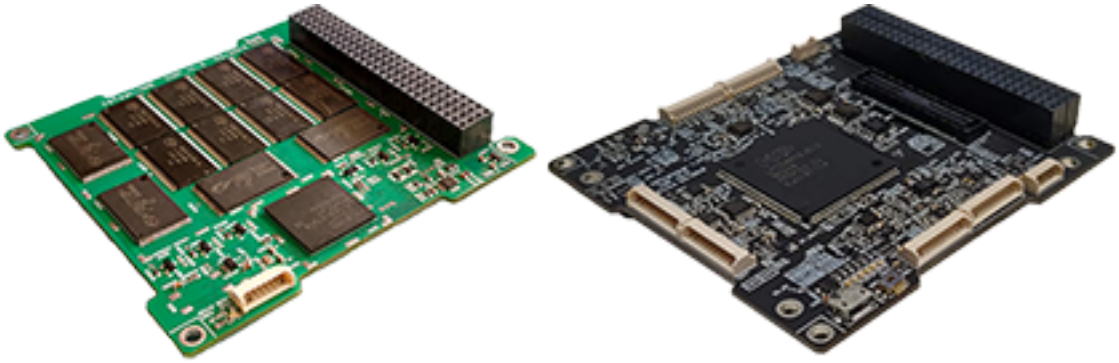
TÉCNICO LISBOA



Evolution of the Communications Processor Board for the ISTnanosat

João Carlos Fonseca Pinto

Thesis to obtain the Master of Science Degree in

Electronics Engineering

Supervisors: Prof. Rui Manuel Rodrigues Rocha
Prof. Moisés Simões Piedade

Examination Committee

Chairperson: Prof. Pedro Miguel Pinto Ramos
Supervisor: Prof. Rui Manuel Rodrigues Rocha
Member of the Committee: Prof. António Joaquim dos Santos Romão Serralheiro

November 2019

Declaração

Declaro que o presente documento é um trabalho original da minha autoria e que cumpre todos os requisitos do Código de Conduta e Boas Práticas da Universidade de Lisboa.

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Agradecimentos

Em primeiro lugar, uma palavra de agradecimento aos meus orientadores, professor Rui Rocha e professor Moisés Piedade, por todo o suporte, dedicação, ensinamentos e conhecimentos transmitidos, bem como pelas críticas e revisões necessárias ao trabalho realizado nesta dissertação de mestrado.

À equipa ISTnanosat, um obrigado pelos conselhos, apoio e companheirismo durante esta etapa, cuja progressão se deve muito a vocês.

Aos meus pais, Manuela e Fernando Pinto, e aos meus irmãos, Pedro e António Pinto, por me ajudarem a crescer cientificamente e suportarem a minha vida, bem como pela motivação e sacrifício dedicados a mim. Sem vocês nada disto seria possível.

Ao Leite, Tomázio, Tiago, Luís, António e Guilherme, pelo apoio e camaradagem durante todo o meu percurso universitário. Convosco tudo ficou bem mais fácil.

À minha namorada, Joana Chumbo, pela paciência e compaixão que dedicaste a mim durante o nosso tempo. Obrigado pelo respeito, carinho e por acreditares comigo.

Abstract

Since the launch of the first man-made object placed into space orbit, the use of artificial satellites has increased, specially in the last years. These satellites have been used in several applications with different sizes and orbits. The ISTsat-1 is the first nanosatellite developed by the ISTnanosat team, having a 10 cm cubic shape that respects the cubesat standard. This satellite is intended to orbit in the Low Earth Orbit in order to test the reliability of an aircraft tracking system, being its scientific payload, plus collecting information about the space environment.

This master thesis focuses on the evolution of the Communications Processor Board subsystem for the ISTnanosat team, which is divided into two distinct versions. While the first version will be integrated in the aforementioned ISTsat-1, the second version is designed for the integration in future ISTnanosat satellites, being an evolution based on two subsystems flying in the first satellite. As the name implies, these subsystems are designed to support the satellite communications, although they may also be considered for the satellite command and data handling and attitude determination and control. Therefore, possible architectures for them are discussed in this master thesis report, describing the final hardware and software architectures. These architectures take into account the harsh space environment that these subsystems will be subjected to, being some radiation mitigation and subsystem level redundancy techniques addressed as well. Finally, some validation tests are implemented for the acceptance of the developed subsystems, confronting the expected results and the obtained ones.

Keywords

ISTsat-1, space environment, cubesat, communications, command and data handling, attitude determination and control, high power processor, memory unit, data storage, sensors, actuators, radiation mitigation.

Resumo

Desde o lançamento do primeiro objecto criado pelo Homem e colocado em órbita espacial, o uso de satélites artificiais tem aumentado, especialmente nos últimos anos. Estes satélites são utilizados em várias aplicações com diferentes tamanhos e órbitas. O ISTsat-1 é o primeiro nanosatélite desenvolvido pela equipa ISTnanosat, tendo uma forma cúbica de 10 cm que respeita o *standard* definido para *cubesats*. Pretende-se que este satélite orbite numa órbita terrestre baixa, tendo como missão científica o teste da fiabilidade de um sistema de rastreamento de aeronaves, além de recolher diversas informações sobre o ambiente espacial.

Esta dissertação foca-se na evolução da Placa do Processador de Comunicações, subsistema pertencente à equipa ISTnanosat que se encontra dividido em duas versões. Enquanto a primeira versão será integrada no satélite ISTsat-1, a segunda versão é desenvolvida para a integração em futuros satélites construídos pela equipa ISTnanosat, sendo uma evolução baseada em dois subsistemas pertencentes ao primeiro satélite. Como o nome indica, estes subsistemas são desenhados para suportar as comunicações do satélite, apesar de poderem ser também considerados para o tratamento de dados e comandos do satélite e para a determinação e controlo da sua atitude. Portanto, neste relatório de dissertação são discutidas possíveis arquitecturas para estes subsistemas, descrevendo as arquitecturas finais de *hardware* e *software*. Estas têm em consideração o ambiente espacial severo no qual estes subsistemas vão operar, sendo por isso abordadas algumas técnicas de mitigação de radiação e de redundância a nível do subsistema. Para concluir, alguns testes de validação são implementados para a aprovação dos subsistemas desenvolvidos, confrontando os resultados esperados com os resultados obtidos.

Palavras Chave

ISTsat-1, ambiente espacial, *cubesat*, comunicações, tratamento de dados e comandos, determinação e controlo de atitude, unidade de alto processamento, unidade de memória, armazenamento de dados, sensores, actuadores, mitigação de radiação.

Contents

List of Figures

List of Tables

Acronyms

ADCS Attitude Determination and Control System

ADC Analog-to-Digital Converter

ADM Antenna Deployment Mechanism

ADS-B Automatic Dependent Surveillance-Broadcast

AFSK Audio Frequency-Shift Keying

AIS Automatic Identification System

API Application Programming Interface

BGA Ball Grid Array

BLS Byte Lane Select

CAN Controller Area Network

C&DH Command and Data Handling

CMSIS Cortex Microcontroller Software Interface Standard

COM Communications Processor Board

COTS commercial off-the-shelf

CS Chip Select

DAC Digital-to-Analog Converter

DoF Degrees of Freedom

DRAM Dynamic Random Access Memory

DSP Digital Signal Processor

EMC Electromagnetic Compatibility

ExtMC External Memory Controller

eMMC Embedded Multimedia Card

EPS Electrical Power System

ESA European Space Agency

FeRAM Ferro-electric Random Access Memory

FPGA Field Programmable Gate Array

FPU Floating-Point Unit

FYS Fly Your Satellite!

GCR Galactic Cosmic Rays

GNSS Global Navigation Satellite System

GPIO General Purpose Input/Output

GPS Global Positioning System

GSMK Gaussian Minimum Shift Keying

GS Ground Segment

HAL Hardware Abstraction Layer

I^2C Inter-Integrated Circuit

IC integrated circuit

IDE Integrated Development Environment

IMU Inertial Measurement Unit

IO I/Oinput/output

IPC Inter-Process Communication

ISIS Innovative Solutions in Space

ISR Interrupt Service Routine

ISS International Space Station

IST Instituto Superior Técnico

JTAG Joint Test Action Group

LEO Low Earth Orbit

LQFP Low Profile Quad Flat Pack

MBM motherboard module

MCU microcontroller

MEMS microelectromechanical systems

MEO Medium Earth Orbit

MISO Master In Slave Out

MOSFET metal–oxide–semiconductor field-effect transistor

MOSI Master Out Slave In

MRAM Magnetoresistive Random Access Memory

OBC On-Board Computer

OE Output Enable

PCB Printed Circuit Board

PPM Pulse Position Modulation

PWM Pulse Width Modulation

QPSK Quadrature Phase-Shift Keying

RAM Random Access Memory

RTC Real-Time Clock

RTOS Real-Time Operating System

RTS Request to Send

SCL Serial Clock Line

SDA Serial Data Line

SDRAM Synchronous Dynamic Random Access Memory

SDR Software Defined Radio

SD Secure Digital

SEE Single Event Effect

SEL Single Event Latchup

SLC single-level cell

SoC system-on-a-chip

SOI Silicon on Insulator

SPI Serial Peripheral Interface

SPIFI Serial Peripheral Interface Flash Interface

SRAM Static Random Access Memory

SWD Serial Wire Debug

TID Total Ionising Dose

TRL Technology Readiness Level

TT&C Tracking, Telemetry and Control

UART Universal Asynchronous Receiver Transmitter

UHF Ultra High Frequency

USART Universal Synchronous Asynchronous Receiver Transmitter

USB Universal Serial Bus

VHF Very High Frequency

V/U Very High and Ultra High Frequency

WDT Watchdog Timer

WE Write Enable

WMM World Magnetic Model

Chapter 1

Introduction

Since the launch of the first artificial Earth satellite into an elliptical Low Earth Orbit, the Sputnik 1 from the Soviet Union in October 1957, the use of artificial satellites has increased, specially in the last years. These satellites have been used in several applications, mainly for communications, scientific, educational and military purposes. Due to the harsh space environment, the manufacturing and the necessary quality assurance and expertise to ensure a high-reliability system that does not fail in orbit is highly expensive. However, due to the technology evolution and to the increase of satellite standardisation, the cost of producing a satellite has decreased, enabling universities to increasingly launch small satellites into space.

One of these standardisations is the Cubesat[1]. This small and low cost satellite is a 10 cm cube unit with a maximum mass of 1.33 kg, which can be stacked to form larger satellites. The use of these satellites was initially intended for Low Earth Orbits, performing scientific researches and exploring new space technologies when the use of larger satellites was not plausible due to their cost. Therefore, a cubesat can be launched into space to test the feasibility of new technologies before deploying bigger satellites. Furthermore, this miniaturised satellite also allows educational studies to be done without a great expense. Nowadays, cubesats are starting to be also considered for deep space missions.

To aid some of these educational space activities, the Fly Your Satellite! (FYS) programme, which is designed and managed by the European Space Agency (ESA) Education Office in collaboration with European universities, has the objective of complementing the academic education of university students, inspiring and preparing them for a more effective introduction to a future work in the space sector. Between 2008 and 2019, this Education Office has enabled more than 10 cubesats to be launched into space, three of them in the first FYS edition: the e-st@r1 from Italy, the AAUSAT-4 from Denmark and the OUTTI-1 from Belgium [1].

[1] The Cubesat standard design was firstly proposed in 1999 by professors Jordi Puig-Suari of California Polytechnic State University and Bob Twiggs of Stanford University [2].

Currently participating in the FYS programme, the ISTsat-1 is an 1U cubesat being developed by the ISTnanosat team, formed by students, professors and radioamateurs settled at Instituto Superior Técnico (IST). This satellite is one of the six educational space projects currently participating in the second edition of the FYS programme.

In addition to the educational purpose that builds on the enthusiasm of the students for space science and technology, complementing their education, the ISTsat-1 mission carries out a feasibility study of the use of nanosatellites to receive the signals of the Automatic Dependent Surveillance-Broadcast (ADS-B) system, which is used in the monitoring of aircraft specially in areas not covered by terrestrial stations. Therefore, the ISTsat-1 main mission is to collect relevant data from several aircraft about their flight status, temporally store it to be further forwarded to a ground station.

The nanosatellite that is being developed in-house is composed of four main subsystems plus the required structure and communication antennas, with the correspondent Antenna Deployment Mechanism (ADM), plus an additional subsystem dedicated to its specific payload. This satellite also follows a typical system structure of a cubesat, whose architecture is explained in Chapter 2.

1.1 Motivation

Nowadays, the existence of specialised companies that build and sell cubesat subsystem modules brings some advantages, as these modules have a guaranteed flight heritage, meaning that they have been successfully used in space and assuring their correct operation with low chance of failure during a mission. However, these commercial modules are highly expensive and inflexible, as they may not be reconfigured or remodulated into a custom mission. Three examples of these companies are *GOMspace*[2], *Pumpkin Space Systems*[3] and *AAC Clyde Space*[4].

However, in the ISTsat-1, the first cubesat built by the ISTnanosat team, all subsystems are designed in-house, only with the purchase of some specialised space components like the V/U antenna and its ADM used for the communication with the Ground Segment (GS), and the solar panels used in the energy harvesting. This means that the creation of the majority of the satellite modules brings some independence upon external implementations and increases the educational purpose of building a custom satellite rather than using a commercial one. These implementations become then more adaptable to the ISTsat-1 specifications as its components can be criteriously chosen, taking advantage of the small satellite space and reducing the power consumption. Because the system is built from scratch, the knowledge of the entire system is persistent during all phases of the satellite engineering, which could not be possible with a commercial module.

[2]GOMspace website: https://gomspace.com/home.aspx
[3]Pumpkin Space Systems website: https://www.pumpkinspace.com/
[4]AAC Clyde Space website: https://www.aac-clyde.space

Considering the Communications Processor Board (COM) that is one of the subsystems present in the ISTsat-1, an unfinished COMv1.0 iteration was done by the former IST student Tiago Carvalho, whose work is detailed in [3]. This iteration, which had some design issues in its external memory unit that needed to be solved for the COMv1 integration in the ISTsat-1 in order to fulfil its data storage requirements, served also as a motivation and starting point for this master thesis. Furthermore, the idea of an evolution of this subsystem to be integrated in future ISTnanosat cubesats also gives an extra motivation to accomplish the objectives set for this master thesis, which are further described.

1.2 Objectives

The purpose of this master thesis report is to describe the evolution of the Communications Processor Board subsystem developed for the ISTnanosat team. The work developed throughout this evolution focuses on the implementation and validation of two versions of this subsystem. The first objective of this master thesis is to detail the finished design of the COM first version (from now on referred to as **COMv1.1**), which integrates on the ISTsat-1 satellite and already had an unfinished iteration as aforementioned.

The COMv1.1 is mainly responsible for the ISTsat-1 communication stack handling and for the satellite data storage when the satellite is in its normal operational mode. In detail, this subsystem must:

- Be able to process the communication stack transceived between the satellite and the Ground Segment.
- Manage and report the correct operation of the various modules integrated in the subsystem itself.
- Handle the storage of telemetry data that reports the operation of the various modules from the satellite subsystems, plus the storage of payload data related with the satellite mission.
- Resort to well-defined interfaces to communicate with the remaining subsystems in the spacecraft.

Additionally to these requirements, the COMv1.1 may also assume the satellite control in its backup operational mode, when the On-Board Computer (OBC) subsystem has failed.

The second objective of this master thesis report is to describe a prototype design of the COM second version (from now on referred to as **COMv2**), being designed for an integration in the next generation of ISTnanosat team cubesats. The COMv2 may be considered as a merge of two subsystems existent on the ISTsat-1, namely the aforementioned COMv1.1 and its On-Board Computer, being an evolution of both. Therefore, additionally to the requirements already defined for the COMv1.1, this subsystem must:

- Determine the satellite attitude through the use of a system of inertial sensors.
- Control this attitude through control signals set on satellite actuators.

- Control the satellite operation, requesting telemetry data from the remaining subsystems and reporting hazardous events to the Ground Segment whenever possible through the subsystem dealing with the communication radio link.

Both COM subsystems described in this master thesis must also fulfil the cubesat standards, which restrains their dimension and the size of the modules used. Furthermore, due to the reduced energy budget, which is shared with the remaining subsystems in the satellite, the power consumption of this system must also be minimised as much as possible.

1.3 Report organization

This report is divided into 7 chapters.

Chapter 1 introduces this document, with a motivation and the objectives of this master thesis.

Chapter 2 describes the main subsystems and functions existent in common satellites, focusing on the increase of small satellite launches. Furthermore, a description of the aforementioned ISTsat-1 cubesat is done as an example of these small satellites. Finally, a brief explanation of the space radiation environment is done with the most significant radiation effects that may harm a satellite.

Chapter 3 provides a state of the art of communication systems, Command and Data Handling systems and Attitude Determination and Control Systems, which are the foundation for both subsystems developed in this master thesis. A comparison between some commercial On-Board Computers and communication modules is also done.

Chapter 4 details the architecture chosen for both COMv1.1 and COMv2 subsystems, listing all the requirements set for them.

Chapter 5 characterises the hardware and software implementations done on both subsystems, detailing the hardware and circuitry choices and the software designed to meet the requirements and architectures previously set.

Chapter 6 outlines the validation of the various modules implemented in these subsystems, evaluating their performance through specific tests and measurements.

Chapter 7 compares the systems developed to the objectives initially set, describing all achievements and problems experienced throughout the work developed in this master thesis. Finally, some future work is also detailed in this chapter.

Chapter 2

Satellite Classification and Space Environment. ISTsat-1 Case Study

Since the early times, mankind has dreamed about how life beyond planet Earth could be, deep into space. The first satellite built by mankind that successfully reached space was the *Sputnik 1*[1], launched on 4 October 1957. This satellite was a polished metal sphere with 58 cm of diameter and four radio antennas that broadcasted beacons (radio pulses transmitted periodically) detected by radio amateurs. This satellite orbited on a Low Earth Orbit, which extends between 160 km and 1 000 km of altitude.

Months later, on 31 January 1958, the American *Explorer 1*[2]was launched, with a missile format with 200 cm of length, 15 cm of diameter and 14 kg of weight, discovering the Van Allen radiation belt. This satellite got into higher Medium Earth Orbits, which extend between 1 000 km and 22 000 km.

These satellites were the first man made objects that were launched into space, starting the Space Race. Since then, the space industry has evolved with different satellite missions and into different standardisations. This chapter serves the purpose of categorising the satellite classification and their several subsystems, focusing on the increase of nanosatellite launches and describing the ISTsat-1 satellite in detail. A brief description of the space radiation environment is also done, as well as its impact in these nanosatellites.

2.1 Satellite Categorization

In the last years, size and weight of artificial satellites has steadily being reduced as technologies evolved, bringing the idea of classifying general satellites by their size.

[1] https://en.wikipedia.org/wiki/Sputnik_1 - Accessed on 23/12/2017.
[2] https://en.wikipedia.org/wiki/Explorer_1 - Accessed on 23/12/2017.

Therefore, according to [4], satellites can be categorised by their weight as:

- **Large satellites** - Mass bigger than 1000 kg;
- **Medium satellites** - Mass between 500 kg and 1000 kg;
- **Minisatellites** - Mass between 100 kg and 500 kg;
- **Microsatellites** - Mass between 10 kg and 100 kg;
- **Nanosatellites** - Mass between 1 kg and 10 kg;
- **Picosatellites** - Mass between 0.1 kg and 1 kg;
- **Femtosatellites** - Mass lower than 0.1 kg.

Although the various satellites built throughout the years may vary significantly in their size, their systems share common functionalities that can be pointed out. According to [5] and [6], the main functions and subsystems that can be found in a satellite are:

- **Communications** - A communication system allows the spacecraft to transmit data and telemetry to Earth and receive commands from it using transceivers. This communication is usually based in the radio frequency spectrum, with frequencies ranging from 30 MHz to 40 GHz, although laser based optical communications are also used. Depending on the satellite purpose and mission, this may be a heavy portion of the satellite.
- **Power** - A system is responsible for the power management and energy harvesting. Solar panels are used in combination with batteries to provide a constant source of electrical power on the satellite, even when the satellite is in eclipse without direct sunlight.
- **Command and Data Handling** - Several subsystems of a satellite are managed and controlled, dealing also with the data storage, filtering and compression.
- **Attitude** - A system monitors (**Determination**) and acts (**Control**) on the satellite attitude, positioning it constantly in its orbit, minimising perturbations like the solar radiation pressure, the interaction of the Earth's magnetic field and the gravity gradient.
- **Structural** - A satellite structure supports and mechanically connects all the subsystems of the spacecraft. However, the structural design must take into account the violent shocks and vibrations that a satellite faces during the rocket launch, being designed to withstand these forces.
- **Thermal** - As many satellite components have rigid temperature ranges in which they operate correctly, specially electronic components, the thermal subsystem regulates the temperature of the various components in a satellite so that they can operate correctly, maintaining their temperature within their specific range, either at eclipse or at direct sunlight.
- **Propulsion** - A system aids the satellite to get or stay in orbit, through chemical or electrical motors used to move the satellite back into the correct orbit when either atmospheric drag, magnetic fields or solar winds deflect the satellite out of its correct trajectory. The propulsion subsystem may also be considered as an active part of the satellite attitude control.

In addition to these subsystems, additional ones can be added according to each satellite developer, commonly related with the its mission, whether it is a commercial, scientific or military mission. In Figure 2.1, a summarised typical satellite structure is shown.

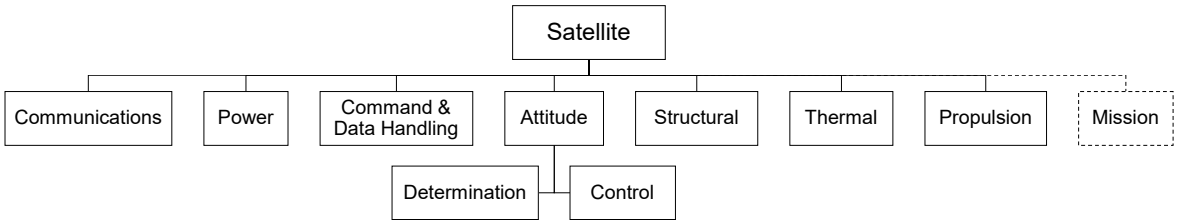


Figure 2.1: Typical satellite structure and their correspondent functions and subsystems.

2.2 The growth of small satellites launches

The use of many small satellites with specific functionalities or missions has increased, rather than using a single and bigger satellite with the same functionalities embedded together in it. This approach has allowed to reduce the cost of the missions, as satellites are built easier, cheaper and faster. Due to their small size, these satellites can also be launched in groups simultaneously from a single spacecraft, something that can not be done with a bigger satellite.

However, some limitations to the use of small satellites can be pointed out. Due to their size, these satellites stay less time in lower orbits than bigger satellites, being in operation for less time. Since the implementation of a propulsion subsystem is challenging due to the lack of space in a small satellite, this subsystem is usually not implemented and the satellite operational time in orbit is not expanded, as this subsystem could be useful to help a satellite keep itself in orbit. Beyond that, the use of less electronic components in small satellites reduces their processing power, energy storage and, consequently, causes less over-the-air data transmission.

Nevertheless, with this decrease in the satellite size, the creation of commercial models brought both cost and time reduction in the satellite manufacturing, as the access to space technology became easier. With similar satellite modules and structures, space deployers also became standardised, increasing the usage of standard satellites and their launches. According to [7], almost 1000 cubesat nanosatellites were launched to space between 2000 and 2018, as shown in Figure 2.2 depicting the evolution of launches per year. Their main missions can be divided into:

- **Communications**, as the primary mission is to relay communications point-to-point, being amateur radio service and Automatic Identification System (AIS) tracking some examples.
- **Educational**, as the main mission is the educational or professional training of the participants in the spacecraft design. Any science returns or technology demonstrations are of secondary value.

- **Earth Imaging**, as the mission returns images of the Earth for commercial and research purposes.
- **Science**, as the mission collects data for scientific research, including Earth science, atmospheric science or space weather.
- **Technology Demonstration**, as the mission involves the first flight of a new technology or design, advancing one or more Technology Readiness Levels[3].
- **Military**, when the mission has military relevance that does not fit in the aforementioned categories.

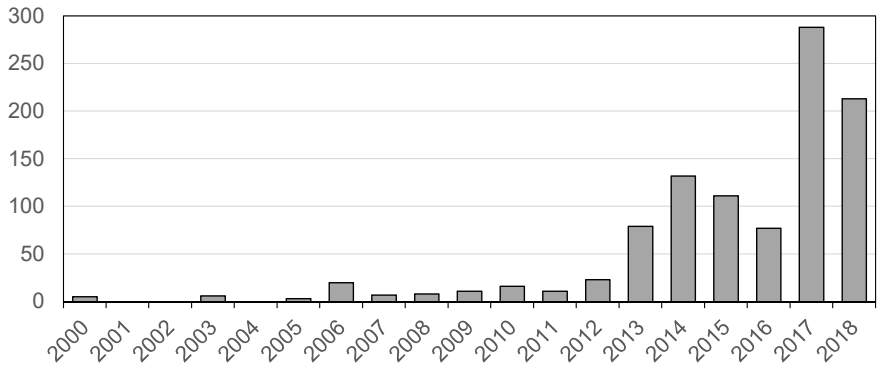


Figure 2.2: Evolution of the cubesat nanosatellite launches between 2000 and 2018 [Adapted from [7]].

2.3 ISTsat-1 and the Fly Your Satellite! programme

With the increased use of low-cost small satellites, the number of educational space studies made by worldwide universities has been increasing in the last years. Allied to that, some specialised space organisations have created programmes to help the launch of these studies and thus introduce university students to the space sector. One of these space organisations is the European Space Agency, whose Education Office has created the Fly Your Satellite! programme. In collaboration with several European universities, this programme has the objective of complementing the academic education of university students, inspiring and preparing them for a more effective introduction to a future work in the space sector [1].

Since 2008, this Education Office has enabled more than 10 cubesats to be launched into space. In the first FYS edition occurring between 2013 and 2016, three 1U cubesat satellites were launched into space. The current second FYS edition consists of four programme phases, where the cubesat teams have to achieve pre-defined objectives and must undergo reviews conducted by ESA specialists [1].

Being one of the educational space projects participating in the current second edition of the Fly Your Satellite! programme, the ISTsat-1 is an 1U cubesat being developed by the ISTnanosat team at Insti-

[3]Technology Readiness Level (TRL) is a type of measurement system used to assess the maturity level of a particular technology, which is evaluated against the parameters of different increasing levels of requirements [6].

tuto Superior Técnico, formed by students, professors and radioamateurs from the Portuguese AMRAD (*Associação Portuguesa de Amadores de Rádio para a Investigação, Educação e Desenvolvimento*).

In addition to the educational purpose that builds the enthusiasm of students for space science and that complements their education, the ISTsat-1 mission carries out a feasibility study of the use of low-cost planar antennas in nanosatellites to receive the signals of the ADS-B system, as these nanosatellites may be used in the monitoring of aircraft in areas not covered by terrestrial stations. Therefore, the ISTsat-1 will collect relevant data from several aircraft about their status, temporally store it to be forwarded further to a ground station, as shown in Figure 2.3.

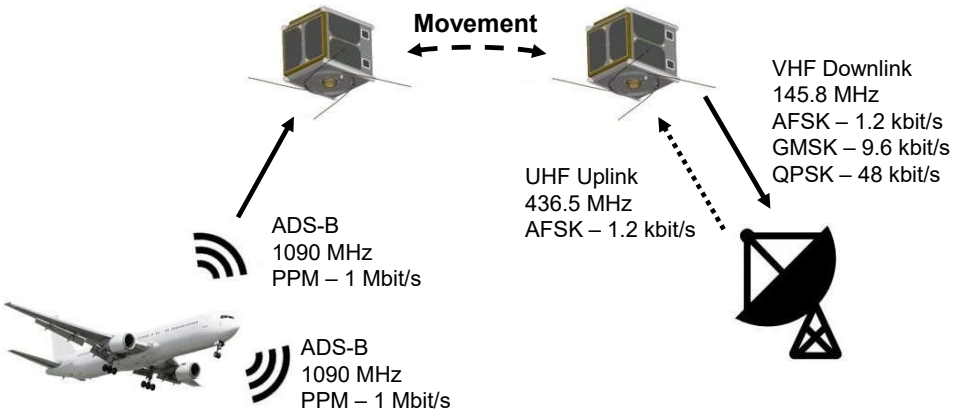


Figure 2.3: ADS-B message reception and storage, followed by the forwarding to the GS when possible.

The nanosatellite being developed is composed of four main subsystems plus an additional one dedicated to its specific mission, following a typical structure of a cubesat. These subsystems are:

- The **Tracking, Telemetry and Control (TT&C)**, responsible for the data uplink and downlink between the satellite and several ground stations, using a UHF frequency of 436.5 MHz and data rate of 1.2 kbit/s for uplink, and a VHF frequency of 145.8 MHz and data rate varying from 1.2 kbit/s up to 48 kbit/s for downlink, as shown in Figure 2.3. Additionally, it has a **Beacon**, which is used to identify and locate the spacecraft, being independent from the rest of the satellite communication system, broadcasting a Morse message with its identifier and some satellite telemetry regarding the remaining subsystems operation. Both TT&C and Beacon share the same V/U antenna for communication.
- The **On-Board Computer (OBC)**, combining two different functionalities:
  - The **Attitude Determination and Control System (ADCS)**, responsible of keeping the satellite well-positioned and in orbit, while collecting data about its space location.
  - The **Command and Data Handling (C&DH)**, responsible for the monitor and control of the other subsystems and for the satellite housekeeping, being considered the satellite main controller.

- The **Electrical Power System (EPS)**, responsible of providing the power to the satellite and managing the energy harvest from the solar panels, storing it in batteries and regulating the power necessary to supply the satellite subsystems.
- The **Communications Processor Board (COM)**, cooperating with the OBC subsystem, is responsible for the satellite communication stack handling and for the satellite data storage, when the spacecraft is in its normal operational mode. In case of OBC failure, the COM subsystem assumes the aforementioned C&DH functionality but not ADCS one, becoming the satellite housekeeper[4] in a backup operational mode.
- The **ADS-B Payload**, being the only scientific payload subsystem, is responsible for receiving the ADS-B signals broadcasted by commercial aircraft and translate them into ADS-B messages to be downloaded to Earth when possible, being connected to the ADS-B antenna.

All these subsystems are being developed and produced by the ISTnanosat team, being the ISTsat-1 cubesat in the *Test Your Satellite!* phase that will last until December 2019 [1]. In Figure 2.4 is shown an overview of the ISTsat-1 architecture and the stacking layout of its subsystems.

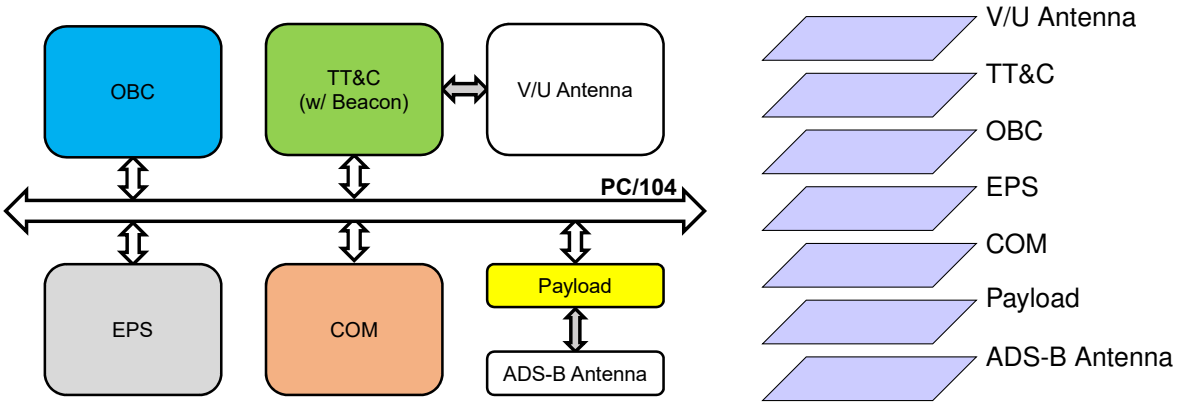


Figure 2.4: ISTsat-1 architecture (left) and subsystem stacking layout (right).

