

SpaceOps-2025, ID # 602

The Module for Event Driven Operations on Spacecraft (MEDOS) – Initial Flight Results

Alexander C. Barrie^{a*}, Miles Bengtson^a, Paul Wood^b, Russell Bjella^c, Liam Greenlee^a, Ryan Kinney^a, Joseph Patton^a, Brandon Stone^c, Marcus Piquette^c, Robert Klar^b

^aAurora Engineering, 4 Redbud Ct., Potomac, 20854, MD, USA

^bSouthwest Research Institute, 6220 Culebra Rd., San Antonio, 78238, TX, USA

^cUniversity of Colorado, Laboratory for Atmospheric and Space Physics, 1234 Innovation Dr., Boulder, 80303, CO, USA

Abstract

The Module for Event Driven Operations on Spacecraft (MEDOS) calculates onboard derived data products that are physically meaningful to mission planners. These derived products are then evaluated against known event definitions using fuzzy sets and, when combined together, provide a likelihood that a given event is occurring. MEDOS has now been verified both on the ground, via flatsat testing, and in space on the MMS2 spacecraft. The in-flight test of MEDOS showcased detection of the radiation belts using in-situ particle measurements, illustrating a more accurate entry and exit trigger than a predictive model of the radiation belt location.

Keywords: MEDOS, autonomy, operations, artificial intelligence, fuzzy logic, MMS

Acronyms/Abbreviations

- Module for Event Driven Operations on Spacecraft (MEDOS)
- NASA core Flight System (cFS)
- Technical Readiness Level (TRL)
- Electronic Data Sheets (EDS)
- Fast Plasma Investigation (FPI)
- Fly's Eye Energetic Particle Sensor (FEEPS)
- Magnetospheric Multiscale mission (MMS)
- Central Instrument Data Processor (CIDP)
- Robot Operating System (ROS)
- Application Program Interface (API)

1. Introduction

Trusted spacecraft autonomy has long been a goal of space flight operations. MEDOS is a NASA cFS[1] (core Flight Software) compliant TRL-7 rated flight software application designed to encode subject matter expert analysis and decision making on a spacecraft. MEDOS consists of two primary modules:

1. A **Derived Telemetry Feature Extractor** which takes raw measurements and converts them into quantities that are physically meaningful to the ground team. For example, a set of count rates, and field measurements may be converted into *plasma density and temperature*. Or, a rate of change of resistance and a frequency may be combined into *methane concentration*. These derived features are intended to be quantities that a subject matter expert works with routinely and can analyze in a meaningful way.
2. **Event Detection** is performed using fuzzy sets to define the likely boundaries of a given event per derived feature. These fuzzy sets are then combined to arrive at a final likelihood that a given event is occurring. For example, a spacecraft crossing the magnetopause can be characterized by fuzzy sets describing changes in the local plasma environment along with the local magnetic field.

The final output of the MEDOS algorithm is a probability that an event has occurred – not a strict yes/no. In this way, it is more forgiving towards uncertainty in the inputs as well as in the fuzzy set definitions. If a sensor is miscalibrated, for example, the confidence of the event trigger may reduce slightly, but the overall picture will remain the same. In this way, MEDOS is much more resilient than a strict rule-based decision tree. At the same time, because the derived telemetry points and the fuzzy sets are described by subject matter experts, heritage, and published knowledge, MEDOS does not rely on in-flight training, or ‘black box’ machine learning approaches. It provides trusted autonomous operational responses to events that are characterized by experts in the field – an important step on the road towards true trusted autonomous operations.

A complete description of the underlying algorithms is available in [Barrie et al., 2025, submitted for review to the Journal of Spacecraft and Rockets\[2\]](#).

In this work, we present results of a ground validation and flight verification of the MEDOS algorithms, highlighting the operational flight readiness of the framework.

2. Flight Ready Implementation of MEDOS

The MEDOS system is divided into two stages, a core library, and a platform-specific app. The core library is written in standard C and contains the logic to process event detections and an interface to receive and process telemetry data. Different platforms have different mechanisms for sending and receiving data, hence the need for a platform-specific app. To process telemetry, the core library uses user-defined functions, which the application is responsible for providing.

