DWIN Linux Screen Development Guide (40&43 Series and ZOS-1 Series)

Contents

Chapter 1 Product Introduction

1 OS Version

linux 4.19 OS: buildroot (40&43 series)
debian10 (ZOS-1 series)

2 Shipping List

- Standard screen*1
- Antenna (for WIFI & Bluetooth)*1

3 Optional Accessories

You can contact the DWIN sales to purchase the following accessories:

- Camera: support DWIN-defined mipi interface horizontal display, 5mp
  support DWIN-defined mipi interface vertical display, 5mp
- Speaker: 8Ω2W, cable length 320mm, front-facing sound output
  8Ω0.8W, cable length 180mm, side-facing sound output
- 4G module: China/India region version
  European region version
  Australian region version

4 Accessories Recommended for Self-Preparation

- 12V 2A power supply
- USB TYPE-A to TYPE-C or Micro-USB adapter cable (according to the type of USB debugging interface indicated in the device specification)
- USB flash drive
- SD card
- Network cable
- The adapter board and related wires for connecting your serial port or power supply device
- Microphone (with socket 2PIN_1.25)
- USB driverless camera

Chapter 2 Tool Introduction

1 The Tools Provided

- Ubuntu 22.04 virtual machine package (with QT Creator and Debian toolchain built-in, and the password is 123)
- RK3566 toolchain: buildroot version

  debian version (already built into the virtual machine package, no additional download is required)
- adb tool: adb-fastboot
- Firmware update package, it contains DWIN-V1_0_0.tar. Meanwhile, there is a folder with the same name inside this compressed package, which contains four files:

  install.sh script file (used to write scripts for various required functions)

  logo folder (used to modify the boot logo)

  wpa_supplicant.conf (the configuration file for WiFi-related functions)

  emcversion (the version number of the update package)
- Burning tools: SD card burning tool: SDDiskTool_V1.69

  USB firmware burning tool: RKDevTool_v2.93
- USB burning driver tool: DriverAssitant_v5.1.1

2 Tools Which Might Require Self-Downloading

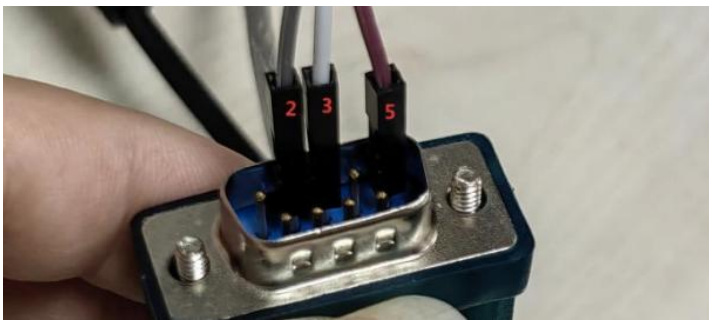
- VMware Workstation (It is recommended to use version 17)
- MobaXterm

Chapter 3 Debugging Method

Currently, DWIN Linux screens 40 series and ZOS-1 series support three debugging methods: serial port, ADB (recommended), and network port SSH.

1 Serial Port (The Operation Is the Same for 40&43 Series and ZOS-1 Series)

- Wiring example (actually subject to the interface type and definition indicated in the datasheet of the corresponding model. The debugging serial port is Serial Port 2).



- Power on the screen.
- Install MobaXterm on the computer and open it. Click "Session" - "New Session" in the top toolbar.
- Select the type "Serial".
- Modify the Serial port to the COM port found in the Device Manager.
- Set the baud rate to "1500000".
- Click "OK" and then you can start the operation.

2 ADB (The Operation Is the Same for the 40&43 Series and the ZOS-1 Series)

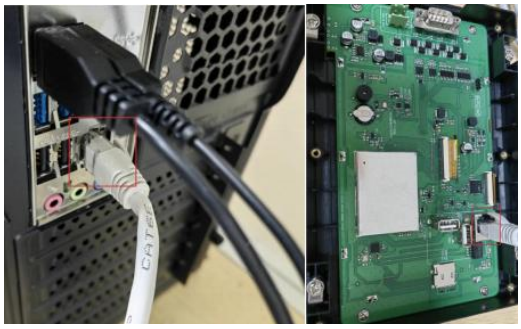
- Connect the device to the computer using a USB TYPE-A to TYPE-C or MICRO-USB adapter cable.
- Power on the screen.
- Open MobaXterm on the computer, and click "Session" - "New Session" in the top toolbar.
- Select the type "Shell".
- Select the terminal shell type "Windows PowerShell".
- Select the startup directory as the directory location where the adb_fastboot folder is located.

- Click "OK".
- → After entering ".\adb.exe shell", you can start the operation.

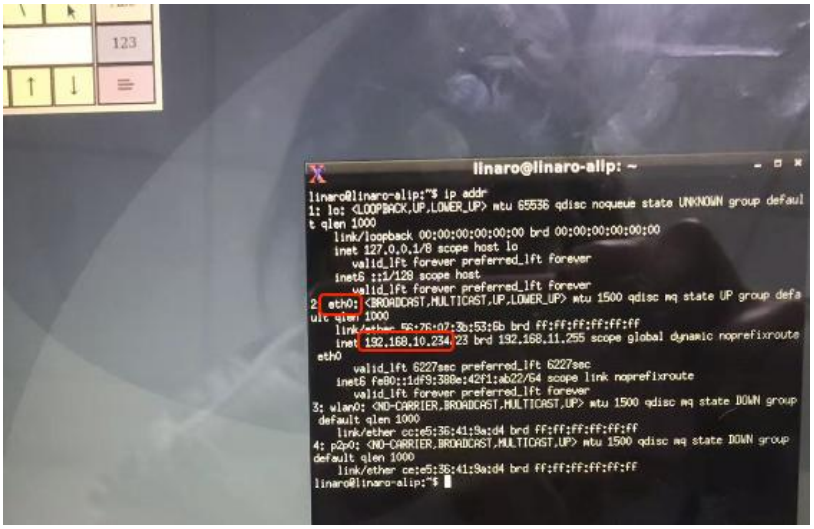
```
PS C:\Users\Administrator\Desktop\adb_fastboot> .\adb.exe  shell
root@RK356X:/#
root@RK356X:/#
```

3 Network Port SSH (There Are Operational Differences between 40&43 Series and ZOS-1 Series)

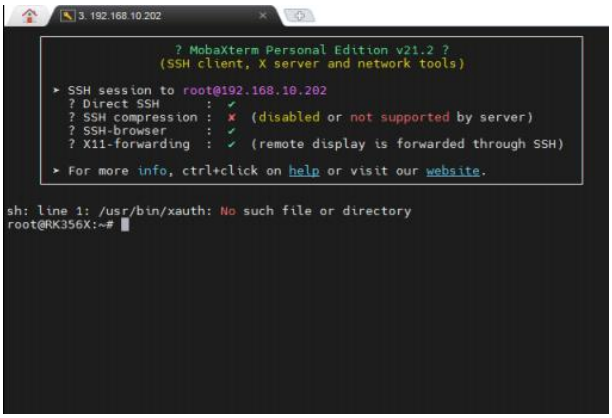
- Connect two new network cables to the router, and insert them into the network ports of the computer and the screen respectively.



- At this time, the device will automatically assign an IP address according to DHCP.
- For the 40 series, you can check the IP address in the default demo of the device.
- For the ZOS-1 series, after entering the desktop, click the menu in the lower left corner. In the menu, click "Universal Access" - "Onboard" to bring up the keyboard. It may take 2 to 3 seconds for the keyboard to load.
- Then click "System Tools" - "XTerm" in the menu. Enter the command "ip addr" in the terminal, and you can see the device IP in the information of "ethe0" as shown in the following figure.



- Open MobaXterm on the computer, click "Session" - "New Session" in the top toolbar, and select the type "SSH".
- Enter the device IP in the Remote host column.
- Click OK.
- At this time, power on the screen, and the following interface will be displayed.



- If you are using the 40 series linux screen, first enter the username "root" and press the Enter key, then enter the password "rockchip", and you can start the operation.
- If you are using the ZOS-1 series linux screen, first enter the username "linaro" and press the Enter key, then enter the password "linaro", and you can start the operation.

Chapter 4 Instruction Examples for Reference

1 Common Instructions of 40&43 Series and ZOS-1 Series

- Modify the volume level

Instruction: amixer set Master 5% (0%~100%)

The numbers from 1 to 9 are for setting the volume to 10, 20, 30... respectively.

Press Shift + "+" or "-" to control the increase or decrease of the single-digit volume.

```
root@RK356X:/# amixer set Master 5%
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch pswitch-joined
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 65536
  Mono:
  Front Left: Playback 3277 [5%] [on]
  Front Right: Playback 3277 [5%] [on]
root@RK356X:/# amixer set Master 50%
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch pswitch-joined
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 65536
  Mono:
  Front Left: Playback 32768 [50%] [on]
  Front Right: Playback 32768 [50%] [on]
```

- Modify the time

The instructions are as follows:

data: check the system time

data -s: set the system time

hwclock: check the hardware clock

hwclock -w: write the system time to the hardware clock

```
root@RK356X:/etc/network# date
Fri Sep 27 07:25:10 UTC 2024
root@RK356X:/etc/network# date -s "2023-10-10 10:10:10"
Tue Oct 10 10:10:10 UTC 2023
root@RK356X:/etc/network# hwclock
Fri Sep 27 07:25:47 2024  0.000000 seconds
root@RK356X:/etc/network# hwclock -w
root@RK356X:/etc/network# hwclock
Tue Oct 10 10:10:35 2023  0.000000 seconds
root@RK356X:/etc/network#
```

- Modify the boot LOGO

Create a folder named "logo" on your computer. Prepare two BMP format pictures (a maximum of two pictures, not a boot animation), name them "logo" and "logo_kernel" respectively, and put them into the "logo" folder. Then, copy the "logo" folder to a USB flash drive. Finally, insert the USB flash drive into the USB port of the

device.

Open MobXterm on the computer, type "df -h" and then press the Enter key. You can see the mounted devices. Locate the path of the USB flash drive (you can confirm it by plugging in and unplugging the USB flash drive). For example:

```
root@RK356X:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       5.9G  650M  5.0G  12% /
devtmpfs        972M     0  972M   0% /dev
tmpfs           981M  140K  981M   1% /tmp
tmpfs           981M  284K  981M   1% /run
tmpfs           981M     0  981M   0% /dev/shm
/dev/mmcblk0p7  123M   13M  104M  11% /oem
/dev/mmcblk0p8  991M   52K  941M   1% /userdata
/dev/sda1        29G   79M   29G   1% /media/udisk0
```

Replace the part with the red background in the following command with the path of the USB flash drive, and then run this command either in the MobaXterm session window created in the third chapter or in the terminal of the Debian device.

```
mkdir -p /userdata/custom_logo
cp -r /media/udis0/logo/* /userdata/custom_logo
```

2 Reference Instructions for ZOS-1 Series

- Wi-Fi

Search for Wi-Fi

```
nmcli dev wifi list
```

```
root@linaro-alip:/# nmcli dev wifi list
IN-USE  BSSID              SSID                  MODE   CHAN  RATE        SIGNAL  BARS  SECURITY
        88:25:93:5D:7E:4A  DWIN-813              Infra  6     405 Mbit/s  57      ▂▄▆_  WPA1 WPA2
        D0:76:E7:12:13:B0  TP-LINK_13B0          Infra  1     405 Mbit/s  47      ▂▄__  --
        4C:10:D5:3B:A2:88  2.4G-AP               Infra  6     405 Mbit/s  39      ▂▄__  WPA1 WPA2
        0A:71:90:37:CE:22  DIRECT-OAKWIOKmsPK    Infra  7     130 Mbit/s  34      ▂▄__  WPA2
        32:24:A9:07:D9:FA  DIRECT-fa-HP 8133     Infra  6     65 Mbit/s   32      ▂▄__  WPA2
```

Show Wi-Fi connections

```
nmcli connection show
```

```
root@linaro-alip:/# nmcli connection show
NAME                UUID                                  TYPE      DEVICE
DWIN-813            395e7634-bfbc-4753-8b25-728ed626b979  wifi      p2p0
4G-UFI-6C37         d46bd0d3-4edb-403b-9473-e36f7cd089be  wifi      --
A8CC                eb4046a1-850f-483e-abe0-ec8633c2b37b  wifi      --
DIRECT-fa-HP 8133   8bf27efd-170d-41f4-a442-1414f94937ee  wifi      --
Wired connection 1  861bbc46-b69b-3fb9-a164-62e11d3f07fa  ethernet  --
```

Delete Wi-Fi connections

```
nmcli con delete id "DWIN-813"
```

```
root@linaro-alip:/# nmcli con show
NAME                UUID                                  TYPE      DEVICE
Wired connection 1  861bbc46-b69b-3fb9-a164-62e11d3f07fa  ethernet  --
```

Add a new Wi-Fi connection

```
nmcli dev wifi connect "DWIN-813" password "123456"
```

```
root@linaro-alip:/# nmcli con show
NAME                UUID                                  TYPE      DEVICE
DWIN-813            f6caf5d7-e52d-4352-8b50-71d059639f45  wifi      p2p0
Wired connection 1  861bbc46-b69b-3fb9-a164-62e11d3f07fa  ethernet  --
```

```
root@linaro-alip:/# nmcli dev wifi connect "Redmi K50 Ultra" password "[redacted]"
Device 'p2p0' successfully activated with '7f8d8286-c80c-418e-9713-2182f94ba6d8'.
root@linaro-alip:/# nmcli con show
NAME                UUID                                  TYPE      DEVICE
Redmi K50 Ultra     7f8d8286-c80c-418e-9713-2182f94ba6d8  wifi      p2p0
DWIN-813            5a5575b3-b114-4d02-ba83-44700aa4286e  wifi      --
Wired connection 1  861bbc46-b69b-3fb9-a164-62e11d3f07fa  ethernet  --
```

Before adding a new Wi-Fi connection, you need to delete other invalid connections.

- 4G Network

```
nmcli con add type gsm ifname "xxx" con-name "xxx"
```

lsusb: check if the 4G module has been successfully loaded.

```
root@linaro-alip:/# lsusb
Bus 006 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 005 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 004 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 002 Device 006: ID 2c7c:6005 Quectel Wireless Solutions Co., Ltd. Android
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

```
ifname "": nmcli d
```

```
root@linaro-alip:/# nmcli d
DEVICE          TYPE      STATE         CONNECTION
ttyUSB2         gsm       disconnected  --
p2p0            wifi      disconnected  --
wlan0           wifi      disconnected  --
p2p-dev-p2p0    wifi-p2p  disconnected  --
p2p-dev-wlan0   wifi-p2p  disconnected  --
eth0            ethernet  unavailable   --
lo              loopback  unmanaged     --
```

con-name "": customize the connection name so that it can be quickly referenced during subsequent operations.

