

EMT536 Buildroot User Manual

V1.0



Boardcon Embedded Design

Overview

This document applies only to the EMT536 development board. It is intended to help users quickly understand the hardware platform and to guide them in completing environment setup, source code compilation, firmware flashing, and functional testing of all hardware interfaces on the board.

System Support

Development Board	Debian11	Buildroot
PICOT536 V1 EMT536-V1	N	Y

Revision History

Version	Date	Author	Revision History
V1.0	2026-03-07	Liu Yuan	Initial version

Disclaimer

The information in this manual is provided for reference only. While Boardcon has made every effort to ensure the accuracy and reliability of the content, no guarantee is given regarding its completeness or correctness. All information in this manual is subject to change without prior notice.

Boardcon reserves the right to revise or update the contents of this manual at any time without prior notification. Boardcon shall not be held liable for any direct or indirect loss or damage arising from the use of this document or the products described herein.

Boardcon embedded design limited

2007-11 Haofang Tianji Plaza, 11008 Beihuan Avenue, Nanshan District,
Shenzhen, Guangdong, China. 518051

URL: www.armdesigner.com | www.boardcon.com

Email: market@armdesigner.com

Technical Support Inquiries: support@armdesigner.com

Tel: +86-755-26481393 | +86-755-27571591

Content

1. Introduction.....	5
1.1 Overview.....	5
1.2 Product Parameters	6
1.3 Hardware Interface Introduction.....	8
2. Install Drivers and Tool	10
2.1 Install CH9102X Driver.....	10
2.1.1 How to Connect the Serial Port Tool.....	10
2.1.2 Install Driver	11
2.2 Install Serial Terminal Tool.....	12
3. Development Environment.....	15
3.1 Preparing the Development Environment.....	15
3.2 Installing Libraries and Toolkits.....	15
4. Compile Source.....	16
5. Burn Image	20
6. Buildroot Test.....	23
6.1 Serial Terminal	23
6.2 Normal Display.....	24
6.3 USB Type-C	25
6.3.1 ADB.....	25
6.3.2 Type-C to USB2.0	26
6.4 USB Host.....	26
6.5 Ethernet.....	27
6.6 4G.....	29
6.7 Micro SD.....	31
6.8 Audio	31
6.8.1 Audio input	32

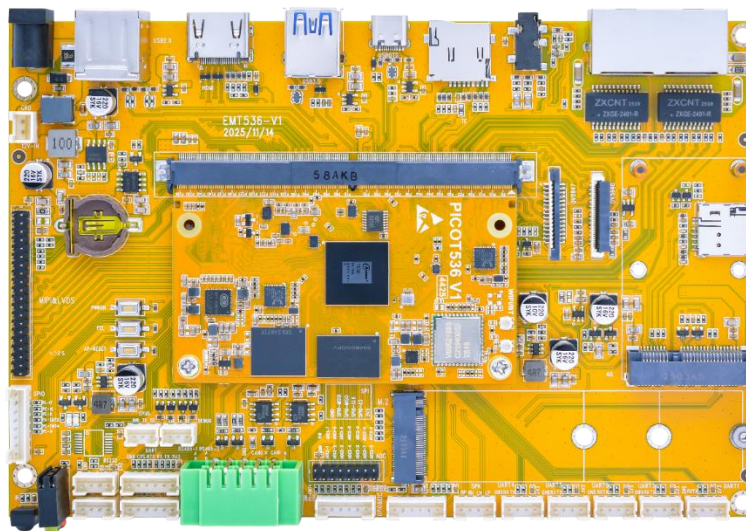
6.8.2 Audio output	32
6.9 WiFi & Bluetooth	33
6.9.1 WiFi	33
6.9.2 Bluetooth.....	35
6.10 IR	37
6.11 RS485.....	38
6.12 CAN.....	40
6.13 UART.....	42
5.14 RTC.....	43
6.15 SPI.....	44
6.16 M.2 SSD.....	44
6.17 GPADC.....	45

1. Introduction

1.1 Overview

EMT536 is a development board based on the Allwinner T536 series SoC, targeting industrial and intelligent hardware applications such as interactive terminals, smart manufacturing, and various industrial devices. The T536 integrates a quad-core Cortex™-A55 CPU and a single-core E907 RISC-V, providing scalable computing performance. It also features an on-chip NPU delivering up to 2 TOPS. With multiple heterogeneous expansion options and support for various OS architectures, it can meet system design requirements across different application scenarios.

In terms of interfaces and expansion, EMT536 supports display interfaces such as MIPI DSI and LVDS, and provides dual GMAC Ethernet ports as well as a USB3.1 Gen1 high-speed interface. The board also integrates 2×CAN, 2×RS485, and multiple UARTs and other common industrial peripheral interfaces, enabling easy on-site device connectivity and functional expansion.

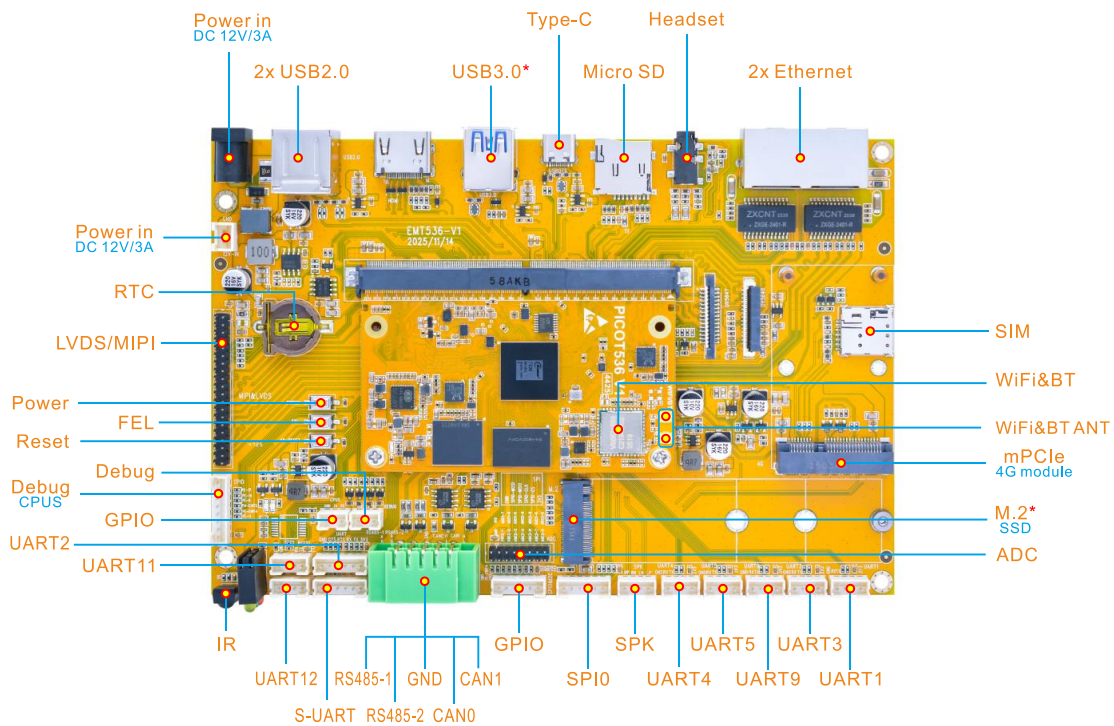


1.2 Product Parameters

Basic Parameters	
SOC	Allwinner T536
CPU	Quad-core ARM Cortex-A55, up to 1.6GHz
NPU	Up to 2 TOPS
Video	Decoder <ul style="list-style-type: none"> • Supports Motion JPEG, up to 3840 x 2160@15fps • Supports JPEG baseline , up to 1920x1080@60fps
	Encoder <ul style="list-style-type: none"> • Support H.264 BP/MP/HP, up to 3840x2160@25fps • Support J JPEG baseline , up to 3840x2160@15fps
RAM	2GB
ROM	8GB eMMC flash
Operating system	Buildroot
Hardware Parameters	
Extended Storage	<ul style="list-style-type: none"> • Support 1x MicroSD Card
Display	<ul style="list-style-type: none"> • Support 1x LVDS, up to 1920x1080@60fps display • Support 1x MIPI DSI up to 1920 x 1200@45fps display
Video input	<ul style="list-style-type: none"> • Support 1x Camera input
Audio	<ul style="list-style-type: none"> • Support 1x Speaker output • Support 1x Headset
USB	<ul style="list-style-type: none"> • Support 2x USB2.0 Host • Support 1x USB3.0 Host
Network	<ul style="list-style-type: none"> • Support 2x Gigabit Ethernet • Support 1x WiFi/BT module • Support 1x 4G module
Peripheral communication	<ul style="list-style-type: none"> • Support 2x RS485 • Support 2x CAN

	<ul style="list-style-type: none"> • Support 9x UART
Other parameters	Support Debug, RTC, IR, ADC, SPI, Power key , Reset key, FEL key
Electrical Parameters	
Power supply input voltage	12V/3A
RTC input voltage	3V/0.6uA
Operating temperature	0~70°
Storage temperature	-40~85°
Structural Parameters	
Core board dimensions	82.0mm x 50.0mm
Motherboard dimensions	120.0mm x 180.0mm

1.3 Hardware Interface Introduction



Interface parameters	
Power in DC 12V/3A	12V DC power input interface
2x USB2.0	Dual-stacked USB2.0 Host interface
USB3.0 (USB3.0/SSD)	USB3.0 Host interface
Type-C	USB Type-C (OTG download interface)
Micro SD	Micro SD slot
Headset	Audio output/input
2xEthernet	Gigabit Ethernet RJ45 interface
Camera0	Camera(ov5640) interface
SIM	SIM Card slot
WiFi&BT	WiFi&Bluetooth module (VS6621S80)
WiFi/BT ANT	WiFi&Bluetooth antenna
mPCIe 4G module	4G module interface

M.2 (USB3.0/SSD)	M.2 SSD interface
ADC	GPADC interface
UART1	UART1 interface
UART3	UART3 interface
UART9	UART9 interface
UART5	UART5 interface
UART4	UART4 interface
SPK	Speaker interface
SPI0	SPI0 interface
GPIO	GPIO interface
CAN1	CAN1 communication interface
CAN0	CAN0 communication interface
GND	Ground signal
RS485-2	RS485-2 communication interface
RS485-1	RS485-1 communication interface
UART16	UART3 interface
UART12	UART9 interface
IR	Infrared receiver
UART11	UART11 interface
UART2	UART2 interface
GPIO	GPIO interface
Debug CPUS	Debug serial port for E902
Debug	Debug serial port (default use)
Reset	Reset key
FEL	FEL key
Power	Power key

Dual-LVDS/Dual-MIPI	LVDS/MIPI display interface
RTC	RTC coin cell connector
Power in DC 12V	12V DC power input interface

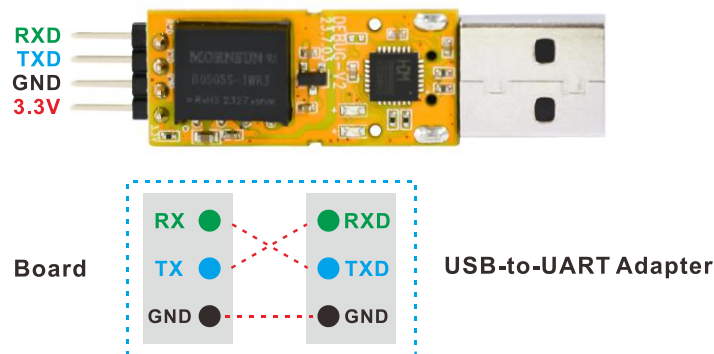
2. Install Drivers and Tool

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

Number	Driver name	Driver	Use
1	CH9102x	SETUP.EXE	Serial port debugging driver
2	Serial Terminal Tool	SecureCRT.exe	Debugging tool

2.1 Install CH9102X Driver

2.1.1 How to Connect the Serial Port Tool



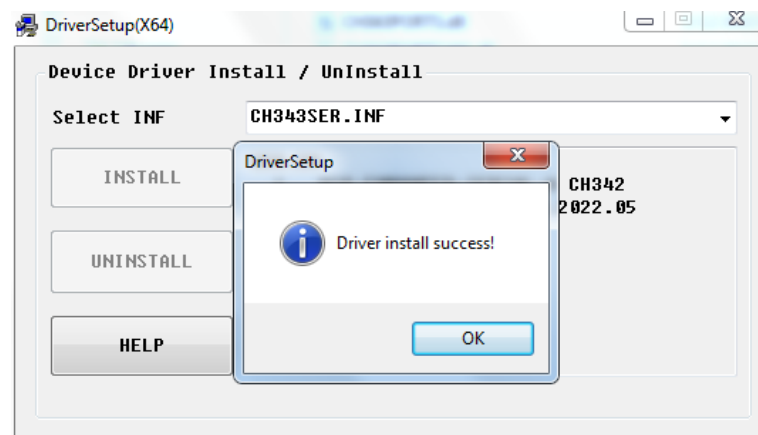
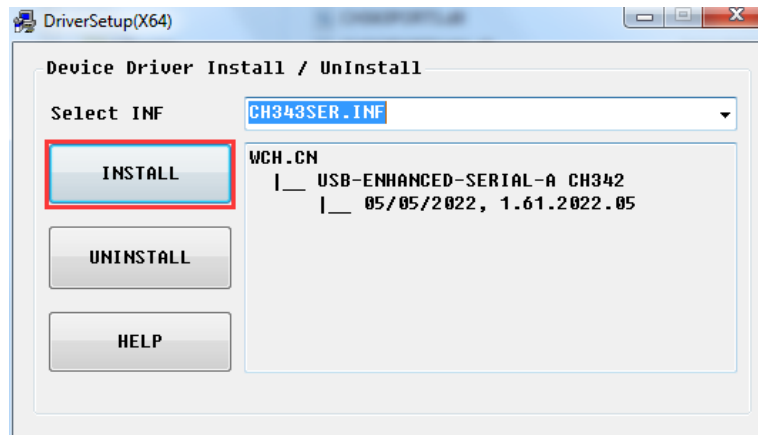
Pin	Connection Description
RXD	Receive, connect to TX pin of the board.
TXD	Transmit, connect to RX pin of the board.
GND	Ground, connect to GND pin of the board.
3V3	No need to connect.

