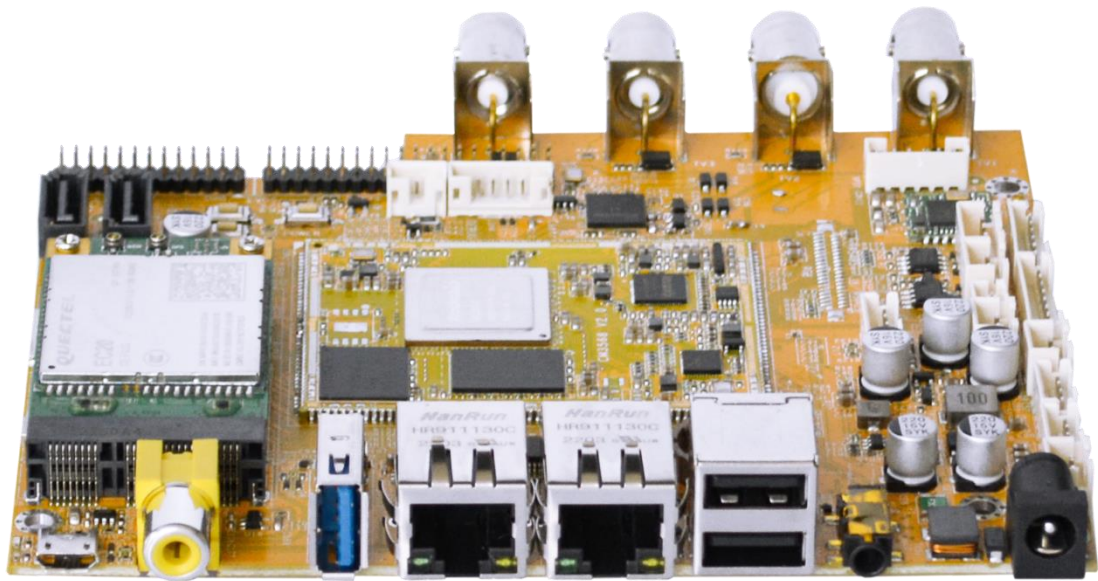


EM3568-AV CAM User Manual

V1.0



Boardcon Embedded Design

www.armdesigner.com

Preface

Overview

The content of this document is only described for the development board EM3568-AV CAM, aiming to help users quickly understand, apply and develop the EM3568-AV CAM development board.

System Support

Development Board	Android12	Debian11	Buildroot
EM3568-AV CAM_V1 CM3568_V2.0	N	Y	N

Revision History

Version	Date	Author	Revision History
V1.0	2023-06-19	Liu Yuan	Initial version

Version update instructions

Due to product version upgrades or other reasons, the content of this document will be updated from time to time. At the same time, it will be synchronized to the Boardcon website.

If you have any questions, concerns or comments about what is stated in the document, please feel free to contact us.

- **Company:** Boardcon Technology Co., Ltd.
- **Address:** 2508 Haofang Tianji Plaza, 11008 Beihuan Avenue, Nanshan District, Shenzhen, GuangDong, China. 518051
- **URL:** www.boardcon.com | www.armdesigner.com
- **Email:** market@armdesigner.com
- **Technical Support Inquiries:** support@armdesigner.com
- **Tel:** +86-755-26481393
+86-755-27571591
- **Fax:** +86-755-27571552

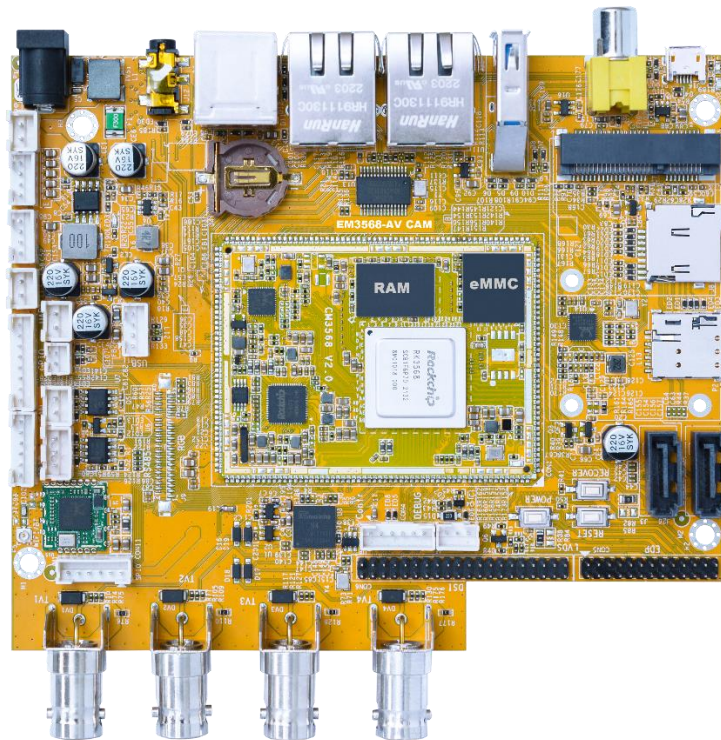
Content

Introduction.....	3
1.Overview	3
2.Product Parameters	3
3. Hardware Interface.....	5
Install Drivers and Tool	7
1.Install RK Driver Assitant.....	7
2.Install CP210X Driver	8
2.1 How to connect the serial port tool.....	8
2.2 Install driver.....	9
3.Install Serial Terminal Tool.....	9
Debian11	12
1.Compiler Environment.....	12
2.Install Tools.....	12
3.Compile Source.....	12
4.Images Operation	13
4.1 Unzip firmware	13
4.2 Pack image	14
5.Burn Images	16
5.1 Burn update firmware.....	16
5.2 Burn split firmware	18
6.Debian Test	19
6.1 Display	19
6.2 SD/USB device	19
6.3 SATA	21
6.4 Video player	21
6.5 Ethernet.....	22
6.6 RTC.....	23
6.7 4G(EC20).....	23
6.8 GPS(EC20)	24
6.9 WiFi.....	25
6.10 Bluetooth.....	26
6.11 CAN.....	28
6.12 RS485	28
6.13 UART	29
6.14 SPI	30
6.15 Camera	31

Introduction

1. Overview

EM3568-AV CAM is a development board designed based on Boardcon CM3568 system on module. The CM3568 SOM is powered by Rockchip RK3568 which is a quad-core ARM Cortex-A55 processor with low power consumption and high performance, support almost full-format H.264 decoder by 4K@60fps, H.265 decoder by 4K@60fps, also H.264/H.265 encoder by 1080p@60fps, high-quality JPEG encoder/decoder and designed for personal mobile Internet devices and AIoT devices.



2. Product Parameters

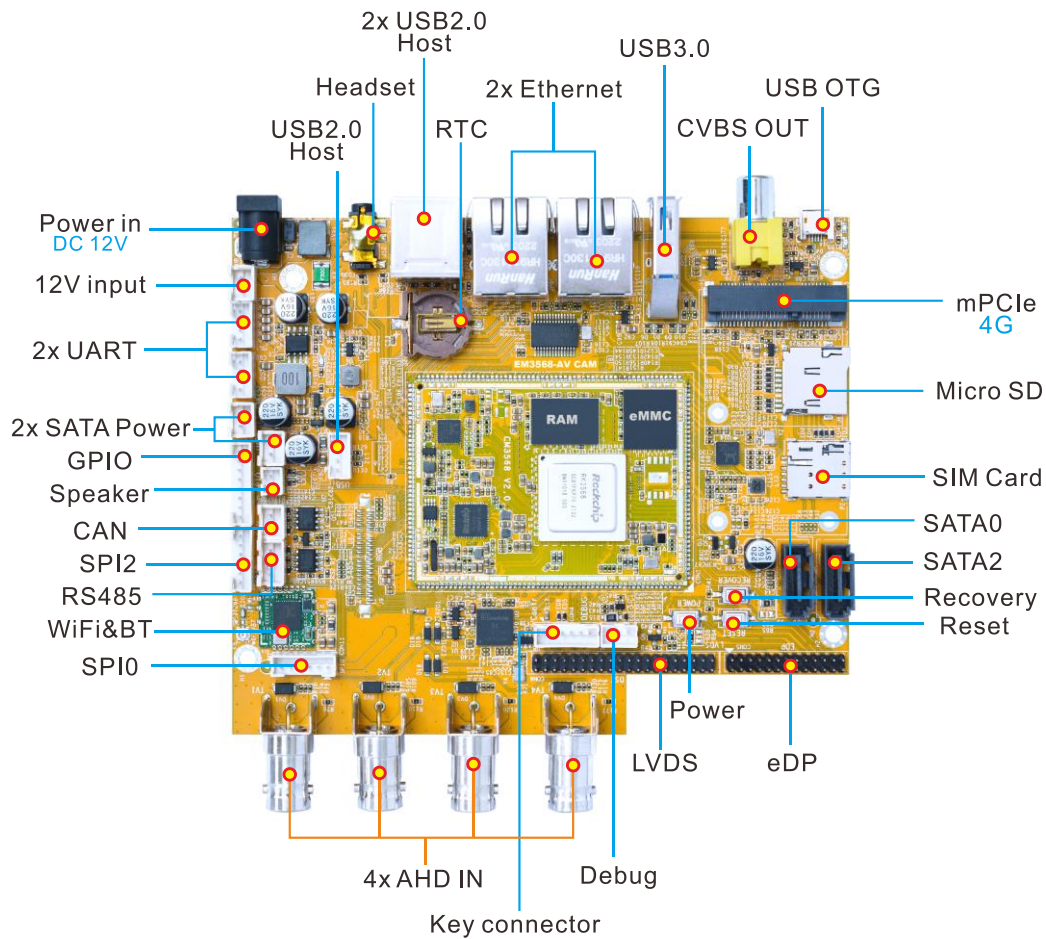
Basic Parameters	
SOC	RK3568
CPU	Quad-core 64-bit ARM Cortex-A55@ up to 2.0GHz
GPU	<ul style="list-style-type: none"> • ARM G52 2EE • Support OpenGL ES 1.1/2.0/3.2, OpenCL 2.0 and Vulkan 1.1 • Embedded high-performance 2D acceleration hardware
NPU	<ul style="list-style-type: none"> • Support 1T computing power