```
root@linaro-alip:/# nmcli con add type gsm ifname "ttyUSB2" con-name "4g"
Connection '4g' (a84b4365-bc93-4995-937a-76459346f4fe) successfully added.
root@linaro-alip:/# nmcli connection show
NAME                UUID                                  TYPE      DEVICE
4g                  a84b4365-bc93-4995-937a-76459346f4fe  gsm       ttyUSB2
Wired connection 1  1613a081-861c-3fc0-8c51-9a49a8f9b1a1  ethernet  --
```

Priorityw: wired connection > wifi > 4g

```
root@linaro-alip:/# nmcli connection show
NAME                UUID                                  TYPE      DEVICE
Wired connection 1  1613a081-861c-3fc0-8c51-9a49a8f9b1a1  ethernet  eth0
DWIN-813            546801f5-c6b0-4732-b7d7-e8a8fcd8bb14  wifi      p2p0
4g                  a84b4365-bc93-4995-937a-76459346f4fe  gsm       ttyUSB2
```

- Bluetooth

Run the instruction "bluetoothctl" to enter the Bluetooth settings interface.

scan on

```
[bluetooth]# scan on
Discovery started
[CHG] Controller F0:A8:82:32:02:11 Discovering: yes
[NEW] Device 41:AB:9A:FE:A7:B6 41-AB-9A-FE-A7-B6
[NEW] Device 6C:A5:9D:3B:67:EA 6C-A5-9D-3B-67-EA
[NEW] Device 5C:F1:A1:4D:2E:68 5C-F1-A1-4D-2E-68
[NEW] Device 5C:60:BA:FC:9F:87 5C-60-BA-FC-9F-87
[NEW] Device 4A:51:B0:04:54:D3 4A-51-B0-04-54-D3
[NEW] Device 6D:02:91:26:E0:CA 6D-02-91-26-E0-CA
[NEW] Device 7E:65:4D:A7:45:17 7E-65-4D-A7-45-17
[NEW] Device 53:41:48:A4:AD:50 53-41-48-A4-AD-50
[NEW] Device 65:15:7F:50:CC:63 65-15-7F-50-CC-63
[NEW] Device A4:C1:38:BD:1A:69 LYWSD03MMC
[NEW] Device 48:02:86:8E:2C:7F 真我GT Neo2
[NEW] Device 52:B0:54:7E:9F:5D 52-B0-54-7E-9F-5D
[NEW] Device EC:30:B3:40:0D:E2 1502026941的Redmi K50 Ultra
[CHG] Device 48:02:86:8E:2C:7F LegacyPairing: yes
[NEW] Device 7E:4A:C4:4B:41:6C 7E-4A-C4-4B-41-6C
```

scan off

pair [dev]

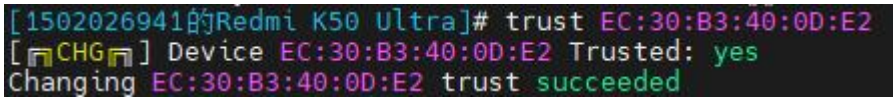
```
[bluetooth]# pair EC:30:B3:40:0D:E2
Attempting to pair with EC:30:B3:40:0D:E2
[DEL] Device 41:AB:9A:FE:A7:B6 41-AB-9A-FE-A7-B6
[DEL] Device 5C:F1:A1:4D:2E:68 5C-F1-A1-4D-2E-68
[DEL] Device 5C:60:BA:FC:9F:87 5C-60-BA-FC-9F-87
[DEL] Device 6C:A5:9D:3B:67:EA 6C-A5-9D-3B-67-EA
[DEL] Device 4A:51:B0:04:54:D3 4A-51-B0-04-54-D3
[CHG] Device EC:30:B3:40:0D:E2 Connected: yes
Request confirmation
[agent] Confirm passkey 074696 (yes/no): [DEL] Device 7E:65:4D:A7:45:17 7E-65-4D-A7-45-17
[agent] Confirm passkey 074696 (yes/no): [DEL] Device 53:41:48:A4:AD:50 53-41-48-A4-AD-50
[agent] Confirm passkey 074696 (yes/no): [DEL] Device 65:15:7F:50:CC:63 65-15-7F-50-CC-63
[agent] Confirm passkey 074696 (yes/no): [DEL] Device 6D:02:91:26:E0:CA 6D-02-91-26-E0-CA
[agent] Confirm passkey 074696 (yes/no): [DEL] Device A4:C1:38:BD:1A:69 LYWSD03MMC
[agent] Confirm passkey 074696 (yes/no): [DEL] Device 52:B0:54:7E:9F:5D 52-B0-54-7E-9F-5D
[agent] Confirm passkey 074696 (yes/no): [DEL] Device 7E:4A:C4:4B:41:6C 7E-4A-C4-4B-41-6C
[agent] Confirm passkey 074696 (yes/no): [DEL] Device 48:02:86:8E:2C:7F 真我GT Neo2
[agent] Confirm passkey 074696 (yes/no): yes
```

```
[CHG] Device EC:30:B3:40:0D:E2 ServicesResolved: yes
[CHG] Device EC:30:B3:40:0D:E2 Paired: yes
Pairing successful
[CHG] Device EC:30:B3:40:0D:E2 ServicesResolved: no
[CHG] Device EC:30:B3:40:0D:E2 Connected: no
```

connect [dev]

```
[bluetooth]# connect EC:30:B3:40:0D:E2
Attempting to connect to EC:30:B3:40:0D:E2
[CHG] Device EC:30:B3:40:0D:E2 Connected: yes
Connection successful
[CHG] Device EC:30:B3:40:0D:E2 ServicesResolved: yes
[1502026941的Redmi K50 Ultra]#
```

```
trust [dev]
```

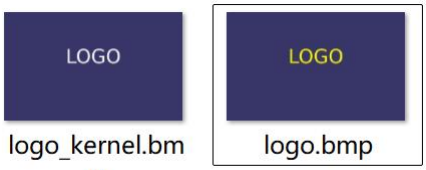


Chapter 5 Usage Guide for the Update Package (For 40&43 Series Only)

1 Boot LOGO Modification

Open the folder named "update" provided by our company, then open the folder named "DWIN_V1-0-0", and you can see a folder named "logo".

Prepare two BMP format pictures (a maximum of two pictures, not a boot animation), name them "logo" and "logo_kernel" respectively, and put them into the "logo" folder.



Open the install script file and add the following code to the script file:

```
#!/bin/sh

copy_dir()
{
  if [ -d $1 ]; then
    for libfile in $1/*; do
      if [ -f $libfile ]; then
        cp $libfile $2/
        chmod $3 $2/${libfile##*/}
       #echo $2/${libfile##*/}
      fi
    done
  fi
}
```

```
instdir=$(cd `dirname $0`; pwd)

#cp $instdir/emcversion /etc/

#cp -f $instdir/lib/libqlinuxfb.so /usr/local/Qt_5.12.5/plugins/platforms/
#chmod 755 /usr/local/Qt_5.12.5/plugins/platforms/libqlinuxfb.so
#
#cp -f $instdir/etc/runqt /etc/
#chmod 777 /etc/runqt

mkdir -p /userdata/custom_logo
cp -r $instdir/logo/* /userdata/custom_logo

sync
#$instdir/serio_app
```

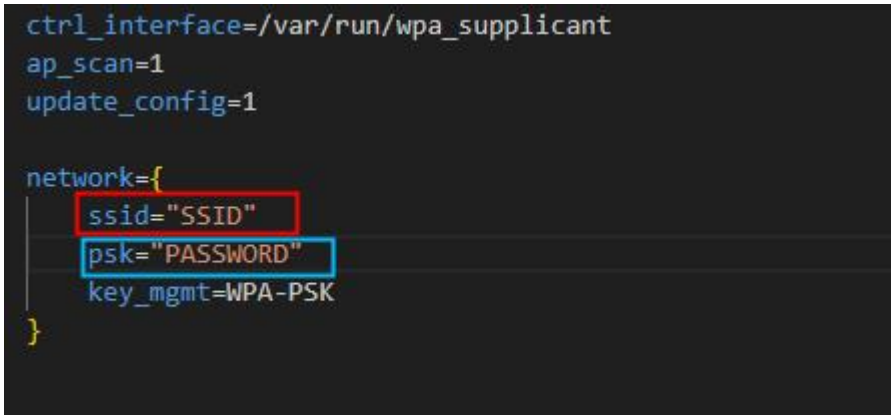
Save it. Then, compress the DWIN_V1-0-0 folder into a tar format. (It is not recommended to use Bandizip, as the hierarchy of the compressed package generated by it will be incorrect. For Windows 11 systems, you can use the default system compression tool. For other Windows system versions, it is recommended to use the 7-zip compression tool.) Store the compressed file in the "update" folder. After that, put the "update" folder into the USB flash drive

Insert the USB flash drive into the Linux screen and power it on again. Wait until the upgrade progress bar turns green, which indicates that the update is successful. Then, it will automatically shut down. At this time, remove the USB flash drive and power it on again.

2 WIFI Connection

Open the folder named "update" provided by our company, and then open the folder named "DWIN_V1-0-0". You can see the code file named "wpa_supplicant.conf".

The code is shown in the following figure. The content marked by the red box is the Wi-Fi name, and the content marked by the blue box is the Wi-Fi password. Users can modify the content within the double quotes according to the actual situation.



Save it after making the modifications.

Then open the install.sh file and add the code within the following green box.

```
mkdir -p /userdata/custom_logo
cp -r $instdir/logo/* /userdata/custom_logo
cp  $instdir/wpa_supplicant.conf   /etc
sync
```

Save it. Then, compress the DWIN_V1-0-0 folder into a tar format. (It is not recommended to use Bandizip, as the hierarchy of the compressed package generated by it will be incorrect. For Windows 11 systems, you can use the default system compression tool. For other Windows system versions, it is recommended to use the 7-zip compression tool.) Store the compressed file in the "update" folder. After that, put the "update" folder into the USB flash drive

Insert the USB flash drive into the Linux screen and power it on again. Wait until the upgrade progress bar turns green, which indicates that the update is successful. Then, it will automatically shut down. At this time, remove the USB flash drive and power it on again.

3 Time Zone Replacement

Open the folder named "update" provided by our company, then open the folder named "DWIN_V1-0-0". You'll find a script file named "install.sh". Add the following code to this script file.

