



NV300-AMB

Military Jetson AGX Orin IP65 Rugged Computer



User's Manual

Revision Date: Nov. 13. 2025



Safety Information

Electrical safety

- To prevent electrical shock hazard, disconnect the power cable from the electrical outlet before relocating the system.
- When adding or removing devices to or from the system, ensure that the power cables for the devices are unplugged before the signal cables are connected. If possible, disconnect all power cables from the existing system before you add a device.
- Before connecting or removing signal cables from the motherboard, ensure that all power cables are unplugged.
- Seek professional assistance before using an adapter or extension cord. These devices could interrupt the grounding circuit.
- Make sure that your power supply is set to the correct voltage in your area.
- If you are not sure about the voltage of the electrical outlet you are using, contact your local power company.
- If the power supply is broken, do not try to fix it by yourself. Contact a qualified service technician or your local distributor.

Operation safety

- Before installing the motherboard and adding devices on it, carefully read all the manuals that came with the package.
- Before using the product, make sure all cables are correctly connected and the power cables are not damaged. If you detect any damage, contact your dealer immediately.
- To avoid short circuits, keep paper clips, screws, and staples away from connectors, slots, sockets and circuitry.
- Avoid dust, humidity, and temperature extremes. Do not place the product in any area where it may become wet.
- Place the product on a stable surface.
- If you encounter any technical problems with the product, contact your local distributor

Statement

- All rights reserved. No part of this publication may be reproduced in any form or by any means, without prior written permission from the publisher.
- All trademarks are the properties of the respective owners.
- All product specifications are subject to change without prior notice



Table of Contents

Safety Information	1
Electrical safety	1
Operation safety	1
Statement	1
Specifications	3
Mother Board Block Diagram	5
System Block Diagram	6
Board Visuals	6
System Front/Rear IO Visuals	7
List of Connectors & Buttons	8
Dimension Drawing	8
Definition of each I/O	9
Ordering Information	11
Software Information	12
Software Configuration	12
System Recovery	12
Host PC	12
JetPack-6.x Installation for NV300-AMB.	12
Including the Kernel Files in Jetson OS Image	12
Jetson OS Installation	21
Jetson SDK Components Installation	26
Change Root File System to M.2 SSD Directly	29
Formatting the M.2 SSD Storage	29
Copying the Root File System	35



Specifications

SYSTEM

AI Performance	275 TOPS
CPU module	Ampere GPU + Arm Cortex-A78AE CPU + 64GB LPDDR5 + 64GB eMMC 5.1
GPU	NVIDIA Ampere architecture with 2 GPX/8TPC, 2048 NVIDIA® CUDA® cores and 64 Tensor Cores, max freq. 1.3GHz
CPU	12-core Arm® Cortex®-A78AE v8.2 64-bit CPU, 3MB L2 + 6MB L3, max freq. 2.2GHz
Memory	64GB 256-bit LPDDR5, 3200MHz
Expansion Slot	2x M.2 2280 M key (PCIe x4) 1x M.2 3042/3052 B key 1x M.2 2030 E Key 2x UART 1x I2S 1x I2C 2x SPI 1x CAN 1x SIM 1x MIPI Camera connector (6CSI camera support)

DISPLAY

Display	1x HDMI 2.0(max resolution 3840x2160)
---------	---------------------------------------

STORAGE

On Board	64GB eMMC 5.1
M.2	1x M.2 2280 M-Key 256GB SSD, up to 2TB

ETHERNET

Ethernet	1x RGMII GbE LAN (10/100/1000 Mbps supported) 1x GbE LAN by Intel® i210
Ethernet	1x MGBE 10GbE LAN

FRONT I/O

X1	12V~30V DC-IN with D38999 connector
X2	1x HDMI with D38999 connector
X3	1x USB3.0 Type-A with D38999 Connector
X4	1x GbE LAN(RJ45) with D38999 connector
X5	1x USB2.0 + 1x RS422 with D38999 Connector



REAR I/O

Debug Access Panel	1x Boot LED
	1x Reset Button
	1x Recover Button
	1x USB type-C for Debug
	1x USB Type-C for Recovery

Ground Screw	1x
--------------	----

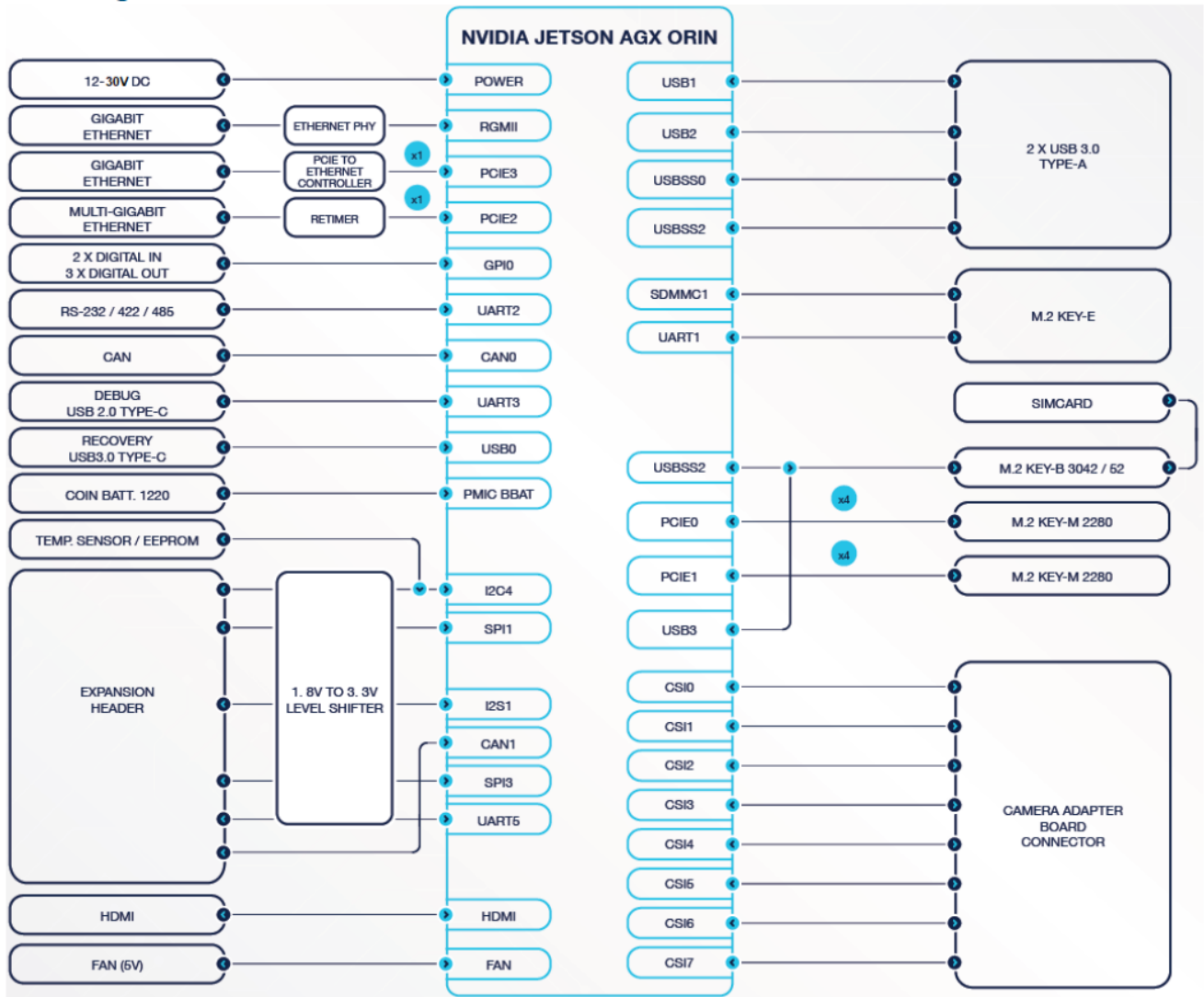
POWER REQUIREMENT

Power Input	12V~30 DC-in
-------------	--------------

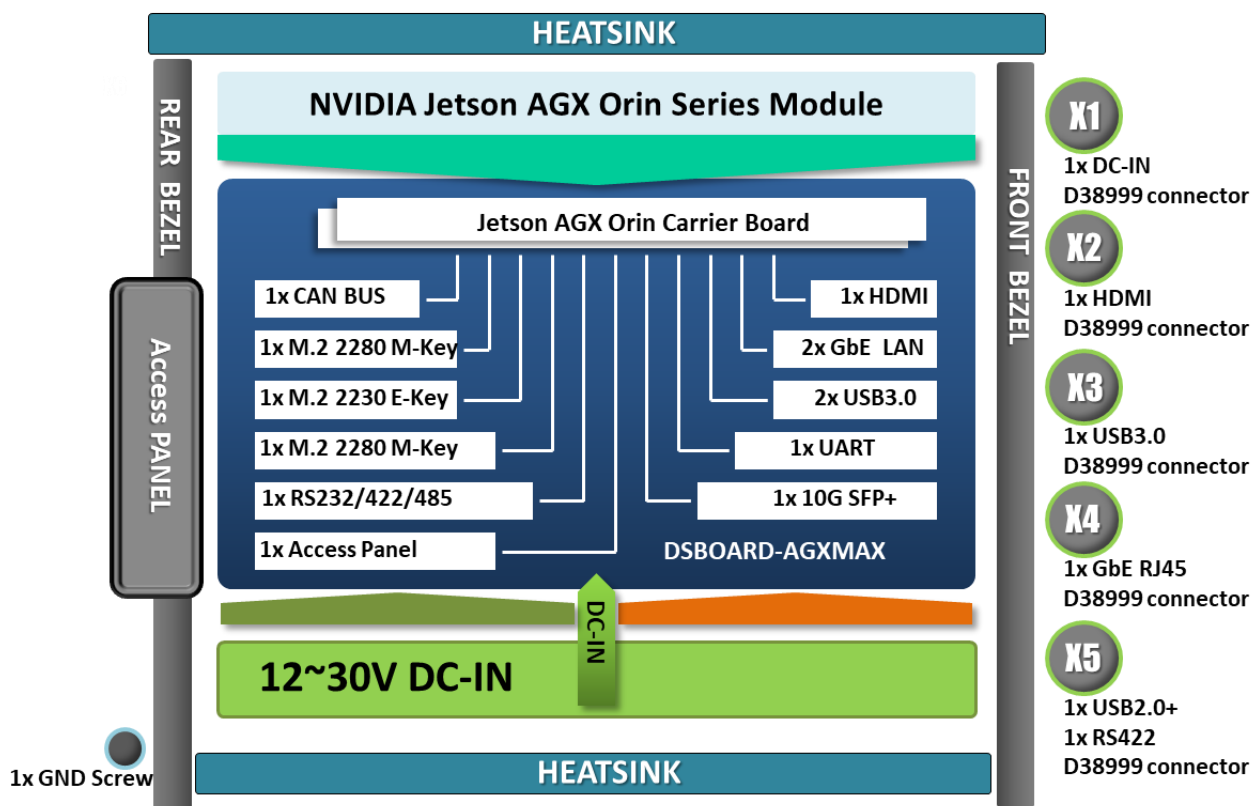
APPLICATIONS, OPERATING SYSTEM

Applications	Energy/Smart Grid/Power Plant Management, Intelligent Automation and manufacturing applications/ AI
Operating System	Ubuntu 20.04 with JetPack6.0 or later