2.1.2 Install Driver

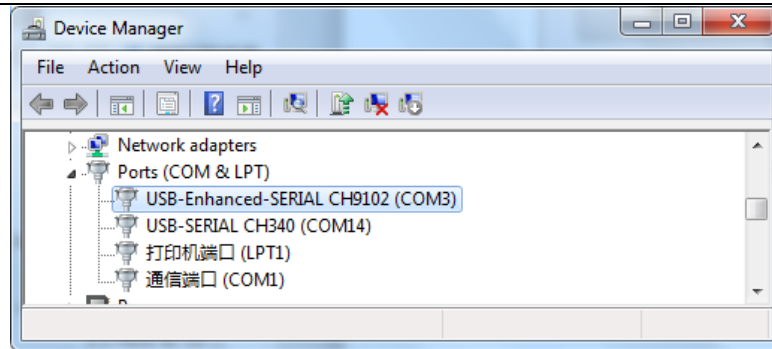
Step 1: Plug the CH9102X Module to the PC

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.

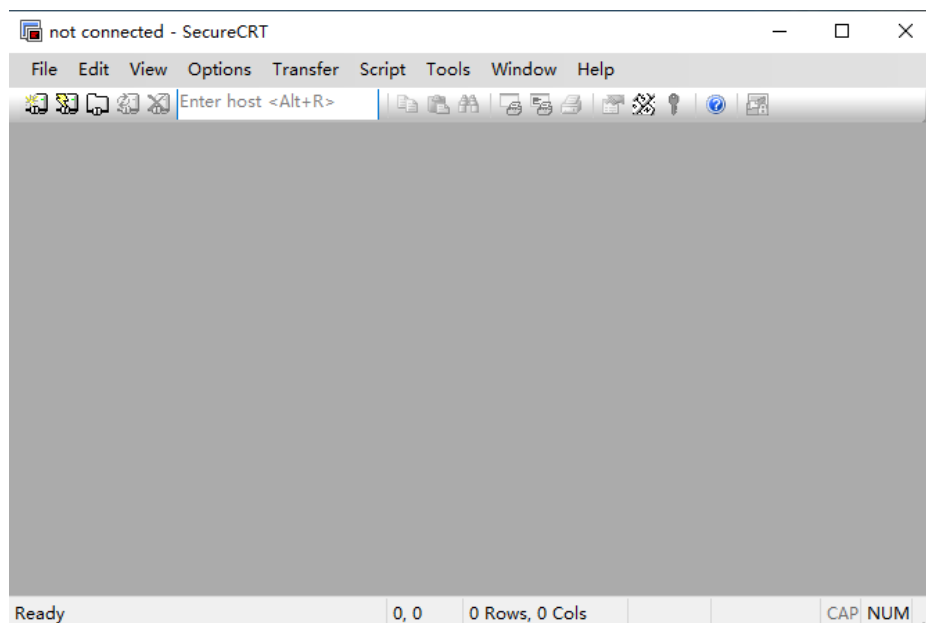


2.2 Install Serial Terminal Tool

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

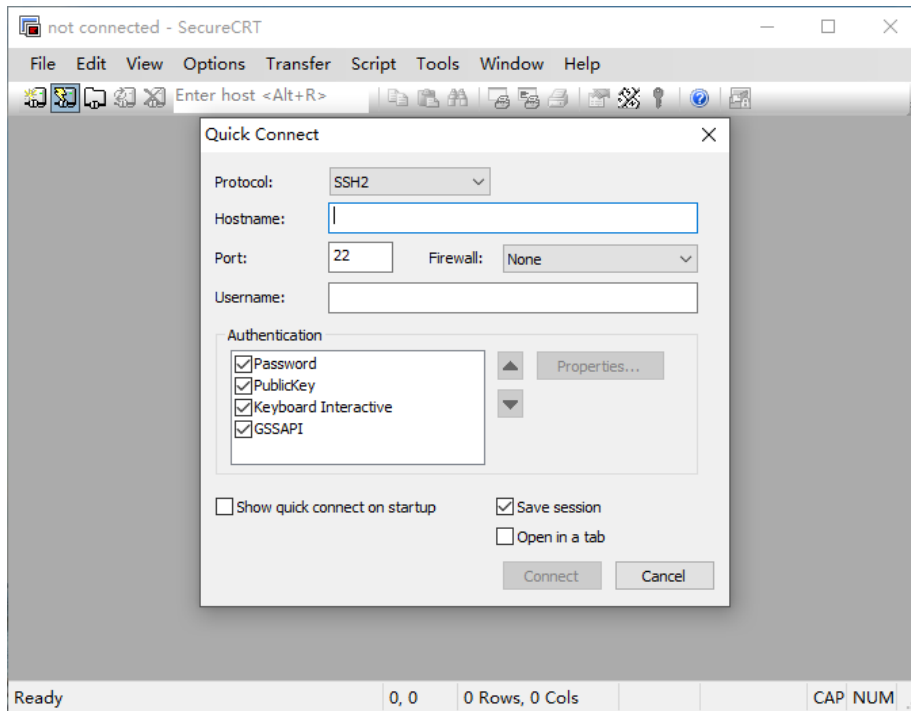
Step 1: Unzip *Platform/SecureCRT.rar* on PC.

Step 2: Click *SecureCRT/SecureCRT.exe* open the SecureCRT.

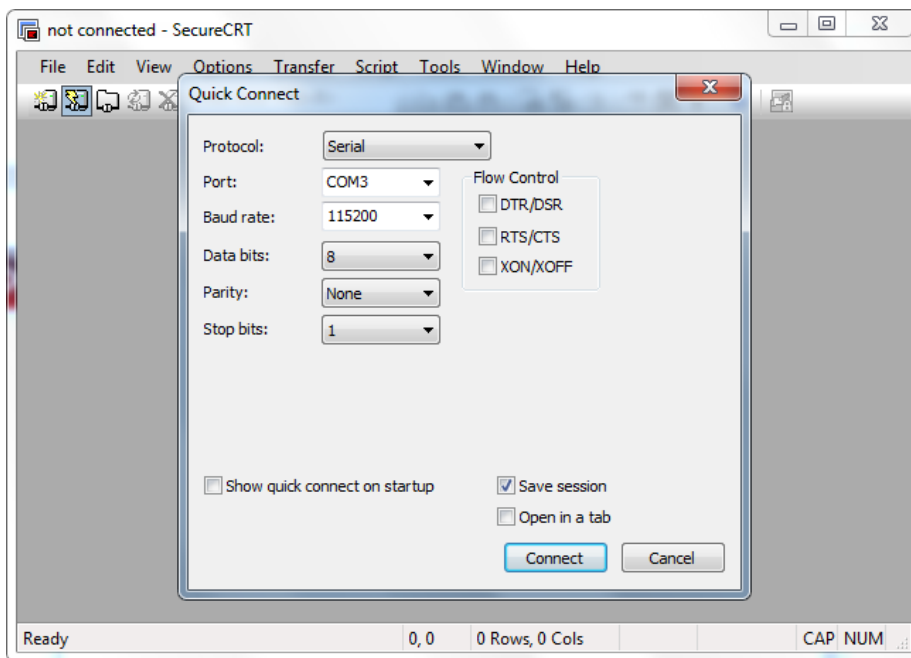


Step 3: Confirm the CH9102X driver has been installed and the CH9102X module is connecting to the PC.

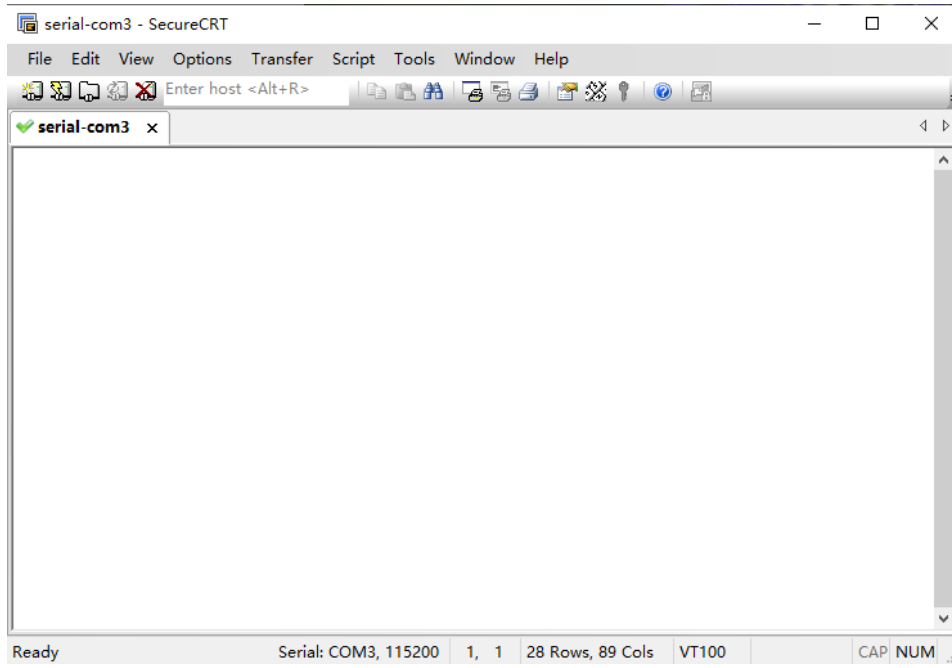
Step 4: Click the “**Quick Connect**” button to go to the Quick Connect configuration screen.



Step 5: Configure as shown in the following figure:



Step 6: After clicking “**Connect**” button, the terminal serial interface will be successfully accessed.



3. Development Environment

3.1 Preparing the Development Environment

It is recommended to use Ubuntu 22.04 or higher version for compilation. If you encounter an error during compilation, user 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 100G. If you do multiple builds, you will need more hard drive space.	Ubuntu 22.04

3.2 Installing Libraries and Toolkits

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	Network	Permission
Ubuntu 22.04	online	root

To install the required tools, execute the following commands:

```
$ sudo apt install -y adb bison build-essential cifs-utils ctags curl fastboot
$ sudo apt install -y flex gawk git g++-multilib jq libelf-dev libncurses5
$ sudo apt install -y libncurses5-dev libncurses-dev libncursesw5 libncursesw5-dev
$ sudo apt install -y libssl-dev libxml2-utils mysql-client-core-8.0 net-tools
$ sudo apt install -y openjdk-11-jre openssl openssh-server python repo sshfs vim
$ sudo apt install -y xrdp zlib1g zlib1g:i386
```

4. Compile Source

Step 1: Unzip the Source

To extract the source files, execute the following commands:

```
$ tar xvf T536_Tina5.0_AIOT *.tar.bz2
$ cd T536_Tina5.0_AIOT-V1.2/tina5.0-t536
```

Step 2: Configure the Compiled Board

To configure the board, execute:

```
$ ./build.sh config
```

In the “**All available board**” menu, the following board options are provided.

Since this board has two hardware multiplexing groups:

- (1) LVDS and MIPI DSI are multiplexed
- (2) PCIe (M.2 SSD) and USB 3.0 are multiplexed

the board configuration is divided into four valid combinations:

- **0. boardcon_lvds1_pcie:** LVDS0 + PCIE.
- **1. boardcon_lvds1_usb3:** LVDS0 + USB3.0.
- **2. boardcon_mipi0_pcie:** MIPI DSI + PCIE.

