

# Development of an Autonomous Redundant Attitude Determination System for Cubesats

Ricardo O. Duarte<sup>1</sup>, Samuel R. C. Vale<sup>1</sup>, Luiz S. Martins-Filho<sup>2,\*</sup>, Fernando. E. Torres<sup>1</sup>

## How to cite

Duarte RO  <https://orcid.org/0000-0001-7904-9074>

Martins-Filho LS  <https://orcid.org/0000-0002-7287-5979>

Torres FE  <https://orcid.org/0000-0002-7708-5538>

Duarte RO; Vale SRC; Martins-Filho LS; Torres FE (2020) Development of an Autonomous Redundant Attitude Determination System for Cubesats. J Aerosp Technol Manag, 12: e3020. <https://doi.org/10.5028/jatm.v12.1166>

**ABSTRACT:** This work deals with the problem of the design of an autonomous redundant system for the attitude determination of nanosatellites. The system consists of a circuit board equipped with three microcontrollers, a magnetic field sensor in three axes, and complementary commercial-of-the-shelf (COTS) components and connectors to provide data signal exchange the vehicle on-board computer. The main goal of this system, named SDATF (of the acronym in Portuguese *Sistema de Determinação de Atitude com Tolerância a Falhas*), is to provide autonomously the vehicle attitude from the information collected from the onboard computer (OBC) and the magnetometer. The adopted attitude determination algorithm is based on the well-known QUEST method, and the goal of this system is to get accurate attitude computations to low-orbit CubeSat satellites using COTS electronic components, and to provide highly reliable data and high availability levels using fault tolerance tools to avoid the harmful consequences of spatial radiation and its faults known as single event upsets (SEU). The article presents the general design of the fault-tolerant systems and experimental tests using bit flip injection methodology.

**KEYWORDS:** Artificial satellites; Attitude control; Embedded software; Fault tolerance; State estimation.

## INTRODUCTION

This paper describes fundamental aspects of the development of a CubeSat nanosatellite attitude determination system with triple redundancy and fault tolerance tools. The system, named SDATF (acronym from the Portuguese name *Sistema de Determinação de Atitude com Tolerância a Falhas*), calculates the attitude parameters that describe the vehicle spatial orientation. The algorithm adopted for attitude determination is the QUEST method, which considers quaternions as parameters, and was tested and compared with other methods before this choice (Duarte *et al.* 2009).

The SDATF has a planned in-flight test on a mission of the satellite NanosatC-BR2, in the context of a program of the Brazilian National Institute for Space Research (INPE, Instituto Nacional de Pesquisas Espaciais), shown in Fig. 1, where it will fly as an experiment payload, but without responsibility on the attitude determination and control strategy of this CubeSat (Garcia *et al.* 2018; Schuch *et al.* 2014).

1. Universidade Federal de Minas Gerais – Escola de Engenharia – Belo Horizonte/MG – Brazil

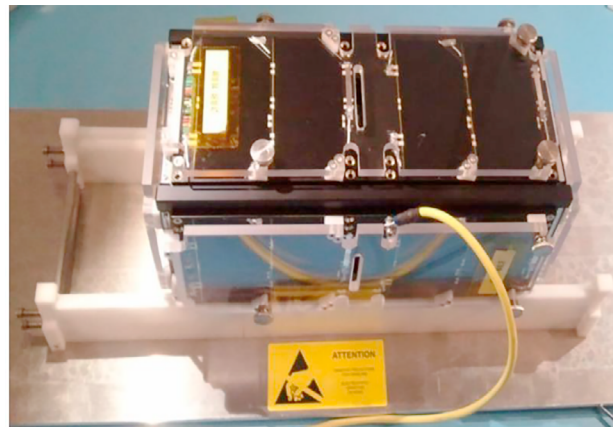
2. Universidade Federal do ABC – Centro de Engenharia, Modelagem e Ciências Sociais Aplicadas – São Bernardo do Campo/SP – Brazil

\*Correspondence author: [luiz.martins@ufabc.edu.br](mailto:luiz.martins@ufabc.edu.br)

