# EM305X User Manual

V1.0

Boardcon Embedded Design

## Overview

The content of this document is only described for the development board Idea3588 , aiming to help users quickly understand, apply and develop the Idea3588 development board.

## System Support

| Development Board | Android12 | Debian11 | Buildroot |
| --- | --- | --- | --- |
| EM305X_V1<br>Mini305X-V1 | N | **N** | **Y** |

## Revision History

| Version | Date | Author | Revision History |
| --- | --- | --- | --- |
| V1.0 | 2024-05-13 | Boardcon Team | 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 embedded design limited

• **Address:** 2508 Haofang Tianji Plaza, 11008 Beihuan Avenue, Nanshan District, Shenzhen, GuangDong, China. 518051

• **URL:** https://www.armdesigner.com/ https://www.boardcon.com/

• **Email:** market@armdesigner.com

• **Technical Support Inquiries:** support@armdesigner.com

• **Tel:** +86-755-26481393

+86-755-27571591

# *Content*

# Introduction

## 1. Overview

The EM305X integrates a dual-core ARM Cortex-A35 processor, which handles the OS, Wi-Fi/BT protocol, and user applications. The built-in NPU features a multi-threaded programmable engine and an ISP. The NPU supports INT8/INT16 inference for all popular deep learning frameworks, including TensorFlow and Caffe. The processor offers a rich set of advanced network and peripheral interfaces, including a 10/100/1000M Ethernet MAC with RGMII, a USB OTG port, multiple SDIO/SD card controllers, as well as UART, I2C, high-speed SPI, and PWM interfaces, making it suitable for smart IP cameras.

## 2. Product Parameters

| **Basic Parameters** | | |
|---|---|---|
| SOC | | C305X |
| CPU | | Dual ARM® Cortex™ A35 CPU |
| NPU | | • 2 NN cores with INT8 inference performance up to 2 Tops<br>• 1 programmable processing unit (PPU) up to 10.56 GFLOPS FP16 or 5.28 GFLOPS FP32 |
| Video | Encoder | Support high-efficient H.265/H.264 video encoding up to 2688x1944@30fps + 1080p@30fps |
| RAM | | 2GB DDR4 |
| ROM | | 8GB eMMC |
| Support system | | Buildroot |
| **Hardware Parameters** | | |
| Extended Storage | | • Support 1x MicroSD Card |
| Display | | • Support 1x RGB TFT_LCD 1.8’ 128x160 output |
| Audio | | • Support 1x Speaker output |
| USB | | • Support 2xUSB2.0<br>• Support 1x USB2.0 Micro USB(shared with USB2.0) |

| | |
|---|---|
| Network | • Support 1x Gigabit Ethernet<br>• Support 1x WIFI/BT module<br>• Support 1x 4G module |
| Camera | Support 1xCamera |
| Peripheral communication | • Support 1xRS485 |
| Other parameters | Support 1xDebug, 1xIR, 1xRTC, 1xPIR |
| **Electrical Parameters** | |
| Power supply input voltage | • 12V/3A<br>• **POE** |
| RTC input voltage | 3V/0.6uA |
| Operating temperature | 0~70° |
| Storage temperature | -40~85° |
| **Structural Parameters** | |
| Core board dimensions | mm x mm |
| Motherboard dimensions | mm x mm |

# 3. Hardware Interface Introduction

| **Interface parameters** | |
|---|---|
| Power in | 12V DC power input interface |
| Reset | Reset key |
| Boot | Boot key |
| USB Micro | USB Micro interface, OTG download interface |
| Ethernet | Gigabit Ethernet RJ45 interface or POE power input interface |
| USB2.0 Host | Dual-layer USB2.0 HOST interface |
| User Key | 6x ADC User key |
| WiFi&BT | WiFi&Bluetooth module |
| RTC | RTC coin cell connector |

| | |
|---|---|
| CSI | Camera CON2 interface |
| Audio O | Speaker output |
| Nano SIM | SIM card interface |
| Micro SD | Micro SD card interface |
| IR | infrared receiver |
| Debug | debug the serial port |
| RS485 | RS485 communication interface |
| PIR | PIR Infrared Sensor Head |
| Light Sensor | Light Sensor interface |
| SPI LCD | 1.8' SPI LCD |

# Install Drivers and Tool

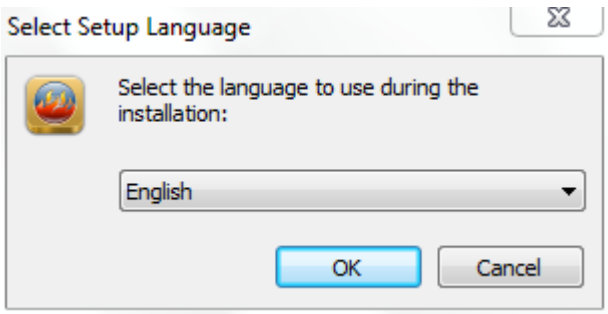
To download firmware and debug in the terminal, the following drivers and software need to be installed(for Windows computers):

| Number | Driver name | Driver | Use |
|---|---|---|---|
| 1 | Aml_USB_Burn_Tool | V3_setup_V3.1.4.exe | OTG USB Burn tool |
| 2 | DNL | dpinst64.exe | OTG USB download driver |
| 3 | CH9102x | PreInstaller.exe | Serial port debugging driver |
| 4 | Serial Terminal Tool | SecureCRT.exe | Debugging tool |

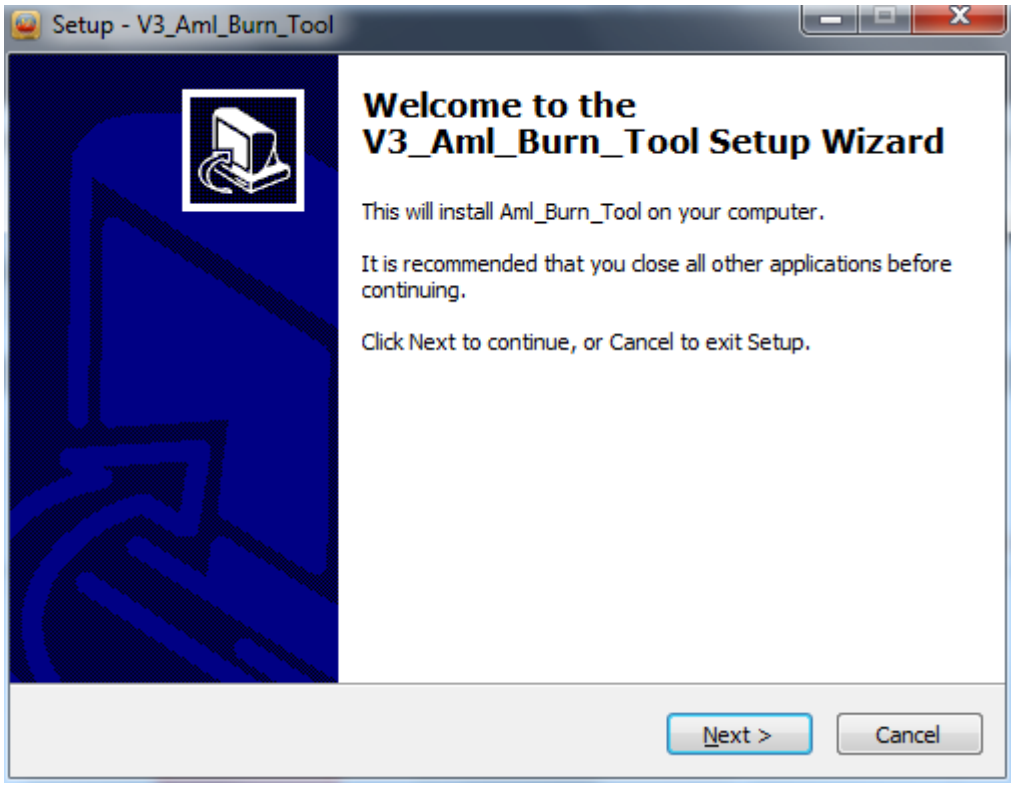
# 1.Install V3_Aml_Burn_Tool and Device Driver

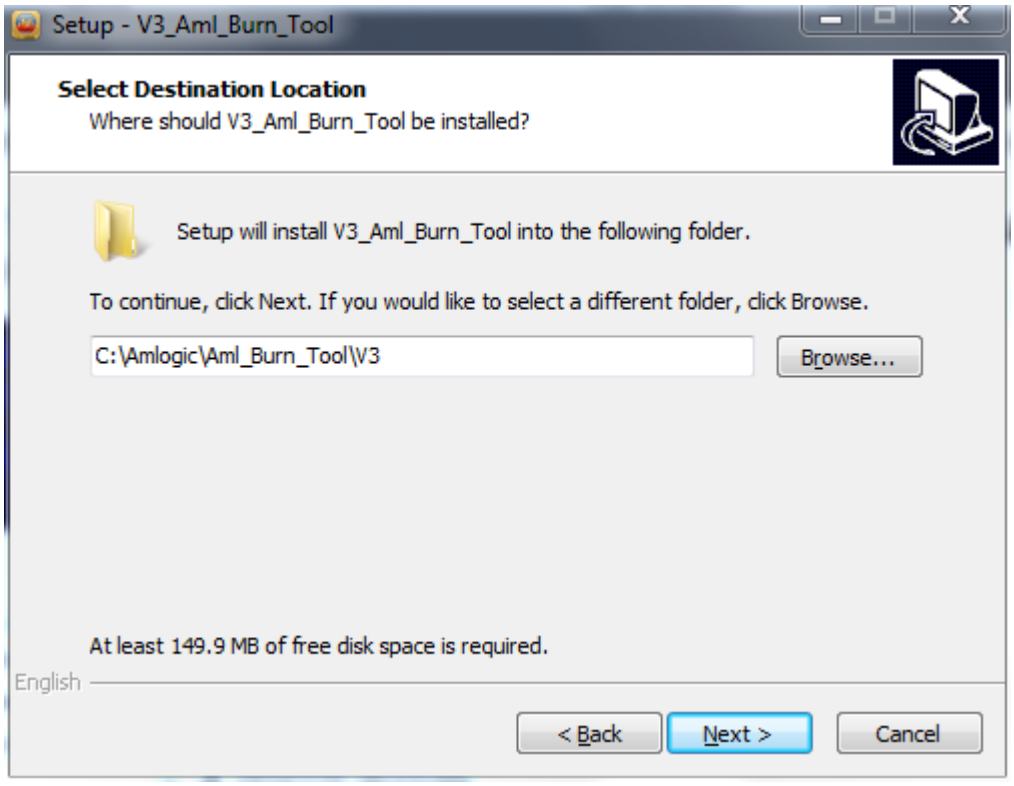
*Step 1*, open Tools/V3_setup_V3.1.4.exe.