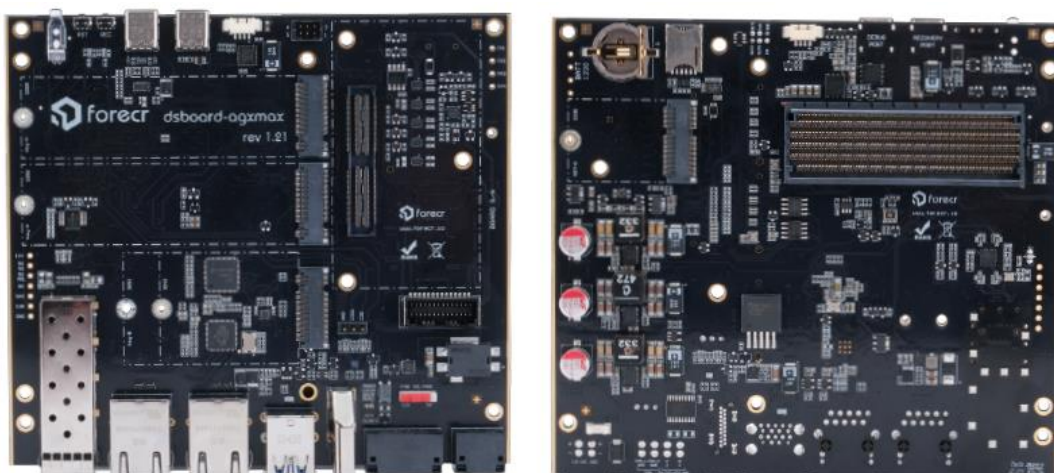
Mother Board Block Diagram



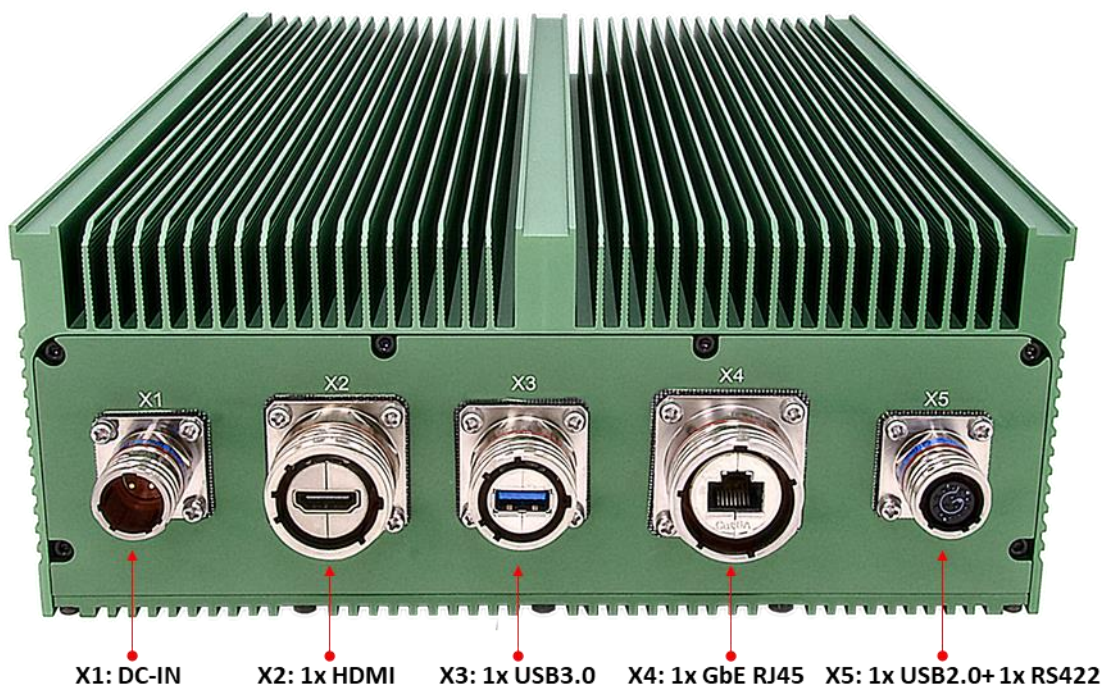
System Block Diagram



Board Visuals



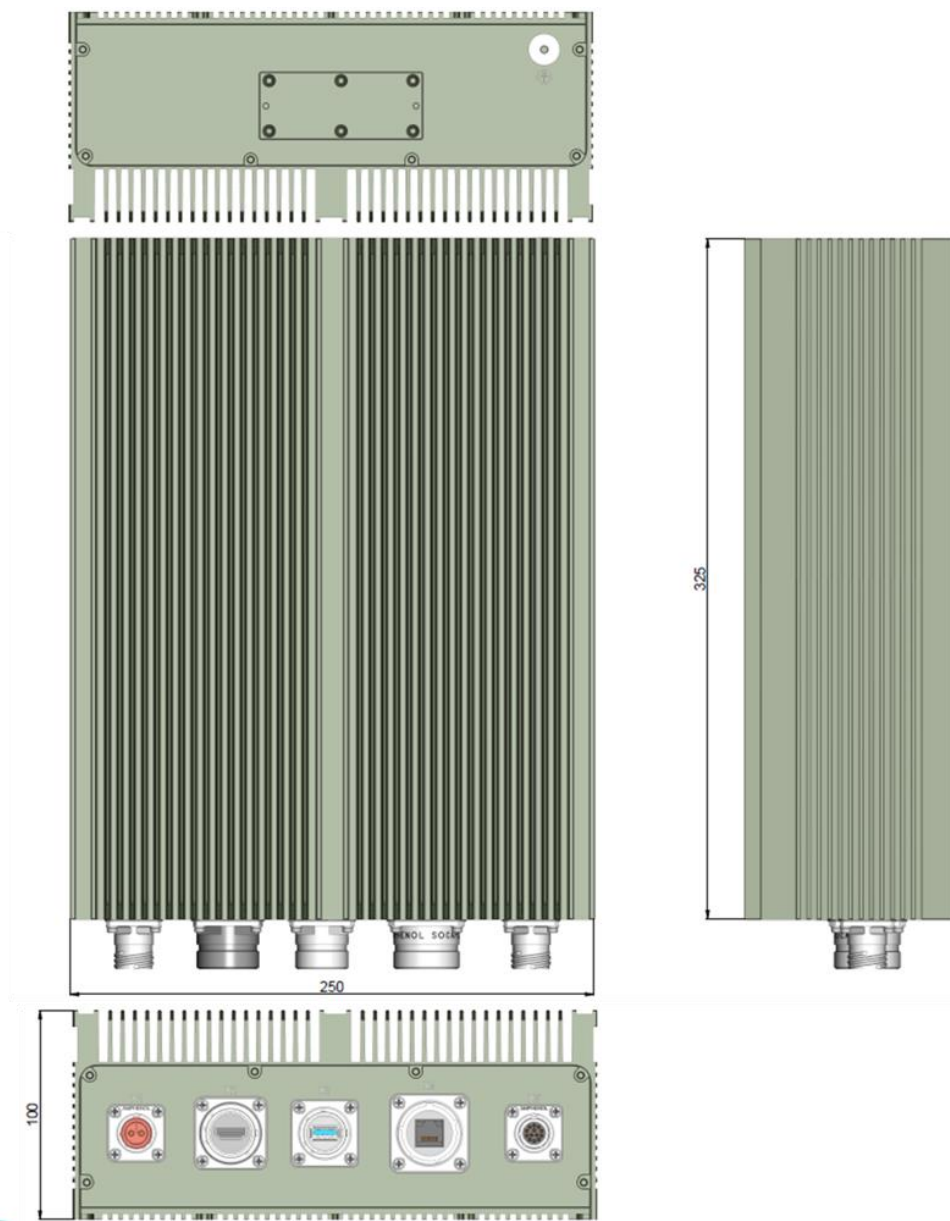
System Front/Rear IO Visuals



List of Connectors & Buttons

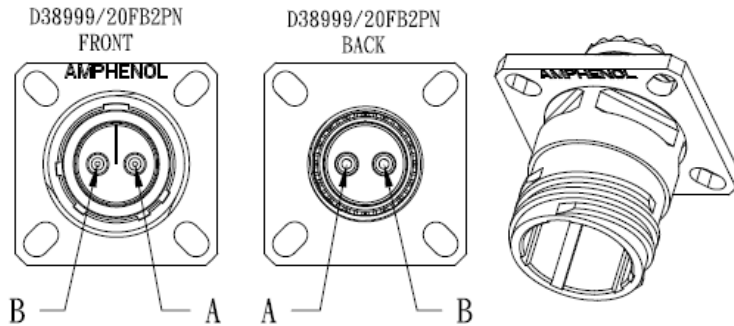


Dimension Drawing



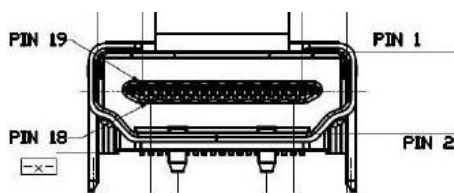
Definition of each I/O

X1: DC Power IN:



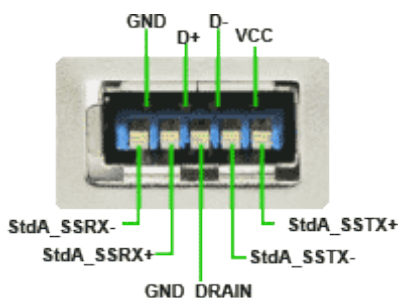
D38999/20FB2PN DC-IN	
A	Vin+
B	Vin-

X2: HDMI 2.0



Pin #	Signal	Pin #	Signal
1	HDMI_TX2_P	2	GND
3	HDMI_TX2_N	4	HDMI_TX1_P
5	GND	6	HDMI_TX1_N
7	HDMI_TX0_P	8	GND
9	HDMI_TX0_N	10	HDMI_CLK_P
11	GND	12	HDMI_CLK_N
13	CEC	14	NC
15	HDMI_SCL	16	HDMI_SDA
17	GND	18	+5 V Power
19	Hot Plug Detect	20	GND
21	GND	22	GND
23	GND		

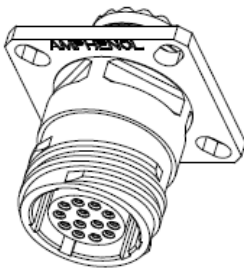
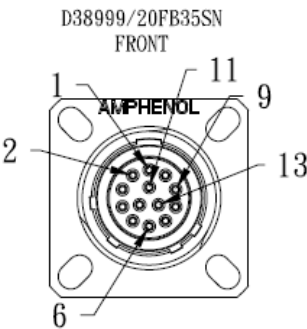
X3: USB3.0



Pin	Name	Direction	Color	Description
1	VBUS		red	+5 V power
2	D-	↔	white	USB 2.0 Data -
3	D+	↔	green	USB 2.0 Data +
4	GND		black	Ground
5	StdA_SSRX-	←	blue	SuperSpeed receiver
6	StdA_SSRX+	←	yellow	SuperSpeed receiver
7	GND_DRAIN		ground	Ground
8	StdA_SSTX-	→	purple	SuperSpeed transmitter
9	StdA_SSTX+	→	orange	SuperSpeed transmitter

X4: 1x GbE LAN RJ45 jack with D38999 connector

X5: 1x USB2.0 + 1x RS422:



D38999/20FB35SN RS422+USB2.0	
1	VCC
2	D-
3	D+
4	GND
5	TX+
6	TX-
7	RX+
8	RX-
13	GND
9	N. C.
10	N. C.
11	N. C.
12	N. C.

NV300-AMB User's Manual

Revision Date: Nov. 13. 2025



Ordering Information

Model	NV300-2L64	NV300-1LS64	NV300-AMB	NV300-2LG64	NV300-2LGS64
GPU	2048-core NVIDIA Ampere with 64 Tensor Cores (AGX Orin 64GB)				
Memory	64GB				
AI Performance	275 TOPs				
CPU	12-core Arm® Cortex®-A78AE v8.2 64-bit CPU 3MB L2 + 6MB L3				
Module total module power	15W 35W 60W, and up to 75W				
Storage	2x M.2 NVMe	1x M.2 NVMe 256GB	1x M.2 NVMe 256GB	2x M.2 NVMe	1x M.2 NVMe
	64GB eMMC 5.1 Flash	64GB eMMC 5.1 Flash	64GB eMMC 5.1 Flash	64GB eMMC 5.1 Flash	64GB eMMC 5.1 Flash
Front I/O					
X1	12-30 VDC with D38999	12-30 VDC with D38999	12-30 VDC with D38999	12-30 VDC with D38999	12-30 VDC with D38999
X2	1x HDMI	1x HDMI with D38999	1x HDMI with D38999	1x HDMI	1x HDMI
X3	1x USB3.0	1x USB2.0 with D38999	1x USB3.0 with D38999	1x USB3.0	1x USB3.0
X4	1x 10GB MPO	1x GbE RJ45 with D38999	1x GbE RJ45 with D38999	1x 10GB MPO	1x 10GB MPO
X5	2x GbE+1x RS232/422/485 +1x CAN+2x DI+3x DO	1x USB2.0+ 1x RS422 with D38999	1x USB2.0+ 1x RS422 with D38999	2x GbE+1x RS232/422/485 +1x CAN+2x DI+3x DO	2x GbE+1x RS232/422/485 +1x CAN+2x DI+3x DO
Rear I/O					
GMSL	N/A	N/A	N/A	4x (up to 8x)	4x (up to 8x)
3G-SDI	N/A	2x 3G-SDI with D38999	N/A	N/A	4x
	TBD	1x RS422	N/A	TBD	TBD
Access panel					
	1x Reset Button				
	1x Recover Button				
	1x Debug Type-C				
	1x Recovery Type-C				
	1x Boot LED				
Dimensions	250 x 325 x 100mm(WxDxH)				



Software Information

Software Configuration

JetPack-6.X Installation can be found here:

<https://www.forecr.io/blogs/installation/jetpack-6-x-installation-for-dsboard-agxmax>

System Recovery

You will need a host PC in order to flash your client device with a new system image.

Host PC

Before flashing the image, you should prepare an OTG cable (USB Type-C) for connecting to NV300-AMB (recovery port), and a host PC with USB Type-A running Ubuntu 20.04.

JetPack-6.x Installation for NV300-AMB.

In this tutorial, we will install JetPack-6 for NV300-AMB. First, we will include our Image, DTB(Device Tree Blob) & pinmux files in Jetson OS image. Then, we will install the Jetson OS into the NV300-AMB. Finally, we will install the Jetson SDK components into it.

Attention: Before starting the flashing process, please remove the 10G SFP+ Ethernet adapter on Carrier Board (DSBOARD-AGXMAX) if you have mounted it.

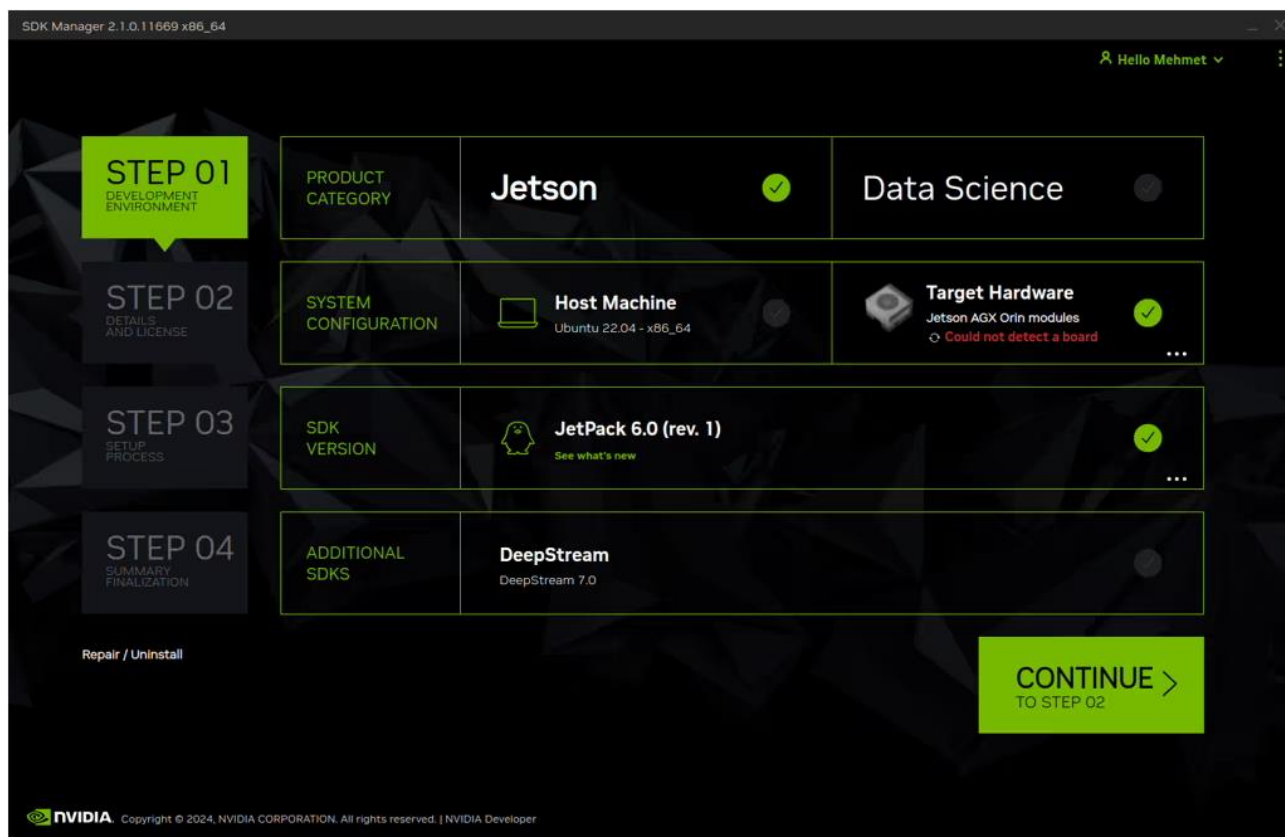
Attention: If you want to transfer the root file-system to an external drive and it has another JetPack version root file system, please format it before starting this blog post.

Known Issues for JetPack-6.0: The desktop service is not loading while the SFP+ transceiver has been connected before powering the DSBOARD-AGXMAX (previous versions from rev-1.22) up. If you will mount physically after powering it up, the desktop will load and the network interface established by the system. We are working on it to fix this problem.

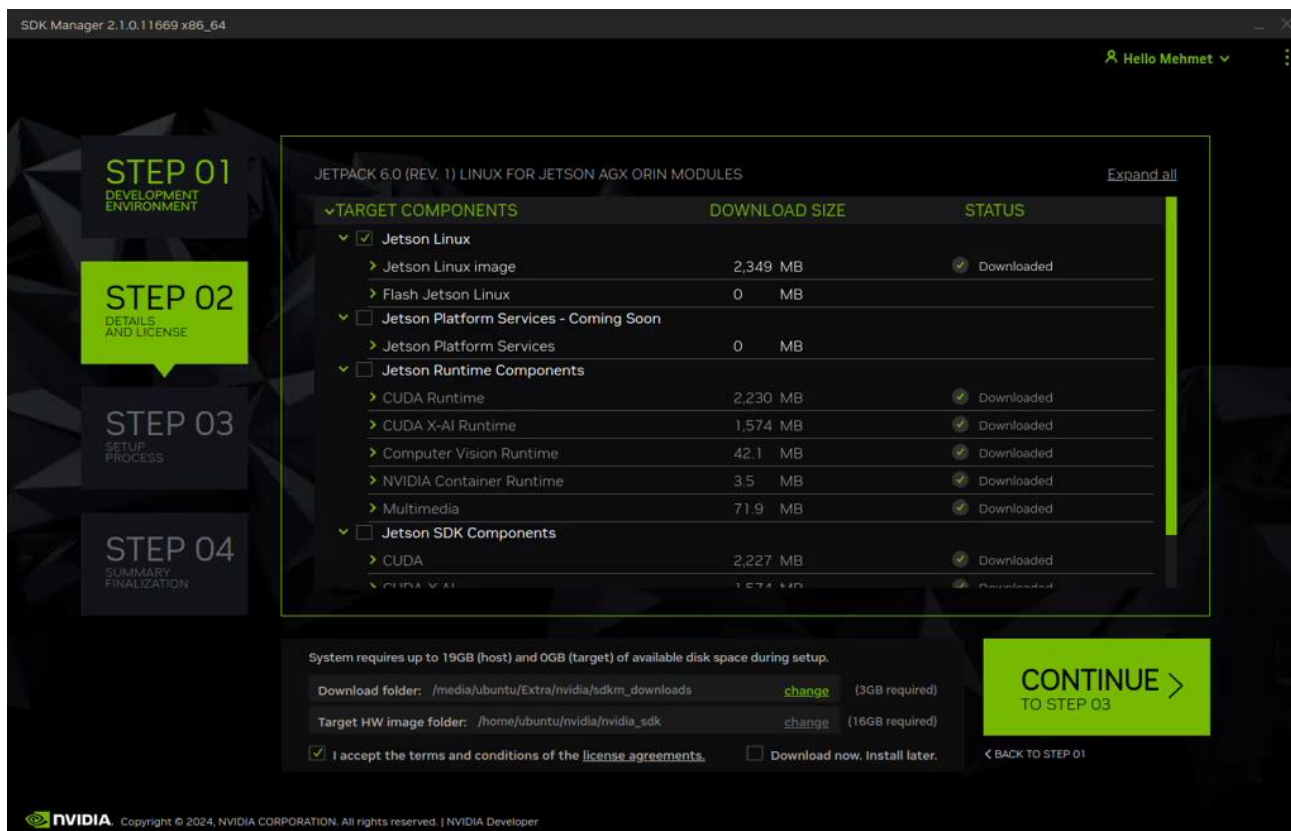
Including the Kernel Files in Jetson OS Image

Open the NVIDIA SDK Manager (<https://developer.nvidia.com/sdk-manager>). Select the correct JetPack version (sdkmanager_2.1.0 or later version) for Target Operating System and “Jetson AGX Orin modules” for Target Hardware (The “Host Machine” components are not required).

