SSC21-WKII-08

STP-H7-CASPR: A Transition from Mission Concept to Launch

Noah Perryman, Theodore Schwarz, Thomas Cook, Seth Roffe, Antony Gillette,
Evan Gretok, Tyler Garrett, Sebastian Sabogal, Alan George
NSF Center for Space, High-performance, and Resilient Computing (SHREC) – University of Pittsburgh
4420 Bayard Street, Suite 560, Pittsburgh, PA, 15213; 814-360-1900
noah.perryman@pitt.edu

Ricardo Lopez
SATLANTIS MICROSATS SL
University Science Park, Sede Building, 48940, Leioa, Bilbao, Spain; +34 944 344 780
lopez@satlantis.com

ABSTRACT

The Configurable and Autonomous Sensor Processing Research (CASPR) project is a university-led experiment developed by student and faculty researchers at the NSF Center for Space, High-performance, and Resilient Computing (SHREC) at the University of Pittsburgh for the Space Test Program – Houston 7 (STP-H7) mission to the International Space Station (ISS). Autonomous sensor processing, the mission theme of the CASPR experiment, is enabled by combining novel sensor technologies with innovative computing techniques on resilient and high-performance flight hardware in a small satellite (SmallSat) form-factor. CASPR includes the iSIM-90, an innovative, high-resolution optical payload for Earth-observation missions developed by SATLANTIS MICROSATS SL. For the CASPR mission, the opto-mechanics of iSIM-90 will be mounted atop a gimbal-actuated platform for agile, low-GRD (ground-resolved distance), and multispectral Earth-observation imaging. This mission will also feature the Prophesee Sisley neuromorphic, event-driven sensor for space situational awareness applications. The CASPR avionics system consists of the following: three radiation-tolerant, reconfigurable space computers, including one flight-proven CSP and two next-gen SSPs; one µCSP Smart Module; one power card; and one backplane. CASPR also features a sub-experiment with an AMD GPU to evaluate new accelerator technologies for space. CASPR is a highly versatile experiment combining a variety of compute and sensor technologies to demonstrate on-orbit capabilities in onboard data analysis, mission operations, and spacecraft autonomy. As a research sandbox, CASPR enables new software and hardware to be remotely uploaded to further enhance mission capabilities. Finally, as a university-led mission, cost is a limiting constraint, leading to budget-driven design decisions and the use of affordable methods and procedures. Other factors, such as a power budget and limited equipment, facilities, and engineering resources, pose additional challenges to the CASPR mission. To address these challenges, we describe cost-effective procedures and methods used in the assembly, integration, and testing of the CASPR experiment.

INTRODUCTION

Due to continued advancements in sensor technology and innovations in spacecraft autonomy, space missions require increasingly more high-performance computing to address these big-data challenges. Dependable, high-performance computers are needed to compress onboard large volumes of sensor data into actionable information to overcome limitations in downlink and to autonomously execute critical mission functions. The capability for onboard autonomous sensor processing, while potentially increasing mission complexity, offers a unique solution to this big-data challenge.

However, these big-data challenges are compounded by stringent constraints in size, weight, power, and cost (SWaP-C) and reliability considerations due to environmental hazards such as radiation, extreme temperatures, vibrations, and vacuum. Radiation effects on electronic devices include transient, single-event effects (SEEs), caused by particles striking the device that can induce a variety of destructive or nondestructive effects. Radiation effects can also be cumulative, such as total ionizing dose (TID), which refers to the parametric degradation due to the device absorption of ionizing radiation over time, and displacement damage dose (DDD), the degradation of the device lattice structure due to continued impacts of nonionizing radiation.[1]

To address these challenges, space missions are increasingly adopting small satellites (SmallSats) as low-SWaP-C platforms enabled by the miniaturization

of key spacecraft components. SmallSats, along with a growing industry in launch ridesharing, has substantially enabled spaceflight opportunities to more military, commercial, and civil operators. Recent surveys have noted that in the past ten years, the nano/microsatellite segment has grown by a factor of ten, with civil operators, such as universities, having launched more satellites in 2019 than any other segment.[2] SmallSat missions often include commercial-off-the-shelf (COTS) devices to improve performance and affordability over traditional radiation-hardened (rad-hard) devices. The evaluation of new capabilities for autonomous sensor processing in space on SmallSat platforms presents a unique opportunity for addressing these design challenges and demonstrating advances in sensor technologies. To exemplify the capabilities of SmallSats for demonstrating onboard autonomous sensor processing, we introduce the Configurable and Autonomous Sensor Processing Research (CASPR) experiment for the upcoming Space Test Program – Houston 7 (STP-H7) mission to the International Space Station (ISS). The CASPR experiment, developed at the National Science Foundation (NSF) Center for Space, High-performance, and Resilient Computing (SHREC) at the University of Pittsburgh, aims to study and evaluate new technologies including sensors, computer systems, and machine-learning and computer-vision (ML/CV) applications for space-based sensing with autonomous sensor processing.

In addition to featuring many complex facets that highlight the aforementioned challenges of designing for spaceflight, CASPR introduced other obstacles as a result of being a university-led mission. Constrained budgets, system and resource limitations, lack of design expertise, student graduation, and knowledge transfer were significant considerations and issues throughout the design process. This article also seeks to inform and equip future university-led, SmallSat mission teams for success. The dilemmas faced in schedule and resources by universities compared to larger commercial mission designers will be considered and addressed.

BACKGROUND

This section provides a cursory overview of the STP-H7 mission along with the key concepts, hardware, and flight history that led to the formulation of CASPR. Introductory descriptions of the sensors surrounding the hardware are also presented. This paper is a revision and extension of a previous paper from the 2020 Small Satellite Conference, which describes the CASPR mission prior to significant development, assembly, and integration and testing (I&T).[3]

STP-H7

The U.S. Department of Defense (DoD) Space Test Program (STP) was created in 1965 to provide an economic and efficient process to enable spaceflight opportunities for the DoD space science and technology community.[4] The STP Houston office serves as the sole interface to NASA for all DoD payloads deployed onto the ISS and other human-rated launch vehicles.

STP-H7 is an upcoming mission that will feature multiple experiments, including CASPR, that will be integrated and flown under the management and direction of the DoD STP Human Spaceflight Payloads Office. STP-H7 is expected to launch on SpaceX-24 in late 2021 and will be installed onto the Columbus External Payload Facility on the exterior of the ISS.

Hybrid Space Computing

The National Science Foundation (NSF) SHREC center, formerly the Center for High-performance Reconfigurable Computing (CHREC), created a hybrid space-computing concept for high-performance, dependable onboard processing.[5]

This concept features a hybrid system-on-chip (SoC) and architecture design supplemented with dependable computing. At the forefront of this concept is a hybrid SoC, combining multiple distinct computing architectures into one device to attain the advantages of each. The hybrid architecture combines COTS technologies, such as the hybrid SoC and memory for performance, energy efficiency, and cost benefits, with rad-hard technologies, such as the watchdog, power circuits, and nonvolatile storage for dependability, to achieve the nexus in performance and dependability. Finally, novel dependable-computing techniques are incorporated for SEE mitigation to further enhance reliability.

The first realization of this concept was the CHREC Space Processor (CSP), developed by student and faculty researchers at the NSF CHREC center in collaboration with CHREC partners. CSP is a 1U single-board computer that features a Xilinx Zynq-7020 SoC, which combines a fixed-logic, dual-core ARM Cortex-A9 CPU and reconfigurable-logic Artix-7 FPGA fabric. The next realization was the µCSP, a sub-1U system-on-module (SoM) designed for reduced SWaP-C, that features a Microchip SmartFusion2 (M2S090) SoC, which combines a fixed-logic ARM Cortex-M3 CPU with a reconfigurable-logic IGLOO2 FPGA fabric.[6] µCSP is paired with the Smart Module system as a framework for designing a series of reusable hardware platforms that can be easily configured, integrated, and tested in preparation for a new mission.[7] The latest iteration of the hybrid space

computing concept is the SHREC Space Processor (SSP). SSP builds upon CSP to create a new platform with improved computational, memory, and communication capabilities. SSP features an upgraded Xilinx Zynq-7030/7035/7045 SoC with FPGA-dedicated DDR memory and multi-gigabit transceivers. These three hybrid space computers in their flight-model (FM) configurations are shown in Figure 1.