• 3. boardcon_mipi0_usb3: MIPI DSI + USB3.0.

```
liuyuan@boardcon:~/opt/EMT536/linux/T536_Tina5.0_AIOT-V1.2/tina5.0-t536$ ./build.sh config
03-03 19:52:26.239 4191338 D mkcommon : =====ACTION List: mk_config ;=====
03-03 19:52:26.241 4191338 D mkcommon : options :
All available platform:
  0. android
  1. linux
Choice [linux]: 1
All available linux_dev:
  0. bsp
  1. buildroot
Choice [buildroot]: 1
All available ic:
  0. t536
Choice [t536]: 0
All available board:
  0. boardcon_lvds1_pcie
  1. boardcon_lvds1_usb3
  2. boardcon_mipi0_pcie
  3. boardcon_mipi0_usb3
  4. demo
  5. demo_amp
  6. demo_kylo
  7. demo_nand
  8. demo_nor
  9. demo_raw_nand
Choice [boardcon_lvds1_pcie]: 3
All available flash:
  0. default
  1. nor
Choice [default]: 0
All available kern_name:
  0. linux-5.10-euler
  1. linux-5.10-origin
  2. linux-5.10-rt
  3. linux-5.10-xenomai
  4. linux-5.15-origin
Choice [linux-5.15-origin]: 4
03-03 19:52:41.193 4193019 D bsp : Setup BSP files
```

Touch Configuration

If LVDS (1280x800) is selected as the display, modify the touch calibration settings in the following file:

tina5.0-t536/platform/thirdparty/gui/lvgl-8/lv_examples/src/lv_drv_conf.h

Use the following configuration for **LVDS**:

```
#if EVDEV_CALIBRATE

//lvds (1280x800)
#if 1
#   define EVDEV_HOR_MIN          5           /*to invert axis swap EVDEV_XXX_MIN by
EVDEV_XXX_MAX*/
#   define EVDEV_HOR_MAX          1197        /*"evtest" Linux tool can help to get the
correct calibraion values>*/
#   define EVDEV_VER_MIN          11
#   define EVDEV_VER_MAX          1913
#endif

//mipi (800x1280)
#if 0
#   define EVDEV_HOR_MIN          7
#   define EVDEV_HOR_MAX          1912
#   define EVDEV_VER_MIN          4
#   define EVDEV_VER_MAX          1197
#endif
```

If MIPI DSI (800x1280) is selected as the display, modify the same file and use the following configuration for MIPI:

```
#if EVDEV_CALIBRATE

//lvds (1280x800)
#if 0
#   define EVDEV_HOR_MIN          5           /*to invert axis swap EVDEV_XXX_MIN by
EVDEV_XXX_MAX*/
#   define EVDEV_HOR_MAX          1197        /*"evtest" Linux tool can help to get the
correct calibraion values>*/
#   define EVDEV_VER_MIN          11
#   define EVDEV_VER_MAX          1913
#endif

//mipi (800x1280)
#if 1
#   define EVDEV_HOR_MIN          7
#   define EVDEV_HOR_MAX          1912
#   define EVDEV_VER_MIN          4
#   define EVDEV_VER_MAX          1197
#endif
```

Note: Only one calibration group should be enabled at a time.

Set **#if 1** for the selected display type and **#if 0** for the unused one.

Step 3: Compile

To start compile, execute the following command:

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

Step 4: Pack Image

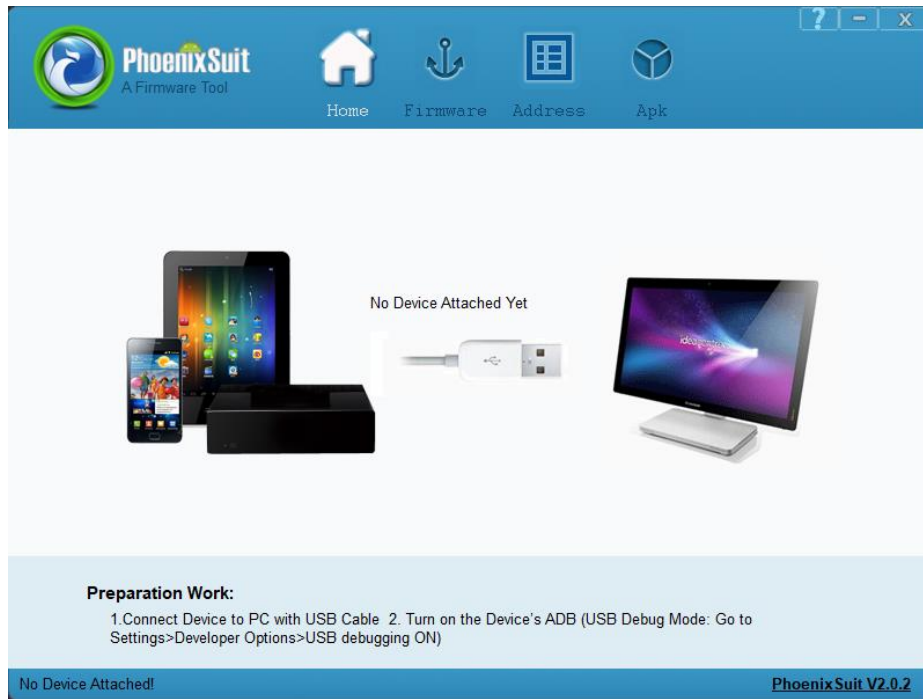
To pack image, execute the following command:

```
$ ./build.sh pack
```

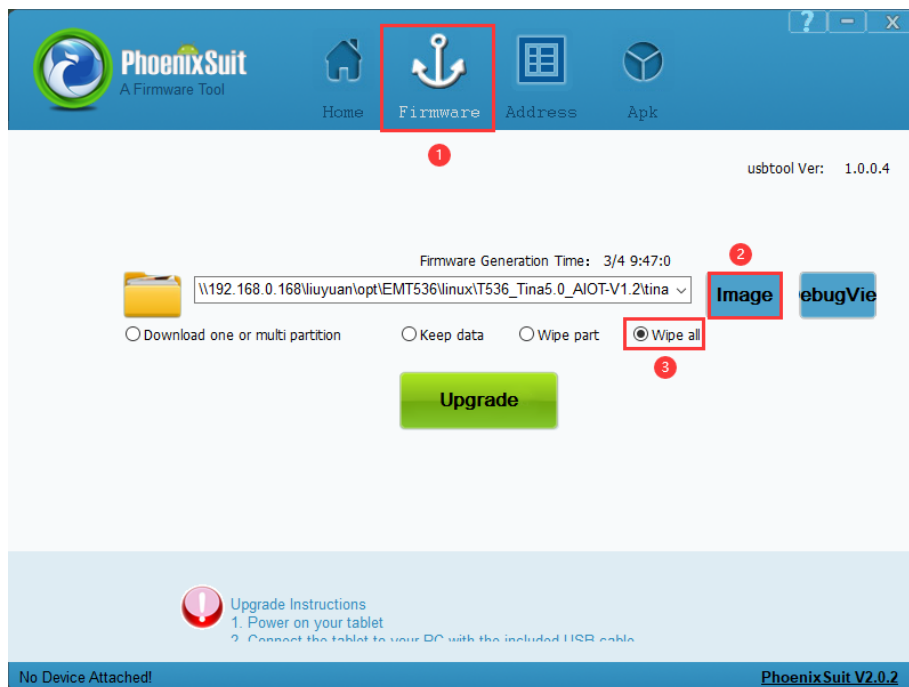
After compiling all, the image will be generated in *out/*.

5. Burn Image

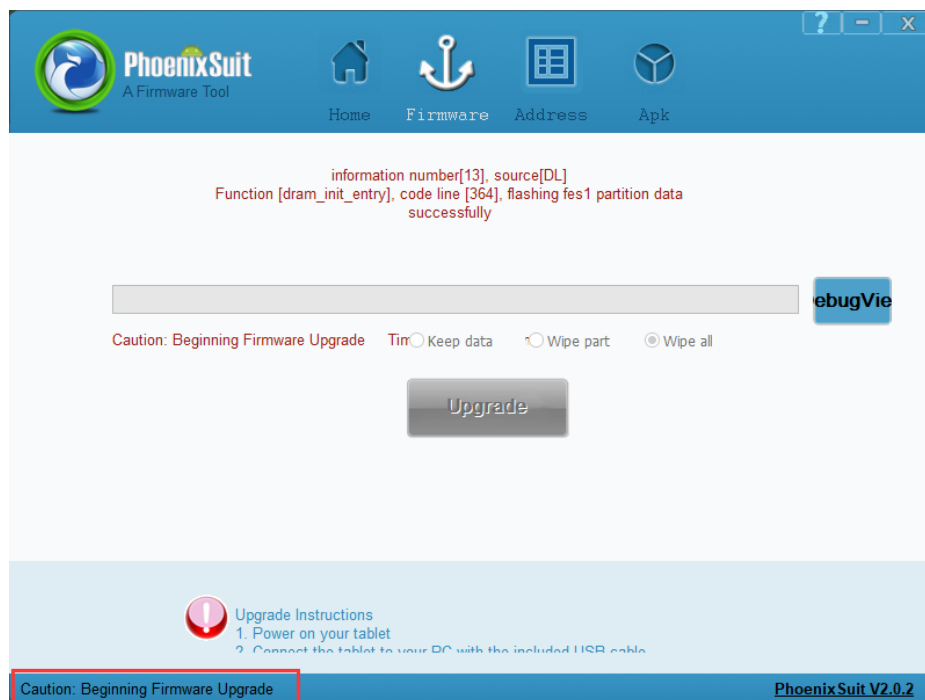
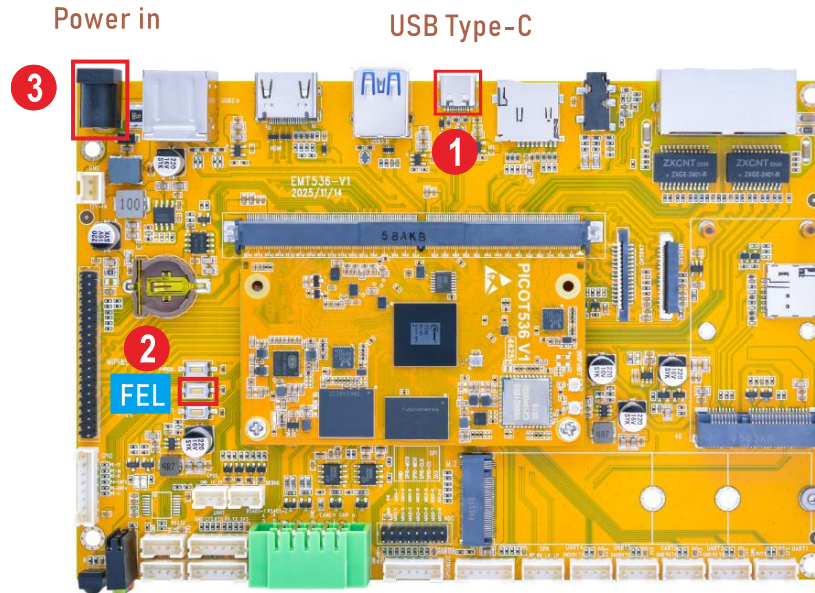
Step 1: Open *PhoenixSuit V2.0.2\PhoenixSuit.exe*.



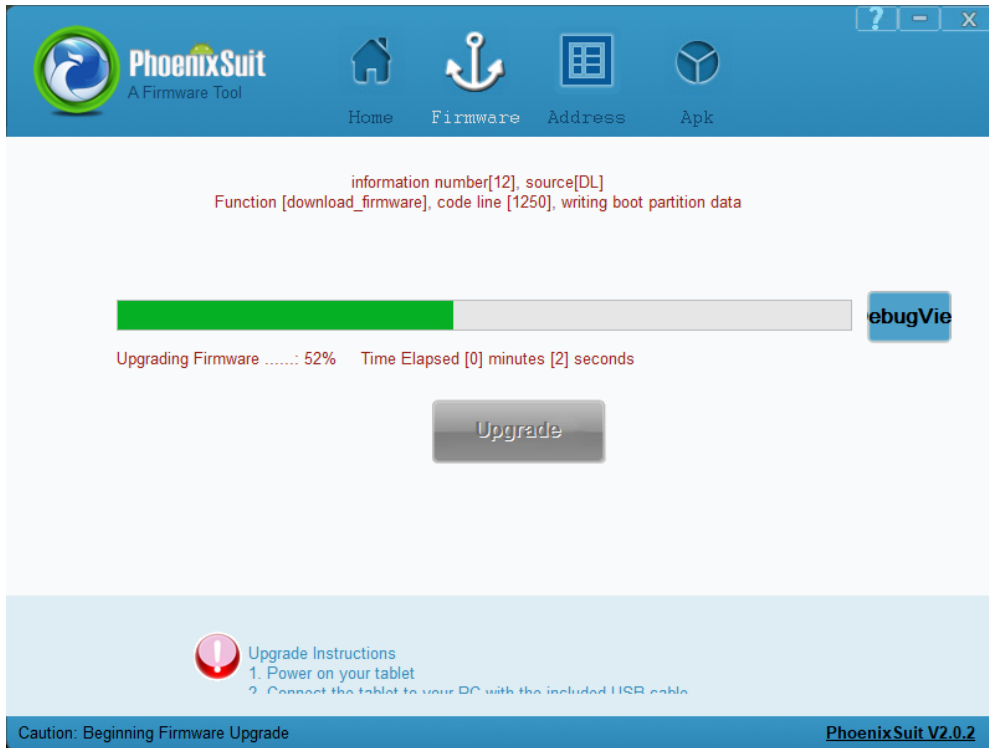
Step 2: Click **Firmware** -> **Image**, select **Image**, and choose **“Wipe all”** as the upgrade mode.



