



ZENITH AEROSPACE

ZENSAT 1.0

RELATÓRIO CUBEDESIGN

Membros na Competição:

Francesco Rossi Lena
João Matheus Siqueira Souza
Leonardo Fernandes Del Vale
Marco Aurélio Pagan Bonaldo
Tiago Bachiega de Almeida
Vinicius Eiji Sasaki

Professor Responsável:

Daniel Varela Magalhães

Colaboradores:

Caíque Abrahao Martins Danilo
Guilherme Victor Chiappa Bento
Orlando Wozniak de Lima Nogueira
Patrícia Di Cunto Bracco

Conteúdo

1	Introdução	3
1.1	Cubesats	3
2	Concepção de Missão	3
2.1	Descrição das necessidades	3
2.2	Descrição dos conceitos de operação da Missão	3
2.3	Descrição dos objetivos da missão	4
2.4	Descrição das medidas de efetividade da missão	4
2.5	Requisitos e justificativas da missão	4
3	Engenharia de Sistemas	4
3.1	Stakeholders	4
3.2	Metas, objetivos, medidas de efetividades e estratégias de qualificação	5
3.3	Cenários Operacionais	5
3.4	Restrições	5
3.5	Requisitos e justificativas da missão	5
3.6	Divisão dos Sistemas do Cubesat	6
3.6.1	ADC - Attitude Determination and Control	6
3.6.2	COM - Command and Telemetry	6
3.6.3	CTRL - Control	7
3.6.4	PS - Power Supply	7
3.6.5	STT - Structure	8
4	Design	8
4.1	Diagrama de fluxo de dados	8
4.2	Product Breakdown Structure	8
5	Construção	10
5.1	Projeto Estrutural	10
5.1.1	Simulação	10
5.1.2	Fabricação e Montagem	10
5.2	Projeto de Hardware	11
5.2.1	Power Supply	12
5.2.2	Commands and Telemetry	14
5.3	Projeto de Software	14
5.3.1	Funcionamento geral da biblioteca “zenith”	15
5.3.2	Comunicação entre Base e CubeSat - Padronizações	15
5.3.3	Inicialização	16
5.3.4	Missão 1 - Telemetria de serviço	16
5.3.5	Missão 2 - Checagem do subsistema de alimentação	16
5.3.6	Missão 3 - Estabilização em um eixo	16
5.3.7	Missão 4 - Determinação de Sol e Horizonte	16
5.4	Missão 5 - Apontamento	16
5.4.1	Missão 6 - Envio de imagem	16
5.4.2	Medição da temperatura na ciclagem térmica	17
5.5	Projeto de Controle e Determinação de Atitude	17
5.5.1	Hardware do ADC	17
5.5.2	Visão Computacional	19
5.5.3	Determinação do Sol e do Horizonte	20
6	Conclusão	20

Lista de Figuras

1	Comparação entre os diferentes tamanhos de Cubesats	3
2	Diagrama de blocos do ADC	6
3	Diagrama de funcionamento do sistema de comando e telemetria	6
4	Diagrama de Fluxo de Dados	8
5	Product Breakdown Structure do ZenSat 1.0	9
6	Resultado obtido para análise modal da estrutura.	10
7	Render do projeto CAD do ZenSat 1.0	10
8	Início da montagem do ZenSat	11
9	Estrutura do Zensat Montada	11
10	Conector ESQ-120-39-G-D da Samtec [3].	11
11	Mapeamento do ZHub	12
12	Padronização das dimensões das placas de circuito impresso do ZenSat	12
13	Render com destaque para o sistema PS do ZenSat	12
14	Placa de Circuito Impresso da placa Power Hub do ZenSat 1.0	13
15	Placa do após soldagem	13
16	Antena Patch na placa dos painéis solares	14
17	Antena com conector SMA	14
18	Exemplificação de um simples Node de comunicação	15
19	Render da localização do Hardware ADC no ZenSat	17
20	Esquemático simplificado da aplicação do driver DRV11873 da Texas Instruments	18
21	Adaptação de um módulo comercial com o driver DRV11873 para controle do motor apelidado carinhosamente de “Gambiarrinha Espacial”.	18
22	Placa de circuito impresso do ADC projetada no software Eagle CAD 9.0 e placa final montada.	18
23	Placa de circuito impresso do Encoder projetada no software Eagle CAD 9.0.	18
24	Montagem da placa do Encoder, da roda de reação com o motor e da caixa com o driver DRV11873.	18
25	Imagen de Saída com a Detecção do Sol e da Linha do Horizonte	20
26	Resultados gravado no arquivo	20

1 Introdução

Este relatório tem como objetivo a documentação dos processos de desenvolvimento do Cubesat experimental ZenSat 1.0 do grupo Zenith Aerospace da Escola de Engenharia de São Carlos - USP. O grupo tem como objetivo a construção do satélite para participação da competição Cubedesign[2] oferecida pelo grupo de Capacitação Tecnológica em Engenharia Espacial - CTEE da Pós-Graduação em Engenharia e Tecnologia Espacial do INPE Instituto Nacional de Pesquisas Espaciais[1].

Muitos dos princípios utilizados para a construção desse projeto foram baseados no livro *Fundamentals of Space Systems*[4], livro o qual apresenta uma boa introdução sobre sistemas espaciais e uma teoria relevante para a construção dos mesmos.

1.1 Cubesats

O surgimento da microeletrônica permitiu que sistemas embarcados pudessem ser criados em tamanhos cada vez mais reduzidos. Esse advento tecnológico oferece uma oportunidade muito grande para os entusiastas em pesquisas aeroespaciais, uma vez que consegue-se fazer sistemas complexos em pequenos espaços. Com sistemas assim, é possível baratear o custo de envio para a órbita terrestre e, ainda, realizar missões de complexidade elevada.

A partir disso, surgiu a ideia da categoria nanosats, os quais são satélites de tamanhos muito inferiores com relação a satélites usuais - com massas de 1 a 10 kg, mas que ainda podem realizar missões de alta complexidade. Chamados de Cubesats, eles cumprem o objetivo de baratear as missões espaciais, uma vez que seu peso e volume reduzidos permitem que sejam lançados em meio a payloads de tamanhos usuais sem que o custo sofra acréscimos relevantes.

Com a popularização desses satélites, foi necessário criar protocolos e normas para o desenvolvimento dos mesmos. Entre estas normas, estão o tipo de envelope desses nanosats. Classifica-se o "tamanho" dos satélites por U e cada um desses é uma unidade do Cubesat. Na Figura 1 é possível ver uma comparação de diversos tipos de cubesat.

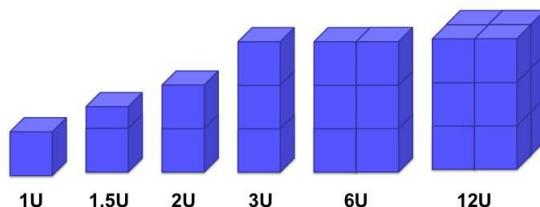


Figura 1: Comparação entre os diferentes tamanhos de Cubesats

2 Concepção de Missão

A primeira fase desta missão consiste no entendimento das missões propostas pelos organizadores da competição Cubedesign e na definição das necessidades do grupo para o desenvolvimento do Cubesat. Para isso foi analisado o regulamento da competição Cubedesign levando em conta as missões propostas.

2.1 Descrição das necessidades

Para a realização do projeto, foi necessário que o grupo conseguisse:

1. Organizar um grupo para desenvolver todas as missões em conjunto visando o aumento da integração funcional do Cubesat e pessoal das equipes.
2. Engajar o grupo para cumprir com as datas estipuladas
3. Conceber um projeto sólido e robusto para uma manufatura fluida e sem eventos inesperados.
4. Prezar pelo menor custo possível e procurar parcerias - principalmente serviços - e otimizar a utilização de recursos.
5. Realizar testes técnicos constantes e antes da finalização geral do projeto, os quais devem ser pautados nas missões.

2.2 Descrição dos conceitos de operação da Missão

A missão realizada pelo ZenSat 1.0 é baseada em missões de imageamento via satélite. Essas consistem na obtenção de imagens de algum alvo a partir do satélite e do envio das mesmas via telemetria.

O satélite deve ser capaz de estabilizar-se em torno de seus próprios eixos, além de ser capaz de realizar o apontamento para uma coordenada determinada por comandos.

No intuito de encontrar essa estabilidade, a espaçonave deve ter um sistema de determinação e controle de atitude para que esta tenha controle da sua orientação no espaço. Em outras palavras, é necessário que a missão determine a posição relativa do CubeSat com relação ao que está ao seu redor.

Além disso, são necessários procedimentos para a análise da saúde do satélite. Devem ser monitorados e ajustados alguns dos meios de operação para que o satélite não sofra com mal-funcionamento. Por exemplo, devem ser monitorados os dados de corrente e tensão dos painéis solares e das baterias de modo a fazer alterações nos modos de operação do satélite e, assim, garantir que não hajam falhas por falta de energia durante a execução de algum comando.

2.3 Descrição dos objetivos da missão

O objetivo da missão é a realização de uma missão de imageamento no Instituto Nacional de Pesquisas Espaciais (INPE).

Um ambiente simulado análogo à região de baixa órbita será criado e a equipe deve desenvolver um CubeSat para a realização da missão. Isto é, o satélite deve ser capaz de realizar todas as funções necessárias para que uma fotografia seja capturada e enviada a uma base de recebimento.

2.4 Descrição das medidas de efetividade da missão

Para o sucesso da missão, foram determinadas as seguintes medidas para computar a efetividade da missão:

1. Enviar e receber dados via telemetria com menor número de pacotes perdidos.
2. Receber por telemetria carregamento relevante nas baterias a partir das células solares. Segurar carga numa situação de eclipse (continuar funcionamento e estimar tempo de uso)
3. Estabilização por roda de reação. Otimizar a velocidade de estabilização
4. Entender a posição do horizonte com melhor precisão
5. Apontamento baseado na posição solar-horizonte.
6. Identificação do comando e maior rapidez na execução.
7. Captar uma imagem e recebê-la com sucesso na base de telemetria.

