SpaceOps Conferences
16-20 May 2016, Daejeon, Korea
14th International Conference on Space Operations

OPS-SAT: FDIR Design on a Mission that Expects Bugs - and Lots of Them

David Evans[1] and Manuel Ortega[2]
ESA/ESOC, Darmstadt, D-64293, Germany

Reinhard Zeif [3]
TU Graz, Graz, 8100, Austria

Tom Sergert[4]
Berlin Space Technologies GmbH. Max-Planck-Str. 3, 12489 Berlin, Germany

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

This paper describes the unique Fault Detection, Isolation and Recovery (FDIR) concept of the ESA OPS-SAT project. OPS-SAT is ESA's first nanosatellite mission and is the first mission world-wide to be designed exclusively to demonstrate ground-breaking satellite and ground control software under real flight conditions. The FDIR system is unique in several ways not least because the mission concept involves replacing the entire on-board software suite, right down to operating system, on a daily basis. This also extends to the ground system which must be designed to be replaceable on the same time scales. The mission will deal with the very acute problem of allowing experimenters to load on a daily basis their own on-board software and ground software with minimal testing. The solution is to design the FDIR system to concentrate on ensuring survivability of the spacecraft rather than the availability of the spacecraft. After survivability, the next priority is ensuring the ability of the ground to recover the mission when a fault is detected. There are many ways into and out of the satellite, some quite unconventional such as a laser uplink path. To ensure a recovery path, the system required to be designed against the normal rules, for example allowing a mechanism to reboot the spacecraft receivers. ESOC is in a unique position for this project, being the first time in control of the Space, ground and payload segment design simultaneously. Hence trade-offs between these three areas are more easily made than in more traditional missions. This paper describes the harmonious interaction between these three areas through the chosen solution of the FDIR system and its current status of implementation, ensuring the success of the mission.

I. Introduction

OPERATING a space mission is not easy. The space environment is terribly unforgiving and it is not the ideal place to be testing new pieces of complicated bespoke hardware or software. Mission critical software, both on-board and on the ground, is therefore selected for its proven, rock-solid reliability rather than other considerations. This results in resistance to experimentation and innovation, especially when the projected benefits are not yet flight proven. Time after time projects settle for reuse rather than innovation.

OPS-SAT is ESA's first nanosatellite mission and is the first mission world-wide to be designed exclusively to demonstrate ground-breaking satellite and ground control software under real flight conditions. The project is being led by the European Space Operations Centre (ESOC) in Germany which has recognised the need to try something very different to break out of the "has never flown, will never fly" cycle in the mission control domain. Having

[1] Project Manager, Special Projects Division, ESA/ESOC
[2] Project Assistant, Special Projects Division, ESA/ESOC
[3] Dipl. Ing./ Msc. System Engineering and Development, IKS TUGraz
[4] Director of Business Development, Beslin Space Technologies GmbH

Copyright © 2016 by European Space Agency. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.

recently undergone the Critical Design Review (CDR) at ESOC and ESTEC facilities, the project has presently passed Phase C and is scheduled to launch in Q3/4 2017.

One of the main mission requirements it that the satellite has to remain constantly safe. It is recognised that with little time for minimal preload, testing the ground and on-board software will contain dangerous bugs and potentially many of them. The concept also has to deal with the inevitable consequences of single event radiation effects on the commercial off-the-shelf hardware and software selected. The solution is to design a Fault Detection, Isolation and Recovery (FDIR) system that concentrates on ensuring the survivability of the spacecraft rather than the availability of the spacecraft. One example is an on-board FDIR system overlay that monitors the payload's current and voltage values centrally but independently of the on-board computer or data bus. In the event of a trigger it can use this bus to switch off the payload without any potential side effects of the problem blocking fast access.

This paper describes the mission, the main requirements and the space segment. It then describes the unique FDIR system that has been designed to fulfil OPS-SAT high safety requirements. First the main FDIR system requirements are presented allowing to quickly perceive the required capacities of this system. The Hardware and Software architectures are included in the paper as well as its Operational Modes. The paper concludes with an Assembly, Integration and Validation section, followed by future work and prospects.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

II. Mission History

OPS-SAT is an ESA nanosatellite mission designed exclusively to demonstrate ground-breaking satellite and ground control software under real flight conditions. This makes it the first mission of its kind worldwide. The project is being led by the European Space Operations Centre (ESOC) in Germany underlining it as a mission designed by operators for operators.

In 2011 the Advanced Operations Concepts Office at ESOC proposed the concept. In January 2012 the ESA General Study Programme funded a feasibility study using the ESA Concurrent Design Facility in ESA/ESTEC. At the end of the concurrent design sessions, the design team declared the project feasible with a preliminary design based on a 3U Cubesat.

Following the study, the ESOC team was contacted by many companies and other national space agencies who expressed an interest in flying their experiments on the satellite. In March 2013, ESA released an open call for experiment ideas. The response was overwhelming with over one hundred experiments from 17 Member States being selected for the next stage. This response was enough to convince the ESA General Support Technology Programme (GSTP) to open a call for Member States to fund the next mission phase. Austria, Germany and Denmark responded positively

The project then kicked off two parallel Phase AB1 studies, starting in July 2013, supported by GSTP. These were led by the Institute of Communication Networks and Satellite Communications of TU Graz (Austria) and GOMSpace of Denmark respectively. TU Graz was supported by Zentrum für Telematik and GOMSpace by GMV GmbH both in Germany.

Following completion of Phase AB1, TU-Graz put together a consortium and proposal to take the mission through Phase B/C/D/E. Poland joined as a supporting member state. GSTP then approved the entire mission and a consortium of the following companies led by TU Graz won the contract to design, test and build OPS-SAT. TU-Graz, MAGNA STYER & UNITEL (Austria); GOMSpace (Denmark); MEW Aerospace UG and Berlin Space Technologies (Germany) and finally SRC and GMV Innovating Solutions (Poland). Core avionics are delivered by GomSpace with core software from GMV. Notable hardware payloads are the advanced ADCS and camera for Berlin Space Technologies, a software defined radio and optical uplink receiver from MEW Aerospace, a CCSDS compatible TMTC decoder/encoder from SRC and a X band transmitter capable of transmitting at 50 Mbps from Syrlinks, France.

