

UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
CURSO DE GRADUAÇÃO EM ENGENHARIA AEROESPACIAL

Lorenzzo Quevedo Mantovani, Mateus Schmitz Venturini

**RELATÓRIO FINAL - CPIO IV: DESENVOLVIMENTO DE UM CANSAT  
PARA COMPETIÇÃO**

Santa Maria, RS  
2019

**Lorenzzo Quevedo Mantovani, Mateus Schmitz Venturini**

**RELATÓRIO FINAL - CPIO IV: DESENVOLVIMENTO DE UM CANSAT PARA  
COMPETIÇÃO**

Trabalho apresentado ao Curso de Graduação em Engenharia Aeroespacial da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para aprovação na disciplina  
**Concepção, Projeto, Implementação e Operação IV.**

ORIENTADOR: Prof. Eduardo Escobar Bürger

Santa Maria, RS  
2019

**Lorenzzo Quevedo Mantovani, Mateus Schmitz Venturini**

**RELATÓRIO FINAL - CPIO IV: DESENVOLVIMENTO DE UM CANSAT PARA COMPETIÇÃO**

Trabalho apresentado ao Curso de Graduação em Engenharia Aeroespacial da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para aprovação na disciplina  
**Concepção, Projeto, Implementação e Operação IV.**

**Aprovado em 15 de maio de 2019:**

---

**Eduardo Escobar Bürger, Dr. (UFSM)**  
(Orientador)

Santa Maria, RS  
2019

## RESUMO

### **RELATÓRIO FINAL - CPIO IV: DESENVOLVIMENTO DE UM CANSAT PARA COMPETIÇÃO**

**AUTOR:** Lorenzzo Quevedo Mantovani, Mateus Schmitz Venturini  
**ORIENTADOR:** Eduardo Escobar Bürger

O desenvolvimento de oportunidades no meio aeroespacial no Brasil tem crescido nos últimos anos; novos cursos de graduação foram abertos em diversas universidade ao longo do país e cada vez novos empregos e funções a serem exercidas surgem no mercado de trabalho. De forma a contribuir com esse avanço, no ano de 2018 a competição Cube Design foi fundada pelo Programa de Pós-graduação em Engenharia e Tecnologias Espaciais do INPE de modo a promover a criatividade, integração e despertar o interesse em atividade no setor espacial, com ênfase em pequenos satélites. Este trabalho apresenta o desenvolvimento de um CanSat, picossatélite em formata de lata, para participar do 2º Cu-beDesign, além de complementar a graduação dos alunos responsáveis por este projeto. Utilizando da metodologia imposta pela disciplina de CPIO na UFSM, busca-se implementar e operar um CanSat.

**Palavras-chave:** Aeroespacial. CanSat. CPIO. Competição. Operação

## ABSTRACT

### FINAL REPORT - CPIO IV: DEVELOPMENT OF A CANSAT FOR COMPETITION

AUTHOR: Lorenzzo Quevedo Mantovani, Mateus Schmitz Venturini  
ADVISOR: Eduardo Escobar Bürger

The development of opportunities in the aerospace sector in Brazil has grown in the last years; new graduation courses opened in several universities around the country, and every moment new jobs and positions arises in the market. In order to contribute to this advancement, in the year 2018 the Cube Design competition was founded by INPE's Post-graduate Program in Engineering and Space Technologies with the objective to promote creativity, integration and awaken interest in the space sector activity, with emphasis in small satellites. Thus, this work presents the proposal of a CanSat, small satellite in a can format, to participate in the 2º CubeDesign, in addition to complementing the graduation of the students responsible for this project. Using the methodology imposed by the discipline of CDIO at UFSM, it is sought to implement and operate a CanSat.

**Keywords:** Aerospace. CanSat. CDIO. Competition. Operation

## LISTA DE FIGURAS

Figura 4.1 – Conceito de operação A .....	16
Figura 4.2 – Conceito de operação B .....	17
Figura 5.1 – <i>Product Tree</i> do projeto do CanSat no Conceito A .....	18
Figura 5.2 – <i>Product Tree</i> do projeto do CanSat no Conceito B .....	18
Figura 6.1 – Arduino UNO .....	20
Figura 6.2 – TM4C123G .....	21
Figura 6.3 – Estrutura externa em PLA .....	23
Figura 6.4 – Formato do paraquedas tipo <i>Cross</i> .....	24
Figura 6.5 – Variação da área do paraquedas com a velocidade terminal para o paraquedas tipo <i>Cross</i> .....	25
Figura 6.6 – Arduino Nano .....	30
Figura 6.7 – Formato do paraquedas tipo <i>flat octagonal</i> .....	31
Figura 6.8 – Variação da área do paraquedas com a velocidade terminal para o paraquedas tipo <i>Flat Octagonal</i> .....	33
Figura 6.9 – Arvore de Decisões .....	37
Figura 7.1 – Estrutura externa após os lançamentos .....	39
Figura 7.2 – Arduino UNO danificado após o segundo lançamento .....	41
Figura B.1 – Circuito elétrico - Conceito B .....	47
Figura B.2 – Circuito elétrico - segunda vista .....	48

## **LISTA DE TABELAS**

Tabela 6.1 – Balanço de Massa .....	19
Tabela 6.2 – Relação dos componentes do sistema e requisitos de missão conforme Conceito A.....	28
Tabela 6.3 – Balanço de Massa .....	29
Tabela 6.4 – Relação dos componentes do sistema e requisitos de missão conforme Conceito B.....	35
Tabela 6.5 – Tabela de custo do Conceito A .....	36
Tabela 6.6 – Tabela de custo do Conceito B .....	36

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>7</b>
1.0.1	Objetivo do projeto .....	7
1.1	2° CUBEDESIGN .....	7
1.1.1	Objetivos de Missão .....	8
1.1.2	Regulamentos da Competição .....	8
1.1.3	Pontuação .....	9
1.2	METODOLOGIA .....	9
<b>2</b>	<b>REQUISITOS PROPOSTOS .....</b>	<b>11</b>
<b>3</b>	<b>CRONOGRAMA E ATIVIDADES REALIZADAS .....</b>	<b>13</b>
<b>4</b>	<b>CONCEITOS DE MISSÃO .....</b>	<b>16</b>
4.1	CONCEITO A .....	16
4.2	CONCEITO B .....	17
<b>5</b>	<b>PRODUCT TREE .....</b>	<b>18</b>
<b>6</b>	<b>ESCOLHA DE CONCEITOS PARA CANSAT .....</b>	<b>19</b>
6.1	CONCEITO A .....	19
6.1.1	Balanço de massa .....	19
6.1.2	Dados de Microcontrolador .....	20
6.1.3	Escolha dos sensores .....	21
6.1.4	Comunicador .....	23
6.1.5	Estrutura .....	23
6.1.6	Paraquedas .....	24
6.1.7	Componentes .....	26
6.1.8	Análise de Requisitos .....	27
6.2	CONCEITO B .....	29
6.2.1	Balanço de Massa .....	29
6.2.2	Dados dos Microcontroladores .....	29
6.2.3	Estrutura .....	30
6.2.4	Paraquedas .....	31
6.2.5	Escolha dos Sensores .....	33
6.2.6	Componentes .....	33
6.2.7	Análise de Requisitos .....	34
6.3	TESTES .....	36
6.4	CONCEITO ESCOLHIDO .....	36
<b>7</b>	<b>LIÇÕES APRENDIDAS .....</b>	<b>38</b>
7.1	SUGESTÕES DE MELHORIAS .....	41
<b>8</b>	<b>CONCLUSÃO .....</b>	<b>43</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>44</b>
	<b>APÊNDICE A – RELATÓRIO DE VERIFICAÇÃO .....</b>	<b>45</b>
A.1	TESTES DE COMUNICAÇÃO COM MÓDULO HC-05 .....	45
A.2	TESTES DE COMUNICAÇÃO COM ESP8266 .....	45
A.3	TESTES DE COMUNICAÇÃO COM NRF24L01 .....	45
A.4	TESTE DE CONECTIVIDADE DOS COMPONENTES .....	46
A.5	INSPEÇÃO DA ESTRUTURA EXTERNA .....	46
	<b>APÊNDICE B – SISTEMA ELÉTRICO - CONCEITO B .....</b>	<b>47</b>

## 1 INTRODUÇÃO

A disciplina de Concepção, Projeto, Implementação e Operação (CPIO) foi adicionada no currículo de Engenharia Aeroespacial da Universidade Federal de Santa Maria (UFSM) de maneira a promover o crescimento inovador e profissional dos alunos. Com o desenvolvimento de novos projetos ao longo de cada uma das quatro fases da disciplina (separada em 4 semestres diferentes, intervalados em 1 ano cada), o aluno tem como resultado um grande acréscimo em sua vida acadêmica e profissional. Neste trabalho estão presentes as quatro fases do CPIO focando, principalmente, na Operação.

Este relatório apresenta o desenvolvimento de um CanSat (do acrônimo das palavras em inglês - *Can* e *Satellite*) para ser testado na competição 2º Cube Design de 24 a 27 de Julho de 2019, em São José dos Campos/SP. Os alunos denominaram sua equipe como "Ready, Sat, Go!" (uma variação da expressão "*Ready, Set, Go!*", do inglês) e ao longo deste semestre trabalham desde o conceito até a operação do sistema espacial a ser testado na competição.

### 1.0.1 Objetivo do projeto

O projeto tem como objetivo desenvolver um satélite de baixo custo da categoria CanSat para participações em competições, tais como o CubeDesign. Por ser o primeiro modelo do gênero desenvolvido na UFSM, foram tomadas como base as regras do 2º CubeDesign. Além disso, o projeto tem como segundo objetivo geral a documentação do processo utilizado e das decisões tomadas, para que sirvam de base para futuras equipes da instituição.

### 1.1 2º CUBEDESIGN

A competição CubeDesign caracteriza-se por apresentar aos competidores a possibilidade de despertar seu interesses em atividades no setor espacial. A competição é separada em três categorias: CubeSat, CanSat e Mockup. O CubeDesign é organizado pelo curso de Pós-graduação em Engenharia e Tecnologia Espaciais (PGETE) do Instituto Nacional de Pesquisas Espaciais (INPE) (CUBEDESIGN, 2019).

### 1.1.1 Objetivos de Missão

O principal objetivo da competição do CanSats é simular o lançamento, a reentrada e o pouso de um módulo espacial carregando experimentos. Para isso, o satélite será lançado a partir de um tubo de ejeção com a mesma dimensão do CanSat. A partir deste momento, o satélite deve analisar a variação de altitude durante toda a operação. Quando em solo, deve realizar uma missão de solo (tal como o acionamento de um dispositivo sonoro) e, ao fim da operação completa, deve-se adquirir os dados de variação de altitude, analisá-los e apresentá-los aos jurados e demais competidores.

### 1.1.2 Regulamentos da Competição

As regras da competição podem ser divididas em gerais (relacionadas à equipe) e de missão (relacionadas ao CanSat e suas especificações). É importante ressaltar que todas estas informações foram retiradas da página do evento do 2º CubeDesign (CUBEDESIGN, 2019).

Das regras gerais, citam-se principalmente:

- Cada equipe deve conter no mínimo dois participantes e um professor responsável;
- Não são permitidas trocas de componentes do satélite entre as equipes durante a execução das prova.

Quanto as regras para a realização da missão do CanSat, têm-se:

- Deve-se seguir as especificações da geometria e dimensões do CanSat;
  - Geometria: Cilindro;
  - Altura: 120 mm;
  - Diâmetro: 60 mm.
- A carga útil deve ser mantida no interior do satélite com as seguintes propriedades:
  - Altura: 22 mm;
  - Largura: 30 mm;
  - Comprimento: 52 mm;
  - Massa: 43.57 gramas.
- A massa total do sistema deve ser de  $400 \pm 100$  g;
- A equipe deve obrigatoriamente recuperar os dados de altitude do lançamento;

- O toque no solo não deve ser catastrófico, para isso, a carga útil está equipada com acelerômetros para medir o nível de impacto;

É importante ressaltar que o descumprimento das regras de dimensão do CanSat, massa total e toque ao solo são eliminatórias para o quesito da competição.