2.5 Requisitos e justificativas da missão

Os requisitos para o cumprimento dessa missão estão especificados na página da competição Cubedesign:

- Requisito 01 - O CubeSat deve ser de 1 à 3 U.
- Requisito 02 - Não precisa conter todos os subsistemas.
- Requisito 03 - Todos devem ter no mínimo o subsistema de estrutura.
- Requisito 04 - Todos os subsistemas devem ser desenvolvidos e construídos pelos alunos.
- Requisito 05 - A equipe deve trazer seus equipamentos de testes elétricos.
- Requisito 06 - O CubeSat deve seguir a *CubeSat Specification rev13[7]*.

3 Engenharia de Sistemas

3.1 Stakeholders

Para o desenvolvimento do Zensat 1.0, foi necessária a análise de instituições relacionadas ao grupo, as quais tem influência no decorrer da sua construção. Dentre essas estão:



- O grupo de Capacitação Tecnológica em Engenharia Espacial - CTEE da Pós-Graduação em Engenharia e Tecnologia Espacial do INPE é o idealizador da competição CubeDesign. Ela é uma iniciativa que promove um ambiente de competição saudável entre as equipes de desenvolvimento de pequenos satélites e, assim, incentiva o estudo dessas áreas. São responsáveis pelo cadastramento e realização da competição.



- A Escola de Engenharia de São Carlos da Universidade de São Paulo (EESC-USP), instituto no qual o grupo realiza suas atividades de desenvolvimento. É a responsável por permitir que os membros possam desenvolver o ZenSat 1.0. O sucesso na realização do projeto é interessante para o desenvolvimento acadêmico dos participantes e, consequentemente, causa uma melhora na qualidade da Universidade.

Patrocinadores do grupo, os quais ajudam financeiramente ou com serviços em troca de publicidade. Dentre os patrocinadores do projeto estão:



Ca And Ma - Componentes Eletrônicos



AcquaNativa Monitoramento Ambiental



Cerebra Sistemas de Controle



Igeagro Consultoria



Grupo Lena de Mineração e Terraplanagem

3.2 Metas, objetivos, medidas de efetividades e estratégias de qualificação

3.3 Cenários Operacionais

O satélite será colocado num ambiente espacial simulado, no qual deverá realizar uma série de missões pré-definidas pelos organizadores da competição CubeDesign. As missões são as seguintes:

1. A primeira missão consiste somente no envio e recebimento de uma telemetria de serviço.
2. Análise da corrente da bateria via telemetria no teste chamado de Sol-Eclipse.
3. A partir de meios iniciais, o CubeSat deve se estabilizar em torno de um eixo em 5 segundos.
4. O Cubesat vai determinar o Sol e o Horizonte, ou seja, irá descobrir os referenciais pelos quais irá se orientar.
5. O CubeSat irá utilizar os meios iniciais e a determinação para apontar para uma direção no plano perpendicular ao seu eixo de estabilização.
6. Será realizada uma captura fotográfica em qualquer resolução e a imagem resultante deverá ser enviada via telemetria

Além das missões supracitadas, há a necessidade de um cenário de operação de baixo consumo. Somente os componentes vitais do Cubesat deverão estar ligados e uma simples telemetria para análise da temperatura do mesmo será enviada. Esta será utilizada durante os testes ambientais

3.4 Restrições

As principais restrições do ZenSat 1.0 foram definidas pelo *CubeSat Specification rev13*, por questões financeiras e pela disponibilidade de tempo para a realização do projeto.

A estrutura física foi projetada com base nas especificações da *CubeSat Specification rev13*, sendo assim houveram diversas limitações de dimensão, peso e processos de fabricação que foram estabelecidas.

Diante do cronograma pré-estabelecido pelo grupo Zenith que contém outros projetos como o lançamento de sondas estratosféricas, a competição CubeDesign nos foi apresentada em torno de noventa dias antes das datas de entrega dos relatórios, como a maioria das equipes já estavam alocadas, o recurso humano foi um primeiro desafio, obrigando o grupo a realocar uma série de participantes para garantir ao desenvolvimento do projeto ZenSat 1.0.

As limitações financeiras que seriam um desafio a parte para as etapas como de a fabricação, obrigaram a equipe a repensar parcerias e buscar novos interessados em colaborar financeiramente com o grupo. O desdobramento desta busca possibilitou a abertura de novas linhas de negociação e chamou a atenção de novas oportunidades de parcerias, não só para este projeto, mas também para com toda linha de desenvolvimento do grupo Zenith.

Inicialmente, o grupo não possuía verba para a realização do projeto, sendo assim foram realizadas ações visando obtenção de fundos. Por meio de patrocínios foi possível arrecadar cerca de R\$5000,00 que foram utilizados no ZenSat 1.0 além da obtenção de vários materiais e componentes eletrônicos que eram indispensáveis para a realização do projeto.

3.5 Requisitos e justificativas da missão

Durante o planejamento da missão e observando as funcionalidades necessárias para completar a missão, foram definidos os seguintes requisitos para a construção do Cubesat:

- É necessária uma base de Telemetria. Essa terá por função o recebimento de dados do cubesat e o envio de comandos para as missões do mesmo.
- Com relação ao sistema de energia, decidiu-se que o consumo máximo por U é de 3W e os dados de corrente e tensão das baterias e painéis solares deverão ser obtidos e armazenados para envio por telemetria.
- Para a determinação de Atitude serão utilizados dois meios: Por obtenção dos dados de Giroscópio, Acelerômetro e Magnetômetro e por visão computacional, a qual irá analisar a posição do CubeSat com referência no Sol e Horizonte descobertos pela mesma.
- Estabilização em 1 eixo por roda de reação
- Manter-se estável apontando para um ponto definido

- Retirar foto em qualquer resolução de um alvo e enviar por telemetria
- Sistemas com processamento independente
- Seguir a *CubeSat Specification rev13*
- Massa: menor que 1,330g por U
- Custo: menor que 5.000 reais

A missão é justificada pela iniciativa de promover um ambiente de competição sadio entre equipes de desenvolvimento de pequenos satélites e, assim gerar troca de conhecimento e experiências nessa área.

3.6 Divisão dos Sistemas do Cubesat

3.6.1 ADC - Attitude Determination and Control

Este sistema tem como função a determinação e controle da atitude do ZenSat. Isso significa determinar e controlar como o satélite está orientado no espaço.

Para a determinação de atitude são utilizados recursos de visão computacional, os quais fazem a análise espacial e encontram a posição do sol e do horizonte e, assim, encontra-se uma referência para o satélite, a partir da qual o sistema de controle de atitude irá funcionar. Além disso, o sistema conta com a análise dos dados de uma IMU de 9 eixos. Assim, o satélite consegue entender seus dados de posição e o controle deste torna-se possível.

Para controlar a atitude por meios inerciais utiliza-se uma roda de reação. Rodas de reação são motores acoplados a um volante de inércia. O funcionamento baseia-se na compensação da rotação do sistema por meio da conservação do momento angular do CubeSat como um todo. A estabilidade do satélite acontece quando a roda atinge uma alta rotação, armazenando uma grande quantidade de momento e reduzindo drasticamente a influência de perturbações externas.

O sistema também conta um controlador PID que tem como função controlar a velocidade do motor para que, além de realizar a estabilização, permita o apontamento para uma coordenada pré-definida, fator necessário para que o CubeSat oriente-se na direção do Sol, para uma maior eficiência dos painéis solares, ou na direção de algum alvo.

Na figura 2 pode-se ver um sistema de determinação e controle de atitude geral de um satélite. A partir dela pode-se observar a malha superior responsável pelo controle da atitude, enquanto a realimentação abaixo é responsável pela determinação da atitude instantânea do CubeSat.

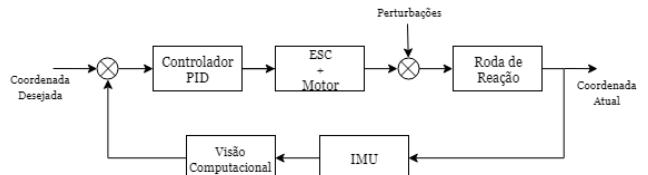


Figura 2: Diagrama de blocos do ADC

Foram determinados alguns requisitos para o funcionamento deste sistema, os quais estão citados a seguir:

- O sistema deve realizar o controle da atitude do satélite com a menor massa possível. No caso foi decidido que 500g deve ser a máxima massa.
- 3W máx de consumo
- Determinação rotação instantânea com encoder óptico
- ESC (Electronic speed control) para controlar a velocidade de rotação do motor que controla a roda de reação.
- Roda de latão, por conta da sua permeabilidade magnética.
- Determinação corrente
- IMU própria
- Medidor de tensão e corrente INA para fail safe

3.6.2 COM - Command and Telemetry

O sistema de telecomando e telemetria é responsável por realizar a comunicação bidirecional entre a base e o satélite. Os comandos são enviados para que o satélite realize o que desejamos, enquanto os dados de telemetria são enviados pelo mesmo com informações sobre seu estado atual. O sistema deve ser otimizado para que haja a menor probabilidade de falhas possível, uma vez que esse é a única madeira de comunicar-se com o satélite depois de seu lançamento.

Os comandos têm como propósito a reconfiguração das maneiras de operação do satélite. Por meio desses são ativados os modos

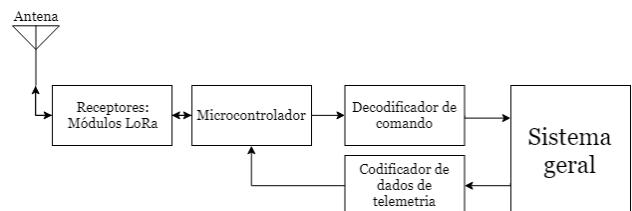


Figura 3: Diagrama de funcionamento do sistema de comando e telemetria

Requisitos gerais do sistema COM:

- Ser capaz de enviar e receber dados por modulação LoRa na faixa de 915MHz.

- Alcance teórico deve ser maior que 300 Km
- Antena patch (no deploy) impressa na placa dos painéis solares
- Base terrestre com redundância equipada com antena yagi comercial de recepção. Ela deve ser capaz de requisitar dados e enviar comandos para o satélite.
- Receber protocolos e gerar arquivo de atuação (funciona com CTRL)

3.6.3 CTRL - Control

O sistema de controle é responsável pela análise dos dados, tomada de decisões e controle dos outros sistemas.