```
ln -sf  /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
```

Here, the time zone is set as Shanghai as an example. You can change it according to your needs./Asia/Shanghai

Save it. Then, compress the DWIN_V1-0-0 folder into a tar format. (It is not recommended to use Bandizip, as the hierarchy of the compressed package generated by it will be incorrect. For Windows 11 systems, you can use the default system compression tool. For other Windows system versions, it is recommended to use the 7-zip compression tool.) Store the compressed file in the "update" folder. After that, put the "update" folder into the USB flash drive

Insert the USB flash drive into the Linux screen and power it on again. Wait until the upgrade progress bar turns green, which indicates that the update is successful. Then, it will automatically shut down. At this time,

remove the USB flash drive and power it on again.

Chapter 6 Product Introduction

1 Firmware Burning via SD Card

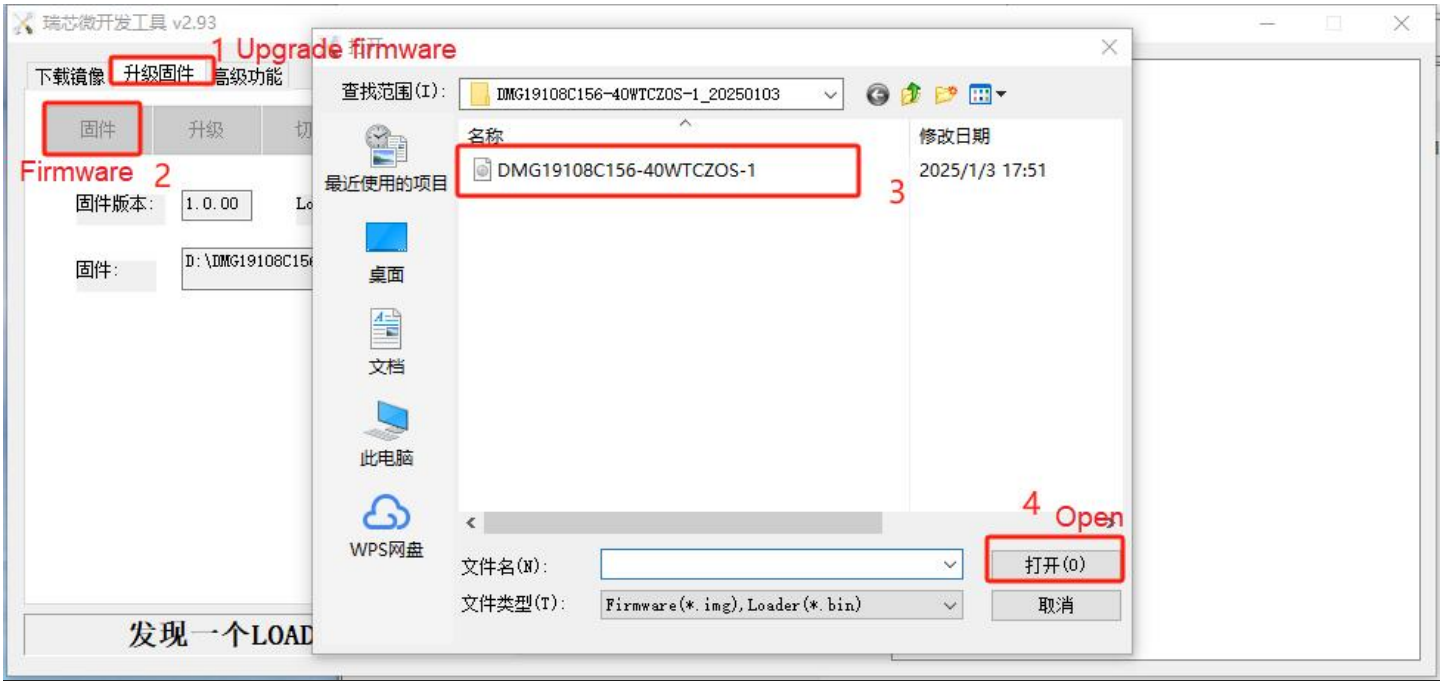
- When upgrading the firmware using an SD card, you need to write the firmware to the SD card using a tool on a computer. Currently, this operation can only be completed on a Windows operating system.
- Insert the card reader with the loaded SD card into the USB port of your computer. Open SDDiskTool_v1.69, and first select this card reader. Check the "Firmware Upgrade" box, then click "Select Firmware" to find the firmware to be programmed.

- After clicking "Create", just wait until the progress is completed.
- Remove the SD card and insert it into the SD card slot of the device. Power on the device with 12V DC. The screen will automatically start the upgrade. After the upgrade is completed, remove the SD card. The screen will automatically restart, indicating that the process is finished.

2 Firmware Burning via USB

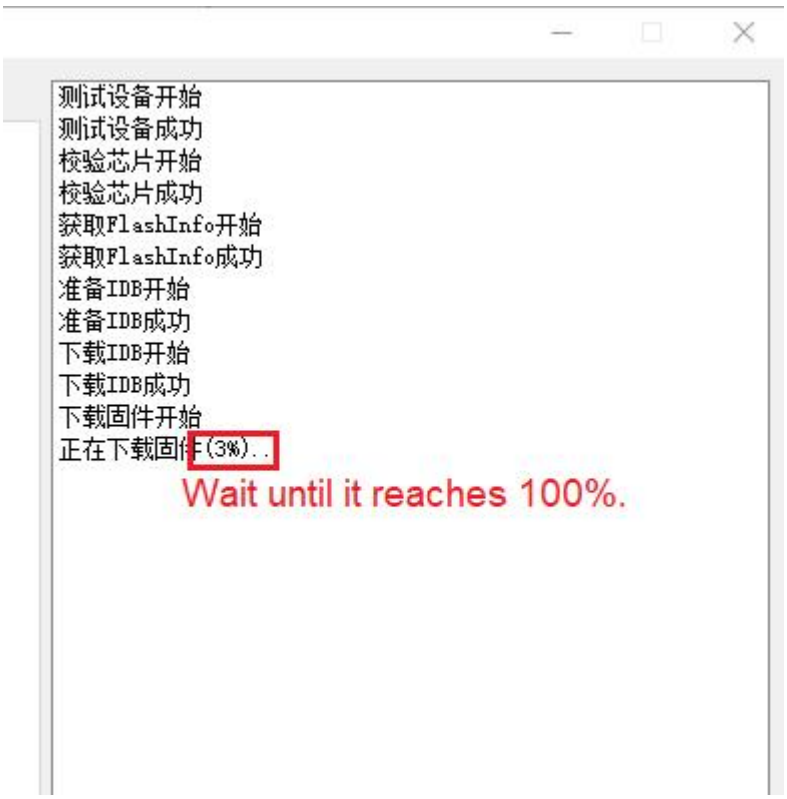
- If it's the first time for the computer used for programming to perform the programming operation, you need to install the driver. Open the "DriverAssitant_v5.1.1" folder of the USB programming driver tool we provided, click on "DriverInstall.exe", and then complete the installation according to the prompts.

- Open RKDevTool_v2.93, click on the second tab "Upgrade Firmware", then click on "Firmware" and upload the img firmware package that needs to be programmed.