2.4 Space radiation environment

Unlike the protected environment in the Earth's atmosphere, the space environment stresses spacecraft components with many hazardous types of cosmic radiation. This radiation may come from a variety of emission sources in the solar system, like the Galactic Cosmic Rays and the Sun radiation. The former are mainly composed of energetic protons and heavier ions with around 2 % of electrons and positrons, with a relative constant presence over time. The latter is composed mostly of low energy particles like protons, being also known as solar wind [8].

[4]The satellite housekeeper subsystems is responsible for the operation of the satellite and, consequently, the operation of the remaining subsystems.

Within space applications, these radiation types can damage electronic components in a satellite. Total Ionising Dose (TID) and Single Event Effects (SEE) are two ways of characterising the impact of this radiation. The TID is measured as the cumulative radiation received by the spacecraft, mostly due to electrons and protons. Some consequences of long term exposure to radiation are the increase of device leakage and power consumption, shifts in component thresholds and clock changes, making the device work out of its specifications or even cease its operation [8].

A Single Event Effect results from high energy particles like cosmic rays and high energy protons that hit the spacecraft electronic components. These disturbances, which result from the ionisation caused by these particles, are considered as *destructive* (D), if a device function is interrupted or permanently damaged, and *non-destructive* (ND), if the state of a device is momentarily changed with no affect to its functionality [8], [9]. A brief description of the most important SEEs is shown in Table 2.1, with the components affected in each case.

Table 2.1: Description of the main Single Event Effects that affect a spacecraft [adapted from [9]].

Single Event Effect	D/ND	Affected Components	Description
Single Event Upset	ND	Memories Latches in logic devices	Corruption of the information stored in a memory element.
Multiple Bit Upset	ND	Memories Latches in logic devices	Several memory elements corrupted by a single strike.
Single Event Functional Interrupt	ND	State Machines Control Sections	Complex devices with built-in corruption of a data path leading to temporal loss of the device functionality.
Single Event Transient	ND	Analog and Mixed Signal circuits	Impulse response on external/internal signals of certain amplitude and duration.
Single Event Disturb	ND	Combinational logic Latches in logic devices	Momentary corruption of the information stored in a bit.
Single Event Latchup	D	CMOS BiCMOS devices	High-current conditions due to short circuits.
Single Event Snapback	D	N-channel MOSFET SOI devices	High-current conditions due to short circuits.
Single Event Burnout	D	BJT N-channel Power MOSFET	Destructive burnout due to high-current conditions.
Single Event Gate Rupture	D	Power MOSFET Non-volatile NMOS	Rupture of gate dielectric due to high electrical field conditions.
Single Hard Error	D	Memories Latches in logic devices	Unalterable change of state in a memory element.

2.4.1 Radiation mitigation

Radiation mitigation is important in the design of space satellites to reduce the severity of the radiation effects into the satellite electronics. Therefore, due to the aforementioned effects, space missions require satellite developers to implement radiation mitigation techniques in their designs.

Although many satellite designers use commercial off-the-shelf (COTS) devices as easy ready-made and sale-available components to build their custom cubesats, this low cost option may not be viable in terms of radiation mitigation since these components are not designed to withstand the space radiation environment. However, some component vendors also provide radiation hardened devices with the same functionalities, although these devices bring an impact not only to the cost budget of the spacecraft, as they are more expensive, but also to the mass budget and system design, as these components are usually bigger and heavier. However, using radiation hardened components is not the only measure taken into account in the satellite radiation mitigation. According to [6], additional measures are:

- The use of **monitor circuits** like a Watchdog Timer (WDT), which is used to check the state of a processing unit. This device resets the processor if its timer is not reset periodically after becoming non-operational due to a SEE. Although most microcontrollers have a WDT, using an external integrated circuit with the same functionality is also a good practice, as a redundant measure.
- The use of **overcurrent protections** that detect and reset components or the whole subsystem before an high current, caused mainly by Single Event Latchups, causes permanent damage.
- **Powering off** some modules within a subsystem separately when these are not required, as many components are more prone to radiation effects when powered on.
- The use of **communication protections** in peripherals like I^2C to prevent hazardous situations when an individual device locks up the communication bus after a SEE or a software glitch. Here, a master controller can monitor and reset the device to restore the bus for all the subsystems. The decouple of both clock and data lines from the main bus is another viable solution, so that each subsystem may isolate its bus from the main bus in a hazardous situation.

Although the aforementioned implementations may also seem important for non-space electronic circuitry, these mitigation techniques are mandatory in a space-based system to increase its robustness and making it fail-proof even in a radiation environment, since the cost and technology used in an in-orbit satellite repair easily outnumbers the cost of the satellite itself, specially in low-cost small satellites, making the repair itself worthless most of the times. Therefore, a satellite and its subsystems must be prepared to prevent and withstand as many fail-cases as possible during their lifetime.

In this Chapter 2, the main structure and functions existent in satellites were introduced, focusing on the increase of nanosatellites launches in the last years and their correspondent missions. As an example of these small satellites, a description of the ISTsat-1 cubesat is done, which supports the development of this master thesis. Additionally, a description of the space radiation environment was done with the most significant radiation effects that may harm a satellite, specially small satellites like cubesats. To prevent a subsystem level failure, some radiation mitigation techniques are detailed, whose implementation is crucial in the electronics design of the several subsystems.

Chapter 3

State of the Art

In Chapter 2, the main functionalities existent in satellites are introduced. Concerning the objectives of this master thesis described in Section 1.2, three of these functionalities prevail, which may take part of the Communications Processor Board in the ISTnanosat nanosatellites:

- The satellite communications.
- The command and data handling;
- The attitude determination and control.

As the COM subsystems designed throughout this master thesis must only process the communication stack transferred between the satellite and the Ground Segment, the design of a radio transceiver used for the communication with the Ground Segment is done in another communication subsystem like a TT&C. The communication stack is then transferred between this subsystem and the COM using a dedicated serial communication interface and protocol defined by the ISTnanosat team. Therefore, the COM subsystem must only process and transfer between the TT&C the protocols inherent to the communication stack. Nevertheless, a brief analysis of communication systems used in nanosatellites is done in this Chapter 3, along with both C&DH and ADCS modules, since their functionalities are also considered for the subsystems designed in this master thesis, specially for the COMv2.

Due to the small form factor imposed by the cubesat standard, these three functionalities are hard to be implemented in a single subsystem, being commonly divided into two separated subsystems: one focused on the satellite communications and the other dedicated to the C&DH and ADCS, being called as an On-Board Computer module. The latter module requires a high power processing and memory units along with determination and control modules for the satellite attitude. Nevertheless, another commercial solutions may separate these functionalities into three correspondent subsystems. Due to this ambiguity, a study regarding different combinations of subsystems that combine these functionalities is done, considering commercial systems developed by companies well-known in the cubesat market. Finally, a comparison between these combined solutions is done, with their pros and cons.

3.1 Communication Systems

The communication system is an essential part of a satellite that enables the transmission of both telemetry and payload data to the Ground Segment, while receiving commands from these stations. The communication between Earth and a satellite is commonly based in the radio frequency spectrum, using frequencies from 30 MHz up to 40 GHz, as shown in Figure 3.1. Additionally, laser based optical communications are also used, which are not considered in the current state of the art.

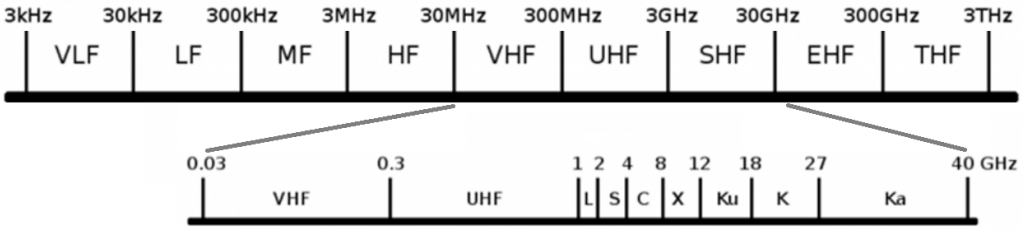


Figure 3.1: Radio frequency spectrum used for satellite communications [adapted from [6]].

Nowadays, many nanosatellites resort to radio frequencies in the VHF and UHF bands to communicate, as many communication modules based on these bands already have flight heritage. However, the use of higher frequency bands in small satellites has been increasing in the past years, with frequencies up to the Ka-band, enabling the use of higher bandwidths and resultant higher data rates [6].

Although the use of nanosatellites is being considered for deep space missions in the future, these satellites still have not been launched to orbits beyond the LEO. Therefore, small satellites launched to these orbits mainly use low gain antennas, like whip or patch antennas, for their communication systems, taking advantage of the short orbit distance. Due to their monopole antenna polarisation, these antennas can also maintain a communication link even when the spacecraft is tumbling, which is also an advantage for nanosatellites that commonly do not have an accurate attitude control. While whip antennas are commonly used for VHF and UHF communications, small patch antennas are used from UHF up to S-bands [6].

Another development that is improving radio frequency based communication systems used in small satellites is the implementation of a Software Defined Radio (SDR). This device is usually based on Field Programmable Gate Array (FPGA) hardware that offers great flexibility, allowing the use of multiple radio frequency bands with different filtering and modulation schemes without hardware changes. Furthermore, since the radio configuration is mostly based on software, the correspondent changes can be done during flight by uploading new software from the Ground Segment, improving the modularity of the communications system. As SDRs are becoming increasingly small and power efficient, their use is also becoming a trend in nanosatellites like cubesats [6].

3.2 Command and Data Handling

Like in any other embedded system, a satellite must rely at least on one system that controls its operation, managing its features and other secondary subsystems, in order to meet its requirements and correct operation. In a cubesat, the main objective of a Command and Data Handling (C&DH) module is to manage and control the other subsystems in the satellite and manage the storage of relevant data to be sent to the Ground Segment. Additionally, this module can also process this data into a well-defined communication protocol stack before sending it to the GS through the satellite transceiver. On the opposite radio link direction, this subsystem may also process the data received from the GS. Therefore, being the satellite main controller, the C&DH subsystem requires high power processing and storage capabilities, being commonly designed with a high power processing unit plus a memory unit to ensure that the data storage of the satellite is not compromised.

3.2.1 High power processing units

Among the recently cubesat implementations, the use of low power microcontrollers and FPGAs can be seen, supporting different processor cores. Nowadays FPGAs provide high levels of integration in small satellite implementations, supplying peripherals, on-chip memories and having an improved performance with low power consumptions. These processing units built a successfull legacy in space as they became one choice on the on-board computing needed in cubesat subsystems [6].

On the other hand, the use of processor-based microcontrollers, like the ARM or the MSP430 product lines, is a trend in cubesats, as they offer a low power solution with fast code executions and the typical on-chip peripherals that are used for the serial communication between the several subsystems within the satellite, like the Controller Area Network (CAN), as well as the Inter-Integrated Circuit (I^2C) interfaces and the Serial Peripheral Interface (SPI). Another feature considered by cubesat developers are microcontrollers that integrate internal memory, which can be programmed more easily [6]. In Section 3.4 a detailed study is done regarding the performance and power efficiency of microcontrollers used in C&DH modules, which are developed by cubesat specialised manufacturers and that have successfully flown in space missions, becoming modules with flight heritage.

3.2.2 Memory unit components

Regarding the selection of memory components to be used in cubesats, many different technology types are available due to their wide ranges, typically starting at 32 kB. However, the main requirement of memory units for satellite systems is reliability, which can be achieved with different types of memory and with different costs, depending on their purpose. Nowadays, the most used memory technologies are the Static Random Access Memory (SRAM) and the Dynamic Random Access Memory (DRAM) for

volatile memory data, and the Flash memory, the Magnetoresistive Random Access Memory (MRAM) and the Ferro-electric Random Access Memory (FeRAM) for persistent memory data. A comparison of these memory types and their performances is shown in Table 3.1.

Table 3.1: Comparison of different memory technologies [adapted from [6]].

Feature	SRAM	DRAM	Flash	MRAM	FeRAM
Volatile	Yes	Yes	No	No	No
Data Retention	None	None	>10 years	>10 years	>10 years
Write/Read Cycles	Unlimited	Unlimited	10^6 cycles	10^{13} cycles	10^{13} cycles
Radiation (TID)	1 Mrad	50 krad	30 krad	1 Mrad	1 Mrad
Single event upset rate	Low	High	Low-med	Nil	Nil
Temperature Range	Military	Industrial	Commercial	Military	Military
Power	500 mW	300 mW	30 mW	900 mW	270 mW
Package	4 MB	128 MB	128-256 MB	1 MB	1.5 MB
Price	Medium	Medium	Low	High	High

For cubesat missions, which usually last no more than 5 years due to their low orbit, factors like the retention time of a memory and its data operation cycles are of low importance. However, the radiation effects on these electronic components are of great importance as they can cease their normal operation, as explained in Section 2.4. Therefore, the selection of memory units must be taken into consideration regarding these factors. Like in any other satellite, the power consumption is another important factor to be considered due to the reduced energy harvesting in small satellites like cubesats.

3.3 Attitude Determination and Control System

Depending on its mission, a satellite is placed into a specific orbit. However, its position can deviate from this orbit commonly due to perturbations like the solar radiation pressure, the interaction of the Earth magnetic field and gravity gradient, and the atmospheric drag. These factors also affect cubesats that orbit at Low Earth Orbits.

The main objective of the Attitude Determination and Control System (ADCS) module is to monitor and act in the satellite attitude, positioning it constantly in its orbit and, therefore, minimising the aforementioned perturbations. The correct attitude of a satellite depends on its status and purpose at a specific time, as a satellite can move itself in orbit. For example, a different positioning may be recommended when radio transmissions are being done to Earth rather than when the satellite is harvesting energy from the sun. Therefore, depending on the situation and objective, an ADCS module can be in one of the following modes, which are normally executed in a specific sequence depending of each spacecraft [10]:

- **Detumbling** - The system measures the speed of rotation and acts on the satellite to reduce it. As the angular momentum is reduced, the determination algorithm can successfully work.
- **Target Pointing** - The satellite focuses on a specific point on the Earth's surface, even with the change of its orbit position.
- **Manoeuvring or Nadir Pointing** - Normal operating mode. The system points the satellite to the spherical centre of the Earth, also known as Earth nadir point.
- **Stabilisation or Inertial Pointing** - The satellite is oriented so that its attitude becomes stabilised to, for example, harvest energy from the Sun by pointing the solar panels directly to it.
- **Desaturation** - As the satellite magnetic sensors and actuators reach their saturation and become unreliable for operation, a magnetic field is forced to overcome this effect and desaturate them.
- **Offline** - The satellite orientation correction system is switched off, having no attitude control to save energy.

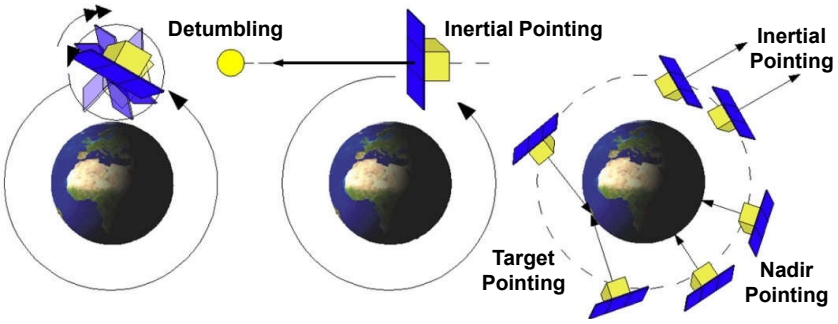


Figure 3.2: Different orientation modes during the satellite attitude determination and control[1].

Allied to these orientation modes, several stabilisation techniques are used, also depending on the satellite mission and its requirements [10]:

- **Gravity Gradient** - The satellite releases a spherical mass connected by a cable and, due to the Newton's law of universal gravitation, the closest mass to Earth will be more attracted and the satellite will line up with the Earth nadir point.
- **Magnetic** - Controllable inductors are used to create magnetic fields that will interact with the Earth's magnetic field.
- **Rotational** - The satellite is rotated on a single axis with a fixed angular momentum vector, making the satellite more resistant to external disturbances.
- **Double rotational** - Using two different angular speeds on the same axis, one is in favour of the angular momentum and the other is against it, stabilising the on-board instruments.
- **Bias Momentum** - With a reaction wheel[2] aligned with an axis, a speed change of that wheel provides a satellite rotation variation in this axis and in the opposite direction, based on Newton's

[1] http://www.kleinsatelliten.de/flying_laptop/systeme/lageregelung.en.html - Adapted on 17/02/2018.
[2] A reaction wheel is a type of flywheel used by spacecraft for three axis attitude control that does not require external torque.

Third Law.

- **Momentum Cancellation** - With reaction wheels in the three axes of the spacecraft, these are accelerated slowly in the same direction of the external perturbation until they reach their maximum speed. Then, with other actuators, the desaturation of these wheels is done.

Among these stabilisation techniques, the accuracy of passive control systems like the gravitational and magnetic gradients is lower than the other active systems, being only an option to consider when the satellite resources are limited, like in a cubesat. However, momentum cancellation is the most used technique because it allows to manoeuvre all three axes of the satellite with 1° of precision [10].

As the name implies, an Attitude Determination and Control System module can be divided into two groups with different functionalities:

- **Attitude Determination**, which relies on sensors to check the satellite position.
- **Attitude Control**, which relies on actuators to intentionally manipulate the satellite, achieving the desired attitude.

The interconnection between these two groups is done using a processing unit, which collects the data from the sensors through an electronic interface, processes and stores it. After that, the necessary corrections to the satellite attitude are calculated and sent to the actuator drivers.

3.3.1 ADCS sensors

A set of sensors is a major requirement in the spacecraft attitude determination, obtaining continuously accurate angular velocities of the satellite and their correspondent variations. Among the existent sensors, two types prevail: the reference sensors, which provide a value against a reference, like the direction of the sun, and the inertial sensors, which provide continuous attitude readings. To improve the performance and redundancy of the system, the two types are usually used together, having implications on the cost, energy consumption and mass of the spacecraft. However, the information processing of both sensor types when working together is essential to create a system that obtains reliable data for the satellite control [10].

The technology of sensor components has evolved significantly over the last years, making electronic devices more reliable and with less size and becoming available in microelectromechanical systems (MEMS) devices. Some sensors used in the satellite orientation are shown in Table 3.2, as well as their purpose, accuracy, advantages and disadvantages.

Among these sensors, the star orientation using a camera is the most accurate method, although it requires a complex processing power to determine the satellite position. Although MEMS magnetometers and gyroscopes are less accurate than other sensors, they are viable options for small satellites due to their size and reduced cost. Moreover, because the gyroscope accuracy changes with its running

Table 3.2: Comparison of different attitude sensors [adapted from [10]].

Sensor	Accuracy [°]	Axis	Pros	Cons	Brief Definition
Sun Sensor	0.1	2	Reduced cost, Simple, Viable	Unusable in solar eclipse	Determines the satellite position relative to the Sun.
Camera (star-oriented)	0.001	3	High accuracy, Absolute positioning	Cost, High mass, Complex use	Uses an internal database and compares it with the observed stars.
Magnetometer (MEMS)	0.5 - 5	1 - 3	Reduced cost, Compact	Only for LEO	Measures the Earth's magnetic field intensity.
Gyroscope (MEMS)	0.01/hour	3	Accuracy	Cost, Changeable accuracy	Expresses the angular changes suffered by the satellite.
GPS	0.1 - 1	-	Easy to use, Absolute positioning	Licensed	Determines the exact position of the satellite with information sent by other satellites.

time, and because accelerometers and magnetometers require gravitational and magnetic fields, they can only work in LEO, where these fields are reasonably felt. Finally, the simplest way to get the absolute position of a satellite is using a GPS, but it requires a license for altitudes higher than 18 km.

3.3.2 ADCS actuators

Similarly to the attitude determination, the satellite attitude control requires a set of actuators to control and impose forces to it. The power consumption, size and lifetime of a satellite are important factors to take into account when choosing actuators, as the attitude control of a small satellite is usually done with a single actuator due to these factors, unlike the attitude determination which may use different sensors.

Each actuator is designed to compensate a specific effect, being divided into inertial actuators that generate torques to modify the angular momentum of the spacecraft, and non-inertial actuators that use favourable external disturbances to achieve the desired attitude control [10]. Table 3.3 lists some of the actuators that can be used in satellites, along with their advantages and disadvantages, range of action and a brief description.

Table 3.3: Comparison of different attitude actuators [adapted from [10]].

Actuator	Range	Pros	Cons	Brief Definition
Gas Propulsion	0.1 – 5 N (cold) 0.5 – 9 000 N (hot)	High accuracy, High range	Size Ceasing Gas	Release of a gas generating a reaction force from the satellite.
Magnetic Inductor	10 μNm – 0.04 Nm	Low mass, No mobile parts	Only in LEO	Creates a magnetic field that interacts with an external field.
Reaction Wheel	0.001 – 1 Nm	High precision	Size, cost	The speed of rotation changes the torque created on a given axis.

3.4 Commercial OBC and communication modules for cubesats

In this Section 3.4, some combinations of both communication and On-Board Computer solutions developed by specialised companies are shown, which are built for cubesat integration. These systems may implement the previously described functionalities, focusing separately or together on communications, C&DH or ADCS purposes. Therefore, factors like the communication frequency band and bandwidths/data rates, the architecture of the processing and memory units, the availability of sensors and actuators and the power consumption are detailed. Please note that the presented combinations are not the only ones available in the market, as other products and companies exist.

GOMspace Software Defined Radio - NanoMind Z7000 and NanoCom TR-600

Being one of the trending companies in the small satellite market, GOMspace designs many different subsystems to be integrated in cubesats. One of them is its integrated SDR composed of two main systems that are interconnected through a docking board following the cubesat standard.

The first system is the *NanoMind Z7000* shown in Figure 3.3, being a high processing power module with flight heritage used for communication and signal and image processing systems. This module has an efficient system-on-a-chip (SoC) with the software programmability of a dual core 800 MHz ARM processor, capable of running a Linux operating system, plus the hardware programmability of an FPGA. For storage, 1 GB of DDR3 RAM and 32 GB of eMMC storage are used. As this product is built to work as a SDR, it does not feature ADCS modules. Regarding its serial communication peripherals, this module can communicate through USB, SPI, I^2C, UART and CAN interfaces. According to its datasheet information, the power consumption of this module is 2.3 W with a 3.3 V power supply [11].

Connected to this module is the *NanoCom TR-600*, also shown in Figure 3.3. This communication module is based on an Analog Devices AD9361 transceiver operating from 70 MHz up to 6 GHz, capable of providing a bandwidth from 200 kHz up to 56 MHz with its integrated 12-bit DACs and ADCs. According to its datasheet information, the maximum current consumption rated for this module is 4.0 A with a 3.6 V power supply [11].

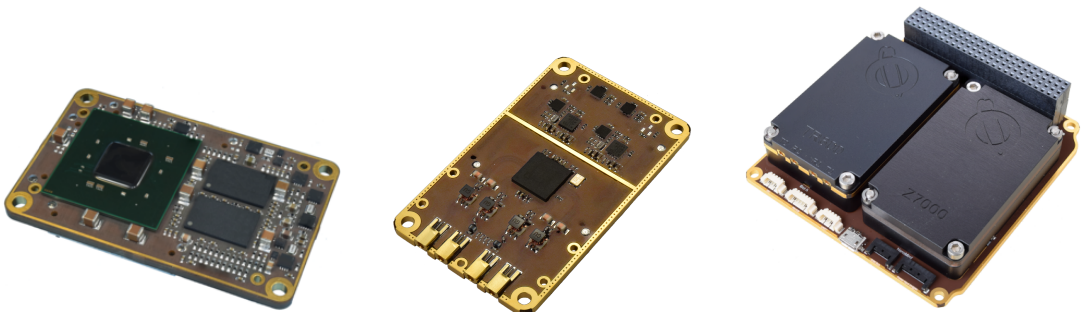


Figure 3.3: *NanoMind Z7000* (left), *NanoCom TR600* (centre) and *NanoDock SDR* (right) [adapted from [11]].

Both modules do not comply with the cubesat standard due to their miniaturised form-factor, requiring the use of a docking board like the *NanoDock SDR* shown in Figure 3.3 for cubesat compatibility, being pluggable modules.

GOMspace OBC and V/U communications module - NanoMind A3200 and NanoCom AX100

Another integrated option designed by GOMspace is the association of its OBC module with a V/U communication module, again interconnected through a docking board following the cubesat standard.

The *NanoMind A3200* On-Board Computer, which is shown in Figure 3.4, has a 64 MHz AVR32 microcontroller as a processing unit and 128 MB NOR Flash combined with 32 MB SDRAM for data storage. A magnetometer and gyroscope sensors are present in this product along with coil driver actuators that assure the ADCS functionalities. According to its datasheet, its typical and maximum power consumptions are 170 mW and 900 mW, respectively, with a 3.3 V power supply [11].

Connected to this module is the NanoCom AX100, also shown in Figure 3.4. This communication module is a half-duplex radio transceiver designed for three VHF and UHF narrow frequency bands (140-150 MHz; 395-405 MHz; 430-440 MHz), delivering configurable data rates from 0.1 kbit/s to 38.4 kbit/s and an output power from 24 to 30 dBm. According to its datasheet, the typical and maximum power consumptions of this module are 2.8 W and 3.2 mW, respectively, with a 3.3 V power supply [11].

Like in the previously described SDR, both modules do not comply with the cubesat standard due to their miniaturised form-factor, requiring the use of a docking board like the *NanoDock DMC-3* shown in Figure 3.4 for cubesat compatibility, being also pluggable modules.

Figure 3.4: *NanoMind A3200* (left), *NanoCom AX100* (centre) and *NanoDock DMC-3* (right) [adapted from [11]].

ISIS On-Board Computer and S-band transmitter

Another well-known company in the cubesat market is Innovative Solutions in Space (ISIS), also developing several subsystems for cubesats. Its On-Board Computer, shown in Figure 3.5, has a power efficient 400 MHz ARM9 processor plus 64 MB of SDRAM and 1 MB of Flash Memory for code storage, and 256 kB of FeRAM for critical data storage. For mass storage, two SD card slots can be used, providing up to 64 GB of data. As a safety implementation, this board has also an external on-board watchdog timer and can communicate through USB, SPI, I^2C and UART interfaces, having also

a JTAG interface for programming and GPIO, ADC and PWM channels. This module has an average power consumption of 400 mW with a 3.3 V power supply [12].

Although this OBC does not integrate ADCS modules directly, the use of optional pluggable daughterboards is another important feature of this product, as it provides a wide range of interfaces for payloads, sensors and actuators in a compact form factor, not only for ISIS daughterboards like the gyroscope or the magnetic control daughterboards, but also for custom ones built by cubesat designers [12].

Another system designed by ISIS is its high data rate S-band transmitter tuned for the 2.20–2.29 GHz frequency range, which is also shown in Figure 3.5. This communication module provides a data rate up to 4.3 Mbit/s and an output power between 27 dBm and 33 dBm, both configurable during flight. According to its datasheet, the typical and maximum power consumptions of this module during transmission are 13 W and 14 W, respectively, with a 16 V power supply [12].

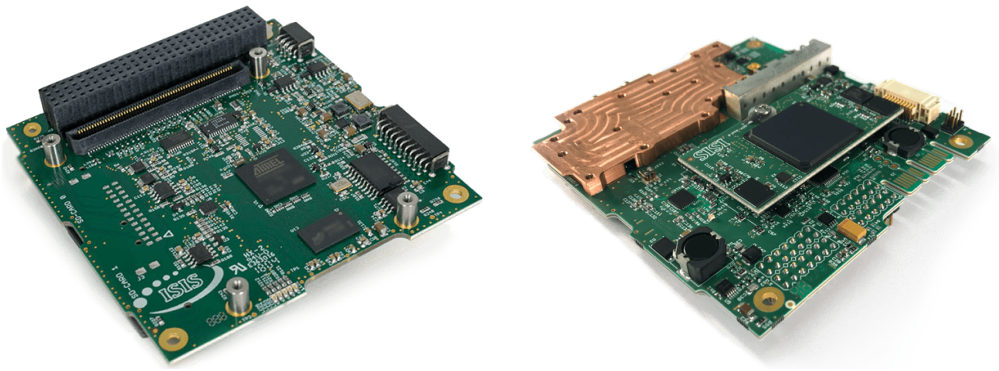


Figure 3.5: ISIS On-Board Computer (left) and S-band transmitter (right) [adapted from [12]].

AAC Clyde Space On-Board Computer and X-band transmitter

Another company in the small satellite market is AAC Clyde Space, with its On-Board Computer shown in Figure 3.6. This board has a SoC with a 50 MHz ARM Cortex-M3 processor plus FPGA hardware capabilities. 8 MB of MRAM are used for code storage and 4 GB of SLC Flash memory for data storage. As communication peripherals, this module communicates through UART, SPI and I^2C interfaces. As safety measures, this board also has an external watchdog and overcurrent protections to protect against SEL occurrences. However, it does not include any ADCS functionalities. This module has 400 mW and 1 W of typical and maximum power consumption, respectively, with a 3.3 V power supply [13].

Regarding high data rate communication modules, AAC Clyde Space offers a X-band transmitter tuned for the 8.025-8.45 GHz frequency range, which is also shown in Figure 3.6. This communication module provides a data rate up to 50 Mbit/s and a maximum output power of 33 dBm. According to its datasheet, the maximum power consumption of this communication module during transmission is 15 W, with power supply voltages ranging from 6.2 V to 17 V [13].

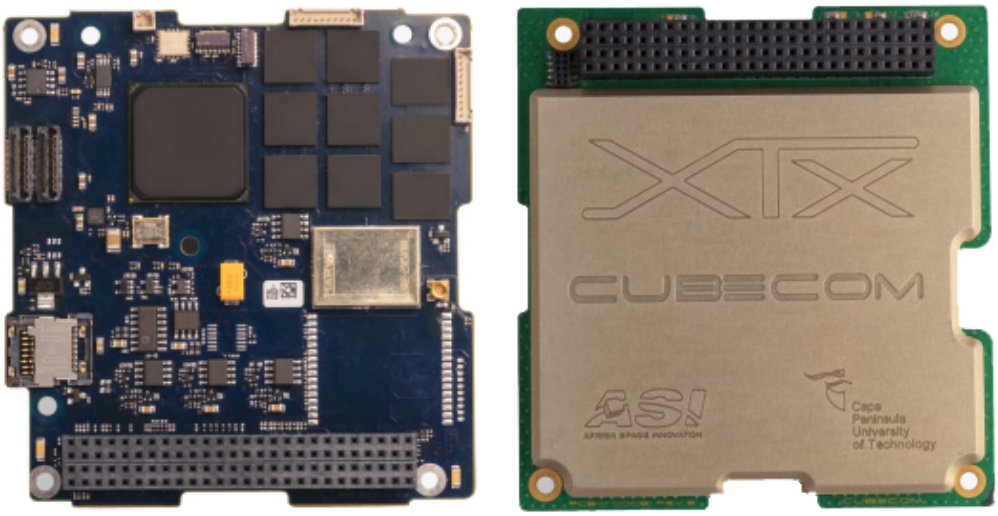


Figure 3.6: AAC Clyde Space On-Board Computer (left) and X-band transmitter (right) [adapted from [13]].

