

WeAct Studio

NANO&XAVIER

TX2 NX CB

Tutorial

Catalog

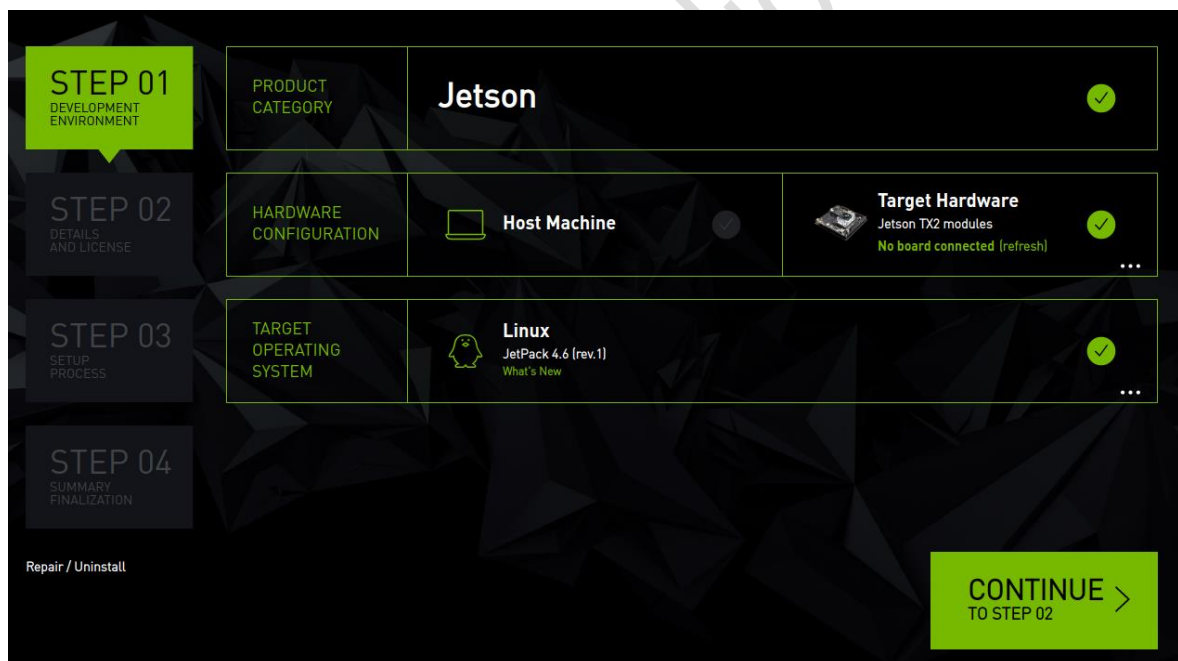
Revision History	3
1. Build a Flash environment	4
2. Update device tree for nano / nx	7
3. Communication using can	11
4. GPIO using on shell	13
Contact with us	14

REVISION HISTORY

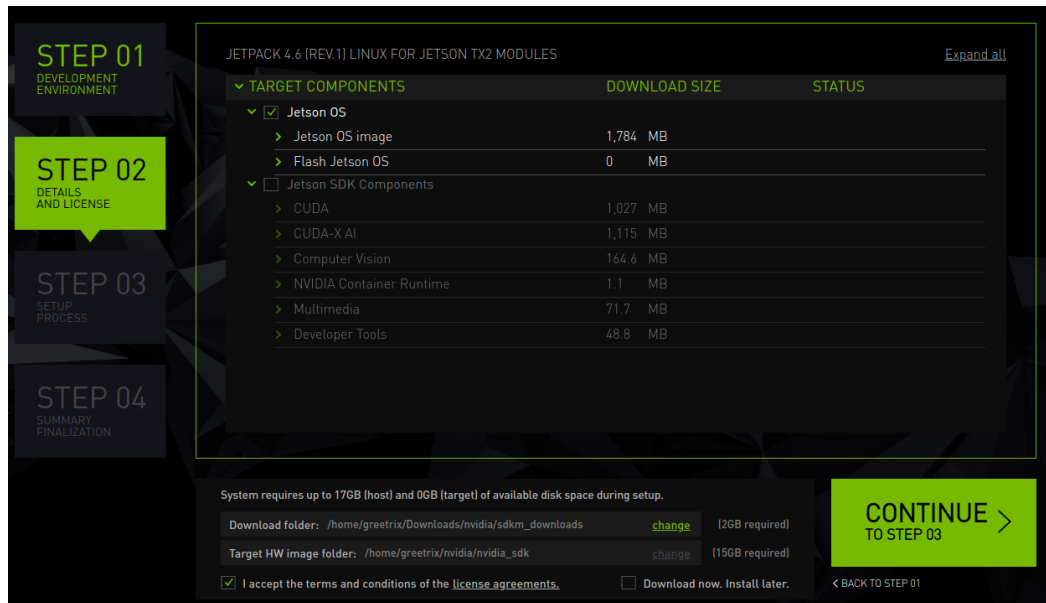
Draft Date	Revision	Description
2021.9.20	V1.0	1. Init Version.