- Next, first use a USB cable to connect the device to the computer, and then power on the device with 12V DC. At this time, the RKDevTool will display the status "Found an ADB device".

- Click the switching function of the tool, and wait until the status of the tool is updated to "Found a LOADER device". At this time, you can click "Upgrade" and wait for the progress to be completed. After the burning is finished, the device will restart.



.

Chapter 7 Environment Setup

1 Virtual Machine Configuration

- Self-download and install VMware Workstation.
- After the installation is completed, click "File" in the menu bar, then select "Open", and finally choose the Ubuntu 22.04 virtual machine package.
- Start the virtual machine and enter the password 123 to log in.

2 Toolchain Installation

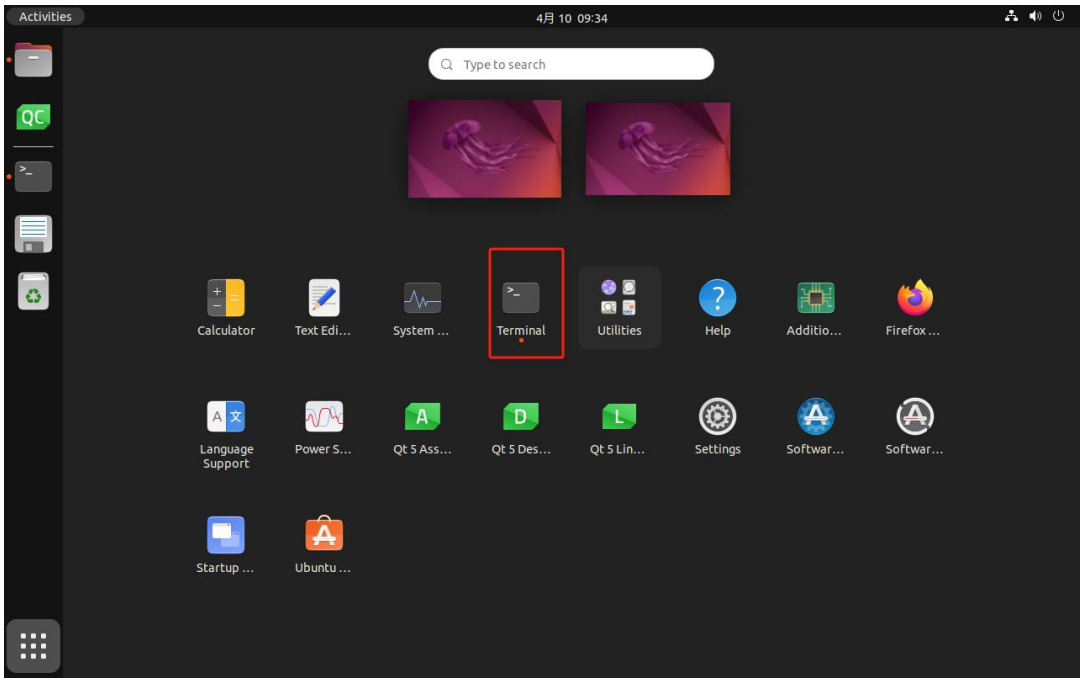
- If you are using the firmware provided by DWIN, we have already packaged the debian version toolchain required for the ZOS-1 series and placed it on the desktop.
- If you are using the 40 series screens, please follow the commands below. Drag and drop the downloaded buildroot version toolchain onto the ubuntu desktop, then extract and install it.
- In some cases, you may find that the toolchain you successfully dragged and dropped doesn't appear. This is because the dragged and dropped files are stored in the virtual machine at ~/.cache/vmware/drag_and_drop/. This is a hidden folder, and generally, we don't have access to it. Therefore, we need to create a folder on the desktop to store the dragged and dropped toolchain.

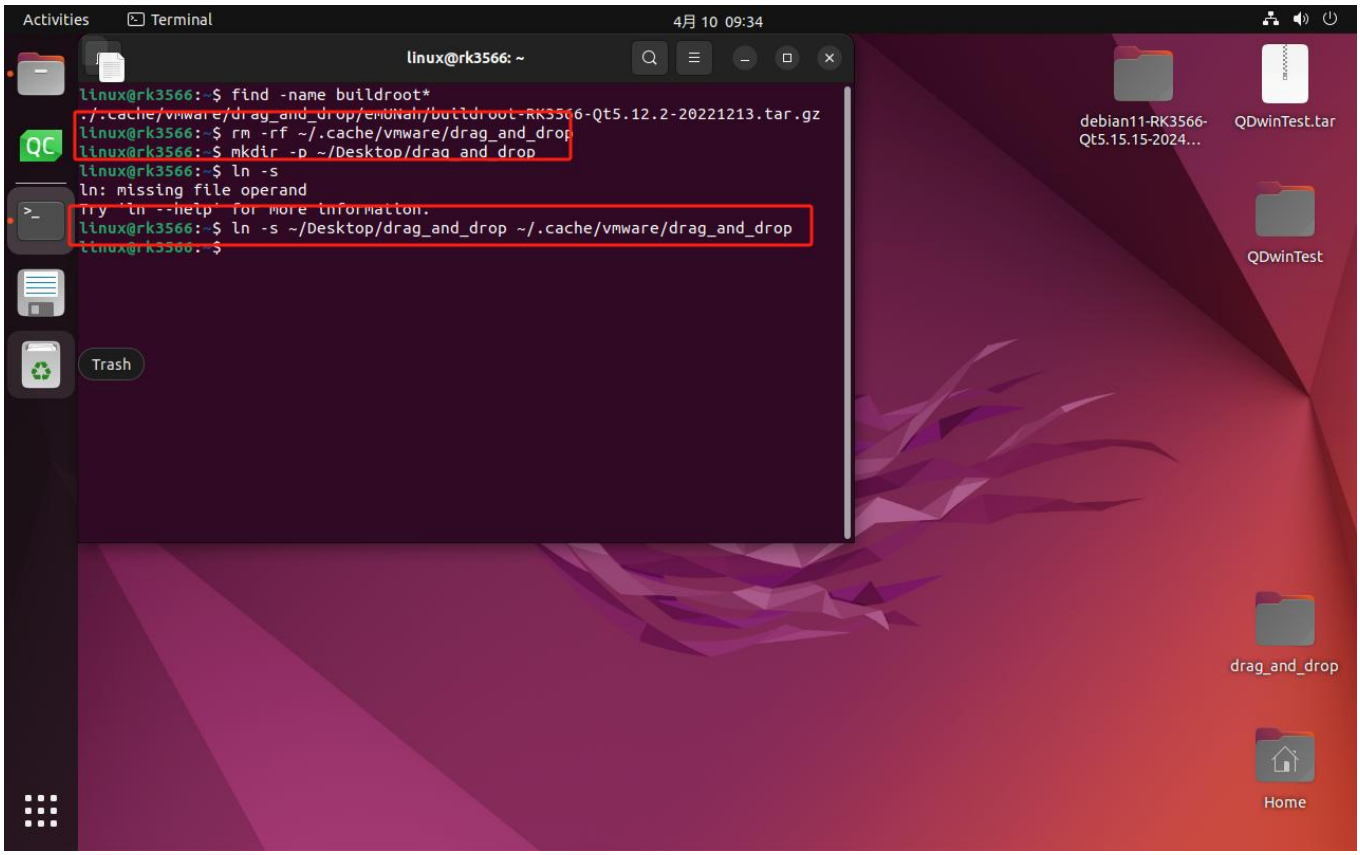
At this point, you can open the terminal of the ubuntu virtual machine and enter the following commands in sequence:

rm -rf ~/.cache/vmware/drag_and_drop #Delete the original cache directory

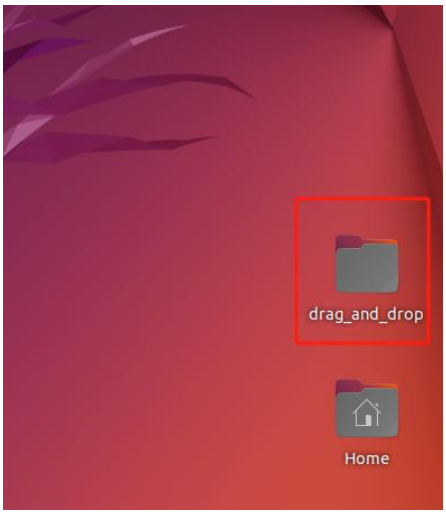
mkdir -p ~/Desktop/drag_and_drop #Create a new directory on the desktop

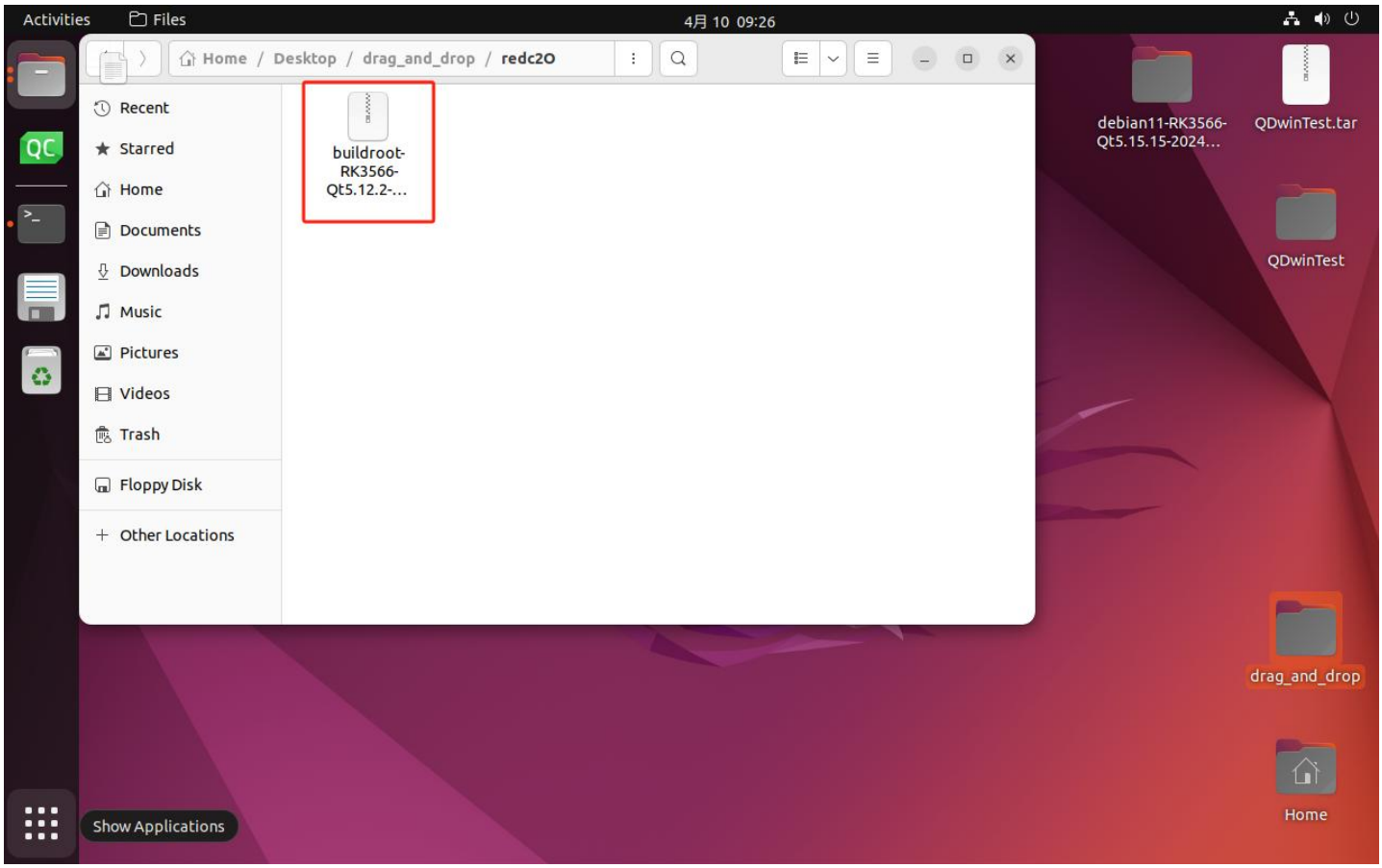
ln -s ~/Desktop/drag_and_drop ~/.cache/vmware/drag_and_drop #Create a symbolic link





In this way, a folder named "drag_and_drop" will appear on the desktop. The dragged and dropped files will directly appear in the folder with the path ~/Desktop/drag_and_drop.


drag_and_drop
Home


Activities
Files
4月 10 09:26
Home / Desktop / drag_and_drop / redc2O
Recent
Starred
Home
Documents
Downloads
Music
Pictures
Videos
Trash
Floppy Disk
Other Locations
buildroot-RK3566-Qt5.12.2-...
debian11-RK3566-Qt5.15.15-2024...