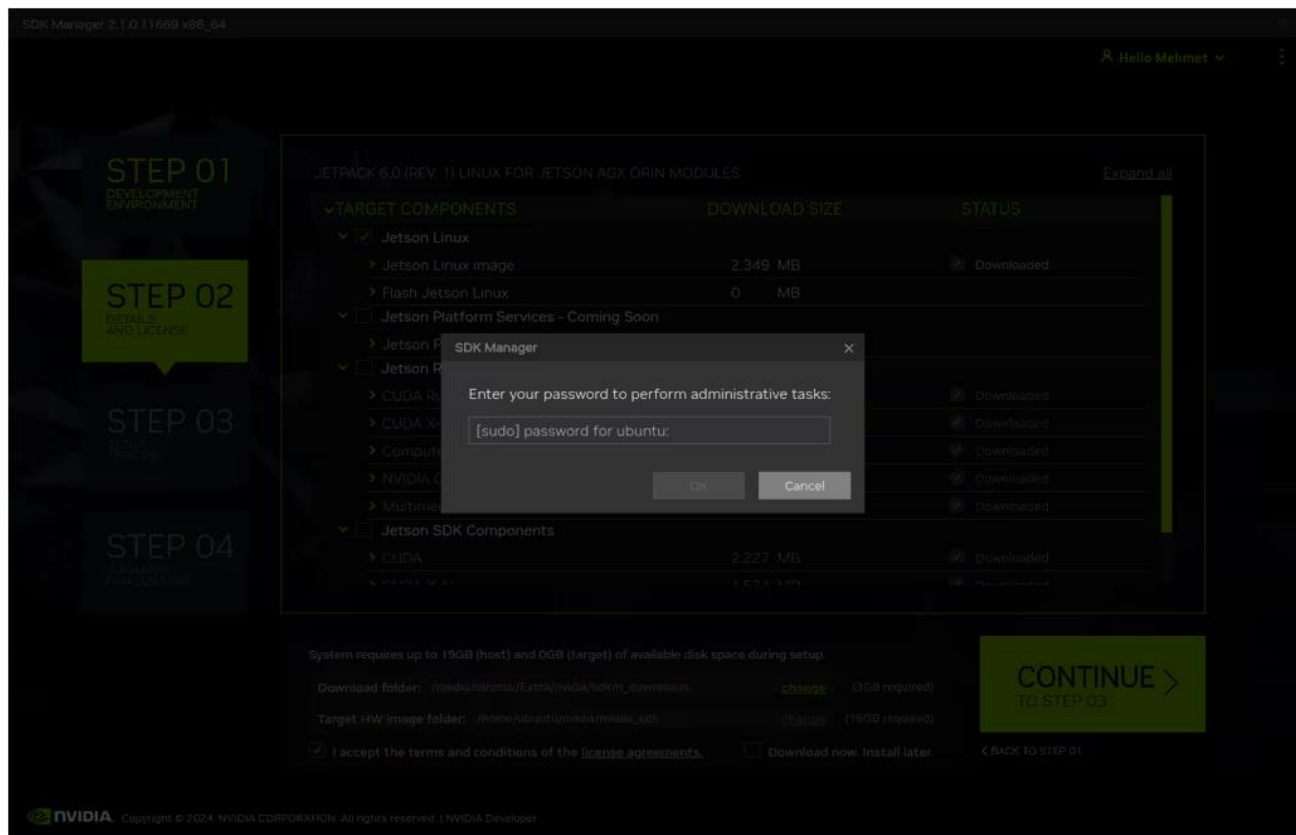
Then, continue to Step 2.



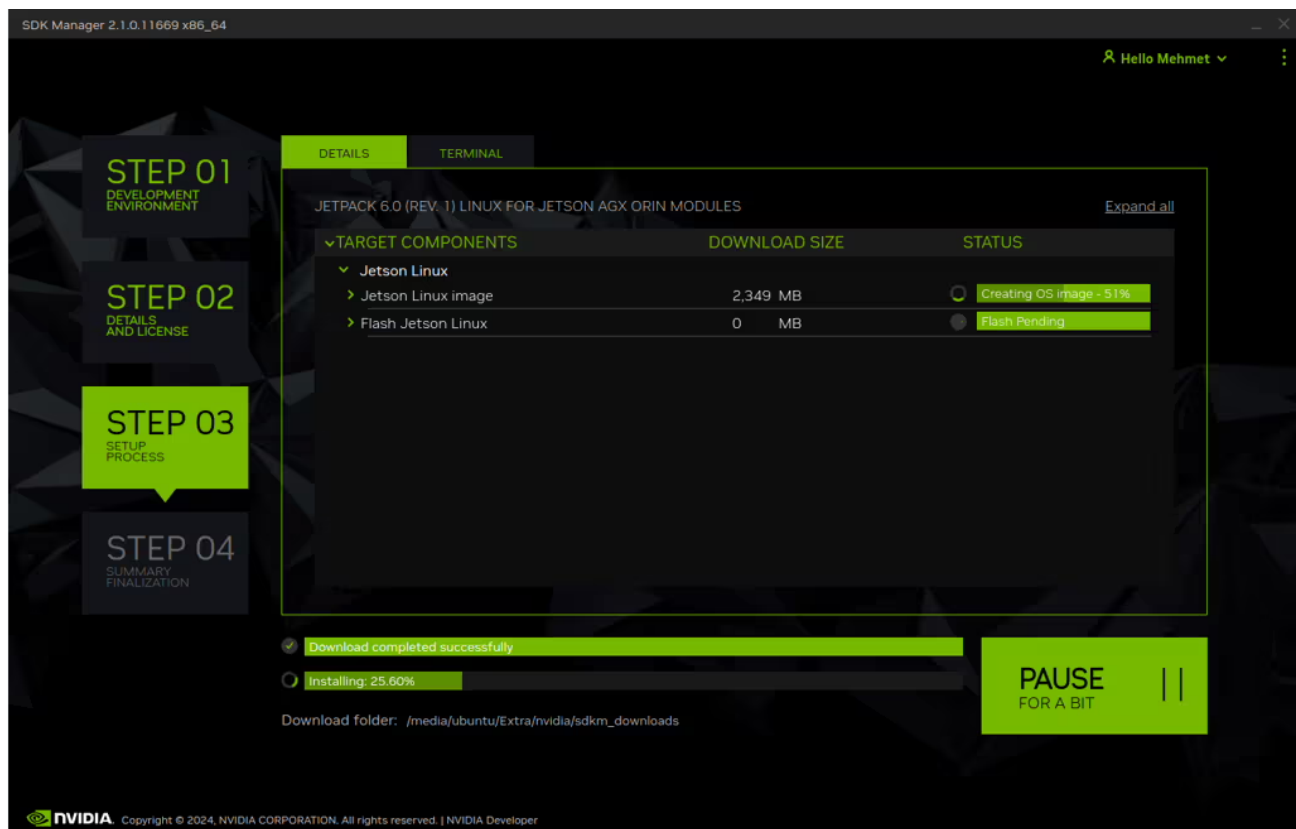
Choose only “Jetson Linux”, accept the terms & conditions and continue to Step 3.



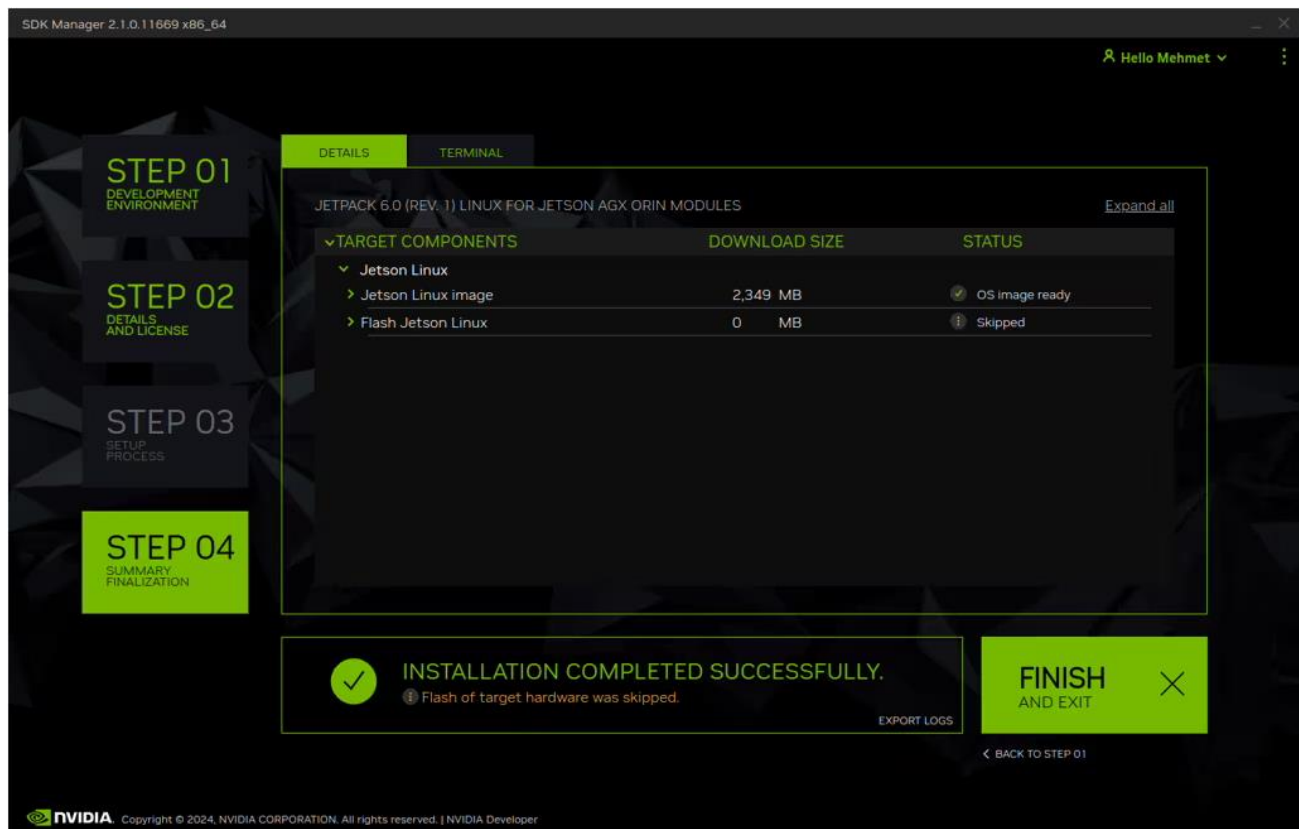
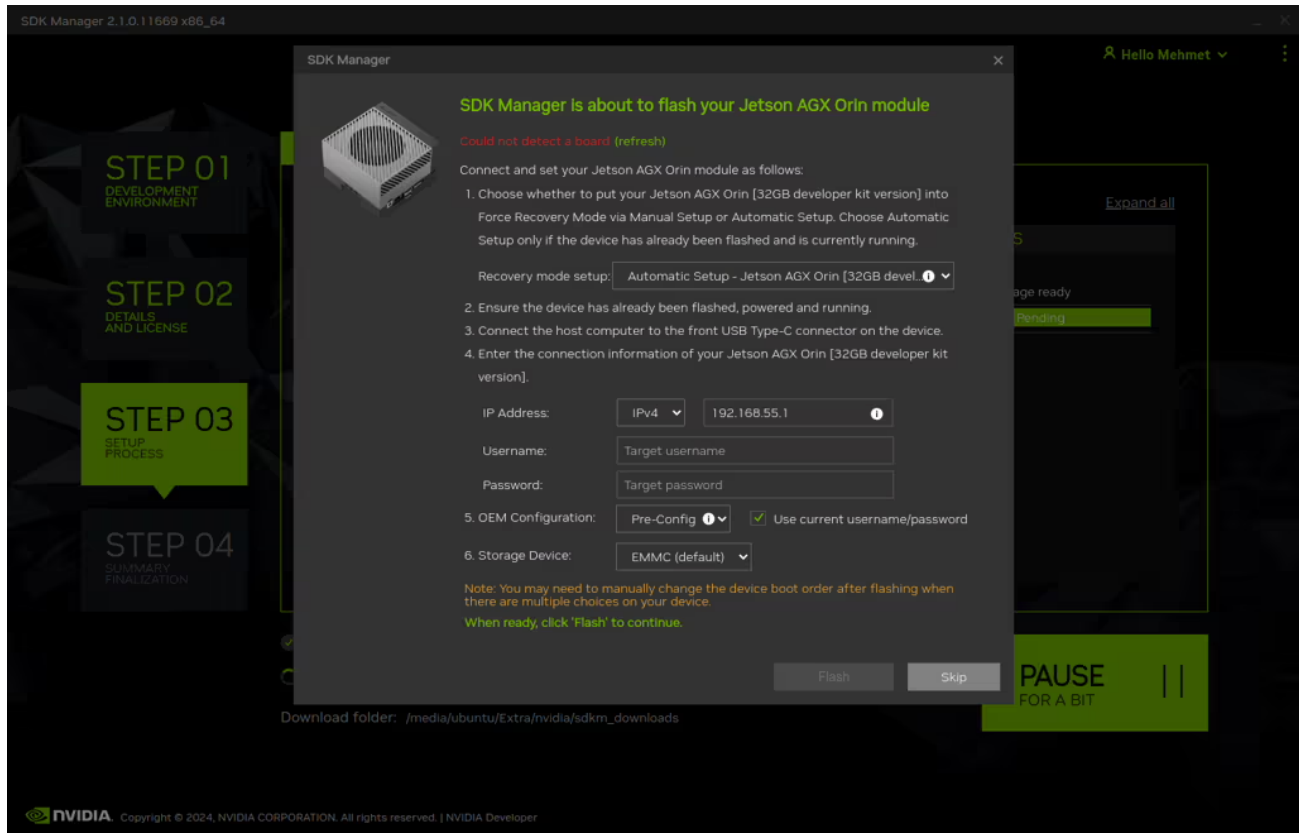
The SDK Manager will ask the username's password. Fill it and continue.



A few seconds later...



After the Jetson OS has created, the SDK Manager asks the Jetson module's flashing style. Just skip it and exit from the SDK Manager.



Open the target HW image folder.

For JetPack-6.0:

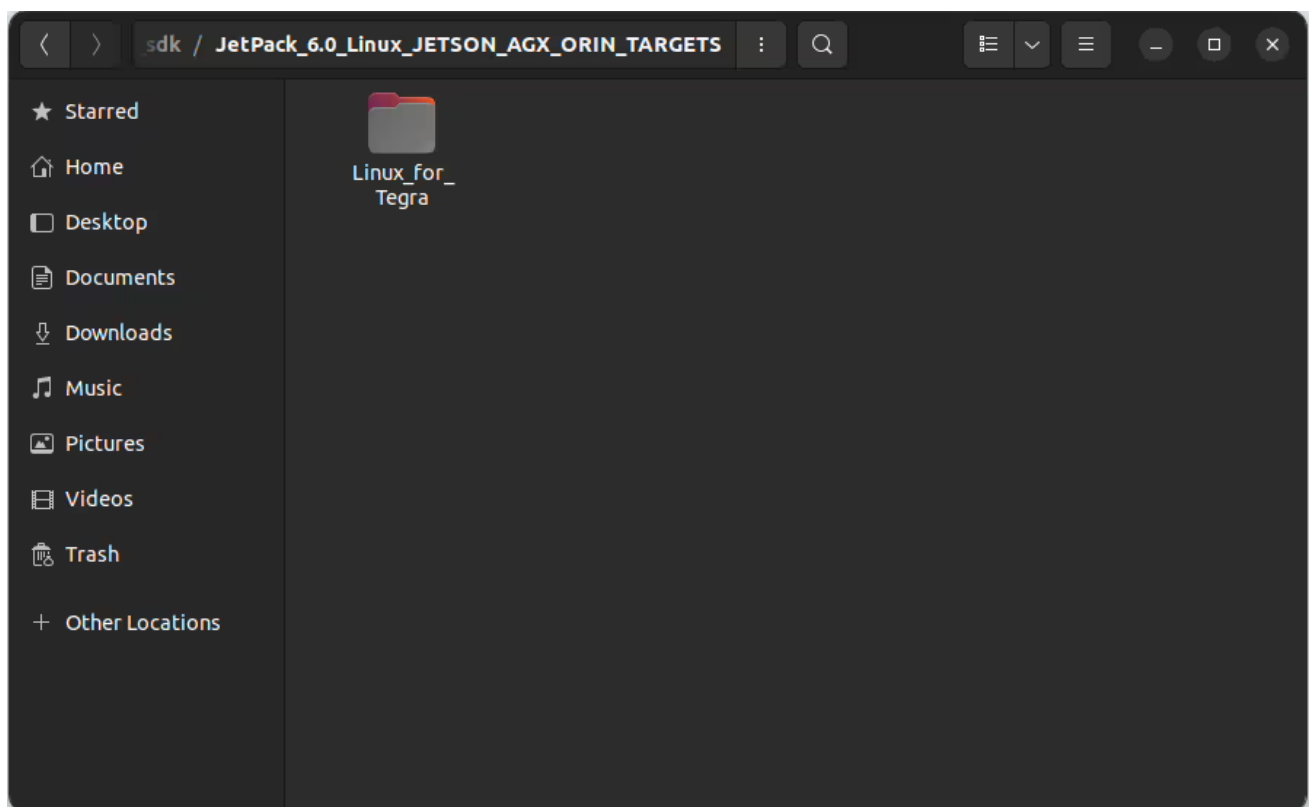
AGX Orin or AGX Orin Industrial: ~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARGETS/

For JetPack-6.1:

AGX Orin or AGX Orin Industrial: ~/nvidia/nvidia_sdk/JetPack_6.1_Linux_JETSON_AGX_ORIN_TARGETS/For JetPack-5.1.1

For JetPack-6.2:

AGX Orin or AGX Orin Industrial: ~/nvidia/nvidia_sdk/JetPack_6.2_Linux_JETSON_AGX_ORIN_TARGETS/



For JetPack-6.0:

Download the BSP files from GitHub link and extract it ([AGX Orin](#), [AGX Orin Industrial](#))

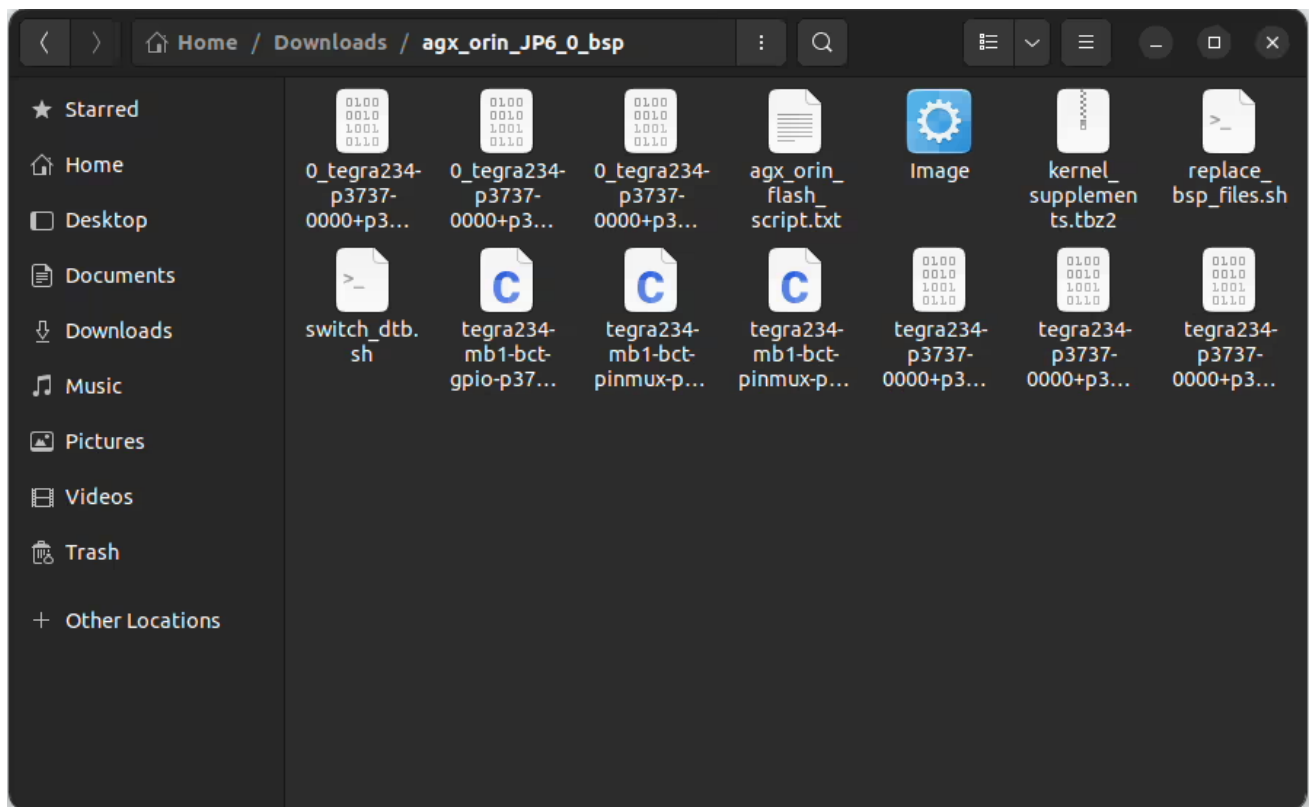
For JetPack-6.1:

Download the BSP files from GitHub link and extract it ([AGX Orin](#), [AGX Orin Industrial](#))

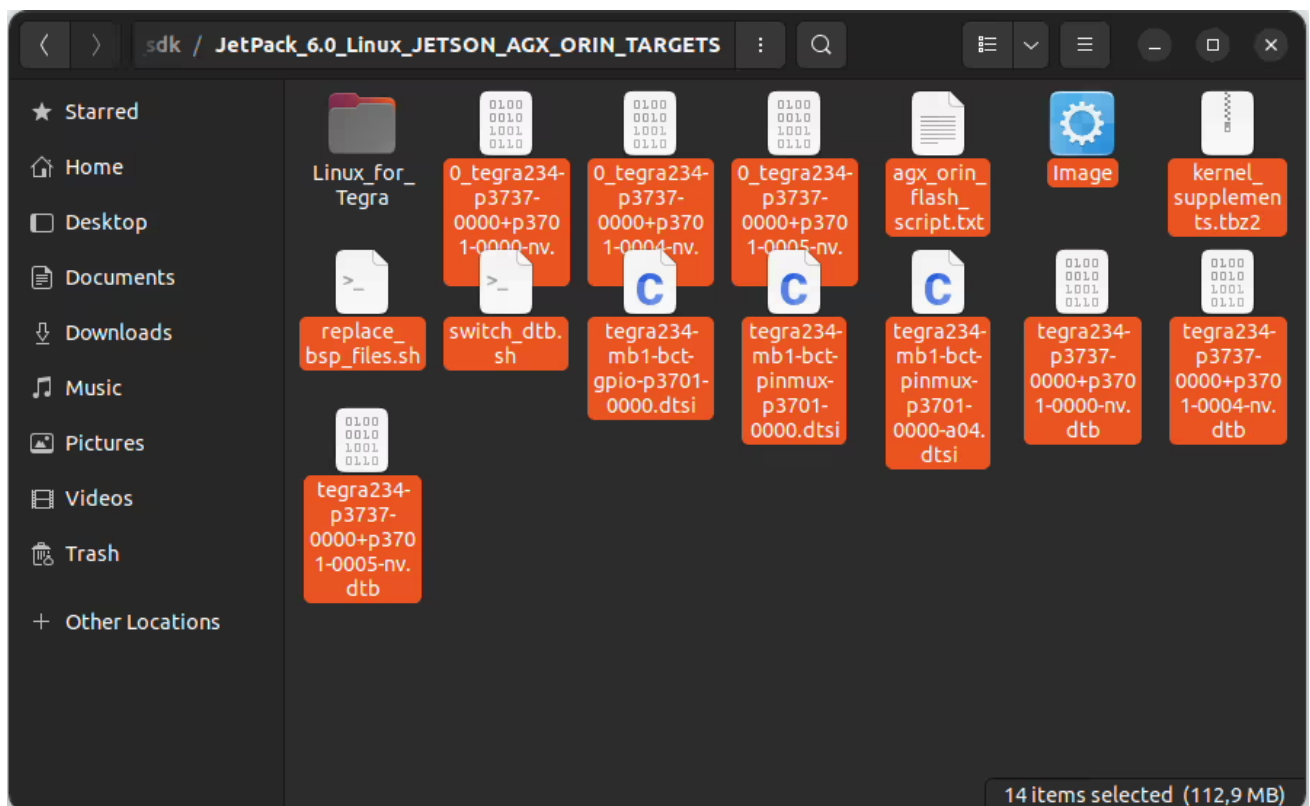
For JetPack-6.2:

Download the BSP files from GitHub link and extract it ([AGX Orin](#), [AGX Orin Industrial](#))

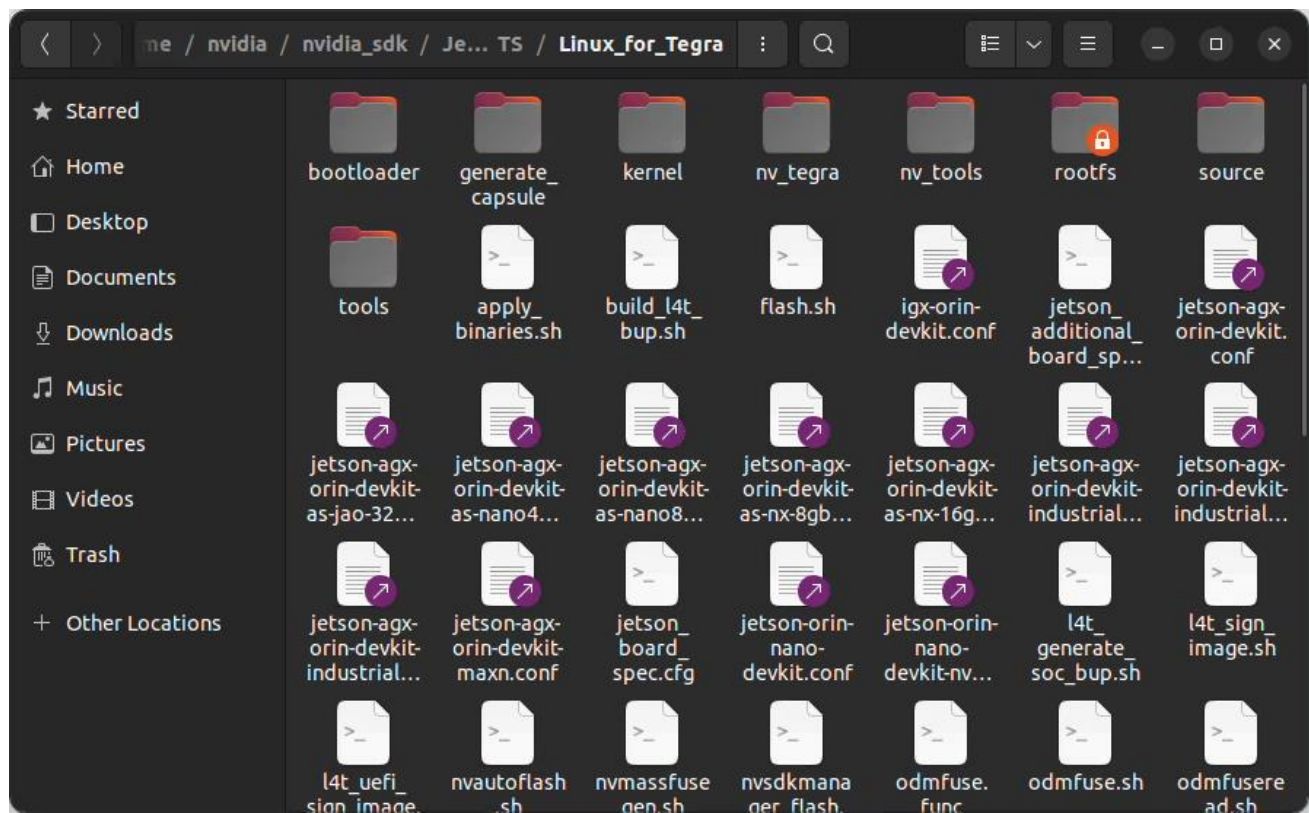
Hint: The following steps have done for AGX Orin, but they are the same for the other Jetson module types (only the BSP files and flashing commands are different).



Copy all files to the target HW image folder.



Open a terminal from the "Linux_for_Tegra" folder



Create the system binaries with these commands below:

```
sudo ./tools/l4t_flash_prerequisites.sh  
sudo ./apply_binaries.sh
```

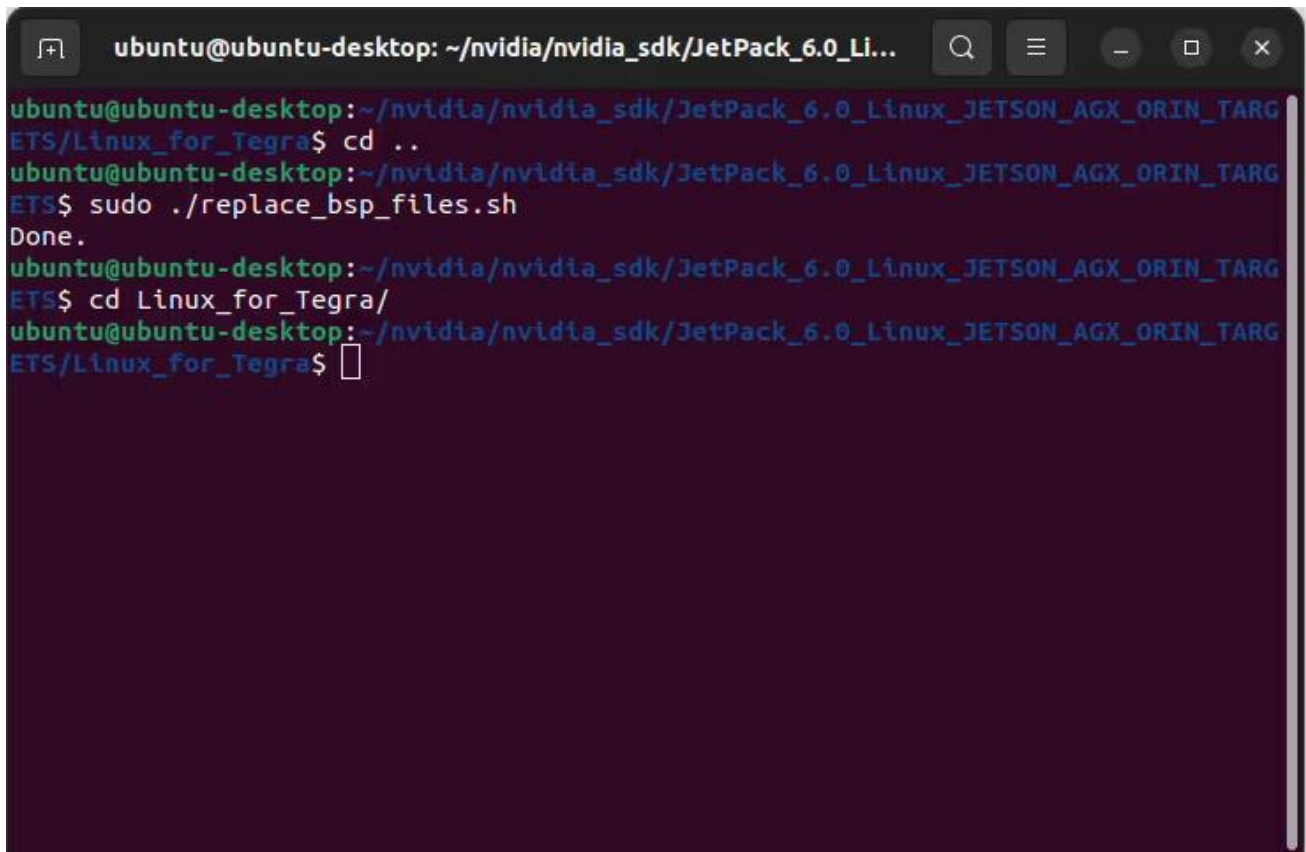
```
ubuntu@ubuntu-desktop: ~/nvidia/nvidia_sdk/JetPack_6.0_Li...
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS/Linux_for_Tegra$ sudo ./tools/l4t_flash_prerequisites.sh
```

```
ubuntu@ubuntu-desktop: ~/nvidia/nvidia_sdk/JetPack_6.0_Li...
device-tree-compiler is already the newest version (1.6.1-1).
dosfstools is already the newest version (4.2-1build3).
lz4 is already the newest version (1.9.3-2build2).
python3-yaml is already the newest version (5.4.1-1ubuntu1).
whois is already the newest version (5.5.13).
zstd is already the newest version (1.4.8+dfsg-3build1).
abootimg is already the newest version (0.6-1build1).
lbzip2 is already the newest version (2.5-2.3).
sshpass is already the newest version (1.09-1).
binutils is already the newest version (2.38-4ubuntu2.6).
cpio is already the newest version (2.13+dfsg-7ubuntu0.1).
libxml2-utils is already the newest version (2.9.13+dfsg-1ubuntu0.4).
nfs-kernel-server is already the newest version (1:2.6.1-1ubuntu1.2).
openssl is already the newest version (3.0.2-0ubuntu1.15).
rsync is already the newest version (3.2.7-0ubuntu0.22.04.2).
udev is already the newest version (249.11-0ubuntu3.12).
uuid-runtime is already the newest version (2.37.2-4ubuntu3.4).
qemu-user-static is already the newest version (1:6.2+dfsg-2ubuntu6.19).
The following packages were automatically installed and are no longer required:
  libwpe-1.0-1 libwpebackend-fdo-1.0-1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 111 not upgraded.
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS/Linux_for_Tegra$ sudo ./apply_binaries.sh
```

```
ubuntu@ubuntu-desktop: ~/nvidia/nvidia_sdk/JetPack_6.0_Li...
ko
Add /lib/modules/5.15.136-tegra/kernel/drivers/pwm/pwm-tegra.ko
Add /lib/modules/5.15.136-tegra/kernel/drivers/hwmon/pwm-fan.ko
Add /lib/modules/5.15.136-tegra/kernel/drivers/pci/controller/dwc/pcie-tegra194.ko
Add /lib/modules/5.15.136-tegra/kernel/drivers/phy/tegra/phy-tegra194-p2u.ko
Add /bin/sh
Add /lib/modules/5.15.136-tegra/kernel/drivers/usb/gadget/udc/tegra-xudc.ko
Add /lib/modules/5.15.136-tegra/kernel/drivers/usb/typec/ucsi/typec_ucsi.ko
Add /lib/modules/5.15.136-tegra/kernel/drivers/usb/typec/ucsi/ucsi_ccg.ko
Add /lib/modules/5.15.136-tegra/kernel/drivers/usb/typec/typec.ko
Cleaning up the temporary directory for updating the initrd..
Processing triggers for nvidia-l4t-kernel (5.15.136-tegra-36.3.0-20240506102626)
...
/home/ubuntu/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARGETS/Linux_for_Tegra
Removing QEMU binary from rootfs
Removing stashed Debian packages from rootfs
L4T BSP package installation completed!
Disabling NetworkManager-wait-online.service
Disable the ondemand service by changing the runlevels to 'K'
Success!
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARGETS/Linux_for_Tegra$
```

Apply the interface configurations with the following commands below:

```
cd ..
sudo ./replace_bsp_files.sh
cd Linux_for_Tegra/
```



```
ubuntu@ubuntu-desktop: ~/nvidia/nvidia_sdk/JetPack_6.0_Li...
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS/Linux_for_Tegra$ cd ..
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS$ sudo ./replace_bsp_files.sh
Done.
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS$ cd Linux_for_Tegra/
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS/Linux_for_Tegra$
```

Hint: If you want to configure your username-password & hostname with default settings, you can create user without the Ubuntu installation wizard. To do this, the user generation command structure should be:

```
sudo tools/l4t_create_default_user.sh -u {USERNAME} -p {PASSWORD} -a -n {HOSTNAME} --accept-license
```

For example (username:"nvidia", password:"nvidia", device-name:"nvidia-agx-orin"):

```
sudo tools/l4t_create_default_user.sh -u nvidia -p nvidia -a -n nvidia-agx-orin --accept-license
```

Jetson OS Installation

Connect the recovery USB (between installer PC & AV710-X4-32G4S's recovery USB) and power connection of your NV300-1LS64.

