POLITECNICO DI TORINO
Repository ISTITUZIONALE

Verification of a CubeSat via Hardware-in-the-loop Simulation

*Original*
Verification of a CubeSat via Hardware-in-the-loop Simulation / Corpino, S., Stesina, F.. - In: IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS. - ISSN 0018-9251. - ELETTRONICO. - 50:4(2014). [10.1109/TAES.2014.130370]

*Availability:*
This version is available at: 11583/2535907 since:

*Publisher:*
IEEE

*Published*
DOI:10.1109/TAES.2014.130370

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)

29 November 2026

NOTICE:
This is the author's version of a work that was accepted for publication in Transactions on Aerospace and Electronic Systems by IEEE. Changes resulting from the publishing process, such as editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Transactions on Aerospace and Electronic Systems, [VOL. 50, ISSUE 4, (October 2014)] DOI#0.1109/TAES.2014.130370

©2014 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

To cite this article:
Corpino S., Stesina F., "Verification of a CubeSat via Hardware-in-the-Loop Simulation", IEEE Transactions on Aerospace and Electronic Systems, Vol. 50, No. 4, 2014, DOI. No. 10.1109/TAES.2014.130370

# Verification of a CubeSat via Hardware-in-the-loop Simulation

Sabrina Corpino and Fabrizio Stesina

***Abstract*—This paper describes the Hardware-In-the-Loop (HIL) simulation methodology used for the verification of functional requirements of e-st@r-I CubeSat. The satellite's behavior has been investigated via HIL simulation and the results obtained are consistent with the expected values in any operative conditions. It has been proven that HIL simulation is a valuable means for supporting the verification process of small satellites, and may help reducing time and cost of the development phase and increasing mission reliability.**

***Index Terms*—CubeSat, Hardware in the Loop, HIL architecture, Model & Simulation, Space Systems Verification**

## I. Introduction

THE design and verification processes of any engineering system make use of modeling and simulation solutions. Several techniques have been developed in this field thanks to the dramatic progress of technology in the last decades. One effective modeling and simulation method is the Hardware-In-the-Loop (HIL) approach. This methodology consists in the combination of computer simulation and hardware in a single coherent platform [1].

Typically, HIL simulation is the last stage of the testing and integration process of an engineering system. We can define a few important stages of interaction between the elements of the system along this process, as detailed by Eickhoff in [2]:

1) Algorithm in the loop (AIL). The algorithms of interest are added to the pure numerical simulation. They are not yet written in the formal language that will be used on the final hardware and are not run on it. AIL is mainly used at design stages with the objective of testing the algorithms
2) Software in the loop (SIL). Algorithms are translated into the final programming language, but they run on ground

Manuscript received May 30, 2013. This work was supported in part by the Politecnico di Torino under Fondo per la Progettualità Studentesca (grant for students hands-on education activities).

S. Corpino is with the Politecnico di Torino, Torino, Italy (corresponding author, phone: +390110906867; fax: +390110906899; e-mail: sabrina.corpino@polito.it).

F. Stesina is with the Politecnico di Torino, Torino, Italy (e-mail: fabrizio.stesina@polito.it).

hardware. The software carries out all the required functions, but in general its performance is different with respect to running it on a flight unit architecture (e.g. PC vs embedded-PC)

3) Hardware in the loop (HIL). Real hardware is included in the simulation loop, and consists typically of sensors and/or actuators. HIL is a hybrid software-hardware simulation architecture, in which the hardware part can vary from a few pieces to the fully integrated system. HIL technique is particularly useful for the verification of all those elements that operate in special environments and conditions which are difficult to reproduce in a laboratory. It may help to detect unexpected behaviors and/or failures arising from the integration of the component in the global system.

In the literature there are some examples regarding simulation based on HIL technology. A wide range of options exists in terms of level of complexity. An example of a complex HIL architecture is the simulation of on-orbit docking between two spacecraft at the Spacecraft Robotic Laboratory of the Naval Post Graduate School in Monterey, CA [3]. In this facility, the target and the chaser are physically reproduced, while the rest of the system is simulated on a real-time simulation computer. In [4], the authors show the verification campaign carried out for the Special Purpose Dextrous Manipulator installed onboard the International Space Station. In this case, the space hardware is simulated and the contact dynamics is emulated using a rigid robot. In [5], the authors describe a ground-based HIL simulation facility for Rendez-Vous and Docking (RVD), called European Proximity Operations Simulator (EPOS), at German Space Center. The laboratory is aimed at providing test and verification capabilities for complete RVD processes of on-orbit servicing missions using two robots for the physical real-time simulation of the maneuvers. Other examples of HIL simulation can be found in literature applied to non space-related systems, as described in [6] – [10]. HIL technique can be applied for verification from component to element level. In [11], HIL simulation is used for the development of a network of sensors for pico-satellite missions. This work underlines how special features of the network are investigable only via the HIL approach. Another interesting example is the development of a HIL simulator for the simultaneous test of an Attitude Heading Reference System and an Attitude Control System based on momentum wheels [12]. Finnset et al. dealt with the simulation of the attitude control system of a small satellite, the European Student Moon Orbiter, using a hybrid SIL-HIL architecture [13]. All these applications show the effectiveness of a test campaign conducted via HIL simulation.

In this paper, the use of the HIL approach to support the engineering team during CubeSat missions development is documented. CubeSats are small satellites which share a common interface and equipment standardization [14]. The basic CubeSat is a 10cm-side cube-shaped platform whose mass is less than 1.33 kilograms [15]. By its nature, a CubeSat is a cheap spacecraft which can be produced in only a few months. CubeSats have been mainly developed for educational purposes, but they are increasingly being used for real science and service missions [16]. CubeSats can be useful for the attainment of a broad set of mission goals, including science, technology demonstration, communications, and Earth observation [17] [18]. This will require the development of adequate technology to increase CubeSat performance. Furthermore, it is necessary to improve

mission reliability, because educationally-driven missions have often failed. The failure of CubeSats is dominated by infant mortality, which can be traced back to design weakness and/or ineffective Assembly-Integration-Verification (AIV) planning and execution. An effective and exhaustive Verification and Validation (V&V) process may help to increase the reliability of small-scale satellites, as claimed also by Dubos et al. [19].

The AeroSpace System Engineering Team (ASSET) has been working on small satellite missions of Politecnico di Torino since 2003 [20]. One of the main research fields deals with defining and implementing effective and efficient methodologies to support the satellite development during all phases of its life cycle. To this purpose, a Model&Simulation-based tool has been developed to support and improve the AIV activities within a CubeSat project. In this paper, the description of a V&V process based on HIL architecture is provided. A HIL simulator has been designed and implemented for a case of interest to demonstrate the feasibility of the approach. The application of HIL simulation improves the understanding of the actual behavior of the onboard devices as well as the interaction among pieces of hardware included in the test. The same results are not achievable with AIL and SIL simulation. The HIL approach also simplifies the modeling activity with respect to AIL and SIL techniques. The real hardware can then be tested under several mission conditions without the need of sophisticated experimental facilities. Furthermore, results from HIL simulation may be used to refine the models used in the AIL/SIL simulations of future projects. In this paper, main opportunities are highlighted together with challenges peculiar to CubeSat applications.