QDwinTest.tar
QDwinTest
drag_and_drop
Home
Show Applications

Chapter 8 Installation and Usage Suggestions for QT Creator

If you are using the ubuntu virtual machine file provided by DWIN and need to develop products in the ZOS-1 series, the virtual machine already contains the pre-configured Qt Creator. You can skip Subsections 1-5 of this chapter and directly click on Section 6 to start the compilation.

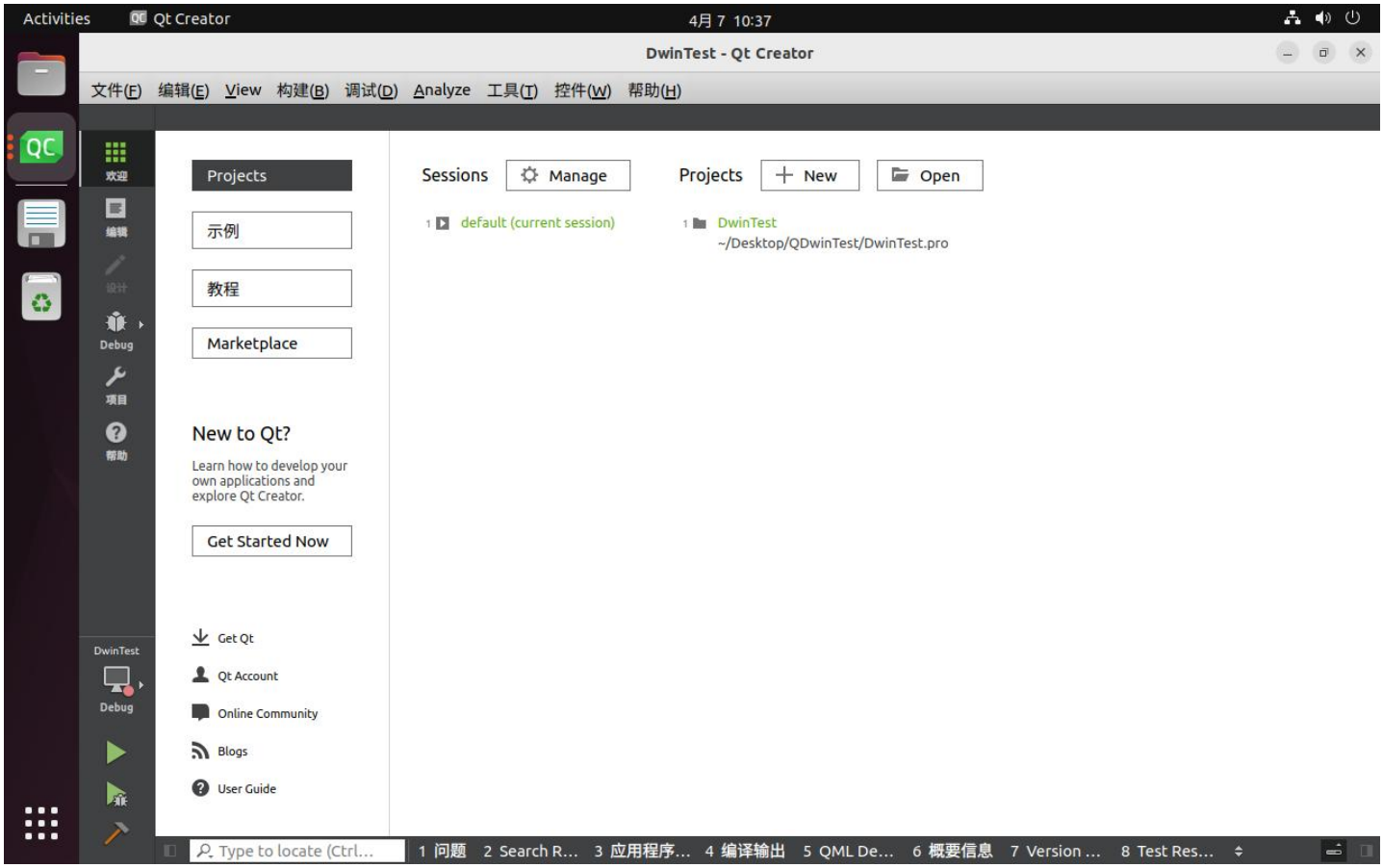
If you are using the ubuntu virtual machine file provided by DWIN and need to develop products in the 40&43 series, you need to start configuring the toolchain from Subsection 3 of this chapter before performing operations such as compilation.

If you create the ubuntu virtual machine by yourself, please read the entire content of this chapter.

1 Installation of Qt Creator in Ubuntu

Enter sudo apt install qtcreator in the Ubuntu terminal.

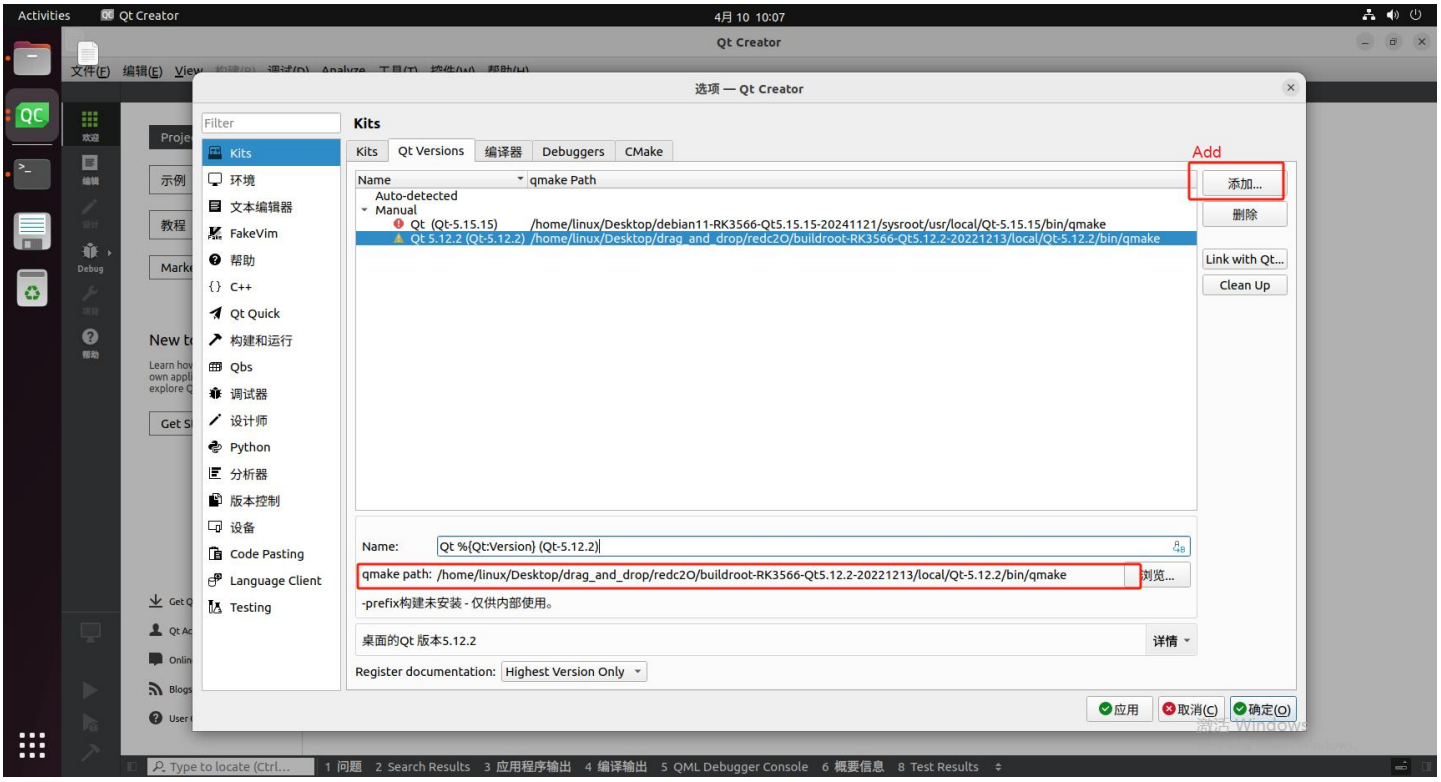
2 QT Creator Software Interface



3 Qmake Settings

Select "Tools" - "Options" - "Kits", then click "Qt versions" and click "Add".

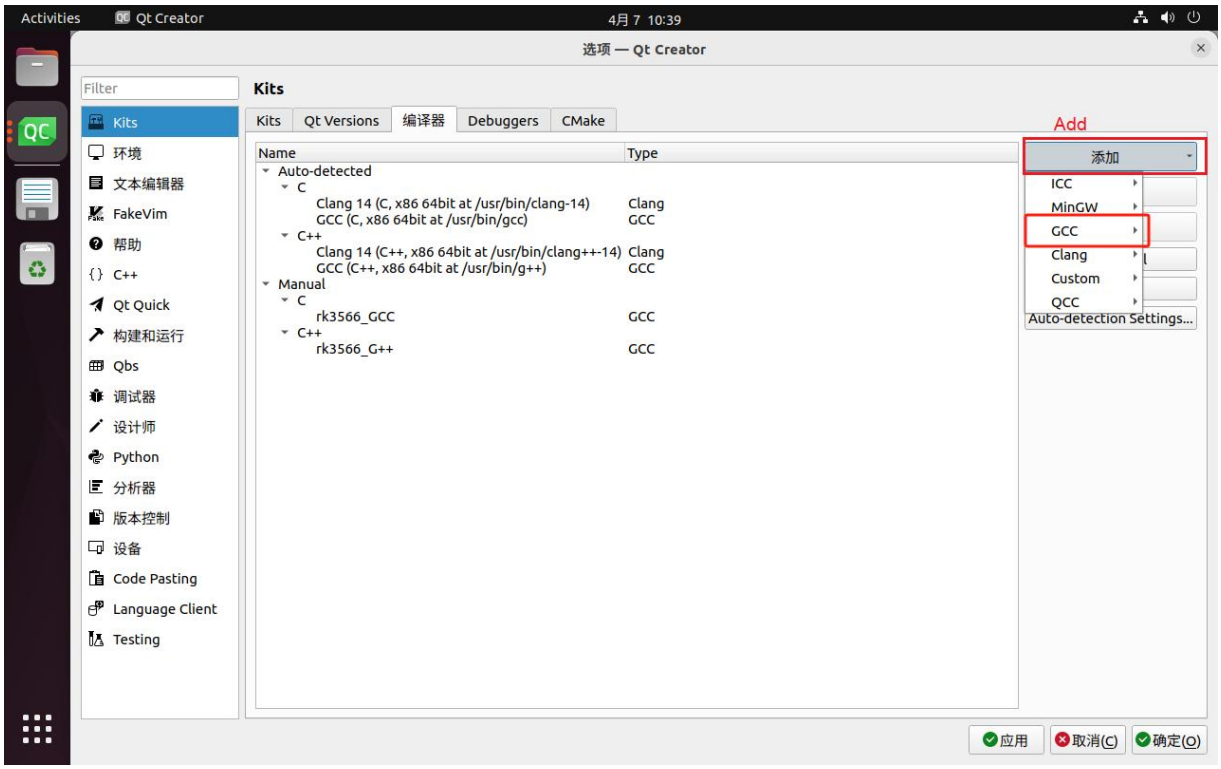
Qmake is generally located in the "local/Qt - 5.12.2/bin/" directory of the toolchain folder.



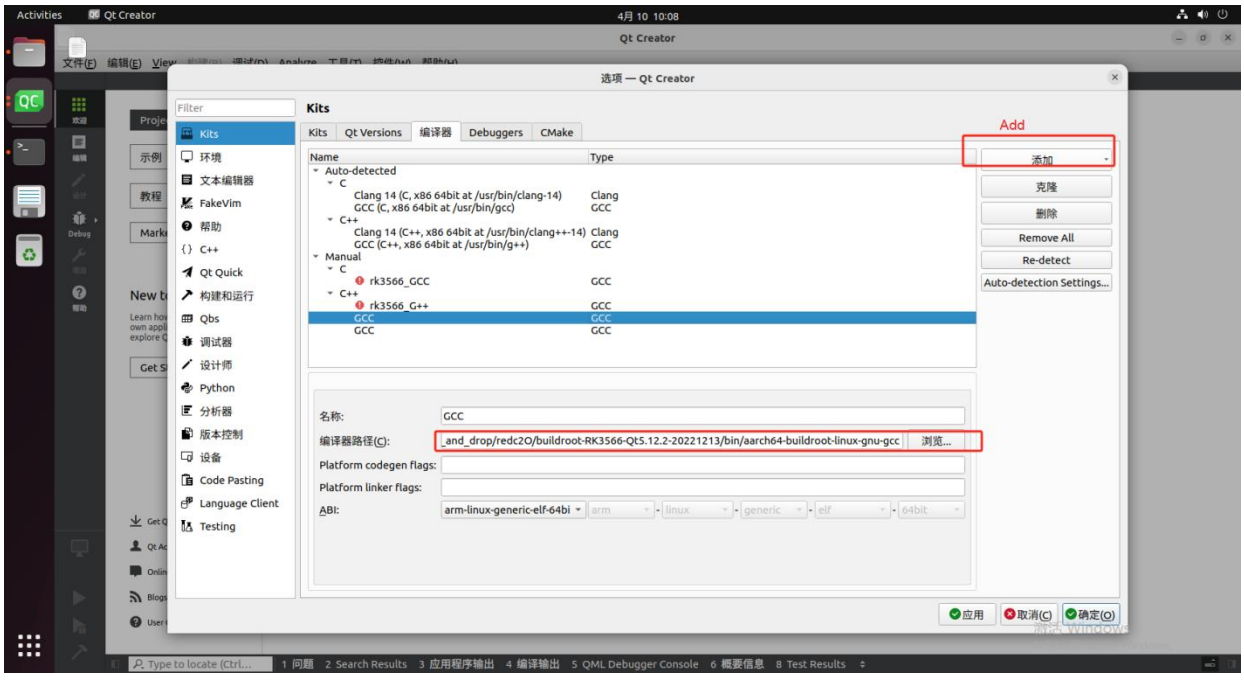
4 Toolchain Settings and Compilation

Select "Tools" - "Options" - "Kits" and click "Compilers". Here, it is recommended to add both C and C++ compilers at the same time. The steps are as follows:

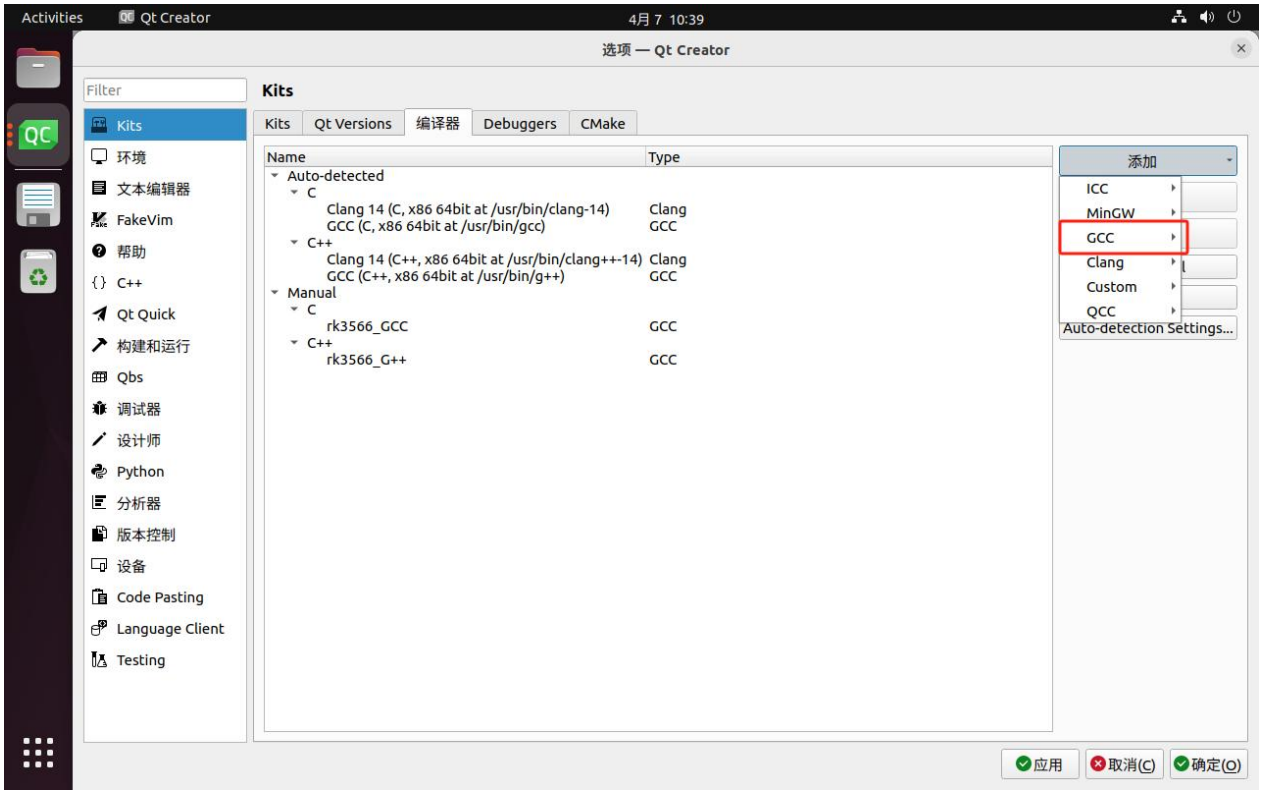
- Click "add"-GCC-C



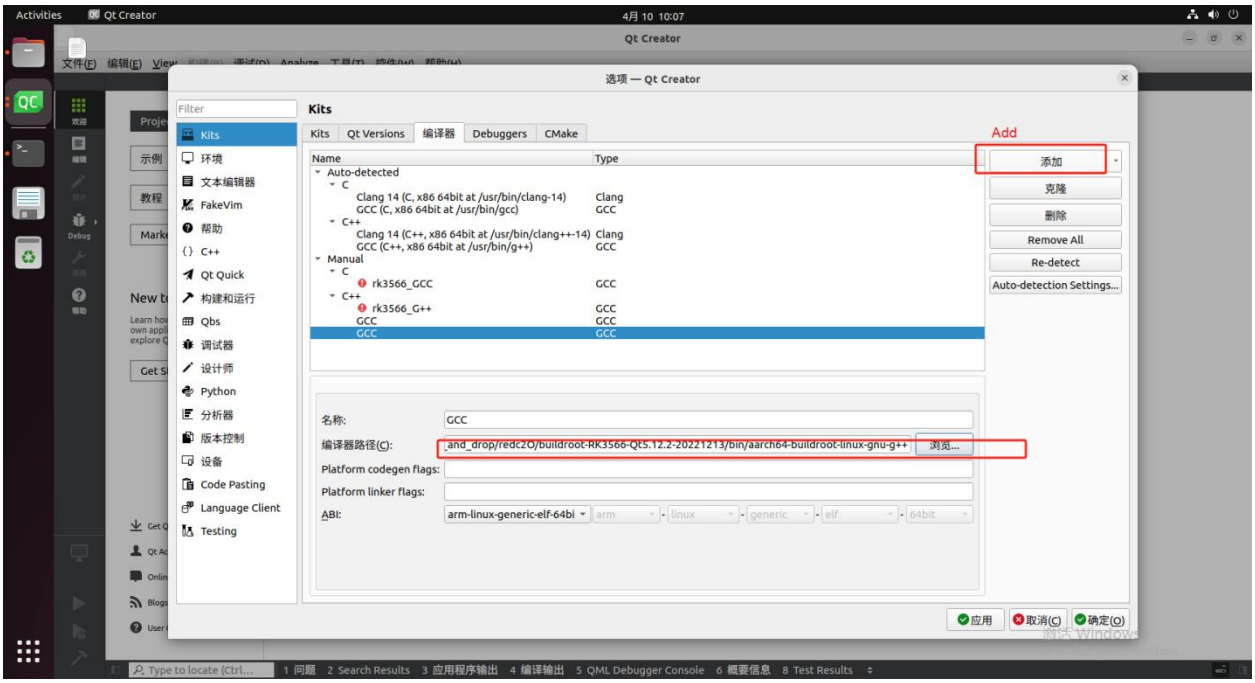
- Click the "Browse" button on the right side of the compiler path. In the bin directory of the toolchain package, select the compiler whose name ends with gcc.



- Repeat the above two steps and click "Add" - GCC-C++



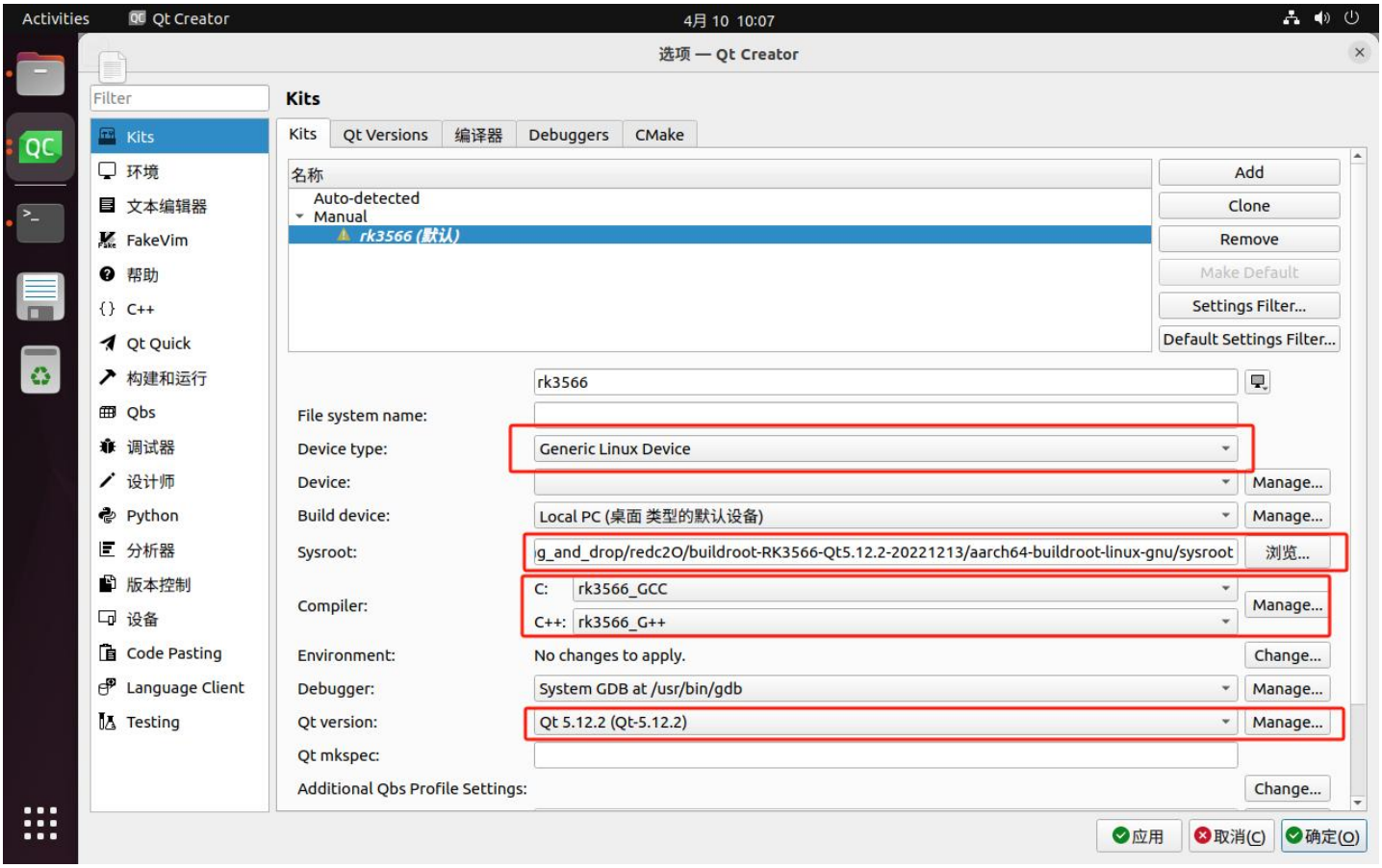
- Click the "Browse" button on the right side of the compiler path. In the bin directory of the toolchain folder, select the compiler whose name ends with g++.



5 Kits Configuration

Configure the Kits page as shown in the following figure..

The sysroot folder is generally located in the aarch64-buildroot-linux-gnu folder of the toolchain folder.



6 Environment Variable Configuration

- Open the project (here, take the qt demo included in our ubuntu virtual machine as an example).
- Add environment variables: click "Project" - "Build", find "Build Environment", and add the following four variables.

