

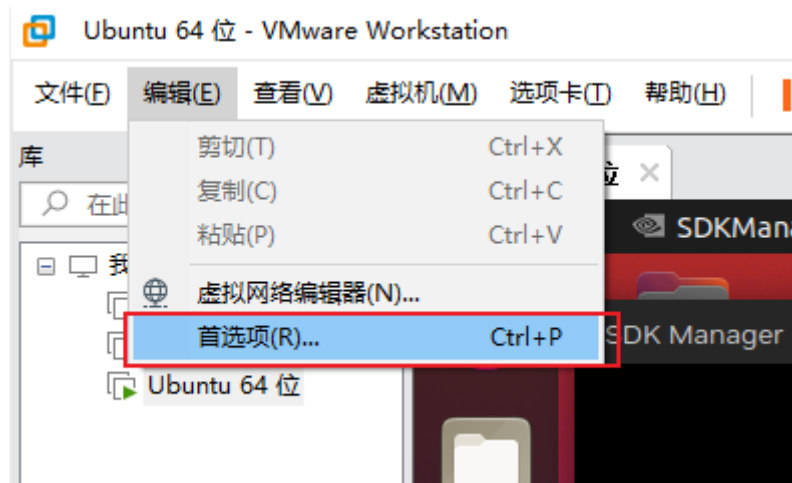
C1902????