Figure 1: CSP Rev. C FM (Left), µCSP Smart Module (Middle), and SSP FM (Right)

SHREC has previously participated in two successful STP missions: STP-H5 and STP-H6. STP-H5 included the ISS SpaceCube Experiment Mini (ISEM) experiment, which featured the CSP sub-experiment developed by SHREC and NASA Goddard Space Flight Center. STP-H5-CSP featured two CSP Rev. B flight models (FMs) with an industrial camera pointing towards nadir for Earth-observation imaging and processing. STP-H5-CSP launched on SpaceX-7 in February 2017 and has been operating on the ISS Express Logistics Carrier 1 (ELC-1) since then. STP-H6 included the Spacecraft Supercomputing for Image and Video Processing (SSIVP) experiment, which featured a cluster of five CSPs (one Rev. C FM and four Rev. B FM), one µCSP Smart Module, and two industrial cameras with varied fields-of-view pointing towards near-nadir. This experiment investigated high-performance computing techniques, such as parallel and FPGA-accelerated processing, for onboard supercomputing.[8] SSIVP flew on SpaceX-14 in May 2019 and has been operating on the ISS Express Logistics Carrier 3 (ELC-3) since then. The STP-H5-CSP and STP-H6-SSIVP experiments are both illustrated in Figure 2.

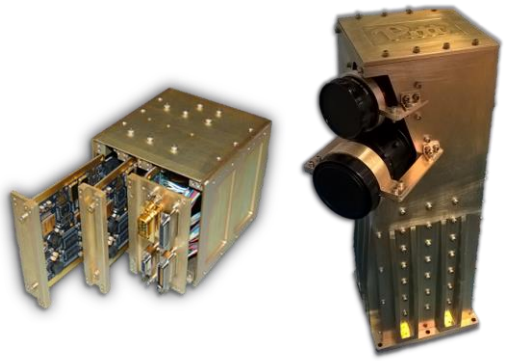

Figure 2: STP-H5-CSP (Left) and STP-H6-SSIVP (Right) Flight Enclosures

Accompanying the hybrid space computers on the STP-H7-CASPR experiment is an AMD Embedded GX-216HC SoC with a 16GB SSD and 4GB of DDR3 memory realized on a Qseven module. The Qseven module, based on the Qseven concept, is a sub-1U COTS SoM card that integrates multiple computer components with a 12W power limit and is mounted onto an application-specific carrier board.[9] The custom carrier card can be used to break out application-specific I/O to provide an interface for external hardware. The Qseven module and custom carrier card, collectively referred to as the Space GPU (SGPU), for the CASPR experiment are shown in Figure 3.

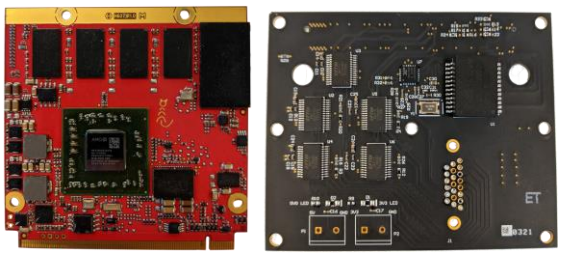

Figure 3: AMD SoM (Left) and SHREC Custom Carrier Card (Right)

Prophesee Sisley Neuromorphic Sensor

The Sisley neuromorphic sensor, developed by Prophesee, is a neuromorphic, event-based sensor. Neuromorphic sensors are composed of independent pixels that are sensitive to the events that occur in their position in the field-of-view (FoV). Unlike conventional vision cameras, which capture entire frames of data at fixed framerates, event-driven, neuromorphic sensors primarily capture in the time domain and generate an asynchronous stream of events, creating a sparse representation of the view by reporting only dynamic changes in light intensity. Events are generated only when changes are detected, and any information associated with the static background is

ignored. This paradigm allows neuromorphic sensors to capture at low temporal resolutions, on the order of microseconds, while maintaining a low data rate, which is useful for resource-constrained systems.

The operating principle of an event-based pixel follows the use of photodiodes to measure brightness in the FoV. The change detector of each pixel individually integrates over the luminosity to determine an increasing or decreasing polarity. These local pixel mechanisms lead to a high dynamic range (>120dB), meaning bright sources do not completely saturate the FoV.

The Sisley, shown in Figure 4, has a 640×480 pixel resolution fabricated in 180 nm CMOS-CIS technology. The maximum event-rate for the sensor is 66 mega-events per second, where events are encoded using address-event representation. This representation describes data in the form of tuples containing the positional coordinates (*x*,*y*), the timestamp, and the polarity of increasing or decreasing light intensity of an event.

Figure 4: Prophesee Sisley Neuromorphic Sensor

SATLANTIS iSIM-90 Sensor

The iSIM-90 (integrated Standard Imager for Microsatellites), developed by SATLANTIS, is a next-generation, multispectral, high-resolution optical imager for Earth observation. The design combines class-leading performance via the use of cutting-edge technologies, significantly reduced build times, and a new level of affordability. This combination approach will provide industries and governments with the ability to acquire and access unparalleled high-resolution data. The iSIM-90 opto-mechanical structure is depicted in Figure 5.

Figure 5: SATLANTIS iSIM-90 Opto-Mechanical Structure

STP-H7-CASPR HARDWARE

The CASPR flight model, shown in Figure 6, houses an approximate 6U avionics payload along with the 8U binocular camera iSIM-90 optics mounted atop a gimbal-actuated platform. The electronics payload contains seven flight cards: one CSP Rev. C, one µCSP Smart Module, two SSP Flight Models, one power card, the SGPU, and one backplane to connect all the flight electronics, sensors, and peripherals. The neuromorphic sensor is also integrated inside the flight enclosure. This section details the flight electronics, how they are connected electrically, and the mechanical and thermal structure.

Figure 6: STP-H7-CASPR Experiment

Flight Electronics

The flight electronics are composed of three major subsystems: the flight computers, power system, and backplane. The flight computers include one CSP, two SSPs, the SGPU, and one µCSP Smart Module. CSP fronts the CASPR experiment as the head node and is

responsible for payload management and command and data handling (C&DH). CSP interfaces directly to the STP-H7 flight computer, called the Data Interface Computer for Experiments (DICE), via RS-422. The SSPs serve as camera nodes and science data processors in a dual-processing configuration. Both SSPs interface directly to both iSIM-90 cameras for acquisition of Earth observation imagery and contain abundant resources in the FPGA to enable onboard acceleration of compute-intensive ML/CV applications. The SGPU serves as a platform for experimental app acceleration. Finally, the µCSP Smart Module performs active thermal management and gimbal motor control. The active thermal-management function of µCSP regulates the temperature of the iSIM-90 optics by monitoring the temperatures of six thermocouples on iSIM-90 and driving solid-state relays (SSRs) to control six heaters to maintain six thermal zones of the iSIM-90 within a 7°C operational temperature range. µCSP also monitors four additional thermocouples on the flight enclosure to track the temperatures of the neuromorphic sensor, SGPU, gimbal motor, and power electronics. The gimbal-motor control function of µCSP operates the gimbal stepper motor via hysteresis mode switching to allow iSIM-90 to tilt its FoV up to 15°.

The power system for CASPR includes a chassis-mounted converter and electromagnetic interference (EMI) filter, the power card, and SSRs used for power distribution attached to the backplane. The frontend converter and EMI filter receive a 28VDC input from the STP-H7 pallet, filter it, and then convert it to a 12VDC output for the power card. The power card, which contains two rad-hard converters, receives 12VDC and generates 5VDC and 3.3VDC rails for distribution to the flight electronics. The power card is illustrated in Figure 7. Finally, a set of high-reliability SSRs, located on both the power card and backplane, are used to control the distribution of the voltage rails to the flight electronics, to activate two external ejector release modules (ERMs) for the gimbal-actuated baseplate beneath the iSIM-90, and to activate the shutters of both iSIM-90 cameras.