Esse sistema deve ser bastante robusto, uma vez que há muitos dados sendo processados ao mesmo tempo.

- Dual core (2x Raspberry pi Zero W)
- 2 câmeras 170° responsáveis por adquirir imagens, as quais passarão por um processo de visão computacional e serão analisadas simultaneamente. Os dados finais devem ser enviados para o sistema ADC, para a determinação e controle da atitude. Além disso essas câmeras devem ser capazes de capturar imagens para que o sistema possa as armazenar e depois sejam enviadas por telemetria para a base de recepção;
- Fornecer posição de sol e Horizonte para controle da roda de reação;
- interface entre PS e ADC e COM, além de ter estabelecido protocolos de comunicação com os mesmos.
- Serviço WatchDog para outros Sistemas (Capacidade de os reiniciar em caso de falha);
- Receber arquivo de atuação e configurar o sistema baseado no protocolo recebido pelo sistema de telecomunicações
- A partir de artifícios de visão computacional, o sistema deve encontrar a posição do Sol e do Horizonte e dar qual a posição instantânea relativa do CubeSat. As análises das imagens das duas câmeras devem acontecer simultaneamente. Essa análise deve ser enviada para o ADC para o controle de atitude.
- Há necessidade de um software que adquire dados de tensão e corrente fornecidas pelos componentes do sistema PS e toma as decisões de estado de energia - modos dia-noite.

3.6.4 PS - Power Supply

O sistema de Power Supply do ZenSat tem como principal função o fornecimento de energia para todo e qualquer componente eletrônico e eletromecânico que precise daquela para o seu funcionamento, tais como processadores, sensores, o motor, entre outros.

Tido como um dos sistemas críticos de um CubeSat, já que uma falta nesse sistema pode vir a comprometer todo o bom funcionamento do projeto, foi investido um grande estudo em cima dos componentes e circuitos integrados que seriam utilizados e fundamentais para a segurança e confiabilidade de tal sistema.

Contamos com o fornecimento principal baseado em baterias, essas que podem ser recarregadas a partir de painéis solares localizados nas laterais do ZenSat. Além disso, todo um sistema de regulação dos níveis de energia também foi empregado, diferenciando a entrega para os componentes de processamento, dos componentes atuadores.

Por fim, utilizamos um módulo que dentre outras funções, realiza a proteção contra curto circuito e possível superaquecimento ou resfriamento das baterias durante a operação normal do ZenSat ou carregamento.

A partir das informações superiores, foram definidos os seguintes requisitos para o bom funcionamento do sistema de Power Supply:

- O sistema deve fornecer energia suficiente para que todos os subsistemas realizem suas operações.
- O sistema deve fornecer informações relevantes para que o sistema de controle possa alterar o modo de funcionamento em caso de baixa quantidade de energia.
- As células de bateria devem ser certificadas com datasheet e obtidas de fornecedor também certificados de modo a termos capacidade fiel à nominal. Além disso espera-se capacidade total maior que 6000mAh para uma tensão de 3,7V. Essas células devem estar fixadas numa placa de circuito impresso por cellholders e têm que ter suporte para carregamento.
- Os painéis solares do sistema devem fornecer potência maior que 1W no ponto máximo de operação.
- Step Ups de tensão de modo a fornecer as tensões necessárias para os vários componentes do satélite.
- Analisar se vai ser possível soldar na PCB e o quanto eficiente são essas pequenas células;
- Corrente máxima 2A;
- Modo low power - Day-Night Switch

3.6.5 STT - Structure

Responsible for the structural part of ZenSat, this system has as function the design of the skeleton and environmental simulations of the cubesat. It is necessary that the development of this follows the norms defined in the document *CubeSat Specification rev13*.

For the satellite project, it must take into account vibrations and temperature gradients that can be applied to it. To do this it is necessary to perform a vibrational and thermal analysis to verify if the structure designed meets the requirements imposed by the environment.

- Dimensions: 2U
- Material: Aluminum 7075-T6
- Fabrication method: CNC
- Center of mass in a sphere of 2cm from the geometric center
- Geometric tolerance: 0,1mm
- Sine vibration: 0-100Hz

4 Design

4.1 Diagrama de fluxo de dados

De modo a organizar o fluxo de dados dentro do ZenSat, foi confeccionado o esquema da figura 4. Ele separa os sistemas do cubesat por cores e as linhas de dados e alimentação da seguinte maneira:

- Azul: Power Supply
- Amarelo: Control
- Verde: Communication and Telemetry
- Vermelho: Attitude Determination and Control
- Linha única com flecha: Dados transportados e sua direção
- Linha dupla: Alimentação do sistema.

É possível analisar na interconexão entre os sistemas que a quantidade de dados trafegados é grande e muito dinâmica. Os subsistemas estão sempre trocando informação e, por isso, assim o projeto do sistema deve ser muito bem organizado e padronizado para que não hajam corrompimentos no sistema.

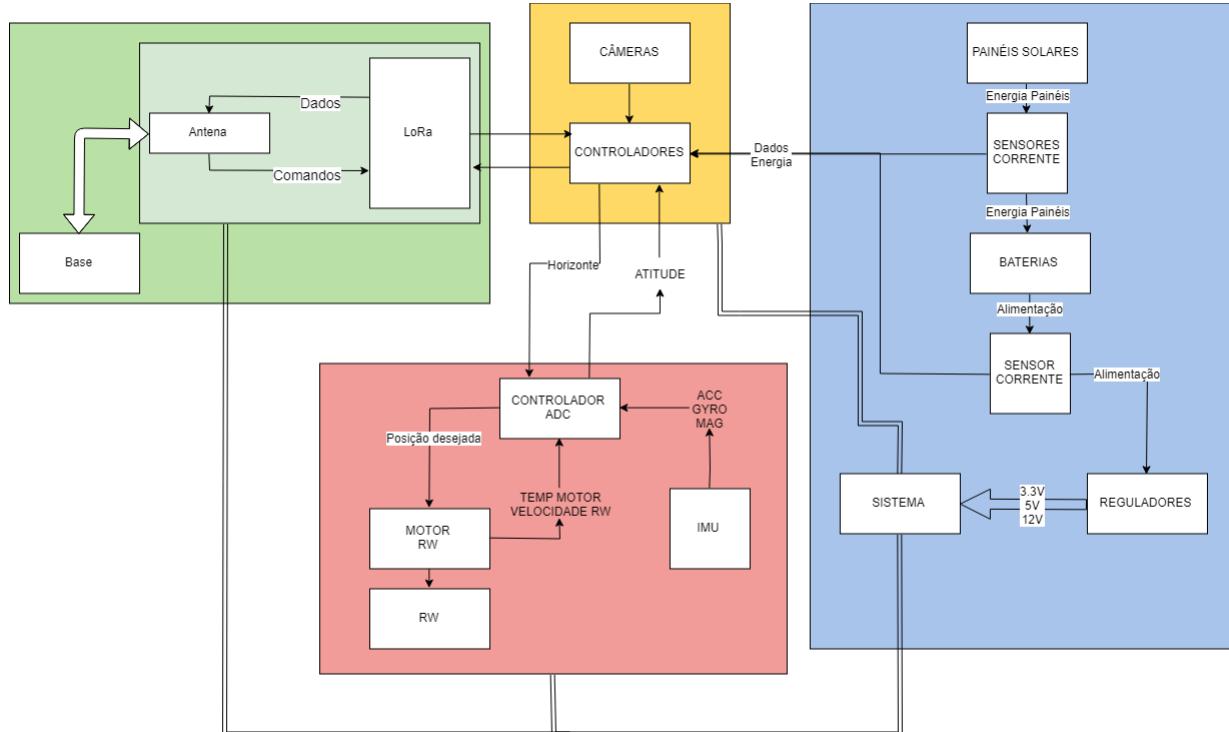


Figura 4: Diagrama de Fluxo de Dados

4.2 Product Breakdown Structure

De modo a ter uma visão geral do projeto, foi elaborado o diagrama apresentado na figura 5. Ele

é uma árvore hierárquica que mostra os componentes gerais do satélite e tem como função uma melhor visão geral do funcionamento do mesmo.

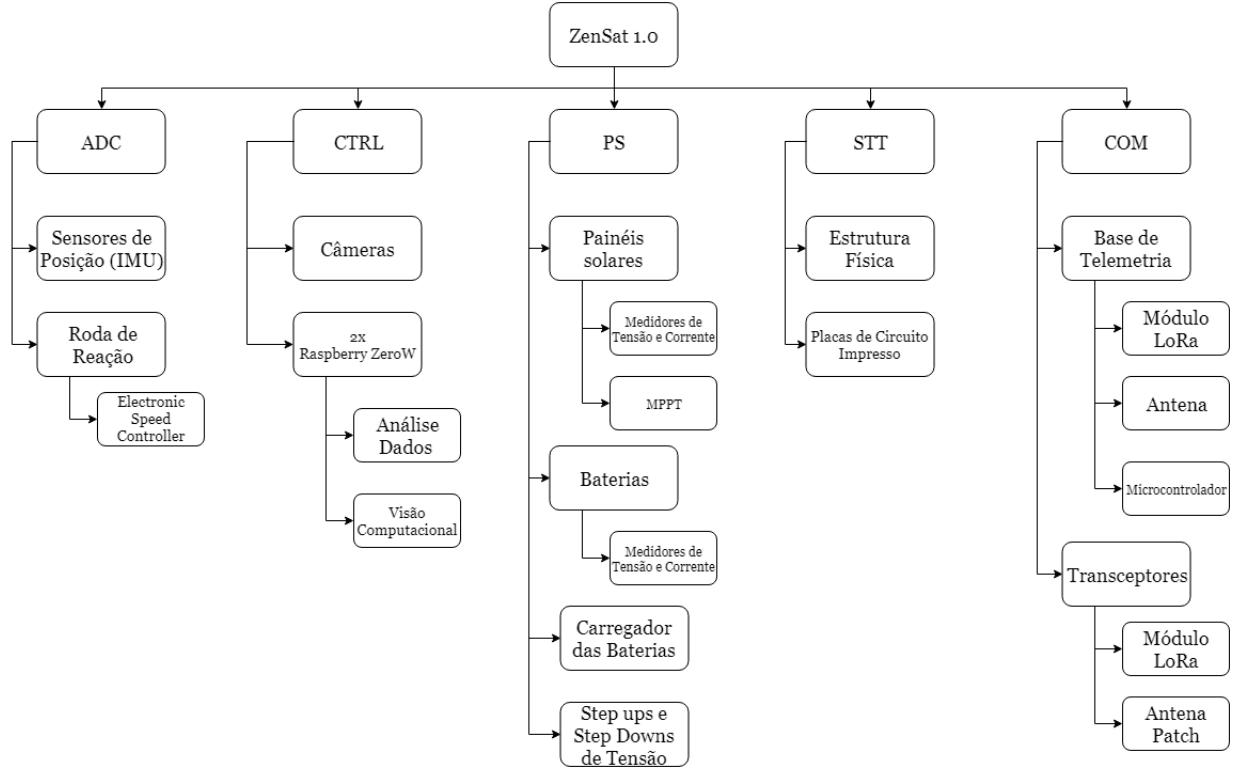


Figura 5: Product Breakdown Structure do ZenSat 1.0

Analisa-se que a árvore hierárquica nos proporciona um panorama interessante da conexão dos componentes dentro de cada subsistema. Com auxílio do diagrama de fluxo de dados, dado pela figura 4, pode-se

entender o funcionamento de cada componente, tanto quanto auxiliar-se para a montagem do software e do hardware do CubeSat.

