

2NOR01 Quick Start

Table of Contents

i.	Overview	2
ii.	Put 2NOR01 config and dtb file from the host computer (Ubuntu 20.04.6)	2
	1. Download & Install NVIDIA SDK Manager	2
	2. Flash the NVIDIA operating software (use JetPack 6.1)	2
	(1) Switch 2NOR01 into Force Recovery Mode	2
	(2) Open SDK Manager and Flash OS	3
	3. Replace 2NOR01 's config and dtb file to Linux_for_Tegra	5
	(1) 2NOR01's p3767.conf.common	5
	(2) 2NOR01's DTB File	5
	4. Flash OS again through command(because need enable eth2)	6
iii.	2NOR01 enable JETSON IO(SPI1/SPI3) and imx219	6
iv.	How to Test	7
	1. TPM	7
	2. SPI (spidev)	7
	3. i2c1 / i2c2 (use PW407)	7
	4. CANBUS	8
	5. CA1	8
	6. MIPI Camera - CAM1-IMX219 and CAM2-IMX219	8
	7. Ethernet(I226IT)	9

i. Overview

The following describes two methods for adding the DTB and configuring the settings of 2NOR01 on NVIDIA Jetson Orin OS.

Note: When using the second method, eth2 will not work due to the absence of the "p3767.conf.common" file.

Once you have completed the first two items, you can use the 'How to Test' function.

ii. Put 2NOR01 config and dtb file from the host computer (Ubuntu 20.04.6)

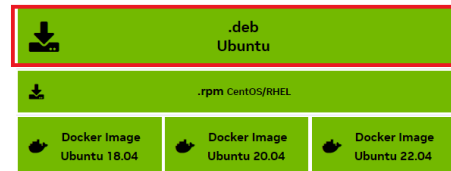
1. Download & Install NVIDIA SDK Manager
<https://developer.nvidia.com/sdk-manager>

SDK Manager

Everything You Need to Set Up Your Development Environment

NVIDIA SDK Manager provides an end-to-end development environment setup solution for NVIDIA's Jetson, Holoscan, Rivermax, DeepStream, Aerial Research Cloud (ARC-OTA), Ethernet Switch, RAPIDS, DRIVE and DOCA SDKs for both host and target devices.

Download NVIDIA SDK Manager 2.1.0



[SDK Manager User Guide](#)

2. Flash the NVIDIA operating software (use JetPack 6.1)
 - (1) Switch 2NOR01 into Force Recovery ModeConnect your Linux host computer to the appropriate USB port on your 2NOR01 (USB typeC)

For 2NOR01:

- 1.Ensure that your 2NOR01 is powered off, and a 32GB or larger NVME card is inserted .
- 2.Enable Force Recovery Mode by placing a jumper across SWP1 , located on the edge of the carrier board up the Jetson module.
- 3.Connect the 2NOR01's power adapter . The 2NOR01 powers on automatically and enters Force Recovery mode.
- 4.Remove the jumper from SWP1

For other 2NOR01 that have hardware buttons:

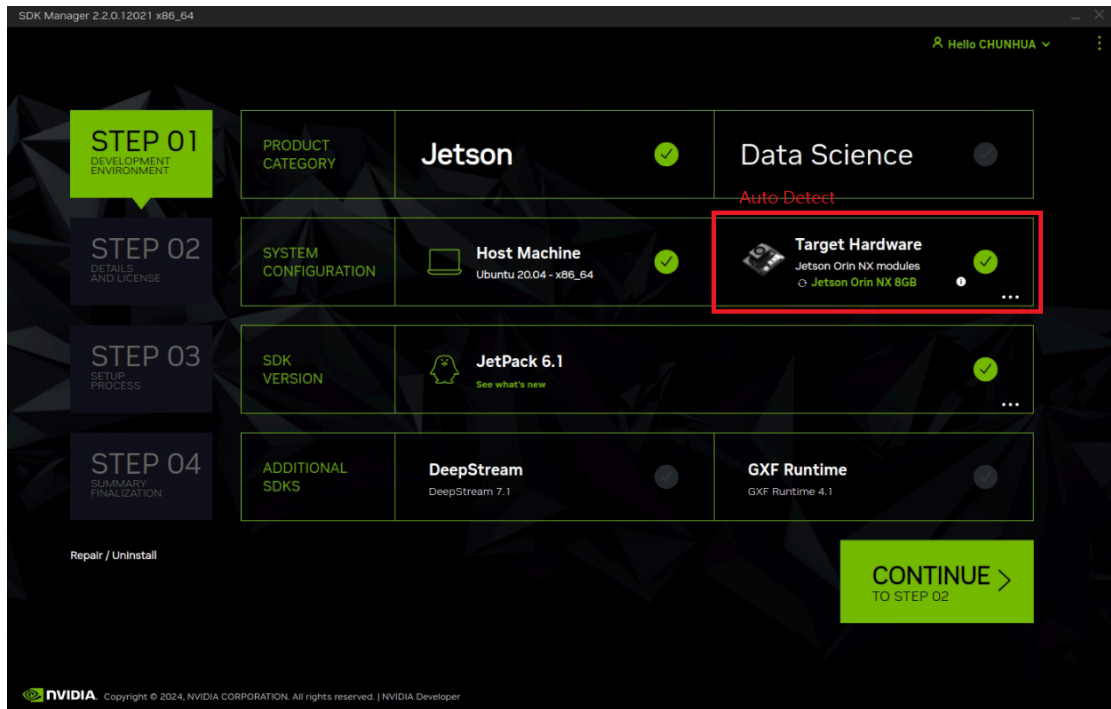
1. Ensure that the developer kit is powered off.
2. Press and hold down the Force Recovery button.
3. Press, then release the Power button.
4. Release the Force Recovery button.