		<ul style="list-style-type: none"> • Support INT8/INT16/FP16/BFP16 MAC hybrid operation
Video	Decoder	<ul style="list-style-type: none"> • Support 4096x2304@60fps H.265/H.264/VP9 video decoding • Support 1920x1088@60fps VP8/VC1/MPEG-4,2,1 video decoding • Support 720x576@60fps H.263 video decoding
	Encoder	<ul style="list-style-type: none"> • Support 1920x1080@60fps H.265/H.264 video encoding • Support YUV/RGB video source with rotation and mirror
RAM		2GB LPDDR4X
ROM		8GB EMMC
Support system		Debian11
Hardware Parameters		
Extended Storage		<ul style="list-style-type: none"> • Support SATA • Support Mirco SD Card
Display		<ul style="list-style-type: none"> • Support LVDS 1280X800@60fps display • Support CVBS display
Audio		<ul style="list-style-type: none"> • Support Headphone output • Support Speaker output
USB		<ul style="list-style-type: none"> • Support 3x USB2.0 • Support USB3.0
Ethernet		Support 2x Gigabit Ethernet
Wireless Network		<ul style="list-style-type: none"> • Support WIFI/BT module • Support 4G module
Camera		Support 4x AHD IN
Peripheral communication		<ul style="list-style-type: none"> • Support CAN • Support RS485 • Support 3x UART • Support 2x SPI
Other parameters		Support Debug, USB OTG, RTC, GPIO
Electrical Parameters		

Power supply input voltage	12V/3A
RTC input voltage	3V
Operating temperature	0~70°
Storage temperature	-40~85°
Structural Parameters	
Core board dimensions	60mm x 45mm
Motherboard dimensions	135mm x 113mm

3. Hardware Interface



Interface parameters	
Power in	12V DC power input interface
USB2.0 Host	USB expansion interface



Headset	Earphone output
2xUSB2.0 Host	Dual-layer USB2.0 HOST interface
RTC	RTC coin cell connector
2x Gigabit Ethernet	Gigabit Ethernet RJ45 interface, via Realtek RTL8211F-CG controller
USB3.0 Host	USB3.0 Host interface
CVBS OUT	CVBS display interface
USB OTG	OTG download interface
mPCIe 4G	The EC20 4G module is used by default
Micro SD	MicroSD card slot
SIM Card	SIM card port
SATA0	SATA0 interface
SATA2	SATA2 interface
Recover	Recovery key
Reset	Reset key
Power	Power key
eDP	eDP screen display interface
LVDS	LVDS screen display interface
Debug	UART2, debug the serial port
Key connector	Including PWM3_IR/Reset/Recover/Power GPIO
4x AHD IN	AHD camera interface
SPI0	SPI0 interface
WIFI&BT	Realtek RTL8723DU WIFI/BT module
RS485	RS485 interface
SPI2	SPI2 interface
CAN	CAN interface
GPIO	GPIO/I2C5 extension interface
Speaker	Speaker interface

2x SATA Power	SATA power interface
2xUART	<ul style="list-style-type: none"> • UART5, TTL level interface • UART4, TTL level interface
12V input	12V DC power input gpio interface
Power in	12V DC power input interface

Install Drivers and Tool

To download firmware and debug in the terminal, the following drivers and software need to be installed(for Windows computers):

Number	Driver name	Driver	Use
1	RK Driver Assitant	DriverInstall.exe	OTG USB driver installation assistant
2	CP210x	PreInstaller.exe	Serial port debugging driver
3	Serial Terminal Tool	SecureCRT.exe	Debugging tool

1.Install RK Driver Assitant

Step 1, open DriverAssitant_v5.12/DriverInstall.exe.

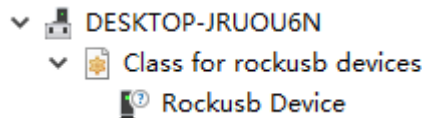
Step 2, to avoid driver conflicts, click **Uninstall Driver** to uninstall the driver.



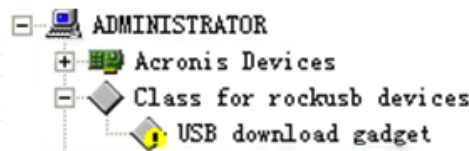
Step 3, click button **Install Driver** to install.



Step 4, after the installation is complete, connect the board and PC with Micro USB cable and press the **Recovery** key and hold then power the board, the following information is displayed in the Computer Device Manager, indicating that the USB driver was successfully installed.



Step 5, if the following device information appears on the computer device manager after the operation in step 4, you need to proceed to the next step.



Step 6, the WINDOW will pop up found New Hardware Wizard dialog box, choose to install from the specified location, and then select: \DriverAssitant_v5.12\ADBDriver.

Step 7, after the installation is completed, the following device information can be seen in the Computer Device Manager.



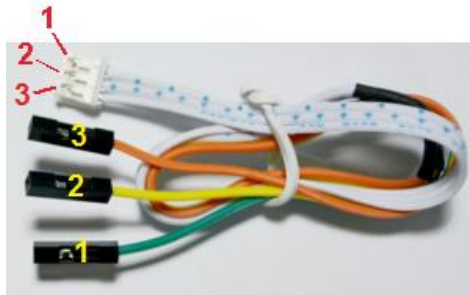
2.Install CP210X Driver

2.1 How to connect the serial port tool



Pin	Connection Description
3V3	No need to connect.
TXD	Transmit, connect to TX pin of the board.

RXD	Receive, connect to RX pin of the board.
GND	Ground, connect to GND pin of the board.
+5V	No need to connect.



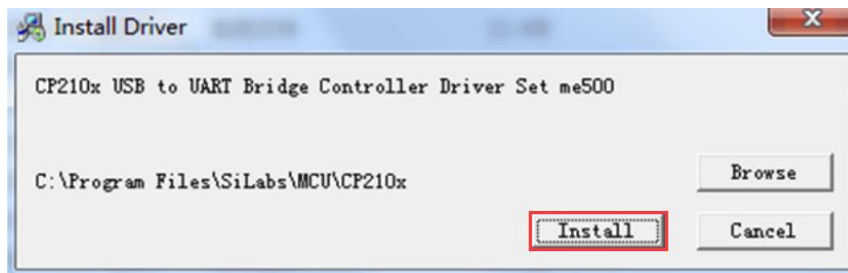
Pin	Connection Description
1	RX, connect to RXD pin of the CP210X Module.
2	TX, connect to TXD pin of the CP210X Module.
3	Ground, connect to GND pin of the CP210X Module.

2.2 Install driver

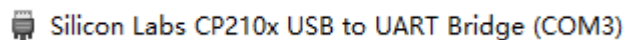
Step 1, plug the CP2102 Module to the PC.

Step 2, unzip CP210x_Windows_Drivers_with_Serial_Enumeration.zip on Windows.

Step 3, select and install the corresponding PreInstaller.exe according to the computer properties.



Step 4, after the installation is completed, the device will be listed under **Device Manager** -> **ports** with unique serial port assigned.

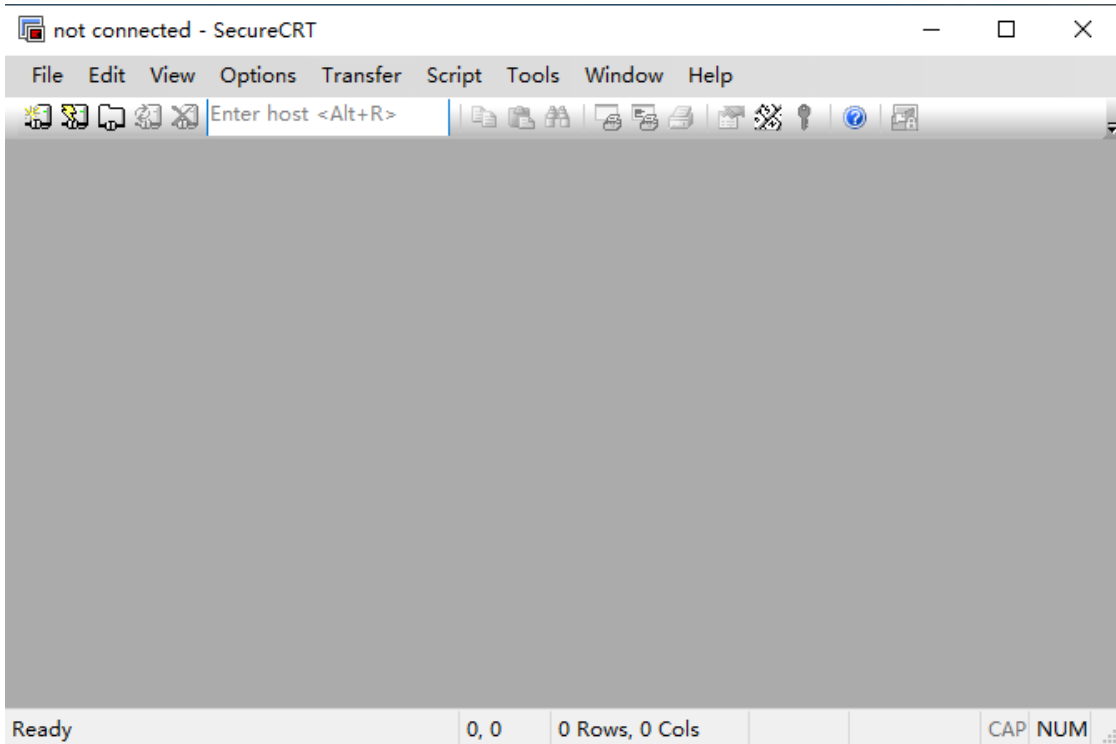


3. Install Serial Terminal Tool

The serial terminal SecureCRT is used for debugging in Windows. It can be used directly after decompression.