*Step 2*, Select Setup Language.

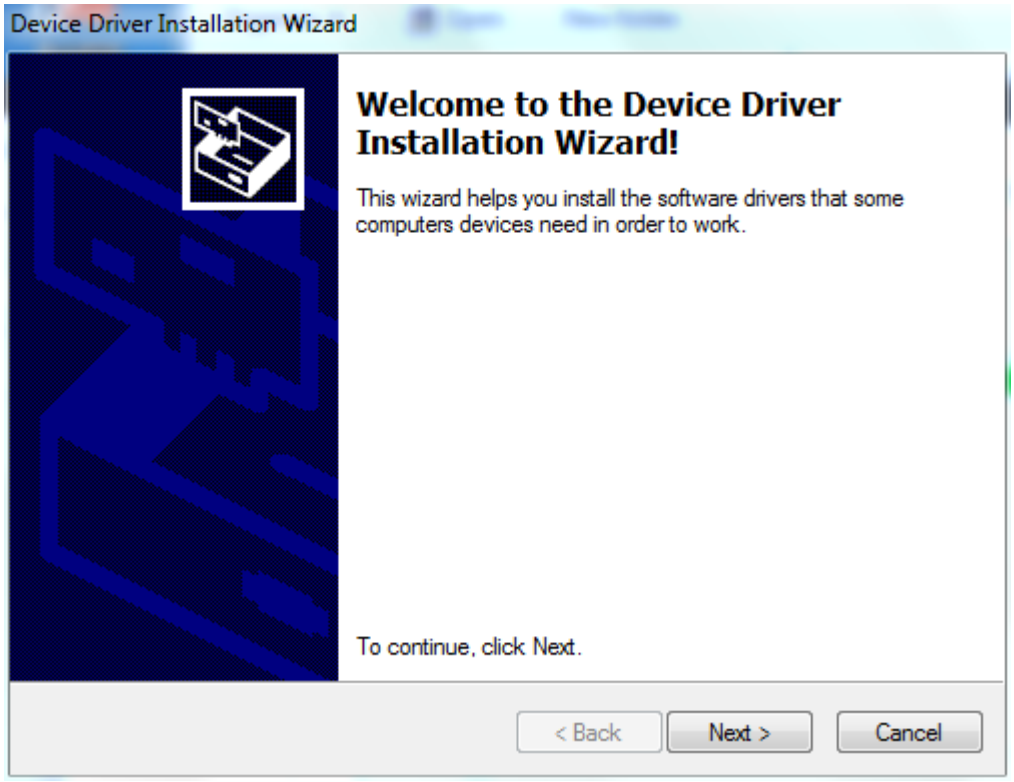


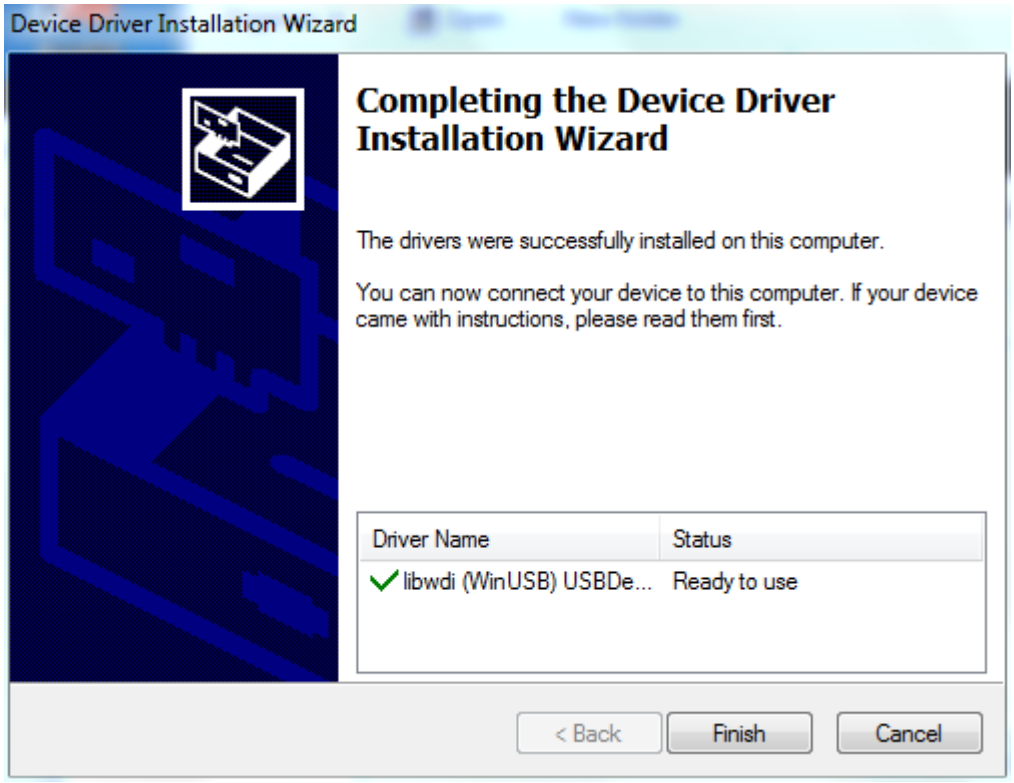
*Step 3*, click button **Next** to install.



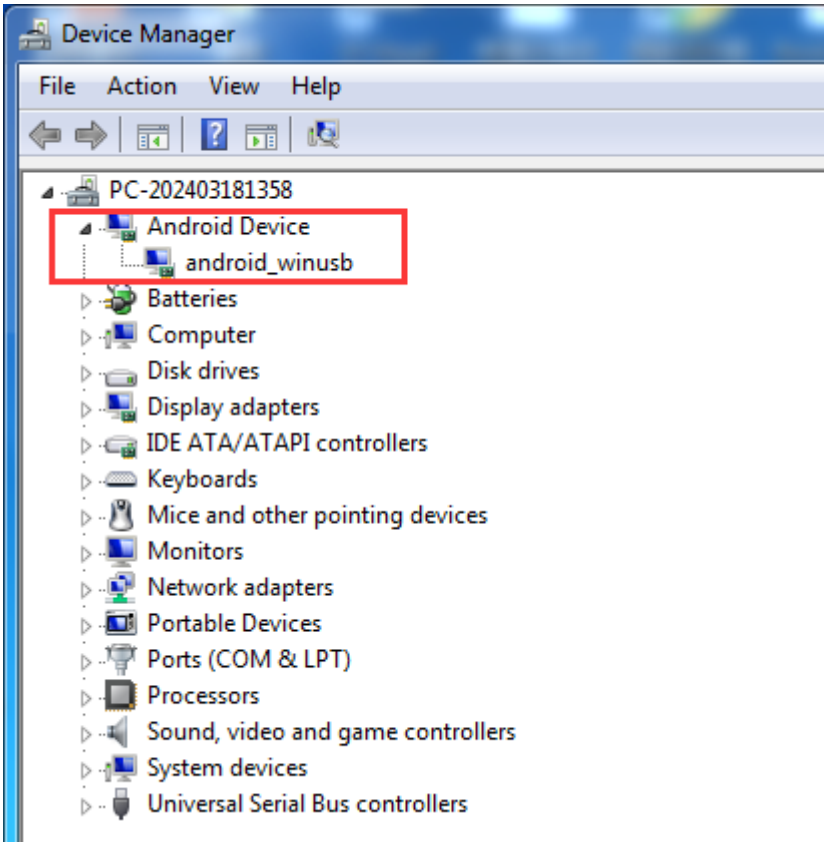




*Step 4*, The Micro USB Device Driver Installation Wizard automatically pops up after the V3_Aml_Burn_Tool installation is completed.

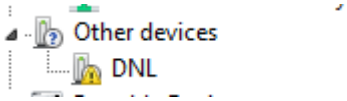


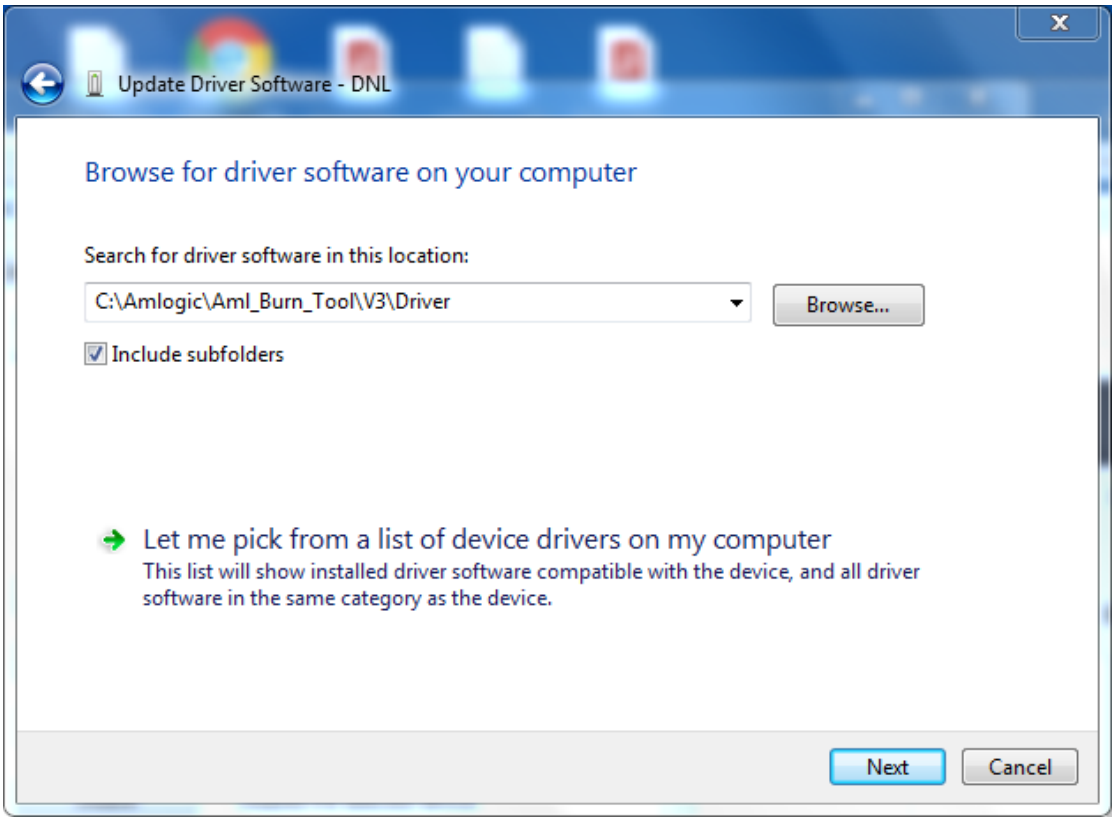

*Step 5,* after the installation is complete, connect the board and PC with Micro USB cable then power the board, the following information is displayed in the Computer Device Manager, indicating that the USB driver was successfully installed.




