Design Lab Jetson Nano Quickstart Guide

May 29, 2024

# Introduction

Capstone teams have run into issues when a Raspberry Pi is no longer powerful enough for the computational needs of the team, such as for running OpenCV. Therefore, the Design Lab has NVIDIA Jetson Nanos that can better fulfill the needs of the teams. Jetson Nanos are also single-board computers like Raspberry Pis, but they have more powerful CPUs and GPUs that can run conventional workloads and some AI workflows.

In this quickstart guide, you will be setting up the Jetson for first boot, as well as working through the NVIDIA “Hello AI World” tutorial. Importantly, use this guide as a companion to “Hello AI World.” This guide would offer tips to help you get set up as quickly as possible, avoiding common pitfalls that aren’t mentioned in the official documentation.

# References

The most important two links to use as references are:

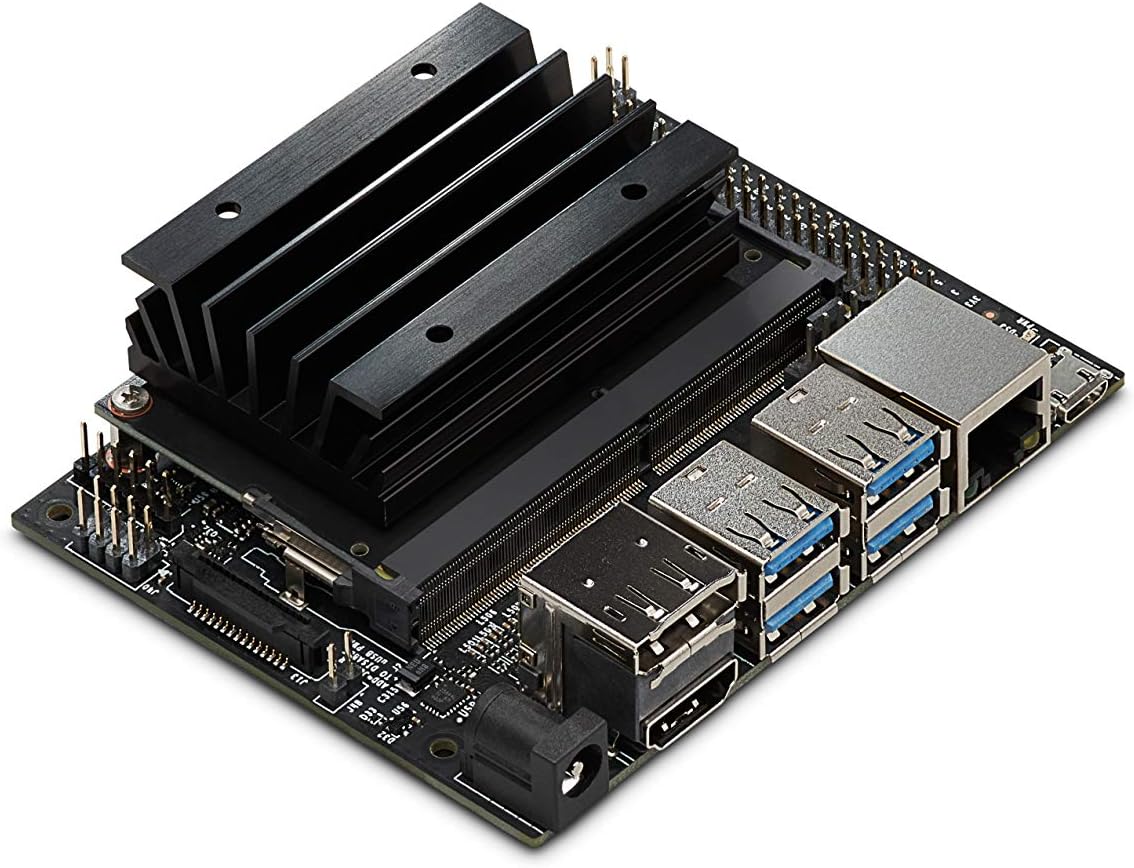
1. <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit>
2. <https://github.com/dusty-nv/jetson-inference>

