Author: Jako Aimsalu, Robotics and Computer engineering MSc
Supervisor: Quazi Saimoon Islam, MSc

Tartu 2026

Development of a Nvidia Jetson platform for the KuupKulgur prototype

Introduction

The KuupKulgur student organization is building a Lunar rover prototype complete with an autonomous navigation system for the Lunar surface. For this, visual and other sensor information is used and processed by the on-board computer.

A popular and powerful commercially available NVIDIA Jetson developer kit is used as a processor, featuring AI-accelerated edge computing. NVIDIA is becoming more active in space computing, and the KuupKulgur project plans to utilize their products in the future. However, to ensure continuous development, the rover requires an intermediate system – specifically designed Jetson hardware.

The Goal

The new carrier board system for the Jetson needs to fit inside the rover's body (Figure 1). The system was decided to be split into two parts: The Main Carrier board PCB implementing essentials for the rover, and Peripheral Debug Board PCB to extend the remaining available interfaces.

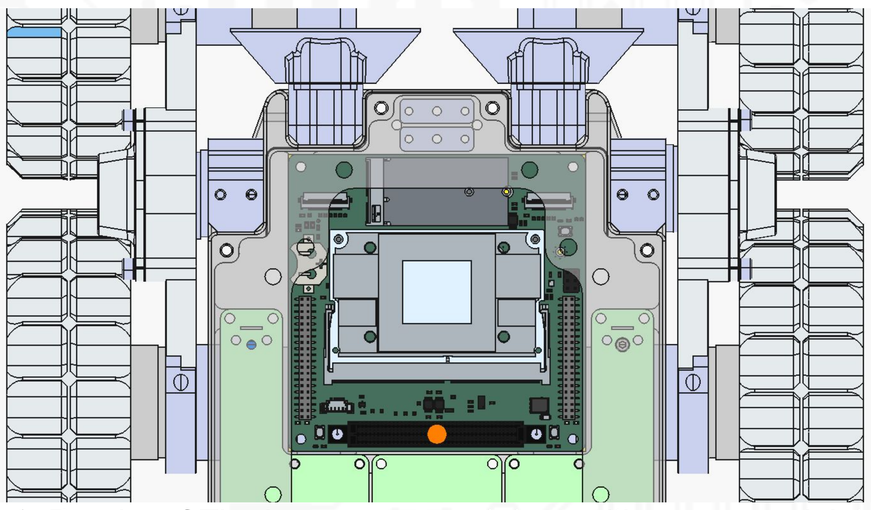

Figure 1. Render of The Lunar rover prototype with a proposed Jetson carrier board inside.

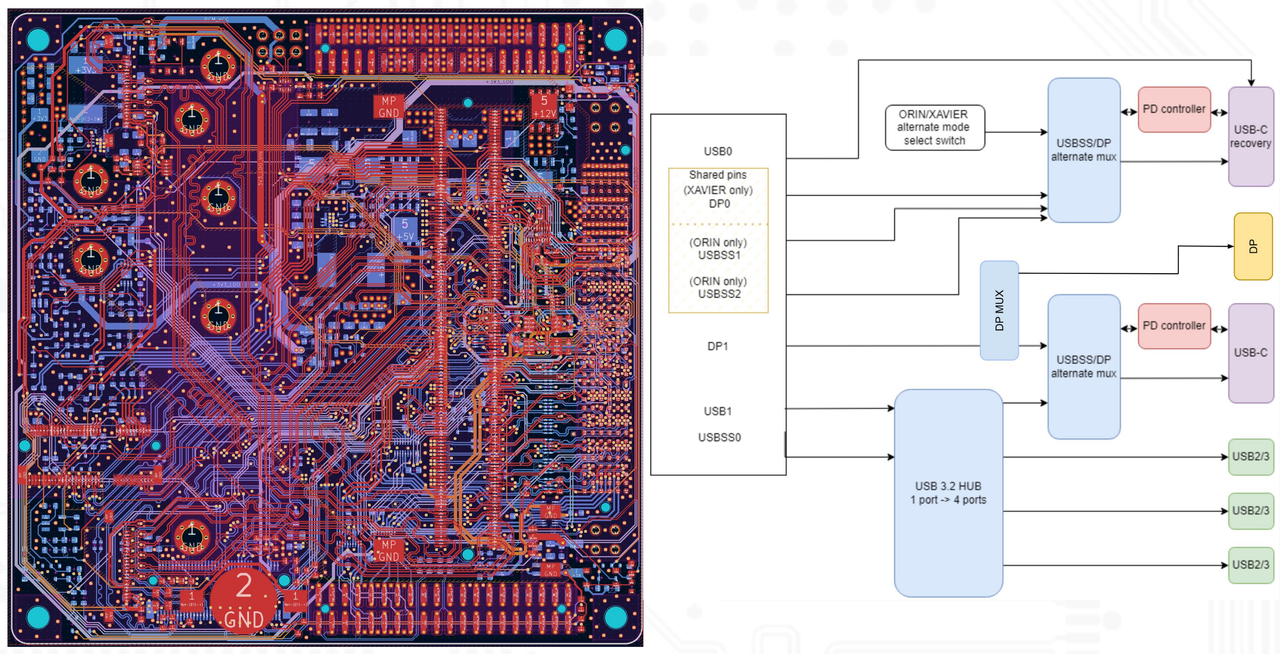

Figure 2. Routed traces on 8-layer board in KiCad and USB/DP interface layout

System Architecture & Interfaces

The new carrier board system was intended to be as similar as possible to the original developer kit to keep software level transition adaption minimal.