Received: Nov. 27, 2019 | Accepted: Mar, 16, 2020

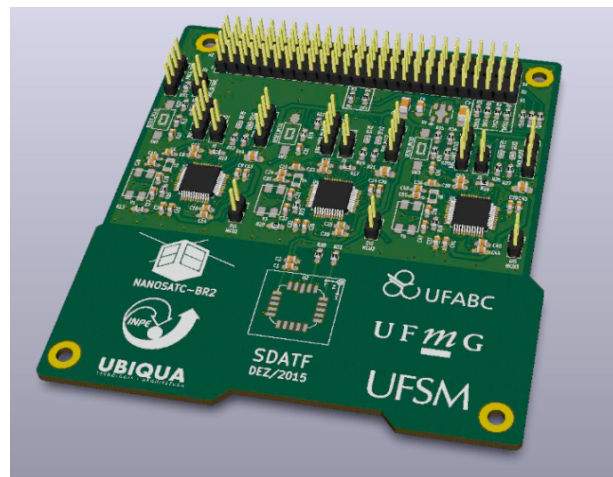
Section Editor: Alison Moraes





**Figure 1.** Engineering model of the NanosatC-BR2 satellite.

The primary objective of this system is to provide the satellite attitude with maximum precision from nonspatially qualified electronic components or commercial-of-the-shelf (COTS) components. A secondary goal, no less important than the first one, is to provide the attitude with maximum reliability and availability during the mission, protecting the system from harmful effects of space radiation. The main COTS devices in the system are three microcontrollers STM32F303CC ARM Cortex – M4F and one three-axis magnetometer XEN 1210. In addition to them, other components complete the system electronics. Figure 2 shows an image of the SDATF board design with the arrangement of its components.



**Figure 2.** The upper face of SDATF board design.

The integrated circuit memory cells are subject to errors due to the impact of energetic particles, such as heavy ions, neutrons and protons, when exposed to space radiation (Mukherjee 2008), leading to temporary errors and unwanted behavior. Such errors are known as single-event effects (SEE) and can be destructive, single-event latchups (SEL), single-event transients (SET), single-event upsets (SEU). This paper focuses on SEU events (Wang and Agrawal 2008).

Excessive radiation loads neutralize the logical state of the volatile memory cells of COTS devices. This phenomenon is logically perceived as an inversion of a bit of memory, which may affect the application, causing a system lock or a harmful error in the calculated attitude values, compromising system reliability and availability (Wang and Agrawal 2008). These SEU type errors are temporary failures affecting mainly the volatile memory of integrated circuits. The faults may go unnoticed by the application if the algorithm does not use at that time the specific location of the affected memory, or write other

information there and the error caused by the fault disappears. In the opposite case, the error affects the calculation results, compromises attitude determination and can cause problems in the satellite control.

Some studies report the occurrence of disorders related to this type of event in space vehicles and stratospheric balloon missions (Harboe-Sorensen *et al.* 1990; Hubert *et al.* 2014). For instance, during 600 days of the ROSAT satellite operation, 40 SEUs were detected (Sims *et al.* 1993). Other works address the development of instrumentation to simulate in a laboratory the incidence of the phenomenon in integrated circuits (Velazco *et al.* 2000; Faure *et al.* 2005), as well as engineering solutions proposals to overcome the problem (Duarte *et al.* 2011; Kastensmidt *et al.* 2006). However, a methodology to do this is systematically used to prove that the proposed solution is effective.

This paper seeks to fill a gap of studies involving methodologies to evaluate the effectiveness of solutions to avoid errors associated with SEU events. The main contribution is the detailed description of choices made in the SDATF autonomous system engineering process. Preliminary fault tolerance tests using bit flip injection in the microcontrollers testify the proposed strategy adequacy.

---

## THE SDATF MODULES

The SDATF embedded software is responsible for calculating the spacecraft attitude. This requires several auxiliary code modules for computing attitude parameters. The main module shall receive from the on-board computer (OBC) the information of the position of the Sun. Other information is received such as modified Julian date (MJD), time of day in seconds, two-line element (TLE) orbit data, and geomagnetic field vector read by the magnetometer installed on the SDATF board itself.

One module is responsible for providing orbit propagation, i.e. it obtains the position and velocity of the satellite at the present time, as this information is required for calculations of the geomagnetic field and the position of the Sun. The inputs are the last TLE and the time elapsed since the TLE definition.

Another module is responsible for calculating the geomagnetic field vector using a truncated model from the IGRF12 model (Thébault *et al.* 2015), which will be used together with the vector measured by the magnetometer in the attitude parameter calculation process. The inputs to this module are time reference of the model (year), distance from the satellite to the center of the Earth, colatitude and east longitude. Another module performs the vector calculations of the Sun's direction in the geocentric inertial reference frame from a mathematical model.

Auxiliary computations, such as coordinate changes, multiplications, and other mathematical operations involving vectors and matrices, are performed in modules specially programmed for these functions. Finally, the attitude determination module itself calculates the quaternions, which are the attitude parameters, using the available information through the algorithm based on the QUEST method.

More details about the SDATF software, as well as tests, results, and analysis regarding the performance and accuracy of calculations from the adopted mathematical models can be found in Garcia *et al.* (2018).

---

## METHODOLOGY

This section describes the sequence of methodological procedures for building the proposed solution. The level of detail aims to allow and stimulate its evaluation and to present a reference for future developments of similar systems.

