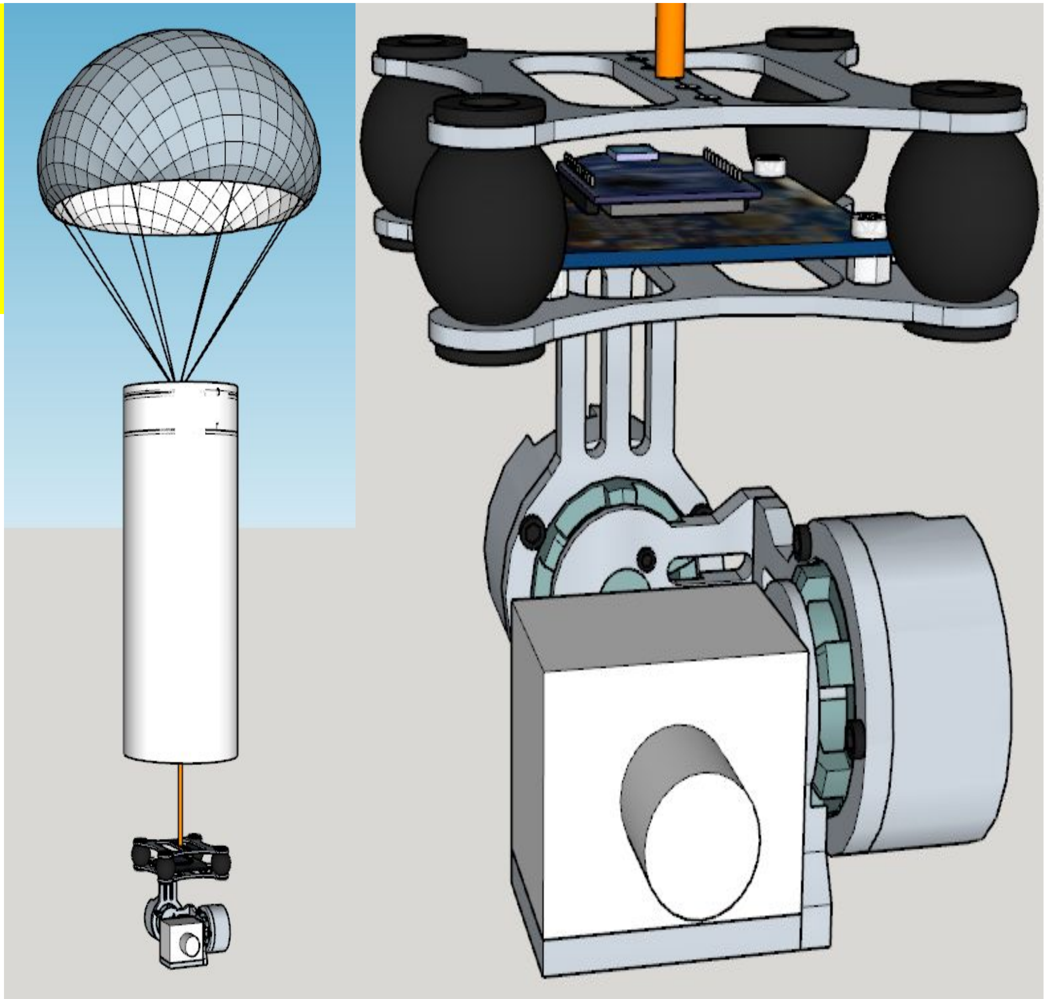


Descent Control Overview

DESCRIPTION
- We consider that the weight of the payload in the bottom on the container and when deployed it will be 10M away will be enough to stabilize it.

PROS:
- Easy to build
- Lightweight

CONS:
- Stability is achieved in a passive way and high speed winds can be a problem



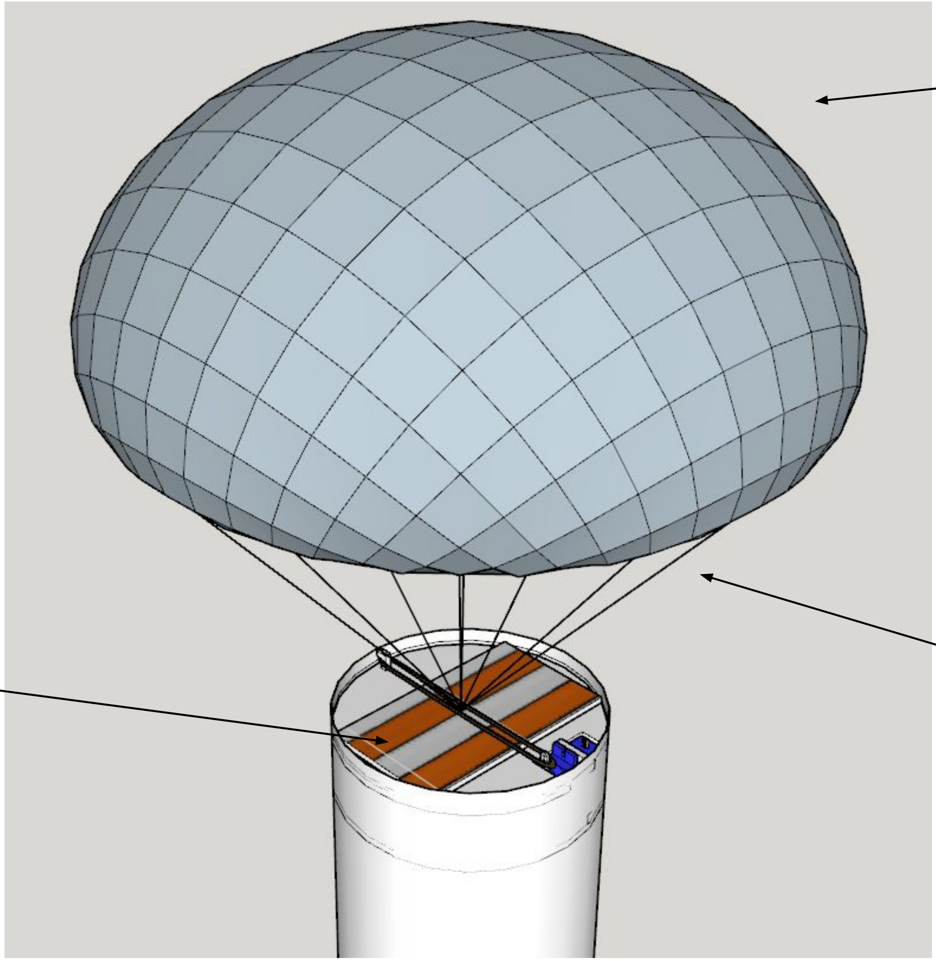
Descent Control Changes Since PDR

- **No changes since the PDR**

Container Descent Control Hardware Summary

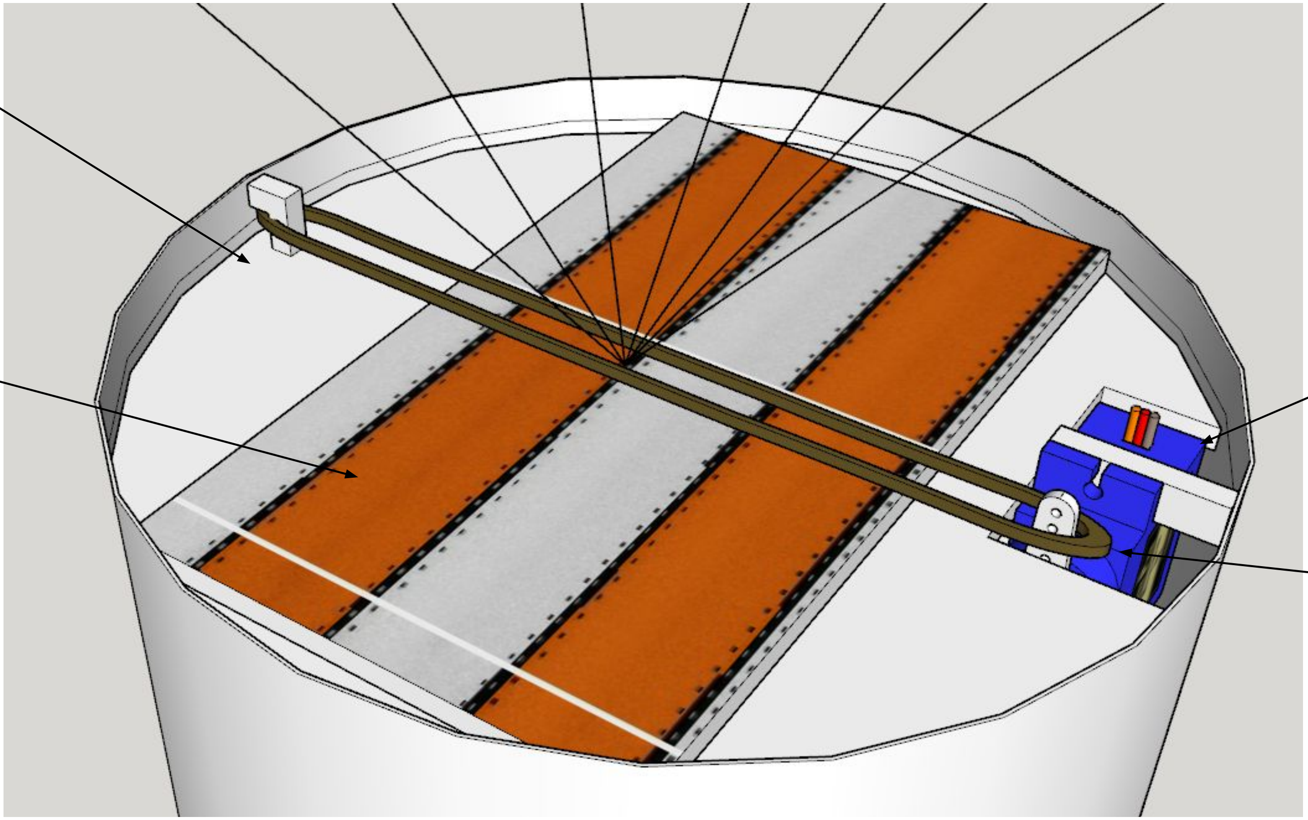
CONTAINER

- The first parachute is knotted to the second, the second is locked with a rubber band and released with a servo, when the second parachute is released the first pulls it out and then free fall

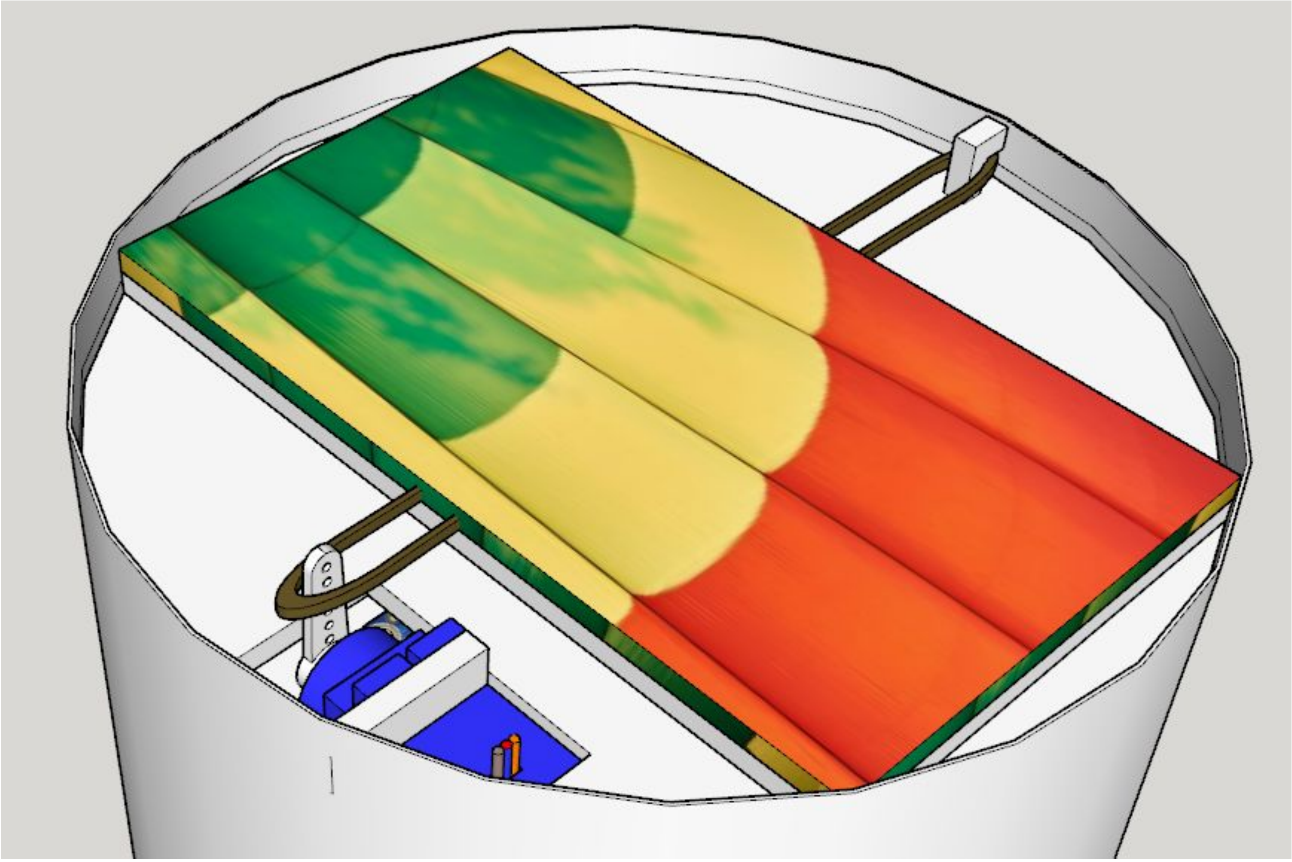


Container Descent Control Hardware Summary

CONTAINER

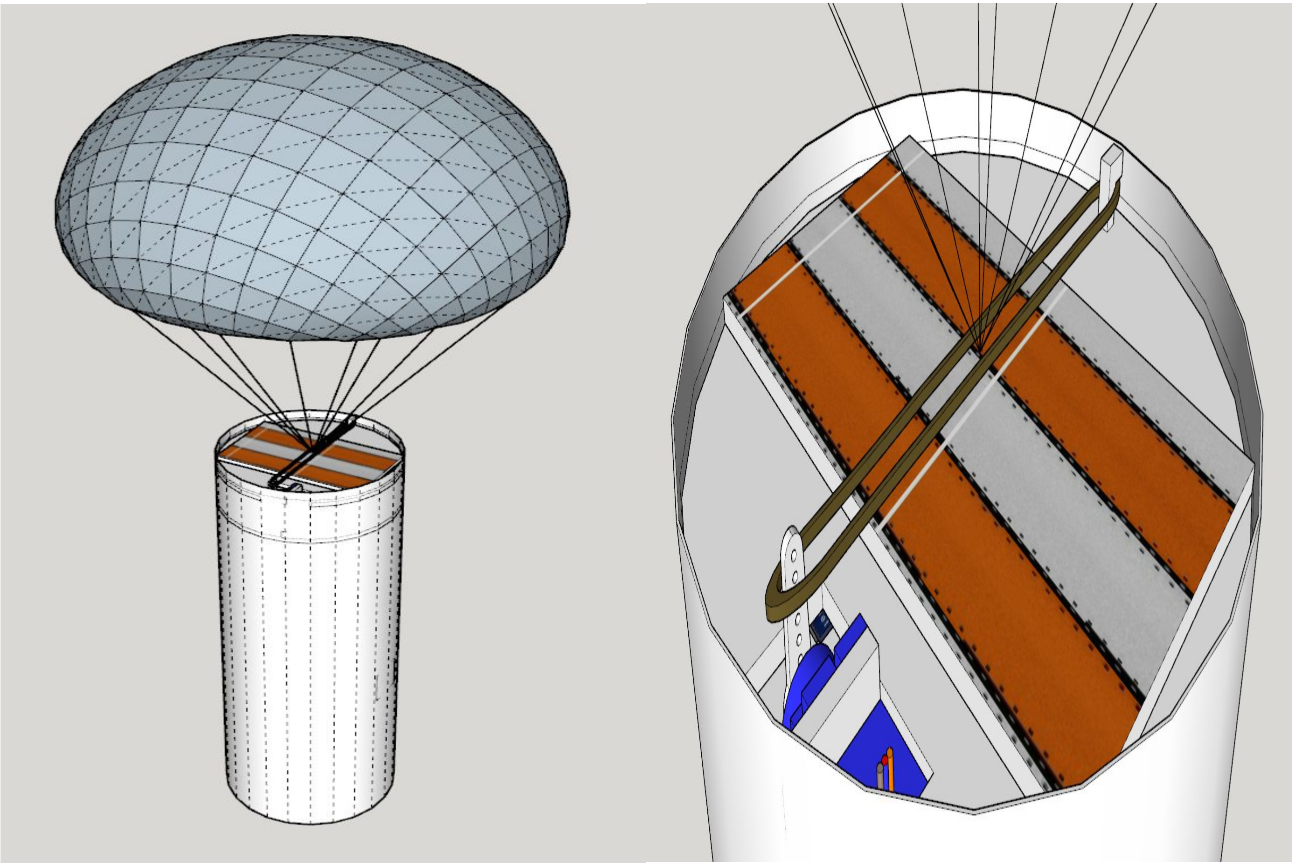


Container Descent Control Hardware Summary



- **The first parachute is tangled to the second parachute, that leaves the first parachute free to open when rocket releases the container**

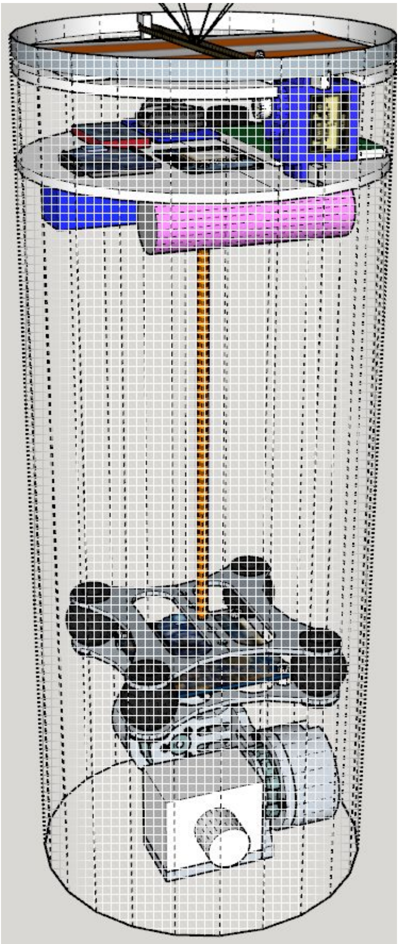
Container Descent Control Hardware Summary



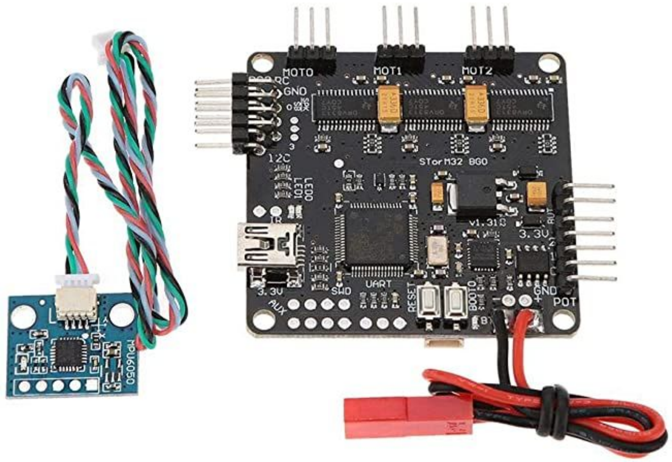
- **When the container is released the parachute deploy and keeps it place due the rubber band that is locked with a hook a rubber band and a servo**

Container and Payload Descent Stability Control Design

- **The container has a low cg because the payload is relative heavy to the rest of the cansat giving them a passive stabilization that prevent swaying**



- The payload consist in a 3 axis gimbal with a controller that is very precise and can kept pointing to nadir under all circumstances



Container Descent Rate Estimate

The area formula of the first parachute with spill hole is

$$S_p = \frac{2 \times m \times g}{\rho \times V^2 \times C_D}$$

where

Sp= Area of the parachute with a spill hole ($m2$)

D= The diameter of the parachute mm

V= Descent speed m/s

CD= 1.5 Drift coefficient of parachute

ϱ= 1.225 kg/m3(Air density at +15 C from sea level)

m=0.600kg Container+Payload

g=9.81m/s2

S_(s_H) =Spill hole area

S_Hr =Spill hole radius

Container Descent Rate Estimate

Replacing the data gives the following results

$$S_p = \frac{2 \times (0{,}6kg) \times \left(9{,}81\,\frac{m}{s^2}\right)}{(1{,}225\,\mathrm{kg}/m^3) \times \left(15\,\frac{\mathrm{m}}{\mathrm{S}}\right)^2 \times 1{,}5}$$

Sp=0.02847$m2$

The diameter can be calculated from the formula

$$S_p = \frac{1}{4}\pi D^2$$

Container Descent Rate Estimate

Clearing terms and replacing the data give us

$$D = \sqrt{\frac{4 \times S_P}{\pi}} = \sqrt{\frac{4 \times 0{,}02847m2}{\pi}} = 0{,}1904036m = 190mm$$

The spill house area is chosen to be 5% of the total parachute area

$$S_{S_H} = S_p \times 5\% = \mathbf{0.02847m2} * \mathbf{0,05} = \mathbf{0,0014235m2}$$

Container Descent Rate Estimate

The spill house radius can be determined by

$$S_{Hr} = \sqrt{\frac{S_{sH}}{\pi}} = \sqrt{\frac{0,0014235\mathrm{m2}}{\pi}} = 0,0212864\mathrm{m} = 21,28\mathrm{mm}$$

This give us a parachute of a 190 mm in diameter with a spill house radius of 21,28 mm for the first parachute

Container Descent Rate Estimate

Now we can estimate the descent speed of the container with the payload

$$V = \sqrt{\frac{2 \times F_{Drag}}{\rho \times S_P \times C_D}}$$

$$V = \sqrt{\frac{2\times(0.6kg)\times(9.81m/s2)}{1.225kg/m3\times(0.02847m2)\times(1.5)}} = 15m/s$$

With these result we accomplish the mission requirements for the first stage

Container Descent Rate Estimate

The area formula of the second parachute with spill hole is $$S_p = \frac{2 \times m \times g}{\rho \times V^2 \times C_D}$$ where

Sp= Area of the parachute with a spill hole ($m2$)

D= The diameter of the parachute mm

V= Descent speed m/s

CD= 1.5 Drift coefficient of parachute

ϱ= 1.225 kg/m3(Air density at +15 C from sea level)

m=0.600kg Container+Payload

g=9.81m/s2

S_(s_H) =Spill hole area

S_Hr =Spill hole radius

Container Descent Rate Estimate

Replacing the data gives the following results

$$S_p = \frac{2 \times (0{,}6kg) \times \left(9{,}81\,\frac{m}{s^2}\right)}{(1{,}225\ \mathrm{kg}/m^3) \times \left(5\,\frac{m}{S}\right)^2 \times 1{,}5}$$

$Sp{=}0.25626m2$

The diameter can be calculated from the formula

$$S_p = \frac{1}{4}\pi D^2$$

Container Descent Rate Estimate

Clearing terms and replacing the data give us

$$D = \sqrt{\frac{4 \times S_P}{\pi}} = \sqrt{\frac{4 \times 0.25626m2}{\pi}} = 0{,}5712109m = 571mm$$

The spill house area is chosen to be 5% of the total parachute area

$$S_{S_H} = S_p \times 5\% = \mathbf{0.25626m2} * \mathbf{0,05} = \mathbf{0,0128m2}$$

Container Descent Rate Estimate

The spill house radius can be determined by

$$S_{Hr} = \sqrt{\frac{S_{sH}}{\pi}} = \sqrt{\frac{0,012813\text{m2}}{\pi}} = 0,06386332\text{m} = 63,86\text{mm}$$

This give us a parachute of a 571 mm in diameter with a spill house radius of 63,86 mm for the second parachute

Container Descent Rate Estimate

Now we can estimate the descent speed of the container with the payload

$$V = \sqrt{\frac{2 \times F_{Drag}}{\rho \times S_P \times C_D}}$$

$$V = \sqrt{\frac{2\times(0.6kg)\times(9.81m/s2)}{1.225kg/m3\times(0.25626m2)\times(1.5)}} = 5m/s$$

