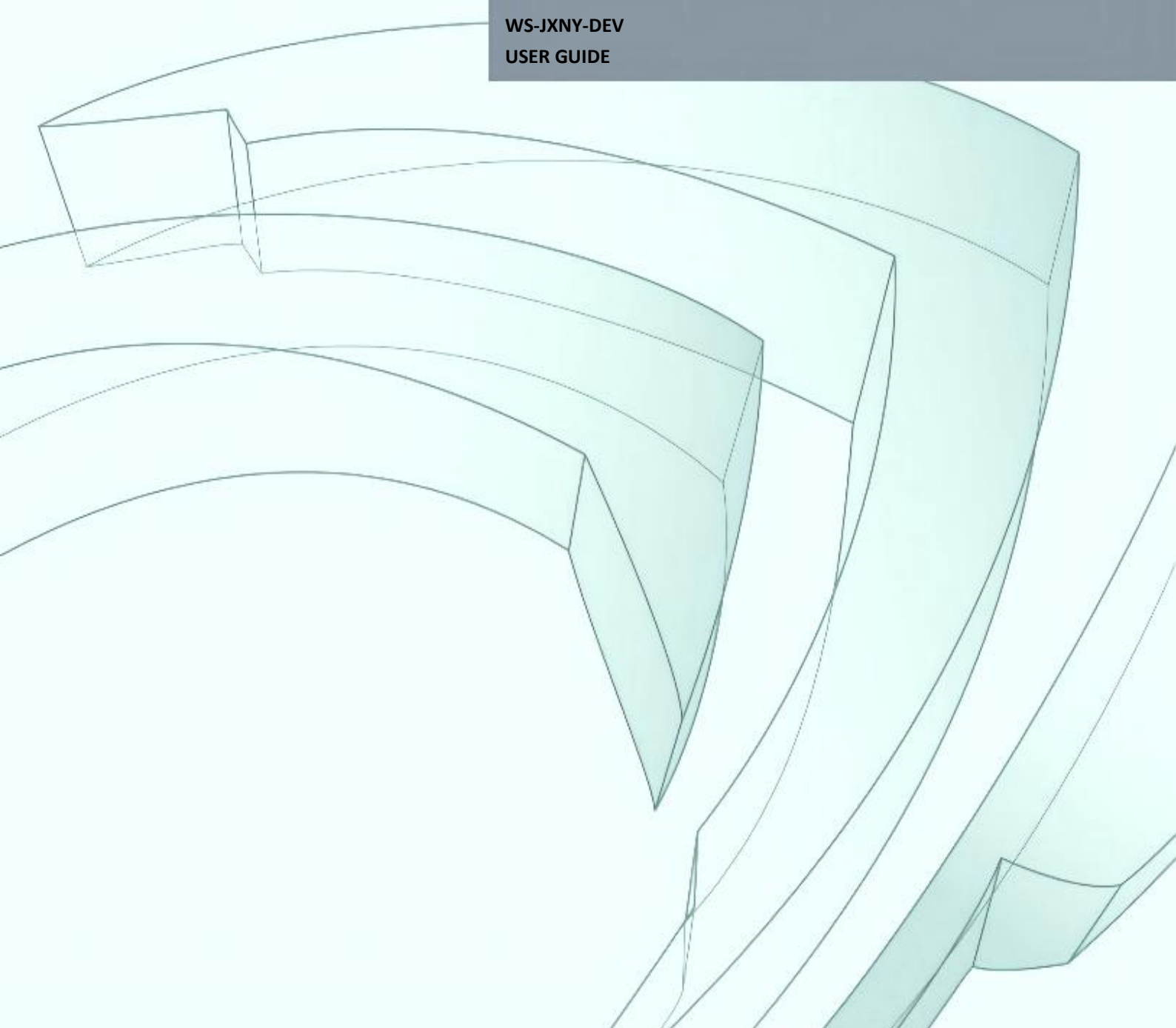
智用星空
WISDOM STARRY

NVIDIA ELITE PARTNER

WS-JXNY-DEV

Development System

WS-JXNY-DEV
USER GUIDE



DOCUMENT CHANGE HISTORY

Version	Date	Authors	Description of Change
1.0	DEC 2, 2021		Initial release

Welcome! There are two key things you should do right away:

0. Sign up for the NVIDIA Developer Program – this enables you to ask questions and contribute on the NVIDIA Jetson Forums, gives access to all documentation and collateral on the Jetson Download Center, and more.
1. Read this User Guide! After that, check out these important links:
    - Jetson FAQ – Please read the FAQ.
    - Support Resources – This web page links to important resources, including the Jetson Forum and the Jetson Ecosystem page.
    - NVIDIA Jetson Linux Driver Package Release Notes –Jetson Linux Driver Package is a key component of the Jetson platform, and provides the sample filesystem for your developer kit. Please read the latest release notes.