*Step 6,* 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.

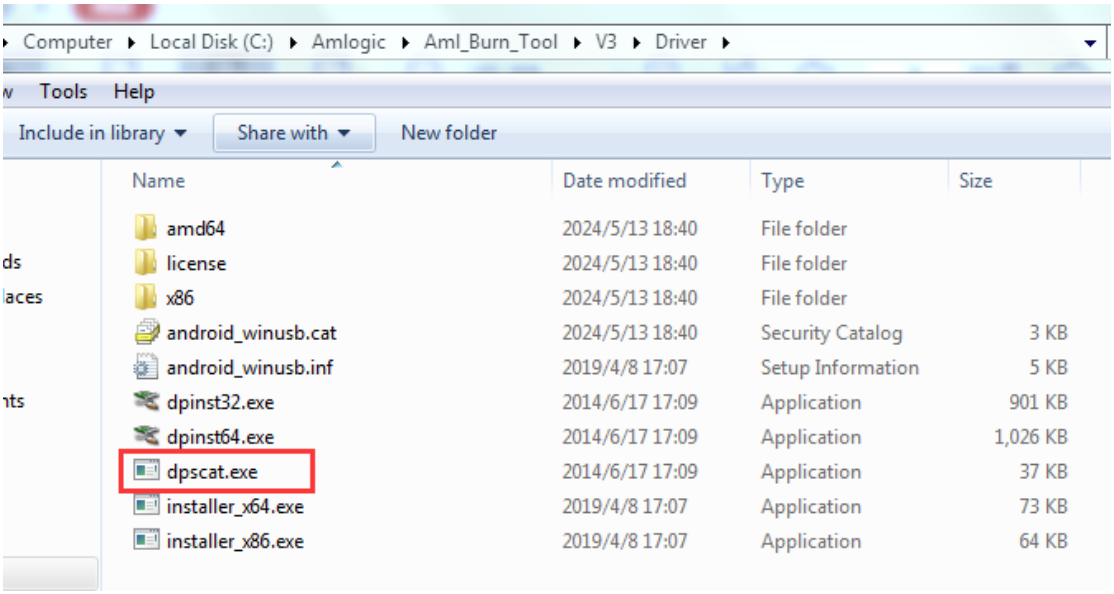



*Method 1,* Right-click on DNL and select **Properties-Driver-Update Driver**, choose to install from the specified location, and then select: \C:\Amlogic\Aml_Burn_Tool\V3\Driver.

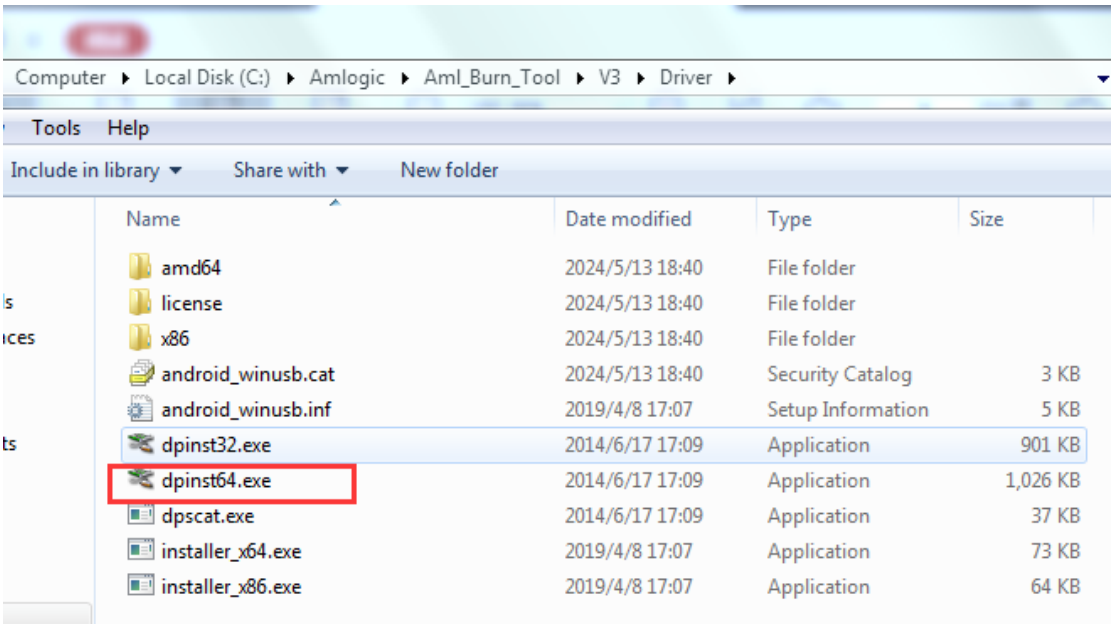




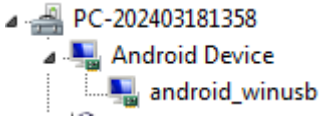
*Method 2,* First go to the directory specified by the tool installation, first click dpscat.exe. signature driver




Then according to machine's system bits, choose the corresponding installation, 64-bit selection dpinst64.exe, 32-bit selection dpinst32.exe.



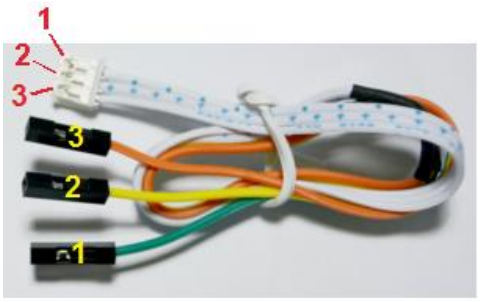

*Step 7*, after the installation is completed, the following device information can be seen in the Computer Device Manager.



# 2.Install CH9102X Driver

## 2.1 How to connect the serial port tool


| Pin | Connection Description |
|---|---|
| RXD | Receive, connect to RX pin of the board. |
| TXD | Transmit, connect to TX pin of the board. |
| GND | Ground, connect to GND pin of the board. |
| 3V3 | No need to connect. |



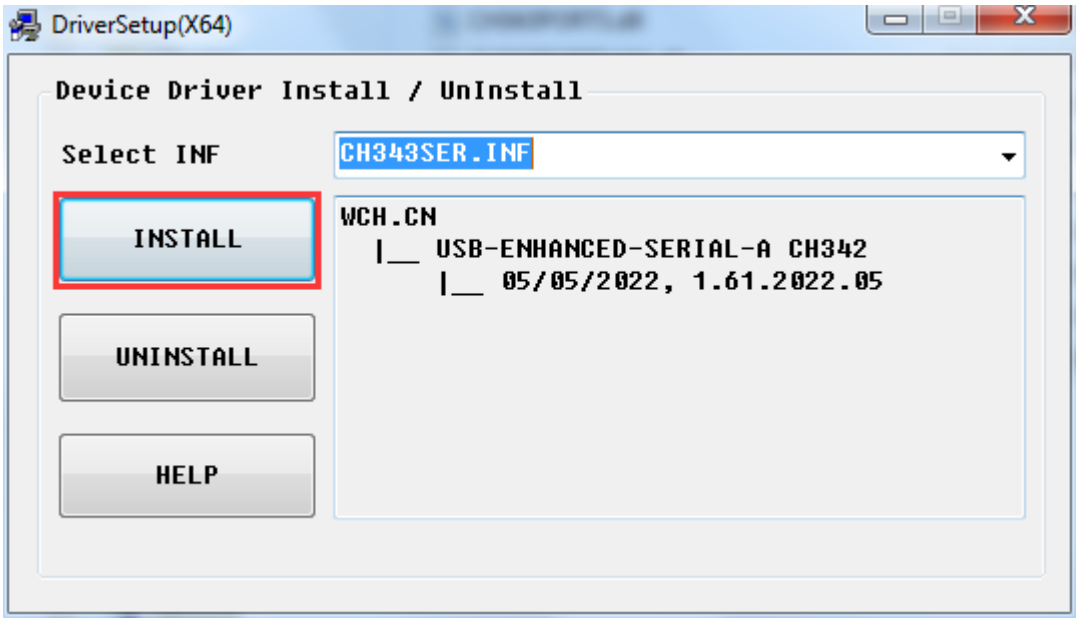

| Pin | Connection Description |
|---|---|
| 1 | RX, connect to TXD pin of the CH9102X Module. |
| 2 | TX, connect to RXD pin of the CH9102X Module. |
| 3 | Ground, connect to GND pin of the CH9102X Module. |

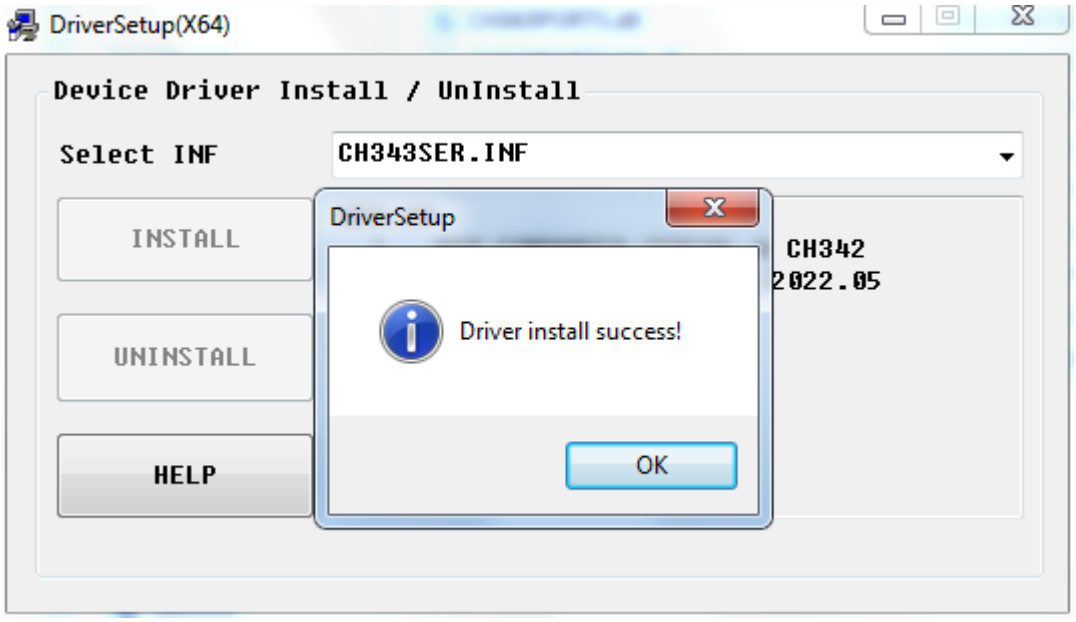
## 2.2 Install driver

*Step 1*, plug the CH9102X Module to the PC.

