

Evolution of the Communications Processor Board for the ISTnanosat

João Carlos Fonseca Pinto

Instituto Superior Técnico - Universidade de Lisboa, Av. Prof. Dr. Cavaco Silva, 2744-016 Porto Salvo

Phone: +351918361224, e-mail: joao.fonseca.pinto@tecnico.ulisboa.pt

Abstract—This paper 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 ISTsat-1 from the ISTnanosat team, 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, being also considered for the satellite command and data handling and attitude determination and control. Therefore, possible architectures for them are discussed in this paper, describing the final hardware and software architectures. These architectures take into account the harsh space environment that these subsystems will be subjected to, as some radiation mitigation and subsystem level redundancy techniques are addressed as well. Finally, some validation tests are implemented for the acceptance of the developed subsystems.

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.

I. INTRODUCTION

Since the launch of the first artificial Earth satellite into orbit, the use of artificial satellites has increased, specially in the last years. These satellites are used in applications like communications, scientific, educational and military purposes. Due to the harsh space environment, the necessary quality assurance to ensure a high-reliability satellite system is highly expensive. However, technology evolution and satellite standardisation has decreased the cost of producing a satellite.

One of these standardisations is the Cubesat, a low cost 10 cm cube satellite that has allowed educational space studies to be done without a great expense. To aid these educational space activities, the Fly Your Satellite! (FYS) programme from the ESA has the objective of inspiring university students to work in the space sector [1]. Currently participating in the FYS is the ISTsat-1, an 1U cubesat developed by the ISTnanosat team. In addition to the educational purpose, the ISTsat-1 mission carries out a feasibility study of the use of satellites to receive ADS-B signals used in aircraft monitoring, collecting relevant data from aircraft and temporally store it to be further forwarded to the Ground Segment (GS).

The purpose of this paper is to describe the evolution of the Communications Processor Board developed for the ISTnanosat, focusing on the implementation and validation of its two versions. The first version COMv1.1 integrates on

the ISTsat-1 and is responsible for its communication stack handling and data storage. The second version COMv2 will integrate on the next generation of ISTnanosat cubesats, being also responsible for satellite attitude determination and control and satellite housekeeping. Both COM subsystems must fulfil the cubesat standards, which restrains their dimensions and power consumptions.

II. SATELLITES CATEGORISATION AND SPACE ENVIRONMENT - ISTSAT-1 CASE STUDY

The satellites built throughout the years vary significantly in their size, as their systems share common functionalities like communications, power management, satellite attitude determination and control, satellite housekeeping and data handling. The use of small satellites has increased and reduced the missions cost, as satellites are built cheaper and faster.

One of these small satellites is the ISTsat-1 participating on the FYS programme from ESA. This nanosatellite follows a typical structure of a cubesat, whose architecture is shown in Figure 1. Its subsystems are:

- The **Tracking, Telemetry and Control (TT&C)**, responsible for the data uplink (UHF) and downlink (VHF) between the satellite and several ground stations.
- The **On-Board Computer (OBC)** responsible of keeping the satellite well-positioned and in orbit and for the monitor and control of the satellite subsystems, being the satellite main controller.
- The **Electrical Power System (EPS)**, responsible for the power management and energy harvesting.
- The **Communications Processor Board (COM)**, responsible for the communication stack handling and the satellite data storage.
- The **ADS-B Payload**, responsible for receiving ADS-B signals broadcasted by commercial aircraft.

Unlike the protected environment in the Earth atmosphere, the space environment damages electronic components with radiation coming from emission sources in the solar system. Total Ionising Dose (TID) and Single Event Effects (SEE) may characterise the impact of this radiation. The former is measured as the cumulative radiation received by the spacecraft (S/C), while the latter results from high energy particles that hit the spacecraft electronic components, changing momentarily their operation or causing permanent damage [2]. A description of important SEEs is done in [3].

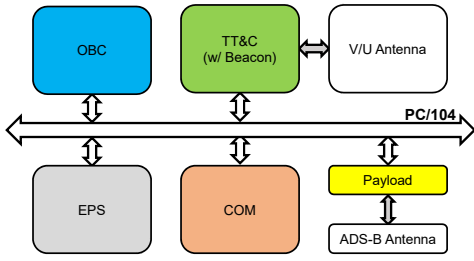


Fig. 1. Block diagram of the ISTsat-1 architecture.

Radiation mitigation techniques are important in the satellite design to reduce the severity of this radiation. Although commercial off-the-shelf devices are easy sale-available components, this low cost options are not designed to withstand the space radiation environment. However, identical radiation hardened devices may exist, although they bring an impact to the cost and mass budget of the S/C. Additional measures to mitigate the radiation effects are the use of microcontroller monitoring and power cut-off circuits, allied with overcurrent and communication protections [4]. These mitigation techniques make a space-based system fail-proof even in an inaccessible radiation environment.

III. STATE OF THE ART

From the main functionalities existent in satellites, the satellite communications, the command and data handling and the attitude determination and control are relevant for the COM subsystems studied in this paper. For that reason, communication systems used in nanosatellites are further summarised, along with both C&DH and ADCS modules.

A. Communication systems

A communication system enables the transmission of both telemetry and payload data from a satellite to the GS, while receiving commands from its stations. This communication is commonly based in the radio frequency spectrum, using frequencies from 30 MHz up to 40 GHz. 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, using higher frequency bands in small satellites is becoming a trend, enabling higher data rates. Small satellites launched to LEOs mainly use low gain antennas for their communication systems, taking advantage of the short orbit distance. Another tendency on small satellites is the implementation of Software Defined Radios (SDR), usually based on FPGAs that allow multiple radio frequency bands with different filtering and modulation schemes [4].

