

6. Bind the device ID

6. Bind the device ID

- 6.1. Device view command
- 6.2. Establish port mapping relationship
 - 6.2.1. Device binding
 - 6.2.2. Introduction to rule file syntax
- 6.3. Verify View
- 6.4. Binding the USB port

When the robot uses two or more USB serial devices, the corresponding relationship between the device name and the device is not fixed, but is allocated according to the order in which the devices are connected to the system. Inserting one device first and then another device can determine the relationship between the device and the device name, but it is very troublesome to plug and unplug the device every time the system starts. The serial port can be mapped to a fixed device name. No matter what the insertion order is, the device will be mapped to the new device name. We only need to use the new device name to read and write the device.

6.1. Device view command

Device ID view

```
lsusb
```

As can be seen from the figure below, the ID number of each device, Astra has the official file for binding the device, the handle generally does not need to be bound, and the main binding is PCB and radar.

```
jetson@yahboom: ~  
jetson@yahboom: ~ 83x41  
jetson@yahboom:~$ lsusb  
Bus 002 Device 002: ID 0bda:0411 Realtek Semiconductor Corp.  
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub  
Bus 001 Device 003: ID 8087:0a2b Intel Corp.  
Bus 001 Device 009: ID c0f4:04e0  
Bus 001 Device 007: ID 413c:301a Dell Computer Corp.  
Bus 001 Device 005: ID 214b:7250  
Bus 001 Device 008: ID 2bc5:0403  
Bus 001 Device 006: ID 2bc5:0501  
Bus 001 Device 004: ID 05e3:0608 Genesys Logic, Inc. Hub  
Bus 001 Device 012: ID 1a86:7523 QinHeng Electronics HL-340 USB-Serial adapter  
Bus 001 Device 018: ID 0079:181c DragonRise Inc.  
Bus 001 Device 013: ID 10c4:ea60 Cygnal Integrated Products, Inc. CP210x UART Bridge / myAVR mySmartUSB light  
Bus 001 Device 010: ID 2109:2813 VIA Labs, Inc.  
Bus 001 Device 002: ID 0bda:5411 Realtek Semiconductor Corp.  
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
jetson@yahboom:~$
```

Device number view

```
11 /dev/
```

```

jetson@yahboom: ~ 117x43
crw----- 1 root root 3, 10 12月 10 17:15 ttya
crw----- 1 root root 3, 11 12月 10 17:15 ttyb
crw----- 1 root root 3, 12 12月 10 17:15 ttyc
crw----- 1 root root 3, 13 12月 10 17:15 ttyd
crw----- 1 root root 3, 14 12月 10 17:15 ttye
crw----- 1 root root 3, 15 12月 10 17:15 ttyf
crw--w---- 1 root tty 4, 64 2月 14 18:01 ttyS0
crw-rw---- 1 root dialout 4, 65 12月 10 17:15 ttyS1
crw-rw---- 1 root dialout 4, 66 12月 10 17:15 ttyS2
crw-rw---- 1 root dialout 4, 67 12月 10 17:15 ttyS3
crw--w---- 1 root tty 238, 1 2月 14 18:01 ttyTHS1
crw-rw---- 1 root dialout 238, 2 12月 10 17:15 ttyTHS2
crwxrwxrwx 1 root dialout 188, 0 12月 10 17:15 ttyUSB0
crwxrwxrwx 1 root dialout 188, 1 2月 14 18:01 ttyUSB1
crw----- 1 root root 10, 239 12月 10 17:15 uhid
crw----- 1 root root 10, 223 12月 10 17:15 uinput
crw-rw-rw- 1 root root 1, 9 12月 10 17:15 urandom
drwxr-xr-x 4 root root 80 12月 10 17:15 v4l/
crw-rw---- 1 root tty 7, 0 12月 10 17:15 vcs
crw-rw---- 1 root tty 7, 1 12月 10 17:15 vcs1
crw-rw---- 1 root tty 7, 2 12月 10 17:15 vcs2
crw-rw---- 1 root tty 7, 3 12月 10 17:15 vcs3
crw-rw---- 1 root tty 7, 4 12月 10 17:15 vcs4
crw-rw---- 1 root tty 7, 5 12月 10 17:15 vcs5
crw-rw---- 1 root tty 7, 6 12月 10 17:15 vcs6
crw-rw---- 1 root tty 7, 128 12月 10 17:15 vcsa
crw-rw---- 1 root tty 7, 129 12月 10 17:15 vcsa1
crw-rw---- 1 root tty 7, 130 12月 10 17:15 vcsa2
crw-rw---- 1 root tty 7, 131 12月 10 17:15 vcsa3
crw-rw---- 1 root tty 7, 132 12月 10 17:15 vcsa4
crw-rw---- 1 root tty 7, 133 12月 10 17:15 vcsa5
crw-rw---- 1 root tty 7, 134 12月 10 17:15 vcsa6
drwxr-xr-x 2 root root 60 1月 1 1970 vfio/
crw----- 1 root root 10, 137 12月 10 17:15 vhci
crw-rw----+ 1 root video 81, 0 12月 10 17:15 video0
crw----- 1 root root 10, 130 12月 10 17:15 watchdog
crw----- 1 root root 244, 0 12月 10 17:15 watchdog0
crw-rw-rw- 1 root root 1, 5 12月 10 17:15 zero
brw-rw---- 1 root disk 252, 0 2月 14 18:01 zram0
brw-rw---- 1 root disk 252, 1 2月 14 18:01 zram1
brw-rw---- 1 root disk 252, 2 2月 14 18:01 zram2
brw-rw---- 1 root disk 252, 3 2月 14 18:01 zram3
jetson@yahboom:~$