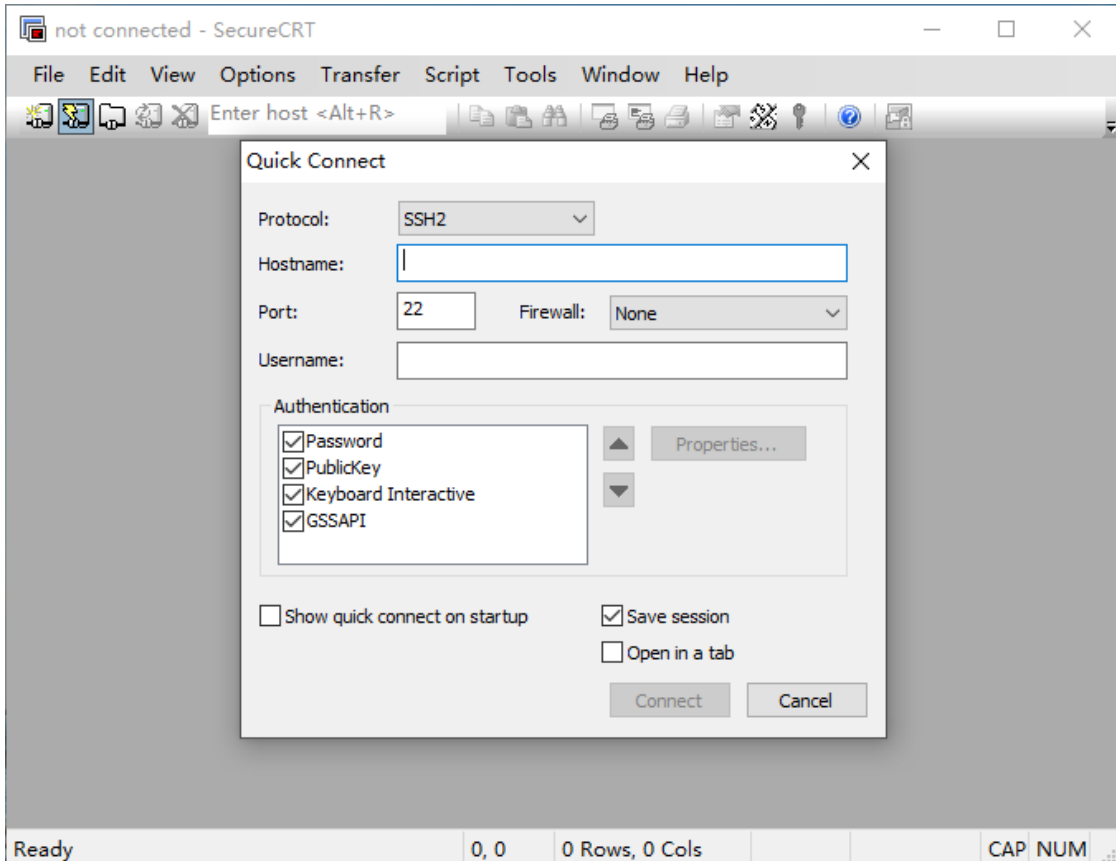
Step 1, unzip Platform/SecureCRT.rar on PC.

Step 2, click SecureCRT/SecureCRT.exe open the SecureCRT.

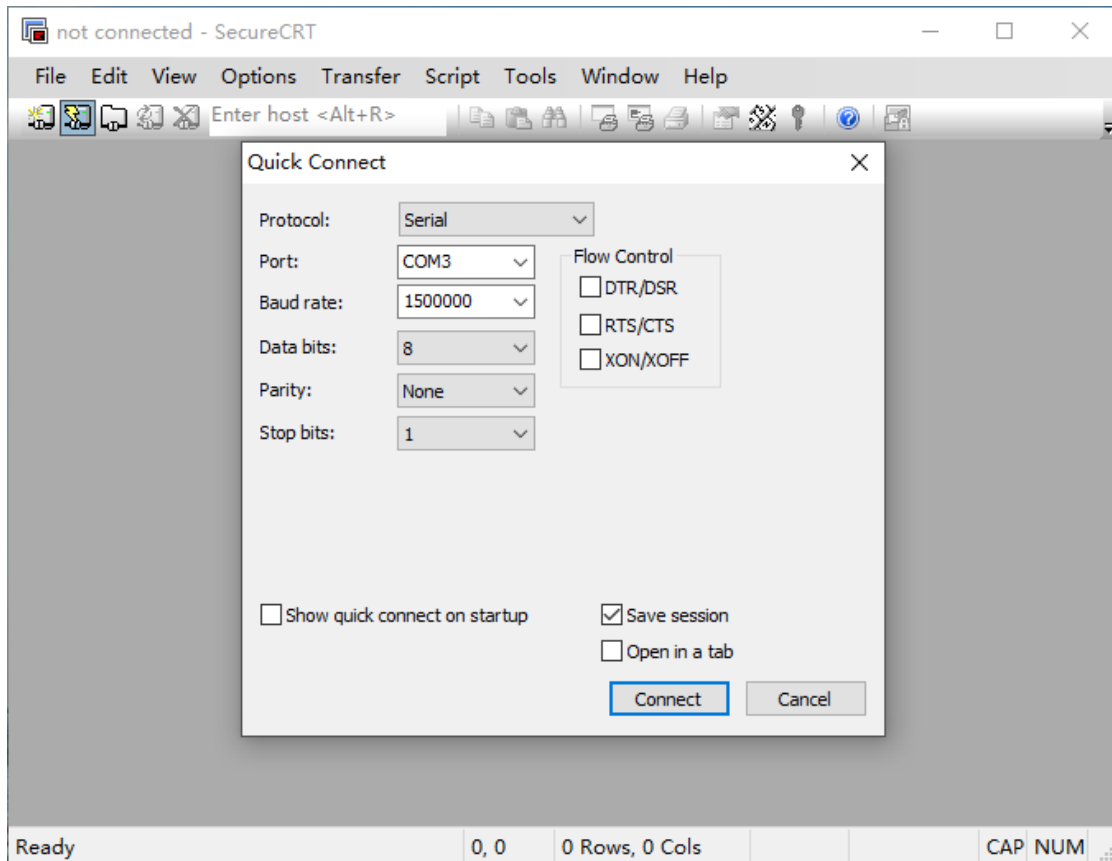


Step 3, confirm that the CP210x driver has been installed and the CP2102 module is connecting to the PC.

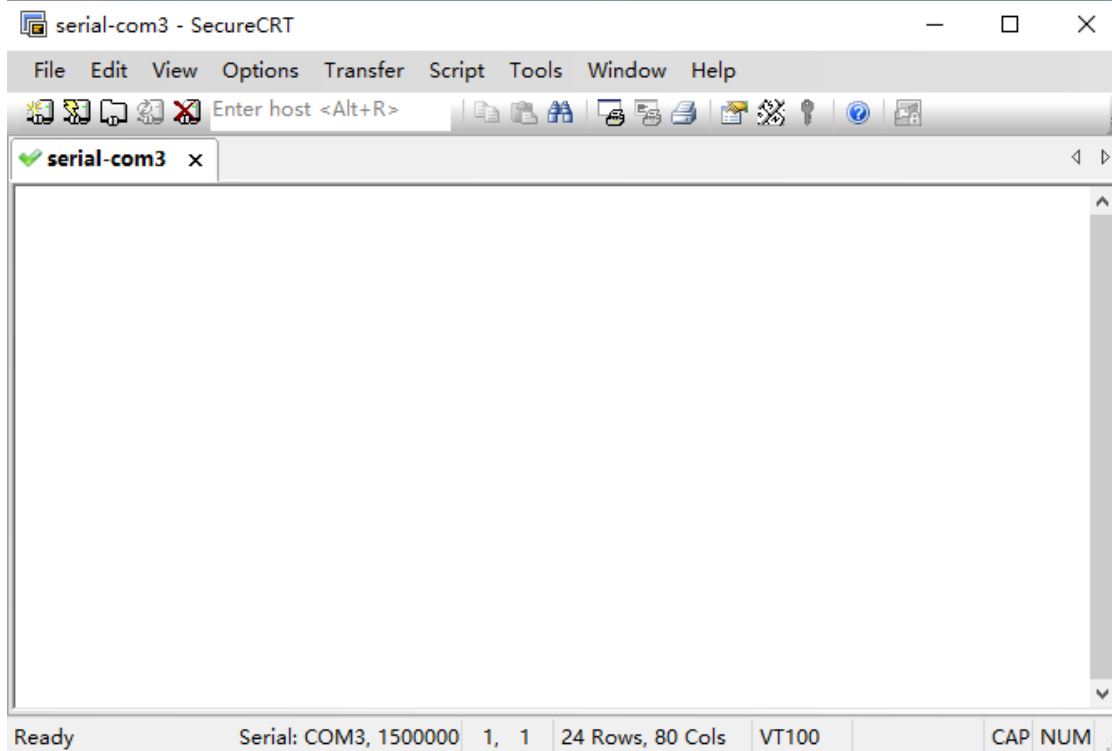
Step 4, click the **Quick Connect** button to go to the Quick Connect configuration screen.



Step 5, configure as shown in the following figure.



Step 6, after clicking connect button, the terminal serial interface will be successfully accessed.



Debian11

1. Compiler Environment

It is recommended to use Ubuntu 22.04 system or above. If you encounter an error during compilation, you can check the error message and install the corresponding software packages accordingly. Other Ubuntu versions may need to adjust the software package accordingly. In addition to the system requirements, there are other hardware and software requirements.

Hardware requirements	Software requirements
64-bit system, hard disk space should be greater than 40G. If you do multiple builds, you will need more hard drive space.	Ubuntu 22.04 system

2. Install Tools

The contents of this directory only provide the software package installation commands that are needed to build the compiled SDK environment. Please install other tools such as samba and ssh yourself.

PC OS: ubuntu 22.04 system

Network: online

Permission: root

```
# sudo apt-get install git ssh make gcc libssl-dev liblz4-tool
# sudo apt-get install expect g++ patchelf chrpath gawk texinfo chrpath diffstat
# sudo apt-get install binfmt-support qemuuser-static live-build bison flex fakeroot
# sudo apt-get install cmake gcc-multilib g++-multilib unzip device-tree-compiler
# sudo apt-get install ncurses-dev libgucharmap-2-90-dev bzip2 expat gpgv2
# sudo apt-get install cpp-aarch64-linux-gnu g++-aarch64-linux-gnu
# sudo apt install python2
# sudo apt install python-is-python3
```

3. Compile Source

Step 1, unzip the source

```
$ tar xvf linux-5.10.tar.bz2
$ cd linux-5.10
```

Step 2, configure the compiled board

```
$ ./build.sh lunch
```

name@boardcon:~/opt/EM3568-TU/linux-5.10\$./build.sh lunch

Log saved at /home/name/opt/EM3568-TU/linux-5.10/output/log/2023-06-19_09-40-22



Pick a defconfig:

1. rockchip_defconfig
2. rockchip_rk3566_evb2_lp4x_v10_32bit_defconfig
3. rockchip_rk3566_evb2_lp4x_v10_defconfig
4. rockchip_rk3568_evb1_ddr4_v10_32bit_defconfig
5. rockchip_rk3568_evb1_ddr4_v10_defconfig
6. rockchip_rk3568_evb2_lp4x_v10_defconfig
7. rockchip_rk3568_uvc_evb1_ddr4_v10_defconfig

Which would you like? [1]: 6 // Select rockchip_rk3568_evb2_lp4x_v10_defconfig

Switching to defconfig: /home/name/opt/EM3568-TU/linux-

5.10/device/rockchip/.chip/rockchip_rk3568_evb2_lp4x_v10_defconfig

Step 3, compile uboot

```
$ ./build.sh uboot
```

Step 4, compile the kernel

```
$ ./build.sh kernel
```

Step 5, compile recovery

```
$ ./build.sh recovery
```

Step 6, compile debian

```
$ ./build.sh rootfs
```

After compilation, generate linaro-rootfs.img in the Debian directory.