With these result we accomplish the mission requirements for the second stage

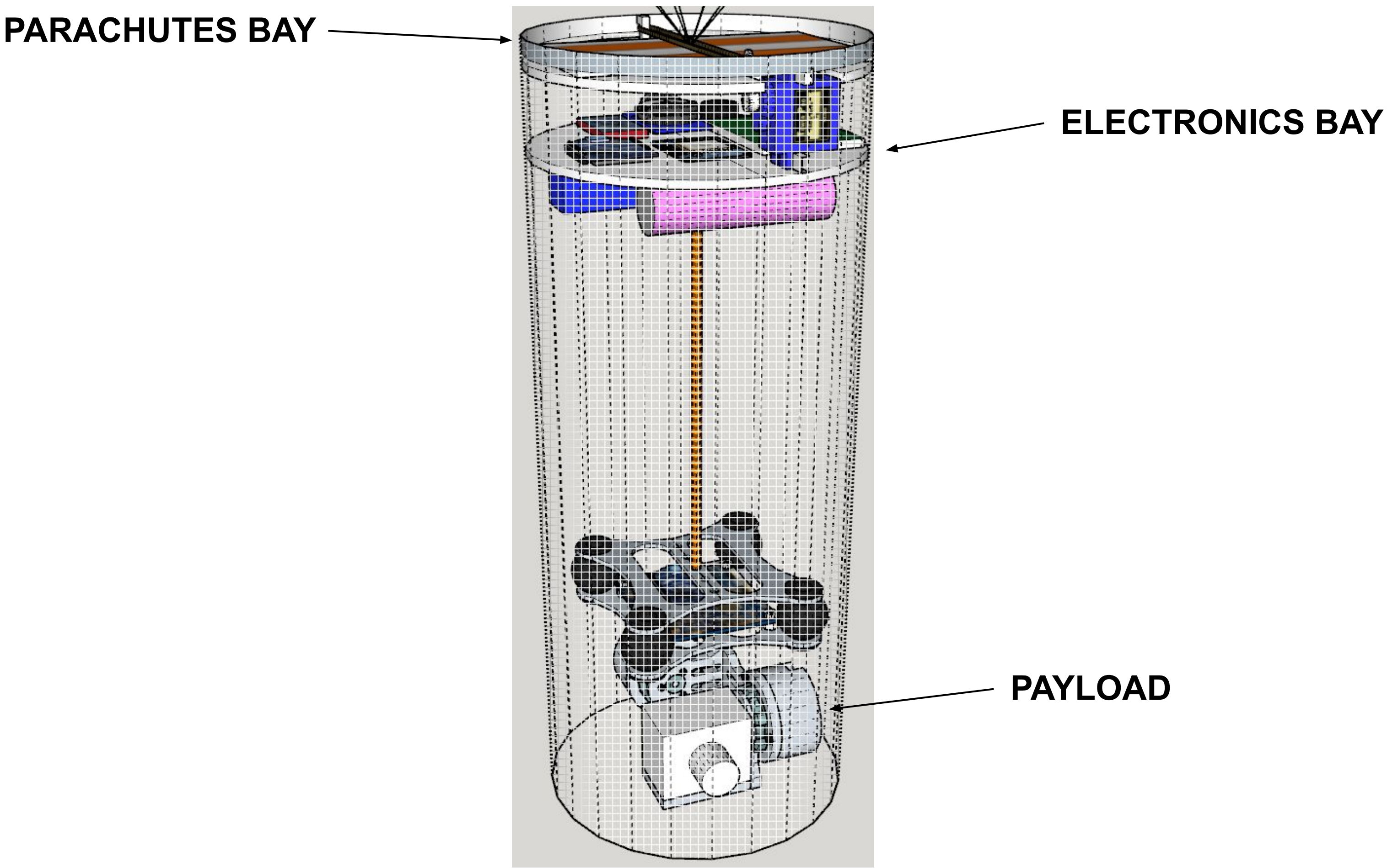
Mechanical Subsystem Design

Jorge Royon

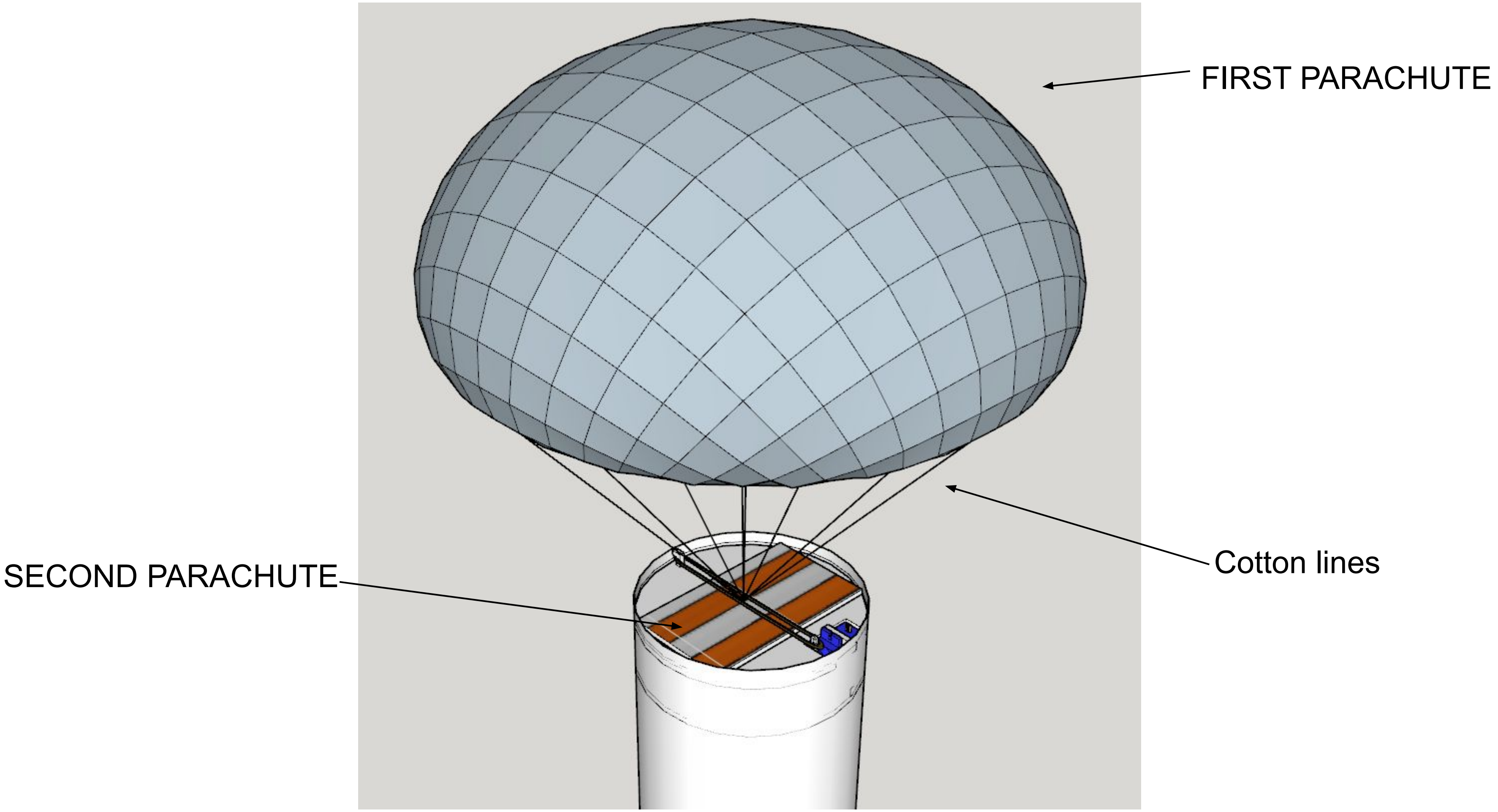
CONTAINER

DESCRIPTION:
•12 cm diameter by 30 cm length
•3D printed in pla in thin walls of 0,8mm
•Estimate weight of 207,5gr

CONTAINER



CONTAINER

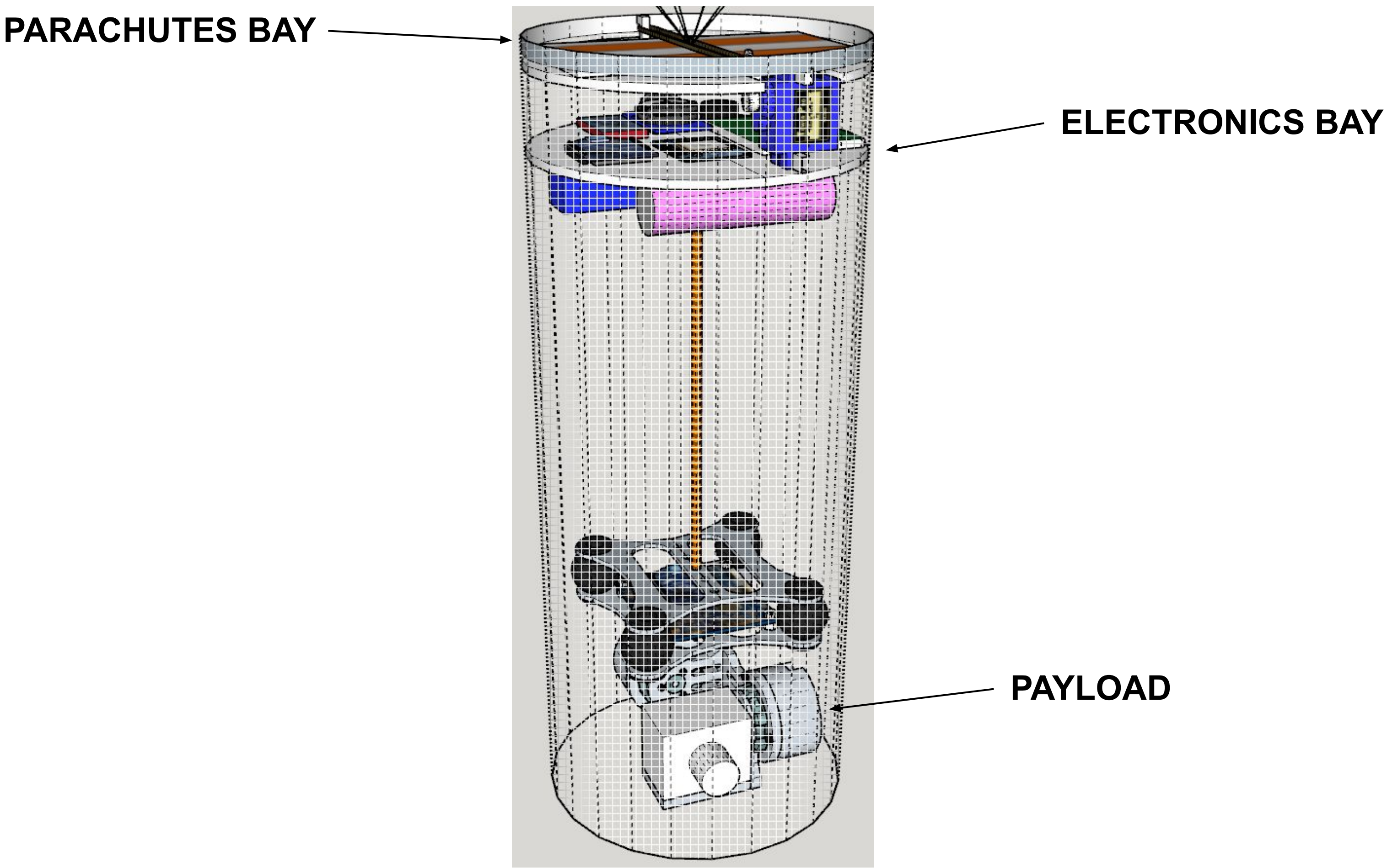


Mechanical Subsystem Changes Since PDR

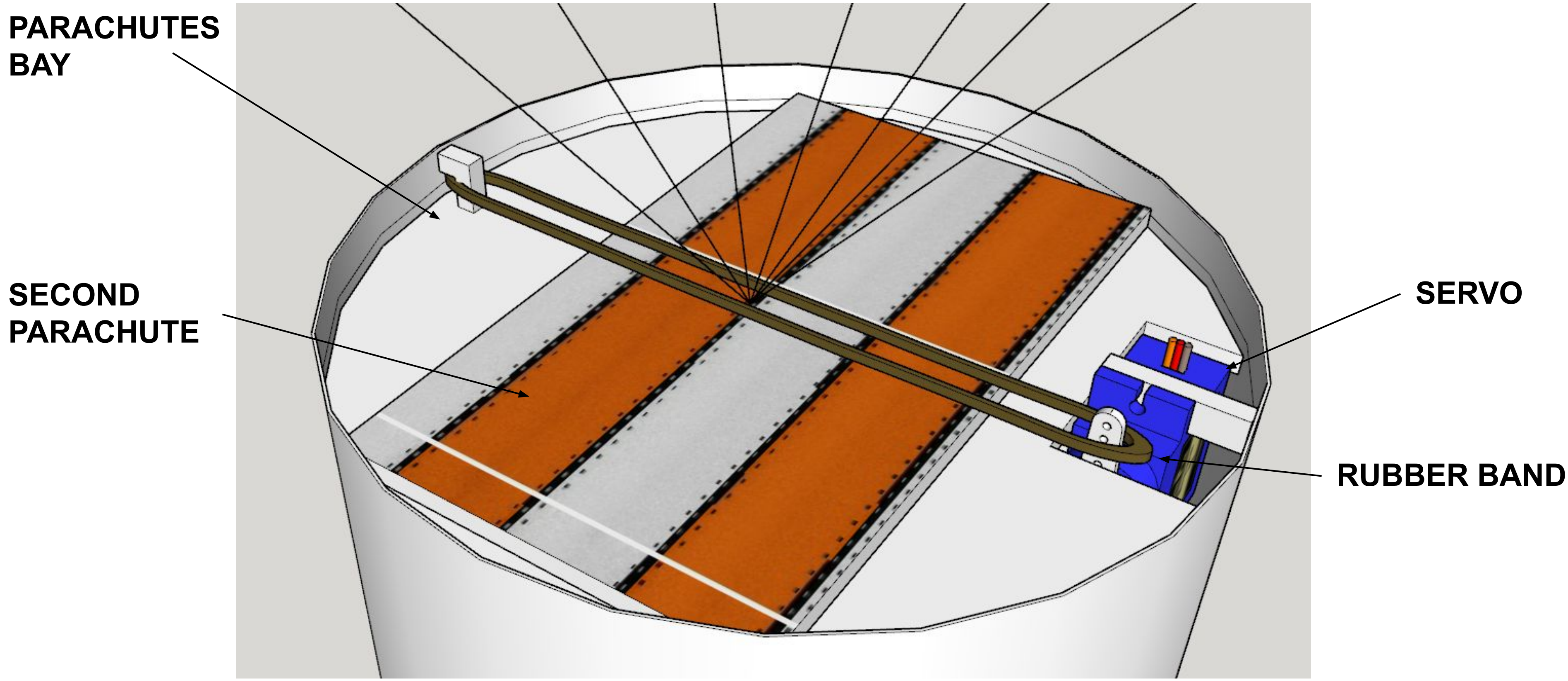
- **No changes since PDR**

Container Mechanical Layout of Components

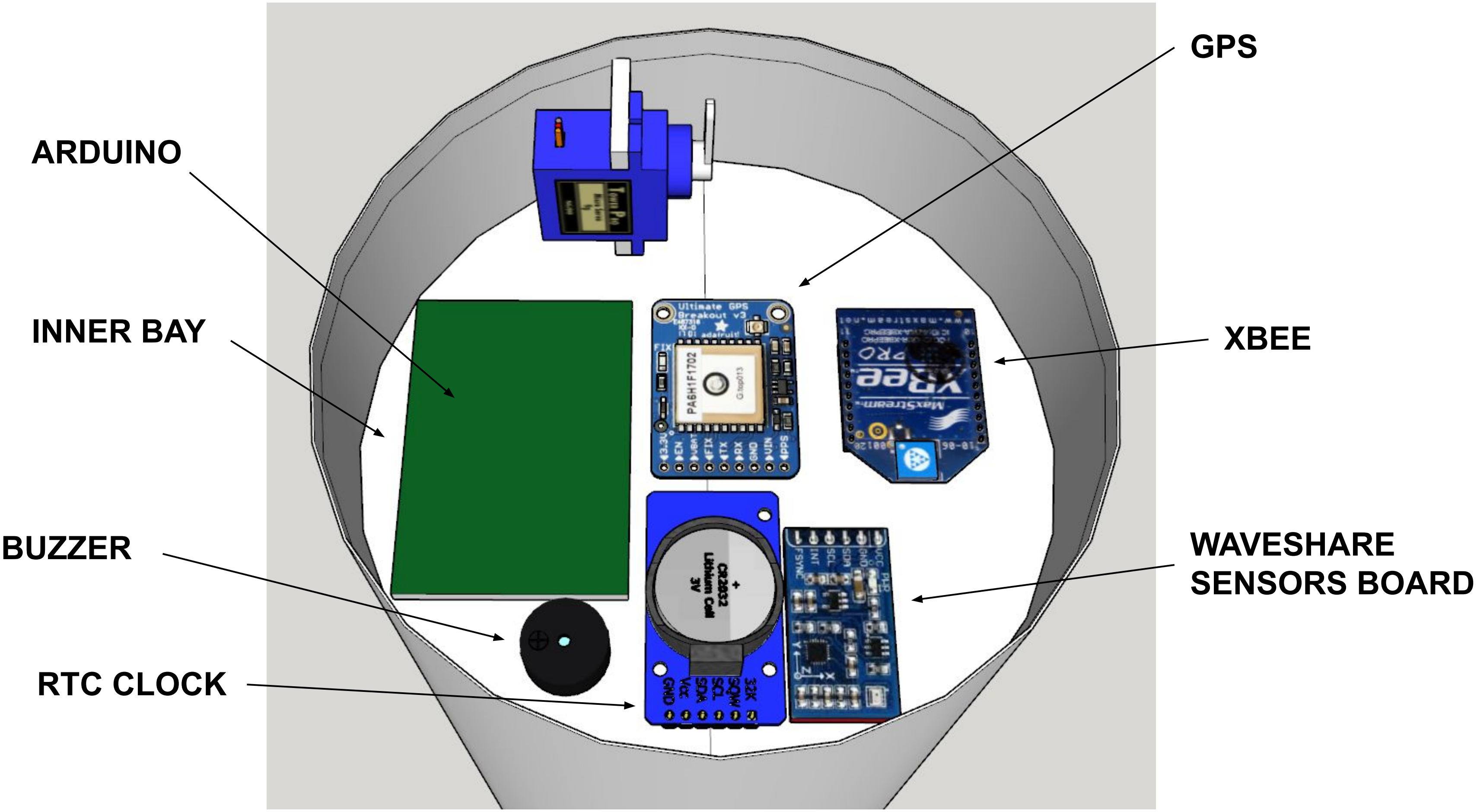
CONTAINER



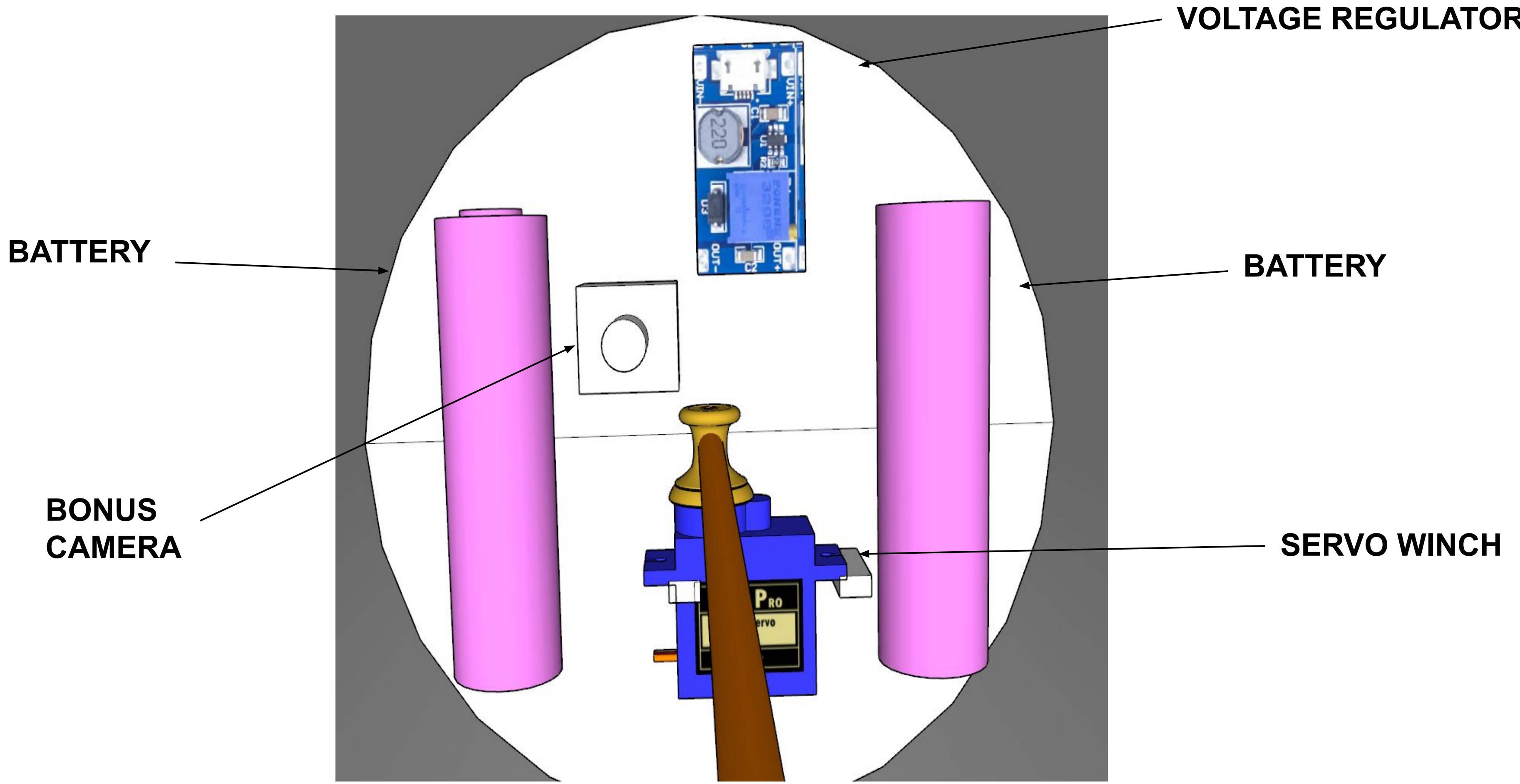
Container Mechanical Layout of Components



Container Mechanical Layout of Components

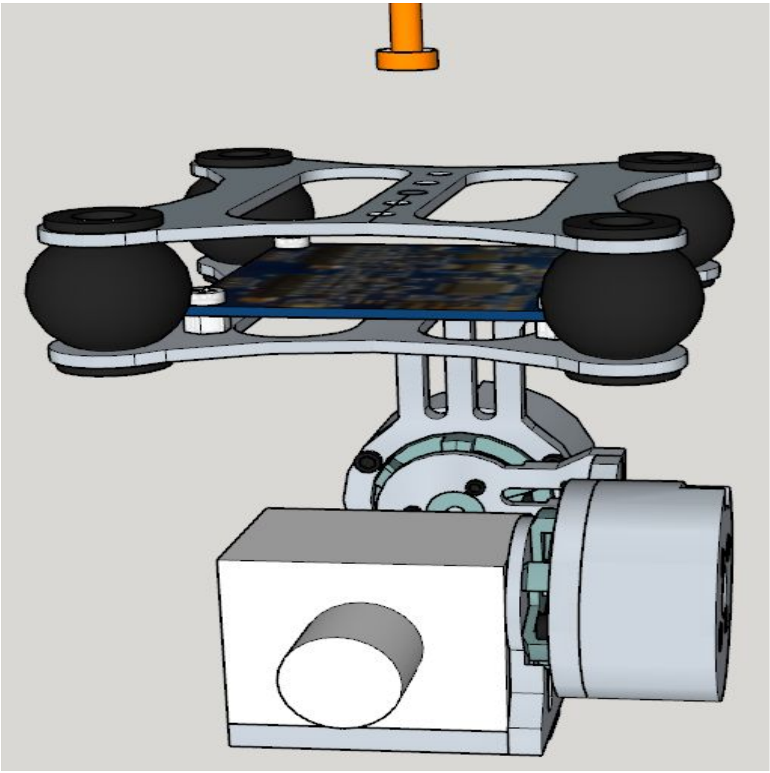


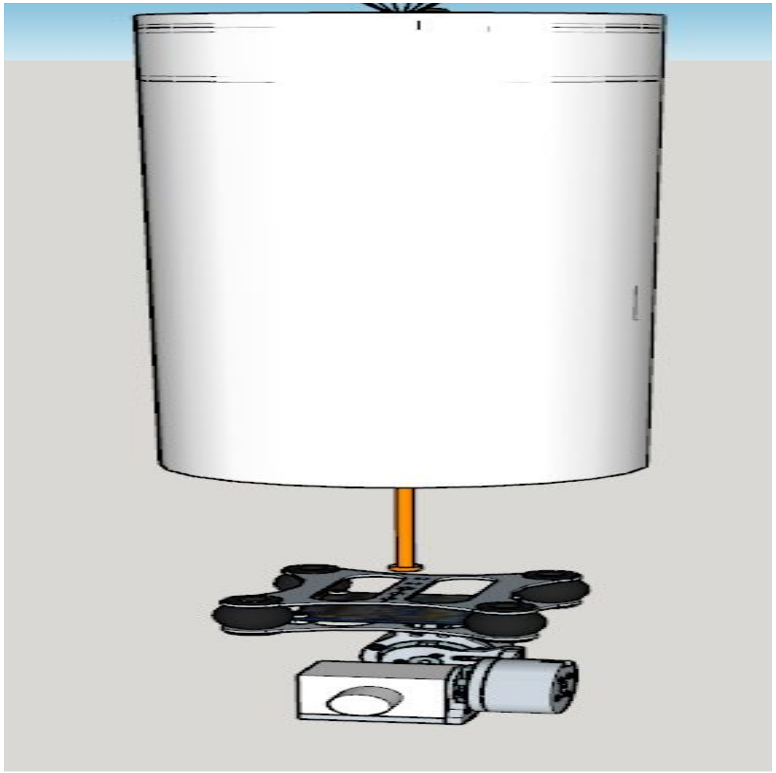
Container Mechanical Layout of Components