Note: To Determine Whether the Developer Kit Is in Force Recovery Mode
Open a terminal window on your host computer and enter command" lsusb" .
The Jetson module is in Force Recovery Mode if you see the message:
For example: Orin NX 8GB

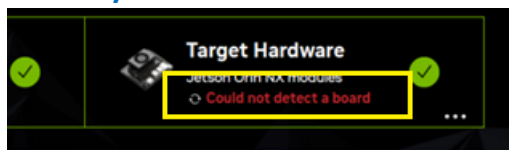
Bus 001 Device 058: ID 0955:7423 NVIDIA Corp

- (2) Open SDK Manager and Flash OS

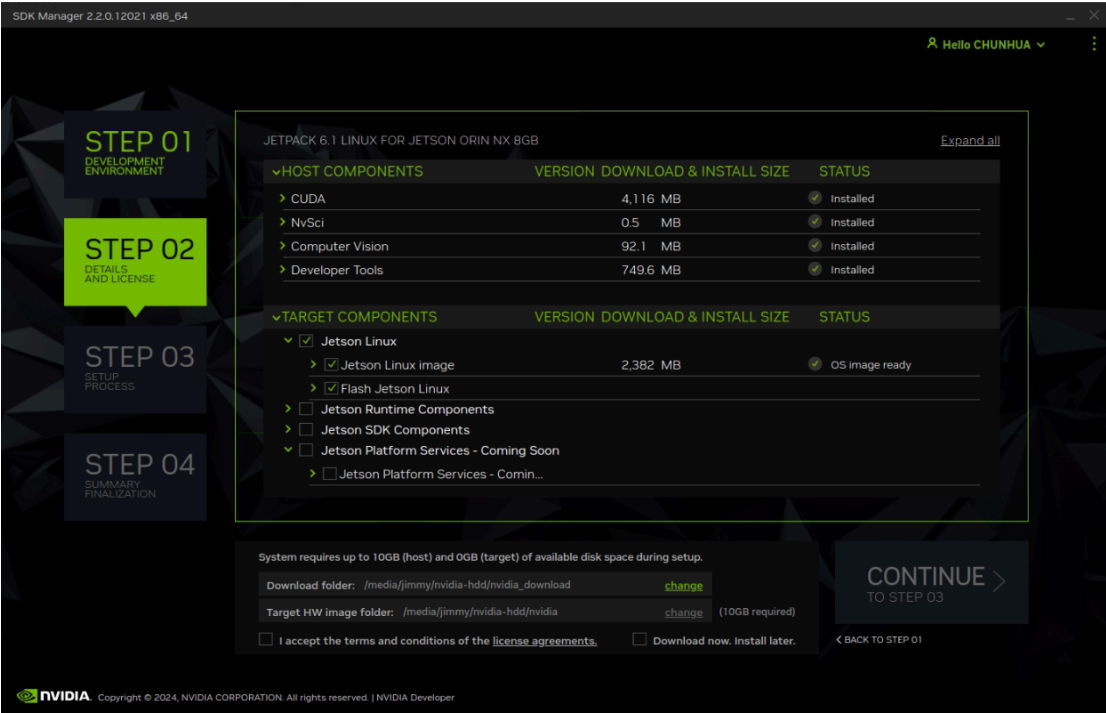
■ STEP 01:



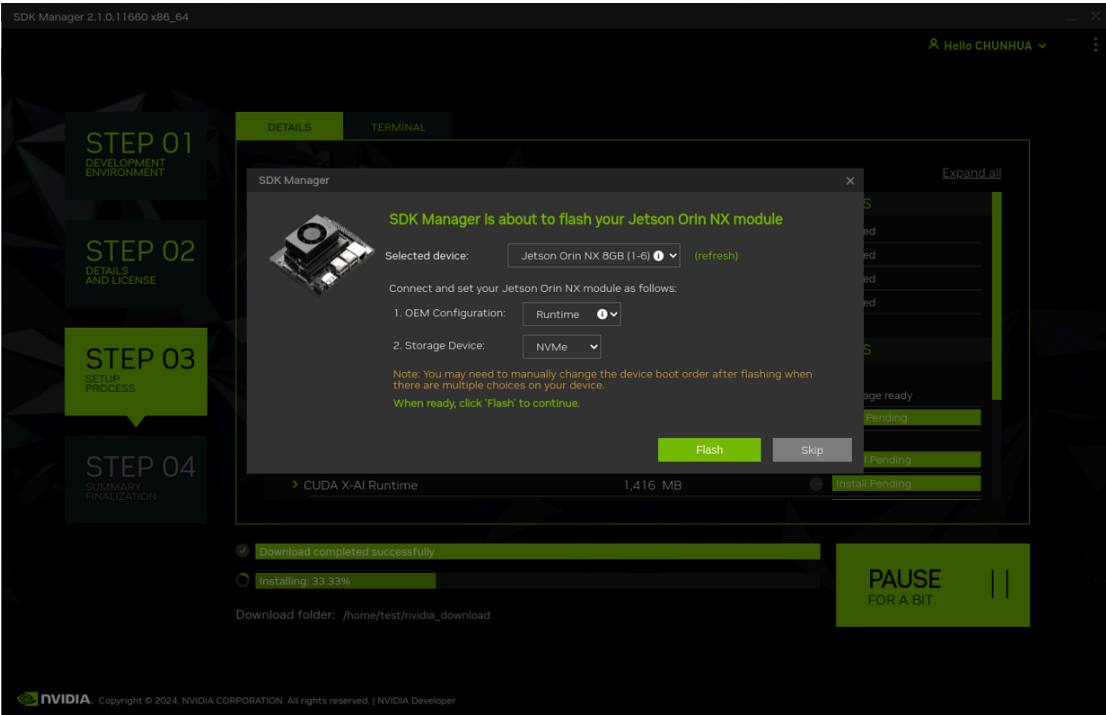
Note: If the host is unable to recognize the CPU board via USB, a message "Could not detect a board" will be displayed. At this time, you can reconnect and disconnect the USB cable to check if the device is in Force Recovery Mode.

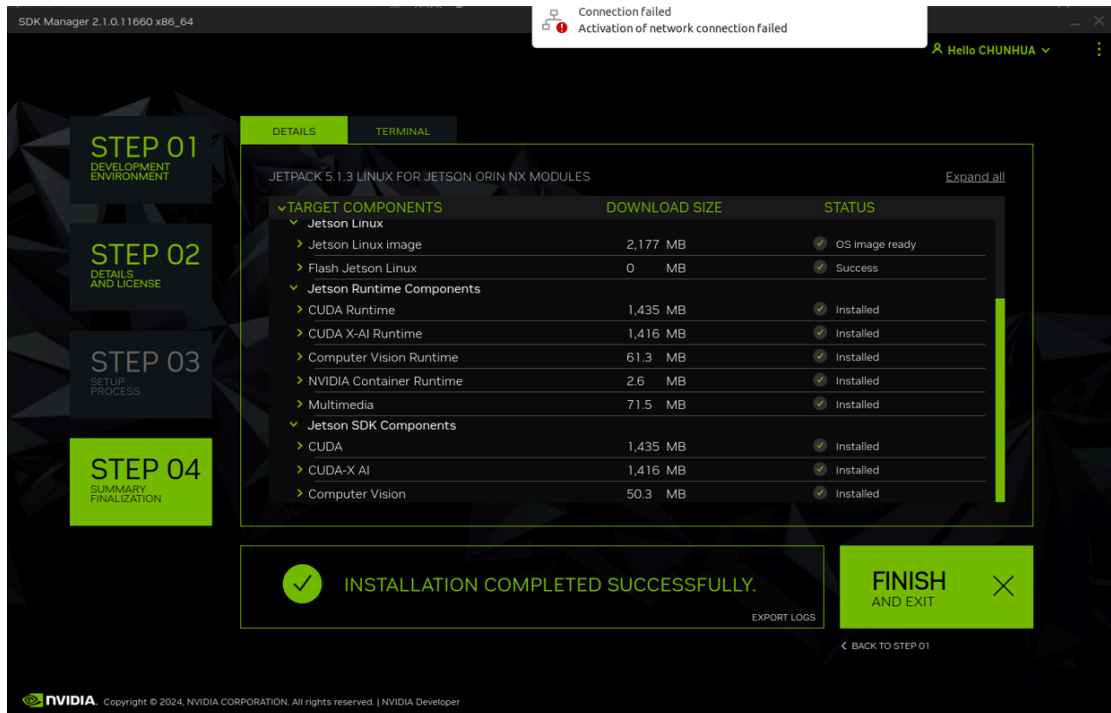


- STEP 02: Uncheck Jetson Runtime Components



STEP 03: Flash





3. Replace 2NOR01 's config and dtb file to Linux_for_Tegra

(1) 2NOR01's p3767.conf.common

Replace 2NOR01 's p3767.conf.common to Linux_for_Tegra.