A NASA Core Flight System (cFS) application for MEDOS was developed as the standard interface layer for future NASA missions. This app uses the cFS functionality for subscribing to telemetry sources and publishing scores. The table services are also used to configure the telemetry and event definitions and for updating them without resetting. The MEDOS app additionally provides the user-defined functions to the library through a dynamically loaded library, which can be updated at runtime. There are downsides to this approach, mainly safety, since these functions are not run in a sandbox environment. Another downside involves processing data received from a system with different alignment or endianness requirements than the system running MEDOS. The telemetry functions must be aware of this and correctly deserialize the data, which increases their complexity. In addition to cFS, a pure C interface, a python interface, and a ROS (Robot Operating System) interface also exist and can be swapped in or out as desired.

MEDOS also includes support for the Electronic Data Sheets (EDS) standard. It provides a platform-agnostic interface for sending and receiving data, which abstracts away the endianness and alignment requirements from application developers. This removes the need for the MEDOS telemetry functions to deserialize data. Additionally, the EDS system contains bindings for Lua language. Lua is a lightweight scripting language that contains a simple C Application Program Interface (API). Most importantly, Lua scripts can be run in a safe and secure sandbox environment. Through the EDS-Lua bindings, MEDOS can use Lua scripts for telemetry processing functions instead of C code. This provides the end user with a way to update operational triggers, event definitions, etc. in a way that is easier to implement and also firewalled from the main flight software. While EDS and Lua scripting are available for the MEDOS interface layer, they are not required.

3. Verification Test Case Setup – Radiation Belt Entry

Many instruments choose to disable high voltage operations while transiting Earth’s radiation belts due to the potential for damage to sensitive electronics. What is not always clear, however, is how the radiation belt should be defined when planning for this safing procedure. Numerous models exist that describe the radiation belts, however these models often result in significantly divergent levels of high energy particle flux for a given L shell (location within the Earth’s magnetic field.[3])

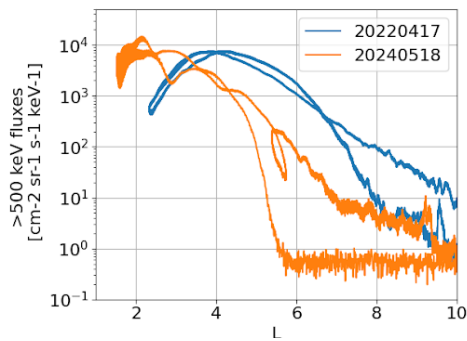


Figure 1. High energy particle flux vs L-shell for two orbital passes of the MMS2 spacecraft.

Figure 1 shows the high energy particle flux (> 500 keV) for the MMS2 spacecraft on two different orbits as a function of L-shell. Each orbit has both the inbound and outbound leg of the orbit, hence why there appear to be two lines of each color. While both traces show an increase in counts at lower L, the place at which the particle counts cross a given threshold is significantly different. For example, the Fast Plasma Investigation (FPI)[4] on the MMS[5] spacecraft shuts off high voltage at L-6. Depending on the day, this could mean an incident high energy flux ranging from 0 up to 1,000 – three order of magnitude difference for the same cutoff of L-6.

Rather than rely on a fixed model such as L-shell, MEDOS demonstrates responsiveness to the actual radiation belt by monitoring this high energy flux directly.

A verification test of MEDOS was set up using radiation belt entry and exit for an event trigger. First, Figure 2 shows two fuzzy set definitions for MEDOS to trigger radiation belt. The high energy count rate was measured using the Fly’s Eye Energetic Particle Sensor (FEEPS)[6] on the MMS spacecraft.

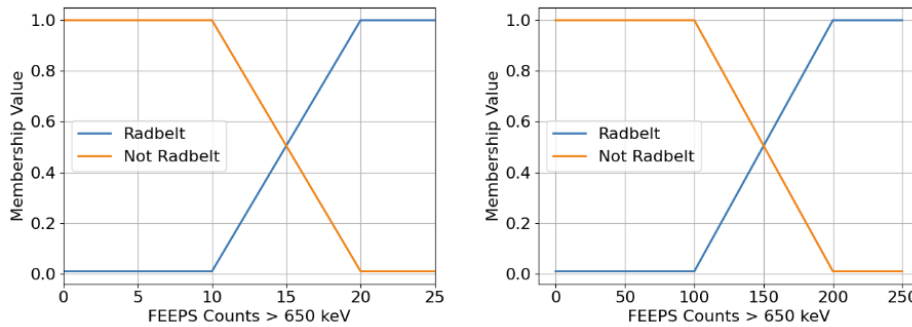


Figure 2. Fuzzy set definitions for a conservative (left) and risk tolerant (right) radiation belt detection.