The first step in designing a solution is the search for information on integrated circuit technologies used to fabricate potential COTS candidates for use on the SDATF board. Next, it was necessary to obtain information from the manufacturer about the failures in time (FIT), defined as a failure rate of 1 per billion hours / Mbit software error measurement of the possible selected COTS components for the system design. Soft errors, i.e. errors changing the

content of a memory cell, increasingly occur on high-density memory cell devices. Integrated circuits of technologies still in wide use, such as the 180 and 40 nm technologies, have soft errors in the range of 100 to 5000 failure in time/Mbit, respectively. The SEU-related soft errors affect SRAM cells (registers, buffers, cache memory, etc.) and microcontroller internal logic cells. As this project considers a microcontroller manufactured with 90 nm technology, the corresponding measure of time failure per Mbit is adopted for the next step. Modern microcontrollers, even equipped with redundancy features to avoid the effects of soft errors, still require some additional redundant application-level techniques to overcome these nondestructive effects.

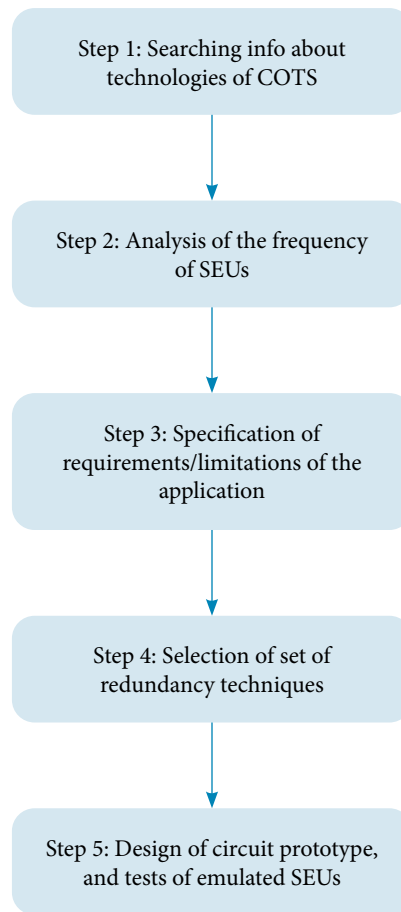
The second step is to conduct a bibliographic survey of the reports studied, which obtained the statistical distribution of frequency with their respective SEUs measurement parameters in integrated circuits manufactured with technology similar to that used in candidate COTS, working in similar environments and conditions that the SDATF will be faced during its operation. The objective of this step was to obtain similar failure rates as the main COTS components of the system will face in order to emulate this situation in the laboratory and validate the proposed fault-tolerant solution. The adopted reference study was presented by Faure *et al.* (2005).

The third step comprises the specification of requirements and limitations imposed by the functionality of the system application, especially regarding its reliability and availability. This stage should consider the study of the conditions of the space environment where the system will operate, and provide the normal and limited conditions of SEU occurrence for decisions taken in the next stage. The SDATF must provide the satellite attitude every second with the highest possible level of reliability and availability, even when the satellite is passing through space zones in its orbit subject to a high level of radiation that can cause SEU.

The fourth step consists of selecting a set of well-established and documented redundancy techniques in order to fully meet all requirements and to comply with the limitations imposed by the system application, under the environmental condition determined by the second and third stages of the methodology. Among the various available redundancy techniques in the fault tolerance area (Kastensmidt *et al.* 2006), e.g., some hardware and software redundancy techniques were selected: a compensation method based on dual redundancy processing with a spare ready to act; a detection method based on mutual activity monitoring with message forwarding in case of no response; and recording the same result in three different memory locations. Finally, by completing the set of selected techniques, the triple vote of the data stored in the three memory positions validates the calculated attitude.

The fifth step deals with the design of a circuit prototype, capable of being tested considering the emulation of SEU under the same environmental conditions shown in step 2. In this same step, the circuit can be tested for more extreme conditions. For example, testing the circuit by increasing the number of SEUs applied to the system memory cells by two, five, and ten times, compared to those reported in the references (Harboe-Sorensen *et al.* 1990; Hubert *et al.* 2014; Sims *et al.* 1993), to compare the limits of the proposed solution with the ones of selected redundancy techniques. For this purpose, a procedure has been developed to emulate SEUs in prototypes of microcontroller-based systems, as well as the SDATF board (Miranda and Duarte 2016). This system allows us to randomly generate a number of events according to the statistical frequency of Poisson distribution, introducing the necessary parameters to emulate the mission. Application outputs recorded in SDATF are monitored. Errors and correct operations are monitored and reported simultaneously to attest to the effectiveness of the proposed solution.

If the previous step is successful, move on to the physical design of an SDATF engineering model, which will be rigorously tested, followed by a flight model, which tests will be moderated to avoid damage to the final system model (see Fig. 2). The flowchart of the proposed methodology is shown in Fig. 3.