This setting is related to eth2 (I226IT)

2NOR01 Orin NANO8G p3767.conf.common/tegra234-p3768-0000+p3767-0003-nv.dtb

file download:https://drive.google.com/drive/folders/15cDonyXOVnYp3MLH7UmKw_kLrPxTY2t1?usp=sharing

2NOR01 Orin NX8G p3767.conf.common/tegra234-p3768-0000+p3767-0001-nv.dtb

file download:https://drive.google.com/drive/folders/1LMtvf_yBSlpkIve_G7xMRwb0-DiC3jn?usp=sharing

2NOR01 Orin NX16G p3767.conf.common/tegra234-p3768-0000+p3767-0000-nv.dtb

file download:https://drive.google.com/drive/folders/1cZ7IzEpMvfpIKK_vv1oJ-77-C4VEsR34?usp=sharing

For example :

```
cp -v p3767.conf.common
/$HOME/JetPack_6.1_Linux_JETSON_ORIN_NANO_TARGETS/Linux_for_Tegra/p3
767.conf.common
```

(2) 2NOR01's DTB File

For Orin Nano example :

```
cp -v tegra234-p3768-0000+p3767-0003-nv.dtb
/$HOME/JetPack_6.1_Linux_JETSON_ORIN_NANO_TARGETS/Linux_for_Tegra/ke
rnel/dtb
```

For Orin NX example :

```
cp -v tegra234-p3768-0000+p3767-0001-nv.dtb
/$HOME/JetPack_6.1_Linux_JETSON_ORIN_NX_TARGETS/Linux_for_Tegra/kernel
/dtb
```

For Orin NX 16G example :

```
cp -v tegra234-p3768-0000+p3767-0000-nv.dtb
/$(HOME)/JetPack_6.1_Linux_JETSON_ORIN_NX_TARGETS/Linux_for_Tegra/kernel
/dtb
```

4. Flash OS again through command(because need enable eth2)
Switch 2NOR01 into Force Recovery Mode again, use l4t_initrd_flash.sh to flash OS

For example : (Orin Nano / Orin NX)

```
sudo ./tools/kernel_flash/l4t_initrd_flash.sh --external-device nvme0n1p1 -c
tools/kernel_flash/flash_l4t_t234_nvme.xml -p " -c
bootloader/generic/cfg/flash_t234_qspi.xml" --showlogs --network usb0
jetson-orin-nano-devkit internal
```

- iii. 2NOR01 enable JETSON IO(SPI1/SPI3) and imx219

File download

imx219 dtbo: https://drive.google.com/drive/folders/1tLID6uMHvMp2_gSo9QImMcdjPdartSXI?usp=sharing

Copy tegra234-p3767-camera-p3768-imx219-dual.dtbo to /boot

Modify /boot/extlinux/extlinux.conf

For example:

```
1 TIMEOUT 30
2 #DEFAULT primary
3 #DEFAULT lex
4
5 MENU TITLE L4T boot options
6
7 LABEL primary
8   MENU LABEL primary kernel
9   LINUX /boot/Image
10  INITRD /boot/initrd
11  APPEND ${cbootargs} root=PARTUUID=9ffa4421-e684-4bc6-8bb0-b2ce4217fe26 rw rootwait rootfstype=ext4 mminit_loglevel=4 console=ttyTCU0,115200 firmware_class.path=/etc/firmware fbcon=map:0
    nospectre_bhb_videofb=off console=tty0
12
13 # When testing a custom kernel, it is recommended that you create a backup of
14 # the original kernel and add a new entry to this file so that the device can
15 # fallback to the original kernel. To do this:
16 #
17 # 1, Make a backup of the original kernel
18 #     sudo cp /boot/Image /boot/Image.backup
19 #
20 # 2, Copy your custom kernel into /boot/Image
21 #
22 # 3, Uncomment below menu setting lines for the original kernel
23 #
24 # 4, Reboot
25
26 # LABEL backup
27 #   MENU LABEL backup kernel
28 #   LINUX /boot/Image.backup
29 #   INITRD /boot/initrd
30 #   APPEND ${cbootargs}
31
32 LABEL lex
33   MENU LABEL lex kernel
34   LINUX /boot/Image
35   INITRD /boot/initrd
36   FDT /boot/tegra234-p3768-0000+p3767-0001-nv.dtb
37   APPEND ${cbootargs} root=PARTUUID=9ffa4421-e684-4bc6-8bb0-b2ce4217fe26 rw rootwait rootfstype=ext4 mminit_loglevel=4 console=ttyTCU0,115200 firmware_class.path=/etc/firmware fbcon=map:0
    nospectre_bhb_videofb=off console=tty0
38   OVERLAYS /boot/tegra234-p3767-camera-p3768-imx219-dual.dtbo
39
```

- iv. How to Test

Next, we can do other verifications

1. TPM

File download

tpm_driver: <https://drive.google.com/drive/folders/197v8lJbLZQ6Ro6XDxzBDeDN9B3JgJacD?usp=sharing>

```
test@test-desktop:~/Factory_Tools/bin$ sudo insmod tpm_tis_core.ko
test@test-desktop:~/Factory_Tools/bin$ sudo insmod tpm_tis_spi.ko
test@test-desktop:~/Factory_Tools/bin$ ls /dev/tpm*
/dev/tpm0  /dev/tpmrm0
test@test-desktop:~/Factory_Tools/bin$
```

2. SPI (spidev)

```
test@test-desktop:~/Factory_Tools$ sudo modprobe spidev
[sudo] password for test:
test@test-desktop:~/Factory_Tools$ ls /dev/spidev*
/dev/spidev0.0  /dev/spidev0.1
test@test-desktop:~/Factory_Tools$
```

3. i2c1 / i2c2 (use PW407)