### 1.1.3 Pontuação

Uma vez que este projeto tem por objetivo apresentar um satélite para a competição 2º Cube Design, é importante que os alunos participantes estejam cientes das pontuações relativas à competição. Abaixo são apresentados valores relacionados a cada fase do evento, conforme o regulamento (CUBEDESIGN, 2019):

- Proteção da carga útil: Leve (700 pts), Moderado (500 pts), Forte (300 pts) e Catastrófico (Eliminatório);
- Recuperação da informação da variação de altitude (200 pts);
- Execução de uma missão e solo (e somente em solo) (200 pts);
- Acionamento do sistema de pouso: Autônomo (200 pts), Telecomando (100 pts) ou Pré-acionado (20 pts);
- Massa total em g (500 pts - massa);
- Projeto: Detalhamento e coerência com a execução (até 150 pts).

## 1.2 METODOLOGIA

Para o desenvolvimento do projeto, inicialmente serão levantados os requisitos da missão baseados nos regulamentos impostos pelo CubeDesign. Com isso, serão formados conceitos de operação que então originarão conceitos de CanSats. Com base nos conceitos, serão levantados os componentes específicos com suas respectivas propriedades para posterior análise. Com os conceitos já estipulados, o modelo final será escolhido baseado no desempenho, custo e facilidade de montagem e operação. Para auxiliar na etapa de decisão, os componentes e/ou subsistemas poderão ser testados de forma a verificar sua eficácia durante ambientes similares aos que serão enfrentados na competição.

Com o conceito definido, a produção, montagem e integração do modelo final será iniciada. No final da integração, irão ser realizados testes para garantir o bom funcionamento do satélite, além de avaliar possíveis falhas, permitindo que correções sejam aplicadas. Em paralelo a todas as etapas, ocorrerá a documentação das decisões tomadas, com suas respectivas justificativas.

## 2 REQUISITOS PROPOSTOS

Com base nos requisitos apresentados pela comissão organizadora do CubeDesign e pelos valores de pontuação estipulados nas Seções 1.1.2 e 1.1.3, a equipe decidiu pelos seguintes requisitos de missão:

**REQ\_ID 001** A estrutura externa do satélite deve possuir dimensões iguais às estipuladas pelas normas da competição;

Tipo de verificação: Inspeção. O satélite deve ser medido.

**REQ\_ID 002** O satélite deve acionar o sistema de pouso de forma automática;

Tipo de verificação: Teste.

Rationale: Segundo as regras da competição, um sistema de paraquedas é categorizado como autônomo quando o sistema de pouso é acionado quando uma determinada condição é percebida, sem telecomandos externos.

**REQ\_ID 003** Deve ser possível determinar a altitude do satélite para cada instante de voo;

Tipo de verificação: Teste. A altitude do satélite deve ser reconstruída com base nos dados coletas.

Rationale: A determinação pode ser feita onboard ou em solo com os dados dos sensores.

**REQ\_ID 004** Deve ser possível obter os dados coletados pelo satélite;

Tipo de verificação: Teste.

**REQ\_ID 005** Deve ser possível remover os componentes internos do satélite em não mais do que 3 minutos;

Tipo de verificação: Teste.

Rationale: Devido ao pequeno tempo de preparação, isso é necessário para posicionar a payload.

**REQ\_ID 006** Deve ser possível posicionar e fixar os componentes internos e paraquedas do satélite em não mais do que 4 minutos;

Tipo de verificação: Teste.

Rationale: Devido ao pequeno tempo de preparação, isso é necessário para posicionar a payload.

**REQ\_ID 007** A remoção dos componentes internos do satélite não deve desativar o funcionamento do sistema;

Tipo de verificação: Teste. A partir da remoção dos componentes, deve-se realizar o teste de funcionamento do sistema.

**REQ\_ID 008** A força de impacto no momento do pouso não deve ser catastrófica;

Tipo de verificação: Teste. Nenhum componente do sistema deve ser danificado no momento do toque no solo.

Rationale: O impacto é considerado catastrófico quando existe dano físico a algum componente.

**REQ\_ID 009** O satélite deve ter massa superior a 300 gramas;

Tipo de verificação: Inspeção. O satélite deve passar pelo processo de pesagem.

Rationale: Quanto menor a massa, mais pontos.

**REQ\_ID 010** A estrutura externa do satélite deve cobrir totalmente os componentes internos durante o lançamento;

Tipo de verificação: Inspeção.

**REQ\_ID 011** A estrutura deve ser capaz de manter os demais componentes do satélite em suas respectivas posições durante toda a missão;

Tipo de verificação: Teste.

**REQ\_ID 012** A face inferior do satélite deve ser a primeira a tocar ao solo;

Tipo de verificação: Teste.

**REQ\_ID 013** Quando em solo, o sistema deve realizar uma missão em solo;

Tipo de verificação: Teste.

Rationale: Missão em solo pode ser, por exemplo, tocar um *buzzer*.

### 3 CRONOGRAMA E ATIVIDADES REALIZADAS

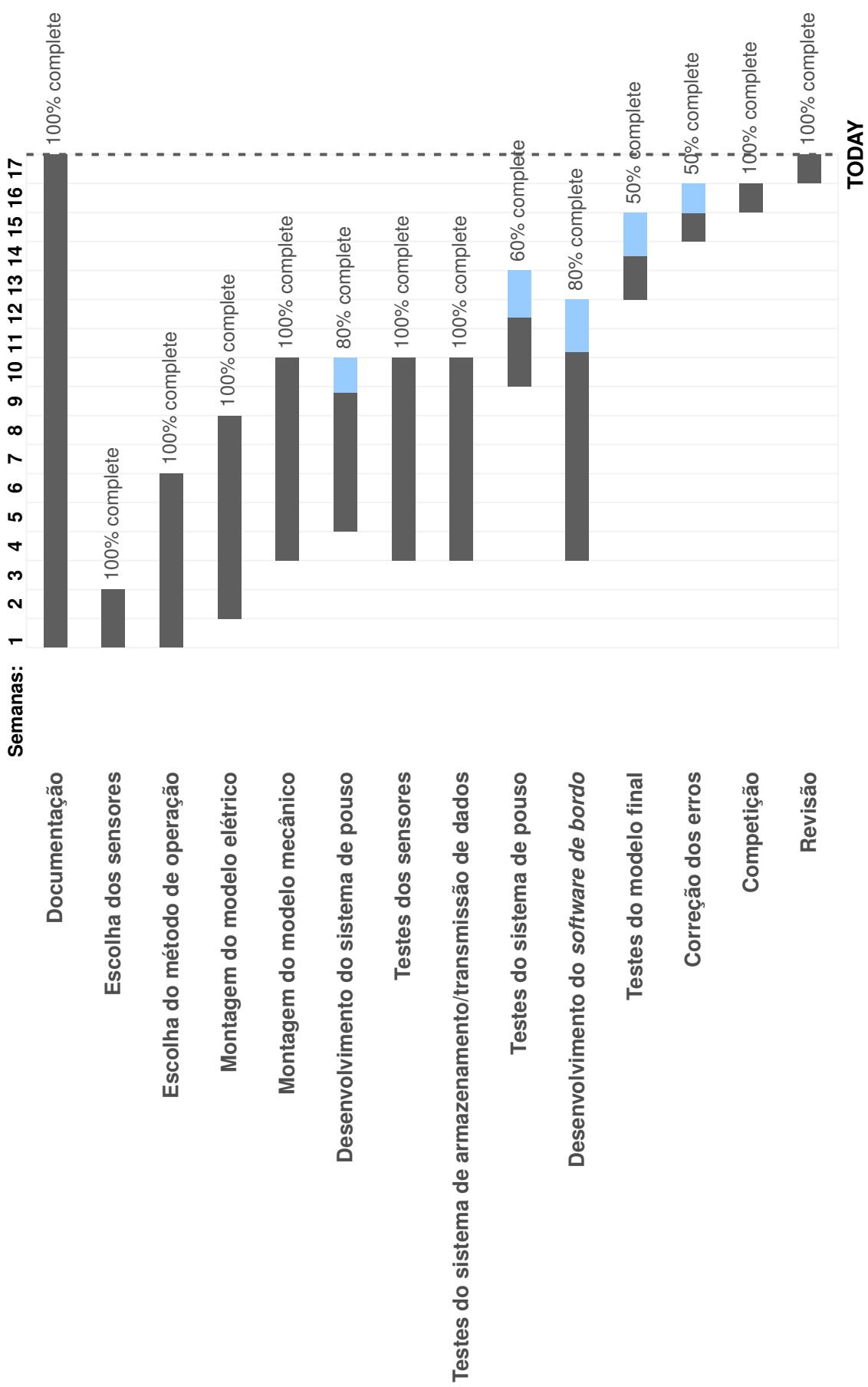
O cronograma está definido considerando o início das atividades como o dia 22/03/2019 e o fim das atividades como a uma semana após o término da competição (03/08/2019), período para considerações finais sobre o desempenho do sistema utilizado. O Gantt Chart abaixo apresenta o cronograma das atividades de forma detalhada, sendo o tempo medido em semanas a partir do início do projeto e porcentagens indicando quão perto as tarefas estão de sua conclusão.

Abaixo estão descritas, de forma geral, atividades realizadas:

- Escolha dos sensores: Para definição dos sensores, contou-se com a ajuda do Prof. André Luís da Silva. Foi determinado que para o bom funcionamento de um filtro de Kalman, é necessária a utilização de sensores para medidas das acelerações, velocidades angulares, campo magnético e pressão atmosférica. Os dados e motivos de escolha dos sensores estão descritos no Capítulo 6.1.3, onde as especificações de cada modelo foram consideradas.
- Escolha do método de operação: Com o suporte dos professores Eduardo Escobar Bürger e André Luís da Silva, foi definido o método de operação do Satélite. De forma a evitar os problemas típicos de comunicações via rádio, como perda de sinal, falta de pacotes, dados corrompidos, não sincronia de tempo, optou-se pelo cálculo do filtro de Kalman ser realizado em solo, como pós processamento de dados. Mais informações são disponibilizadas no capítulo 6.
- Montagem do modelo elétrico: Foi realizada a integração do comunicador NRF24L01 - transmissor - e do sensor MPU6050 com a placa Arduino UNO. A placa Arduino MEGA foi conectada com o comunicador NRF24L01 - receptor -. De forma similar, outros comunicadores foram testados. Sensores também foram conectados e testados, além do adaptador de cartão SD para gravação de dados. O modelo elétrico em sua versão final inicialmente foi montado em uma *protoboard* para que testes fossem realizados. Só então a forma final na estrutura interna e com fios soldados foi utilizada.
- Montagem do modelo mecânico: A estrutura foi gerada em impressora 3D pelo Professor Eduardo Bürger. A estrutura interna foi modelada em CAD e então produzida por impressão 3D. Foi necessário atualizar o modelo em CAD e realizar mais de uma impressão para cada componente devido a problemas de encaixe que surgiram pela expansão do plástico durante a impressão.
- Desenvolvimento do sistema de pouso: Entrou-se em contato com a equipe Carancho Aerodesign para averiguar o sistema de paraquedas utilizado. Apesar da função

ser ligeiramente diferente (gerar leve força de arrasto e suportar 3 *kg*), o modelo de dobradura utilizado pode ser empregado. O material utilizado no paraquedas da equipe do Carancho é o de guarda-chuvas, e sua viabilidade para o CanSat está sendo estudada. Foi adquirido 4*m*<sup>2</sup> de material Dácron para a confecção do paraquedas, sendo que foram realizados testes para determinar o tamanho do paraquedas (limitado pelo espaço disponível no interior do CanSat) e seu formato, conexões de fios, dobraduras, etc. Entretanto, o quesito do mecanismo de acionamento acabou sendo finalizado pouco tempo antes da competição, e não teve a atenção necessária dedicada. Portanto, essa etapa é considerada como não concluída.