Comparison of the previously detailed commercial solutions

Chapter 3 focuses on three main functionalities present in a nanosatellite: communications, command and data handling and attitude determination and control. From the study of commercial modules made in Section 3.4, these functionalities are differently divided between several subsystems. For example, the GOMspace combination of its On-Board Computer plus its V/U communications module provides a solution where the three functionalities are offered on a single cubesat standard board, while the ISIS solution separates the communications functionalities into another subsystem, keeping the C&DH and optional ADCS functionalities on its OBC. Additionally, both GOMspace SDR and AAC Clyde Space solutions does not focus on the satellite attitude, which must be provided by another system. Nevertheless, these solutions offer an improved performance on their communication systems.

Regarding the different OBC modules, different architectural solutions may be pointed out, where this system may be in a single board or have a daughterboard attached to a main board. In the former, a simpler but non-modular solution is achieved. In the latter, the main processing unit may be on daughterboard, making this OBC system a modular processing solution where different microcontrollers may be used, or it may be on the motherboard, leaving the daughterboard free to be modular for the use of ADCS, data storage or payload functionalities, depending on the requirements and mission of each cubesat. Furthermore, the use of radiation mitigation techniques is crucial for the reliability of these systems due to the harsh space environment where they will operate.

In Table 3.4, the specifications obtained with the solutions previously studied in Section 3.4 are summarised, concerning the three desired functionalities and the achieved maximum power consumption.

Table 3.4: Summary of the specifications obtained with the previously detailed commercial modules.

Feature		GOMspace SDR	GOMspace OBC + VHF/UHF	ISIS OBC + S-band TX	AAC Clyde Space OBC + X-band TX
C&DH	Processing Power	2-Core ARM A9 800 MHz + FPGA	64 MHz AVR32	400 MHz ARM9	50 MHz ARM Cortex-M3
	Memory Storage	1 GB DDR3	128 MB Flash 32 MB SDRAM	1 MB Flash 256 kB FeRAM 64 MB SDRAM	8 MB MRAM
	Mass Storage	32 GB SD Card	None	64 GB SD Card	4 GB SLC Flash
ADCS	Sensors	None	Gyro + Mag	Yes*	None
	Actuators	None	Coil Drivers	Yes*	None
Comms	RX/TX	Both	Both	Only TX**	Only TX**
	Frequency Range	70 MHz-6 GHz	140-150 MHz 395-405 MHz 430-440 MHz	2.20–2.29 GHz	8.025-8.45 GHz
	Bandwidth/Data rate	200 kHz-56 MHz	0.1-38.4 kbit/s	Up to 4.3 Mbit/s	Up to 50 Mbit/s
	TX Output power	2-10 dBm	24-30 dBm	27-33 dBm	Up to 33 dBm
Maximum Power Consumption		$\approx$ 16 W	$\approx$ 4 W	$\approx$ 14 W	$\approx$ 16 W
Price		Price available only after request.			

*Available with daughterboard. **Data reception is possible with another V/U communication subsystem.

The GOMspace SDR offers the most modular solution for communications due to its full duplex transceiver. As a Software Defined Radio, its operation can be easily reconfigured during flight, having a wide operational frequency range and bandwidths. Although no satellite attitude functions are considered, this module can be also used for the C&DH requirements due to its high power processor and large memory unit. Even though its transmitter output power is much lower than the remaining solutions, this combination still has the highest maximum power consumption.

On the other hand, the GOMspace OBC plus V/U module offer the less power consumable solution, ideal for satellites with a low energy harvesting and with a low data rate transferred with the GS. Additionally, both attitude determination and control functionalities are considered, making this combination also the most space efficient, as the three desired functionalities are implemented in a single cubesat standard board.

Although both ISIS and AAC Clyde Space also provide a V/U communication module for both data transmission and reception, the correspondent S-band and X-band transmitters can be used when a high downlink data rate is required, being comparable with the performance of the GOMspace SDR. Although AAC Clyde Space solution provides a higher data rate resultant of its higher frequency band, the transmitter output power and power consumption of both modules are similar. However, the ISIS On-Board Computer may also be considered as an ADCS module with the use of its daughterboards, which is not possible with the AAC Clyde Space solution.

Chapter 4

System Architecture

In this Chapter 4, a summarised description is done regarding the system architecture of the subsystems proposed and developed in this master thesis, that illustrate the evolution of the Communications Processor Board for the ISTnanosat. Since the architecture of the subsystems integrating a satellite is directly related with the requirements that must be complied, the requirements for both COM versions are listed further, considering the permanent and specific functionalities to be implemented.

4.1 COMv1.1 Architecture

Being one of the five subsystems present in the ISTsat-1, the first objective for COMv1.1 is to be responsible of processing the ISTsat-1 communication stack, decoding and encoding all data in the space-link between the spacecraft and the Ground Segment, requiring for that a high-power processor. Additionally, to support the storage of both telemetry and mission data, an external memory unit is considered so that this storage is not compromised by the usual small and insufficient internal memory present in a microcontroller. From these main design ideas, requirements for the COMv1.1 arise considering its communication and C&DH functionalities, and input and output commands for the other subsystems, which are detailed further.

4.1.1 Communication and C&DH requirements

Permanent functions are procedures and operations that can be done in background while specific functions are running. Regarding the communication requirements, the main permanent function is to process the communication stack used between the satellite and its Ground Segment. On the order hand, the permanent functions for the C&DH requirements are:

- Provide data storage capabilities for telemetry and payload data, relying on a parallel memory unit.

- Apply compression techniques to this data to decrease the data flow from the satellite to the GS, and to mitigate the saturation of the aforementioned memory unit.
- Monitor the occurrence of Single Event Effects affecting the external memory unit.
- Assume the satellite housekeeping functions only in case of the permanent failure of the OBC (satellite backup operational mode).

A specific or secondary function is only done on demand by any subsystem in the satellite, including the COM itself, or through a Ground Segment command. These functions are:

- Store new satellite telemetry/housekeeping data after receiving it from the satellite housekeeper.
- Store new payload data after receiving it from the satellite payload.
- On memory storage fill-up, decide which data is discarded if no more compression can be done.
- Act accordingly to the occurrence of SEE, shutting down the memory unit partially or totally.
- Reset the processing unit after a hazardous SEE or software glitch blocks its operation.

4.1.2 Subsystem IO requirements

Like any other subsystem, the COMv1.1 subsystem must be able to receive and respond to commands or informations issued by the other satellite subsystems or by the Ground Segment. Therefore, the following functions are also considered:

- Communicate with other subsystems through the two main I^2C buses.
- Provide a dedicated communication interface with the TT&C for the satellite communication stack to be transferred between the satellite and the GS.
- Provide an heartbeat signal sent to the OBC housekeeping subsystem through the system bus, used to monitor and hard reset the COM subsystem if necessary.
- Report the state of the memory unit to the housekeeping subsystem.
- Enter a low-power mode after a GS command.

Additionally to these system and radiation mitigation requirements, the COMv1.1 must also be as power efficient as possible due to the low energy harvesting capabilities of the ISTsat-1 cubesat. Additionally, the design must follow the mechanical specifications defined by the PC/104[1] standard, in which the interconnection of the several subsystems is well-defined. Based on these requirements, Figure 4.1 summarises the architecture chosen for the COMv1.1.

[1] The PC/104 standard is a form factor for a stacked system bus with 104 pins and a well-defined pinout, commonly used in cubesats.

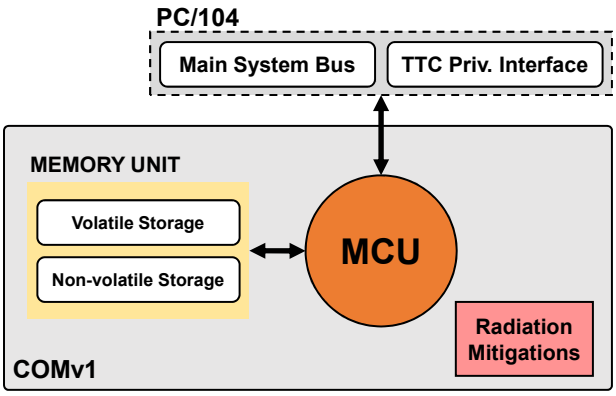


Figure 4.1: Architecture of the first version of the Communications Processor Board subsystem.

4.2 COMv2 Architecture

Being an evolution of the COM subsystem, the idea for the second version of the Communications Processor Board proposed in this master thesis is to expand the subsystem to an architecture where both functionalities of two subsystems existent on the ISTsat-1, namely its OBC and the aforementioned COMv1.1, can be merged into a single subsystem in a simpler architecture, or can be separate into two equally subsystems in a more complex architecture, performing its ADCS and C&DH functionalities separately or redundantly.

This modular solution allows the use of similar COM subsystems with different functionalities. For example, although a mission is not well-defined yet, one idea for the second ISTnanosat team mission suggests the use of one 3U cubesat which later splits into two different cubesats after launch, a 2U and an 1U. Both of these satellites shall have the same COM, but one of them may not require a big data storage unit, being more beneficial to have an additional small payload or have more ADCS components for a more precise attitude. Nevertheless, minimum storage and ADCS capabilities must be assured in both satellites, independently of the modular solution previously described.

Before analysing possible architectures, the requirements for the COMv2 are listed further, being divided into its communication, C&DH and ADCS requirements. Additionally, requirements regarding input and output commands for the other subsystems in the spacecraft are also detailed.

4.2.1 ADCS requirements

Regarding the ADCS requeriments, the aforementioned permanent functions are:

- Calculate the satellite angular momentum on a axis oriented to Earth.
- Determine the satellite attitude through sensor measurements, redundantly if possible.
- Manage the correct operation of the attitude sensors.

Additionally, the specific or secondary functions are:

- Generate PWM signals with different duty cycles for the magnetorquer drivers.
- Disable the ADCS functionalities when not required to safe energy.
- Provide an optional interface with a Global Navigation Satellite System (GNSS) for an accurate satellite position.

4.2.2 Communication and C&DH requirements

Regarding the communication requirements, the main permanent function remains as the processing of the communication stack used between the satellite and its Ground Segment. Regarding the C&DH requirements, the permanent functions are:

- Be the satellite housekeeper, controlling all subsystems in the satellite through commands and information requests.
- Provide data storage capabilities for telemetry and payload data, relying on a parallel memory unit.
- Apply compression techniques to this data to decrease the data flow from the satellite to the GS, and to mitigate the fill-up of the aforementioned memory unit.
- Monitor the occurrence of Single Event Effects affecting subsystem components.
- Verify the correct operation of implemented radiation mitigation techniques.

Furthermore, the specific or secondary functions are:

- Ask and store new satellite telemetry/housekeeping data after receiving it.
- Store new payload data after receiving it from the satellite payload.
- On memory storage fill-up, decide which data is discarded if no more compression can be done.
- Keep track of a time reference on power shutdown for at least one orbit of 90 minutes.
- Reset the processing unit after a hazardous SEE or software glitch blocks its operation.
- Act accordingly to the occurrence of a SEE affecting important subsystem components.
- Enter a low-power mode to safe energy whenever the processing unit is not required.

4.2.3 Subsystem IO requirements

Like any other subsystem, the COMv2 subsystem must be able to receive and respond to commands or information issued by the other satellite subsystems or by the Ground Segment. Therefore, the following communication functions also prevail:

- Communicate with other subsystems through the two main I^2C buses.
- Provide an alternative interface for the main system bus through a redundant CAN bus.

- Have a dedicated communication interface with the TT&C to transceive this communication stack.
- Provide an heartbeat signal sent to the EPS subsystem through the system bus, used to monitor and hard reset the COM subsystem if necessary.
- Receive commands from the GS and dispatch these for the remaining subsystems.
- Provide information of the satellite attitude to the GS.
- Report the state of the memory unit, sensors and other modules in the subsystem.
- Enter a low-power mode on demand by the GS, keeping only its sporadic radiation mitigation functions.
- Optionally provide, through the satellite system bus, the same signals used for the actuators in the attitude control.
- Have an additional serial communication interface dedicated to the satellite payload.

4.2.4 Architecture solutions

Recalling Section 3.4, different architecture solutions exist in commercial On-Board Computers. Using a single board, the space limitations due to the cubesat standard force the architecture to be simpler and not modular. For example, the use of a fast parallel memory unit like the one proposed for the COMv1.1 together with ADCS sensors and actuators is impossible due to the lack of PCB space, making this simple architecture not feasible for this COM design.

Another commercial solution for the COM architecture is the use of daughterboards, where the main processor unit can be placed on them or in the main board. The former solution is preferable when the use of different processing units is needed. For example, in the aforementioned mission with 2U and 1U cubesats, one of the COMs may require a more powerful processor to additionally manage payload data, also requiring high power consumption, while the other one may have a less powerful but low-power processor to deal only with ADCS and C&DH functionalities. However, this solution is also not viable because the use of large connectors with a large pin count is needed to connect all the modules to the several peripherals of the microcontroller placed on the daughterboard, occupying much board space and increasing the system complexity. Additionally, the modularity of the processing unit also increases the software complexity in the satellite since different software and driver development may be needed for each MCU.

The latter solution, where the motherboard has the main processing unit, leaves the daughterboard free to be modular, being used only if required. In that case, this board can be dedicated to ADCS, data storage or payload functionalities, depending on the requirements and mission of each cubesat. Furthermore, only one processing unit is chosen for the motherboard, easing the software development as opposed to the aforementioned architecture.

Therefore, the last option is chosen for the Communications Processor Board architecture, where a fixed motherboard has the processing unit with a small memory unit to assure the minimal storage requirements, plus the sensors and driver actuators required for minimal ADCS functionalities.

On the other hand, the daughterboard is optional with different possible designs, being directly connected to the microcontroller through a dedicated connector, following the approach of the ISIS On-Board Computer. Based on these requirements, Figure 4.2 summarises the architecture chosen for the COMv2.w≪

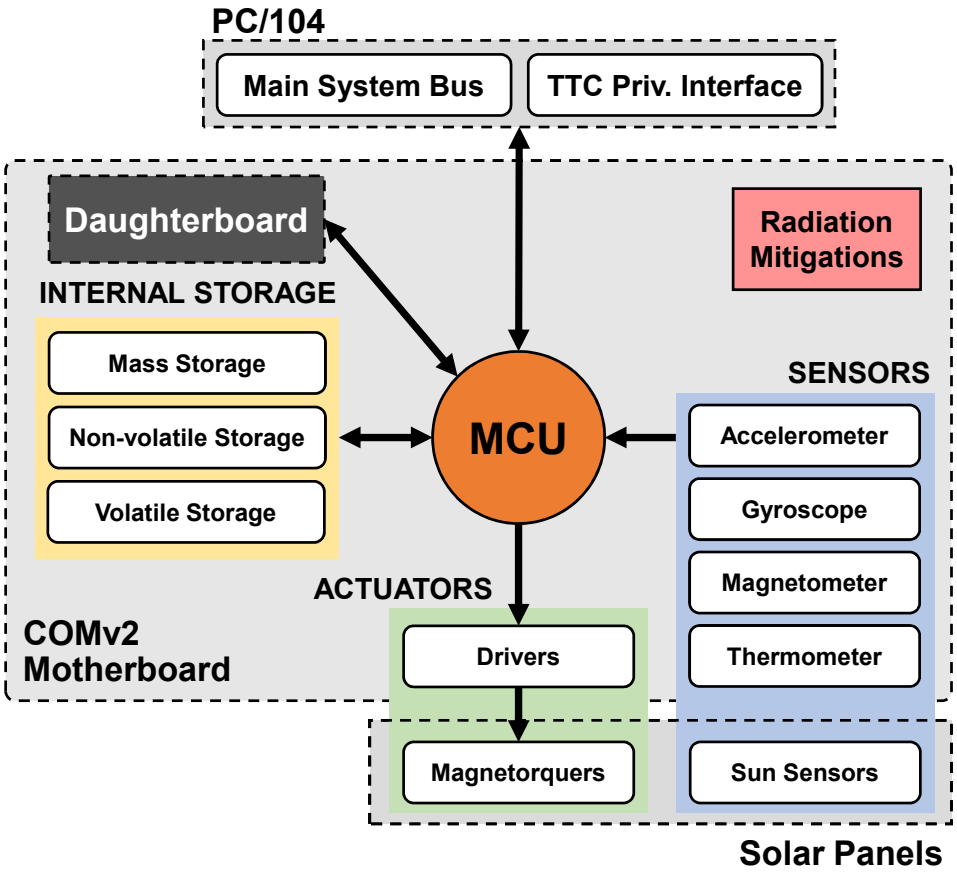


Figure 4.2: Architecture of the second version of the Communications Processor Board subsystem.

Chapter 5

System Implementation

In this Chapter 5, the components and modules chosen to achieve the previously discussed COM architectures are projected, describing also the interfaces and electrical connections between them. The purpose of each module is explained considering their hardware and software functionalities.

In the hardware project, the logical connections between the different modules are explained, followed by the explanation of the subsystem power regulation, control and distribution. The hardware structure of both subsystems differ since the COMv2 has additional functionalities and requirements when compared to the COMv1.1. For that reason, two different sections exist in this chapter dedicated to the hardware project of the first and second COM versions.

Finally, the software implementation is detailed, as well as the logical boot sequence that manages the subsystem peripherals. Since the software structure is similar in both subsystems, a single section dedicated to the system software project exists for both subsystems in this chapter.

5.1 COMv1.1 Hardware project

To support the explanation of the COMv1.1 hardware modules, the block diagram from Figure 5.1 represents an overview of the hardware architecture of this subsystem, where all the components and relevant interconnections are shown. This hardware architecture, which is described in detail in further sections, is mainly composed of:

- The microcontroller unit, whose connection to the subsystem modules and peripherals is done.
- An external parallel memory unit of both volatile and non-volatile memory.
- Peripherals chosen for the communication with the remaining satellite subsystems.
- Radiation mitigation circuits regarding the operation of the microcontroller and memory units.
- A Serial Wire Debug (SWD) programming and debugging interface.

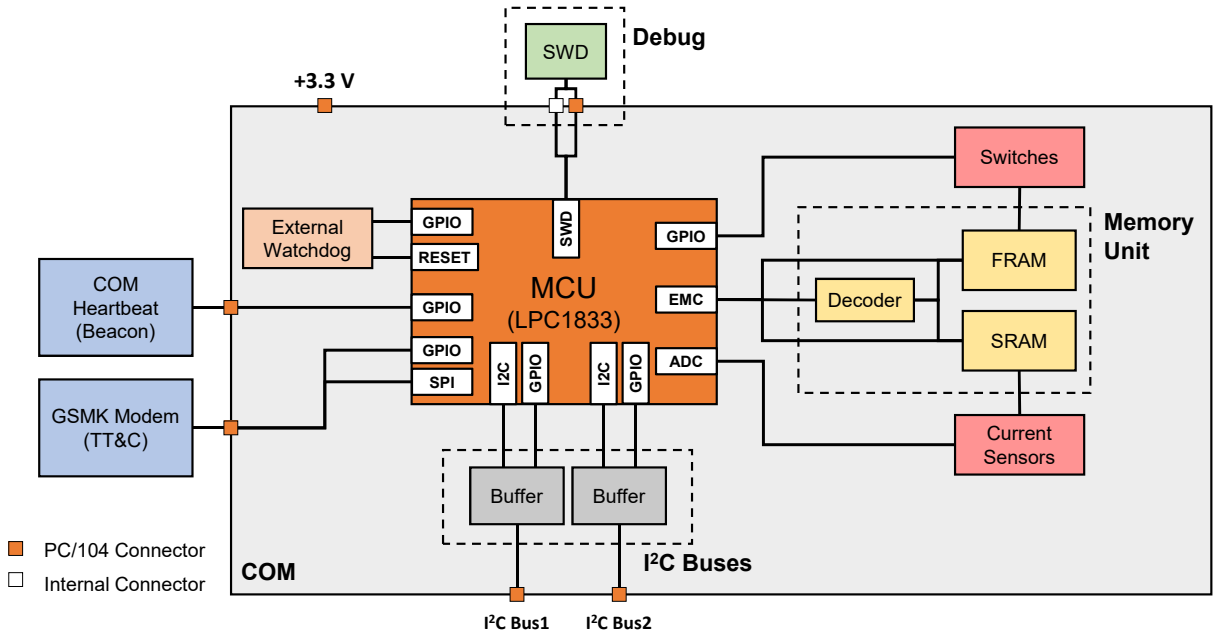


Figure 5.1: Block diagram of the COMv1.1 hardware architecture.

5.1.1 High-power processing unit selection

Because the COMv1.1 subsystem requires a high-power processing unit to meet its requirements, the first thing to be considered is the use of a powerful processing unit with low power consumption. The choice of this processing unit was done on [3] in the unfinished iteration COMv1.0, which considered the requirements shown in Table 5.1.

Table 5.1: List of COMv1.1 microcontroller requirements.

Characteristic	Requirement
Power consumption (without peripherals)	$< 300\ \mu$A/MHz
Serial Communication Peripherals	At least I^2C and SPI
External memory management unit	Static memory
Analog-to-digital converter	Yes
Low-power modes	Yes

Considering different manufacturers and their options, the microcontroller chosen is the LPC1833 from NXP Semiconductors, an 180 MHz ARM Cortex-M3 with 136 kB of internal RAM and 512 kB of internal Flash memory. This microcontroller is chosen due to its low power consumption and its External Memory Controller (ExtMC) capable of doing double word parallel addressing, which enables the use of fast memories that represent a major requirement of this subsystem [14].

5.1.2 Subsystem Power and PCB design

Since the ISTnanosat team has decided that the EPS subsystem must provide separate regulated 3.3 V power lines for each subsystem in the satellite, all the COMv1.1 subsystem is simply powered by its dedicated 3.3 V power line.

In order to keep the Electromagnetic Compatibility (EMC) with the remaining subsystems in the satellite, a multi-layered Printed Circuit Board (PCB) is used, with power and ground planes separated into different PCB layers, and analog and digital planes separated from each other whenever possible on the component layers, as shown in Figure 5.2.

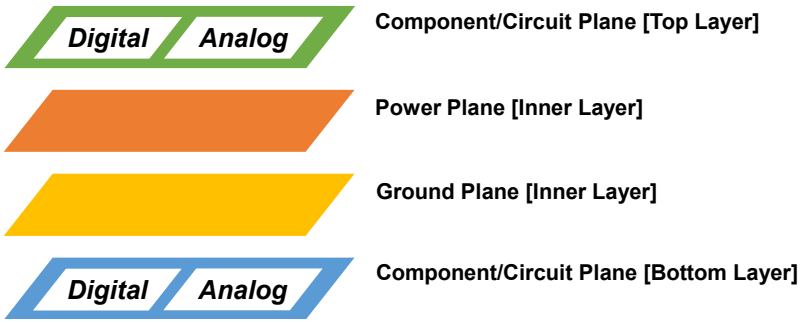


Figure 5.2: PCB layer design scheme, with power and ground planes separated into two distinct inner layers.

5.1.3 External Watchdog Timer

Although the chosen microcontroller has an internal watchdog timer, an external one is also considered to provide redundancy to the system and act when a Single Event Effect or software glitch prevents the internal one from operating correctly. For that reason, the voltage monitor TPS3823A[1] from Texas Instruments was chosen. Besides monitoring the microcontroller supply voltage, this integrated circuit also offers an external watchdog timer, being connected to the microcontroller RESET active low line.

Regarding the MCU supply voltage monitoring, when this voltage drops below a threshold voltage of 3.0 V, the RESET line is forced low for at least 200 ms or until the supply voltage becomes greater than this threshold voltage, stopping the microcontroller operation if the supply voltage is not sufficient. Considering the watchdog timer function, which is demonstrated in Figure 5.3, this IC provides an input WDT_Feed signal that must be toggled by the microcontroller under a 2 s timeout period, usually through a GPIO. If this commutation fails due to a software glitch, the RESET line is forced low for 200 ms, enabling the MCU to restart its operation successfully.

[1] http://www.ti.com/lit/ds/symlink/tps3823.pdf - Accessed on 08/01/2018

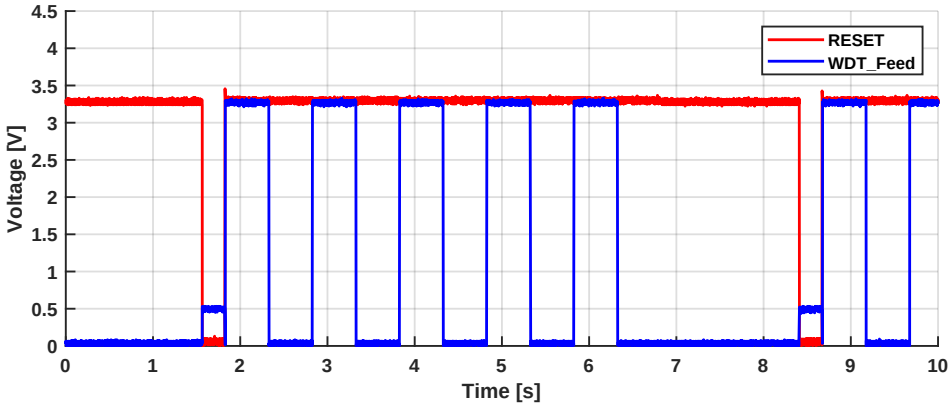


Figure 5.3: Operation of the external watchdog timer, which resets the microcontroller if its feed is stopped.

5.1.4 External memory unit

An external memory unit was also planned for the unfinished iteration COMv1.0, as detailed in [3]. In this design, the memory types used were SRAM as volatile memory and Flash as non-volatile memory. However, this iteration had design problems on this memory unit as these memories were incorrectly addressed. Additionally, as studied in Section 3.2, Flash memories are very susceptible to radiation, which can make them cease operation easily. For this reason, the aforementioned Flash memories were replaced by FeRAM ones in the new COMv1.1 design proposed in this master thesis, keeping the SRAM memory type for volatile memory.

Since a parallel memory unit is used and no strict requirement for its size was defined, the amount of memory used in the COMv1.1 is mostly limited by the physical space defined in the cubesat standard, as parallel memories integrated circuits require a big form factor due to their considerable number of data and address lines. As a result, this subsystem implements a static memory unit with four Cypress CY62177EV30[2] SRAM memories with 4 MB each, plus eight Cypress FM22L16[3] FeRAM memories with 512 kB each, having a total of 16 MB of volatile memory and 4 MB of persistent memory. These memory chips have a 16-bit data bus and may be addressed as bytes or words.

As aforementioned, the memory addressing is achieved by the microcontroller External Memory Controller, which can handle both static or dynamic parallel memory. Using static memory like the ones chosen, this controller provides:

- The **data** signals, with 32 signals *D[0..31]*, addressing up to 32-bit double words of data.
- The **address** signals, with up to 24 signals *A[0..23]*, that provide a 16 MB addressing range.
- The **chip-select** active low signals, with up to 4 signals *CS[0..3]*, which combined with the address bus allow up to 64 MB of total addressing range.

[2] https://www.cypress.com/file/44341/download - Accessed on 02/12/2017
[3] https://www.cypress.com/file/136476/download - Accessed on 02/12/2017

- The **output enable** active low signal *OE*, used to read data from the memory.
- The **write enable** active low signal *WE*, used to write data into the memory.
- The **byte-lane select** active low signals *BLS[0..3]*, which determine if a 8-bit, 16-bit or 32-bit addressing is done inside a double word, whose logic is described in Table 5.2.

Table 5.2: Logic table for the ExtMC addressing type, using the Byte Lane Select signals.

BLS0	BLS1	BLS2	BLS3	Data Transferred	Adressing Type
0	0	0	0	D_0 - D_{31}	32-bit (double word)
0	0	1	1	D_0 - D_{15}	16-bit (word)
1	1	0	0	D_{16} - D_{31}	16-bit (word)
0	1	1	1	D_0 - D_7	8-bit (byte)
1	0	1	1	D_8 - D_{15}	8-bit (byte)
1	1	0	1	D_{16} - D_{23}	8-bit (byte)
1	1	1	0	D_{24} - D_{31}	8-bit (byte)
Note: Other BLS combinations are invalid and the ExtMC does not output them.					

Although the memories chosen allow byte and word addressing, the MCU also provides double word addressing through the use of the Byte Lane Select (BLS) signals. For this to happen, the 16-bit memories must be addressed in pairs as depicted in Figure 5.4, where the lower BLS signals and the lower 16 data lines connect to one memory, and the upper BLS signals and the upper 16 data lines connect to the other memory. Additionally, the lower A0 and A1 addressing signals are no longer required since the BLS signals replace them. For example, as shown in Figure 5.4, for a 4 MB memory with a 20-bit address bus, signals A2 to A22 from the ExtMC are used (and not signals A0 to A20). Finally, the CS, WE and OE are connected in the same way to both memories [14].

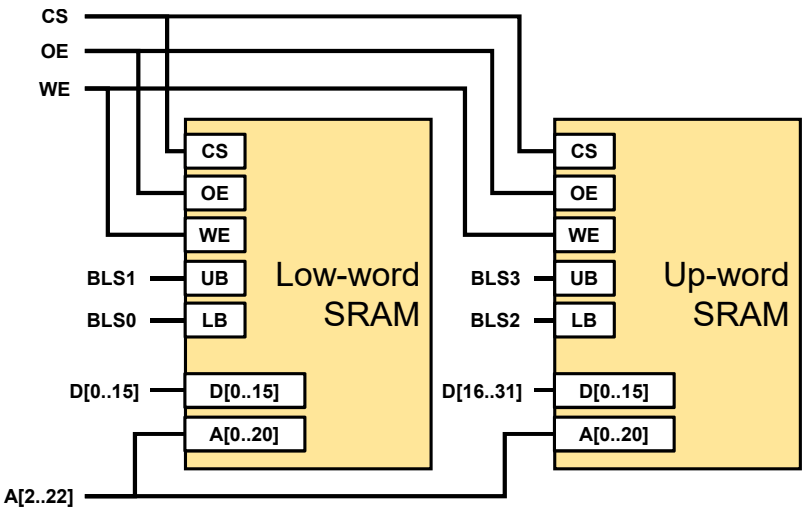


Figure 5.4: Use of the ExtMC for a 32-bit memory bank interfaced with two 16-bit memories [adapted from [14]].

This new requirement of addressing pairs of memory simultaneously to achieve a double word addressing leaves six pairs of memory to be addressed separately: two SRAM pairs plus four FeRAM pairs. Since only four CS signals are provided by the ExtMC, these signals can not be directly divided between these six pairs of memory. Therefore, an external decoder is considered to select which pair of memory is addressed by the microcontroller at a certain address. The solution chosen is the Texas Instruments SN54HC139-SP[4], a dual 2-line to 4-line decoder with a special IC package qualified for space usage. Both decoders have a separate enable line to which two CS signals from the ExtMC are respectively connected, as shown in Figure 5.5. The decoder logic is also achieved using specific address signals from this memory controller so that the memory unit may be addressed continuously, achieving the decoding logic detailed in Table 5.3. Please note that the CS signals on the memories and on the decoder output are both active low signals, being connected to each other.

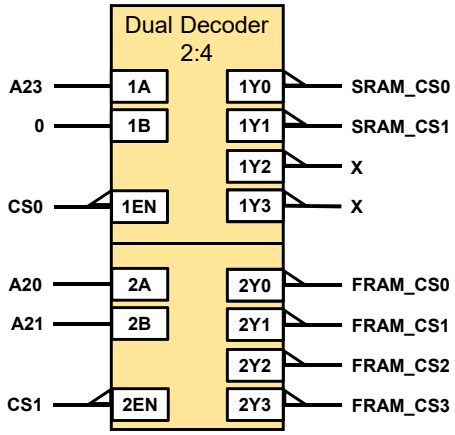


Figure 5.5: ExtMC connection to the COMv1.1 memory decoding unit to address six different memory pairs.

