Virtual Reality In Space

Why Virtual Reality in space?

Benefits:

- Commodity computing hardware is inexpensive.
- Commodity computing hardware is computationally powerful!

Caveats:

- Real-time streaming of stereoscopic video from space is bandwidth heavy!
- Not feasible to stream real-time from cubesat in real-time *at this time*.

Possible future implementations:

- ExoBrake deployment vision recognition and analysis.
- Vision-based cube satellite (cubesat) stabilization and orientation.
- Real-time artificial intelligence (AI) and augmented reality (AR) assisted missions.
- Moon and Mars autonomous vehicles.

Thinking about a solution…

When presented with a problem or challenge, how will you define your solution?

- What are the physical constraints?
  Will the computer fit inside a cube satellite?

- What are the power constraints?
  What are the power requirements to operate the computer?

- What are the cost constraints?
  Is the computer relatively low-cost and appropriate?

- What are the time constrains?
  When must our phase of the cube satellite project be finished?

- Are there other constraints?
  How will we cool the CPU/GPU in space?

What are Cube Satellites?

Cube satellites also called “cubesats” come in various sizes:

- 10 cm x 10 cm x 10 cm (1U or one-unit)
- 10 cm x 10 cm x 20 cm (2U or two-unit)
- 10 cm x 10 cm x 30 cm (3U or three-unit)
- 10 cm x 10 cm x 60 cm (6U or six-unit)

Source: https://www.nasa.gov/mission_pages/cubesats/overview

Mission: Transmit stereoscopic video from Low Earth Orbit back to earth.

Hardware used:

- Nvidia Jetson TX2
- Intel Edison
- ETTUS wide-band, programmable, radio transceiver
- Leopard Imaging stereoscopic cameras

Software used:

- GNU/Linux (Ubuntu 14.04): The operating system powering the Nvidia Jetson TX2.
- FFMPEG: An open source library to record and compress video from camera input.
- GNU Radio: An open source Software Defined Radio (SDR) for Digital Signal Processing (DSP).

Programming languages used:

- C: To divide huge video into small chunks. Each chunk has an MD5 checksum computed.
- Perl: A simple scripting language to create a series of batch processing commands.

Launch of TechEdSat-8 from Cape Canaveral Air Force Station

Our particular 6U cubesat, TechEdSat-8 (Technology Education Satellite), was onboard a SpaceX Falcon 9 rocket.

This rocket launched from Space Launch Complex 40 (SLC-40) at Cape Canaveral Air Force Station in Florida on December 5, 2018, at 1:16 EST (Eastern Standard Time).

Deployment of TechEdSat-8 from the I.S.S.

Our TechEdSat-8 cubesat was deployed from the International Space Station (I.S.S.) on January 31, 2019 at 8:45am Pacific Standard Time.

