IONSAT - Bus Communication Interface

Prepared by

Romaric SALLUSTRE M.Sc.(C)
Flight Software Developer Internship
November 2023 - February 2024

Mentor

Dr.Luca Bucciantini Ph.D.

Supervisor

Mr. Ricardo Colpari, M.Sc

Submitted
in Partial Fulfillment of the Requirements for the degree of
Master of Science AIRE – Digital Sciences

TABLE OF CONTENTS

1. Abstract

Nowadays, Cube Satellites are used to discover space with various payloads. The IonSat project represents a pioneering effort in CubeSat technology, aimed at designing, deploying, and maintaining a 6U CubeSat equipped with the Iodine NPT-30i miniaturized thruster It is designated as a stand-alone unit, thus this propulsion system demonstrates its integration capabilities during internal mission synchronization lasting more than a year. The main focus is on changes to the orbit irrespective of orbital inclination and launching in the Very Earth Low Orbit of 300-km. This project is in line with the emerging interest in minimal earth cycles, offering many benefits such as reduced communication delays, increased resolution and lower shipping costs. The internship function facilitates the communication between the subsystems and the onboard computer using the Intra communication protocols.

Keywords: CubeSat, IonSat, Thruster, Iodine NPT-30i, Very Low Earth orbit, Communication Protocol

1.1. General Context

CubeSats are a class of nanosatellites associated with a standard size and form factor. The standard size is a "one unit" or "1U" measuring 10x10x10 cm (Figure 1), with a maximum weight of 1.33 kg per "Unit", and multiple units can be combined in order to form satellites up to 24U. This format was developed in 1999 by California Polytechnic State University and Stanford University to provide a low-cost and easy-to-deploy platform for research, space exploration and education reasons. The CubeSat standard allows for different organizations an easy integration.

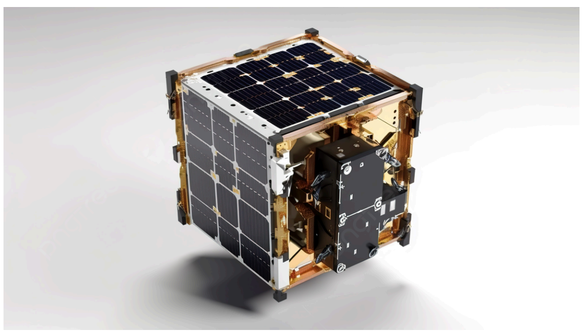


Figure 1: 1U CubeSat

Nowadays, due to their small size and low cost, CubeSats have become popular among universities, research institutions, and private companies. That fact enabled more opportunities for space research and exploration by allowing a wider range of organizations to participate in space missions.

In France, the CubeSat activities started with the development of "ROBUSTA 1-A" by the University of Montpellier with the technical support of the CNES(French space agency). Launched in February 2012 it only functioned for a few days. Nevertheless, this student project led to the creation of the JANUS program by the CNES. The JANUS program goal is to give technical and financial support for the realization of nanosatellites in universities and schools of higher education, thus training students in space engineering. By May 2023, 21 French CubeSats have been successfully launched and several others are in development.

1.1.1. Host Institution Presentation

The Centre Spatial de l'École Polytechnique (CSEP) is a student space center situated at the DRAHIX-nnovation Center of École Polytechnique. Founded in 2010, it oversees rocket and nano-satellite projects with support from X-Fab and LPP. CSEP facilitates collaboration among students, teachers, researchers, and industrial partners from various french organizations. The DRAHIX-nnovation Centre is fully equipped with renewable energy consumption.

2. Introduction

IonSat is a 6U nanosatellite student project of the Ecole Polytechnique, in partnership with the CNES (via the Nanolab Academy programme), ThrustMe and ONERA. This project is, above all, an educational project. IonSat will teach students the technical aspects of building and operating a CubeSat. Nevertheless, the objectives are not limited to education and train- ing. They also include an important technological innovation, which is the demonstration of a nanosatellite orbit control using electric propulsion in a Very Low Earth Orbit (VLEO). IonSat will also carry two payloads as secondary objectives, which won't compromise the primary goal.