Container Payload Mount

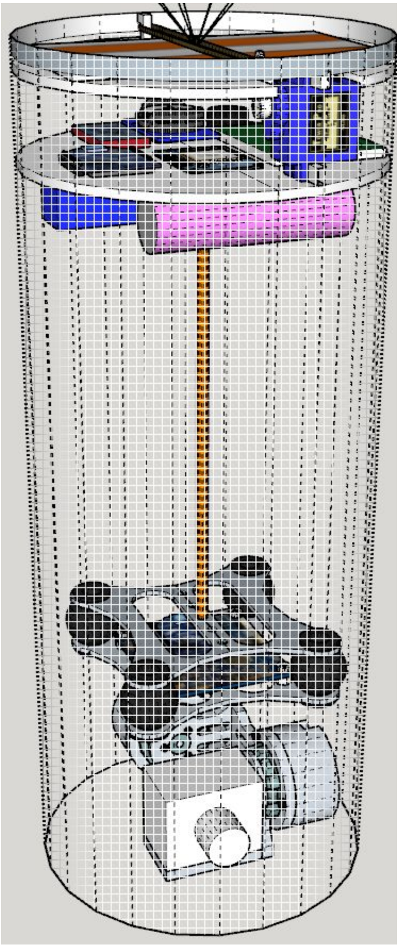
- **The payload is locked in the tether previously slightly extended from the container**

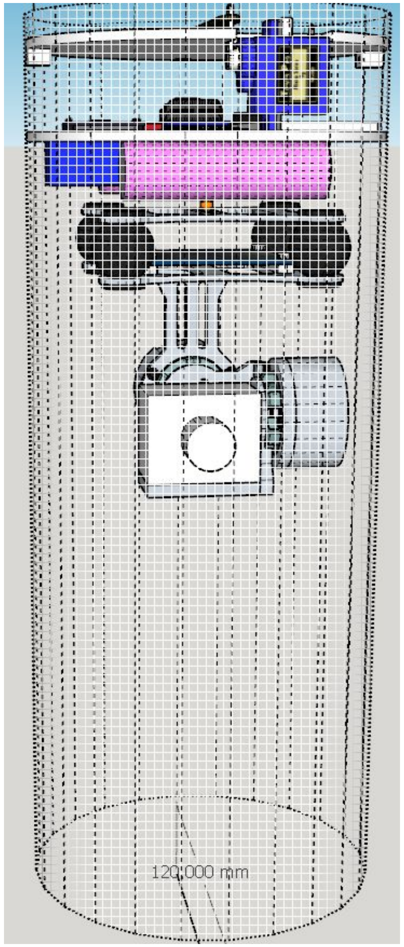




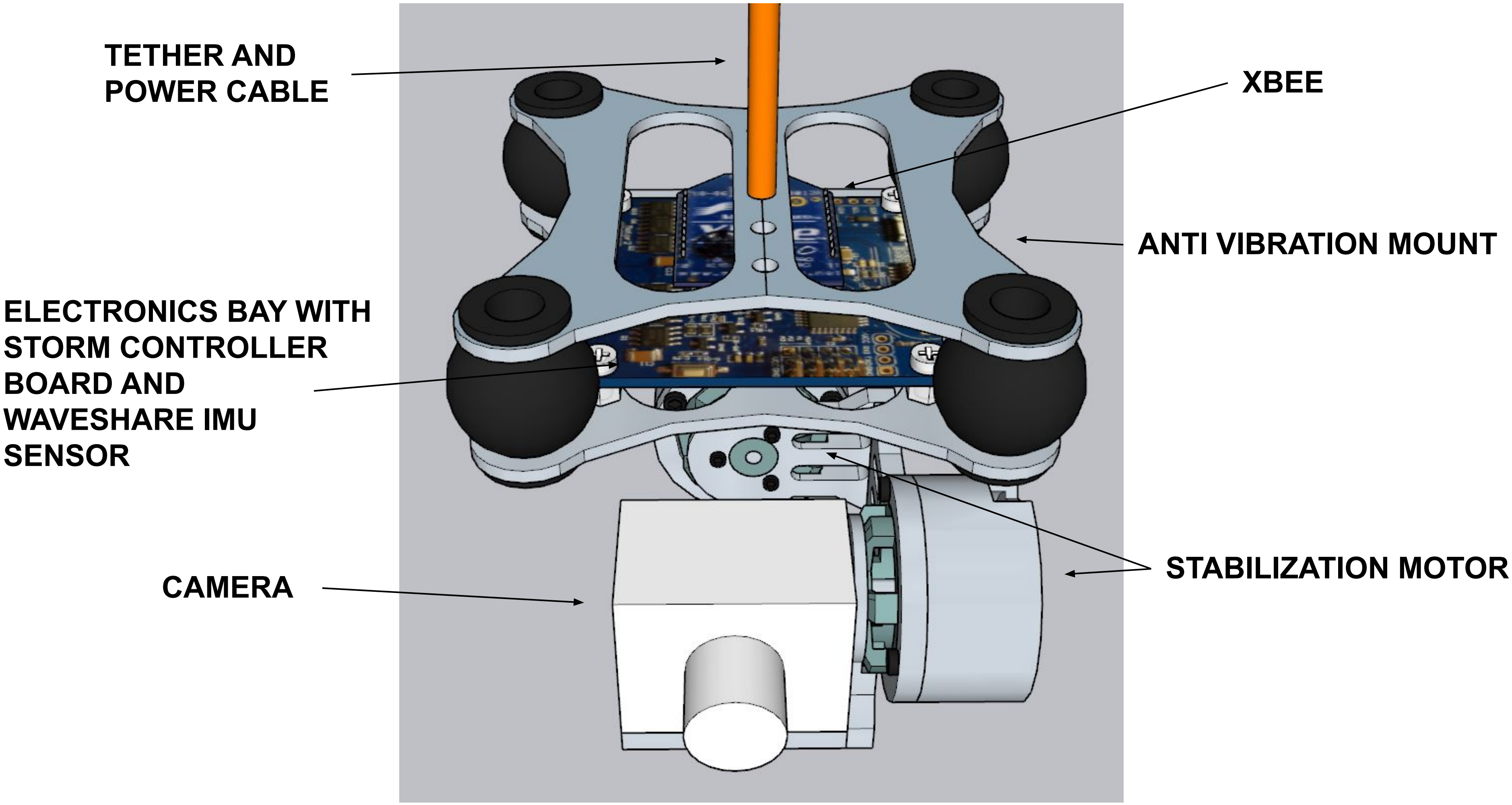
Container Payload Mount

- **Then the servo is activated and the payload is retracted and locked by tension in place**
- **The payload is locked in place by the tension created by the teather and the payload upper plate against the main battery**

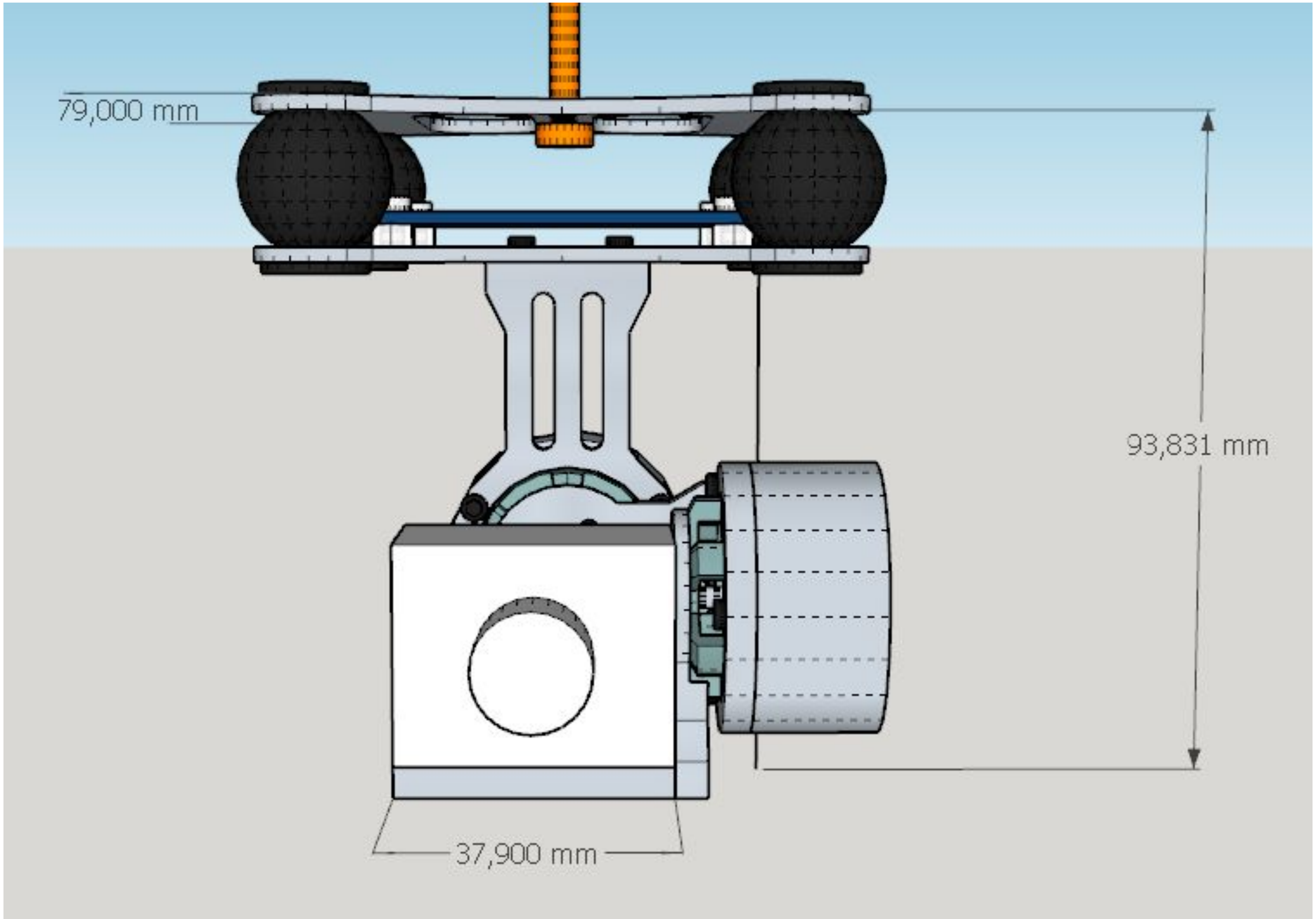




Payload Mechanical Layout of Components

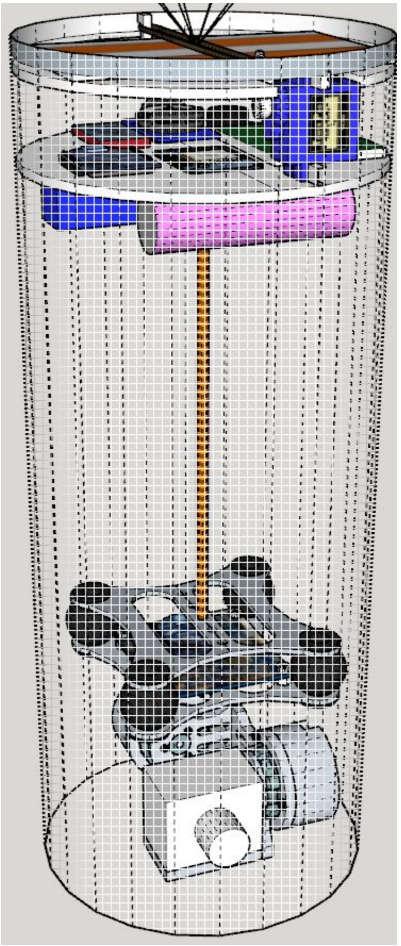


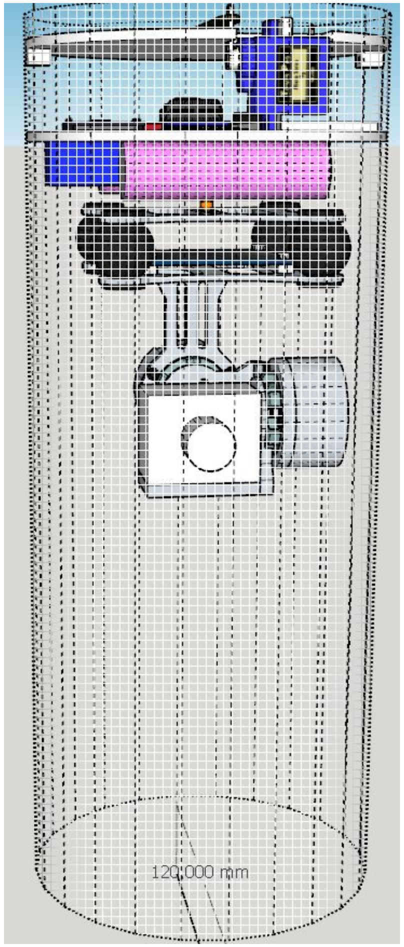
Payload Mechanical Layout of Components



Payload Pre Deployment Configuration

- **Then the servo is activated and the payload is retracted and locked by tension in place**
- **The payload is locked in place by the tension created by the teather and the payload upper plate against the main battery**

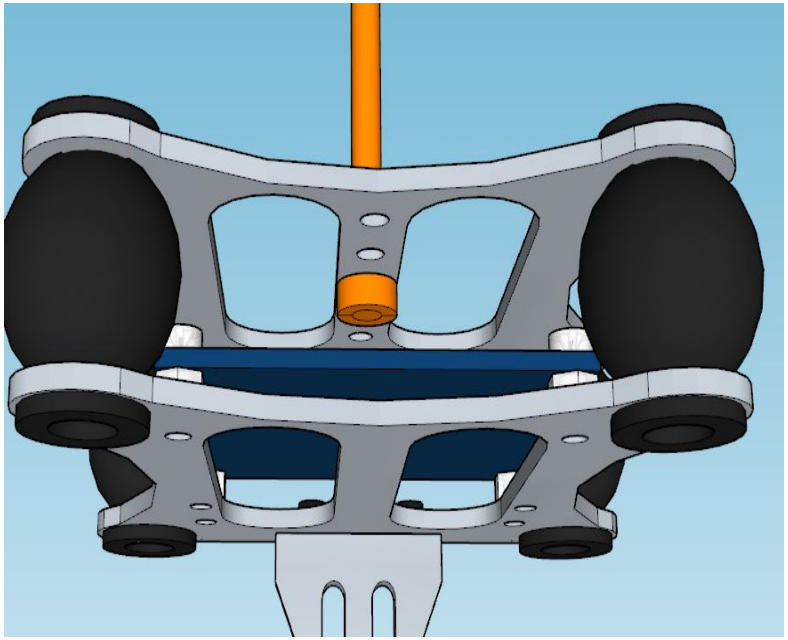


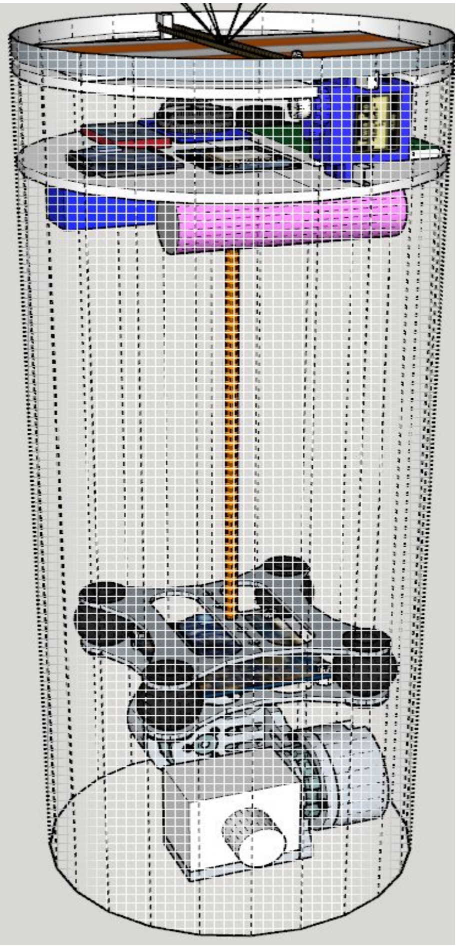


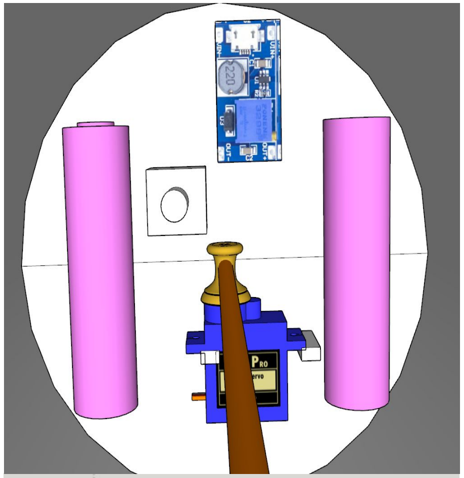
Payload Deployment Configuration

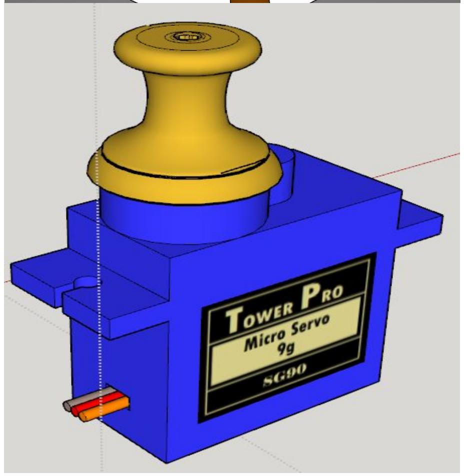
DESCRIPTION:

- **The payload is attached to one end of the tether with a 3D printed bulb of TPU, the same material of the tether and locked with glue and a servo with a spool rolls to tangle or untangle the tether so the payload be deployed or retracted as wish**





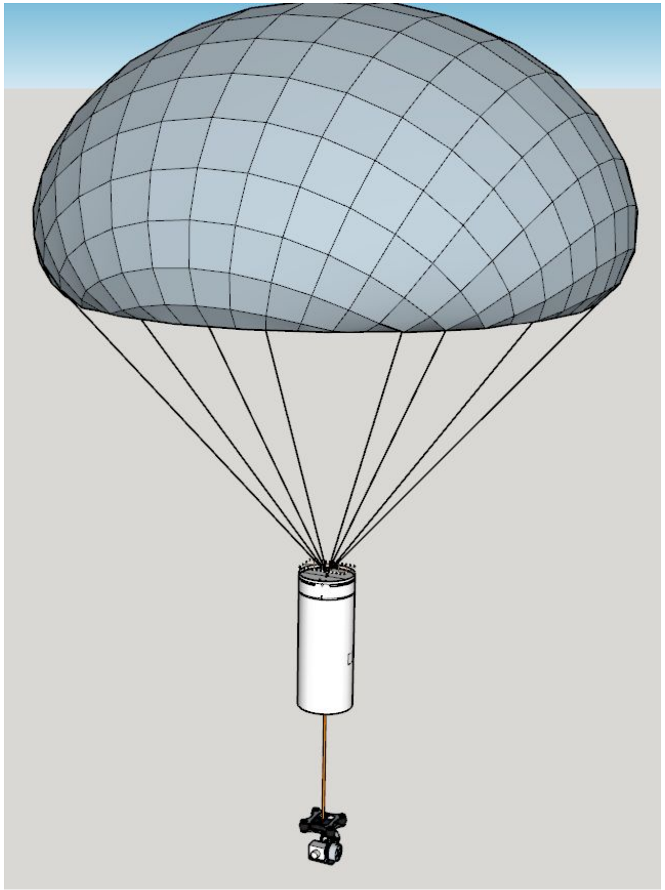




Payload Deployment Configuration

DESCRIPTION:

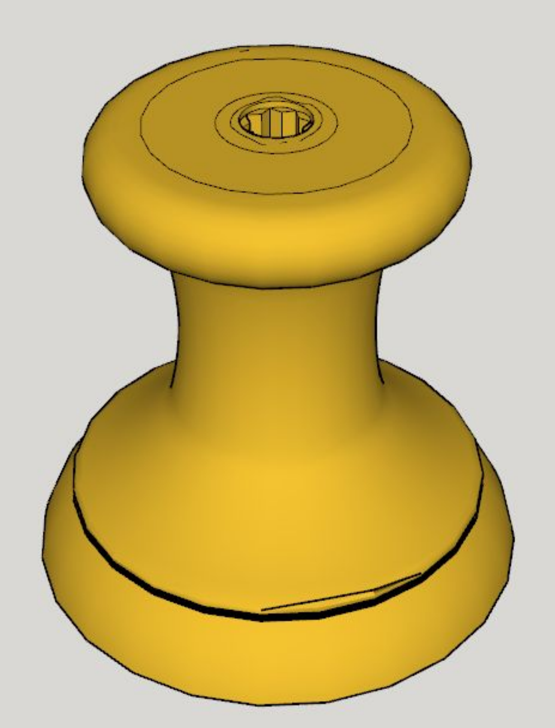
- **The payload is attached to one end of the tether with a 3D printed bulb of TPU, the same material of the tether and locked with glue and a servo with a spool rolls to tangle or untangle the tether so the payload be deployed or retracted as wish**

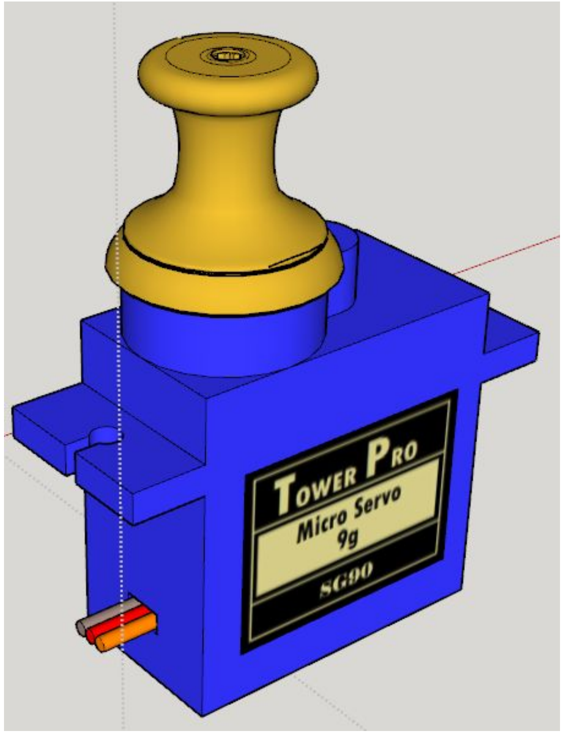


Tether Design Description

Description:

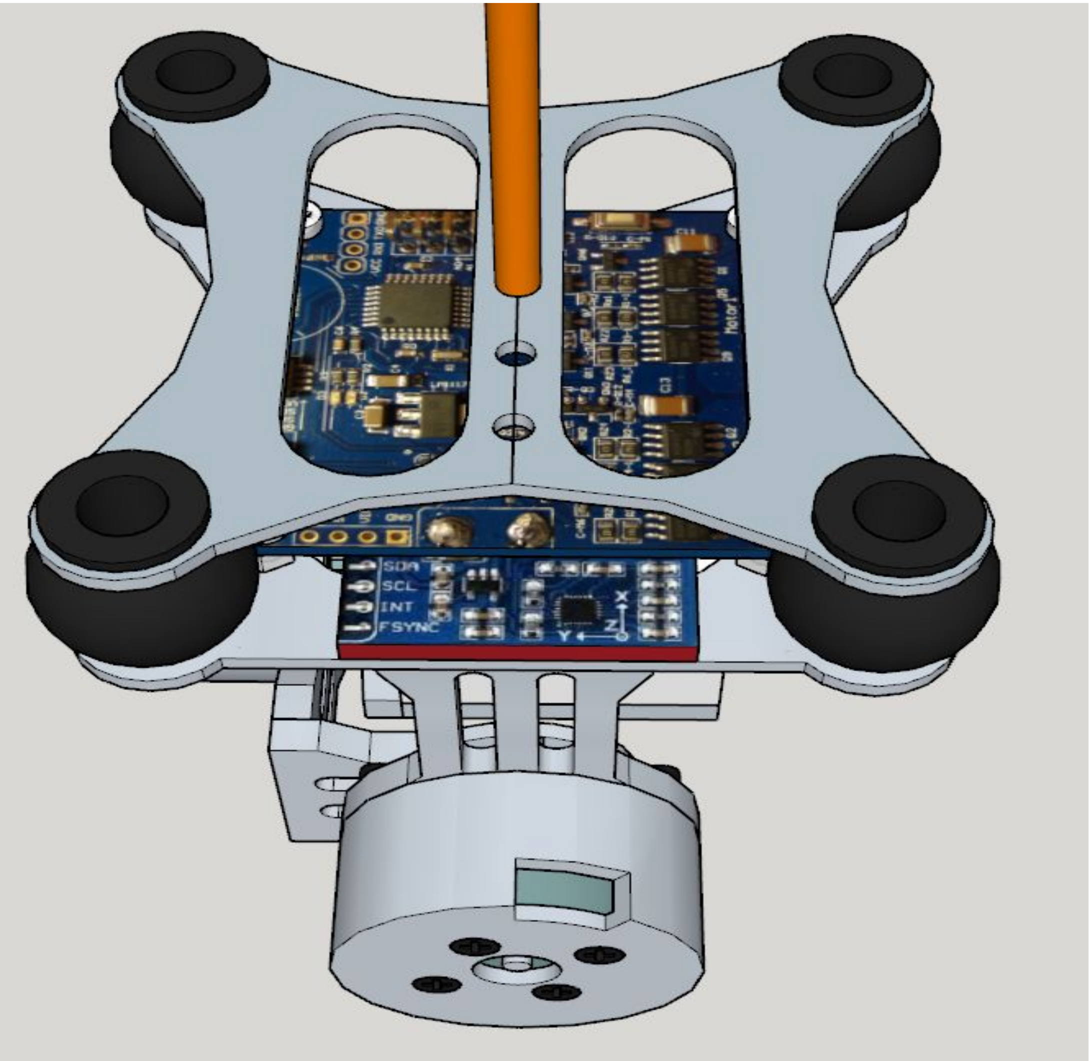
- **A special designed spool is used as servo arm on a modified servo that spins in continuous rotation in both directions that allows the tether to be tangled or untangled. A tiny and very flexible awg 35 cable with shrink tube will be used to give power to the payload, and for the tether material will choose TPU printing filament because is very flexible and can withstand high forces before fail**