Table 5.3: Decoding logic to address six different memory pairs in the COMv1.1 using the ExtMC.

CS1	CS0	A23	A20	A21	Active Signal	Memory Pair Selected
1	0	0	X	X	*SRAM_CS0*	SRAM1 + SRAM2
1	0	1	X	X	*SRAM_CS1*	SRAM3 + SRAM4
0	1	X	0	0	*FRAM_CS0*	FRAM1 + FRAM2
0	1	X	0	1	*FRAM_CS1*	FRAM3 + FRAM4
0	1	X	1	0	*FRAM_CS2*	FRAM5 + FRAM6
0	1	X	1	1	*FRAM_CS3*	FRAM7 + FRAM8

To confirm the correct operation of this decoder, a simple access test is performed, with write and read operations to addresses correspondent to each memory pair. During these operations, the CS lines of each memory pair are measured plus the ExtMC address and CS signals that provide the decoding logic from Table 5.3.

[4] https://www.ti.com/lit/ds/symlink/sn54hc139-sp.pdf - Accessed on 12/12/2017

As shown in Figure 5.6 and Figure 5.7, the R0 and R1 chip selects are asserted according to the decoding logic implemented with both CS0 and A23 signals, meaning that each SRAM memory pair is addressed correctly. Likewise, the F0 to F3 chip selects are also asserted correctly considering the CS1, A20 and A21 signals, meaning that FeRAM memory pairs are correctly addressed.

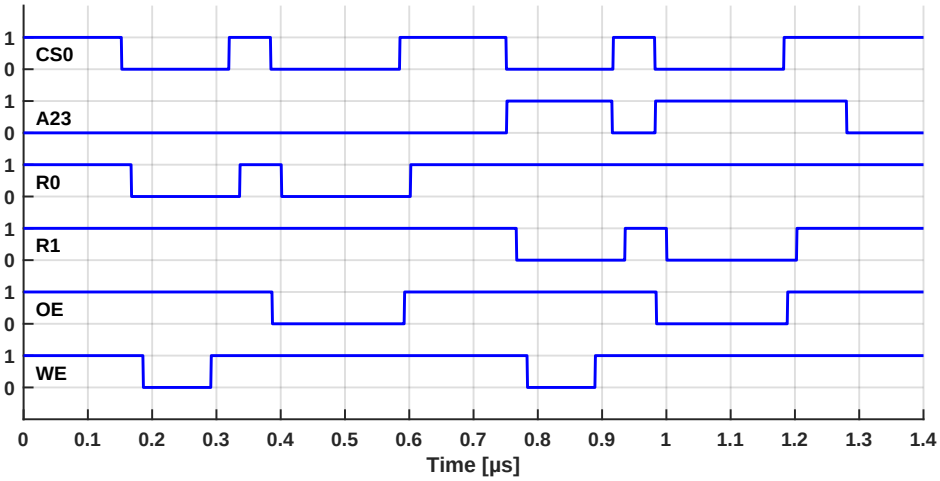


Figure 5.6: Decoding of SRAM memories, where **R0** and **R1** signals are asserted low according to Table 5.3.

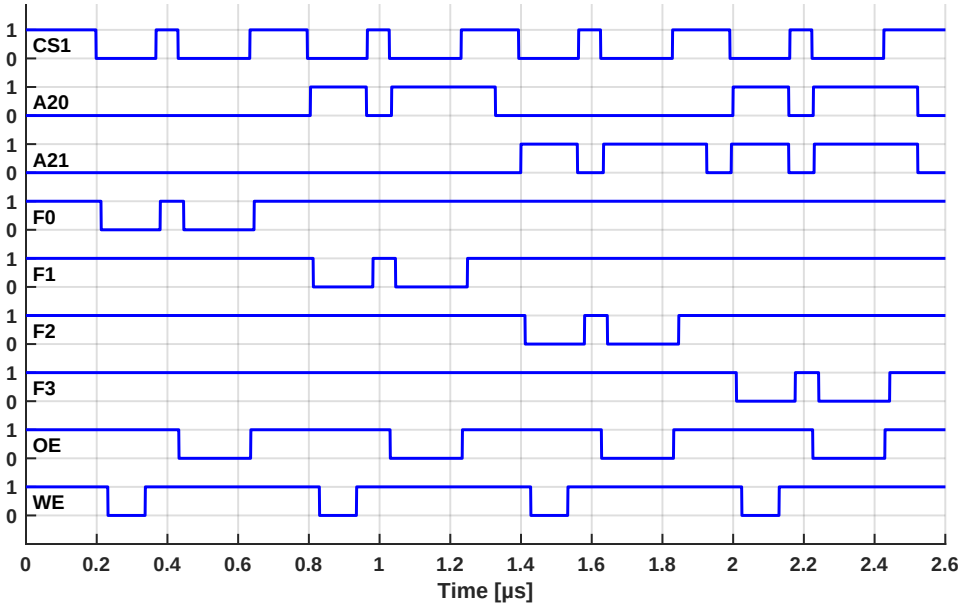


Figure 5.7: Decoding of FeRAM memories, where **F0** to **F3** signals are asserted low according to Table 5.3.

Furthermore, the memory access times can be measured from both OE and WE signals, getting a write access time of 105 ns and a read access time of 205 ns. These times depend on the ExtMC configuration done on the microcontroller startup sequence, which is configured for the fastest memory access times possible. The validation and performance tests of this memory unit are done in Chapter 6.

5.1.5 Radiation protections for the external memory unit

With the addition of the aforementioned decoder, the microcontroller ExtMC became fully operational for the six memory pairs. However, because external memory units are likely to be the major point of failure on cubesat data handling subsystems due to their sensibility to radiation, a radiation mitigation technique relying on memory current sensing and power cut-off was also thought for the COMv1.0, where the radiation protection would rely on a single current sensor to measure the consumption of the whole memory unit, plus two switches to cut off the power supply separately from each of both memory types, namely SRAM and Flash. However, in the new design implemented in the COMv1.1, two current sensors for each memory type and a switch per memory pair are used, as explained further and shown in Figure 5.8.

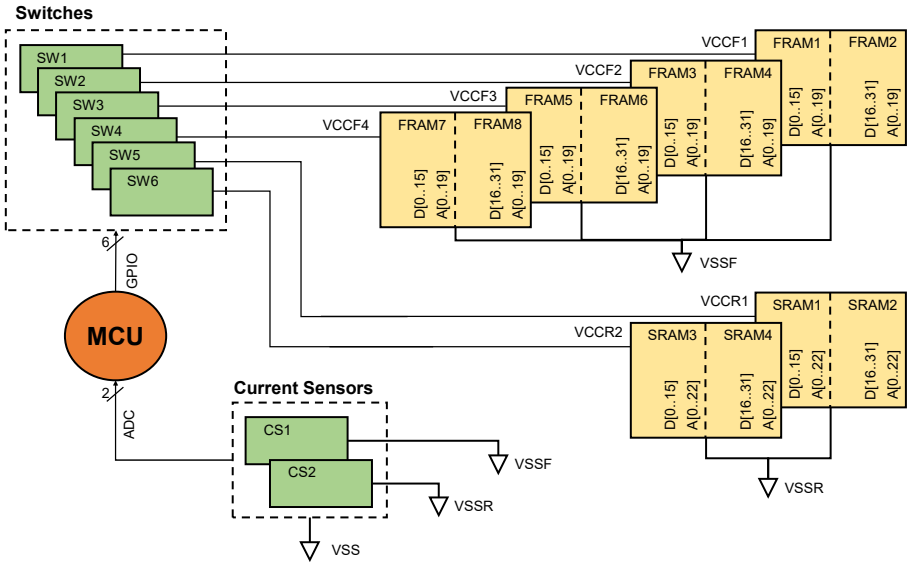


Figure 5.8: Block diagram of the COMv1.1 radiation protections implemented in its memory unit.

One current sensor is used for each of both memory types chosen, namely SRAM and FeRAM. In the occurrence of a SEE, for example, a Single Event Latchup (SEL) causing a high-current condition in these memories, the processing unit can detect this occurrence through two ADC channels, each one connected to the correspondent current sensor of each memory type. Since two current sensors are used, a direct correlation between the detection of a high-current in one sensor and the correspondent memory type may be done, making it easier for the processing unit to act accordingly.

After detecting this high current condition, the microcontroller may cut off the power from every pair of SRAM and FeRAM memories through six different switches, which are controlled by six independent microcontroller GPIO lines. Then, the microcontroller gradually turns them on separately, rechecking the current sensors. If the power cut-off has been done fast enough so that the spontaneous SEE did not caused a permanent damage to a memory cell, the current sensors shall return to a low current state.

However, if the damage is permanent and high currents are continuously detected, the memory housekeeping software will learn that there is a problem in a memory pair and turn it off permanently, until the Ground Segment commands to retry its use. Since SEEs may put devices operating out of operation, this radiation mitigation method reduces the chance of a memory component to be permanently damaged due to a high-current condition. Nonetheless, since six switches are used instead of only two, these memory pairs can be shutdown gradually if only one memory within a pair fails its operation permanently, instead of disabling all the memories of a certain type like thought for the former COMv1.0. This design improvement increases the memory unit lifetime.

The hardware implemented in this radiation protection as a switch and as a current sensor is shown in Figure 5.9. On its left, a single P-channel power MOSFET (BSS315P) is used for each memory pair as its switch. The gate of this transistor is directly driven by the correspondent GPIO enable line from the microcontroller, which controls if the 3.3 V subsystem power supply is provided to the correspondent memory pair.

Additionally, a pull-up resistor R_1 connected between the gate and source is added to keep this switch opened while the microcontroller enable line is at a high impedance state, mainly during the startup sequence. Without this resistor R_1, the transistor Q_1 could conduct during the startup sequence, meaning that the several memory pairs would be turned on simultaneously after the subsystem power-on. Due to the memory unit decoupling capacitors, a high current surge would occur, meaning that the EPS subsystem, which monitors the current consumption of every subsystem, would shutdown the COMv1.1 subsystem after detecting this current spike. This process would repeat itself, meaning that the COMv1.1 would never start operation. Furthermore, to attenuate this current surge even with the pull-up resistor, the memory pairs are turned on sequentially and not simultaneously by the boot-up sequence software, with a predefined delay between each power-on.

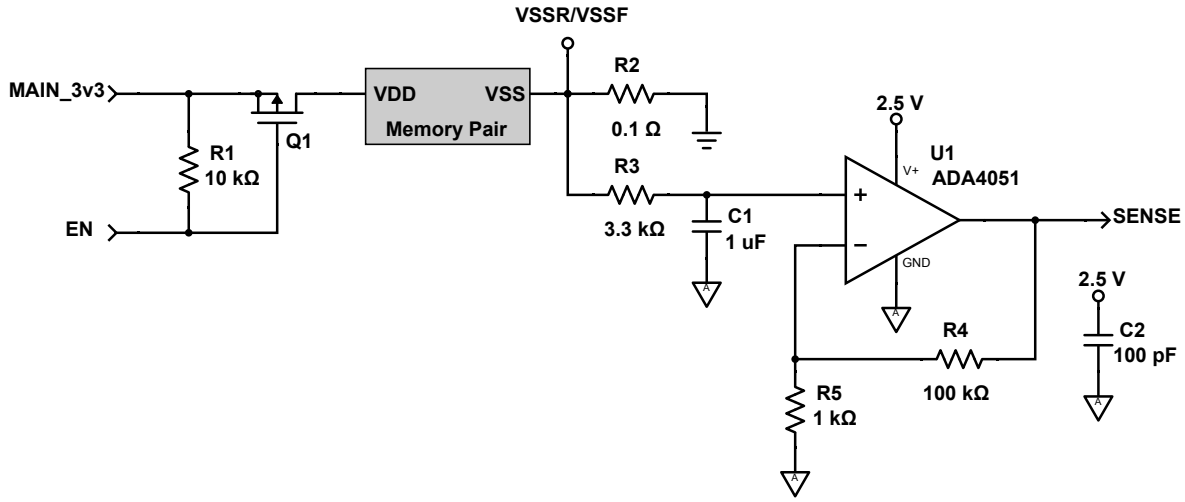


Figure 5.9: COMv1.1 power cut-off switch and current sensor.

On the right part of Figure 5.9, a current sensor made by a shunt resistor R_2 and an operational amplifier (ADA4051) in a non-inverting configuration is implemented for each memory type. In this configuration, the output of the operational amplifier is connected to an ADC channel, whose voltage is given by

$$V_{SENSE} = I_{R_2} \times R_2 \left(1 + \frac{R_4}{R_5}\right) \tag{5.1}$$

where I_{R2} is the current flowing from the memories through the shunt resistor R_2 to the main digital ground plane. Therefore, a current sensor gain

$$G = R_2 \left(1 + \frac{R_4}{R_5}\right) \tag{5.2}$$

is achieved. Since the power consumption of each memory type differ, different gains are used for each current sensor, considering that the maximum output voltage is achieved near the latch-up current specified in the datasheets of each memory selected. Additionally, a low-pass filter is added to the input of the operational amplifier, so that the high-frequency current spikes in the memories are filtered before being acquired by the ADC, designed with a cut-off frequency

$$f_c = \frac{1}{2\pi R_3 C_1} \tag{5.3}$$

of 50 Hz. Finally, a voltage reference REF3325 from Texas Instruments, powered by the 3.3 V subsystem power supply, is also used to power the microcontroller ADC channels and the current sensors, providing a stable voltage reference of 2.5 V. Considering this voltage as the maximum current sensor output voltage, the SRAM current sensor is designed with a gain G of 10 V/A and the FeRAM sensor with a gain G of 20 V/A, since the latch-up currents of these memories are 250 mA and 125 mA, respectively.

In order to validate the design of the aforementioned current sensors, different currents are directly applied to the correspondent shunt resistors within the current range expected. These currents, which are measured with an ammeter, are then compared to the currents obtained by the microcontroller, resorting to its ADC channels to measure the output voltage of the current sensors. Since a linear characteristic between the output voltage and the shunt current is expected for these current sensors, linear regressions are computed with the measured values. These linear regressions are useful for a further software calibration of the current sensors, where a direct proportionality between the current flowing through the shunt resistors and the measured ADC values is obtained.

The validation of the COMv1.1 current sensors is shown in Figure 5.10. The desired current sensors gains of 10 V/A and 20 V/A are approximately achieved for both SRAM and FeRAM memory types, with the expected linear characteristic. However, an unexpected difference of 20 mA between the real current and the measured current is obtained on both current sensors.

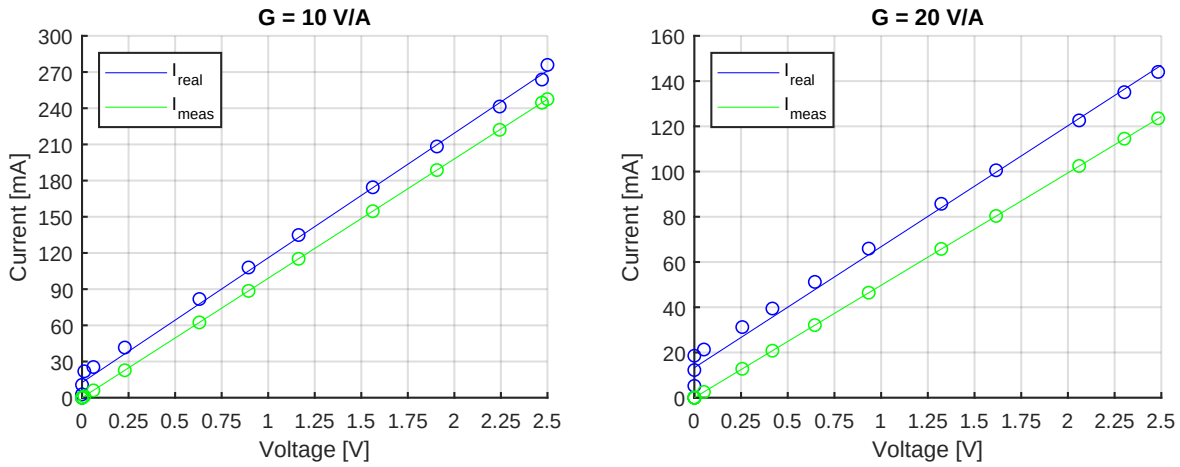


Figure 5.10: Operation of the COMv1.1 current sensors, with the real and measured currents, obtained by the MCU.

Recalling the circuit from Figure 5.9, this difference is confirmed by an offset voltage of -2 mV measured from the positive to the negative inputs of the operational amplifier used, when no current is following through the shunt resistor. Since the maximum input offset voltage defined on the datasheet of this operational amplifier is 17 μV, the obtained offset voltage may be set due to the designed low-pass filter influence on the inputs of the operational amplifier, although this hypothesis was not confirmed. Therefore, operational amplifier only operates for currents higher than 20 mA, where this offset voltage becomes positive due to a voltage drop greater than 2 mV on the 0.1 Ω shunt resistor.

Although this behaviour is not the one expected, the typical current consumptions for both memories used when in operation range between 30 mA and 40 mA, meaning that an abnormal current consumption higher than these values is detected by the microcontroller using this current sensor. Additionally, the detected current offset is taken into account on the software calibration of the current sensors, adding the 20 mA offset current to the current values obtained through the ADC channels. Therefore, despite the detected offset, these current sensors are validated for the current measurement of the COMv1.1 memory unit.

5.1.6 Communication peripherals

Following one of the options provided by the cubesat standard, the I^2C interface is chosen for the satellite main communication bus. However, the use of a single interface for all communications between the subsystems of the spacecraft is risky, specially considering the space environment whose radiation may affect the bus operation. For that reason, the ISTnanosat team has decided to use a second I^2C interface as a redundant one.

Additionally to this bus redundancy, the LTC4303[5] bus buffer from Linear Technology is used on both I²C buses, mitigating a stuck bus situation made by a software or hardware problem. This buffer, whose connections implemented are shown in Figure 5.11, automatically separates the main satellite bus from a microcontroller if any of the I²C data (SDA) or clock (SCL) lines becomes low for more than 30 ms. After breaking connection, 16 clock pulses are generated on the microcontroller SCL line in an attempt to free the bus, being the connection automatically reestablished when the bus becomes free and ready for communication. Additionally, this IC provides a READY signal, which indicates if the connection between the MCU and the main system bus is correctly established, and an ENABLE signal, which can be used to break this connection intentionally.

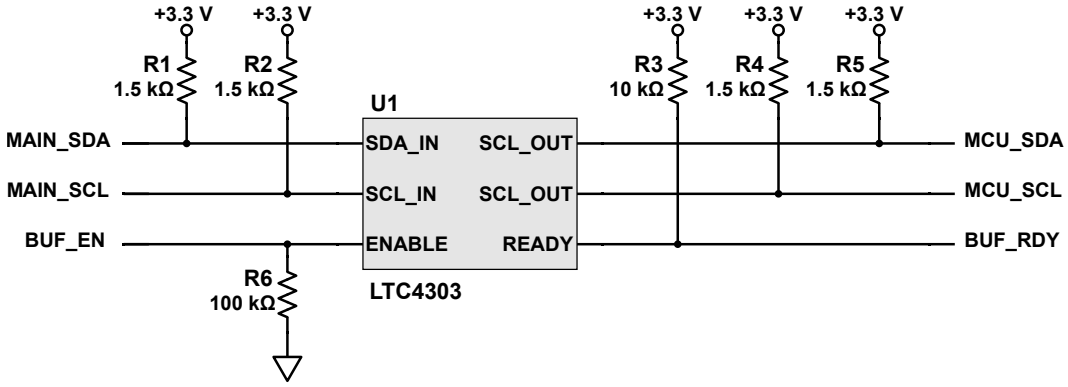


Figure 5.11: Use of the LTC4303 I²C bus buffer to decouple the system I²C bus from the microcontroller bus.

Since the main objective of this subsystem is to process the satellite communication stack, a dedicated communication interface is implemented between the TT&C subsystem responsible for the space-link. This additional interface separates the data related with this communication stack from the main system buses, preventing them from becoming flooded. A SPI interface was chosen by the ISTnanosat team for this dedicated communication interface, with the TT&C as the master node and the COMv1.1 as the slave node, since the DSP used in the TT&C subsystem does not support a SPI interface as a slave node, operating only as a master node. Additionally, to inform the TT&C that new data to be transmitted to the GS exists, an additional Request to Send (RTS) signal is used by the COM subsystem so that the former knows when a SPI transfer is necessary, which is based on a GPIO line. Although the SPI protocol is full-duplex, the ISTnanosat team has decided to use this interface in a half-duplex way, where only one node can transmit data at a time.

The operation of this interface is depicted in Figure 5.12, where two bytes are sent from the TT&C to the COM subsystem through the MOSI line, being afterwards resent in the opposite direction through its MISO line, after the assertion of the RTS line. Since the TT&C is the master node, both SPI transfers are initialised by this subsystem, with an operating frequency of 500 kHz.

[5] https://www.analog.com/media/en/technical-documentation/data-sheets/4303fb.pdf - Accessed on 12/01/2018

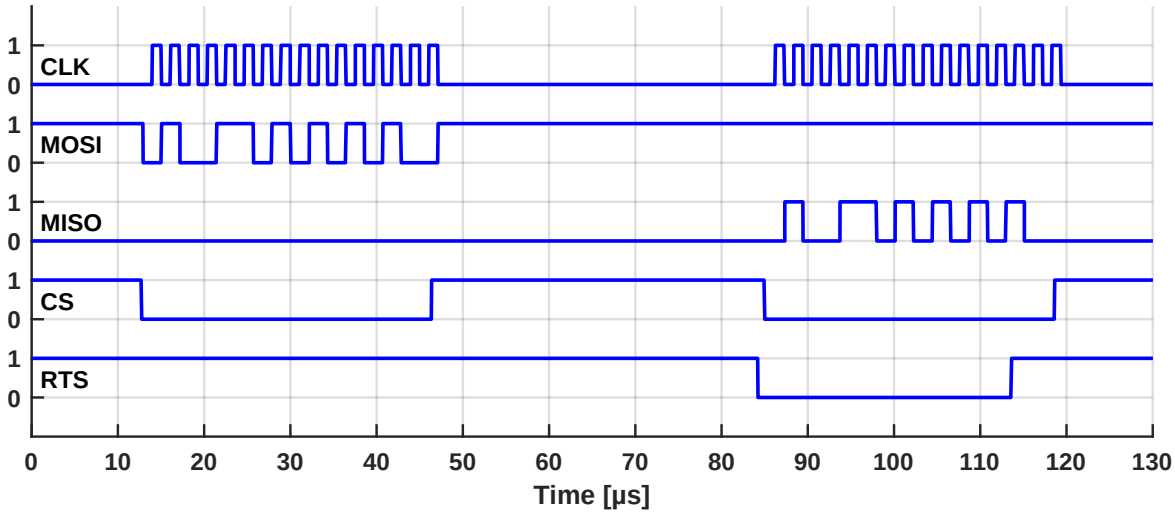


Figure 5.12: Operation of the dedicated SPI+RTS interface, with two bytes being transferred on both directions.

Finally, a GPIO pin is used to provide a heartbeat signal to the OBC housekeeper subsystem in the satellite and for the beacon signal sent to the GS, which toggles its state within a heartbeat period.

Regarding the debug of the subsystem, a programming SWD interface is used for programming and debugging the microcontroller. Additionally to an internal connector used when programming the subsystem outside the satellite, another connection of this programming interface is done to the satellite PC/104 bus, which then connects to the satellite Service Port[6] on the Payload subsystem.

5.1.7 COMv1.1 subsystem flight model board

A final COMv1.1 board has been designed with the modules defined in the previous sections. This PCB, with six layers and with a size of 93.0 mm x 86.4 mm, respects the cubesat standard and will be integrated in the ISTsat-1 satellite. In Figure 5.13 is shown how the different modules populate the designed board.

[6] When the satellite is fully integrated, the solar panels enclose the several subsystems in the cubesat and it becomes impossible to access them. Therefore, to provide the programming interfaces for the several subsystems when the satellite is fully integrated, a Service Port board is plugged on the bottom face of the satellite into the Payload subsystem, whose signals connect directly to the PC/104 bus.

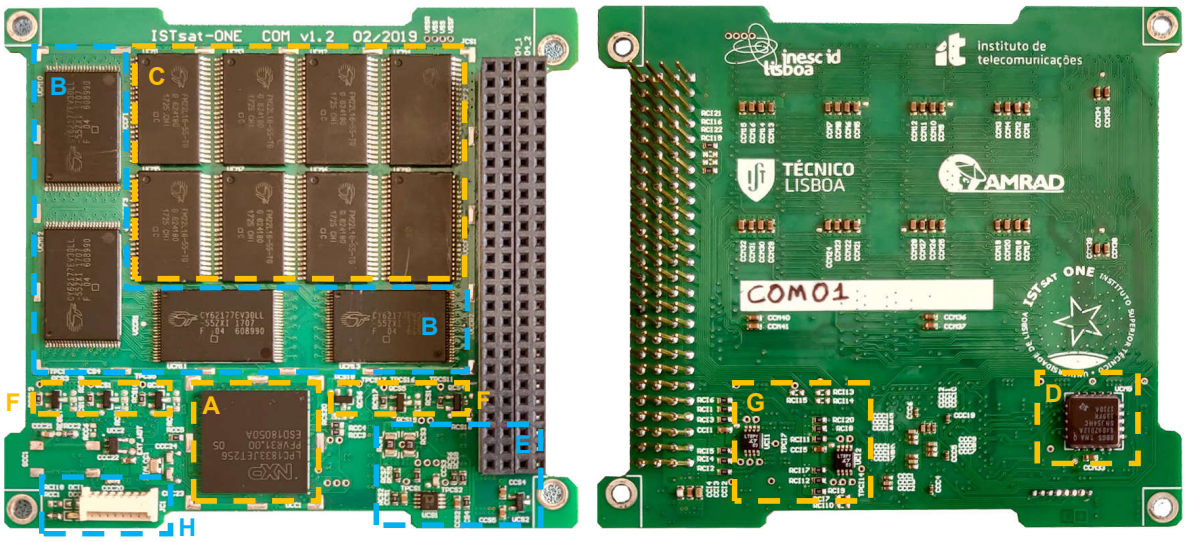


Figure 5.13: Top (left) and bottom (right) views of the COMv1.1 final design board, highlighting the main modules.

A: Microcontroller
B: SRAM memory bank
C: FeRAM memory bank
D: Decoder for the memory unit
E: Memory unit current sensors
F: Memory unit switches
G: I^2C interface buffers
H: Debug connector

5.2 COMv2 Hardware project

To support the explanation of the COMv2 hardware modules, the block diagram from Figure 5.14 represents an overview of the hardware architecture of this subsystem, where all components and relevant interconnections are shown. In the following sections, the several power and digital modules of this subsystem are detailed, finishing with its power distribution. The COMv2 hardware architecture is mainly composed of:

- The microcontroller unit, whose connection to the subsystem modules and peripherals is done.
- An external serial memory unit of non-volatile memory.
- Two IMU sensors and conditioning circuits for sun sensors used in the attitude determination.
- Driving circuits of magnetorquers used for the attitude control.
- Peripherals chosen for the communication with the several satellite subsystems.
- Radiation mitigation circuits regarding the operation of the microcontroller and the several subsystem modules.
- A programming interface plus an USB and SD Card interfaces used for debug.

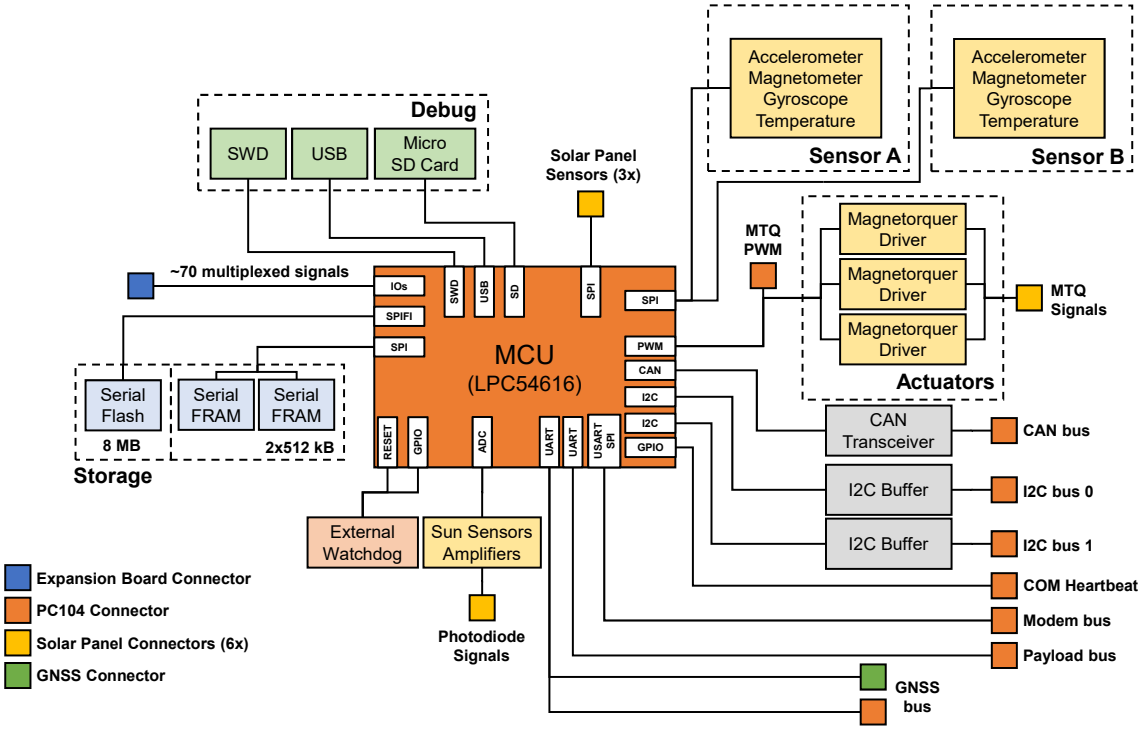


Figure 5.14: Block diagram of the COMv2 hardware architecture.

5.2.1 High-power processing unit selection

Because the COMv2 subsystem also requires a high-power processing unit to meet its communications, command and data handling and attitude determination and control requirements, the first thing to be considered is the use of a powerful processing unit with low power consumption. In Table 5.4 are summarised the requirements set for this processing unit.

Table 5.4: COMv2 microcontroller requirements.

Characteristic	Requirement
Internal Program Memory	$>$ 512 kB
Internal Data Memory	$>$ 192 kB
Power consumption (without peripherals)	$<$ 150 μA/MHz
Serial Communication Peripherals	I^{2}C, SPI, UART, USART, CAN
External memory management unit	Static memory, SD Card
Pulse Width Modulation (for ADCS)	Yes
Analog-to-digital converter	Yes
Low-power modes	Yes

For the selection of this high power processing unit, three well-known microcontroller manufacturers were considered:

- Texas Instruments, studying the MSP432 family, which is divided into two product lines: the *MSP432P4* and the *MSP432E4*.
- STMicroelectronics, studying the STM32F4 family composed of two main product lines: the *access* and the *foundation* lines.
- NXP Semiconductors, studying the *LPC546XX* and the *LPC540XX* product lines from the LPC54XXX family.

The requirements set in Table 5.4 were considered during the study of the aforementioned microcontroller families, whose correspondent specifications are summarised in Table 5.5.

Table 5.5: Comparison of available processing unit product lines for the COMv2 subsystem.