The mission CDR phase is on-going and planned to be completed in April 2016. Launch is planned for the second half of 2017.

III. Mission Requirements

The objective of the OPS-SAT mission is to drastically improve ESA mission control capabilities by providing an in-flight experimentation platform on which European industry and academia can explore the opportunities that arise when present barriers are removed. These barriers are the present low computer processing capabilities on-board satellites and the strict requirements on pre-flight testing/heritage imposed on mission critical software.

The mission will explore these opportunities by launching a low cost, low risk, experimental platform that flies the latest miniature processor hardware and reconfigurable firmware. Experimenters will be able to reconfigure any part of the end-to-end chain on ground or on-board to demonstrate new operational concepts under flight conditions

The OPS-SAT spacecraft will be designed to be extremely robust, in the sense that it can survive any possible impact caused by incorrectly running experimental software. It will be able to survive until contacted by the ground and then being recovered for the next experiment.

The driving requirements are as follows:

1) OPS-SAT shall allow experimentation with on-board and ground software by offering a safe and reconfigurable environment for execution of software experiments that are relevant for future mission operation needs at ESA.
2) The spacecraft shall be power and thermally safe even if tumbling. The mission shall be robust against single event upsets, latching events or faulty experimental software. It shall be demonstrated that despite using COTS components a reconfigurable and yet re-liable platform can be delivered.
3) The OPS-SAT payload shall deliver as a minimum; two processors running at 500+MHz with 500MB of RAM, 10 GB solid storage and a reconfigurable FPGA.
4) Software experiments shall have open access to all on-board resources and systems un-less justified due to safety.
5) At least one configuration shall be representative of an ESA mission (including ground software and OBSW).
6) S-Band uplink rates of at least 256 kbps and S-Band downlink rates of 1 Mbps shall be supported. The high uplink data rate is due to the fact that this mission is focused on new software experiments. Therefore it is needed to upload the frequently changing software experiments in reasonable times.
7) The spacecraft shall be recoverable and resettable by at least two independent communications routes in hardware and software. The spacecraft shall be able to communicate with the respective ground station in any orientation

ESA and its European industry partners generate every year many new and innovative ideas for advancing European space technology regarding mission operations but the majority of these innovations never make it to orbit. OPS-SAT emerged, providing a low cost in-orbit laboratory available for authorized experimenters to test, demonstrate and validate their development software experiments. OPS-SAT is the first CubeSat designed by ESA and is a safe experimental platform which shall fly in a low-earth sun synchronous orbit. OPS-SAT makes available a reconfigurable platform, at every layer from channel coding upwards, and it will be available for experimenters wishing to test and demonstrate new software and mission operation concepts.

IV. The Space Segment

OPS-SAT can be viewed as four interconnected parts: a cubesat bus, an ESA communications module, a payload and a FDIR system. The payload can be further broken down into a processing core, various peripherals (camera, GPS, advanced ADCS subsystem) and several payloads of opportunity. The cubesat bus consists of an on-board computer called the Nanomind, a power subsystem, a UHF communications subsystem and a basic ADCS subsystem.

The mechanical architecture of the OPS-Sat is a 3U CubeSat structure with double folded deployable solar panels. It has a size of 10x10x30 cm (not including deployable) and a mass of approx. 5.4 kg. Two deployable solar array panels generate 30 W of electrical (peak) power. The spacecraft outside front view and rear views is shown in Fig.1.

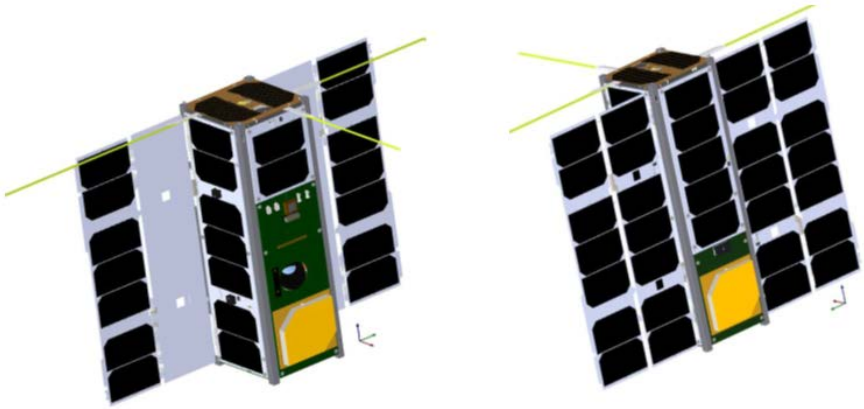

Figure 1. OPS-SAT front and rear views.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

A. Bus

Approximately 1U of the satellite accommodates the cubesat COTS components, including: the UHF antenna deployment system (as well as the software defined radio receiver payload antenna, see later), a motherboard with the UHF transceiver, the NanoMind OBC, the BP4 battery pack with 4 battery cells, the EPS system, consisting of a motherboard accommodating two power input boards for the body mounted and deployable solar array strings, and two power output boards for power regulation and distribution and the Z axis magnetorquer integrated in a PCB. The double deployable solar arrays are provided by ClydeSpace whereas the remaining components are provided by GomSpace. Using a quasi-single provider keeps the integration costs and risks of the satellite bus low. GOMSpace also provide the ground terminal for the UHF communications. The one area where some development was required on the COTS components was the power conditioning subsystem. The mission sometimes generates over 30 W of peak power and is connected in 11 strings. To accommodate this GOMSpace proposed an architecture with two separate power conditioning and distribution boards, each connected to half of the solar arrays on the satellite. Solar arrays cover the satellite except for the Z faces, as shown in Fig. 1. The deployable solar arrays have integrated sun sensors and magnetorquers. The bus also includes a GPS unit which is linked to the NanoMind OBC to provide GPS functionality to the ADCS system. The GPS antenna is integrated into the –X panel next to the umbilical connectors of the spacecraft. As the GPS will be integrated with the CubeSat bus and not directly to the processing platform, all data from the GPS can be made available over the CAN and I2C payload bus interface from the NanoMind to processing platform. In addition, the GPS will be used for timekeeping on the Nanomind OBC, which keeps the on-board time (OBT).