```

PCB

laser

Astra

6.2. Establish port mapping relationship

6.2.1. Device binding

- Astra binding

The binding rules file for Astra camera is [56-orbbec-usb.rules], It is provided by Astra manufacturers and demonstrated here with AstraPro Plus.

Place the [56-orbbec-usb.rules] file in the /etc/udev/rules.d directory of the main control

```
/etc/udev/rules.d/56-orbbec-usb.rules
```

You can find the [56-orbbec-usb.rules] file, which is the Astra camera device binding file.

Run the following command to refresh the USB rules to bind the Astra camera

```
sudo udevadm control --reload-rules && sudo udevadm trigger
```

Check whether the binding is successful:

```
jetson@ubuntu:~$ ll /dev/astra*
lrwxrwxrwx 1 root root 15 May  5 17:42 /dev/astradepth -> bus/usb/001/007
#astradepth port
lrwxrwxrwx 1 root root 15 May  5 17:42 /dev/astrauvc -> bus/usb/001/009    #RGB
port
```

If the preceding information is displayed, the binding is successful.

- **PCB and Radar Bonding**

Go to the rules.d directory

```
cd /etc/udev/rules.d/
```

Create a new [usb.rules] file and edit it

```
sudo vim usb.rules
```

write the following

```
KERNEL=="ttyUSB*", ATTRS{idVendor}=="1a86", ATTRS{idProduct}=="7523",
MODE:="0777", SYMLINK+="myserial"
KERNEL=="ttyUSB*", ATTRS{idVendor}=="10c4", ATTRS{idProduct}=="ea60",
MODE:="0777", SYMLINK+="rplidar"
```

Exit to make the rules take effect

```
sudo chmod 777 usb.rules
sudo udevadm control --reload-rules && sudo udevadm trigger
```

Check whether the binding is successful:

```
jetson@ubuntu:~$ ll /dev | grep ttyUSB*
lrwxrwxrwx 1 root root 7 5月 18 20:13 gps1 -> ttyUSB1
lrwxrwxrwx 1 root root 7 5月 18 20:13 myserial -> ttyUSB0    #pcb binding
ttyUSB0 port
lrwxrwxrwx 1 root root 7 5月 18 20:13 rplidar -> ttyUSB1    #radar
binding ttyUSB1 port
crwxrwxrwx 1 root dialout 188, 0 5月 18 20:13 ttyUSB0
crwxrwxrwx 1 root dialout 188, 1 5月 18 20:13 ttyUSB1
```

If the preceding information is displayed, the binding is successful.