Feature	TI - MSP432		STM - STM32F4		NXP - LPC54XXX	
	MSP432P4	MSP432E4	Access	Foundation	LPC546XX	LPC540XX
Core	ARM Cortex-M4 with FPU					
Max. Clock Frequency	48 MHz	120 MHz	100 MHz	168 MHz	180 MHz	180 MHz
Program Memory	2 MB	1 MB	1 MB	1 MB	512 kB	None*
Data Memory	256 kB	256 kB	256 kB	256 kB	200 kB	360 kB
Power Consumption (μA/MHz)	100	642	112	238	140	190
Ext. memory interface	No	Yes (32-bit)	Yes (16-bit)		Yes (32-bit)	
Required Peripherals	No CAN	Yes	Yes	Yes	Yes	Yes
ADC	14-bit 1 MS/s	12-bit 2 MS/s	12-bit 2.4 MS/s		12-bit 5 MS/s	
Package	LQFP100	BGA212	LQFP144	LQFP144	LQFP208	LQFP208
Price	12.02 €	17.57 €	9.57 €	11.84 €	7.93 €	7.03 €
*External Flash memory may be used up to 4 MB						

All the studied options are based on an ARM Cortex M4 with a Floating-Point Unit (FPU), which provides a high processing power capability to the system. Nevertheless, different maximum clock frequencies are available for the selected processing units.

Considering the Texas Instruments MSP432P4, this option provides the lowest power consumption with the smallest IC package with 100 pins. However, this product line does not allow the use of external static memory unit and does not provide a CAN peripheral, which are required for COMv2. On the other hand, the MSP432E4 fulfils these requirements but its power consumption in considerably above the maximum tolerated. For these reasons, the MSP432 family is excluded from the processing unit selection panel.

Considering the STM32F4 family, the main difference between its access and foundation product lines lies on the processing unit performance, with a higher operating frequency available for the latter.

However, this product line exceeds the power consumption requirement, consuming near 100 μA/MHz above the limit set. Nevertheless, the access product line fulfils all the requirements set in Table 5.4 and is a valid option for the COMv2 processing unit, providing the required program and data memory, the required memory interface and communication peripherals, with an interesting clock frequency and power consumption.

Finally, the LPC540XX product line from NXP provides the cheapest processing units with the highest data memory. However, this product line requires the use of an external flash memory for program data, resulting on an additional power consumption. Nonetheless, the LPC546XX product line, which fulfils all the requirements set in Table 5.4, may be considered as a valid processing unit for the COMv2. When compared with the first valid option, the STM32F4 access line, the NXP product line offers a higher clock frequency, although its power consumption is slightly higher and its package is bigger, which are also two constraints for this subsystem. Additionally, the available external memory interface is the same ExtMC interface already studied in the COMv1.1, easing the development of an external parallel memory unit in the COMv2 subsystem. Also due to the ISTnanosat team knowledge of this manufacturer, since two MCUs used in ISTsat-1 subsystems come from the LPC family of NXP, the **LPC54616** microcontroller of the LPC546XX family is chosen as the processing unit of the COMv2.

Another important aspect taken into account on this microcontroller choice are its multiplexed pins, which allow them to have different functionalities that are programmed at the processor boot sequence. This is particularly important when considering the daughterboard set by the COMv2 archictecture because the pins connected between the MCU and this daughterboard are also multiplexed, meaning that functionalities like communication protocols, common peripherals or the external memory interface are easily available for different daughterboards with different purposes designed in the future, making this daughterboard extremely modular. For the connection with the COMv2 motherboard, an 80-pin connector is implemented to provide the aforementioned functionalities.

5.2.2 External memory unit

The use of parallel memory for the subsystem internal storage is not considered in the COMv2 motherboard, since this type of memory requires a lot of PCB space for its components to maintain a data storage capacity similar to the one used in the COMv1.1 subsystem, and since a complex routing is required in parallel memory due to its data and address buses. Therefore, this memory type is replaced with serial memory, which has slower addressing times but ensures more data storage capacity per PCB space with less routing complexity.

To assure that the storage of critical data is not compromised, serial FeRAM is considered. Recalling Section 3.2.2, which studies the different memory types to be used in a cubesat, this type of memory is a non-volatile memory more resistant to radiation effects than a normal Flash memory. Two Cypress

CY15B104Q[7] serial FeRAM memories with 512 kB each are considered, since FeRAM memories with bigger capacity are not available in the market. These are accessed through a common SPI interface and the correspondent CS signals.

Allied to this critical data storage, serial Flash memory is considered to increase the data storage capabilities, with one Cypress S25FL064L memory with 8 MB of data capacity. The interface chosen for this memory is SPIFI instead of a normal SPI. This interface, developed by NXP, is a quad SPI Flash interface initially thought for the LPC family microcontrollers without internal Flash memory, enabling its boot and code execution from external Flash memories instead. Nevertheless, this interface may also be easily used on runtime for normal data storage as in any other memory, since the entire Flash content is directly memory mapped by the processing unit, providing fast memory access times comparable with the access times of parallel memory due to the quad SPI bus. Although this type of memory is very susceptible to radiation, one way to reduce these effects is to power off this memory whenever not necessary, benefiting from the persistent memory type to keep the data stored without power.

COMv2 first daughterboard

In the COMv1.1 subsystem, parallel memory is used for the satellite main data storage, where both satellite telemetry and payload data are continuously stored before being sent to the Ground Segment. A design idea for the first COMv2 daughterboard is to implement this storage hardware as well, in addition to the aforementioned FeRAM and Flash memories. This implementation delivers more storage capabilities to the system, which may be used optionally if required by the satellite mission.

Therefore, the first daughterboard project is a memory expansion board, having the same design as the COMv1.1 subsystem memory unit referred in Section 5.1.4. All the memory addressing is also achieved by the same ExtMC static memory controller and external decoder used to select the memory pairs. Additionally, the same radiation mitigation architecture from Section 5.1.5 are considered.

Since this external memory unit is already implemented and studied in the COMv1.1 design, only the validation of this memory expansion daughterboard is studied in this master thesis, with the same test procedures and acceptance done for the COMv1.1, which are described in Chapter 6. Nevertheless, Figure 5.15 summarises the memory unit considered for the COMv2 subsystem, with the internal motherboard memory unit plus an external one provided when using this memory expansion board.

Since the COMv2 has some of its implementations based on the ISTsat-1 OBC subsystem, the hardware architecture of the latter is briefly described in Appendix A. Nevertheless, any similarities between these subsystems are highlighted in the following sections.

[7] https://www.cypress.com/file/209146/download - Accessed on 22/11/2018
[7] https://www.cypress.com/file/316661/download - Accessed on 12/12/2018

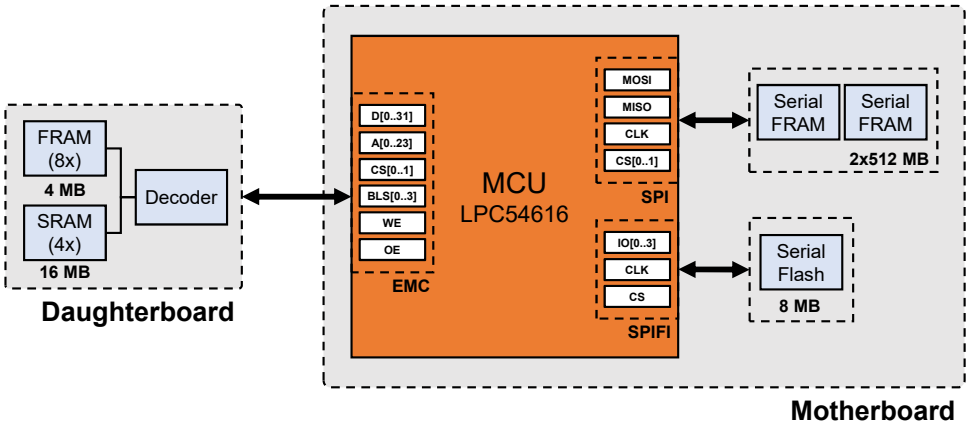


Figure 5.15: COMv2 memory unit block diagram, considering also a memory expansion board as daughterboard.

5.2.3 Subsystem power regulation

In the ISTsat-1, the Electrical Power System is capable of providing and monitoring the power supply of each subsystem independently, meaning that if a short-circuit condition is detected in one subsystem by the EPS and, consequently, its power is cut-off, the power supply of the remaining subsystems will not be affected. This system level architecture is also considered in the COMv2, where a dedicated VCC_COM power supply bus of 3.3 V is provided to this subsystem by the EPS through the PC/104 main satellite bus, as implemented in the COMv1.1.

During an EPS nominal state of operation, this power supply line shall be sufficient for the COMv2 to operate. However, since this subsystem may be the satellite housekeeper, a redundant BACKUP power supply is considered to keep it powered up during an EPS failure, which connects directly to the solar panels through the PC/104 bus and through the EPS subsystem. The BACKUP line voltage may vary between 0 V and 4.65 V depending on the solar exposure, considering the ISTsat-1 case. Therefore, an additional power regulation circuitry is needed to provide a 3.3 V power supply to the COMv2 subsystem from this BACKUP bus.

The circuit used for the power regulation is shown in Figure 5.16, which is also implemented in the ISTsat-1 OBC. The TPS62162[8] 3 V to 17 V step-down converter from Texas Instruments is used following the manufacturer recommendations, with an input capacitor C_1 of 10 μF, an output inductor L_1 of 2.2 nH, an output capacitor C_2 of 22 μF and an output resistor R_4 of 100 kΩ, providing a 3.3 V output power supply with an efficiency of 90 % for an 100 mA current consumption.

However, this circuit must work only when the VCC_COM power supply is not provided by the EPS. For that, two Schottky diodes D_1 and D_2 are connected in series to each of the power supplies, setting a regulated power supply VCC_REG to the subsystem modules of 3.0 V, considering a voltage drop of 0.3 V in these diodes. Additionally, to manage the converter operation, a control circuit connected to its

[8] http://www.ti.com/lit/ds/symlink/tps62162.pdf - Accessed on 20/10/2018

ENABLE pin is used. Whenever the VCC_COM power supply exists, the N-channel MOSFET transistor Q_1 is turned on, keeping ENABLE low. If this power supply is cut off by the EPS, the pull-up resistor R_1 sets ENABLE high and the operation of the step-down converter starts, considering that a sufficient solar panel voltage exists in the BACKUP bus. Finally, two Zener diodes D_3 and D_4 are used to limit voltage spikes on VCC_COM and VCC_REG to 3.6 V, preventing the damage of the subsystem circuitry [10].

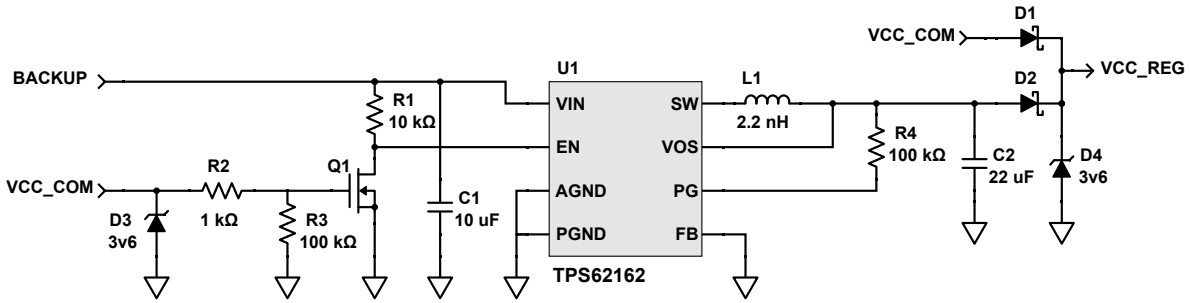


Figure 5.16: Regulation circuit for the BACKUP power supply bus, implemented in the COMv2.

The validation of this circuit is shown in Figure 5.17. From the left graph, a voltage drop of 0.3 V is confirmed whenever VCC_COM is provided with 3.45 V[9], resulting on a regulated power supply VCC_REG of 3.15 V. However, when VCC_COM is cut off, the buck regulator does not start instantaneously, being the VCC_REG voltage provided again after 270 ms and with a voltage of 3 V.

From the right graph of Figure 5.17, the fast transition from the BACKUP power supply back to the VCC_COM bus does not influence the regulated VCC_REG voltage, as this voltage increases again to 3.15 V, meaning that the subsystem is not temporally shutdown during this transition.

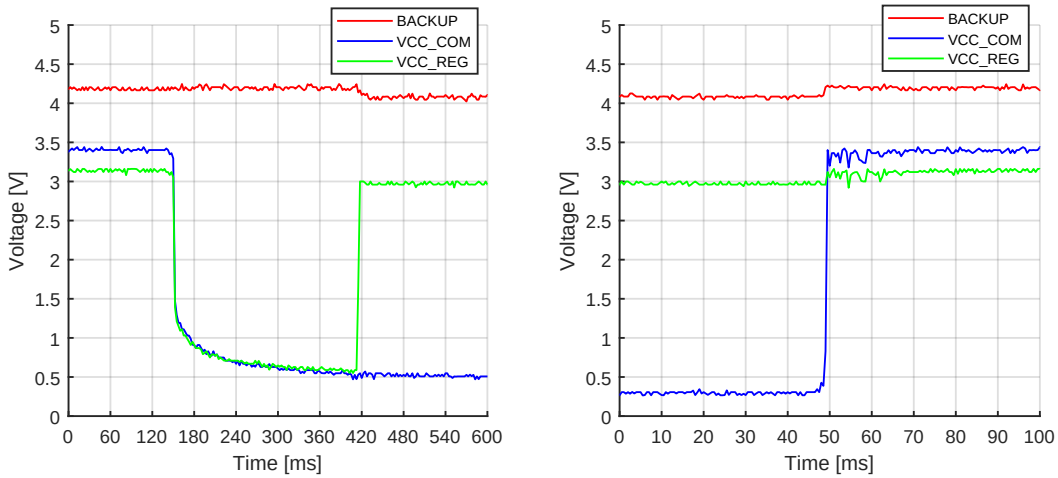


Figure 5.17: BACKUP power supply regulator startup (left) and shutdown (right), according to the VCC_COM bus.

[9] In the design of the ISTsat-1 EPS subsystem, an higher VCC_COM voltage is provided due to a power supply requirement of the TT&C subsystem, which requires a regulated voltage of 3.45 V instead of 3.3 V. This regulated power supply is shared with the COM subsystem, although both subsystems may be cut off separately.

Other power lines considered that may not be used during flight are the main 3.3 V and 5 V provided on the PC/104 main system bus. Additionally, an USB 5 V line is also considered. These options are also connected to the aforementioned regulator, which may be selected through the use of jumpers throughout the subsystem debug as a replacement for the BACKUP bus.

5.2.4 Power cut-off and current measurement circuits

Being the subsystem regulated power line VCC_REG set to 3.0 V, the next step is to detail the radiation mitigation implementations used to verify the power consumption of the several subsystem modules. For that, a power cut-off switch and a current sensing circuit are implemented, as shown in Figure 5.18.

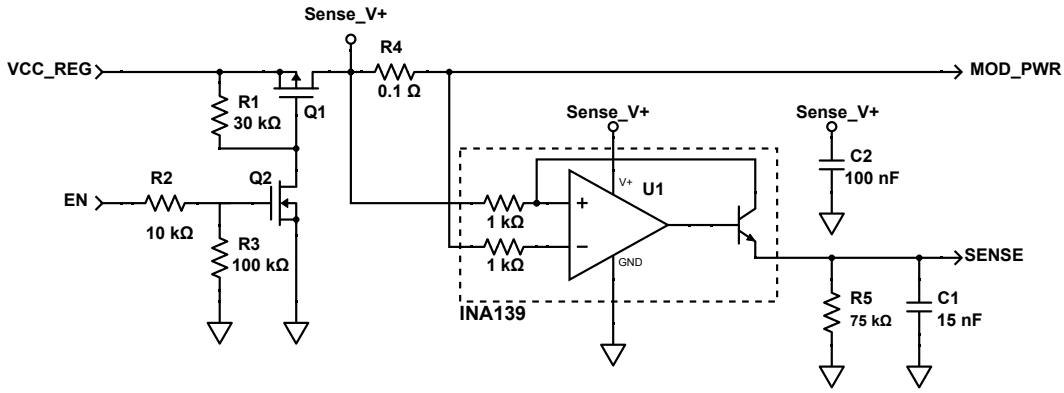


Figure 5.18: Power cut-off switch and current measurement circuit for the COMv2 modules.

On the left, a power cut-off switch is implemented with a P-channel MOSFET (PMV65XP) in series with the regulated power supply VCC_REG, plus a N-channel MOSFET (PMV30UN) that controls the former transistor through a GPIO line provided by the microcontroller. This design makes the power supply of each module only controllable by the MCU. Additionally, these modules stay without power at its startup due to the pull-up resistor R_1 and pull-down resistor R_3, which force the power cut-off until the microcontroller ENABLE line leaves an high impedance state at its boot sequence.

On the right side of Figure 5.18, a current sensing circuit made by a shunt resistor R_4 and a current shunt monitor INA139[10] from Texas Instruments is implemented. This integrated circuit, internally made by an operational amplifier and a bipolar transistor, converts a differential input voltage to a current output, which is converted back to a voltage with an external load resistor R_5. The output of this circuit is connected to a ADC channel, whose voltage is, according to its datasheet, given by

$$V_{SENSE} = 0.001 \times I_{R_4} R_4 R_5, \tag{5.4}$$

where I_{R4} is the current flowing through the shunt resistor R_4, whose resistance is fixed to 0.1 Ω in

[10] http://www.ti.com/lit/ds/symlink/ina169.pdf - Accessed on 21/11/2018

this design. Therefore, the different gains of each current sensor are directly controlled by the output resistor R_5. However, due to a voltage drop of 0.7 V on the internal bipolar transistor (value defined by the manufacturer), this output voltage is limited to 2.3 V, considering that the INA139 is powered with 3.0 V provided by VCC_REG. This design condition also influences the choice of the R_5 resistance, as the output voltage range of the current sensing circuit becomes lower than the expected 3.0 V provided by VCC_REG.

Additionally, a low-pass filter is added to the output of this circuit with the output capacitor C_1, so that the high-frequency current spikes in the modules are filtered before being acquired by the ADC. This filter is designed with a cut-off frequency

$$f_c = \frac{1}{2\pi R_5 C_1} \tag{5.5}$$

of 150 Hz. Since the different modules composing the COMv2 have different current consumptions, different values of R_5 and C_1 are chosen for each case. Nevertheless, only three different gains are used for the several current sensors implemented for this subsystem, which are further detailed in the power distribution analysis from Section 5.2.11. The values obtained for these current sensors are shown in Table 5.6, considering different maximum currents of 1 A, 300 mA and 100 mA.

Table 5.6: Values of R_5 and C_1 chosen for the different COMv2 current sensors.

Maximum current (mA)	Desired gain (V/A)	Desired cut-off frequency (Hz)	R_5 (kΩ)	C_1 (nF)	**Selected R_5 (kΩ)**	**Selected C_1 (nF)**
1000	2.3	150	23	46.1	22	47
300	7.7	150	77	13.8	75	15
100	23	150	230	4.6	220	4.7

For example, considering a current sensor whose maximum current consumption is 1 A, the desired shunt current to output voltage gain is 2.3 V/A, considering the aforementioned maximum output voltage obtained with this current sensor typology. For that, an output resistor

$$R_5 = \frac{V_{SENSE}}{0.001 \times I_{R_4} R_4} \tag{5.6}$$

of 23 Ω is required, where I_{R4} is the 1 A current flowing through the 0.1 Ω shunt resistor R_4. Being the output resistor defined, an output capacitor

$$C_1 = \frac{1}{2\pi R_5 f_c} \tag{5.7}$$

of 46.1 nF is required, considering a cut-off frequency of 150 Hz. Analysing typical values for capacitors and resistors available in the marker, an output resistor and capacitor of 22 kΩ and 47 nF are used.

In order to study the operation of these current sensors, the same procedure done in the COMv1.1 current sensors is repeated, where different currents are directly applied to the correspondent shunt resistors within the current range expected, obtaining the real current and measured current from Figure 5.19 for the different gains implemented.

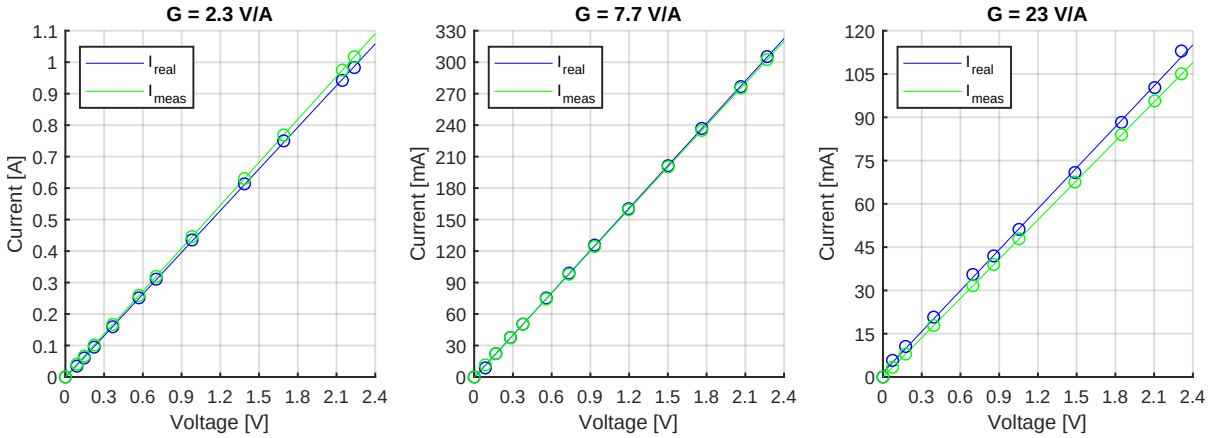


Figure 5.19: Operation of the COMv2 current sensors, with the real and measured currents, obtained by the MCU.

A linear characteristic is achieved for the three different gains, although a small deviation exists between the slopes of the real and measured currents, which can be easily solved through a software calibration, achieving a direct proportionality between the current flowing through the shunt resistors and the measured ADC values.

5.2.5 Capacitor bank for the time reference

Being the satellite housekeeper, one requirement for the COMv2 is to keep track of a time reference not only during the normal satellite operation, but also during a power shutdown, for at least a maximum of 90 minutes[11] correspondent to an orbit. This requirement is imposed so that this time reference is not lost if this subsystem is powered off by the EPS, even during an eclipse where the BACKUP power line is expected to be unavailable.

This time reference is maintained by the internal Real-Time Clock (RTC) of the microcontroller, which is updated by a Ground Segment command. The microcontroller selected for the COMv2 subsystem provides a dedicated V_{RTC} pin to power its RTC, which is not connected directly to the regulated VCC_REG power line as the rest of the microcontroller. Instead, a capacitor bank and a diode with a low reverse current (BAV170HMFH) connected to the V_{RTC} pin is considered, as shown in Figure 5.20. Although this implementation is also used in the OBC subsystem, an additional external RTC is required for this subsystem, since the chosen OBC microcontroller does not provide a dedicated V_{RTC} pin.

[11] An orbit time of approximately 90 minutes is considered because this is also the orbit time of the ISS from which cubesats like the ISTsat-1 are deployed into space.

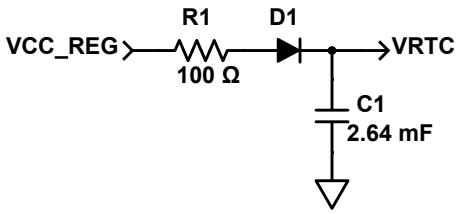


Figure 5.20: Capacitor bank circuit used to keep the RTC time reference over an orbit.

According to the LPC54616 datasheet, the typical current consumption of the internal RTC when the microcontroller is in a deep power-down mode is 340 nA with a V_{RTC} of 3.0 V. To calculate the necessary capacitance for the capacitor bank, this current consumption is assumed also when only the V_{RTC} pin is powered and the rest of the microcontroller is shutdown. Additionally, the minimum voltage $V_{RTC\,min}$ required for RTC to operate is 1.71 V [15].

Considering a capacitor bank discharge time of 90 minutes correspondent to an orbit and that the V_{RTC} is initially powered by 2.5 V (due to voltage drop on the diode D_1 of 0.5 V), its capacitance given by

$$C = \left(\frac{I_{RTC}}{V_{RTC_{max}} - V_{RTC_{min}}} \right) t_{orbit} \tag{5.8}$$

must be greater than 2.32 mF to meet this requirement, considering a constant current discharge.

Analysing possible ceramic and tantalum capacitors available in the market, eight 330 uF tantalum capacitors are considered, providing a parallel capacitor bank of 2640 uF. Additionally, to limit the in-rush current on the capacitor bank from VCC_REG, a resistor of 100 Ω in series with the aforementioned diode is used.

The obtained capacitor bank operation is shown in Figure 5.21, whose discharge is much faster then expected. This means that current consumption assumption of 340 nA characteristic of the microcontroller deep power-down mode is not valid, as the current consumption

$$I_{RTC} = \frac{C}{t_{orbit}} \times (V_{RTC_{max}} - V_{RTC_{min}}) \tag{5.9}$$

proofs to be approximately 16 μA, considering an approximate discharge time of 130 s between the initial voltage of 2.5 V and the minimum RTC operating voltage of 1.71 V, as shown in Figure 5.21.

This analysis proofs that this simpler RTC design is not feasible for the 90 minutes requirement. Even if a bank with a bigger capacitance was considered, the time and current consumption required for recharging this bank after a power down would be considerable and not desirable for a small satellite. This means that the same implementation done on the ISTsat-1 OBC must be considered in the future, where the same capacitor bank typology is used but powering an external RTC integrated circuit instead of connecting to the microcontroller, which is accessed by a SPI interface to retrieve the time reference during the boot-up sequence or on normal operation. This implementation is detailed in Appendix A.

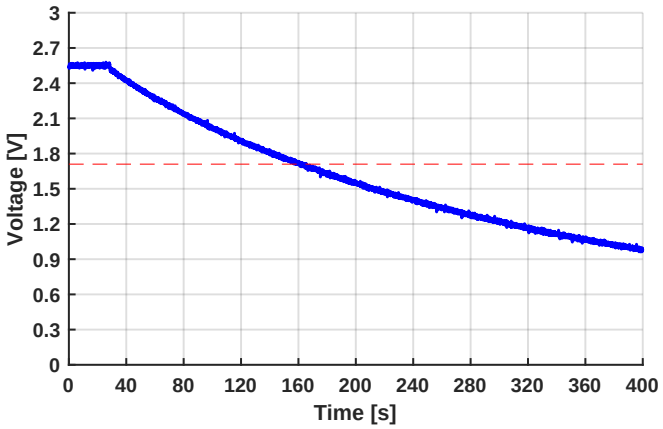


Figure 5.21: Discharge of the COMv2 RTC capacitor bank, with the RTC minimum operating voltage.

5.2.6 Solar panels interface

One implementation done in the ISTsat-1 OBC subsystem is the interface with several modules present on the satellite solar panels, such as solar sensors, magnetorquers and gyroscopes, which are vital for the attitude determination and control. The solar panel boards used in the ISTsat-1, which are bought from EnduroSat[12], have a well-defined interface to these modules with a 12-pin connector. Like implemented in the ISTsat-1 OBC, the COMv2 is designed to support these interfaces, with one connector for each one of the six cubesat planes. The hardware implemented between these modules and the processing unit is detailed in the further attitude determination and control sections.

5.2.7 Attitude determination sensors

The first thing considered concerning the satellite attitude is the sensor selection for its determination, which is based on a 9 DoF[13] system, requiring a gyroscope, an accelerometer and a magnetometer to characterise the satellite orientation.

To determine which sensors are used, several small and low-cost MEMS sensors from Table 5.7, Table 5.8 and Table 5.9 are analysed, being divided in magnetometers, accelerometers and gyroscopes, respectively. However, some of this MEMS options integrate multiple sensors, as seen in the last column of these tables. Additionally, all these sensors can measure the correspondent physical quantity in a 3 axis reference system. The sensors used in the ISTsat-1 OBC subsystem are also shown in the last rows as a reference, whose production was discontinued.

[12] EnduroSat website: https://www.endurosat.com

[13] The Degrees of Freedom (DoF) of a rigid body are the number of independent movements it has. In this case, the translation in a given X,Y,Z axis system (accelerometer), the rotation around any of these three axis (gyroscope) and the pointing direction in that axis system (magnetometer) characterises a satellite with 9 DoF.

Table 5.7: Comparison of magnetometer MEMS sensors.

Device	Bits	Measurement Range (mT)	Sensitivity (nT/LSB)	Spectral Noise Density (nT/sqrt(Hz))	Sampling Rate (Hz)	Supply Current (mA)	Manufacturer	Price (€)	Other Sensors
LSM303C	16	1.6	58	110	80	0.27	STMicro	2.32	acc
LIS2MDL	16	5	150	95	100	0.2	STMicro	1.55	-
ICM-20948	16	4.9	150	NA	100	0.09	InvenSense	6.20	acc/gyr
LSM9DSI	16	1.6	14	NA	80	0.3	STMicro	5.36	acc/gyr
BMX055	16	1.3	300	190	100	0.17	Bosch	6.13	acc/gyr
MPU9250*	16	4.8	600	NA	8	0.3	InvenSense	12.82	acc/gyr
HMC5983*	12	0.8	72	200	220	0.1	Honeywell	3.37	-
NA - Not Available on datasheet. *Used on ISTsat-1 OBC and discontinued.									

Table 5.8: Comparison of accelerometer MEMS sensors.

Device	Bits	Measurement Range (g)	Sensitivity (mg/LSB)	Spectral Noise Density (μg/sqrt(Hz))	Sampling Rate (Hz)	Supply Current (mA)	Manufacturer	Price (€)	Other Sensors
LSM303C	16	2/4/8/16	0.061	140	800	0.18	STMicro	2.32	mag
ICM-20602	16	2/4/8/16	0.061	100	4000	0.321	InvenSense	2.86	gyr
ICM-20948	16	2/4/8/16	0.061	230	4500	0.068	InvenSense	6.20	mag/gyr
LSM9DSI	16	2/4/8/16	0.061	NA	950	0.3	STMicro	5.36	mag/gyr
BMX055	12	2/4/8/16	0.976	150	1000	0.13	Bosch	6.13	mag/gyr
MPU9250*	16	2/4/8/16	0.061	300	4000	0.5	InvenSense	12.82	mag/gyr
NA - Not Available on datasheet. *Used on ISTsat-1 OBC and discontinued.									

Table 5.9: Comparison of gyroscope MEMS sensors.

Device	Bits	Measurement Range (°/s)	Sensitivity ((m°/s)/LSB)	Spectral Noise Density ((m°/s)/sqrt(Hz))	Sampling Rate (Hz)	Supply Current (mA)	Manufact.	Price (€)	Other Sensors
L3GD20H	16	2000	8.75	11	750	5	STMicro	2.32	acc
ICM-20602	16	2000	7.6	4	8000	2.6	STMicro	1.55	-
ICM-20948	16	2000	7.6	15	9000	1.2	InvenSense	6.20	acc/gyr
LSM9DSI	16	2000	8.75	NA	950	4	STMicro	5.36	acc/gyr
BMX055	16	2000	3.81	14	2000	5	Bosch	6.13	acc/gyr
MPU9250*	16	2000	7.6	5	8000	3.2	InvenSense	12.82	acc/gyr
NA - Not Available on datasheet. *Used on ISTsat-1 OBC and discontinued.									

Analysing Table 5.7, the LSM9DSI magnetometer provides a better sensitivity but no noise density is specified, having also the highest current consumption compared to the BMX055 or the LIS2MDL. From Table 5.8, the ICM-20948 offers the highest sampling rate and the lowest power consumption when compared to the remaining sensors, although its noise density is the highest. Finally, from the gyroscopes detailed in Table 5.9 the same characteristics occur with the ICM-20948, although the BMX-055 provides the best sensitivity for this sensor type.