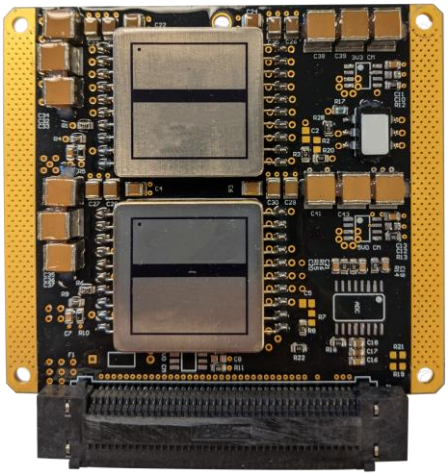

Figure 7: STP-H7-CASPR Power Card

Power management on CASPR leverages the power distribution system on the backplane to maintain an operational constraint to a budget of 60W. Under a full load, with all systems powered on, the operational power can exceed this power budget. To avoid exceeding this allotment, SSRs, controlled by the CSP and µCSP, are activated such that only a predetermined subset of the flight electronics and sensors are powered on simultaneously. The SSRs form power islands for both iSIM-90 cameras, shutter mechanism, gimbal motor, and heaters, the neuromorphic sensor, the SGPU, and each SSP. The CSP and µCSP Smart Module are always powered on.

The backplane implements the interconnect network and power distribution for all flight electronics, sensors, and peripherals. The CSP, SSPs, µCSP Smart Module, and power card are mounted directly onto the backplane. The backplane also exposes multiple Micro-D connectors for external interfaces (RS-422 for C&DH, power input, and debug) and internal interfaces, including the gimbal motor (control and limit switches), the iSIM-90 (Camera Link, thermocouples and heaters, and shutter control and telemetry), the thermocouples on the flight box, the SGPU (power, networking, and debug), ERMs, and limit switches. Finally, the backplane includes a ruggedized USB 2.0 Mini B connector for interfacing with the neuromorphic sensor. The flight backplane is shown in Figure 8.

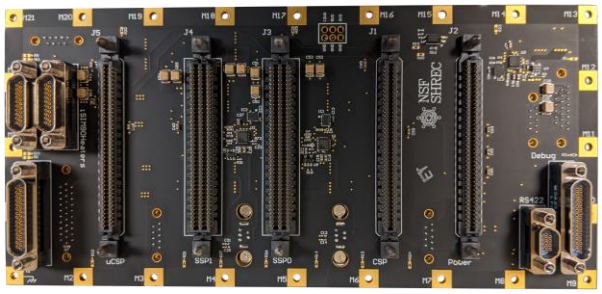

Figure 8: STP-H7-CASPR Backplane

System Architecture

The CASPR system architecture, named the Galaxy Strider, networks all flight electronics, sensors, and peripherals to create a complex platform for demonstrating multiple technologies. As the head node, CSP is central to the system architecture and interfaces directly to DICE via RS-422 for C&DH. The CSP and both SSPs form a trio to allow the CSP direct access to image data generated by SSP from iSIM-90 and to allow both SSPs to cooperate for dual processing. CSP interfaces with the neuromorphic sensor for event-data acquisition. CSP also interfaces with the µCSP Smart Module for commanding and telemetry of the thermal-management and gimbal-motor control functions. CSP also drives the SSRs to release the ERMs and shutters or to control the power states of several modules for power management. CSP, and primarily SSP0, interface to the SGPU for data processing with GPU acceleration.

To improve network dependability, the CASPR architecture features redundant interfaces and routing that allow for multiple connections between any two flight computers. For example, CSP and SSP0 can connect directly via a high-bandwidth SpaceWire interface or a redundant UART interface. Due to redundant routing, CSP and SSP0 can also connect indirectly via SSP1. Similarly, the Camera Link interfaces from both cameras are branched on the backplane to allow either SSP to operate both cameras and to enable CASPR to continue to use iSIM-90 in case of failure of an SSP node. The Galaxy Strider architecture is illustrated in Figure 9.

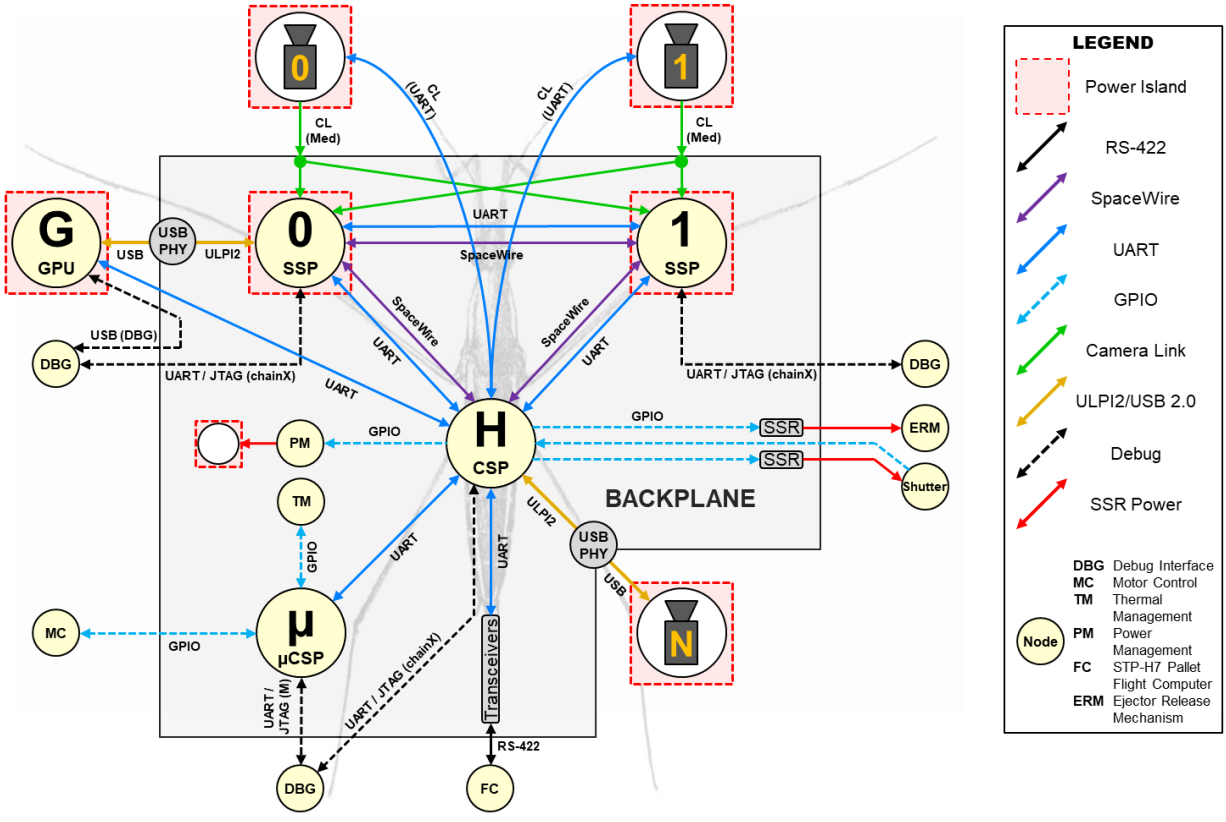

Figure 9: STP-H7-CASPR System Architecture

Mechanical Structure

The CASPR mechanical structure is divided into two main assemblies. The top half consists of the iSIM-90 vision system, and the bottom half consists of the anodized-aluminum avionics chassis with a gimbal-powered platform connecting the two. Data and power connectors protrude from the side of the chassis to connect to the STP-H7 pallet. For integration into a pallet with many experiments, CASPR was designed to have as small of a footprint as possible. This footprint was mostly defined by the size of iSIM-90 and the range of motion allowed in the gimbal mechanism.