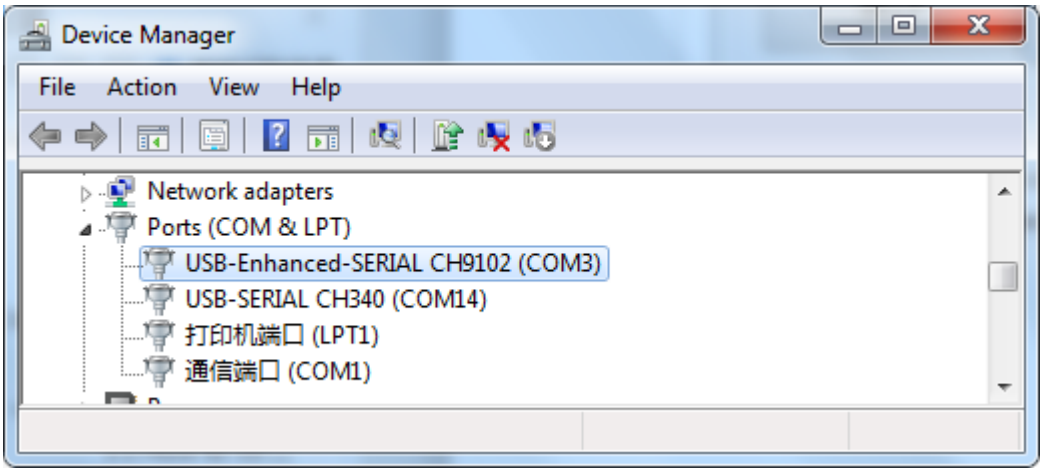
*Step 2*, unzip CH343SER.ZIP on Windows.

*Step 3*, select and install the corresponding SETUP.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.

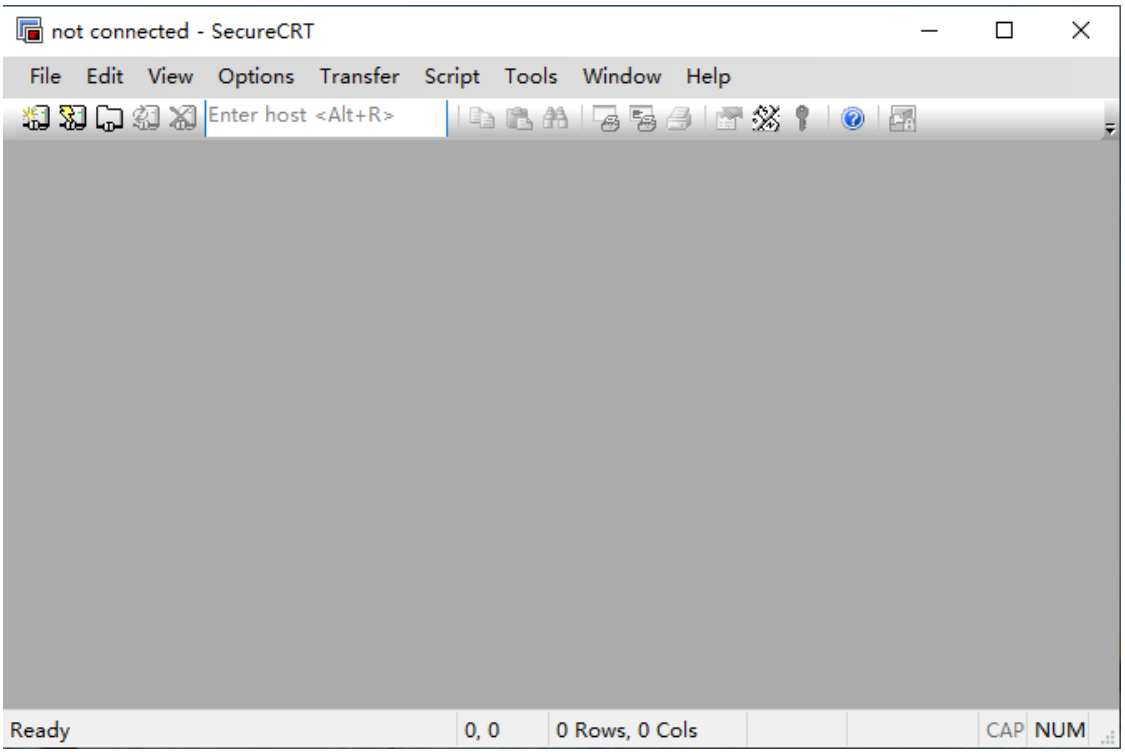


# 3.Install Serial Terminal Tool

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

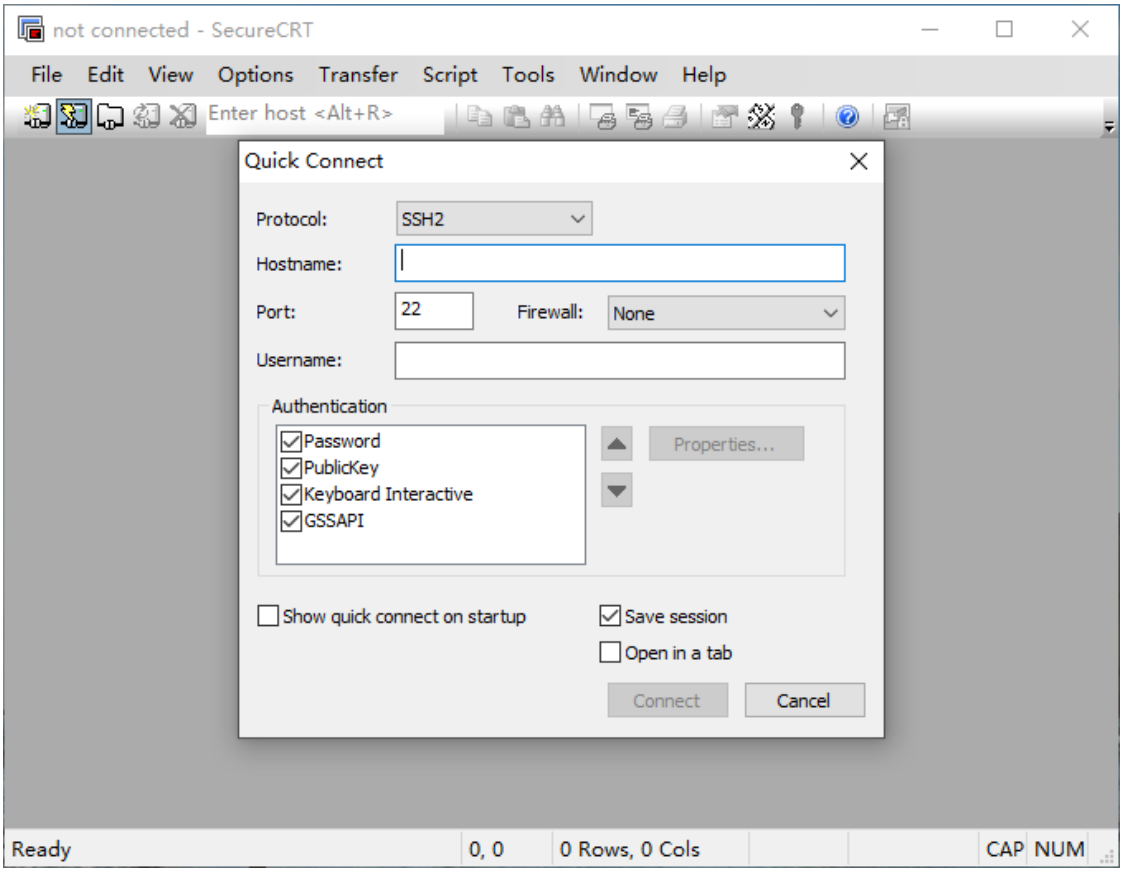
*Step 1,* unzip Platform/SecureCRT.rar on PC.

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

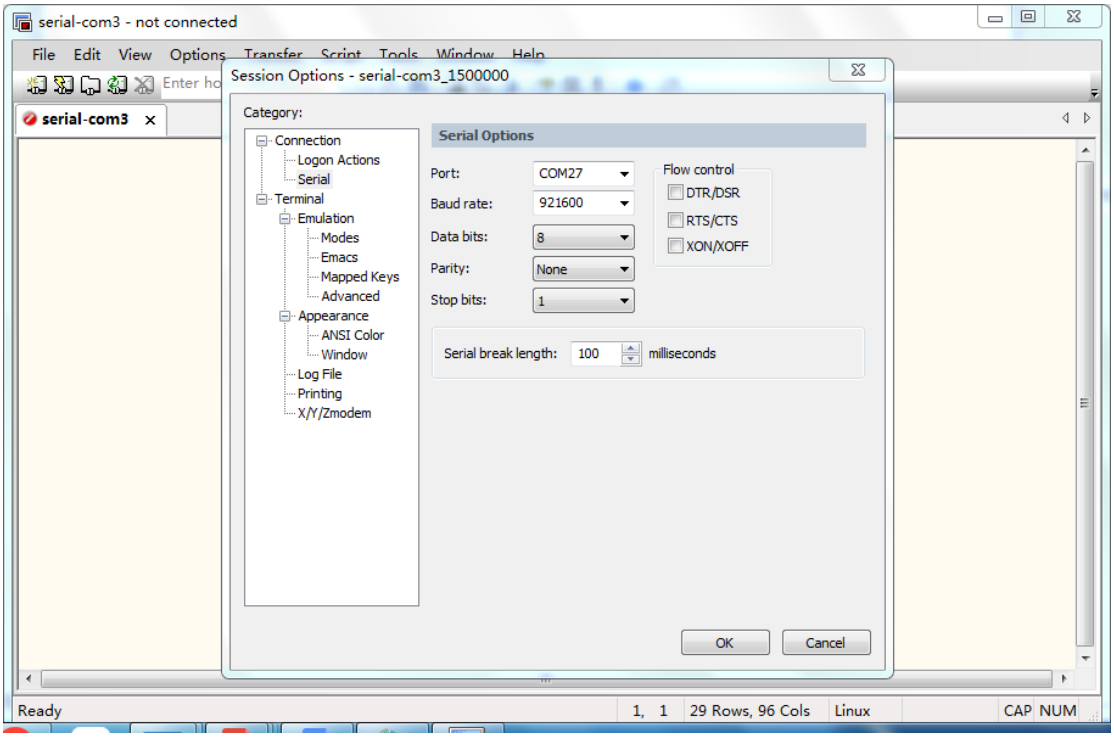



*Step 3,* confirm that the CH9102X driver has been installed and the CH9102X module is connecting to the PC.