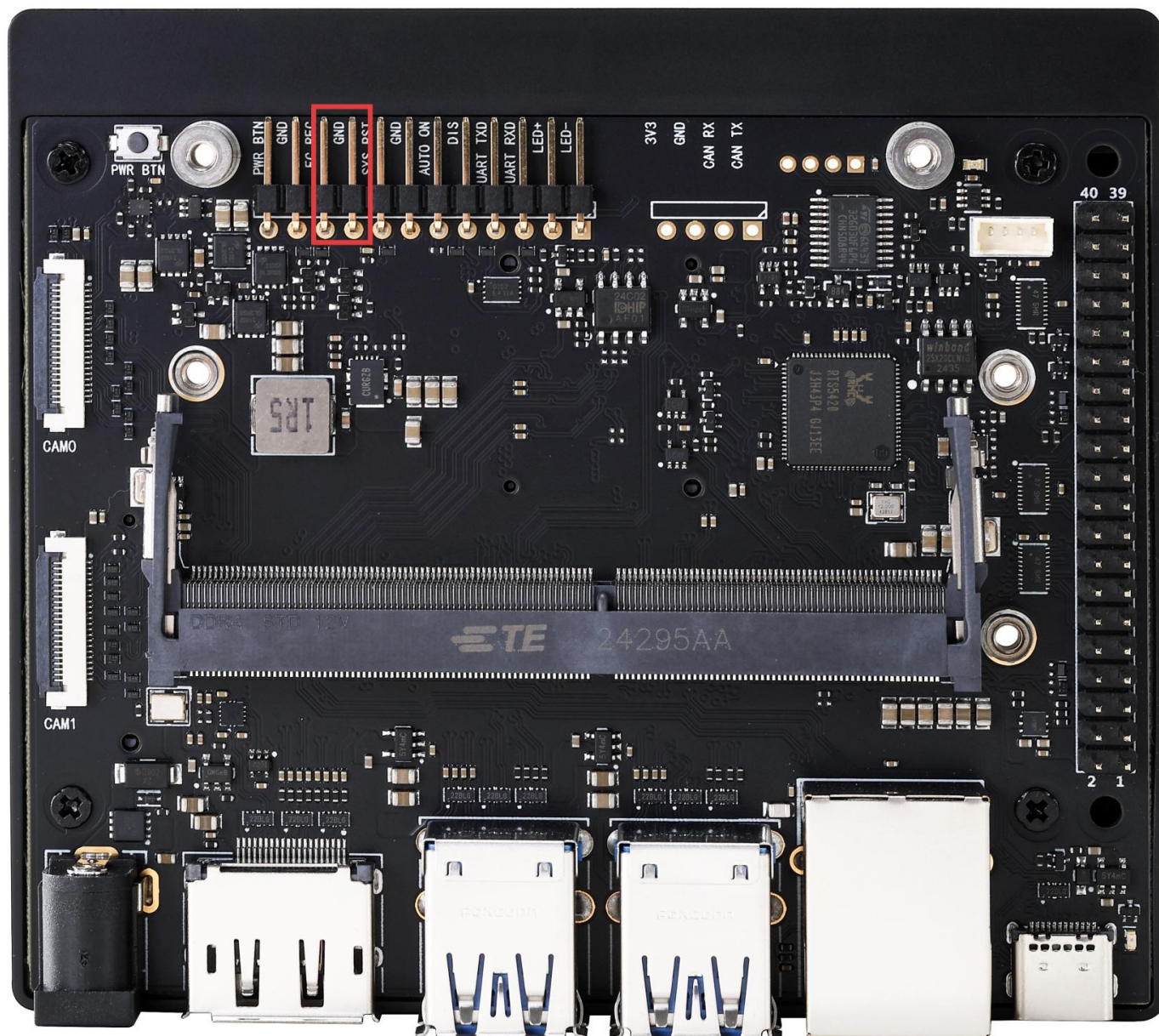


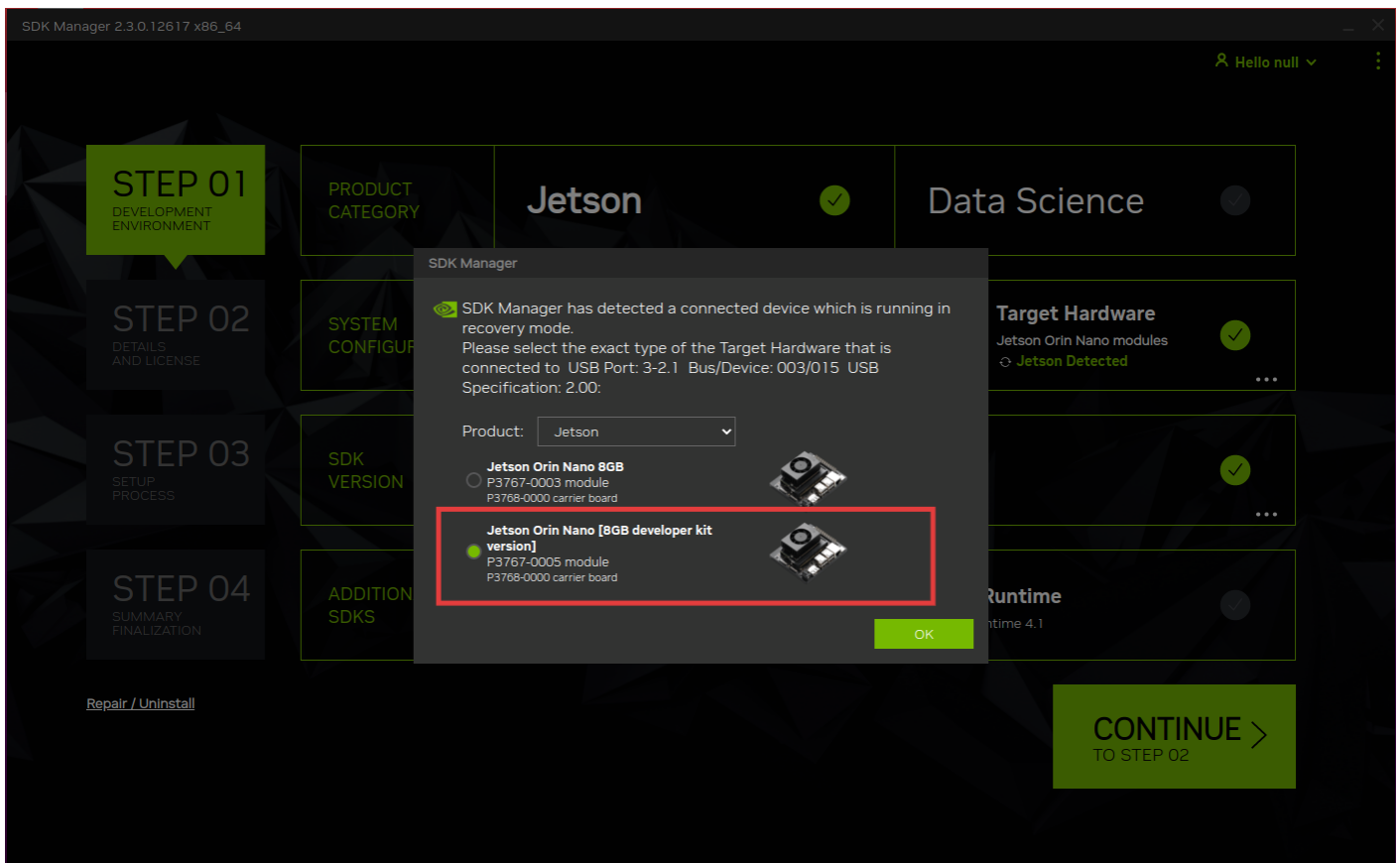
1.??NVIDIA SDK Manager????