B. Command and Data Handling (C&DH)

A satellite relies on a system to control its correct operation, manage its subsystems and store relevant data to be sent to the GS, compressing and encoding it into a known communication protocol stack. Therefore, the C&DH subsystem requires high power processing and storage capabilities.

The use of low power microcontrollers and FPGAs can be observed on recently cubesats, supporting different processor cores, providing typical serial communication peripherals, on-chip memories and improved low power performances with fast code executions. Additionally, different memory technology types offer wide capacity ranges, although the main requirement for satellite memory units is reliability. A study of the most used volatile and non-volatile memory types and their performances may be seen in [4]. For short cubesat missions, the memory retention time and operation cycles are not important but its power consumption and susceptibility to radiation must be considered.

C. Attitude Determination and Control (ADCS)

Depending on its mission, a satellite is placed into a specific orbit but its position can deviate from it due to space perturbations. The ADCS is used to monitor and act in the satellite attitude, positioning it constantly in its orbit. Furthermore, a satellite can move itself in orbit depending on its status and purpose at a specific time. Therefore, an ADCS module can put a satellite in detumbling mode or on a target, inertial or nadir pointing, as shown in Figure 2.

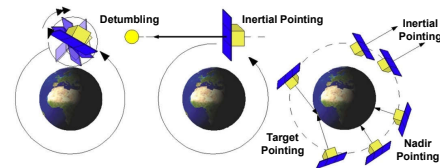


Fig. 2. Orientation modes of the satellite attitude determination and control.

Allied to these orientation modes, passive and active stabilisation techniques are used. The accuracy of passive control systems, like the gravitational and magnetic gradients, is lower than the active ones, like the double rotational technique or the momentum cancellation, being only an option to consider when the satellite resources are limited, like in a cubesat [5].

As the name implies, an ADCS module is divided into the attitude determination, relying on sensors to check the satellite position, and into the attitude control, relying on actuators to intentionally achieve the desired satellite attitude. Between these two groups is a processing unit, which collects the data from sensors and calculates the necessary corrections to the satellite attitude before acting on its actuators.

Sensors are a requirement for the attitude determination, obtaining continuously accurate angular velocities of the satellite. Their technology has evolved in the last years, as electronic devices are more reliable, small and available in MEMS devices, like magnetometers and gyroscopes. These are viable options for small satellites due to their size and reduced cost, although the star orientation method with a camera offers more accuracy. Similarly, the satellite attitude control requires actuators to control it. In a small satellite, this control is usually done with a single actuator due to its size and power consumption. Some options are magnetorquers, reaction wheels or gas propulsion [5].

D. Commercial modules for cubesats

A study of commercial modules available in well-known companies of the cubesat market (GOMspace, ISIS and AAC Clyde Space) was done concerning the previously studied communications, C&DH and ADCS functionalities, in order to evaluate different architectures, as these functionalities are differently divided between several subsystems.

For example, the GOMspace combination of its On Board Computer (OBC) plus its V/U communications module provides a solution where the three functionalities are offered on a single cubesat board. Another solution from ISIS separates the communications module into another subsystem, keeping the C&DH and optional ADCS functionalities on its OBC. Finally, the AAC Clyde Space solution separates the three functionalities into three separate subsystems. Furthermore, different architectural solutions exist for the OBC, whether this system is designed on a single board or has a daughterboard attached to a main board, either with the main processing unit or with another ADCS, data storage or mission functionalities.

IV. SYSTEM ARCHITECTURE

Since the architecture of a satellite subsystem is directly related with the requirements that must be complied, the requirements for both COM versions are listed further, considering their permanent and secondary functionalities.

A. COMv1.1 architecture

The COMv1.1 is responsible of processing the ISTsat-1 communication stack, decoding and encoding all data in the space-link, requiring for that a high-power processor. To support the storage of both telemetry and mission data, an external memory unit is also considered. The requirements for the COMv1.1 arise as:

Permanent functions:

- Decode/encode the satellite communication stack.
- Provide data storage for telemetry/payload data.
- Apply compression techniques to this data to decrease the data flow on the space-link.
- Monitor the occurrence of SEEs.
- Assume the satellite housekeeping, after an OBC permanent failure (satellite backup mode).

Secondary functions:

- Store new telemetry/housekeeping data after receiving it.
- Store new payload data after receiving it.
- On memory storage fill-up, decide which data is discarded if no more compression can be done.
- Act accordingly to the occurrence of SEE.
- Reset its blocked processor after a SEE/software glitch.

Input/Output functions:

- Use two I²C buses for main communication.
- Provide a private communication interface with the TT&C dedicated to the satellite communication stack.
- Provide an heartbeat signal for the satellite housekeeper.
- Report the memory unit state to the satellite housekeeper.
- Enter a low-power mode after a GS command.

The COMv1.1 must also comply with the cubesat PC/104 standard for integration in the ISTsat-1, being as power efficient as possible and obeying to its mechanical specifications.

B. COMv2 architecture

After a study of different possible architectures for the COMv2 subsystem, an architecture with a fixed motherboard and an optional daughterboard was chosen, where the former has the processing unit, a small memory unit for minimal storage capabilities and the sensors and driver actuators required for the ADCS functionalities, allowing the daughterboard to have different possible designs. From this architecture, the requirements for the COMv2 arise as:

Permanent functions:

- Determine the S/C attitude through sensor measurements.
- Manage the correct operation of the attitude sensors.
- Be the satellite housekeeper.
- Decode/encode the satellite communication stack.
- Provide data storage for telemetry/payload data.
- Apply compression techniques to this data to decrease the data flow on the space-link.
- Monitor the occurrence of SEEs.