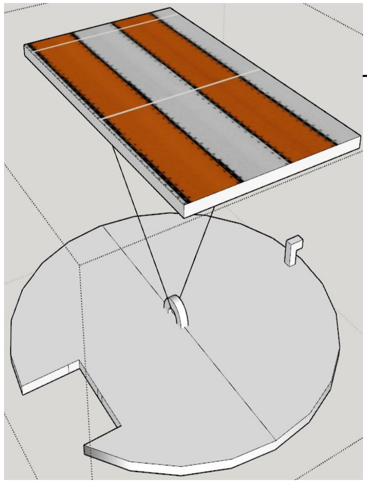
Description:

- **For keeping the descent speed of the payload to the mission requirement in the camera gimbal is a Waveshare IMU sensor board which can measure the altitude very precisely and with a PID algorithm in the arduino we can achieve precise speed under all circumstances accelerating or decelerating the servo rotation**

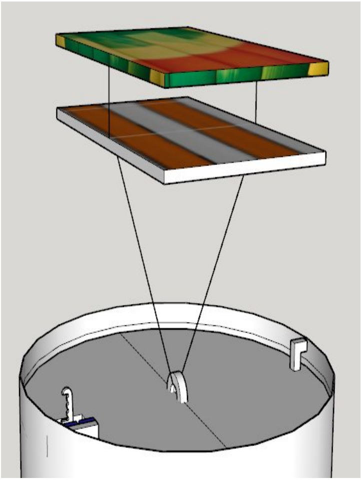


Container Parachute Attachment Mechanism

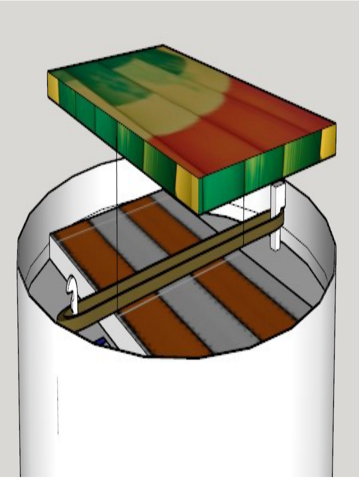
1)The second Parachute is folded and tied to the hook of the parachute bay



2)The parachute bay is secured with screw to the container, the the first parachute is folded and tied to the second one

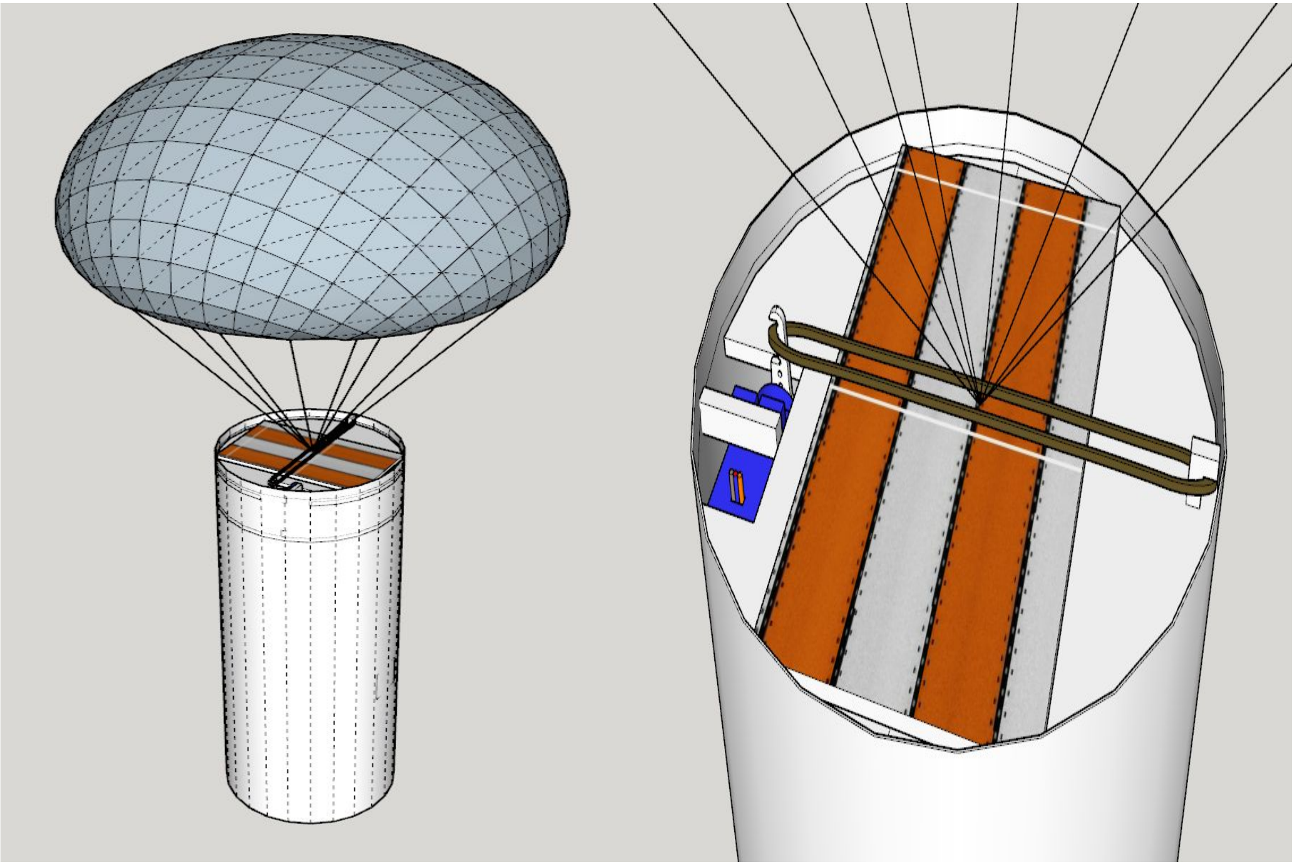


3)The second parachute is locked with a rubber band



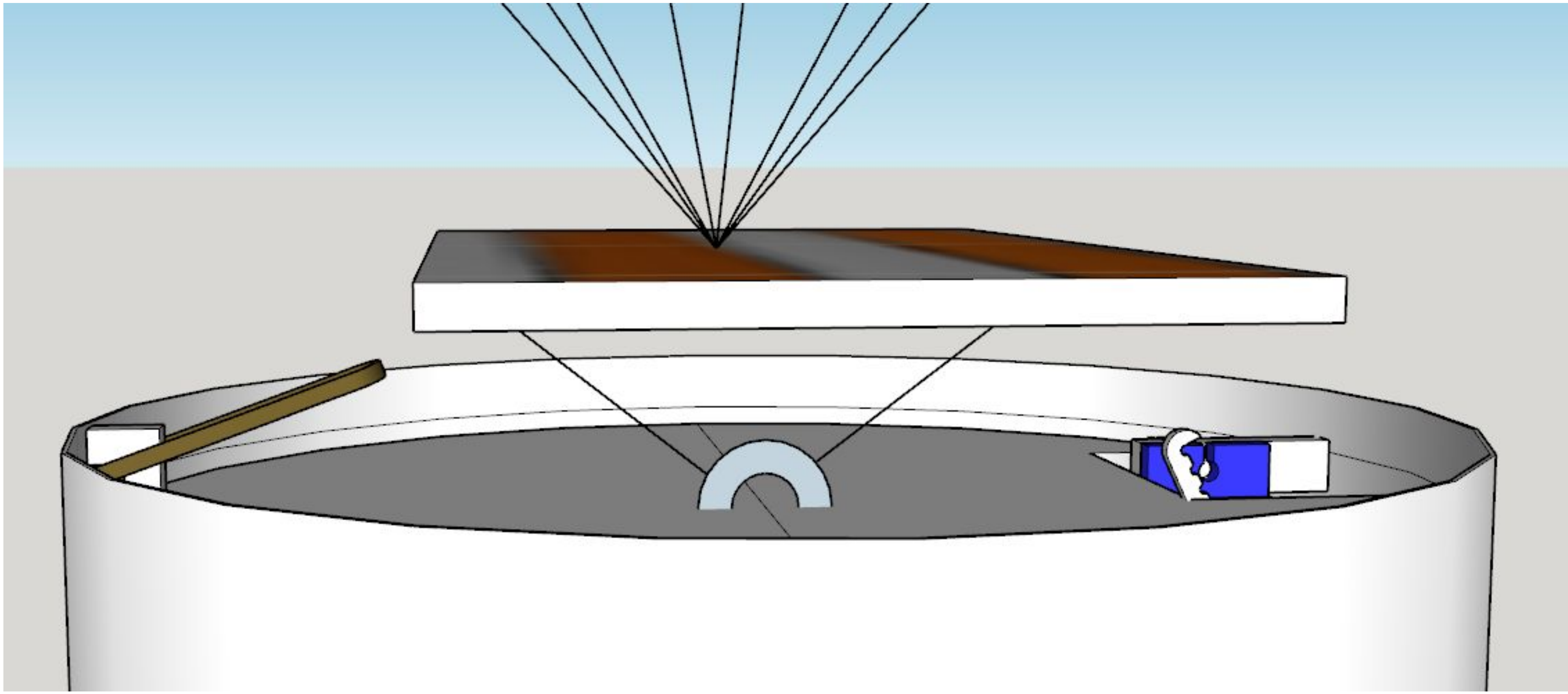
4)The first Parachute is folded and placed upside the second parachute on the top container

Container Parachute Attachment Mechanism



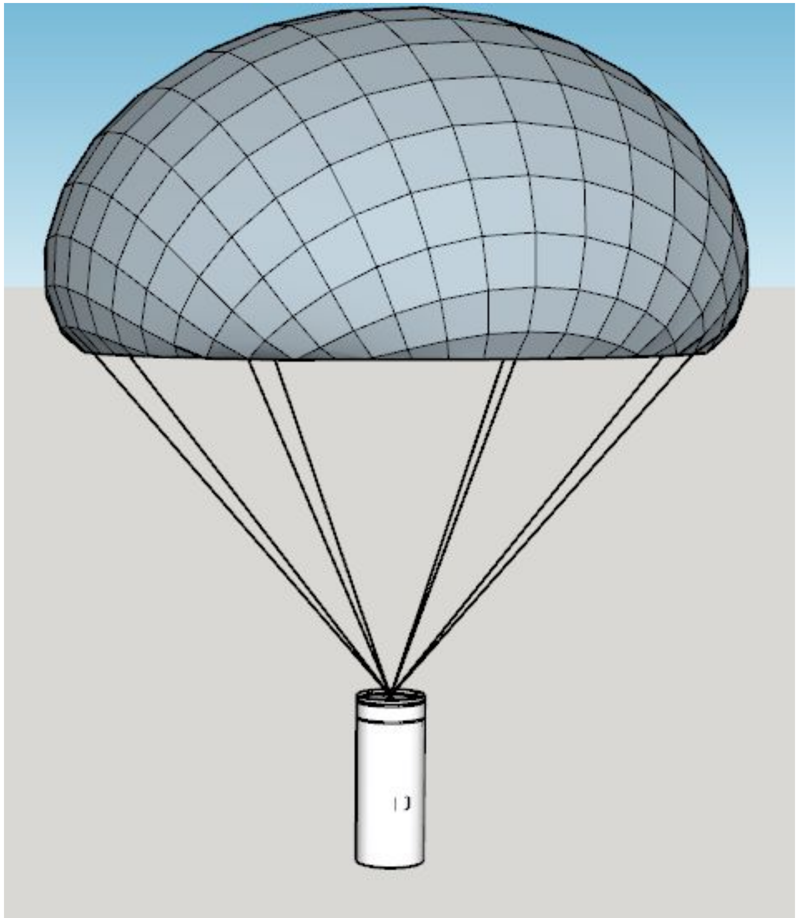
- **When launched the first parachute is deployed and put tension on the second parachute that is locked with the rubber band**

Container Parachute Attachment Mechanism



- **Then the servo releases the rubber band and the first parachute pulls out the second one and because is only tangled in horizontal way to the second the cottom lines releases the first parachute**

Container Parachute Attachment Mechanism



- **Then the second larger parachute fully deploy and the container descend**

Structure Survivability

Securing Electrical Connections

•Depending on the connection components, proper methods for securing will be used.

Electrical connection methods are:

- **Soldering**
- **Cyanoacrylate**
- **Electric tape**
- **Heat shrink tube**
- **hot glue**

PAYLOAD MASS

PART	mass gr	Data type
XBee	40	D
10 DOF IMU Sensor	4	D
Storm32 Controller	2	D
Brushless motor (x3)	93	D
Air frame	80	E
Camera	13	D
TOTAL PAYLOAD MASS	232 gr	

D:Datasheet
E:Estimated

CONTAINER MASS

PART	mass gr	Data type
XBee	40	D
10 DOF IMU Sensor	1	D
Arduino mega 2560 mini	5	D
18650 battery (x2)	96	D
Frame	139,5	E
Container first parachute	2	E
Adafruit Ultimate GPS	8,5	D
Servo (x2)	18	D
Tether	2	E
Buzzer	15	D

CONTAINER MASS

PART	mass gr	Data type
3D printed parts	3	E
Container first parachute	4	E
misc, (cotton wires, screw)	8	E
Real time clock	8	d
Camera	13	D
TOTAL CONTAINER MASS	363 gr	

CANSAT MASS

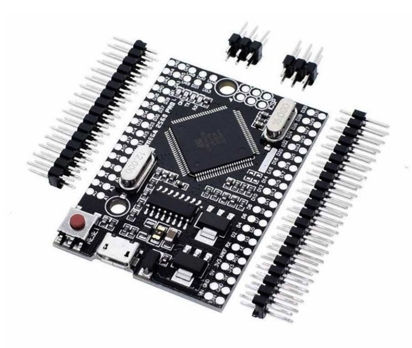
PART	Mass gr
PAYLOAD	232
CONTAINER	363
TOTAL	595

Total mass of the cansat is estimated in 595 gr, to reach the 600 gr mark we can add little plumbs in the container, and if its too heavy we can make holes in the lateral walls of the container and cover them with MONOKOTE to reduce weight

Communication and Data Handling (CDH) Subsystem Design

Jorge Royon
Juan Esteban Yurquina
Juárez Enzo

Container Command Data Handler (CDH) Overview



Arduino Mega 2560 Mini → XBee PRO S2C

Arduino 2560 mega mini: An Arduino mega, is used to get data from sensors, transmit it to the XBee PRO S2C and control the container.

A XBee PRo S2C module is a transceiver. It is used to transmit and receive data between the payload and the ground station.

CDH Changes Since PDR

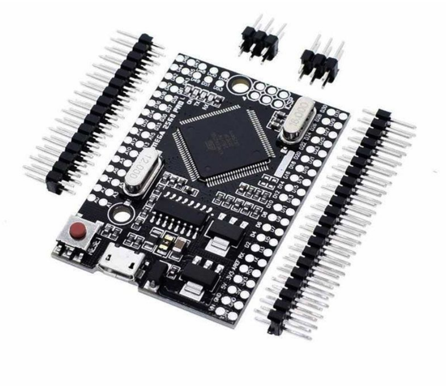
- **We haven’t changes to this section since the submission of the PDR.**

Container Processor & Memory Selection

Arduino Mega 2560 mini

MODEL	BOOT TIME	PROCESSOR SPEED	I/O PINS	INTERFACES	MEMORY STORAGE REQUIRED	SUPPLY VOLTAGE	COST
Arduino Mega 2560 mini	3.50 s	16 MHz	Analog In[16] PWM [14] Digital[70]	I2C [1] SPI [5] UART [4]	Flash-256 KB SRAM-8 KB EEPROM-4 KB	5V-12V	$ 24,93

- **We use Arduino Mega 2560 mini due to availability and performance**



Container Real-Time Clock

Arduino RTC DS3231

MODEL	TYPE	VOLTAGE	CURRENT	ACCURACY	RESET TOLERANCE	COST
Arduino RTC DS 3231	Hardware	3,3V	200 µA	±2 ppm	In reset condition external clock continues keeping time	$ 2,00

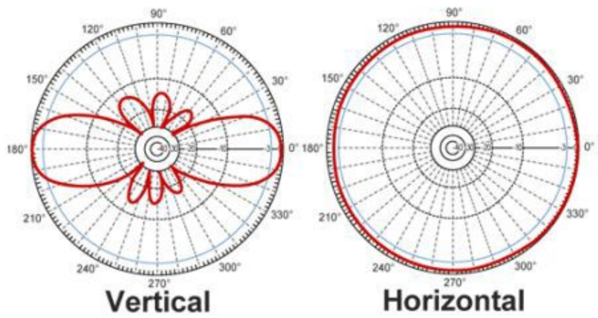
- **We use the Arduino DS3231 because it is cheaper, consumes less energy, and is enough for what we need**

Container Antenna Selection

Generic 2.4GHz

MODEL	CONNECTION TYPE	FREQUENCY	DIRECTION	GAIN	BEAM WIDTH HORIZONTAL/ VERTICAL	RANGE	COST
Generic 2.4GHz	RP - SMA	2.4 GHz	Omnidirectio nal	2dBi	360°/32°	10km	$2.00

We use the generic antenna because we can peel off the plastic cover of the antenna to fit inside the container, plus the xbee's range exceeds the operational limit easily, so not as much gain is needed.Transmit power of XBee S2C is 250 mW 24dBm software selectable.



Container Radio Configuration

- **XBEE radio selection: XBee PRO S2C**
- **NETID: 1064 Container and 6064 for Payload**
- **TRANSMISSION CONTROL:** The XBee in the container is set as Coordinator and the XBee in the ground station is set at Endpoint In all of the mission phases the Coordinator transmits the telemetry data to the Endpoint. In addition, the coordinator in the container receives the data from the modules in the payload while they are in flight and sends it to the ground station.
- **Data rate:** The xbee in the container is set to transmit de data to the ground station at 1 Hz transmission rate and 4 Hz when payload deployed
- Both xbees communicate in unicast mode

Container Telemetry Format

DATA FORMAT	SAMPLE DATA	DESCRIPTION
<TEAM_ID>	1064	Is the assigned team identification.
<MISSION_TIME>	08:12:55	Is UTC time in format hh:mm:ss, where hh is hours, mm is minutes, and ss is seconds.
<PACKET_COUNT>	1485	Is the count of transmitted packets, which is to be maintained through processor reset. One count is used for transmission of all packets, regardless of type.
<PACKET_TYPE>	C	Is the ASCII character ‘C’ for Container telemetry, character ‘T’ for tethered Payload (deployed first) relayed telemetry at 4 Hz.
<MODE>	F	= ‘F’ for flight (the default mode upon system start) and ‘S’ for simulation
TP_RELEASED	R	= ‘N’ for not released and ‘R’ for released

Container Telemetry Format

DATA FORMAT	SAMPLE DATA	DESCRIPTION
<ALTITUDE>	84.2	is the altitude in units of meters and must be relative to ground level. The resolution must be 0.1 meters.
<TEMP>	24.3	is the temperature in degrees Celsius with a resolution of 0.1 degrees C.
<VOLTAGE>	4.45	is the voltage of the CanSat power bus. The resolution must be 0.01 volts.
<GPS_TIME>	08:12:55	is the time generated by the GPS receiver. The time must be reported in UTC and have a resolution of a second.
<GPS_LATITUDE>	25.0458	is the latitude generated by the GPS receiver in decimal degrees with a resolution of 0.0001 degrees.
<GPS_LONGITUDE>	-160.5804	is the longitude generated by the GPS receiver in decimal degrees with a resolution of 0.0001 degrees.

Container Telemetry Format

DATA FORMAT	SAMPLE DATA	DESCRIPTION
<GPS_ALTITUDE>	84.2	is the altitude generated by the GPS receiver in meters above mean sea level with a resolution of 0.1 meters.
<GPS_SATS>	6	is the number of GPS satellites being tracked by the GPS receiver. This must be an integer number.
<SOFTWARE STATE>	LAUNCH_WAIT	iis the operating state of the software. (e.g., LAUNCH_WAIT, ASCENT, ROCKET_SEPARATION, DESCENT, TP_RELEASE, LANDED, etc.). Teams may define their own states.
<CMD_ECHO>	CMD1000SIMP1 0 1325	is the fixed text command id and argument of the last received command with no commas.

Container Telemetry Format

- **Data format**
  <TEAM_ID>, <MISSION_TIME>, <PACKET_COUNT>, <PACKET_TYPE>, <MODE>, <TP_RELEASED>, <ALTITUDE>, <TEMP>, <VOLTAGE>, <GPS_TIME>, <GPS_LATITUDE>, <GPS_LONGITUDE>, <GPS_ALTITUDE>, <GPS_SATS>, <SOFTWARE_STATE>, <CMD_ECHO>

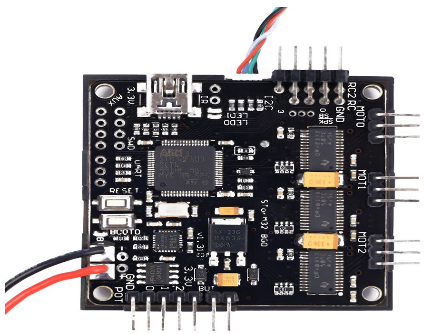
- **Telemetry data file name for the Container:**
  - for Container: Fligth_1064_C.csv

- Data from payload shall transmit continuously at 4Hz sample rate

Container Command Formats

COMMAND	DESCRIPTION	DATA INCLUDED	DATA SAMPLE
CX	Container Telemetry On/Off Command	<ON_OFF> is the string 'ON' to activate the Container telemetry transmissions and 'OFF' to turn off the transmissions.	CMD,1064,CX,ON
ST	Set Time	<UTC_TIME> is UTC time in the format hh:mm:ss where hh is hours, mm is the minutes and ss is the seconds.	CMD,1064,ST,13:35:59
SIM	Simulation Mode Control Command	<MODE> is the string 'ENABLE' to enable the simulation mode, 'ACTIVATE' to activate the simulation mode, or 'DISABLE' which both disables and deactivates the simulation mode.	CMD,1064,SIM,ENABLE
SIMP	Simulated Pressure Data	<PRESSURE> is the simulated atmospheric pressure data in units of pascals with a resolution of one Pascal.	CMD,1064,SIMP,101325

Payload Command Data Handler (CDH) Overview



Storm32 Control Board → **XBee³**

A Storm32 Control Board, is used to get data from sensors and transmit it from the XBee³ to the container.

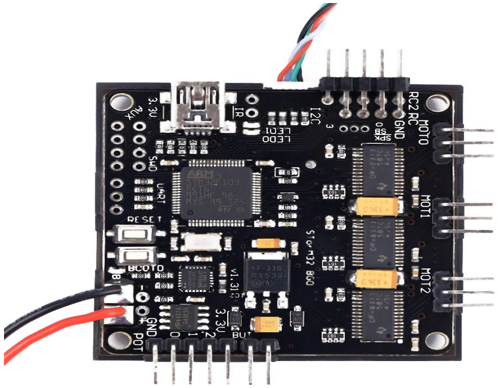
An XBee³ module is used to transmit data to the payload and to the ground station.

Payload Processor & Memory Selection

Storm32

MODEL	BOOT TIME	PROCESSOR SPEED	I/O PINS	INTERFACES	MEMORY STORAGE REQUIRED	SUPPLY VOLTAGE	COST
Storm32	4.84 s	16 MHz	Analog In[8] PWM [6] Digital[62]	I2C [1] SPI [1] UART [1]	Flash-32 KB SRAM-2 KB EEPROM-1 KB	5V-9V	$ 22,00

We use the Arduino Storm32 it is more compact and cheaper. It is not as complete as the Arduino Mega mini, but for what we need in the payload it is enough.