Variable Name 1: RK3566_SDK_PATH

Variable Value 1: The root directory of the buildroot-RK3566-Qt5.12.2-20221213.tar.gz package

Variable Name 2: RK3566_SYSROOT

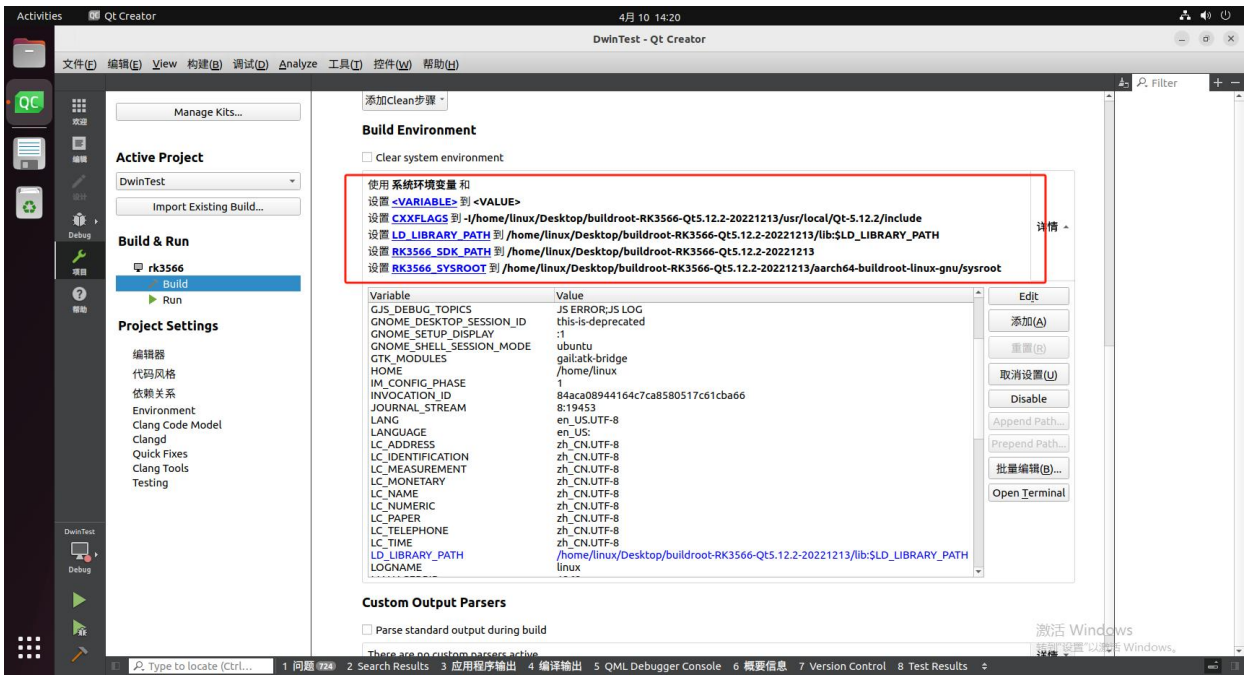
Variable Value 2: The aarch64-buildroot-linux-gnu/sysroot directory of the buildroot-RK3566-Qt5.12.2-20221213.tar.gz package

Variable Name 3: CXXFLAGS

Variable Value 3: The include directory of the buildroot-RK3566-Qt5.12.2-20221213.tar.gz package

Variable Name 4: LD_LIBRARY_PATH

Variable Value 4: The lib:$LD_LIBRARY_PATH directory of the buildroot-RK3566-Qt5.12.2-20221213.tar.gz package.

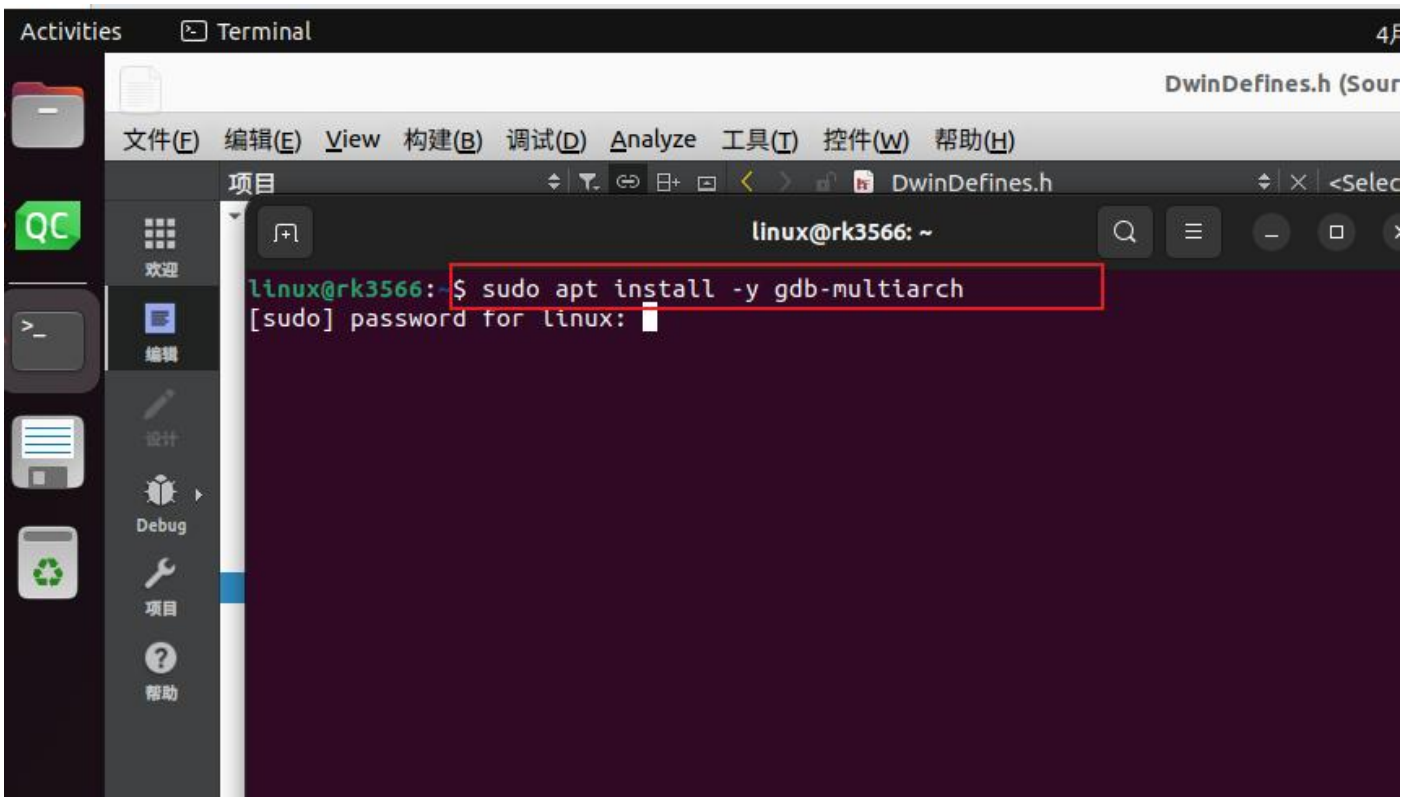


7 Online Deployment of Qt Program

This section will teach you how to deploy a Qt program on DWIN Linux screen via remote connection, allowing you to see the effects of program modifications in real time. If you don't need online debugging and want to directly compile the Qt program, you can skip this section and go straight to the content in Section 8.

- Set up the debugger

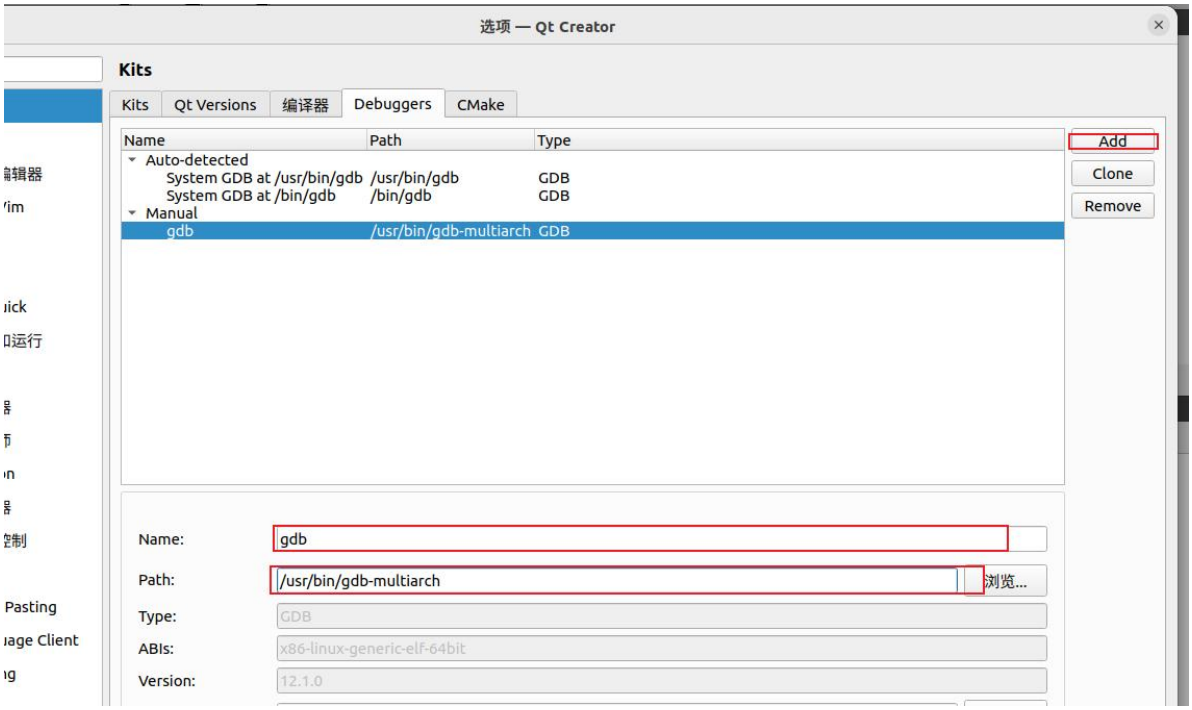
In the Ubuntu virtual machine, open the terminal and enter the command sudo apt install -y gdb-multiarch. Then input the virtual machine password 123 to install GDB required for remote debugging.



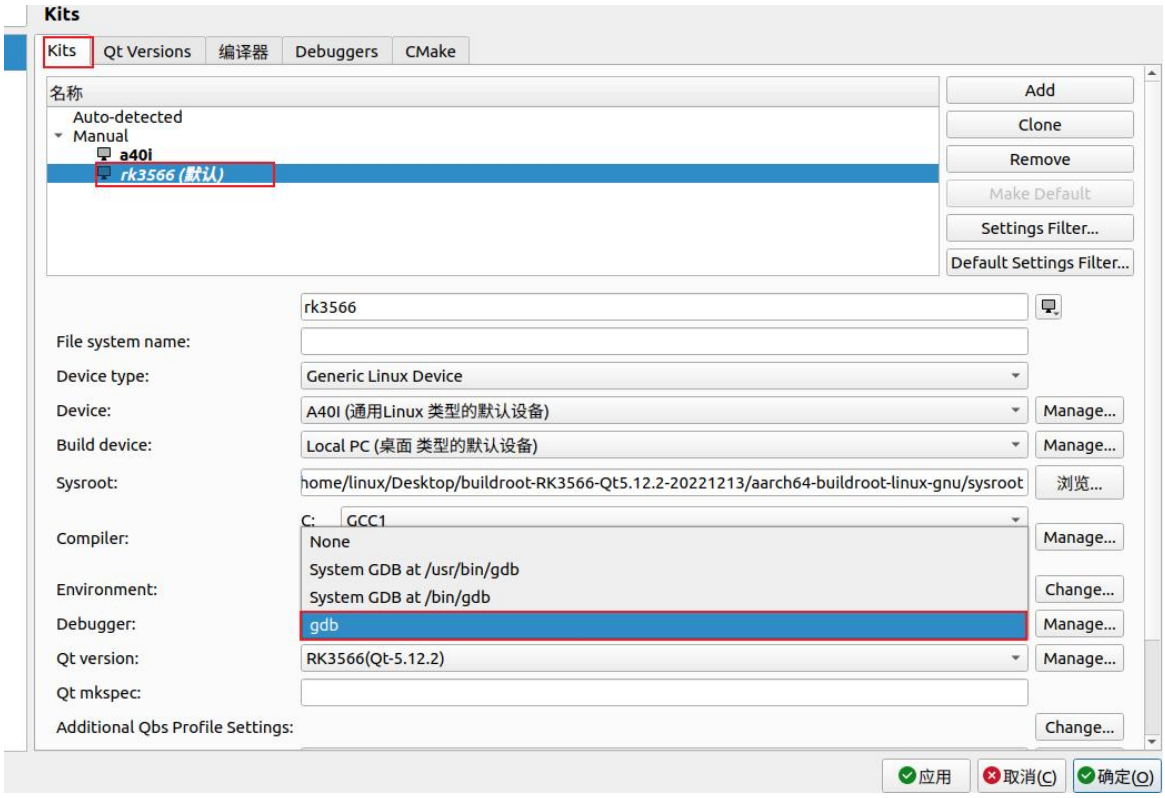
Open Qt Creator, click "Tools" - "Options" in the top menu bar. In the pop-up window, select "Debuggers", then click "Add". Fill in the content as shown in the figure.

Name: user-defined

Path: /usr/bin/gdb-multiarch

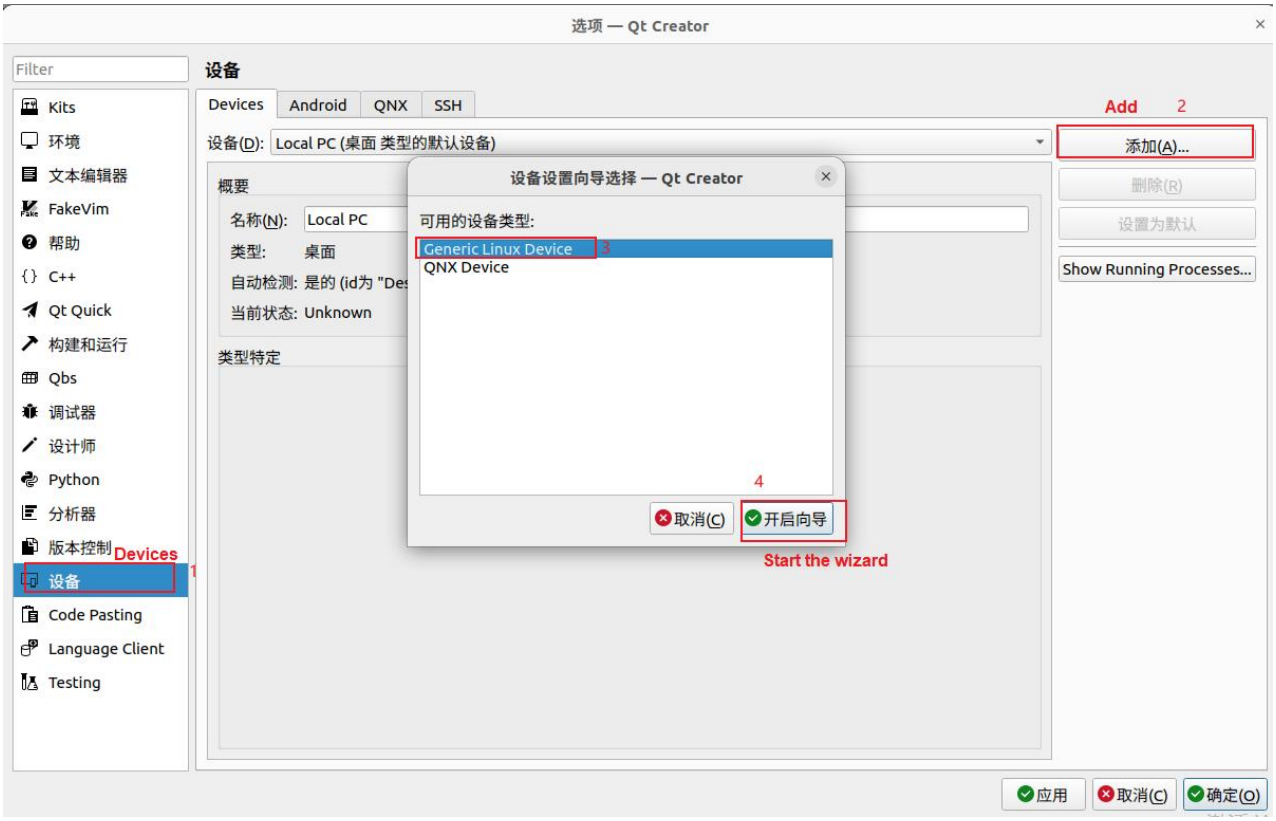


Click "Apply" in the bottom-right corner. Then, click the "Kits" tab in the pop-up window's menu bar. Select the kit you created earlier, and as shown in the figure below, choose the GDB you just created. Finally, click "Apply" again in the bottom-right corner.



- Add a remote device

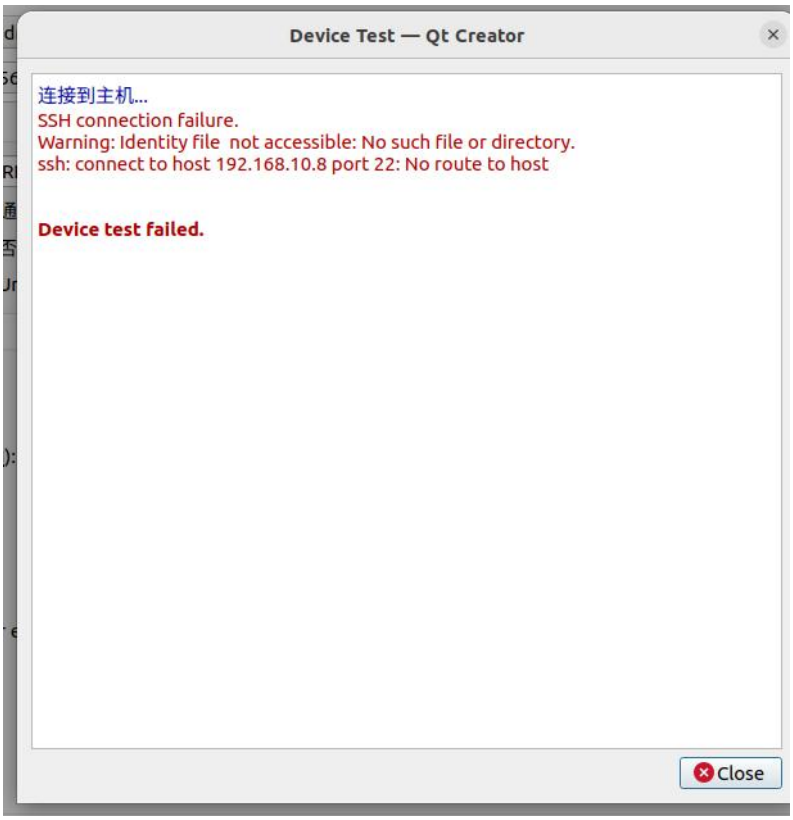
Click the "Devices" tab on the left, then click "Add". Set the device type to Generic Linux Device and start the wizard as shown below.



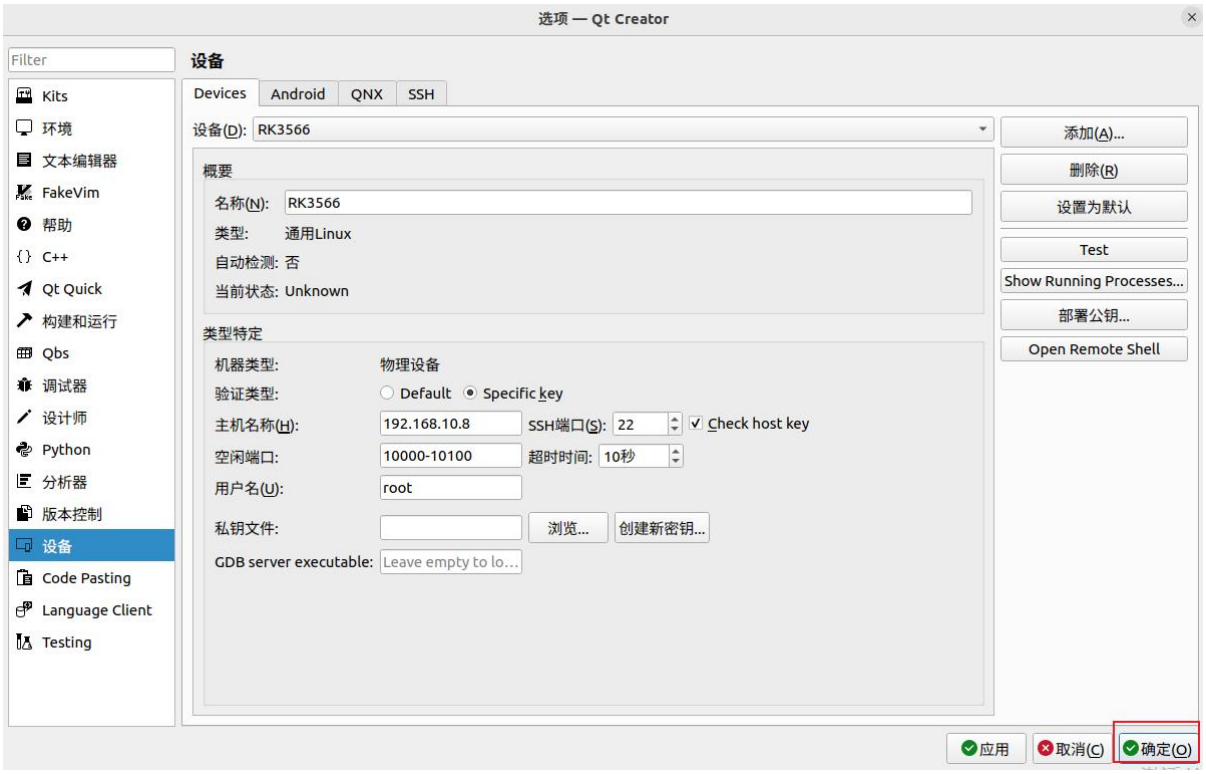
Continue with the parameter settings and here is an explanation of the parameters: The first column is the name you assigned to the DWIN Linux screen device in Qt Creator; the second column is the IP address automatically assigned to the DWIN Linux screen; the third column is the username used to indicate the DWIN Linux screen in the code during remote debugging connection.