The primary objective is the operation of maintaining the CubeSat in a Very Earth Low Orbit (VLEO) using an electric propulsion system. The electric thruster that will be integrated into IonSat is the NPT30-I2 iodine engine provided by the partner ThrustMe.

Figure 2: Electric thruster NPT30-I2; from ThrustMe

Regarding the flight plan and operations, the IonSat team is responsible for their development. In Figure 3 below, illustrating the altitude evolution since deployment. IonSat's mission has two main goals. The first is to maintain a station in a very low orbit (less than 300 km), which involves utilizing a propulsion system and a precise flight plan. The NPT30-I2 iodine ion engine from ThrustMe, serves as the propulsion system, and the IonSat team is in charge of developing the flight plan and overseeing operations.

The second objective focuses on studying the atmosphere at the visited altitudes, particularly characterizing density and composition. Density can be indirectly determined by measuring the drag force (orbit impact) or aerodynamic torque (attitude impact). The ONERA ResistPack system from ONERA helps measure the composition of atomic oxygen. Additionally, the team is conducting a study on the use of iodine on the platform, specifically examining the contamination of the solar panels.

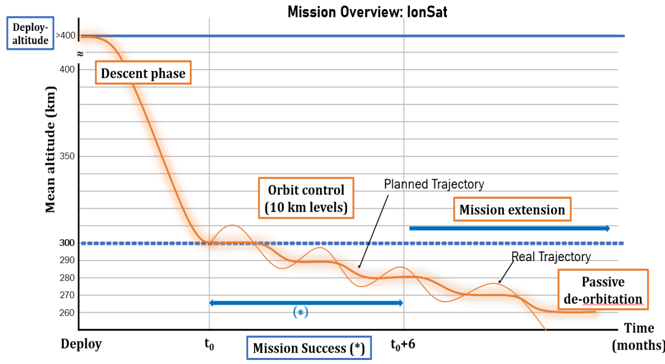


Figure 3: IonSat Mission Plan

2.1. System Overview

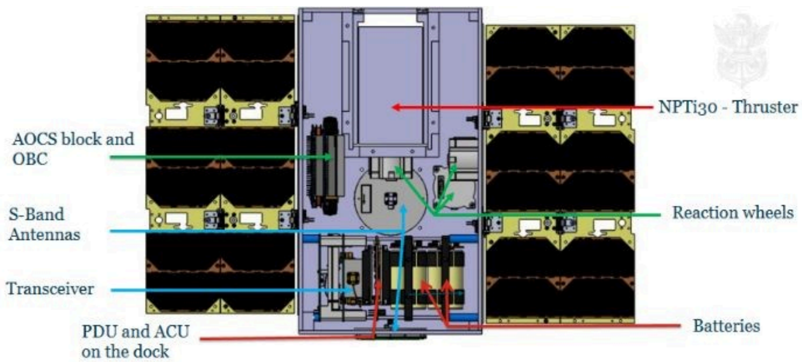


Figure 4: Positioning of subsystems

Although Ionsat is a 6U nanosatellite, It must still contain all the essential components of a larger satellite. Here's a basic breakdown of the key system components of satellite:

- **Electrical Power Subsystem:** This consists of solar panels to generate electricity, batteries to store the power, and power distribution and regulation subsystems. The power system is responsible for providing energy for all the satellite's functions.
- **Ground Segment:** This isn't on the satellite itself, but is an essential part of the system. The ground segment includes the ground stations that communicate with the satellite, as well as the systems for controlling the satellite and processing the data it sends back.
- **Communications Subsystem:** The satellite needs antennas and transceivers to communicate with ground stations. The communication system could include a receiver, a transmitter, or a transceiver which combines both.
- **Onboard Computer Subsystem:** This is the 'brain' of the satellite. It controls the various functions, manages systems, stores and processes data, and handles communications.
- **Attitude Determination and Control Subsystem (ADCS):** This subsystem is used to determine and control the satellite's orientation in space. It may use sensors such as star trackers, sun sensors, magnetometers, and gyroscopes to determine attitude, and reaction wheels, magnetorquers, and thrusters to control it.
- **Payload:** This is the reason why the satellite is in space – it carries the instruments that perform the satellite's primary mission. In IONSat, the ONERA ResistPack system is used to measure the composition of atomic oxygen.