Secondary functions:

- Generate PWM signals for the magnetorquer drivers.
- Be able to disable the ADCS functions when not needed.
- Store new telemetry/housekeeping data after receiving it.
- Store new payload data after receiving it.
- On memory storage fill-up, decide which data is discarded if no more compression can be done.
- Act accordingly to the occurrence of SEE.
- Reset its blocked processor after a SEE/software glitch.
- Keep track of time for one orbit (90 minutes) after a power shutdown.
- Be able to put its processor in a low-power mode.

Input/Output functions:

- Use two I²C buses for main communication.
- Provide a CAN bus for alternative main communication.
- Provide a private communication interface with the TT&C dedicated to the satellite communication stack.
- Receive GS commands and dispatch them to the remaining subsystems.
- Provide satellite attitude information to the GS.
- Report the state of its modules to the GS.
- Enter a low-power mode after a GS command.

V. SYSTEM IMPLEMENTATION

The components and modules chosen to achieve the previously discussed COM architectures are studied in this section, concerning the hardware and software functionalities of each module. In the hardware project, the logical connections between the different modules are explained, focusing on the power regulation, control and distribution. Finally, the software implementation that manages the subsystem peripherals is also detailed.

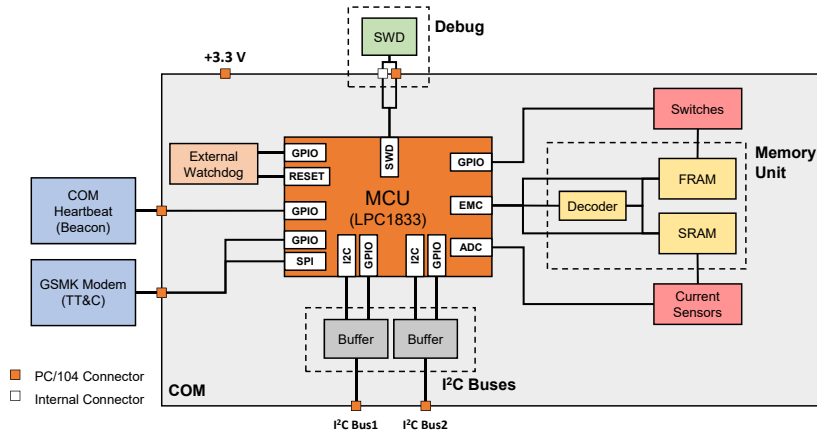


Fig. 3. Block diagram of the COMv1.1 hardware architecture.

A. COMv1.1 Hardware project

To support the explanation of the COMv1.1 hardware modules, the block diagram from Figure 3 represents an overview of the hardware architecture of this subsystem.

The required high-power processing unit for this subsystem is based on a LPC1833 microcontroller (MCU) from NXP Semiconductors, an 180 MHz ARM Cortex-M3 with 136 kB and 512 kB of internal RAM and Flash memory, respectively. The choice of this processing unit was done on an unfinished iteration COMv1.0 by Tiago Carvalho [6]. This microcontroller is chosen due to its low power consumption and its External Memory Controller (ExtMC) capable of doing double word parallel addressing.

Regarding the power supply of the subsystem, all the COMv1.1 is directly powered by a regulated 3.3 V power line provided by the EPS.

Although the chosen MCU has an internal watchdog timer (WDT), an external one like the TPS3823A from Texas Instruments is considered, providing redundancy and acting when a SEE or software glitch prevents the operation of the internal one. Considering its watchdog timer function, this integrated circuit provides an input signal that must be toggled by the MCU under a 2 s timeout period, otherwise the MCU is reset through its reset line.

An external memory unit was also planned for the unfinished iteration COMv1.0, where Static RAM (SRAM) was used as volatile memory and Flash as non-volatile memory [6]. In the new COMv1.1, Ferroelectric RAM (FeRAM) is used instead of Flash due to its higher robustness to radiation, keeping SRAM as volatile memory. The obtained memory unit comprises four Cypress CY62177EV30 SRAM memories with 4 MB each, plus eight Cypress FM22L16 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.

The chosen microcontroller ExtMC provides several signals used to address static parallel memory like the ones chosen, namely its 32 data signals, 24 address signals, 4 chip

selects (CS) and write/output enable signals. This controller is capable of doing either byte, word or double word addressing, chosen through its 4 byte-lane select (BLS) signals, meaning that the chosen memories must be addressed in pairs in order to provide a double word addressing, where the lower BLS signals and 16 data lines connect to one memory, and the upper BLS and 16 data lines connect to the other memory.

This new requirement of addressing pairs of memory leaves six pairs of memory to be addressed separately: two SRAM pairs plus four FeRAM pairs. The Texas Instruments SN54HC139-SP external decoder is considered to select which pair of memory is addressed by the microcontroller at a certain address. Connected to this dual 2-line to 4-line decoder, qualified for space usage, are two CS signals plus the address signals required from the ExtMC so that all memory unit may be addressed continuously. The correct operation of this decoder was confirmed with simple access test to each memory pair, obtaining a write access time of 105 ns and a read access time of 205 ns.

With this decoder, the memory unit became fully operational for the six memory pairs. To decrease its susceptibility to radiation, a mitigation technique relying on current sensing and power cut-off is implemented, with two current sensors for each memory type and six switches for each memory pair. This implementation is important so that an occurrence of a SEE causing a high-current condition does not damage these memories permanently. The hardware implemented in this radiation protection as a switch and as a current sensor is shown in Figure 4.