While the NV300-1LS64's power connector plugged in,

- press reset & recovery buttons together
- release reset button
- release the recovery button after 3 seconds later. This will set it to Recovery mode.

Attention: To be able to get more stable USB connection, please do not connect the Jetson via a USB hub or docking station to the host PC. Use a USB cable to the host PC directly.



Then, type "lsusb" and check the device connected in Recovery mode.

- **"0955:7023 NVidia Corp." for AGX Orin 64GB**
- "0955:7223 NVidia Corp." for AGX Orin 32GB
- "0955:7023 NVidia Corp." for AGX Orin Industrial

```
ubuntu@ubuntu-desktop: ~/nvidia/nvidia_sdk/JetPack_6.0_Li...  
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG  
ETS/Linux_for_Tegra$ lsusb  
Bus 008 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub  
Bus 007 Device 002: ID 046d:0825 Logitech, Inc. Webcam C270  
Bus 007 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 006 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub  
Bus 005 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub  
Bus 003 Device 002: ID 0b05:18f3 ASUSTek Computer, Inc. AURA LED Controller  
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub  
Bus 001 Device 005: ID 8087:0029 Intel Corp. AX200 Bluetooth  
Bus 001 Device 003: ID 05e3:0610 Genesys Logic, Inc. Hub  
Bus 001 Device 022: ID 413c:301a Dell Computer Corp. Dell MS116 Optical Mouse  
Bus 001 Device 021: ID 1c4f:0026 SiGma Micro Keyboard  
Bus 001 Device 020: ID 0409:005a NEC Corp. HighSpeed Hub  
Bus 001 Device 019: ID 1a40:0101 Terminus Technology Inc. Hub  
Bus 001 Device 024: ID 0955:7023 NVIDIA Corp. APX  
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG  
ETS/Linux_for_Tegra$
```

For AGX Orin, flash the Jetson OS with this script below:

```
sudo ./flash.sh jetson-agx-orin-devkit mmcblk0p1
```

For AGX Orin Industrial, flash the Jetson OS with this script below:

```
sudo ./flash.sh jetson-agx-orin-devkit-industrial mmcblk0p1
```

```
ubuntu@ubuntu-desktop: ~/nvidia/nvidia_sdk/JetPack_6.0_Li...
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS/Linux_for_Tegra$ lsusb
Bus 008 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 007 Device 002: ID 046d:0825 Logitech, Inc. Webcam C270
Bus 007 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 006 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 005 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 002: ID 0b05:18f3 ASUSTek Computer, Inc. AURA LED Controller
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 005: ID 8087:0029 Intel Corp. AX200 Bluetooth
Bus 001 Device 003: ID 05e3:0610 Genesys Logic, Inc. Hub
Bus 001 Device 022: ID 413c:301a Dell Computer Corp. Dell MS116 Optical Mouse
Bus 001 Device 021: ID 1c4f:0026 SiGma Micro Keyboard
Bus 001 Device 020: ID 0409:005a NEC Corp. HighSpeed Hub
Bus 001 Device 019: ID 1a40:0101 Terminus Technology Inc. Hub
Bus 001 Device 024: ID 0955:7023 NVIDIA Corp. APX
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS/Linux_for_Tegra$ sudo ./flash.sh jetson-agx-orin-devkit mmcblk0p1
```

At the end of the script, the device will reboot. Complete your Ubuntu installation wizard (if you have not created a user with tools/l4t_create_default_user.sh script file) from the NV300-1LS64 (language, keyboard type, location, username & password etc.).

```
ubuntu@ubuntu-desktop: ~/nvidia/nvidia_sdk/JetPack_6.0_Li...
pt
[ 691.8354 ] Bootloader version 01.00.0000
[ 692.0381 ] Writing partition A_MEM_BCT with mem_coldboot_sigheader.bct.encrypt
[ 243712 bytes ]
[ 692.0387 ] [.....] 100%
[ 695.0769 ] tegradevflash_v2 --write B_MEM_BCT mem_coldboot_sigheader.bct.encrypt
pt
[ 695.0774 ] Bootloader version 01.00.0000
[ 695.8982 ] Writing partition B_MEM_BCT with mem_coldboot_sigheader.bct.encrypt
[ 243712 bytes ]
[ 695.8987 ] [.....] 100%
[ 698.9364 ] Flashing completed

[ 698.9365 ] Coldbooting the device
[ 698.9371 ] tegrarcm_v2 --chip 0x23 0 --ismb2
[ 698.9376 ] MB2 version 01.00.0000
[ 699.7903 ] Coldbooting the device
[ 699.7909 ] tegrarcm_v2 --chip 0x23 0 --reboot coldboot
[ 699.7913 ] MB2 version 01.00.0000
*** The target generic has been flashed successfully. ***
Reset the board to boot from internal eMMC.

ubuntu@ubuntu-desktop:~/nvidia/nvidia_sdk/JetPack_6.0_Linux_JETSON_AGX_ORIN_TARG
ETS/Linux_for_Tegra$
```

[Optional] If you will use the recovery USB port as host (to be able to connect USB devices), please verify that the FDT parameter has been applied in the extlinux.conf file. In JetPack-6 installation, this parameter may not include in it. If this parameter is missing in it, please open it with a text editor and write "FDT" with the DTB file (located in the /boot/dtb/ folder).:

```
nvidia@nvidia-desktop: ~  
nvidia@nvidia-desktop:~$ cat /boot/extlinux/extlinux.conf  
TIMEOUT 30  
DEFAULT primary  
  
MENU TITLE L4T boot options  
  
LABEL primary  
    MENU LABEL primary kernel  
    LINUX /boot/Image  
    INITRD /boot/initrd  
    FDT /boot/dtb/kernel_tegra234-p3737-0000+p3701-0005-nv.dtb  
    APPEND ${cbootargs} root=/dev/mmcblk0p1 rw rootwait rootfstype=ext4 mminit  
_loglevel=4 console=ttyTCU0,115200 console=ttyAMA0,115200 firmware_class.path=/e  
tc/firmware fbcon=map:0 net.ifnames=0 nospectre_bhb video=efifb:off console=tty0  
  
# When testing a custom kernel, it is recommended that you create a backup of  
# the original kernel and add a new entry to this file so that the device can  
# fallback to the original kernel. To do this:  
#  
# 1, Make a backup of the original kernel  
#     sudo cp /boot/Image /boot/Image.backup  
#  
# 2, Copy your custom kernel into /boot/Image
```

Then, please open a terminal from the Jetson Orin and type the following command below. This will update its current device-tree and reboot it.

Otherwise, you can use this port for virtual network communication (file transfer etc. between host PC with 192.168.55.1 IP address) in default.

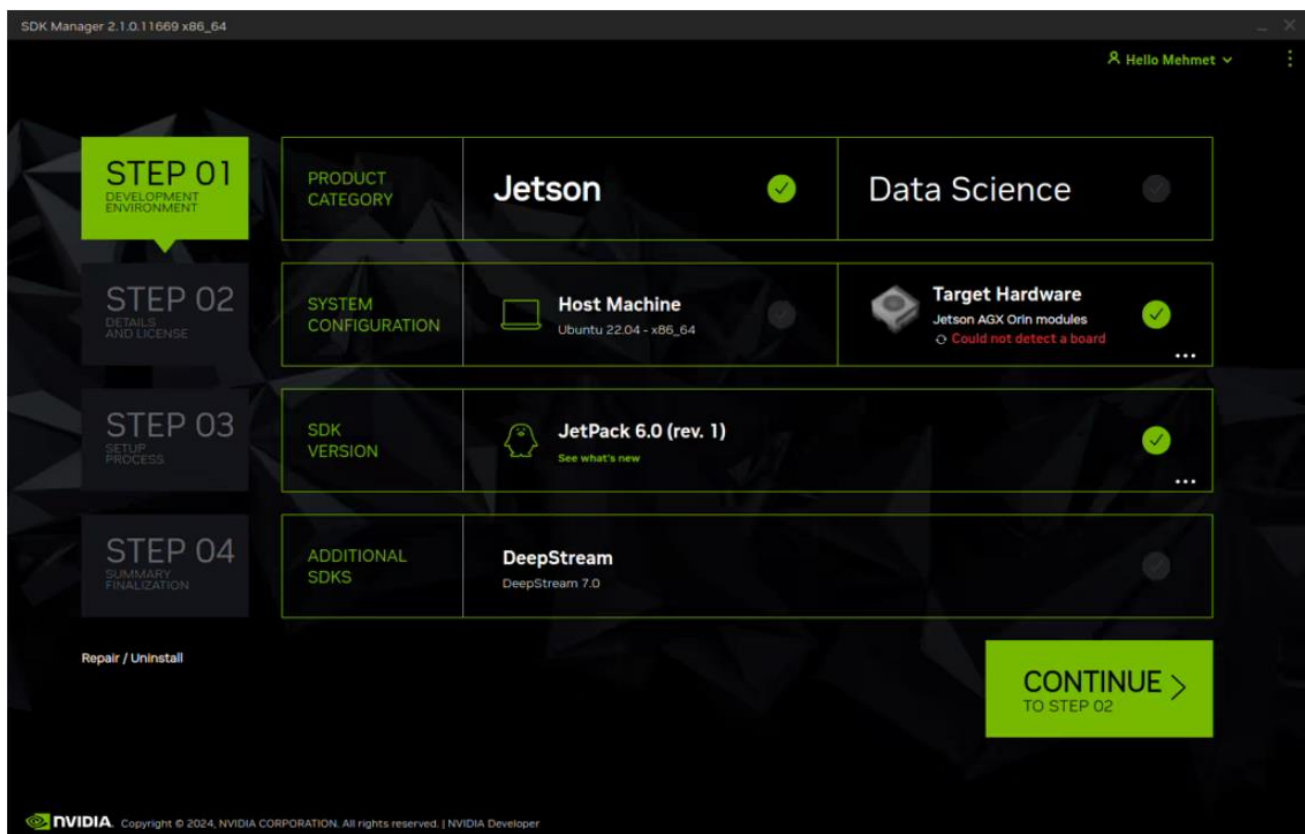
```
sudo switch_dtb.sh
```

```
nvidia@nvidia-desktop: ~  
nvidia@nvidia-desktop:~$ sudo switch_dtb.sh  
Base DTB: kernel_tegra234-p3737-0000+p3701-0005-nv.dtb  
New DTB: tegra234-p3737-0000+p3701-0005-nv.dtb  
Done.  
Rebooting...  
█
```

Jetson SDK Components Installation

Connect the NV300-1LS64 to the Internet. Then, Open the NVIDIA SDK Manager. Select the correct JetPack version for Target Operating System and “Jetson AGX Orin modules” for Target Hardware (The “Host Machine” components are not required. Additional SDKs (DeepStream) are optional).

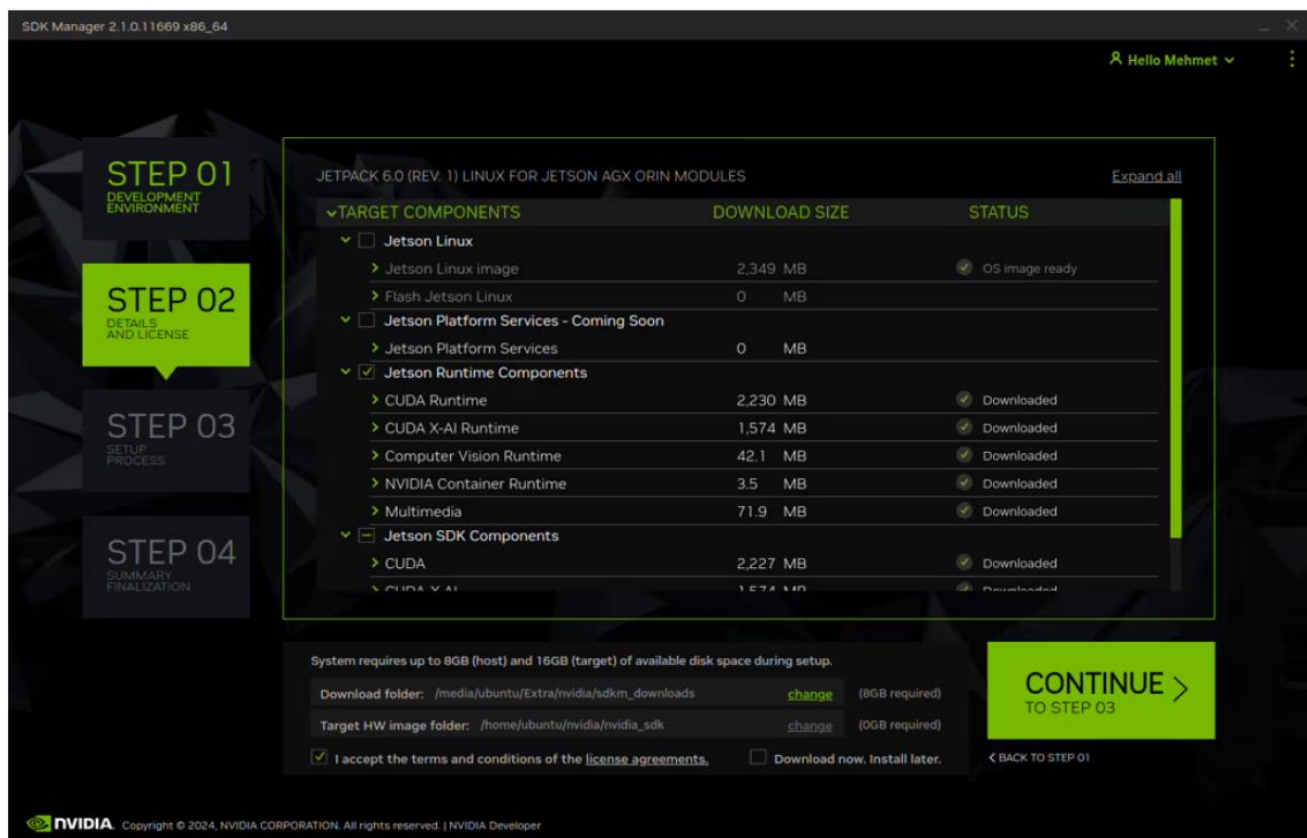
Then, continue to Step 2.