- Testes dos sensores: Os testes dos sensores foram realizados em conjunto com a etapa "Montagem do modelo elétrico", de forma a avaliar a correta ligação e conexão dos componentes. Mais testes foram realizados com o modelo elétrico já finalizado. Para essa etapa, contou-se com a ajuda do aluno Renan Sacco (graduando no curso de Engenharia Elétrica pela Universidade Federal de Santa Maria) que disponibilizou parte do código para leitura do sensor MPU9250 e os modos de calibração para a IMU.
- Testes do sistema de comunicação: Os testes foram realizados com os módulos bluetooth HC-05, Wi-Fi ESP8266 e NRF24L01. Os dois primeiros se mostraram ineficientes e incapazes de cumprir o requisito **REQ 004**. Os testes realizados com o módulo NRF24L01 apresentaram os melhores resultados, entretanto com baixa confiabilidade. O relatório dos testes está no Anexo A.
- Testes do sistema de pouso: Essa etapa foi parcialmente cumprida devido a testes com o sistema completo e em condições mais próximas a do lançamento não ter sido realizada.
- Desenvolvimento do *software* de bordo: Etapa não concluída totalmente. Realizada em conjunto com as etapas Montagem do modelo elétrico, Teste de sensores e Testes do sistema de comunicação. Após isso, as rotinas de abertura de paraquedas e de missão de solo foram introduzidas, e as condições para cada etapa adotadas. Durante a competição, conforme os problemas surgiam, o *software* foi atualizado e corrigido. Ainda assim, existem erros e diversos pontos a serem melhorados. Portanto é considerada como inacabada.



## 4 CONCEITOS DE MISSÃO

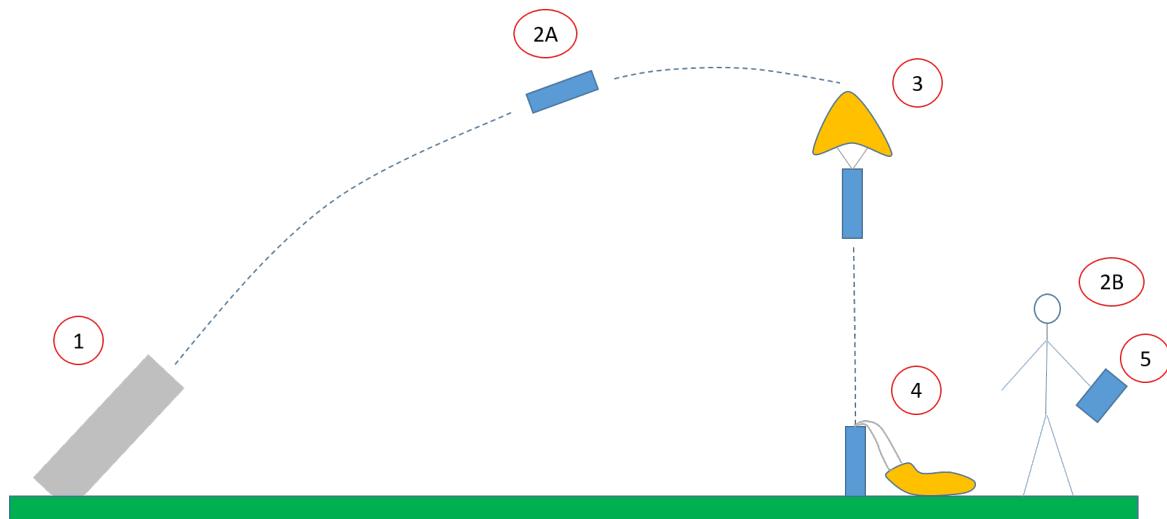
Para a missão a ser realizada, foram propostas e analisados dois conceitos de operação, A e B. Os dois conceitos possuem os mesmos princípios básicos, atendendo aos requisitos propostos; entretanto a forma como isso é realizado varia.

### 4.1 CONCEITO A

Este desenvolvido consiste na utilização de comunicadores *wireless* para a transmissão em tempo real da altitude do satélite, não sendo armazenados na memória. O processamento dos dados e a rotina de programação são embarcados no microcontrolador, para que somente os valores de tempo e altitude sejam transmitidos. A abertura do paraquedas ocorre de forma automática baseado nos dados dos sensores. No momento de toque no solo, o satélite realiza a missão de solo, que consiste em emitir um som através de um *buzzer*. O segmento em solo, então, somente gera o gráfico da altitude pelo tempo.

Neste conceito as etapas de operação, como mostradas na Figura 4.1, são:

Figura 4.1 – Conceito de operação A.



Fonte: Autores.

**1** Lançamento do CanSat;

**2A** Voo, realizando a coleta e processamento dos dados;

**2B** Recepção de dados, simultânea a etapa 2;

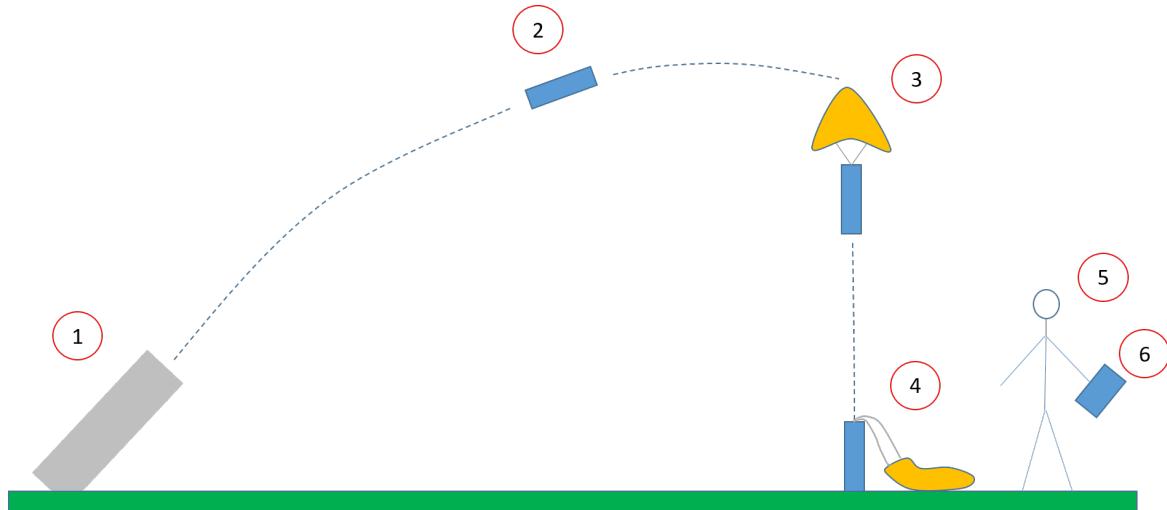
- 3** Acionamento automático do paraquedas;
- 4** Pouso e operação em solo;
- 5** Pós processamento dos dados.

#### 4.2 CONCEITO B

Neste conceito de operação, o satélite coleta os dados durante o voo e os armazena em um cartão de memória. O microcontrolador embarcado realiza operações simples, com os dados brutos, para determinar se o satélite está subindo, descendo ou em solo. O acionamento do paraquedas é feito de forma automática. Após o pouso, o satélite inicia a missão em solo.

A equipe de solo então remove o cartão de memória para coletar os dados registrados e realiza o pós processamento.

Figura 4.2 – Conceito de operação B.



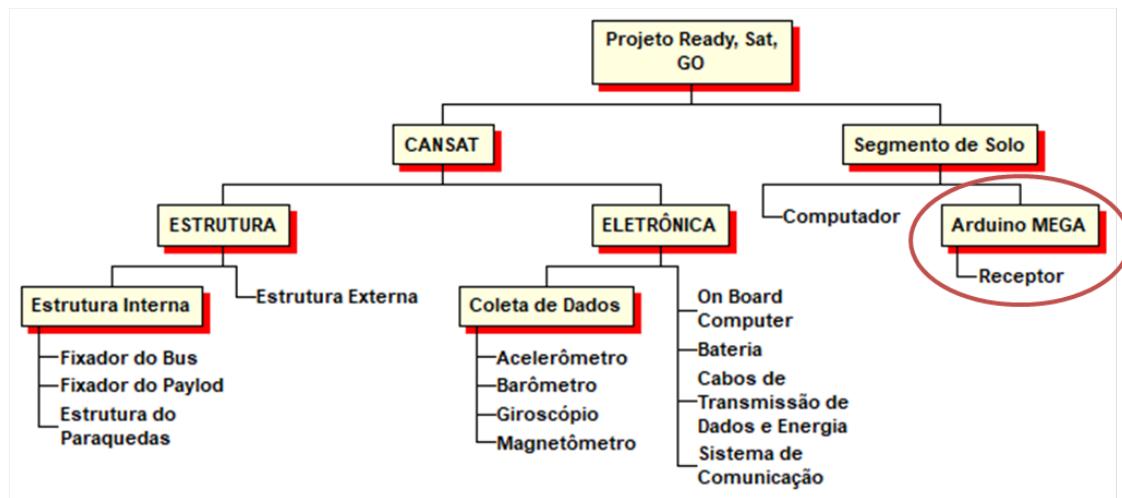
Fonte: Autores.

- 1** Lançamento do CanSat;
- 2** Voo, realizando a coleta e armazenamento dos dados;
- 3** Acionamento automático do paraquedas;
- 4** Pouso e operação em solo;
- 5** Coleta do cartão de memória;
- 6** Pós processamento dos dados;

## 5 PRODUCT TREE

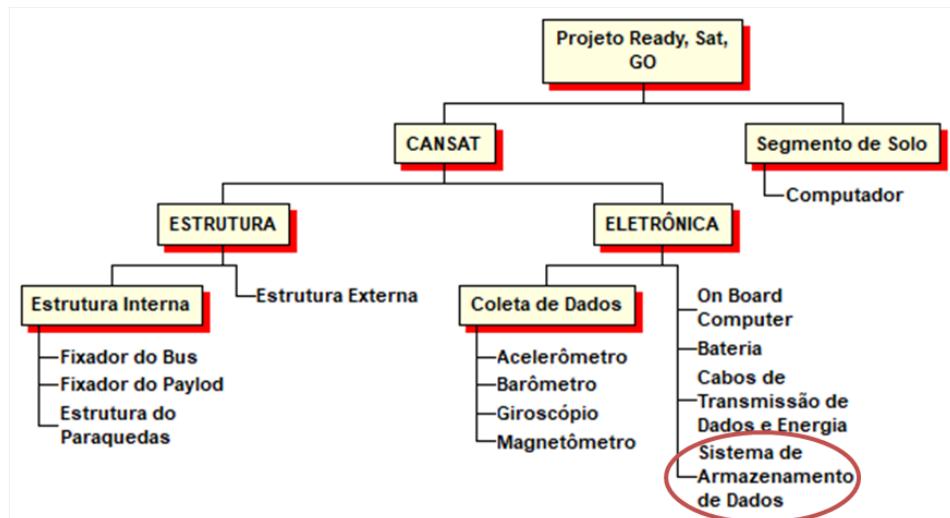
Uma ferramenta importante no desenvolvimento de projetos é a *Product Tree*, esta apresenta os diversos segmentos presentes no trabalho. Dessa forma, é possível analisar separadamente cada subsistema do CanSat. A Figura 5.1 apresenta a *Product Tree* estruturada do satélite CanSat utilizando o conceito A, enquanto que a Figura 5.2 apresenta o conceito B.

Figura 5.1 – *Product Tree* do projeto do CanSat no Conceito A.



Fonte: Autores.

Figura 5.2 – *Product Tree* do projeto do CanSat no Conceito B.



Fonte: Autores.

Nota-se que, como apresentado na Seção 4, o Conceito B não utiliza de sistemas de recepção (Arduino MEGA em 5.1). Além disso, no lugar do sistema de comunicação, tem-se o Sistema de Armazenamento.

## 6 ESCOLHA DE CONCEITOS PARA CANSAT

Esta seção apresenta com mais detalhes os principais conceitos estudados para a construção e operação de CanSats para a competição 2º Cube Design. Ao longo desta seção serão apresentados os principais fatores de cada conceito, assim como seus componentes, vantagens e desvantagens quando comparados entre si. Para concretizar a escolha do conceito a ser utilizado, são apresentados os custos relativos à construção de cada conceito ao final desta seção, assim como o veredito final.