- **Propulsion System:** Depending on the mission, some small satellites may include a propulsion system for maneuvers in space, orbit adjustments, or deorbiting at the end of the mission.
- **GNSS:** Global Navigation Satellite System(GNSS) is used primarily for Orbit Determination.
- **Thermal Control System:** Satellites are subject to extreme thermal conditions in space, so they need a thermal control system to keep electronic components within their operating temperature ranges. This can be achieved through passive methods like using coatings and radiators, or active methods like heaters and coolers.
- **Structural System:** This is the 'body' of the satellite that houses all the systems and payload. It must be strong enough to withstand the forces of launch, yet as light as possible to reduce launch costs.

All the above subsystems are connected to the onboard Computer which is called the Ninano Processor Board [9].

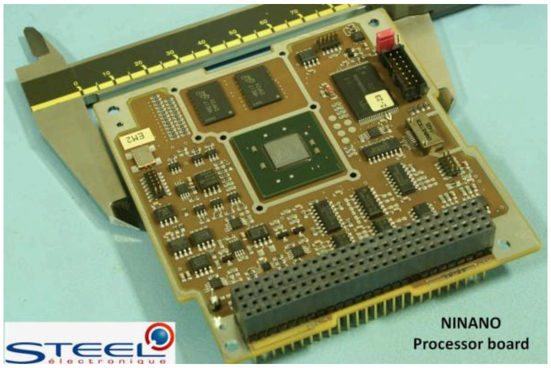


Figure 5: NINANO Processor

The Ninano board acts as the master computer which controls all the subsystems and takes the decision. As each year the project improves their interfaces and construction. IonSAT is still developing each subsystem. The next section will show the more technical part of the data transfer and communication between the subsystems.

3. Methods

3.1. Bibliographic Review

The Ionsat project has been the subject of a few papers by students at international conferences where each year's paper has an advancement in the IONSat which helped me to learn the structure of the research project. The stand alone propulsion system is a novel method in the cube satellite with a complex structure[1].Very Earth Low Orbit (VLEO). The atmospheric drag acting opposite to the satellite and the positioning of the cube satellite is briefed with the help of a simple drag model[2]. The usage of Iodine based electric propulsion thrusters has been described with less latency and low cost to float under the altitude of 400km (Low Earth Orbit)[3]. The design is floating in the Very Earth Low Orbit (VLEO) between the altitude of 250-400 km [4].

3.2. Overview of Activities Undertaken

The objective of this internship is to contribute to the development of the communication bus of IonSat and to explore the functionalities of the project. The IonSat platform envisions a distributed architecture where all nodes can communicate with each other via a common bus. Commands are received and transmitted via the common bus to the onboard main computer (OBC), which my focus is based on communication between Thruster, ADCS to the OBC via CAN and I2C bus. Additionally, Also joined as a part of the E4C (Energy for Climate) Challenge in a team of 4 for elaborating a Energy demand forecasting model.