On its left, a single P-channel power MOSFET is used for each memory pair as its switch, being driven by a GPIO line from the microcontroller, which controls if the 3.3 V subsystem power supply is provided to the correspondent memory pair. A pull-up resistor R_1 is used to keep this switch closed while the microcontroller enable line is at a high impedance state, mainly during the startup sequence.

On its right, a current sensor made by a shunt resistor R_2 and an operational amplifier in a non-inverting configuration is implemented for each memory type.

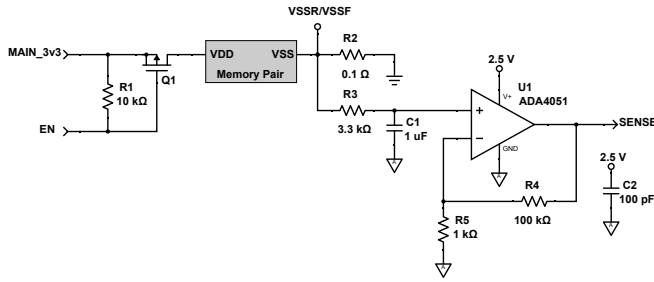


Fig. 4. COMv1.1 power cut-off switch and current sensor.



Fig. 5. COMv1.1 final design board.

In this configuration, a current to voltage sensor gain

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

is achieved. Additionally, a low-pass filter with a cut-off frequency of 50 Hz is added with R_3 and C_1 , so that the high frequency current spikes in the memories are filtered before being acquired by the correspondent ADC channel. Finally, a stable voltage reference of 2.5 V is also used to power the MCU ADC channels and the current sensors. Since the power consumption of each memory type differ, different gains are used for each current sensor, where the SRAM current sensor is designed with a gain of 10 V/A and the FeRAM sensor with a gain of 20 V/A. The correct operation of these circuits was confirmed as the desired gains with a linear characteristic were achieved, being the current sensors also calibrated.

As communication peripherals, two I²C interfaces are chosen for the satellite main communication bus. Additionally to this redundancy, the LTC4303 bus buffer from Linear Technology is used on both buses, mitigating a stuck bus situation made by a software or hardware problem. This buffer automatically separates the main satellite bus from the microcontroller if any of the I²C data or clock lines becomes low for more than 30 ms, being the connection automatically reestablished when the bus is again ready for communication.

A dedicated SPI interface is implemented between the subsystem responsible for the space-link to exchange the satellite communication stack, where the TT&C is the master node and the COMv1.1 is the slave node. Additionally, to inform the former that new data has to be transmitted to the GS, an additional RTS signal is used by the latter so that the former knows when a SPI transfer is necessary.

Finally, a GPIO pin is used to provide a heartbeat signal to the satellite housekeeper subsystem, whose state is toggled within a heartbeat period. A programming SWD interface is used for programming and debugging the microcontroller.

A final COMv1.1 board, shown in Figure 5, has been designed with the previously defined modules. This PCB, with six layers and a size of 93.0 mm x 86.4 mm, respects the cubesat standard and will be integrated in the ISTsat-1 satellite.

B. COMv2 architecture

To support the explanation of the COMv2 hardware modules, the block diagram from Figure 6 represents an overview of the hardware architecture of this subsystem.

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, a study of microcontrollers from a selection of well-known manufacturers is done. Considering the architecture set for this subsystem with a motherboard and optional daughterboards, the LPC54616 microcontroller from NXP Semiconductors is chosen for the main board, delivering an 180 MHz ARM Cortex-M4 with 200 kB and 512 kB of internal RAM and Flash memory.

The same external WDT considered for the COMv1.1 is used to reset this microcontroller after SEEs or software glitches.

The use of parallel memory for this subsystem is not considered in its motherboard, as parallel memory requires a lot of PCB space to maintain a reasonable data storage capacity, being this memory type replaced with serial memory. To assure that the storage of critical data is not compromised, serial FeRAM is considered, with two 512 kB Cypress CY15B104Q, which are accessed through a common SPI interface. Allied to this critical data storage, a 8 MB Cypress S25FL064L serial Flash memory is considered for normal data storage, interconnected to the MCU through a quad SPI interface. Additionally to this serial storage, an optional memory expansion daughterboard is considered, which has the same design as the COMv1.1 memory unit previously described. The addressing of this memory is also achieved by the static ExtMC and external decoder used to select the memory pairs, with the same radiation mitigation architecture.

Two different power sources are considered for the power supply of the COMv2. The first is a regulated VCC_COM power supply bus of 3.3 V provided by the EPS. The second is a BACKUP power supply that connects directly to the satellite solar panels, considered to keep the COMv2 powered up during an EPS failure. This BACKUP voltage may vary between 0 V and 4.65 V depending on the solar exposure. Therefore, an additional power regulation circuitry is needed to provide a 3.3 V power supply to the COMv2 subsystem from this BACKUP bus, which is shown in Figure 7.

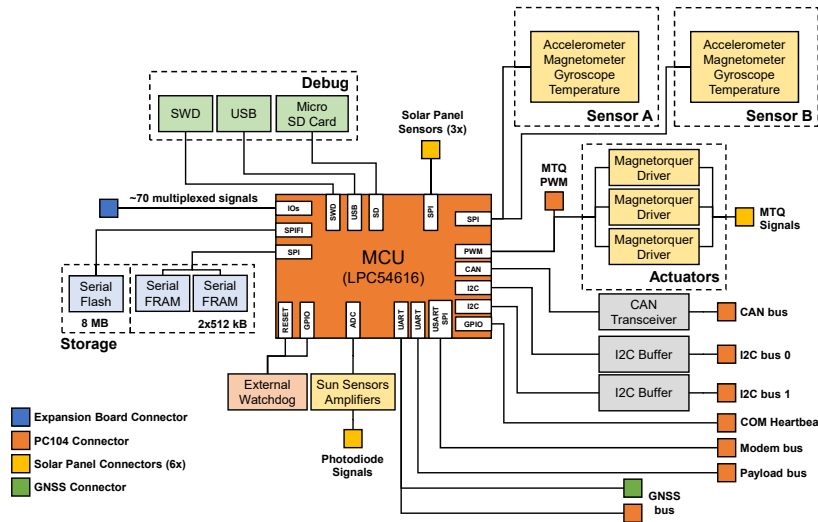


Fig. 6. Block diagram of the COMv2 hardware architecture.