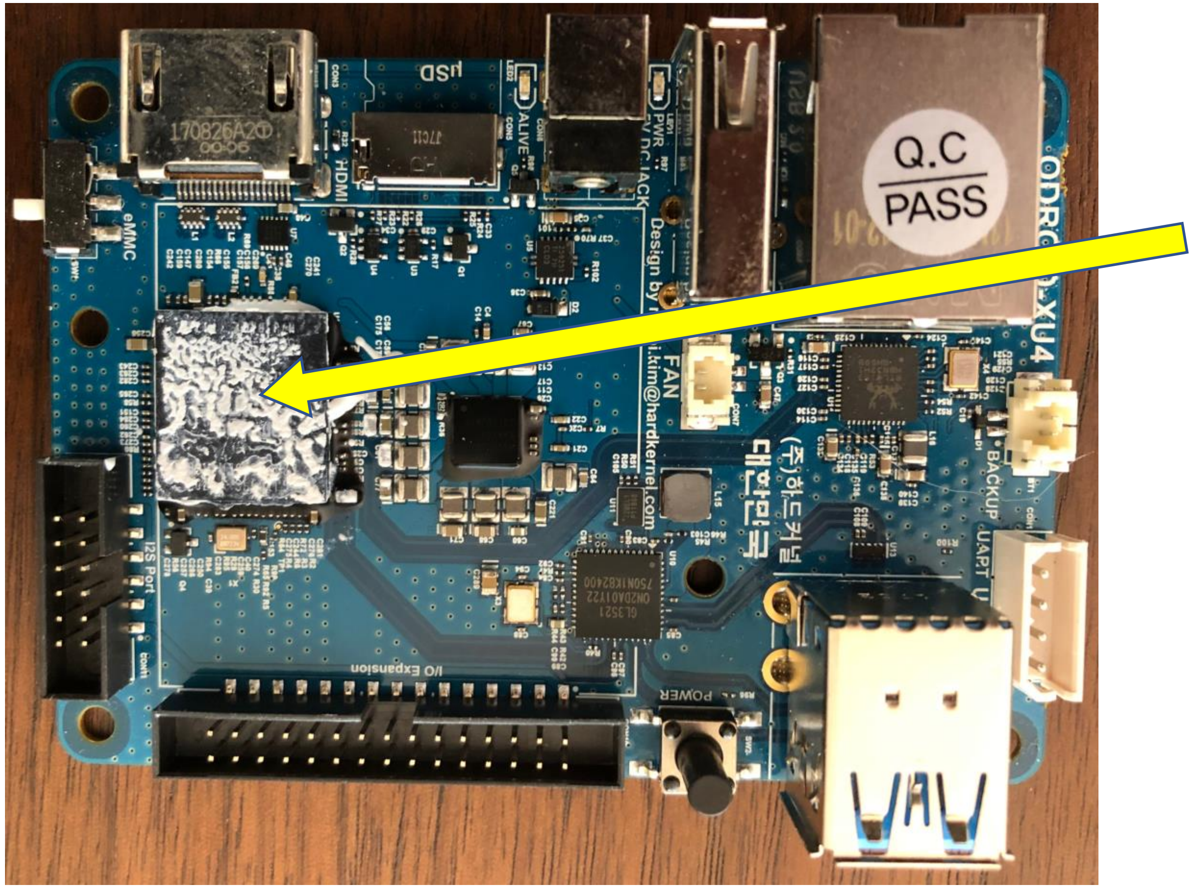
Here’s a video link to the deployment:
https://vimeo.com/345855751

Computing hardware considered for this mission

We considered two different computers for recording, compressing, then transcoding video for this mission:

- Odroid XU4
- Nvidia Jetson TX2

ODROID XU4



Samsung Exynos5422 System-on-a-chip (SoC).

This SoC includes:
Four ARM Cortex-A15 cores (2.0GHz processor)
Eight ARM Cortex-A7 cores (1.4GHz processor)
One graphics processor (GPU): Mali-T628 MP6
Two gigabytes LPDDR3 RAM

Note: Heat sink has been removed. The residue on the SoC is thermal grease.

ODROID XU4: Advantages

Advantages:

- **Form factor**: 82mm x 58mm

  Odroid XU4 form factor is compatible with size constraints of our cubesat.

- **Power consumption** (based on my own stress tests): approximately 18W @ 5V

  Odroid XU4 runs on 5V, there is no need for a DC-to-DC voltage converter.

  CubeSat native power supply runs at 5V.

- **High speed input.**

  Odroid XU4 features USB 3.0 input.

  This enables higher transfer speeds for camera USB input.

- **Low cost**:

  Odroid XU4 costs approximately $62.00 (in Spring 2018 semester – the time of development).

ODROID XU4: Disadvantages

Disadvantages:

- Total onboard RAM (memory) limited to 2GB. Expansion of RAM impossible beyond 2GB.
- No driver support for GPU acceleration.
- USB 3.0 was still bandwidth constrained! We could not handle input data flow rate of stereoscopic camera input at 3840x1080 video resolution (LEFT: 1920 x 1080 and RIGHT: 1920x1080).

ODROID XU4: feasible or not?

We could continue development of ODROID XU4 but we would need to make some compromises:

- Dramatically reduce camera video resolution.
- Use single camera video (no stereo camera input).

A longer term solution (if we had significantly more time for this mission):

- Optimize the Android Operating System designed for the Odroid XU4 to handle large video input files (possible with GPU acceleration for video transcoding and compression).
- Modify FFMPEG to enable GPU-accelerated video compression and transcoding on the MALI GPU.

