Development of a Compact Space Imaging Module for Nano- and Pico- Multisatellite Earth Observation Systems

Pavel Lukashevich
UIIP NAS Belarus
Minsk, Republic of Belarus
e-mail: pavel.lukashevich@newman.bas-net.by

Alexei Belotserkovsky
UIIP NAS Belarus
Minsk, Republic of Belarus
e-mail: alex.belot@gmail.com

***Abstract*— This paper introduces the concept and prototype of EO-NANO, a compact space imaging module designed for nano- and picosatellite Earth Observation (EO) systems. The module features a hybrid architecture that integrates a Raspberry Pi Compute Module 4 (RP CM4) as the main processor and a Raspberry Pi Pico (RP2040) microcontroller for auxiliary control and data transfer. Key hardware features include a multiplexing solution that accommodates up to four camera interfaces and a shared SD card mechanism to enhance fault tolerance and power efficiency. The primary objective of this work is to develop an integrated hardware and software system that simplifies and standardizes the integration of ERS payloads into small satellite platforms, thereby making space imaging more accessible and affordable.**

***Keywords*—CubeSat, Payload, Nanosatellite, Earth Observation (EO), Software, Raspberry Pi, Compute Module 4, Raspberry Pi Pico, CAN, SD Card.**

I. Introduction

In recent years, nano- and picosatellite technologies have advanced significantly, driven by the miniaturization of electronic components and the standardization of platforms such as CubeSat. These satellites, typically classified by mass (nano: 1–10 kg, pico: <1 kg), are becoming increasingly capable of performing complex missions, including EO, thanks to improvements in attitude control systems, onboard data processing, and imaging sensors [1]. The use of constellations of these small satellites allows for a much higher revisit frequency, which is critical for applications requiring timely data, such as disaster monitoring. For instance, a proposed 6U CubeSat constellation could replace five large 156 kg RapidEye satellites with 35 smaller 8 kg satellites, enabling imaging every 3.5 hours instead of every 24 hours [2, 3].

The integration of nanotechnology has further spurred the development of highly efficient, compact sensors suitable for small satellites [1], while missions like NASA's Mars Cube One (MarCO) have demonstrated their potential even for interplanetary travel [4]. In the commercial sector, companies such as Planet Labs and Spire Global have successfully deployed large CubeSat constellations for Earth observation and weather forecasting, confirming the operational viability of small satellites [5]. Moreover, these platforms play a vital role in educational and research programs, expanding access to space for smaller nations and academic institutions [6].

II. Motivation

Despite these advancements, integrating optical-electronic systems as a primary or secondary payload remains a non-trivial challenge. High costs, complex integration, and limited imaging frequency are common obstacles. This project addresses these issues by proposing a standardized and simplified ERS payload designed to be more accessible for a broad community of small spacecraft developers.

The result is the EO-NANO, an experimental prototype of a compact space imaging module. It is a fully integrated hardware-software complex built on a hybrid architecture that employs the high-performance Raspberry Pi Compute Module 4 (RP CM4) for image processing and a Raspberry Pi Pico (RP2040) microcontroller for control and redundancy. By using widely available and cost-effective components, the EO-NANO module provides an efficient solution for image acquisition, onboard processing, and data management, paving the way for more flexible and scalable ERS missions.

The EO-NANO experimental prototype, software, image processing algorithms, design documentation, and a demonstration model of the satellite platform (DMSP) were created as part of the "Complex-SG" union program for 2023-2026 (9SG4.2-219).

III. Architecture and Design

The EO-NANO module was conceived as an experimental prototype to investigate the creation of a compact space imaging payload using low-cost, commercially available components. The primary objective is to develop a universal, customizable, and scalable payload that can be integrated into nano- and picosatellites to perform Earth Remote Sensing tasks. The design targets key technical specifications, including dimensions up to 100x100x25 mm (without cameras), a mass not exceeding 0.250 kg, and a ground spatial resolution of 100–2000 meters per pixel from a 500 km orbit. The system is engineered to automatically capture, process,

https://doi.org/10.51408/csit2025_64

compress, and store images for subsequent transmission to Earth.