### 6.1 CONCEITO A

#### 6.1.1 Balanço de massa

Para o balanço de massa, os componentes (até o momento) estão listados na tabela abaixo:

Tabela 6.1 – Balanço de Massa

TM4C123G	20 g	Servo	11 g
Comunicador	5 g	Paraquedas	
Antena	8 g	Cabos	
MPU6050+Magnetômetro	2 g	Bateria	42 g
Barômetro	2 g	Buzzer	2 g
Estrutura interna			
Estrutura externa	60 g	Total	≤ 300 g

Considerando o atual balanço de massa, restam 230 gramas para serem distribuídos entre as estruturas internas e externas, sistema de paraquedas e cabos. A *payload* não entra na conta. Dessa forma o requisito **REQ 007** é cumprido.

Inicialmente os testes estão sendo realizados utilizando o microcontrolador Arduino UNO, entretanto o objetivo é migrar para o TM4C123G devido ao maior poder de processamento para que o filtro de Kalman possam ser implementado. Caso não haja compatibilidade entre as bibliotecas dos sensores e o TM4C123G, será utilizado o Arduino UNO - sem o filtro de Kalman -, coletando somente os dados do barômetro.

Figura 6.1 – Arduino UNO



Fonte: Arduino (2019b).

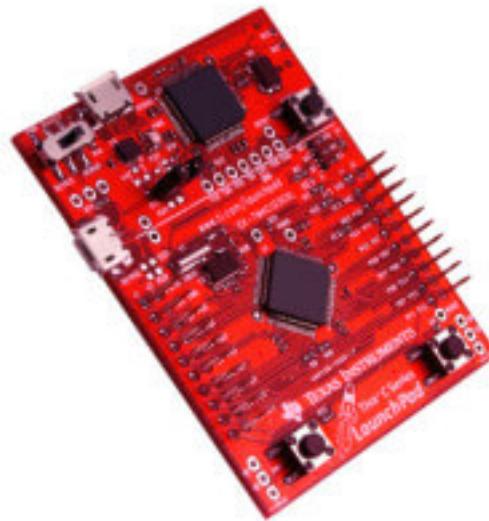
### 6.1.2 Dados de Microcontrolador

As características principais do Arduino UNO (Figura 6.1), conforme Arduino (2019b), são:

- Tensão de operação: 5 V ;
- Tensão de entrada: 7 V até 12 V ;
- Flash Memory : 32 KB;
- SRAM: 2 KB;
- EEPROM: 1 KB;
- Clock Speed : 16 M Hz;
- Número de pinos digitais: 14;
- Número de pinos analógicos: 6;
- Número de portas com PWM: 6;
- Protocolos de comunicação: UART, SPI, I2C.

As características principais do TM4C123G (Figura 6.2), conforme Texas Instruments Incorporated (2014) são:

Figura 6.2 – TM4C123G



Fonte: Texas Instruments Incorporated (2014).

- Tensão de operação: 3.3 V;
- Tensão em GPIO- 5 V - algumas portas só aceitam 3.6 V;
- Tensão de entrada: 7 V até 12 V;
- *Flash Memory*: 256 KB;
- *SRAM*: 32 KB;
- *EEPROM*: 2 KB;
- *Clock Speed*: 80 MHz;
- Processador: 32 bits;
- Protocolos de comunicação: UART, SPI, I2C.

### 6.1.3 Escolha dos sensores

Para a determinação do deslocamento do satélite optou-se pela utilização em conjunto de um acelerômetro e um barômetro. Dessa forma, é possível aplicar um filtro de Kalman para aprimorar e reduzir os erros de medida.

O acelerômetro escolhido (por já ter sido adquirido para outro projeto) é o MPU-6050. As características são (INVENSENSE, 2013):

- Tensão de operação: 3 V até 5 V;
- Protocolo de comunicação: I2C;
- Precisão do acelerômetro:  $\pm 2, \pm 4, \pm 8, \pm 16 g$ ;
- Precisão do giroscópio:  $\pm 250, \pm 500, \pm 1000, \pm 2000 \text{ graus/s}$ ;
- Dimensões: 20x16x1 mm.

Caso se torne necessário fazer a aquisição dos dados de velocidade angular, o sensor já possui essa capacidade. Inicialmente espera-se utilizar o MPU 9250 por conter o magnetômetro, o qual já foi adquirido.

O sensor barométrico optado inicialmente é o MS5611. Conforme TE Connectivity (2017), as características são:

- Resolução: 10 cm - modo de alta precisão;
- Consumo de potência:  $< 0.15 \mu A$ ;
- Tensão de alimentação: 1.8 V até 3.6 V;
- Sensor de pressão integrada: 24 bits;
- Protocolo de comunicação: I2C e SPI com até 20MHz;
- Temperatura de operação:  $-40^{\circ}C$  até  $+85^{\circ}C$ .

Como substituto pode ser utilizado o BMP180, cujas características são (BOSCH, 2013):

- Resolução: 17 cm - modo de alta precisão
- Consumo de potência:  $< 32 \mu A$  - modo de alta precisão;
- Tensão de alimentação: 1.8 V até 3.6 V;
- Sensor de pressão integrada: 16 bits;
- Protocolo de comunicação: I2C;
- Temperatura de operação:  $-40^{\circ}C$  até  $+85^{\circ}C$ ;
- Tempo de conversão: Max 51 ms - modo de alta precisão.

#### 6.1.4 Comunicador

O módulo de comunicação inicialmente escolhido é o NRF24L01, e possui, de acordo com Nordic Semiconductor (2006), as características:

- Velocidade de Transmissão: 1  $Mbps$  ou 2  $Mbps$ ;
- Protocolo de comunicação: SPI;
- Tensão de alimentação: 1.8  $V$  até 3.6  $V$ ;
- Consumo de corrente em modo TX com ganho de 0  $dB$ : 11.3  $mA$ ;
- Consumo de corrente em modo RX: 12.3  $mA$ ;
- Frequência: 2.4 – 2.5  $GHz$ .

#### 6.1.5 Estrutura

De forma a cumprir os requisitos **REQ 001, 008, 010 e 011**, a estrutura externa proposta é feita em polímero utilizando uma impresso 3D de forma a se adequar aos requisitos e ter menor densidade que estruturas de metal. Além disso, é possível modificar a estrutura interna para acomodar os componentes de forma adequada. A primeira versão da estrutura (Figure 6.3) utiliza o PLA, com densidade de aproximadamente  $1.24 g/cm^3$ . A espessura da estrutura externa está projetada inicialmente como tendo 2  $mm$ .

Figura 6.3 – Estrutura externa em PLA.



Fonte: Autores.

### 6.1.6 Paraquedas

De modo a cumprir os requisitos **REQ. 006 e 008**, deve-se estudar as possibilidades de tipos e formas de acionamento do sistema de paraquedas.

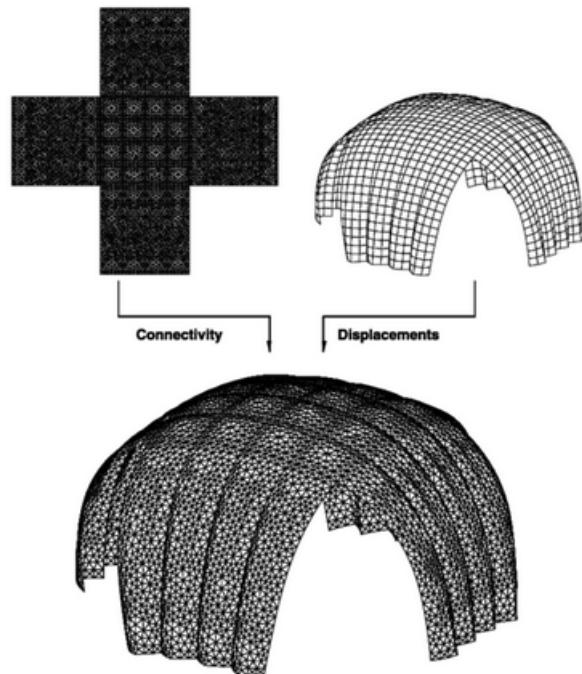
Uma vez que o tamanho e o peso dos componentes do CanSat devem ser controlados de maneira a não ultrapassar os limites estipulados na Seção 2, é necessário analisar o formato e as dimensões do sistema de paraquedas. Conforme Brohm (2004), a área mínima necessária para que o sistema de paraquedas leve o CanSat de maneira segura ao chão é dada pela equação:

$$A_p = \frac{2gm}{\rho C_d V^2} \quad (6.1)$$

Onde  $A_p$  é a área do paraquedas [ $m^2$ ], g a aceleração da gravidade [ $9,81 \text{ m/s}^2$ ], m é a massa do paraquedas [ $kg$ ],  $\rho$  é a densidade do ar à nível do mar [ $1,225 \text{ kg/m}^3$ ],  $C_d$  é o coeficiente de arrasto do paraquedas (que varia conforme o seu formato) e V é a velocidade de descida (ou velocidade terminal) do sistema [ $m/s$ ].

Para este conceito, a categoria de paraquedas *cross* (também conhecido por *cruciform*) foi selecionada. A Figura 6.4 apresenta como é esta configuração. Neste conceito, têm-se a presença de cinco quadrados de mesma área que, assim que juntos, formam a estrutura em "X".

Figura 6.4 – Formato do paraquedas tipo *Cross*



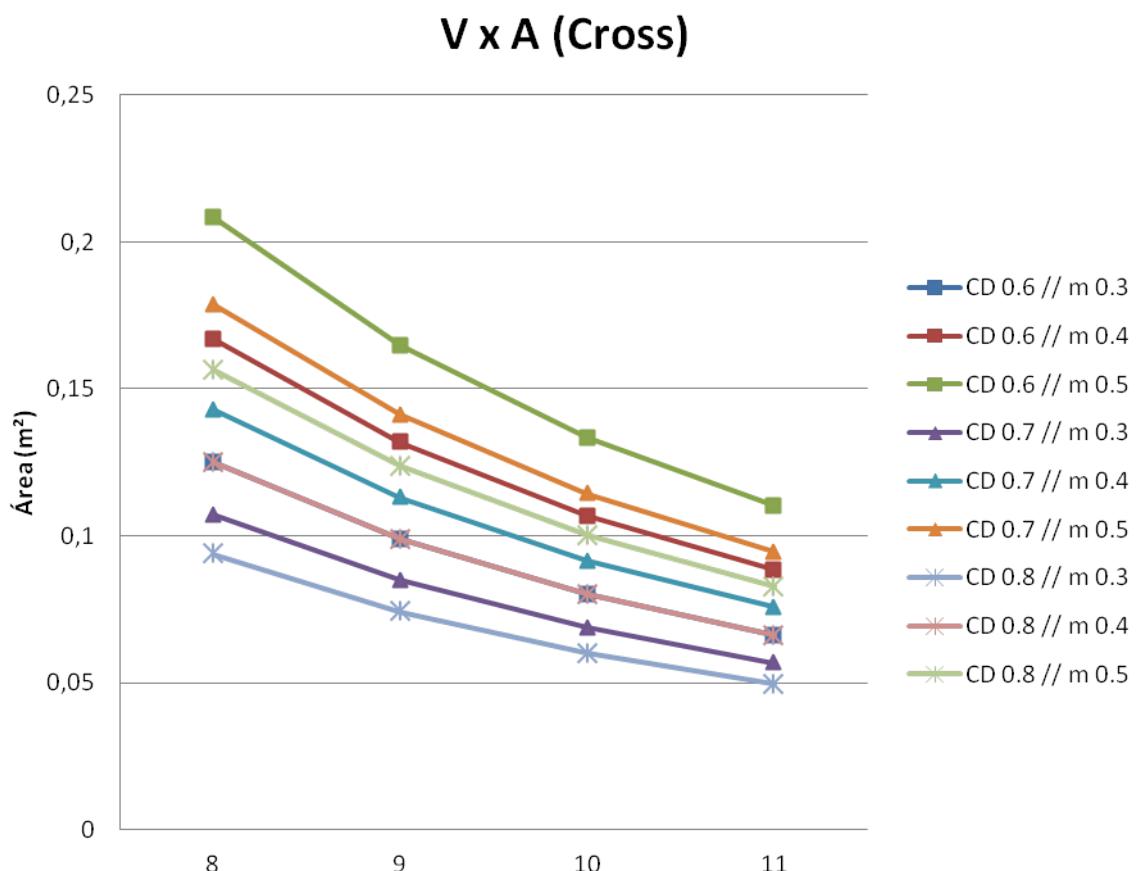
Fonte: Stein et al. (2001).