1. BUILD A FLASH ENVIRONMENT

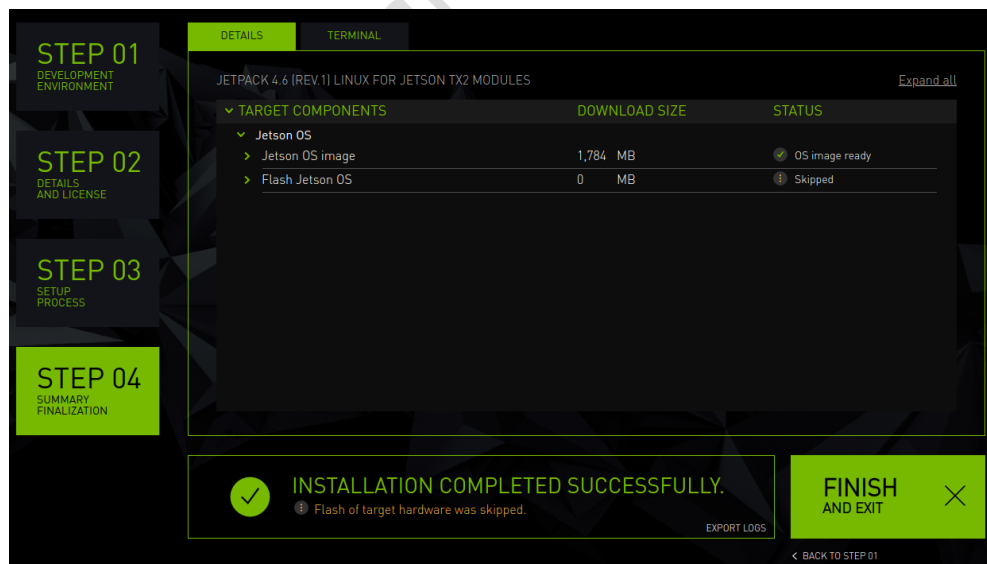
- a) First, you need a computer with Ubuntu 16.04 or above as the host to burn nano / NX, or you can install VMware on windows.
- b) Download the latest SDK manager from NVIDIA and install it in Ubuntu 18.04 (You need to register an NVIDIA account, which will also be used later)
 - SDK-Manager Download: <https://developer.nvidia.com/nvidia-sdk-manager>
- c) Select the target hardware and jetpack version required, uncheck the host machine, take tx2nx as an example, and click continue.



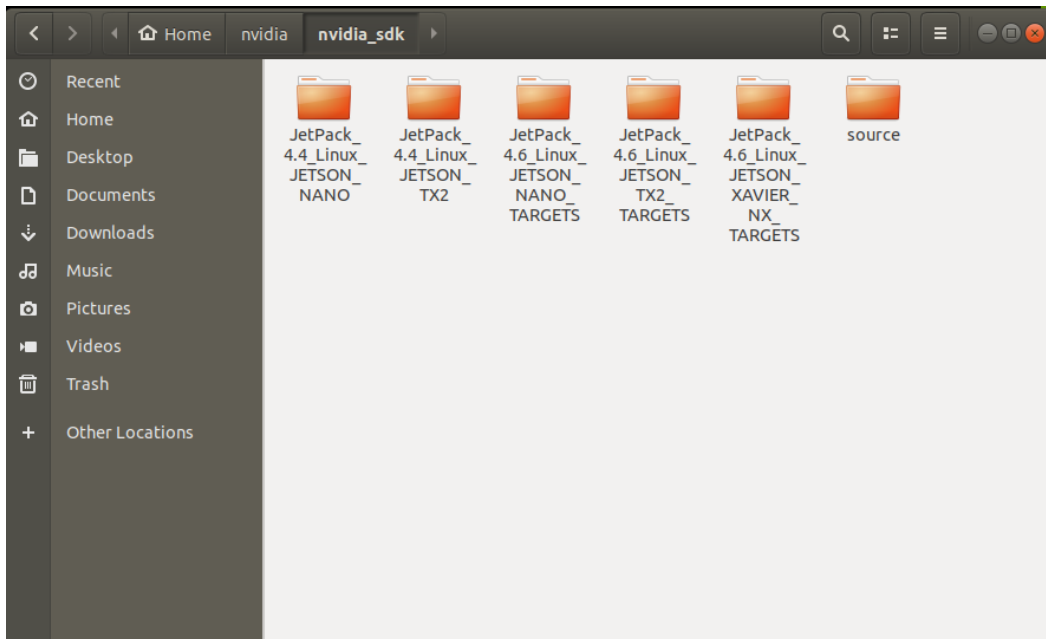
- d) Check I accept the terms and conditions of the license agreements, uncheck the Jetson SDK components, and click continue to proceed to the next step.