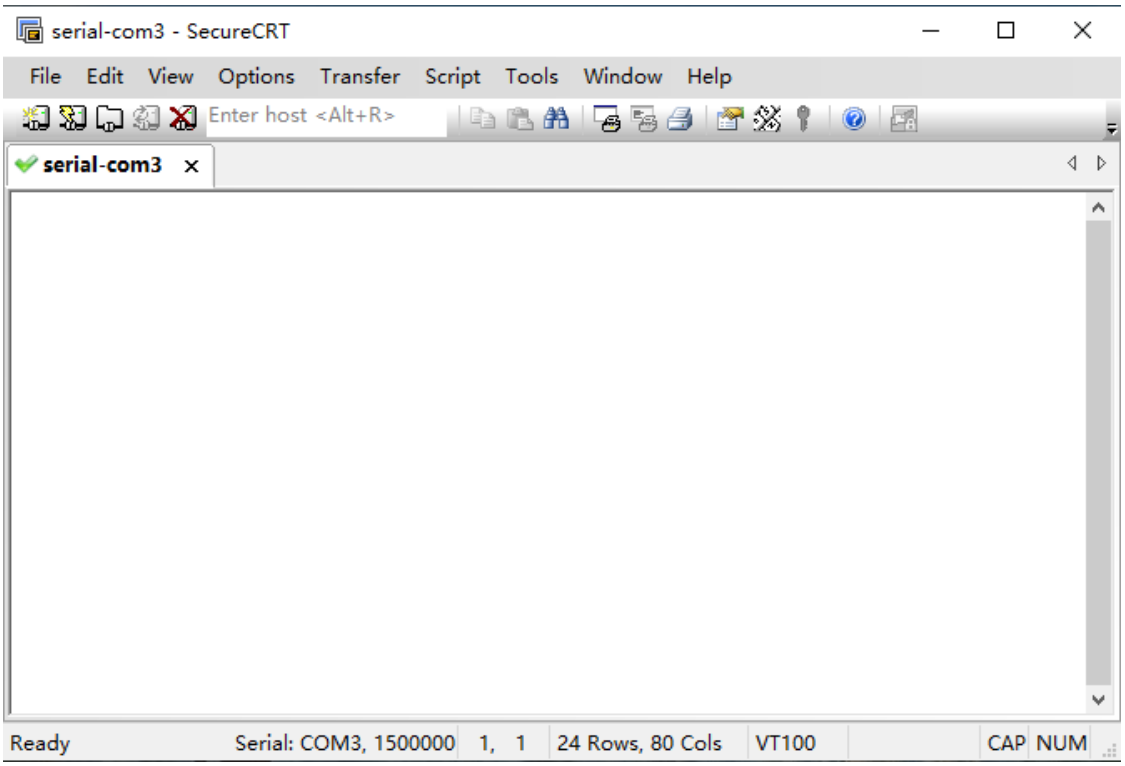
*Step 4,* click the **Quick Connect** button to go to the Quick Connect configuration screen.




*Step 5,* configure as shown in the following figure:




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




# Buildroot

## 1.Compiler Environment

It is recommended to use Ubuntu 18.04 or higher version for compilation. If you encounter an error during compilation, you can check the error message and install the corresponding software packages accordingly. Other Linux 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 200G. If you do multiple builds, you will need more hard drive space. | Ubuntu 18.04 system |

## 2.Install Tools

## 3.Compile Source

In the root directory of the source code, execute the following command:

$ ./build_all.sh

======== Build Options ========

1.c1-ae400

2.c1-ae401

```
        3.c1-ae402
        4.c2-af400
        5.c2-af401
        6.c2-af400-nand
        7.c1-ae401-nand
        8.c2-af401-nand
        9.c2-af402
        ...
================================
please select a num：4
Build all of c2-af400
======== NN Version Options ========
        1. v6.4.2.1
        2. v6.4.3
        3. v6.4.4.3
        4. no NN
        ...
====================================
please select a num：2
Build NN v6.4.4.3 version driver
======== Build WIFI Options ========
        1.AP6398
        2.AP6256
        3.AP6201BM
        4.AP6201BM-MINIME
        5.RTL8723DS
        6.no WIFI
================================
please select a num:5
Build WIFI AP6256 driver
======== Build USB Options ========
        1.UVC
        2.UAC1
        3.UAC2
        4.UVC+UAC1
        5.UVC+UAC2
        6.UVC+RNDIS
        7.UVC+UAC1+RNDIS
        8.UVC+UAC2+RNDIS
        9.no USB
================================
please select a num:9
Don't build USB
```

Compile ipcdemo

$ ./build_ipcdemo.sh

**********start build ipcdemo**********

======== Build Options ========

1.c1-ae400
2.c1-ae401
3.c1-ae402
4.c2-af400
5.c2-af401
6.c2-af400-nand
7.c1-ae401-nand
8.c2-af401-nand
9.c2-af402
10.clean

==========

please select a num：4

Build ipcdemo for c2-af400

======== Build NN Options ========

1.v6.4.2.1
2.v6.4.3
3.v6.4.4.3
4.no NN

==========

please select a num：3

Pack image

$ ./build_usbimage.sh c2_af400

If you debug the module separately, after the complete compilation once, the next debugging can directly compile each module, such as after modifying the dts, only need to compile:

$ ./build_dtb.sh c2_af400

$ ./build_usbimage.sh

Pack image:

$ ./build_usbimage.sh c2_af400

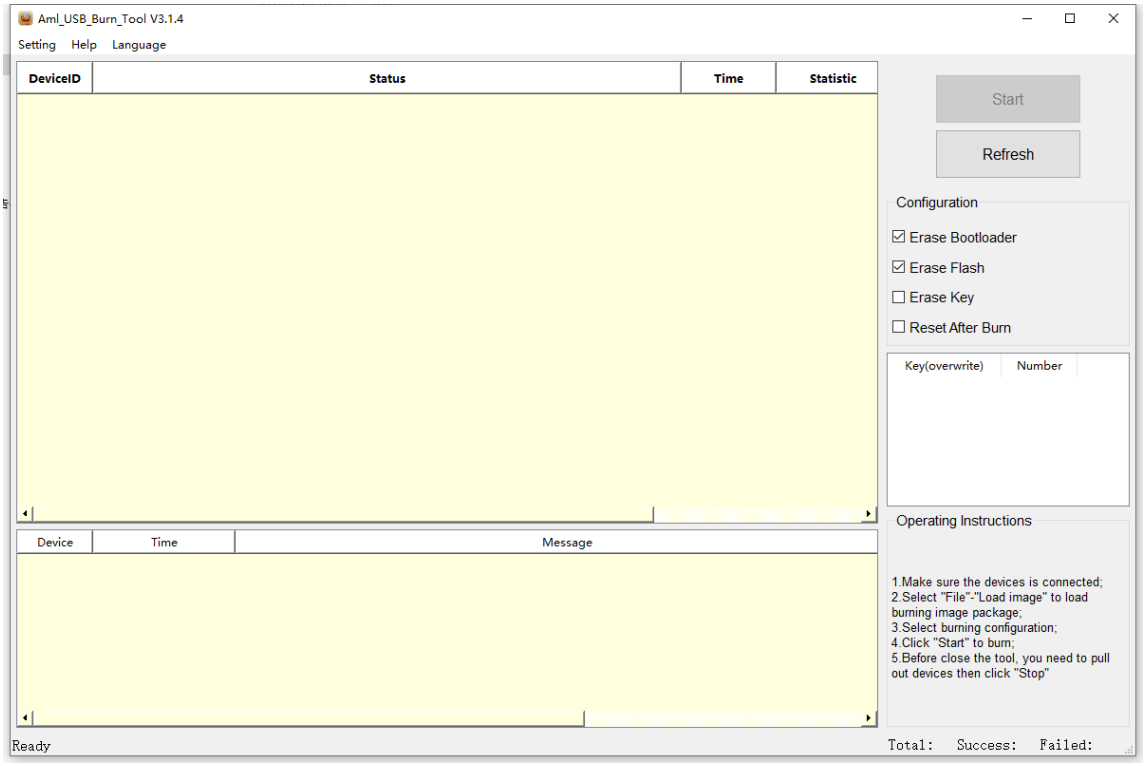
After compiling, the generated firmware is stored in the following directory:

```
wuguankang@boardcon:~/opt/c305x/c305x-new/ipc-sdk-full-release-20211125/images/c2_af400$ ls
aml_sdc_burn.ini              dtb.img                        meson-c2-c305x-af400.dtb  u-boot.bin
aml_upgrade_package.conf      Image.gz                       platform.conf             u-boot.bin.sd.bin
aml_upgrade_package_emmc.conf logo.img                       rootfs.ext2               usb_flow.aml
aml_upgrade_package.img       meson-c2-c305x-af400-1g.dtb    rootfs.ext2.img2simg
boot.img                      meson-c2-c305x-af400-256m.dtb  rootfs.ext4
wuguankang@boardcon:~/opt/c305x/c305x-new/ipc-sdk-full-release-20211125/images/c2_af400$
```

