

Custom Computational Payloads for DJI Matrice 350

Unlocking Advanced Drone Imaging Capabilities for Geo-mapping Research



Custom Payload for DJI Matrice 350

This is a guide for integrating a computational payload, centered around a NVIDIA Jetson Nano, into the DJI Matrice 350. Our goal is to intercept the video downlink from the Matrice, conduct localized processing and then transmit to the ground station.

Seamless Integration

The payload fits within the drone's case, mimicking a standard DJI payload for easy transport and deployment.

Customizable

Utilizes 3D-printed components and affordable NVIDIA Jetson Nano, making it customizable.

Full Drone Interactivity

Leverages the drone's power and telemetry, controls drone behavior and other payloads, and accesses sensor and camera feeds.

Real-time Video Processing

Processes video and streams it to the operator's controller in real time, enhancing situational awareness.

DJI SDKs for Custom Development

DJI offers three Software Development Kits (SDKs) to interact with their drones, each serving different integration needs.

1

Mobile SDK (MSDK)

Develops Android/iOS applications for drone control and data viewing via the controller or connected device.

2

Onboard SDK (OSDK)

Enables processing directly on the drone, enhancing speed and reliability by reducing reliance on the controller's data link.

3

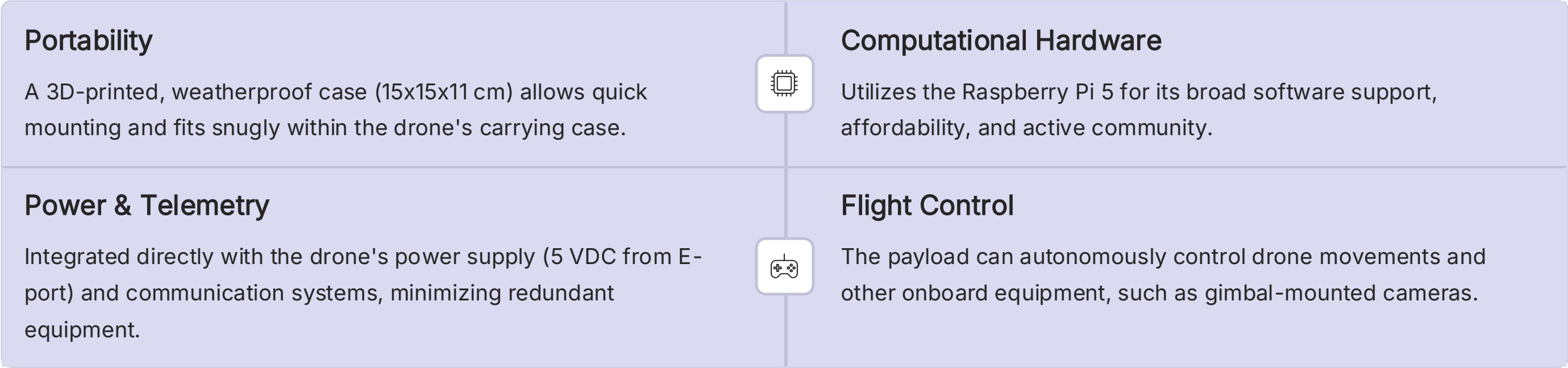
Payload SDK (PSDK)

The newest and most relevant for custom payloads, allowing deep integration into the drone's hardware and software systems. Targets onboard computers like Manifold 2 and NVIDIA Jetson series.

While the PSDK provides extensive capabilities, its complexities and proprietary nature present a significant barrier, often requiring direct communication with DJI support to overcome undocumented nuances.

Building the Payload: Physical and Computational Setup

Our custom payload adheres to strict requirements for portability, ease of use, and full integration with the Matrice 350.