**Figure 3.** The flowchart of the proposed methodology.

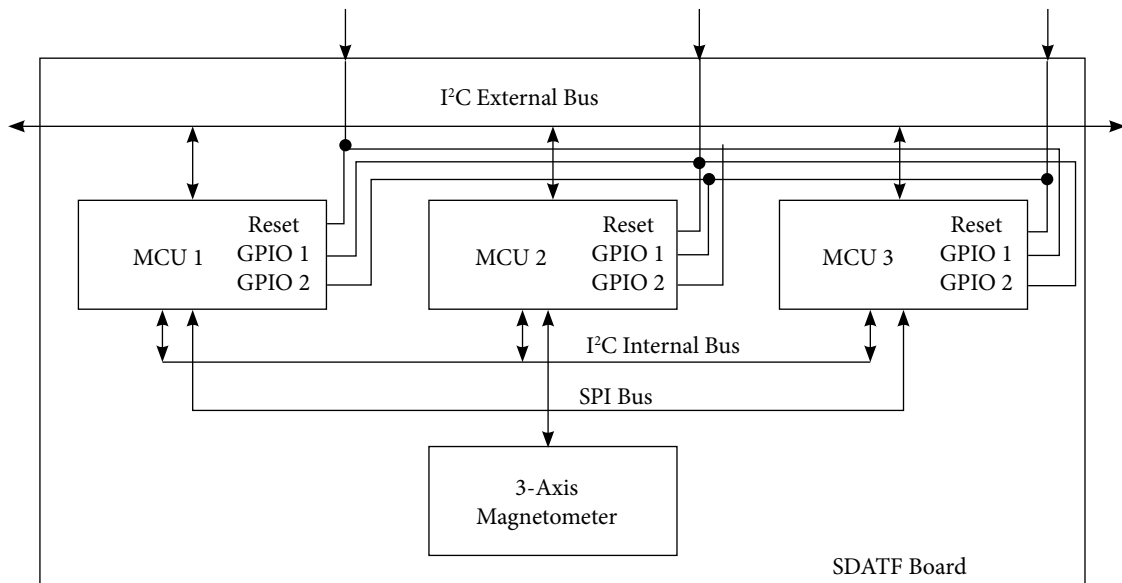
## THE PROPOSED SOLUTION

This section describes in more details the proposed solution, including a description of the hardware structure, system operation and its interaction with the on-board computer (OBC), the fault tolerance scheme, and future flight testing.

### BRIEF DESCRIPTION OF THE ON-BOARD HARDWARE STRUCTURE

Figure 2 presents a description of the representative structure of the SDATF board with its main components. Above the microcontrollers, one can see an I<sup>2</sup>C bus, called an external I<sup>2</sup>C bus, which is used for communication between the SDATF system and the OBC.

On this bus, there is the traffic of data packets PCK\_TASK (containing the information about the calculation task) and PCK\_RESULT (containing the obtained results in the calculation tasks). Below the microcontrollers, there is a second I<sup>2</sup>C bus, called I<sup>2</sup>C internal bus (Fig. 4). It is through this bus that microcontrollers exchange data with each other, such as magnetometer measurement data, calculated attitude data, heartbeats, and internal control commands. Still below the microcontrollers, the SPI bus, which is another communication bus, can be observed. The microcontroller set in Master mode obtains data from the magnetometer through this bus.



**Figure 4.** The structural description of SDATF with its main COTS devices and the internal and external OBC connections.

Figure 4 also shows the reset signals of microcontrollers that have an external physical connection with different general-purpose input/output (GPIO) pins of the OBC. This way, the OBC can reset all or individually any SDATF microcontroller on demand from a telemetry command coming from the base station on Earth. Each microcontroller pin is also physically connected to another GPIO pin of each microcontroller. Thus, any active microcontroller on the SDATF board can restart each microcontroller as long as there is any suspicion of inactivity of its active monitored microcontroller.

## THE INTERACTION BETWEEN SDATF AND OBC

The SDATF acts as an independent attitude calculation server and asynchronous system with respect to OBC. The on-board computer at any time provides data packets with various calculation tasks containing the measurements and parameters sent via an I<sup>2</sup>C packet to the SDATF. This package, called PCK\_TASK, contains the uplink telemetry commands and solar sensor data, with the respective time reference marked.

After some time, which is necessary for the attitude calculation by the SDATF, the OBC requests the results via a second I<sup>2</sup>C packet. This package is called PCK\_RESULT and contains the calculated attitude and all data required for the ground validation process via downlink to the base station. The fault tolerance techniques involved in the system design are transparent to the OBC working mode. Therefore, the OBC has no control over the SDATF internal operation, internal exchange modes, or communication between the system calculation nodes. A single hardware reset interface has been added to allow OBC, via a remotely sent command from Earth, to perform an on-demand reset individually on any of the three microcontroller units (MCU) or a single simultaneous reset signal on all three MCUs. The interface that allows this direct command is the GPIOs connected to the respective microcontroller reset inputs. After power-up, the system executes internal boot routines and waits for the OBC command while in standby mode. No further command or intervention requires OBC's attention to start the SDATF system.