1.1 ??VMware?USB??

- USB??

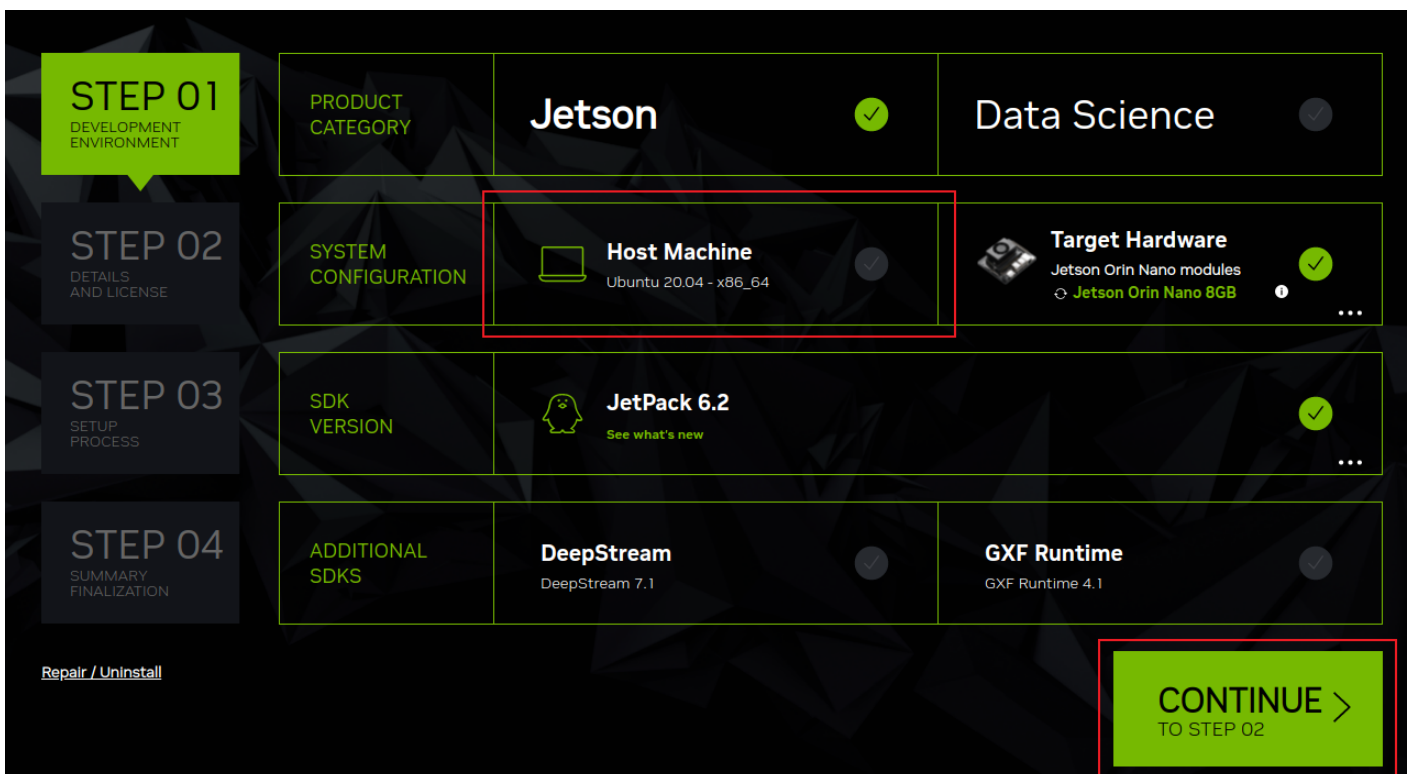