A driving mission requirement is that a satellite configuration should exist that is indistinguishable to the ground from a typical ESA satellite. Among other things this means that the spacecraft has to fly firmware and software that implement CCSDS protocols. The solution is to deploy the IP core of an ESA TM/TC encoder/decoder chip onto a commercially available FPGA. This unit is referred to as the CCSDS engine. This chain will be used for nominally communicating between the Nanomind OBC and the ESA ground control system. However it will be possible for the experimenters to bypass this unit and go directly between the S band transponder and the processing platform. This will allow configurations that use non-CCSDS protocols such as TCP/IP on the mission.

Since over 90% of the experimenters want to load large software images to the spacecraft as part of their experiment it was decided that the mission had to allow the fast upload. The S-band receivers must therefore be able to accept an uplink signal at 256 kbps. For comparison the UHF transceiver on the Cubesat bus can only support data rates of 9.6 kbps. This is much higher than the highest uplink rate for normal ESA spacecraft which is 4 kbps rising to a maximum of 64 kbps in some rare cases. This requirement is already driving innovation on the ground as during ground prototype testing ESA realized its ground segment could not pro-duce such a fast telecommand stream without major modifications. This presents us with a prime example of the nanosatellite world challenging long standing and accepted limitations in the world of big space. On-board the new EWC31 S-band transceiver from Syrlinks, France has been selected to provide this high uplink and a variable downlink rising to 1 Mbps.

B. Satellite Experimental Processing Platform (SEPP)

The Satellite Experimental Processing Platform (SEPP) is the heart of the OPS-SAT payload. It is a powerful ALTERA Cyclone V system-on-chip module with sufficient on-board memory in order to carry out advanced software and hardware experiments. It is the reconfigurable platform required on OPS-SAT on which all major experiments will be processed. The Altera Cyclone V SX System-on-Chip (SoC) digital core logic device provides with a 800MHz CPU clock and 1GB DDR3 RAM a powerful processing capability. All Altera SoC SX devices consist of an internal Hard Processing System (HPS) and a Field Programmable Gate Array (FPGA) portion. The Altera Cyclone V SX SoC HPS is a fully functional computer and contains a dual core ARM CPU with several built-in hardware blocks and device interfaces. It also has built-in error correction coding (ECC) features..

The system offers the possibility to use DDR2, LPDDR2 or DDR3 RAM. The ARM CPU is connected to a large number of HPS hardware blocks and interfaces. Bridges enable high speed data exchange between FPGA and HPS portions. Linux is used as default operation system (OS) for the SoC. All HPS blocks can be accessed from the installed OS application software. The HPS portion has to be configured at system startup to setup the system in accordance to the implemented hardware design. The SoC configuration data is part of the SEPP software image stored in the external memory. The image is based on the Altera reference Yocto Linux and U-Boot boot loader software.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

C. Fine ADCS

OPS-SAT contains two ADCS systems. One is provided as part of the bus and is referred to as the coarse ADCS. The control algorithms are implemented on the Nanomind OBC and it relies on magnetotorquers as actuators and sun sensors and magnetometers as sensors. The other is implemented as part of the payload and is referred to as the fine-pointing ADCS or iADCS. Experimenters can use this for carrying out attitude control experiments and to provide higher pointing accuracy for camera and optical data transmission experiments. Control algorithms can be placed directly on the iADCS FPGA or on the SEPP. The iADCS-100 by Berlin Space Technologies and their partner Hyperion Technologies has been chosen allowing a pointing accuracy well below 1°. The iADCS provides a set of high performance sensors and actuators such as the ST-200 star tracker and miniature reaction wheels. In combination with the proven ADCS algorithms derived from the LEOS microsatellites, the iADCS-100 offers a number of autonomous modes such as nadir pointing and target pointing that were before only available for larger spacecraft.

The evaluation of the experimenters' proposals showed that there is significant interest in camera experiments. Several experimenters will develop on-board processing algorithms for simple remote sensing applications.

D. Camera

The baseline for the OPS-SAT optical camera is the BST IMS-100. This is a small space camera developed by Berlin Space Technologies together with its partner Hyperion based on the ST200 star tracker. The ST200 has been developed and tested for the Earth Video Camera project for the International Space Station. Sensor and MCU have been tested with proton irradiation of 130MeV for a TID of 10 krad. It can provide still images as well as video, whereby image processing will be performed on the processor core (SEPP). For video download the X-band transmitter will be used. The camera performance is provided in the following table:

Parameter	Value	Comment
Resolution	53m	@ 600km orbit height
Field of View / Scene	108x103km	@ 600km orbit height
Channels	3	RGB via on sensor Bayer Pattern
Frame Rate (Burst Mode)	20	2040x1944px / 1280x780px
Frame Rate (Continuous)	1.3 / 5	2040x1944px / 1280x780px

Table 1. IMS-100 Camera Performance.

E. Syrlinks EWC27 X-band transmitter

This CNES funded mini X band transmitter from Syrlinks, France is capable of transmitting up to 50 Mbps. It was subsequently selected to fly because it has a great deal of synergy with the other experiments. The on-board camera will support both still image and streaming video modes and many experiments intend to exploit this. Such camera experiments will require substantial downlink data rates for which the EWC27 X-band transmitter will be needed, particularly when bearing in mind real-time applications and the short contact times (typically 10 minutes for a ground station pass, four times a day). In fact the EWC27 has already flown as part of the ESA/GOMSpace mission GOMX-3 which was re-leased from ISS on August 19th 2015. The unit has already been tested successfully at its maximum limit of 3 Mbps (due to ITU regulations).