Section II presents the rationale for the research and potential advantages stemming from the application of the HIL technique to CubeSat projects. In Section III, the CubeSat project assumed as test-case is briefly described. Section IV represents the main content of the article. It includes the description of the HIL simulator's architecture, the working principle, test settings and run. In Section V the results of the tests are illustrated and discussed, then conclusions and final remarks are suggested in Section VI.

## II. CubeSat verification and HIL simulation

Mission success is highly influenced by a robust design on the one hand, and by an effective verification and validation campaign on the other hand. With respect to the V&V activity, it shall be noticed that a major difference between CubeSats and traditional spacecraft exists. To date, the main objective of CubeSat V&V campaigns has been the demonstration of the satellite's capability to survive the launch phase without jeopardizing the launch vehicle and other spacecraft in the payload bay. This implies that CubeSats have been extensively tested against launch environment requirements upon request of launch authorities. Unfortunately, minor effort was committed to the verification of functional and operational requirements. These verifications are fully demanded to CubeSats developers, and no guidelines exist.

The International Organization for Standardization (ISO) is working on the definition of a new standard devoted to the

verification of small-scale satellites. The standard is meant to improve the reliability of this class of satellites, while keeping its nature of low-cost and fast-delivery. In fact, some tests are necessary to improve the reliability of small-scale satellites to a level acceptable as a subject of commercial investment. At the same time, applying the same test requirements and methods as the ones for traditional large/medium satellites would undermine the advantage of low-cost and fast-delivery. The new ISO standard will reflect the work done within the Nano-satellite Environmental Tests Standardization project described in [21]. The European Space Agency (ESA) has recently launched the *Fly Your Satellite!* initiative: selected university teams from EU and Canada are supported by ESA's experts for the development of their CubeSats. The project aims at increasing CubeSat mission reliability through several actions: to improve design implementation, to define best practice for conducting the verification process, and to make the CubeSat community aware of the importance of verification.

The activity discussed in the paper is located within this framework and it is aimed at improving the effectiveness of the functional verification of CubeSat. It is worth mentioning that for CubeSat programs the functional verification is usually carried out via simulation with no hardware in the loop. In most cases, CubeSats are tested against functional requirements in ambient condition, then they undergo the environmental tests, followed each by a reduced functional test. Rarely CubeSat's functionalities are tested in orbit-like conditions mainly because of two reasons. First, this test would require resources, facilities and costs usually unsustainable for a student initiative. The facilities devoted to these purposes are few and not appropriate for the test of so-small objects; they are expensive to hire, and inaccessible for unqualified users, i.e. the students. Second, it often occurs that the operational orbit is still unknown when the CubeSat is designed, integrated and tested. To reproduce physically the whole set of orbit conditions would result in a tremendous effort, in terms of both budget and testing time. The HIL approach represents the right balance between performance and cost of the testing activity.

From the available literature and the authors' experience, HIL simulation is a methodology which has not been largely used for small satellites, and in particular for CubeSats. HIL applications for small space platforms are quite recent and they are mainly oriented to the verification of Attitude Determination and Control System (ADCS) performances [22] or, more in general, the dynamic behavior of satellites [23]. It is the authors' opinion that CubeSat projects may benefit to a great extent from the application of HIL technology in V&V activity for the following reasons:

1) HIL simulation guarantees that we get the actual response from the real hardware included in the loop. HIL testing results much more authentic if compared with other simulation techniques. The presence of the hardware allows for example the accurate measurement of lag, signal noise, signal dropout, actuator saturation, and other parameters
2) Including the real hardware in the simulation loop simplifies the modeling activity. It may happen that accurate mathematical models of some devices are too complex to build or they are not readily available. This is particularly true in case of a CubeSat project, which makes extensive use of Commercial Off The Shelf components

3) Unpredictable interactions among pieces of hardware can be detected and eventually straightened out. A better understanding of the integrated system's behavior can be achieved early in the development process
4) The hardware can be easily tested in several orbit conditions, as these are virtually reproduced. This is particularly useful for CubeSat platforms which shall be flexible enough to adapt to different mission conditions.

Notwithstanding these opportunities, using HIL simulation for CubeSat projects entails a couple of concerns. First, the HIL approach necessitates test facilities equipped with flight hardware and dedicated support equipment. On the one hand, this means additional cost and longer test preparation time if compared with pure, AIL, or SIL simulations. On the other hand, the increased cost and time can be justified taking into account that HIL simulation yields greater insight into system's performance than other simulations. Second, traditional space systems have been tested via the HIL technique for decades in the industry. CubeSats are usually developed by universities, research centers, and small medium enterprises, which are not generally provided with certified simulators. Brand-new simulators require to be validated and the process may represent an issue for CubeSat programs, for example in terms of budget constraints and return of the investment. If the simulator's architecture is flexible enough, it can be used for several projects and not only for a specific mission, thus justifying the investment.

The abovementioned challenges could be overcome by the standardization of ground support equipment and facilities devoted to HIL testing of CubeSats. The cooperation within the CubeSat Community plays a major role to this purpose.

## III. E-ST@R PROGRAM

The e-st@r (Educational SaTellite @ politecnico di toRino) project is a structured hands-on education and research program based on CubeSats development. The main objective is to prepare future generation of space professionals. University programs shall take up the technological challenges issued by the scientific community and the industry, and they must help to improving knowledge necessary to build future space missions. The big challenge is doing space missions at low cost while keeping reliability as high as possible. For these reasons, one of the main tasks of the e-st@r project is the development of methodologies to support both effectively and efficiently the entire satellite lifecycle.

E-st@r-I is the first element of the CubeSats family we are working on. It is a 1U satellite developed by students with the aim of testing in space new technologies and design solutions. The CubeSat was chosen by ESA within the "Educational Payload on the Vega Maiden Flight" initiative. The launch took place in 2012, February 13$^{th}$, from Europe's spaceport in French Guyana. The second 1U CubeSat (e-st@r-II) is under test at the Systems and Technologies for Aerospace Research Laboratory (STAR Lab) of the Department of Mechanical and Aerospace Engineering. The team is also working at the development of a 3U CubeSat for Earth Observation based on remote sensing techniques.

### *A. CubeSat Configuration*

CubeSats included in the e-st@r program are equipped with a set of subsystems for supporting the mission and the payload. The payload of the first two CubeSats is constituted by an Active Attitude Determination and Control System (A-ADCS). The attitude is determined by algorithms and Kalman filter in the A-ADCS software using data sensed by an Inertial Measurement Unit (IMU), a magnetometer and the solar panels used as coarse sun sensors. The actuation is accomplished by three magnetic torquers (MT). The A-ADCS is controlled by a dedicated micro-controller (ARM9), on which the algorithms for determination and control run [24].