Note: If it is the first-time compiling Debian, it required install dependency packages. (Permission: root)

```
# cd debian
```

```
# sudo apt-get install binfmt-support qemu-user-static live-build
```

```
# sudo dpkg -i ubuntu-build-service/packages/*
```

```
# sudo apt-get install -f
```

Step 7, generate and check firmwares

```
$ ./build.sh firmware
```

Step 8, build update image

```
$ ./build.sh updateimg
```

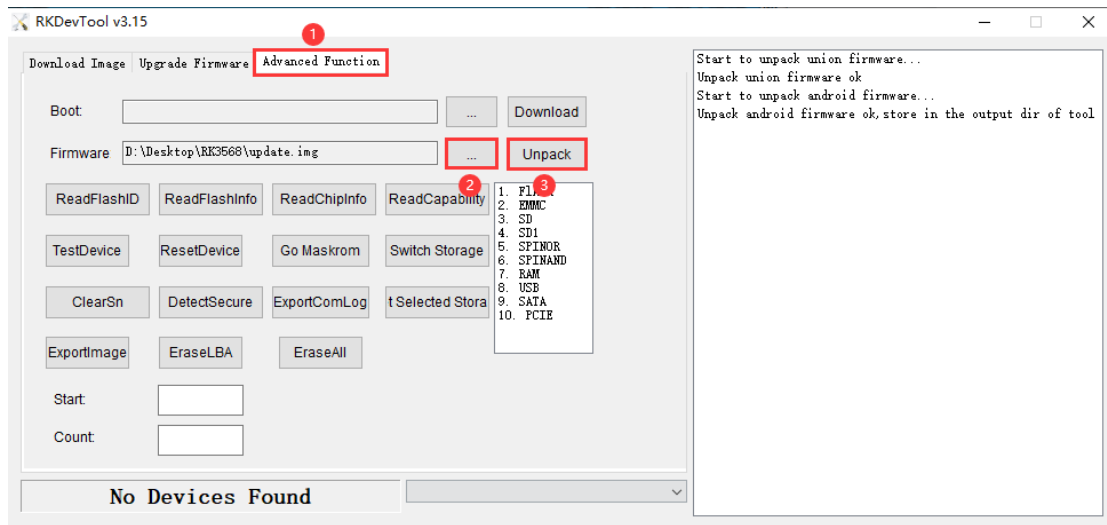
Images and update.img are generated in `rockdev/` directory.

4.Images Operation

4.1 Unzip firmware

Step 1, open `RKDevTool_Release\RKDevTool.exe`.

Step 2, click **Advanced Function** -> **Firmware**, select **update.img**, then click **Unpack** to unzip.



Step 3, the unzip files will be generated in `\\RKDevTool\\RKDevTool_Release\\Output\\Android` and `\\RKDevTool\\RKDevTool_Release\\Output\\Android\\Image` directory.

RKDevTool > RKDevTool_Release > Output > Android >

- 名称
- Image
 - MiniLoaderAll.bin
 - package-file
 - parameter.txt

RKDevTool > RKDevTool_Release > Output > Android > Image










- 名称
- boot.img
 - misc.img
 - oem.img
 - recovery.img
 - rootfs.img
 - uboot.img
 - userdata.img

4.2 Pack image

Step 1, copy the firmware file to be packaged to windows `RKDevTool/rockdev/Image`.

RKDevTool > rockdev > Image










名称

-  boot.img
-  MiniLoaderAll.bin
-  misc.img
-  oem.img
-  parameter.txt
-  recovery.img
-  rootfs.img
-  uboot.img
-  userdata.img

Step 2, enter RKDevTool/rockdev/, double-click to run **rk356x-mkupdate.bat**.

RKDevTool > rockdev

名称

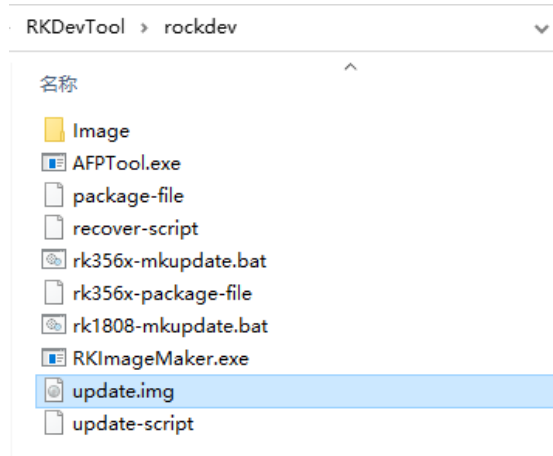
-  Image
-  AFPTool.exe
-  package-file
-  recover-script
-  **rk356x-mkupdate.bat**
-  rk356x-package-file
-  rk1808-mkupdate.bat
-  RKImageMaker.exe
-  update-script

```

Android Firmware Package Tool v1.65
D:\Desktop\RK3568\RKDevTool\rockdev>afptool -pack ./ Image/update.img
Android Firmware Package Tool v1.65
----- PACKAGE -----
Add file: .\package-file
Add file: .\package-file done, offset=0x800, size=0x118, userspace=0x1
Add file: .\Image\parameter.txt
Add file: .\Image\parameter.txt done, offset=0x1000, size=0x1f4, userspace=0x1
Add file: .\Image\MiniLoaderAll.bin
Add file: .\Image\MiniLoaderAll.bin done, offset=0x1800, size=0x6f9c0, userspace=0xe0
Add file: .\Image\uboot.img
Add file: .\Image\uboot.img done, offset=0x71800, size=0x400000, userspace=0x801
Add file: .\Image\misc.img
Add file: .\Image\misc.img done, offset=0x472000, size=0xc000, userspace=0x19
Add file: .\Image\boot.img
Add file: .\Image\boot.img done, offset=0x47e800, size=0x2273e00, userspace=0x44e8
Add file: .\Image\recovery.img
Add file: .\Image\recovery.img done, offset=0x26f2800, size=0x2e03800, userspace=0x5c08
Add file: .\Image\rootfs.img
Add file: .\Image\rootfs.img done, offset=0x54f6800, size=0xe6200000, userspace=0x1cc401
Add file: .\Image\oem.img
Add file: .\Image\oem.img done, offset=0xeb6f7000, size=0x10a6000, userspace=0x214d
Add file: .\Image\userdata.img
Add file: .\Image\userdata.img done, offset=0xec79d800, size=0x444000, userspace=0x839
Add CRC...
Make firmware OK!
----- OK -----
D:\Desktop\RK3568\RKDevTool\rockdev>RKImageMaker.exe -RK3568 Image\MiniLoaderAll.bin Image/update.img update.img -os_type:androidos
*****RKImageMaker ver 1.66 *****
Generating new image, please wait...
Writing head info...
Writing boot file...
Writing firmware...
Generating MD5 data...
MD5 data generated successfully!
New image generated successfully!

```

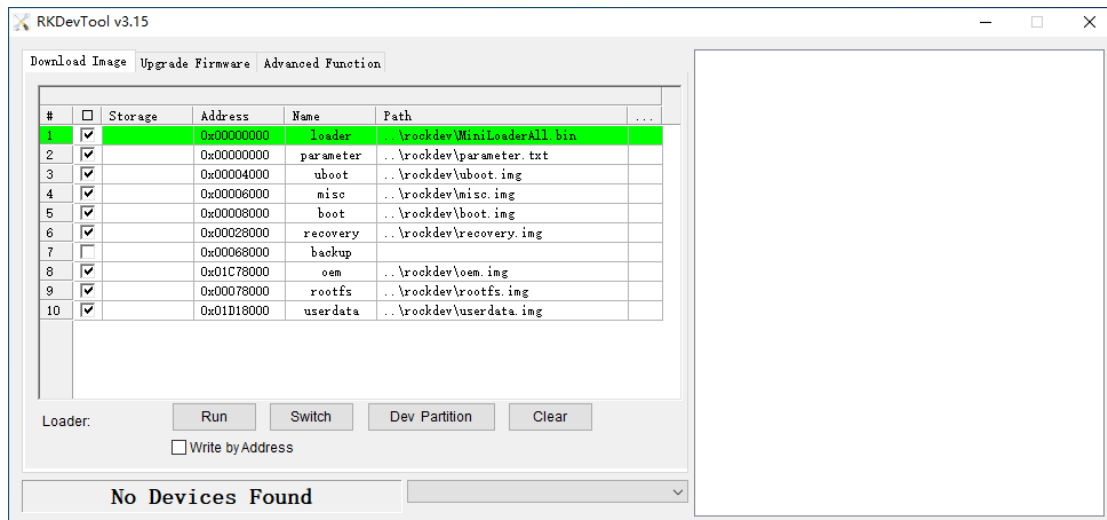
Step 3, the **update.img** will be generated in **rockdev/** directory.