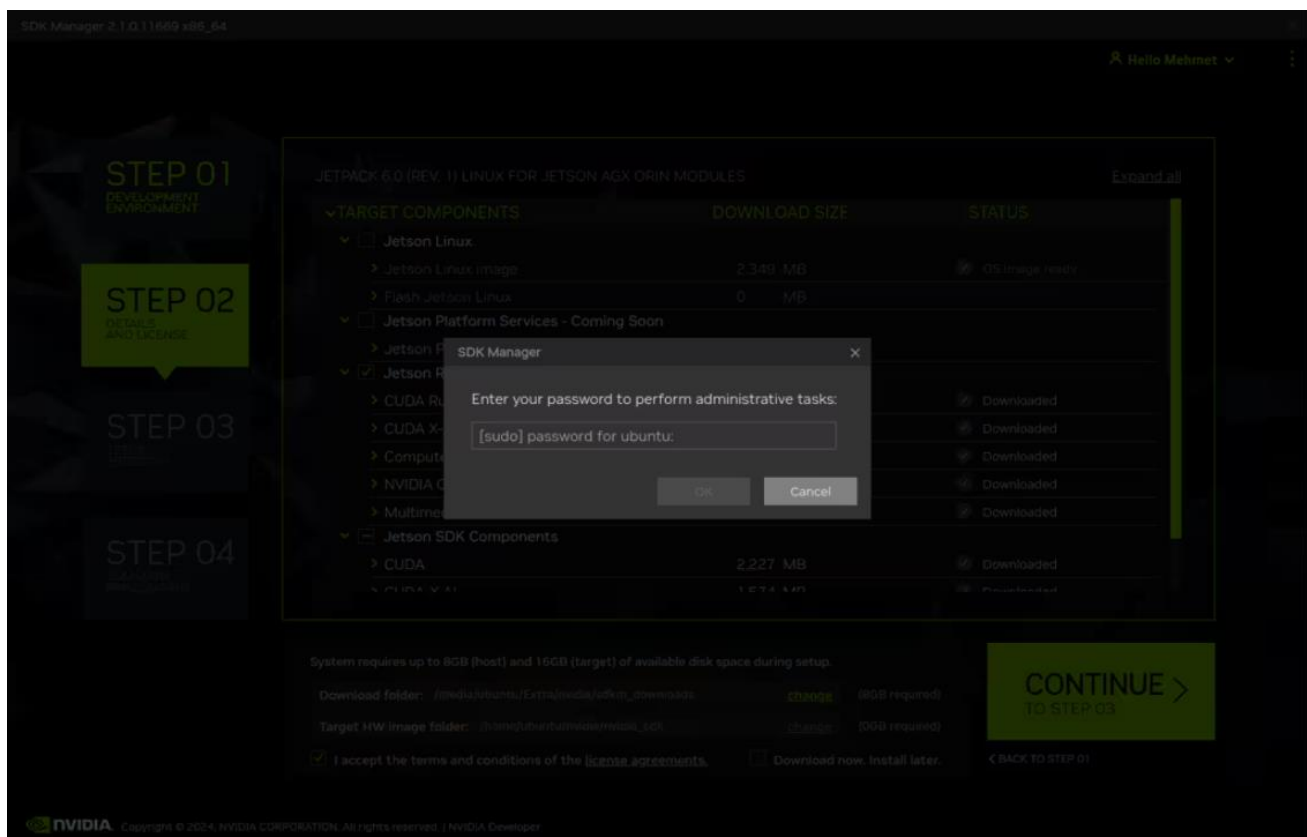
Choose at least “Jetson Runtime Components” (“Jetson SDK Components” are optional. It depends on your use case), accept the terms & conditions and continue to Step 3.

NV300-AMB User's Manual

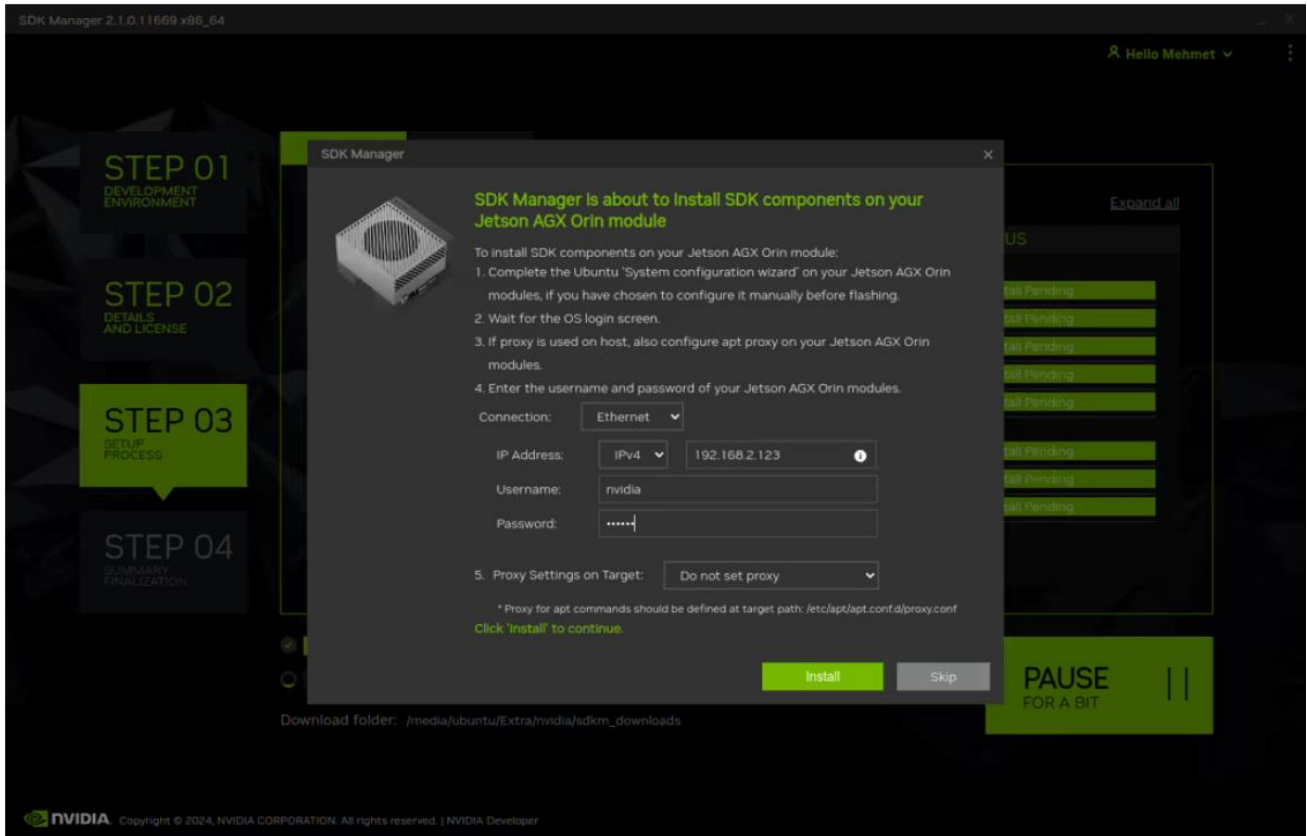
Revision Date: Nov. 13. 2025



The SDK Manager will ask the username's password. Fill it and continue.



Type the IP address, username and password of Jetson AGX Orin module and install the SDK Components.



At the end of the installation, the NV300-1LS64 becomes ready.

To avoid kernel update with "apt upgrade" or "apt-get upgrade" commands, please follow [this guide](#) on the Jetson module.

Change Root File System to M.2 SSD Directly

Formatting the M.2 SSD Storage

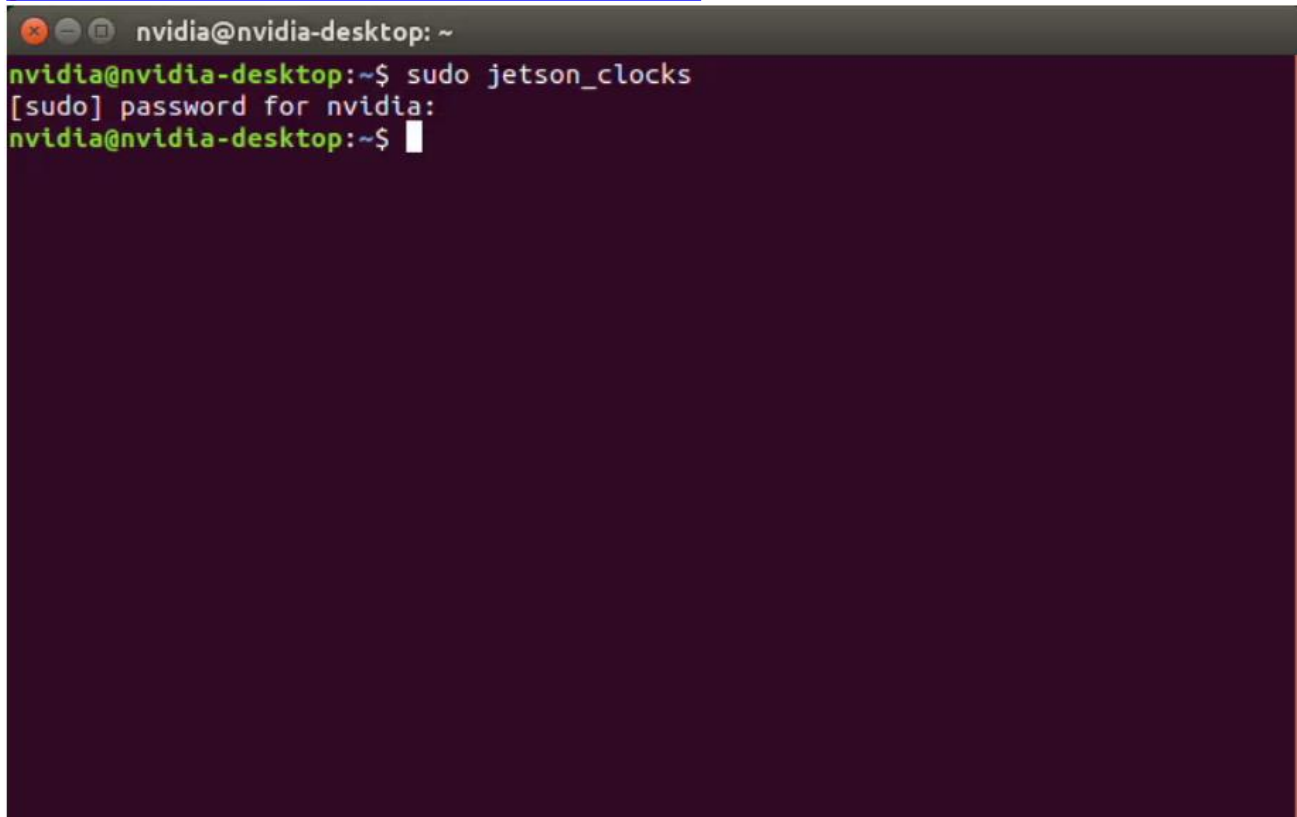
we will explain how to move your root file system on EMMC flash to SSD storage on M.2 slot directly. The previous version of this post is here:

<https://www.forecr.io/blogs/bsp-development/changing-storage-of-the-root-file-system-emmc-to-m-2-ssd>

The advantage of this version is the boot up speed. In the previous version, the file system into the SSD mounted with a service after the file system into the eMMC mounted. This process increases the boot up time. In this post, the file system into the SSD mounted directly without any service. On the other hand, the Jetson module won't boot up without SSD until the extlinux.conf file changed. To avoid this problem, you can backup your Jetson module before changing the root file system. First, connect your M.2 compatible storage device to M.2 connector and connect the basic interfaces (Ethernet, HDMI, keyboard, mouse) then power on.

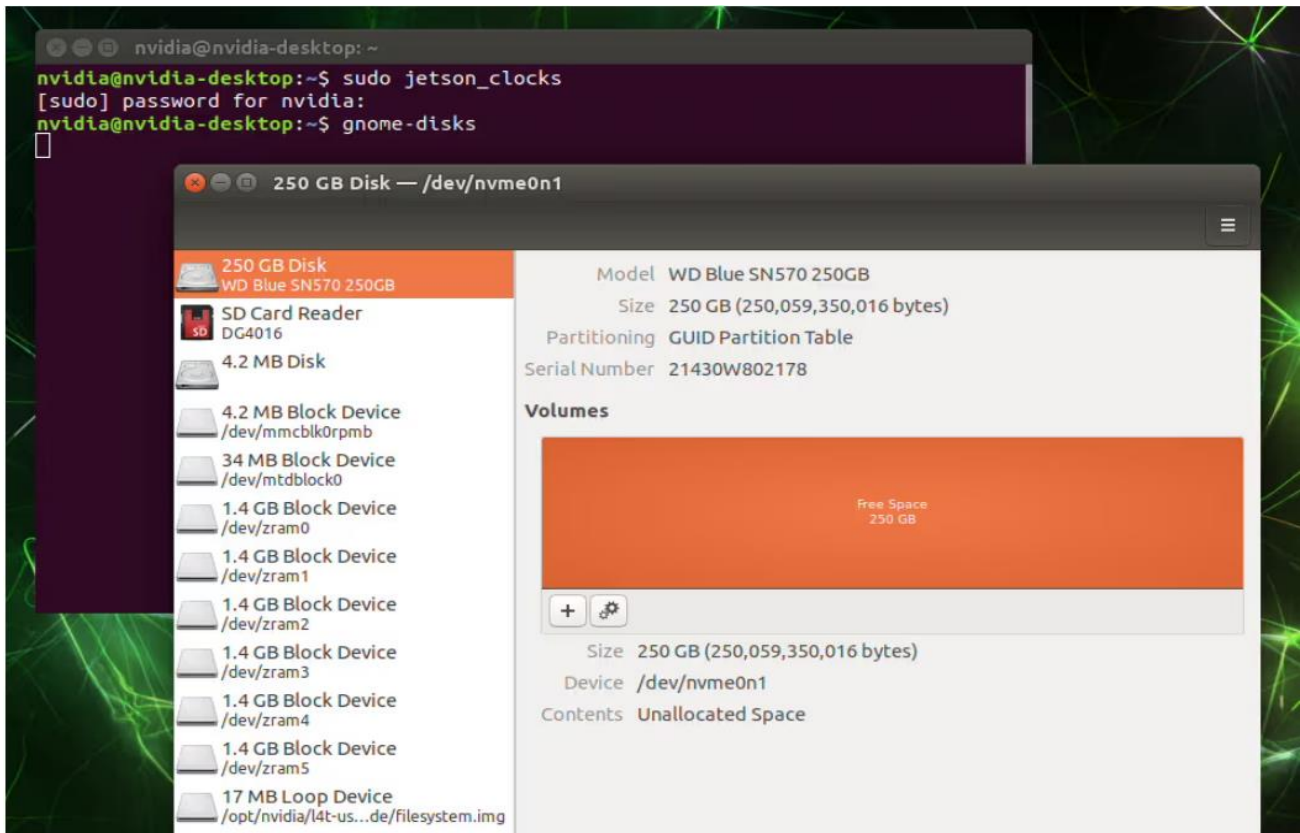
Open a terminal and type these commands below:

```
sudo jetson_clocks  
gnome-disks
```