## OPERATION OF THE SDATF

The goal of SDATF is to calculate the CubeSat three-axis attitude from measurements of the three-axis magnetometer installed on its own board, as well as from measurements of solar sensors. To calculate attitude, the system must receive as OBC data input:

- The satellite position in orbit data;
- The data read from solar sensors installed on the satellite;

- The time in UTC format associated with solar sensor measurements;
- The reset signal of the magnetometer and calibration commands.

The same firmware program is installed on three microcontrollers. This firmware provides a differentiated configuration for each microcontroller according to SDATF operation in flight. During normal CubeSat operation, the system has one of its microcontrollers configured in Master mode, another in Partner mode, and the third in Sleeper mode. As long as there is no change in the configuration mode between microcontrollers, this configuration is called the SDATF duty cycle. All MCU configuration modes with their responsibilities and tasks are described below:

- **Master Mode:** It receives from OBC a data packet via an external I<sup>2</sup>C bus, called PCK\_TASK packet. The microcontroller (MCU) is responsible for sending another data packet through the external bus I<sup>2</sup>C, called the PCK\_RESULT packet, to the OBC computer, at its request. The MCU is responsible for processing the measurements, filtering, and calibration of data received from the magnetometer, and the data to be associated with UTC time. These data are obtained at a commonly adopted recording frequency of 40 Hz. The MCU is also responsible for recording the processed magnetometer data in three separate memory locations. The MCU shares this data with the other two microcontrollers via the internal I<sup>2</sup>C bus. This same MCU is responsible for calculating the attitude with a voting procedure of data stored in its memory resulting from magnetometer data processing. The MCU also requests the attitude calculated by the two other microcontrollers and resends the calculated attitude for the validation of the other two microcontrollers via the internal I<sup>2</sup>C bus. Validation of the calculated attitude, compared to the attitude received by two other microcontrollers, takes place in MCU Master. It regularly sends activity messages (called heartbeats) to the MCU in Partner mode and controls MCU Partner activity by regularly listening for this microcontroller with a maximum timeout. When it is suspected that Partner is not communicating, MCU Master should reset MCU in Partner mode. In this case, the Master must restart the suspected locking MCU Partner, which will resume activity in Sleeper mode; and wake up the MCU in Sleeper mode to become the new MCU Partner in the fault-tolerant system.
- **Partner Mode:** The MCU receives an I<sup>2</sup>C external bus data packet, called the PCK\_TASK packet, from OBC. It is the microcontroller responsible for receiving the processed data from the magnetometer supplied by the MCU configured in Master mode via the internal I<sup>2</sup>C bus. The Partner must calculate attitude using magnetometer data received from Master, along with data received by PCK\_TASK package. The MCU Partner is responsible for sending the calculated attitude to the other two microcontrollers. It receives the calculated attitude of the other microcontrollers via the internal I<sup>2</sup>C bus. The MCU Partner must proceed with voting on its calculated attitude against the attitude received by two other microcontrollers. This MCU must record the voted attitude in order to keep a backup to prepare the PCK\_RESULT package in case it becomes the new Master of a new round. The Partner regularly sends messages to the MCU in Master Mode, recording their activity (heartbeats). The Partner is responsible for controlling MCU Master activity through Master's regular heartbeat demand, within a maximum timeout limit.
- **Sleeper Mode:** This MCU receives a data packet via an external I<sup>2</sup>C bus, called a PCK\_TASK packet, from OBC. It also receives processed data from the magnetometer supplied by the MCU configured in Master mode via the internal I<sup>2</sup>C bus. The MCU Sleeper also calculates attitude using magnetometer data received from the Master, along with data received by the PCK\_TASK packet. It sends the calculated attitude to the other two microcontrollers. The Sleeper must receive the calculated attitude of the other microcontrollers also via the internal I<sup>2</sup>C bus. The MCU Sleeper must proceed with voting on its calculated attitude compared to the attitude received by two other microcontrollers. It records the voted attitude in order to keep a backup to prepare the PCK\_RESULT package for the case it becomes the new Master in the next round. The rest of the time is in low-power mode, waiting for a new PCK\_TASK package to perform a new duty cycle or waiting for a control signal from MCU Master to become the new MCU in Partner mode in case of any communication failure of the Partner noted by the Master of the round.

A microcontroller in Master or Partner configuration mode in a round is called active node. Each active node is responsible for monitoring the activity of the other active nodes in the round. There are two different ways for one node to monitor the activity of the other. The first way is to control the time of useful information exchange between them, where useful information is the treated data from magnetometer readings and attitude calculation data performed individually by each active node. The second way is

to control the timeout limit for receiving a heartbeat between active nodes configured as Master and Partner mode. One MCU monitors the operational state of the other. After the attitude is calculated and verified by the master node, the Master and Partner microcontrollers are monitored by heartbeat exchange via the I<sup>2</sup>C internal bus. The Partner microcontroller continues to process the collected magnetometer data continuously, and store the digitally filtered sensor data in three different positions in its SRAM memory.