Nvidia Jetson TX2: Advantages

Advantages:

- **Form factor**: 87 mm x 50 mm

  Nvidia Jetson TX2 is compatible with size constraints of our cubesat.

- **On-board RAM: 8 GB**

  Eight gigabytes of ram gives us plenty of buffer space to record stereoscopic video at 3840x1080 (1920x1080 for the left frame and 1920x1080 for the right frame).

- **High speed input.**

  Nvidia Jetson TX2 features USB 3.0 input and was designed for stereoscopic camera input.

- **Low cost**:
  Nvidia Jetson TX2 costs approximately $500 (in 2018 at time of our mission).

Nvidia Jetson TX2: Disadvantages

Disadvantages

- **Power consumption** (based on my own stress tests): approximately 18W @ 9V

  Nvidia Jetson TX2 runs minimally on 9V. We will need a DC-to-DC voltage converter to boost the 5V main power to 9V for the Nvidia TX2. Power consumption under full-load was similar to Odroid XU4.

- **GPU acceleration**

  At the time of our mission launch, Nvidia did not fully support GPU-accelerated video transcoding and compression. It was a work-in-progress. Therefore, we were unable to take advantage of this hardware accelerated feature at the time of our launch.

Nvidia Jetson TX2: Feasible or not?

We continued to use Nvidia Jetson TX2 for several reasons:

- Nvidia had GPU-accelerated drivers in development. We could take advantage of hardware acceleration on a future mission.
- Nvidia Jetson TX2 featured the most RAM (8 GB) of any small form factor SoC (system-on-a-chip) that was both readily available and inexpensive.

The compromise solution:

- Buffering raw video directly to a solid state device (SSD).
- After entire video is captured and stored on SSD, transcode and compress the raw video into a smaller file using ONLY the built-in ARM processor that Nvidia Jetson TX2 uses. It is slow ***but effective***.

Other factors that affected our decision:

- We also chose Nvidia Jetson TX2 because it was effectively a supercomputer in a small form factor. We knew the accelerated hardware built into the GPU could be used for all sorts of scientific experiments such as spectroscopy to determine the chemical composition of the atmosphere in Low Earth Orbit.

Programming the Nvidia Jetson TX2: The Batch Process (in Perl)

Capture thirty seconds of raw, uncompressed video at resolution 3840x1080.

↓

Transcode raw video using H.264 codec while maintaining original resolution 3840x1080.