At its core, the EO-NANO module features a hybrid architecture designed to balance performance, power consumption, and reliability. This architecture pairs a Raspberry Pi Compute Module 4 (RP CM4) as the main processing unit with a Raspberry Pi Pico (RP2040) microcontroller. The RP CM4 is responsible for the computationally intensive tasks of image acquisition and processing from up to four integrated cameras, which can operate in RGB or RGB+NIR spectra. The RP2040 serves as an auxiliary controller, managing CAN bus communications and providing redundant data access pathways. To accommodate multiple sensors, the hardware implements an analog multiplexing solution that allows the system to switch between different cameras, effectively expanding its imaging capabilities despite the limited physical interfaces on the main processor. The physical layout of the experimental prototype, featuring its 3D-printed housing and camera arrangement, is shown in Fig. 1.

Figure 1. Experimental prototype of the EO-NANO module with 3D printing housing elements and four cameras

A critical feature for enhancing fault tolerance and power management is the shared storage mechanism. Using two HEF4053 triple analog switches, the SD card interface is dynamically multiplexed between the RP CM4 and the RP2040. By default, the card is connected to the CM4 for data writing during an imaging session. However, this design allows the RP2040 to independently take control of the SD card via a GPIO signal, enabling it to read stored data and transmit it over the CAN bus. This ensures mission data can be retrieved even if the main processor is powered down to conserve energy or has experienced a failure. The module's power system operates from an 8.5V onboard bus, which is first regulated by a primary DC-DC converter to a stable 5V main power line. This line, in turn, feeds several low-dropout linear regulators to generate the distinct 3.3V and additional 5V lines required by the cameras, processors, and various transceivers.

IV. Configuration

Prior to the execution of the primary application software, the underlying Raspberry Pi OS requires specific configuration to enable and correctly interface with the module's custom hardware peripherals. This is achieved primarily through the use of Device Tree Overlays (DTOs), which inform the Linux kernel how to interact with non-standard components. For the EO-NANO module, several custom overlays are employed. A key overlay configures the MCP2515 CAN controller, mapping it to the SPI1 bus and assigning the correct GPIO for its interrupt line, thereby creating the can0 network interface. Another critical overlay modifies the behavior of the secondary SD card interface, forcing it into a slower, 1-bit communication mode. This adjustment is essential for ensuring reliable operation with the hardware multiplexer that arbitrates access between the main processor and the auxiliary microcontroller. The system's master configuration file also explicitly defines the initial state for GPIOs controlling camera bank selection and power, ensuring the system boots into a safe, predictable state.

To validate these hardware configurations and ensure system integrity before operational deployment, a suite of diagnostic scripts was developed. These scripts facilitate targeted testing of each key subsystem. A comprehensive camera diagnostic script automates the process of verifying I2C communication, enumerating all detected sensors via the libcamera framework, and performing test captures at all supported resolutions. For low-level debugging, an interactive script provides manual control over the GPIOs managing camera power and bank selection. Further scripts were created to benchmark the read/write performance of the external SD card and to provide a quick, system-wide status report on all critical interfaces, including processor temperature, CAN bus state, and SD card parameters. This testing framework allows for robust hardware verification and rapid fault isolation during integration.

V. Software

The operational logic of the EO-NANO module is managed by a modular software suite developed in Python, designed for autonomous execution on the configured platform. The software orchestrates the complete mission cycle, from image acquisition to final data storage. The process begins with the main controller script initiating the capture sequence. It programmatically manages hardware resources, issuing GPIO commands to select a specific camera bank and apply power. Subsequently, it uses the Picamera2 library to execute a scripted imaging session, capturing a series of frames from each available sensor with varying exposure settings to ensure data quality across different lighting conditions. This procedure is systematically repeated for each camera bank, guaranteeing comprehensive data collection.