Since one of the ADCS requirements of the COMv2 subsystem is to acquire sensor measurements to determine the satellite attitude with a possible redundancy, the first design criteria considered is to have at least two sensors of each type to provide two redundant 9 DoF systems. Therefore, two IMUs with the three sensor types in a single integrated circuit are preferably considered to minimise the use

of PCB space. Since the power consumption must be reduced whenever possible, the ICM-20948 and the BMX055 sensors are chosen as the sensors used for the attitude determination, discarding the remaining LSM9DSI 9-axis option due to its greater power consumption.

The ICM-20948[14] from InvenSense provides the three sensor types in a 3 axis reference system. Its gyroscope can be programmed with a measurement range of $\pm$250/500/1000/2000 °/s and its accelerometer with $\pm$2/4/8/16 g. The magnetometer has a fixed measurement range of $\pm$4900 μT. Additionally, an integrated temperature sensor with a sensitivity of 0.003 °C/LSB is available, which is also important to determine the temperature inside the spacecraft. An SPI or an I^2C interface may be selected to access this device.

The BMX055[15] from Bosch also provides the three sensor types in a 3 axis reference system. Its gyroscope can be programmed with a measurement range of $\pm$125/250/500/1000/2000 °/s and its accelerometer with $\pm$2/4/8/16 g. The magnetometer has a fixed measurement range of $\pm$1300 μT for its X and Y axis and of $\pm$2500 μT for its Z axis. A temperature sensor with a sensitivity of 0.5 °C/LSB is also integrated. Again, the communication with this device can be done either by a SPI or an I^2C interface.

Table 5.10 summarises the electrical specifications for both sensors selected. Special care must be taken with the I/O power supply voltage of the ICM-20948, which ranges only between 1.71 V and 1.95 V. Therefore, a voltage reference REF3318[16] from Texas Instruments is used to provide a stable 1.8 V I/O power supply voltage from the regulated voltage VCC_REG. Although being a voltage reference, its output provides an output current up to 5 mA, which is sufficient to power the digital circuitry of this sensor. Finally, a single SPI interface is chosen to communicate with these two sensors due to its faster operating frequency, operating in a single master with multiple slaves approach. The validation and calibration of these IMU sensors is done in Chapter 6.

Table 5.10: Electrical specifications of the COMv2 attitude determination sensors.

Parameter		ICM-20948	BMX055
Power Supply Voltage (V)	Min	1.71	2.4
	Max	3.6	3.6
I/O Power Supply Voltage (V)	Min	1.71	1.2
	Max	1.95	3.6
Typical Current Consumption (mA)		3.11	5.3
Operating Temperature (°C)	Min	-40	-40
	Max	85	85
Maximum Operating Frequency (kHz)	I^2C	400	400
	SPI	7000	10000

[14] https://www.invensense.com/wp-content/uploads/2016/06/DS-000189-ICM-20948-v1.3.pdf - Accessed on 03/01/2019

[15] https://www.mouser.com/ds/2/621/BST-BMX055-DS000-01v2-371988.pdf - Accessed on 05/01/2019

[16] http://www.ti.com/lit/ds/symlink/ref3333.pdf - Accessed on 03/01/2019

Additionally to the aforementioned sensors, transimpedance amplifiers are also used to amplify the current of six photodiode solar sensors available on the solar panels into ADC channels, as these sensors are required for the attitude determination. The circuit implemented for each photodiode is shown in Figure 5.22, being also present in the ISTsat-1 OBC.

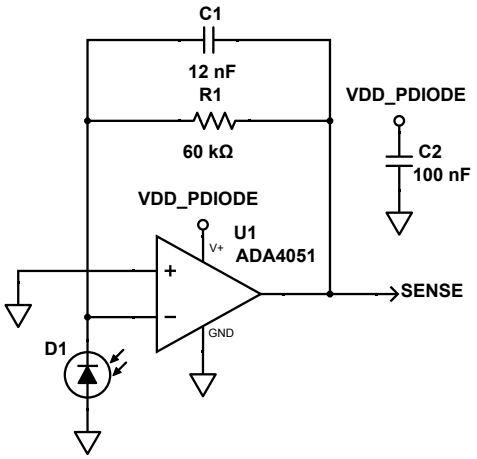


Figure 5.22: Transimpedance amplifier circuit used for the sun sensors present in solar panels.

To study the performance of this circuit, the SFH 2716[17]photodiode from OSRAM is chosen, with a typical photocurrent of 0.57 μA for an illuminance of 1000 lux. Analysing its datasheet and considering that the current flowing through this photodiode is directly proportional to the intensity of the incident light, a photodiode current of approximately 50 μA is obtained when in direct solar exposure, which is approximately equivalent to an illuminance of 100 000 lux [16]. Knowing that the maximum output voltage of the amplifier is set to 3.0 V by the regulated power supply VCC_REG, a feedback resistor

$$R_1 = \frac{V_{o_{Max}} - V_{o_{Min}}}{I_{i_{Max}}} \tag{5.10}$$

of 60 kΩ is chosen. Additionally, this circuit bandwidth is designed with a cut-off frequency f_c of 200 Hz, meaning that the feedback capacitor

$$C_1 = \frac{1}{2\pi R_1 f_c} \tag{5.11}$$

must be smaller than 13.2 nF to reach this bandwidth. For that, a feedback capacitor with a value of 12 nF is chosen, resulting in a cut-off frequency of 221 Hz, approximately. Finally, the output of this transimpedance amplifier is connected to an ADC channel, whose voltage is given by

$$V_{SENSE} = -R_1 I_{D_1} \tag{5.12}$$

where I_{D1} is the photocurrent flowing through the photodiode when exposed to incident light.

[17] https://www.osram.com/media/resource/hires/osram-dam-5467175/SFH+2716+A01_EN.pdf - Accessed on 06/07/2019

Please note that the component values achieved refer to a preselected photodiode, being this circuit operation studied in Chapter 6. However, these formulations are valid for other photodiodes whose current flowing through them is directly proportional to the intensity of the incident light.

The last implementation regarding the attitude determination sensors is a SPI interface connected to the aforementioned solar panel connectors. This communication interface is also considered in the ISTsat-1 OBC where, in the case of the EnduroSat solar panels, three identical 1-axis gyroscope sensors are present in three solar panels, one for each axis of the satellite, providing a 3 DoF system. This sensors, whose model can not be revealed due to confidentiality issues, are accessed by this common SPI interface with three CS signals for the different solar panel connectors.

5.2.8 Attitude determination actuators

Regarding the actuators used for the attitude control system, the use of the magnetorquers also available on the solar panels is considered, keeping the same architecture used on the ISTsat-1 OBC. For that, PWM signals are generated by the microcontroller to drive three magnetorquers present in three solar panels, one for each axis of the satellite, using three DRV8837[18] H-bridge drivers from Texas Instruments. Capable of providing an output current of 1.8 A with a power supply of 3.3 V, this driver operates with two PWM signals which control the current flow through the magnetorquers like in a H-bridge. A SLEEP pin is also provided by this IC to enter a low-power mode when these drivers are not necessary, which is controlled by a GPIO pin. The operation of this circuit is studied in Chapter 6.

Although this circuit was firstly thought considering the magnetorquers present in the EnduroSat solar panels, other magnetorquers may be used by these drivers through the same solar panel connectors. Additionally, these PWM signals may also be available on the PC/104 main system bus through the use of jumpers if another magnetorquers outside the solar panels are considered by the ISTnanosat team in the future.

5.2.9 External Watchdog Timer

As implemented in the COMv1.1 and explained in Section 5.1.3, the TPS3823A voltage monitor and external watchdog timer is also used in the COMv2 to provide redundancy to the system and act when a Single Event Effect or software glitch prevents the microcontroller internal one from operating correctly. If this external watchdog timer input is not refreshed through the commutation of a GPIO pin within a 2 s timeout, the microcontroller is reset.

[18] http://www.ti.com/lit/ds/symlink/drv8838.pdf - Accessed on 10/02/2019

5.2.10 Communication peripherals

Like in the COMv1.1, the communication with the other subsystems in the spacecraft is done through the two standard I^2C interfaces, with the same external LTC4303 I^2C bus buffers referred in Section 5.1.6. These are connected to each I^2C interface of the microcontroller, automatically disconnecting the main bus from this microcontroller if it tries to block the bus.

Additionally, a new CAN communication protocol is introduced as an alternative interface to complement or replace the aforementioned I^2C interfaces. Being not implemented in the ISTsat-1 satellite, this interface may be used for telemetry or payload data. For this, a CAN transceiver driver is required to provide the transmission and reception capability between the differential CAN bus and the CAN controller on the microcontroller. The use of this robust protocol adds more redundancy to the communication between the several subsystems. In Table 5.11 a brief comparison between the two aforementioned communication protocols is detailed.

Table 5.11: Comparison between the I^2C and CAN interfaces.

Parameter	I^2C	CAN
Communication Type	Node oriented, where a device is selected through its address. Half-duplex and multi-master.	Message oriented, where the messages on the bus are defined by its ID. Half-duplex and multi-master.
Payload Size	Unlimited (although big payloads prevent the bus from being used by other devices).	Up to 8 bytes on normal CAN, requiring fragmentation for bigger payload sizes.
Data Rate	Up to 400 kHz when protected with I^2C buffers.	Up to 1 Mbit/s.
Bus Pinout	Synchronous bus, SDA/SCL	Asynchronous differential bus, CANL/CANH
Availability	Typical protocol available on most microcontrollers.	Requiring a demanding CAN controller, this protocol is not available on some low-power microcontrollers.
Power Consumption	Low, limited to the node I^2C peripherals and pull-up resistors.	High, as the CAN controller and transceiver drivers required per node for the differential bus have a considerable power consumption.
Reliability	Low, since a software fault may hang the bus, although bus buffers exist to free the bus.	High, with fault tolerance, data validation and error detection on the bus messages.
Protocol Complexity	Medium, due to its node addressing and acknowledging.	High, due to its complex frame format.

Although CAN provides an higher data rate than I^2C, being also more reliable in its fault management, this interface has a more complex message oriented protocol, requiring fragmentation for considerable payload sizes. Additionally, a bigger power consumption is expected due to the hardware required for its differential bus. Nevertheless, these communication protocols are two viable choices for the system communication in a cubesat due to the multi-master topology, where several subsystems can use a

same interface to communicate between them. Therefore, the use of one interface or both for the system communication may be chosen by the ISTnanosat team in the future. Nevertheless, a comparison of the throughput achieved with both interfaces using the COMv2 subsystem is done in Chapter 6.

Additionally to these main system interfaces, the same dedicated interface with the TT&C subsystem is again considered for the communication stack interchange, with a SPI interface plus a RTS signal, as implemented in the COMv1.1 and referred in Section 5.1.6. However, the use of a synchronous serial interface is also considered by the ISTnanosat team to replace the former interface. Although this implementation is not studied in this master thesis, a performance analysis of the performance between these interfaces may be done in the future by the ISTnanosat team.

As in the COMv1.1, a GPIO pin is connected to the Beacon through the PC/104 main system bus to inform the operation status of the COMv2.

Two additional optional asynchronous serial communication interfaces are also considered. The former may be used as an interface with a GNSS, used for a more accurate satellite positioning. This interface is available on the PC/104 main connector or on a dedicated internal connector, selected through the use of jumpers. The latter interface may be used as a dedicated interface between a payload subsystem to receive payload data, also relieving the main I^2C and CAN buses.

Regarding the debug of the subsystem, a programming SWD interface is used for the microcontroller programming and debugging. Although not studied in this master thesis, an USB interface is considered to allow the system to communicate easily with any computer, plus a mass storage SD card.

5.2.11 Power distribution and protections for the subsystem modules

In the previous sections, the several power and digital modules implemented in the COMv2 are detailed. However, the study of the power distribution within the subsystem modules is also important, considering the radiation protections designed to verify their correct operation. Therefore, the proposed system power distribution is shown in the block diagram from Figure 5.23. Recalling Section 5.2.4, power cut-off switches and current measurement circuits are implemented:

- For the attitude determination sensors, powered by the regulated VCC_REG bus, where a single current measurement circuit is considered together with two separate switches that cut-off the power of each sensor independently.
- For the six transimpedance amplifiers for the photodiode sun sensors, powered by the regulated VCC_REG power bus, where a current sensing circuit and a single switch are used.
- For the attitude determination actuators, powered by the regulated VCC_REG power line, with a current sensing circuit and a cut-off switch used to control the power of the magnetorquer drivers.
- For the additional sensors placed on the solar panels, which can be powered with the regulated VCC_REG power line or by the 5 V power bus provided on the PC/104 main system bus, chosen

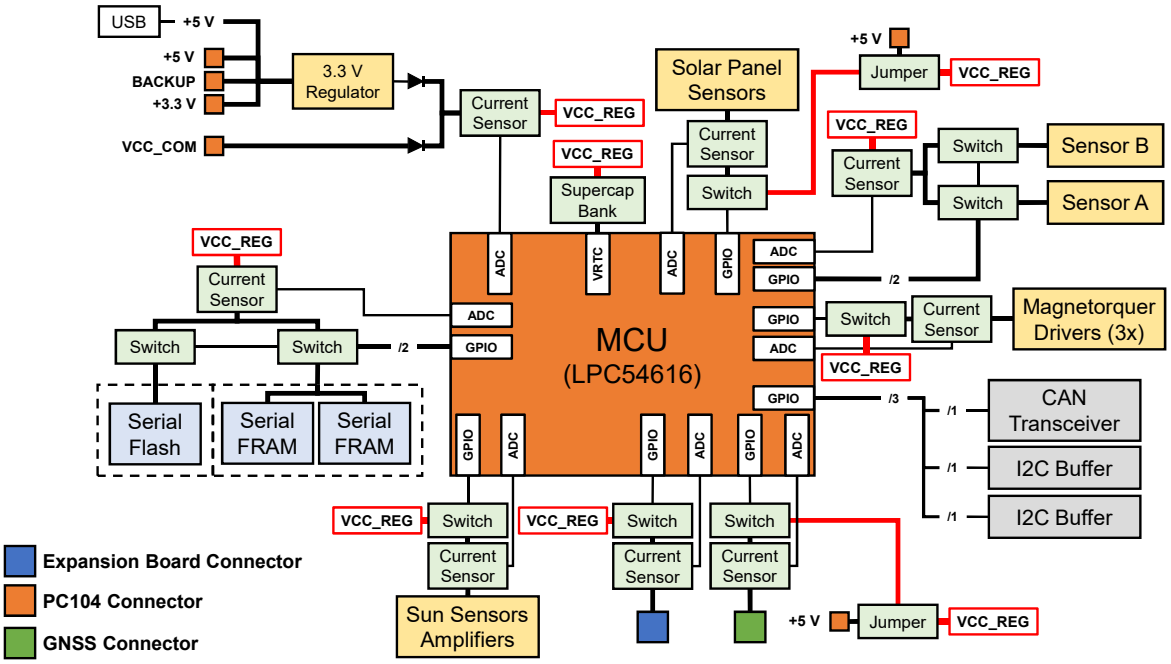


Figure 5.23: Block diagram of the COMv2 power distribution.

through a jumper, where another current sensing circuit and a single switch are considered.

- For the internal storage, powered by the regulated VCC_REG power line, a single current measurement circuit is considered together with two separate switches that cut-off the power of the Flash memory and both FeRAMs independently.
- For the daughterboard, powered by the regulated VCC_REG power line, a current sensing circuit and a single switch are used.
- For the GNSS internal connector, which can be powered with the regulated VCC_REG power line or by the 5 V power bus provided on the PC/104 main system bus, chosen through a jumper, where another current sensing circuit and a single switch are considered.

Additionally, a current sensor in series with the output of the power regulation circuit that provides the VCC_REG power line is considered, which is used to measure the whole subsystem power consumption on this power bus. For the I^2C buffers and for the CAN transceiver, used to communicate with the remaining subsystems, no current measurement or power cut-off is done, although optional GPIO lines are reserved for these components so that they can be put in a sleep mode by the microcontroller.

5.2.12 COMv2 subsystem and memory daughterboard prototype boards

A prototype COMv2 board has been designed with the modules defined in the previous sections, which is shown in Figure 5.24 with how these different modules populate the board. This PCB, with six layers and with a size of 93.0 mm x 86.4 mm, respects the cubesat standard and serves as a feasibility study of

its use in the future ISTnanosat team satellites. Additionally, a prototype of the aforementioned memory expansion daughterboard has been designed, which is shown in Figure 5.25.

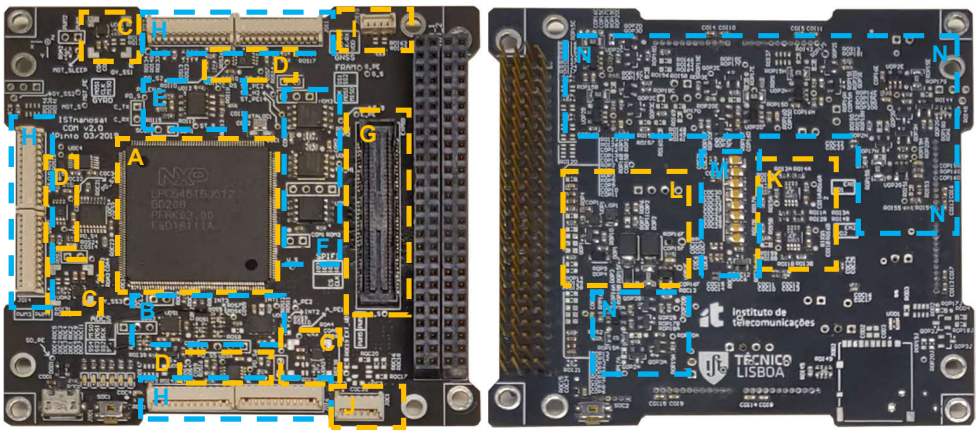


Figure 5.24: Top (left) and bottom (right) views of the COMv2 prototype board, highlighting the main modules.

A: Microcontroller
B: ADCS sensors
C: ADCS magnetorquer drivers
D: Sun sensor amplifiers
E: CAN transceiver driver
F: FeRAM and Flash memories
G: Daughterboard connector
H: Solar panel connectors
I: GNSS connector
J: Debug connector
K: I^2C interface buffers
L: Subsystem power regulation
M: RTC capacitor bank
N: Current sensors/cut-off switches

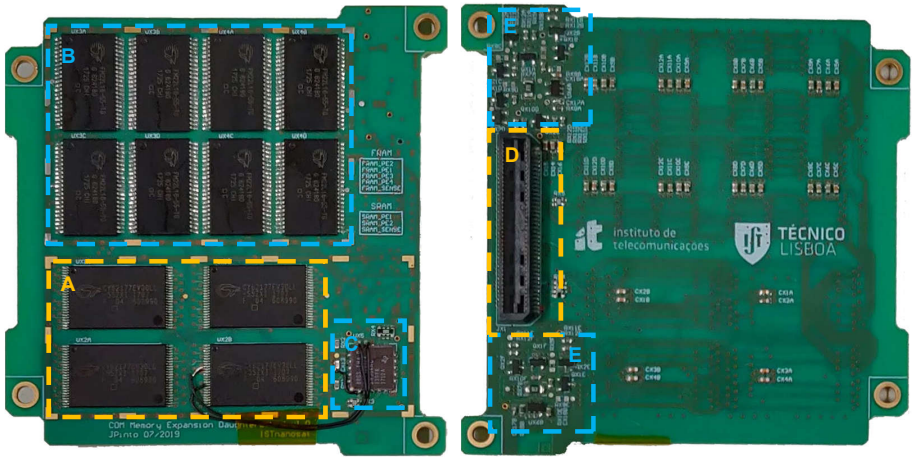


Figure 5.25: Top (left) and bottom (right) views of the COMv2 memory expansion daughterboard prototype.

A: SRAM memory bank
B: FeRAM memory bank
C: Decoder for the memory unit
D: Daughterboard connector
E: Current sensors/cut-off switches

5.3 Software Implementation

In this section, the software structure developed to meet the requirements set for both versions of the Communications Processor Board is detailed. The objective of the software developed in this master thesis is to provide the necessary abstraction layers between the hardware described in the previous sections and an application running in these subsystems. Therefore, this section focuses on the system software rather than on the software application layer, which is further referred as application software. This system software includes inherently the hardware abstraction layers used for the virtualisation of the underneath hardware platform, which are required for the application.

Despite different COM versions are implemented, the system software stack developed is similar, which is shown in Figure 5.26. The first layer that translates the hardware present in these subsystems into the system software is the low level layer of the Hardware Abstraction Layer (HAL), mainly composed of component drivers code from ARM and NXP. Above this layer is the high level layer of this HAL, which provides an understandable API to work with the required microcontroller peripherals. This API was developed by the ISTnanosat team to provide a common interface to the hardware of the different subsystems present in its satellites.

As the next software layer, a set of subsystem drivers that resorts to this HAL is designed to access the different hardware modules present in the COM subsystems. Additionally, board configuration files are implemented to identify every peripheral and its correspondent configuration, allowing a faster reconfiguration without changing the HAL implementation.

The final layer of the system software is provided with the use of a Real-Time Operating System (RTOS). This layer provides to the application software an easier way to deal with the real-time requirements set for the COM subsystems, which must respond to several hardware and software events within certain time limits, easing the communication and synchronisation between the system software and the application software through the use of IPC mechanisms. The *Services* layer is the first layer of the application software stack that also uses this RTOS layer, translating, managing and validating the data transfer between the system software and the application software.

The different layers of the system software stack are described in detail in the further sections. All the software is developed in C language, being its file structure directly connected with this system software stack. For the software development, the MCUXpresso IDE from NXP is firstly used. Additionally, a development environment created for the ISTsat-1 test and integration is also used, built during the development of this master thesis and by the ISTnanosat team. This development environment is based on a set of *makefiles* that select the platform and project to be compiled and run on the several ISTsat-1 subsystems, easing the software version control and the continuous integration of software made by different members of the ISTnanosat team.

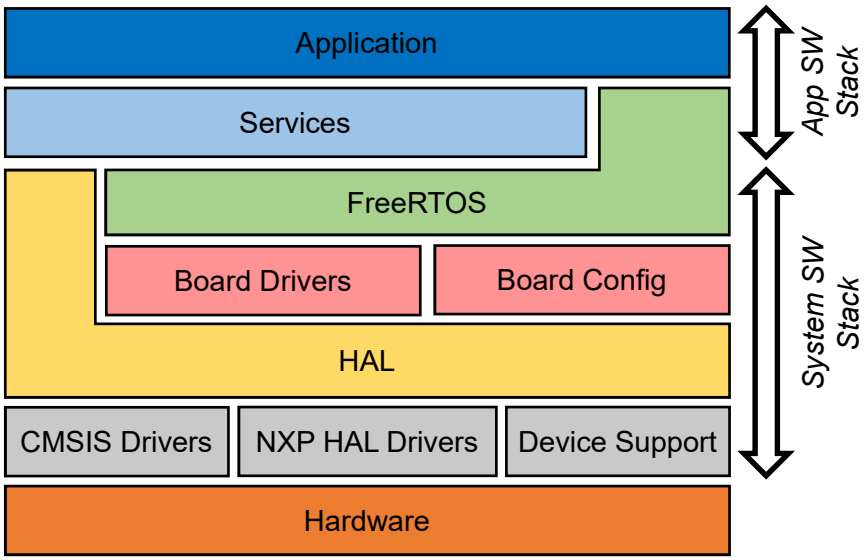


Figure 5.26: Software stack implemented on both versions of the Communications Processor Board.

5.3.1 Hardware Abstraction Layer

The HAL is the first abstraction layer that isolates the differences between the several peripherals required by the microcontroller of each subsystem. The ISTnanosat team has decided that this HAL shall be equal in the different ISTsat-1 subsystems whenever possible, providing a similar, portable and modular software API, which can be used equally by the application software of the different subsystems. This also applies to both COM versions, whose system software tries to follow this specification.

Considering both COM subsystems developed, this HAL is composed of two distinct parts. The former part is divided into three different libraries, which are provided by the manufacturer of the microcontrollers chosen:

- The **CMSIS drivers**, which are an ARM software interface standard between its ARM Cortex processors and their peripherals, being designed regardless of the different MCU vendors[19]. In this case, these drivers define different access layers to the core peripherals, providing different compiling definitions and the interrupt vector tables present on all Cortex-M3 and Cortex-M4 cores, which are the ones used in the chosen LPC1833 and LPC54616 microcontrollers, respectively.
- The **NXP HAL drivers**, which are already a hardware abstraction layer designed by NXP that provides an API to work with the different MCU peripherals. However, this software API differs from the LPC1833 to the LPC54616, removing portability and modularity between them.
- The **device support drivers**, which are specific for each LPC1833 and LPC54616 processing units, providing the definition of peripheral registers, memory maps and basic boot-up sequences.

[19] https://developer.arm.com/tools-and-software/embedded/cmsis - Accessed on 2019/08/14.

Although NXP already provides a functional software API which could be used as hardware abstraction layers in both microcontrollers, these APIs do not respect the software specification defined by the ISTnanosat team. Therefore, the latter part of the HAL developed for the COM subsystems is an adaptation of the aforementioned NXP HAL drivers to a new software API that respects the ISTnanosat software specification.

Taking into account the requirements of each subsystem, several abstraction layers are implemented for the peripherals required, based on the ISTnanosat specification. Starting with the simple GPIO and ADC peripherals, some main functions are implemented for each peripheral:

- `gpio_config()`, used to set a pin as an output or as an input.
- `gpio_read()` and `gpio_write()`, used to read or write a pin logical state.
- `adc_config()`, used to configure an ADC with its resolution bits and acquisition periods.
- `adc_start()` and `adc_stop()`, used to start or stop the ADC operation.
- `adc_read()`, used to read the last value acquired by an ADC channel.

Additionally, functions for internal microcontroller peripherals are also implemented, such as the WDT, the internal RTC and internal timers:

- `wdt_config()`, used to configure the internal WDT with a predefined timeout;
- `wdt_start()` and `rtc_stop()`, used to start or stop the WDT operation;
- `wdt_pet()`, used to reset the WDT to zero, avoiding its timeout;
- `wdt_resetSource()`, used at the boot sequence to check if the microcontroller has restarted due its RESET line or due to a watchdog timeout;
- `rtc_config()`, used to configure the RTC with an epoch[20] time reference;
- `rtc_start()` and `rtc_stop()`, used to start or stop the RTC operation;
- `rtc_setTime()` and `rtc_getTime()`, used to set or get a RTC time reference;
- `timer_config()`, used to configure a timer with a predefined timeout;
- `timer_start()`, `timer_stop()` and `timer_reset()`, used to start, stop or reset a timer to zero;
- `timer_getValue_us()`, used to read the timer value, given in microseconds.

Regarding the PWM peripheral necessary for the magnetorquer drivers on the COMv2 subsystem, several routines are also implemented:

- `pwm_config()`, used to configure PWM channels with a predefined frequency;
- `pwm_start()` and `pwm_stop()`, used to start or stop the PWM operation;
- `pwm_set_dutycycle()` and `pwm_get_dutycycle()`, used to set or get a channel PWM duty cycle;
- `pwm_set_frequency()` and `pwm_get_frequency()`, used to set or get the PWM frequency.

[20] The Unix epoch time reference is the number of seconds that have elapsed since 1 January 1970.

The configuration of the ExtMC peripheral required for the parallel memory unit is done at the boot sequence, configuring the memory access times and the addressing type. Thereafter, the memory unit is accessed directly through addresses that are memory-mapped on the microcontroller.

The implementation of the hardware abstraction layers for the several communication peripherals required for the COM subsystems (UART, SPI, I^2C and CAN) is more complex, being mostly based on interrupts and non-blocking function calls, where the microcontroller can continue its code execution while the peripherals are dealing with the input or output of data from another device.

All the peripherals are initialised through their correspondent configuration routines, namely `uart_config()`, `spi_config()`, `i2c_config()` and `can_config()`. The resources required for these interfaces to work, like the dedicated buffers statically allocated for the data to be transmitted or to be received, are passed to these routine calls. For example, each UART interface has two dedicated circular buffers, one used for the byte array transmission and another for the byte array reception. Additionally, other configurations like the speed or bit rate desired, operational modes (in the case of SPI) or slave addresses (in the case of I^2C) are also passed to these function calls.

Considering the UART interfaces and the SPI interface only when operating in slave mode (used on the dedicated TT&C interface), these interfaces operate through the use of interrupts and circular buffers. In Figure 5.27 is shown the flowchart diagram of these interrupts, where the the reception and the transmission handling can be distinguished.

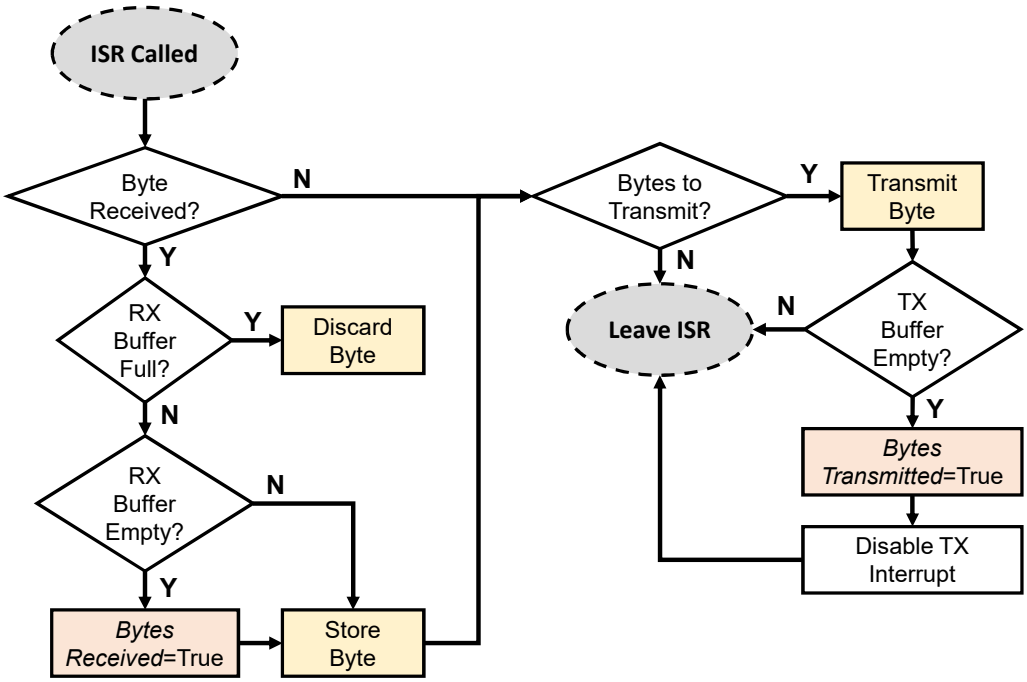


Figure 5.27: Flowchart of the byte reception and transmission on the communication interfaces of both COMs.

On the reception side, a new byte is stored in the reception buffer if this buffer is not full. If this buffer is empty, a flag is set to the application software to inform that the buffer has been filled with a new byte. Then, this application may use the `uart_read()` routine to retrieve this data from the buffer until it becomes empty again, when this flag is unset again. Please note that the reception interrupt is always enabled, meaning that if a byte is received, the ISR is automatically called.

On the transmission side, if the transmission buffer is not empty due to an `uart_write()` function call from the application software, which fills this buffer with data to transmit and enables the transmission interrupt flag, successive calls to the ISR are done after every a byte transmission until this buffer becomes empty again. At this point, another flag is set to the application software to inform that the data has been transmitted and the transmission interrupt is disabled.

Regarding the I^2C and CAN communication interfaces, a similar process is used, although the correspondent ISRs have to deal with state machines intrinsic to these protocols, making them more complex. Additionally, both reception and transmission flags are only set after a message has been received and not at every byte. Nevertheless, the application may use similar routines to send or receive messages from other nodes, namely `can_send()`, `i2c_send()`, `can_recv()` and `i2c_recv()`.