3.3. Bus Communication Interface

The communication Protocol is the set of rules and regulations that allow two electronic devices to connect to exchange the data with one another. It is classified into two types:

1. Inter System Protocol - It is basically communication between two different devices The different categories of intersystem protocol Includes the UART(universal asynchronous transmitter and receiver), USART (universal synchronous and asynchronous transmitter and receiver) and USB(Universal Serial Bus)
2. Intra System Protocol - It is used to communicate the two devices within the circuit board. While using these intra system protocols, without going to intrasystem protocols we will expand the peripherals of the microcontroller. Using intra system protocols circuit complexity and power consumption, the cost is decreased and it is very secure to accessing the data. The different categories are I²C(Inter Integrated Circuit), CAN(Controller Area Network) and SPI(Serial Peripheral Interface). The internship is explored with CAN and I²C protocol to communicate between subsystems. Each protocol will be explained briefly below.

3.3.1. CAN Protocol

The CAN communications protocol is called Controller Area Network which is commonly used in automobiles where the CAN module can act as a multi master controller. It is a serial communication protocol. The working of CAN bus requires two wires CAN High (H+) and CAN low (H-). It is based on a message-oriented transmission protocol[5]. The CAN bus is used in Attitude Determination Control System(ADCS) where Arduino is used to communicate between the systems.

3.3.2. I²C Protocol

I2C stands for the inter-integrated circuit and it requires only two wires connecting all peripherals to the microcontroller. I2C requires two wires SDA (serial data line) and SCL (serial clock line) to carry information between devices. It is a master of a slave communication protocol. Each slave has a unique address. The master device sends the address of the target slave device and reads/writes the flag. The address matches any slave device that the device is ON, the remaining slave devices are disabled mode.

3.4. Components Used

The softwares used to contribute for the communication interface are Arduino IDE and STM32Cube IDE. The microcontroller development boards used are Arduino UNO, Arduino NANO and STM32F429I Discovery. I was exposed to the transceivers of MCP 2565 and SN65231D.

3.5 Attitude Determination Control Subsystem (ADCS)

The objective of the Attitude Determination and Control System (ADCS) is to ensure a stable orientation control of the satellite and a specific pointing precision. The below figure shows an overview of how the ADCS subsystem works with all of its main components and interfaces. The ADCS subsystem includes the following main components like sensors, actuators, microcontroller and interfaces.

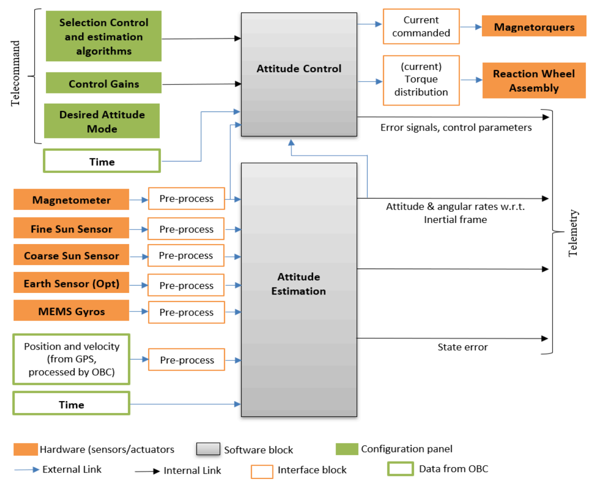


Figure 6: Structure of the ADCS Subsystem

In IONSat, The ADCS is purchased from CubeADCS which is the first project of Centre Spatial de Ècole Polytechnique.

3.6. Thruster Engineering Model

In IONSat, we use NPT30-I2 (shown in Figure 6)as a thruster. The NPT30-I2 is an electric propulsion system leveraging gridded ion thruster technology. The unit is entirely self-contained. Its PPU (Power Processing Unit), thruster, fuel tank, and thermal management system are fully integrated into a single system, conforming to CubeSat 1U or 1.5U geometry. The pro- pellant utilized is solid Iodine (I2), and the fuel tank remains unpressurized during launch, storage, and operation.

The table below shows the commands that the OBC can send to the NPT30-I2. A command consists of 8 bytes. One data type, one data number and six data parameters. If a data parameter byte is not used, it defaults to 0x55. To start the propulsion system, the "Arm Thruster" command must be followed, with a maximum delay of 10 seconds, by a "Start Firing Sequence" command. Any command other than "start firing" or "telemetry request" sent after arming the system will disarm the system. After 10 seconds of delay, the system will be disarmed.