The gimbal mechanism includes no space-rated parts to save on costs. Instead, several readily available COTS gears were used with a custom vacuum-rated stepper motor from Lin Engineering. By incorporating a 27:1 planetary gearbox and a worm gear set, the NEMA11 size motor can output enough torque to raise the gimbal platform. The gimbal mechanism is important to the successful completion of mission objectives, but failure of the gears or motor do not pose a significant risk to structural integrity or safety. Each component was subjected to testing in a vacuum chamber to ensure there were no mass-loss outgassing concerns.

The design of the gimbal mechanism was such that the teeth of the worm gear were not sufficient to keep the gimbal platform stationary during launch. For this purpose, redundant TINI E500 ERMs were used. These mechanisms allow a bolted joint to fasten two components together, and, using shape-memory alloy actuation, disengage that connection. Each ERM can support the preload required to keep the joint secure during launch loads. A stainless-steel spring is used to deflect the fastener after actuation to ensure the gimbal platform can raise and lower without interference. While mechanical stops prevent over rotation of the gimbal platform in the event the motor fails, limit switches for the raised and lowered position are used to ensure the motor control logic can prevent excessive rotation.

The gimbal platform, on which the iSIM-90 is mounted, allows the optics to cover a field of regard 15 times larger than a fixed alternative. The range of motion begins at nadir, and rotates 15° cross-track towards starboard at 1°/s. While the mechanism can account for some adjustment of the ISS attitude in roll, variation in pitch still has a large impact on iSIM90 performance. The FoVs of the iSIM-90 and the neuromorphic sensor are illustrated in Figure 10.

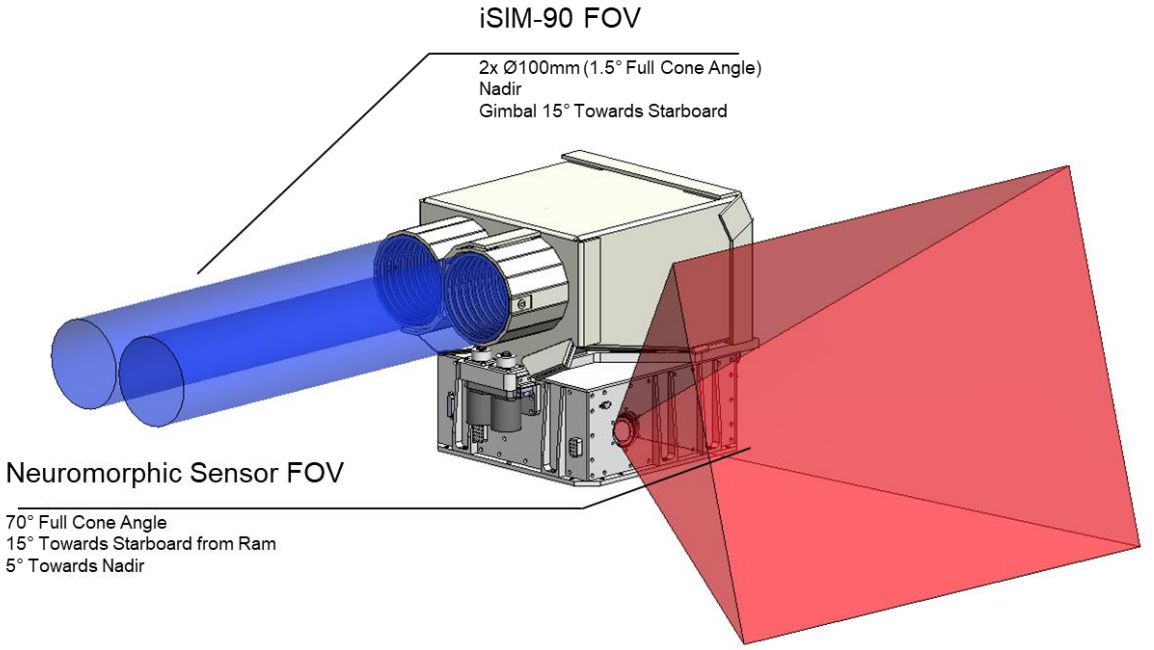

Figure 10: Sensor FoVs

Sensors

iSIM-90 is a modified Maksutov-Cassegrain optical design with a focal length of 775 mm and effective aperture diameter of 77.5 mm. The imager is designed to provide diffraction-limited images for Earth observations from blue band to near-infrared (NIR) band, with a spatial resolution of 2.5 m red/green/blue and 3 m NIR with a 13 km swath. The system relies on the technological integration of three key foundations: a binocular diffraction-limited set of telescopes working at visible and near-infrared wavelengths, a high-precision, robust, and light alloy structure supplemented with carbon-fiber rods, and a set of innovative CMOS array detector units. The construction elements of iSIM-90 can be divided into the opto-mechanics, the thermal control system, and the thermal cover, as shown in Figure 11.

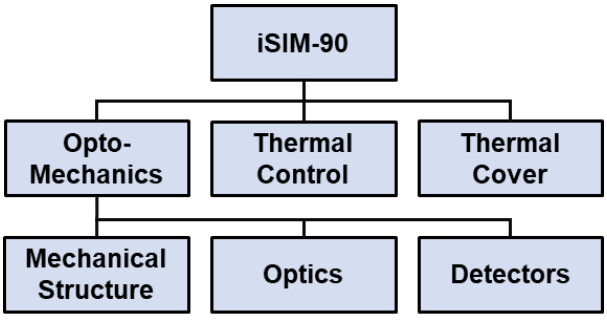

Figure 11: iSIM-90 Construction Elements

The opto-mechanical system of iSIM-90 is divided into two identical optical channels reinforced by a compact, lightweight, and thermally stable structure with a singular modified COTS detector mounted at the end of each optical channel. The overall system schematic is shown in Figure 12.

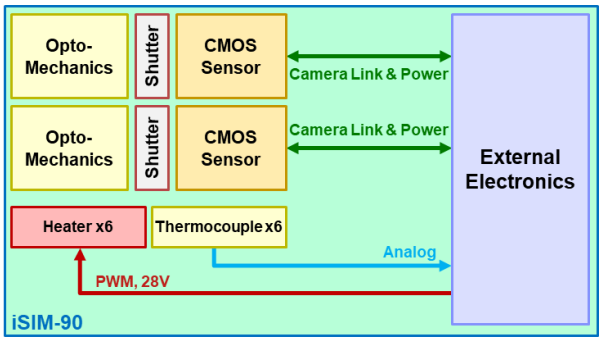

Figure 12: iSIM-90 System Schematic

The neuromorphic sensor is integrated inside the CASPR avionics chassis and aims towards ram/limb to view Earth's horizon. Half of the FoV is Earth, and the other half is space. The neuromorphic sensor has a custom aluminum enclosure to ensure survival under launch vibration loads. The neuromorphic sensor is used for space situational awareness and can be used to detect ISS movements (e.g., solar panels, moving arms, etc.), capsule docking, CubeSat deployment, space debris, and satellites. The sparse representation of the FoV generated by the neuromorphic sensor will enable CASPR to efficiently detect and track these objects.

STP-H7-CASPR SOFTWARE

The primary components of CASPR software included a Linux operating system, the core Flight Software, drivers for sensors, networking, and peripherals, and additional services and applications. The software architecture of CASPR includes command and telemetry handling, application management, networking, and fault tolerance. Additionally, CSP and SSP on CASPR include complete hardware/software stacks to facilitate high-throughput imaging and FPGA acceleration.

Flight Software Architecture

Both the CSP and SSP run Wumbo Linux, a custom operating system based on buildroot with Xilinx's Linux kernel fork and busybox userland. Wumbo Linux uses a volatile, initramfs-based filesystem that is reset on system reboot, and a persistent filesystem in the onboard NAND flash memory to store science data products and newly uploaded files. The boot image, which includes Wumbo Linux, the first-stage bootloader (FSBL), u-boot (second-stage bootloader), and FPGA configuration bitstream, are programmed into the flash memory. To improve the reliability of the boot process for CSP and SSP, these boot images are signed by RSA keys and programmed redundantly into flash memory. The RSA authentication and failover features of the Zynq SoC are enabled. Combined, RSA authentication allows the Zynq SoC to verify the integrity of a boot image, and, if corrupted, RSA failover enables the Zynq SoC to continue to attempt to verify subsequent images until it can boot successfully.