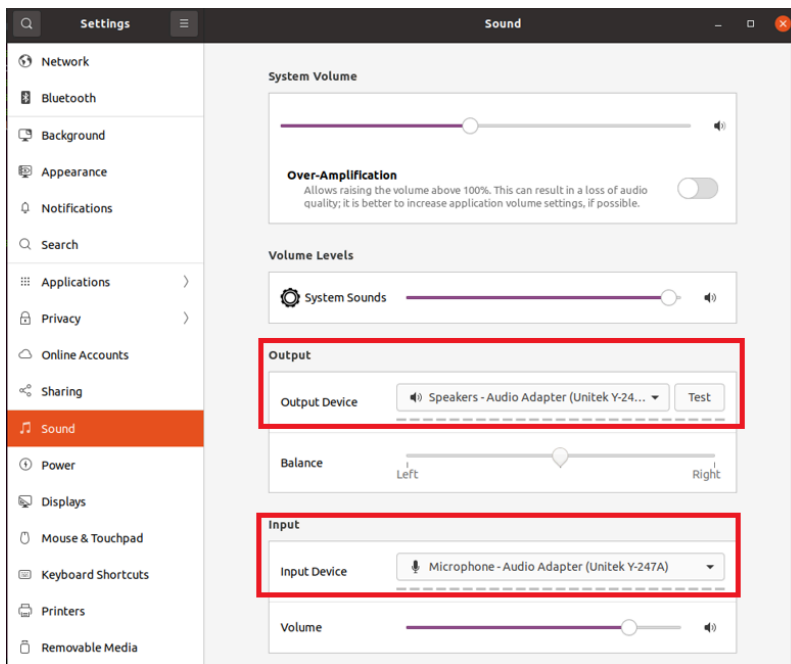
```
root@test-desktop:/home/test# i2cdetect -y 7
Warning: Can't use SMBus Quick Write command, will skip some addresses
  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:
10:
20:
30: -- -- -- -- -- -- -- 37
40:
50: -- -- -- -- -- -- -- -- -- -- -- 5f
60:
70:
root@test-desktop:/home/test# i2cdetect -y 1
Warning: Can't use SMBus Quick Write command, will skip some addresses
  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:
10:
20:
30: -- -- -- -- -- -- --
40:
50: -- -- -- -- -- -- -- -- -- -- -- 5f
60:
70:
root@test-desktop:/home/test#
```

4. CANBUS

```
test@test-desktop:~/Factory_Tools/bin$ sudo modprobe can
[sudo] password for test:
test@test-desktop:~/Factory_Tools/bin$ sudo modprobe can_raw
test@test-desktop:~/Factory_Tools/bin$ sudo modprobe mttcan
test@test-desktop:~/Factory_Tools/bin$ ifconfig can0
can0: flags=128<NOARP> mtu 16
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 10 (UNSPEC)
    RX packets 2 bytes 16 (16.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 1 overruns 0 carrier 1 collisions 0
    device interrupt 113

test@test-desktop:~/Factory_Tools/bin$
```