The spacecraft bus is constituted by the Electrical Power System (EPS), the Communication System (COMSYS), the Onboard Computer (OBC), and the Structure and Mechanism System (S&M). The EPS collects and transforms solar radiation energy by means of five solar panels made of GaAs cells. Electrical power is stored in two battery packs, then regulated and distributed to other subsystems through two power buses at 3,3 Volt and 5 Volt. The COMSYS is an in-house developed design and manages downlink and uplink signals. It exchanges information with the OBC by a piece of equipment based on commercial components: a PIC16 modulates/demodulates and checks the signal, a radio transceiver amplifies the signal and a dipole antenna transmits/receives the signal to/from Earth. The OBC performs all onboard activities related to the command and data handling functions, the data storage in the SD memory card, and time synchronization. The structure is an aluminum alloy box designed to endure launch loads, and to host and protect the electronics from the harsh space environment. It also includes the deployment system of the antenna. The CubeSat flight unit and the blocks scheme are shown in Fig. 1. The main bus represents the common interface among subsystems, for both the data flow and the power line.

### *B. E-st@r-I verification campaign*

The CubeSat verification campaign has been designed according to ECSS (European Cooperation for Space Standardization) rules. It includes environmental, functional and miscellaneous verifications at different levels in different project phases. The following functions of the CubeSat can be verified in the HIL simulator at qualification stage [25]:

1) to transmit/receive telemetry/commands
2) to determine and control attitude
3) to charge/discharge batteries

The electric power consumption is also monitored and verified during the test.

Requirements related to operations can also be demonstrated, such as:

1) the capability of the CubeSat to change mode of operation upon command from GCS
2) the autonomous deployment of the antenna
3) the execution of the activation sequence, i.e. the start-up sequence which runs once the CubeSat is powered immediately after the deployment from the P-POD (Poly Picosatellite Orbital Deployer).

In order to verify the correct behavior of the satellite with respect to requirements and constraints, the remarkable mission and housekeeping telemetry parameters have been individuated:

1) Status bytes: operative modes tag, mission time, packet ID number
2) EPS telemetry: voltage, current and temperature of the batteries; battery bus voltage; current of the 5V and 3.3V buses; and voltage, current and temperature of the solar panels
3) ADCS telemetry: quaternions, angular velocities and Earth Magnetic Field (EMF)

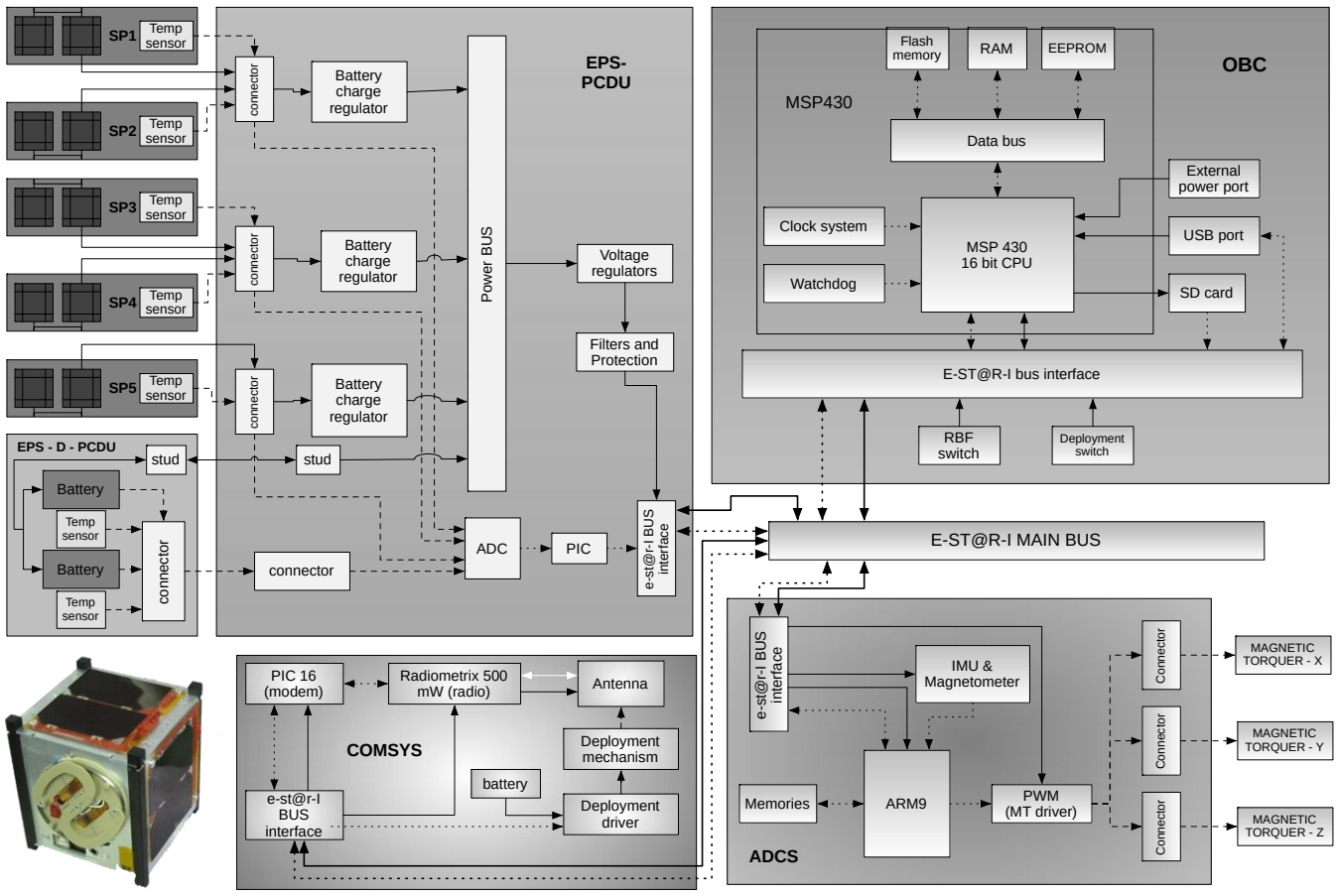


**Fig. 1. e-st@r-I picture and blocks scheme. Main subsystems and components are indicated in the boxes: EPS (Electrical Power System), OBC (On Board Computer), COMSYS (Communication System), and ADCS (Attitude Determination and Control System). All subsystems interface through the e-st@r-I main bus block.**

## IV. HARDWARE IN THE LOOP SIMULATOR

This section describes the HIL simulator used for the test campaign carried out on the e-st@r-I CubeSat. The simulator has been designed to adapt to other missions and CubeSat configurations. It can be used throughout the verification process of the system, from design stage to qualification. For these reasons, it is characterized by an intrinsic modular architecture, as described hereafter. The design of the simulator follows the guidelines given by Eichkoff in [2]. A key feature of the simulator is the use of freeware. However, well known commercial products have been used to "validate" the HIL simulation results wherever possible.

### *A. Simulator architecture*

Fig. 2 illustrates the global architecture of the HIL simulator. The two main blocks represent the simulation station and the test object, on the left hand side and on the right hand side respectively. The blocks are connected by the GSE (Ground Support Equipment) box. The simulation station includes a user interface represented by the Control Console block, whereas the test object interfaces also with the Ground Control Station (GCS) via Radio Frequency.

The simulation unit is constituted by four main elements: 1) the simulator core, 2) the mission models, 3) the equipment models, and 4) the interface unit.