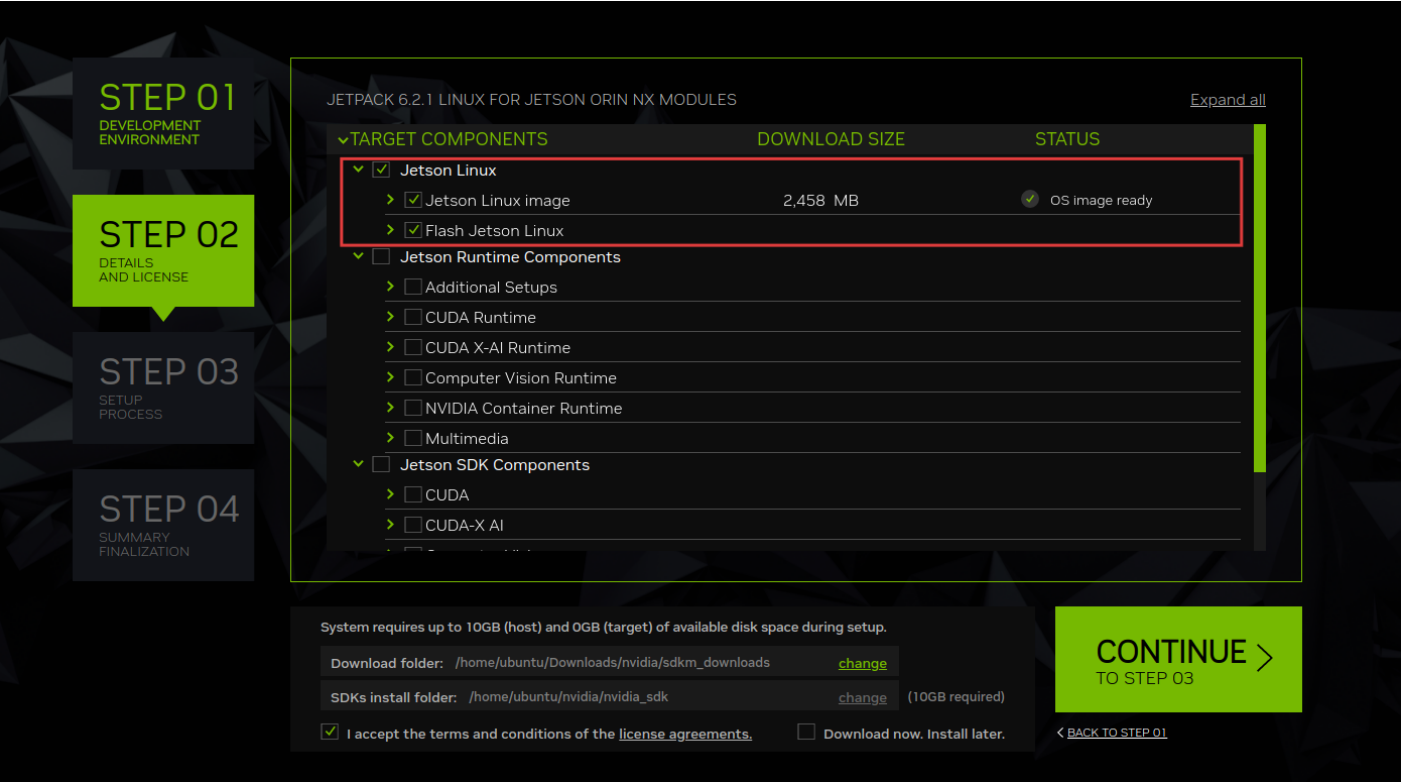


1.3 ??SDK

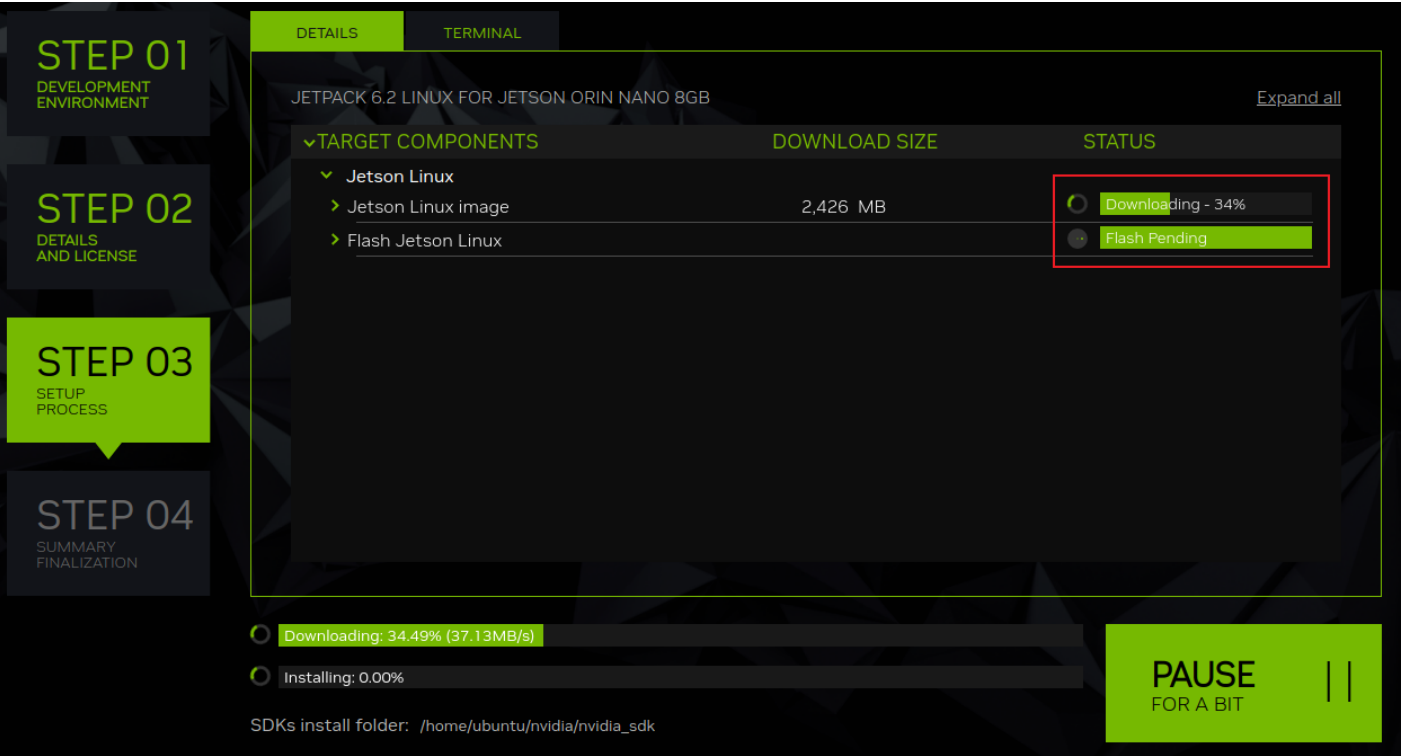
- Host Machine CONTINUE