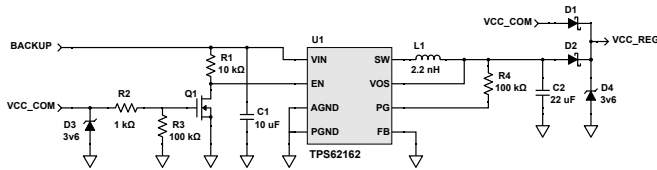


Fig. 7. Regulation circuit for the BACKUP power supply bus, implemented in the COMv2.

The TPS62162 step-down converter from Texas Instruments is used following the manufacturer recommendations, providing a 3.3 V output power supply. Since this circuit must work only when the VCC_COM power supply is not provided, two Schottky diodes D_1 and D_2 are connected in series to each power supply, setting a regulated power supply VCC_REG of 3.0 V. Additionally, a control circuit connected to its ENABLE pin is used to shutdown the converter when the VCC_COM power supply exists, and to start it when VCC_COM is not available. The correct operation of this circuit was validated, resulting on an instantaneous transition from the BACKUP to VCC_COM power supply, but with a temporary shutdown of 270 ms when switching from the VCC_COM to the BACKUP power supply.

The power consumption of the several subsystem modules is verified with a power cut-off switch and a current sensing circuit, as shown in Figure 8. On the left, a power cut-off switch is implemented with a P-channel MOSFET in series with the regulated power supply VCC_REG, plus a N-channel MOSFET that controls the former transistor through a GPIO line provided by the microcontroller. These modules stay without power during the MCU boot sequence due to the resistors R_1 and R_3 . On the right, a current sensing circuit made by a shunt resistor R_4 and a current shunt monitor INA139 from Texas Instruments is implemented. With this

configuration, a current to voltage sensor gain

$$G = 0.001 \times R_4 R_5, \quad (2)$$

is achieved. When this circuit is powered with the VCC_REG 3.0 V, the output voltage of this current sensor is limited to 2.3 V due to the internal design of the INA139. Additionally, a low-pass filter of 150 Hz 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. The gains of each current sensor are directly controlled by the output resistor R_5 , being three different gains used in this subsystem for maximum currents of 1 A, 300 mA and 100 mA. The correct operation of this circuit was confirmed as the desired gains with linear characteristics were achieved, being the current sensors also calibrated.

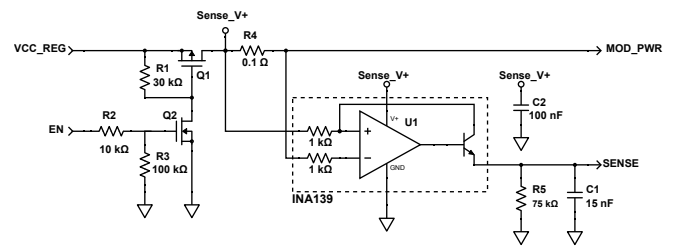


Fig. 8. Power cut-off switch and current measurement circuit for the COMv2 modules.

The satellite attitude determination is based on a redundant 9 Degrees of Freedom system with a gyroscope, an accelerometer and a magnetometer, used to characterise the satellite orientation. From a selection of several small and low cost MEMS sensors in the market, the InvenSense ICM-20948 and the Bosch BMX055 sensors are considered as the sensors used for the attitude determination. Both provide the three sensor types in a 3 axis reference system plus an integrated

temperature sensor, used to determine the temperature inside the spacecraft. To access them, a single SPI interface is chosen, operating in a single master with multiple slaves approach.

Additionally to these MEMS sensors, the COMv2 is prepared with a SPI interface for additional sensors on the solar panels. Furthermore, transimpedance amplifiers with a feedback resistor and capacitor are also used to amplify the current of six photodiode solar sensors, available in the solar panels, into ADC channels, as these sensors are also required for the attitude determination. The component selection and circuit validation was done considering an OSRAM SFH 2716 photodiode.

Regarding the actuators used for the attitude control system, PWM signals are considered to drive three magnetorquers present in the solar panels, one for each axis of the satellite, using three DRV8837 H-bridge drivers from Texas Instruments. This driver operates with two PWM signals that control the current flow through the magnetorquers like in a H-bridge.

As implemented in the COMv1.1, the COMv2 relies on two I²C interfaces with the same LTC4303 bus buffer for the the satellite main communication bus. Additionally, the same SPI+RTS interface is considered for the communication stack transfer between the TT&C, keeping this subsystem as the master node and the COMv2 as the slave node.

However, a new CAN interface is introduced as a feasibility study to complement or replace the aforementioned I²C interfaces, requiring a CAN transceiver driver 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. Furthermore, two UART interfaces are considered, one for a Global Navigation Satellite System (GNSS) used for a more accurate satellite positioning, and another as a dedicated interface between a payload subsystem, relieving the main I²C and CAN buses. Again, a SWD interface is used for programming and debugging the subsystem.