The simulator core is constituted by a network of computers (PCs) with Linux as Operating System (OS). All the OS's unnecessary functionalities are inhibited in order to guarantee as much as possible the real time activity of the simulator. This C++ program is able to: 1) schedule all tasks according to priorities, and time and logical sequence, 2) manage the logical interfaces with the test-object, i.e. the e-st@r-I CubeSat, 3) manage the I/O with the user through the Control Console (display, keyboard and mouse), and 4) save all the information related to the simulation session. Moreover, *ad hoc* kernels are defined to allow different types of simulations (real-time or off real-time) for several arrangements of the test object hardware. The interface unit must guarantee communications between the simulation unit and the satellite. The communication between the simulation unit and two programmable power packs (ISO-TECH IPS 3202), which are part of the GSE, is made through a RS232 serial interface. The power units provide the correct values of currents and voltages to the solar panels connectors on the Power Control and Distribution Unit (PCDU) of the EPS, as if the five solar panels were in orbit. The communication to and from the A-ADCS ARM9 is provided by a serial cable; this communication is also used for debug purposes. The physical interfaces of the HIL test bench are shown in Fig. 3.

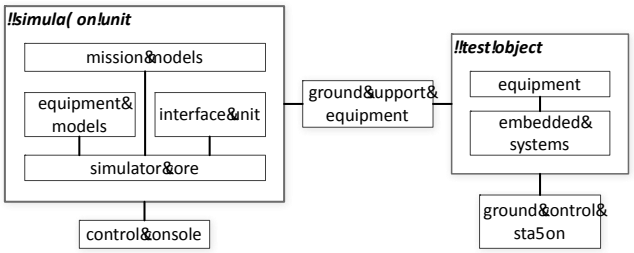


**Fig. 2. Simulator architecture.**

Channel 0 (ttyUSB0) connects the simulation unit to MAX232 and MAX232 to ARM9's debug port on the A-ADCS board. The A-ADCS board returns the Pulse Width Modulation (PWM) commands to the simulation unit. Through this channel, the operator controls the A-ADCS's behavior in real time.

Channel 1 (ttyUSB1) connects the simulation unit to power pack N1. The simulation software conveys solar panels current and voltage values to set power pack N1 in order to simulate solar panels +x, +y, -y. The three power pack's outputs are connected directly to the six-pin connectors on the PCDU.

Channel 2 (ttyUSB2) connects the simulation unit to power pack N2. The simulation software conveys solar panels current and voltage values to power pack N2 in order to simulate solar panels +z and -z . Two out of the three power pack's outputs are connected directly to the six-pin connectors on the PCDU.

Channel 3 (ttyUSB3) connects the simulation unit to the IMU on the A-ADCS board. Simulated IMU's raw data are transmitted to the A-ADCS board in terms of angular velocities and local EMF, formatted as specified in the IMU data-sheet.

The simulation unit also includes electrical emulation of sensors and actuators. The output value of each sensor is computed by the simulator and it is read by the embedded system under test. In the same way, the embedded system commands the actuators by means of proper control signals and it sends back to the simulator the control values.

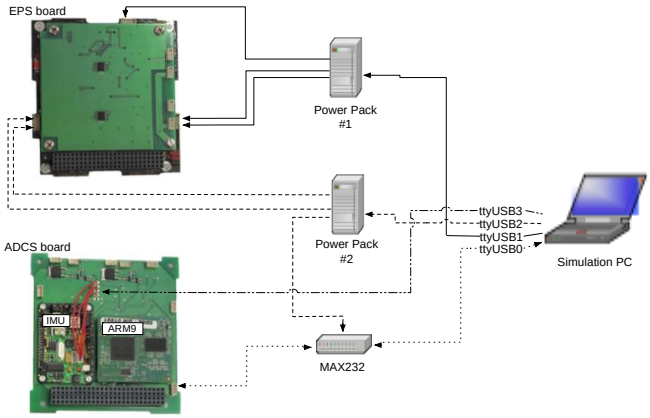


**Fig. 3. Simulator interfaces. See text for discussion.**

The fundamental part of any simulation is the ability of modeling the elements of the system under study. A database with different mathematical and stochastic models, divided in several categories, has been built and is periodically updated. The simulator has been designed according to the principle that any simulated system should behave as the real one not only in terms of numerical or physical outputs, but also in terms of interface with the other elements in the simulation loop. Potential anomalies in actual sensor's measurements shall also be reproduced in the virtual sensor. The level of models' detail shall be carefully evaluated taking into account the accuracy of the simulation on the one side, and the computational cost on the other side. Usually, increasingly more detailed models are used according to the advancement of the design. The models come from literature when available, or they are derived from information given by the manufacturers, or they can be obtained from data acquired during open-loop testing of the specific equipment.

The simulated satellite's equipment are:

1) solar panels and EPS's temperature sensors. Their models have been developed from data according to [26]. Using the simulated information of Sun position, satellite attitude, and thermal conditions, the voltage and current supplied by each panel are computed. These information are used to set the power units that are connected to the EPS board to power the CubeSat with the desired voltages and currents
2) IMU. The models of the gyroscopes and of the tri-axial magnetometer are included in the simulator [27]. These models are used in the attitude determination loop
3) MT. The models of the magnetic actuators [27] and the PWM drivers [28] are used to calculate the magnetic dipole that, together with the EMF simulated data, generates the control torque.

The orbital conditions are simulated through the models of: 1) the orbit by a Simplified General Perturbation propagator (SGP4); 2) the rotational dynamics and kinematics of the satellite; 3) the thermal flows hitting the satellite [29]; and 4) the EMF [30]. The orbital position is necessary for obtaining the local sun vector and the local magnetic field. The sun vector is used to calculate thermal flows and temperatures on solar panels. All these models are implemented in the simulation unit of the HIL simulator. The test object used for the simulation described in this paper is the e-st@r-I CubeSat. It can be split into two main parts, as shown in Fig. 3. The "equipment" block includes D-PCDU, batteries and antenna, i.e. all those devices without processing units and which need only to be powered. The "embedded system" block encompasses EPS PCDU, COMSYS board, A-ADCS board and OBC, i.e. all those units with micro-processors. Sensors and actuators of the CubeSat are electrically connected to the power bus, but only for verification of total power consumption. They are not used for sensing data nor for generating torques.

### *B. Working principle*

This section explains how the simulator works. The working cycle is a closed loop illustrated in Fig. 4.

After the simulation is initialized, PWM commands are calculated in terms of duty cycles. These commands are sent to the simulation unit through Channel 0. The simulation software calculates the applied voltages, and the currents flowing into the MTs. In the end, the generated dipole moment is determined.

The dipole moment together with the Earth's magnetic field (measured by the modeled magnetometer) generate the value of the magnetic control torque. Using the dynamics and kinematics models of the CubeSat it is possible to obtain the values of the angular velocities around the satellite's body axes with respect to the inertial reference frame.

The angular velocities values (measured by gyros) and EMF values (measured by magnetometer) are converted in data strings in order to emulate the actual IMU's behavior; they are finally transmitted to the A-ADCS board through Channel 3.