F. Optical Receiver

This optical communications experiment will see an optical uplink for the first time on a Nanosat. It provides a transmission rate of 16 Kbps using a small optical receiver which fits into OPS-SAT. A photon counting module with a built-in multi-pixel photon counter is the heart of this system. A prototype of the receiver was tested on ground and it was demonstrated that transmissions with the specified data rate can be carried out. For the uplink the

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

Satellite Laser Ranging Station operated by TU Graz and the Space Research Institute of the Austrian Academy of Sciences at the Lustbühel Observatory shall be used. The optical receiver will be connected to the SEPP so that uplink data can be received and processed by on-board experimental software. The laser ranging station at TU Graz will be used as the ground segment for the experiment. The laser ranging station at TU Graz will be used as the ground segment for the experiment. Optical retro-reflectors on the surfaces of the nanosatellite will provide the means to locate and track the spacecraft by laser tracking & ranging stations, assisted by an on-board GPS which is integrated in the core avionics.

An interesting aspect of the experiment is that this will be the first time a nanosatellite has been communicated with via an optical channel. Also the application will allow the secure uplink of one-time pad encryption keys. These can be used on the RF downlink making the broadcast extremely secure. Such an experiment has never been done before.

G. Software Defined Radio Frontend

This is a very small radio front-end consisting of a tuner, down-converter and analogue to digital converter. Complex signal samples are delivered to the SEPP where signal processing (e.g. demodulation and decoding) can be performed by on-board experimental software. The SDR theoretically receives signals with a bandwidth of up to 14 MHz in a frequency range of 300MHz to 3GHz. This allows the monitoring and demodulation of radio signals for a wide frequency range. For OPS-SAT mission the SDR is customized in accordance to the used antennas. The received frequency range includes the radio amateur UHF bands and this community has indicated interest in using the mission to perform a spectrum survey of those bands to investigate the increasing interference problems in uplink.

H. Optical Retro-Reflectors on Panels for Attitude Determination

This passive experiment will allow attitude monitoring and precise tracking using laser stations on the ground. There is an interest, especially among the space debris community to investigate the spacecraft dynamics of asymmetric objects such as OPS-SAT when uncontrolled. This can be done on this mission by simply disabling the actuation for periods when the mission is running and after mission termination using the laser reflections to determine attitude sate and evolution as it slowly descends.

V. FDIR Primary System Requirements

To ensure that the 3U Cubesat remains safe throughout the whole mission time a clear set of subsystem requirements had to be derived. The main subsystem requirements for the FDIR system include:

1) The FDIR system shall detect and identify failures: Failure detection and identification is the main task of the FDIR system.
2) The FDIR system shall collect housekeeping data from subsystems periodically: There shall be constant monitoring of the subsystems. This data includes power statuses and voltages of the payloads.
3) The FDIR system shall collect housekeeping data from the payloads. This data includes power statuses and voltages of the payloads.
4) The housekeeping data shall be read from enabled devices only. Only devices that are switched on shall generate housekeeping data.
5) The FDIR system shall be able to log hardware and software reset events. After a hardware reset, it shall inform the operator about the nature of the reset.
6) The FDIR system shall be able to restart automatically after a hard system reset.
7) Essential data shall not be lost in case of a system reset or a FDIR system restart. After a reset, the data concerning the cause of the reset shall not be lost.
8) The FDIR system shall be able to resume operation after a hard reset. To safeguard the system, parameters on warning and fault limits need to be stored in the FDIR system.
9) The FDIR system shall forward messages to the Nanomind OBC in case of a parameter exceeding fault limits.
10) The FDIR system shall disable the payload device in case of a parameter exceeding fault limits. If the payload reaches a fault limit, it has to be switched off automatically to prevent damage.
11) It shall be possible for the ground to selectively modify, enable and inhibit all limits used for the spacecraft FDIR system: operations may prove that other limits for warning and fault are necessary. They need to be modifiable in flight.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

12) The FDIR system internal clock shall be configurable from the Nanomind. The FDIR system and Nanomind clock need to be synchronous to allow tracking of events and parameters.
13) The FDIR system shall implement I2C CSP commands for immediate/direct housekeeping data retrieval.

These requirements have driven the design of the FDIR system and the main contractor, Berlin Space Technologies (BST), has proposed a hardware and software architecture as described in section VI.

VI. FDIR system Overview

The approach followed for mission survival and recoverability was to design the FDIR system as an overlay of the rest of the spacecraft. The design had to be simple and robust therefore a cold redundant option was taken for the FDIR computer see FDIR OBC1 and OBC2 in Fig.2. The active computer collects housekeeping data such as temperature, current and voltage statuses of the payloads and ensures safety by providing reset capabilities when certain configurable limits are exceeded. In addition the FDIR computer can communicate to the ground via the Nanomind but also directly both in S-band and UHF, providing an essential communication route should the Nanomind or on-board buses become blocked. The cubesat can in essence function without the FDIR overlay, but with extra risk e.g. bugs inside the uploaded experiments. As depicted in the block diagram of Fig.2, the payloads are essentially slaved to the FDIR system who can directly activate and deactivate their power and data lines via the bus switches (BS). A bus switch consists of two parts, a power switch and a data interface switch. Both parts can be used independently of each other. The data bus switches safely separate data interfaces to allow bus usage even if a device generates a physical failure on one of its interfaces. The power switch safely separates payload module power lines by isolated switching, providing suitable protection from latching and protects the systems in case of overcurrent, over-voltage or under-voltage events.

As each payload has its own bus switch it means the FDIR system can isolate each individual unit for the data and power buses in the event of failure. The bus couplers allow the payloads to communicate their status data to the FDIR OBC and receive commands on a dedicated OBDH bus ensuring that even in the event of a prime OBDH bus lock up the FDIR system can function. This also provides independent measurements of the health state of the payloads during experiment monitoring in addition to those functions running on the Nanomind.