- Jetson Linux



1.4 ??????



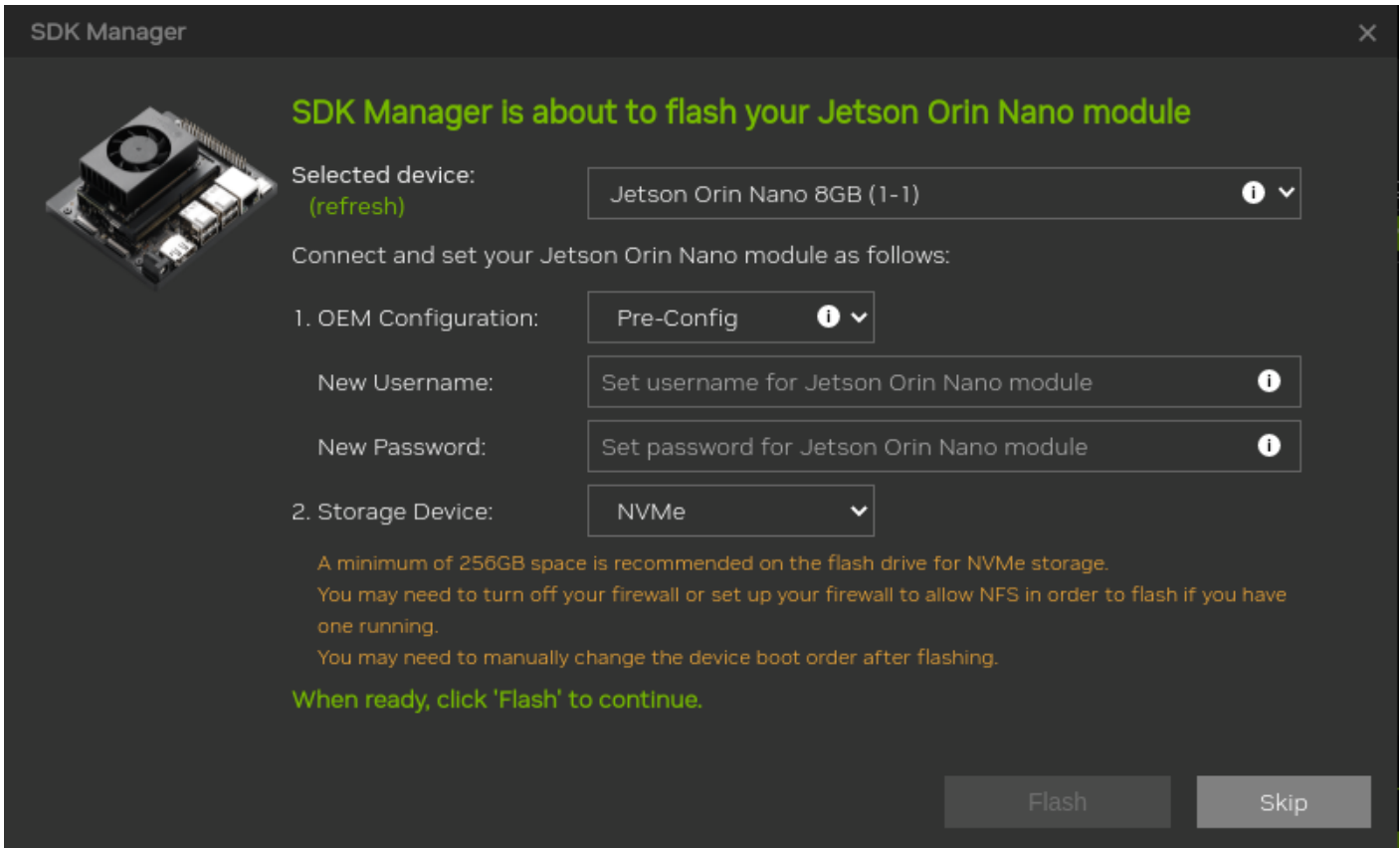
1.5 ??????