6.2.2. Introduction to rule file syntax

```
KERNEL=="ttyUSB*", ATTRS{idVendor}=="1a86", ATTRS{idProduct}=="7523",
MODE:="0777", SYMLINK+="myserial"
KERNEL=="ttyUSB*", ATTRS{idVendor}=="10c4", ATTRS{idProduct}=="ea60",
MODE:="0777", SYMLINK+="rplidar"
```

Parse

```
KERNEL          # The device name that matches the event
ATTR{filename}  # Match the sysfs attribute of the event device.
idVendor        # Vendor ID
idProduct       # product number
SYMLINK          # Generate symbolic links for device files under /dev/. Just
                 # give this device an alias.
MODE            # Set permissions for the device.
```

From [6.1], it is easy to see that the device number of the PCB is [ttyUSB0], and the ID number is [1a86, 7523], which is fixed. 0, 1, 2, 3, 4, ...] are all bound to [myserial]; the same is true for radar device [ttyUSB1]; the same is true for other devices that need to be bound.

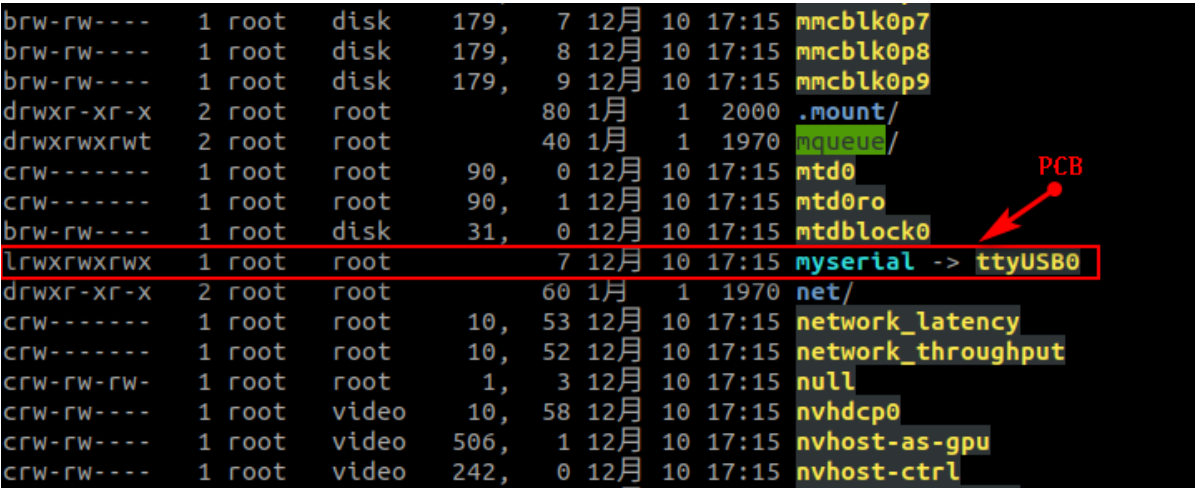
Note: When taking an alias, do not take some device names that already exist in the system, otherwise it will fail.

6.3. Verify View

Device number view

```
11 /dev/
```

PCB




laser

```

brw-rw---- 1 root disk 1, 2 12月 10 17:15 ram2
brw-rw---- 1 root disk 1, 3 12月 10 17:15 ram3
brw-rw---- 1 root disk 1, 4 12月 10 17:15 ram4
brw-rw---- 1 root disk 1, 5 12月 10 17:15 ram5
brw-rw---- 1 root disk 1, 6 12月 10 17:15 ram6
brw-rw---- 1 root disk 1, 7 12月 10 17:15 ram7
brw-rw---- 1 root disk 1, 8 12月 10 17:15 ram8
brw-rw---- 1 root disk 1, 9 12月 10 17:15 ram9
crw-rw-rw- 1 root root 1, 8 12月 10 17:15 random
crw-rw-rw- 1 root netdev 10, 62 12月 10 17:15 rfkill
lrwxrwxrwx 1 root root 7 2月 14 18:01 rplidar -> ttyUSB1
lrwxrwxrwx 1 root root 4 12月 10 17:15 rtc -> rtc1
crw----- 1 root root 252, 0 12月 10 17:15 rtc0
crw----- 1 root root 252, 1 12月 10 17:15 rtc1
drwxr-xr-x 4 root root 80 12月 10 17:15 serial/
drwxrwxrwt 2 root root 60 2月 14 18:01 shm/
drwxr-xr-x 4 root root 600 12月 10 17:15 snd/
lrwxrwxrwx 1 root root 15 12月 10 17:15 stderr -> /proc/self/fd/2
lrwxrwxrwx 1 root root 15 12月 10 17:15 stdin -> /proc/self/fd/0
lrwxrwxrwx 1 root root 15 12月 10 17:15 stdout -> /proc/self/fd/1
crw-rw---- 1 root video 10, 60 12月 10 17:15 tegra_camera_ctrl
crw-rw---- 1 root video 10, 59 12月 10 17:15 tegra_cec
crw-rw---- 1 root crypto 10, 38 12月 10 17:15 tegra-crypto
crw-rw---- 1 root video 239, 1 12月 10 17:15 tegra_dc_0
crw-rw---- 1 root video 239, 2 12月 10 17:15 tegra_dc_1
crw-rw---- 1 root video 239, 0 12月 10 17:15 tegra_dc_ctrl
crw-rw---- 1 root video 10, 63 12月 10 17:15 tegra_mipi_cal