Data types	Data numbers	Data parameters						Slave message reply	Description
0x00	0x01	0x55[1]	0x55	0x55	0x55	0x55	0x55	Telemetry Data Package	telemetry request
0x05	0x01	0x55	0x55	0x55	0x55	0x55	0x55	No reply	Arm thruster[2]
0x05	0x02	0x55	0x55	0x55	0x55	0x55	0x55	No reply	Start the firing sequence[3]
0x05	0x03	0x55	0x55	0x55	0x55	0x55	0x55	No reply	Stop the propulsion system
0x05	0x04	Mode[4]	0x55	0x55	0x55	0x55	0x55	No reply	Change the operation mode
0x05	0x05	0x55	0x55	0x55	0x55	0x55	0x55	No reply	Start ground test
0x05	0x06	0x55	0x55	0x55	0x55	0x55	0x55	No reply	Stop ground test
0x05	0x0A	0x55	0x55	0x55	0x55	0x55	0x55	No reply	Reset error code[5]

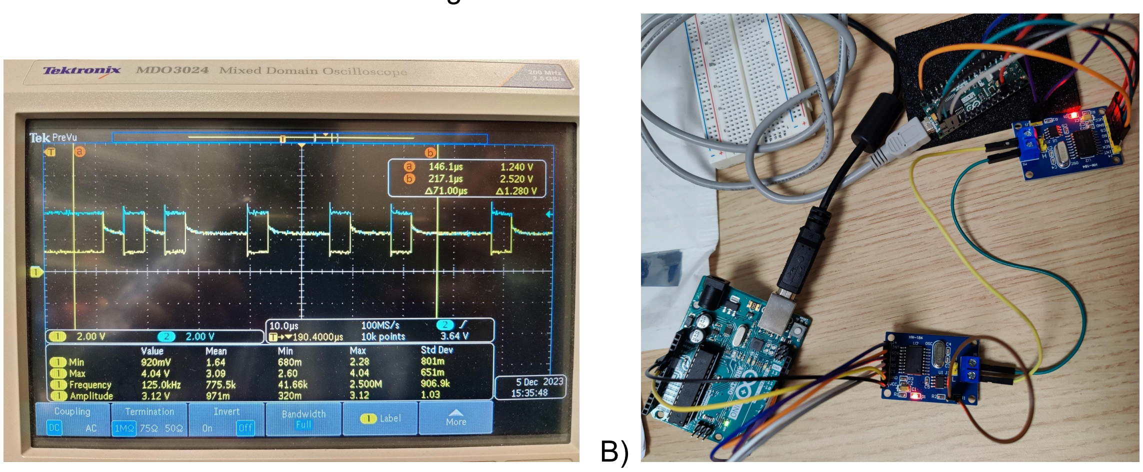
Table 1: Data Commands of the thruster

4. Executions & Results

4.1. CAN Communication between ADCS & OBC

The first initialisation was started by establishing a CAN communication between two Microcontroller Arduino boards with MCP2565 CAN module. The communication is monitored by the serial monitor of the Arduino IDE. The pinout configurations are the power, ground, CAN High, CAN Low and other pins. The signals produced between the boards are verified by the Oscilloscope. The result from the oscilloscope showcases the differential voltage with the dominant and recessive state for the voltage and time.

A)



B)

Figure 7: A) showcases the signal distribution between the CAN module. B) CAN Communication between two Arduino controllers

As the discussed CubeADCS insulated box was still under construction. The ADCS was not installed and further processes were put off. As we received the Thruster Engineering Model from ThrustME the building of communication protocol was started.