The SGPU runs Lubuntu 18.04 LTS on a persistent filesystem. The SGPU is connected to SSP0 via 100 Mbps Ethernet to CSP via UART. Conventional networking applications, such as SSH and SCP, can be used to transfer files and issue commands to and from the SGPU. The AMD SoC and embedded GPU of the SGPU are equipped with OpenCL for app acceleration, as well as Python, TensorFlow, OpenCV, and other frameworks not typically supported nor deployed in space platforms. One mission objective is to evaluate the survivability of the SGPU, an all-COTS computer, in the radiation environment of the ISS orbit and to evaluate fault-mitigation software designed to mitigate

radiation-induced data errors in parallelized workloads. Should the SGPU perform adequately in the harsh environment, there is significant opportunity for deployment of additional mainstream software in orbit.

Services and Applications

The CSP flight image uses NASA Goddard Space Flight Center's Core Flight Executive and Core Flight System (cFE/cFS) to enable services such as commanding, telemetry, and scheduling. Custom cFS applications were developed to manage communication with the STP-H7 DICE and handle operations such as file transfers. Camera control for both the iSIM-90 and Sisley are performed with custom applications called using the shell (SHL) cFS application. A list of custom cFS applications created for CASPR are described in Table 1. A communication diagram between CASPR cFS apps and other components is shown in Figure 13.

Table 1: CASPR cFS Applications

App	Function
AUTODL	Enables file downlink
CI	Modified version of the NASA CI app to receive and forward multi-part commands from DICEIF
CSM	Enables schedule upload and cFS app control
DICEIF	Controls the serial interface with the pallet communication system
HS	Gathers health and status information from each device, such as temperature and memory usage
FT_CSP	Enables file transfer over the software bus (e.g., to TO)
FTDP	Enables file upload from the DICE
SHL	Runs a Linux shell command on CSP
TO	A modified version of the NASA TO app to collect telemetry output for downlink

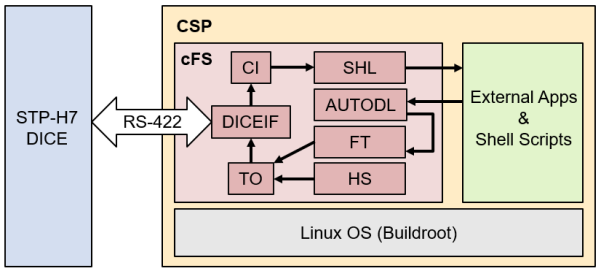

Figure 13: CASPR App Communication Diagram

cFE/cFS was not included on the SSPs as these devices do not require core C&DH and thus do not require the same suite of reliable flight software. The reduced compute and memory overhead allow both SSPs to be more effectively utilized for other mission operations, such as capturing image frames from the iSIM-90 or accelerating ML/CV apps on the FPGA.

FPGA Management

Both CSP and SSP feature complete hardware/software stacks for FPGA-based dynamic partial reconfiguration (PR) and Camera Link frame grabber (CLFG) for the iSIM-90, as shown in Figure 14. The PR stack enables a subset of the FPGA to be reconfigured dynamically at runtime, allowing for new FPGA designs (e.g., accelerators, soft-core processors, and other IP cores) to be deployed for evaluation on CASPR. This stack includes a large PR region in the FPGA with multiple clock and reset signals and AXI interfaces, allowing AXI-compliant IP cores to be interfaced as PR modules. When a partial reconfiguration module is programmed, the corresponding Linux drivers can be loaded for use by userspace applications and libraries. One example is the Xilinx Deep-Learning Processing Unit, a general-purpose convolutional neural-network (CNN) accelerator, for processing ML/CV apps.

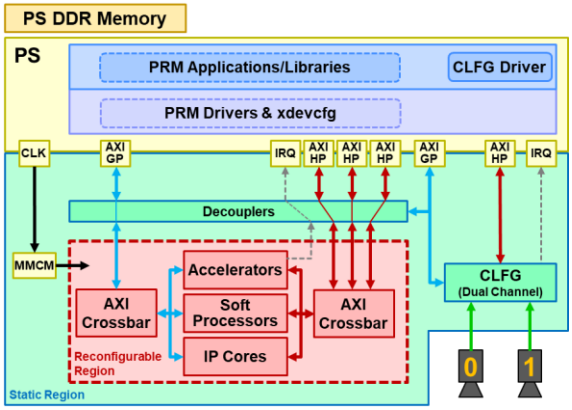

Figure 14: HW/SW Stacks for FPGA Reconfiguration, Acceleration, and Imaging

Both SSPs also feature a hardware/software stack for an FPGA-based CLFG with a supporting Linux driver. The CLFG contains dual Camera Link Medium Configuration (4-TAP, 12-bit) channels to enable two image-data streams for each camera on iSIM-90. The CLFG can capture multiple consecutive, variably sized frames (up to 4096×3072 or 12MP) at up to 26 frames per second.

In CASPR, CSP signals both SSPs to synchronously capture multiple frames from both cameras (all four filters) in a swath. These frames are then input into the SATLANTIS super-resolution application to generate a multispectral, low-GRD image for that swath.

Operations

CASPR will be operated from a ground-station system located at the NSF SHREC lab at the University of Pittsburgh. The command and telemetry systems are designed for versatility. Most ground commands encapsulate shell commands that will be run on the flight computers on orbit, allowing for the addition of numerous apps and scripts developed by student researchers. Results and data generated by commands and apps can be stored on the nonvolatile filesystem for later processing or downlinked directly to the ground-station computer for viewing, storage, and further processing. CASPR will support the upload of new software apps and FPGA hardware bitstreams throughout the life of the experiment, helping support a valuable stream of research, especially from a university perspective, that will make the investment into CASPR development worthwhile.

DEVELOPMENT, INTEGRATION, & TESTING

As a platform intended to demonstrate multiple technologies, CASPR manifested into an overwhelmingly complex system for a university-led mission, necessitating the need for efficient methods to develop CASPR in many aspects (hardware, software, mechanical, etc.). In this section, CASPR mission development, including formulation, design, integration, assembly, and testing, are discussed. Development principles, methods, and practices are detailed and analyzed to provide lessons learned and to compare with industry standards.

Initial Design and Development

CASPR was formulated with the objective to demonstrate and evaluate novel technologies (computers and sensors) and methods (ML/CV applications and operations) for autonomous sensor processing. The mission concept for CASPR, including the concept of operations, mission objectives, and preliminary specifications, was proposed to the Space Experiment Review Board (SERB) at the U.S. Air Force and U.S. DoD sessions in 2018. The mission proposal specified the inclusion of five flight computers and two novel sensors (iSIM-90 and Sisley) to demonstrate capabilities for autonomous sensor processing.

Once the mission concept was formulated, the CASPR architecture was designed using a modular and iterative approach to minimize complexity. All computers, sensors, and electronics acted as distinct modules, and a network topology was designed iteratively to connect all modules into one system architecture. The CASPR architecture required several design considerations, including the number of I/O and compatibility of I/O voltages and standards, the characteristics of the network (dataflow, controlflow, and bandwidth), and the separation of modules in distinct power islands for power management. For example, the gimbal-actuated iSIM-90 platform required several interfaces, including two Camera Link (medium configuration), I/O for heaters and thermocouples, I/O for shutter control and telemetry, and I/O for gimbal control and telemetry, and power. To interface this platform with the flight computers, these interfaces were divided and addressed individually by function. The SSPs, which feature an abundance of resources for data acquisition and onboard processing, were assigned the role of camera nodes and science data processors and connected to the Camera Link interface (data subset) for imaging and processing. The µCSP Smart Module, a flight-proven, low-power controller, was assigned the role of active thermal management and gimbal-motor control and was connected to the I/O interfaces for heaters, thermocouples, and gimbal control and telemetry. CSP, serving as the payload manager, was connected to the Camera Link interface (control subset) for camera configuration, I/O interfaces for shutter telemetry, and I/O interfaces for SSRs to control the camera shutters and experiment-wide power states. As another example, the neuromorphic sensor only required a sole USB 2.0 interface and, due to its low data rate, was connected to CSP for event data acquisition and processing.