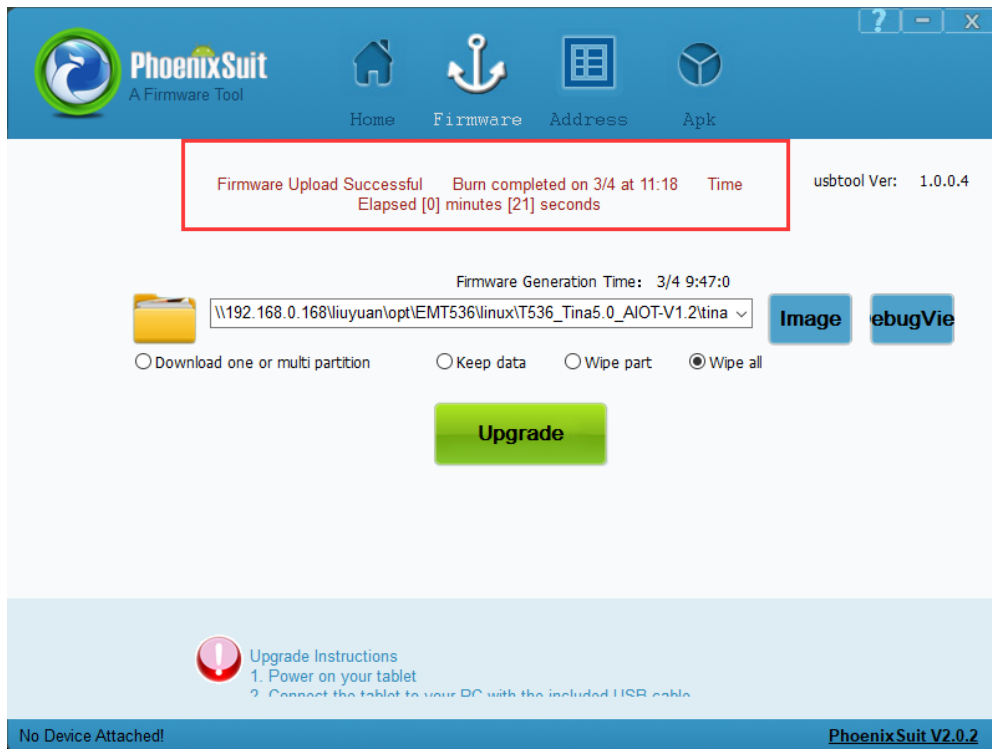
Step 3: Connect PC and development board with Type-C USB cable, keep pressing the **FEL Key**, Power on the board until the pop-up window appears as shown below, or the flashing tool displays **“Beginning Firmware Upgrade”**.



Step 4: Release the button, the download will start automatically.



Step 5: After the upgrade is complete, it will be displayed as follows.

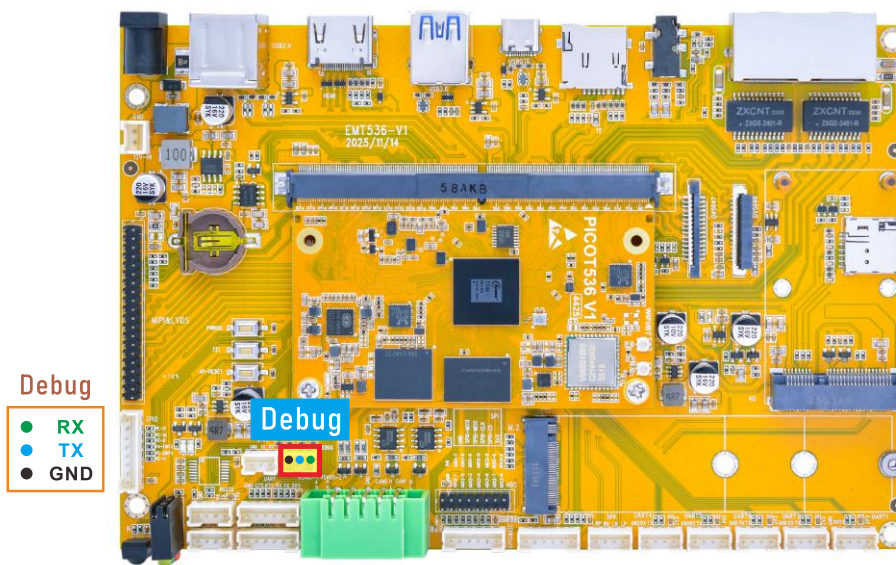


6. Buildroot Test

Longan login: root

6.1 Serial Terminal

Connect the board and PC with USB Serial cable, then power on, the terminal will output boot information. The default baudrate is 115200.



```

serial-com6 - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
serial-com6 x
[ 10.483688] SND_SOC_BIAS_STANDBY---es8323_set_bias_level:677
[ 10.490557] SND_SOC_BIAS_OFF---es8323_set_bias_level:690
[ 10.518938] dma dma0chan3: sunxi:dmaThe timeout func is not supported or chan->private is NULL, timeout
mode not used
[ 10.532874] SND_SOC_BIAS_STANDBY---es8323_set_bias_level:677
[ 10.539747] SND_SOC_BIAS_PREPARE---es8323_set_bias_level:657
[ 10.546343] SND_SOC_BIAS_ON---es8323_set_bias_level:653
[ 12.322166] random: crng init done
[ 12.325990] random: 9 urandom warning(s) missed due to ratelimiting
[ 12.417884] FAT-fs (mmcblk0p7): volume was not properly unmounted. Some data may be corrupt. Please run
fsck.
[ 12.450039] FAT-fs (mmcblk0p7): volume was not properly unmounted. Some data may be corrupt. Please run
fsck.
Trying to connect to swupdate...

Welcome to Allwinner Longan Platform
Longan login: [ 15.635970] SND_SOC_BIAS_PREPARE---es8323_set_bias_level:657
[ 15.642602] SND_SOC_BIAS_STANDBY---es8323_set_bias_level:677
[ 15.649503] SND_SOC_BIAS_OFF---es8323_set_bias_level:690
[ 25.203255] sunxi:sunxi_pd_test-soc@3000000:pd_vi_test@0:[WARN]: runtime_suspend disable clock
[ 25.212966] sunxi:sunxi_pd_test-soc@3000000:pd_ve_test@0:[WARN]: runtime_suspend disable clock
[ 25.222649] sunxi:sunxi_pd_test-soc@3000000:pd_npu_test@0:[WARN]: runtime_suspend disable clock
[ 25.232427] sunxi:sunxi_pd_test-soc@3000000:pd_serdes_test@0:[WARN]: runtime_suspend disable clock
[ 25.242491] sunxi:sunxi_pd_test-soc@3000000:pd_vo_test@0:[WARN]: runtime_suspend disable clock
[ 38.114199] axp1530-dcdc3: disabling
[ 38.118349] axp2202-aldol: disabling
[ 38.122477] axp2202-cldo4: disabling
[ 38.126726] axp2202-vmid: disabling

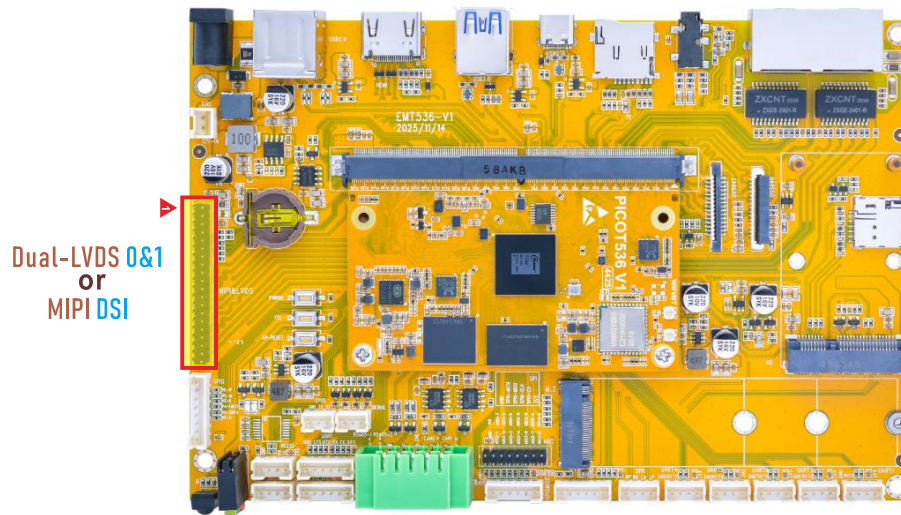
Welcome to Allwinner Longan Platform
Longan login: root
#
#
#
Ready Serial: COM6, 115200 34, 3 34 Rows, 105 Cols VT100 CAP NUM

```

6.2 Normal Display

EMT536 supports only one independent display output: either LVDS or MIPI DSI.

They are mutually exclusive and cannot be used simultaneously.

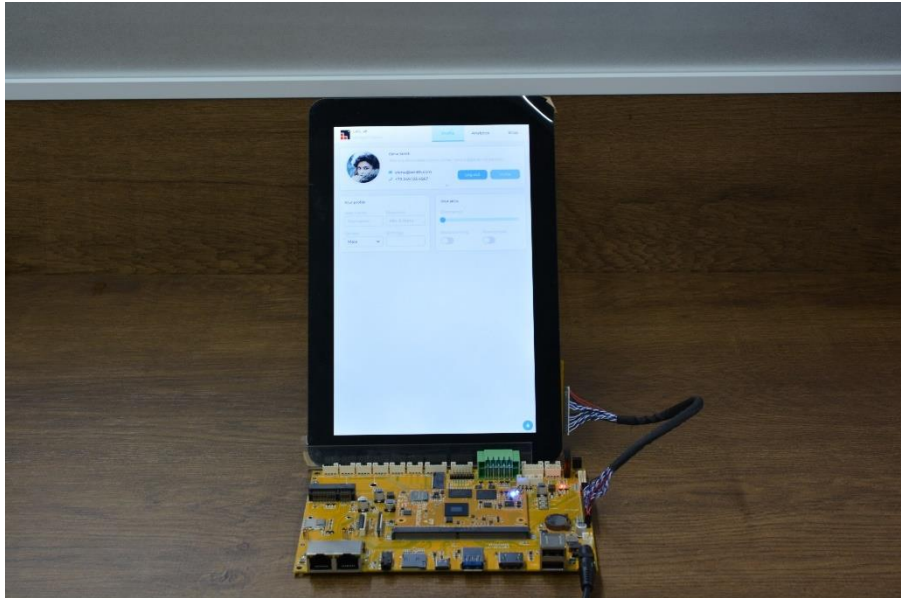


Display output combinations:

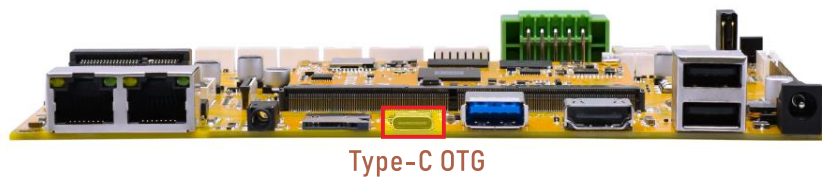
LVDS0:



MIPI DSI:



6.3 USB Type-C



The Type-C OTG port operates in Host mode by default.