4.2. I2C Communication for Thruster & OBC

The I2C (Inter-Interconnected Circuit) is the communication protocol for communication between thruster engineering model and on board computer(OBC). The onboard computer used for testing the communication was STM32 microcontroller. As we need to configure various commands for the communication. STM 32 was highly reliable to use. The microcontroller board

STM32F429I Discovery allows users to develop rich applications easily with advanced graphic user interfaces. The software pinout configurations and code interfaces are carried out with STM32 CubeIDE. The debugging is worked out with an inbuilt debugger embedded with the board and connected via ST LINK V2.

The initiation of the I2C is also carried out between STM32F429I Discovery and Arduino Nano Controller with a sample data transmission. The output is monitored with the serial monitor of Arduino. The next process is to connect the thruster engineering model to the STM32F429I where a handshake is carried out to check whether the data is verified or not. Handshake is basically a multibyte communication between controllers with a call and response.

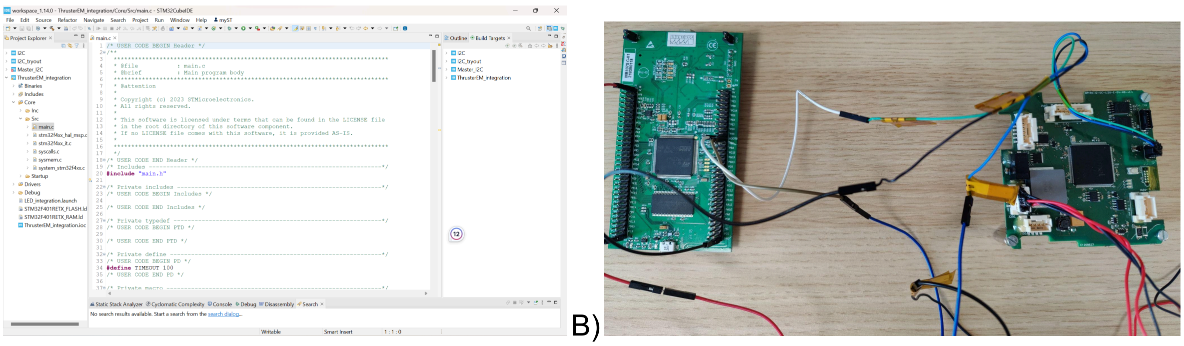
A)  B)

Figure 8: A) STM32 Cube IDE interface B) Thruster and STM32F429I model

Figure 8 A shows the STM32 CubeIDE interface where the code generation and pinout configuration is done in the same software and Figure B shows the connections between the thruster emulator as a slave and stm32F429I as the master. The pinout configuration basically consists of SDL (Serial Clock Line) and SDA (Serial Data Line) with the power connections. So, the further communication will be analyzed using a logic analyser incase of issues.The files of stm32 are uploaded in github and the links are given in the appendix.

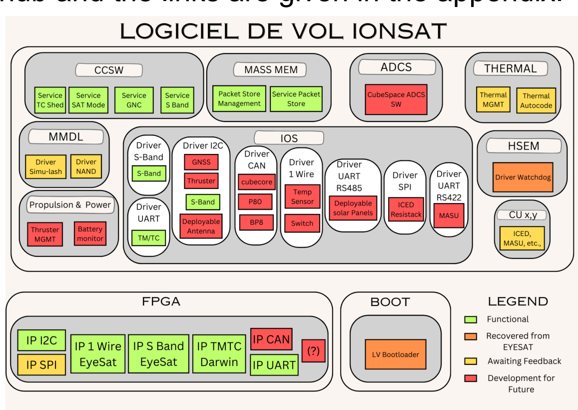


Figure 9: Software Architecture of IONSAT

The below picture shows the various subsystems and their interface connections. As IonSat is a long term and more efficient project where it integrates systems from different french organizations like CNES, ONERA and ThrustME. I contributed to strategizing the below chart for IONSAT.