P. S: please download and install in a smooth network environment. When the download or installation fails, click Retry to continue until all the status is installed and green is displayed. During the installation process, a network burning message will pop up and select skip.



e) After the installation is successful, the required files will be burned with the corresponding version under `~ / NVIDIA / nvidia_sdk /`



- f) Install Python support through `sudo apt-get install python` on the terminal for subsequent environment burning.

2. UPDATE DEVICE TREE FOR NANO / NX

P.S: the Weact device tree is different from the official device tree (Other functions are the same). If there is no need, the device tree can not be updated.

!!! Note that updating the device tree does not affect any system files. Please be assured to update

NVIDIA and WeAct device tree diff

	NVIDIA	WeAct Studio
Nano-SD	Same	Same
Nano-EMMC	Cant use SD	Can use SD
TX2NX	Cant use SD&UART1	Can use SD &UART1
XavierNX	Cant use SD	Can use SD

a) Here, take tx2nx as an example, download the corresponding device tree file on the GitHub of Weact studio.

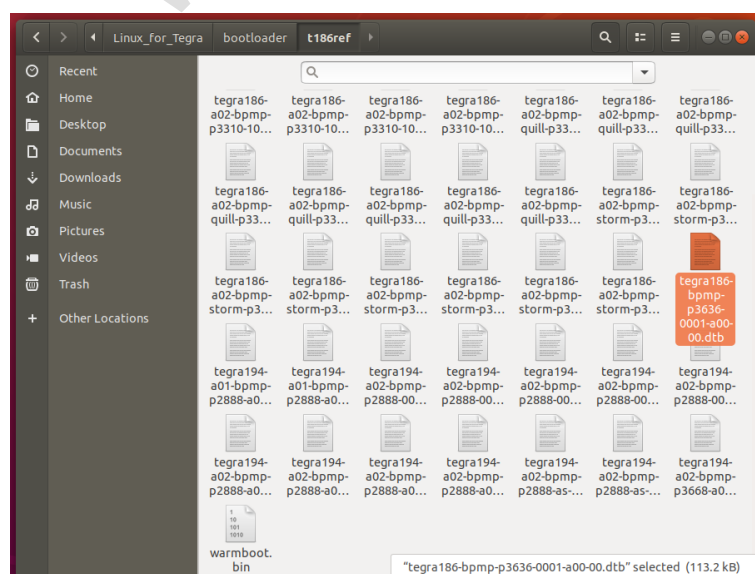
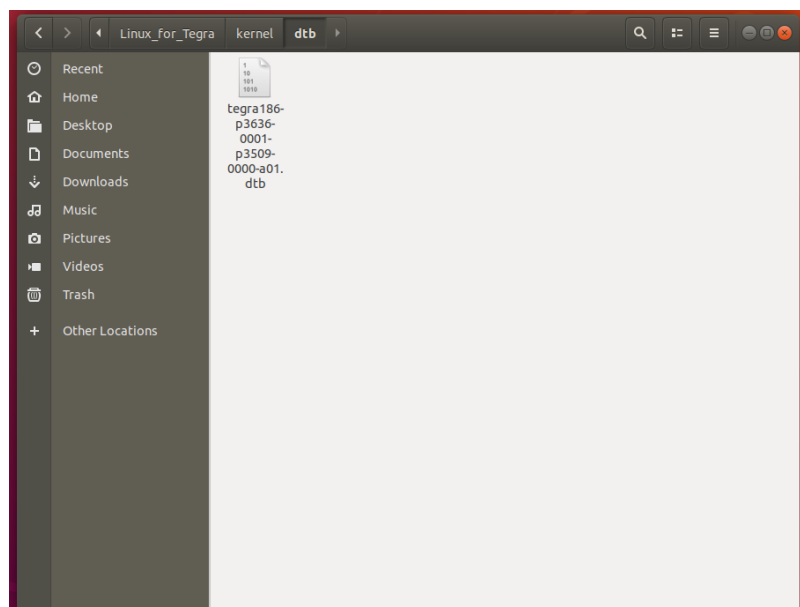
➤ Github: https://github.com/WeActTC/Nano_TX2-Xavier_NX-CB