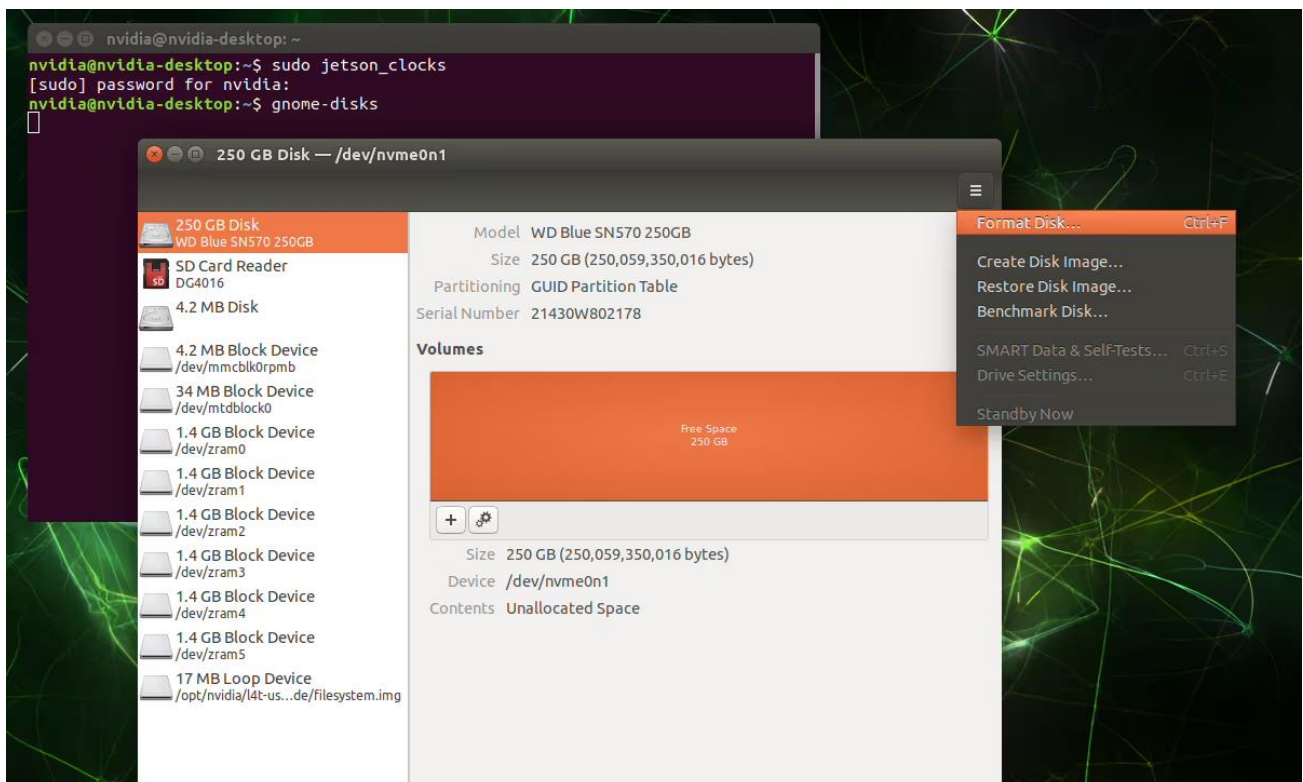


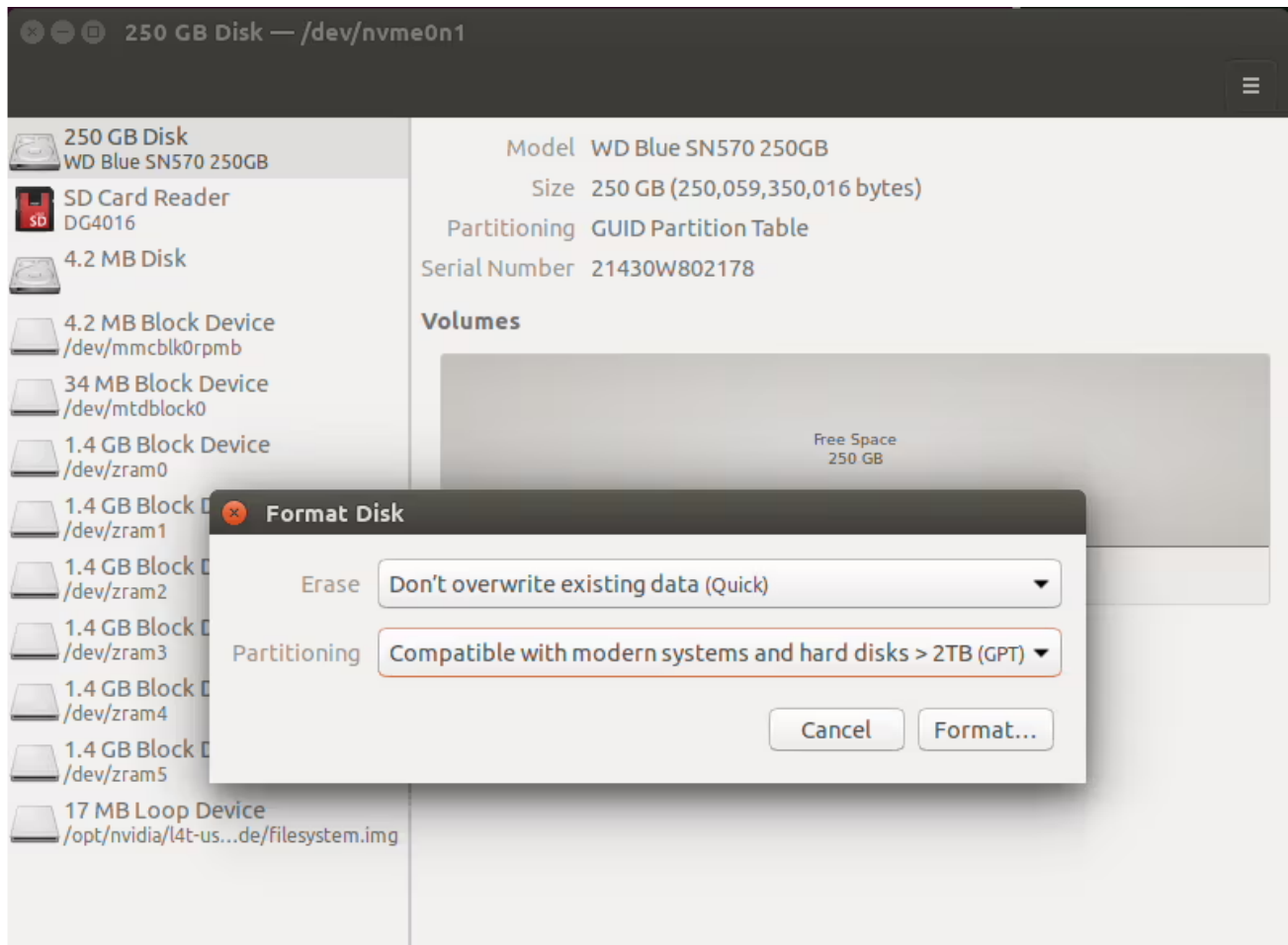
```
nvidia@nvidia-desktop: ~  
nvidia@nvidia-desktop:~$ sudo jetson_clocks  
[sudo] password for nvidia:  
nvidia@nvidia-desktop:~$
```

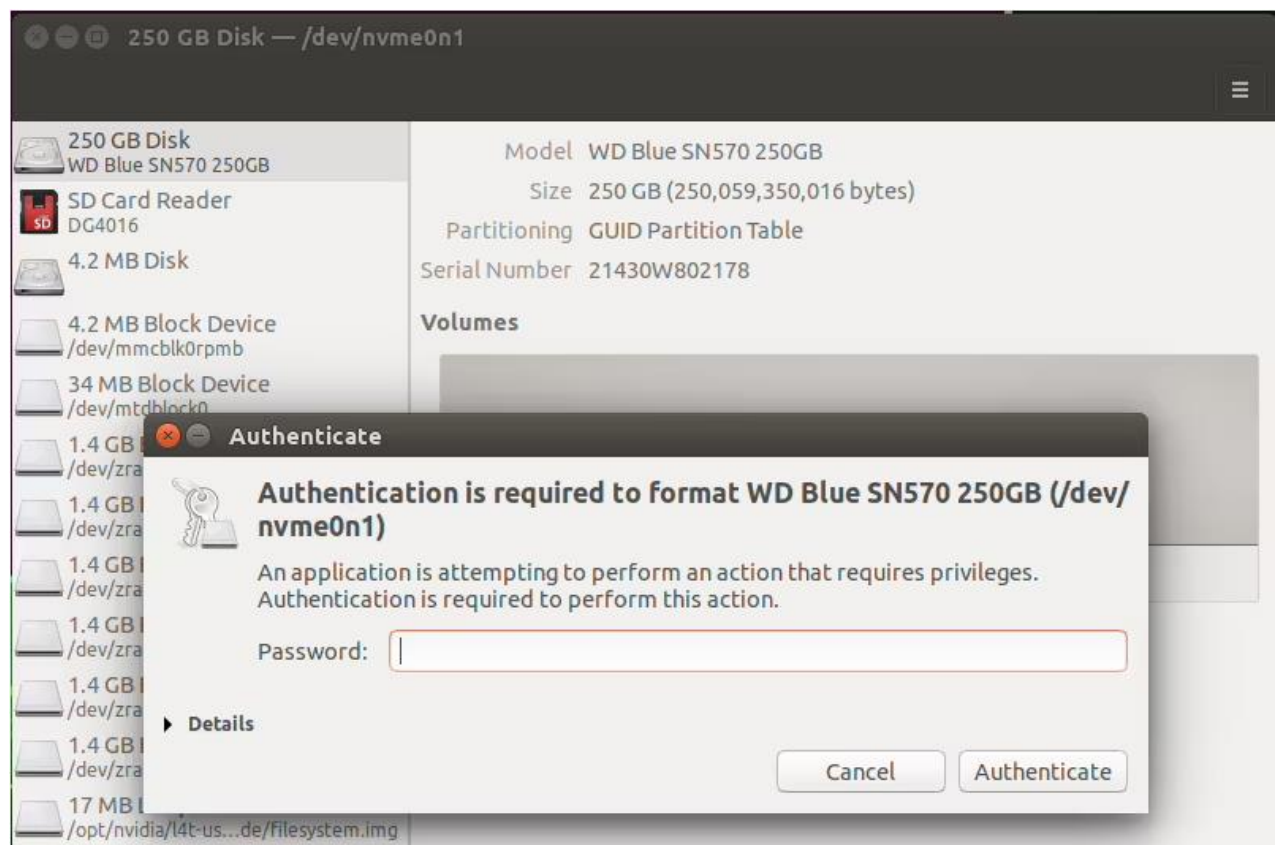
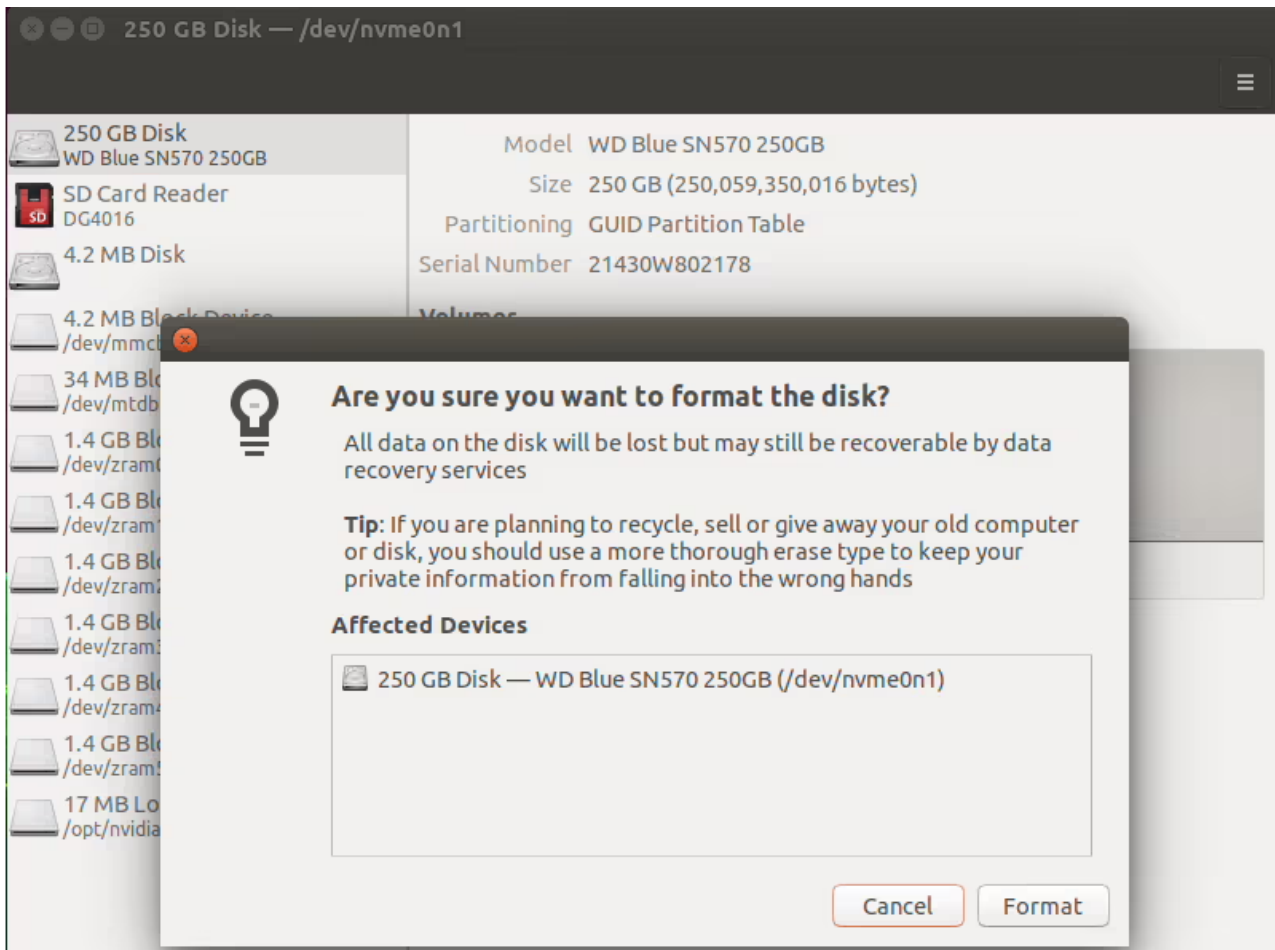
The first command allows the Jetson module's whole sources to use. The next command opens GNOME Disks application below.



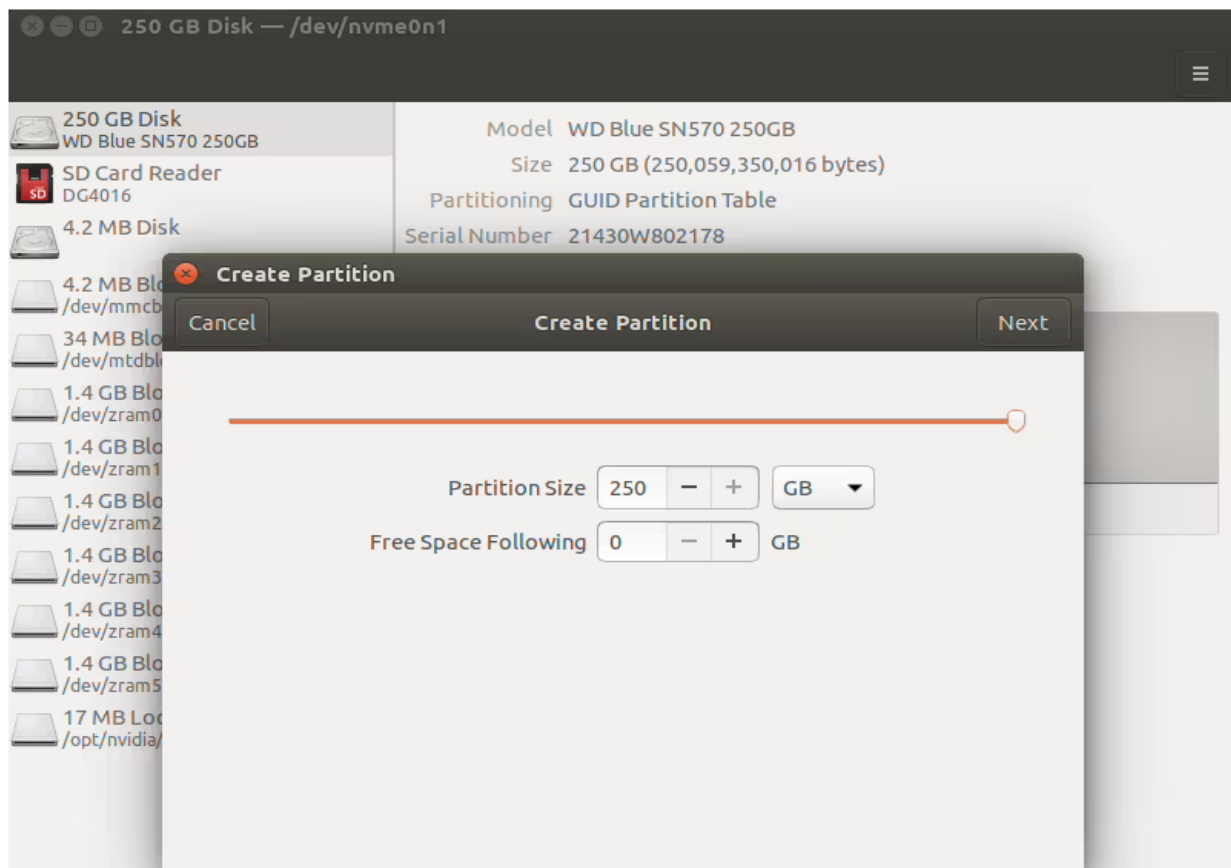
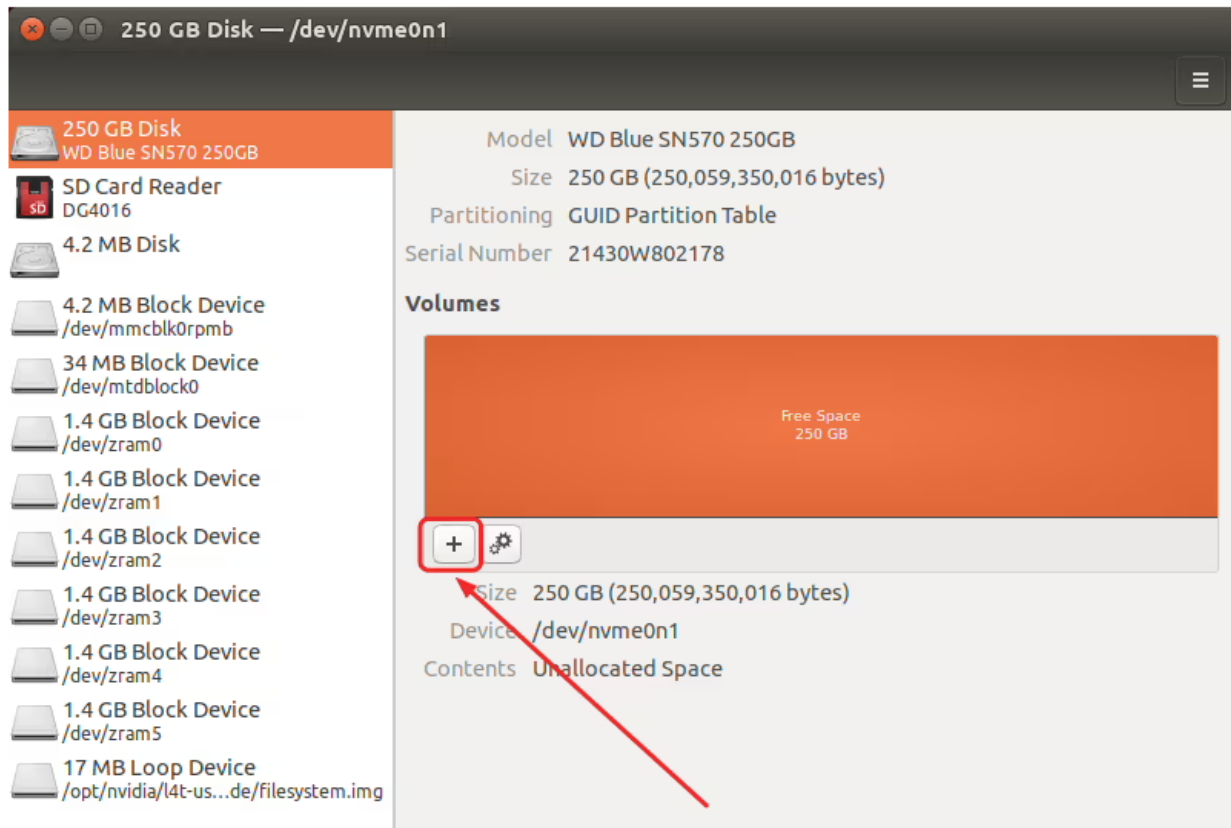
Format the whole disk before creating the storage.