Upon the conclusion of an imaging session, the software enters a data processing and packaging phase. All captured images and associated metadata logs from the session are consolidated and compressed into a single ZIP archive. To provide a robust method for data integrity verification, a SHA-256 checksum of the resulting archive is computed and stored in a corresponding file. Following packaging, the software manages the transfer of this data to the external storage medium. It first mounts the SD card, copies the data package and its checksum file, and then performs a critical on-device verification by recalculating the hash of the file on the SD card to confirm that the write operation was completed without corruption. Once the data is securely stored and verified, the software broadcasts a notification message over the CAN bus, informing the main satellite bus that a new data package is available for downlink.

VI. DEMONSTRATION PLATFORM

To validate the integration and functionality of the universal imaging payload, a dedicated test environment, the Demonstration Model of the Satellite Platform (DMSP), was engineered. This platform was designed to emulate the core functionalities of a cost-effective small satellite bus, providing a realistic framework for testing the EO-NANO module against constraints analogous to those found in space applications. The DMSP allows for the comprehensive validation of key operational scenarios, including power consumption analysis, communication protocol verification, and end-to-end data input/output demonstrations.

The central element of the DMSP is a custom power distribution and management board. This board is architected around a battery pack monitored by an integrated Battery Management System (BMS), which ensures cell balancing and provides protection against deep discharge. The main bus voltage is distributed to four independent, stabilized power channels, each managed by its own power controller. This configuration allows for precise, real-time current consumption monitoring and provides automatic overload protection for each connected payload. A Raspberry Pi Pico microcontroller serves as the low-level power manager, using an analog multiplexer to sequentially read the current draw from each of the four channels via its internal ADC. The overall system is coordinated by a higher-level Raspberry Pi Compute Module, which communicates with the Pico and provides external data interfaces via SPI-to-CAN converters.

A key design feature of the DMSP is its versatile payload interface system, which was developed to ensure broad compatibility. It includes a 100-pin external connector designed for the current EO-NANO module, as well as an internal slot intended for a future, more compact iteration named EO-PICO. This dual-interface approach was intentionally designed to be compatible with the common payload connection standard utilized by established CubeSat platforms, such as those in the SPUTNIX [7] product line, ensuring a clear pathway for future integration with industry-standard buses. A 3D model of the DMSP assembly is shown in Fig. 2.

The utility of the DMSP extends beyond the validation of the EO-NANO and its direct successors. It is also intended to serve as a standardized integration and testing platform for other collaborative development efforts within the broader research program. For instance, it is planned to be used for the validation of a separate module, developed by program partners, which is designed to perform intelligent, real-time processing of satellite imagery directly onboard the spacecraft. This positions the DMSP as a foundational asset for fostering parallel development and ensuring interoperability between different payload systems.