Finally, the SPI interface operating in master mode is based on an interrupt and polling mechanism, where a `spi_transfer()` routine is used. In this function call, a full-duplex SPI transfer is done, where the application remains blocked until all bytes have been transferred. On the other side, the interrupt handles each byte reception and transmission as depicted in Figure 5.27 until the operation is complete, with the new data received being available to the freed application. This blocking approach is used since the SPI interface is configured in master mode, controlling all of its transfers, which are known *a priori* to be small and based on a command-response procedure due to the hardware used, consuming low processing time.

5.3.2 Board hardware drivers

Being the hardware abstraction layer of the peripherals developed, the next step is to develop the software drivers to work with remaining hardware present in the COM subsystems, using this HAL as a bridge between the processing unit and this hardware.

Starting with the COMv2 subsystem, two SPI interfaces operating in master mode are used, one to access both attitude determination sensors and another to access both FeRAM memories. In the former, two setup functions, `ICM20948_setup()` and `BMX055_setup()`, are used to access the configuration registers of each sensor, enabling their continuous operation and programming, for example, their measurement ranges and rates of acquisition. Thereafter, read routines are used to obtain the values measured by each sensor, which are returned to the application already in their correspondent physical quantity.

In the latter, two functions calls are used to write or read a byte array from a predefined address, namely `FRAM_write()` and `FRAM_read()`, respectively. The same approach is done with the external Flash memory, where an `ExtFlash_read()` routine is used to read a byte array from a predefined address, and an `ExtFlash_write()` routine is used to write a byte array to a Flash memory sector.

Regarding the magnetorquer drivers, a function call `mtq_set_dutycycle()` is created so that the different PWMs required for each magnetorquer driver are updated at the same time with different duty cycles. The frequency of these PWMs is set at the driver configuration with `mtq_config()`, which remains constant throughout the satellite attitude control.

Concerning the solar panel sun sensors, a `sunsensor_read()` routine is created to provide to the application the intensity of the incident light present in each sun sensor, resorting to the correspondent ADC interface for this measurement.

All the aforementioned hardware have also initialisation or configuration routines, which configure the aforementioned microcontroller peripherals required for its correct operation. Additionally, the GPIO and ADC channels required for the correspondent current sensors and power cut-off switches are also configured in these routines, which are thereafter used to enable or disable the power of each module (with `enablePower()` and `disablePower()`) and read its current consumption (with `readCurrent()`).

All the hardware present in the COMv1.1 subsystem is dealt directly through its HAL, not requiring any specific board driver.

5.3.3 Real-Time Operating System

A Real-Time Operating System is an operating system designed for systems with real-time requirements, where this system must respond to several events within certain time limits. To ensure that these requirements are met, a scheduler is used to assign different priorities to different threads of execution, providing a predictable and deterministic execution pattern, where different tasks are executed concurrently one after another.

Both COM subsystems rely on the FreeRTOS[21] operating system to meet their real-time requirements. This RTOS is designed specially for embedded systems with low processing power, with its simple kernel requiring a low memory around 4 kB.

Recalling the software stack from Figure 5.26, the *Services* layer is mainly responsible for the data exchange between the system software and the application software, resorting to different services based on FreeRTOS tasks that are used, for example, to manage the data transferred through the UART, I^2C, SPI or CAN peripherals. The IPC mechanisms selected for the synchronisation between the system software and these services are task notifications, used to notify the application software whenever a new hardware event is triggered. For example, after an I^2C message has been successfully

[21] https://www.freertos.org/about-RTOS.html - Accessed on 2019/09/29.

received, the task correspondent to the I^2C service is notified that a new message can be collected from the internal HAL buffers in order to be validated and handled to the application software, whenever a valid message is received.

Furthermore, the FreeRTOS is also ported to these subsystems to allow the application software to easily manage the real-time requirements with real time scheduling functionality, inter-task communication, timing and synchronisation primitives.

5.3.4 Board configuration files

Another important part of the developed software stack are its board configuration files, in which are detailed the identification of every peripheral and their correspondent configuration. These files are particularly useful because every configuration is detached from the HAL, which must be prepared *a priori* to apply each one of these configurations. For example, considering the speed of a SPI interface, different speeds may be selected on the correspondent configuration file but the HAL does not need to be modified after an user selects the necessary speed, enabling a fast reconfiguration only through the use of different definitions. Another example applies to different COMv2 daughterboards, where the microcontroller multiplexed pins may be configured differently depending on the daughterboard used, as the HAL that initialises these pins is the same for every daughterboard but the pin definitions are changed only with different configuration files correspondent to each daughterboard.

5.3.5 Boot sequence for top-level application

Being the system software set, a similar boot sequence is implemented in both COM versions, mainly dealing with the initialisation of the required peripherals and hardware for each subsystem. This startup sequence is detailed in Figure 5.28.

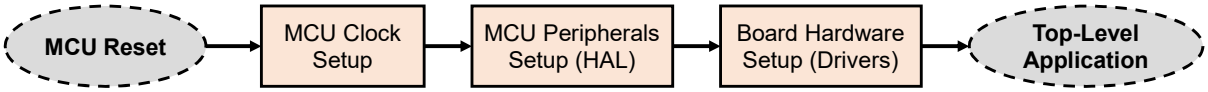


Figure 5.28: Boot sequence implemented on both COM versions.

After a microcontroller reset, made either by a power-on reset or by a watchdog timer timeout, the first step is to set the clock frequency in which the microcontroller will operate. In the case of the COMv1.1 subsystem, the operating frequency is set to 30 MHz. Although the ARM Cortex-M3 used allows a clock frequency up to 180 MHz, the ISTnanosat team has decided to reduce it to lower the power consumption of this subsystem. In the case of the COMv2 subsystem, though the maximum clock frequency possible is also 180 MHz, the main clock frequency of its microcontroller is reduced to 96 MHz not only to reduce

the power consumption but also due to a limitation on the CAN peripheral, whose clock requires the main clock to be a multiple of 8 MHz [15]. Nevertheless, both clock frequencies can be easily reconfigured in the future if more processing power or less power consumption are required.

Being the main clock frequency of the processing units set, the next step is to initialise the microcontroller peripherals required for the correct subsystem operation. This initialisation deals with the different pinout configuration and the setup of internal peripherals, like the WDT and the RTC, and the setup of the communication peripherals, like the I^2C and the SPI. After all microcontroller peripherals are initialised, the remaining subsystem hardware can be configured as well, resorting to these peripherals.

All initial setup takes into account the aforementioned configuration files, booting up all the hardware present in the subsystems, which can be accessed by the application software using the routines detailed on the previous sections. Being all the hardware configured, the real-time operating system is started and the top-level application software begins.

In this Chapter 5, the components and modules designed for both COM subsystems were described taking into account the purpose of each module, considering their hardware and software functionalities.

Chapter 6

System Validation

This Chapter 6 serves the purpose of detailing the validation of the modules selected and designed for both COMv1.1 and COMv2, with a description of the procedures done and tests developed. This validation is performed considering the different functionalities of these modules rather than studying both subsystems separately, since some requirements are shared between these subsystems. Therefore, this chapter is divided considering the validation of:

- Both parallel and serial external memory units implemented.
- The communication peripherals.
- The attitude determination and control modules.

6.1 Memory units validation

Due to the different memory typologies used in both COM subsystems, the validation of the memory units is done differently in the parallel and serial memories. For the parallel memory unit, the validation is done resorting to three memory tests developed by Michael Barr, detecting either electrical connection problems on the parallel memory bus, missing memory chips or improperly inserted memories [17], [18].

The first test is performed to confirm if the data bus is correctly connected between the microcontroller and the memory unit, detecting either an open connection on the data bus or a short circuit between any of the data wires. For that, a *walking 1's* test is performed, where a single data bit is set to 1 and shifted through the entire data word, being the remaining bits set to zero (0001b $\rightarrow$ 0010b $\rightarrow$ 0100b $\rightarrow$ ...). The successive writes are done to a same address and verified afterwards. This test is done for all width of the data bus (which is 32 bits, in the case of the memory unit developed).

Being the data bus verified, a second test is done to confirm that the address bus is correctly connected between the processing unit and the memory units, in order to detect overlapping

memory locations. This address test follows the same *walking 1's* approach but regarding the address bus, isolating each address bit and confirming that each address line can be set to 0 and 1 without affecting the others. The addresses tested are the ”power-of-two” addresses (000h $\rightarrow$ 001h $\rightarrow$ 002h $\rightarrow$ 004h $\rightarrow$ 008h $\rightarrow$ 010h $\rightarrow$ 020h $\rightarrow$ ...). The test procedure starts with writing a data pattern to each power-of-two address offset within a memory. Then, an inverted data pattern is written to the first test offset, verifying that the initial pattern is still stored at every other power-of-two addresses. If any of these addresses contain the inverted pattern, a problem with the current address bit exists. If no overlapping is found, the procedure is repeated for the remaining offsets until all the address bus width is tested (which is 24 bits, in the case of the memory unit developed[1]).

After both data and address buses are successfully tested, a third test is used to ensure the integrity of the memory devices, where every bit in a device must be capable of holding both logical values. The procedure of this test is based on a longer *increment* test, where every memory location is written and verified twice. On the first pass, incremental data is written and verified, being inverted, written and verified again on a second pass, as shown in Table 6.1.

Table 6.1: Example of an incremental test, with the values written on both passes for each address offset.

Address Offset	First Pass	Second Pass (Inverted)
00h	00000001	11111110
01h	00000010	11111101
02h	00000011	11111100
...	...	...
FDh	11111110	00000001
FEh	11111111	00000000
FFh	00000000	11111111

Being the whole memory unit correctly addressed through its decoder, the aforementioned third test was done to ensure the integrity of whole memory unit of both COMv1.1 and COMv2 memory expansion board, with an 100 % successful rate after 1000 test repetitions, meaning that the parallel memory unit is fully operational.

Considering the serial memory unit implemented on the COMv2 subsystem, the access to the memories chosen is mostly based on commands sent by the microcontroller through the correspondent serial peripherals, usually followed by a response from the memory:

- For a read procedure, a specific command and desired address are sent to the memory, followed by subsequent transactions where data is read from the memory.
- For a write procedure, a specific command and desired address are sent to the memory, being the data written into the memories on the following transactions.

[1] To verify the correct connection of every memory pair used separately, smaller address bus widths of 23 bits and 20 bits may be used for the SRAM and FeRAM memories, respectively.

- Additionally, on the FeRAM chosen, a Write Enable command must be issued before every write for a valid write procedure. Due to the Flash memory architecture, a command to erase the sector to which new data is going to be written must also be sent to the memory. The microcontroller must then wait for this erase to be finished before writing the new data, polling a memory register that informs when the sector is ready to be written again through a specific command. These write enable and erase procedures are required operations that decrease the memory bandwidth.

For the validation of the serial memory unit implemented on the COMv2 subsystem, another memory test is done based on a buffer with a fixed size and random data, which is written to an address and verified afterwards with a read procedure. This test is then repeated several times for the following addresses, and the success rate of the whole test is measured. This test assumes that both SPI and SPIFI peripherals, used to access the FeRAM and Flash memories, are correctly operating, with a chosen operating frequency of 2 MHz and 48 MHz, respectively. After 1000 repetitions of the aforementioned test, an 100 % successful rate was achieved, meaning that the serial memory unit is fully operational.

Performance comparison between parallel and serial memory units

To evaluate the memory bandwidth obtained with both parallel and serial memory units, a performance test was done to each memory type, whose results are shown in Table 6.2. The memory bandwidth is determined for both read and write accesses, where a buffer with a predefined size is written to a memory several times, measuring the total time elapsed separately for all write operations. Afterwards, the same procedure is done but reading the same buffer size from the memory. These times are measured using the internal microcontroller timers and doing an average of 10 test repetitions. Therefore, a memory bandwidth

$$Bw = \frac{B_{size} \times N}{t_{av}} \tag{6.1}$$

is calculated, where B_{size} is the buffer size for each transfer, N the number of transfers done and t_{av} the time elapsed for these transfers, averaged for 10 test repetitions.

Considering the serial memory unit, a difference between the read and write memory bandwidths is confirmed, being the write memory accesses slower than the read accesses. Furthermore, this difference is more evident with the serial Flash, where the write memory bandwidth decreases substantially due to the time waiting for the erase of the Flash sectors, whose rate is set to 61 kB/s, according to this memory datasheet. However, the read memory bandwidth obtained in this Flash memory is considerably high, being comparable with the parallel memory unit, with a lower memory bandwidth being obtained for byte addressing. Furthermore, recalling Figure 5.6 and Figure 5.7, the fact that write accesses are faster than read accesses is confirmed, with a higher write memory bandwidth obtained for the parallel memory when compared with the read memory bandwidth.

Table 6.2: Performance comparison between implemented memory units, with the memory bandwidths obtained.

Memory Type	Addressing Type	Buffer Size (Bytes)	Transfers	Time* (ms)		**Bandwidth (MB/s)**	
				Read	Write	Read	Write
Serial FeRAM	-	64	100	44	48	0.15	0.13
Serial Flash	-	4096	10	10	750	3.98	0.054
Parallel (COMv1.1)	32-bit	16384	100	260	164	6.28	9.95
	16-bit	16384	100	521	329	3.14	4.98
	8-bit	16384	100	1042	659	1.57	2.49
Parallel (COMv2)	32-bit	16384	100	154	94	10.66	17.44
	16-bit	16384	100	307	205	5.33	7.99
	8-bit	16384	100	581	376	2.82	4.36
*Average time calculated for 10 test repetitions.							

Please note that different processing units were used to achieve these results, as the parallel memory unit is tested on both COM subsystems but the serial memory unit is only tested on the COMv2 subsystem. Additionally, an increase of the memory bandwidth occurs when using the parallel memory expansion daughterboard on the COMv2 subsystem, since the microcontroller is operating at a higher frequency (96 MHz instead of 30 MHz) and has a better ARM architecture (Cortex-M4 instead of Cortex-M3), improving the memory access times.

6.2 Validation of the communication peripherals

The validation of the several communication interfaces implemented on both COM subsystems starts with a *ping* test between these subsystems and another device with these interfaces already debugged and fully operational. For that, a Teensy 3.5 development board with an 120 MHz ARM Cortex-M4 is used, resorting to a simple Arduino framework for a fast software development. During this test, bugs detected on the developed HAL are fixed, improving the robustness of the communication peripherals.

After this ping test is successful in both directions, a performance test is done to measure the throughput of each communication peripheral. The throughput is measured as the rate of successful messages delivered over an interface, where messages with a predefined size are written to a single device several times, measuring the total time elapsed for all transactions. Again, these times are measured using the internal microcontroller timers and doing an average of 10 test repetitions. Therefore, a throughput

$$Th = \frac{B_{size} \times N}{t_{av}} \tag{6.2}$$

is calculated, where B_{size} is the buffer size of each message, N the number of messages sent and t_{av} the time elapsed for all transactions to be done, averaged for 10 test repetitions. The results for these

throughput tests are shown in Table 6.3, where different buffer sizes, number of transfers and operating frequencies or baud rates are used for the implemented SPI, I^2C, UART and CAN interfaces.

Table 6.3: Performance comparison between implemented serial peripherals, with the throughputs obtained.

Protocol	Operating Frequency Baud Rate	Buffer Size (Bytes)	Transfers	Time* (s)	**Throughput (kB/s)**
SPI (w/ RTS)	500 kHz	8	4096	0.76	42.99
	500 kHz	300	256	1.63	47.01
I^2C	100 kHz	8	4096	3.68	8.91
	100 kHz	64	1024	6.39	10.26
	400 kHz	8	4096	1.24	26.49
	400 kHz	64	1024	2.07	31.69
UART (8N1)	115.2 kb/s	8	4096	2.84	11.54
	115.2 kb/s	64	1024	5.68	11.55
CAN	500 kb/s	8	4096	1.03	31.68
	1 Mb/s	8	4096	0.55	59.90
*Average time calculated for 10 test repetitions.					

For a fixed operating frequency, both SPI and I^2C interfaces have a higher throughput when transferring buffers with a higher size, since the overhead per transfer intrinsic to these protocols and latency between transfers are reduced. However, the same does not happen for the UART interface, whose overhead is not dependable of the size of the transferred buffer, being the same for every byte. Comparing both I^2C and CAN peripherals, which are the options considered for the satellite main bus, a higher throughput is achieved for the latter when transferring buffers with the same size, using an operating frequency of 400 kHz and a baud rate of 500 kbit/s, respectively. Nevertheless, the throughput of these interfaces becomes similar as the buffer transferred size increases in the I^2C interface. This can not be done in the CAN interface due to its message size limitation of 8 bytes, requiring fragmentation. Nevertheless, when using a 1 Mbit/s rate, the CAN peripheral provides the highest throughput, surpassing the throughput achieved with the SPI interface for 500 kHz, even when big buffer sizes are transferred through this interface.

6.3 ADCS modules validation

Since different attitude modules are implemented on the COMv2 subsystem, whose purpose focuses either on the attitude determination or on the attitude control, different acceptance procedures are described in this section to validate these modules.

Starting with the attitude determination, the first validated modules were the inertial sensors implemented on the COMv2, namely the two sets of gyroscopes, accelerometers and magnetometers. Although these MEMS offer an easy integration of inertial sensors on this subsystem, the correct

use of these sensors can not be done without a prior calibration that eliminates structural errors, which may result on differences between the expected and measured outputs. According to [19], the following sensor model may be used to calibrate inertial sensors, where the calibrated values for the three axis

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 1 & M_{XY} & M_{XZ} \\ M_{YX} & 1 & M_{YZ} \\ M_{ZX} & M_{ZY} & 1 \end{bmatrix} \times \begin{bmatrix} 1/S_X & 0 & 0 \\ 0 & 1/S_Y & 0 \\ 0 & 0 & 1/S_Z \end{bmatrix} \times \left(\begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix} - \begin{bmatrix} X_{off} \\ Y_{off} \\ Z_{off} \end{bmatrix} \right) \tag{6.3}$$

are easily computed considering the $(X, Y, Z)_r$ measured raw values, where the M_{ij} matrix provides the axis misalignment errors existent due to the non-orthogonality of the three sensor axis, the S_i matrix gives the scale factors between the measured and expected values, and the $(X, Y, Z)_{off}$ vector represents the constant bias offset that needs to be removed from the non calibrated values. The referential axis system used for this calibration is depicted in Figure 6.1, which is ported from the ISTsat-1 satellite.

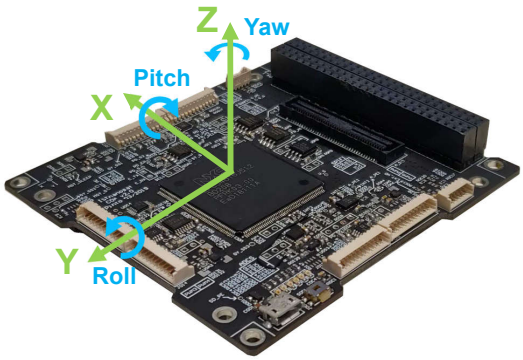


Figure 6.1: COMv2 referential axis system, adapted from the ISTsat-1 cubesat.

The accelerometer calibration is based on the Earth gravity field, where each sensor axis is placed orthogonally to the surface of the Earth in both directions, expecting either a positive or negative 1 g (9.807 m/s^2) of acceleration in that axis, as shown in Table 6.4. An acquisition of raw acceleration values was done for each orientation, for at least 10 s with a sampling rate of 100 Hz, to which was applied a least square method in order to obtain the correspondent sensor calibration values of Equation 6.3, as detailed in [20].

Table 6.4: Expected values for each accelerometer axis when placed orthogonally to the Earth's surface.

Orientation	X axis	Y axis	Z axis
+X	+1 g	0	0
-X	-1 g	0	0
+Y	0	+1 g	0
-Y	0	-1 g	0
+Z	0	0	+1 g
-Z	0	0	-1 g

After calculating the aforementioned calibration values, a tilt test was applied to the COMv2 board, whose results are shown in Figure 6.2. This test studies the response of both accelerometers available in the ICM-20498 and BMX055 sensors. The COMv2 board was rotated in different orientations, staying at rest in the different positions given by Table 6.4.

Although a small correction can be observed after the calibration, mainly regarding the axis offsets of both sensors, the BMX055 provides a more stable response than the ICM-20498, where a noisier response is achieved for the latter. Nevertheless, both sensors have similar responses that are optimised after the calibration, meaning that the calibration procedure was validated.

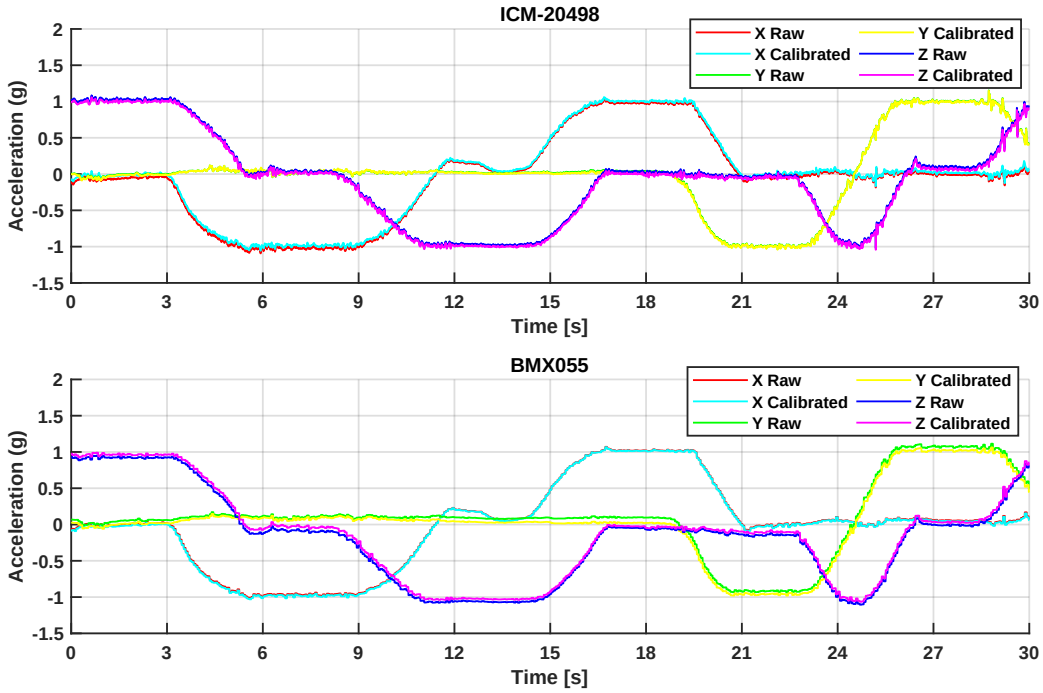


Figure 6.2: Measured accelerometer values with and without calibration for each referential axis.

The gyroscope calibration is based on precise angular velocities that are applied to a single axis, resorting to a precise turn table used to reduce the calibration errors. Since small angular velocities below 10 °/s are expected during the satellite pointing orientation mode, three different velocities were considered for the calibration: 2, 5 and 10 °/s, as shown in Table 6.5. An acquisition of raw angular velocities was done for each orientation and for each velocity, for at least 10 s and with a sampling rate of 100 Hz. To this data was applied a least square method, as previously done with the accelerometers, and the calibration parameters given by Equation 6.3 were calculated. After that, the final calibration matrices of each IMU sensor were calculated with average of the three matrices obtained for the three different angular velocities.

Table 6.5: Expected values for each gyroscope axis when rotating orthogonally to the Earth's surface.

Orientation	X axis	Y axis	Z axis
+X	+2, +5, +10 °/s	0	0
-X	-2, -5, -10 °/s	0	0
+Y	0	+2, +5, +10 °/s	0
-Y	0	-2, -5, -10 °/s	0
+Z	0	0	+2, +5, +10 °/s
-Z	0	0	-2, -5, -10 °/s

After calculating the gyroscope calibration values, a rotation test procedure was performed with each axis of the COMv2 board place orthogonally to the Earth's surface, to study the response of both gyroscopes available in the ICM-20498 and BMX055 sensors. In this test, whose results are depicted in Figure 6.3, an acquisition of raw data was done during the following scenarios, with a sampling frequency of 100 Hz:

- Board at rest;
- Board rotating precisely 90 ° in one axis with an angular velocity of 5 °/s;
- Board back at rest for 5 seconds;
- Board rotating precisely -90° in same axis back to the initial position, with -5 °/s of angular velocity.

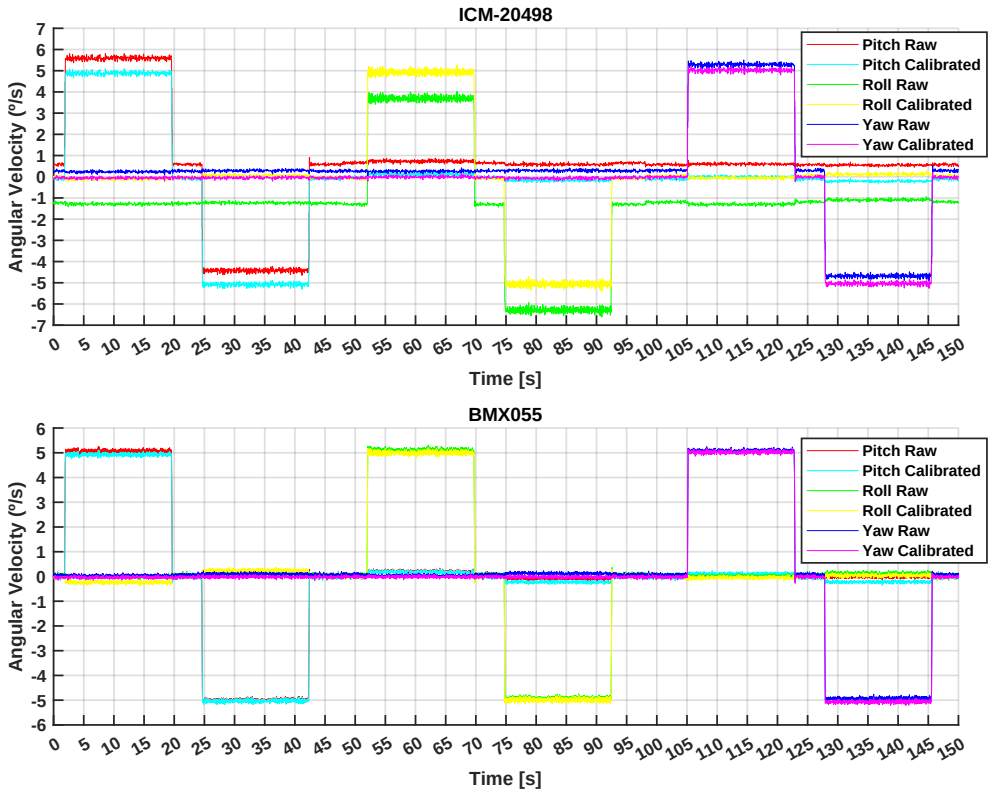


Figure 6.3: Measured gyroscope values with and without calibration.

After applying the calibration matrices to the measured values, the existent offsets were removed, specially on the ICM-20498 sensor, as the obtained angular velocities during the ± 90 º rotations become near the expected ± 5 º/s for the expected 18 s rotational time. However, a misalignment between the different axis still exists, which is noticeable during a rotation, specially on the BMX055 gyroscope. For example, during the pitch axis rotations on this sensor, a small deviation from the zero angular velocity occurs on the roll axis, and vice-versa. One way that might mitigate this misalignment is to repeat the calibration procedure for longer acquisition times at rest and for another angular velocities, acquiring a considerable amount of samples to recalculate the calibration matrices. Nevertheless, the results from this calibration are satisfactory, as the performance of both gyroscopes was improved.

The magnetometer calibration is based on the Earth magnetic field. Considering an ideal magnetometer, measured magnetic field vectors would have the same magnitude independently of the sensor orientation, meaning that if a rotation in the three axis is performed, the values obtained would create a spherical surface whose centre is zero and whose radius is the magnitude of this magnetic field. However, this does not happen with a real magnetometer, where soft iron distortions distort the circular output into an elliptical shape and hard iron distortions shift the centre of this new ellipse from the zero centre[2] [10] [19]. Furthermore, since the direction and magnitude of a magnetic field at a given location are hard to know, as the magnetic field vector differs within the location on the planet, the same procedure done for the previous sensors can not be applied directly.

During the development of this master thesis, a Helmholtz cage was being developed by the IST-nanosat team that would allow to isolate the space inside it from the external Earth magnetic field. Additionally, this cage would provide, upon the excitation of its coils, different precise magnetic field vectors, which might be used to apply the calibration method used in the previous sensors, since a well-known magnetic field vector could be applied to the magnetometers.

Since the Helmholtz cage is not completed yet, another coarse solution was used to perform the magnetometer calibration through a software developed by Yury Matselenak named *MagMaster*. This software is mainly designed for the calibration of magnetometers used on drones, where only the direction of the magnetic field is necessary for the attitude. Therefore, this software only calibrates this parameter, not providing the correct magnitude of the measured magnetic field.

For this calibration software to work, the COMv2 board must be positioned in 12 different orientations, while an acquisition of magnetic field measurements is applied to this software in order to calculate the calibration values of Equation 6.3. For example, considering the X axis, the sensor must be placed with this axis orthogonally to the surface of the Earth in both directions and in two different positions rotated 180º from each other, as shown in Figure 6.4. These orientations are then repeated for the remaining Y

[2]Magnetic measurements are subjected to either hard iron or soft iron distortions. The former is created by objects that produce a magnetic field, like a magnetised iron. The latter is considered as a deflection in the existing magnetic field, whose distortion depends on which direction the field acts relative to the sensor [19].

and Z axis orthogonally placed to the Earth's surface, ending the calibration procedure.

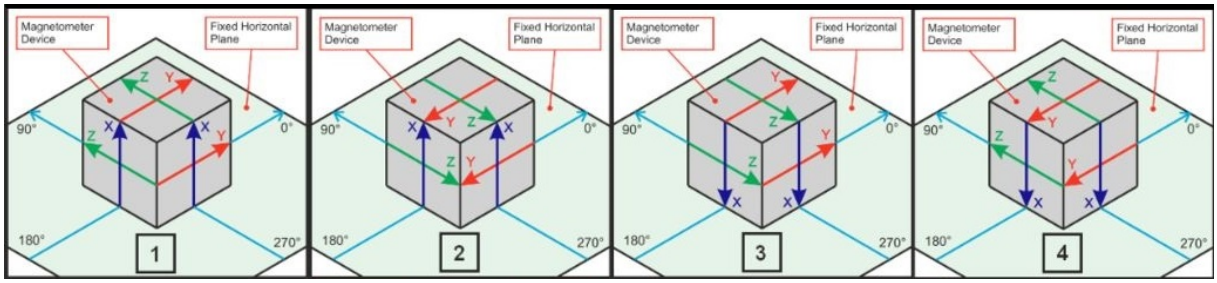


Figure 6.4: Orientations required for the magnetometer calibration[3].

The result of this calibration is depicted in Figure 6.5, where raw values for both ICM-20948 and BMX055 magnetometer (represented with blue dots) are firstly acquired using a spherical air bearing mechanism, where the COMv2 subsystem is freely rotating inside a sphere. As expected, the achieved result were elliptical shapes with an offset, which is considerably high on the ICM-20948 sensor. After the calibration, whose result is represented with green dots in Figure 6.5, the offset of both sensors was removed and the magnetic field directions obtained from both magnetometers were successfully calibrated.