Pre-Config■■■■■■■

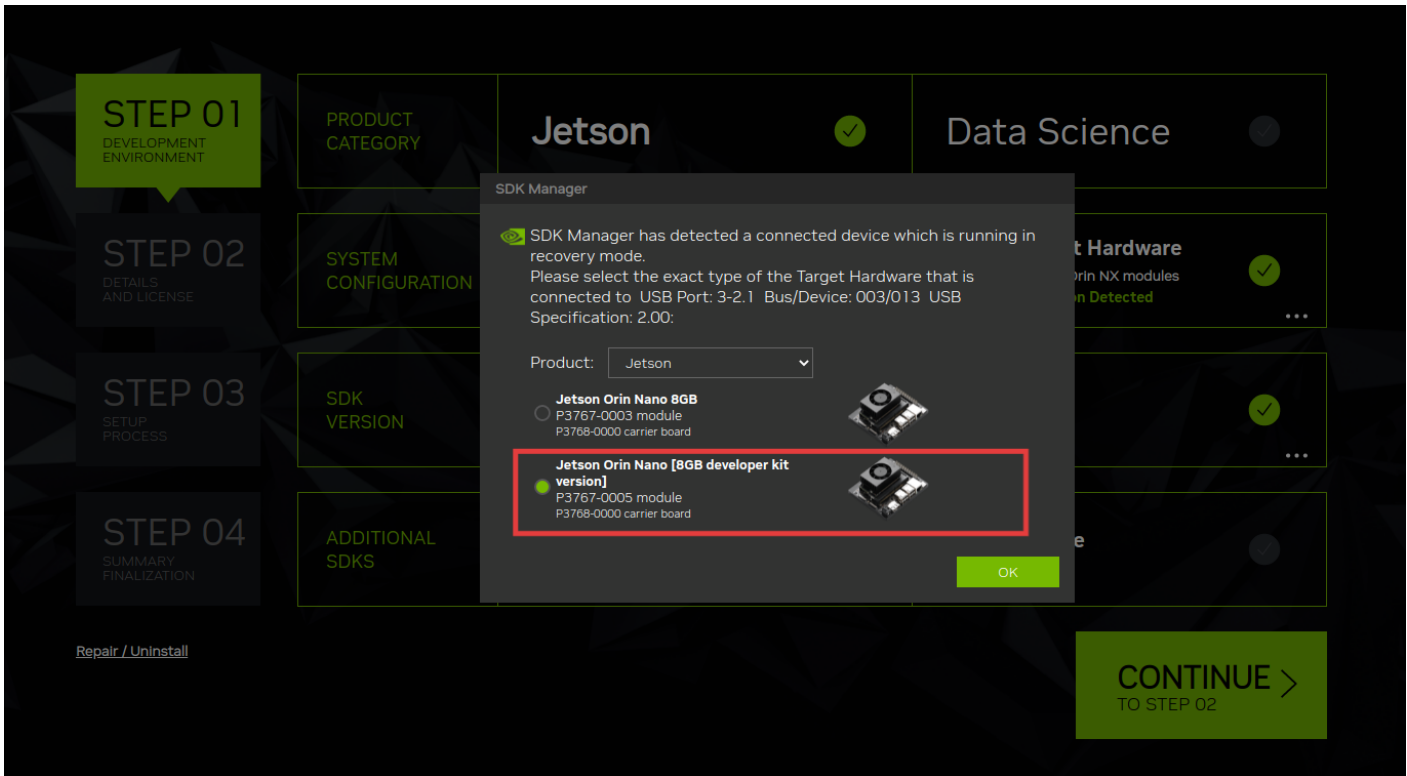
Runtime■■■■■■■■■■

Storage Device■■■■■■■■■■■■■■■■■■■■

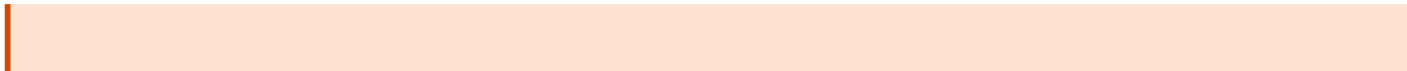
(■■■) /■■■■ /U ■■■)



- ☐ ☐ developer kit version ☐



- ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ 10~20 ☐ ☐ ☐ ☐



STEP 01
DEVELOPMENT
ENVIRONMENT

STEP 02
DETAILS
AND LICENSE