Additionally, to reduce the complexity of CASPR, lessons learned from prior missions, including STP-H5-CSP and STP-H6-SSIVP, were considered. For example, due to the continued success and flight-heritage of CSP and µCSP, both platforms were reused and assumed critical mission functions in CASPR. Similarly, software (cFE/cFS and Wumbo Linux), FPGA designs, and interfaces (e.g., Camera Link and SpaceWire) from previous missions were leveraged and adapted for CASPR. One improvement upon previous missions was designing for testability to reduce complexity during testing and validation of flight hardware. The backplane was designed to allow for testing of individual modules without depending on essential cards to avoid delays (e.g., supply power directly to the backplane to bypass an unavailable power card). Furthermore, a debug board was created to provide debug interfaces (serial and JTAG) to enhance the testability of CASPR flight computers.

The complexity of CASPR introduced significant schedule and timeline challenges. One prominent example was the concurrent development of the SSP during mission development. Issues encountered during the development of SSP required multiple revisions that introduced delays in I&T. The complexity of nearly all PCBs included in CASPR required significant time for

manufacturing and assembly. Donations were sought for many rad-hard parts required for the mission to maintain budgetary requirements and reduce the impact of long lead times. Furthermore, assembly procedures, such as lead forming, epoxying, and conformal coating introduced additional scheduling overhead and delays. Finally, development milestones needed to be carefully coordinated according to the STP-H7 mission schedule.

Power management was another source of complexity in CASPR. The total power consumption of all modules on CASPR exceeds the maximum power budget allotted by STP, as well as the thermal limits of the flight chassis. To support multiple technologies on CASPR, SSRs were added to control the power state of several modules and to only enable a subset of modules whose total power consumption is within the power budget. Preset power modes were drafted to maximize experiment capability without exceeding power limits.

Finally, CASPR development incurred additional impediments due unique limitations associated with a university-led team. All contributors to CASPR were graduate students, many with full academic course loads, research obligations and deadlines, and other commitments, which posed challenges to development scheduling. Many students were new to mission development, had limited experience, and required time to become familiar with NASA standards and industry practices before being able to contribute to CASPR. Due to the multiyear timeline, multiple students graduated during mission development, and the transfer of knowledge to successors was a consistent challenge. As an academic mission with limited funding, many decisions were budget-driven to preserve expenditures (e.g., part selection, requesting part donations, selecting longer lead times, etc.). To maintain budget requirements and schedule deadlines, there were limited opportunities to troubleshoot and respin hardware. Fortunately, SHREC sponsors and vendors have provided substantial assistance to support CASPR development, including part donations, design reviews, and other opportunities.

Preliminary Testing

To minimize complexity, a modular and iterative approach was used for CASPR development and testing. Initially, development was performed on commercially available development kits, such as the Digilent PYNQ, Xilinx ZC706, and Emcraft SmartFusion2 Starter Kit that served as low-cost representations of CSP, SSP, and µCSP for module-level development. Active evaluation kits for CSP, SSP, and µCSP were also used to test directly on flight hardware. Subsequently, a pre-FlatSat platform, combining multiple development kits, was created for system-level development to account for interactions between multiple modules. Once the flight electronics were manufactured and assembled, a FlatSat platform was created allowing for the development and testing of actual flight hardware. The backplane and debug board, both designed for testability, were substantially beneficial for modular and iterative testing of the flight electronics.

To achieve this modular and iterative approach, a generalized testing procedure was used to systematically test and isolate design issues. Testing began with visual inspections of the flight electronics to ensure all components were placed correctly. The flight electronics were then tested for shorts at the node level (i.e., circuits on individual boards) followed by testing on custom active evaluation kits when applicable. Once testing at the node-level was completed, system-level evaluation began. Modules were connected on the backplane and then tested for shorts between boards. Upon successful testing at the node- and system-levels, the system was powered on, and testing of the software and FPGAs began. Testing at this level started with confirming that the electrical connections between boards, specified by the system architecture, worked as expected. Once all connections were tested, functional testing began to ensure the correct operation of all flight electronics and sensors. Any problems identified throughout the testing procedure were isolated and fixed ad hoc. The culmination of this testing was the functional CASPR FlatSat, shown in Figure 15.

Figure 15: STP-H7-CASPR FlatSat

During CASPR development, multiple issues were encountered that required an improvised solution to satisfy mission requirements using only the resources that were available. For example, one significant issue encountered in CASPR was the nonfunctional FPGA-side 4GB DDR memory on SSP. This memory was

critical, relied upon as a frame buffer for storing images captured by SSP from the iSIM-90. To maintain core functionality for the mission's primary objective, an improvised, alternative design was created that split the CPU-sided 1GB DDR memory into two 512MB segments: one for Linux and one for frame buffering. Operation plans were also adjusted to allow both SSPs to capture from both iSIM-90 cameras in parallel into their respective frame buffers. Although this limitation decreased the memory available for Linux and frame buffering, it allowed the primary mission objectives to be completed despite a significant hardware issue.

Another outstanding issue was the nonfunctional primary JTAG chain used to program the CSP and SSPs. Efforts were made during the design process to enable two separate, redundant chains for programming all devices together or the CSP alone. Fortunately, a software-based workaround was created to program boot images directly into NAND flash memory from Linux, resolving the issue and enabling full redundancy without performance degradation despite the nonfunctional JTAG interface.

Another challenge was the nonfunctional watchdog timer circuitry on SSP. A design issue prevented SSP from being restored by watchdog-triggered resets in the event of a crash. To resolve this issue, a multi-level workaround was created. SSPs directed their heartbeat signals to CSP, which would monitor these heartbeats. Whenever an SSP ceased to signal, CSP would power cycle the unresponsive SSP to restore it. The 1Gbps Ethernet connection between SSP0 and SGPU was also nonfunctional, limiting communication to a slow, secondary UART connection. Due to known issues with the SSP RGMII Ethernet PHY interface, a 100Mbps USB-to-Ethernet circuit was created that interfaces directly with the SSP ULPI2 (USB 2.0 PHY) interface instead. Although this workaround is suboptimal, the new connection substantially improved the communication bandwidth to the SGPU.

The µCSP Smart Module 2.5VDC converter was originally nonfunctional due to design oversights. Given timeline and budgetary constraints, a second revision of this board was not a viable solution. Instead, a drop-on module with an alternative 2.5VDC converter was used and placed on top of the footprint of the previous 2.5VDC converter. A similar solution was used for the motor driver circuit. The original operational amplifier used did not meet the necessary power requirements, and thus a replacement was needed. No suitable replacements were pin-compatible, so another drop-on module with an appropriate operational amplifier was used.

While SHREC students were testing the flight hardware, SATLANTIS performed the optical characterization test campaign for the iSIM-90 FM during the months of August and September of 2020. The characterization measurements were executed in an ISO5 cleanroom and in a vacuum chamber and included several processes. In the following paragraph, the main results of such characterization are presented, by means of optical performance indicators, including the modulation transfer function (MTF) and point spread function (PSF).

Among other parameters, the MTF is a good indicator of the system optical performance. The MTF is a metric quantifying the sharpness of the reconstructed image. In Figure 16, the MTFs of the iSIM-90 FM are presented for each of the four filters. The MTFs were obtained by imaging a resolution test target in the laboratory setup. These MTFs are obtained after applying SATLANTIS image super-resolution algorithm.