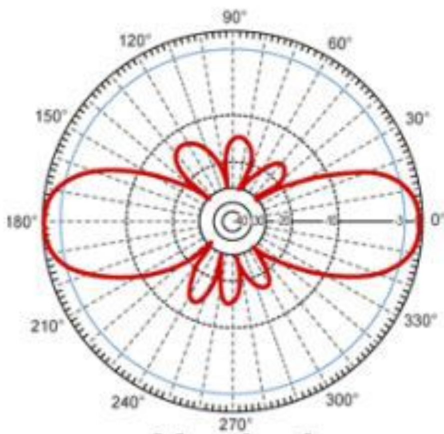


Payload Antenna Selection

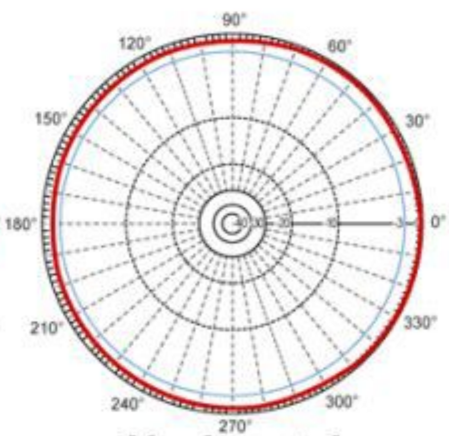
Antenna of XBee³ module

MODEL	CONNECTION TYPE	FREQUENCY	DIRECTION	GAIN	BEAM WIDTH HORIZONTAL/ VERTICAL	RANGE	COST
Antenna of XBee³ module	-	2.4 GHz	Omnidirectional	2dBi	360°/32°	10km	$0.00

We chose to use the antenna that is included in the XBee module because in this way we reduce costs and and it is enough to cover our needs.



Vertical



Horizontal

Payload Radio Configuration

- **XBEE radio selection: XBee³**
- **NETID: 6064**
- **TRANSMISSION CONTROL:** The XBEEs in the Payload are set as Endpoint and the XBee in the container is set at Coordinator In all of the mission phases the Endpoints transmit the telemetry data only to the Coordinator, while they are in flight and then, the Coordinator sends it to the ground station.
- **Data rate:** The XBEEs in the Payload are set to transmit de data to the container at 4 Hz transmission rate
- All the XBEE communicate in unicast mode

Payload Telemetry Format

DATA FORMAT	SAMPLE DATA	DESCRIPTION
<TEAM_ID>	1064	Is the assigned team identification.
<MISSION_TIME>	13:15:55	Is UTC time in format hh:mm:ss, where hh is hours, mm is minutes, and ss is seconds.
<PACKET_COUNT>	4237	Is the count of transmitted packets, which is to be maintained through processor reset. One count is used for transmission of all packets, regardless of type.
<PACKET_TYPE>	T	is the ASCII character 'C' for Container telemetry, character 'T' for Tethered Payload relay telemetry, and character 'X' for optional custom packets.
<TP_ALTITUDE>	84.1	Is the altitude in units of meters and must be relative to ground level. The resolution must be 0.1 meters.
<TP_TEMP>	28.5	Is the measured temperature in degrees Celsius with a resolution of 0.1 degrees C.

Payload Telemetry Format

DATA FORMAT	SAMPLE DATA	DESCRIPTION
<TP_VOLTAGE>	4.45	is the voltage of the CanSat power bus. The resolution must be 0.01 volts.
<GYRO_R>	1	are the gyro readings in degrees per second for the roll, pitch, and yaw axes
<GYRO_P>	1	are the gyro readings in degrees per second for the roll, pitch, and yaw axes
<GYRO_Y>	1	are the gyro readings in degrees per second for the roll, pitch, and yaw axes
<ACCEL_R>	2	are the accelerometer readings for the roll, pitch and yaw axes.
<ACCEL_P>	2	are the accelerometer readings for the roll, pitch and yaw axes.

Payload Telemetry Format

DATA FORMAT	SAMPLE DATA	DESCRIPTION
<ACCEL_Y>	2	are the accelerometer readings for the roll, pitch and yaw axes.
<MAG_R>	1	are magnetometer readings in the roll, pitch and yaw axes in gauss.
<MAG_P>	1	are magnetometer readings in the roll, pitch and yaw axes in gauss.
<MAG_Y>	1	are magnetometer readings in the roll, pitch and yaw axes in gauss.
<POINTING_ERROR>	1	is the yaw pointing error in degrees. Zero degrees is due South.
<TP_SOFTWARE_STATE>	STANDBY	is the operating state of the Tethered Probe software. (e.g., STANDBY, RELEASED, ACQUIRING_TARGET, TARGET_POINTING, etc.). Teams may define their own states.

Payload Telemetry Format

- **Data format**

  TEAM_ID, MISSION_TIME, PACKET_COUNT, PACKET_TYPE, TP_ALTITUDE, TP_TEMP, TP_VOLTAGE, GYRO_R, GYRO_P, GYRO_Y, ACCEL_P, ACCEL_P, ACCEL_Y, MAG_R, MAG_P, MAG_Y, POINTING_ERROR, TP_SOFTWARE_STATE

- **Telemetry data file name:**
  - for the Container Telemetry: Flight_1064_C.csv
  - for the Tethered Payload: Flight_1064_T.csv

Electrical Power Subsystem Design

Juárez Enzo

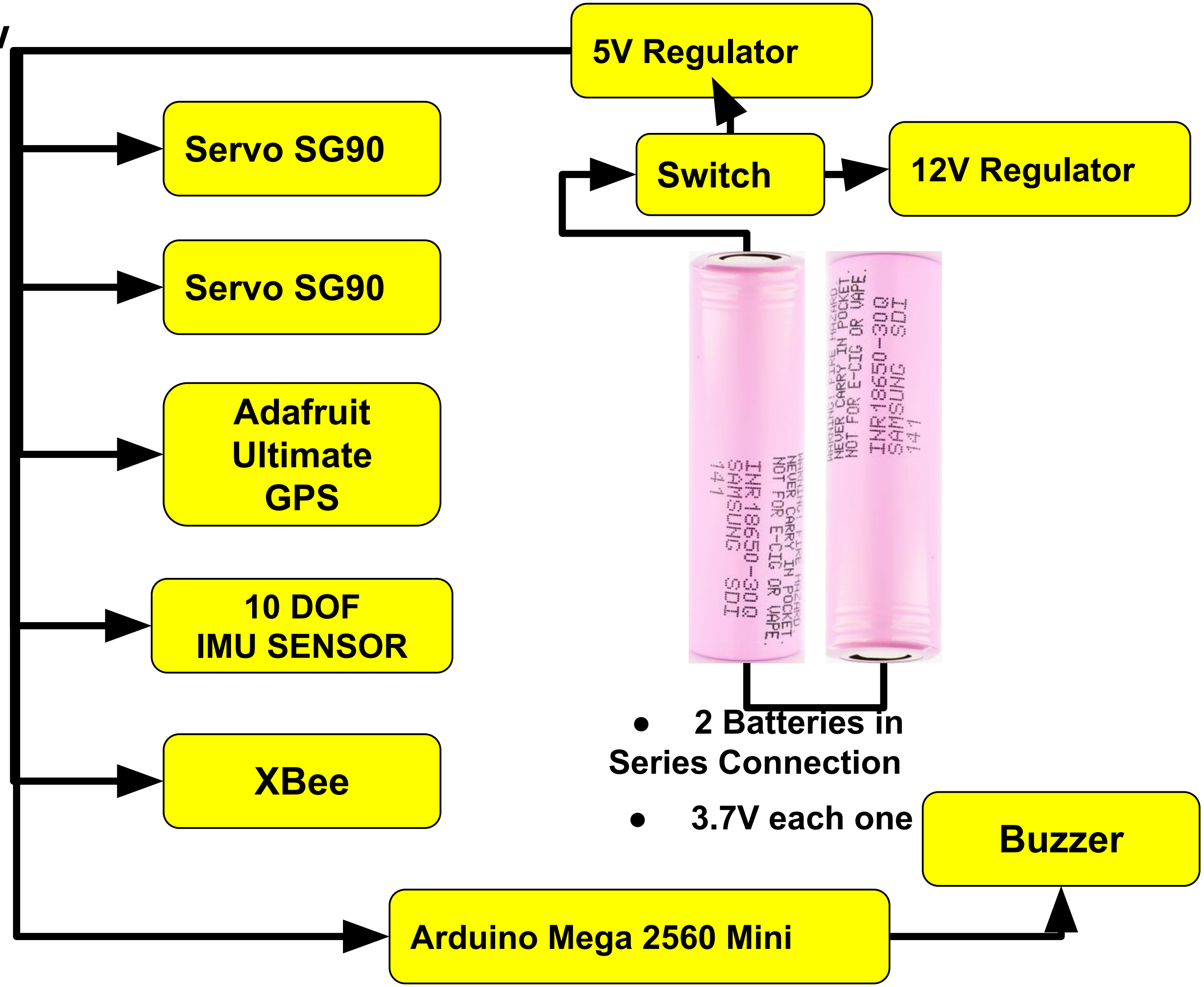
EPS Overview (1 of 4)

Component	Purpose
Battery Samsung 30Q 18650	Power Source
XBee	Radio Communicator Container
XBee	Radio Communicator Ground Station
Adafruit Ultimate GPS	GPS Sensor
Micro Servo SG90 1.5 Kg	Deploy 1st container parachute
Micro Servo SG90 1.5 Kg	Deploy 2nd container parachute
Mini Spy Camera (x2)	Record the descent and bonus camera
Switch	ON/OFF the Cansat
Buzzer	Audio Beacon

EPS Overview (2 of 4)

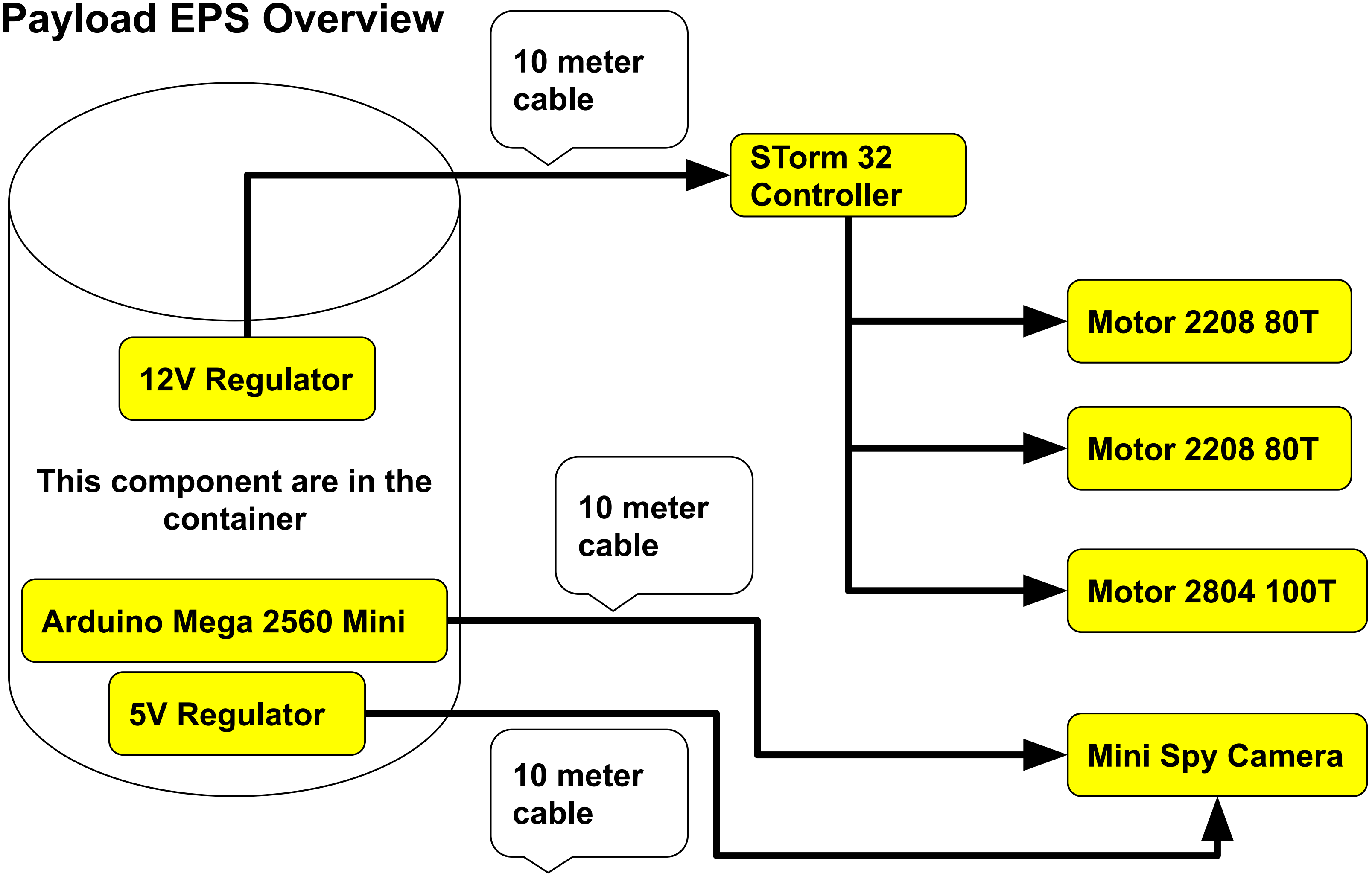
Component	Purpose
STorm32 Controller	Gimbal Camera and Motor Controller
Motor 2208 80T	Move and stabilize Camera
Motor 2208 80T	Move and stabilize Camera
Motor 2804 100T	Move and stabilize Camera
10 DOF IMU Sensor	Stabilization, Navigation, temperature and pressure sensor
Module Step UP 2A DC-DC XY-016	Gimbal Controller Power Supply Stabilizer
AMS1117 5.0 HT242E	Provides clean 5V Supply
Arduino Mega 2560 Mini	Microcontroller Based Control Board

Container EPS Overview



EPS Overview (4 of 4)

Payload EPS Overview

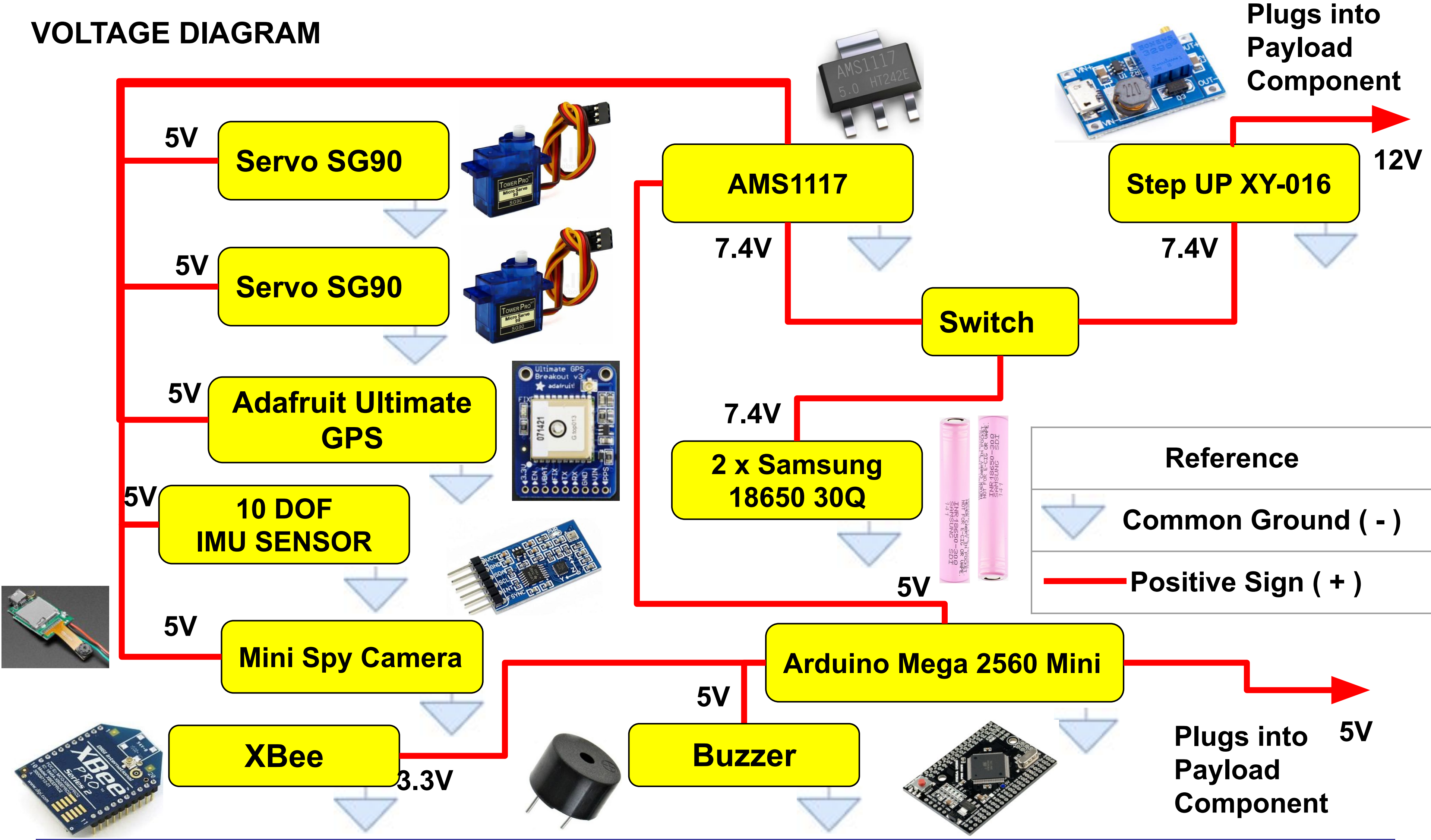


EPS Changes Since PDR

- **We haven't changes to this section since the submission of the PDR.**

Container Electrical Block Diagram (1 of 2)

VOLTAGE DIAGRAM

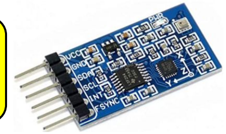


INTERFACE DIAGRAM

Servo SG90

Servo SG90

Adafruit Ultimate GPS



10 DOF IMU SENSOR

XBee

Arduino Mega 2560 Mini

Plugs into Payload Component

Mini Spy Camera

Buzzer

Reference
Control Sign (~)

Arduino Mega 2560 Mini is our microprocessor and control the state, the data and behaviour to all our components in the container and the camera in the Payload.

Samsung 30Q 18650 Battery

NAME / MODEL	TYPE	Nominal Voltage (V)	Capacity (mAh)	Weight (g)	Max Continuous Discharging Rate (A)	Price ($)	Source
Samsung 30Q 18650 Battery	Lithium Ion	3.6	3000	48	30	7.99	Datasheet

- **We will use 2 x Samsung 30Q 18650 Battery will be used in SERIE configuration, where we will obtain 7.2V of Nominal Voltage and 3000mAh of Nominal Current.**
- **Both Samsung 30Q 18650 will mounted on container using battery socket**
- **It has capability to deliver enough current and power for the container**

Container Power Budget (1 of 2)

NO	COMPONENT	Voltage (V)	Current (mA)	Duty Cycles (Hr)	Power Consumption (Wh)	Source
1	**Servo SG90 N°1**	5	650	0.016	0.052	Datasheet
2	**Servo SG90 N°2**	5	650	0.016	0.052	Datasheet
3	**Arduino Mega 2560 Mini**	5	0.51	2	0.0051	Datasheet
4	**Xbee Module**	3.3	23	2	0.1518	Datasheet
5	**Adafruit Ultimate GPS**	5	20	2	0.2	Datasheet
6	**10 DOF IMU Sensor**	5	42	2	0.42	Datasheet
7	**Step Up XY-016**	7.4	26	2	0.385	Estimate
8	**AMS1117**	5	0.2	2	0.12	Datasheet
9	**Buzzer**	5	500	0.016	0.04	Datasheet
10	**Adafruit Mini Spy Camera**	5	110	2	1.1	Datasheet

Total Power Consumed in Container (for Two Hours)
2.5259 Wh

Available Total Current Capacity (for Two Hours)	Available Total Power Capacity (for Two Hours)
3000mAh	21.6Wh

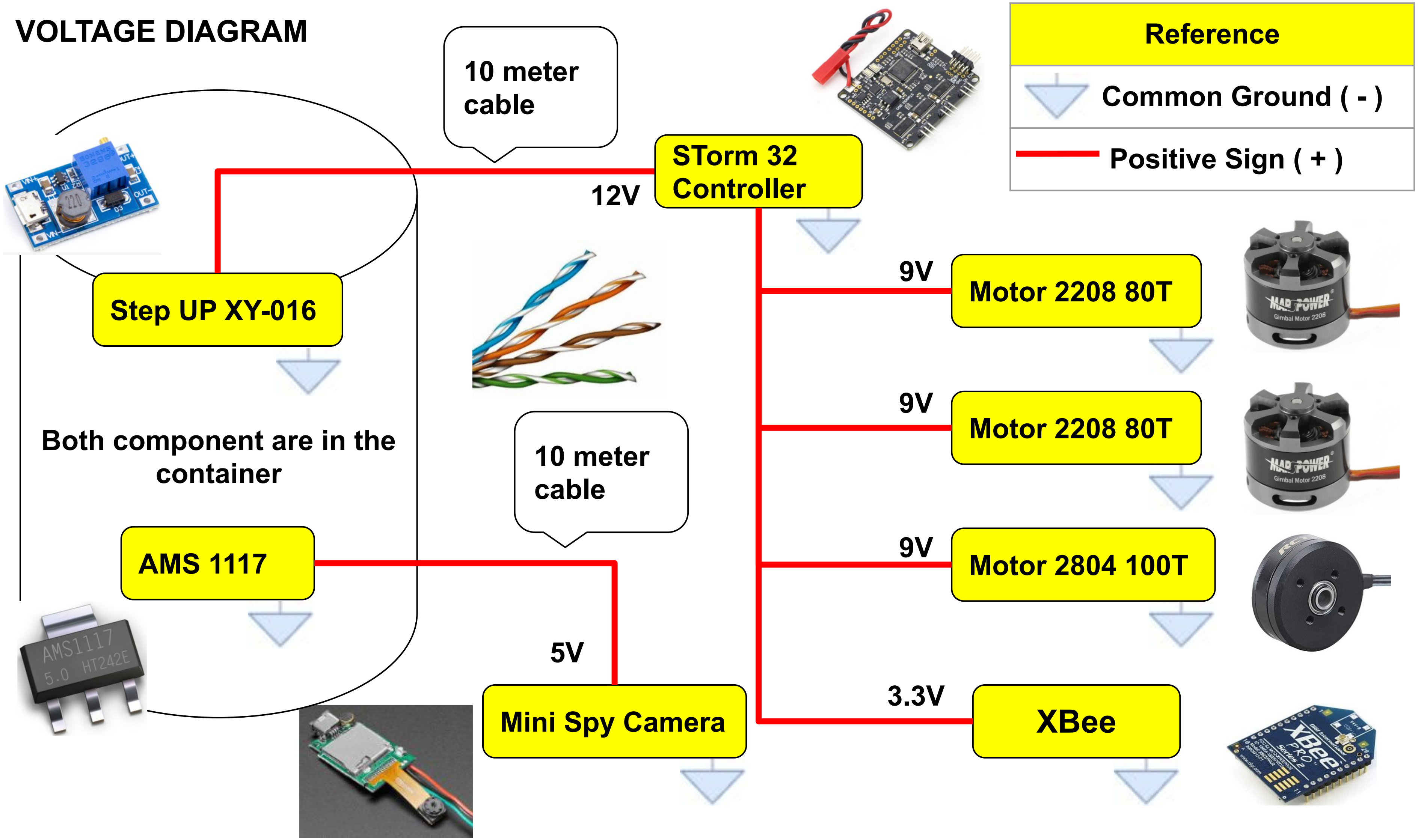
Margin of Power Consumption (for Two Hours)
21.6 Wh - 2.5259 Wh = 19.0741 Wh (Margin 88.3%)

Conclusion: Our Battery is sufficient to power all component through the flight

We will add a final slide adding the consumption calculations of the container and the Payload, since they use the same battery as a source and the final calculation will be better represented there.