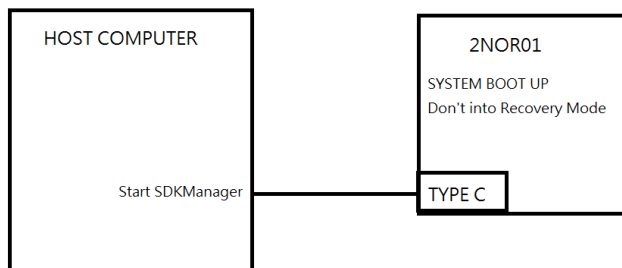
5. CA1



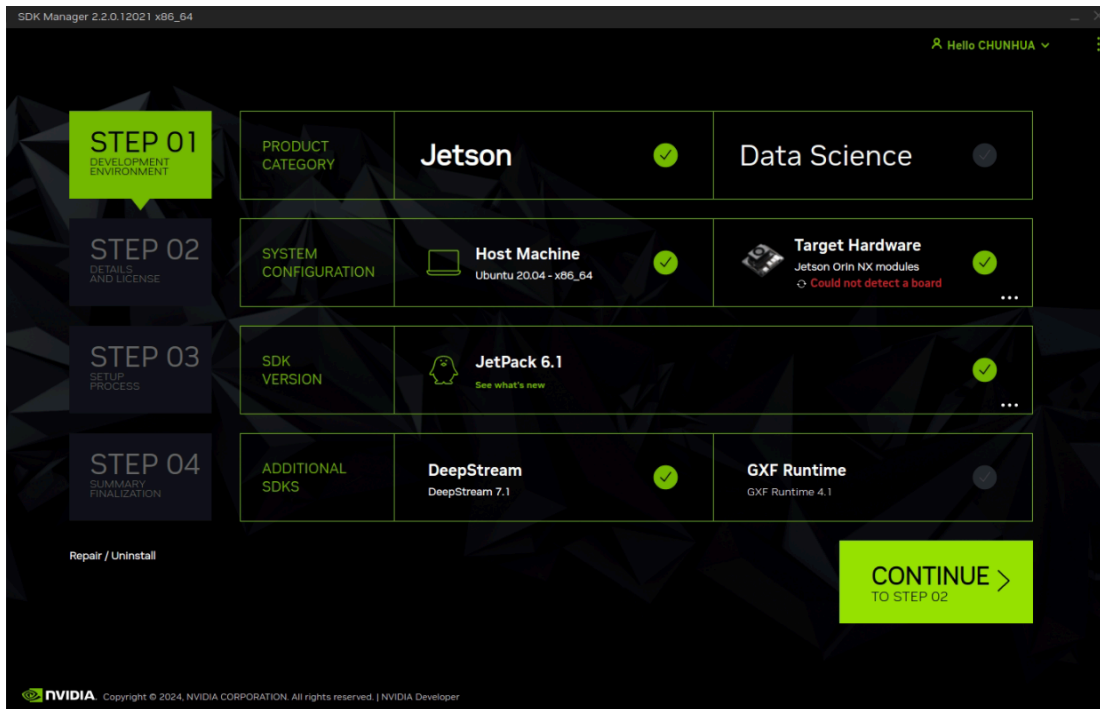
6. MIPI Camera - CAM1-IMX219 and CAM2-IMX219

```
root@test-desktop:/home/test# gst-launch-1.0 nvarguscamerasrc sensor-id=0 ! 'video/x-raw(memory:NVMM),width=(int)1280,height=(int)720,format=(string)NV12,framerate=(fraction)30/1' ! nvvidconv ! queue ! xvimagesink
WARNING: erroneous pipeline: no element "nvarguscamerasrc"
root@test-desktop:/home/test# gst-launch-1.0 nvarguscamerasrc sensor-id=1 ! 'video/x-raw(memory:NVMM),width=(int)1280,height=(int)720,format=(string)NV12,framerate=(fraction)30/1' ! nvvidconv ! queue ! xvimagesink
WARNING: erroneous pipeline: no element "nvarguscamerasrc"
```

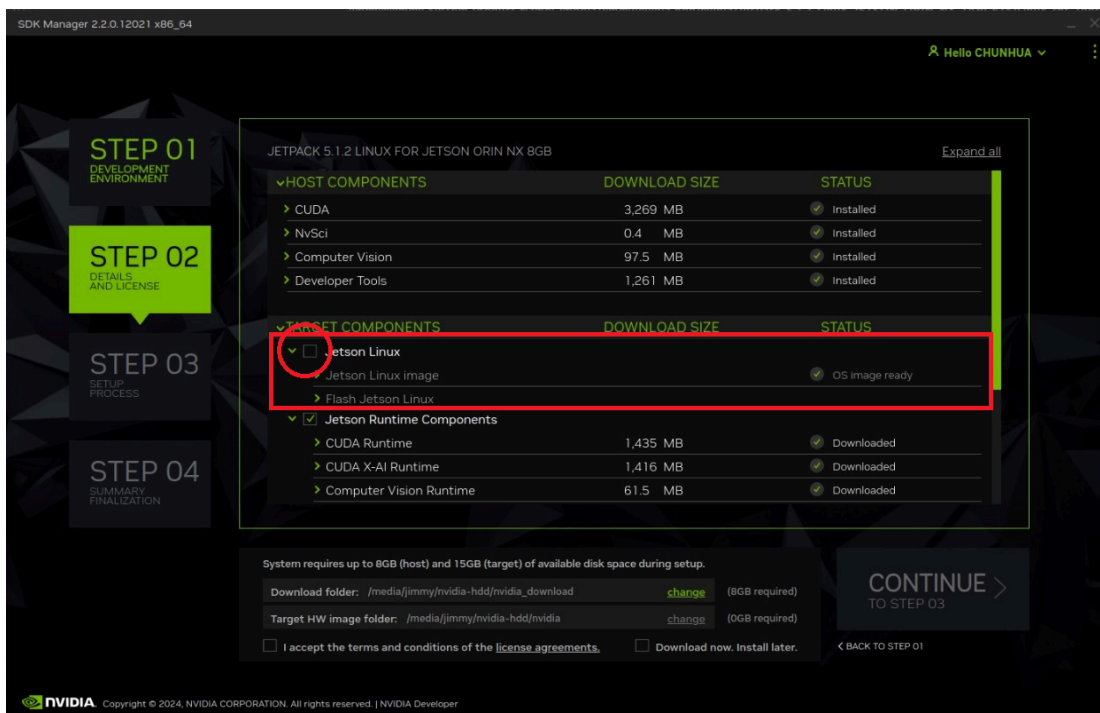
If a warning appears when you execute mipi camera, you must install the jetson runtime component because some packages are missing.