Thanks,
Wisdom Starry -- the NVIDIA Jetson Partner's team

目录

GETTING STARTED

The Development System provides a full-featured development platform designed to get you up and running NVIDIA JETSON quickly. The included carrier board exposes many standard hardware interfaces, enabling a highly flexible and extensible platform for rapid prototyping.

NVIDIA JetPack™SDK supports both your Development System and host development platform. It includes:

- Sample Linux filesystem with NVIDIA drivers
- AI and Computer Vision libraries and APIs
- Developer tools
- Documentation and sample code

Before using your Development System, you must install JetPack. A Linux host computer is required; for details, see How to Install JetPack, below.

Minimum system requirements for the host computer are:

- Ubuntu Linux x64 v18.04
- A valid Internet connection
- At least 23GB of disk space

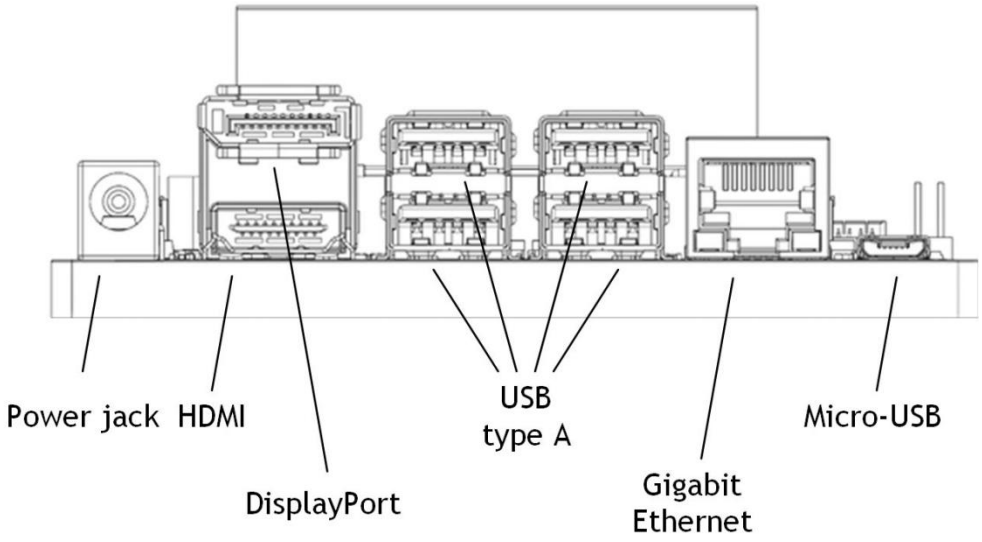
INCLUDED IN THE BOX

The Development System includes:

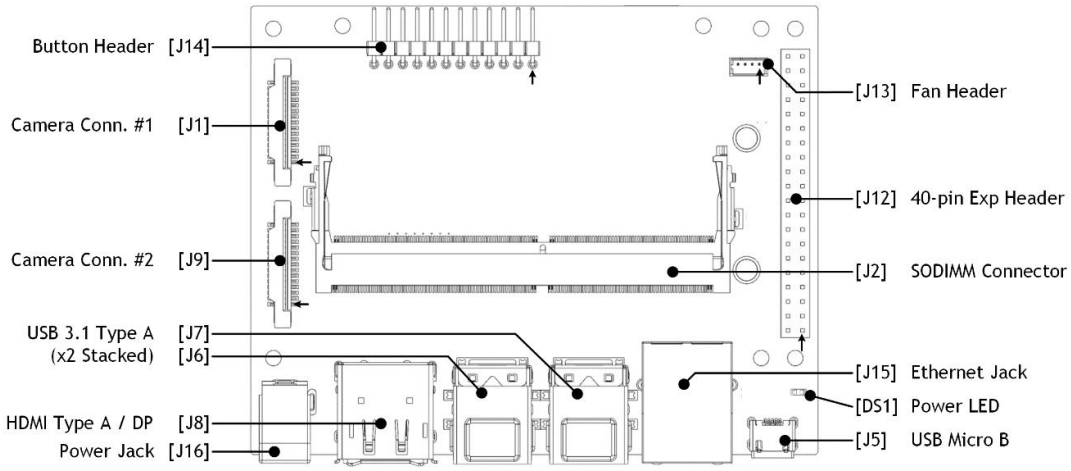
- NVIDIA Jetson Xavier™ NX module (P3668) with thermal solution
  (Or NVIDIA Jetson™ TX2-NX module (P3636) with thermal solution)
- Reference carrier board (WS-MDN-501)
- Power supply with 12V5A AC cord