A prototype COMv2 board, shown in Figure 9, has been designed with the previously described modules. This PCB, with six layers and 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 also been designed, being also shown in Figure 9.

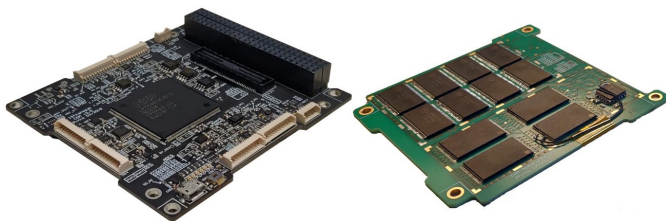


Fig. 9. COMv2 (left) and its memory expansion (right) prototype boards.

C. Software architecture

The software developed for both COM subsystems and further described in this paper is called as system software, which provides the necessary abstraction layers between the hardware previously described and an application running on these subsystems.

Although different COM versions are implemented, the system software stack developed is similar, which is shown in Figure 10. 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 proprietary ARM and NXP software files. Above this layer is the high level layer of this HAL, which provides an understandable API, developed by the ISTnanosat team, to work with the required microcontroller peripherals. Board drivers that resort to this HAL is then designed to access the different hardware modules present in the COM subsystems. Board configuration files are also 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. 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.

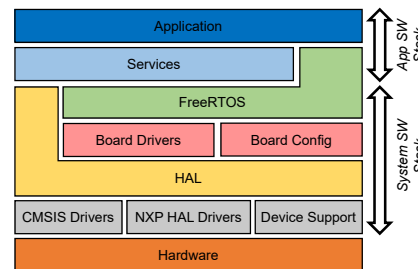


Fig. 10. Software stack implemented on both COM subsystems.

A similar boot sequence is implemented in both COM versions, mainly dealing with the initialisation of the required MCU peripherals and hardware modules for each subsystem. 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 used plus a development environment created for the ISTsat-1 test and integration, being based on a set of *makefiles* that select the platform and project to be compiled and run on the several ISTsat-1 subsystems.

VI. SYSTEM VALIDATION

The validation of the designed modules is performed considering their different functionalities rather than studying both subsystems separately. For that, the validation of both parallel and serial external memory units, communication peripherals and attitude determination and control modules is considered.

The memory units validation is done differently in parallel and serial memories. The validation of the parallel memory unit is done resorting to three memory tests developed by Michael Barr, detecting either electrical connection problems, missing memory chips or improperly inserted memories. The first and second tests are based on a *walking 1's* test done to test separately the parallel data and address buses, detecting either an open or short circuit connection on any of the data wires. The third test is based on an *increment* test used to ensure the integrity of the memory devices, testing if every bit in a device is capable of holding both logical values. These tests are explained in detail in [7] and [8]. Being the whole memory unit correctly addressed through its decoder, the aforementioned third test was run to ensure the integrity of the whole parallel memory units of both COMv1.1 and COMv2 memory expansion board, with an 100 % successful rate after 1000 test repetitions, meaning that these memory units are fully operational.

Considering the serial memory unit implemented on the COMv2 subsystem, the access to its memories is mostly based on commands sent by the microcontroller through the correspondent serial peripherals, usually followed by a response from the memory. For the validation of this memory unit, a memory test based on a buffer with a fixed size and random data is done, 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. After 1000 repetitions, an 100 % successful rate was achieved for both FeRAM and Flash, meaning that the serial memory unit is fully operational.

A performance test was done to each memory unit to evaluate the memory bandwidth, whose results are shown in Table I. The memory bandwidth is determined for both read and write accesses, where a buffer with a predefined size is written to and read from a memory several times, measuring the total time elapsed separately for all operations. These times are measured using the internal microcontroller timers, doing an average of 10 test repetitions. Therefore, a memory bandwidth

$$Bw = \frac{B_{size} \times N}{t_{av}} \quad (3)$$

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 due to longer write access commands, being the write memory accesses slower than the read accesses. This difference is more evident with

TABLE I

PERFORMANCE COMPARISON BETWEEN IMPLEMENTED MEMORY UNITS, WITH THE MEMORY BANDWIDTHS OBTAINED.

Memory Type	B _{size} (Bytes)	N	t _{av} * (ms)		Bw (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 32-bit COMv1.1	16384	100	260	164	6.28	9.95
Parallel 16-bit COMv1.1	16384	100	521	329	3.14	4.98
Parallel 8-bit COMv1.1	16384	100	1042	659	1.57	2.49
Parallel 32-bit COMv2	16384	100	154	94	10.66	17.44
Parallel 16-bit COMv2	16384	100	307	205	5.33	7.99
Parallel 8-bit COMv2	16384	100	581	376	2.82	4.36

Flash due to the time waiting for the erase of the Flash sectors. 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. In parallel memory, the fact that write accesses are faster than read accesses is confirmed, with a higher write memory bandwidth than the read one. Finally, an increase of the memory bandwidth occurs when using the parallel memory expansion daughterboard on the COMv2 subsystem, since its microcontroller is operating at a higher frequency and has a better ARM architecture, improving the memory access times.

The validation of the communication interfaces implemented on both COM subsystems starts with a *ping* test between these subsystems and another device with these interfaces fully operational (*Teensy 3.5*). During this test, bugs detected on the developed HAL are fixed, improving the robustness of the communication peripherals. A performance test is then done to measure the throughput of each peripheral, where messages with a predefined size are written to a single device several times, measuring the total time elapsed for all transactions. These times are measured using the internal microcontroller timers and doing an average of 10 test repetitions. Therefore, a throughput

$$Th = \frac{B_{size} \times N}{t_{av}} \quad (4)$$

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 II, where different operating frequencies or baud rates are used.

