Spectral Ocean Color (SPOC): Lessons Learned from the University of Georgia Small Satellite Research Laboratory's First Satellite

Sydney Whilden, Deepak Mishra
University of Georgia Small Satellite Research Laboratory
220 Cedar St Rm 107, Athens, GA 30602; 706-542-2856
sydney.whilden25@uga.edu, dmishra@uga.edu

David L. Cotten, Nicholas Neel
Oak Ridge National Laboratory
1 Bethel Valley Road, Oak Ridge, TN 37830; 865-576-7658
cottendl@ornl.gov, neelnc@ornl.gov

ABSTRACT

In October 2020, the University of Georgia Small Satellite Research Laboratory launched its first CubeSat, a 3U Earth-observation mission designed to collect multispectral data from Georgia's coastal environments for UGA's Center for Geospatial Research to make recommendations on environmental conservation, care, and use. SPOC successfully detumbled, but after approximately a month in orbit, a coronal mass ejection (we speculate) caused us to lose contact. Despite our disappointment at the loss of SPOC, we are leveraging the lessons learned for our upcoming missions. These lessons can be categorized in four principal areas: software (flight and payload), mission operations, testing, and educational program structure. Specifically, we learned how to carefully design mission controls, how to plan and execute robust batteries of tests, and how to work together to reach our potential as young scientists and engineers. We will show how we implement these lessons on our upcoming missions – the Multi-view Onboard Computational Imager (MOCI), a 6U mission using on-orbit Structure-from-Motion to create 3D terrain maps; and the Mission for Education and Multi-media Engagement Satellite (MEMESat-1), a 2U non-profit-sponsored outreach mission designed to introduce undergraduates to building satellites and K-12 students to the world of satellite and and radio communications. We aim to share what we have learned with other young CubeSat development programs to help them pioneer new space system technology, gain scientific insight from payload data, build strong university space programs, and enrich their surrounding communities.

INTRODUCTION

Program Structure

The University of Georgia Small Satellite Research Laboratory (SSRL) is overseen largely by undergraduate students. In the case of SPOC, faculty oversight provided by Principal Investigator Dr. David L. Cotten was crucial, but Dr. Cotten's strategy was to provide support for undergraduate efforts rather than to head the mission himself. The result was that SPOC's development was motivated by a combination of scientific need and student enthusiasm, rather than solely by faculty research interests.

The SSRL is officially part of the UGA Department of Geography, but is housed in the same building as the Department of Physics and Astronomy. Both departments are a part of Franklin College of Arts and Sciences, as opposed to the College of Engineering. Accordingly, while a large number of undergraduate lab members are students in various disciplines of engineering, the SSRL's research interests are a healthy mix of engineering- and space technology-related pursuits and geographical questions whose answers have payoff on the ground.

Over 250 students have been members of the SSRL. They come from a variety of majors, including Mechanical, Electrical, and Computer Systems Engineering, Computer Science, Physics, and even Biology and Natural Resources Management.

Mission Overview

SPOC was funded by the NASA Undergraduate Student Instrument Project (USIP). It was expected to launch as part of NASA's eight CubeSat Launch Initiative class. It was initially scheduled to launch in 2018.

SPOC's goal was to acquire moderate-resolution spectral imagery of coastal Georgia environments such as Sapelo island, as well as ocean color. The idea was to measure coastal wetlands status, estuarine water quality, and near-coastal ocean productivity. Of particular interest were suspended sediments and organic matter including algal blooms. Data was to be acquired across a wide range of spectral bands, between 433 and 866 nm.[1]

The payload is SPOCeye, an adjustable multi-spectral imager designed by Cloudland Instruments. The design for SPOCeye is based off of a previous Cloudland design called HawkEye, which was launched on board the SeaHawk mission from UNC Wilmington. Its main sensor is a complementary metal oxide sensor (CMOS) array.[1]

The stack features an OBC from Clyde Space (since incorporated into AAC Microtec).