STEP 03
SETUP
PROCESS

STEP 04
SUMMARY
FINALIZATION

DETAILS

TERMINAL

JETPACK 6.2 LINUX FOR JETSON ORIN NANO 8GB

Expand all

✓TARGET COMPONENTS	DOWNLOAD SIZE	STATUS
✓ Jetson Linux		
➤ Jetson Linux image	2,426 MB	✓ OS image ready
➤ Flash Jetson Linux		✓ Success

✓

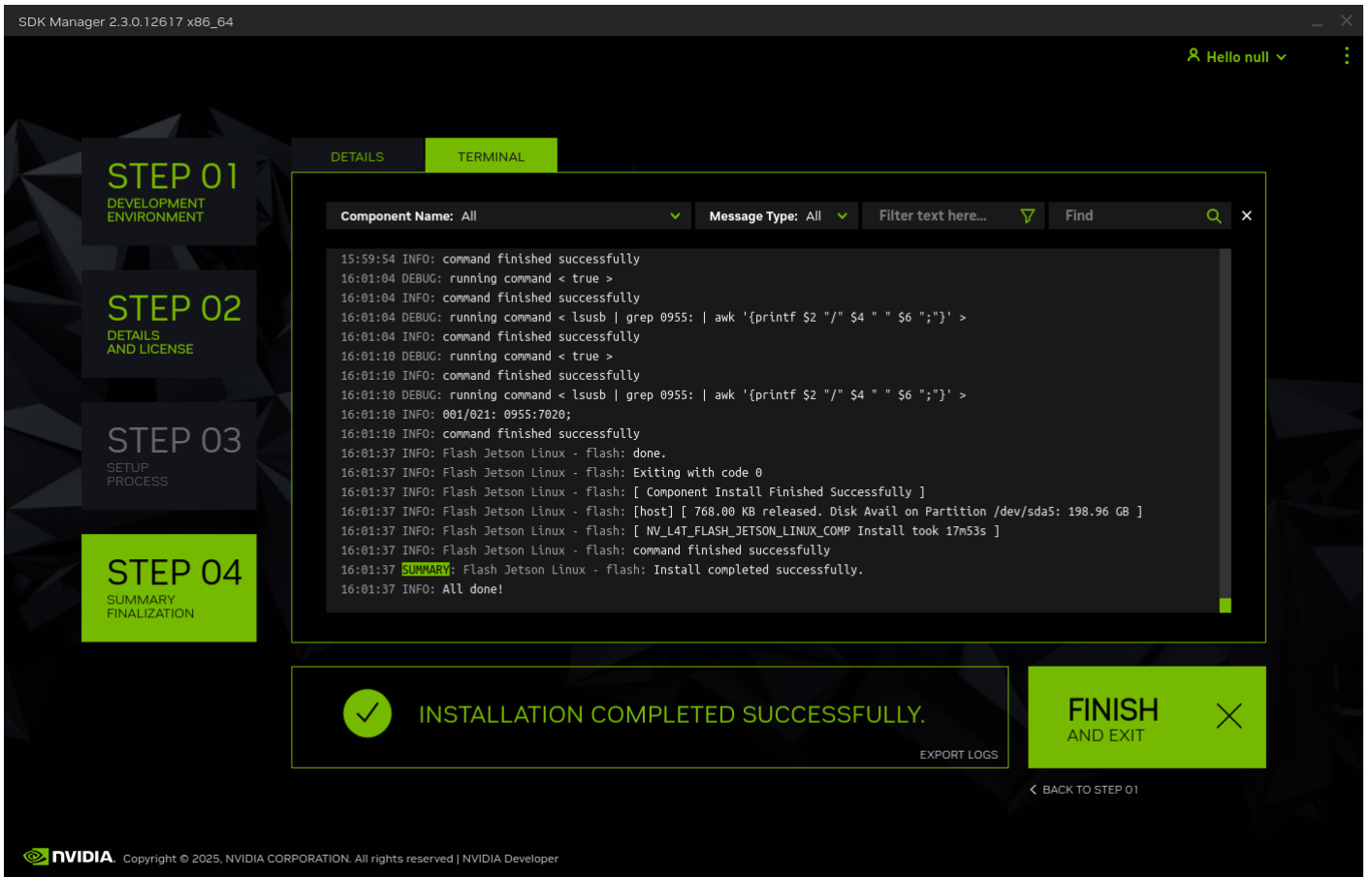
INSTALLATION COMPLETED SUCCESSFULLY.