Conforme McWilliams (2013) e ESA (2018), este formato de paraquedas provê ao sistema redução de oscilação e descidas mais estáveis. Além disso, esta configuração

pode diminuir a velocidade de descida em até 30%. De suas características, citam-se a facilidade de montagem à mão e a pouca utilização de material. Esta configuração é popular na construção de CanSats, pois apresenta uma boa relação eficiência/peso, uma vez que seu coeficiente de arrasto varia entre 0.6 a 0.8.

Uma vez que as normas da competição não apresentam valores específicos para a velocidade terminal,  $V$ , do satélite, buscou-se em outras competições os valores médios para a mesma. Nos regulamentos da ESA (2018) e da ASC (2018), encontrou-se que os valores médios para a velocidade  $V$  variam entre 8 e 11  $m/s$ . Assim, pode-se calcular os valores de áreas do paraquedas para a configuração *cross*. A Figura 6.5 apresenta os valores de área do paraquedas em relação à velocidade terminal do sistema, assim como variando os valores de  $C_d$  e massa do CanSat.

Figura 6.5 – Variação da área do paraquedas com a velocidade terminal para o paraquedas tipo *Cross*.



Fonte: Autores.

Nota-se pelo gráfico apresentado na Figura 6.5 que a menor área do paraquedas encontrada é de  $A \approx 0,05 m^2$  para o caso de  $C_d = 0,8$  e massa  $m = 0,3 kg$ . Dessa forma, cada um dos cinco quadrados presentes na configuração apresentam áreas iguais à  $A_q \approx 0,01 m^2$ . No entanto, este é visto como o melhor caso para a configuração do sistema, onde têm-se a melhor configuração do paraquedas (obtendo maior valor de  $C_d$ ) e

a menor massa, conforme estipulado na Seção 1.1.2 - e pode não ser atingido durante a confecção do sistema de paraquedas-, mas também apresenta a maior velocidade terminal - o que leva a um maior impacto no solo. Caso a velocidade terminal seja mínima ( $8 \text{ m/s}$ ), obtém-se os seguintes valores:  $A_p \approx 0,093 \text{ m}^2$  e  $A_q \approx 0,0186 \text{ m}^2$ , um aumento de 86% na área comparado ao melhor caso.

Quanto à escolha de materiais, existem diversos que apresentam resultados satisfatórios na confecção de paraquedas, o mais utilizado é o nylon "rip stop", no entanto este possui custo elevado e não foi encontrada em regiões próximas à Santa Maria. Outra opção muito utilizada neste setor é o tecido de poliéster Dácron, que possui preços inferiores ao nylon e apresenta características próximas do mesmo. Dessa forma, o tecido Dácron foi selecionado para a confecção do paraquedas.

### 6.1.7 Componentes

De maneira a facilitar a identificação dos componentes a serem analisados os seguintes itens são apresentados:

**C 001** TM4C123G ;

**C 002** NRF24L01;

**C 003** MS5611;

**C 004** MPU9250;

**C 005** Bateria;

**C 006** Servo;

**C 007** Paraquedas;

**C 008** Estrutura interna;

**C 009** Estrutura externa;

**C 010** Arduino Mega;

**C 011** Antena;

**C 012** Cabos;

**C 013** Buzzer;

**C 014** Computador de solo.

### **6.1.8 Análise de Requisitos**

A tabela 6.2 apresenta a relação dos componentes do sistema com os requisitos estipulados na Seção 2 utilizando do Conceito A:

Tabela 6.2 – Relação dos componentes do sistema e requisitos de missão conforme Conceito A.

Requisitos/Componentes	C 001	C 002	C 003	C 004	C 005	C 006	C 007	C 008	C 009	C 010	C 011	C 012	C 013	C 014
REQ_ID 001														
REQ_ID 002														
REQ_ID 003														
REQ_ID 004														
REQ_ID 005														
REQ_ID 006														
REQ_ID 007														
REQ_ID 008														
REQ_ID 009														
REQ_ID 010														
REQ_ID 011														
REQ_ID 012														
REQ_ID 013														

## 6.2 CONCEITO B

A principal diferença entre o conceito B e o conceito A é em qual etapa da missão o processamento refinado dos dados ocorrerá. No conceito A, o processamento é feito *onboard*, sendo o resultado final - altitude - transmitido via rádio, sendo que os dados não são armazenados durante a missão.

O conceito B é caracterizado pelo armazenamento dos dados coletados, sendo que a aplicação do filtro de Kalman é feita em solo. Durante o voo, o computador de bordo realiza cálculos superficiais para determinar se o satélite está subindo, descendo ou em solo. Isso para poder determinar o momento de abertura do paraquedas e iniciar a missão em solo. Dessa forma, ainda empregam-se os sensores MPU9250 e o MS5611. A estrutura externa é a mesma apresentada anteriormente. Além do mais, esse conceito permite que a equipe de solo escolha quais dados serão utilizados para efetivamente reconstruir a altitude durante o voo. Portanto, caso os dados coletados pelo MPU9250 se mostrem não confiáveis, ou o modelo com o filtro de Kalman aplicado não esteja condizente com o esperado, pode-se utilizar somente os dados obtidos pelo barômetro.

O esquema de ligações elétricas dos componentes é apresentado no apêndice B.

### 6.2.1 Balanço de Massa

Tabela 6.3 – Balanço de Massa

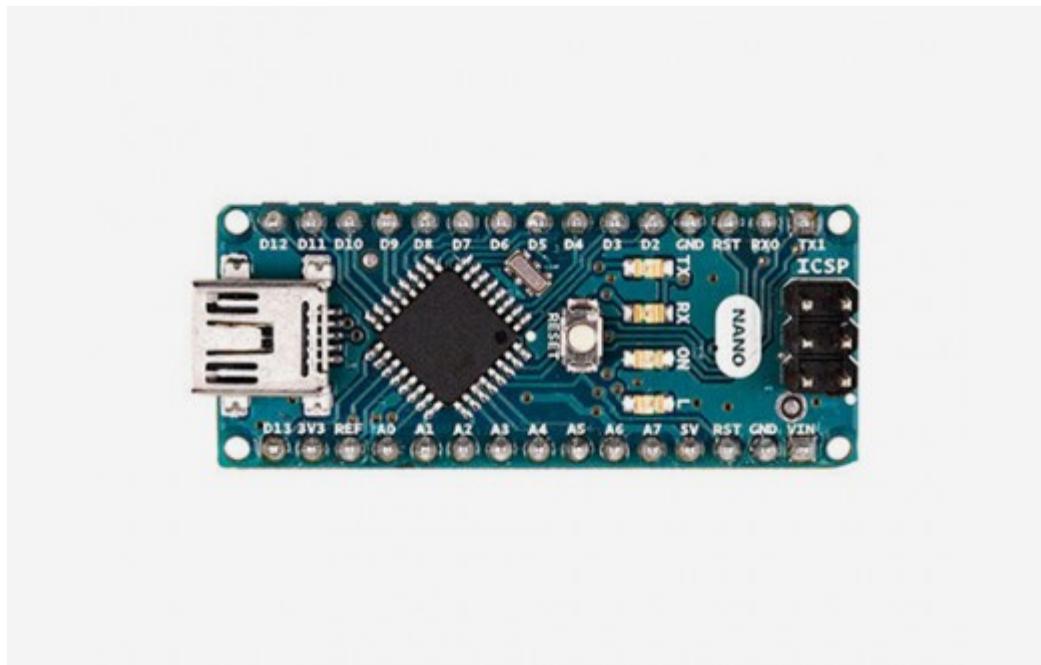
Arduino Nano	7 g	Servo	11 g
Adaptador para gravação em Cartão microSD	3 g	Paraquedas	
Cartão microSD	1 g	Cabos	
MPU6050+Magnetômetro	1 g	Bateria	42 g
Barômetro	1 g	Buzzer	2 g
Estrutura interna		Total	
Estrutura externa	60 g		$\leq 300$ g

### 6.2.2 Dados dos Microcontroladores

As características do Arduino Nano são:

- Tensão de operação: 5 V ;
- Tensão de entrada: 7 V até 12 V ;
- Flash Memory : 32 KB;

Figura 6.6 – Arduino Nano



Fonte: Arduino (2019a).

- SRAM: 2 KB;
- EEPROM: 1 KB;
- Clock Speed : 16 M Hz;
- Número de pinos digitais: 13;
- Número de pinos analógicos: 8;
- Número de portas com PWM: 6;
- Protocolos de comunicação: UART, SPI, I2C.

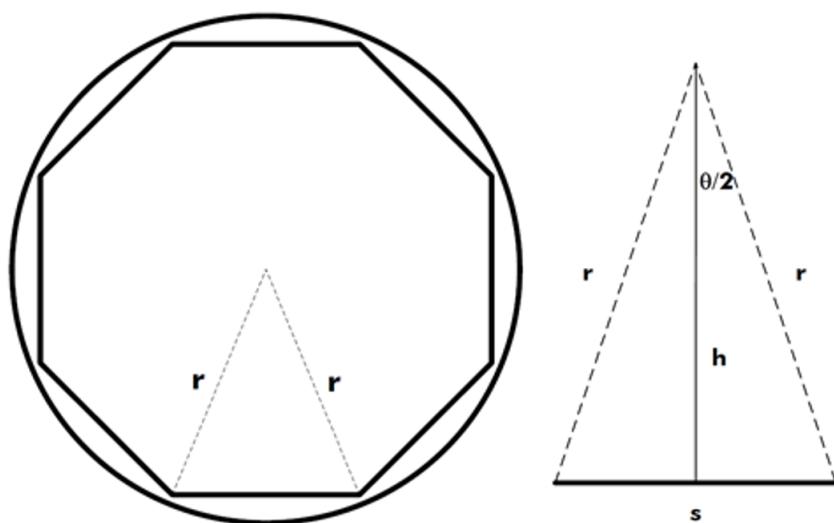
### 6.2.3 Estrutura

A estrutura externa é mesma apresentada no Conceito A, uma vez que a estrutura deve cumprir os regulamentos da competição e os requisitos **REQ 001**, **REQ 008**, **REQ 010** e **011**.

#### 6.2.4 Paraquedas

De maneira a cumprir as mesmas funções estipuladas no Conceito A, a configuração do sistema de paraquedas escolhido para o conceito B é o *flat octagonal*, conforme apresenta a Figura 6.7. Esta configuração, por sua vez, apresenta características semelhantes ao tipo *Cross*, uma vez que também é muito utilizada na confecção de CanSats, pois também possui facilidade de montagem. Além disso, seus coeficientes de arrasto apresentam melhoria quanto à configuração *cross*, uma vez que seus valores de  $C_d$  variam de 0,75 a 0,8 (BROHM, 2004; ASC, 2018).

Figura 6.7 – Formato do paraquedas tipo *flat octagonal*.



Fonte: Brohm (2004).