PSDK Port Functionalities Overview

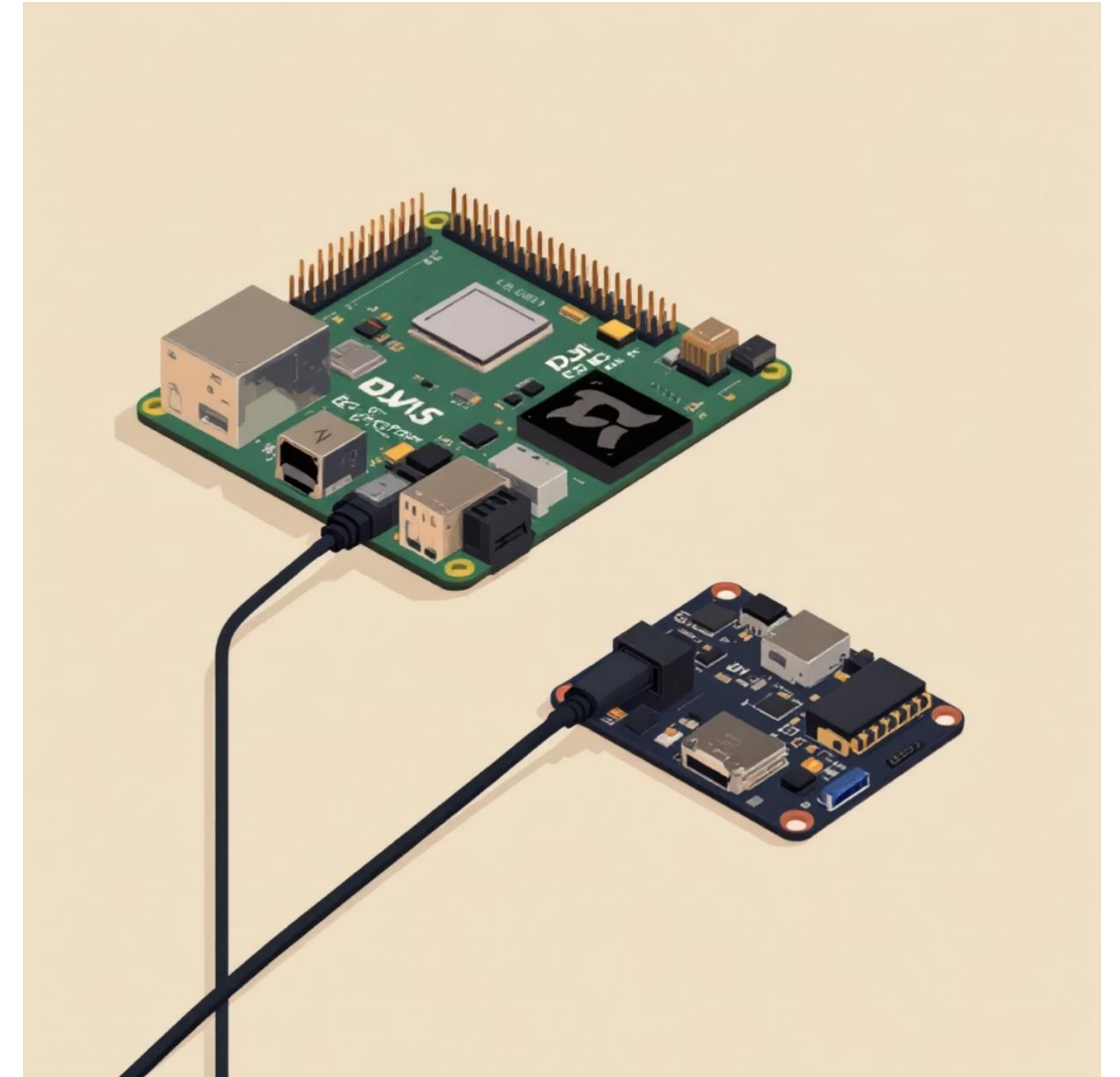
E-port	✓	✓	✓	✓	×
SkyPort V2	✓	×	✓	×	✓

To achieve full functionality (control, sensor access, and real-time streaming), our payload leverages both the E-port and SkyPort V2, requiring two separate PSDK applications running in parallel on the Raspberry Pi 5.

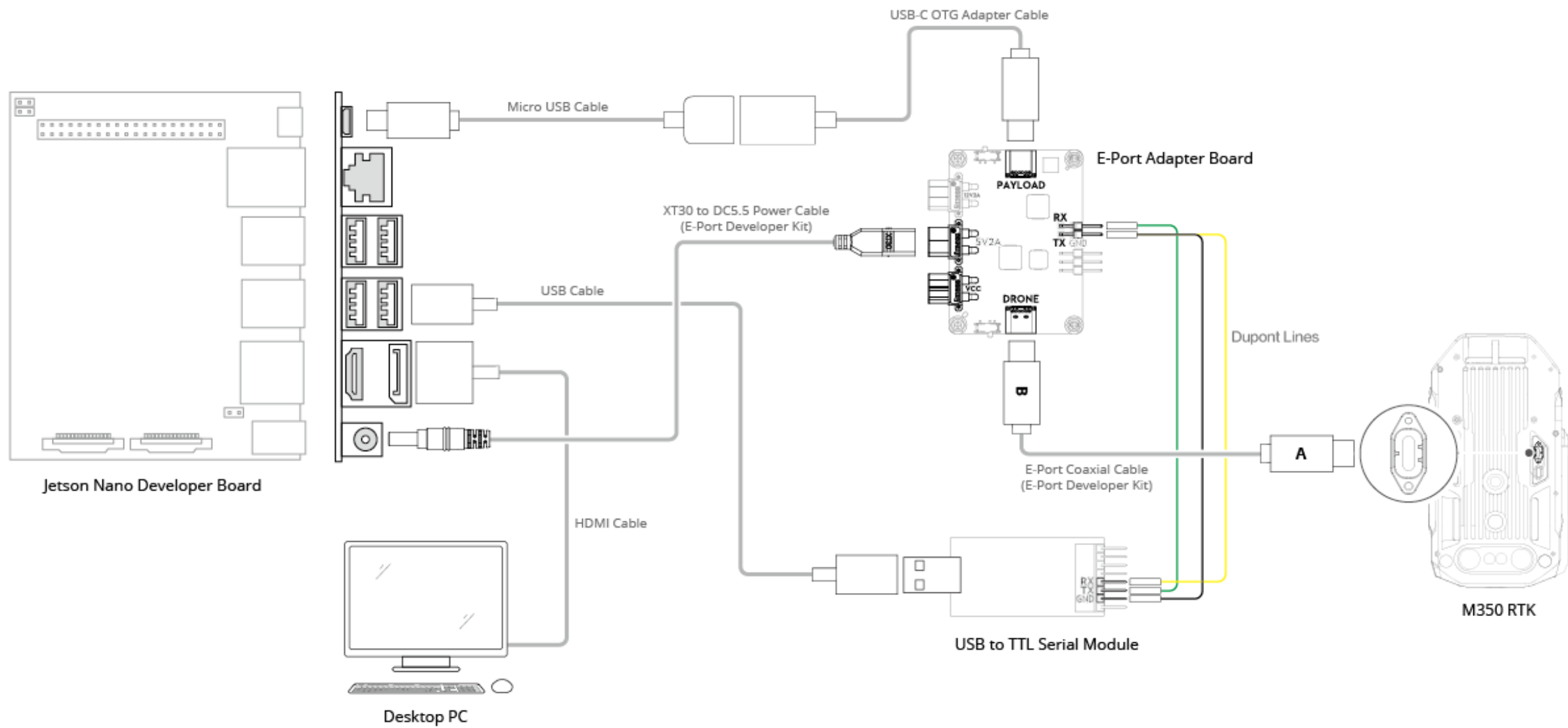
E-Port Application: Unlocking Drone Sensor & Camera Feeds