The system has been designed so that it can have direct communication with the ground, who can configure it directly via two different routes. One is through the CCSDS engine using the S-band antennas, see Fig.2. The second possibility is to communicate via the UHF antennas using direct CSP commands.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

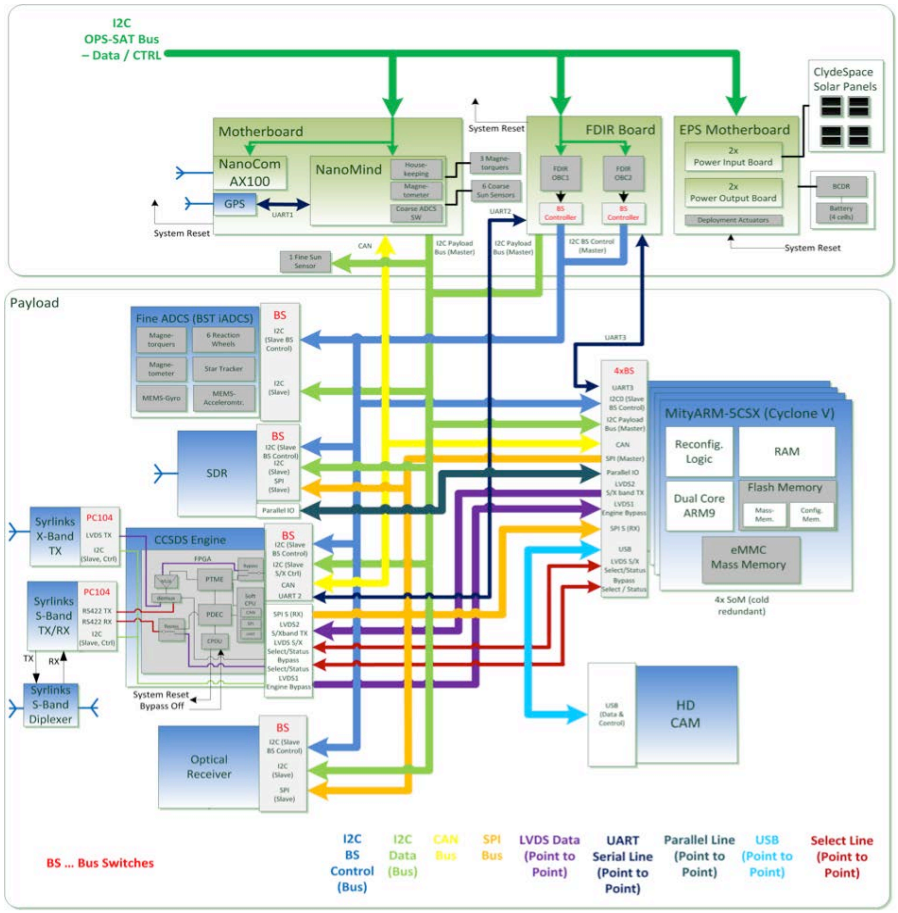

Figure 2. OPS-SAT Block Diagram.

To improve the robustness of the system, the FDIR system provides a hard reset of the spacecraft using a hammer circuit (Hammer Reset Timer in Fig.4). The hammer circuit is an independent circuit on the FDIR board which uses a counter with a fixed predefined value, initially set to 14 days. The counter triggers a spacecraft reset when the predefined value is reached and ensures that a reset occurs autonomously even if other reset mechanisms cannot be triggered. All FDIR events are also forwarded to the Nanomind OBC which also communicates regularly with the FDIR OBC as depicted by Fig. 3.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

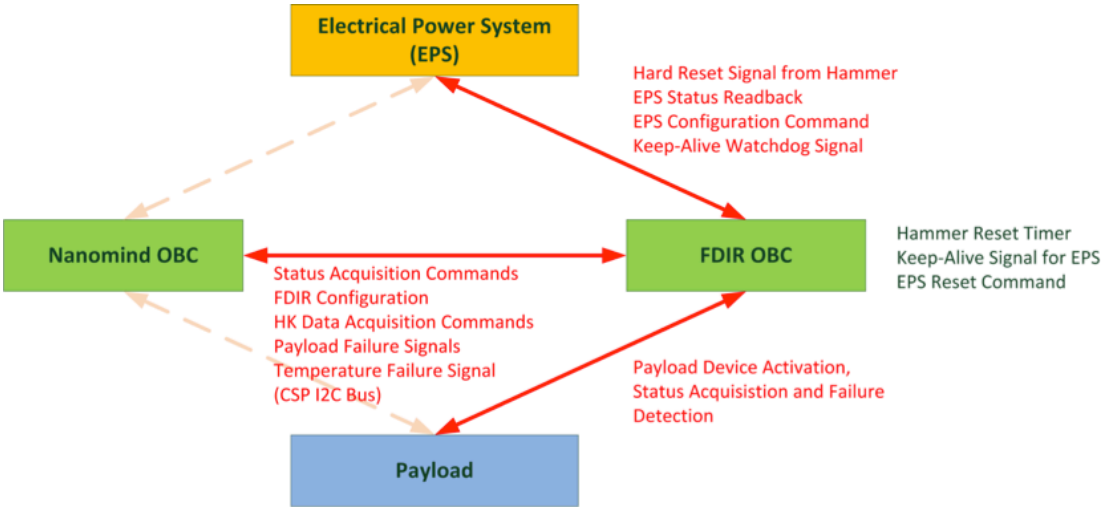

Figure 3. OPS-SAT FDIR bus capabilities

A. Hardware Architecture.

The FDIR system consists of two computing platforms (FDIR system #1 and #2) with two separate Micro Controller Units (MCU). Each one has a 2GB Flash memory to store the required telemetry in the event of failure, and as expressed before they are set in cold redundancy. Both units are connected to a 3.3V power supply through a De-Latch circuit, as seen in Fig.4. To transmit telemetry data towards the Nanomind an I2C Master (M) bus is used. A Payload I2C Data bus (I2C P/L Data in Fig.4) is implemented to communicate with the payload peripherals.