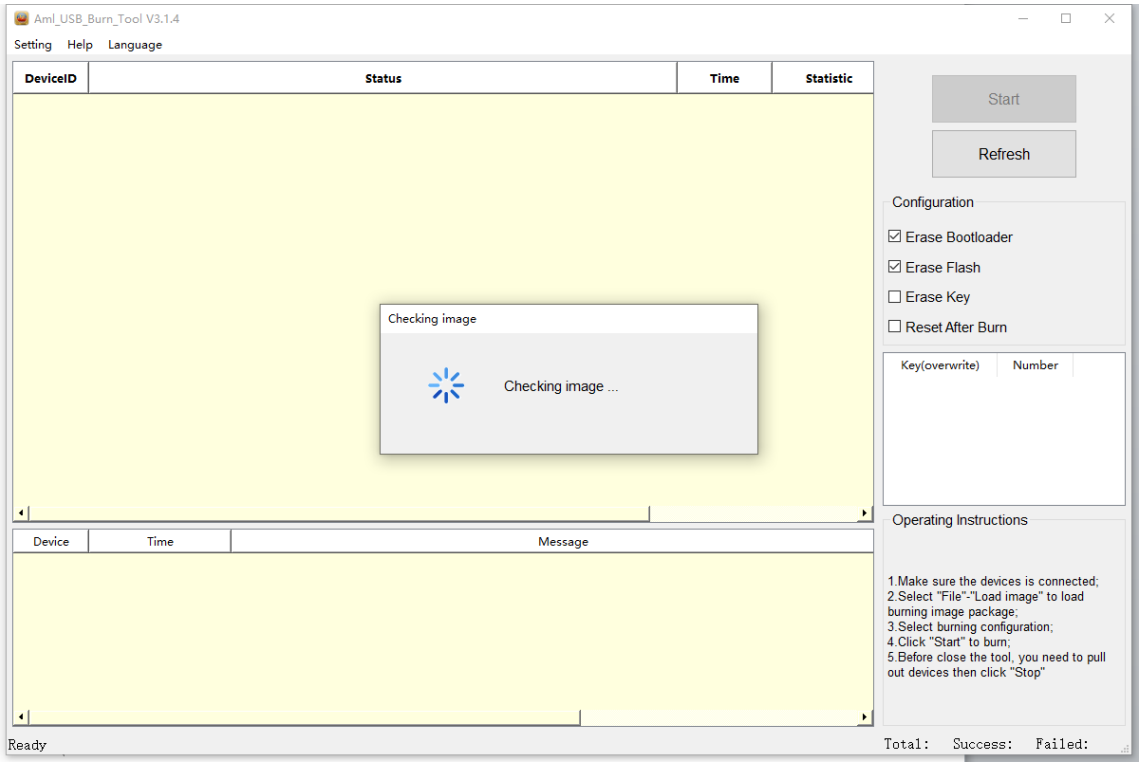
The aml_upgrade_package.img file is the firmware we need.

# 4.Burn Images

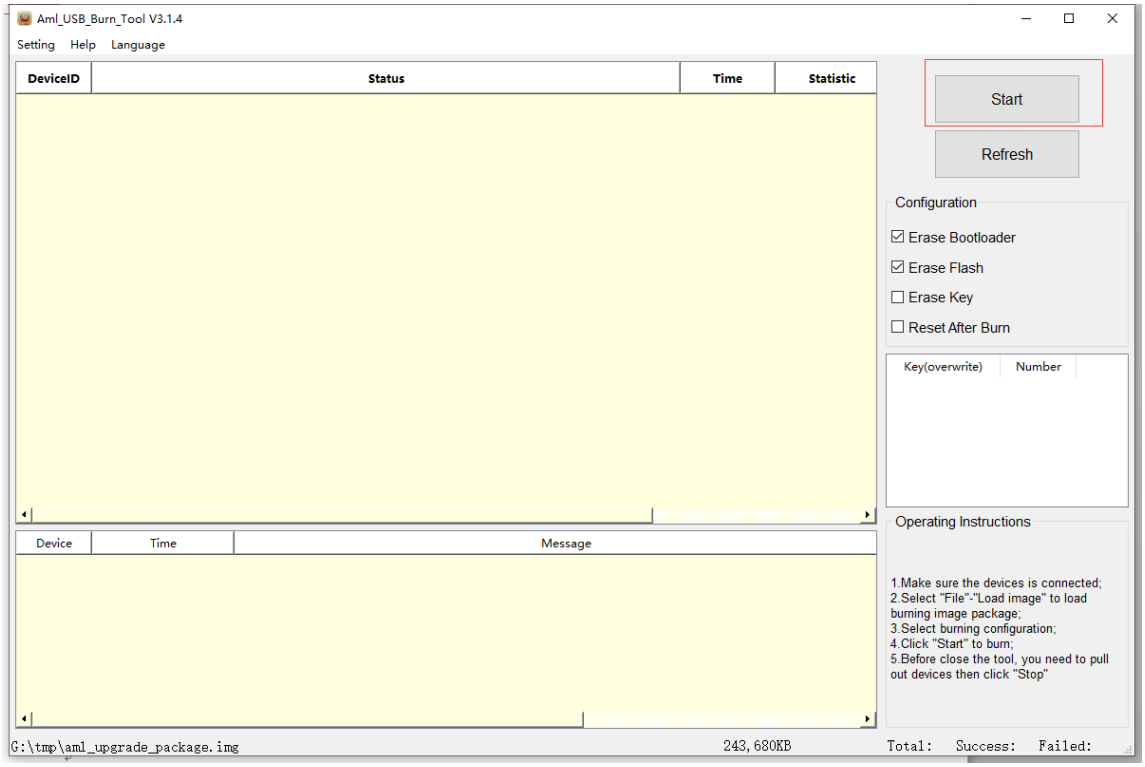
Step 1, Open the Aml_Burn_Tool software.



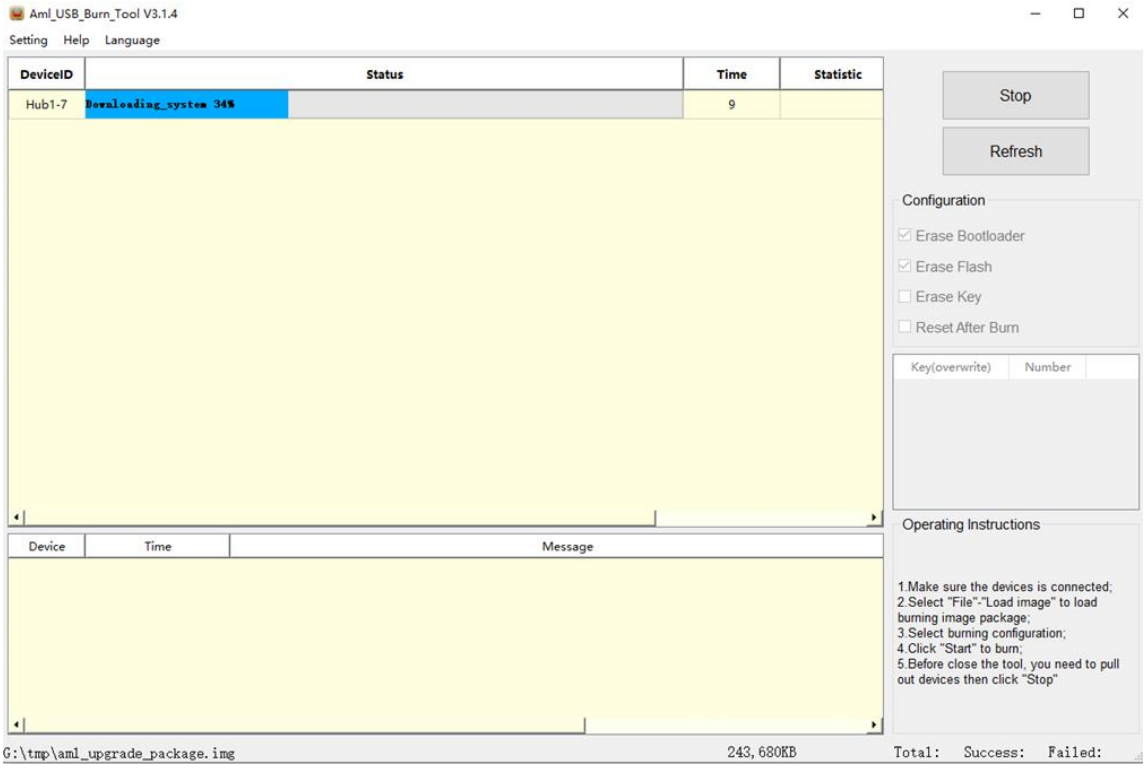

Step2,Load the firmware to burn.




Step3,Click the start button.




Step4, Plug in the OTG, hold down the Update button, power it back on, and the firmware will be burned.



# 5.Buildroot Application

## 5.1 SPI display

The EM305X default support SPI display. Only display “Boardcon”.

## 5.2 SD card

*Step 1,* insert the microSD card into the card slot, then power on.

*Step 2,* view the SD card automatic mount directory.

# df -h

```
# df -h
Filesystem                Size      Used Available Use% Mounted on
/dev/system             487.9M    222.5M    229.6M  49% /
devtmpfs                397.0M         0    397.0M   0% /dev
tmpfs                   692.4M         0    692.4M   0% /dev/shm
tmpfs                   692.4M    160.0K    692.3M   0% /tmp
tmpfs                   692.4M    180.0K    692.2M   0% /run
/dev/data                 4.4G     72.0K      4.2G   0% /data
/dev/mmcblk1p1           57.0G      3.0G     51.1G   6% /media/mmcblk1p1
#
#
```

## 5.3 USB host

*Step 1,* insert the USB 2.0 flash drive into the USB interface.

*Step 2,* Disconnect the OTG device

*Step 3,* view the USB flash automatic mount directory.

# df -h

```
# df -h
Filesystem                Size      Used Available Use% Mounted on
/dev/system             487.9M    222.5M    229.6M  49% /
devtmpfs                397.0M         0    397.0M   0% /dev
tmpfs                   692.4M         0    692.4M   0% /dev/shm
tmpfs                   692.4M    164.0K    692.2M   0% /tmp
tmpfs                   692.4M    204.0K    692.2M   0% /run
/dev/data                 4.4G     76.0K      4.2G   0% /data
/dev/mmcblk1p1           57.0G      3.0G     51.1G   6% /media/mmcblk1p1
/dev/sda1                28.9G     25.9G      3.0G  90% /media/sda1
#
```

## 5.4 Ethernet

*Step 1,* connect the network cable to the Ethernet port.

*Step 2,* according to the terminal serial port printing information, it can be seen the Gigabit Ethernet recognition is successful.

```
[ 6212.621232@1] meson6-dwmac ff080000.ethernet eth0: Link is Up - 1Gbps/Full - flow control rx/tx
[ 6212.621694@1] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
```

# ifconfig

```
# ifconfig
eth0      Link encap:Ethernet  HWaddr 02:AD:33:01:5B:8C
          inet addr:192.168.0.187  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::ad:33ff:fe01:5b8c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1367 errors:0 dropped:0 overruns:0 frame:0
          TX packets:143 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:170006 (166.0 KiB)  TX bytes:26644 (26.0 KiB)
          Interrupt:42

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:16 errors:0 dropped:0 overruns:0 frame:0
          TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1184 (1.1 KiB)  TX bytes:1184 (1.1 KiB)
```

*Step 3*, internet test.