6.3.1 ADB

To switch Host mode to Device mode user can execute the following command:

```
# echo usb_device >/sys/devices/platform/soc\@3000000/10.usbc0/otg_role
```

```
# echo usb_device >/sys/devices/platform/soc\@3000000/10.usbc0/otg_role  
[ 497.922645] sunxi_usb_udc 4100000.udc-controller: supply udc not found, using dummy regulator  
#
```

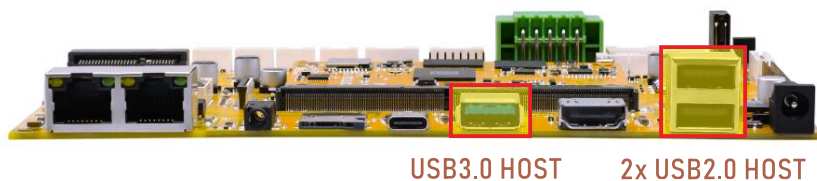
6.3.2 Type-C to USB2.0

By default, the system boots in Host mode. If need to switch from Device mode to Host mode, run the following command:

```
# echo usb_host > /sys/devices/platform/soc@3000000/10.usbc0/otg_role
```

```
# [10592.866457] sunxi-ehci 4101000.ehci0-controller: supply hci not found, using dummy regulator
[10592.877688] sunxi-ehci 4101000.ehci0-controller: EHCI Host Controller
[10592.884971] sunxi-ehci 4101000.ehci0-controller: new USB bus registered, assigned bus number 3
[10592.894742] sunxi-ehci 4101000.ehci0-controller: irq 138, io mem 0x04101000
[10592.918125] sunxi-ehci 4101000.ehci0-controller: USB 2.0 started, EHCI 1.00
[10592.925950] debugfs: Directory 'sunxi-ehci' with parent 'ehci' already present!
[10592.934267] usb usb3: New USB device found, idVendor=1d6b, idProduct=0002, bcdDevice= 5.15
[10592.943551] usb usb3: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[10592.951659] usb usb3: Product: EHCI Host Controller
[10592.957130] usb usb3: Manufacturer: Linux 5.15.147 ehci_hcd
[10592.963384] usb usb3: SerialNumber: sunxi-ehci
[10592.968776] hub 3-0:1.0: USB hub found
[10592.973059] hub 3-0:1.0: 1 port detected
[10592.978004] sunxi-ohci 4101400.ohci0-controller: supply hci not found, using dummy regulator
[10592.988144] sunxi-ohci 4101400.ohci0-controller: OHCI Host Controller
[10592.995444] sunxi-ohci 4101400.ohci0-controller: new USB bus registered, assigned bus number 4
[10593.005609] debugfs: Directory 'sunxi-ohci' with parent 'ohci' already present!
[10593.013867] sunxi-ohci 4101400.ohci0-controller: irq 139, io mem 0x04101400
[10593.082221] usb usb4: New USB device found, idVendor=1d6b, idProduct=0001, bcdDevice= 5.15
[10593.091504] usb usb4: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[10593.099610] usb usb4: Product: OHCI Host Controller
[10593.105078] usb usb4: Manufacturer: Linux 5.15.147 ohci_hcd
[10593.111331] usb usb4: SerialNumber: sunxi-ohci
[10593.116734] hub 4-0:1.0: USB hub found
[10593.120998] hub 4-0:1.0: 1 port detected
```

6.4 USB Host



Note: The USB 3.0 interface shares pins with PCIe, so only one of the two can be used at a time.

After connecting the USB flash drive, it will be automatically mounted. The user can identify whether the mounted flash drive is USB 2.0 or USB 3.0 from the debug log.

- USB2.0 print information in **high-speed** mode.

```
# [ 133.364130] usb 1-1.2: new high-speed USB device number 8 using xhci-hcd
[ 133.474989] usb 1-1.2: New USB device found, idVendor=0951, idProduct=1666, bcdDevice= 2.00
[ 133.484386] usb 1-1.2: New USB device strings: Mfr=2, Product=3, SerialNumber=4
[ 133.492693] usb 1-1.2: Product: DataTraveler 3.0
[ 133.497896] usb 1-1.2: Manufacturer: Kingston
[ 133.502787] usb 1-1.2: SerialNumber: E0D55EA573F8194168681327
[ 133.510092] usb-storage 1-1.2:1.0: USB Mass Storage device detected
[ 133.517534] scsi host1: usb-storage 1-1.2:1.0
[ 134.528500] scsi 1:0:0:0: Direct-Access Kingston DataTraveler 3.0 0000 PQ: 0 ANSI: 4
[ 134.538689] sd 1:0:0:0: [sdb] 121077760 512-byte logical blocks: (62.0 GB/57.7 GiB)
[ 134.547603] sd 1:0:0:0: [sdb] Write Protect is off
[ 134.553006] sd 1:0:0:0: [sdb] Mode Sense: 23 00 00 00
[ 134.559037] sd 1:0:0:0: [sdb] Write cache: disabled, read cache: enabled, doesn't support DPO or FUA
[ 134.574720] sdb: sdb1
[ 134.581499] sd 1:0:0:0: [sdb] Attached SCSI removable disk
create /dev/sdb
create /dev/sdb1
[ 134.617325] FAT-fs (sdb1): Volume was not properly unmounted. Some data may be corrupt. Please run
fsck.
```

- USB3.0 print information in **SuperSpeed** mode.

```
# [ 88.892508] usb 2-1: new SuperSpeed USB device number 2 using xhci-hcd
[ 88.925634] usb 2-1: New USB device found, idVendor=0dd8, idProduct=3b00, bcdDevice= 0.02
[ 88.934839] usb 2-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 88.942960] usb 2-1: Product: OnlyDisk
[ 88.947179] usb 2-1: Manufacturer: Netac
[ 88.951587] usb 2-1: SerialNumber: C0E8BFA3EC38F796
[ 88.958423] usb-storage 2-1:1.0: USB Mass Storage device detected
[ 88.965687] scsi host0: usb-storage 2-1:1.0
[ 90.228546] scsi 0:0:0:0: Direct-Access Netac OnlyDisk 8.01 PQ: 0 ANSI: 6
[ 90.238866] sd 0:0:0:0: [sda] 121610240 512-byte logical blocks: (62.3 GB/58.0 GiB)
[ 90.248066] sd 0:0:0:0: [sda] Write Protect is off
[ 90.253470] sd 0:0:0:0: [sda] Mode Sense: 23 00 00 00
[ 90.259306] sd 0:0:0:0: [sda] Write cache: disabled, read cache: enabled, doesn't support DPO or FUA
[ 90.274568] sda: sda1
[ 90.280883] sd 0:0:0:0: [sda] Attached SCSI removable disk
create /dev/sda
create /dev/sda1
[ 90.315359] FAT-fs (sda1): Volume was not properly unmounted. Some data may be corrupt. Please run
fsck.
```

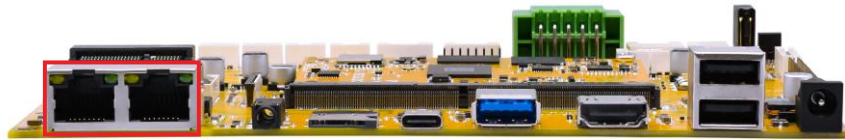
Execute the following command to view the path where the device is mounted:

```
# df -h
```

```
# df -h
Filesystem      Size      Used Available Use% Mounted on
/dev/root       990.7M    232.1M    742.5M   24% /
devtmpfs        980.0M    0          980.0M   0% /dev
tmpfs           982.5M    0          982.5M   0% /dev/shm
tmpfs           982.5M    168.0K    982.3M   0% /tmp
tmpfs           982.5M    196.0K    982.3M   0% /run
/dev/by-name/UDISK 6.1G      4.0K      6.1G     0% /mnt/UDISK
/dev/sda1       58.0G     33.1G     24.9G    57% /mnt/usb/sda1
/dev/sdb1       57.7G     10.9G     46.8G    19% /mnt/usb/sdb1
```

6.5 Ethernet

Step 1: Connect the network cable to the Ethernet port.



eth1 eth0

Ethernet

According to the log, it can be seen that the Gigabit Ethernet recognition is successful.

```
# [ 67.933951] dwmac-sunxi 4510000.ethernet eth1: Link is Up - 1Gbps/Full - flow control rx/tx
[ 67.943376] IPv6: ADDRCONF (NETDEV_CHANGE): eth1: link becomes ready
[ 73.885941] dwmac-sunxi 4500000.ethernet eth0: Link is Up - 1Gbps/Full - flow control rx/tx
[ 73.895363] IPv6: ADDRCONF (NETDEV_CHANGE): eth0: link becomes ready
```

Step 2: View network interface information.

```
# ifconfig
```

```
# ifconfig
eth0      Link encap:Ethernet  HWaddr AE:CE:1C:9A:8B:7B
          inet addr:192.168.0.122  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::e8d2:393:a1de:b293/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:521 errors:0 dropped:2 overruns:0 frame:0
          TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:49084 (47.9 KiB) TX bytes:1620 (1.5 KiB)
          Interrupt:155 Base address:0x8000

eth1      Link encap:Ethernet  HWaddr 42:7F:F3:4C:29:AE
          inet addr:192.168.0.158  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::f5b2:e2d3:7c47:aa7c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:596 errors:0 dropped:1 overruns:0 frame:0
          TX packets:22 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:55731 (54.4 KiB) TX bytes:2472 (2.4 KiB)
          Interrupt:160
```

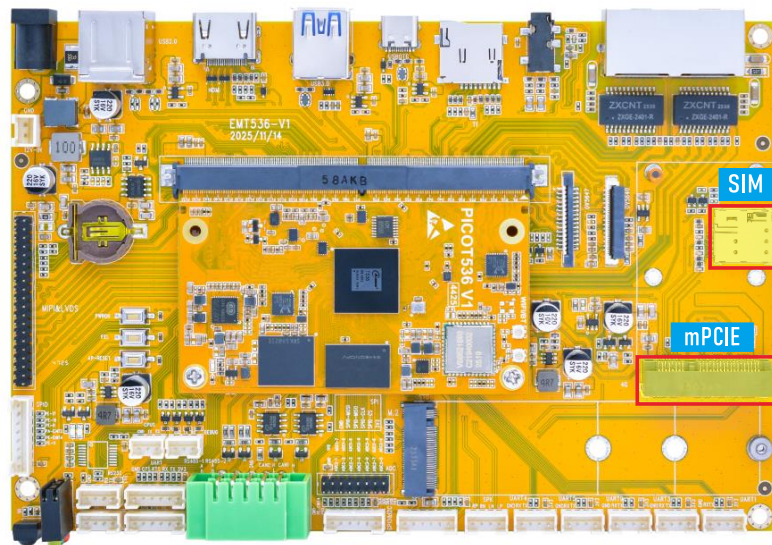
Step 3: Network connection test.

```
# ping -I eth0 www.armdesigner.com
# ping -I eth1 www.armdesigner.com
```

```
# ping -I eth0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=49 time=190.225 ms
64 bytes from 67.222.54.196: seq=1 ttl=49 time=2897.364 ms
64 bytes from 67.222.54.196: seq=2 ttl=49 time=1903.591 ms
64 bytes from 67.222.54.196: seq=3 ttl=49 time=916.991 ms
64 bytes from 67.222.54.196: seq=4 ttl=49 time=189.838 ms
64 bytes from 67.222.54.196: seq=5 ttl=49 time=189.515 ms
^C
--- www.armdesigner.com ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 189.515/1047.920/2897.364 ms
#
# ping -I eth1 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=49 time=185.158 ms
64 bytes from 67.222.54.196: seq=1 ttl=49 time=181.667 ms
64 bytes from 67.222.54.196: seq=2 ttl=49 time=180.991 ms
64 bytes from 67.222.54.196: seq=3 ttl=49 time=861.634 ms
64 bytes from 67.222.54.196: seq=4 ttl=49 time=817.352 ms
64 bytes from 67.222.54.196: seq=5 ttl=49 time=352.452 ms
^C
--- www.armdesigner.com ping statistics ---
7 packets transmitted, 6 packets received, 14% packet loss
round-trip min/avg/max = 180.991/429.875/861.634 ms
```

6.6 4G

Step 1: Insert 4G module to mPCIe socket (4G model: EC20).



Step 2: Connect antenna and insert SIM card.

Step 3: Initiate the PPP connection.

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

```
# pppd call quectel-ppp &
[1] 563
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)
noccps       # (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
send (AT^M)
expect (OK)
AT^M^M
OK
-- got it

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

Step 4: Check the status of the network interfaces.