Update path and equipment tree name of each equipment tree

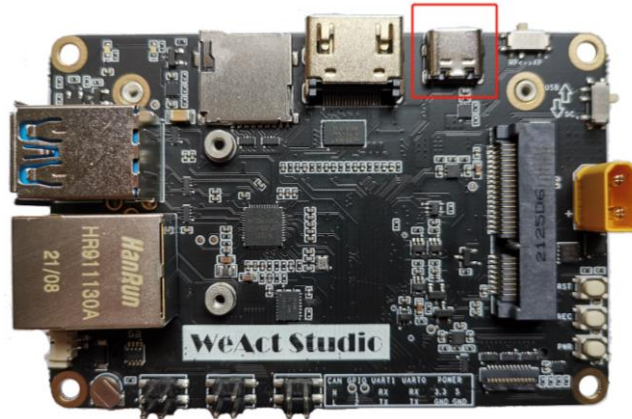
	Linux_for_Tegra/kernel/dtb	Linux_for_Tegra/bootloader/t186ref(t210f)
Nano-EMMC	tegra210-p3448-0002-p3449-0000-b00	None
TX2NX	tegra186-p3636-0001-p3509-0000-a01	tegra186-bpmp-p3636-0001-a00-00
XavierNX	tegra194-p3668-all-p3509-0000	None

b) Find the corresponding version of the device tree

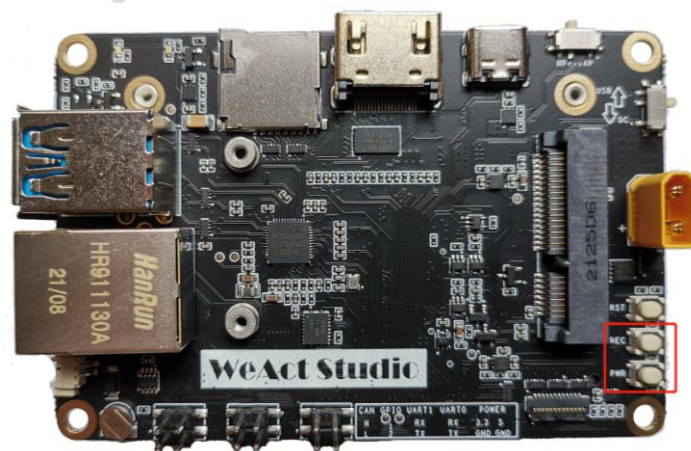
1. Enter `~/nvidia/nvidia_sdk/JetPack_4.6_Linux_JETSON_TX2_TARGETS/Linux_for_Tegra/kernel/dtb`, copy device-tree **tegra186-p3636-0001-p3509-000-a01.dtb** to this dir.
2. Enter `~/nvidia/nvidia_sdk/JetPack_4.6_Linux_JETSON_TX2_TARGETS/Linux_for_Tegra/bootloader/t186ref`, Copy device-tree **tegra186-bmp-p3636-0001-a00-00.dtb** to this dir **【Only TX2NX】**



3. Use the USB type-C cable to connect the USB OTG interface on the carrier board.



4. Turn the power on key to MP (Manual power on), press the rec key, then press the PWR key to power on, release the rec key to enter the recovery mode, at this time, NVIDIA USB drive sign will appear in the lower right corner of VMware, or open the terminal and enter lsusb command, NVIDIA Corp will be found.



5. Enter `~/nvidia/nvidia_sdk/JetPack_4.6_Linux_JETSON_TX2_TARGETS/Linux_for_Tegra`, open command, run `sudo ./flash.sh -r -k kernel-dtb jetson-xavier-nx-devkit-tx2-nx mmcblk0p1`, wait updated successful.