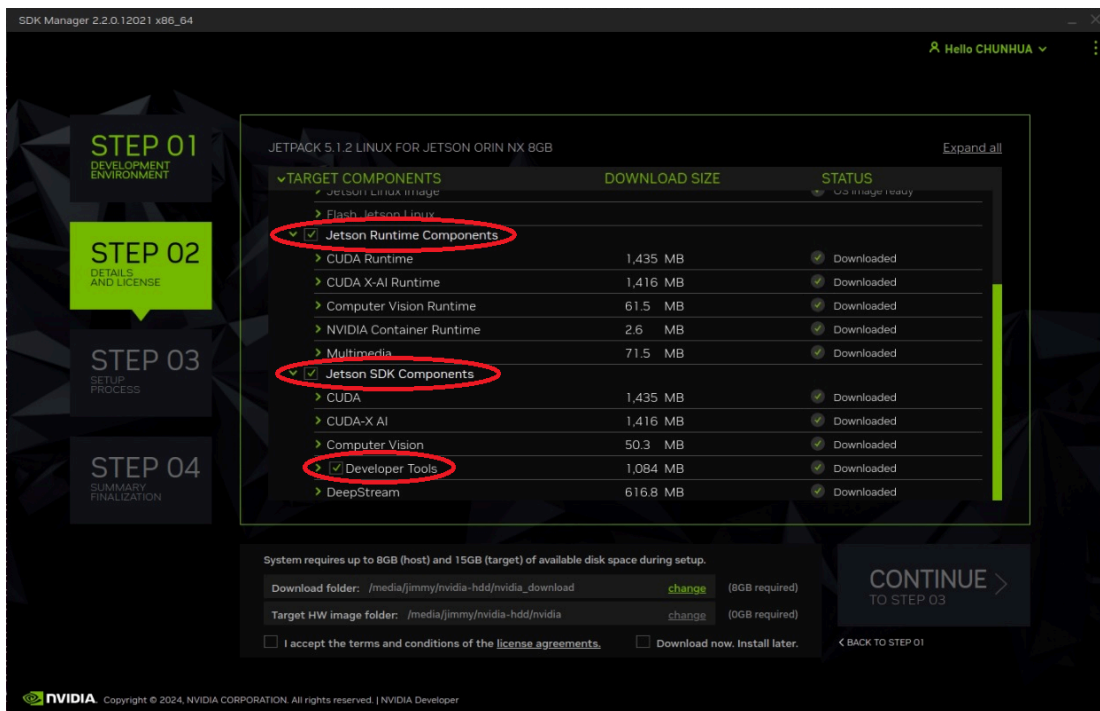
Use sdkmanager install DeepStream



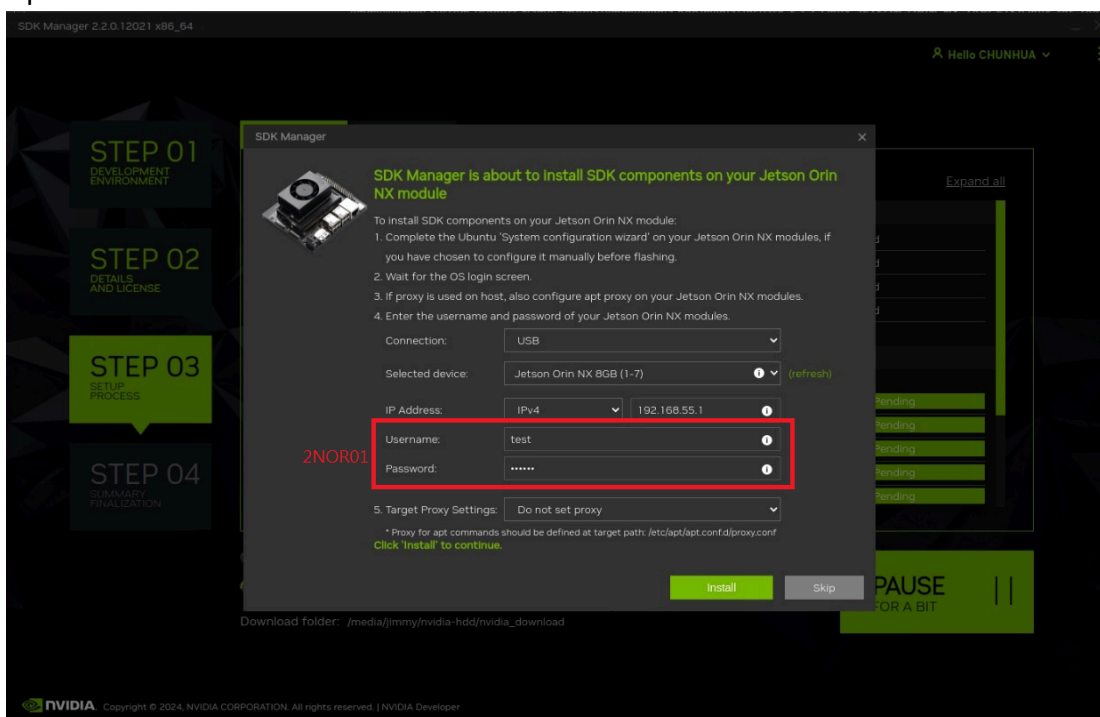
Cancel mark Jetson Linux



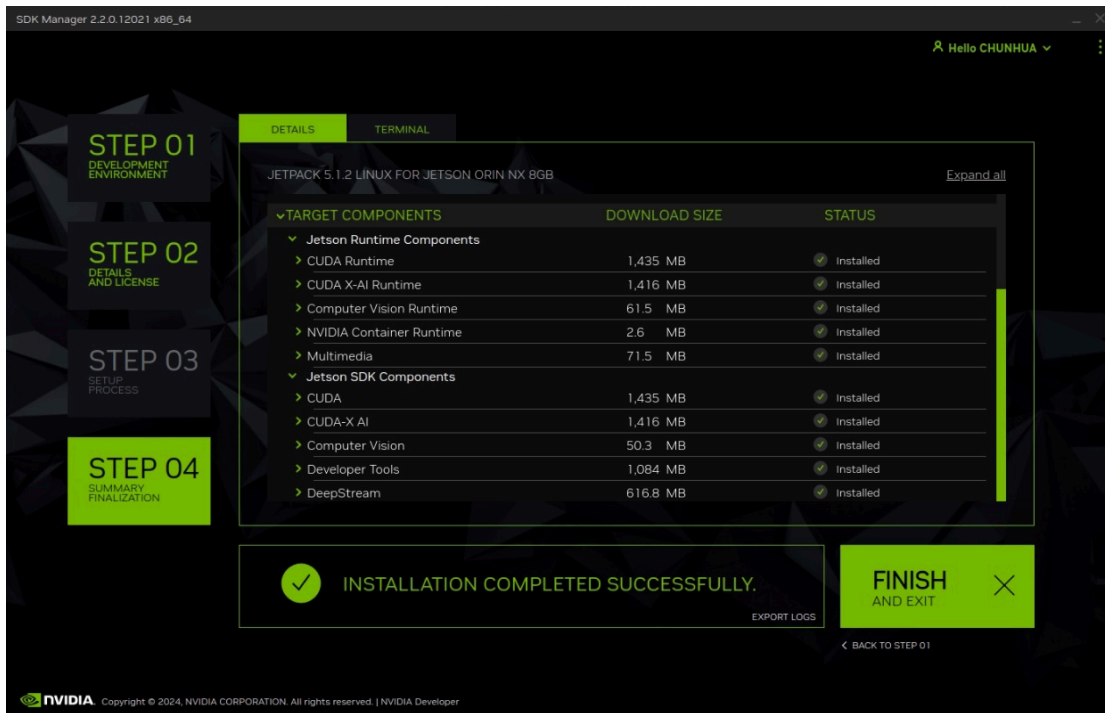
choice Jetson Runtime Components/Jetson SDK Components/Developer Tools



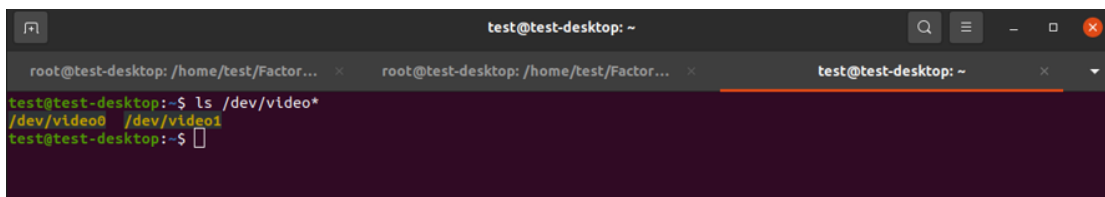
input 2NOR01 username and Password



Install Completed Successfully.



Check MIPI node



Play Command:

7. Ethernet(I226IT)
Install driver

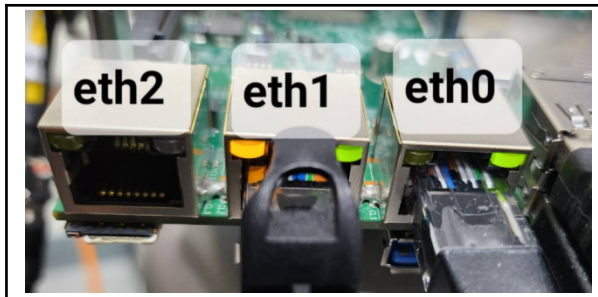
```

test@test-desktop:~$ sudo insmod igc.ko
[sudo] password for test:
test@test-desktop:~$ ifconfig
enP7p1s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 00:a0:c9:00:00:00 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device memory 0x322800000-32280ffff
enP8p1s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 48:b0:2d:ea:ef:3a txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 254 base 0xd000
enP9p1s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.1.111.182 netmask 255.255.255.0 broadcast 10.1.111.255
    inet6 fe80::39fa:2e6a:32ce:680b prefixlen 64 scopeid 0x20<link>
    ether 00:a0:c9:00:00:00 txqueuelen 1000 (Ethernet)
    RX packets 44 bytes 5023 (5.0 KB)
    RX errors 0 dropped 20 overruns 0 frame 0
    TX packets 51 bytes 6165 (6.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device memory 0x328000000-32800ffff

```

Remark:

eth2



Note: eth2 needs to modify the “p3767.conf.common” ,so just replacing dtb cannot make eth2 work, must re-flash image again .
[/JetPack_6.1_Linux_JETSON_ORIN_NANO_TARGETS/Linux_for_Tegra/p3767.conf.common](#)

```
ODMDATA="gbe-uphy-config-9,hsstp-lane-map-3,hsio-uphy-config-0";
```