Similarly, the simulation software determines the heat fluxes on each face of the CubeSat and the temperatures on each solar panel. Values of heat fluxes and temperatures enter the solar cells model in order to get the power generated by solar panels in terms of voltage and current. These information are transmitted to the two 3-channels power packs which supply the onboard PCDU. The onboard software calculates the Sun vector which, together with EMF and angular velocities, allows to determine the satellite's attitude. These information are used by the A-ADCS board to control the attitude of the satellite computing the new PWM duty cycles and finally sending them back to the simulation unit. Telemetry is also radio-transmitted by the CubeSat to the GCS where it can be visualized and saved for further analysis. The CubeSat may also receive commands from GCS during a simulation session.

### *C. Test settings*

Key parameters are defined before the simulation starts, such as the duration of the simulation, the time step, and the models to be used. CubeSat's characteristics are also chosen and their values (peak or trends) are continuously monitored as they shall be verified against requirements. For e-st@r-I, the parameters are the EPS and ADCS telemetries and other system's health and status data.

The analysis of simulation results is made from data collected through three independent sources: 1) GCS log files: updated every time that a radio communication between the satellite and the GCS occurs and a new packet, containing the telemetry data, is received; 2) Satellite SD card files: updated every 2 minutes; they contain the same telemetry strings received by the GCS but they are free of transmission errors; 3) Simulator log files: upated every 0.5 seconds; they hold all data about the environmental and system simulation and not only the telemetry data.

At the present moment, data analysis is performed off real time, following a procedure that consists of importing data from

files, handling, converting and interpreting them. The most meaningful parameters are chosen and the respective graphs are plotted and evaluated.

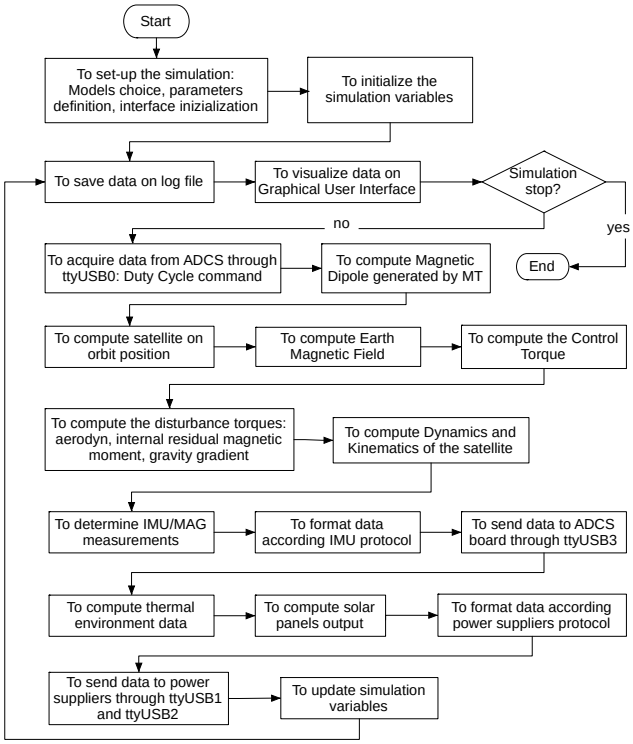


**Fig. 4: simulation software flow chart**

## V. RESULTS AND DISCUSSION

In this section, the results of a simulation session are presented and discussed. The global objective of the campaign was the study of the satellite's behavior in the first day of in-orbit operations. The test aims at demonstrating that the CubeSat performs according to a reference mission as described hereafter. Mission and orbit data are given in Table 1.

**TABLE I**
**MISSION AND ORBIT DATA**

| | |
|---|---|
| Semi-major axis [km] | 7359.46 |
| Eccentricity | 0.06664 |
| Inclination [deg] | 69.5 |
| RAAN | 0 |
| Argument of perigee [deg] | 130 |
| Mean anomaly [deg] | 0 |
| Epoch | 1 Jan 2012 |
| Date of launch | 13 Feb 2012 |
| Launch site | Kourou, French Guyana |
| Launch Vehicle | Vega |

The CubeSat is released into orbit at time T0. It enters the *detumbling* mode during which the A-ADCS reduces the angular velocities until threshold values are reached. For the purpose of the HIL simulation, the value of 0.5 rad/s has been assumed as

initial angular velocity of the CubeSat after release from the deployer (typical reference values are in the range 0.1-1.0 rad/s). The detumbling phase is considered completed when the rotation rate decreases below 0.005 rad/s. The CubeSat remains in this mode of operations for about 2 hours. At T0 + 2 hours: 1) the antenna deploys autonomously, 2) COMSYS is activated by OBC, and 3) the communication link from/to Earth is established. In the nominal mission, telemetry packets are sent to GCS every 2 minutes. According to the test plan, commands from the GCS are sent to the CubeSat to update mission parameters and to change modes of operations. At the end of the simulation session, the log files are analyzed to evaluate the CubeSat's behavior, and to verify if the requirements are met. In the following graphs, most interesting parameters' trends are plotted from data in the GCS log file. The simulation has been running for 100000 seconds.

In Fig. 5, battery-bus voltage, 5V-bus current, and 3.3V-bus current trends are shown. The battery bus voltage graph shows that the batteries are recharged during the orbits, reaching the maximum voltage value equal to 8.17 V in less than 10 hours (given a starting value of 7.80 V). Power consumption on the 5V-bus is almost constant. The peak in the plot is due to a wrong data acquisition; power consumption on the 3.3V-bus varies depending upon the use of OBC and A-ADCS boards.

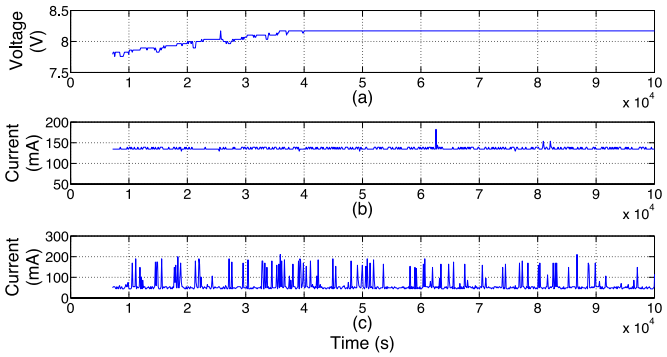


**Fig. 5. (a) Battery-bus voltage as a function of time, (b) 5V-bus current as a function of time, and (c) 3.3V-bus current as a function of time**

In Fig. 6, the trends of relevant parameters of one battery cell are shown. Each battery pack is constituted by two Li-Ion (Lithium Ions) cells. As far as voltage of cell is concerned, the graph confirms the trend seen in Fig. 5 for the battery-bus voltage. The plot of the current shows consumption peaks which depend on the variable instantaneous operations of the subsystems. Since the test was performed in laboratory conditions, the temperature is almost constant and is equal to about 27 °C (which is higher than the room temperature, due to the fact that the battery heats up during operations, in particular during charge).