The left panel shows fuzzy set definitions for a conservative risk posture and the right panel shows settings for a more risk tolerant posture. Note that the count rate required to trigger for the risk tolerant approach is ten times the level (150 vs 15) of the conservative approach. The FEEPS counts would then be measured continually while MEDOS evaluates the trigger for entry into the radiation belt.

3. Ground Test Results

Ground verification was performed at the Laboratory for Atmospheric and Space Physics (LASP) at the University of Colorado. The flatsat used for testing operational flight sequences was utilized, with a telemetry buffer that can pull data from the MMS science data center and feed it directly to MEDOS as if it were a real time telemetry stream. Figure 3 shows the results of testing on the MMS Flatsat test environment housed at CU/LASP. The MMS Flatsat houses a flight-spec Central Instrument Data Processor (CIDP) which was used to run the MEDOS software. Data was fed into the CIDP via the telemetry interface for the FEEPS instrument through an independent telemetry feeding program. For this test the FEEPS flight data from MMS2 on the March 10, 2024 perigee pass was used as input.

The top panel shows the raw count data at the highest energy level (> 500 keV). The center panel shows the MEDOS results with a “conservative” risk posture. In this case, the red “radiation belt” exclusion event is much larger than the “risk tolerant” approach shown in the bottom panel. This showcases the ability of the MEDOS algorithm to be tuned to different operational concepts. The conclusion of this ground test campaign on the MMS flatsat raised MEDOS to the NASA TRL-6 level, allowing it to be deployed in flight.