5 Construção

5.1 Projeto Estrutural

O desenvolvimento completo da estrutura do ZenSat foi realizado, inicialmente, com base em modelos e aspectos levantados no início do projeto, buscando adequar sua construção aos parâmetros estabelecidos no desafio e os recursos disponíveis. Baseando-se principalmente na disponibilidade de espaço interno, definiu-se que o projeto partaria para uma estrutura final 2U, seguindo os padrões de medida conforme as normas fornecidas para orientação *CubeSat Specification rev13*.

Para viabilizar a aplicação em missão, além de suprir as necessidades estruturais envolvendo a manutenção total da geometria em atividade, proteção e suporte dos elementos internos, o ZenSat foi projetado com base em uma montagem robusta e leve envolvendo 4 trilhos dispostos nas arestas maiores da geometria, tendo estes função de união e suporte, e duas placas semelhantes aplicadas nas faces superior e inferior da estrutura com função de união e proteção. Além disso, há uma placa diferenciada para reforço estrutural e suporte na região central e uma placa para fixação dos componentes da roda de reação. Os métodos de união também cuidadosamente selecionados de forma a atribuir rigidez à estrutura e contribuir para a leveza e fácil montagem da estrutura.

5.1.1 Simulação

Foram realizadas simulações para diferentes desenhos com intuito de obter a frequência natural da estrutura e analisar qual seria mais adequada para as missões. Segundo *Falcon 9 User's Guide* [6], a frequência aplicada para sua payload varia entre 0 e 100 Hz, sendo assim as simulações foram realizadas visando encontrar uma estrutura capaz de suportar essas condições. A estrutura final escolhida está retratada na Figura 6 e os modos de vibração estão retratados na Tabela 1.

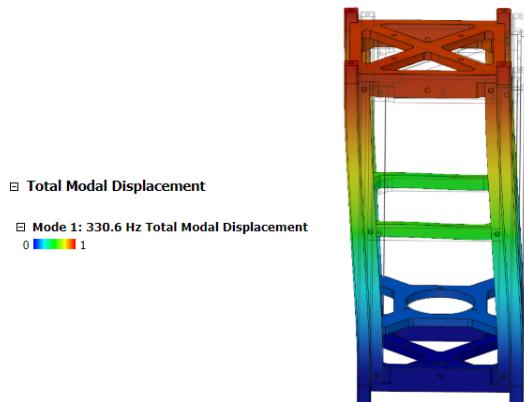


Figura 6: Resultado obtido para análise modal da estrutura.

Tabela 1: Frequências naturais de vibração do ZenSat 1.0

Modo	Frequência
1	330,6 Hz
2	337,3 Hz
3	514,9 Hz
4	548,7 Hz
5	594,9 Hz
6	666,4 Hz
7	1025 Hz
8	1129 Hz

5.1.2 Fabricação e Montagem

A estrutura do ZenSat totalmente modelada pela equipe se mostrou, através das simulações, ser capaz de suportar os desafios impostos e apresentou resultados satisfatórios para o início da etapa de fabricação.

Conforme foi projetada, a estrutura de liga de Alumínio 7075-T6, deveria ser totalmente usinada através dos processos de Fresamento e Furação. Em uma parceria com o grupo Zenith, a empresa situada em São Carlos - SP, AcquaNativa, se dispôs a viabilizar todo o processo de fabricação da estrutura, possibilitando assim o cumprimento da etapa de fabricação da estrutura do ZenSat. Diante do acordo, nas reuniões desta parceria, foram disponibilizados os desenhos técnicos mecânicos à empresa que, utilizando de parâmetros de usinagem estabelecidos junto à equipe de projetistas do grupo Zenith, iniciou a etapa de fabricação.



Figura 7: Render do projeto CAD do ZenSat 1.0

A montagem da estrutura desde o início do projeto foi uma das pautas mais importantes nas discussões, sempre houve a consciência de que a montagem da estrutura seria uma etapa delicada e que, se projetada com atenção, evitaria problemas de fragilidade estrutural, vibrações indesejadas e de alocação dos componentes internos. Começando pela seleção dos parafusos utilizados na montagem, juntamente do projeto dos métodos de união, até a utilização de eixos de aço 1045 que suportam conectores desenvolvidos e fabricados por impressão 3D para fixar placas e componentes eletrônicos internos; a estrutura ZenSat é projetada, fabricada e montada para garantir que todos os seus

componentes superem as condições de trabalho de uma missão.

Especificação de parafusos utilizados:

- M3x3mm *button head*
- M3x6mm
- M2x4mm para fixação das placas solares
- M3x30mm

A estrutura do ZenSat foi montada juntamente com a alocação dos sistemas eletrônicos. Iniciou-se a montagem pela região inferior, instalando os eixos na placa inferior e o suporte da roda de reação. Em seguida, os outros componentes eletrônicos e suportes foram alocados e por fim os trilhos laterais e as placas solares.

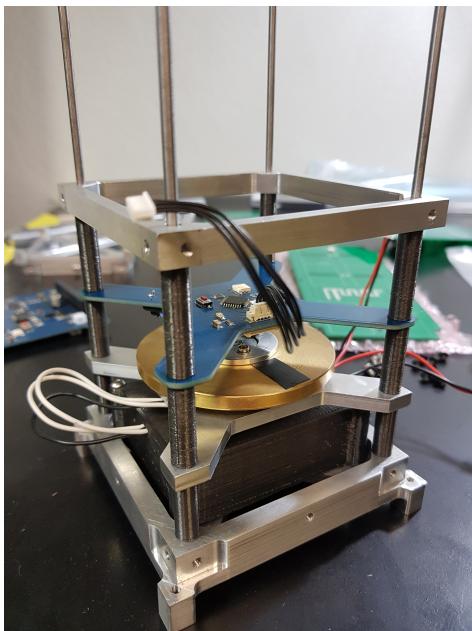


Figura 8: Início da montagem do ZenSat



Figura 9: Estrutura do Zensat Montada

5.2 Projeto de Hardware

Endo o primeiro projeto do grupo Zenith EESC a se basear nos padrões Cubesat, o ZenSat 1.0 representa uma oportunidade da equipe desenvolver sistemas análogos ao que serão embarcados eu uma futura missão real. Desse modo, os sistemas eletrônicos foram projetados tendo em mente a consolidação de padrões de conexão e barramento, geometria de placas e técnicas de manufatura, testando ideias que serão amaduradas nos futuros protótipos. Um dos sistemas aplicados ao projeto de hardware foi o uso de sistemas de “Stacking Headers” para criar um barramento de pinos comuns entre placas que compartilham sinais e energia. Das necessidades de pinos, escolheu-se um barramento de 40 pinos posicionados em duas fileiras de 20 pinos com o conector ESQ-120-39-G-D da Samtec mostrado na figura 10. Das suas características destaca-se os contatos banhados a ouro de baixa força de inserção e capacidade de suportar correntes superiores a 5A, resistência a temperaturas entre -40° e 125°C e espaçamento padrão de pinos de 2,54mm.



Figura 10: Conector ESQ-120-39-G-D da Samtec [3].

Com o intuito de se padronizar as conexões, a equipe criou o mapeamento de pinos ZHub que possibilitou que diferentes membros trabalhassem em diferentes designs de placas sem esperar a conclusão das placas com conexões comuns.

PS2	PS 1	ADC	TM 2	TM 1	CTR	MAP	Row 1	Row 2	MAP	CTR	TM 1	TM 2	ADC	PS1	PS2
						GND	1	2	GND						
						GND	3	4	GND						
						BAT +	5	6	BAT +						
						SOL +	7	8	SOL -						
						5V	9	10	5V						
						3v3	11	12	3v3						
						MISO 1	13	14	MOSI 1						
						SCK 1	15	16	CE01						
						MISO2	17	18	MOSI 2						
						SCK 2	19	20	CE02						
						CE11	21	22	CE12						
						SDA 1	23	24	SCL 1						
						SDA 2	25	26	SCL 2						
						TX1	27	28	RX1						
						TX2	29	30	RX2						
						SDA3	31	32	SCL3						
						DIG1	33	34	DIG2	ENABLE MPPT					
						DIG3	35	36	DIG4						
						DIG5	37	38	DIG6						
						DIG7	39	40	DIG8						
		RESET ADC												ENABLE MPPT	

Figura 11: Mapeamento do ZHub

Destaca-se também a padronização da geometria das placas internas para uma melhor organização física dos cabos e conectores, bem como do sistema de prendimento das placas na estrutura do Cubesat. Uma vez que as mesmas seriam feitas por processos industriais, uma geometria complexa com recortes não seria problema. As dimensões são apresentadas na figura 12:

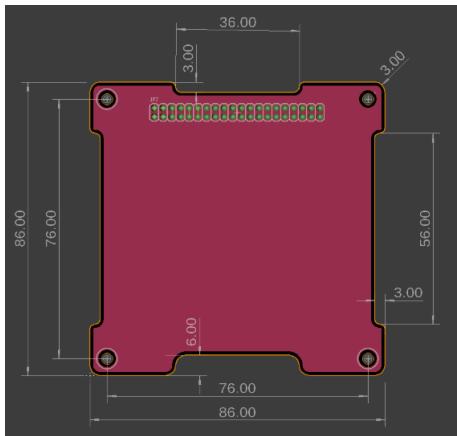


Figura 12: Padronização das dimensões das placas de circuito impresso do ZenSat

Todas os sistemas eletrônicos foram integrados em um total de sete placas de circuito impresso desenvolvidas pelo grupo com o software Eagle CAD 9.0 e fabricadas pela empresa chinesa PCBWay em compósito FR4, resistência ao calor TG 130-140, acabamento HASL e 1 Oz de cobre por pé quadrado.