On the same FDIR system platform, several auxiliary components for the Cubesat have been implemented by BST. This includes three extra reaction wheels connected to the integrated payload ADCS (iADCS-100), a precision gyro (RS422) and a High Resolution camera. In the same board, the Hammer circuit that provides the hard system reset is built-in next to the bus switches that are connected to the payload peripherals.

Safe mode triggers are implemented in the FDIR board to safeguard the spacecraft against events, such as single-event-upsets, that leave the spacecraft in an unknown and potentially unsafe state. These triggers cause the spacecraft to go through a complete power-cycle and restart in a fully known configuration. Power cycling is performed by the Electrical Power Supply in the cubesat bus and can be initiated in three ways:

1) ***Hard Reset (trigger)****:* Autonomously by the EPS unit / FDIR hammer circuit.

2) ***Hard Reset (command)****:* By electrical signal connected to the CCSDS engine via the USART interface.

3) ***Soft Reset****:* By TC (CSP command) to the EPS unit originating from ground or from another subsystem.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

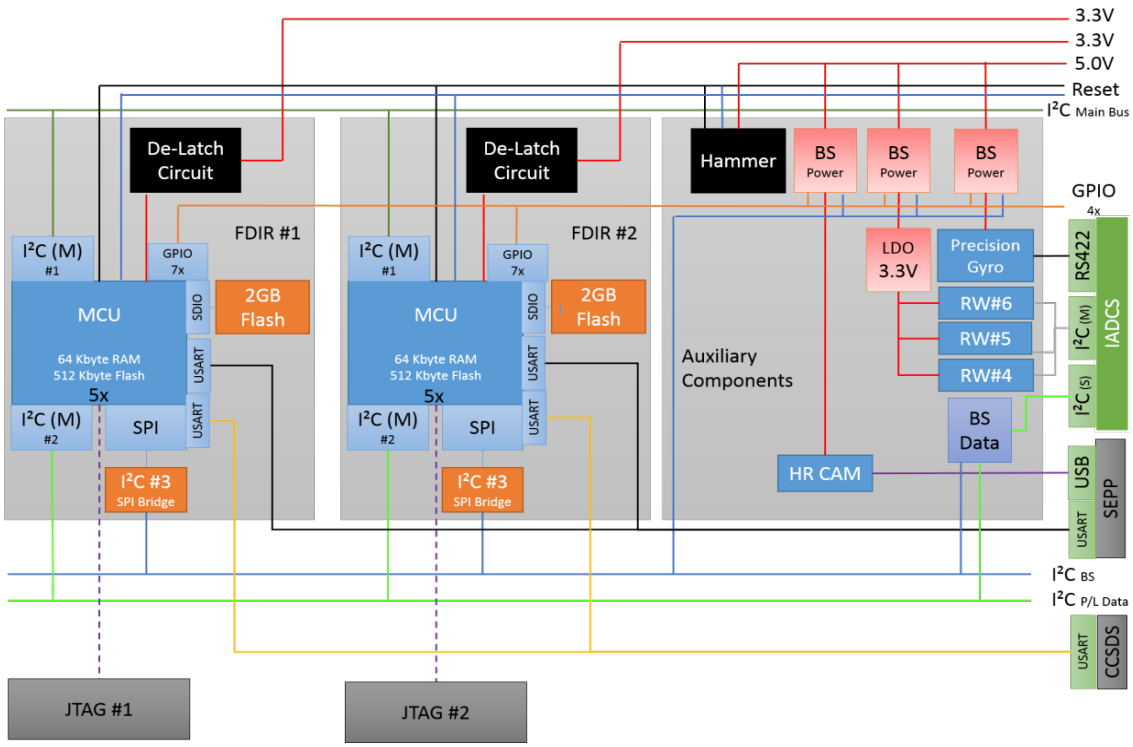

Figure 4. FDIR Hardware Architecture

B. Software Architecture.

With regard to the FDIR system on-board software, BST has made the decision to not use an operating system but a firmware oriented design. An OS based system is not required nor would it be advisable due to the large overheads it would require and the low processing resources that are available for the FDIR MCU due to the power constraints. Since the FDIR system works as the main master and communicator between all the payload subsystems, it is important for the FDIR system to be active all the time and with a better performance and throughput. The functional essence is that once the software is started and initiated, it is in a constant loop updating the state machine and scheduler. This loop continues until there is a hammer reset or is shut down by a failure or power shut down. The FDIR system architecture has three main layers of software:

1) ***The High Level Software Application (HLSA):*** High level software application mainly contains applications to handle outside/ground communication providing access to the payload subsystems, handling the state of the complete system in case of failure and re-initiating the system to start afresh after hammer reset.

2) ***The Management and Service Layer (MSL):*** Management and service layer is responsible for handling all the modules that need servicing or regular monitoring. It keeps a check on all the communications that are in progress and gives a priority to the external commands and initiates them. It is also responsible for collecting housekeeping data and most of it is initially stored in the RAM and then moved to the secondary storage device for future use and thus provides memory management too. It keeps a regular time count and is always aware of the next necessary tasks to be serviced and follows the procedures to handle these time initiated tasks and processes such as regular monitoring of the status of the subsystems, reading telemetry data, etc.

3) ***The Low Level Layer (LLL):*** Low level layer mainly consists of the boot-loader and device drivers that are the basic building blocks of the complete software architecture. As the system is switched on by the EPS module with the 3.3V power supply, the boot-loader kicks in, initiating all the device drivers until the complete software is built. The built in redundancy in the hardware with 2 MCUs is also handled by this to choose which MCU is active at a given time and automatically chooses the new MCU if the system was previously shutdown due to failure or system fault.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