It is recommended to directly skip the key setting. Click "Next".

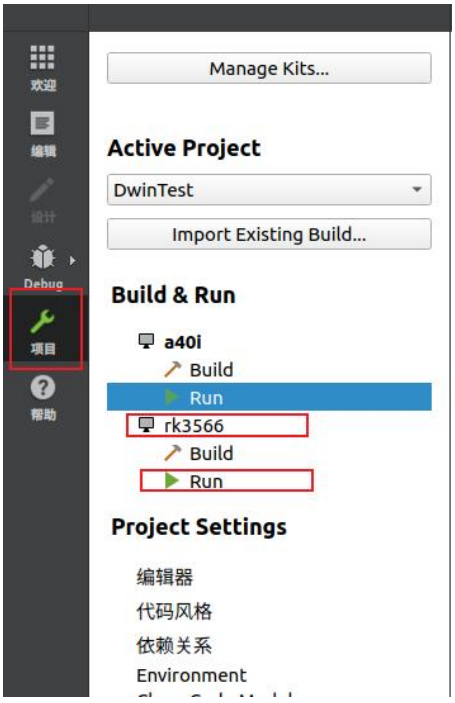
At this time, a device test failure will be displayed. This is because we still need to configure other parameters. Just click "Close" to exit.



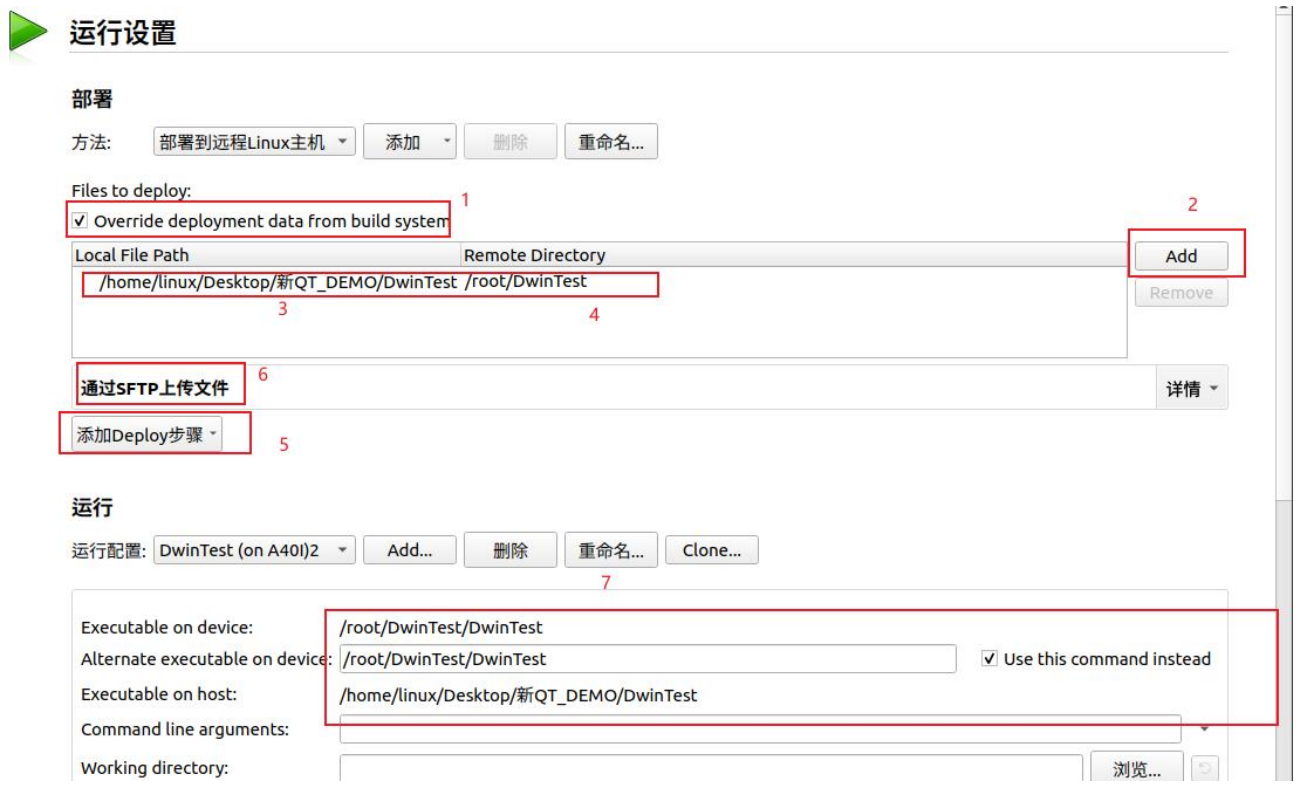
Click "OK" to exit the "Options" dialog box.



In the left menu bar of Qt Creator, click "Projects", select the "kits" that have just been configured, click "Run" to configure the running parameters.

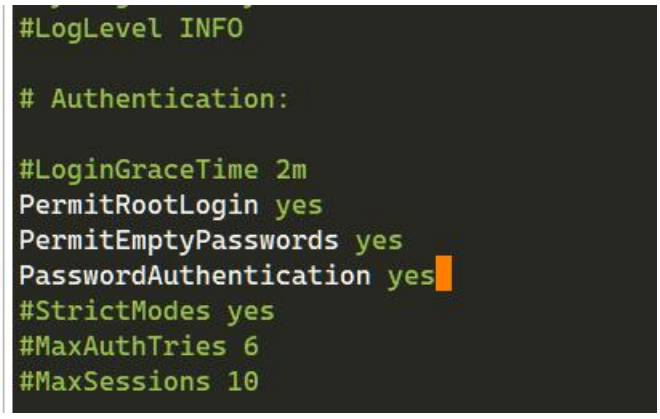


Configure as shown in the figure below. It should be noted that the path in the third step refers to the directory where the Qt project generates its executable file.



- Modify the screen configuration

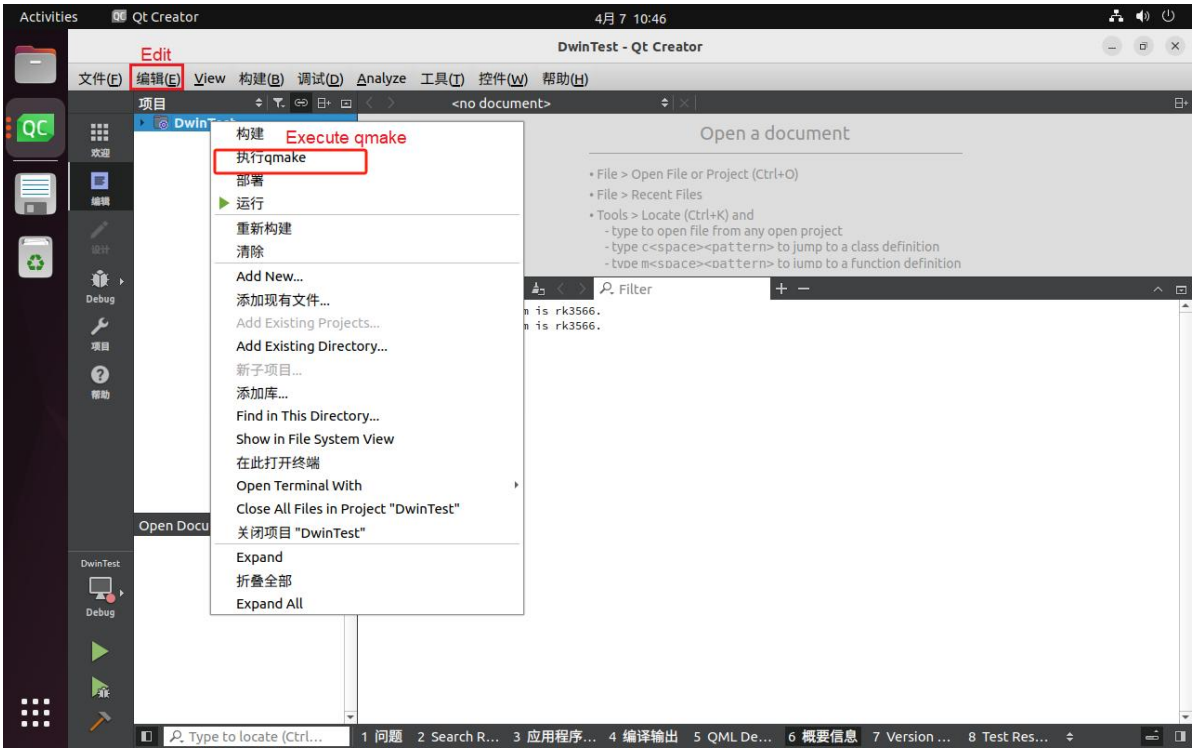
Move the cursor to the position shown in the figure below, press "i" on the keyboard to enter insert mode, and configure according to the image. After configuration, press "Esc", then type ":wq" and hit Enter to save and exit.



8 Project Compilation and Download

- Execute qmake

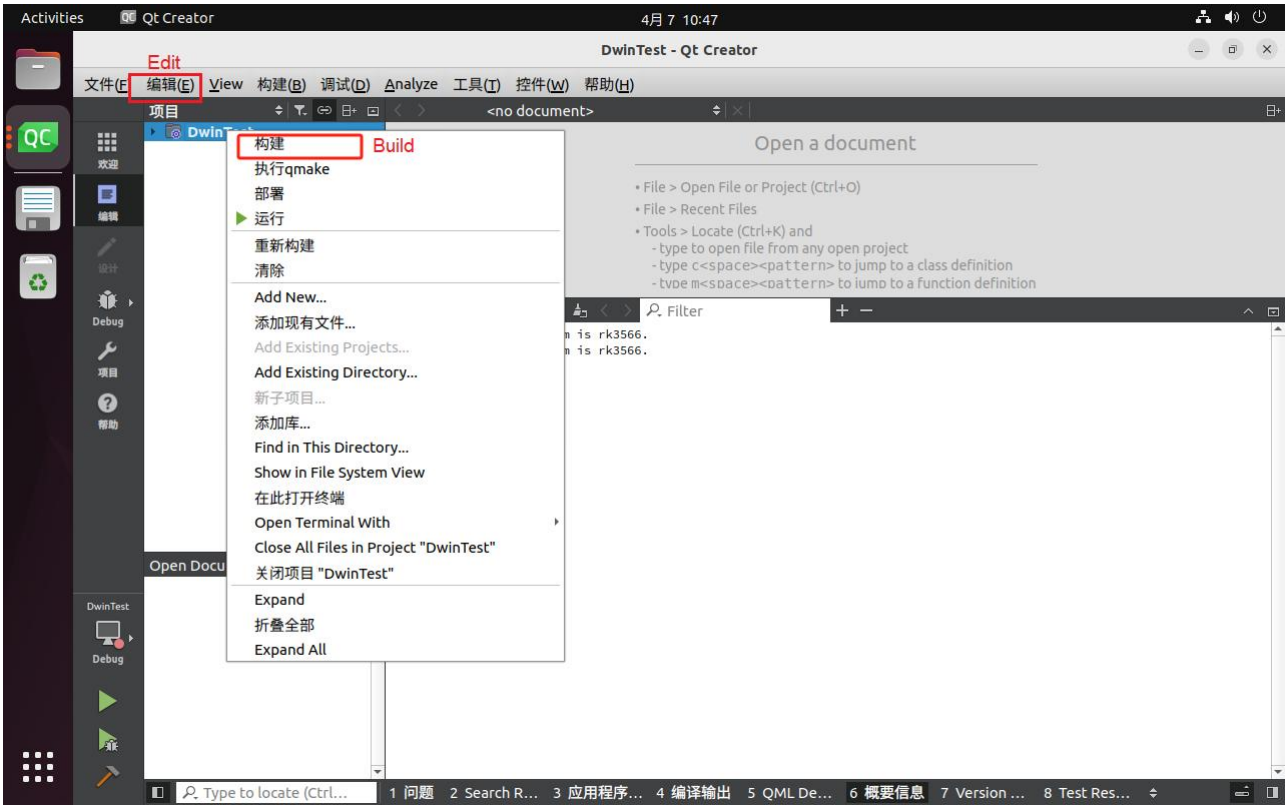
Click "Edit", select the demo project, right-click the mouse, and select "Execute qmake" from the menu bar.



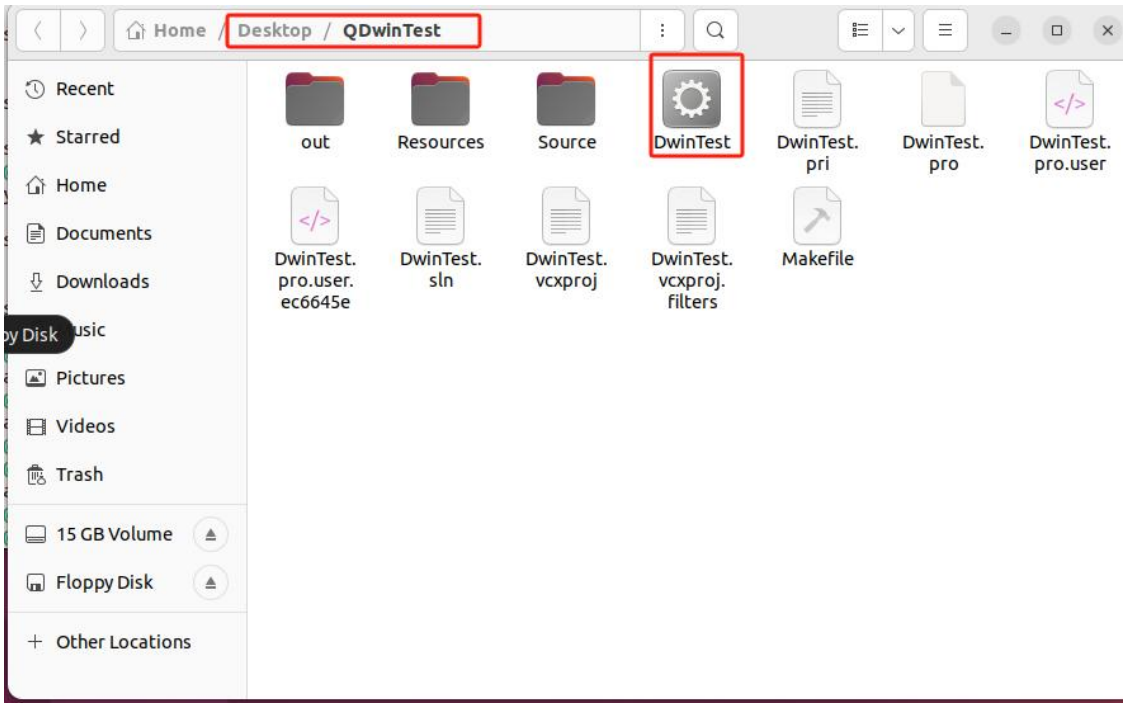
When qmake succeeds, it will be as shown in the following figure (the red part is the printout of DWIN_QT_DEMO.pri, which does not affect the result):

- Build

Click "Edit", select the demo project, right-click the mouse, and select "Build" from the menu bar.



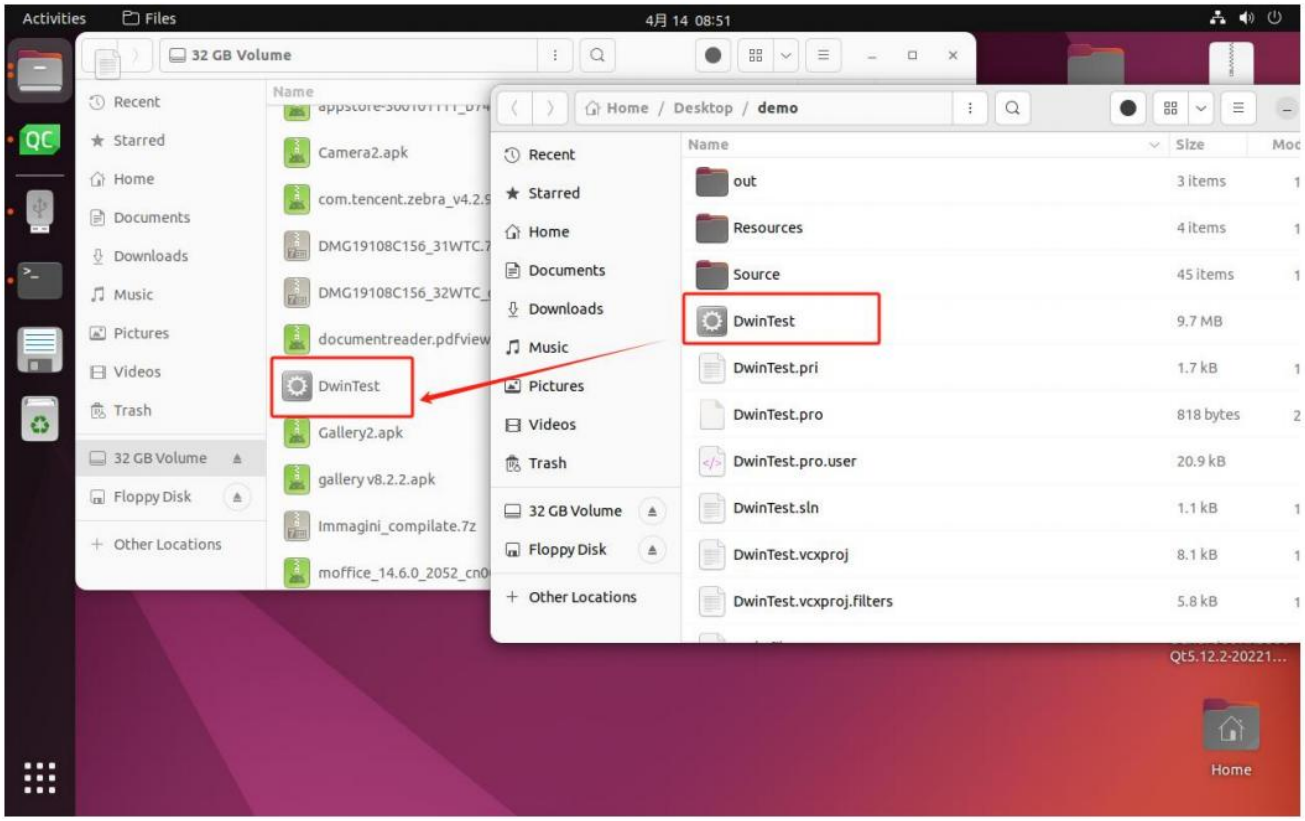
The target file has been generated in the project directory. To download it to the device, you need to recordits storage path.



- Download via USB flash drive

(1) Insert the USB flash drive into the computer and make selections in the order shown in the following figure.

(2) Drag the generated project files to the USB flash drive.



(3) Remove the USB flash drive from the computer and insert it into the device. Open the MobaXterm session window that you configured as required in Chapter 3. You can first use the df -h command instruction to query the path of the USB flash drive and the path on the device where you want to store the files. Here, we'll directly store the files in the root directory of the device. Use the cp command to copy the project files from the USB flash drive to the device. The reference code is as follows:

```
kunos login: root
Password:
# cp /mnt/usb/sda1/DwinTest /
cp: can't stat '/mnt/usb/sda1/DwinTest': No such file or directory
# df -h
Filesystem                Size      Used Available Use% Mounted on
/dev/root               991.9M    299.3M    676.6M  31% /
tmpfs                    44.7M         0     44.7M   0% /dev/shm
tmpfs                    54.4M     28.0K     54.3M   0% /tmp
tmpfs                    54.4M     24.0K     54.3M   0% /run
devtmpfs                 44.7M         0     44.7M   0% /dev
/dev/mmcblk0p8            6.0G     27.9M      5.7G   0% /extp
/dev/sda1                29.7G      4.8G     24.9G  16% /mnt/usb/sda1
# cp /mnt/usb/sda1/DwinTest /
```

df -h: search for the mounted device

/: The path of the root directory

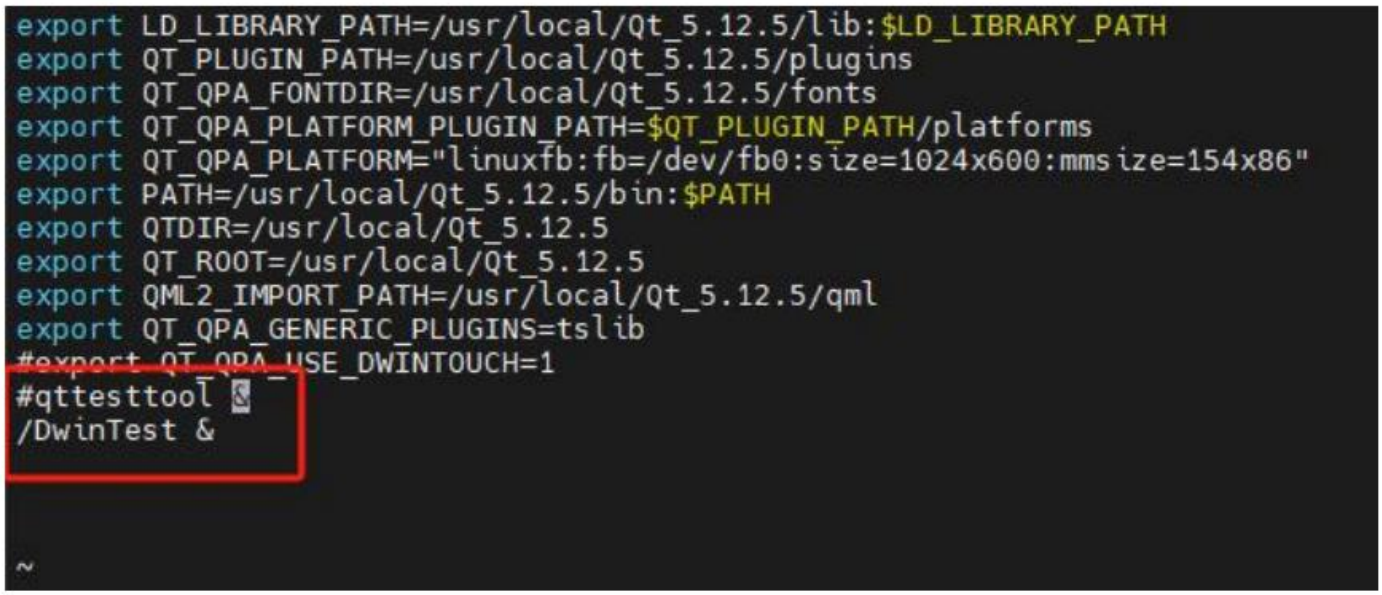
/mnt/usb/sdal: USB drive path

cp command: cp (space) the path of the file + the file name + (space) + the target path

- Run the qt project

Type vi (with a space here) /etc/init.d/S99qtlinuxfb

Move the cursor to the beginning of the line with "qttesttool". Press the "i" key to enter the input mode. Type "#" to comment out this line. Then move the cursor to the end of the line and press the Enter key to go to the next line. Next, enter the absolute path of the "dwinqtdemo" program followed by a space and "&". Then press the "Esc" key to exit the input mode. Type ": (colon) wq" to save the changes to the file.



Then continue to enter the following code. The qt program will run after the system restarts.

cd/etc/iniit.d/

reboot (Press the Enter key to make it effective)

Chapter 9 Revised Version

Rev	Revise Date	Content	Editor
00	2023-2-20	First Edition	Yu Yihe
01	2023-3-17	English version	Chen Lvzhi
02	2024-3-20	Added examples about brightness adjustment and system time settings	Chen Yan
03	2024-7-25	Add QT creator compile configurate, compile Linux 4.19 firmware, and chapter 1.	Chen Yan
04	2024-11-07	Add chapter 4.4 (40 series & 40ZOS-1 series)	Chen Xian
05	2025-02-08	Add Boot logo update method and other QT screen configuration options. Add examples of Debian screen configuration.	Chen Xian
06	2025-02-13	Update the location of "Run dwinqtdemo"	Chen Xian
07	2025-02-17	Delete the part of "obtaining the SDK by downloading via repo" , Sync code and Download SDK	Chen Xian
08	2025-03-10	Update “Modify the Startup LOGO”	Chen Xian
09	2025-04-16	Rewrite the logic and description of the development guide, add some accompanying figures, adjust the position of the environment setup chapter, update the applicable objects of the development package, and supplement the method of using commands to update the logo. Add content about downloading the QTdemo to the device and environment configuration variables in Chapter 8. Optimize the file storage location for downloading the QTdemo to the screen and the reference code in Chapter 8.	Chen Xian
10	2025-05-08	Add the description of the 43 series	Chen Xian

Please contact us if you have any questions about the use of this document or our products, or if you would like to know the latest information about our products:

Customer service Tel: +86-400-018-9008

Customer service email: dwinhmi@dwin.com.cn

Website: www.dwin-global.com

DWIN Developer Forum: https://forums.dwin-global.com/index.php/forums

Thank you all for continuous support of DWIN, and your approval is the driving force of our progress!