Fig. 7 illustrates the plots of solar panels voltages during the one-day mission, showing the sunlight/eclipse cycles. The graphs show that the satellite experiences, within the same day, both orbits with short-eclipse/long-sunlight periods and orbits with no eclipse. This is due to the combination of orbit geometry and launch date. It represents a transition condition toward a one-week period of full sunlight. During the simulated mission, the power consumption of the CubeSat is quite low, as only vital functions

are executed. Moreover, Fig. 7 shows very short-duration eclipses. For these reasons, we can see very light voltage drops in Fig. 5(a).

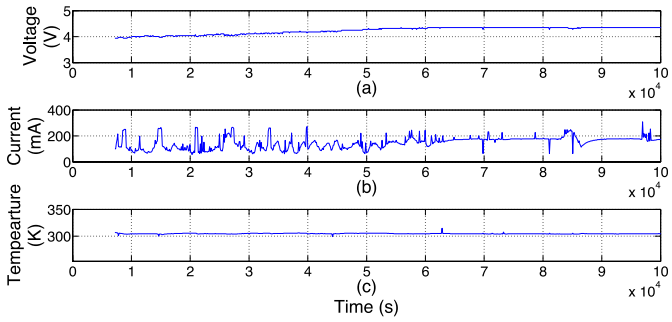


**Fig. 6. Telemetries of battery pack no. 1 as a function of time. (a) Voltage, (b) current, and (c) temperature of one cell of battery pack.**

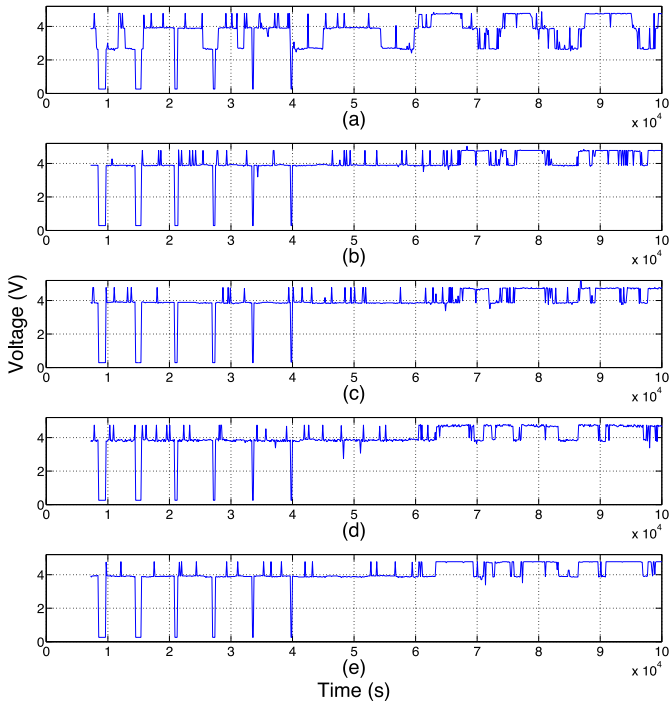


**Fig. 7. Solar panel voltages. (a) panel +x, (b) panel +y, (c) panel –y, (d) panel +z, and (e) panel –z. Panels are named after the perpendicular vector of the face they lie on, in body axes.**

After the *detumbling* phase ends, the angular velocities of the CubeSat remain constant with respect to the inertial frame, therefore the satellite is not spinning anymore. In Fig. 8, it is possible to notice that the angular velocities with respect to the orbital frame is close to 0. The trends of the angular velocities during the *detumbling* phase are not visible in Fig. 8 because the graphs have been plotted from the GCS log file which does not record any data before the *detumbling* phase is completed. The peak in the $\omega_x$ plot is ascribable to an acquisition error.

Fig. 9 shows that the desired attitude (antenna pointing to the ground, reference quaternion [0 0 0 1]) is achieved with the

expected accuracy in the first day of the mission. The pointing error measured at the end of the test is less than 10 deg.

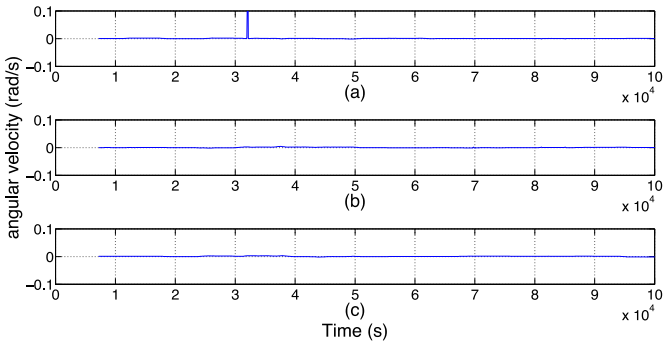


**Fig. 8. Angular velocities as a function of time. (a) angular velocity along x axis, (b) angular velocity along y axis, and (c) angular velocity along z axis.**

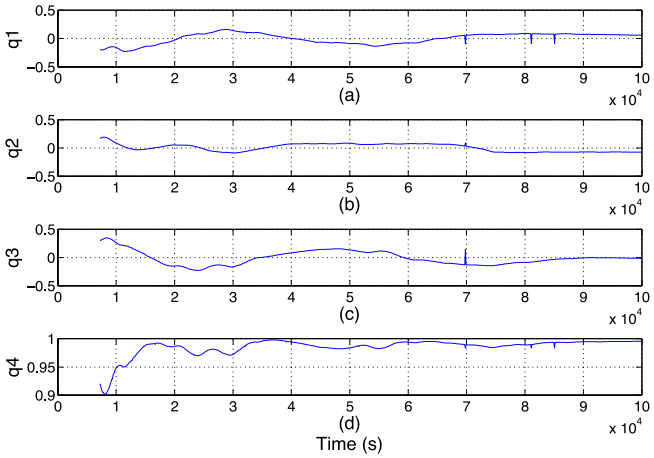


**Fig. 9. Quaternion. (a) first component of quaternion vector part, (b) second component of quaternion vector part, (c) third component of quaternion vector part, and (d) quaternion scalar part.**

Moreover, from all graphs, it is possible to see that no data are received at GCS during the first 2 hours after injection into orbit. This was one of the operational requirements to be verified. The set of requirements and constraints verified in the same HIL simulation session is reported in Table 2. In addition to the closeout of these requirements, other interesting information are obtained from the test:

1) No packet is lost (not saved and/or not received) in one-day length simulation
2) One system reset occurred over one-day length simulation, demonstrating the robustness of the software design
3) 3 out of 720 packets contain no-coherent data (comparison between data saved on the onboard SD card and data recorded in the GCS log file). The data error rate (measured in terms of packets per day) is less than $5x10^{-3}$.

## VI. Conclusion

The paper shows an application of the HIL simulation technique based on a custom simulator which is under development at Politecnico di Torino. The simulator has been developed to provide the engineering team with a versatile tool which can be used throughout the design and development process of a space system. Thanks to its modular architecture, the simulator shows a good flexibility with respect to project phases and types of missions/systems. It may be tailored for different applications without major changes and, in principle, it can handle simulation of several kinds of space missions and systems. Simulation models may be loaded in the simulator according to the specific design stage: models with different levels of detail and complexity are available and new models can be implemented and integrated in the simulator in the future. The type of mission of interest can also be chosen by setting the simulator in terms of specific mission parameters and systems involved in the simulation session.