1.1 Development System INTERFACES

Top view (left)

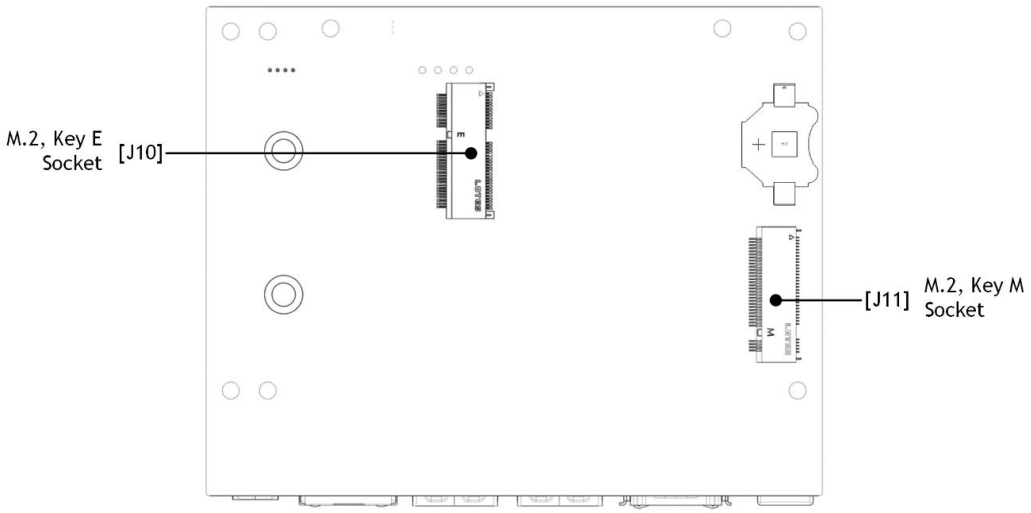


Top view of Development System carrier board



bottom view of Development System carrier

board



1.1.1 Parameter table

JETSON moudle	NVIDIA JETSON NANO	NVIDIA JETSON TX2-NX	NVIDIA JETSON XAVIER NX
AI Performance	472 GFLOPs	1.3 TFLOPs	21 TOPs
OS	UBUNTU 18.04	UBUNTU 18.04	UBUNTU 18.04
CPU	Dual-Core Denver 1.5 64-Bit CPU and ARM-A57	Dual-Core NVIDIA Denver 1.5 64-Bit CPU and Quad-Core ARM® Cortex-A57	6-core 64-bit CPU NVIDIA Carmel ARMv8.2
GPU	128-core NVIDIA Maxwell™ GPU	256-core NVIDIA Pascal™ GPU	384-core NVIDIA Volta™ GPU with 48 Tensor Cores
Memory	4 GB 64-bit LPDDR4	4 GB 128-bit LPDDR4x	8 GB 128-bit LPDDR4x
Storage	16GB eMMC 5.1 M.2 M key NVME 2242 SSD	16GB eMMC 5.1 M.2 M key NVME 2242 SSD	16GB eMMC 5.1 M.2 M key NVME 2242 SSD
Power	5V/12V DC 5W/10W	5V/12V DC 7.5W/15W	5V/12V DC 10W/15W
Display	MICRO HDMI	MICRO HDMI	MICRO HDMI
USB	USB3.0*4,MICRO USB2.0*1, USB2.0*2	USB3.0*4,MICRO USB2.0*1, USB2.0*2	USB3.0*4,MICRO USB2.0*1, USB2.0*2
NETWORK	Gigabit Lan ,4G-LTE，WIFI	Gigabit Lan ,4G-LTE，WIFI	Gigabit Lan ,4G-LTE，WIFI
Interfaces	mPCIE,UART,GPIO,I²C,CAN 5V-PWM-FAN	mPCIE,UART,GPIO,I²C,CAN 5V-PWM-FAN	mPCIE,UART,GPIO,I²C,CAN 5V-PWM-FAN
Mechanical	100*80*43mm, 150g	100*80*43mm, 150g	100*80*43mm, 150g
Temperature	0℃-50℃	0℃-50℃	0℃-50℃