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

```
# ping -I eth0 www.boardcon.com
PING www.boardcon.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=1 ttl=48 time=174.579 ms
64 bytes from 67.222.54.196: seq=2 ttl=48 time=174.763 ms
64 bytes from 67.222.54.196: seq=3 ttl=48 time=174.418 ms
64 bytes from 67.222.54.196: seq=4 ttl=48 time=174.560 ms
64 bytes from 67.222.54.196: seq=5 ttl=48 time=174.781 ms
64 bytes from 67.222.54.196: seq=6 ttl=48 time=174.145 ms
64 bytes from 67.222.54.196: seq=7 ttl=48 time=177.189 ms
```

## 5.5 WIFI

*Step 1, Configure the SSID and password for the WiFi you want to connect to.*

```
# cat /etc/wpa_supplicant.conf
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1
update_config=1

network={
        ssid="Boardcon"
        psk="Boardcon43435656"
        key_mgmt=WPA-PSK
}
#
```

*Step 2, Connect to WiFi.*

```
# wpa_supplicant -i wlan0 -Dnl80211 -c /etc/wpa_supplicant.conf  -B
Successfully initialized wpa_supplicant
[ 4070.262055@1] RTW: cfg80211_rtw_get_txpower(wlan0) total max: 1600 mbm
[ 4070.262309@1] RTW: cfg80211_rtw_get_txpower(wlan0) total max: 1600 mbm
[ 4070.263597@1] RTW: cfg80211_rtw_get_txpower(wlan0) total max: 1600 mbm
[ 4070.267079@1] RTW: cfg80211_rtw_get_txpower(wlan0) total max: 1600 mbm
[ 4070.268755@1] RTW: rtw_ndev_notifier_call(wlan0) state:4
[ 4070.269405@1] RTW: rtw_ndev_notifier_call(wlan0) state:4
[ 4070.269784@1] RTW: rtw_ndev_notifier_call(wlan0) state:4
[ 4070.289113@1] RTW: cfg80211_rtw_del_key(wlan0) key_index=0, addr=          (null)
[ 4070.289718@1] RTW: cfg80211_rtw_del_key(wlan0) key_index=1, addr=          (null)
[ 4070.290433@1] RTW: cfg80211_rtw_del_key(wlan0) key_index=2, addr=          (null)
[ 4070.291344@1] RTW: cfg80211_rtw_del_key(wlan0) key_index=3, addr=          (null)
[ 4070.292346@1] RTW: cfg80211_rtw_del_key(wlan0) key_index=4, addr=          (null)
[ 4070.294490@1] RTW: cfg80211_rtw_del_key(wlan0) key_index=5, addr=          (null)
[ 4070.295917@1] RTW: rtw_ndev_notifier_call(wlan0) state:4
[ 4070.298426@1] RTW: rtw_ndev_notifier_call(wlan0) state:4
# [ 4070.395806@1] RTW: cfg80211_rtw_scan(wlan0)
[ 4070.399143@0] RTW: [HW_VAR_CHECK_TXBUF] no packet in tx packet buffer (0)
[ 4070.399578@0] RTW: wlan0 sleep m0=0x00000003, ori reg_0x4d4=0x00000000
```

*Step 3*, Dynamic IP acquisition.

```
# udhcpc -i wlan0 -R
udhcpc: started, v1.31.1
udhcpc: sending discover
[ 4156.868740@0] RTW: rtl8723d_fill_default_txdesc(wlan0): SP Packet(0x0800) rate=0x0 SeqNum = 4
[ 4156.872837@0] RTW: rtw_cpwm_polling: polling cpwm OK! rpwm_retry=0, cpwm_orig=80, cpwm_now=00 , 0x100=0x3f
[ 4156.874235@0] RTW: rtw_set_ps_mode(wlan0) Leave 802.11 power save - WIFI-LPS_CTRL_SPECIAL_PACKET
[ 4156.874652@0] RTW: rtl8723d_set_FwPwrMode_cmd(): FW LPS mode = 0, SmartPS=2
udhcpc: sending select for 192.168.0.123
[ 4156.912755@0] RTW: rtl8723d_fill_default_txdesc(wlan0): SP Packet(0x0800) rate=0x0 SeqNum = 5
udhcpc: lease of 192.168.0.123 obtained, lease time 86400
deleting routers
adding dns 202.96.134.33
adding dns 202.96.128.86
#
```

*Step 4,* View WiFi devices.

```
# ifconfig
eth0      Link encap:Ethernet  HWaddr 02:AD:33:01:5B:8C
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:31691 errors:0 dropped:0 overruns:0 frame:0
          TX packets:37134 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3211104 (3.0 MiB)  TX bytes:48368139 (46.1 MiB)
          Interrupt:42

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:16 errors:0 dropped:0 overruns:0 frame:0
          TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1184 (1.1 KiB)  TX bytes:1184 (1.1 KiB)

wlan0     Link encap:Ethernet  HWaddr 0C:CF:89:0A:7A:A2
          inet addr:192.168.0.123  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::ecf:89ff:fe0a:7aa2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:568 errors:0 dropped:80 overruns:0 frame:0
          TX packets:5 errors:0 dropped:22 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:85294 (83.2 KiB)  TX bytes:1146 (1.1 KiB)
```

*Step 4,* internet test:

```
# ping www.boardcon.com
PING www.boardcon.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=48 time=277.105 ms
64 bytes from 67.222.54.196: seq=1 ttl=48 time=223.432 ms
64 bytes from 67.222.54.196: seq=2 ttl=48 time=282.189 ms
64 bytes from 67.222.54.196: seq=3 ttl=48 time=293.511 ms
64 bytes from 67.222.54.196: seq=4 ttl=48 time=430.171 ms
64 bytes from 67.222.54.196: seq=5 ttl=48 time=227.906 ms
```

## 5.6 Bluetooth

*Step 1,* connect the antenna.

*Step 2,* execute the following command to enable Bluetooth:

# bt_init_bcm.sh

```
# bt_init_bcm.sh
killall: brcm_patchram_plus1: no process killed
[   92.920199@1] BT_RADIO going: off
[   92.920243@1] AML_BT: going OFF
[   93.453797@1] BT_RADIO going: on
[   93.453947@1] AML_BT: going ON
###AMPAK FW Auto detection patch version = [v1.1 20161117]###
FW folder path = /etc/bt
proc_resetCan't get device info: No such device
# chip id = BCM4345C5
FW path = /etc/bt/BCM4345C5.hcd
send hci_download_minidriverproc_resetDone setting line discpline
```

*Step 3,* viewing device status:

```
# hciconfig -a
hci0:   Type: Primary  Bus: UART
        BD Address: AA:AA:AA:AA:AA:AA  ACL MTU: 1021:8  SCO MTU: 64:1
        DOWN
        RX bytes:2169 acl:0 sco:0 events:120 errors:0
        TX bytes:1362 acl:0 sco:0 commands:120 errors:0
        Features: 0xbf 0xfe 0xcf 0xfe 0xdb 0xff 0x7b 0x87
        Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
        Link policy: RSWITCH SNIFF
        Link mode: SLAVE ACCEPT
```

## 5.7 4G(EC20)

*Step 1,* insert 4G module to PCIe slot (4G model:EC20).

*Step 2,* connect antenna and insert SIM card And disconnetct otg.

*Step 3,* execute the following command to realize ppp dialing:

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

```
# pppd call quectel-ppp &
# pppd options in effect:
debug           # (from /etc/ppp/peers/quectel-ppp)
nodetach                # (from /etc/ppp/peers/quectel-ppp)
dump            # (from /etc/ppp/peers/quectel-ppp)
noauth          # (from /etc/ppp/peers/quectel-ppp)
user test               # (from /etc/ppp/peers/quectel-ppp)
password ??????         # (from /etc/ppp/peers/quectel-ppp)
remotename 3gppp                # (from /etc/ppp/peers/quectel-ppp)
/dev/ttyUSB3            # (from /etc/ppp/peers/quectel-ppp)
115200          # (from /etc/ppp/peers/quectel-ppp)
lock            # (from /etc/ppp/peers/quectel-ppp)
connect chat -s -v -f /etc/ppp/peers/quectel-chat-connect               # (from /etc/ppp/peers/quectel-ppp)
disconnect chat -s -v -f /etc/ppp/peers/quectel-chat-disconnect         # (from /etc/ppp/peers/quectel-ppp)
nocrtscts               # (from /etc/ppp/peers/quectel-ppp)
modem           # (from /etc/ppp/peers/quectel-ppp)
hide-password           # (from /etc/ppp/peers/quectel-ppp)
novj            # (from /etc/ppp/peers/quectel-ppp)
novjccomp               # (from /etc/ppp/peers/quectel-ppp)
ipcp-accept-local               # (from /etc/ppp/peers/quectel-ppp)
ipcp-accept-remote              # (from /etc/ppp/peers/quectel-ppp)
ipparam 3gppp           # (from /etc/ppp/peers/quectel-ppp)
noipdefault             # (from /etc/ppp/peers/quectel-ppp)
ipcp-max-failure 30             # (from /etc/ppp/peers/quectel-ppp)
defaultroute            # (from /etc/ppp/peers/quectel-ppp)
usepeerdns              # (from /etc/ppp/peers/quectel-ppp)
noccp           # (from /etc/ppp/peers/quectel-ppp)
abort on (BUSY)
abort on (NO CARRIER)
abort on (NO DIALTONE)
abort on (ERROR)
abort on (NO ANSWER)
timeout set to 30 seconds
```

```
timeout set to 30 seconds
send (AT^M)
expect (OK)
AT^M^M
OK
 -- got it

send (ATE0^M)
expect (OK)
^M
ATE0^M^M
OK
 -- got it

send (ATI;+CSUB;+CSQ;+CPIN?;+COPS?;+CGREG?;&D2^M)
expect (OK)
^M
^M
Quectel^M
EC20F^M
Revision: EC20CEFILGR06A02M1G^M
^M
SubEdition: V02^M
^M
+CSQ: 29,99^M
^M
+CPIN: READY^M
^M
+COPS: 0,0,"Suning\00",7^M
^M
+CGREG: 0,1^M
^M
OK
```

```
^M
^M
CONNECT
 -- got it

Script chat -s -v -f /etc/ppp/peers/quectel-chat-connect finished (pid 1917), status = 0x0
Serial connection established.
using channel 1
Using interface ppp0
Connect: ppp0 <--> /dev/ttyUSB3
sent [LCP ConfReq id=0x1 <asyncmap 0x0> <magic 0xbf7b460e> <pcomp> <accomp>]
rcvd [LCP ConfReq id=0x0 <asyncmap 0x0> <auth chap MD5> <magic 0xd0ff60df> <pcomp> <accomp>]
sent [LCP ConfAck id=0x0 <asyncmap 0x0> <auth chap MD5> <magic 0xd0ff60df> <pcomp> <accomp>]
rcvd [LCP ConfAck id=0x1 <asyncmap 0x0> <magic 0xbf7b460e> <pcomp> <accomp>]
rcvd [LCP DiscReq id=0x1 magic=0xd0ff60df]
rcvd [CHAP Challenge id=0x1 <2f29f649d9d90144696c9b94823155b4>, name = "UMTS_CHAP_SRVR"]
sent [CHAP Response id=0x1 <e47ff6a44af9320bc3c891899d86b159>, name = "test"]
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>]
rcvd [IPCP ConfReq id=0x0]
sent [IPCP ConfNak id=0x0 <addr 0.0.0.0>]
rcvd [IPCP ConfNak id=0x1 <addr 10.138.181.9> <ms-dns1 120.80.80.80> <ms-dns2 221.5.88.88>]
sent [IPCP ConfReq id=0x2 <addr 10.138.181.9> <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 10.138.181.9> <ms-dns1 120.80.80.80> <ms-dns2 221.5.88.88>]
Could not determine remote IP address: defaulting to 10.64.64.64
local  IP address 10.138.181.9
remote IP address 10.64.64.64
primary   DNS address 120.80.80.80
secondary DNS address 221.5.88.88
```