```



6.4. Binding the USB port

The above situations are all different ID numbers. If the ID numbers of the radar and the PCB are the same, or there are two or more PCBs (radars) with the same ID, the above binding will be confused.

Then, we need to bind the USB port. After binding, the **cannot be changed at will**, and each device **can only be linked to a fixed** USB port.


Binding method, take [ttyUSB0] as an example, check the port of the device at this time

```
ll /dev | grep ttyUSB*
```

```

jetson@jetson-desktop:~$ ll /dev | grep ttyUSB*
lrwxrwxrwx 1 root root 7 5月 18 20:13 gps1 -> ttyUSB1
lrwxrwxrwx 1 root root 7 5月 18 20:13 myserial -> ttyUSB0
lrwxrwxrwx 1 root root 7 5月 18 20:13 rplidar -> ttyUSB1
crwxrwxrwx 1 root dialout 188, 0 5月 18 20:13 ttyUSB0
crwxrwxrwx 1 root dialout 188, 1 5月 18 20:13 ttyUSB1

```



The ttyUSB0 corresponding device is myserial

```
udevadm info --attribute-walk --name=/dev/ttyUSB0 | grep devpath
```

```

jetson@jetson-desktop:~$ udevadm info --attribute-walk --name=/dev/ttyUSB0 | grep devpath
udevadm info starts with the device specified by the devpath and then
ATTRS{devpath}="2.4"
ATTRS{devpath}="2"
ATTRS{devpath}="0"
jetson@jetson-desktop:~$

```

What we need is to modify the rules for myserial in the Rules file:

```
# before modification:
# KERNEL=="ttyUSB*", ATTRS{idVendor}=="1a86", ATTRS{idProduct}=="7523",
# MODE:="0777", SYMLINK+="myserial"
# after modification:
KERNEL=="ttyUSB*", ATTRS{devpath}=="2.4", ATTRS{idVendor}=="1a86",
ATTRS{idProduct}=="7523", MODE:="0777", SYMLINK+="myserial"
```

Save and exit to make the rule take effect. Execute on the master:

```
sudo udevadm control --reload-rules && sudo udevadm trigger
```

Check whether the binding is successful:

```
jetson@ubuntu:~$ ll /dev | grep ttyUSB*
lrwxrwxrwx 1 root root 7 5月 18 20:13 gps1 -> ttyUSB1
lrwxrwxrwx 1 root root 7 5月 18 20:13 myserial -> ttyUSB0 #pcb binding
ttyUSB0 port
lrwxrwxrwx 1 root root 7 5月 18 20:13 rplidar -> ttyUSB1 #radar
binding ttyUSB1 port
lrwxrwxrwx 1 root root 7 5月 18 20:13 myspeech -> ttyUSB2 #Voice
control panel binding ttyUSB2 port
crwxrwxrwx 1 root dialout 188, 0 5月 18 20:13 ttyUSB0
crwxrwxrwx 1 root dialout 188, 1 5月 18 20:13 ttyUSB1
crwxrwxrwx 1 root dialout 188, 1 5月 18 20:13 ttyUSB2
```

If the preceding information is displayed, the binding is successful.

If you do not understand the binding method above, you can also refer to the content of Chapter 14 of the course document <14.Voice control courses \ Voice control module port binding>