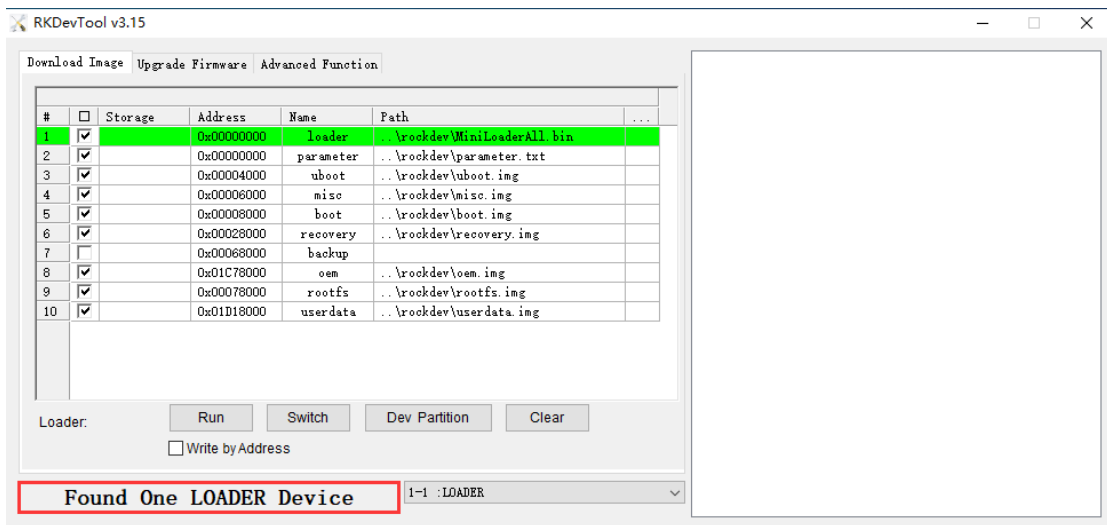
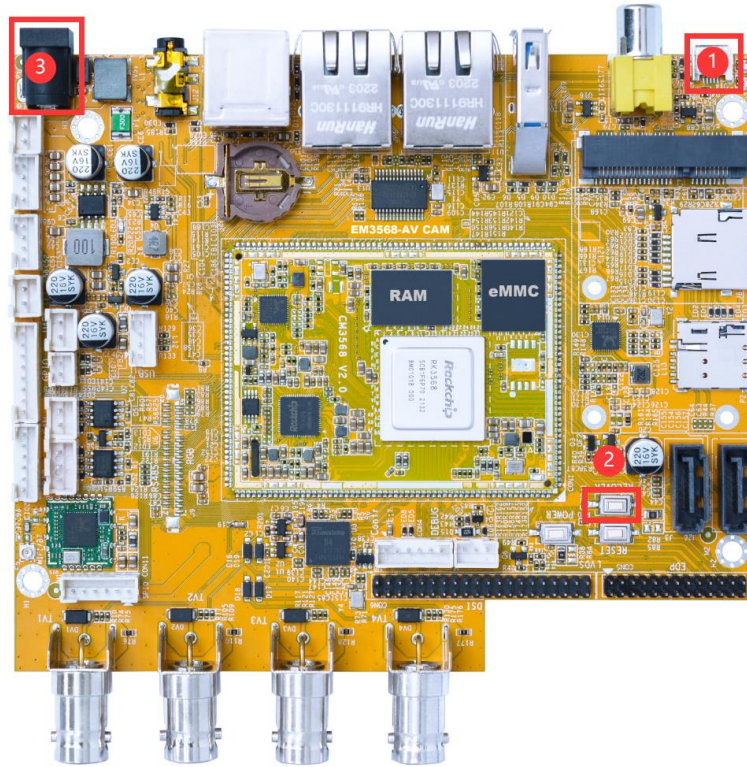
5. Burn Images

5.1 Burn update firmware

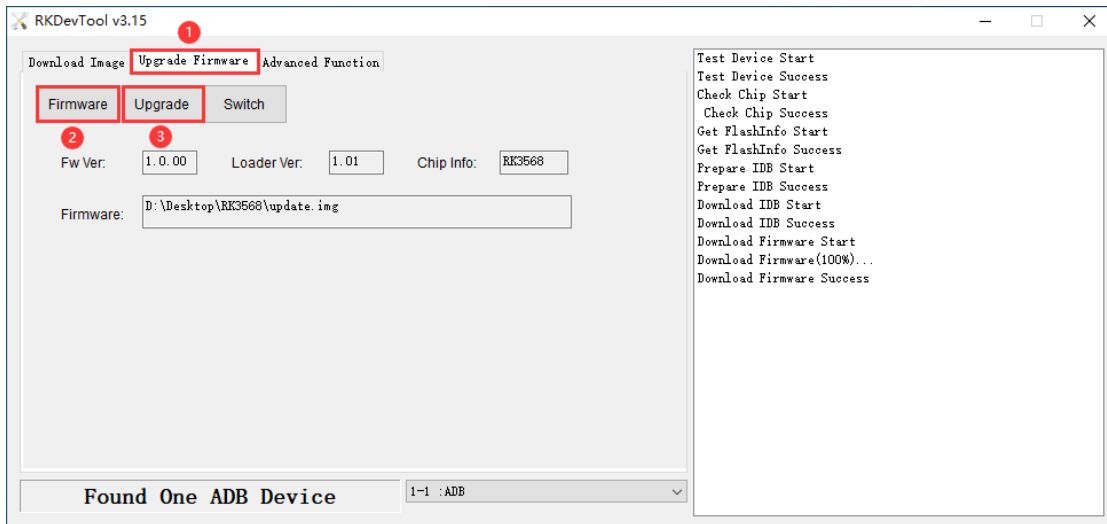
Step 1, open RKDevTool_Release\RKDevTool.exe.



Step 3, connect PC and development board with Micro USB cable, keep pressing the **Recovery Key** and power the board until the windows PC shows **Found one LOADER Device** release the **Recovery Key**.



Step 4, click **Upgrade Firmware** -> **Firmware**, select **update.img**, then click **Upgrade** to flash.

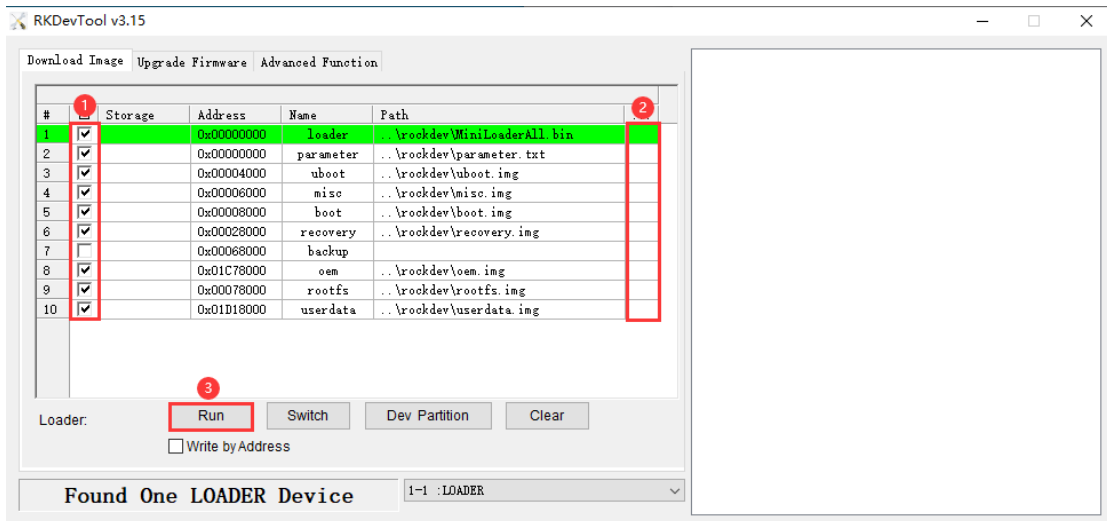


5.2 Burn split firmware

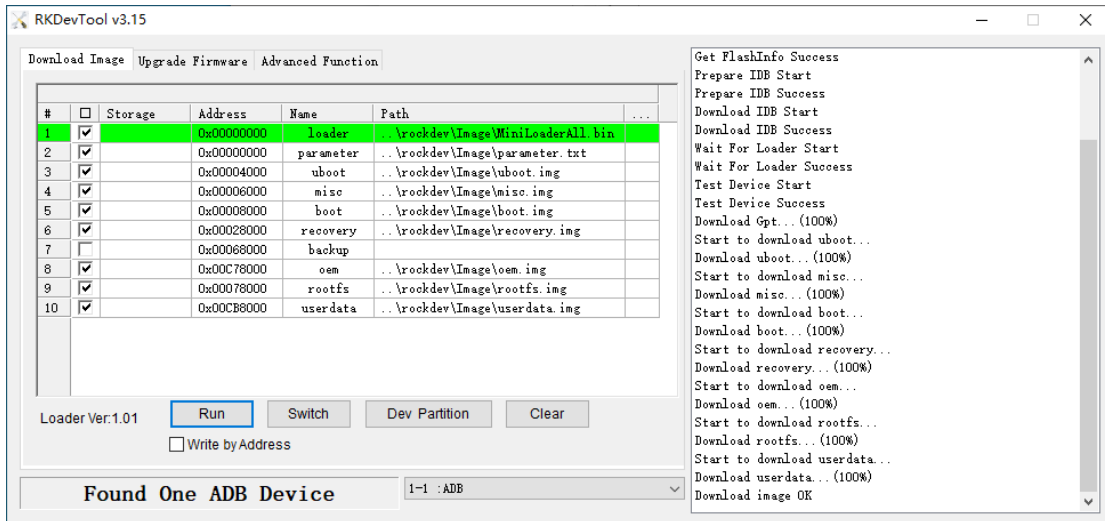
Step 1, select the checkbox on the left.

Step 2, click the column on the right side for the path of the file want to flash.

Step 3, click **run** button to flash the image.



Step 4, wait for the completion of burning.



6. Debian Test

6.1 Display

Supports LVDS and TV display. If you want to use the LVDS screen touch, you need to do the following to bind TP to the LVDS screen:

Step 1, installing xinput:

```
# apt-get update
# apt-get install xinput
```

Step 2, view display name:

```
# xrandr
```

```
root@linaro-alip:/# xrandr
Screen 0: minimum 320 x 200, current 2560 x 800, maximum 16384 x 16384
LVDS-1 connected primary 1280x800+0+0 (normal left inverted right x axis y axis) 0mm x 0mm
1280x800 59.19*+
```

Step 3, view the entered device ID:

```
# xinput
```

```
root@linaro-alip:/# xinput
Virtual core pointer          id=2    [master pointer (3)]
├─ Virtual core XTEST pointer id=4    [slave pointer (2)]
├─ Microsoft Microsoft Basic Optical Mouse id=7    [slave pointer (2)]
├─ goodix-ts                  id=11   [slave pointer (2)]
Virtual core keyboard        id=3    [master keyboard (2)]
├─ Virtual core XTEST keyboard id=5    [slave keyboard (3)]
├─ adc-keys                   id=6    [slave keyboard (3)]
├─ rk805 pwrkey               id=8    [slave keyboard (3)]
├─ fdd70030.pwm              id=9    [slave keyboard (3)]
├─ hdmi_cec_key              id=10   [slave keyboard (3)]
root@linaro-alip:/#
```

Step 4, bind TP:

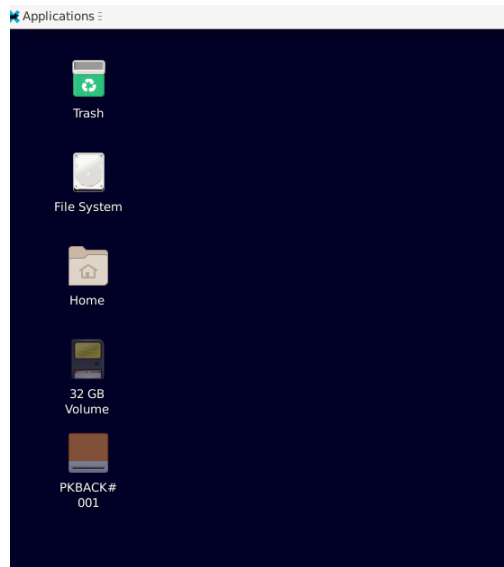
```
# xinput map-to-output [ID] [display name]
```

```
root@linaro-alip:/#
root@linaro-alip:/# xinput map-to-output 11 LVDS-1
root@linaro-alip:/#
```

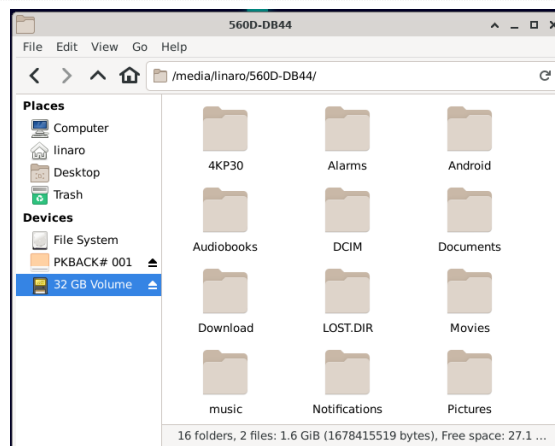
6.2 SD/USB device

After inserting an SD card or USB device, a corresponding icon will be formed on the desktop.