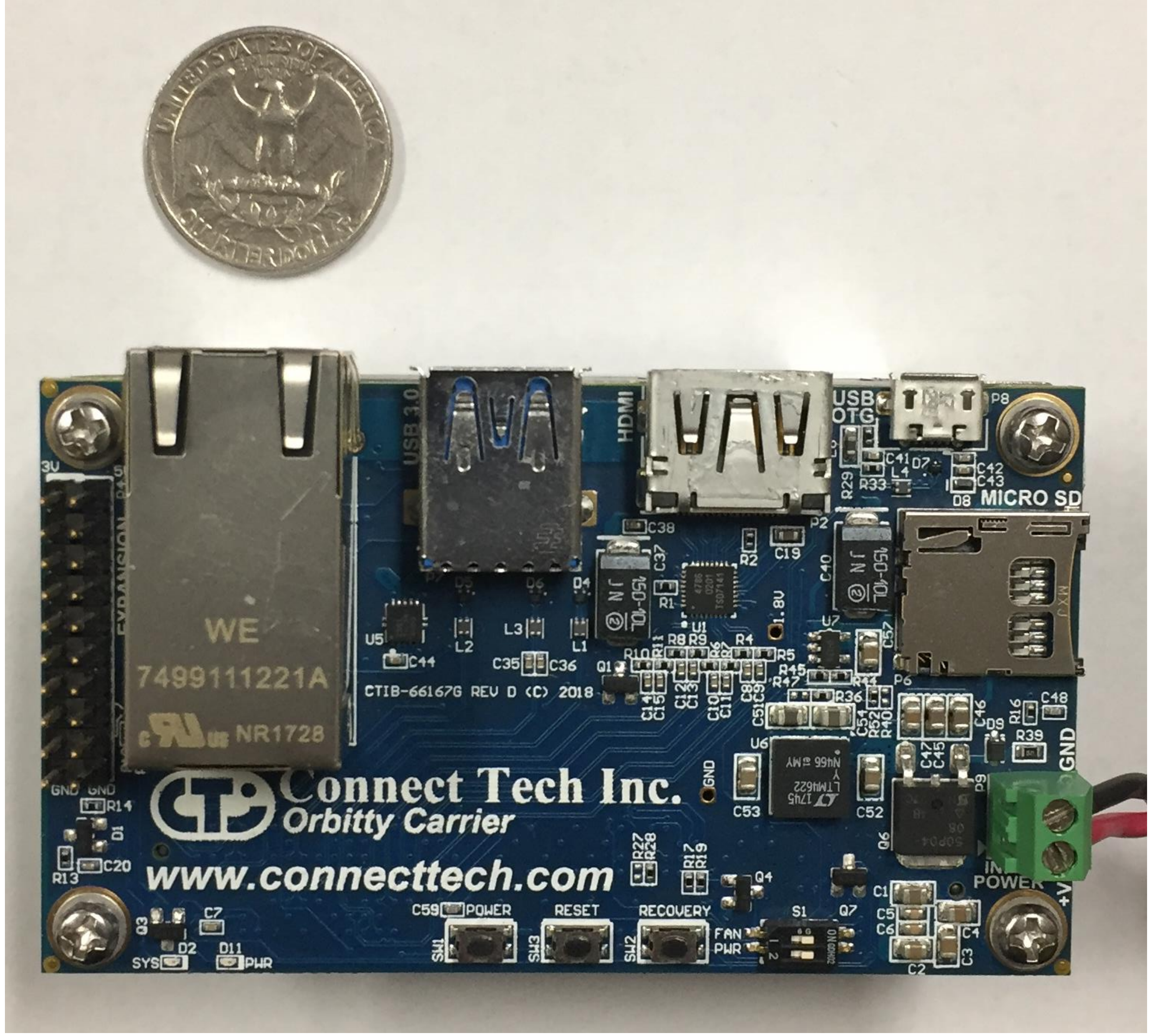
↓

Transfer H.264 video file to an Intel Edison board (on same CubeSat as Nvidia TX2) via wired UART (Universal Asynchronous Receive Transmit) and wireless 802.11n Wi-Fi.

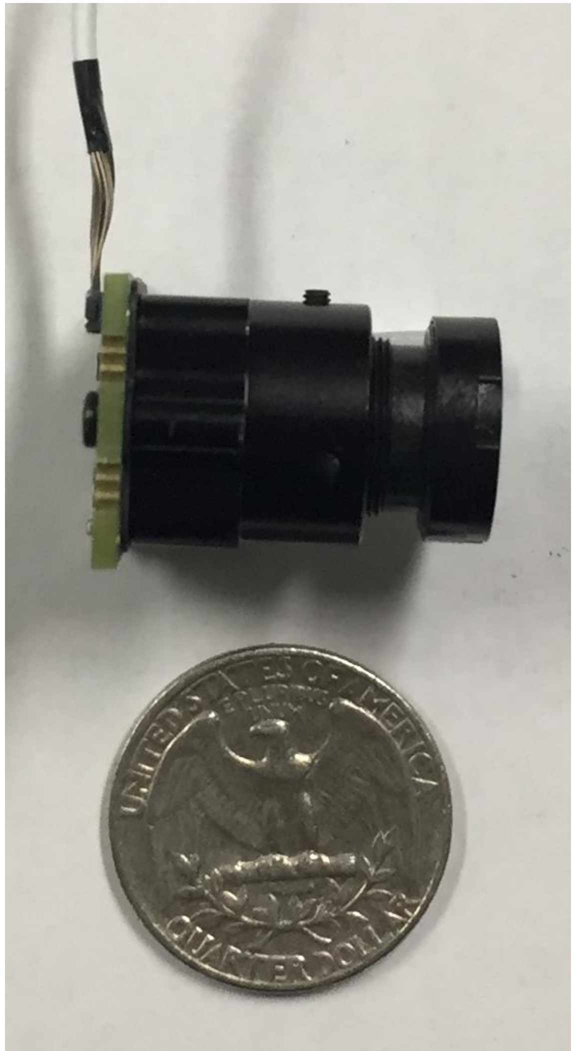
↓

Intel Edison board packetizes H.264 video file into small chunks via SMS (satellite phone) and via GNU/Radio with an ETTUS S-band radio transceiver.