5.2.1 Power Supply

O sistema de gerenciamento e fornecimento de energia do ZenSat exigiu o projeto de quatro placas, dada a grande quantidade de requisitos técnicos relativos à energia. Portanto, nota-se que esse foi o subsistema que demandou maior esforço durante o projeto de hardware do ZenSat. A seguir, todas as quatro placas projetadas serão detalhadas.



Figura 13: Render com destaque para o sistema PS do ZenSat

- Placa das baterias

A principal função desta placa é alojar as três baterias de modelo 18650, as quais consistem na principal fonte de energia do ZenSat. Este modelo foi escolhido devido à alta densidade energética e baixo custo, fatores que o tornam popular em nanossatélites desta natureza.

As três baterias foram dispostas em série na placa, resultando numa tensão total de 11,1V (3,7V de cada bateria). Foram escolhidos suportes metálicos do tipo "battery clips" para fixar as baterias e garantir o contato com as trilhas da placa.

Outro elemento importante desta placa é a chave de interrupção do fornecimento de energia para todo o satélite (também denominada "kill switch"). A presença dessa chave é de grande importância, pois permite que o ZenSat seja ligado e desligado de maneira simples e robusta.

Quatro sensores de tensão e corrente do modelo INA219 são dispostos na placa, realizando medições dos painéis das quatro faces do nanossatélite e das baterias. Esse sensor, da fabricante Texas Instruments, exige que o resistor "shunt" de medição seja inserido em série com o circuito no qual se deseja medir a corrente. É interessante ressaltar que o INA219 apresenta alta precisão, capacidade de realizar medições em diferentes frequências e resoluções, bem como capacidade de se comunicar via I2C, fatores esses que motivam sua utilização neste projeto.

Nesta placa estão ainda quatro diodos, cada um

conectado em série com o arranjo de células solares de uma das faces do ZenSat. Esses diodos (também chamados "blocking diodes") têm a função de impedir que os painéis solares consumam potência elétrica enquanto não estão gerando energia (situação de não-iluminação). Portanto, esta placa possui os conectores para receber como entradas as saídas dos painéis presentes nas faces laterais.

O esquemático elétrico da placa no software EAGLE pode ser visto na figura 14. A placa após fabricação e soldagem dos componentes pode ser vista na figura 15.

- Power Hub

Esta placa, vital para o funcionamento do ZenSat, possui diversas funções, entre elas: extraer a máxima potência dos painéis solares, carregar a bateria e adequar os níveis de tensão provenientes da bateria. Devido aos diversos blocos funcionais, esta placa é a que mais apresenta componentes em todo o ZenSat.

Para extrair a máxima potência possível proveniente dos painéis solares, utilizou-se um circuito integrado totalmente dedicado a isso: o SPV1040. Esse dispositivo utiliza um algoritmo de MPPT (Maximum Power Point Tracking, mais especificamente o algoritmo "perturb and observe"), o qual ajusta a carga aplicada aos painéis de modo a garantir que a máxima potência seja transferida à carga. O SPV1040 foi escolhido por ser um circuito integrado de alta eficiência.

A tensão de saída do bloco constituído pelo MPPT não é alta o suficiente para carregar a bateria, portanto fez-se necessário utilizar um conversor DC-DC. Para isso, utilizou-se o step-up LM2621, conversor boost de bom desempenho, para elevar a tensão até aproximadamente 12V.

Para utilizar os 12V presentes na saída do conversor para carregar as baterias, utilizou-se um circuito integrado específico para a carga de baterias íon-lítio: o CI MAX1757. Dessa forma, acoplando em cascata todos os blocos descritos acima, é possível carregar as baterias 18650 (principais fontes de energia do ZenSat) a partir da energia gerada a partir dos painéis solares existentes nas faces do nanossatélite.

Finalmente, fez-se necessário adequar os níveis de tensão para os níveis utilizados nas outras placas (portanto, 5V e 3,3V). Para isso, utilizou-se dois conversores buck MP2307. Ajustando-se os resistores dos divisores de tensão presentes no circuito do MP2307, foi possível garantir dois blocos fornecendo as tensões desejadas (5V e 3,3V), as quais são conectadas ao barramento comum de 40 pinos.

Nesta placa ainda estão presentes três sensores de tensão e corrente modelo INA219, os quais realizam a aquisição de dados da saída dos reguladores de tensão e das baterias.

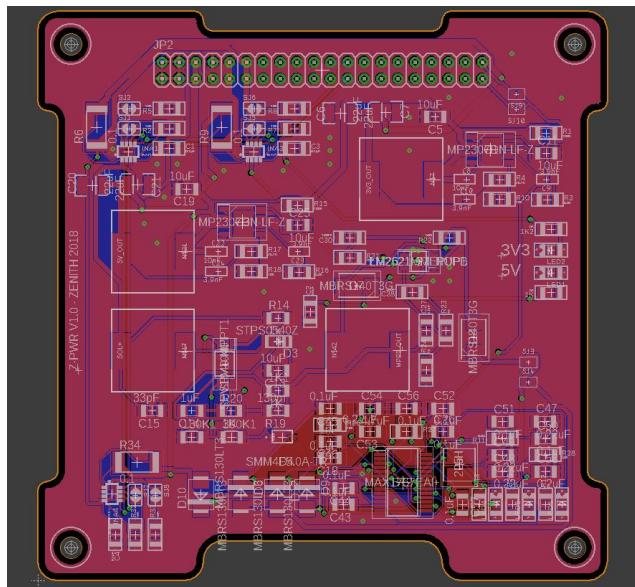


Figura 14: Placa de Circuito Impresso da placa Power Hub do ZenSat 1.0

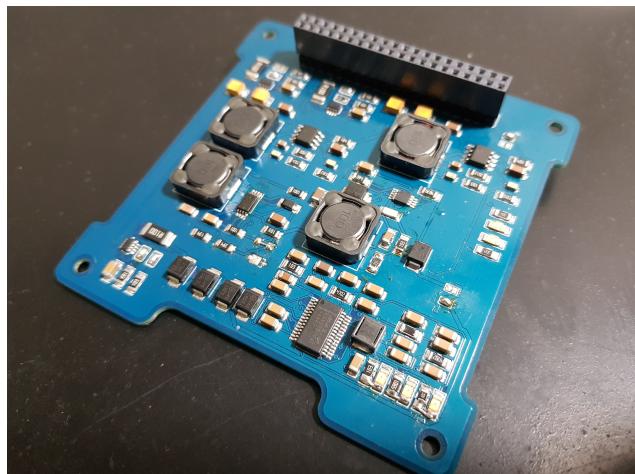


Figura 15: Placa do após soldagem

- Placa lateral antena

Foram projetadas duas placas laterais com os painéis solares e as antenas de comunicação. Além disso, estão presentes o microcontrolador Atmega 328P e rádio LoRA, necessários para o sistema de telecomunicação.



Figura 16: Antena Patch na placa dos painéis solares

- Placa lateral câmera



Figura 17: Antena com conector SMA

5.3 Projeto de Software

O software foi concebido para realização das missões propostas pela competição CubeDesign, sem muitas funções adicionais. No entanto, para implementação do código, o grupo Zenith se propôs a criar uma biblioteca de funções de uso geral para projetos no setor aeroespacial, a ZenithLib. Essa biblioteca conta com funções de manipulação de arquivos e pacotes de informação, pertinentes para implementação da comunicação no sistema; funções de recuperação de sistema, para o caso de falhas; funções de monitoramento de saúde do sistema; funções de determinação de altitude do satélite e algumas outras funções mais específicas. Como este é o primeiro protótipo de CubeSat criado pelo grupo, a biblioteca não está completa. A proposta é que ela evolua junto com os próximos projetos do grupo, tornando-se mais geral em todos os tipos de aplicações aeroespaciais. A biblioteca e os códigos criados podem ser encontrados no link:

www.github.com/zenitheesc/ZenSat_1.0

O desenvolvimento do software iniciou-se com a criação de um diagrama de blocos, o Diagrama de Fluxo de Software, disponível em:

Diagrama de Fluxo de Software

Esse diagrama representa os processos necessários para realização das missões propostas para a competição. Determinou-se que o CubeSat será uma máquina de estados, para facilitar o controle de suas ações, enumerando cada estado como uma missão a ser realizada. Além disso, determinou-se um processo de ini-

5.2.2 Commands and Telemetry

O projeto de comunicação entre CubeSat e Base posto utilizou um rádio de tecnologia LoRa, sendo interfaceado com uma Raspberry Pi 0 por meio de um microcontrolador, um Atmega 328P-AU. Utilizou-se o microcontrolador para não haver riscos de interrupção nos sistemas principais, gerenciados pela Raspberry. Tanto na Base quanto no CubeSat, o hardware criado é semelhante. Ocorre comunicação entre o microcontrolador e o rádio LoRa por protocolo SPI, sendo o microcontrolador configurado como Master, e comunicação entre a Raspberry e o microcontrolador por protocolo I2C, sendo a Raspberry configurada como Master.

Além da parte lógica, produziu-se uma antena na própria PCB do circuito do CubeSat, esta antena possui ganho de -0,1dB. Para a Base, utilizou-se uma antena com conector SMA de ganho 3dBi.

cialização, no qual são checados os subsistemas do nanosatélite e configurado variáveis que possibilitem seu funcionamento antes que a máquina de estados comece a operar. Com base no diagrama, iniciou-se a construção do software.