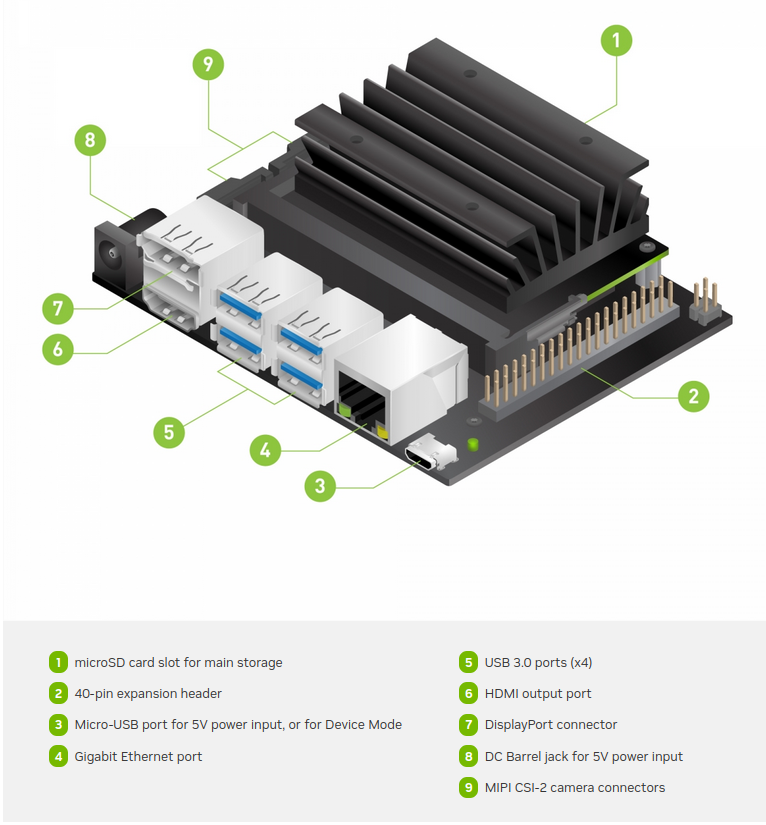
Everything outlined in this guide can be done just by following these two links. You may run into a headache or two, but it’s absolutely doable. Use the first link to set up the hardware side of the Jetson, and use the second link to start using neural networks to process images, such as classification and object detection.

Further references can be found in the Jetson Nano Developer Kit User Guide, linked here: <https://developer.nvidia.com/embedded/dlc/Jetson_Nano_Developer_Kit_User_Guide>

# Things You’ll Need

1. A Jetson Nano Developer Kit

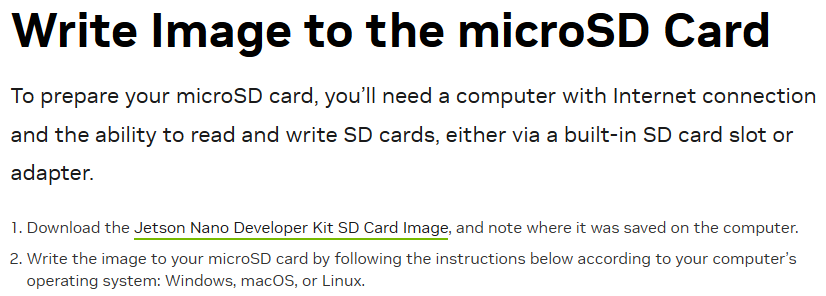


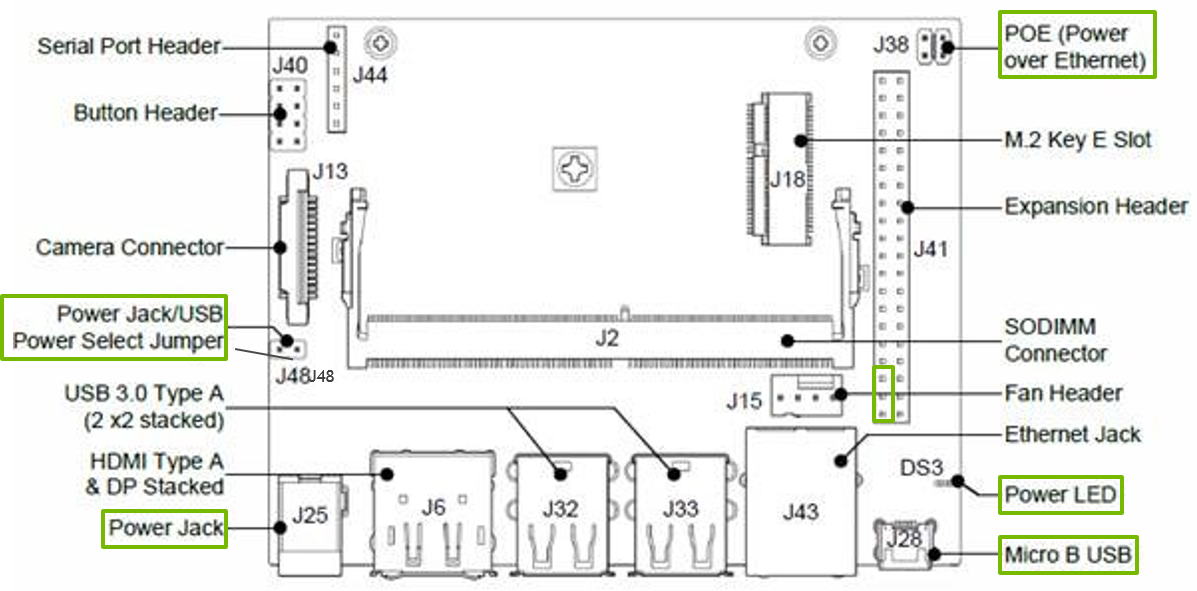


1. microSD card with at least 32GB
2. A USB microSD card adapter
3. A 5V/2A micro-USB power supply OR a 5V/2A or above DC barrel jack supply
   1. See “First Boot” step 4 for more information on the choice between micro-USB and DC barrel jack.
4. A monitor connected through HDMI or DisplayPort
5. A keyboard
6. A mouse
7. (Optional but recommended) A Wi-Fi dongle or an Ethernet cable
   1. NOTE: Jetson Nanos do not come with onboard Wi-Fi, unlike the later models of Raspberry Pis
8. (Optional) A webcam
   1. This is used to test the image recognition capabilities of the Jetson Nano