Neste modelo, o sistema de paraquedas fica próximo aos convencionais vistos em atividades de paraquedismo, onde a superfície do canopi apresenta superfície circular (ou próxima). Aqui, a superfície circular é dividida em um polígono de  $n$  lados, em geral  $n$  varia de 6 a 8, os vértices deste polígono são tangentes ao círculo, e sua distância até o centro do mesmo é igual ao raio,  $r$ . Na Figura 6.7, pode-se notar que duas linhas são apresentadas, com origem no centro do círculo e fim nos vértices do polígono, formando um triângulo isósceles. Para este projeto, o polígono escolhido apresenta 8 lados (octógono). A área total do octógono - que representa a área do paraquedas  $A_p$  - pode ser dividida em 8 triângulos isósceles idênticos de área  $A_t$ . As áreas do paraquedas e dos triângulos são dadas conforme as equações:

$$A_p = 8A_t \quad (6.2)$$

$$A_t = \frac{sh}{2} \quad (6.3)$$

Onde  $s$  é a base do triângulo [m] e  $h$  a altura [m]. Estas podem ser escritas em função do raio  $r$ , tal que:

$$s = 2r \sin\left(\frac{180^\circ}{n}\right) \quad (6.4)$$

$$h = r \cos\left(\frac{180^\circ}{n}\right) \quad (6.5)$$

Modificando a Equação 6.3 e aplicando as devidas transformações trigométricas, encontra-se que:

$$A_t = \frac{r^2}{2} \sin\left(\frac{360^\circ}{n}\right) \quad (6.6)$$

Lembrando que  $r = d/2$ , tem-se que área do paraquedas  $A_p$  é dada por:

$$A_p = \frac{nd^2}{8} \sin\left(\frac{360^\circ}{n}\right) \quad (6.7)$$

Como o polígono já foi estipulado para ser um octágono, uma forma de auxiliar na construção do paraquedas é modificar a Equação 6.7 em função do diâmetro,  $d$ , do paraquedas/círculo, tal que (BROHM, 2004):

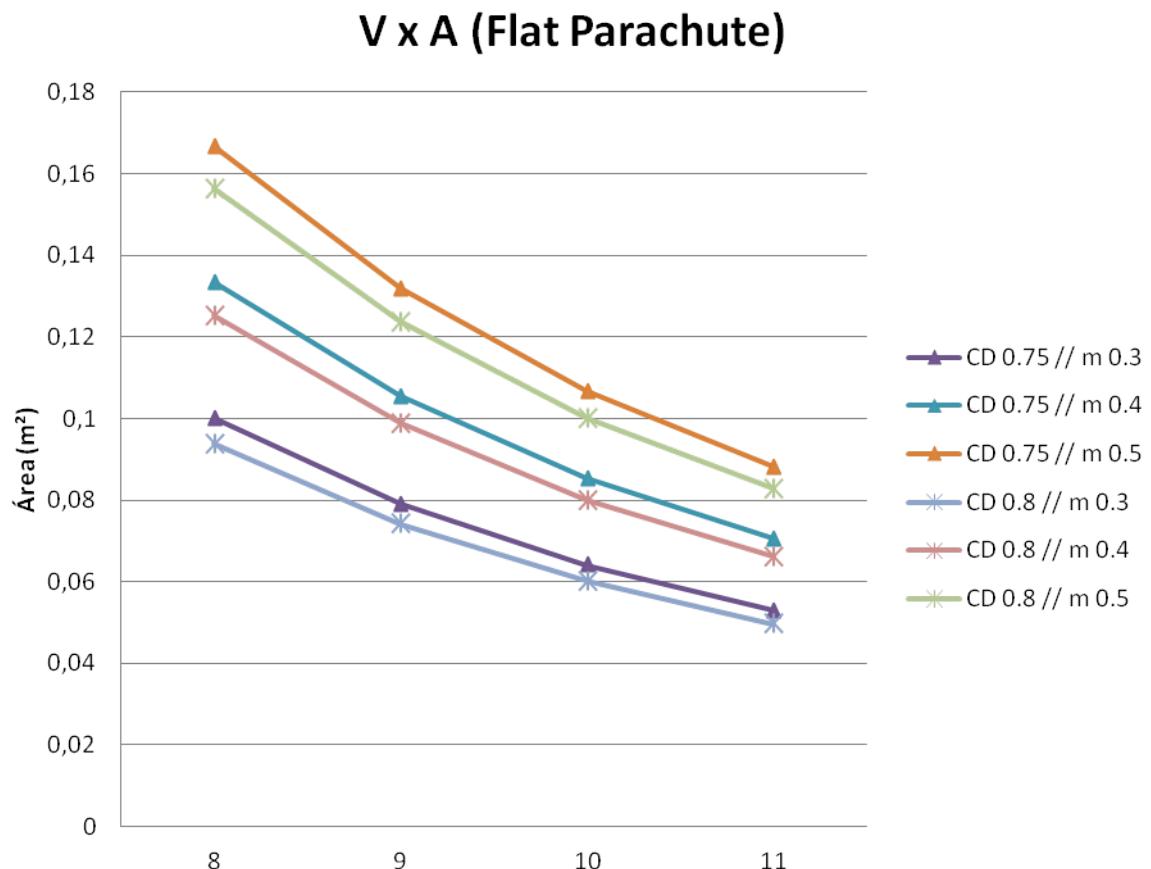
$$d = 2 \sqrt{\frac{2A_p}{nsin\left(\frac{360^\circ}{n}\right)}} \quad (6.8)$$

Para  $n = 8$ :

$$d \approx 1.1892 \sqrt{A_p} \quad (6.9)$$

Quanto ao cálculo da área do paraquedas, este é feito da mesma maneira como apresentado na Equação 6.1 da Seção 6.1.6. Os valores de  $A_p$  para o paraquedas do tipo *flat octagon* são apresentados na Figura 6.8. Pode-se notar que as áreas do paraquedas encontradas são as mesmas que para o Conceito A somente o intervalo de menor e maior área foi modificado. Além disso, como visto em ambas as configurações, a distribuição da área é feita de maneira distinta.

Figura 6.8 – Variação da área do paraquedas com a velocidade terminal para o paraquedas tipo *Flat Octagonal*.



Fonte: Autores.

Assim, o diâmetro do paraquedas encontrado para o caso de menor área ( $A_p \approx 0.5 \text{ m}^2$ ) é igual a  $d \approx 0,266 \text{ m}$ , enquanto que para o caso de menor velocidade ( $A_p \approx 0.093 \text{ m}^2$ ) é igual a  $d \approx 0,362 \text{ m}$ . Novamente, acredita-se que os valores dos coeficientes de arrasto podem ficar abaixo dos ideais ( $C_d < 0,8$ ), assim as áreas podem sofrer pequenas variações durante a confecção do paraquedas.

### 6.2.5 Escolha dos Sensores

As opções de sensores são as mesmas disponíveis para o conceito A.

### 6.2.6 Componentes

De maneira a facilitar a identificação dos componentes a serem analisados os seguintes itens são apresentados:

**C 001** Arduino Nano;

**C 002** Adaptador de cartão SD;

**C 003** Cartão SD;

**C 004** MS5611;

**C 005** MPU9250;

**C 006** Bateria;

**C 007** Servo;

**C 008** Paraquedas;

**C 009** Estrutura interna;

**C 010** Estrutura externa;

**C 011** Buzzer;

**C 012** Cabos;

**C 013** Computador de solo.

### **6.2.7 Análise de Requisitos**

A Tabela 6.4 apresenta a relação dos componentes do sistema com os requisitos estipulados na Seção 2 utilizando do Conceito B:

Tabela 6.4 – Relação dos componentes do sistema e requisitos de missão conforme Conceito B.

### 6.3 TESTES

De forma a identificar a funcionalidade do sistema, assim como a eficiência dos sensores citados na seção anterior, diversos testes foram realizados. Dentre eles, foram realizados testes de comunicação em diferentes distâncias de forma a identificar possíveis problemas na transmissão de dados entre os comunicadores. O Anexo A apresenta o relatório de teste realizados.

### 6.4 CONCEITO ESCOLHIDO

O conceito B foi escolhido devido a maior confiabilidade do método de aquisição dos dados coletados pelo satélites. Como mostram os relatórios de testes apresentados, os comunicadores *wireless* se mostraram ou com pequeno alcance, ou apresentaram grandes instabilidades. Assim sendo, o conceito A foi descartado.

Além disso, o balanço de massa do conceito B se mostrou melhor que o do conceito A. Caso o peso do paraquedas, cabos e estrutura interna sejam desconsiderados em ambos (por serem similares), o conceito B apresenta 18 gramas a menos e, portanto, apresenta vantagens na pontuação durante a competição. Em relação aos custos, o conceito A se mostra mais caro, devido ao comunicador, computador de bordo e receptor em solo, como indica a Tabela 6.5. A tabela 6.6 apresenta os custos do conceito B.

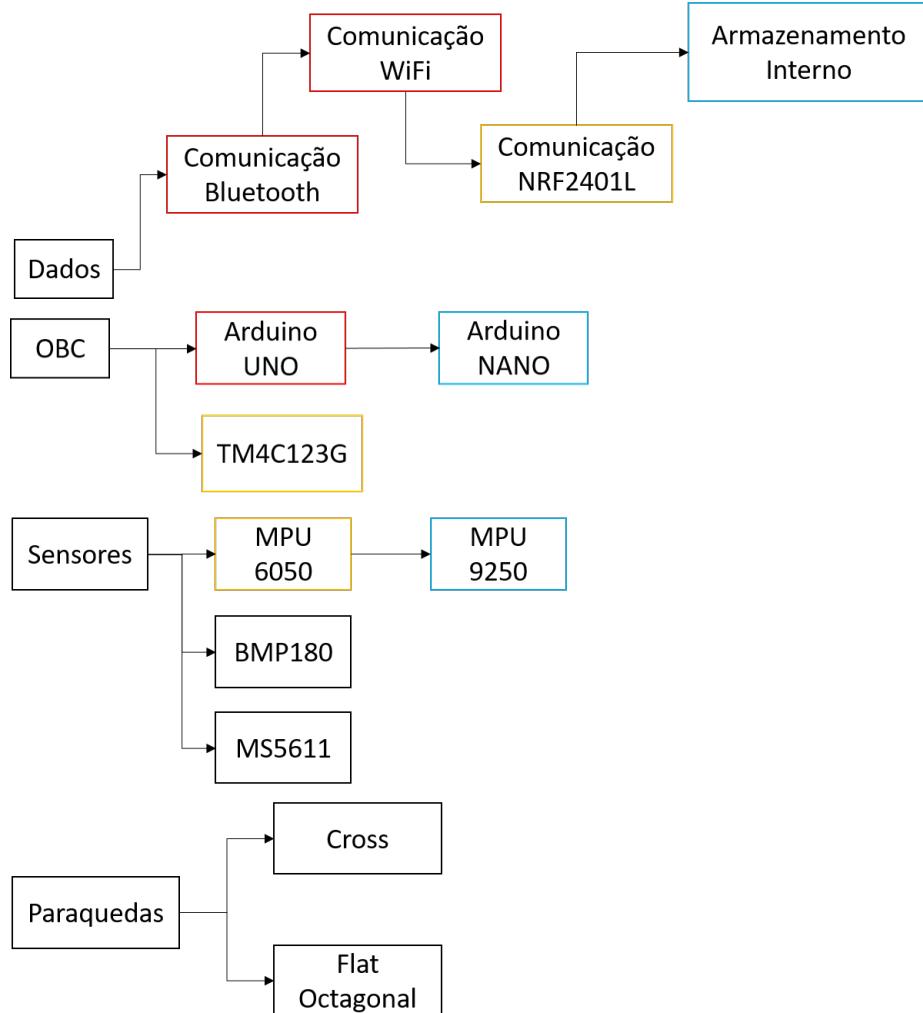
Tabela 6.5 – Tabela de custo do Conceito A

Componente	Custo
Arduino Uno	55 reais
NRF2401L	30 reais
MPU9250	26 reais
MS5611	49 reais
Arduino Mega	149 reais
Dácron	10 reais/m <sup>2</sup>
Total	309

Tabela 6.6 – Tabela de custo do Conceito B

Componente	Custo
Arduino NANO	39 reais
Adaptador MicroSD	15 reais
MPU9250	26 reais
MS5611	49 reais
Dácron	10 reais/m <sup>2</sup>
Total	129

Figura 6.9 – Arvore de Decisões



A sequência de escolha e decisões tomadas é ilustrada na Figura 6.9, onde é apresentado a ideia temporal dos testes e decisões tomadas. Em vermelho estão os componentes considerados impróprios para uso no projeto, enquanto que os em amarelo apontam problemas nas sua utilização. Os componentes marcados em azul são indicados para as aplicações desejadas. É importante ressaltar que os modelos de paraquedas foram arbitrariamente alocados em cada conceito, entretanto poderiam ser alternados sem perdas para a configuração ou escolha do conceito (ambos os paraquedas são opções viáveis tanto para o conceito A como para o conceito B). Inicialmente planeja-se empregar o modelo Flat Octagonal devido a sua menor variação da taxa de CD entretanto, caso surjam dificuldades para realizar sua dobra ou mesmo manufatura pode-se optar pelo modelo Cross.