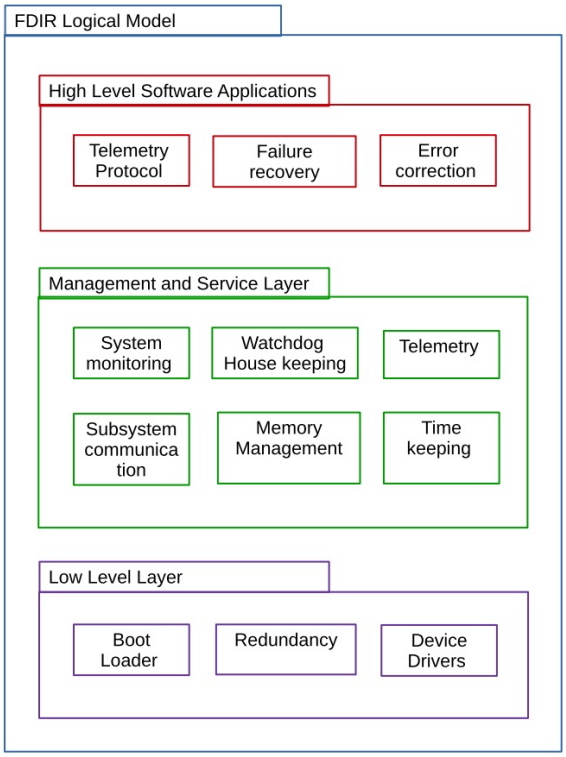

Figure 5. FDIR Software Logical Model

C. Bus Switches.

A bus switch consists of two parts, a power switch and a data interface switch. Both parts can be used independently of each other. The data bus switches safely separate data interfaces to allow bus usage even if a device generates a physical failure on one of its interfaces. The power switch safely separates payload module power lines by isolated switching, providing suitable protection from latching and protects the systems in case of overcurrent, over-voltage or under-voltage events.

The data bus switches are designed to monitor and control the connections between the payload interfaces and the external satellite bus. Furthermore, they provide an independent health measurement of payloads and consist of several components such as an intelligent power monitor and a controller with latch-up, overcurrent, overvoltage and under-voltage protection.

The bus switches allow the payloads to communicate their power and status data to the FDIR OBC and to receive commands on an independent dedicated I2C bus. As an example, see the Data Bus Switches of SEPP in Fig. 6. The FDIR computer is connected to the I2C Bus Expander which controls the individual data switches.

Different interfaces can be equipped with data bus switches to be able to selectively connect and disconnect the data signals from the satellite bus. All data switches can be controlled independently. Each bus switch is different due to the fact that the number of interfaces differs between the units. The bus switches have to be monitored and configured after payload device start-up by the FDIR computer to change the bus switch state if necessary and to detect system failures.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

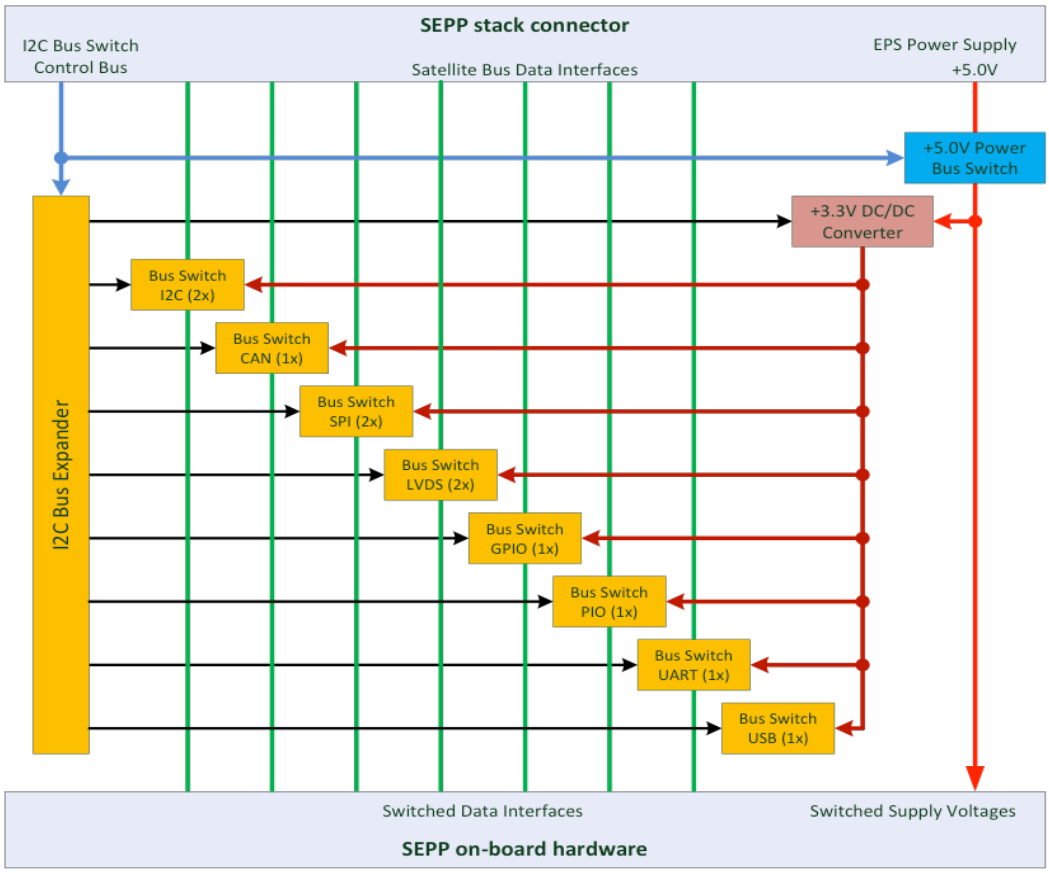

Figure 6. SEPP Bus switch lay-out.

On OPS-SAT all payload modules are supplied by a single EPS output channel. Each of these EPS channels contains its own protection circuitry providing TM used by the FDIR for status monitoring and failure detection. The EPS switches for the payload devices are controlled by the FDIR OBC as well to enable the hardware in accordance to the needs of the experiments. The concept of bus switches enables the implementation of a redundant SEPP PCB stack with two or more identical SEPP modules in so-called cold redundancy where all boards share a common EPS power supply. Hence, to activate a single SEPP board a kind of switch is required to establish the connection to the power supply while all other spare SEPP modules are deactivated. This switch and powerful monitoring and control features are implemented in the SEPP power bus switch. Because of the limited number of EPS channels, the power bus switches are a very important concept to provide the necessary flexibility required by the OPS-SAT experiments.