Device tree updated command

Device	Command
Nano-SD	<code>sudo ./flash.sh -r -k DTB jetson-nano-qspi-sd mmcblk0p1</code>
Nano-EMMC	<code>sudo ./flash.sh -r -k DTB jetson-nano-emmc mmcblk0p1</code>
TX2-NX	<code>sudo ./flash.sh -r -k kernel-dtb jetson-xavier-nx-devkit-tx2-nx mmcblk0p1</code>
Xavier-SD	<code>sudo ./flash.sh -r -k kernel-dtb jetson-xavier-nx-devkit-qspi mmcblk0p1</code>
Xavier-EMMC	<code>sudo ./flash.sh -r -k kernel-dtb jetson-xavier-nx-devkit-emmc mmcblk0p1</code>

After updating the device tree, it will be successful! Display, as shown in the following figure.

```
File Edit View Search Terminal Help
[ 11.1401 ] tegradevflash_v2 --iscpubl
[ 11.1423 ] Cannot Open USB
[ 11.9533 ]
[ 12.9584 ] tegrarcm_v2 --isapplet
[ 13.2306 ]
[ 13.2341 ] tegradevflash_v2 --iscpubl
[ 13.2354 ] Bootloader version 01.00.0000
[ 13.3996 ] Bootloader version 01.00.0000
[ 13.4611 ]
[ 13.4611 ] Writing partition
[ 13.4647 ] tegradevflash_v2 --write kernel-dtb 1_kernel_tegra186-p3636-0001-p3
509-0000-a01_sigheader.dtb.encrypt
[ 13.4676 ] Bootloader version 01.00.0000
[ 13.6334 ] Writing partition kernel-dtb with 1_kernel_tegra186-p3636-0001-p350
9-0000-a01_sigheader.dtb.encrypt
[ 13.6352 ] [.....] 100%
[ 13.7256 ]
[ 13.7259 ] Coldbooting the device
[ 13.7283 ] tegradevflash_v2 --reboot coldboot
[ 13.7306 ] Bootloader version 01.00.0000
[ 13.9214 ]
*** The [kernel-dtb] has been updated successfully. ***
```

3. COMMUNICATION USING CAN

- a) Two CAN controllers (CAN 0 / CAN 1) are integrated on TX2 NX / xaviernx. In addition, a CAN transceiver (CAN 0) is designed on the carrier board of Weact studio, which can be directly attached to the CAN physical bus.
- b) TX2 NX / xaviernx has its own CANbus driver and is integrated into the image. It already supports CANbus without further processing. We need to install CANbus module. (enter the following command in the terminal or put it into rc.local to start the self startup)

```
modprobe can
modprobe can-raw
modprobe can-bcm
modprobe can-gw
modprobe can_dev
modprobe mttcan
```

- c) Check whether the installation is successful through lsmod.

```
nvidia@localhost:~$ lsmod
Module                  Size  Used by
fuse                    103841  2
mttcan                   66251  0
can_dev                 13306  1 mttcan
can_gw                   10919  0
can_bcm                  16471  0
can_raw                 10388  0
can                      46600  3 can_raw,can_bcm,can_gw
zram                     26166  6
overlay                 48691  0
bcmhdhd                  934274  0
cfg80211                 589351  1 bcmhdhd
spidev                   13282  0
nvgpu                   1575721  20
bluedroid_pm             13912  0
ip_tables                19441  0
x_tables                 28951  1 ip_tables
```

- ```
sudo ip link set can0 type can bitrate 500000
sudo ip link set up can0
```

- ```
nvidia@localhost:~$ ifconfig
can0: flags=193<UP,RUNNING,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 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 131
```

- [illegible]

4. GPIO USING ON SHELL

- a) Nano / TX2 NX / Xavier NX can directly control GPIO input and output through shell commands.

	GPIO1	GPIO2
Nano	194	38
TX2-NX	338	269
Xavier-NX	196	105

- b) Take tx2-nx gpio1 as an example
- Activate IO first : `sudo echo 338 > /sys/class/gpio/export`
 - Set IO direction: `echo out > /sys/class/gpio/gpio338/direction`
 - Set output level : `echo 1 > /sys/class/gpio/gpio338/value`

CONTACT WITH US

- Github: <https://github.com/WeActTC>
- Gitee: <https://gitee.com/WeAct-TC>
- Site: <https://www.weact-tc.cn/>
- Aliexpress:
<https://www.aliexpress.com/item/1005003334440054.html?spm=5261.ProductManageOnline.0.0.48104edfJwGktm>

WeAct Studio