1.2 Description of interface

1.2.1 MICRO HDMI

pin	Signal	pin	Signal
1	Hot Plug Detect	2	Utility
3	TMDS Data2+	4	TMDS Data2 Shield
5	TMDS Data2-	6	TMDS Data1+
7	TMDS Data1 Shield	8	TMDS Data1-
9	TMDS Data0+	10	TMDS Data0 Shield
11	TMDS Data0-	12	TMDS Clock+
13	TMDS Clock Shield	14	TMDS Clock-
15	CEC	16	DDC /CEC Ground
17	SCL	18	SDA
19	+5V Power		

1.2.2 USB 3.0

pin	Signal	pin	Signal
1	VBUS	2	USB 2.0 D-
3	USB 2.0 D+	4	GND
5	SSRX-	6	SSRX+
7	GND	8	SSTX-
9	SSTX+		

1.2.3 USB-OTG

pin	Signal	pin	Signal
1	VBUS	2	USB 2.0 D-
3	USB 2.0 D+	4	USB ID
5	GND		

1.2.4 USB 2.0 Socket

pin	Signal	pin	Signal
1	5V	2	USB 22_ D_N
3	USB 22_ D_P	4	GND

1.2.5 I²S Sound

pin	Signal	pin	Signal
1	VDD _5V_IN	2	GND
3	GPIO12_LS	4	GPIO09_LS
5	I²C2_SCL_LS	6	I²S0_SCLK_LS
7	I²C2_SDA_LS	8	I²S0_SDOUT_LS
9	I²S0_LRCK_LS	10	I²S0_SDIN_LS

1.2.6 FAN_PWM

pin	Signal	pin	Signal
1	GND	2	+5V
3	FAN_TACH_CON	4	FAN_PWM

1.2.7 Gigabyte LAN

pin	Signal	pin	Signal
1	TP0+	2	TP0-
3	TP1+	4	TP2+
5	TP2-	6	TP1-
7	TP3+	8	TP3-

1.2.8 12V POWER

pin	Signal	pin	Signal
1	12V	2	GND
Voltage range +12V			

1.2.9 5V POWER

pin	Signal	pin	Signal
1	5V	2	GND
Voltage range +5V			

1.3.1 Extension 40pin

pin	Signal	pin	Signal
1	3.3V	2	3.3V
3	UART0_TX	4	UART0_RX
5	UART1_RX	6	UART1_TX
7	GPIO_0	8	GPIO_1
9	GPIO_2	10	GPIO_3
11	I²C_GP1_CLK	12	I²C_GP1_DAT
13	RECOVERY	14	RTC_BAT_INPUT
15	RESET	16	PC_LED+
17	POWER_BUTTON	18	PC_LED-
19	GND	20	GND
21	-	22	-
23	CAN0H*	24	CAN0L*
*Pin 23 24 only for NX moudle use			

1.3.1.1 URAT0

pin	Signal	pin	Signal
3	UART0_TX	4	UART0_RX

*:URAT0 for UART console Print debug information，device name /dev/ttyTHS0

Tools can be used for referencecutecom：sudo apt-get install cutecom

Connect NX/NANO with HOST program cutecom，Baud rate 115200/8N1。

1.3.1.2 URAT1

pin	Signal	pin	Signal
5	UART1_RX	6	UART1_TX

*:device name /dev/ttyTHS0

1.3.1.3 CAN0

pin	Signal	pin	Signal
23	CAN0H	24	CAN0L

1.3.1.4 GPIO

pin	Signal	pin	Signal
7	GPIO_0	8	GPIO_1
9	GPIO_2	10	GPIO_3

*:GPIO TEST:

```
#check gpio
cd /sys/class/gpio
#load gpio
echo '388' | sudo tee /sys/class/gpio/export
echo '298' | sudo tee /sys/class/gpio/export
echo '480' | sudo tee /sys/class/gpio/export
echo '486' | sudo tee /sys/class/gpio/export
 #set gpioOutput direction
cd gpio388
echo 'out' | sudo tee /sys/class/gpio/gpio388/direction
#gpio 3.3v
echo '1' | sudo tee /sys/class/gpio/gpio388/value
#gpio 0v
echo '0' | sudo tee /sys/class/gpio/gpio388/value
#set gpio lutput direction, File directory /sys/class/gpio
#check gpio value，set input value 0
cat ./gpio480/value (print 0，)
echo 'in' | sudo tee /sys/class/gpio/gpio480/direction
#High level，3.3v,retune 1
cat /sys/class/gpio/gpio480/value
```