# First Boot

1. Go to <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#write>, and click on the link “Jetson Nano Developer Kit SD Card Image” under “1.” to download the image to be flashed onto a microSD card.



1. Prepare a microSD card, but do not insert it into the computer yet.
   1. NOTE: If the provided Jetson already comes with a microSD card and you wish to flash it, press it down in order to release it. Do not pull!
2. Follow the instructions on <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#write> in order to flash the microSD card. This website has instructions for Windows, Mac, and Linux. On Linux and Mac, it has instructions for using either Balena Etcher, a GUI program, or the command line. The Mac command-line requires the executables diskutil, fgrep, unzip and dd. The Linux command-line requires dmesg, tail, awk, unzip, and dd. All methods would work well.
3. Follow the instructions under “Initial Setup with Display Attached” on <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#setup> .
   1. IMPORTANT: There is a power LED next to the micro-USB port on the Jetson Nano. After plugging in the micro-USB power supply, observe if the LED has lit up. If it hasn’t, there are at least two possibilities:
      1. The power supply isn’t powerful enough. Double check to make sure that it can support 5V/2A, or try a different power supply.
      2. The Jetson has been preconfigured to be powered through the DC barrel jack instead. This can be verified by checking the “Power Select Jumper” indicated in the following image:

If the two pins are connected, then the Jetson can only be powered through the DC barrel jack. Either disconnect the jumper connecting the two pins, or find a compatible barrel jack supply. Note that while officially the Jetson prefers a 5V/4A barrel jack supply, 5V/2A has been known to work.

1. After this step, congratulations! You have the Jetson set up, and are ready to do “Hello AI World.”

# (Optional) Jetson JetPack version / L4T Version

You may encounter cases where you’re asked for the JetPack version of your Jetson Nano, but you’re not sure what it is. Instructions to do so are as follows:

1. Open the terminal by clicking the Ubuntu icon on the top left and searching for “terminal”, and run cat /etc/nv\_tegra\_release in the terminal.
2. Take the output and find the L4T version.
   1. For example, if your output is something like # R32 (release), REVISION: 7.1, GCID: 29818004, BOARD: t210ref, EABI: aarch64, DATE: Sat Feb 19 17:05:08 UTC 2022, then your L4T version is the combined values of the number after the first capital R and the revision number. For me, my JetPack version is 32.7.1, from “R32 (release), REVISION: 7.1”.
3. After finding your L4T version, go to <https://developer.nvidia.com/embedded/jetpack-archive> and find the corresponding JetPack version number according to your L4T version. For example, since I got 32.7.1, I search for it on the page, and it corresponds to JetPack version 4.6.1.

# Hello AI World

1. From here, go to <https://github.com/dusty-nv/jetson-inference/blob/master/docs/aux-docker.md> and follow the instructions there to complete the Hello AI World tutorial.
   1. The L4T version and JetPack version number here would be crucial to help you get the correct Docker container. For example, for the L4T version number I used above, I went to <https://hub.docker.com/r/dustynv/jetson-inference/tags>, found the correct Docker container, and ran the corresponding pull command in the terminal: docker pull dustynv/jetson-inference:r32.7.1
   2. If you run into issues running the command, run it with sudo: sudo docker pull dustynv/jetson-inference:r32.7.1
2. After this, continue following the guide at [https://github.com/dusty-nv/jetson-inference/blob/master/docs/aux-docker.md#launching-the-containe](https://github.com/dusty-nv/jetson-inference/blob/master/docs/aux-docker.md#launching-the-container)r all the way till the end, or until you are satisfied.
   1. For the purposes of time, the Design Lab recommends trying out the following four functionalities that Hello AI World provides:
      1. Image Classification, found here: <https://github.com/dusty-nv/jetson-inference/blob/master/docs/imagenet-console-2.md>
      2. Object Detection, found here: <https://github.com/dusty-nv/jetson-inference/blob/master/docs/detectnet-console-2.md>
      3. Semantic Segmentation (parsing an image into its different parts), found here: <https://github.com/dusty-nv/jetson-inference/blob/master/docs/segnet-console-2.md>
      4. Monocular Depth (estimating depth with just one camera): https://github.com/dusty-nv/jetson-inference/blob/master/docs/depthnet.md