5.3.1 Funcionamento geral da biblioteca “zenith”

Como os sistemas aeroespaciais são bastante complexos, com vários subsistemas se comunicando um com o outro, iniciamos a criação da biblioteca ZenithLib com inspiração no funcionamento do ROS - Robot Operating System, que utiliza Nodes para a comunicação de vários subsistemas e dispositivos. Um Node consiste, de forma abstrata, num ponto de entrada e saída de informações, pelo qual se unem subsistemas distintos. Ou seja, quando um subsistema A deseja se comunicar com um subsistema B, o subsistema A envia a informação para o Node que os une e o subsistema B lê a informação no Node. Para implementar esse conceito de Node, utilizou-se uma estrutura de dois arquivos, para comunicação unidirecional, e quatro arquivos, para comunicação bidirecional. Em cada estrutura, há um arquivo contendo a informação enviada de um subsistema e um outro arquivo de indexação, que contém a posição no arquivo da última informação adicionada. No caso de comunicação bidirecional existem dois arquivos de informação e dois arquivos de indexação. Não utilizou-se o próprio ROS para desenvolvimento do CubeSat pois este ainda têm sua versão de operação RealTime em testes e não se fazia necessário de todas as funções desse sistema. Com base nesse sistema de comunicação todo o restante do software foi desenvolvido.

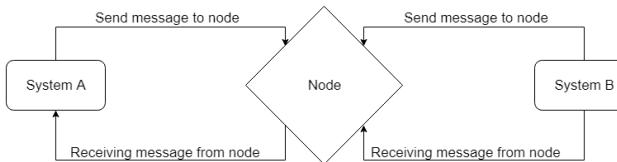


Figura 18: Exemplificação de um simples Node de comunicação

Os Nodes existentes no software são:

TM/TC : estes Nodes , TM e TC, têm funcionamento análogo, e são responsáveis pela correspondência do subsistema de comunicação entre CubeSat e Base. Estes, em especial, possuem, cada um, um arquivo a mais. Isso ocorreu pois decidiu-se fazer a indexação dos arquivos de telemetria e telecomando do satélite em relação à base com apenas dois bytes, para poupar espaço de informação nos pacotes de comunicação. Desse modo, utilizou-se um arquivo por byte. Um arquivo contendo o número primário do pacote de informação (valor entre 0 e 255) e outro, o valor de ciclo de pacotes de informação (também entre 0 e 255), que representa quantas vezes foram enviados 255 pacotes. Esses dois foram salvos em arquivos distintos para trazer mais segurança quanto aos dados, evitando

que esses sejam corrompidos. Os arquivos referentes a estes Nodes são definidos no arquivo “constants.h” da biblioteca, e são referenciados por:

TM

TM_FILE : Contém os pacotes de informação da telemetria;

TM_NUMBER : Contém a indexação primária da telemetria;

TM_CYCLE : Contém os valores de ciclo da telemetria;

TC_TC_FILE : Contém os pacotes de informação dos telecomandos;

TC_NUMBER : Contém a indexação primária dos telecomandos;

TC_CYCLE : Contém os valores de ciclo dos telecomandos;

PS : este Node é responsável pela comunicação do subsistema de alimentação do CubeSat. Ele apresenta comunicação unidirecional, não recebendo informações de outros subsistemas, apenas fornecendo-as. Os arquivos referentes a esse node são referenciados na biblioteca por:

PS_FILE : contém as informações enviadas pelo subsistema;

PS_NUMBER : contém a indexação;

ADC : este Node é responsável pela comunicação do subsistema de controle de atitude do CubeSat. Ele apresenta comunicação bidirecional, recebendo informações de configuração e retornando informações com resultados de operação. Os arquivos referentes a esse Node são referenciados na biblioteca por:

ADC_TX_FILE : contém as informações enviadas pelo sistema de controle de atitude;

ADC_TX_NUMBER : contém a indexação das informações enviadas pelo sistema de controle de atitude;

ADC_RX_FILE : contém as informações enviadas para o sistema de controle de atitude;

ADC_RX_NUMBER : contém a indexação das informações enviadas para o sistema de controle de atitude;

5.3.2 Comunicação entre Base e CubeSat - Padronizações

Como a comunicação entre Base e CubeSat é de vital importância para o funcionamento do sistema. Padroniza-se os pacotes de informação responsáveis pela troca de informação entre os dois sistemas para que seja mais eficiente a comunicação e dificulte a existência de erros e interferências que a comprometam. As padronizações principais se deram quanto ao número do pacote, número de ciclo do pacote e modo de operação. Em cada pacote enviado tanto pela Base quanto pelo CubeSat, as posições 0, 123 e 253 (considerando cada posição como um byte) são para guardar o número do pacote; as posições 2, 124 e 254 são para guardar o modo de operação e a posição 1 para guardar o ciclo do pacote. Para o número do pacote e o modo de operação são guardados três bytes para cada

em função da possibilidade de checagem do correto valor. Uma vez que é possível que haja interferência no pacote de informações, essas duas informações, número de pacote e modo de operação, devem ser checadas três vezes, e são assumidas como válidas apenas se dois desses valores forem iguais.

Além disso, cada missão requer informações específicas de telemetria e telecomando. A padronização de cada um desses pacotes se encontra no repositório do Github.

5.3.3 Inicialização

As funções referentes a inicialização cuidam da manutenção dos arquivos de dados do CubeSat. O cubesat ao iniciar, executa uma função de inicialização que faz a checagem de todos os arquivos de Nodes e de outras informações do CubeSat. Esses arquivos caso corrompidos são reinicializados de acordo com o último backup de arquivos existente. Após isso, inicia-se a comunicação entre as duas Raspberries presentes no CubeSat, indicando que o sistema está sendo inicializado e que a RaspSlave deve checar seus arquivos também. Após esses arquivos checados, inicia-se o código principal, que é responsável por ativar a máquina de estados na qual foi baseado o CubeSat.

5.3.4 Missão 1 - Telemetria de serviço

Em uma missão espacial, é fundamentalmente necessário que a base tenha informações sobre as condições nas quais se encontra o satélite, como nível de energia, status de carregamento, temperatura, velocidade angular do satélite e algumas outras medidas. Assim, para realizar a primeira missão proposta pela competição CubeDesign, criamos um pacote de telemetria com informações da saúde dos sistemas, os quais são enviados quando solicitados pela base ou a cada intervalo constante de tempo que é configurado pelo operador. Para isso, ativamos os Nodes responsáveis por cada subsistema, requisitando dados dos sensores. Estes dados são salvos nos arquivos de informação de cada Node e depois é montado o pacote com as leituras desses sensores com o posterior envio para a base.

5.3.5 Missão 2 - Checagem do subsistema de alimentação

Este estado do CubeSat é responsável por enviar informações sobre o sistema de alimentação à Base. Inicialmente, a base envia um telecomando, informando quantas amostras serão necessárias. O CubeSat recebe o telecomando e configura seu funcionamento de modo que possa fazer a checagem. A checagem consiste em ler através de sensores INA219 a tensão e a corrente de vários pontos do circuito: charger, bateria, regulador de tensão 5V, regulador de tensão 3V3, painel Solar A, painel Solar B, painel Solar C e painel Solar D. Essas leituras são enviadas pelo Node de checagem de alimentação para a RaspMaster e em seguida são montadas em um pacote para envio à base.

5.3.6 Missão 3 - Estabilização em um eixo

Neste estado, a base apenas envia que é necessário que o CubeSat entre em modo de estabilização. Recebendo a mensagem, a RaspMaster envia ao subsistema de ADC o comando de estabilização, via Node . O sistema realiza as rotinas de estabilização e retorna à RaspMaster uma mensagem de estabilização. Nessa mensagem de retorno, estão contidos ângulos de euler, azimute, elevação, velocidade da roda de reação e velocidade angular do CubeSat. A RaspMaster concatena essas informações e envia uma telemetria de informação de estabilização.

5.3.7 Missão 4 - Determinação de Sol e Horizonte

Para a determinação do Sol e do Horizonte foi implementada uma função em C interage com o código em Python da Visão Computacional para informar ao sistema o que foi encontrado na imagem capturada. Esta função envia um comando para a Raspberry para que seja capturada uma imagem pela PiCamera conectada. Isto feito, ela chama o código de Visão Computacional implementado e, após finalizado seu processamento, lê do Node de determinação de atitude, atualizado pelo código da visão, onde estão os valores das coordenadas do centro do segmento de reta pelo qual aproximamos o Horizonte e do círculo pelo qual aproximamos o Sol. Com isso, a função analisa os valores encontrados e retorna 0 caso nada tenha sido encontrado, 1 caso o Sol e o Horizonte tenham sido encontrados, 2 caso somente o Sol tenha sido encontrado e 3 caso somente o Horizonte tenha sido encontrado.

5.4 Missão 5 - Apontamento

Pra realização da missão de apontamento, o sistema obtém referências de posicionamento determinadas pela missão 4. Após isso, espera por comandos de posicionamento esperado, os quais são enviados pela Base e devem ser dados em relação à altitude determinada na missão 4. Com base nas informações recebidas, o Node de comunicação do ADC envia informações para esse subsistema, o qual, através de controle PID implementado, faz o apontamento em função dos parâmetros passados pelo operador. Quando o sistema termina sua movimentação, ele retorna ao Node informações sobre estado atual, como Azimute e Velocidade da roda de reação, além de uma foto da visão atual da câmera após o apontamento.

5.4.1 Missão 6 - Envio de imagem

Nesse estado o CubeSat realiza uma captura de imagem e a envia para a Base. Primeiramente, a imagem capturada é dividida sequencialmente em pacotes que é enviado para o ATMEGA, e então este realiza a transmissão com o uso do LoRa. Em sincronia com o CubeSat, a Base realiza a recepção dos pacotes com o uso do ATMEGA e do LoRa, assim como na transmissão, após armazenar todos os pacotes possíveis, as

informações úteis são extraídas para realizar a remontagem da imagem original. Os pacotes são estruturados seguindo o modelo padrão aqui descrito, sendo que o primeiro pacote a ser enviado contém informações sobre a quantia total a ser recebida pela Base.

5.4.2 Medição da temperatura na ciclagem térmica

No teste ambiental, o CubeSat terá de fazer medições de sua temperatura a cada período de tempo T determinado pelo operador, para isso, implementou-se uma função que lê o sensor de temperatura, monta o pacote mínimo de telemetria para envio da temperatura e o envia a base. Após isso, entra em modo de economia de energia, durante um intervalo T. Quando este período acaba, novas medições são feitas e enviadas à base.