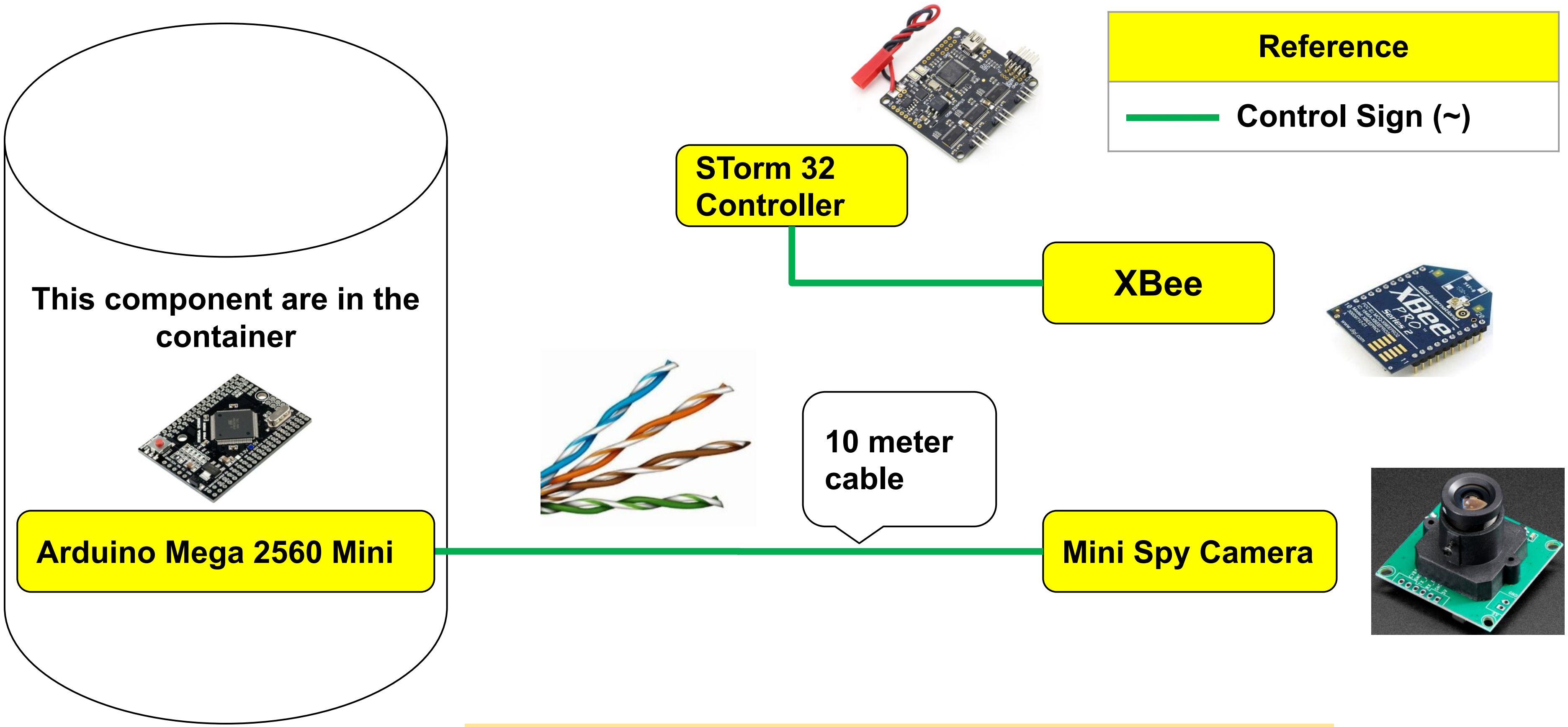
Payload Electrical Block Diagram (1 of 2)

VOLTAGE DIAGRAM



Payload Electrical Block Diagram (1 of 2)

INTERFACE DIAGRAM



The Arduino Mega 2560 Mini tells the camera when to start recording and when to end

Payload Power Source

Samsung 30Q 18650 Battery

NAME / MODEL	TYPE	Nominal Voltage (V)	Capacity (mAh)	Weight (g)	Max Continuous Discharging Rate (A)	Price ($)	Source
Samsung 30Q 18650 Battery	Lithium Ion	3.6	3000	48	30	7.99	Datasheet

- **We will use 2 x Samsung 30Q 18650 Battery will be used in SERIE configuration, where we will obtain 7.2V of Nominal Voltage and 3000mAh of Nominal Current.**
- **Both Samsung 30Q 18650 will mounted on container using battery socket**
- **It has capability to deliver enough current and power for the container**

Payload Power Budget (1 of 2)

NO	COMPONENT	Voltage (V)	Current (mA)	Duty Cycles (Hr)	Power Consumption (Wh)	Source
1	STorm32 Controller	12	15	2	0.36	Estimate
2	Motor 2208 80T	7.4	30	2	0.44	Datasheet
3	Motor 2208 80T	7.4	30	2	0.44	Datasheet
4	Motor 2804 100T	7.4	7.4	2	0.44	Datasheet
5	Adafruit Mini Spy Camera	5	110	2	1.1	Datasheet
6	Module XBee	3.3	23	2	0.1518	Datasheet

Payload Power Budget (2 of 2)

Total Power Consumed in Payload (for Two Hours)
2.932 Wh

Available Total Current Capacity (for Two Hours)	Available Total Power Capacity (for Two Hours)
3000mAh	21.6Wh

Margin of Power Consumption (for Two Hours)
21.6 Wh - 2.932 Wh = 18.668 Wh (Margin 86.42%)

Conclusion: Our Battery is sufficient to power all component through the flight

We will add a final slide adding the consumption calculations of the container and the Payload, since they use the same battery as a source and the final calculation will be better represented there.

Container and Payload Power Budget

Total Power Consumed in Container (for Two Hours)
2.5259 Wh

Total Power Consumed in Payload (for Two Hours)
2.932 Wh

Available Total Current Capacity (for Two Hours)	Available Total Power Capacity (for Two Hours)
3000mAh	21.6Wh

Margin of Power Consumption (for Two Hours)
21.6 Wh - 2.526Wh - 2.932 Wh = 16.142 Wh (Margin 74.73%)

Final Conclusion

Our Battery is sufficient to power all component through the flight and can operate Two Hours without problems.

Flight Software (FSW) Design

Juárez Enzo

FSW Overview (1 of 2)

Programming Languages:

- C/C++

Development environments:

- Arduino IDE

Summary of FSW tasks:

- To collect sensors data and convert it to a value that complies with CanSat requirements.
- Send data to the Ground control station and store it to an onboard backup EEPROM.
- Monitor altitude changes to trigger payload releases.
- Activate audio beacon after landing.
- Activate mechanisms in the correct time
- Ensure telemetry frequency is 1Hz and 4 Hz when payload is released
- Design a software architecture such that the processor can fully recover from a sudden power loss that might occur in any time during the mission and for arbitrary duration

FSW Overview (2 of 2)

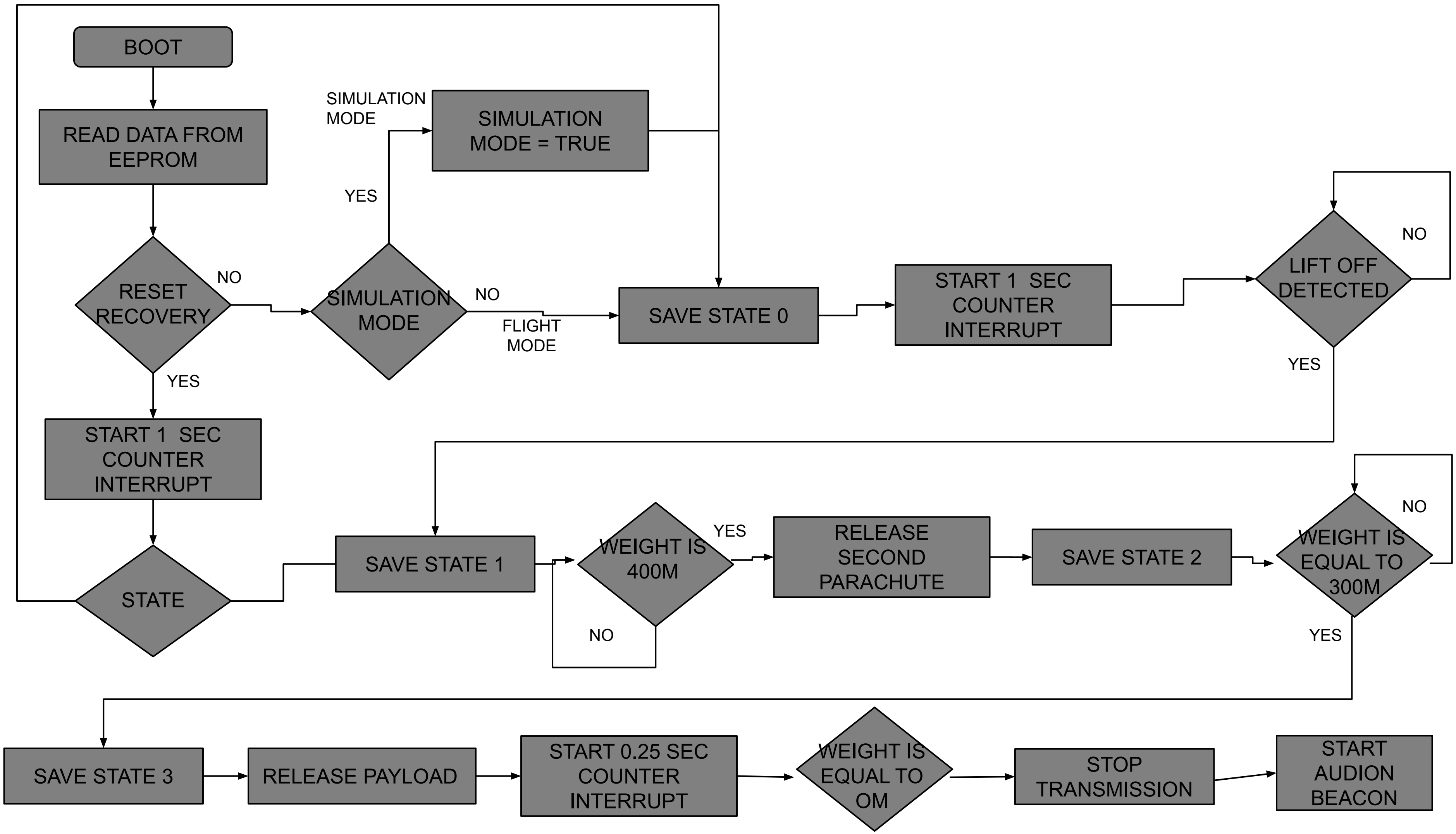
FSW Task:

Container	Payload
To collect sensors data and convert it to a value that complies with CanSat requirements.	**To collect sensors data and convert it to a value that complies with CanSat requirements.**
Send data to the Ground control station and store it to an onboard backup EEPROM.	**Send data to the container and store it to an onboard backup EEPROM.**
Monitor altitude changes to trigger payload releases	**Ensure telemetry frequency is 4Hz**
Activate audio beacon after landing	**Recover from a sudden power loss that might occur in any time during the mission and for arbitrary duration**
Activate mechanisms in the correct time	
Ensure telemetry frequency is 1Hz	

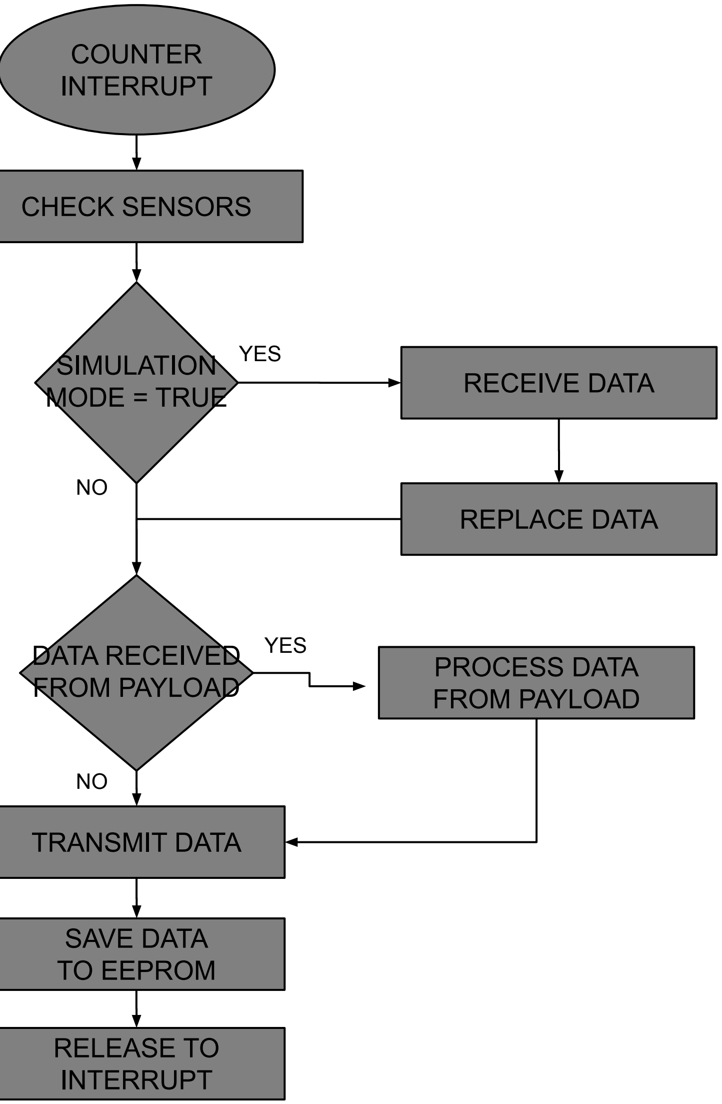
FSW Changes Since PDR

- We changed the flowcharts, designing them directly in the interface, since in the scoresheet they mentioned that they were not well appreciated.

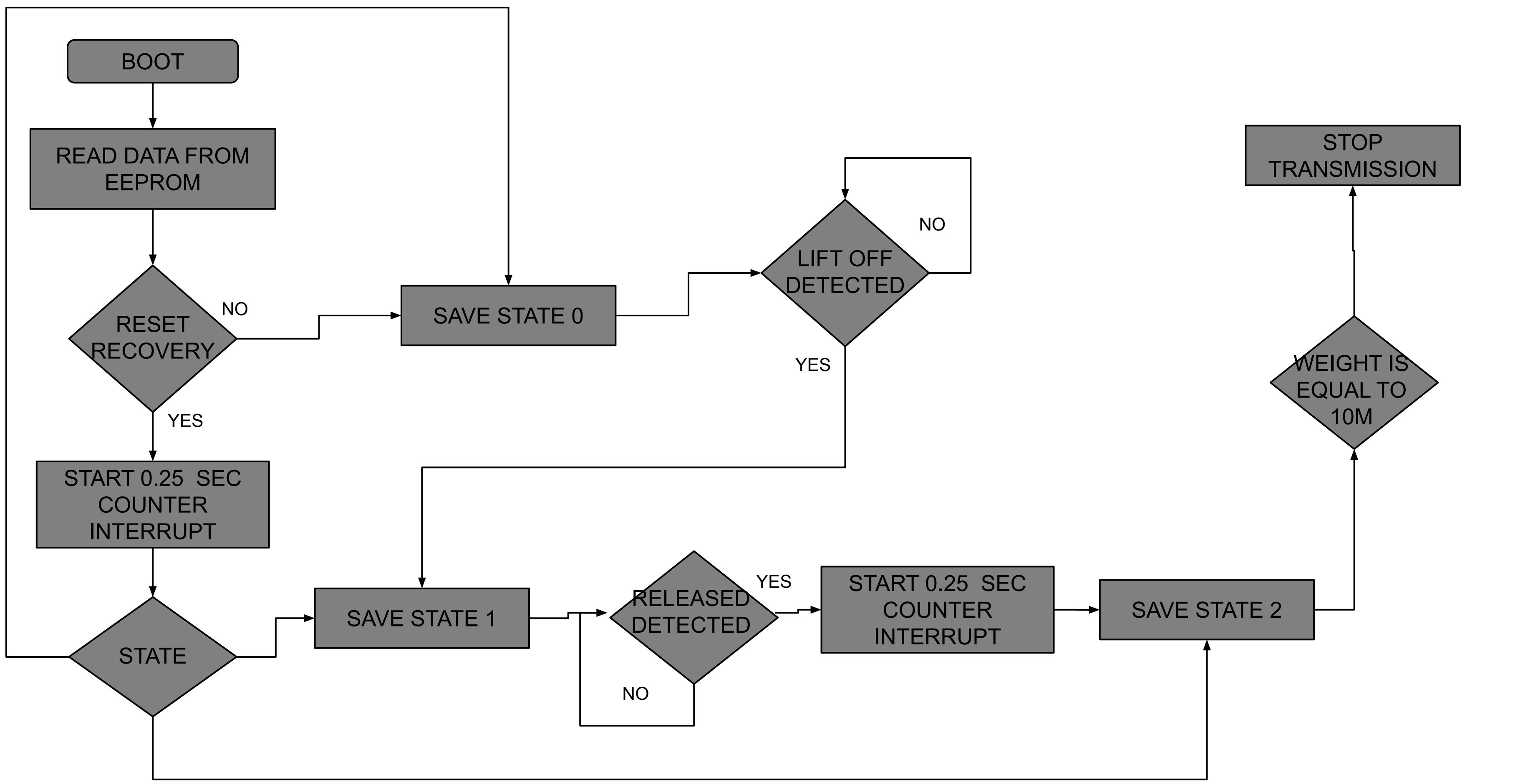
Container CanSat FSW State Diagram



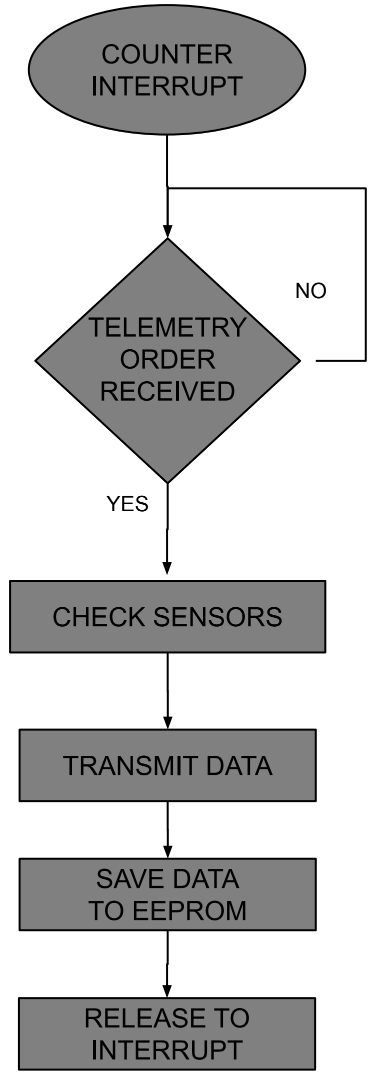
Container CanSat FSW State Diagram



Payload CanSat FSW State Diagram



Payload CanSat FSW State Diagram



Simulation mode

The ground station will receive a file with the simulated data, then the data is transmitted to the container where the container firmware would calculate the altitude with the supplied data, then it will be formatted to complain requirements and transmitted to the ground station

Simulation commands

The simulation commands are:

- CMD,<TEAM_ID>,SIM,<MODE>

  Where:

  1. CMD and SIM are static text.
  2 .<TEAM_ID> is the assigned team identification.
  3.<MODE> is the string ‘ENABLE’ to enable the simulation mode, ‘ACTIVATE’ to activate the simulation mode, or ‘DISABLE’ which both disables and deactivates the simulation mode.

  Example: Both the CMD,1000,SIM,ENABLE and CMD,1000,SIM,ACTIVATE commands are required to begin simulation mode.

SIMP - Simulated Pressure Data (to be used in Simulation Mode only)

- CMD,<TEAM ID>,SIMP,<PRESSURE>

  Where:

  1. CMD and SIMP are static text.
  2.<TEAM_ID> is the assigned team identification.
  3.<PRESSURE> is the simulated atmospheric pressure data in units of pascals with a resolution of one Pascal.

Example: CMD,1000,SIMP,101325 provides a simulated pressure reading to the Container (101325 Pascals = approximately sea level).

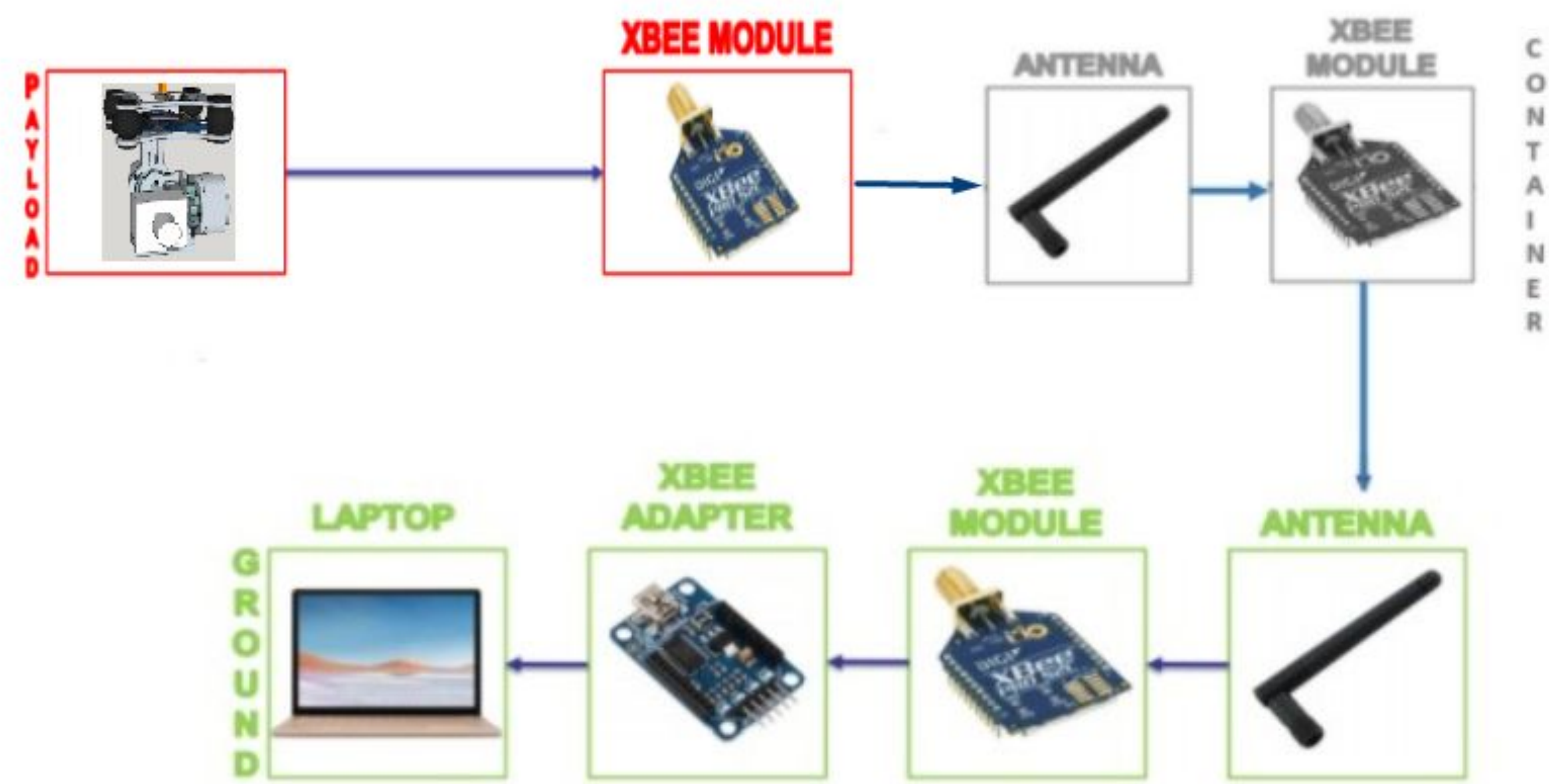
Software Development Plan

Manager/s	Component	Description	Order
Ivan Maccio and Mateo Figallo	Ground station	Program interface	Simultaneous
		Program Xbee	Simultaneous
Jorge Royon and Esteban Yurquina	Container	Program arduino for each sensor	First
		Servo module	Simultaneous Second
		Simulation module	Simultaneous Second
Jorge Royon and Enzo Juarez	Payload	Program Xbee	First
		Program sensors	Second
All Team	Final integration	Full test of all function	Final task