For a fixed operating frequency, both SPI and I²C interfaces have a higher throughput when transferring buffers with a higher size, since the overhead per transfer and latency between transfers are reduced. However, the same does not happen for the UART interface, whose overhead is not

TABLE II
PERFORMANCE COMPARISON BETWEEN IMPLEMENTED SERIAL PERIPHERALS, WITH THE THROUGHPUTS OBTAINED.

Protocol	Freq/Baud	B _{size} (Bytes)	N	t _{av} * (s)	Th (kB/s)
SPI+RTS	500 kHz	8	4096	0.76	42.99
	500 kHz	300	256	1.63	47.01
I ² C	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

dependable of the buffer size. Comparing both I²C and CAN peripherals, 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 size increases in the I²C interface. 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.

Concerning the attitude determination a prior calibration of the chosen inertial sensors and sun sensors is done. For the former, the sensor model used for the calibration may be seen in [9], where the calibrated values are computed with measured raw values, from which a constant bias offset is removed, multiplied by a matrix of the axis misalignment errors, existent due to the non-orthogonality of the three sensor axis, and by the matrix of the scale factors between the measured and expected values.

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 of acceleration in that axis. The gyroscope calibration is based on precise angular velocities of 2, 5 and 10 %s that are applied to a single axis, resorting to a precise turn table used to reduce the calibration errors. For these calibrations, an acquisition of 10 s with a sampling frequency of 100 Hz is done, applying a least square method to obtain the calibration parameters, as detailed in [10]. The magnetometer calibration is based on the Earth magnetic field, using a software developed by Yury Matselenak named *MagMaster*, designed for the direction calibration of magnetometers used on drones [11]. Then, a coarse magnitude calibration is done considering the World Magnetic Model calculator to normalise the obtained magnitude field vectors to a magnitude of 43.87 μ T, present at IST Taguspark on the day of calibration.

A tilt test, a rotation test and an air bearing test were performed to study the response of the ICM-20498 and BMX055 sensors after the three calibrations, which are shown in Figure 11, Figure 12 and Figure 13, respectively.

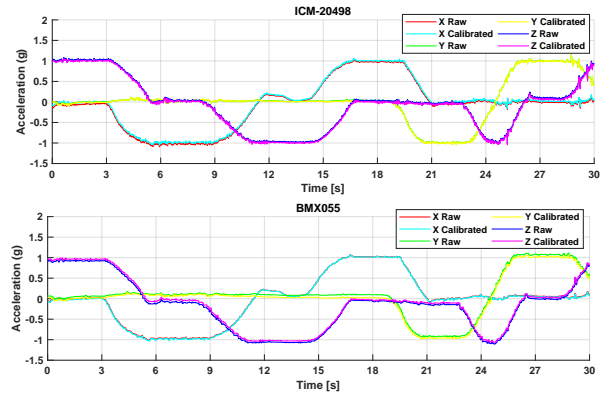


Fig. 11. Accelerometer tilt test values with and without calibration.

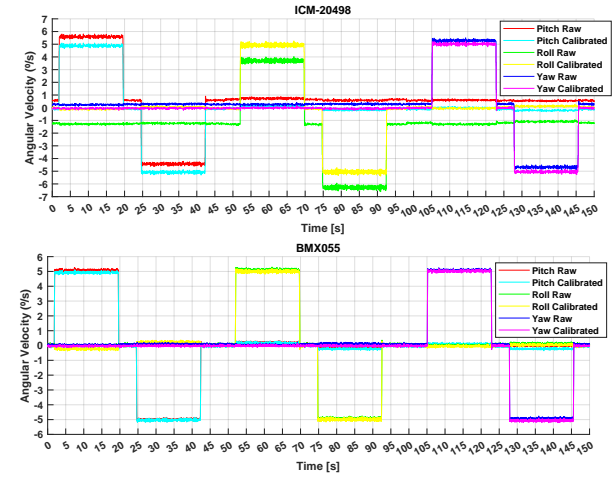


Fig. 12. Gyroscope rotational test values with and without calibration.

A small correction is observed after the accelerometer calibration, with the BMX055 providing a more stable response than the ICM-20498. On the gyroscope calibration, the existent offsets were removed, specially on the ICM-20498 sensor. However, a misalignment between the different axis is still observed during a rotation, specially on the BMX055 gyroscope. Finally, the magnetometer offsets were removed after the direction calibration done with the *MagMaster*, and a spherical surface with no offset is achieved after the magnitude calibration using the WMM considering the IST Taguspark location. Overall, the calibrations were satisfactory but the magnetometer one must be repeated with a known magnetic field vector on a Helmholtz cage being designed by the ISTnanosat team.

The performance of the sun sensor transimpedance amplifiers was also studied, comparing illuminance levels obtained with the designed circuit with a ISO-TECH 1335 digital light meter. The two photodetectors were placed next to each other on a stand rotating in a non-orthogonal axis, providing different light incidence angles. The sun at direct midday sunlight was chosen as the light source for the test scenario, whose results are shown in Figure 14.

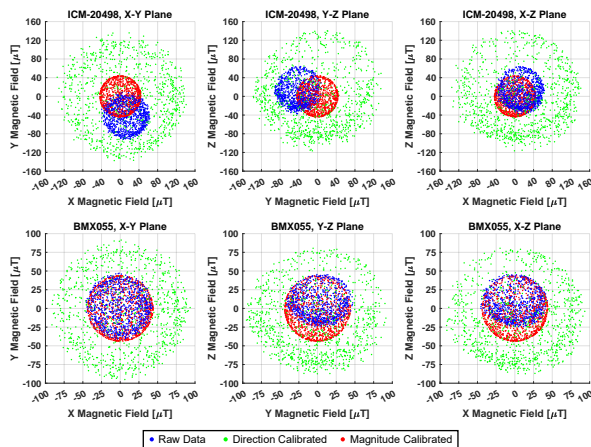


Fig. 13. Magnetometer air bearing test values with and without calibration.