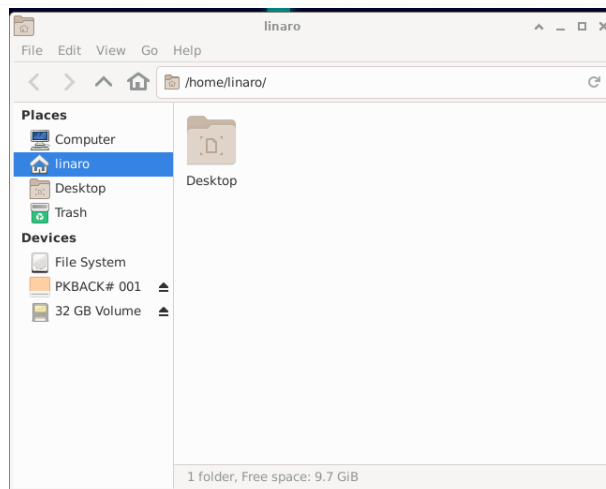
Automatic mounting of SD/USB storage devices can be achieved through the following two methods.



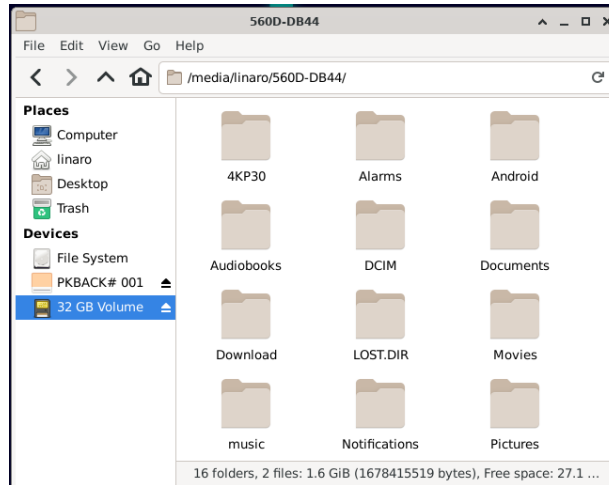
Step 1, double click on the desktop icon and the SD/USB storage device will automatically mount:



Step 2, double click on the **Home** icon to enter the main directory:



Then click on the **Devices** under the devices directory to achieve automatic mounting:



The USB Host can be used to connect USB mouse, USB keyboard, U-Disk or other USB devices.

6.3 SATA

Connect the SATA0 and the sata power to the board. Execute follow command to mount SATA.

```
# ls /dev // view sata device name
# mkdir /mnt/sata // create a new directory file
# mount /dev/sda1 /mnt/sata/ // mount sata to the new directory
# ls /mnt/sata/ // view the contents of the sata
```

```
root@linaro-alip:~# mkdir /mnt/sata
root@linaro-alip:~# mount /dev/sda1 /mnt/sata
root@linaro-alip:~# ls /mnt/sata
'$RECYCLE.BIN' Documents Notifications test
2.wav Download Pictures test.wav
Alarms Go-home.mp3 Podcasts test~
Android LOST.DIR Recordings
Audiobooks Movies Ringtones
DCIM Music big_buck_bunny_1080p_h264.mp4
root@linaro-alip:~#
root@linaro-alip:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        14G  3.6G  9.6G  27% /
devtmpfs        3.8G  8.0K  3.8G   1% /dev
tmpfs           3.9G   0  3.9G   0% /dev/shm
tmpfs           1.6G  1.8M  1.6G   1% /run
tmpfs           5.0M  4.0K  5.0M   1% /run/lock
tmpfs           3.9G  16K  3.9G   1% /tmp
/dev/mmcblk0p7  121M  12M  101M  11% /oem
/dev/mmcblk0p8   15G   36K   14G   1% /userdata
tmpfs           793M  28K  793M   1% /run/user/0
tmpfs           793M  40K  793M   1% /run/user/1000
/dev/sda1       112G  851M  111G   1% /mnt/sata
root@linaro-alip:~#
```

6.4 Video player

You can choose multiple commands to play videos:

Step 1, this method is quite troublesome and mainly used for playing videos in H256 format:

```
# cat /sys/kernel/debug/dri/0/state | grep "plane\[" // Select the display for playback
```



```
root@linaro-alip:~# cat /sys/kernel/debug/dri/0/state | grep "plane\[\"
plane[54]: Esmart0-win0
plane[76]: Esmart1-win0
plane[98]: Esmart2-win0
plane[120]: Esmart3-win0
plane[142]: Cluster0-win0
plane[156]: Cluster1-win0
plane[170]: Cluster2-win0
plane[184]: Cluster3-win0
root@linaro-alip:~#
```

export GST_MPP_VIDEODEC_DEFAULT_ARM_AFBC=1 // Be sure to open AFBC

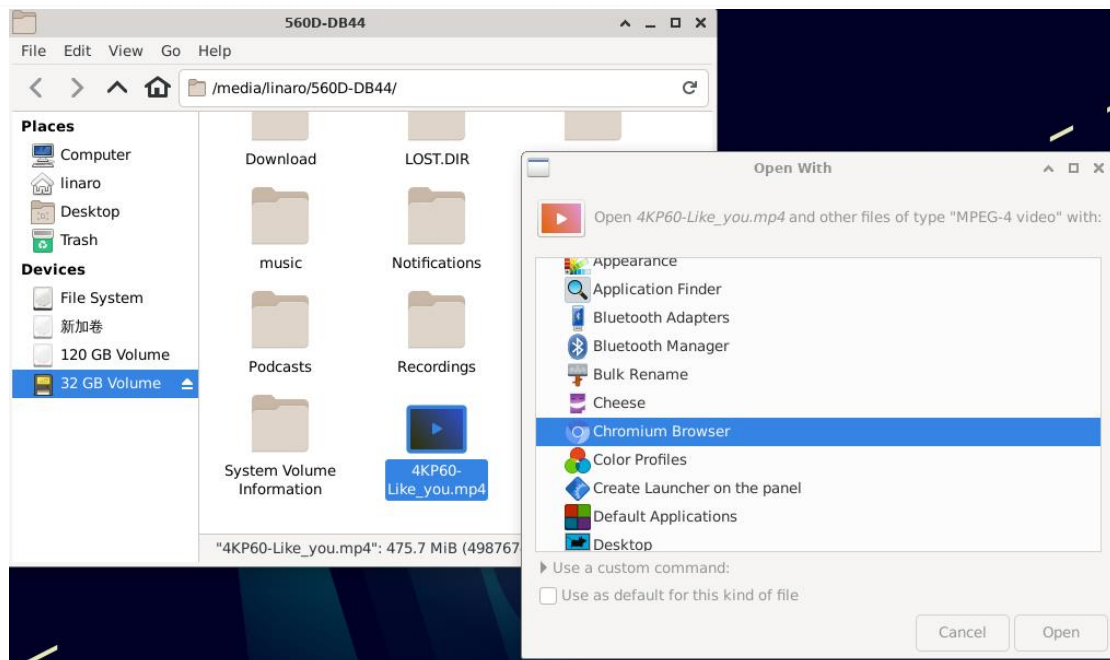
```
root@linaro-alip:~# export GST_MPP_VIDEODEC_DEFAULT_ARM_AFBC=1
```

GST_DEBUG=*mpp*:4 gst-play-1.0 --flags=3 --videosink="kmssink plane-id=142"
 /media/linaro/144460C44460AA66/video/H265_HEVC/I.O.I-Dream.Girl.2160p.UHDTV.H265.ts

Step 2, playing videos using Chromium (video files in H265 format are not supported):

chromium --no-sandbox file:///media/linaro/560D-DB44/4KP60-Like_you.mp4

Or select the video file you want to open, right-click and select Chromium Browser to open it



6.5 Ethernet

```
# ifconfig

root@linaro-alip:/#
root@linaro-alip:/#
root@linaro-alip:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.139 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::8c16:b07b:f836:fc3 prefixlen 64 scopeid 0x20<link>
    ether 36:31:c3:de:ed:19 txqueuelen 1000 (Ethernet)
    RX packets 212 bytes 13864 (13.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 13 bytes 1572 (1.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 51

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.138 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::c9b7:8ee6:dfb7:a851 prefixlen 64 scopeid 0x20<link>
    ether 32:31:c3:de:ed:19 txqueuelen 1000 (Ethernet)
    RX packets 59 bytes 4130 (4.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 11 bytes 1424 (1.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 56
```

```
# ping -I eth0 www.boardcon.com
```

```
# ping -I eth1 www.boardcon.com
```

```
root@linaro-alip:/#
root@linaro-alip:/# ping -I eth0 www.boardcon.com
PING www.boardcon.com (67.222.54.196) from 192.168.0.139 eth0: 56(84) bytes of data.
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=1 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=2 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=3 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=4 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=5 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=6 ttl=47 time=215 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=7 ttl=47 time=213 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=8 ttl=47 time=211 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=9 ttl=47 time=213 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=11 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=12 ttl=47 time=211 ms
^C
--- www.boardcon.com ping statistics ---
13 packets transmitted, 11 received, 15.3846% packet loss, time 16277ms
rtt min/avg/max/mdev = 211.033/212.315/214.843/1.009 ms
root@linaro-alip:/#
root@linaro-alip:/# ping -I eth1 www.boardcon.com
PING www.boardcon.com (67.222.54.196) from 192.168.0.138 eth1: 56(84) bytes of data.
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=1 ttl=47 time=214 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=2 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=3 ttl=47 time=213 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=5 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=6 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=7 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=8 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=9 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=10 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=11 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=12 ttl=47 time=212 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=13 ttl=47 time=212 ms
^C
--- www.boardcon.com ping statistics ---
13 packets transmitted, 12 received, 7.69231% packet loss, time 12027ms
rtt min/avg/max/mdev = 211.639/212.371/214.058/0.561 ms
root@linaro-alip:/#
```

6.6 RTC

Execute the follow command to set the RTC time:

```
# date -s "2023-06-20 10:22:00"
```

```
# hwclock -w
```

```
# hwclock
```

```
root@linaro-alip:/# date -s "2023-06-20 10:22:00"
date: invalid date '2023-06-20 10:22:00'
root@linaro-alip:/# hwclock -w
root@linaro-alip:/# hwclock
2023-06-20 02:22:08.874674+00:00
root@linaro-alip:/# hwclock
2023-06-20 02:22:32.843655+00:00
root@linaro-alip:/# hwclock
2023-06-20 02:23:00.434740+00:00
root@linaro-alip:/#
```

If it takes time to be saved after power failure, please connect the button battery.