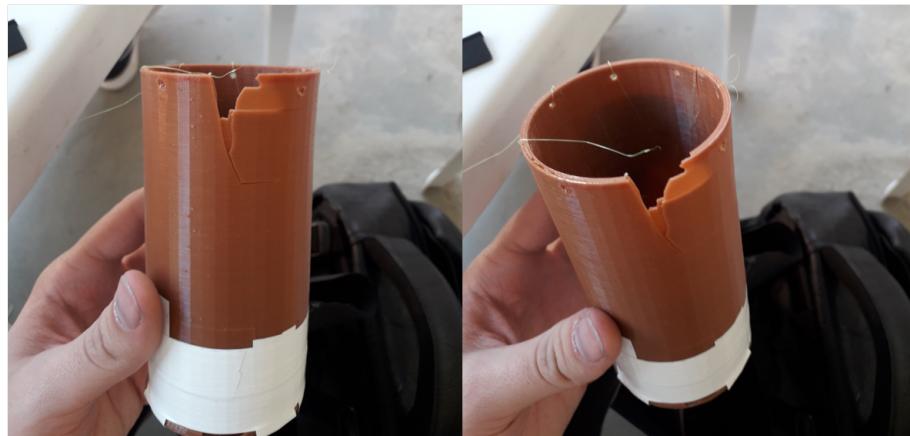
## 7 LIÇÕES APRENDIDAS

Com o decorrer da competição algumas lições foram aprendidas, envolvendo os problemas enfrentados, soluções adotadas e até mesmo aprendizado com outras equipes. Assim, serão apresentados os aspectos considerados mais relevantes.

A principal dificuldade enfrentada durante a competição foi uma falha no Arduino NANO, que não permitia que o *upload* de novos códigos para o dispositivo. Assim, novas rotinas de programação não podiam ser implementadas e os erros encontrados não podiam ser corrigidos. Mesmo procurando e tentando diversas abordagens, não foi possível solucionar o problema. Dentre as diversas tentativas, pouquíssimas eram capazes de atualizar o *software* embarcado e, portanto, não foi possível realizar testes de abertura de paraquedas de forma eficiente, nem corrigir os erros que faziam a missão de solo iniciar antes do esperado. O problema enfrentado piorou dois dias antes da competição, mas já havia ocorrido esporadicamente. Uma forma de contornar isso seria possuir um Arduino NANO de reserva para substituições.

Com o erro de *software* no Arduino, alguns outros problemas surgiram, tais como dificuldade de abertura do paraquedas e de missão de solo. A abertura do paraquedas de forma automática não ocorreu durante o primeiro lançamento e, portanto, o CanSat sofreu grande impacto com o solo após o lançamento, conforme pode ser visto na Figura 7.1. Tanto a estrutura interna como a estrutura externa foram danificadas, além de algumas conexões necessitarem de novas soldas sendo que diversos reparos foram necessários para torna-lo novamente operacional para o segundo lançamento. Percebe-se que o uso de sistema autônomo de paraquedas requer diversos testes, e está sujeito a várias dificuldades técnicas. Isso também é perceptível por todas as outras equipes presentes na competição apresentarem falha durante a abertura autônoma do paraquedas durante os lançamentos oficiais. Além dos danos físicos, grandes impactos como o sofrido pelo CanSat, podem tornar necessário fazer novamente a calibração das IMUs para a aplicação do filtro de Kalman.

Figura 7.1 – Estrutura externa após os lançamentos.



O erro inicialmente presente consistia em uma constante de variação de pressão ( $\Delta P$ ) a ser comparada pelo Arduino estar muito pequena. Quando o sensor MS5611 medisse uma variação de pressão superior à constante, o paraquedas se abriria. Entretanto, devido a constante estar muito pequena, a operação de abertura de paraquedas era iniciada mesmo com o CanSat parado e antes do lançamento. E, com o CanSat percebendo que a condição de abertura do paraquedas havia sido iniciada, a ação de missão de solo poderia começar assim que as medidas de pressão obtidas pelo MS5611 indicassem que o CanSat estava parado. Mas, com o erro para abertura do paraquedas, a missão de solo era iniciada antes da momento correto. De maneira geral, as falhas acima podem ser ligadas ao erro do Arduino e aos poucos testes realizados antes da competição.

Por fim, com a missão de solo iniciando antes do tempo correto, os dados obtidos pelos sensores MPU9250 e MS5611 não eram coletados e gravados no cartão de memória. Isso se deve ao uso do comando *delay* durante a execução da missão de solo. Novamente devido ao problema no Arduino, o erro não pode ser corrigido e nenhuma aquisição de dados foi realizada durante o primeiro lançamento. Umas das maneiras de corrigir o problema seria substituir o comando *delay* por algo similar (como uma comparação do tempo transcorrido) para que a aquisição de dados prosseguisse mesmo com a missão de solo em andamento.

Outro problema que surgiu durante a competição foi a estrutura interna ficar presa na estrutura externa durante o procedimento de *fitcheck*. O problema surgiu devido ao espaço ocupado pelos cabos ligando os componentes eletrônicos. Isso ocasionou diversas dificuldades de operação, principalmente durante as etapas de preparação do CanSat para lançamento, que deve ficar dentro do tempo limite de 2 minutos. Outra forma de conectar os componentes poderia ter sido adotada, ou então o espaço para cabos dimensionado - procedimento que não foi levado em consideração no desenvolvimento do CanSat. Outras arquiteturas e estruturas também são viáveis, como apresentado pelas outras equipes presentes na competição, estruturas desmontáveis ou rosqueadas, por exemplo.

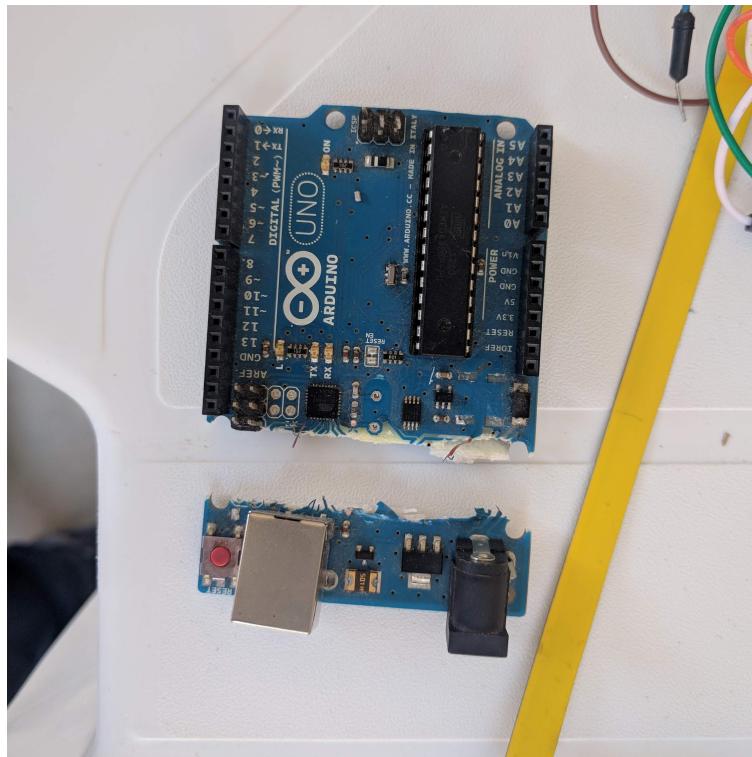
Para o segundo lançamento, o Arduino NANO foi substituído por um Arduino UNO

e, para isso, a estrutura interna foi removida e somente os componentes eletrônicos posicionados no interior do CanSat. Para o Arduino UNO alguns erros do *software* puderam ser corrigidos, tais como a missão de solo e a coleta de dados dos sensores. Mas pela maior dimensão do Arduino UNO quando comparado ao Arduino NANO, o atuador para abertura autônoma não poderia ser utilizado e, portanto, o modo de operação do paraquedas foi alterado pra pré-acionado. Infelizmente, devido a forma de posicionamento do paraquedas, a dimensão do CanSat (com paraquedas) superou os limites máximos estipulados pelas regras, o que acarretou na estrutura do CanSat não desacoplar do dispositivo lançador e novamente sofreu grande impacto com o solo. No segundo lançamento o Arduino UNO separou-se em dois devido ao impacto e portanto a missão de solo não foi realizada. O cartão de memória também foi danificado durante o pouso e, consequentemente, nenhum dado foi recuperado. A Figura 7.2 apresenta como ficou o Arduino UNO após o impacto catastrófico.

Em relação a condição de acionamento do paraquedas, inicialmente foi adotado que quando cinco medidas consecutivas fossem crescentes (indicando uma queda) o paraquedas seria acionado. Entretanto, devido aos erros de leitura do barômetro, essa condição não era obtida e, portanto, foi alterada para uma variação de pressão  $\Delta P$ , assumida inicialmente como  $10Pa$ . Esse valor foi obtido simulando as condições de abertura do paraquedas com a coleta de dados do CanSat em queda livre. Entretanto o valor de  $\Delta P$  foi posteriormente alterado para  $15Pa$  devido ao acionamento do paraquedas ainda em solo. Um  $\Delta P$  grande se torna necessário devido aos erros de leitura presentes no barômetro - que aumentam quando expostos ao Sol devido ao aumento de temperatura. O valor de  $10Pa$  foi obtido durante a noite em condições de temperaturas mais baixas.

A missão de solo, de forma similar, era iniciada quando os cinco últimos valores de pressão lidos se mantivessem dentro de um limite. Ou seja, o mínimo e o máximo valor dentre as cinco medidas não excedesse  $5Pa$ , entretanto devido ao erro de acionamento do paraquedas, foi adotada uma condição temporal para inicio da missão de solo. Assim, a missão de solo era iniciada 2.5 minutos após o CanSat ser ligado.

Figura 7.2 – Arduino UNO danificado após o segundo lançamento.



Como pontos positivos, a alta precisão das dimensões da estrutura externa foram decisivos na etapa de *fitcheck* e na pontuação final obtida na competição. O *fitcheck* é uma etapa eliminatória, mas somente foi atendida pela equipe Ready, Sat, Go!.

Além disso, a utilização de técnicas de Engenharia de Sistemas - durante todo o projeto do CanSat - foi um ponto positivo ressaltado pelos avaliadores durante a apresentação do projeto. É uma prática essencial e muitas vezes negligenciada durante as fases de um projeto. Ademais, possuir uma apresentação clara, com os objetivos, metodologia e resultados alcançados se mostrou um ponto positivo para a equipe Ready, Sat, Go!, além do relatório detalhado.

Por fim, deve-se ressaltar a necessidade de sempre ter dispositivos reservas disponíveis durante uma competição. A principal falha já mencionada neste projeto foi dada pelo mal funcionamento da eletrônica, o que poderia ter sido resolvida de maneira simplificada se existisse um Arduino NANO extra.

## 7.1 SUGESTÕES DE MELHORIAS

Considerando as lições aprendidas apresentadas acima, algumas sugestões de melhorias são apresentadas. Como sugestões para o sistema:

- Utilizar um sistema de paraquedas mais eficaz;

- Utilizar microprocessadores embarcados mais confiáveis;
- Empregar um sistema que facilite a integração da carga útil em pouco tempo;
- Desenvolver uma estrutura interna mais prática para reparo dos componentes, e de mais fácil colocação/remoção;
- Otimizar o *software* de bordo para filtragem do barômetro e rotinas de acionamento de paraquedas e missão de solo.

Como sugestões para o projeto em si:

- Seguir o cronograma de forma a evitar atrasos e não realização de atividades;
- Realizar os testes no caso mais próximo possível das condições reais da competição, de forma a tentar abranger todas as situações possíveis.