5.5 Projeto de Controle e Determinação de Atitude

5.5.1 Hardware do ADC



Figura 19: Render da localização do Hardware ADC no ZenSat

Para o cumprimento das missões de controle de atitude do modelo de Cubesat em torno de um eixo (eixo Z), a equipe determinou o uso de um sistema de roda de reação única.

Seu princípio se baseia na conservação do momento angular dos elementos de modo que uma aplicação de um torque à uma massa gera um torque contrário na estrutura geral do satélite, possibilitando o controle do seu giro em torno do eixo de rotação da massa com torque aplicado.

O sistema depende então intrinsecamente de um sistema atuador sobre a massa (roda) e um sistema para a determinação do torque a ser aplicado. Esse segundo caracteriza o sistema de determinação de atitude e um algoritmo de controle. A ser descrito à frente, o ZenSat 1.0 possui um sistema de determinação de atitude com base em referenciais ópticos pelo reconhecimento do horizonte e sol por visão computacional, mas tal sistema é lento para um controle funcional. Desse

modo, escolheu-se o uso de IMUs de 9 eixos para a determinação da atitude com referenciais inerciais. Tais IMUs selecionadas foram duas BNO055 da Bosch Sensortec em um SiP (System in Package) que integra um processador ARM Cortex M0+ com algoritmos de fusão dos sensores, fornecendo dados com baixo ruído e dados de orientação absoluta como vetor de euler e quatérnios. Sua comunicação é via I2C.

Para o tratamento dos dados, comunicação e comando, escolheu-se o uso de um microcontrolador ARM Cortex M4 STM32L476RG da St Microelectronics de baixo consumo, 1MB de memória flash, 128 Kbyte de SRAM, podendo operar até 80MHz e programável por JTAG. Destaca sua compatibilidade com o ambiente Arduino, o que facilitou o desenvolvimento do código pois possibilita o uso extensivo de bibliotecas desenvolvidas para outros microcontroladores pela comunidade. Sua comunicação com o subsistema de controle é intermediada por um microcontrolador AVR ATMEGA 328p-au da Microchip de modo que esse é responsável por intermediar as interrupções, sendo escravo de ambos os sistemas, não sendo necessária a implementação de rotinas de interrupções em nenhum dos sistemas principais.

Devido a necessidade de um sistema de baixo consumo (menor que 3W), o dimensionamento do ADC se deu em torno do uso de motores Brushless de alta eficiência, tendo os motores de discos rígidos (HDD) como uma solução plausível. O grupo realizou um benchmark de aproximadamente 10 motores diferentes chegando no motor Nidec 2A2232IEC como a melhor solução por apresentar a melhor controlabilidade e menor consumo (menor que 1W com a roda dimensionada e em velocidade máxima constante).

Para o controle do motor, o grupo escolheu o driver DRV11873 da Texas Instruments por possibilitar o controle de motores Brushless com um simples sinal PWM, proporcional a velocidade desejada. O driver possui comutação de 15° e configuração sensorless, tensão de operação entre 5 e 16V e 1,5A de corrente máxima em operação contínua. Pode ser utilizado em motores com 3 ou 4 fios. A figura 20 representa simplificadamente a aplicação do driver para um motor de 4 fios (3 polos e um sinal comum). Uma vez que o motor utilizado pelo grupo não possui a conexão comum (apenas 3 fios), tal conexão foi emulada ligando-se três resistores aos polos (U, V e W) a um ponto comum (COM). O projeto original integrar trés drivers na placa de circuito impresso do ADC para um futuro controle de mais rodas para manipulação do Cubesat em novos eixos.

Infelizmente os drivers integrados demonstraram-se instáveis em tensões superiores a 6V e não podem ser utilizados para o controle da roda no Cubedesign. Os motivos para tal instabilidade são incertos, mas têm-se a suspeita de que o baixo espaçamento entre as trilhas da placa possibilitem interferência em tensões maiores. Uma nova versão da placa foi projetada e está em manufatura na china no momento da escrita desse documento, não devendo chegar até a competição. Para contornar esse problema, o grupo adaptou

um módulo comercial que integra o mesmo driver em um compartimento impresso em PetG por uma Prusa I3, conectado à placa do ADC por conectores previamente pensados para caso ocorresse problemas com os drivers na placa.

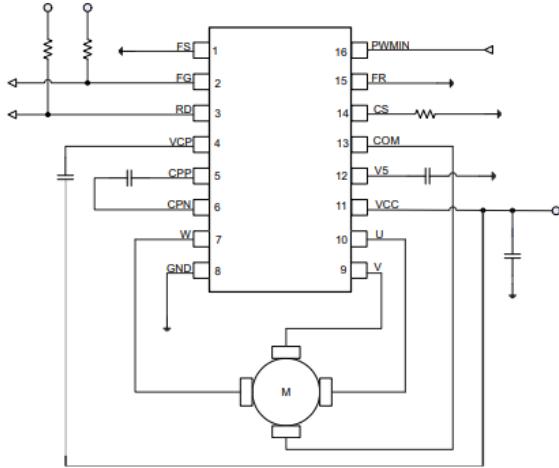


Figura 20: Esquemático simplificado da aplicação do driver DRV11873 da Texas Instruments



Figura 21: Adaptação de um módulo comercial com o driver DRV11873 para controle do motor apelidado carinhosamente de “Gambiarrinha Espacial”.

Os componentes descritos foram integrados em uma única placa de circuito impresso nos mesmos padrões que as outras placas do Cubesat, possibilitando o uso do mesmo barramento para fornecimento de energia e comunicação. Seu layout foi produzido com o software Eagle CAD 9.0 da Autodesk e sua manufatura foi realizada pela empresa chinesa PCBWay. A montagem dos componentes foi realizada pela parceira Cerebra Sistemas de Controle LTDA., em São Carlos-SP.

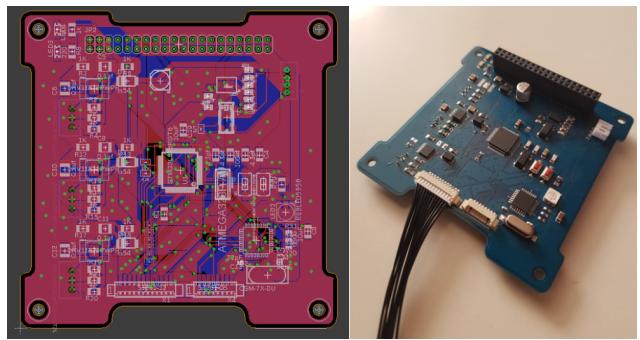


Figura 22: Placa de circuito impresso do ADC projetada no software Eagle CAD 9.0 e placa final montada.

Ainda, o grupo desenvolveu uma placa dedicada para a determinação da velocidade de rotação da roda de reação com base em um sensor óptico reflexivo infravermelho e um microcontrolador AVR ATMEGA 328p- au de modo a contar a passagem de fitas pretas fixadas na roda. Os valores de velocidade são utilizados pelo sistema principal do ADC, sendo a comunicação feita por I2C por um cabo Molex Picoblade.

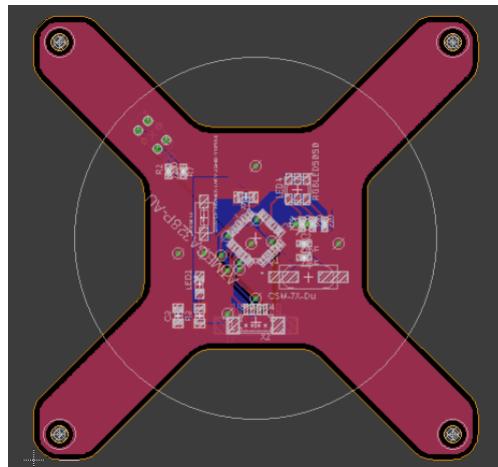


Figura 23: Placa de circuito impresso do Encoder projetada no software Eagle CAD 9.0.

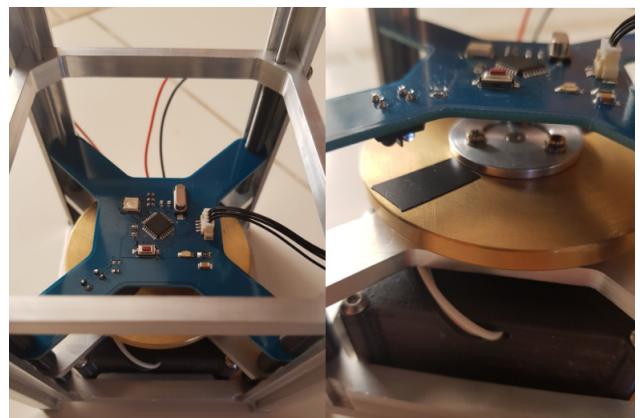


Figura 24: Montagem da placa do Encoder, da roda de reação com o motor e da caixa com o driver DRV11873.

O dimensionamento da roda de reação levou em

conta a velocidade máxima possível de ser obtida com o motor (aproximadamente 7000 rpm com pouca variação entre rodas diferentes) e a rotação máxima a qual deseja-se estabilizar o CubeSat, bem como uma estimativa do momento de inércia do mesmo.