*Step 4*, execute the following to view the network interface status:

```
ppp0      Link encap:Point-to-Point Protocol
          inet addr:10.138.181.9  P-t-P:10.64.64.64  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:52 (52.0 B)  TX bytes:58 (58.0 B)
```

*Step 5, ADD DNS*

```
# cat /etc/resolv.conf
# Generated by dhcpcd
# /etc/resolv.conf.head can replace this line
# /etc/resolv.conf.tail can replace this line
nameserver 8.8.8.8
```

*Step 6*, execute the following to check the connectivity of the detection network:

```
# ping -I ppp0 www.boardcon.com
PING www.boardcon.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=47 time=548.762 ms
64 bytes from 67.222.54.196: seq=1 ttl=47 time=505.090 ms
64 bytes from 67.222.54.196: seq=2 ttl=47 time=464.702 ms
64 bytes from 67.222.54.196: seq=3 ttl=47 time=425.226 ms
64 bytes from 67.222.54.196: seq=4 ttl=47 time=384.604 ms
64 bytes from 67.222.54.196: seq=5 ttl=47 time=352.323 ms
```

## 5.8 KEY

Pressing different keys will print different key values.

```
# [ 4876.108884@0] input input2: key 115 down
[ 4876.300815@0] input input2: key 115 up

#
# [ 4884.205551@0] input input2: key 29 down
[ 4884.364850@0] input input2: key 29 up

# [ 4886.444425@0] input input1: key 116 down.
[ 4886.708424@0] input input1: key 116 up.

# [ 4887.820936@0] input input2: key 114 down
[ 4888.012957@0] input input2: key 114 up

# [ 4889.741028@0] input input2: key 141 down
[ 4889.872907@0] input input2: key 141 up

# [ 4891.208981@0] input input2: key 28 down
[ 4891.405538@0] input input2: key 28 up

# [ 4901.800881@0] input input2: key 141 down
[ 4901.960829@0] input input2: key 141 up

# ■
```

## 5.9 PIR

*Step 1,* execute the following command to test PIR

# pir_test.sh

The PIR sensor obtains a value of 1 if it senses the object displacement in place, and 0 otherwise.

```
# pir_test.sh
0
1
1
1
1
1
0
```

## 5.10 RTC

*Step 1,* install the coin cell battery.

*Step 2,* the RTC synchronization test is performed while connected to the network.

date -s "2024-05-17 15:26:00" //use the command date -s "YY-MM-DD hh:mm:ss" to modif the date and time of the system

# hwclock -w // set RTC from system time

# hwclock

```
# date -s "2024-05-17 15:26:00"
Fri May 17 15:26:00 CST 2024
#
#
# hwclock -w
#
# hwclock
Fri May 17 15:26:11 2024  0.000000 seconds
#
# hwclock
Fri May 17 15:27:43 2024  0.000000 seconds
#
```

After setting the time zone correctly, confirm that the time matches the actual time. After confirming the RTC time synchronization, power off, only the RTC coin cell battery is kept, other devices are disconnected, the development board is idle for a period of time, and then connect the power supply to confirm whether the time is running.

## 5.11 IR

*Step 1,* execute the following command to turn on IR receiving data information printing:

# echo 1 > /sys/class/remote/amremote/debug_enable

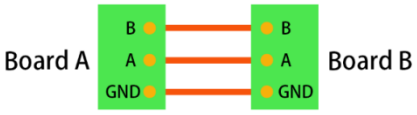
# echo 7 > /proc/sys/kernel/printk

*Step 2,* press the remote control against the infrared receiver head, and the return value can be obtained in the terminal printing.

```
# echo 1 > /sys/class/remote/amremote/debug_enable
#
# echo 7 > /proc/sys/kernel/printk
#
# [ 1384.196777@1] meson-ir:framecode=0xe916fe01
[ 1384.196819@1] meson-ir:receive scancode=0x16
[ 1384.197151@1] meson-ir:keypressed=0x0
[ 1384.197653@1] meson-ir:report key!!
[ 1384.277209@1] meson-ir:ir controller busy flag = 0
[ 1384.277282@1] meson-ir:keyup!!
[ 1385.801657@1] meson-ir:framecode=0xbf40fe01
[ 1385.801700@1] meson-ir:receive scancode=0x40
[ 1385.802033@1] meson-ir:keypressed=0x0
[ 1385.802526@1] meson-ir:report key!!
[ 1385.885217@1] meson-ir:ir controller busy flag = 0
[ 1385.885294@1] meson-ir:keyup!!
```

## 5.12 RS485

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




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

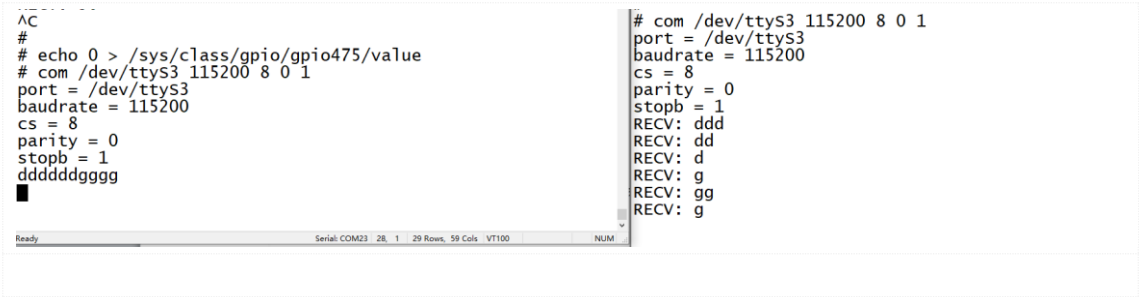
gpio475 value=1 receive ,gpio475 value=0 send

Left receives, right sends.

```
# echo 475 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio475/direction
# echo 1 > /sys/class/gpio/gpio475/value
# com /dev/ttyS3 115200 8 0 1
port = /dev/ttyS3
baudrate = 115200
cs = 8
parity = 0
stopb = 1
RECV: 1
RECV: 23
RECV: 4
RECV: 5
RECV: 6
RECV: 78
RECV: 90
```

```
# echo 475 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio475/direction
# echo 0 > /sys/class/gpio/gpio475/value
# com /dev/ttyS3 115200 8 0 1
port = /dev/ttyS3
baudrate = 115200
cs = 8
parity = 0
stopb = 1
1234567890
```

Left sends,right receives.




## 5.13 Camera

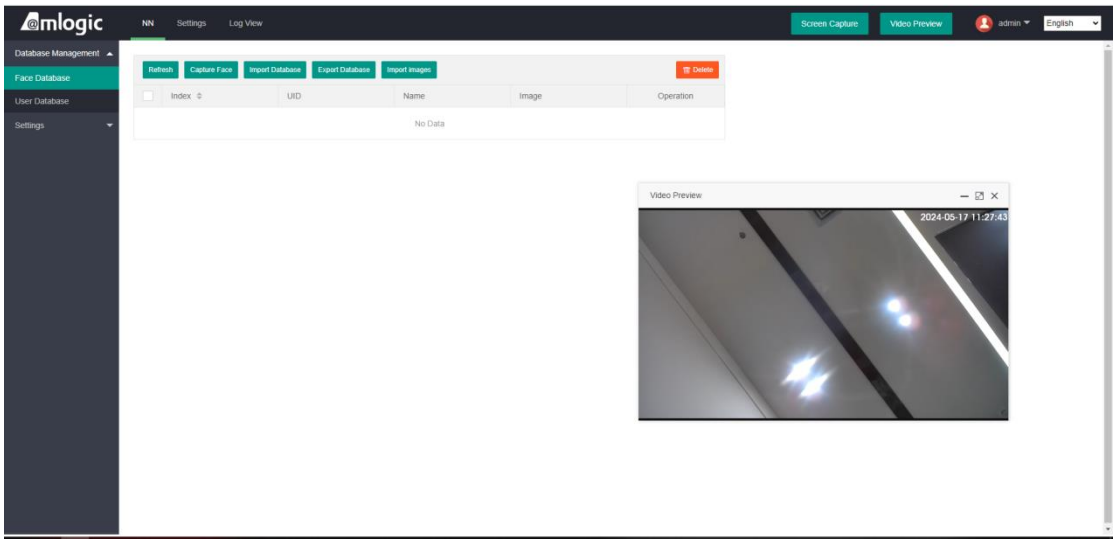
*Step 1*, connect the IMX307 module to the camera interface.

*Step 2*, camera test.

Enter while browsing:
http://192.168.xx.xx:8888/

account: admin
password: admin


Enter the main screen to preview the camera.




# 5.14 Speaker

Step 1: Plug in the speaker device.

Step 2, check the sound card device

```
# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: AMLAUGESOUND [AML-AUGESOUND], device 0: TDM-B-a1-acodec-hifi-alsaPORT-i2s fe004800.a1_codec-0 []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
#
#
#
```

Step 3, use the following command to play the audio.

# aplay -Dhw:0,0 test.wav

```
# aplay -Dhw:0,0 test.wav
[ 5145.733834@1] snd_tdm fe050000.audiobus:tdm@1: enable AED
Playing WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
[ 5145.735716@1] asoc-aml-card auge_sound: tdm playback enable
```