The E-port application is critical for enabling the Raspberry Pi to interact deeply with the drone's internal systems. This setup facilitates access to vital data and control functionalities.

- **Hardware Connection:** The DJI E-port development board connects to the Raspberry Pi 5 via UART for low-bandwidth communication and USB-C for high-bandwidth data, including video streams.
- **Control & Data Subscription:** With proper UART configuration, the PSDK application gains the ability to control drone flight and other payloads, and subscribe to various data topics (e.g., IMU, GPS, altimeter).
- **Video Retrieval:** Two USB bulk virtual devices are configured on the Nano, allowing the drone to directly write RGB and stereo video files using the FunctionFS protocol over the USB-C port. This is a complex setup, adapted from limited DJI documentation for different drone and Nano models.
- **USB Gadget Configuration:** The Nano is declared as a multifunction composite USB gadget, ensuring unambiguous identification by the drone and establishing endpoints for data management.



The E-port application provides critical access to the drone's internal systems.



SkyPort Application: Real-time Video Streaming to Controller

The SkyPort application enables real-time streaming of the NVIDIA desktop to the operator's controller, crucial for in-flight visualization and interaction.

Desktop Streaming

By switching to the X11 desktop environment, ffmpeg can efficiently capture and stream the Pi's desktop as raw frames, encoded into H.264 format.

Optimized Performance

To maintain CPU load below 10% while ensuring adequate video quality, the desktop resolution is set to 640x480 pixels at 24 FPS, with ffmpeg encoding at 1.2 Mbps using the "ultrafast" preset.

Pilot Interaction

Basic interaction with the Pi desktop is achieved through `xdotool`, mapping pilot clicks on the controller's screen to corresponding actions on the Pi.

This independent application runs in parallel with the E-port application, ensuring that any visualization or output from the Pi is visible to the pilot, significantly enhancing operational capabilities.

Operation

Developing custom payloads for commercial drones often involves navigating undocumented features and unexpected behaviors. We've identified and addressed several critical quirks:

- 1** **Boot-up Sequencing**
Desktop streaming applications require a minimum 180-second uptime before stable operation; the SkyPort application must start before the E-port to prevent data channel interference.
- 2** **Interface Decoupling**
Critical to use distinct interfaces for each application (Pi's onboard UART and USB bulk for E-port; USB-to-UART converter and Ethernet for SkyPort) to avoid crashes.
- 3** **Stereo Video Retrieval**
Stereo video streams are only retrievable via USB bulk devices, requiring adaptation of existing tutorials for the Matrice 350 and NVIDIA Jetson Nano.
- 4** **Firmware Update Complexities**
SkyPort V2 connector firmware updates can only be performed on a Matrice 300 with an older version of DJI Assistant software, highlighting documentation gaps and support challenges.
- 5** **System Interdependencies**
Proprietary microprocessors across the drone ecosystem mean that errors can propagate unexpectedly, requiring careful troubleshooting and system reboots.

Our work provides a blueprint and configuration scripts for researchers to integrate custom computational payloads into the DJI Matrice 350, opening doors for advanced drone applications in various domains.

Important Links

<https://developer.dji.com/doc/payload-sdk-tutorial/en/payload-quick-start/quick-guide/jetson-nano.html>

<https://github.com/DJI-Mobile-SDK-Tutorials/iOS-SimulatorDemo>

<https://arxiv.org/html/2405.06176v1>

<https://developer.dji.com/mobile-sdk/documentation/ios-tutorials/SimulatorDemo.html>