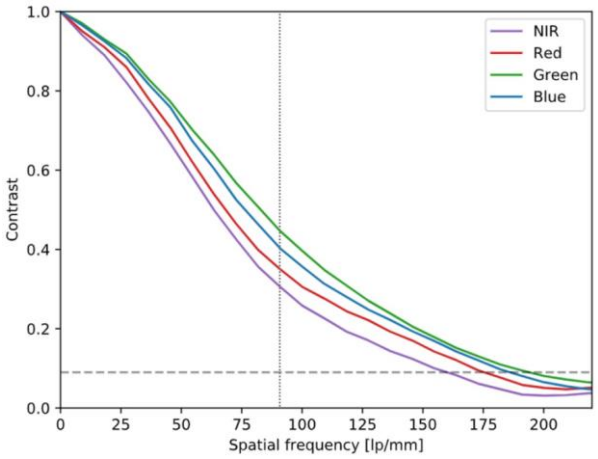

Figure 16: MTF Curves for the Four Bands of iSIM-90 FM, After Super -Resolution Algorithm. Credit SATLANTIS

The dashed horizontal line represents the Rayleigh limit of diffraction, and the dotted vertical line represents the Nyquist frequency. The MTF curves show the satisfactory spatial frequency response of the iSIM-90 FM for the four band-pass filters. These curves are proof of the desired optical performance of iSIM-90 and the high resolution of images after super-resolution. In addition, the same graph shows the MTF at Nyquist frequency is between 35% and 50% per band, which is also proof of remarkable optical performance.

From the resolution test target, the PSFs in each filter can also be derived. The PSFs shown in Figure 17 are native, meaning that the super-resolution algorithm has not been applied to them.

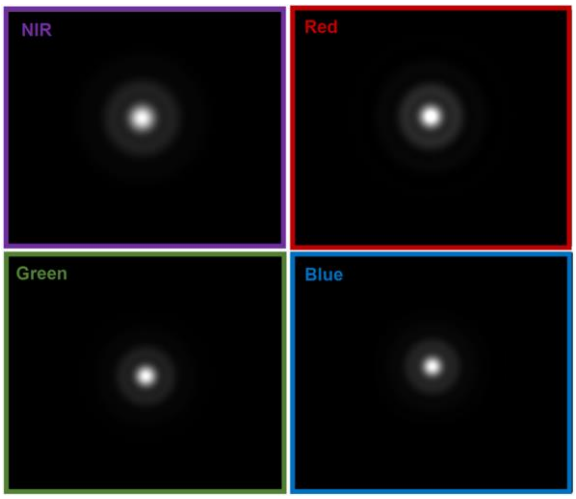

Figure 17: PSFs in the Four Filters of iSIM-90 FM. Credit SATLANTIS

These PSFs have the characteristic Airy rings, a property from diffraction-limited imaging systems. The PSF Airy radii (i.e., the radii of the first ring) are shown in Table 2.

Table 2: PSF Airy Radii for Each Filter

Filter	PSF Airy Radii
NIR	7.7 µm
Red	6.6 µm
Green	6.1 µm
Blue	5.7 µm

Design for Flight

Since the arrival of the flight hardware was staggered, the design and preparation of hardware for flight was intertwined with preliminary testing. Initial designs, except for the SSP engineering models, already incorporated rad-hard parts, but additional steps were required to prepare the cards for flight. Each design was assembled using a leaded solder process to avoid connection failures and increase reliability. Parts on the flight cards were epoxied for stability and to survive launch conditions. Each flight card was also conformally coated to survive the harsh environment of space. The flight harnesses were also designed and assembled inhouse for each of the connectors on the backplane, which included selecting proper materials, identifying all necessary connections between interfaces, and assembling the harnesses to fit within the flight enclosure.

Integration and Testing

After receiving the flight cards and preparing them for flight, all designs were assembled in the chassis, as shown in Figure 18. One significant challenge was assembling and integrating the flight harnesses for each of the connectors on the backplane. Each harness required precise measurements that could only be determined once the enclosure was partially assembled, leaving little time or space for error. Any issues had to be handled with limited space and available options. There were no replacement parts and no opportunities for revision at this stage in the timeline.

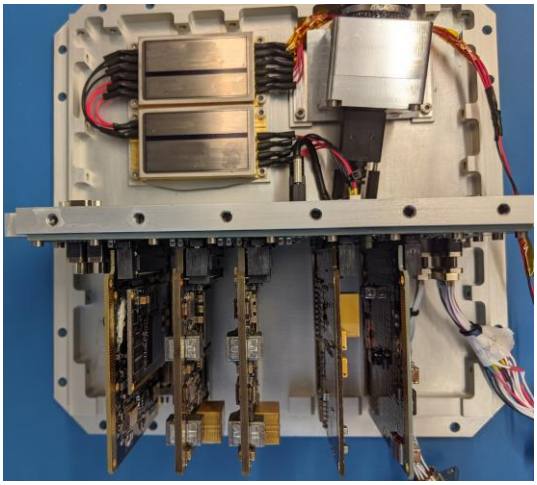

Figure 18: STP-H7-CASPR Internal Structure

As the hardware was assembled and tested, software was iteratively installed, verified, and adapted to reflect modifications to the hardware. The most critical pieces of software included cFE/cFS and associated components, userspace drivers for the network interfaces, CLFG, neuromorphic event capture, and PR framework. Additional third-party apps were added to add capabilities for image compression, format conversion, and other features. Scripts were prepared to properly initialize and configure hardware and to govern the transition between power modes. A single overarching functional-test script was designed to confirm the functionality of all sensors and interfaces via a single command. This script can be run inflight to verify continued functionality of all systems.

Testing the µCSP Smart Module was especially difficult due to its close integration with the iSIM-90, gimbal, and heaters. Once iSIM-90 was fully installed, as shown in Figure 19, the thermocouples were tested for accurate temperature measurement from each thermal zone on the iSIM-90. The heaters were then interfaced with the µCSP Smart Module and tested one by one for proper mapping to their respective thermocouples. In particular, the gimbal-motor driver circuit was a challenging, yet successfully validated component in CASPR. A hardware bug in the stepper-motor driver circuit prevented the motor from operating properly in a full-step configuration. Fortunately, the

motor-control logic could be rewritten in software to achieve half-stepping functionality. With careful adjustment and a slight reduction in speed, the motor was able to generate the required torque to open the iSIM-90 gimbal in microgravity.

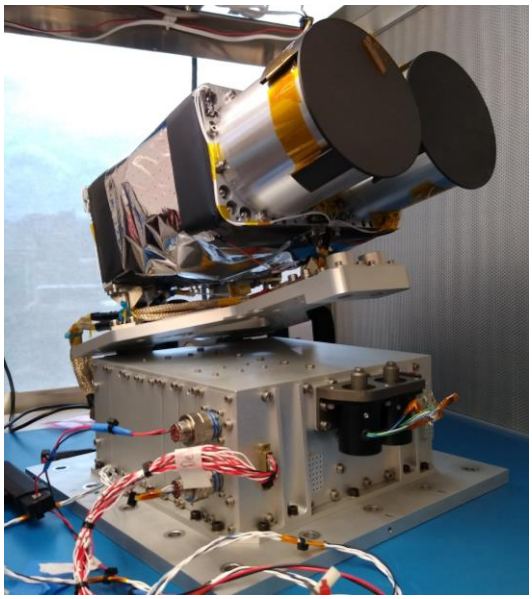

Figure 19: STP-H7-CASPR Flight Unit with Gimbal in Open Configuration

The precise optical verification testing of the integrated iSIM-90 instrument, depicted in Figure 20, also brought challenges. While the initial testing performed by SATLANTIS was supported by a live video stream that allowed for targeting and focus to be completed quickly, the low data rate of the flight interconnects this process impractical. To overcome this challenge, onboard image compression was performed on entire image frames for targeting, where quality was not essential. Next, for high-detail images of the focused fiber-optic target, a cropping feature was incorporated into the CLFG driver that output only the immediate area around the focused dot of light, allowing for the rapid capture of the required frames to confirm optical performance.

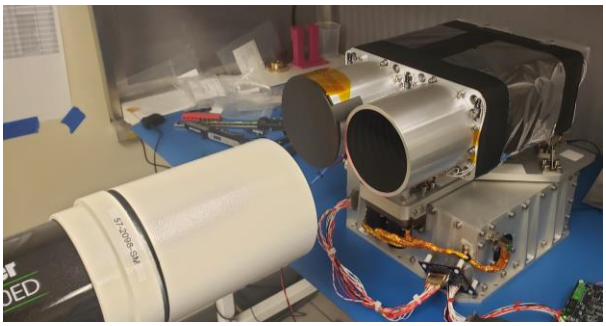

Figure 20: STP-H7-CASPR Optical Testing

Environmental Testing

As part of the requirements for satellite design, environmental testing is vital to ensure the survival of the mission. To ensure CASPR survives launch without any risks towards the safety requirements, a workmanship vibration test was performed at Naval Research Laboratory (NRL). Inertial measurement units (IMUs) were used to measure the frequency at various locations on CASPR. A random vibrational load representing the SpaceX Falcon 9 launch vehicle was performed in all three cartesian directions for 60 seconds. After each test, a sine sweep was performed to measure the shift in resonance frequencies at each IMU location. After each load test, a functional test was performed to ensure functionality of each component. Once the functionality was verified, the next random vibrational test was performed for the next coordinate. One of the configurations for these tests is shown in Figure 21.