6.7 4G(EC20)

Step 1, execute the following command to realize ppp dialing:

```
# pppd call quectel-ppp &
```



```
-- got it
send (ATD*99#^M)
expect (CONNECT)
^M
^M
CONNECT
-- got it

Script chat -s -v -f /etc/ppp/peers/quectel-chat-connect finished (pid 2069), status = 0x0
Serial connection established.
using channel 1
Using interface ppp0
Connect: ppp0 <--> /dev/ttyUSB3
sent [LCP ConfReq id=0x1 <asyncmap 0x0> <magic 0x90083d31> <pcomp> <accomp>]
rcvd [LCP ConfReq id=0x0 <asyncmap 0x0> <auth chap MD5> <magic 0xd0ffd81b> <pcomp> <accomp>]
sent [LCP ConfAck id=0x0 <asyncmap 0x0> <auth chap MD5> <magic 0xd0ffd81b> <pcomp> <accomp>]
rcvd [LCP ConfAck id=0x1 <asyncmap 0x0> <magic 0x90083d31> <pcomp> <accomp>]
sent [LCP EchoReq id=0x0 magic=0x90083d31]
rcvd [LCP DiscReq id=0x1 magic=0xd0ffd81b]
rcvd [CHAP Challenge id=0x1 <43d9b9cbde2f79d1322750e45f3af0eb>, name = "UMTS_CHAP_SRVR"]
sent [CHAP Response id=0x1 <e901d6812caedab20322f36a7ecd8f888>, name = "test"]
rcvd [LCP EchoRep id=0x0 magic=0xd0ffd81b 90 08 3d 31]
rcvd [CHAP Success id=0x1 ""]
CHAP authentication succeeded
CHAP authentication succeeded
sent [IPCP ConfReq id=0x1 <addr 0.0.0.0> <ms-dns1 0.0.0.0> <ms-dns2 0.0.0.0>]
sent [IPV6CP ConfReq id=0x1 <addr fe80::0480:8be6:0cde:5c36>]
rcvd [IPCP ConfReq id=0x0]
sent [IPCP ConfNak id=0x0 <addr 0.0.0.0>]
rcvd [IPCP ConfNak id=0x1 <addr 10.65.22.200> <ms-dns1 120.80.80.80> <ms-dns2 221.5.88.88>]
sent [IPCP ConfReq id=0x2 <addr 10.65.22.200> <ms-dns1 120.80.80.80> <ms-dns2 221.5.88.88>]
rcvd [IPCP ConfReq id=0x1]
sent [IPCP ConfAck id=0x1]
rcvd [IPCP ConfAck id=0x2 <addr 10.65.22.200> <ms-dns1 120.80.80.80> <ms-dns2 221.5.88.88>]
Could not determine remote IP address: defaulting to 10.64.64.64
Script /etc/ppp/ip-pre-up started (pid 2078)
Script /etc/ppp/ip-pre-up finished (pid 2078), status = 0x0
not replacing default route to eth0 [192.168.0.2]
local IP address 10.65.22.200
remote IP address 10.64.64.64
primary DNS address 120.80.80.80
secondary DNS address 221.5.88.88
Script /etc/ppp/ip-up started (pid 2081)
Script /etc/ppp/ip-up finished (pid 2081), status = 0x0
sent [IPV6CP ConfReq id=0x1 <addr fe80::0480:8be6:0cde:5c36>]
sent [IPV6CP ConfReq id=0x1 <addr fe80::0480:8be6:0cde:5c36>]
sent [IPV6CP ConfReq id=0x1 <addr fe80::0480:8be6:0cde:5c36>]
sent [IPV6CP ConfReq id=0x1 <addr fe80::0480:8be6:0cde:5c36>]
```

Step 2, execute the following to view the network interface status:

```
# ifconfig -a
```

```
ppp0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
    inet 10.65.22.200 netmask 255.255.255.255 destination 10.64.64.64
    ppp txqueuelen 3 (Point-to-Point Protocol)
    RX packets 4 bytes 52 (52.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 10 bytes 142 (142.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Step 3, execute the following to check the connectivity of the detection network:

```
# ping -I ppp0 www.boardcon.com
```

```
root@linaro-alip:~#
root@linaro-alip:~# ping -I ppp0 www.boardcon.com
PING www.boardcon.com (67.222.54.196) from 10.65.22.200 ppp0: 56(84) bytes of data.
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=1 ttl=47 time=289 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=2 ttl=47 time=202 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=3 ttl=47 time=207 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=4 ttl=47 time=198 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=5 ttl=47 time=197 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=6 ttl=47 time=210 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=7 ttl=47 time=197 ms
```

6.8 GPS(EC20)

Plug the EC20 module and connect GPS antenna, then power on and execute the follow command:

```
# echo -e "AT+QGPS=1\r\n" > /dev/ttyUSB2
```

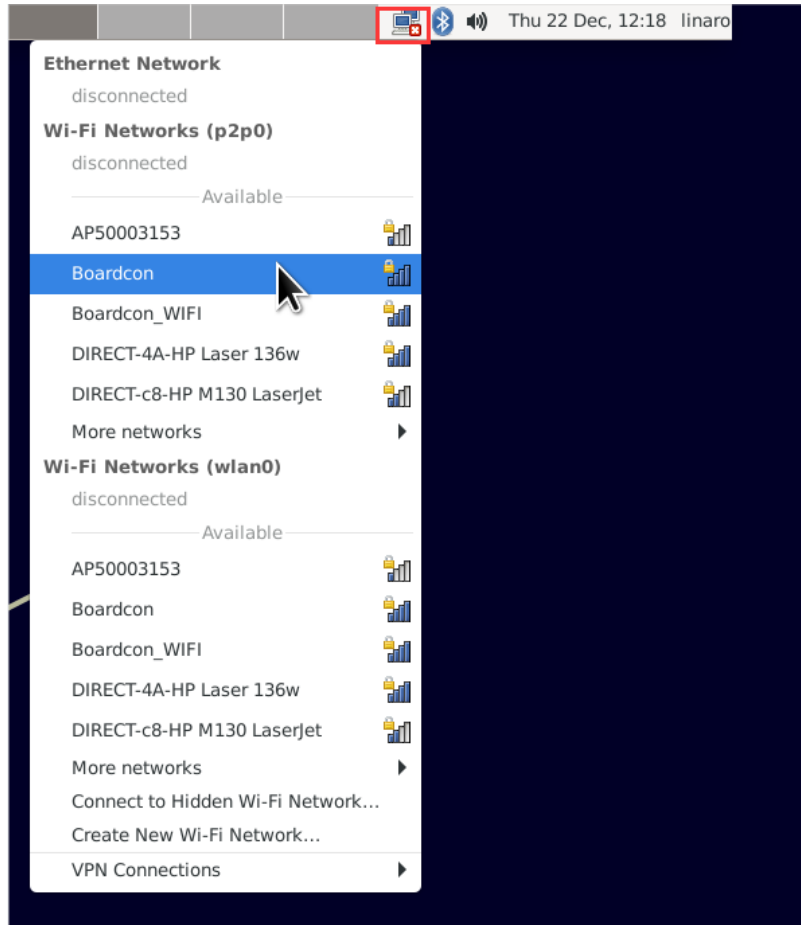
```
# cat /dev/ttyUSB1
```



```
root@linaro-alip:~# echo -e "AT+QGPS=1\r\n" > /dev/ttyUSB2
root@linaro-alip:~# cat /dev/ttyUSB1
$GPVTG,,T,,M,,N,,K,N*2C
$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPGGA,,,,,0,,,,,,,,,*66
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,T,,M,,N,,K,N*2C
$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPGGA,,,,,0,,,,,,,,,*66
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,T,,M,,N,,K,N*2C
$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPGGA,,,,,0,,,,,,,,,*66
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,T,,M,,N,,K,N*2C
$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPGGA,,,,,0,,,,,,,,,*66
$GPRMC,,V,,,,,,,,,N*53
$GPGSV,1,1,03,33,,34,38,,34,39,,34*7C
$GPVTG,,T,,M,,N,,K,N*2C
$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPGGA,,,,,0,,,,,,,,,*66
$GPRMC,,V,,,,,,,,,N*53
$GPGSV,1,1,04,33,,34,38,,34,39,,34,40,,34*78
$GPVTG,,T,,M,,N,,K,N*2C
$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPGGA,,,,,0,,,,,,,,,*66
$GPRMC,,V,,,,,,,,,N*53
```

6.9 WiFi

Step 1, connect the WiFi antenna, then click the network icon in the top right corner of the UI interface, select the SSID from the list of available networks and enter the password.



Step 2, execute the following to view the network interface status:

```
# ifconfig
```

```
[ 116.831201] IPv6: ADDRCONF(NETDEV_CHANGE): wlx367de41c8192: link becomes ready
```

```
wlx367de41c8192: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.0.146 netmask 255.255.255.0 broadcast 192.168.0.255
inet6 fe80::da4b:cda8:d7b:b1a9 prefixlen 64 scopeid 0x20<link>
ether 36:7d:e4:1c:81:92 txqueuelen 1000 (Ethernet)
RX packets 45 bytes 6079 (5.9 KiB)
RX errors 0 dropped 4 overruns 0 frame 0
TX packets 18 bytes 2811 (2.7 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