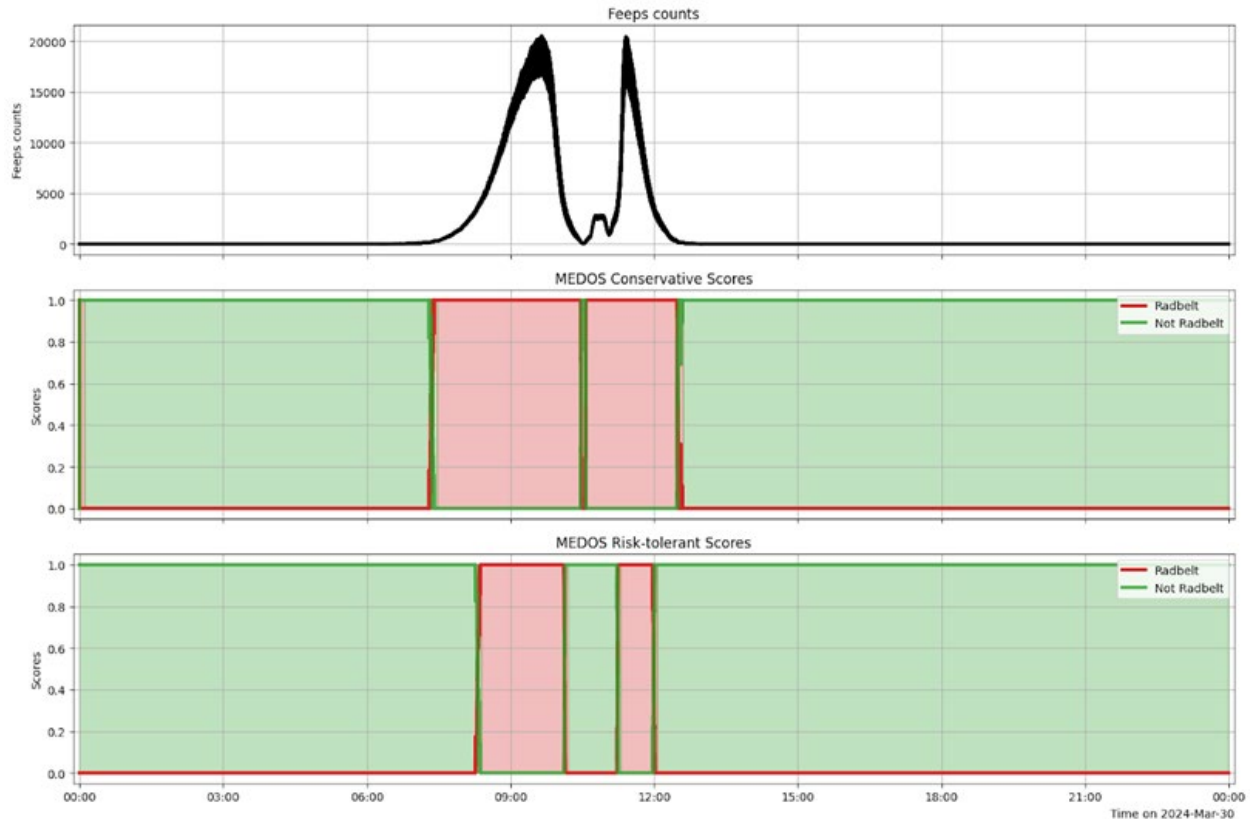


Figure 3. Ground test results of radiation belt crossing for conservative (top) and risk tolerant (bottom) flatsat tests.

4. Flight Verification Results

A flight verification of the MEDOS framework was conducted in March of 2025, bringing it to TRL-7. Three successive orbits were run: two with the conservative risk posture, and a third with the risk tolerant posture. In each case, the FEEPS raw telemetry was fed into the MEDOS system, which analyzed the packets against the defined radiation belt event trigger. When the radiation belt was detected, an MMS action point would enable and send a notification of the detected event. A corresponding message was sent on detection of the exit from the radiation belt.

Figure 4 shows the results of the three orbits. In all three cases, MEDOS successfully detected the peak of the radiation belts, when the highest energy counts were at their peak. The detection events are numbered for reference. The risk tolerant orbit does not detect the lesser of the two belt passes and only triggers on the more intense pass. This illustrates the operational flexibility of MEDOS – given the low intensity of the high energy particles in that area of the radiation belt, a spacecraft could likely operate safely through the first part of the radiation belt. This flexibility would not be possible with a pure “Turn off after L-6” type of rule.

To illustrate why the two areas of the radiation belt appears asymmetric for these three orbits, Figure 5 shows the orbit of the MMS2 spacecraft during the second pass of the verification test and the magnetic field lines associated with L shell. The event numbering matches the time order numbering of Figure 4. For the first MEDOS event trigger (upper red band), the spacecraft is just grazing the radiation belt, whereas for the second event trigger (lower red band) the spacecraft is passing through the center of the radiation belt. It can be difficult to predict the exact shape of the radiation system, particularly in three dimensional space with an inclined orbit and the dynamic nature of the Sun-Earth system.