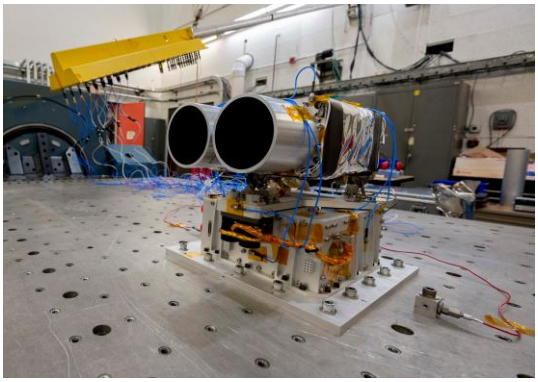

Figure 21: Vibration Testing for STP-H7-CASPR Experiment

After vibrational testing, thermal-vacuum (TVAC) testing was also performed at NRL to ensure CASPR's electronics systems could survive extreme temperatures in the vacuum of space. Thermocouples were placed around major areas of CASPR's structure to monitor temperatures during the test. One survival test was performed, ranging from -20°C to 70°C with the system

unpowered. After the cycle, functional tests were performed to confirm system functionality. Additionally, two operational tests were performed, ranging from 0°C to 60°C. At the extreme temperatures, functional tests and stress tests were performed to observe the response of the system operating in extreme thermal conditions. Similarly, power measurements were collected during stress tests to ensure the power budget was maintained.

On-Dock Integration and Software Update

CASPR was successfully delivered to STP at NASA Johnson Flight Center to be integrated onto the STP-H7 pallet in March of 2021. The purpose of on-dock integration testing was to ensure a fully functional interface between the STP-H7 pallet and the CASPR experiment, which involved verifying the receipt of CASPR telemetry on the DICE interface computer and the successful transmission of a command to CASPR from the DICE. With software testing complete, the mechanical interface to the pallet was validated and preliminary mechanical installation of CASPR was completed. The CASPR experiment on the STP-H7 pallet is shown in Figure 22.

Figure 22: Delivery of STP-H7-CASPR Flight System

In preparation for the launch of CASPR, a final software update will allow for refinements and deployment of additional apps. Many of the apps and scripts included in this update will leverage flight testing aboard SSIVP to ensure resilience. The capability of CASPR to support on-orbit software uploads extends the life of the mission and offers the opportunity to leverage its sophisticated sensor payloads for a growing body of novel research.

CONCLUSIONS

As sensor technologies continue to advance in capability, they introduce big-data challenges due to massive datasets and limitations in downlink. In this paper, we introduced the CASPR experiment for the STP-H7 mission to address these big-data challenges on a SmallSat platform. With autonomous sensor processing as the mission theme, CASPR combines novel sensor technologies with innovative computing techniques on resilient and high-performance flight hardware.

As a university-led, complex experiment, the CASPR mission faced many difficulties beyond the standard challenges that arise when designing for spaceflight. Issues such as budget constraints, resource limitations, design expertise, knowledge transfer, and student graduation introduced unique challenges to the mission development. This paper describes the design principles and methods used for CASPR development, examples of difficulties faced with solutions, and recommendations for university-led mission teams.

The STP-H7-CASPR experiment was successfully delivered to STP in March of 2021, with a scheduled launch on SpaceX-24 in late 2021. Upon arrival to the ISS, this mission will provide a unique sandbox for evaluating and demonstrating novel research in autonomous sensor processing.

Acknowledgments

The research was funded by industry and government members of the NSF SHREC Center, and the National Science Foundation I/UCRC Program under Grant No. CNS-1738783. The authors would like to especially acknowledge several individuals and organizations that supported and contributed to CASPR. Justin Goodwill, Christopher Siak, Aidan Phillips, Eric Sherman, Mitchell Moran, Brad Shea, Nicholas Franconi, and Christopher Wilson for supporting and contributing to the hardware and software development for CASPR. Renesas, Infineon, and the Space Power group at Texas Instruments for their generous part donations essential for this CASPR. The U.S. Naval Research Lab for their feedback on design reviews, part donations, and supporting CASPR with TVAC and vibration testing. NASA Goddard Space Flight Center Code 587 for providing design reviews for SSP. Ryad Benosman for the provision of the Sisley neuromorphic sensor. Jackson Smith, Jennings Leavitt, and Michael Wirthlin of SHREC at Brigham Young University for developing their hybrid CRAM scrubber for CSP and SSP. The U.S. Air Force Research Laboratory: Space Vehicles Directorate for their DoD sponsorship for this experiment. Finally, Space Test Program - Houston team for their continued support and guidance throughout this mission.

References

1. National Academies of Sciences, Engineering, and Medicine, “Testing at the Speed of Light: The State of U.S. Electronic Parts Space Radiation Testing Infrastructure,” The National Academies Press, Washington, DC, 2018.
2. Williams, C., DelPozzo, S., Carapaica, L., and Gerber, D. (2020). Nano/Microsatellite Market Forecast (10th ed.).
3. Roffe, Seth, Theodore Schwarz, Thomas Cook, Noah Perryman, Justin Goodwill, Evan Gretok, Aidan Phillips, Mitchell Moran, Tyler Garrett, and Alan George. "CASPR: Autonomous Sensor Processing Experiment for STP-H7," Proc. of the 34th Annu. AIAA/USU Conf. on Small Satellites, Logan, UT, Aug. 1-6, 2020..
4. Braun, B., Sims, S., McLeroy, J., and Brining, C.B., “Breaking (Space) Barriers for 50 Years: The Past, Present, and Future of the DoD Space Test Program,” Proc. of the 31st Annu. AIAA/USU Conf. on Small Satellites, Logan, UT, Aug. 5-10, 2017.
5. Rudolph, D., Wilson, C., Stewart, J., Gauvin, P., George, A.D., Lam, H., Crum, G., Wirthlin, M., Wilson, A., and A. Stoddard, “CSP: A Multifaceted Hybrid Architecture for Space Computing,” Proc. of the 28th Annu. AIAA/USU Conf. on Small Satellites, Logan, UT, Aug. 2-7, 2014.
6. Wilson, C., Stewart, J., Gauvin, P., MacKinnon, J., Coole, J., Urriste, J., George, A.D., Crum, G., Timmons, E., Beck, J., Flatley, T., Wirthlin M., Wilson, A., and Stoddard A., “CSP Hybrid Space Computing for STP-H5/ISEM on ISS,” Proc. of the 29th Annu. AIAA/USU Conf. on Small Satellites, Logan, UT, Aug. 8-13, 2015.
7. Wilson, C., Mackinnon, J., Gauvin, P., Sabogal, S., George, A.D., Crum, G., and Flatley, T., “μCSP: A diminutive, Hybrid, Space Processor for Smart Modules and CubeSats,” Proc. of the 30th Annu. AIAA/USU Conf. on Small Satellites, Logan, UT, Aug. 6-11, 2016.
8. Sabogal, S., Gauvin, P., Shea, B., Sabogal, D., Gillette, A., Wilson, C., George, A.D., Crum, G., Barchowsky, A., Flatley, T., “SSIVP: Spacecraft Supercomputing Experiment for STP-H6,” Proc. of the 31st Annu. AIAA/USU Conf. on Small Satellites, Logan, UT, Aug. 5-10, 2017.
9. Standardization Group For Embedded Technologies, “Qseven Design Guide,” Version 2.0, March 5, 2015.