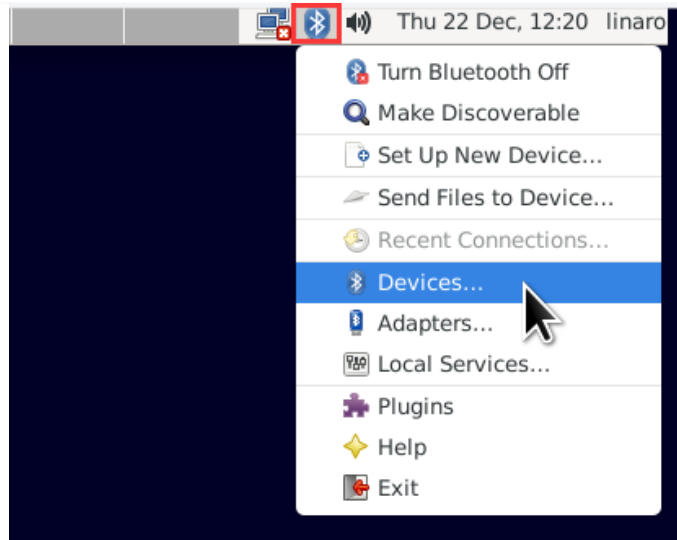
Step 3, execute the following to check the connectivity of the detection network:

```
# ping -I wlx367de41c8192 www.boardcon.com
```

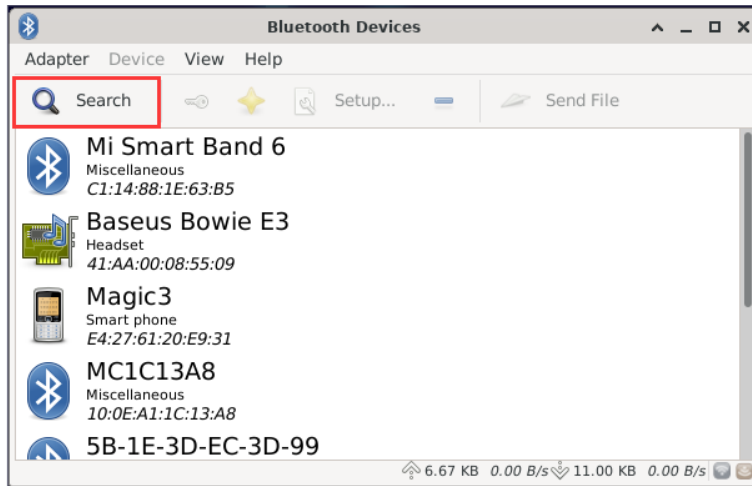
```
root@linaro-alip:/# ping -I wlx367de41c8192 www.boardcon.com
PING www.boardcon.com (67.222.54.196) from 192.168.0.146 wlx367de41c8192: 56(84) bytes of data.
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=1 ttl=47 time=205 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=2 ttl=47 time=204 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=3 ttl=47 time=209 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=4 ttl=47 time=204 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=5 ttl=47 time=203 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=6 ttl=47 time=203 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=7 ttl=47 time=207 ms
```

6.10 Bluetooth

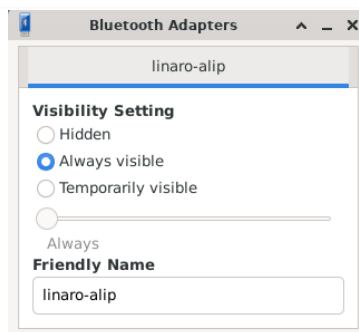
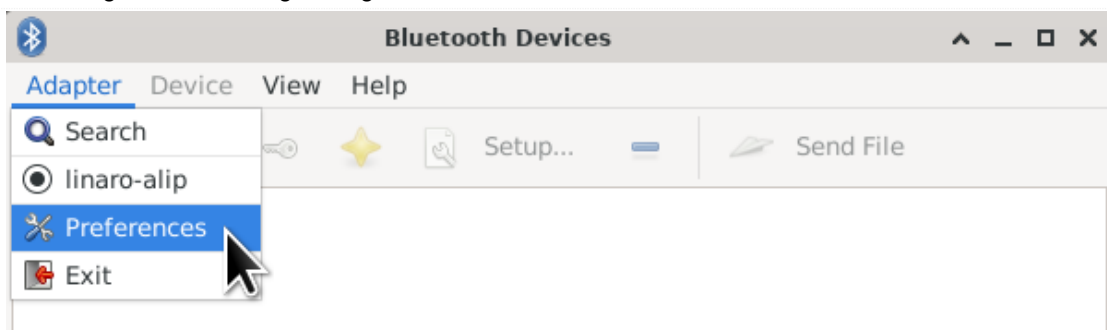
Step 1, click the Bluetooth icon in the top right corner of the UI interface, Select **"Devices"** enter the Bluetooth Devices.



Step 3, click the “Search” button to start searching and select the available device in the list to pair.

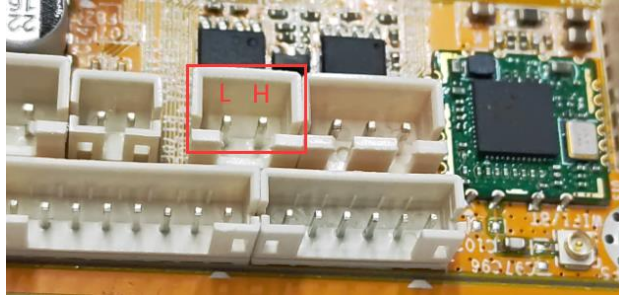
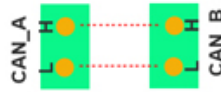


Step 4, the Bluetooth device name is hidden by default. Set the Bluetooth device name to be visible according to the following settings:



6.11 CAN

Step 1, connect the CAN ports of board A and board B as follows:



Step 2, execute the following commands on the serial terminal of board A and board B respectively.

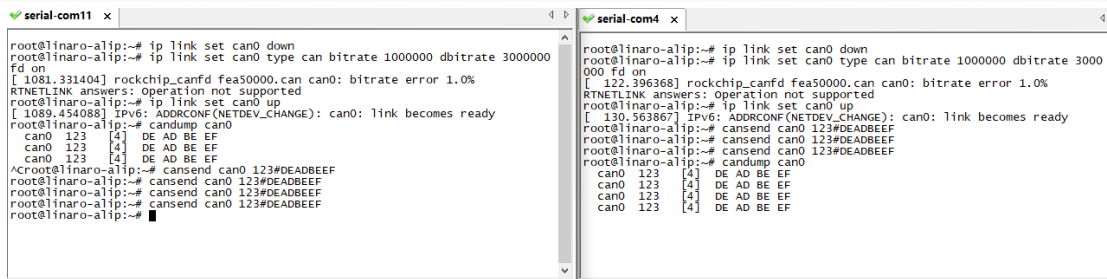
```
# ip link set can0 down
# ip link set can0 type can bitrate 1000000 dbitrate 3000000 fd on
# ip link set can0 up
```

Step 3, execute the following instructions as the receiver.

```
# candump can0 // set CAN0 as receive
```

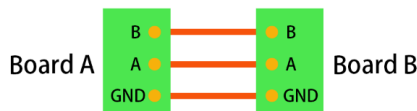
Step 4, execute the following commands as the transmitter.

```
# cansend can0 123#DEADBEEF // CAN0 send characters 0xDE 0xAD 0xBE 0xEF
```



6.12 RS485

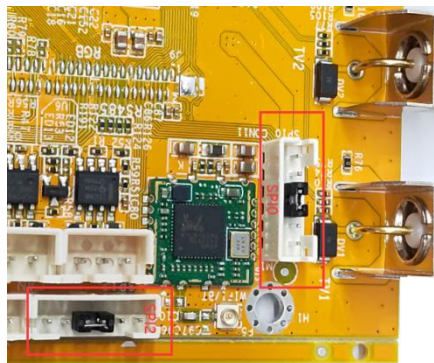
Step 1, connect the RS485 ports of board A and board B as follows:




```
root@linaro-alip:~# com /dev/ttyS5 115200 8 0 1
port = /dev/ttyS5
baudrate = 115200
cs = 8
parity = 0
stopb = 1
[ 690.566204] of_dma_request_slave_channel: dma-names property of node '/serial@fe690000' missing or empty
[ 690.566313] dw-apb-uart fe690000.serial: failed to request DMA, use interrupt mode
KKKKK
RECV: KKKKK
QW789
RECV: QW789
369ASD
RECV: 369ASD
```

6.14 SPI

Step 1, short circuit MISO_M1 and MOSI_M1 pins of SPI.



Step 2, SPI0 test:

```
root@linaro-alip:~#
root@linaro-alip:~# spidev0.0_test
spi mode: 0
bits per word: 8
max speed: 500000 Hz (500 KHz)

FF FF FF FF FF FF
40 00 00 00 00 95
FF FF FF FF FF FF
FF FF FF FF FF FF
FF FF FF FF FF FF
DE AD BE EF BA AD
F0 0D
root@linaro-alip:~#
```

Step 3, SPI2 test:



```
root@linaro-alip:/#
root@linaro-alip:/# spidev2.0_test
spi mode: 0
bits per word: 8
max speed: 500000 Hz (500 KHz)

FF FF FF FF FF FF
40 00 00 00 00 95
FF FF FF FF FF FF
FF FF FF FF FF FF
FF FF FF FF FF FF
DE AD BE EF BA AD
F0 0D
root@linaro-alip:/#
```

6.15 Camera

Step 1, camera preview command:

TV1

```
# gst-launch-1.0 v4l2src device=/dev/video8 ! video/x-raw,format=NV12,width=1280,height=720,
framerate=30/1 ! xvimagesink &
```

TV2

```
# gst-launch-1.0 v4l2src device=/dev/video9 ! video/x-raw,format=NV12,width=1280,height=720,
framerate=30/1 ! xvimagesink &
```

TV3

```
# gst-launch-1.0 v4l2src device=/dev/video10 ! video/x-raw,format=NV12,width=1280,height=720,
framerate=30/1 ! xvimagesink &
```

TV4

```
# gst-launch-1.0 v4l2src device=/dev/video11 ! video/x-raw,format=NV12,width=1280,height=720,
framerate=30/1 ! xvimagesink &
```

Step2, camera video recording command:

```
# gst-launch-1.0 v4l2src device=/dev/video8 num-buffers=100 ! \
video/x-raw,format=NV12,width=1280,height=720,framerate=30/1 ! \
videoconvert ! mpph264enc ! h264parse ! mp4mux ! \
filesink location=/tmp/h264a.mp4
```

Step3, camera take a picture command:

```
# gst-launch-1.0 -v v4l2src device=/dev/video8 num-buffers=10 ! \
video/x-raw,format=NV12,width=1280,height=720 ! mppjpegenc ! \
multifilesink location=/tmp/test1%05d.jpg
```