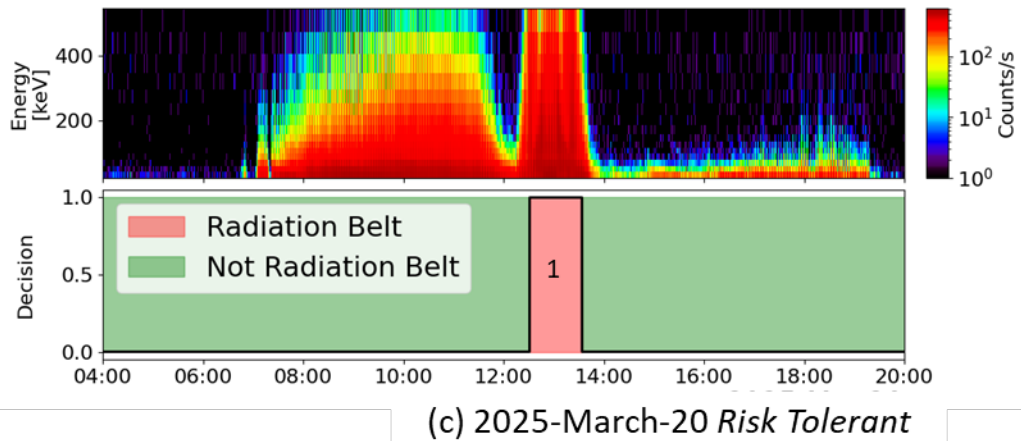
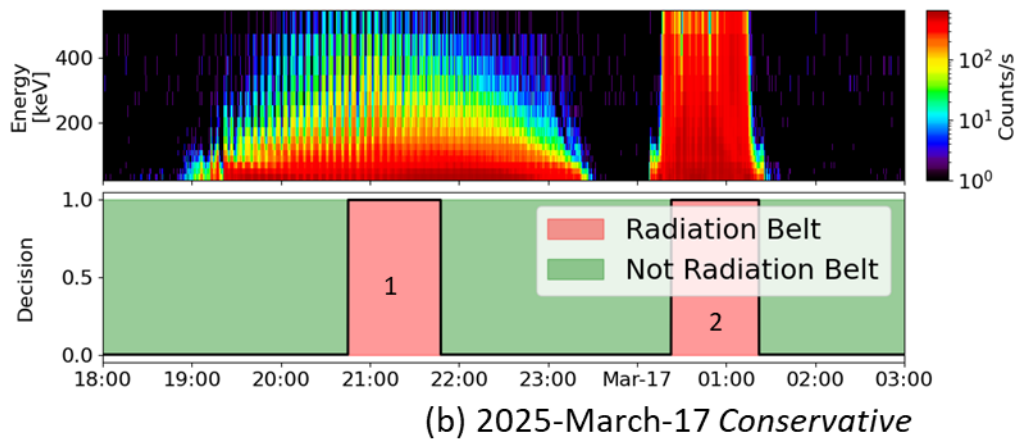
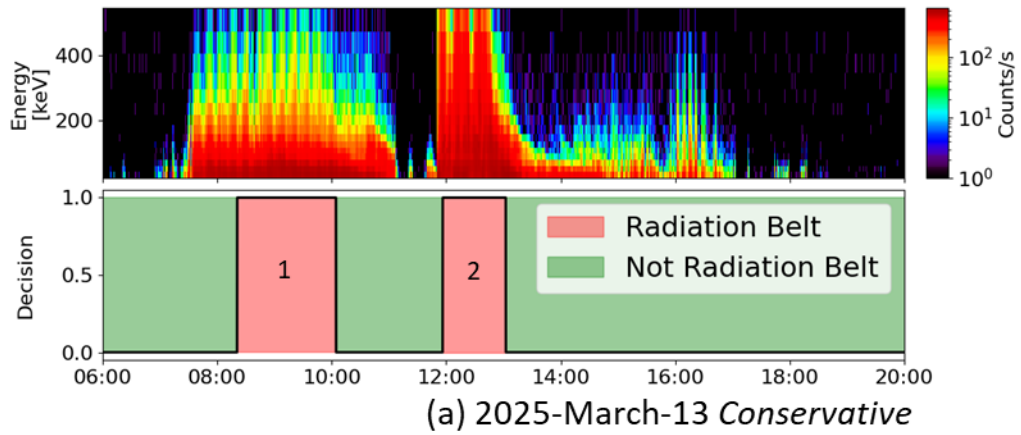


Figure 4. Three passes through perigee of the MMS2 spacecraft for MEDOS verification campaign.