In order to make this monitoring scheme possible, some timer peripherals are configured in the software of each microcontroller. A first timer is dedicated to setting the heartbeats send timeout to another active system node. A second timer must set the elapsed time limit to wait for the heartbeat from the other active node. A third timer is dedicated to controlling the time limit that the microcontroller must wait to receive magnetometer data for attitude calculation. In addition, the same timer is used to control the time limit for receiving the calculated attitude of the other two active microcontrollers in an SDATF time window.

The calculation attitude record, as well as the voted attitude, should be stored in every three distinct volatile memory positions of each MCU. The goal is that each data used to calculate the attitude or each possible data to be sent to another MCU or OBC first go through a voting scheme (described in detail below) to ensure the reliability of the processed or in-use data.

The voting scheme consists of two phases. The first phase is a simple vote. Only the values stored in the three volatile memory positions are compared. The correct value is considered to be the value contained in most of the three different memory locations. If there is a triple disagreement in this first step, a second voting phase is required, which consists of a bit to bit comparison of the contents of the locations of the three memory locations that failed in the first voting stage. The bit value contained in most equivalent memory locations is considered to be the correct value to obtain the correct voted final value of the data to be used in the attitude calculation or sent to the other nodes.

To validate its calculated attitude with the attitude calculated by other system nodes, the Master mode microcontroller performs the same voting scheme described in the previous paragraph. The attitude calculation calculated by the Master is then stored in three different positions of volatile memory to assemble the PCK\_RESULT package, with the result that will be required at any time by the OBC.

The OBC will be responsible for collecting the packets, storing them in its internal memory and periodically sending the packet to the ground telemetry station for ground validation. This package is called PCK\_RESULT. The data collected by the packages obtained during the flight will be statistically analyzed and the results will be used for any corrections and improvements in the SDATF system.

---

## FAULT TOLERANCE TESTS RESULTS

In order to perform fault tolerance tests, a prototype was assembled based in the microcontroller PIC24FJ64GA002 (Microchip Technology Inc 2007), thus allowing the mapping of sensitive system memory elements and the undesirable effects of the injected errors. In this context, the bit flip injection was emulated with a radioactive particle arrival rate as proposed by Nicolaidis and Velazco (2007), and with the setting of external event interruption on one target at a time of the attitude determination firmware. The bit flip injection was produced using the portable testbed for harsh environment of single event upset (PORTHES) system, a tool developed at the Intelligent Systems Laboratory of the Federal University of Minas Gerais (UFMG), which allows the application of the code emulating upsets (CEU) methodology proposed by Velazco *et al.* (2000).

In these batteries of tests, the automatic mode was used to inject the bit flips through the PORTHES graphical interface, where targets are selected by random upset injection, following the Poisson distribution with a rate of 2.3 upsets per second. In this graphical interface, it was possible to identify the system behavior and the effects caused by inserting upsets on this specific target. Table 1 presents the results obtained in those SDATF tests. The duration of each battery of tests was 2 min, and the showed results are the mean values calculated of the observations after three repetitions of the test procedure.

It can be seen that some register failures caused a system lock after terminating the PORTHES bit flip injection process. These registers are U2TXREG, used by the UART2 communication peripheral, which acts to transmit data between Master and Sampler, the WREG14, used as a stack user-defined frame pointer, has the function of allocating the memory, the address of the local variables (context change) and the LATA, which has the pins to reset the other two MCUs, among others.



**Table 1.** Results of the tests with fault injection in the SDATF.

MCU special function register	Injected bit flips	Effect caused by the upset			System availability (%)
		Tolerated errors (%)	Results error (%)	Loss sequence errors (%)	
T1CON	304	87.59	0	12.41	91.09
TMR1	303	89.51	1.23	9.26	90.08
PR1	346	86.52	1.12	12.36	85.60
T2CON	296	99.32	0	0.68	100
TMR2	291	100	0	0	100
PR2	318	98.80	0	1.20	99.20
T3CON	327	100	0	0	100
TMR3	292	100	0	0	100
PR3	326	100	0	0	100
T5CON	273	100	0	0	100
TMR5	278	100	0	0	100
PR5	267	100	0	0	100
IFS0	397	62.36	28.65	8.99	89.6
IFS1	437	70.98	24.11	4.91	93.60
IEC0	358	2.36	1.57	96.07	3.20 (system lock)
IEC1	278	100	0	0	100
IPC0	279	100	0	0	100
IPC1	271	100	0	0	100
IPC2	292	100	0	0	100
U2MODE	337	93.67	5.88	0	100
U2STA	330	94.84	5.16	0	100
U2BRG	334	70.22	29.78	0	100
U2TXREG	347	23.30	75.99	0.71	99.20
U2RXREG	265	100	0	0	100
INT0	278	100	0	0	100
INT1	297	100	0	0	100
INT2	333	100	0	0	100
CLKDIV	284	48.41	51.58	0	100
WREG0	278	100	0	0	100
WREG1	287	99.21	0.79	0	100

Continue...