Key carrier board interfaces:

- NVMe SSD and Wi-Fi/BT over PCIe4.0
- Cameras over MIPI-CSI-2
- DisplayPort 1.4a HBR3 and USB3.2
- SPI, I2C, I2S, CAN, GPIOs

Additional implemented features and approaches:

- Hot-swappable Peripheral Debug Board
- ATtiny9-based Power Control Module (PCM)
- Multi-generation support for both Orin and Xavier models
- Full-featured USB-Cs with Power Delivery & DisplayPort support (Figure 2)
- Power with protection from multiple sources (Figure 4)

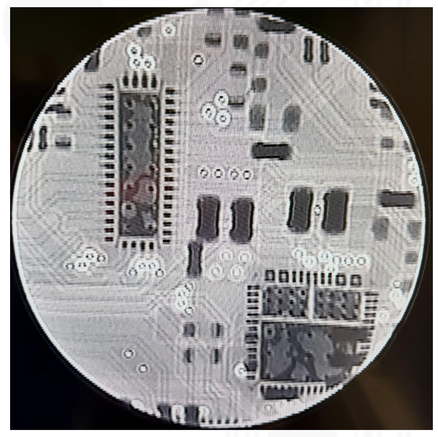

Figure 3. Solder X-ray inspection

Development Strategy & Hardware Realization

NVIDIA provides concept PCB references, recommendations, and guidelines for engineers creating Jetson products. This 8-layer PCB was designed using the open-source electronic design automation (EDA) software, KiCad (Figure 2). Compared to "simple" PCB designs, a much stricter adherence to design rules was required to maintain signal integrity across high-speed interfaces for optimal performance. While AI can assist in explaining advanced concepts and considerations, it was not used for this design because it lacks critical, contextual systems-engineering knowledge.

Even with prior experience designing simpler boards, a massive amount of new knowledge had to be gathered to simulate signal energy transfers and component behaviors. Major issues were frequently discovered during the process, requiring the design to be started all over again. At least two complete redesigns were executed without ordering intermediate test boards, as low-volume multi-layer PCB manufacturing is very expensive. Furthermore, soldering was done entirely by hand following Tartu Observatory's strict lead-free requirements. While professional assembly services were considered, the complexity of the board meant it could only be partially machined, which would significantly increase costs without fully automating the process.

Conclusion & Next Steps

Two custom 8-layer PCBs (96×96 mm and 155.6×74.6 mm) were successfully engineered and assembled at Tartu Observatory as a part of the thesis. Rigorous hardware-level testing validations were conducted (Figure 3); a few major design flaws were identified and successfully isolated and rectified thereafter. The NVIDIA Jetson operating system to successfully boot on the custom hardware was enabled, validating system stability and clearing the path for software integration. An over 2-year development cycle was completed, in which a total of 900+ components were precision-placed and hand-soldered, with over 20 meters of high-density trace lines routed. Following successful software validation and further rover prototype development, the architecture will be ready for further iterations and potential series production of the system.

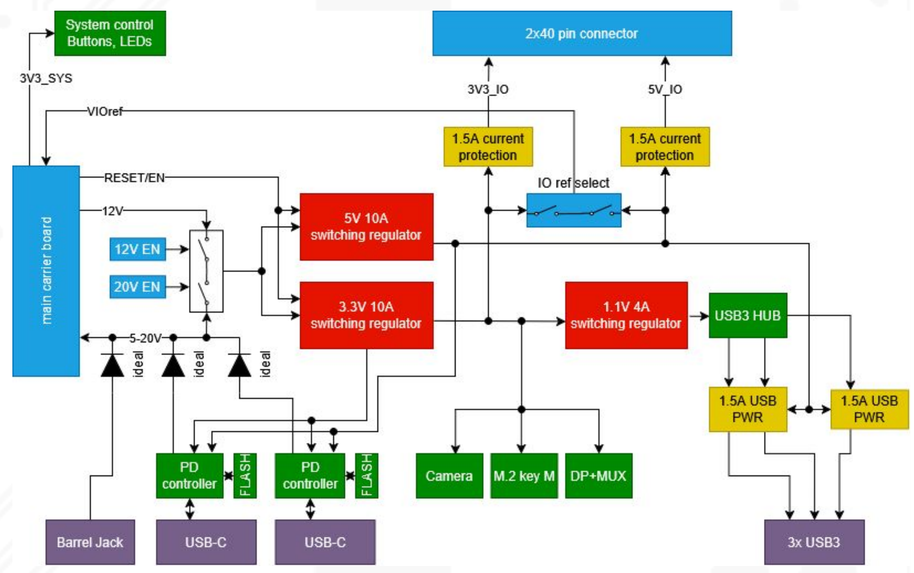

Figure 4. Peripheral Debug Board power distribution diagram

https://kodu.ut.ee/~jakoai/maka/index.html https://kuupkulgur.space