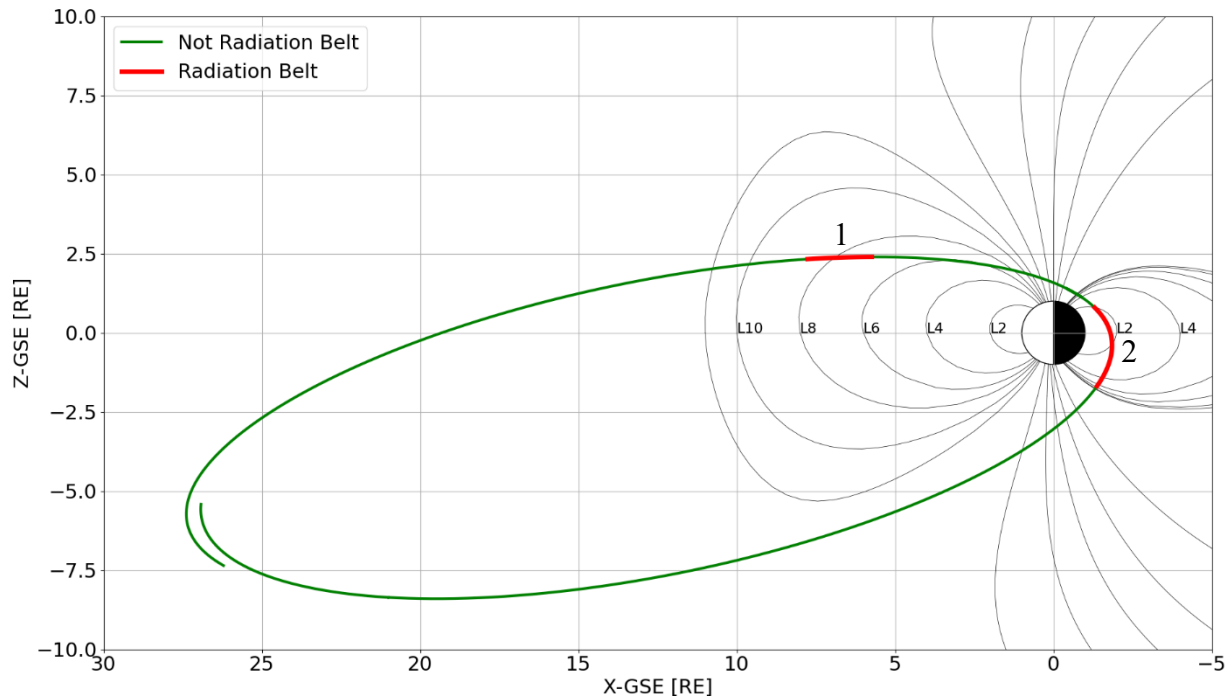


Figure 5. Orbit of MMS2 including the March 17, 2025 MEDOS events.

5. Discussion

Validation of the MEDOS algorithms was previously performed over several months of sample data to detect a variety of key mission events[2]. The verification of the MEDOS algorithms provide confidence that they will perform as intended in an operational space environment. In particular, the in-flight demonstration emphasizes the utility of event driven operations. Space is a chaotic and uncertain environment and it is not possible to accurately to predict the location of every event of interest in a space flight mission.

Rather than trying to predict and pre-plan operational sequences ahead of time, MEDOS allows for the in-situ real time detection of events as they occur. This is a key asset for transient events that may be missed if not timed perfectly, such as subtle science events that are few and far between. Similarly, any event that affects health and safety, such as the radiation belt entry presented here, can benefit from real time detection as the exact timing of these events can be highly variable (as demonstrated in [Figure 1](#)).

6. Conclusions

Presented here are the verification tests of the MEDOS algorithms, both on the ground via flatsat, and in flight on the MMS2 spacecraft. The tests successfully demonstrated functionality of autonomous event detection in a flagship NASA mission. Because MEDOS has interface definitions for cFS, ROS[7], python, and C (with an F'[8] implementation under consideration,) it represents an off the shelf toolkit capable if reliable, trustworthy, autonomous event detection and response based on subject matter expertise and heritage knowledge.

MEDOS does not require on-orbit training, and is resilient against uncertainty in event definitions, environmental percepts, or in the detection and measurement system. It represents a key step forward on the path toward trusted autonomous operations.

Acknowledgements

The authors would like to individually thank Wayne Yu and Conrad Schiff of NASA's Goddard Space Flight Center, as well as the MMS mission team, Southwest Research Institute, and the Laboratory for Atmospheric and Space Physics at the University of Colorado for the support to bring this flight test to fruition.

References

- [1] McComas, D., "NASA/GSFC's Flight Software Core Flight System," *Flight Software Workshop Flight Workshop*, 2012.
- [2] Barrie, A., et al., "Module for Event Driven Operations on Spacecraft," [Under Review], 2025.
- [3] Daly, E. J., et al. "Problems with models of the radiation belts." *IEEE Transactions on Nuclear Science* 43.2 (1996): 403-415.
- [4] Pollock, C., et al. "Fast plasma investigation for magnetospheric multiscale." *Space Science Reviews* 199 (2016): 331-406.
- [5] Burch, J. L., et al. "Magnetospheric multiscale overview and science objectives." *Space Science Reviews* 199 (2016): 5-21.
- [6] Blake, J. B., et al. "The fly's eye energetic particle spectrometer (FEEPS) sensors for the magnetospheric multiscale (MMS) mission." *Space Science Reviews* 199 (2016): 309-329.
- [7] Quigley, Morgan, et al. "ROS: an open-source Robot Operating System." *ICRA workshop on open source software*. Vol. 3. No. 3.2. 2009.
- [8] Bocchino, Robert, et al. "F Prime: an open-source framework for small-scale flight software systems." (2018).