**Table 1.** Continuation.

MCU special function register	Injected bit flips	Effect caused by the upset			System availability (%)
		Tolerated errors (%)	Results error (%)	Loss sequence errors (%)	
WREG2	294	43.68	56.32	0	100
WREG3	278	100	0	0	100
WREG4	323	100	0	0	100
WREG5	337	57.14	0.79	42.07	58.40 (system lock)
WREG6	298	99.24	0.76	0	100
WREG7	304	100	0	0	100
WREG8	274	100	0	0	100
WREG9	313	100	0	0	100
WREG10	339	100	0	0	100
WREG11	297	100	0	0	100
WREG12	289	100	0	0	100
WREG13	290	100	0	0	100
WREG14	394	5.38	1.54	93.08	3.20 (system lock)
WREG15	312	100	0	0	100
RPOR4	303	100	0	0	100
RPINR0	267	100	0	0	100
RPINR1	269	100	0	0	100
RPINR19	287	100	0	0	100
TRISA	276	100	0	0	100
TRISB	355	7.74	38.09	54.17	10.40 (system lock)
PORTA	300	100	0	0	100.
PORTB	262	36.05	63.95	0	99.2
LATA	463	3.21	28.84	67.95	3.27 (system lock)
LATB	308	97.67	2.33	0	100
SPLIM	383	45.16	7.09	47.74	36.80 (system lock)
PC - asm ("GOTO 0x000000")	349	89.58	9.90	0.52	90.0
PC - asm ("GOTO 0x000002")	351	36.97	62.56	0.47	93.6

These observed problems indicate the necessity of providing some design changes in hardware and in the software improved to the flight release. Some special attention was dedicated to provide software redundancy in the case where low-system availability cases observed on bit-flip emulation sessions (see last column of Table 1) in order to achieve 100% availability. Concerning registers of which SEUs dealt to sequence-loss errors, the fault-tolerant system was able to resume operation without external intervention. For instance, in the case of IFS0 register, the system recovered and resumed communication with PORTHES after time latency. The general evaluation of these tests results is quite positive in terms of the capacity of the system operational functioning keeping.

Even so, in order to overcome the detected problems, the use of diverse communication channels and hardware interfaces was adopted in order to avoid the low availability observed due to UART communication use during test sessions. Due to this low availability cases observed, the flight SDATF design was modified for using two I2C communication channels and one SPI (as can be seen in Fig. 4). These design modifications resulted in providing diversity channels to data exchange and consequently higher rates of availability and fault recovery. The fault injection tools and the session tests accomplished, provide the necessary data to enhance the fault tolerance design and achieve the availability required by the mission.

---

## CONCLUSION

The main contribution of this work is the procedure described and detailed in the engineering decisions and the sequence of steps required for the designing of a fault-tolerant scheme for the attitude determination system. The few detected problems concerning the system lock during SEU injection sessions allowed to observe possible weaknesses in the proposed fault-tolerant solutions and adapt it to overcome these problems. The SEU emulation campaigns allowed to identify potential weaknesses in the fault-tolerant solution and propose some software design adequacy in order to overcome the low-availability situations. The next steps of this system development will include the analysis of experimental data and validation of the SDATF in the planned flight test as a payload of the NanosatC-BR2 mission.

In addition, a new system proposal, which is more complete in terms of use on CubeSat nanosatellites, should include a control module, in this case constituting an attitude determination and control system (ADCS).

---

## ACKNOWLEDGMENTS

The authors acknowledge Dr. Hélio Koiti Kuga, Dr. Otávio Santos Cupertino Durão, and Dr. Valdemir Carrara of the Brazilian Institute for Space Research (INPE) for the knowledge and experience shared during the development of the project.

Editors and authors are thankful to Fundação Conrado Wessel for providing the financial support for publishing this article.

---

## FUNDERS

Conselho Nacional de Desenvolvimento Científico e Tecnológico

[<http://doi.org/10.13039/501100003593>]

Fundação de Amparo à Pesquisa do Estado de Minas Gerais

[<http://doi.org/10.13039/501100004901>]

Agência Espacial Brasileira

[<http://doi.org/10.13039/501100008138>]

---

## AUTHOR'S CONTRIBUTION

Conceptualization, Duarte R. O., Vale S. R. C. , Martins-Filho L. S. and Torres F. E.; Methodology, Duarte R. O., Vale S. R. C., Martins-Filho L. S. and Torres F. E.; Investigation, Duarte R. O., Vale S. R. C. , Martins-Filho L. S. and Torres F. E.; Writing – Original Draft, Duarte R. O., Vale S. R. C. , Martins-Filho L. S. and Torres F. E.; Writing – Review and Editing, Duarte R. O., Vale S. R. C., Martins-Filho L. S. and Torres F. E.; Resources, Duarte R. O., Vale S. R. C. , Martins-Filho L. S. and Torres F. E.; Supervision, Duarte R. O., Vale S. R. C. , Martins-Filho L. S. and Torres F. E.