Additionally to the software used, an approximation to the correct magnitude of the magnetic field vector was achieved using a geomagnetic field calculator based on the World Magnetic Model (WMM)[4]. According to this model, the magnitude of the magnetic field present in the IST Taguspark facility on the calibration day was 43.87 μT. After normalising the obtained calibration vectors and multiplying then by this magnitude, the result obtained is represented with red dots in Figure 6.5, where a spherical surface with no offset is achieved, meaning that the magnetometer was calibrated for this location.

This calibration procedure would be acceptable if these magnetometers were used on a single location on Earth, updating the magnitude of the magnetic field vector using the WMM as an approximation. However, this procedure is not valid for a satellite since the magnetic field vector is not constant during an orbit. Therefore, a proper calibration must be done using the previously discussed Helmholtz cage, also taking into account the integration of the COMv2 subsystem in a satellite, since the other subsystems and its metallic structure may influence the magnetic field measurements.

Additionally to the aforementioned IMU sensors used in the attitude determination, the performance of the sun sensor transimpedance amplifiers was also studied, comparing the illuminance levels obtained with the circuit design studied in Section 5.2.7 with the illuminance levels obtained with a ISO-TECH 1335[5] digital light meter. For that, the test photodiode used for the former and photo detector of the latter were placed next to each other on a stand rotating in a non-orthogonal axis, as shown

[3] https://github.com/YuriMat/MagMaster - Accessed on 2019/09/20.
[4] https://www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml - Accessed on 2019/09/20
[5] http://webx.ubi.pt/~smogo/disciplinas/alunos/manual-luximetro-ISOTEX-Lux1335.pdf - Accessed on 2019/09/20

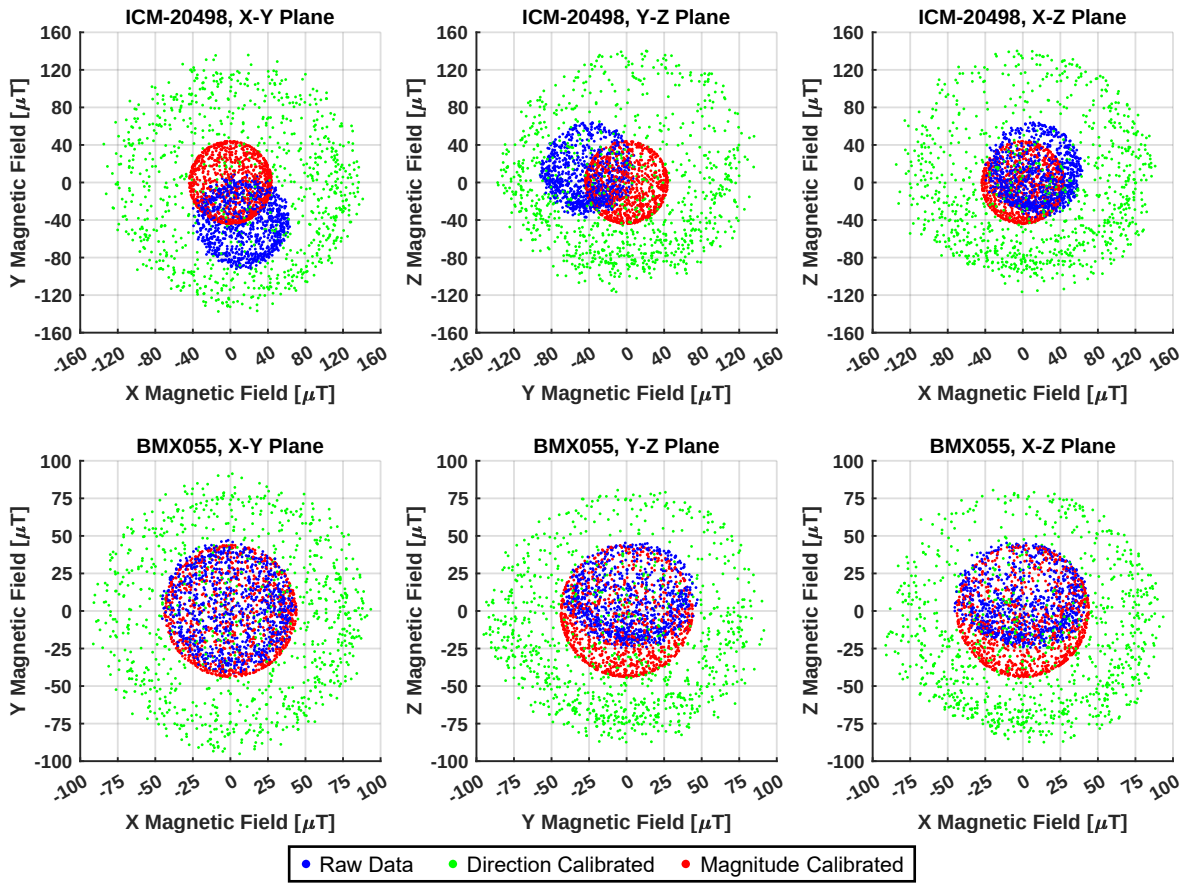


Figure 6.5: Obtained magnetometer values without calibration and after the direction and magnitude calibrations.

in Figure 6.6. This setup provides different incidence angles θ_i between these devices and a light source, where the maximum illuminance level is obtained with a zero incident angle. Furthermore, this rotating stand tries to simulate the cubesat faces where a sun sensor is available.

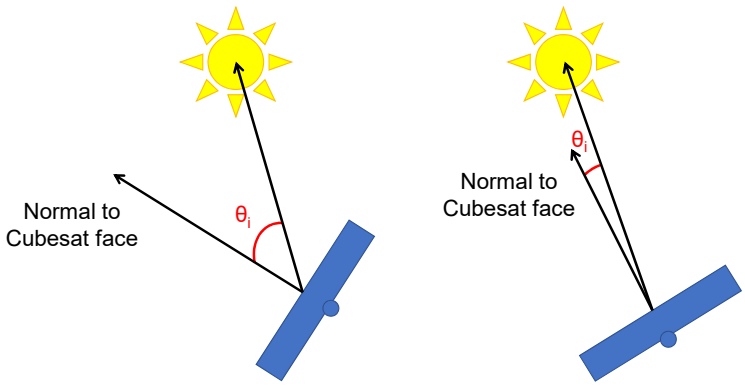


Figure 6.6: Different light incidence angles obtained with a rotating stand, providing different illuminance levels.

The sun at direct midday sunlight was chosen as the light source for the test scenario, simulating the satellite light source. Illuminance measurements are done with incident angles from -90 º to 90 º with 10 º steps, whose results are shown in Figure 6.7.

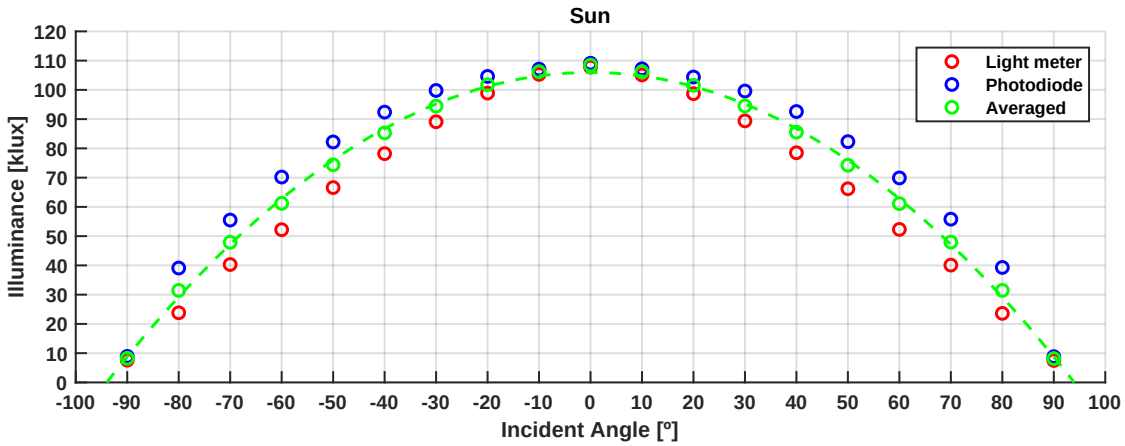


Figure 6.7: Illuminance levels obtained using the COMv2 sun sensor amplifiers, compared with a digital light meter.

As expected, the maximum illuminance level was obtained for an incident angle of 0 º, with similar values for both light sensors that are near the expected value of 100 000 lux, which is correspondent to direct sunlight illumination [16]. Although the measured characteristics when considering incidence angles between 0 º and $\pm$ 90 º are not linear, these characteristics can be fitted into a second order polynomial regression

$$L = a{\theta_i}^2 + b\theta_i + c \tag{6.4}$$

in order to achieve a direct correlation between the measured illuminance level L and the incident angle of the light source θ_i. This is particularly important for the attitude determination since the orientation of the satellite can be calculated when considering the sun as a reference. As a result, the polynomial regression resultant from the average of the two light sensors characteristics can be considered, which is also traced in Figure 6.7. Although two mathematical solutions exist for this polynomial regression for an illuminance value given by a single sun sensor, the satellite orientation may be achieved when every sun sensor placed on the solar panels is considered, using a matrix of the six measured illuminance levels on an attitude algorithm.

The results from Figure 6.7 were obtained considering the test photodiode SFH 2716, meaning that this process must be repeated for every sun sensor after the integration of the COMv2 subsystem in a satellite, since a different photodiode may be used as a sun sensor.

Finally, the validation of the magnetorquer drivers implemented on the COMv2 studied only the current delivery to a magnetorquer with certain a PWM driving signal, not focusing on an attitude control algorithm, which is out of the scope of this master thesis.

A magnetorquer is usually made of electromagnetic coils that include a small resistance. Taking into account its inductive part, a voltage applied to the magnetorquer

$$V = \frac{I}{R} + L\frac{\Delta I}{\Delta t}, \tag{6.5}$$

where L is the magnetorquer inductance and R its resistance, gives a current transient response

$$I = \frac{V}{R}\left(1 - e^{-tR/L}\right), \tag{6.6}$$

which must be taken into account in the attitude control for the magnetorquer actuation.

Due to the unavailability of the flight magnetorquers for this validation, their inductance was neglected in this first validation, whose test only focuses on resistive loads. Furthermore, the ISTnanosat team has decided that an actuation frequency of 10 Hz is used for the attitude control algorithm applied in the ISTsat-1, being this frequency also considered for the PWM signals used in the COMv2. Due to this low frequency, the aforementioned current response may be assumed as a square signal whenever a PWM is applied, which results in a current response similar to a resistive load. Nevertheless, this circuit must also be tested and accepted after the integration of the COMv2 subsystem in a cubesat, using the flight magnetorquers and verifying the current transient responses.

Considering a magnetorquer as a resistive load with resistance R, an average current

$$I = \frac{V}{R} \times D_c, \tag{6.7}$$

is obtained, where D_c is the PWM duty cycle. Recalling that a VCC_REG regulated voltage of 3.0 V is used to power the magnetorquer drivers, three different loads of 33 Ω, 150 Ω and 330 Ω were used to test this circuit. The current delivered to these loads was measured with duty cycle steps of 5 %, whose results are shown in Figure 6.8.

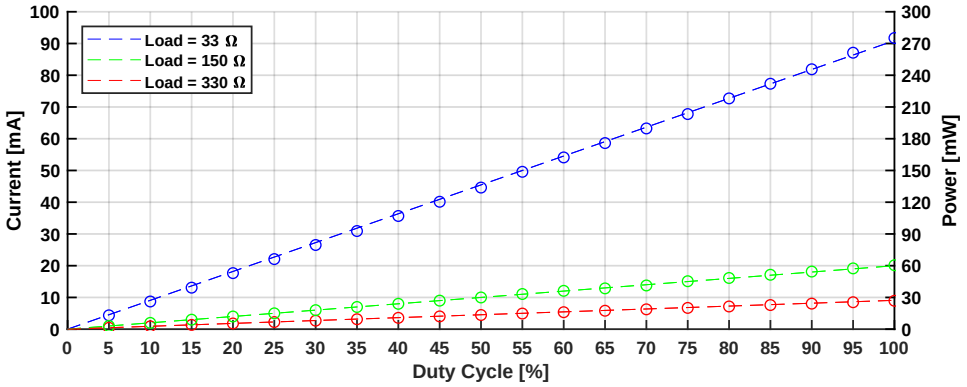


Figure 6.8: Current and power consumption of the magnetorquer drivers for different duty cycles and resistive loads.

The measured currents for each load follow the linear characteristic given by Equation 6.7. However, depending on the inductance of the flight magnetorquers used on the satellite and on the PWM actuation frequency, this characteristic may cease to be linear as these parameters become higher. Furthermore, average currents lower than the ones given by Equation 6.7 are expected as the duty cycle decreases, since transient current response might occur.

Chapter 7

Conclusion and Future Work

The communication systems present in a satellite are usually divided into two distinct functionalities: the management of the radio link with the Ground Segment and the processing of the communication stack transferred through this space link. Depending on the satellite structure, these functionalities may be separated into different subsystems. Furthermore, a high data rate may be required on this radio link depending on the satellite mission, which also demands the design of data storage hardware so that this data is not compromised while no communication with the GS can be done.

The main objectives set for this master thesis were the implementation and validation of two versions of the Communications Processor Board subsystem, being designed for the integration on cubesats developed by the ISTnanosat team. The first COM version is mainly responsible for the ISTsat-1 communication stack handling and for the satellite data storage when the satellite is in its normal operational mode. The second COM version is designed for an integration in the next generation of ISTnanosat team cubesats, having not only the aforementioned requirements, but also the responsibility for the attitude determination and control and for housekeeping of the satellite, managing its operation and the remaining subsystems.

The COMv1.1 subsystem is designed with a high power processing unit not only capable of processing the satellite complex communication stack, but also capable of being interconnected with a parallel memory unit. Using the microcontroller External Memory Controller and a decoder, this static memory unit with non-volatile and volatile memory chips is addressed sequentially, providing either byte, word or double-word addressing. The correct operation of this memory unit was validated with a test procedure that verifies not only the memories integrity but also the correct interconnection between this unit and the microcontroller. Additionally, a performance test was done to measure the memory bandwidth achieved with this design.

To protect the memory unit operation against the space environment, a radiation mitigation technique based on current measurement and memory power shutdown is implemented, where the power supply

of these memories is cut-off temporarily after detecting an high current situation caused by a Single Event Effect, preventing the permanent damage of the memory units. The circuitry implemented for this radiation mitigation technique was validated, being the current measurement circuits calibrated.

The COMv2 subsystem is designed with a modular architecture, having a motherboard plus an optional daughterboard. For the former, another high power processing unit is chosen to meet the new attitude determination and control and satellite housekeeping requirements, which are added to the requirements set for the COMv1.1 subsystem.

Regarding its data storage requirements, a compact serial memory unit is implemented with non-volatile memory, designed for both critical and mission data storage. Again, this memory unit was validated through a performance test that also measured the memory bandwidth achieved. Regarding the COMv2 optional daughterboard, a first prototype of a memory expansion daughterboard is designed with a similar implementation of the COMv1.1 memory unit, providing additional data storage capabilities. This memory unit was validated with the same performance test applied to the COMv1.1 memory unit, measuring the memory bandwidth achieved with this new design.

For the attitude determination requirements, this subsystem relies on two redundant IMU sensors, and is prepared for the interconnection with sun sensors and other IMU sensors available on solar panels, relying on transimpedance amplifiers for the former and a dedicated SPI interface for the latter. To improve the performance of the IMU and sun sensors, the correspondent calibration was done, meaning that these sensors are ready for the integration in an attitude algorithm. However, a rough calibration of the magnetometers was done, which must be repeated in a controlled scenario with a well-known magnetic field vector. Furthermore, all calibrations were done with the COMv2 subsystem isolated, meaning that these must be repeated upon its integration on a satellite, when the remaining subsystems and structures may influence the performance of these sensors, specially the magnetometers do to the influence of the metallic satellite structure.

For the attitude control, the COMv2 subsystem is designed with driving circuits for the magnetorquer coils available on solar panels, which are commanded with PWM signals. These magnetorquer drivers were validated by evaluating the current delivery to a resistive load rather than to the real magnetorquers, which could not be used due to their unavailability. This means that the acceptance of these magnetorquer drivers must be done again in the future, since testing with resistive loads may not be representative of the real flight magnetorquers.

Both COM versions rely on the PC/104 cubesat standard for the satellite system bus, with two redundant I^2C main buses used to communicate with the remaining subsystems. These buses are protected with dedicated I^2C buffers to prevent that a hazardous event does not block the bus operation. To ensure that the communication stack data flow in the satellite is not compromised, a dedicated SPI interface with the TT&C subsystem is also implemented in both COM versions.

Additionally, the COMv2 subsystem is also designed with a CAN bus, which may be considered for the main system bus in the future ISTnanosat team satellites, together with the aforementioned I^2C buses. Two asynchronous serial communication interfaces thought for payload data are also implemented. All these interfaces were validated with a performance test, measuring the throughput achieved when transmitting data to another device.

As in the COMv1.1, the COMv2 subsystem is populated with power cut-off switches and current measurement circuits to protect the several implemented modules against the space radiation environment. This circuitry was also validated, being the several current measurement circuits calibrated.

A similar system software is implemented for both COM subsystems to provide the necessary abstraction layers between the hardware developed and an application running in these subsystems. This system software provides an efficient way of interacting with the hardware of these subsystems through a Hardware Abstraction Layer that uses an API well-defined by the ISTnanosat team, along with the use of FreeRTOS to easily meet the real-time requirements of these embedded systems. Furthermore, this system software allowed the hardware implemented to be validated in a simpler way.

Overall, the objectives set for this master thesis were reached. Nevertheless, some improvements and tests must be done to finish the characterisation of the COMv2 subsystem, in addition to the ones previously described that regarded the attitude determination and control:

- Redesign and validate the circuit used to keep a time reference for an orbit as implemented in the ISTsat-1 OBC, since the designed circuit failed to meet its requirements.
- Implement and validate the USART peripheral, that may replace the SPI interface used for the dedicated communication interface with the TT&C.
- Implement and validate the debug SD Card interface. Although this optional interface was not set as a requirement for the COMv2, it was implemented on its prototype and may provide additional mass storage that may be required for the future ISTnanosat satellites.
- Characterise the power consumption with the different modules operating or shutdown, providing an overall characterisation of the subsystem power consumption.
- Perform an Electromagnetic Compatibility test to characterise the electromagnetic emissions of this subsystem, as the parallel memory unit design in the daughterboard may be a good emitting source while operating.
- Integrate the application software being developed by the ISTnanosat team into the subsystem, assuring the correct operation of its hardware. This regards not only the housekeeping and communication stack requirements but also the attitude algorithm regarding the satellite attitude determination and control.

After this, the COMv2 will be ready for the integration on a future ISTnanosat satellite, following the steps of its predecessor COMv1 integrated in the ISTsat-1.

Bibliography

[1] ESA. (2018, October) Fly Your Satellite! programme. Accessed on 23 December 2017. [Online]. Available: http://www.esa.int/Education/CubeSats_-_Fly_Your_Satellite/Fly_Your_Satellite!_programme

[2] Cubesat. (2016, August) History of the CubeSat. Accessed on 26 December 2017. [Online]. Available: http://www.spacedaily.com/reports/History_of_the_CubeSat_999.html

[3] T. Carvalho, "ISTnanosat-1 High Power Processor," Master's thesis, Instituto Superior Técnico - Universidade Técnica de Lisboa, January 2013, accessed on 28 September 2017.

[4] Nanosats Database. (2017, December) What is a CubeSat & other picosatellites. Accessed on 27 December 2017. [Online]. Available: https://www.nanosats.eu/cubesat

[5] InetDaemon. (2018, May) Satellite Subsystems. Accessed on 25 January 2019. [Online]. Available: https://www.inetdaemon.com/tutorials/satellite/satellite_subsystems.shtml

[6] NASA. (2018, December) State of the Art of Small Spacecraft Technology. Accessed on 20 January 2019. [Online]. Available: https://sst-soa.arc.nasa.gov/

[7] M. Swartwout. (2019, February) CubeSat Database. Accessed on 20 February 2019. [Online]. Available: https://sites.google.com/a/slu.edu/swartwout/home/cubesat-database

[8] P. Coelho, "ISTNanosat-1 Quality Assurance, Risk Management and Assembly, Integration and Verification Planning," Master's thesis, Instituto Superior Técnico - Universidade Técnica de Lisboa, May 2016, accessed on 09 October 2017.

[9] F. Sturesson. (2009, June) Single Event Effects (SEE) Mechanism and Effects. Accessed on 08 September 2017. [Online]. Available: https://docplayer.net/65061561-Single-event-effects-see-mechanism-and-effects.html

[10] R. Afonso, "Sistema de Gestão e Determinação de Atitude do ISTNanosat-1," Master's thesis, Instituto Superior Técnico - Universidade Técnica de Lisboa, October 2016, accessed on 28 December 2018.

[11] GOMSpace. (2019, September) Cubesat Products - Communication systems, Command & Data Handling systems. Accessed on 30 September 2019. [Online]. Available: https://gomspace.com/shop/subsystems/default.aspx

[12] ISIS. (2019, September) Cubesat Products - Communication systems, Command & Data Handling systems. Accessed on 30 September 2019. [Online]. Available: https://www.isispace.nl/products/

[13] Clyde Space. (2019, September) Satellite Bits - Communication systems, Command & Data Handling systems. Accessed on 30 September 2019. [Online]. Available: https://www.aac-clyde.space/satellite-bits

[14] NXP Semiconductors, *LPC185x/3x/2x/1x User Manual*, 5th ed., March 2016. [Online]. Available: https://www.nxp.com/docs/en/data-sheet/LPC185X_3X_2X_1X.pdf

[15] NXP Semiconductors, *LPC546xx User Manual*, 2nd ed., October 2018. [Online]. Available: https://www.nxp.com/docs/en/data-sheet/LPC546XX.pdf

[16] National Optical Astronomy Observatory. (2015, December) Recommended Light Levels. Accessed on 20 April 2019. [Online]. Available: https://www.noao.edu/education/QLTkit/ACTIVITY_Documents/Safety/LightLevels_outdoor+indoor.pdf

[17] M. Barr, *Programming Embedded Systems in C and C++*, 1st ed. O'Reilly, January 1999.

[18] M. Barr, "Software-Based Memory Testing," *Embedded Systems Programming*, July 2000, pp. 28-40.

[19] VectorNav Technologies. (2019, August) Sensor Calibration. Accessed on 29 August 2019. [Online]. Available: https://www.vectornav.com/support/library/calibration

[20] STMicroelectronics. (2014, June) AN3182 Application note - Tilt measurement using a low-g 3-axis accelerometer. Accessed on 31 August 2019. [Online]. Available: http://www.bdtic.com/download/ST/AN3182.pdf

Appendix A

ISTsat-1 OBC architecture

This Appendix serves the purpose of summarising the hardware architecture of the On-Board Computer to be integrated in the ISTsat-1, since this system highly relates with the COMv2 subsystem developed in this master thesis, as some implementations of the latter rely on the former. This OBC was firstly developed by Ruben Afonso, whose work is detailed in [10], with small modifications done afterwards by the ISTsat-1 team in the final iteration integrated in the ISTsat-1 cubesat. This subsystem is mainly responsible for the satellite housekeeping and attitude determination and control. The block diagram from Figure A.1 represents the hardware architecture of this On-Board Computer.

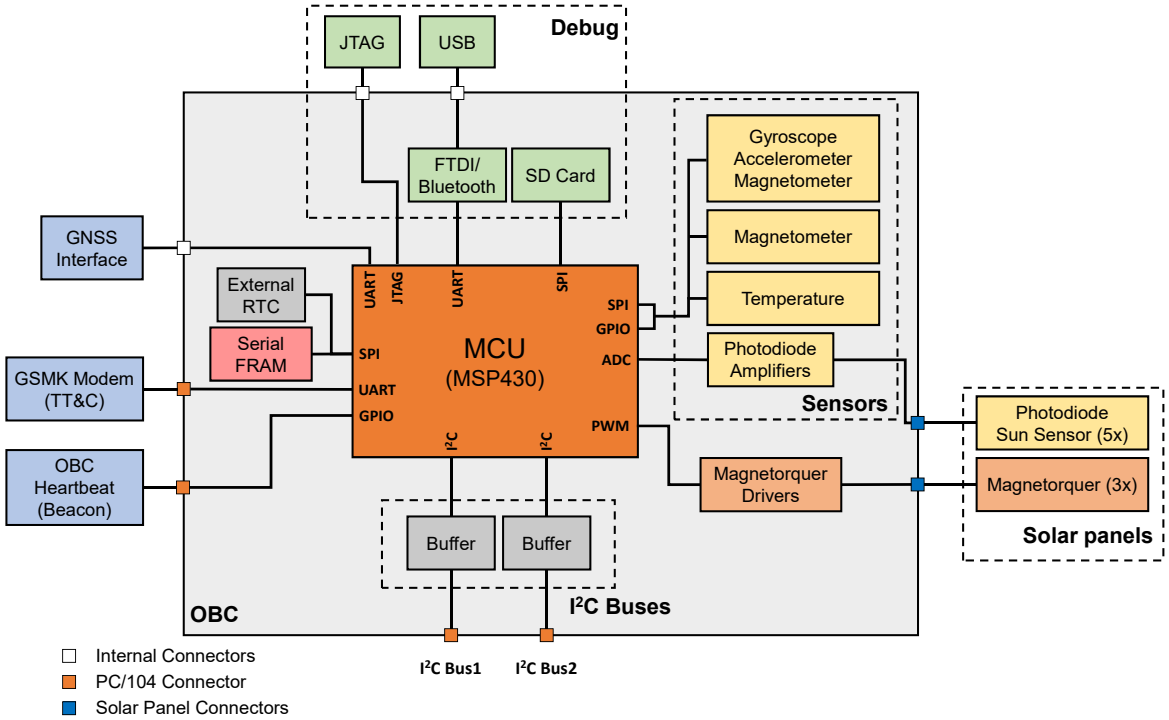


Figure A.1: ISTsat-1 On-Board Computer subsystem hardware architecture [adapted from [10]].

Starting by the ADCS funcionalities, the first thing considered in the design of this subsystem was the sensor selection for the attitude determination, where an IMU sensor with three sensor types was used: a gyroscope, an accelerometer and a magnetometer. In addition to this IC, an additional magnetometer is used to provide redundancy and better accuracy in the magnetic field measurements. Additionally, a dedicated temperature sensor is also used to accurately determine the temperature inside the spacecraft. These three circuits are interconnected to the main processing unit by a SPI interface [10]. Resorting to photodiode sun sensors present in the solar panels, transconductance amplifiers are also used to amplify their signal into ADC channels, as studied in Section 5.2.7.

Regarding actuators for the control system, PWM signals are used to drive three magnetorquer drivers, that then connect to the magnetorquers on the solar panels through specific connectors, as described in Section 5.2.8.

The communication with the other subsystems in the spacecraft is done through the two standard I^2C interfaces. External I^2C buffers are also used, disconnecting each interface of the microcontroller from the main bus if this MCU tries to block these interfaces [10]. Additionally, the OBC can communicate with the TT&C through a dedicated UART interface to exchange information related with the communication stack, mainly used during the satellite operation in safe mode, where the COMv1.1 subsystem is shutdown. Additionally, a GPIO pin connected to the Beacon is used to inform the OBC operation status.

Regarding the debug of the subsystem, a programming JTAG interface is used for the microcontroller programming and debugging, and a serial/USB converter is added to allow the system to communicate easily with any computer, plus a Bluetooth module that enables the system to be used remotely. For mass storage, SD card storage may also be used.

The MSP430F5438A from Texas Instruments is the chosen microcontroller for the OBC processing unit, with a 16-bit architecture, 25 MHz of operating frequency, 256 kB of Flash and 16 KB of RAM [10]. Additionally, a 512 MB serial FeRAM memory with a SPI interface is used for critical data storage.

Although the microcontroller chosen for the OBC has an embedded Real-Time Clock, this MCU can not keep a time reference in case of a subsystem power shutdown like thought in the design of the COMv2 subsystem. Therefore, a PCF2123[1] from NXP is used as an external RTC to keep this time reference during at least one orbit of approximately 90 minutes, as this is also a requirement set for the ISTsat-1 OBC. This integrated circuit is accessed through the same SPI interface used for the aforementioned FeRAM.

Following the same typology described in Section 5.2.5, this external RTC is powered by a capacitor bank[2] and a diode with a low reverse current as shown in Figure A.2. This circuit holds the power supply voltage required to keep the time reference during at least one orbit, during a power shutdown.

[1] https://www.nxp.com/docs/en/data-sheet/PCF2123.pdf - Accessed on 15/03/2019

[2] Since common batteries like coin-cells are not allowed to be used by ESA without proper qualification, the team decided to use a capacitor bank instead.

According to the PCF2123 datasheet, its typical current consumption when not being addressed by its SPI interface is 350 nA with a supply voltage V_{RTC} of 3.0 V. Additionally, the minimum voltage $V_{RTC_{min}}$ required for the RTC to keep its time reference is 1.1 V. Considering a capacitor bank discharge time of an orbit of 90 minutes and that the V_{RTC} is initially powered by 3 V, its capacitance given by

$$C = \left(\frac{I_{RTC}}{V_{RTC_{max}} - V_{RTC_{min}}} \right) t_{orbit} \tag{A.1}$$

must be greater than 995 uF to meet this requirement, considering a constant current discharge.

Analysing possible ceramic and tantalum capacitors available in the market, eight 220 uF tantalum capacitors are considered, providing a parallel capacitor bank of 1760 uF. This oversized value is implemented to ensure that the requirement is met even if an higher current consumption or different maximum and minimum voltages occur. Additionally, to limit the in-rush current on the capacitor bank when charging, a resistor of 100 Ω in series with the aforementioned diode is used.

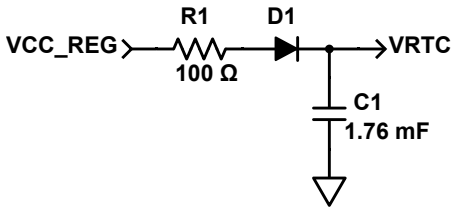


Figure A.2: Capacitor bank circuit used to keep the RTC time reference over an orbit.

The obtained capacitor bank operation is shown in Figure A.3, whose discharge meets the 90 minutes requirement, as the power supply voltage V_{RTC} is greater than the 1.1 V minimum operating voltage of the external RTC for more than 5400 s correspondent to an orbit.

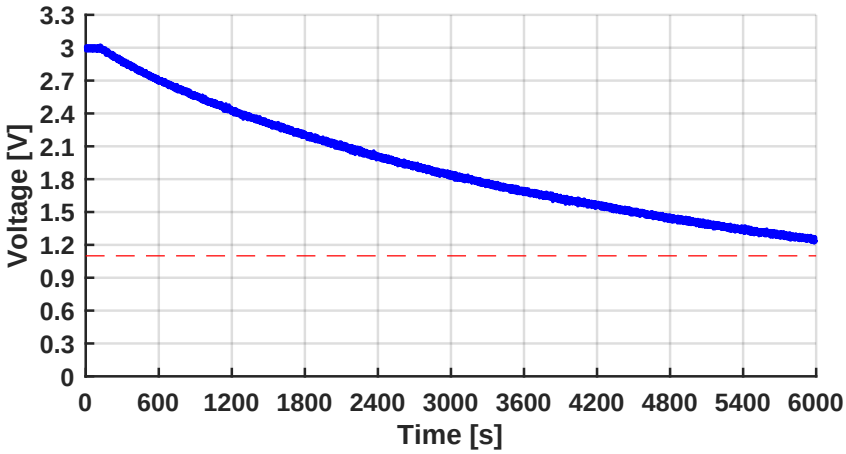


Figure A.3: Discharge of the OBC time reference capacitor bank, with the RTC minimum operating voltage.