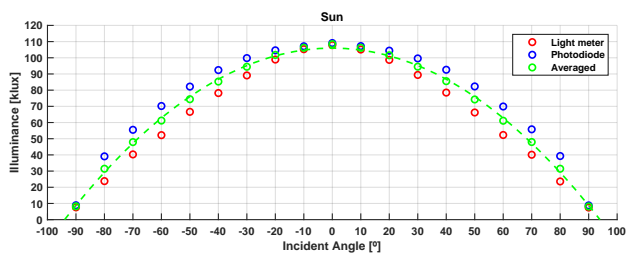


Fig. 14. Performance of the sun sensor amplifiers and a digital light meter.

An expected maximum illuminance level near 100 klux was obtained for an incident angle of 0°. Additionally, a direct correlation between the measured illuminance levels and the light incident angle is achieved with a second order polynomial regression, calculated from the average of the two light sensors characteristics. The satellite orientation may be then 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.

A study of the current delivery to a resistive load is done to validate the magnetorquer drivers, since the real magnetorquers were not available. Considering a magnetorquer as a resistive load with resistance R , an average current

$$I = \frac{V}{R} \times D_c, \quad (5)$$

is obtained, where D_c is the PWM duty cycle. Three different loads of 33 Ω , 150 Ω and 330 Ω were used to test this circuit considering a supply voltage V_{CC_REG} of 3.0 V, whose resultant currents followed the linear characteristic given by Equation 5 for different duty cycles.

VII. CONCLUSION AND FUTURE WORK

Overall, the main objectives set for this work were achieved, with the implementation and validation of two versions of the COM subsystems.

The first COM version is designed with a high power processing unit and a parallel memory storage unit, both protected with radiation mitigation techniques. All these modules were validated with performance and calibration tests, as the COMv1.1 is ready for the integration in ISTsat-1.

The second COM version is designed with a modular architecture comprising a motherboard and an optional daughterboard. The main board keeps a high power processing unit but provides a serial memory storage unit upgradable with an optional parallel memory expansion daughterboard. Both memory units were validated with performance tests.

Additionally, a set of sensors and actuator drivers are implemented in the motherboard to provide the required attitude determination and control functionalities. A prior calibration to these sensors was done, whose results were valid. However, some coarse assumptions were made, mainly on the magnetometer, whose calibration is only valid for the IST Taguspark location and not for a LEO, and on the magnetorquer drivers, since only resistor loads were used, neglecting the inductive part that exists on magnetorquer coils. Furthermore, since these results were obtained with this COMv2 isolated, the sensor and actuator drivers performances must be evaluated again on a flight model, considering the influence of the satellite remaining subsystems and structure.

Finally, additional future work prevails to finish the characterisation of the COMv2, namely the redesign of a circuit used to keep a time reference for an orbit, the implementation of a SD Card interface used for mass storage, and the power consumption and thermal characterisations of this subsystem.

REFERENCES

- [1] ESA, "Fly Your Satellite! programme", May 2018. [Online]. Available: http://www.esa.int/Education/CubeSats_-_Fly_Your_Satellite/Fly_Your_Satellite!_programme. [Accessed: 23-Dec-2017].
- [2] P. Coelho, "ISTNanosat-1 Quality Assurance, Risk Management and Assembly, Integration and Verification Planning", Instituto Superior Técnico - Universidade Técnica de Lisboa, May 2016. [Accessed: 09-Oct-2017].
- [3] F. Stursson, "Single Event Effects (SEE) Mechanism and Effects", June 2009. [Online]. Available: <https://docplayer.net/65061561-Single-event-effects-see-mechanism-and-effects.html>. [Accessed: 08-Set-2017].
- [4] NASA, "State of the Art of Small Spacecraft Technology", December 2018. [Online]. Available: <https://sst-soa.arc.nasa.gov/>. [Accessed: 20-Jan-2019].
- [5] R. Afonso, "Sistema de Gestão e Determinação de Atitude do ISTNanosat-1", Instituto Superior Técnico - Universidade Técnica de Lisboa, October 2016. [Accessed: 28-Dec-2018].
- [6] T. Carvalho, "ISTNanosat-1 High Power Processor", Instituto Superior Técnico - Universidade Técnica de Lisboa, January 2013. [Accessed: 28-Sep-2017].
- [7] M. Barr, "Software-Based Memory Testing", Embedded Systems Programming, pp. 28-40, July 2000.
- [8] M. Barr, "Programming Embedded Systems in C and C++", 1st ed., O'Reilly, January 1999.
- [9] VectorNav Technologies, "Sensor Calibration", August 2019. [Online]. Available: <https://www.vectornav.com/support/library/calibration>. [Accessed: 29-Aug-2019].
- [10] STMicroelectronics, "Tilt measurement using a low-g 3-axis accelerometer", June 2014. [Online]. Available: <http://www.bdtic.com/download/ST/AN3182.pdf>. [Accessed: 31-Aug-2019].
- [11] Y. Matselenak, "Easy Hard and Soft Iron Magnetometer Calibration", June 2014. [Online]. Available: <https://www.instructables.com/id/Easy-hard-and-soft-iron-magnetometer-calibration/>. [Accessed: 01-Sep-2019].