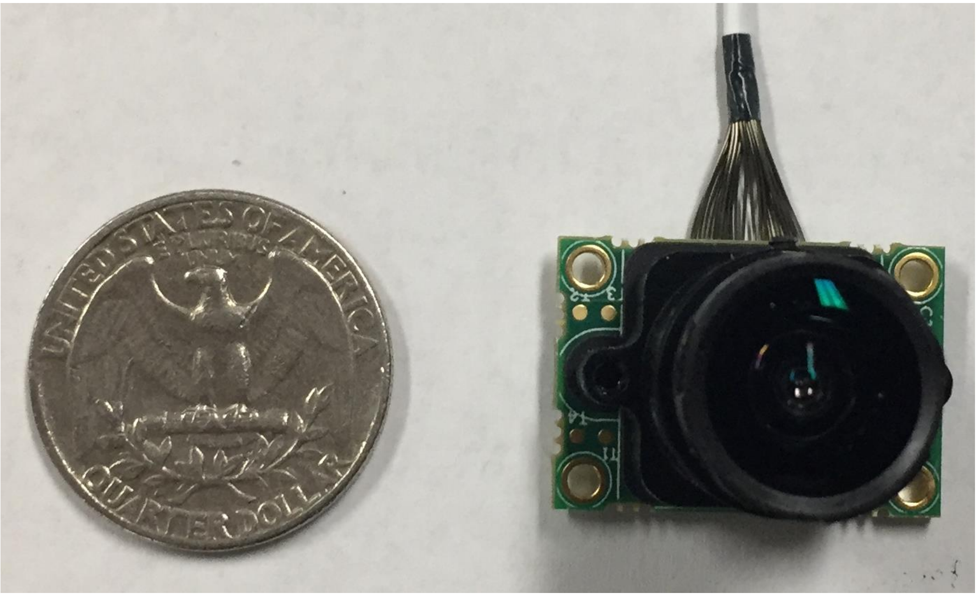
TechEdSat-8: Carrier Board Hardware



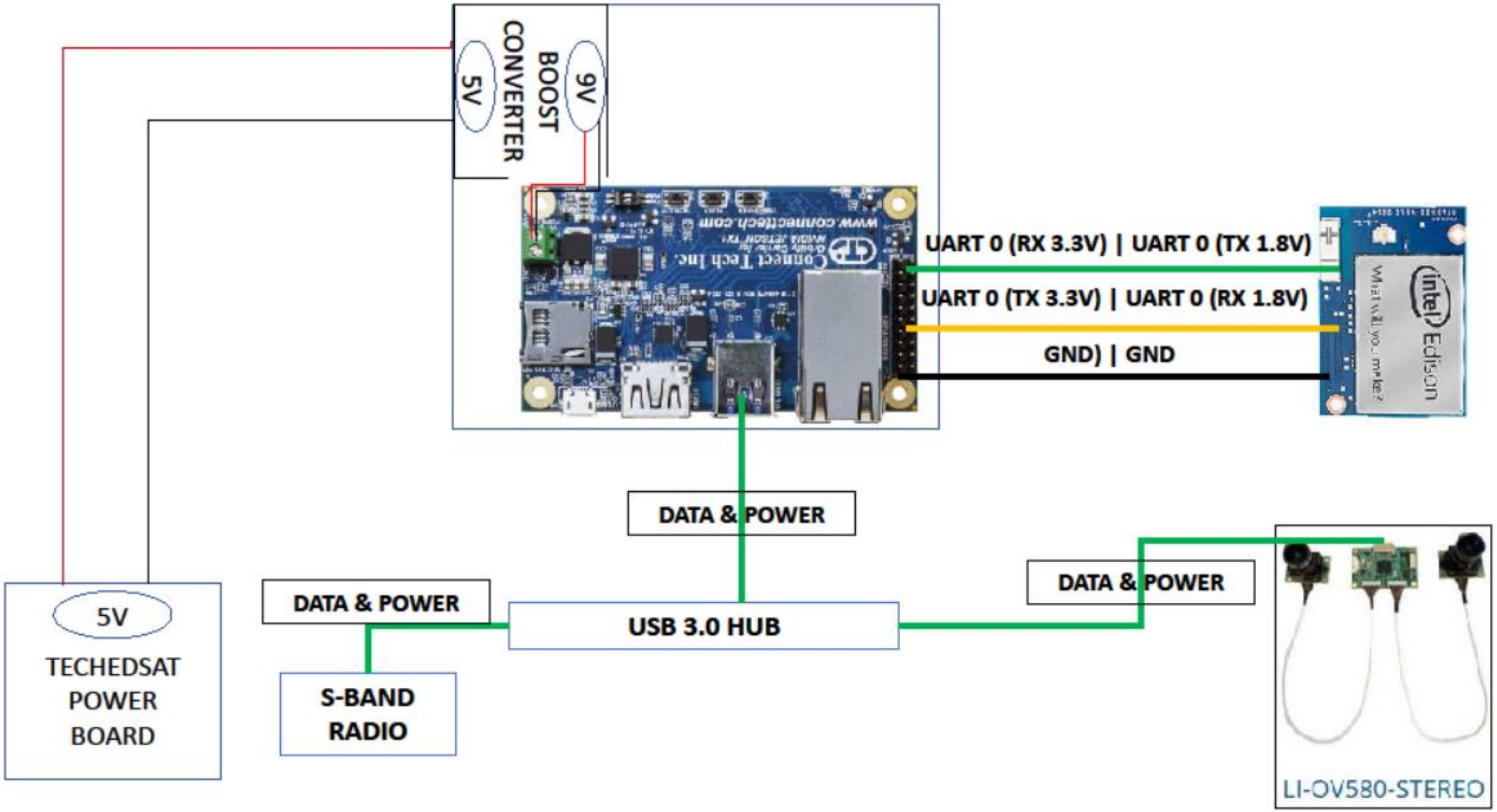
TechEdSat-8: Stereoscopic Cameras



Leopard Imaging stereoscopic cameras (model number LI-OV580-STEREO).



TechEdSat-8: Wiring Diagram (Simplified)



TechEdSat-8: Status

Once jettisoned from I.S.S. what happened to TechEdSat-8?

- Diagnostic messages sent from TechEdSat-8 in Low Earth Orbit (LEO) indicated the cubesat successfully recorded and transcoded 30 seconds of stereoscopic video on the Nvidia TX2 immediately after it was deployed from the International Space Station.

- Sadly, the video transmission was never sent due to a failure in the battery charging system.

What went wrong?

- The on-board batteries failed to recharge from the onboard photovoltaic cells. Consequently, there was not enough battery power to operate TechEdSat-8's radio transceiver to send back the recorded video.

- TechEdSat-8 burned up in earth's atmosphere approximately three weeks after deployment.

TechEdSat-8: Success and Failure

What made this mission a success?

- To the best of our knowledge, this was the first time an Nvidia Jetson TX2 has ever been launched into space.
- We proved that Nvidia Jetson TX2 is a viable compute device for space.
- Using common readily-available hardware (i.e. Nvidia Jetson TX2) meant we could put a supercomputer capable of multi-teraflop performance into cube satellite using relatively low power (under 20 watts).

Future missions:

- The minor setback of a failed battery recharge has not stopped us from continuing to explore Virtual Reality in space.

The takeaway

- Utilize common technologies in non-traditional ways outside their intended environment or scope.
- Remain optimistic and persevere even when failures inevitably occur.

THANK YOU!

Email:

Robert.Bruce@sjsu.edu

Lab 246:

http://www.sjsuvrlab.org/

This presentation is archived at:

https://www.robertjamesbruce.com/pdf/vr_in_space.pdf