EXPORT LOGS

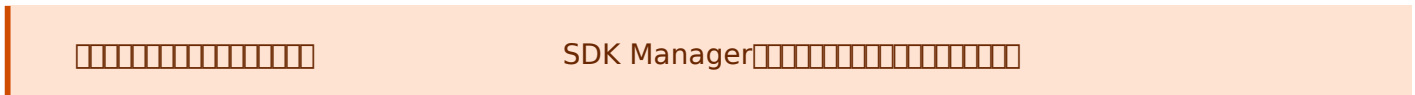
FINISH
AND EXIT

✕

⏪ BACK TO STEP 01



2.???????super??



2.1 ??????

-  **FC REC**  **GND**  **Tpye-C**


2.2 ?????????????



SDK Manager

10~20

- JetPack6.2

```
cd /home/ubuntu/nvidia/nvidia_sdk/JetPack_6.2.1_Linux_JETSON_ORIN_NANO_TARGETS/Linux_for_Tegra
#
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-super internal
```

- JetPack5.1.5

```
cd /home/ubuntu/nvidia/nvidia_sdk/JetPack_5.1.5_Linux_JETSON_ORIN_NANO_TARGETS/Linux_for_Tegra
#
sudo ./tools/kernel_flash/l4t_initrd_flash.sh --external-device nvme0n1p1\
  -c tools/kernel_flash/flash_l4t_external.xml -p "-c
bootloader/t186ref/cfg/flash_t234_qspi.xml"\
  --showlogs --network usb0 jetson-orin-nano-devkit-super internal
```



```
sudo ./tools/backup_restore/l4t_backup_restore.sh -b -e nvme0n1 jetson-orin-nano-devkit-nvme
```

- | | | | |
|--|--|--|--|
| | | | |
|--|--|--|--|

```
sudo ./tools/backup_restore/l4t_backup_restore.sh -r -e nvme0n1 jetson-orin-nano-devkit-nvme
```

`nvrestore_partitions.sh` 292~296

```

278 # This block will check to make sure the model of the target board matches the
279 # model of the board the images came from.
280 for value in $(grep -v -e '(\^ *$|^#)' < "${FILE_NAME}"); do
281     declare -a FIELDS
282     for part in {1..6}; do
283         FIELDS[part]=$(echo "$value" | awk -F, -v part=${part} '{print $part}')
284     done
285     if [ "${FIELDS[1]}" = 'board_spec' ]; then
286         set_board_spec
287         if [[ "${FIELDS[2]}" == "${BOARD_SPEC}" ]]; then
288             BOARD_MATCH=true
289         fi
290     fi
291 done
292 if [ ${BOARD_MATCH} = false ]; then
293 #     echo "${SCRIPT_NAME}: You are trying to flash images from a board model that does not"
294 #     echo "match the current board you're flashing onto."
295 #     exit 1
296 #fi
297
298 declare -A able_to_delete

```

#5

edit 11 2025 06:46:48

edit 18 2025 07:55:53