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

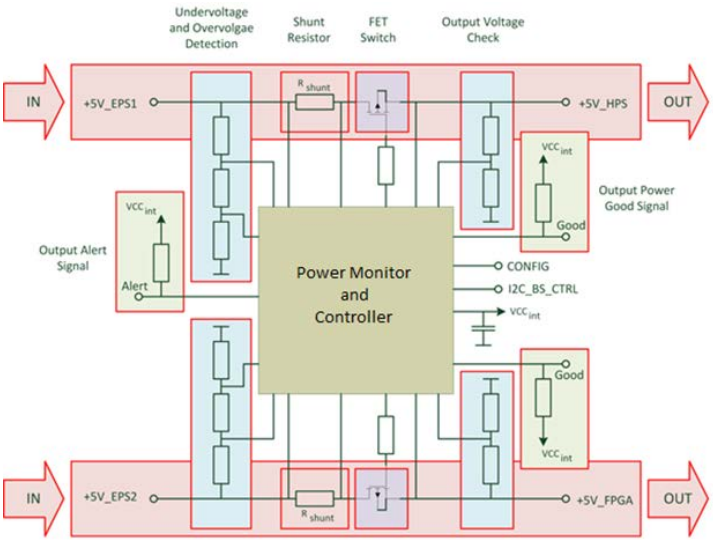

Figure 7. OPS-SAT Power Bus Switch used for SEPP

The power bus switch is used by the FDIR OBC to retrieve information about the consumed current and the level of the voltage. It should be emphasized that a power bus switch is much more than only a simple switch. A power monitor and controller IC is used to realize the power bus switch providing several features for signal monitoring and failure detection. Hence, the switch itself is only one aspect of the power bus switch. As an example, the power bus switch used for SEPP is shown in Fig. 7.

Irrespective of the power supply type both channels can be activated by the FDIR OBC at the same time or in a custom sequence by sending write commands to the I2C bus switch control bus. The power monitor and controller IC features adjustable analogue foldback current limit and a dedicated pin which be used to configure the time spent in overcurrent before declaring a fault. Depending on the used device, the part to latch off may be configured or automatically restarted after the controller detects a current limit fault. The power monitor and control IC can trigger digital notification/fault signals when a fault was detected. Optionally, the controller may notify when output power is good, and power-up either automatically or wait for an action to turn on the FET switch. All internal status information can be read by the FDIR OBC via I2C bus switch control bus. The status of the digital alert and fault signals are represented in the internal registers as well. The electrical switch itself is realized with an external field effect transistor (FET) which is controlled by internal power monitor and controller logic.

VII. FDIR Operational Modes

The operational modes of the FDIR system have been divided into six different functional states, as described below:

1) ***Idle (initial state):*** After the boot loader loads all the device drivers and sets up the high level software, the system is in an *Idle* state. In this state, the system is running a time keeping counter which schedules regular intervals of *Safety Check and Scheduled Queue Processing Check.* The system jumps periodically to one of these states to perform the scheduled processes. The state can also change when there is an external command or a master command is received. Then the *Idle* state jumps to the *Process Master Command* state to take care of the task. After completion of the process, it returns back to *Idle* state and waits for the next master command or for a scheduled run.
2) ***Process Master Command****:* When a master command can be self-processed within the FDIR system, it is handled by the system software. If the master command is to handle a client, the command will be forwarded to a client subsystem. The FDIR system then jumps to the *Send Command to Client Pending* state. Several client commands can be invoked by the FDIR system main system in parallel. After the client subsystem completes, the system returns a confirmation message acknowledging successful completion.
3) ***Safety Check****:* In this state the system checks for faults and a queue of sub-states is processed. Each sub-state invokes a self-check read and registers the collected status data. The system jumps to the *Send*

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481

Commands to Client Pending state until the client responds. This system may also request telemetry data from the client and then the system jumps to the *Wait for Client Reply* state until this data is returned. After completion, the FDIR system returns to the *Safety Check* state and a message is sent to confirm the successful completion of the whole process.

4) ***Scheduled Queue Processing****:* In this state the system starts a queue of processes to maintain the FDIR system itself in a fault free working state. It checks for several features such as temperature check, memory over flow check, maintaining the secondary storage for future data retrieval when requested by the master command, etc. These processes may also initiate sending sequence of commands to the client and collecting data. After successful completion, a confirmation is sent back.
5) ***Send Command to Client Pending****:* When a request or command is sent to a client to perform a task, the system goes into this state. The system needs to queue commands to the client if it is busy before initiating the next command.
6) ***Wait for Client Reply****: It* takes a certain amount of time for a client to initiate and then respond to the requests and during that time the system is in the wait state as described. This wait may be caused due to performing a sequence of tasks independently by the client for example.

VIII. Conclusion

Designing a space mission in which you know that there will be lots of critical software loaded on-board that has many bugs is not easy. Add to this budget constraints that mean you cannot rely on the usual amount of pre-loading tests (in fact you can practically do very little) then the problem borders close on impossible. However on OPS-SAT, we did have one advantage - we knew this challenge right from the start and so could prepare for it. Thanks to a very productive collaboration between the ground, space and experiment segments of this mission we have been able to put together an FDIR concept at system, software and hardware level that we believe is up to the job. Having the idea in the back of your head that the on-board software is full of nasty bugs that want to end to mission is not an easy design rule to live with, but it certainly makes you better system engineers.

Acknowledgments

The authors wish to acknowledge the provider of the FDIR system, Berlin Space Technologies, and the prime TU Graz, for their contribution to the safety of this mission.

References

[1]D. Evans, Alexander Lange, OPS-SAT: Operational Concept for ESA'S First Mission Dedicated to Operational Technology, Spaceops 2016

Downloaded by ESA-ESTEC on May 30, 2016 | http://arc.aiaa.org | DOI: 10.2514/6.2016-2481