## 8 CONCLUSÃO

De maneira a produzir o primeiro CanSat desenvolvido na UFSM, este projeto utilizou a competição 2º CubeDesign como forma de guiar o projeto e testar o desempenho do modelo produzido. A partir dos requisitos adotados com base nos regulamentos da competição, assim como focando nos quesitos que forneciam uma maior pontuação - aumentando as chances de uma melhor classificação final - foi possível desenvolver um conceito de CanSat que cumprisse com as funções estipuladas e iniciar sua montagem e integração. Além disso, de maneira a apresentar este como um possível documento para futuros projetos de CanSats, o trabalho foi apresentado de maneira instrucional, mostrando algumas possibilidades de CanSat e seus vantagens/desvantagens.

Para que fosse possível determinar uma boa forma de operação para o CanSat, dois conceitos foram criados e seus componentes e características levantados. Com base nessas informações e nos balanços de massa e custo, foi possível definir qual dos dois conceitos melhor se enquadram nos requisitos apresentados e qual deles poderia apresentar o melhor desempenho. Com isso optou-se pelo segundo conceito, o qual utiliza um adaptador de cartão MicroSD para armazena os dados coletados em voo, para que então o pós processamento ocorra em solo.

Para a determinação da altitude, optou-se por um filtro de Kalman que faça a integração das medidas inerciais com os sensores de magnetômetro e barômetro. Essa é uma medida adotada para tentar reduzir os erros que um sensor barométrico sozinho ocasionalmente, além de ser possível determinar quais medidas serão efetivamente utilizadas para obter a altitude (caso os valores coletados pelo MPU não se mostrem adequados, pode-se utilizar somente os do MS5611). Ao final, as etapas propostas (apresentadas no Gantt no capítulo 3) foram totalmente concluídas, exceto "Desenvolvimento de um sistema de pouso", "Testes do sistema de pouso", "Desenvolvimento do *software* de bordo", "Testes do modelo final" e "Correção de erros".

Durante a competição, diversos problemas foram enfrentados, principalmente devido a falhas no Arduino NANO. Ainda assim, a equipe Ready, Sat, Go! obteve primeiro lugar na competição na categoria CanSat. Com isso, é possível afirmar que os objetivos propostos - apresentados no capítulo 1.0.1 - de desenvolver um CanSat de baixo custo para competições, assim como documentar o processo e decisões tomadas foram atingidos com sucesso.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ARDUINO. **Arduino Nano**. 2019. Disponível em: <<https://store.arduino.cc/usa/arduino-nano>>.
- \_\_\_\_\_. **Arduino Uno Rev3**. 2019. Disponível em: <<https://store.arduino.cc/usa/arduino-uno-rev3>>.
- ASC. **Parachute Design**. 2018. Disponível em: <<https://www.narom.no/undervisningsressurser/the-cansat-book/the-primary-mission/parachute-design>>.
- BOSCH. **BMP180 Digital pressure sensor**. 2.5. ed. [S.I.], 2013.
- BROHM, J. R. **The Mathematics of Flat Parachutes**. [S.I.], 2004.
- CUBEDESIGN. 2º **Cube Design**. 2019. Disponível em: <<http://www.inpe.br/cubedesign/2019/>>.
- ESA. **Design Your Parachute – A Guide to Landing Your CanSat Safely | Teach with Space T10**. [S.I.], 2018.
- INVENSENSE. **MPU-6000 and MPU-6050 Product Specification Revision 3.4**. 3.4. ed. 1197 Borregas Ave, Sunnyvale, CA 94089 U.S.A., 2013.
- MCWILLIAMS, P. **A summary of different parachute types**. 2013. Disponível em: <<https://awe365.com/a-summary-of-different-parachute-types>>.
- NORDIC SEMICONDUCTOR. **PRELIMINARY PRODUCT SPECIFICATION: Single chip 2.4 GHz Transceiver - nRF24L01**. 1.2. ed. Vestre Rosten 81, N-7075 Tiller, Norway, 2006.
- STEIN, K. et al. Fluid-structure interactions of a cross parachute: numerical simulation. **Computer Methods in Applied Mechanics and Engineering**, v. 191, p. 673–687, 12 2001.
- TE CONNECTIVITY. **MS5611-01BA03 Barometric Pressure Sensor, with stainless steel cap**. [S.I.], 2017.
- TEXAS INSTRUMENTS INCORPORATED. **Tiva™ TM4C123GH6PM Microcontroller**. 108 Wild Basin, Suite 350 Austin, TX 78746, 2014.

## APÊNDICE A – RELATÓRIO DE VERIFICAÇÃO

### A.1 TESTES DE COMUNICAÇÃO COM MÓDULO HC-05

O módulo HC-05 é um módulo bluetooth que possibilita a comunicação sem fio de um microcontrolador com dispositivos compatíveis, tais como *smartphones*, computadores, etc. Os testes para esse sistema consistiam em avaliar a distância máxima de comunicação, assim como o atraso de resposta baseado na distância.

O local de testes escolhidos foi a frente Centro Regional Sul do Instituto Nacional de Pesquisas Espaciais (INPE/CRS). Para tanto, um aluno ficava em um ponto fixo com o aparelho receptor (RX), enquanto outro se deslocava com o transmissor (TX).

Como resultado, foi avaliado que havia perda de sinal a uma distância superior a 20 metros e, portanto, o requisito **004** não era cumprido. Tentativas de melhorar a recepção e o envio de dados se mostraram ineficazes. Dessa forma, esse método de comunicação foi descartado.

### A.2 TESTES DE COMUNICAÇÃO COM ESP8266

O teste dessa forma de comunicação consistiu em testar o sensor em uma bancada e avaliar a taxa de dados e atualização das informações, que foi avaliada como insuficiente - não cumprindo o requisito **004**. Alguns outros problemas poderiam surgir pela forma como a comunicação é estabelecida, como perda de dados e estouro de memória. Dessa forma, esse método de comunicação foi descartado.

### A.3 TESTES DE COMUNICAÇÃO COM NRF24L01

Os testes de comunicação realizados com o nRF24L01 foram feitos na frente do INPE/CRS, com um aluno permanecendo em uma posição com o receptor, enquanto o outro se deslocava com o transmissor. O dispositivo de TX estava sem a antena para que o requisito **001** fosse atendido. As configurações utilizadas foram:

- TX: Ganho configurado para alto;
- TX: Velocidade de comunicação de 2 *Mbps*;

- RX: Velocidade de comunicação de 2 *Mbps*;
- Canal de comunicação: 001.

Nessas condições foram realizados diversos testes, que apontam uma capacidade de recepção a mais de 70 metros, satisfazendo o requisito **004**. Entretanto, durante os testes, os comunicadores apresentaram respostas intermitentes e não condizentes após a perda de sinal. Isso é um problema grave pois caso ocorra algo similar durante o voo do CanSat, os dados pós perda de sinal seriam perdidos. Esse problema foi corrigido via *software*, sendo os testes realizados para isso em local fechado e de curta distância.

Mesmo com a correção aplicada e a distância de comunicação excedendo 70 metros em alguns casos - cumprindo então o requisito **004**, algumas condições de testes indicaram perda do sinal por tempo prolongado ou distâncias muito inferiores aos 70 metros.

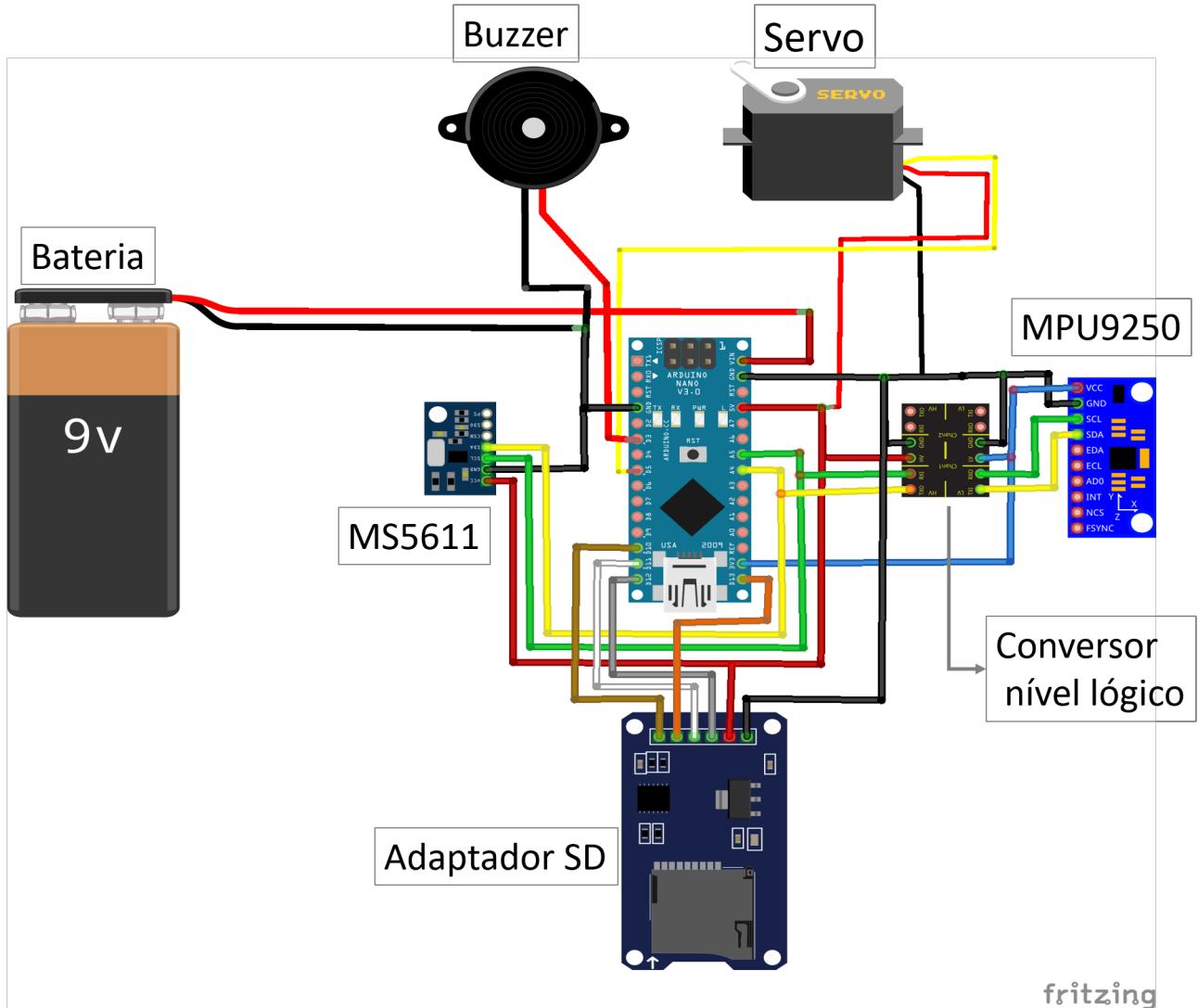
#### A.4 TESTE DE CONECTIVIDADE DOS COMPONENTES

Durante as etapas de montagem do modelo elétrico, teste dos sensores e desenvolvimento do *software* de bordo, os componentes foram integrados e testados. Avaliou-se o correto funcionamento, apresentando comportamento dentro do esperado. Mais testes serão necessários conforme mais sensores forem integrados ao sistema.

#### A.5 INSPEÇÃO DA ESTRUTURA EXTERNA

Após a fabricação por impressão 3D, a estrutura externa foi medida com um paquímetro para verificar se suas dimensões estavam de acordo com o estipulado pelo requisito **001**.

Figura B.1 – Circuito elétrico - Conceito B



## APÊNDICE B – SISTEMA ELÉTRICO - CONCEITO B

O esquema de ligações do sistema elétrico do conceito escolhido (conceito B) é apresentado na Figura B.1.

Importante ressaltar a operação do MPU9250 em 3.3V, assim gerando a necessidade de elevar a tensão de suas saídas digitais para 5V de forma a poder ser colocado no mesmo circuito com o MS5611 que opera em 5V. A Figura B.2 apresenta mais próximos os componentes MS5611, MPU9250, Arduino NANO e conversor de nível lógico.

Figura B.2 – Circuito elétrico - segunda vista