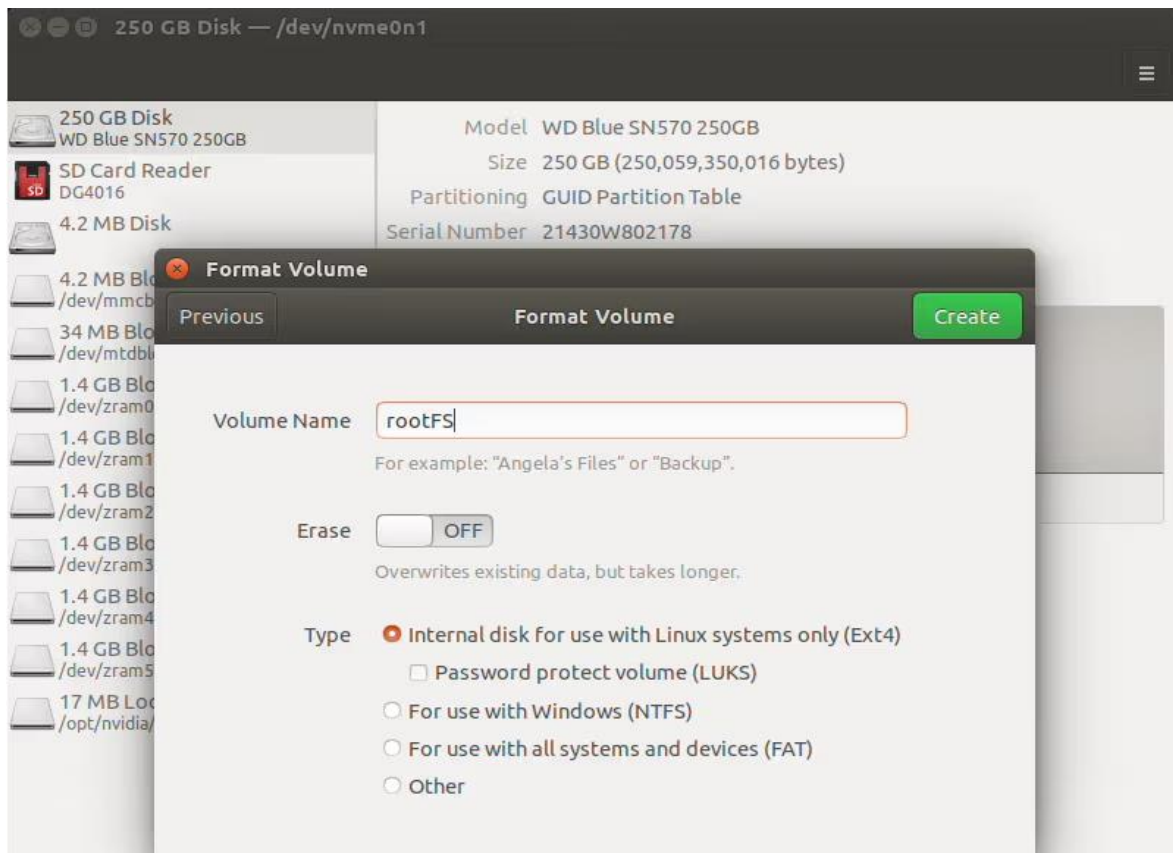




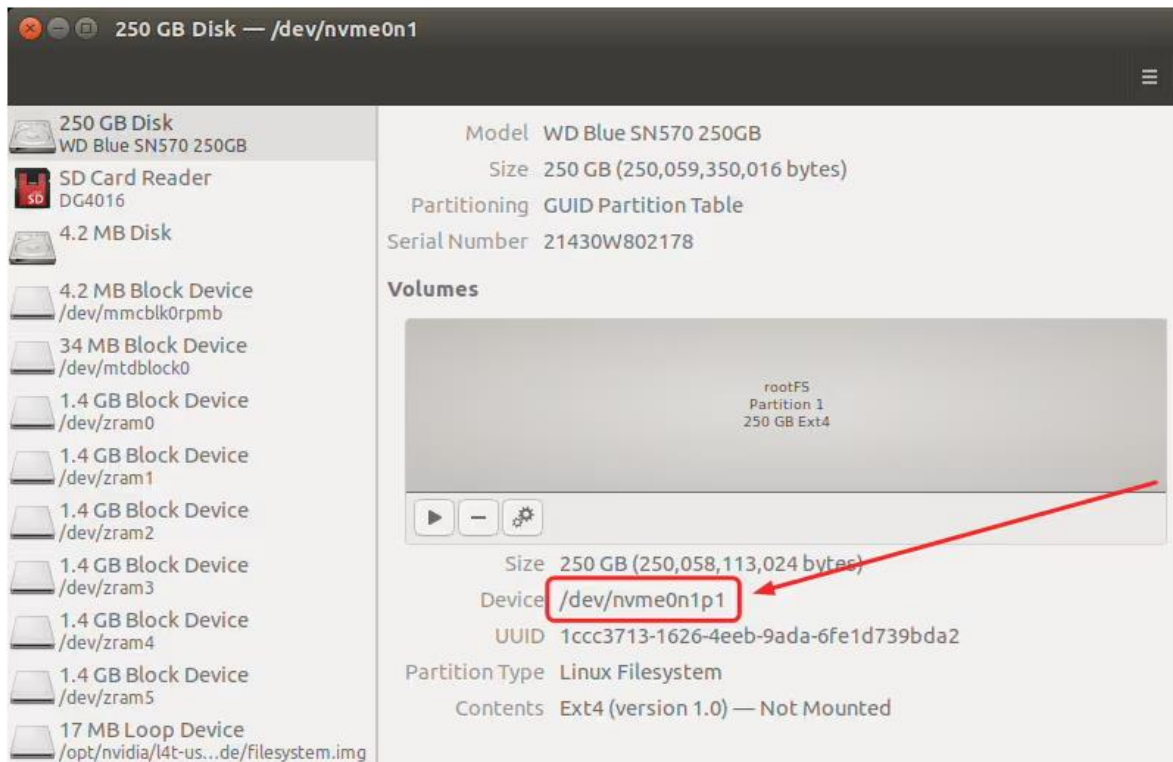
Then, create a new partition from SSD storage.



Format the disk as ext4 format (partition size is up to you but must be min current file system's size).



After creating the partition, check it's name (/dev/nvme0n1p1).



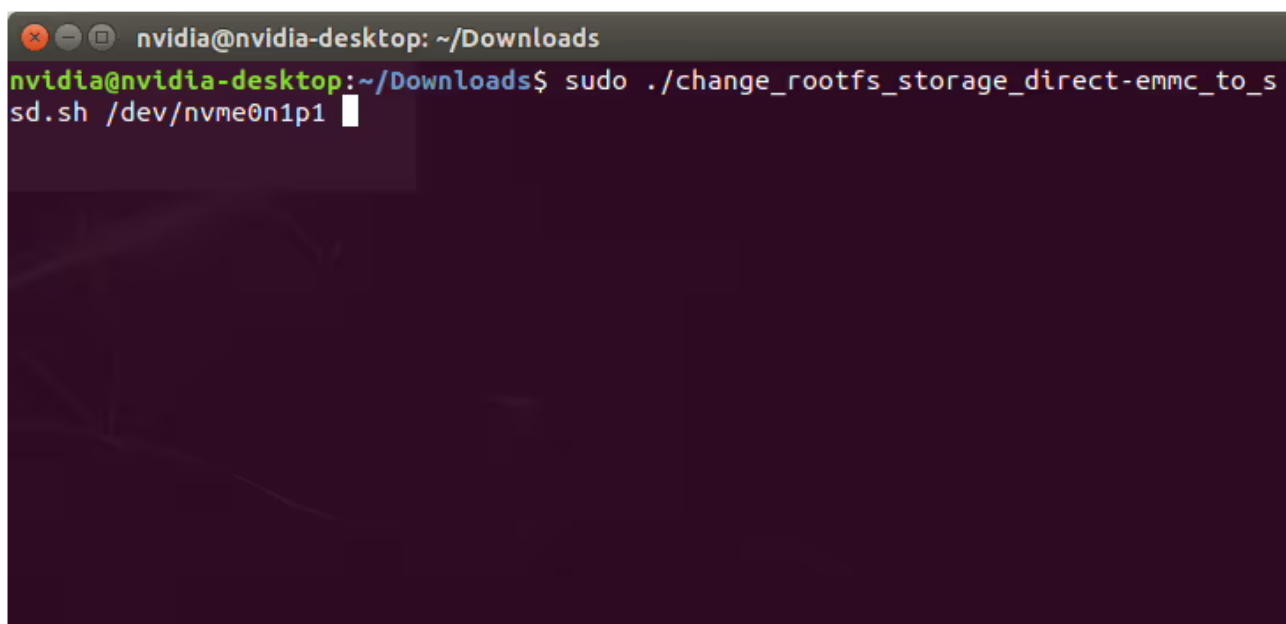
Copying the Root File System

Download the script file from here and extract it. Then, run it with this command below:

```
sudo ./change_rootfs_storage_direct-emmc_to_ssd.sh {EXTERNAL_STORAGE}
```

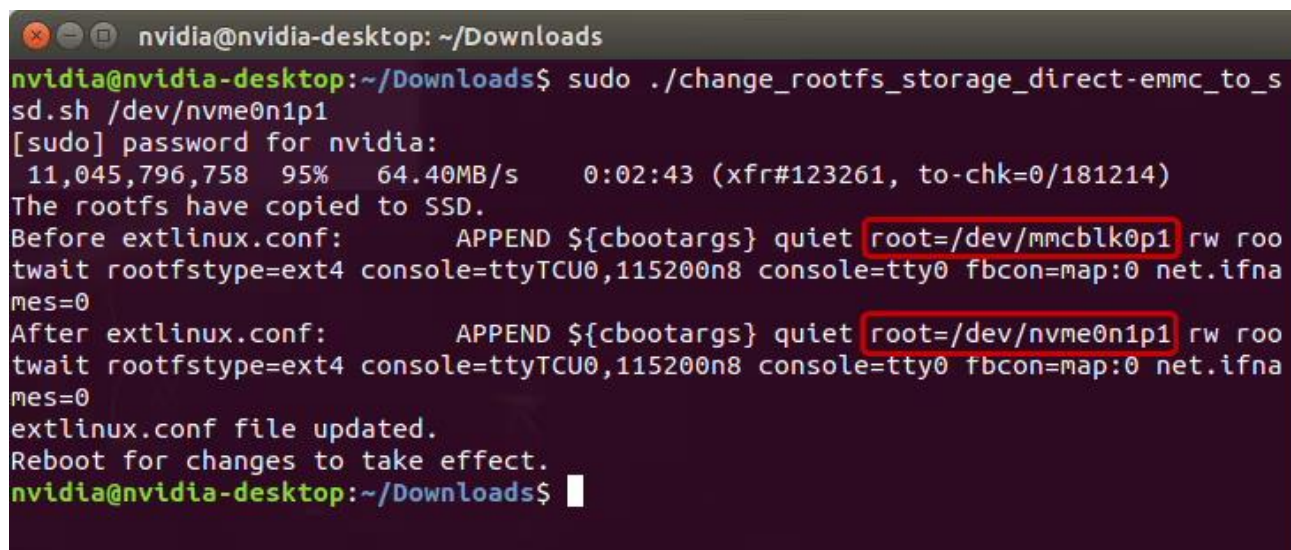
In our setup, we typed this command below:

```
sudo ./change_rootfs_storage_direct-emmc_to_ssd.sh /dev/nvme0n1p1
```



```
nvidia@nvidia-desktop: ~/Downloads
nvidia@nvidia-desktop:~/Downloads$ sudo ./change_rootfs_storage_direct-emmc_to_ssd.sh /dev/nvme0n1p1
```

A few times later, the whole file system copied and the root path changed.



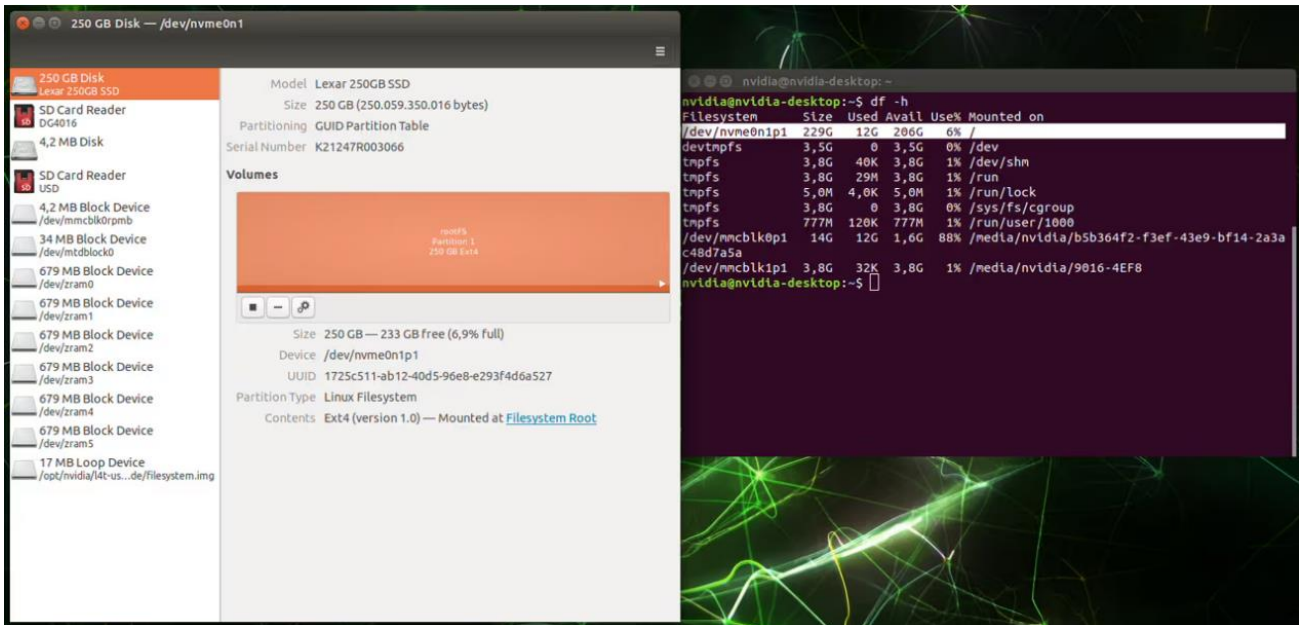
```
nvidia@nvidia-desktop: ~/Downloads
nvidia@nvidia-desktop:~/Downloads$ sudo ./change_rootfs_storage_direct-emmc_to_ssd.sh /dev/nvme0n1p1
[sudo] password for nvidia:
 11,045,796,758 95% 64.40MB/s 0:02:43 (xfr#123261, to-chk=0/181214)
The rootfs have copied to SSD.
Before extlinux.conf: APPEND ${cbootargs} quiet root=/dev/mmcblk0p1 rw rootwait rootfstype=ext4 console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0
After extlinux.conf: APPEND ${cbootargs} quiet root=/dev/nvme0n1p1 rw rootwait rootfstype=ext4 console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0
extlinux.conf file updated.
Reboot for changes to take effect.
nvidia@nvidia-desktop:~/Downloads$
```

It's time to reboot the Jetson module. Reboot it and check the Root File System copied successfully.

Assignment of the Root File System

Open a terminal and type this command to check the root mounted from SSD below:

```
df -h
```



After rebooting you can see that the new storage is assigned as root file system.