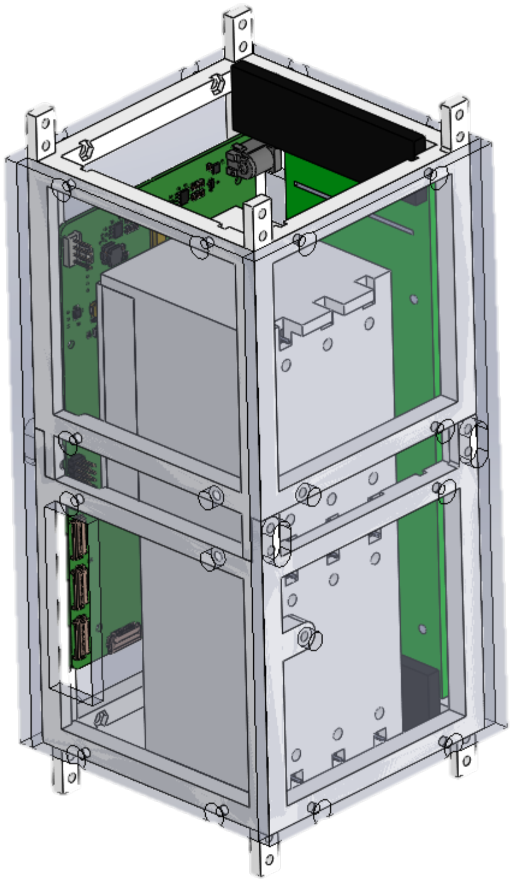

Figure 2. 3D model of experimental prototype of DMSP

VII. FUTURE WORK

The EO-NANO module has successfully completed its initial phase of autonomous testing, validating the functionality of its entire operational workflow, from image acquisition through to data packaging and storage, in a standalone configuration. This successful validation has established a robust baseline for the next stage of development. Applying the design principles and lessons learned from this initial work, the subsequent phase will involve the implementation of a second-generation module, designated EO-PICO. This new module is being engineered for the more compact internal slot of the Demonstration Model of the Satellite Platform (DMSP) and will undergo a similar regime of autonomous trials to validate its core functions independently.

Following the successful individual validation of both modules, a comprehensive integrated testing campaign will be conducted. This crucial stage will involve operating both the EO-NANO and EO-PICO payloads concurrently within the DMSP environment. The primary objective of this campaign is to identify and resolve potential interoperability challenges, which will inform any necessary refinements to the payload modules and the DMSP's management software. It is anticipated that this iterative process of design, autonomous testing, and integrated validation will provide a substantial empirical foundation and establish the technical expertise necessary to address the complexities inherent in the development, programming, and practical deployment of advanced EO payloads.

VIII. CONCLUSION

This paper has detailed a comprehensive approach to the development of an affordable and flexible payload for Earth Observation, culminating in the EO-NANO experimental prototype. The work demonstrates that by utilizing a hybrid architecture of commercial off-the-shelf components and implementing a robust, modular software suite, it is possible to create a highly capable and adaptable imaging system. The methodical validation of this system, supported by a dedicated demonstration platform, establishes a foundational framework for integrating advanced payloads into nano- and picosatellite systems. The presented concept, with its emphasis on fault-tolerant design and a complete, verifiable data workflow, directly addresses the key challenges of accessibility and reliability in small satellite missions. The continued development outlined herein promises not only to refine this technology but also to build critical expertise for future innovations in this field.

REFERENCES

[1] A. Denizli, T. Kocakulak and M. Moretto, “Bringing satellite and nanotechnologies together: unifying strengths against pollution and climate change”, *Frontiers in Nanotechnology*, vol. 3, 2024. DOI: 10.3389/fnano.2024.1332820.

[2] Nanostar Project, First Steps, NANOSTAR’s Documentation, version 1.0rc2. [Online]. Available: https://nanostar-project.gitlab.io/main/source/first-steps/index.html. [Accessed: June 30, 2025].

[3] S. R. Tsitas and J. Kingston, “6U CubeSat commercial applications”, *The Aeronautical Journal*, vol. 116, no. 1176, pp. 189–198, 2012. DOI: 10.1017/S0001924000006692.

[4] NASA Jet Propulsion Laboratory, Mars Cube One (MarCO). [Online]. Available: https://www.jpl.nasa.gov/missions/mars-cube-one-marco/. [Accessed: June 30, 2025].

[5] C. Foster, J. Mason and C. Beierle, “Orbit Determination and Differential-drag Control of Planet Labs Cubesat Constellations”, *arXiv:1509.03270* [astro-ph.IM], 2015.

[6] J. N. Pelton and S. Madry, “Student Experiments, Education, and Training with Small Satellites”, *Handbook of Small Satellites*, 2020. DOI: 10.1007/978-3-030-36308-6_34.

[7] SPUTNIX, Typical configurations of CubeSat platforms. [Online]. Available: https://sputnix.ru/en/satellite-platforms/cubesat-platforms/. [Accessed: June 30, 2025].