---

## REFERENCES

- Duarte RO, Martins-Filho LS, Kuga HK (2009) Performance comparison of attitude determination Algorithms developed to run in a microprocessor environment. Paper presented 20th International Congress of Mechanical Engineering. Gramado: COBEM.
- Duarte RO, Torres FE, Gomes TH, Martins-Filho LS, Kuga HK (2011) An attitude determination system implementation to low orbit small satellite with fault-tolerant techniques. Paper presented Proceedings of the 8th IAA Symposium on Small Satellites for Earth Observation. Berlin: IAA.
- Faure F, Velazco R, Peronnard P (2005) Single-event-upset-like fault injection: a comprehensive framework. *IEEE Trans Nucl Sci.* 52(6):2205-2209. <https://doi.org/10.1109/TNS.2005.860689>
- Garcia CBA, Vale SRC, Martins-Filho LS, Duarte RO, Kuga HK, Carrara V (2018) Validation tests of attitude determination software for nanosatellite embedded systems. *Measurement.* 116:391-401. <https://doi.org/10.1016/j.measurement.2017.11.040>
- Harboe-Sorensen R, Daly EJ, Underwood CI, Ward J, Adams L (1990) The behaviour of measured SEU at low altitude during periods of high solar activity (spacecraft memories). *IEEE Trans Nucl Sci.* 37(6):1938-1946. <https://doi.org/10.1109/23.101212>
- Hubert G, Bezerra F, Nicot J-M, Artola L, Cheminet A, Valdivia J-N, Mouret J-M, Meyer J-R, Cocquerez P (2014) Atmospheric radiation environment effects on electronic balloon board observed during polar vortex and equatorial operational campaigns. *IEEE Trans Nucl Sci.* 61(4):1703-1709. <https://doi.org/10.1109/TNS.2014.2303497>
- Kastensmidt FL, Carro L, Reis R (2006) *Fault-tolerance techniques for SRAM-based FPGAs.* Boston: Springer. <https://doi.org/10.1007/978-0-387-31069-5>
- Microchip Technology Inc (2007) *PPIC24FJ64GA004 Family Data Sheet 28/44-Pin General Purpose, 16-Bit Flash Microcontrollers.* [accessed Oct 2 2019]. <http://ww1.microchip.com/downloads/en/devicedoc/39881b.pdf>
- Miranda BCD, Duarte RO (2016) Behavior modelling and simulation of a fault tolerant attitude determination system for NanosatC-Br2. Paper presented Proceedings of the 2nd IAA Latin American CubeSat Workshop. Florianópolis: IAA.
- Mukherjee S (2008) *Architecture design for soft errors.* Burlington: Morgan Kaufmann.
- Nicolaidis M, Velazco R (2007) Architectures for robust and complex integrated systems (ARIS). TIMA Annual Report. [accessed Apr 18 2019]. <http://tima.imag.fr/research/files/gr-08/aris.pdf>
- Schuch NJ, Silva MR, Durao OSC (2014) NanosatC-BR CubeSat development & capacity building programs: NanosatC-BR1 & NanosatC-BR2. Paper presented Proceedings of 1st Latin American IAA CubeSat Workshop. Brasília: IAA.
- Sims MR, Sims AJ, Bentley RD, Guttridge PR, Gourlay JA (1993) Measured single event upset rates in low earth orbit for the ROSAT Wide Field Camera. *Nucl Instrum Methods Phys Res A.* 329(1-2):329-336. [https://doi.org/10.1016/0168-9002\(93\)90951-D](https://doi.org/10.1016/0168-9002(93)90951-D)
- Thébault E, Finlay CC, Toh H (2015) Special issue "International Geomagnetic Reference Field — the twelfth generation". *Earth Planet Sp.* 67:158. <https://doi.org/10.1186/s40623-015-0313-0>
- Velazco R, Rezgui S, Ecoffet R (2000) Predicting error rate for microprocessor-based digital architectures through C.E.U. (code emulating upsets) injection. *IEEE Trans Nucl Sci.* 47(6):2405-2411. <https://doi.org/10.1109/23.903784>
- Wang F, Agrawal VD (2008) Single event upset: an embedded tutorial. Paper presented 21st International Conference on VLSI Design (VLSID 2008). Hyderabad: IEEE. <https://doi.org/10.1109/VLSI.2008.28>