At the present moment, only simple designs can be investigated through this platform, but it can be extended for testing more complex mission without re-designing the simulator's architecture. Some improvements can be done in order to enhance the simulator performance and accuracy, as well as to simplify the test bench and to reduce the time for test set up. Most common interfaces are already implemented in the simulator, but we are working to include other protocols. For example, the integration of the CAN bus interface could make the simulator suitable for testing pieces of hardware not supported at present. The addition of a robotic arm for improving attitude simulation capability is also planned.

The HIL simulator has been used for the verification of the functional and operational requirements of the e-st@r-I CubeSat. A one-day mission has been simulated during which the CubeSat demonstrated to be able to accomplish the required functions and to operate as expected.

The whole mission simulation has been carried out without relevant errors, showing that the HIL simulator can support the verification process of a CubeSat. A large number of requirements have been verified within a single simulation session, thus saving resources in terms of time for test set-up and running, man-hours and costs if compared to other testing techniques. Moreover, the test allowed to follow out the behavior of the equipment included in the simulation loop. For example, the actual power consumption of each piece of hardware has been measured and used to refine the power budget.

The HIL testing results have been compared with AIL and SIL simulations. These comparisons showed a good correlation between the different simulations, even if some differences have been noticed. For example, using the HIL facility helped to identify unexpected interactions between different devices during the integration of the CubeSat. The design has been reviewed to eliminate these potential failure causes using the information given by the HIL simulations. In particular, the onboard software has been adjusted as the HIL simulation unveiled synchronization conflicts between processes running on different onboard microcontrollers. A few tasks have then been re-scheduled in the ADCS and OBC software taking into account communication delays or overlapping between the two systems. Another interesting result of the HIL test has been the verification of the

conversion equations used to obtain the housekeeping telemetry of the EPS. In fact, this is a COTS equipment for which the conversion relationships were given by the supplier, and the HIL simulation allowed to refine these equations. In the case of in-house developed systems, the HIL simulation has been used for calibration of sensors, such as for example the gyros, after their integration in the CubeSat.

In conclusion, it has been proven that HIL simulation can effectively support the AIV process of a CubeSat. Improving the verification activity of CubeSats is a key factor for the development of innovative CubeSat missions. It has been demonstrated that a simple HIL architecture can serve well the purpose, even if some limitations exist. The HIL simulator's capability can be further extended to support a broader class of applications. Over one hundred CubeSats have been launched into orbit in the last decade and exciting missions are under development. The definition and standardization of methods and tools for supporting the development of CubeSat projects could be a key element to increase mission success while maintaining their peculiar features of low cost and fast delivery. This paper is intended to share our experience on the verification process of the e-st@r platform with other universities which are running or want to start CubeSat programs, and to show how university projects can contribute to the advance of science and technology while educating students.

**TABLE II**
**REQUIREMENTS AND CONSTRAINTS (VERIFIED BY HIL SIMULATION)**

| Description | Verification closeout | Remarks |
|---|---|---|
| The CubeSat shall send telemetry packets every 120 ± 15s | Verified | |
| Telemetry packets shall be formatted according to KISS AX.25 protocol | Verified | |
| Telemetry shall be stored on board on a non volatile memory (SD card) | Verified | |
| The CubeSat shall execute commands received from GCS | Verified | |
| Current consumption @5V shall be less than 150 mA in nominal mode of operation (mean value) | Verified | See Fig. 5(b) |
| Current consumption @3.3V shall be less than 180 mA in nominal mode of operation (mean value) | Verified | See Fig. 5(c) |
| Current consumption @5V shall be less than 200 mA in nominal mode of operation (peak value) | Verified | See Fig. 5(b) |
| Current consumption @3.3V shall be less than 230 mA in nominal mode of operation (peak value) | Verified | See Fig. 5(c) |
| Battery shall be charged in less than 10 hours | Verified | See Fig. 5(a) |
| CubeSat shall determine its attitude | Verified | See Fig. 9 |
| CubeSat shall control its attitude | Verified | See Fig. 8 and 9 |
| CubeSat shall be able to maintain its attitude | Verified | See Fig. 9 |
| The pointing accuracy shall be at least 10° | Verified | See Fig. 9 |
| At the end of the detumbling phase, the angular velocities shall not exceed 0.005 rad/s for each axis | Verified | See Fig. 8 |
| Current consumption of each MT shall be less than 150 mA | Verified | |
| Antenna deployment shall not create any space debris | Verified | By visual inspection |
| Antenna shall wait to deploy a minimum of 30 minutes after the CubeSat deployment switch (DS) is activated from P-POD ejection | Verified | Antenna deployed 2 hours after DS activation |
| RF transmitter shall wait to transmit a minimum of 30 minutes after the CubeSat's DS is activated from P-POD ejection | Verified | No transmission to GCS in the first 2 hours (Fig. 5 to 9) |
| The CubeSat shall be able to receive a transmitter shutdown command, as per FCC regulation | Verified | |
| The CubeSat shall de able to switch from one operation mode to another operation mode upon command from GCS | Verified | |
| No electronics shall be active during launch to prevent any electrical or RF interference with the launch vehicle and primary payloads. | Verified | |
| All systems shall be turned off by the DS, including real time clocks | Verified | |

## REFERENCES


[1] J. A. Ledin, "Hardware in The Loop Simulation", *Embedded Systems Programming*, Vol.12(2), pp. 42-60, 1999. https://edit.ethz.ch/idsc/Courses/embedded_control_systems/Exercises/Hardware-in-the-Loop.pdf,last access July 2013.

[2] J. Eickhoff, *Simulating Spacecraft Systems*, New York: Springer, 2009.

[3] J. S. Hall, M. Romano, "Laboratory Experimentation of Guidance and Control of Spacecraft During On-orbit Proximity Maneuvers," in *Mechatronic Systems Simulation Modeling and Control*, A. Milella, D. Di Paola, G. Cicirelli, Eds. Rijeka, Croatia: InTech, 2010, ch. 11, pp. 187-225.

[4] J. de Carufel, E. Martin, J.C. Piedboeuf, "Control strategies for hardware-in-the-loop simulation of flexible space robots", *IEE Proc. Control Theory and Applications*, 2000, vol.147(6), pp. 569-579.

[5] T. Boge, T. Wimmer, O. Ma, M. Zebenay, "EPOS: A Robotics-Based Hardware-in-the-Loop Simulator for Simulating Satellite RvD Operations", *i-SAIRAS*, Sapporo, Japan, 2010.

[6] R.Isermann, J. Schaffnit, S.Sinsel, "Hardware-in-the-Loop simulation for the design and testing of engine-control systems", *Control Engineering Practice*, vol. 7(5), pp. 643-653,1999.

[7] E.R. Mueller, "Hardware-in-the-loop Simulation Design for Evaluation of Unmanned Aerial Vehicle Control Systems", *AIAA Modeling and Simulation Technologies Conference*, Collection of Technical papers, 2007, vol. 1, pp. 530-543, Hilton Head, SC.