5. Discussions

The above observations can show the preliminary results carried out for doing a communication I2C communication between the thruster and onboard computer. The ongoing step is to implement an algorithm for managing the data transfer between the OBC and the thruster emulator with some commands of initiation of propulsion which can be seen from the picture below. I am also exposed in meetings with the organizations for managing the structure of the IonSat. However, internship limitations include insufficient thruster emulator data hindering communication progress. Complications arose with Cubecore ADCS delivery, interrupting work initiated on ADCS-OBC bus communication. Additionally, the absence of a software engineer at CSEP posed challenges in resolving queries and addressing software-related tasks.

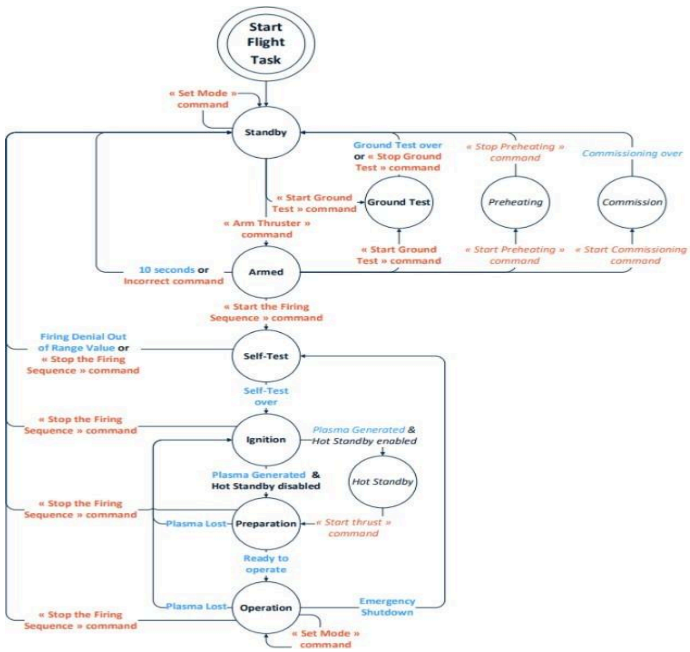


Figure 10: List Command Frames of the Thruster Engineering Model

My internship experience was a rich source of learning and personal growth, yielding several key takeaways:

1. Proficiency in the microcontrollers
2. The importance of various subsystems for communication protocol and overview
3. The ability to construct a communication between master and slave.
4. Competence in generating logic for connecting the subsystems
5. A comprehensive understanding of carrying out for building a cube satellites

6. Conclusions

The IonSat team played a significant role in guiding my learning process, and based on preliminary results, the thruster's interface has been successfully established through I2C communication. The primary focus now shifts towards implementing the algorithm for the Arm thruster command block of the thruster engineering model.

Looking ahead, I would like to propose some enhanced suggestions for the future development of the IonSat project:

1. Integration of the Command frame into the thruster engineering model.
2. Collaboration on Cube core ADCS integration with the onboard computer (OBC).

7. Acknowledgements

I wish to express my profound gratitude to key individuals who played pivotal roles in my transformative internship journey. First and foremost, my heartfelt thanks go to my mentor, Dr. Luca Bucciantini, an esteemed Astrophysicist whose unwavering support, invaluable guidance, and wealth of knowledge have been instrumental in shaping my understanding of various aspects of academia. Dr. Luca's grounded nature serves as a remarkable example of professionalism, and I feel truly privileged to have learned under his tutelage.

Equally deserving of my sincere appreciation is my supervisor, Mr. Ricardo Colpari. His unwavering dedication and commitment have provided invaluable direction for my learning journey. Mr. Ricardo's extensive knowledge and patience in reiterating concepts, along with generously sharing expertise, have been a cornerstone of my learning experience. His willingness to assist me in resolving issues by working through them together has been akin to a compass, guiding me on the right track whenever needed. I am profoundly grateful for the wisdom and perspective he has graciously imparted. The knowledge, skills, and experiences acquired during this internship have truly transformed my perspective on my interest in Space. With the invaluable support and experiences garnered during this internship, I eagerly anticipate continuing my academic journey with newfound confidence and boundless enthusiasm.