Ground Control System (GCS) Design

Ivan Maccio
Mateo Figallo

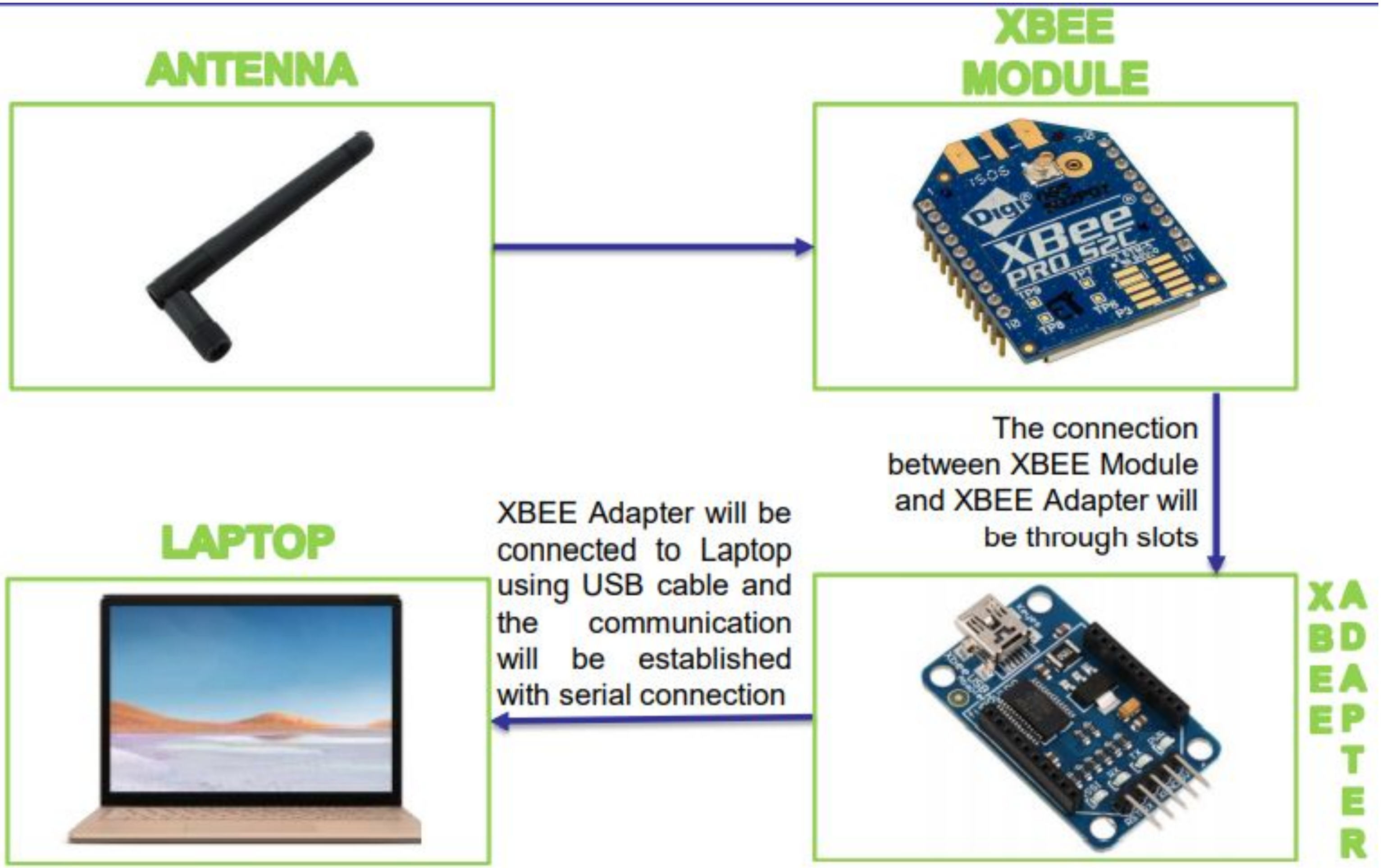
GCS Overview



GCS Changes Since PDR

- **No major changes since PDR**

GCS Overview



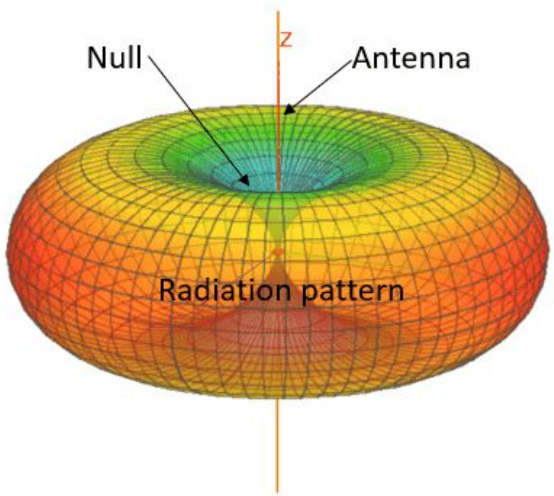
GCS Design

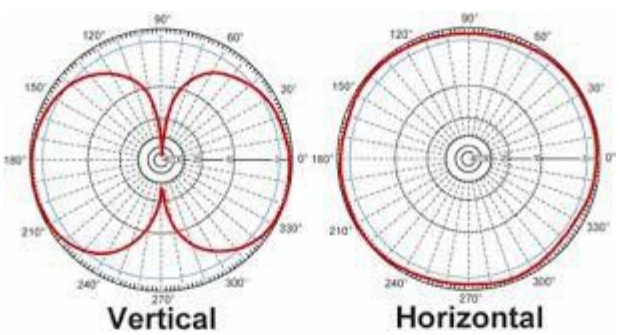
- **Specifications**

- Battery: The Ground Control Station Laptop can operate 3 hours with battery.
- Overheating Mitigation: There will be an umbrella to avoid direct sun exposure
- Auto update mitigation: The option of auto update will be disabled before the launch

GCS Antenna

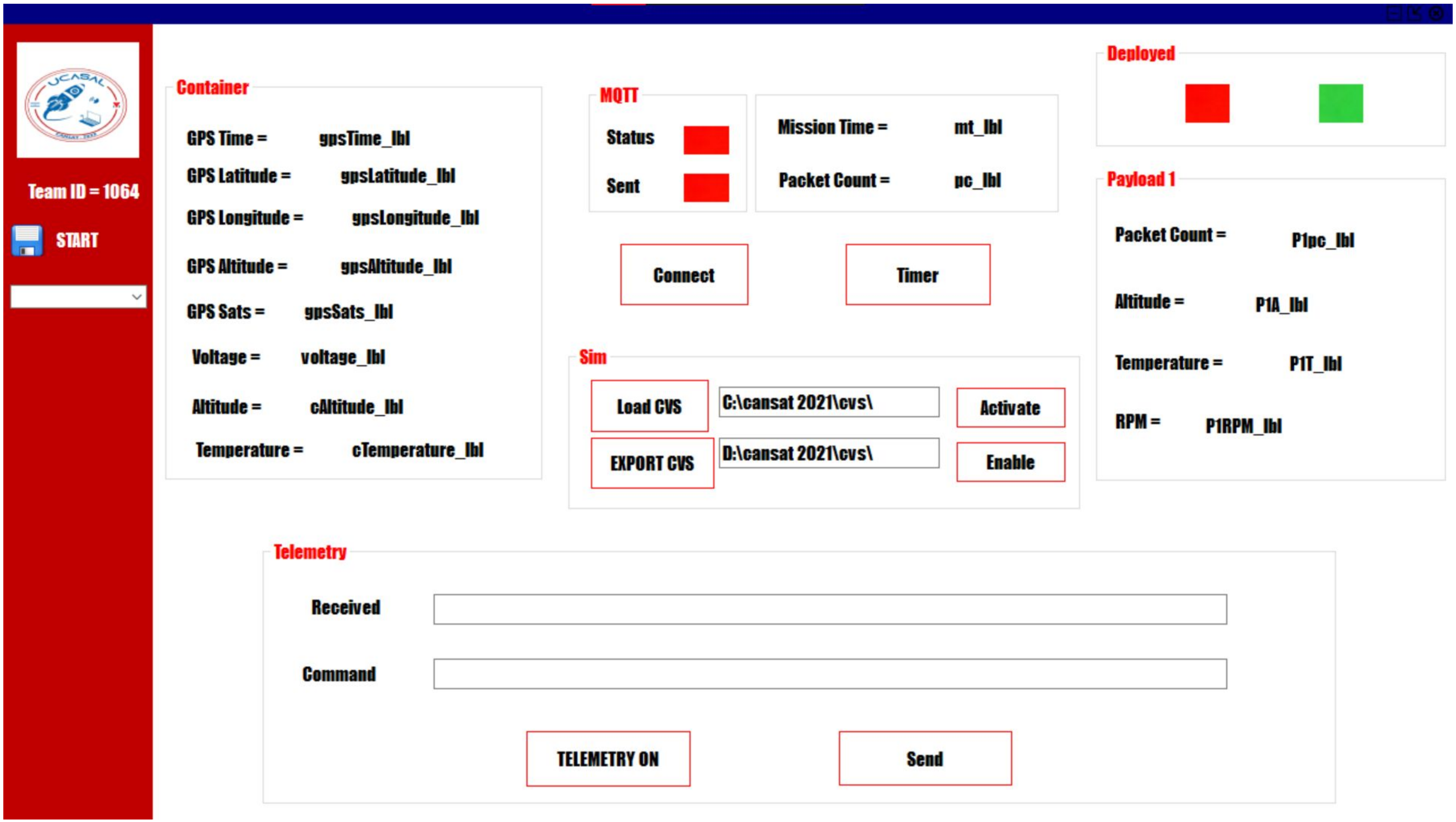
Model	Connection Type	Frequency	Direction	Gain	Beam Width Horizontal / Vertical	Price
Bingfu Dual Band WiFi 2.4GHz	RP-SMA	2,4 ghz	Omnidirectional	9dBi	360°/ 32°	$ 7,99





GCS Software

- **Telemetry display screen shots**



GCS Software

Sample CSV file from GCS

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1																		
2	TeamId	MissionTir	PacketCou	PacketTyp	Mode	Sp1Releas	Altitude	Temperatu	Voltage	GpsTime	GpsLatituc	GpsLongit	GpsAltituc	GpsSats	SoftwareState	Sp1Packet	CmdEcho	
3	1064	4:02:08	22	C	F	FALSE	0	20.4	3.42	4:58:47	0	0	0	0	LAUNCH_WAIT	0	CXON	
4	1064	4:02:09	23	C	F	FALSE	0	20.4	3.17	4:58:48	0	0	0	0	LAUNCH_WAIT	0	CXON	
5	1064	4:02:09	23	C	F	FALSE	0	20.4	3.17	4:58:48	0	0	0	0	LAUNCH_WAIT	0	CXON	
6	1064	4:02:10	24	C	F	FALSE	0	20.4	2.51	4:58:49	0	0	0	0	LAUNCH_WAIT	0	SIM_ENABLE	
7	1064	4:02:10	24	C	F	FALSE	0	20.4	2.51	4:58:49	0	0	0	0	LAUNCH_WAIT	0	SIM_ENABLE	
8	1064	4:02:11	25	C	F	FALSE	0	20.4	2.31	4:58:50	0	0	0	0	LAUNCH_WAIT	0	SIM_ENABLE	
9	1064	4:02:11	25	C	F	FALSE	0	20.4	2.31	4:58:50	0	0	0	0	LAUNCH_WAIT	0	SIM_ENABLE	
10	1064	4:02:12	26	C	F	FALSE	0	20.4	2.29	4:58:51	0	0	0	0	LAUNCH_WAIT	0	SIM_ENABLE	
11	1064	4:02:12	26	C	F	FALSE	0	20.4	2.29	4:58:51	0	0	0	0	LAUNCH_WAIT	0	SIM_ENABLE	
12	1064	4:02:13	27	C	F	FALSE	0	20.4	2.46	4:58:52	0	0	0	0	LAUNCH_WAIT	0	SIM_ENABLE	
13	1064	4:02:13	27	C	F	FALSE	0	20.4	2.46	4:58:52	0	0	0	0	LAUNCH_WAIT	0	SIM_ENABLE	
14	1064	4:02:43	57	C	S	FALSE	0	20.3	2.49	4:59:22	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
15	1064	4:02:43	57	C	S	FALSE	0	20.3	2.49	4:59:22	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
16	1064	4:02:44	58	C	S	FALSE	0	20.3	2.76	4:59:23	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
17	1064	4:02:44	58	C	S	FALSE	0	20.3	2.76	4:59:23	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
18	1064	4:02:45	59	C	S	FALSE	0	20.3	2.83	4:59:24	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
19	1064	4:02:45	59	C	S	FALSE	0	20.3	2.83	4:59:24	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
20	1064	4:02:46	60	C	S	FALSE	0	20.3	3.2	4:59:25	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
21	1064	4:02:46	60	C	S	FALSE	0	20.3	3.2	4:59:25	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
22	1064	4:02:47	61	C	S	FALSE	0	20.3	3.33	4:59:26	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
23	1064	4:02:47	61	C	S	FALSE	0	20.3	3.33	4:59:26	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
24	1064	4:02:48	62	C	S	FALSE	0	20.3	3.59	4:59:27	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
25	1064	4:02:48	62	C	S	FALSE	0	20.3	3.59	4:59:27	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
26	1064	4:02:49	63	C	S	FALSE	0	20.3	3.65	4:59:28	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	
27	1064	4:02:49	63	C	S	FALSE	0	20.3	3.65	4:59:28	0	0	0	0	LAUNCH_WAIT	0	SIM_ACTIVATE	

- **Commercial off the shelf (COTS) software packages used**
  - MATLAB GUI Package for Visual Studio 2015

- **Command software and interface**

The Ground Station will send the following commands from the container:

- CX - Container Telemetry On/Off Command
    - Generic one: CMD,<TEAM_ID>,CX,<ON_OFF>
    - Example of command to be sent: CMD,1231,CX,ON, where:
        - CMD and CX are static text
        - ON is set to activate the Container telemetry transmissions. OFF would be use to turn off the transmissions
- ST - Set Time
    - Generic one: CMD,<TEAM_ID>,ST,<UTC_TIME>
    - Example of command to be sent: CMD,1231,ST,13:35:59, where:
        - CMD and ST are static text
        - 13:35:59 is the parameter -in UTC format- which sets the mission time to the value given

- **Simulation mode**

The Ground Station will send the following commands from the Ground Station:

- SIM - Simulation Mode Control Command
  - Generic one: CMD,<TEAM_ID>,SIM,<MODE>
  - Example of command to be sent: CMD,1064,SIM,ENABLE, where:
    - CMD and SIM are static text
    - ENABLE enables the simulation mode. ‘ACTIVATE’ activates the simulation mode, and ‘DISABLE’ both disables and deactivates the simulation mode.
- PX - Science Payload Transmission On/Off
  - Generic one: CMD,<TEAM_ID>,SP1X/SP1X,<ON_OFF>
  - Example of command to be sent: CMD,1064,SP1X,ON. This will trigger the Container to relay a command to the Science Payload 1 to begin telemetry transmissions.
- SIMP - Simulated Pressure Data (to be used in Simulation Mode only)
  - Generic one: CMD,<TEAM ID>,SIMP,<PRESSURE>
  - Example of command to be sent: CMD,1064,SIMP,101325, which provides a simulated pressure reading to the Container (101325 Pascals = approximately sea level)

•Ground station software is nearly finished and ready for real testing with simulated and real data in a few days.

MQTT Integration

Once the ground station is turned on, we will proceed to connect it with the MQTT server:

Client.connect(“mqtt://username:password@cansat.info”);

When the ground station software saves a data line to a file, that line will be sent to the broker (1 line per second):

mqttc.publish(“teams/1064”, “<TEAM_ID>,<MISSION_TIME>,<PACKET_COUNT>,<PACKET_TYPE>,<MODE>,<SP1_RELEASED>,<SP2_RELEASED>,<ALTITUDE>,<TEMP>,<VOLTAGE>,<GPS_TIME>,<GPS_LATITUDE>,<GPS_LONGITUDE>,<GPS_ALTITUDE>,<GPS_SATS>,<SOFTWARE_STATE>,<SP1_PACKET_COUNT>,<SP2_PACKET_COUNT>,<CMD_ECHO>”);

At the end of sending all the lines, the client will be disconnected from the server:

Client.disconnect();

All this process was carried out using the C# (.Net) language and the M2Mqtt.Net library.

CanSat Integration and Test

Ivan Maccio
Mateo Figallo

CanSat Integration and Test Overview

CANSAT Integration and Test Overview

- **Subsystem Level Testing Plan**
  - Sensor
  - CDH
  - ESP
  - Radio Communication
  - FSW
  - Mechanical
  - Descent Control
- **Integrated Level Functional Test Plan**
  - Descent Testing
  - Communications
  - Mechanisms
  - Deployment
- **Environmental Test Plan**
  - Drop Test
  - Thermal Test
  - Vibration Test
  - FIT Check
  - Vacuum Test

Subsystem Level Testing Plan

In order to make sure everything works as designed, a series of tests will be conducted:

- We are planning to test every sensor alone with an Arduino mega.
- Then, we are going to test the sensor in additive way, meaning that when we finish testing all sensors individually, we are start to test two sensors at once,then three sensors at once, then four, etc., until full integration is completed.
- In parallel we are going to make mechanical tests to the tethered payload.
- We are going to drop it from a drone
- There will be a Flight Software test as well, which consists in simulating all possible processor shut down -in any period of time- and check if the software works correctly in the facing of mentioned situation.
- The sending of data in the correct order will be tested
- We will check the accuracy of the received data from sensors

Integrated Level Functional Test Plan

Integrated Level Functional Test Plan

Container

Mass test.
Free fall test with parachutes.
Test descent rate with chronometer.

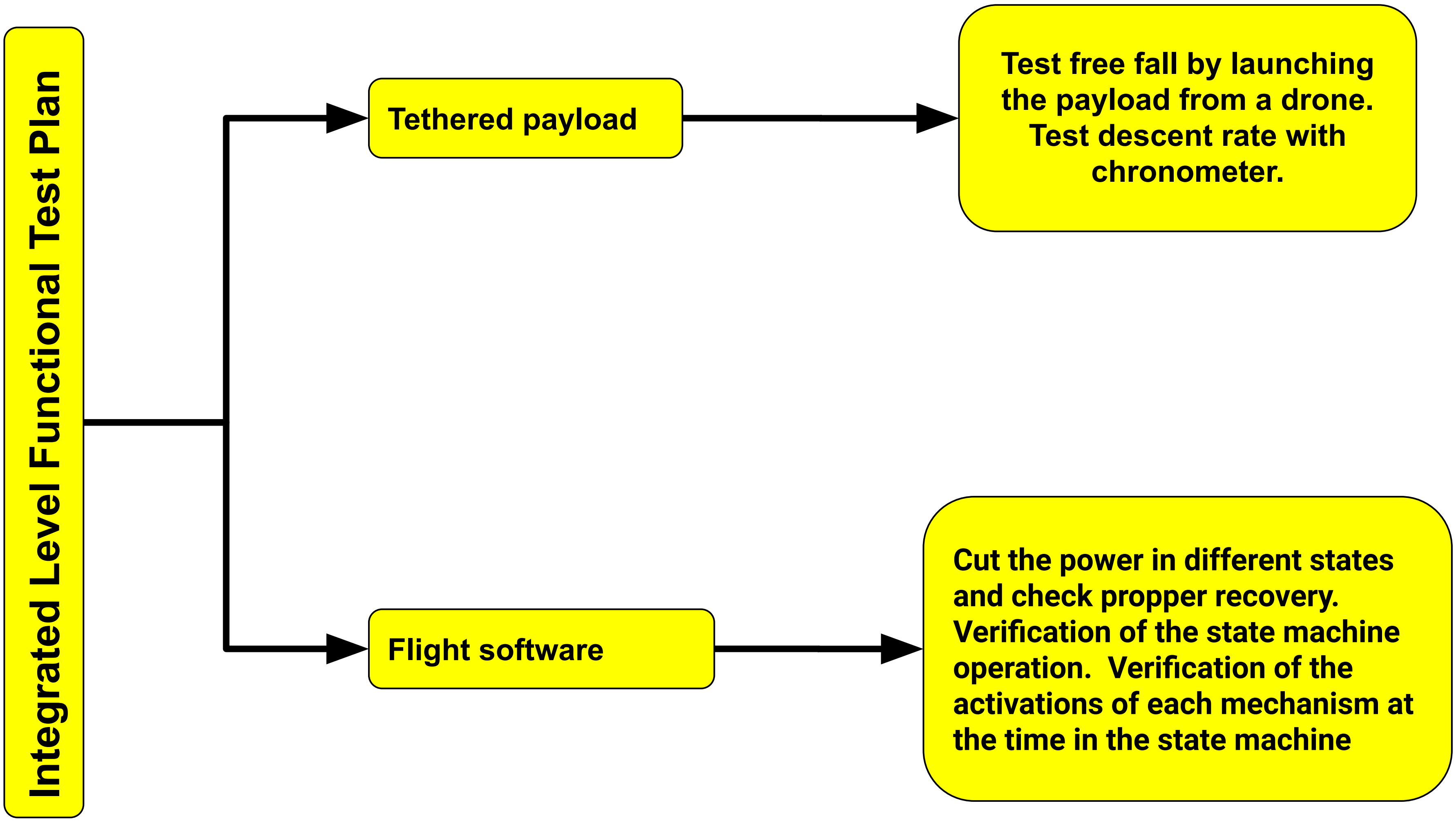
Electronics Sensors

Test each sensor individually with an arduino uno.
Test xbee configuration with one arduino and a arduino pc adaptor.
Xbee modules should had radio link in ground of about 120 m to ensure air link.

Ground Station

Simulate payload data in excel and compare results.
Repeat the simulation with an xbee connected with an arduino that only transmit the simulation data.

Integrated Level Functional Test Plan



Environmental Test Plan

The tests are going to be executed as suggested in the mission guide, for the drop test we are going to use a non flexible string of the spec detailed in mission guide ,for the thermal test we are going to build a isolated foam box and use a hair dryer or a heat gun with temperature regulation for rise the temperature.
For the vibration test we had an orbital sander, we are going to 3D print a locking mechanism to hold the CanSat to the sander.

Test Procedures Descriptions

DROP TEST		
Test Procedure	**Test description**	**Pass Fail Criteria**
1	Power on	Pass if CanSat turns on. Otherwise fail.
2	Verify telemetry is being received	Pass if data is received on the GCS. Otherwise fail.
3	Raise CanSat by the attached cord, so that the attachment points of the cord, on the eye bolt and the parachute, are at the same height.	-
4	Release the CanSat.	-
5	Verify the CanSat did not lose power.	Pass if power is still on. Otherwise fail.
6	Inspect for any damage, or detached parts	Pass if parts are still intact. Otherwise fail.
7	Verify telemetry is still being received.	Pass if data is still being received by GCS. Otherwise fail.

Test Procedures Descriptions

THERMAL TEST		
Test Procedure	**Test description**	**Pass Fail Criteria**
1	Build a foam box and heated up with a heat gun with temperature control and heat the box to 60 degrees	
2	Put powered on CanSat in the box and wait for two hours	Pass if Cansat powers on and sending data to GCS. Otherwise fail.
3	Inspect mechanical and structural to ensure everything is intact	Pass if everything is intact. Otherwise fail.
4	Verify that CanSat is still sending data to GCS	Pass if CanSat is sending data to GCS. Otherwise fail.

Test Procedures Descriptions

VIBRATION CHECK		
Test Procedure	**Test description**	**Pass Fail Criteria**
1	Power on Cansat	Pass if CanSat turns on. Otherwise fail.
2	Verify accelerometer data is being collected	Pass if accelerometer data is received on the GCS. Otherwise fail.
3	Power up the sander	
4	Once the sander is up to full speed, wait 5 seconds	
5	Power down the sander to full stop	
6	Inspect CanSat for damage and functionality	Pass if CanSat is damage free and still functioning. Otherwise fail
7	Verify accelerometer data is still being collected	Pass if accelerometer data is received on the GCS. Otherwise fail.

Test Procedures Descriptions

FIT CHECK		
Test Procedure	**Test description**	**Pass Fail Criteria**
1	Measure CanSat dimension	Pass if CanSat dimension is < 125mmx400mm. Otherwise fail.
2	Visually inspect for sharp edges that will prevent it from deploying out of the rocket	Pass if no sharp edges. Otherwise fail
3	Test the cansat inside the rocket payload	Pass Cansat enter and exit freely

Simulation Test Plan

The tests are going to be executed as suggested in the mission guide.

We are going to use dummy data to test that the ground station correctly sends the data to the MQTT server. Corroborating them from cansat.info/plot.html.

Mission Operations & Analysis

Mateo Figallo

Overview of Mission Sequence of Events

Event	Duties and Responsibilities
Arrival at the simulation site	• All team members will have to put their own facemasks • Ground Control Station, Container and Payload will be assembled and checked.
Setting up	• Antenna will be checked. • Xbee communications will be checked as well. • Fit check and mass measurement. • Container release mechanism check. • Inspection of electronics. • Battery charge check. • Radio links check.
Pre launch	• Review of the checklist. • Team leader approval in order to proceed.
CanSat preparation	• Integrate payload and container.
Final checks preparation	• Team approval.
Rocket integration	• Integrate CanSat with the rocket.

Overview of Mission Sequence of Events

Event	Duties and Responsibilities
Launch	• Take photos • Record
Recovery	• Recover CanSat
Data presentation	• Prepare data • Print / graph results • Deliver results • Save the equipment

Field Safety Rules Compliance

Mission Operations Manual					
1	2	3	4	5	6
GCS Configuration	CanSat Preparation	Rocket CanSat Integration	Launch Preparations Procedure	Launch Procedure	Removal Procedures

- **Mission Operations Manual will include checklists for the each of the topics mentioned above.**
- **Each member will familiarize themselves with the manual before launch day. Safety of each person on launch day is paramount.**
- **The document has been downloaded from CanSat competition website and had been modified to include our instructions. It will be updated as we encounter changes before launch day.**

CanSat Location and Recovery

- A member of the team will detect the container and payload during the flight and identify its possible landing zone.
- Container color will be chosen from red, Pink or orange.
- Payload color will be chosen from red, Pink or orange.
- The CanSat container and the payload will have a label with the following information:
  - Team leader name
  - Team leader email and phone number
  - Team number

Mission Rehearsal Activities

Ground system radio link check procedures	• Assembly of components, testing and troubleshooting Powering on/off the CanSat • Visual verification of loose electronic components before turning on • Turn on container • Turn on the payload • Wait for the container send telemetry indicating the payload working properly
Launch configuration preparations	• Visual verification of mechanical system • Installation of the payload in the container and locking with rubber bands • Close the container top hatch • Stowing the parachute on top of container to ensure deployment
Loading the CanSat in the launch vehicle	• Installation of the Cansat into the rocket payload section

Mission Rehearsal Activities

Telemetry processing, archiving, and analysis	• Reading graphs and indicators on GCS GUI • Verification of CSV file
• **Recovery**	• The member of the team in charge of identifying the possible fall will locate the container and the payload.

The purpose of this section is to summarize and cross reference the compliance to the CanSat Competition Mission Guide requirements.