Teve-se o valor de 100 RPM, fornecido pelos organizadores da competição, como a rotação máxima a qual o CubeSat seria suscetível à estabilizar. No momento do dimensionamento da roda o grupo não possuía o CAD finalizado, portanto as características geométricas e de massa para a determinação do momento de inércia do CubeSat não estavam disponíveis. Considerou-se então um paralelepípedo com as dimensões de 2U e 2Kg como o objeto a ser estabilizado. Seu momento de inércia, em torno do eixo Z, considerando sua maior dimensão alinhada a esse eixo, é dada então por:

$$I = (b^2 \times L^2) \times \frac{M}{12}$$

Onde:

$$\begin{aligned} \text{Base } b &= 0,1m; \\ \text{Largura } L &= 0,1m; \\ \text{Massa } M &= 2Kg; \end{aligned}$$

Resultando em

$$I = 3,3 \exp -3KgM^2$$

A equação da conservação do momento angular pode ser escrita como:

$$I_{sat} \times \omega_{sat} = I_{roda} \times \omega_{roda}$$

Com $W(sat)$ e $W(roda)$ as velocidades angulares do Cubesat e da roda de reação respectivamente. Desse modo:

$$I_{roda} = I_{sat} \times \frac{\omega_{sat}}{\omega_{roda}}$$

Para os valores das velocidades angulares convertidos das rotações máximas esperadas para a roda e para o Cubesat, tem-se:

$$I_{roda} = 4,7 \exp -5KgM^2$$

Como o valor mínimo para o momento de inércia da roda. O momento de inércia para um disco homogêneo com um furo no centro, posicionado no plano X-Y, em torno do eixo Z é dado por:

$$I = (R^2 - r^2) \times \frac{M}{2}$$

A escolha de um disco homogêneo como a geometria da roda de reação se deu com o fim de se facilitar a usinagem, embora não otimize o uso da massa para um maior momento de inércia. Com a massa M dada por

$$M = \pi \times (R^2 - r^2) \times T \times \rho$$

Sendo T a altura e ρ a densidade. Logo:

$$I = \pi \times (R^2 - r^2)^2 \times T \times \frac{\rho}{2}$$

Os valores para o raio do furo do centro “r” e para a altura “T” foram determinados de modo a possibilitar uma fixação simples no motor selecionado e correspondem a 12,5 e 5mm respectivamente. O material escolhido é o latão por ter alta densidade e alta permeabilidade magnética, de forma a mitigar interferências no funcionamento do motor e nas IMUs. Sua densidade é de aproximadamente $8,7g/cm^3$.

Resolvendo para o valor I_{roda} mínimo, têm-se o raio mínimo de 31 milímetros.

A fim de se evitar a operação na velocidade máxima e mitigar a possível rápida saturação do sistema, fabricou-se uma roda de 35mm de raio, com momento de inércia de $7,8 \exp -5$ e 146g.

5.5.2 Visão Computacional

A Determinação de Atitude utiliza-se de um programa de Visão Computacional e Processamento de Imagens para identificar a posição do Sol e da linha do horizonte, para que seja possível determinar a posição relativa do ZenSat em relação à estes referenciais. O programa utilizado recebe duas capturas periódicas, uma da câmera frontal e outra da câmera traseira, e processa cada imagem de forma a identificar se existem elementos nela e, se existir, estabelecer qual dos elementos é a linha do horizonte, qual é o Sol, e as posições relativas ao sistema de coordenadas do OpenCv para imagens destes objetos. Também é possível se determinar quantos graus o centro do Sol está defasado do centro da câmera. O sistemas de coordenadas que é utilizado pelo OpenCv possui o canto superior esquerdo como a origem, sendo que o eixo Y cresce deste canto até o inferior esquerdo e o eixo X cresce deste canto ao superior direito.

Para a detecção da linha do horizonte foi utilizado a Transformada de Hough [8], que pode ser utilizada para a detecção de retas ou de formas aproximadas a uma. Este algoritmo associa a reta a um ponto

$$(\rho, \theta)$$

no espaço, de forma que sua equação se torna

$$\rho = x \cos \theta + y \sin \theta$$

. Assim, foi possível estimar um segmento de reta delimitado pelo tamanho da imagem de entrada e determinar a posição do centro deste segmento.

Para determinar a localização do Sol, ele foi aproximado a um círculo e utilizado o algoritmo de Transformada de Hough para Círculos [5]. Este algoritmo retorna as coordenadas cartesianas do centro do círculo encontrado com base no sistema cartesiano da imagem e também o tamanho do raio r deste círculo. Com isso, foi possível determinar a posição do centro do sol em relação à imagem capturada.

Também é retornado pelo código a diferença angular entre o centro do Sol e o centro da imagem em

x, tendo vista que o ZenSat irá ser rotacionado apenas no eixo z. Assim sendo, estabelecemos o centro da imagem como o ângulo 0° , sendo que o máximo de desfase angular que o Sol pode ter para a direita é 80° e para a esquerda é -80° , visto que a abertura angular a do campo de visão da câmera é de 160° . Assim, comparamos a posição x do centro do Sol com o tamanho da imagem obtida e determinamos aproximadamente quantos graus o ZenSat deve girar para que o centro da câmera esteja alinhado com a posição x do centro do Sol

Caso nada seja encontrado na captura de alguma das câmeras, são retornados valores absurdos, como posição do centro do Sol ou do centro do segmento de reta descriptor da linha do horizonte com o valor $(-1, -1)$ e diferença angular do sol com valor -360° .

Foi feita uma simulação com fitas de LED e lâmpada para validar o funcionamento do código e foi obtida a imagem resultado da Figura 25

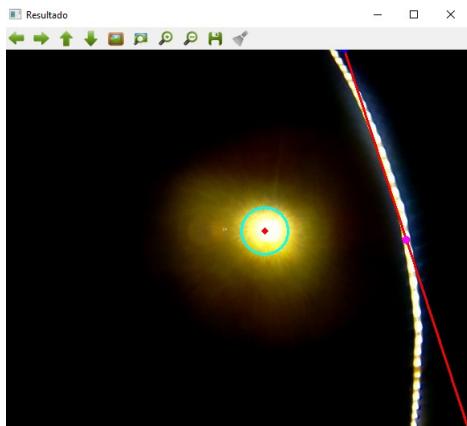


Figura 25: Imagem de Saída com a Detecção do Sol e da Linha do Horizonte

Os valores obtidos das coordenadas dos centros e do ângulos são gravados em um arquivo com a seguinte sintaxe:

```
[x do centro do segmento de Horizonte]
[y do centro do segmento de Horizonte]
[x do centro do Sol]
[y do centro do Sol]
[diferença de ângulo do Sol]
```

A Figura 26 é a imagem dos dados gravados em um arquivo para uma imagem semelhante à Figura 25. Este arquivo é utilizado pelos outros componentes do sistema embarcado.

1	000517.0
2	000240.0
3	00000382
4	00000260
5	025.9272
6	

Figura 26: Resultados gravado no arquivo

5.5.3 Determinação do Sol e do Horizonte

A determinação do Sol e do Horizonte é implementada na função *horizonDetermination.c*. Esta função realiza toda a parte de informar ao sistema o que foi detectado pelo código da visão computacional e também provê a imagem de entrada necessária para o processamento.

A função em questão inicialmente envia um comando para a Raspberry que a executa para que seja capturada uma foto com a PiCamera. O comando utilizado é *system("raspistill -w 640 -h 480 -o captura.jpg -q 10")*. Em seguida, com a foto capturada, é possível executar o código da visão com o comando *system("python VisaoCubeSatCompeticao.py")*, que, como explicado na seção anterior, processa a imagem de entrada e determina as coordenadas do centro de segmento de reta interpretado como o horizonte, do círculo interpretado como o Sol e da diferença angular do centro do Sol com o centro da câmera com base no eixo x.

Após executar o código, o programa lê o arquivo onde os dados foram salvos e analisa os resultados obtidos para determinar o que foi encontrado. Assim sendo, a função retorna 0 para o sistema caso nada tenha sido determinado, 1 caso tenha encontrado tanto o Sol quanto o horizonte, 2 caso tenha encontrado somente o Sol e 3 caso tenha encontrado somente o horizonte. Com estes valores, os demais componentes do sistema podem tomar as decisões adequadas à situação.

6 Conclusão

Em suma, o projeto do ZenSat 1.0 cumpriu seus objetivos, mesmo tendo alguns problemas durante seu desenvolvimento.

O sistema de energia do CubeSat cumpriu com o objetivo de fornecer energia para todo o sistema em funcionamento e os painéis solares carregaram as baterias com taxa suficiente para o funcionamento do satélite em modo de baixo gasto energético.

No caso da estrutura física, cuja fabricação foi realizada por uma empresa terceirizada e por este motivo os membros do grupo não participaram ativamente deste processo, algumas medidas ultrapassaram a tolerância definida para os desenhos em 0,1mm, o que pode impedir um encaixe efetivo do cubesat no P-POD.

Em relação a vibração, a estrutura desenvolvida pelo grupo mostrou-se capaz de suportar as frequências necessárias. Isso foi comprovado nas simulações de análise modal, pois o valor obtido de frequência natural da estrutura é superior ao valor das frequências atuantes no processo.

A padronização da comunicação cumpriu com as necessidades e permitiu uma troca de informações eficiente entre subsistemas e entre o Satélite e a Base.

A Determinação de Atitude a partir da Visão Computacional e Processamento de Imagens se mostrou um método muito eficiente para a determinação de posição do Sol e Horizonte, fornecendo as informações necessárias para o ADC e o CTRL realizem a

determinação de atitude.

Apesar do mal funcionamento do controlador eletrônico desenvolvido pelo grupo, a utilização do sistema comercial demonstrou-se funcional e permitiu a estabilização do CubeSat.

Por fim é possível notar que há inúmeras oportunidades de melhoria no projeto realizado, porém vale ressaltar que o desenvolvimento do ZenSat 1.0 possibilitou um grande aprendizado do grupo tanto em questões organizacionais quanto em quesitos técnicos relacionados ao desenvolvimento de satélites.

Referências

- [1] Instituto Nacional de Pesquisas Espaciais. inpe.br. = <http://www.inpe.br/>.
- [2] Instituto Nacional de Pesquisas Espaciais. Cubedesign - categoria cubesat, 2018. =<http://www.inpe.br/cubedesign/categorias/cubesat.php>.
- [3] RS Delivers. Detalhes do esq12039gd. <https://il.rsdelivers.com/product/samtec/esq12039gd/samtec-esq-series-254mm-pitch-40-way-2-row/2546132>.
- [4] Vicent L. Pisacane. *Fundamentals Of Space Systems*. Oxford University Press, 2005.
- [5] Harvey Rhody. Lecture 10: Hough circle transform. https://www.cis.rit.edu/class/simg782/lectures/lecture10/lec782_05_10.html.
- [6] SpaceX. Falcon 9 user's guide. http://www.spacex.com/sites/spacex/files/falcon9usersguide_rev2.0.pdf.
- [7] California Polytechnic State University. cubesat.org, 2018. =<http://www.cubesat.org/resources/>.
- [8] USP. Transformada de hough para detectar retas. <http://www.lps.usp.br/hae/apostila/hough.pdf/>.