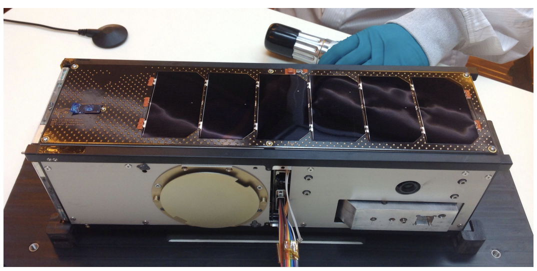

Figure 1: SPOC after integration.

Launch

On October 2, 2020, SPOC launched on a Cygnus NG-14 commercial re-supply mission to the International Space Station. It deployed from a Nanoracks deployer on November 5 of that year. As of November 6, the satellite was trackable by COSMO, the UGA ground station.[2] The first telemetry beacon was successfully received by that time by SatNOGS ground stations.

However, around December 9, temperature on the battery board and the CMC began to spike. On December 14, the last telemetry frame was received from SPOC by SatNOGS ground station EU1XX. SPOC's last known state was Safe Reboot Mode on the Failsafe software image.

Our best interpretation of these results is that a solar flare taking place on December 14 (the tenth largest of 2020) caused a single-event upset (SEU) or single-event latch-up (SEL), damaging hardware components on the electrical power system (EPS).[3]

LESSONS LEARNED

General Principles

1. Keep it as simple as you can. You have to walk before you can run.

The overarching lesson we took from the mission of SPOC, both with respect to the satellite and the ground station, is that whenever a process may be simplified, one generally ought to simplify it. SPOC required pointing accuracy, a large data budget, two antennas, detailed on-orbit scheduling, and particular power needs.

As an example, individual selections had to be made among available COTS parts for different subsystem components. Buying a 3U CubeSat bus made by a single manufacturer would have cut down on the complication associated with integrating the products of multiple manufacturers into a single bus. Parts would be guaranteed to work together.

Likewise, for the ground station, by working with a larger and more complicated suite of equipment designed to communicate using both UHF and S-band, we passed up the chance to use a simpler UHF station that was easier to develop. So much time was put into the bigger system that some basic functionality went untested.

One way we found success in adhering to this mantra of simplicity was by using the software designed for SPOC's hardware. Even while we were integrating various wildly different hardware components together, much complication was avoided by simply following manufacturer recommendation.

Radiation Mitigation

1. Prepare for the inevitable.
2. Choose your hardware carefully.

The unofficial motto of the SSRL is "Space is hard." This is because sometimes, even when the best precautions have been taken, the natural environment intervenes.

It is not typically possible to put heavy radiation shielding on a CubeSat. This is because most CubeSats use low-cost, standardized commercial off-the-shelf (COTS) parts. Like all small satellites in low Earth orbit, SPOC was subject to levels of ionizing radiation from the South Atlantic Anomaly that posed significant risk to mission objectives. The risk primarily originated from the possibility of single-event effects (SEEs), which are not necessarily mission-fatal, and single-event latch-ups, which usually are.

As such, for radiation mitigation, SPOC used software-level methods. It had an alternate Failsafe software image, designed to be booted into only upon two consecutive unscheduled, unexpected reboots. Failsafe contained only the Safe Reboot Mode, a mode which powered only the onboard computer (OBC), electrical power system (EPS), and CMC. A manual command from the ground station would have been required to boot back into the primary software image with full functionality.[2]

There also existed a simple Safe Mode, which could be triggered by a manual command owing to hardware degradation due to ionizing radiation. Safe Mode was different from Safe Reboot in that Safe Mode existed in the primary software image and was entered via a command from the ground station rather than an automatic transition triggered by consecutive reboots.

The truth is that these measures did not represent a robust counter-attack to the dangers of the low Earth orbit radiation environment. Other missions have used co-processors to back up primary hardware and error-correcting codes (ECCs) to counteract bit flips induced by SEEs. What we ultimately learned is that a certain degree of radiation-related risk is inevitable in low Earth orbit, but if one has the funds available, one can invest in redundancy measures that provide some peace of mind.

Mission Control Design

1. Separate the team designing the mission controls from the team intended to operate the mission using those controls.

2. Prioritize basic functionality over an attractive front end.

It seems natural to have the team of people who will operate the mission be the team that designs the mission operations. However, the processes of designing mission operations and actually operating the mission are very different, and in the end, it behooves one to have different teams on each task.