```
# ifconfig
```

```
# ifconfig
inet addr:10.113.45.22 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:14 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:3
RX bytes:52 (52.0 B) TX bytes:198 (198.0 B)
```

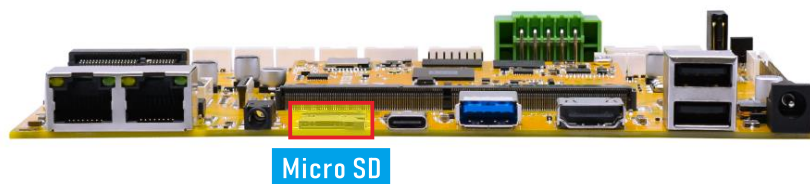
Step 5: Test the PPP connection.

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

```
# ping -I ppp0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=40 time=422.016 ms
64 bytes from 67.222.54.196: seq=1 ttl=40 time=288.065 ms
64 bytes from 67.222.54.196: seq=2 ttl=41 time=281.476 ms
64 bytes from 67.222.54.196: seq=3 ttl=41 time=272.951 ms
64 bytes from 67.222.54.196: seq=4 ttl=41 time=275.303 ms
64 bytes from 67.222.54.196: seq=5 ttl=41 time=275.213 ms
^C
--- www.armdesigner.com ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 272.951/302.504/422.016 ms
```

6.7 Micro SD

Step 1: Insert the micro SD card into the card slot.



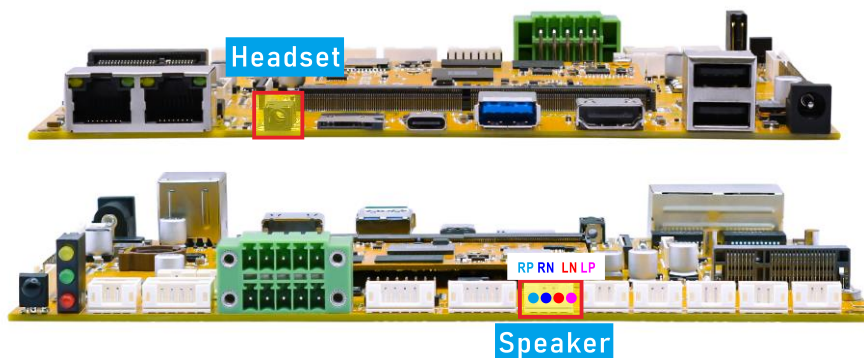
Step 2: The system will automatically mount it, view the device mount path.

```
# df -h
```

```
# df -h
Filesystem      Size      Used Available Use% Mounted on
/dev/root        990.7M    232.6M    742.1M    24% /
devtmpfs        980.0M    0          980.0M    0% /dev
tmpfs           982.5M    0          982.5M    0% /dev/shm
tmpfs           982.5M    168.0K    982.3M    0% /tmp
tmpfs           982.5M    252.0K    982.3M    0% /run
/dev/sda1       57.7G     10.9G     46.8G    19% /mnt/usb/sda1
/dev/mmcblk1p1  59.5G     16.4M     59.4G    0% /mnt/sdcard/mmcblk1p1
/dev/by-name/UDISK 6.1G      4.0K      6.1G    0% /mnt/UDISK
```

6.8 Audio

Step 1: Plug in the headset, and connect the speaker.



6.8.1 Audio input

- Recording

Execute the following command to start recording:

```
# arecord -Dhw:0,0 -f cd test.wav
```

```
# arecord -Dhw:0,0 -f cd test.wav
Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

6.8.2 Audio output

- Headset Output

Execute the following command to switch the audio output to the **headset** channel:

```
# amixer -c 0 cset name='SPK Switch' on
# amixer -c 0 cset name='OUT1 Switch' 1
```

```
# amixer -c 0 cset name='SPK Switch' on
numid=32,iface=MIXER,name='SPK Switch'
; type=BOOLEAN,access=rw-----,values=1
: values=on
#
# amixer -c 0 cset name='OUT1 Switch' 1
numid=44,iface=MIXER,name='OUT1 Switch'
; type=BOOLEAN,access=rw-----,values=1
: values=on
```

- Speaker Output

Execute the following command to switch the audio output to the **speaker** channel:

```
# amixer -c 0 cset name='SPK Switch' on
# amixer -c 0 cset name='OUT2 Switch' 1
```

```
# amixer -c 0 cset name='SPK Switch' on
numid=32,iface=MIXER,name='SPK Switch'
; type=BOOLEAN,access=rw-----,values=1
: values=on
#
# amixer -c 0 cset name='OUT2 Switch' 1
numid=45,iface=MIXER,name='OUT2 Switch'
; type=BOOLEAN,access=rw-----,values=1
: values=on
```

- Play Audio

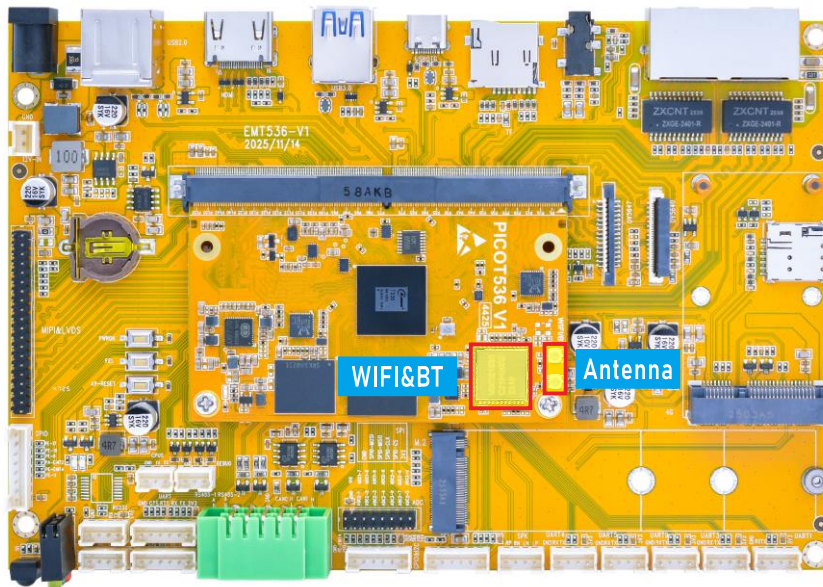
Execute the following command to play audio:

```
# aplay -Dhw:0,0 test.wav
```

```
# aplay -Dhw:0,0 test.wav  
Playing WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

6.9 WiFi & Bluetooth

To use Wi-Fi and Bluetooth functions properly, the antenna needs to be connected.



6.9.1 WiFi

Step 1: View the device information.

```
# ifconfig
```

```
# ifconfig  
wlan0  Link encap:Ethernet HWaddr 60:48:9C:42:0F:D8  
UP BROADCAST MULTICAST MTU:1500 Metric:1  
RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
```

Step 2: Scan for available WiFi hotspots.

```
# iw wlan0 scan
```

```
# iw wlan0 scan
[ 760.905422] [chip1][SKWIFI6621S DBG] skw_scan: STA: chip: 1, nr_chan: 39, n_ssids: 1, ie_len: 0
[ 764.901523] [chip1][SKWIFI6621S DBG] skw_scan_done: inst: 0, aborted: 0, scan result: 89
BSS 38:20:28:50:0f:81(on wlan0)
  TSF: 760699739 usec (0d, 00:12:40)
  freq: 2412
  beacon interval: 100 TUs
  capability: ESS Privacy ShortPreamble ShortSlotTime RadioMeasure (0x1431)
  signal: -73.00 dBm
  last seen: 3960 ms ago
  Information elements from Probe Response frame:
  SSID: Casvi
  Supported rates: 1.0* 2.0* 5.5 11.0 6.0 9.0 12.0 18.0
  DS Parameter set: channel 1
  Country: CN Environment: Indoor only
    Channels [1 - 13] @ 20 dBm
  ERP: <no flags>
  Extended supported rates: 24.0 36.0 48.0 54.0
  RSN:
    * Version: 1
    * Group cipher: CCMP
    * Pairwise ciphers: CCMP
    * Authentication suites: PSK
    * Capabilities: 1-PTKSA-RC 1-GTKSA-RC (0x0000)
  RM enabled capabilities:
```

Step 3: Connect to the hotspot.

```
# wifi-connect.sh SSID PSK
```

```
# wifi-connect.sh Boardcon Boardcon43435656
connect to WiFi ssid: Boardcon, Passwd: Boardcon43435656
killall: wpa_supplicant: no process killed
Successfully initialized wpa supplicant
[ 866.003035] [chip1][SKWIFI6621S DBG] skw_mgmt_frame_register: STA add filter ACTION
[ 866.072550] [chip1][SKWIFI6621S DBG] skw_add_virtual_intf: p2p-dev-wlan0(P2P_DEVICE), mac:
00:00:00:00:00:00
[ 866.083636] [chip1][SKWIFI6621S INFO] skw_add_iface: P2P_DEVICE, inst: 3, mac: 00:00:00:00:00:00,
bitmap: 0x9
[ 866.114026] [chip1][SKWIFI6621S DBG] skw_mgmt_frame_register: P2P_DEVICE add filter ACTION
[ 866.125597] [chip1][SKWIFI6621S DBG] skw_start_p2p_device: traced
[ 866.135350] sunxi-can 4546800.can: PM runtime resume.
# [ 866.141439] sunxi-can 4546800.can: PM runtime suspend.
[ 866.148304] [chip1][SKWIFI6621S DBG] skw_scan: STA: chip: 1, nr_chan: 39, n_ssids: 1, ie_len: 139
[ 869.274209] [chip1][SKWIFI6621S DBG] skw_scan_done: inst: 0, aborted: 0, scan result: 45
[ 869.283974] [chip1][SKWIFI6621S DBG] skw_dump_survey: last idx: 0
[ 869.292622] [chip1][SKWIFI6621S INFO] skw_auth: wlan0, bssid: b4:f1:8c:6d:d1:24, auth type: 0, state:
NONE
[ 869.303491] [chip1][SKWIFI6621S DBG] skw_abort_scan: inst: 0, scanning: 0
[ 869.311045] [chip1][SKWIFI6621S DBG] skw_parse_center_chn: band: 0, ppe: 0xe42407ff, phy_cap_info[0]:
0x2
```

Step 4: View the network interface status.

```
# ifconfig
```

```
# ifconfig
wlan0 Link encap:Ethernet HWaddr 60:48:9C:42:0F:D8
inet addr:192.168.0.103 Bcast:192.168.0.255 Mask:255.255.255.0
inet6 addr: fe80::a225:b17c:7c2a:8ab5/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:200 errors:0 dropped:9 overruns:0 frame:0
TX packets:23 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:18582 (18.1 KiB) TX bytes:1578 (1.5 KiB)
```

Step 5: Test the WiFi network.

```
# ping -I wlan0 www.armdesigner.com
```

```
# ping -I wlan0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=49 time=203.025 ms
64 bytes from 67.222.54.196: seq=1 ttl=49 time=213.264 ms
64 bytes from 67.222.54.196: seq=2 ttl=49 time=186.845 ms
64 bytes from 67.222.54.196: seq=3 ttl=49 time=240.466 ms
64 bytes from 67.222.54.196: seq=4 ttl=49 time=196.769 ms
64 bytes from 67.222.54.196: seq=5 ttl=49 time=240.842 ms
64 bytes from 67.222.54.196: seq=6 ttl=49 time=196.849 ms
64 bytes from 67.222.54.196: seq=7 ttl=49 time=216.692 ms
^C
--- www.armdesigner.com ping statistics ---
8 packets transmitted, 8 packets received, 0% packet loss
round-trip min/avg/max = 186.845/211.844/240.842 ms
```

6.9.2 Bluetooth

The audio output channel is disabled by default. When the device is used as a Bluetooth speaker, the audio output channel needs to be manually enabled.

```
# amixer -c 0 cset name='SPK Switch' on
# amixer -c 0 cset name='OUT2 Switch' 1
```

Step 1: View the Bluetooth device status.

```
# hciconfig -a
```

```
# hciconfig -a
hci0: Type: Primary Bus: SDIO
BD Address: 60:48:9C:42:0F:D9 ACL MTU: 1021:6 SCO MTU: 255:4
UP RUNNING
RX bytes:0 acl:0 sco:0 events:61 errors:0
TX bytes:2352 acl:0 sco:0 commands:63 errors:0
Features: 0xbf 0xee 0x0d 0xfe 0xdb 0xfd 0x7b 0x87
Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV3
Link policy: RSWITCH SNIFF
Link mode: SLAVE ACCEPT
Name: 'BlueZ 5.54'
Class: 0x0c0400
Service Classes: Rendering, Capturing
Device Class: Audio/Video, Uncategorized
HCI Version: (0xc) Revision: 0x5302
LMP Version: (0xc) Subversion: 0xd
Manufacturer: not assigned (3149)
```

Step 3: Control and configure the Bluetooth device.

```
# bluetoothctl
```

```
# bluetoothctl
Agent registered
[CHG] Controller 60:48:9C:42:0F:D9 Pairable: yes
[bluetooth]#
```

Step 4: Scan for nearby Bluetooth devices.

(After scanning, turn off scanning to prevent continuous searching and refreshing.)