[8] N.R. Gans, W.E. Dixon, R. Lind, A. Kurdila, "A hardware in the loop simulation platform for vision-based control of unmanned air vehicles", *Mechatronics*, vol. 19, pp. 1043-1056, 2009. doi: 10.1016/j.mechatronics.2009.06.014

[9] E. Tara, S. Filizadeh, E. Dirks, "Battery-in-the-Loop Simulation of a Planetary-Gear-Based Hybrid Electric Vehicle", *IEEE Trans. Veh. Technol.*, vol. 62(2), Feb, 2013. doi 10.1109/TVT.2012.222.6921

[10] D. Bullock, B. Johnson, R. Wells, M. Kyted, Z. Li, "Hardware In the Loop simulation", *Transportation Research Part C: Emerging Technologies*, vol.12(1), pp. 73-89, 2004.

[11] X. Wu, T. Vladimirova, "Hardware-in-Loop Simulation of a Satellite Sensor Network for Distributed Space Applications", *Conference on Adaptive Hardware and System*, NASA/ESA, 2008, pp. 424 – 431.

[12] D. Kim, S.Y. Park, J.W. Kim, K.H. Choi, "Development of a Hardware In-Loop (HIL) Simulator for Spacecraft Attitude Control Using Momentum Wheels", *Journal of Astronomy and Space Sciences*, vol. 25(4), pp. 347-360, 2008.

[13] R. Finnset, S. K. Rao, J. Antonsen, "Real time hardware-in-loop simulation of ESMO satellite attitude control system", *Model Identification and Control Journal*, vol. 27(2), pp. 125-140, 2006.

[14] A. Toorian, K. Diaz, S. Lee, "The cubesat approach to space access", *Proc. IEEE Aerospace Conference*, pp. 1-14, 2008.

[15] CubeSat Design Specification Rev12, The CubeSat Program, Cal Poly SLO, 2009. http://dww.cubesat.org/images/developers/cds_rev13_final.pdf, last access July 2013.

[16] J. Bouwmeester, J. Guo, "Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology", *Acta Astronautica*, vol. 67(7-8), pp. 854-862, 2010.

[17] K. Woellert, P. Ehrenfreund, A. J. Ricco, H. Hertzfeld. "Cubesats: Cost-effective science and technology platforms for emerging and developing nations", *Advances in Space Research*, vol. 47, pp. 663-684, 2011

[18] R. Staehle, et al., "Interplanetary CubeSats: Opening the Solar System to a Broad Community al Lower Cost", Journal of Small Satellites, vol.2(1), pp. 161-186, 2013

[19] G. F. Dubos, J. Castet, J. H. Saleh, "Statistical reliability analysis of satellites by mass category: does spacecraft size matter?", *Acta Astronautica*, vol. 67(1-2), pp. 584-595, 2010.

[20] S. Corpino, S. Chiesa, F. Stesina, N. Viola, "*Small satellite and Cubesat missions: ten years of activities at Politecnico di Torino",* 2nd IAA Conference on University Satellite Mission and Cubesat Workshop, Rome (Italy), 2013.

[21] M. Cho, H. Masui, T. Hatamura, K. Date, S. Horii, S. Obata, "Overview of Nano-satellite Environmental Tests Standardization Project: Test Campaign and Standard Draft", 26th Annual AIAA/USU Conference on Small Satellites, 2012.

[22] N. K. Ure, Y. B. Kaya, G. Inalhan, "The Development of a Software and Hardware-in-The- Loop Test System for ITU-PSAT II Nano Satellite ADCS", *Proc. IEEE Aerospace Conference*, Big Sky, MT, 2011.

[23] M. Raif , U. Walter, J. Bouwmeester, "Dynamic system simulation of small satellite projects", *Acta Astronautica*, vol.67, pp 1138–1156, 2010.

[24] F. Stesina, S. Corpino, R. Mozzillo, G. Obiols Rabasa, "Design of the active attitude determination and control system for e-st@r Cubesat", *Proceedings of the 63rd IAC*, Naples (Italy), 2012.

[25] S. Corpino, F. Stesina, "Hardware In The Loop Test Campaign for E-ST@R Cubesat", *The 4S-Symposium (ESA)*, Portoroz (Slovenja), 2012.

[26] A. K. Hayder, R. L. Wiley, G. Halpert, D. J. Flood, S. Sabripour, *Spacecraft Power Technologies*, London, UK: Imperial College Press, 2003.

[27] Computer Science Corporation (Members of the Technical Staff Attitude Systems Operation), *Spacecraft Attitude Determination and Control*, J. R. Wertz, Ed. Dordrecht, The Netherlands: D. Reidel Publishing Company, 1978.

[28] M. Polites, C. Quarles, D. Kaderbek, "Pulse Width Modulating Low Power Magnetic Torquers for Precise Spacecraft Attitude Stabilization", *Proceedings of the Institution of Mechanical Engineers Part G: Journal of Aerospace Engineering*, vol.219, pp. 471-482, 2005.

[29] D. G. Gilmore, *Spacecraft Thermal Control Handbook, Volume 1: Fundamental Technologies*, D. G. Gilmore, Ed. Reston, VA: American Institute of Aeronautics and Astronautics, Inc., 2002.

[30] International Association of Geomagnetism and Aeronomy (Members of the Working Group V-MOD), "International Geomagnetic Reference Field: the eleventh generation", *Geophysical Journal International*, vol.183(3), pp 1216–1230, 2010.



**Sabrina Corpino** received the MSc degree in Aerospace Engineering from Politecnico di Torino, Torino, Italy, in 1999 and the PhD degree in Aerospace Engineering from Politecnico di Torino, Torino, Italy, in 2003.

She has been working as Assistant Professor at the Department of Mechanical and Aerospace Engineering of Politecnico di Torino since July 2003 and has been confirmed in June 2009. Since 2011 she is Professore Aggregato with tenure of the module "Space systems and missions design" of the Master degree in Aerospace Engineering. She is responsible for all satellites' programs developed by the Aerospace Systems Engineering Team, ASSET. She is program manager of the CubeSat program at Politecnico di Torino, which involves other researchers and many students. Her skills and interests led her to take part in many past and present research programs, funded by public institutions and private companies. She is author of several articles, conference proceedings and book chapters. Her main scientific interests are: Systems of Systems analysis, Design Methodologies for Aerospace Systems, Techniques for systems simulation and verification.



**Fabrizio Stesina** received the MSc degree in Computer Engineering from Politecnico di Torino, Torino, Italy, in 2004. He is currently pursuing the PhD degree in Aerospace Engineering at Politecnico di Torino, Torino, Italy.

From 2004 to 2010, he was Research Assistant with the Department of Mechanical and Aerospace Engineering of Politecnico di Torino. He has been working on small satellite programs carried out within the Aerospace Systems Engineering Team, ASSET, and he has the role of System Engineer in CubeSat student projects. He is also participating in a research program on space debris removal funded by the Piedmont Regional Government and led by Aviospace srl (an EADS-Astrium Company). His main scientific interests are: Guidance, Navigation and Control Systems for space applications, Control Theory, and Techniques for systems simulation and verification.