The reason for this is one of time preservation. The person who spends an inordinate amount of time designing the mission control system for the satellite may assume that those to follow will understand the design of the system. However, the proper flow of the design should be that the mission operators specify their mission control needs and the ground control specialists design the software according to those needs. The two roles should not be conflated. The reason why is that when the mission operators, who may not be from a software background, are relied upon to develop the ground station software, their focus is diverted from their actual purpose, which is to operate the mission. The proper flow of mission requirements is from the mission operations team to the ground station software development team, rather than the reverse.

Testing

1. Do not let fear of contamination deter you from testing ground and flight systems.

2. For software testing, employ test-driven development and continuous integration.

3. For hardware testing, test on the flight model, not just the engineering model.

4. Use the testing software that is recommended for your hardware.

The overarching lesson learned was that the key to a successful mission is the testing done on the ground. The truth, however, is that rigorous prelaunch testing can be challenging to carry out. Reasons for this may include the necessary frequency and physical effort of testing (some tests run for as long as 24 straight hours with constant monitoring); the expense of testing equipment, including engineering development units (EDUs); the logistical difficulty of transporting the satellite and the equipment required to test it (some testing had to be performed at neighboring universities with more suitable testing equipment, e.g. a vibration table); or fear of damage to the satellite itself when it is transported from place to place to perform certain tests.

One specific example is long-range ground station testing. Long-range communications testing necessitates that satellite-to-ground station communications be tested over a range of at least one mile. This necessitates that either an EDU ground station be carried outside of the laboratory, or that in the EDU's absence the satellite itself must be taken outside. Access to an EDU ground station may be lacking due to such reasons as a lack of designated funding, a common problem for early-stage university CubeSat laboratories. However, as we learned, sometimes difficult decisions must be made. Taking the satellite outside the laboratory poses a risk to its physical integrity, but to avoid long-range testing for this reason is taking a gamble with mission success. In the end, we did not remove the satellite from the lab for long-range communications testing for fear of damaging it; thus, at the time of launch, the ability for the satellite to transmit to or receive from the ground station over long distances was not thoroughly verified.

Another complicating factor in testing is the fear of failure. Students will naturally want their systems to pass the tests to which they subject them; it can be daunting to begin a test of hardware or software that one knows the system will fail. However, failing early tests is an important part of learning how to pass them. Tests should be viewed as diagnostic tools to identify shortcomings.

As it pertains to software, a way to employ this outlook is by using test-driven development. In this process, both unit tests and integration tests are performed on the code. If possible, it is beneficial to use continuous integration – that is to say, these unit and integration tests should be automated and performed frequently. Software versions should be regularly frozen and tested.

The same principle applies in a way to hardware. Instead of testing only on isolated subsystems, the whole system should be tested. In addition, testing should be performed on the flight model, not just the engineering model. Students might be afraid of damaging the flight hardware, which can be difficult or impossible to replace before launch, especially as the phase of most intense testing is reached late in mission development. However, even if subsystems function effectively together on the flatsat, that is no guarantee they will function equally well on the flight model. Thus, the entire system should be integrated and tested in the form it will take in orbit.

CONCLUSION

Testing is crucial to the success of any satellite mission. The students in the SSRL by and large put forth their best effort to make SPOC a reality. That the students who built SPOC were almost exclusively undergraduates should not go unnoticed; they learned above and the beyond the call of curriculum to make SPOC function. Every lesson the students learned will benefit the development of MOCI and MEMESat-1, the SSRL missions currently in development. It is the hope of the authors that other university CubeSat labs can use these lessons to build CubeSats that fulfill and exceed mission success criteria.

ACKNOWLEDGEMENTS

Thanks to Alex Lin and Caleb Adams for providing their crucial perspectives on lessons learned from the SPOC mission. The authors are grateful also to the faculty who have supported the SSRL's development into the organization it is today. Most of all, we thank the dozens of students who worked hard to put SPOC in space.

References

[1] Nirav Cotten, David L.; Ilango. Spoc mission overview. Internal document., 2017.

[2] P.; Grable G.; King A. Leicher, B.; Copenhaver. Concept of operations: A guide to spoc functionality. 2020.

[3] Viewing archive of monday, december 14, 2020, June 2022.