```
[bluetooth]# scan on          # Enable scanning
[bluetooth]# scan off        # Disable scanning
[bluetooth]# devices          # List devices
```

```
[bluetooth]# scan on
...
[bluetooth]# scan off
...
[bluetooth]# devices
Device C1:14:88:1E:63:B5 Mi Smart Band 6
Device 52:8A:F6:CA:D3:EB 52-8A-F6-CA-D3-EB
Device 5D:A3:1C:A4:48:6F 5D-A3-1C-A4-48-6F
Device 50:8A:06:FA:35:A5 TY
Device 6A:8C:DF:9D:FC:67 NGMolwBu/A+AXAZtJ0jjFQi5Y
Device 7E:CB:C0:DA:29:27 7E-CB-C0-DA-29-27
Device 43:74:BF:F2:FB:57 43-74-BF-F2-FB-57
Device 80:B6:55:03:AB:64 80-B6-55-03-AB-64
Device 61:96:E1:96:60:3C MPNXzf0IR5tpjCpD6B0pcl4Mo
Device 4E:CD:BB:B9:13:36 4E-CD-BB-B9-13-36
Device 7E:06:A7:25:31:3C NPoHej0iBURNVBokM2TsFGPyw
Device 6F:5F:80:53:FE:3C 6F-5F-80-53-FE-3C
Device 41:61:6F:B5:96:ED 41-61-6F-B5-96-ED
Device 55:B7:7E:0E:95:99 NGOWhsn5+LBN2L142t3pTKFXo
Device EA:F2:AF:23:CF:7C KD-89524
Device 72:9C:F5:BB:95:78 72-9C-F5-BB-95-78
Device 6D:ED:49:16:B6:9C 6D-ED-49-16-B6-9C
Device CB:4E:FD:D9:23:C2 F44EFDD923C2
Device 4A:FB:03:BB:C1:05 4A-FB-03-BB-C1-05
Device 7E:5A:B1:F2:B2:92 NGM+HXdaFSp72XW6hz1ht7/H4
Device 70:06:81:7A:3A:24 70-06-81-7A-3A-24
Device A8:35:12:9A:EB:4D liuy
```

Step 5: Pair the device.

```
[bluetooth]# pair A8:35:12:9A:EB:4D
```

```
[bluetooth]# pair A8:35:12:9A:EB:4D
Attempting to pair with A8:35:12:9A:EB:4D
[CHG] Device A8:35:12:9A:EB:4D Connected: yes
Request confirmation
[agent] Confirm passkey 826524 (yes/no): yes
[CHG] Device A8:35:12:9A:EB:4D Modalias: bluetooth:v010Fp107Ed1436
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001105-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000110a-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001112-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000111f-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000112f-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001132-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001200-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001800-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001801-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000fe35-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 11c8b310-80e4-4276-afc0-f81590b2177f
[CHG] Device A8:35:12:9A:EB:4D UUIs: 8ce255c0-200a-11e0-ac64-0800200c9a66
[CHG] Device A8:35:12:9A:EB:4D UUIs: 9664aa26-d76c-43ad-9775-d310f253a408
[CHG] Device A8:35:12:9A:EB:4D UUIs: c574edb3-6551-473b-bad8-65035f07e4a8
[CHG] Device A8:35:12:9A:EB:4D ServicesResolved: yes
[CHG] Device A8:35:12:9A:EB:4D Paired: yes
Pairing successful
[CHG] Device A8:35:12:9A:EB:4D ServicesResolved: no
[CHG] Device A8:35:12:9A:EB:4D Connected: no
```

Step 6: Connect the device.

```
[bluetooth]# connect A8:35:12:9A:EB:4D
```

```
[bluetooth]# connect A8:35:12:9A:EB:4D
Attempting to connect to A8:35:12:9A:EB:4D
[CHG] Device A8:35:12:9A:EB:4D Connected: yes
Connection successful
[CHG] Device A8:35:12:9A:EB:4D ServicesResolved: yes
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001105-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001108-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000110a-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001112-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000111f-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000112f-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001132-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001200-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001800-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 00001801-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 0000fe35-0000-1000-8000-00805f9b34fb
[CHG] Device A8:35:12:9A:EB:4D UUIs: 11c8b310-80e4-4276-afc0-f81590b2177f
[CHG] Device A8:35:12:9A:EB:4D UUIs: 8ce255c0-200a-11e0-ac64-0800200c9a66
[CHG] Device A8:35:12:9A:EB:4D UUIs: 9664aa26-d76c-43ad-9775-d310f253a408
[CHG] Device A8:35:12:9A:EB:4D UUIs: c574edb3-6551-473b-bad8-65035f07e4a8
Authorize service
[agent] Authorize service 00001108-0000-1000-8000-00805f9b34fb (yes/no): yes
#
```

6.10 IR



IR

Execute the following command to retrieve the data reported by the IR module:

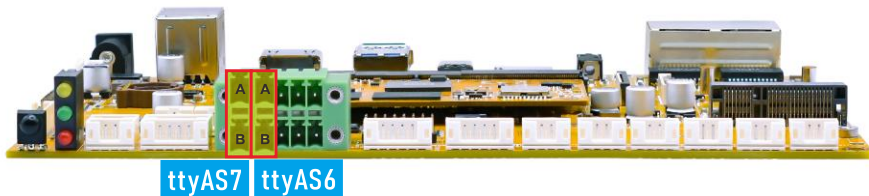
```
# hexdump /dev/input/event0
```

```
# hexdump /dev/input/event0
00000000 03c3 0000 0000 0000 a27c 000b 0000 0000
00000010 0004 0004 1864 0018 03c3 0000 0000 0000
00000020 a27c 000b 0000 0000 0000 0000 0000 0000
00000030 03c3 0000 0000 0000 90db 000c 0000 0000
00000040 0004 0004 1864 0018 03c3 0000 0000 0000
00000050 90db 000c 0000 0000 0000 0000 0000 0000
00000060 03c5 0000 0000 0000 fd1a 0005 0000 0000
00000070 0004 0004 1867 0018 03c5 0000 0000 0000
00000080 fd1a 0005 0000 0000 0000 0000 0000 0000
00000090 03c5 0000 0000 0000 eb83 0006 0000 0000
000000a0 0004 0004 1867 0018 03c5 0000 0000 0000
000000b0 eb83 0006 0000 0000 0000 0000 0000 0000
000000c0 03c6 0000 0000 0000 b34b 000a 0000 0000
000000d0 0004 0004 1866 0018 03c6 0000 0000 0000
000000e0 b34b 000a 0000 0000 0000 0000 0000 0000
000000f0 03c6 0000 0000 0000 a1b8 000b 0000 0000
00001000 0004 0004 1866 0018 03c6 0000 0000 0000
00001100 a1b8 000b 0000 0000 0000 0000 0000 0000
00001200 03c7 0000 0000 0000 dae2 0009 0000 0000
00001300 0004 0004 1865 0018 03c7 0000 0000 0000
00001400 dae2 0009 0000 0000 0000 0000 0000 0000
00001500 03c7 0000 0000 0000 c6f8 000a 0000 0000
00001600 0004 0004 1865 0018 03c7 0000 0000 0000
00001700 c6f8 000a 0000 0000 0000 0000 0000 0000
```

View the event node for the input device corresponding to the IR-RX module:

```
# cat /proc/bus/input/devices
```

6.11 RS485

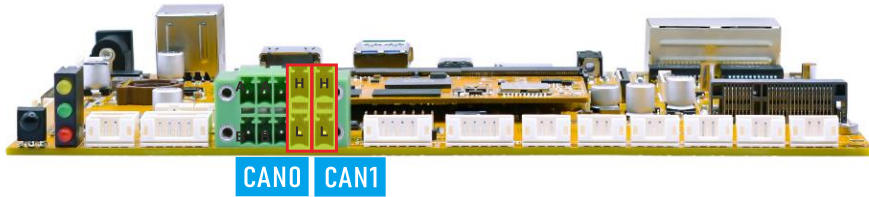


Step 1: Connect the RS485 test tool to the board as shown in the diagram below.



Step 2: Open the corresponding serial terminals, setting the baud rate to 115200.

6.12 CAN



CAN0 corresponds to device node can0, and CAN1 corresponds to device node can1.

The following test procedure uses CAN0 as an example. To test CAN1, simply replace the device node with can1.

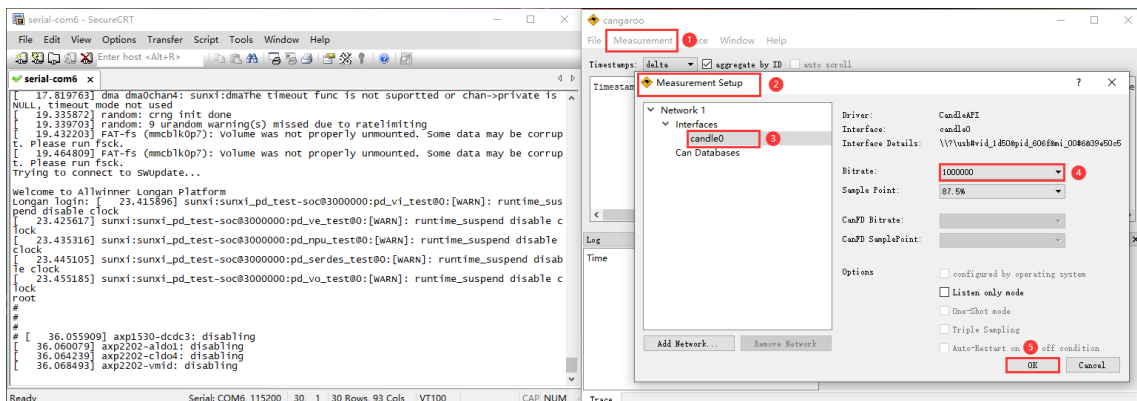
```
# ifconfig -a
can0    Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
        NOARP MTU:16 Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:10
        RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
        Interrupt:175

can1    Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
        NOARP MTU:16 Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:10
        RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
        Interrupt:178
```

Step 1: Connect the CAN test tool to the board as shown in the diagram below.

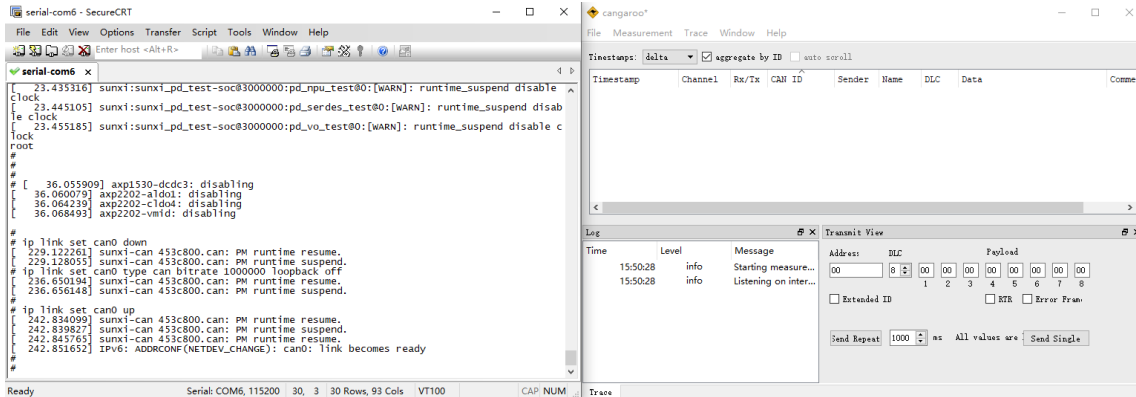


Step 2: Open the CAN test software and set the baud rate to 1000000.



Step 3: Configure and activate the CAN network, setting the bitrate to 1000000.

```
# ip link set can0 down
# ip link set can0 type can bitrate 1000000 loopback off
# ip link set can0 up
```



The screenshot shows the SecureCRT terminal window with the following commands and output:

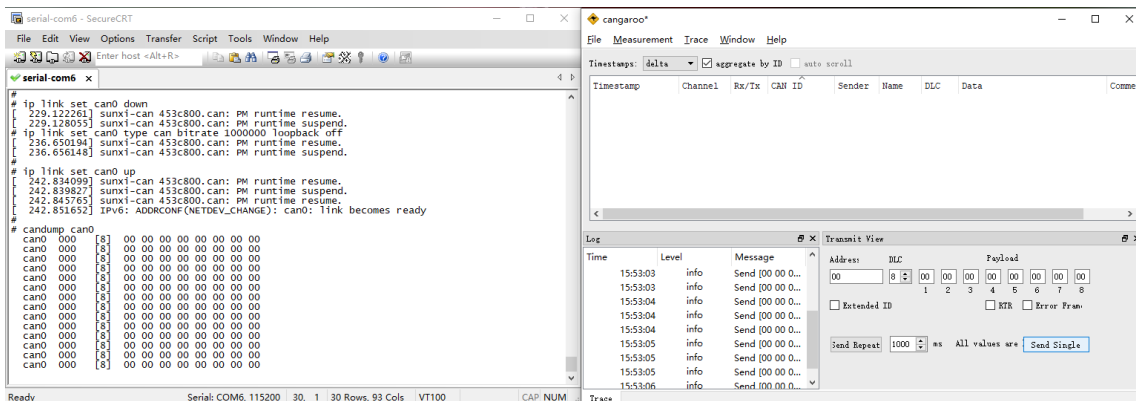
```
# ip link set can0 down
[ 229.122261] sunxi-can 453c800.can: PM runtime resume.
[ 229.128055] sunxi-can 453c800.can: PM runtime suspend.
# ip link set can0 type can bitrate 1000000 loopback off
[ 236.650194] sunxi-can 453c800.can: PM runtime resume.
[ 236.656148] sunxi-can 453c800.can: PM runtime suspend.
# ip link set can0 up
[ 242.834099] sunxi-can 453c800.can: PM runtime resume.
[ 242.839827] sunxi-can 453c800.can: PM runtime suspend.
[ 242.845765] sunxi-can 453c800.can: PM runtime resume.
[ 242.851652] IPv6: ADDRCONF(NETDEV_CHANGE): can0: link becomes ready
```

The Trace window shows the following log entries:

Time	Level	Message
15:50:28	info	Starting measure...
15:50:28	info	Listening on inter...

Step 4: Configure CAN as the receiver.