Requirements Compliance

Mateo Figallo

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
1	Total mass of the CanSat (science payloads and container) shall be 600 grams +/- 10 grams	Y
2	CanSat shall fit in a cylindrical envelope of 125 mm diameter x 400 mm length. Tolerances are to be included to facilitate container deployment from the rocket fairing	Y
3	The container shall not have any sharp edges to cause it to get stuck in the rocket payload section which is made of cardboard	Y
4	The container shall be a fluorescent color; pink, red or orange	Y
5	The container shall be solid and fully enclose the science payloads. Small holes to allow access to turn on the science payloads are allowed. The end of the container where the payload deploys may be open	Y
6	The rocket airframe shall not be used to restrain any deployable parts of the CanSat	Y
7	The rocket airframe shall not be used as part of the CanSat operations	Y
8	The container's first parachute shall not be enclosed in the container structure. It shall be external and attached to the container so that it opens immediately when deployed from the rocket	Y

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
9	The Parachutes shall be fluorescent Pink or Orange	Y
10	The descent rate of the CanSat (container and science payload) shall be 15 meters/second +/- 5m/s after deployment while above 400 meters	Y
11	The descent rate of the CanSat shall be reduced to 5 meters/second +/-2 m/s when the CanSat descends below 400 meters	Y
12	0 altitude reference shall be at the launch pad	Y
13	All structures shall be built to survive 15 Gs of launch acceleration	Y
14	All structures shall be built to survive 30 Gs of shock	Y
15	All electronics shall be hard mounted using proper mounts such as standoffs, screws, or high performance adhesives	Y
16	All mechanisms shall be capable of maintaining their configuration or states under all forces	Y

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
17	Mechanisms shall not use pyrotechnics or chemicals	Y
18	Mechanisms that use heat (e.g., nichrome wire) shall not be exposed to the outside environment to reduce potential risk of setting vegetation on fire	Y
19	Both the container and payload shall be labeled with team contact information including email address	Y
20	Cost of the CanSat shall be under $1000. Ground support and analysis tools are not included in the cost. Equipment from previous years should be included in this cost, based on current market value	Y
21	XBEE radios shall be used for telemetry. 2.4 GHz Series radios are allowed. 900 MHz XBEE radios are also allowed	Y
22	XBEE radios shall have their NETID/PANID set to their team number	Y
23	XBEE radios shall not use broadcast mode	Y
24	The container shall include electronics to receive sensor payload telemetry	Y

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
25	The container shall include electronics and mechanisms to release the science payload on a tether	Y
26	The container shall include a GPS sensor to track its position	Y
27	The container shall include a pressure sensor to measure altitude	Y
28	The container shall measure its battery voltage	Y
29	The container shall transmit its telemetry once per second (1 Hz) in the formats described in the Telemetry Requirements section	Y
30	The container shall poll the payload for telemetry and relay that data four times per second (4 Hz) in the formats described in the Telemetry Requirements section	Y
31	The container shall stop polling and transmitting telemetry when it lands	Y
32	The container and science payload must include an easily accessible power switch that can be accessed without disassembling the cansat and science payloads and in the stowed configuration	Y

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
33	The container and payload must include a power indicator such as an LED or sound generating device that can be easily seen or heard without disassembling the cansat and in the stowed state	Y
34	An audio beacon is required for the container. It shall be powered after landing	Y
35	The audio beacon must have a minimum sound pressure level of 92 dB, unobstructed	Y
36	Battery source may be alkaline, Ni-Cad, Ni-MH or Lithium. Lithium polymer batteries are not allowed. Lithium cells must be manufactured with a metal package similar to 18650 cells. Coin cells are allowed	Y
37	An easily accessible battery compartment must be included allowing batteries to be installed or removed in less than a minute and not require a total disassembly of the CanSat	Y
38	Spring contacts shall not be used for making electrical connections to batteries. Shock forces can cause momentary disconnects	Y
39	The Cansat must operate during the environmental tests laid out in Section 3.5	Y
40	The Cansat shall operate for a minimum of two hours when integrated into the rocket	Y

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
41	The science payload shall have their NETID/PANID set to their team number plus 5000. If the team number is 1000, sensor payload NETID is 6000	Y
42	The science payload shall transmit sensor telemetry to the container when polled	Y
43	Missing in mission guide	?
44	The science payload shall include a pressure sensor, temperature sensor and rotation sensor	Y
45	The science payload shall include a video camera pointing 45 degrees up from the payload NADIR direction	Y
46	The science payload shall maintain orientation so the camera always faces south within +/- 20 degrees	Y
47	The payload shall be connected to the container with a 10 meter tether	Y
48	At 300 meters, the payload shall be released from the container at a rate of .5 meters per second	Y

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
49	The flight software shall maintain a count of packets transmitted which shall increment with each packet transmission throughout the mission. The value shall be maintained through processor resets	Y
50	The container shall maintain mission time throughout the whole mission even with processor resets or momentary power loss	Y
51	The container shall have its time set to UTC time to within one second before launch	Y
52	The container flight software shall support simulated flight mode where the ground station sends air pressure values at a one second interval using a provided flight profile csv file	Y
53	In simulation mode, the flight software shall use the radio uplink pressure values in place of the pressure sensor for determining the container altitude	Y
54	The container flight software shall only enter simulation mode after it receives the SIMULATION ENABLE and SIMULATION ACTIVATE commands	Y
55	The ground station shall command the Cansat to start transmitting telemetry prior to launch	Y
56	The ground station shall generate csv files of all sensor data as specified in the Telemetry Requirements section	Y

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
57	Telemetry shall include mission time with one second or better resolution. Mission time shall be maintained in the event of a processor reset during the launch and mission	Y
58	Configuration states such as if commanded to transmit telemetry shall be maintained in the event of a processor reset during launch and mission	Y
59	Each team shall develop their own ground station	Y
60	All telemetry shall be displayed in real time during descent on the ground station	Y
61	All telemetry shall be displayed in engineering units (meters, meters/sec, Celsius, etc.	Y
62	Teams shall plot each telemetry data field in real time during flight	Y
63	The ground station shall include one laptop computer with a minimum of two hours of battery operation, XBEE radio and a hand-held antenna	Y
64	The ground station must be portable so the team can be positioned at the ground station operation site along the flight line. AC power will not be available at the ground station operation site	Y

Requirements Compliance (multiple slides, as needed)

RQ	Description	Comply
65	The ground station software shall be able to command the container to operate in simulation mode by sending two commands, SIMULATION ENABLE and SIMULATION ACTIVATE	Y
66	When in simulation mode, the ground station shall transmit pressure data from a csv file provided by the competition at a 1 Hz interval to the container	Y
67	The science payloads shall not transmit telemetry during the launch, and the container shall command the science payloads to begin telemetry transmission upon release from the container	Y
68	All video cameras shall be in color, have a resolution of at least 640x480 and record at a minimum of 30 frames a second	Y

Management

Mateo Figallo

Status of Procurements

Component	Status	Component	Status
Waveshare 10 DOF IMU C	Obtained	Parachute Apogee 29118	Obtained
Adafruit Ultimate GPS	Obtained	2.4 ghz antenna	Obtained
Arduino mega 2560 mini	Obtained	Buzzer	Obtained
TowerPro sg90 servo	Obtained	3D printed parts	Obtained
Adafruit spy camera	Obtained	Gimbal	Obtained
Xbee Pro S2C	Obtained	TPU 3D Printing Filament	Obtained
Xbee Pro 3	Obtained	Samsung 30Q 18650 Battery	Obtained
Parachute Apogee 29126	Obtained	Real clock module	Obtained

CanSat Budget – Hardware

Component	Quantity	Unit Price ($)	Total ($)	Status
Waveshare 10 DOF IMU C	1	17.99	17.99	NEW
Adafruit Ultimate GPS	1	12.99	12.99	NEW
Arduino mega 2560 mini	1	8.84	8.84	NEW
TowerPro sg90 servo	2	2.42	4.84	NEW
Adafruit spy camera	1	12.5	12.5	NEW
Xbee Pro S2C	2	32.56	65.12	NEW
Xbee Pro 3	1	32.56	32.56	NEW
Parachute Apogee 29126	1	3.74	3.74	NEW
Parachute Apogee 29118	1	7.38	7.38	NEW
2.4 ghz antenna	3	2.7	8.1	NEW
Buzzer	1	5	5	NEW
3D printed parts	6	7	42	NEW

CanSat Budget – Hardware

Component	Quantity	Unit Price ($)	Total ($)	Status
Gimbal	1	55.99	55.99	NEW
TPU 3D Printing Filament	1	33.99	33.99	NEW
Samsung 30Q 18650 Battery	2	7.99	15.98	NEW
Real clock module	1	2.1	2.1	NEW
Total	**329.12$**			

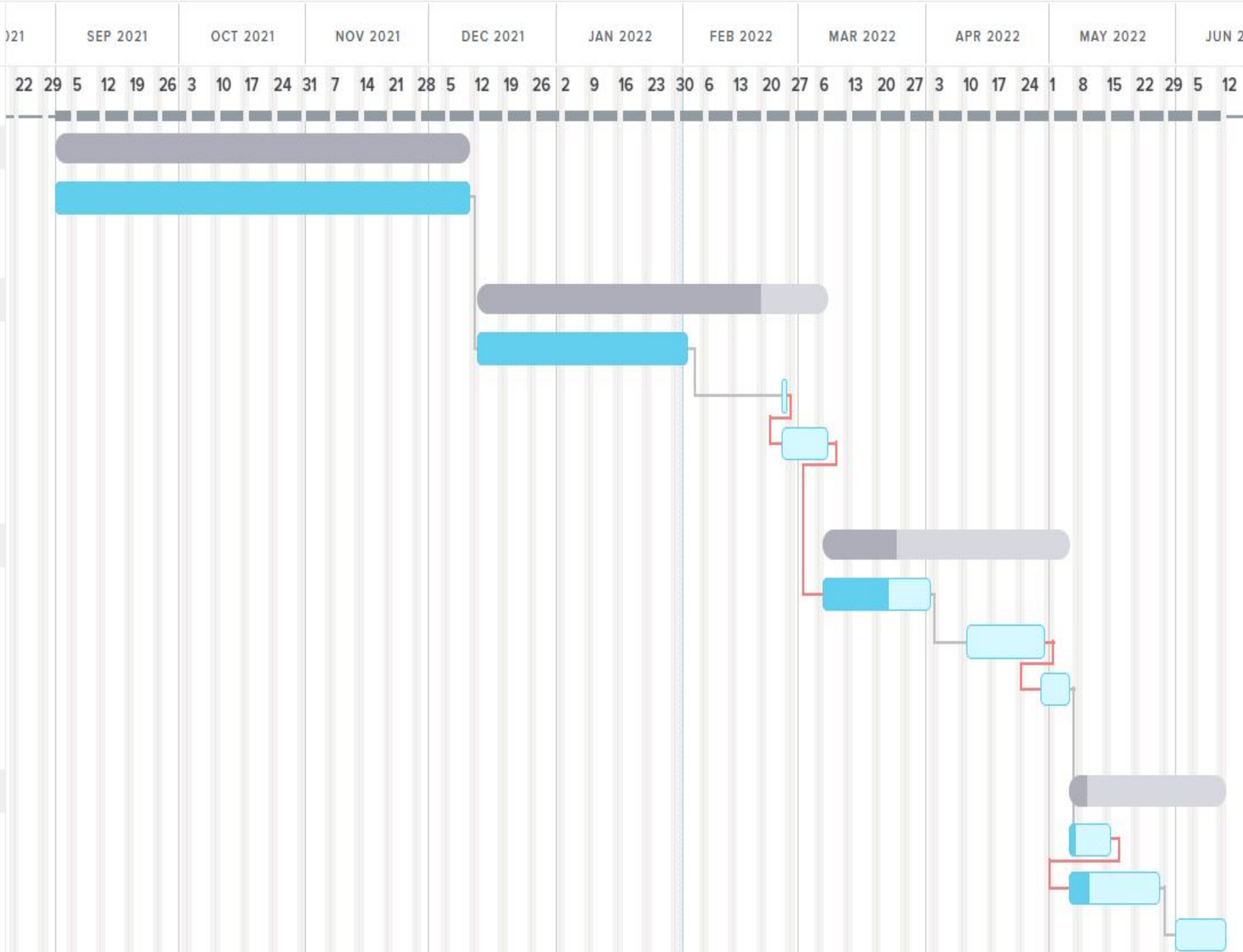
CanSat Budget – Other Costs

Category	Quantity	Unit Price ($)	Total Price($)	Consideration
Travel	5	500	2.500	Estimate
Visa	5	190	950	Estimate
Hotel	5	-	1.050	Estimate
Transportation	1	-	670	Estimate
Food	5	175	875	Estimate
Travel insurance	5	50	250	Estimate
Total	**5.246,5$**			

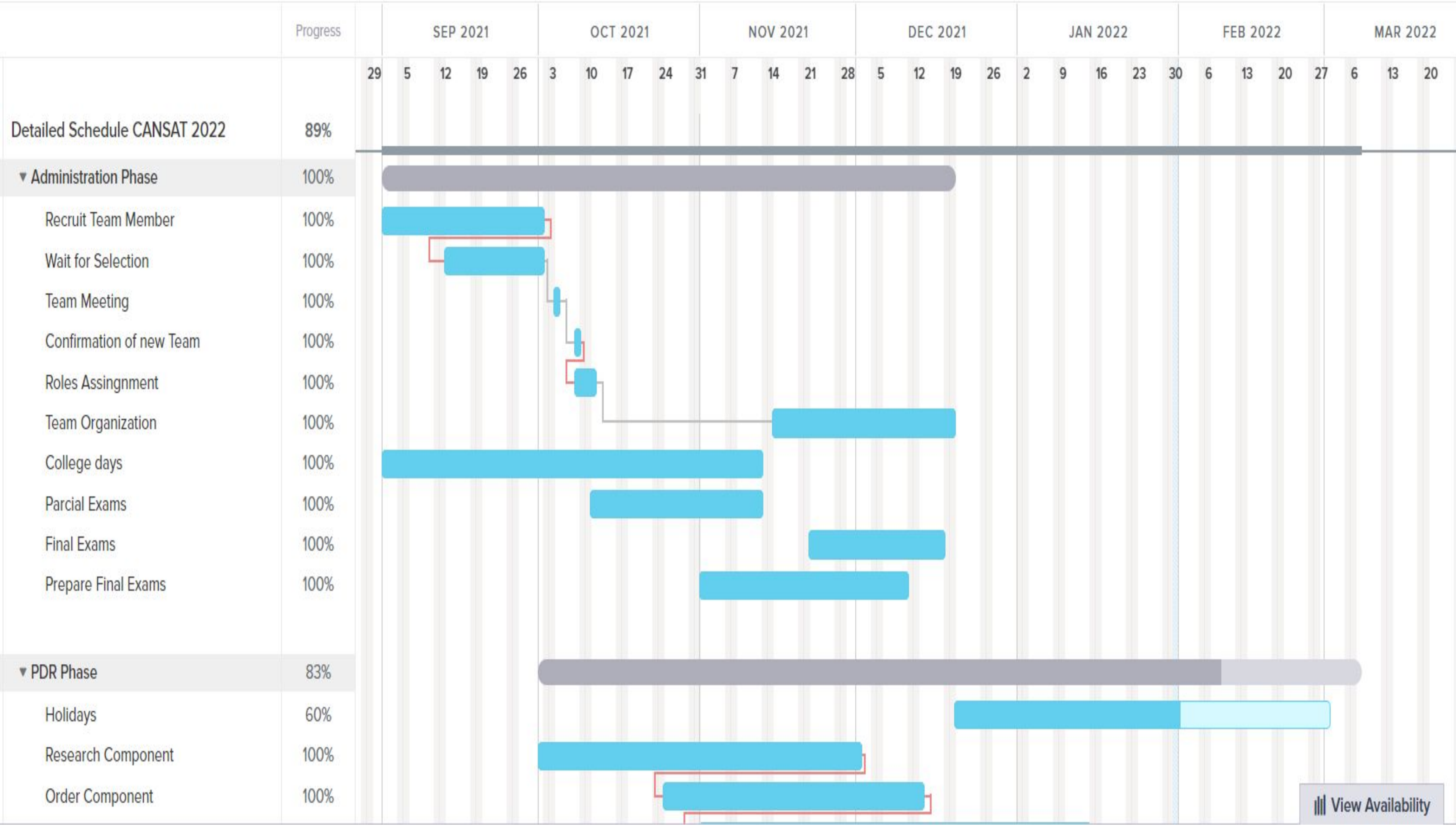
The computer, its support, clothing and other components are not included because they are already purchased

Program Schedule Overview

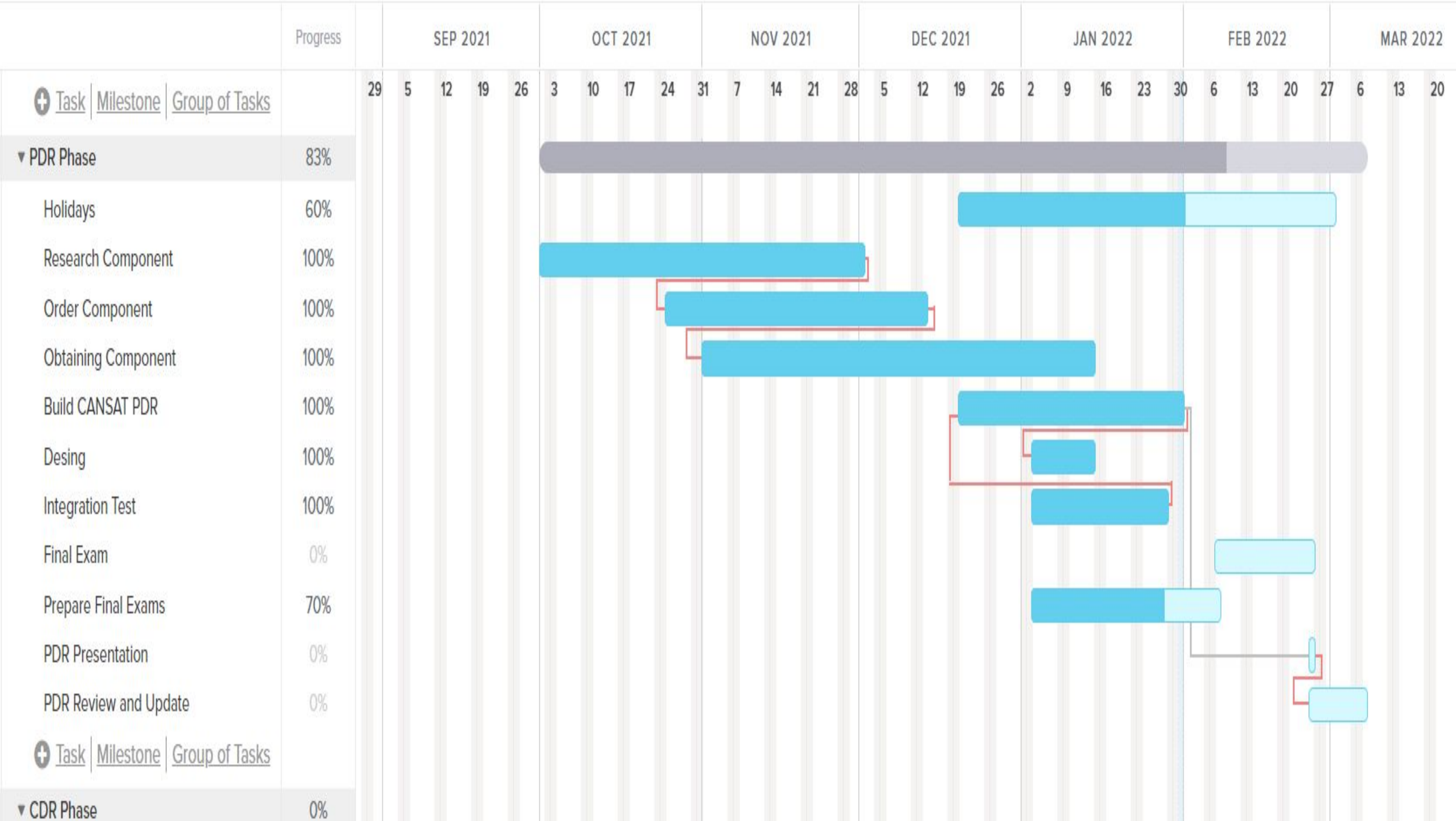
	Progress
Schedule Overview CANSAT 2022	67%
▼ Administrative Phase	100%
	100%
▼ PDR Phase	81%
Work	100%
PDR Presentation	0%
PDR Review and Update	0%
▼ CDR Phase	30%
Work	60%
CDR Presentation	0%
CDR Review and Update	0%
▼ Final Stage	12%
Create Backup Payload and Container	10%
Travel Preparation	20%
Preparation Towards Competition	0%



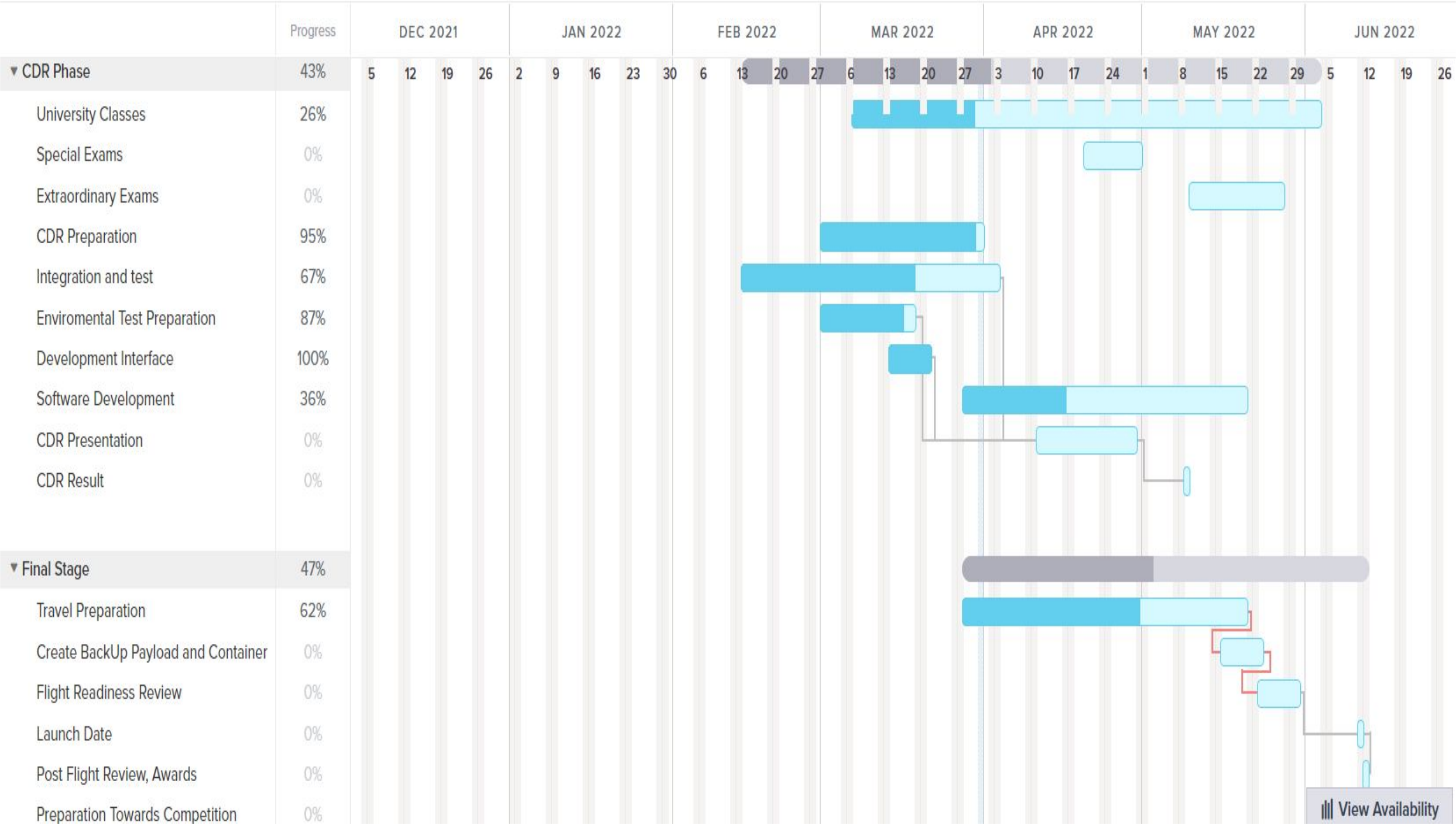
Detailed Program Schedule



Detailed Program Schedule



Detailed Program Schedule



Mission Rehearsals

- Team members and roles

TEAM MEMBER	ROLE
Jorge Royon	Mechanical builder
Juan Esteban Yurquina	Electronic developer and tester
Enzo Juarez	Electronic developer and tester
Ivan Maccio	Environmental tester and Software developer
Mateo Figallo	Environmental tester and Software developer

Mission Rehearsals

- **Mission dry-runs**

25/5/2022 - Full simulated test and error correction if any has occurred, then retry the test until all is performed as it should

Criteria of pass: No error glitches in telemetry, no physical damage or loose components, check telemetry data and ensure coherent results.

1/6/2022 - Repeat the full simulated test to ensure everything is ok

Criteria of pass: No error glitches in telemetry, no physical damage or loose components, check telemetry data and ensure coherent results.

Conclusions

- **We are close to finish the prototype and start testing, we had a good time advantage and trust that our model will accomplish every mission requirement**