1.3.1.5 I²C

pin	Signal	pin	Signal
11	I²C_GP1_CLK	12	I²C_GP1_DAT

*:I²C TEST:

```
#check I2C BUS
sudo i2cdetect -l
# Check whether the slave device is identified on the bus (the bus ID is the bus number), and the result shows that there is a number and UU on behalf of the device
sudo i2cdetect -y BUSID
#Read 16 bit data (busid is bus number) (W writes 2 bits) ((0x50 is the address corresponding to the number and Uu), (0x00; 0x20 is the register address) (R is to read 16 bits)sudo i2ctransfer -f -y BUSID w2@0x50 0x00 0x20 r16
#Write 4-bit data (busid is the bus number) (W writes 4 bits) ((0x50 is the address corresponding to the number and Uu), (0x00; 0x20 is the deposit address) (0x77-0x77 is the new content to be modified))
sudo i2ctransfer -f -y BUSID w4@0x50 0x00 0x20 0x77 0x77
```

2、System flashing

2.1 Prepare

Download Image compression package

Jetson Download Center.

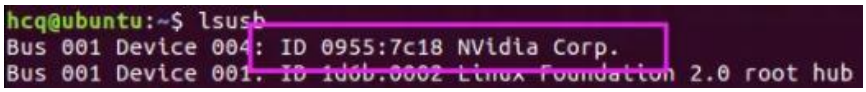
2.2 Flashing

（1）Assemble carrier board+NANO/TX2-NX/Nx core moudle+ Radiator，connect 12V2pin green terminal power supply；

（2）Used to enter Force Recovery Mode. Button is held down while either system is first powered on, or by

pressing & releasing reset button while Recovery button is pressed.

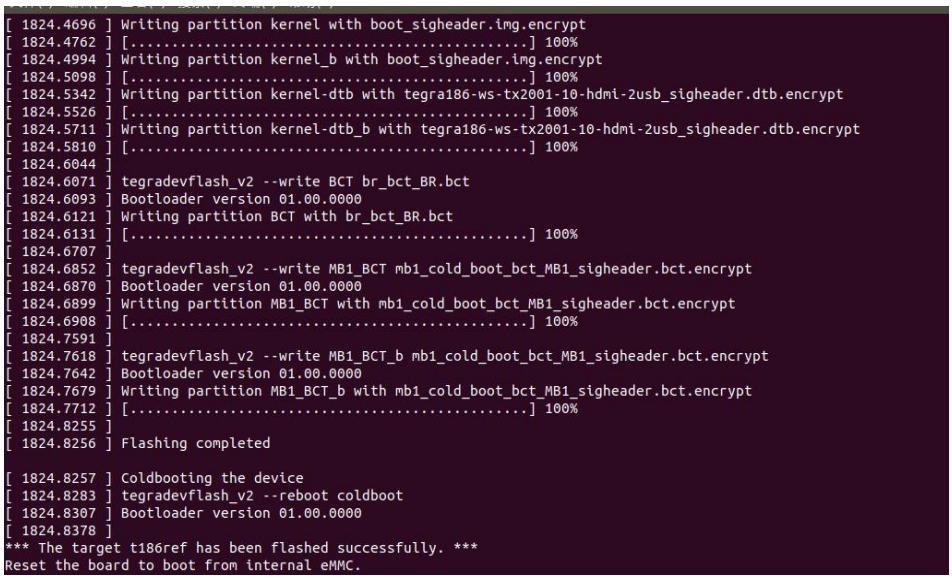
（3）To determine whether or not to enter the recovery mode successfully, you can use the lsusb command to check whether there are "NVIDIA Corp" devices.



如图所示即表示已进入 recovery 模式

（4）following to install.sh The text description when the run is completed, or the readme.txt The introduction of the document flash.sh Burning operation. For example: burning records

sudo ./flash.sh jetson-xavier-nx-devkit-emmc mmcblk0p1



2.3 flashing custom image

2.3.1 Backup custom image:

```
sudo ./flash.sh -r -k APP -G backup.img jetson-xavier-nx-devkit-emmc mmcblk0p1
```

Generated backup.img.raw Image file of (recommended to be compressed to zip file storage)

2.3.2 Restore custom image:

Copy the image of to Linux_ for_ Tegra / bootloader / directory and rename to system.img

```
sudo mv backup.img.raw bootloader/system.img
```

```
cd Linux_for_Tegra/bootloader/
sudo ./flash.sh -r jetson-xavier-nx-devkit-emmc mmcblk0p1
```

Note: - r parameter refers to using the system.img to flash.

Note: both backup and recovery must be in the mode of receover.

2.4 verification

Connect the MICRO HDMI interface display to power on verification, and the image burning is successful when the system is star.