```
# candump can0
```



The screenshot shows the SecureCRT terminal window with the following commands and output:

```
# ip link set can0 down
[ 229.122261] sunxi-can 453c800.can: PM runtime resume.
[ 229.128055] sunxi-can 453c800.can: PM runtime suspend.
# ip link set can0 type can bitrate 1000000 loopback off
[ 236.650194] sunxi-can 453c800.can: PM runtime resume.
[ 236.656148] sunxi-can 453c800.can: PM runtime suspend.
# ip link set can0 up
[ 242.834099] sunxi-can 453c800.can: PM runtime resume.
[ 242.839827] sunxi-can 453c800.can: PM runtime suspend.
[ 242.845765] sunxi-can 453c800.can: PM runtime resume.
[ 242.851652] IPv6: ADDRCONF(NETDEV_CHANGE): can0: link becomes ready
```

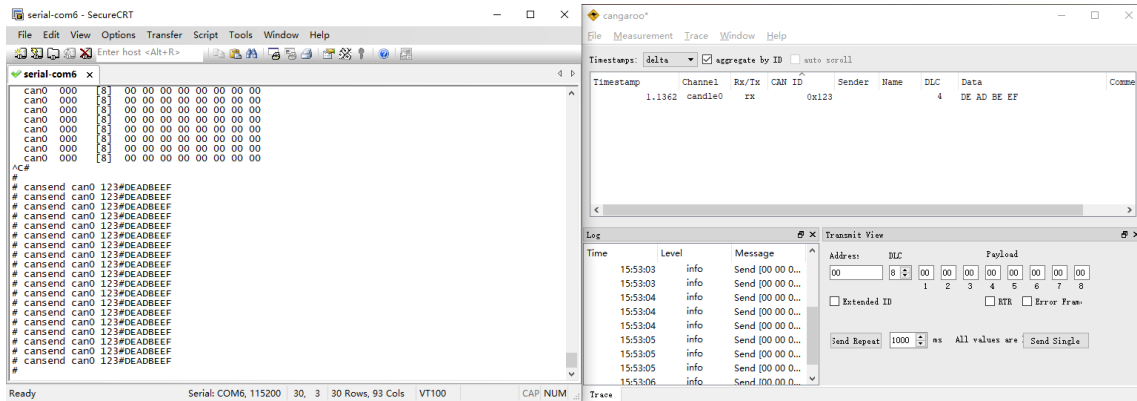
```
# candump can0
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
can0 000 [8] 00 00 00 00 00 00 00
```

The Trace window shows the following log entries:

Time	Level	Message
15:53:03	info	Send [00 00 0...
15:53:03	info	Send [00 00 0...
15:53:04	info	Send [00 00 0...
15:53:04	info	Send [00 00 0...
15:53:05	info	Send [00 00 0...
15:53:05	info	Send [00 00 0...
15:53:05	info	Send [00 00 0...
15:53:06	info	Send [00 00 0...

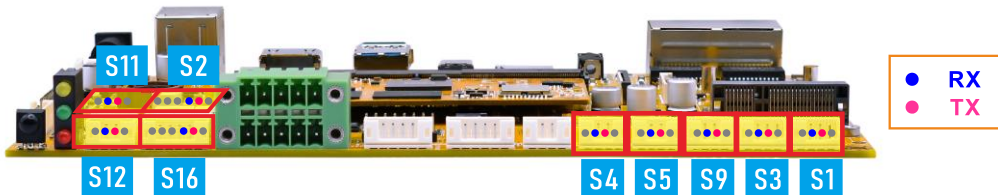
Step 5: Configure CAN as the sender.

```
# cansend can0 123#DEADBEEF
```



6.13 UART

Step 1: Short circuit RX and TX pins of UART.



```
# com /dev/ttyAS* 115200 8 0 1
```

Parameter description:

```
com : serial port test utility
/dev/ttyAS* : UART device node to be tested
115200 : baud rate
8 : data bits
0 : no parity
1 : one stop bit
```

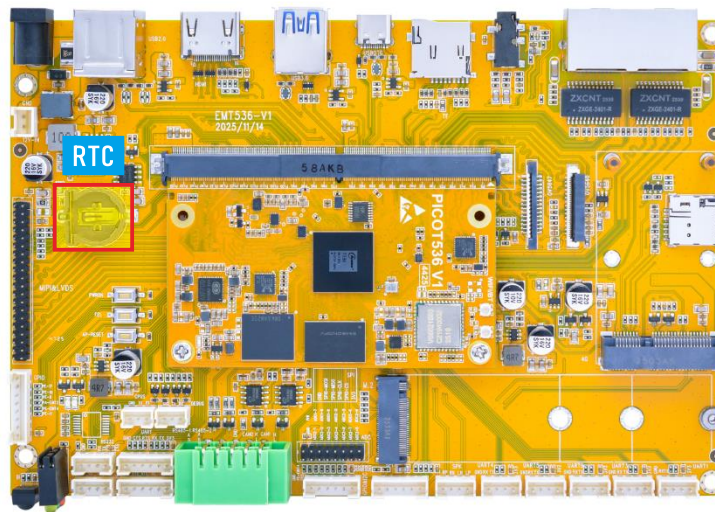
Step 2: Take UART1 (S1) as an example.

```
# com /dev/ttyAS1 115200 8 0 1
```

```
# com /dev/ttyAS1 115200 8 0 1
port = /dev/ttyAS1
baudrate = 115200
cs = 8
parity = 0
stopb = 1
jkjkkjkjkjkjk
RECV: jkjkjkjkjkjkjk
5656565656
RECV: 5656565656
ooooo
RECV: ooooo
jkjju6677
RECV: jkjjju6677
```

5.14 RTC

Step 1: Install the Lithium Cell (3V. CR1220), then power on.



Step 2: Set the system time.

```
# date -s "2026-03-11 17:24:30"
```

Step 3: Write the system time to the hardware clock.

```
# hwclock -w
```

Step 4: Display the current hardware clock time.

```
# hwclock
```

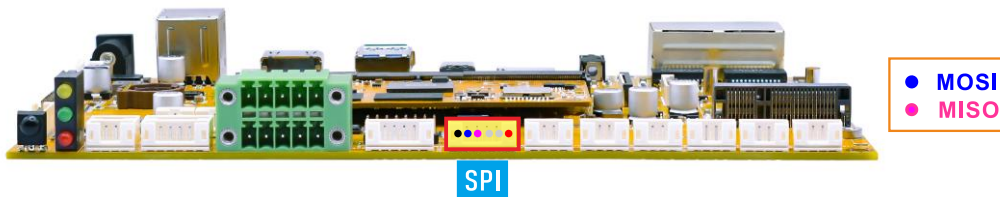
```
# date -s "2026-03-11 17:24:30"
Wed Mar 11 17:24:30 UTC 2026
# hwclock -w
# hwclock
Wed Mar 11 17:24:37 2026  0.000000 seconds
# hwclock
Wed Mar 11 17:24:51 2026  0.000000 seconds
# hwclock
Wed Mar 11 17:25:07 2026  0.000000 seconds
```

Step 5: Power off, after a period of time to turn on the power again, check whether the time is saved.

```
# hwclock
Wed Mar 11 17:35:38 2026  0.000000 seconds
# hwclock
Wed Mar 11 17:36:01 2026  0.000000 seconds
# hwclock
Wed Mar 11 17:36:59 2026  0.000000 seconds
# hwclock
Wed Mar 11 17:37:04 2026  0.000000 seconds
```

6.15 SPI

Step 1: short circuit MISO and MOSI pins of SPI.



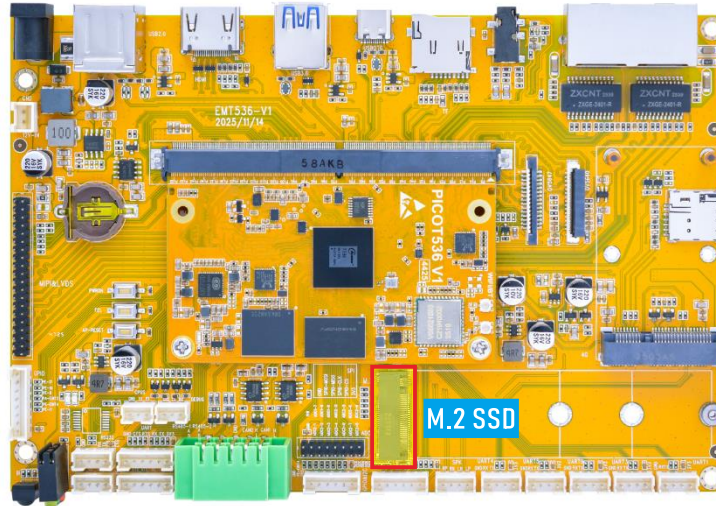
Step 2: Test spi.

```
# spidev_test -s 20000000 -v
```

6.16 M.2 SSD

Note: The M.2 SSD interface is multiplexed with USB 3.0, so only one of the two can be used at a time.

Step 1: Insert the SSD into the M.2 slot, then power on the device.

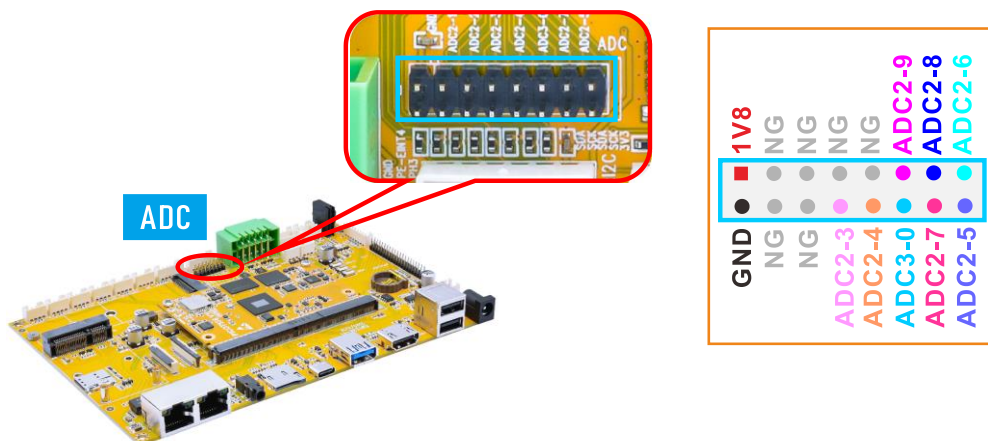


Step 2: The system will automatically mount it, view the device mount path.

```
# df -h
```

```
# df -h
Filesystem      Size      Used Available Use% Mounted on
/dev/root       990.7M    232.2M   742.4M   24% /
devtmpfs        980.0M    0         980.0M   0% /dev
tmpfs           982.5M    0         982.5M   0% /dev/shm
tmpfs           982.5M    216.0K   982.3M   0% /tmp
tmpfs           982.5M    288.0K   982.2M   0% /run
/dev/nvme0n1p1  468.5G    32.0K    463.7G   0% /mnt/nvme/nvme0n1p1
/dev/by-name/UDISK  6.1G     4.0K     6.1G    0% /mnt/UDISK
```

6.17 GPADC



Note: The ADC input voltage must not exceed 1.8V.

GPADC test command:

```
ADC2-3 : cat /sys/bus/iio/devices/iio\:device0/in_voltage3_raw
ADC2-4 : cat /sys/bus/iio/devices/iio\:device0/in_voltage4_raw
ADC3-0 : cat /sys/bus/iio/devices/iio\:device1/in_voltage0_raw
ADC2-7 : cat /sys/bus/iio/devices/iio\:device0/in_voltage7_raw
ADC2-5 : cat /sys/bus/iio/devices/iio\:device0/in_voltage5_raw
ADC2-6 : cat /sys/bus/iio/devices/iio\:device0/in_voltage6_raw
ADC2-8 : cat /sys/bus/iio/devices/iio\:device0/in_voltage8_raw
ADC2-9 : cat /sys/bus/iio/devices/iio\:device0/in_voltage9_raw
```

The following logs use ADC2-3 as an example and show the ADC readings when the input is connected to 0V and 1.8V, respectively:

```
# cat /sys/bus/iio/devices/iio\:device0/in_voltage3_raw
0
# cat /sys/bus/iio/devices/iio\:device0/in_voltage3_raw
1796
```