8. References

01. *Hui, J., Olichon, K., Autric, A., Taleb, H., Deroo, V., Sueiro, J., Launois, A., Herma, Z., & Maloum, M. A. (2022). DETAILED DESIGN OF IONSAT: a STATION-KEEPING MISSION AT ALTITUDES BELOW 300KM. www.academia.edu. https://www.academia.edu/86960427/DETAILED_DESIGN_OF_IONSAT_A_STATION_KEEPING_MISSION_AT_ALTITUDES_BELOW_300KM*
02. *Paper information (60923) — IAF*. (n.d.). https://iafastro.directory/iac/paper/id/60923/summary/
03. Marmuse, F. (2019, October 21). *Fitting a high total impulse electric propulsion system in a student CubeSat to compensate the atmospheric drag in low-earth orbit*. https://polytechnique.hal.science/hal-02341692/
04. Pellouin, C. (2018, October 1). *IonSat: challenging the atmospheric drag with a 6U nanosatellite*. https://hal.science/hal-02341691
05. 2018.B.2.6. IONSAT: Challenging the atmospheric drag with a 6U nanosatellite. (2021, March 1). Retrieved from https://icubesat.org/papers/2018-2/2018-b-2-6-ionsat-challenging-the-atmospheric-drag-with-a-6u-nanosatellite/
06. *NINANO - On Board Computer for Nanosatellites | satsearch*. (n.d.). https://satsearch.co/products/steel-electronique-ninano-on-board-computer-for-nanosatellites

07. Agarwal, T. (2021, February 15). *Communication Protocols : Basics and Types with Functionality*. ElProCus - Electronic Projects for Engineering Students. https://www.elprocus.com/communication-protocols/

9. Appendix

9.1. Cubesat Space Protocol:

When ADCS communication faces interruptions, an alternative protocol, Cubesat Space Protocol (CSP), becomes essential for ADCS-OBC communication. In Ionsat, our software focuses solely on interfacing with ADCS and EPS via CSP. The emphasis lies in comprehending the transmitted protocol format and executing parsing tasks.

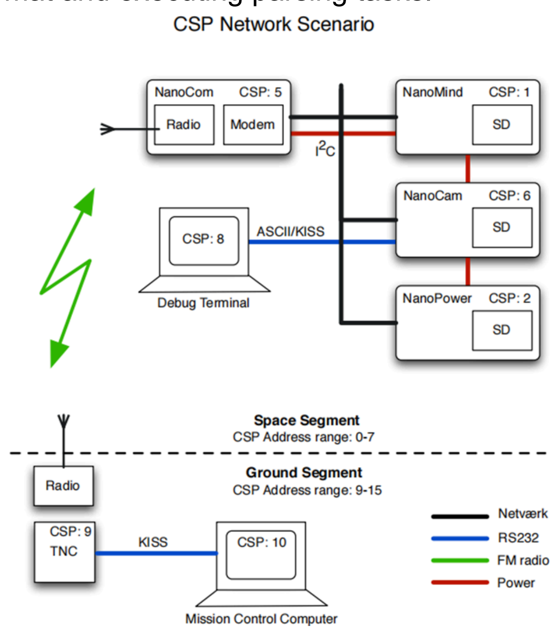


Figure 11: Cubesat Space Protocol

9.2 Additional Work

The E4C challenge is to minimize the emissions of a building located on the IP Paris campus (DrahiX: the startup incubator). As the project started on Jan 15, I don't have a result to produce. Basically, The challenge consists of three tasks:

- Elaborate an energy demand forecasting model based on data provided by us.
- Elaborate an energy consumption management strategy, based on interactions with occupants, the available data, and the model, to minimize CO2 emissions.
- Discuss the acceptability of this strategy and explore ways to gain acceptance.

9.3 Documentation

All the code files, pinout configurations and other implementations are documented in [GitHub].