ORIENT DISPLAY

Your Total LCD Solution Provider

http://www.orientdisplay.com

AIY-A005M

User Manual ver01

Based on **(ROCKCHIP)RK3288**
ARM® Cortex®-A17 quad-core processor

Customer's Approval:

	SIGNATURE	DATE
PREPARED BY (RD ENGINEER)		
CHECKED BY		
APPROVED BY		

Contents

Chapter 1

Product Overview

This chapter mainly introduces

- ◆ Introduction to the motherboard
- ◆ Technical parameters

1.1 Overview

The AIY-A005M motherboard adopts Rockchip's RK3288 quad-core ARM Cortex-A17 architecture processor with a maximum frequency of 1.8GHz. GPU adopts Mali-T764, supports 4K H265/H264 video decoding, supports VC-1, MPEG-1/2/4, VP8 and other formats of video decoding, supports H.264, VP8 format video encoding. The 3D graphics processing unit supports OpenGL ES1.1/2.0/3.0, penCL 1.1/1.2. The motherboard is equipped with Android 7.1 system, standard 2GB DDR3 and 8GB EMMC, support TF card expansion.

The motherboard supports LVDS, MIPI, eDP, HDMI display output; provides 7 USB ports, 6 GPIO, 4 serial ports, 2 I2C, 1 CAN and 1 OTG, which can meet the needs of different customers; integrates Gigabit RJ45 Ethernet Network port, 2.4G WiFi (optional 2.4G/5G dual-band WiFi), support 4G wireless network card, can adapt to a variety of network environments.

1.2 Features

High performance: RK3288 chip adopts quad-core A17 solution, the main frequency is up to 1.8GHz, and it integrates super-powered Mali-T764 GPU. Compared with the common single-core, dual-core, quad-core and eight-core solutions in the market, it has a qualitative leap in performance, can play various formats of high-definition 4K*2K video, and can handle complex interactive operations.

High scalability: Seven USB, four RS232 serial ports (one of which can be reused as RS485), two I2C interfaces, one CAN bus, two I2C buses, and six GPIO expansion ports can expand more peripheral devices.

High integration: The motherboard integrates Gigabit Ethernet, WiFi, Bluetooth, dual-channel 5W audio output, microphone input, TF card expansion, built-in Mini-PCIe module interface, supports HDMI output, supports LVDS, eDP, MIPI and other mainstream display interfaces, Which greatly simplifies the design of the whole machine.

It perfectly supports a variety of mainstream touch screens such as infrared, optical, capacitive, and resistance, and supports HID configuration of drive-free touch screens without debugging.

Support Android system customization, perfect support for customer upper-level application APP development.

1.3 Product parameters

1.1.1 Function parameters

Processor:

— Rockchip's ARM Cortex™-A17 quad-core high-performance processor, clocked at up to 1.8GHz

— Mali-T764 GPU, support AFBC (frame buffer compression)

— Support OpenGL ES 1.1/2.0/3.1, OpenCL 1.1, DirectX11

— Support 1080P multi-format video decoding: VC-1, MPEG-1/2/4, VP8, H.263, H.264, AVS, MVC, HEVC, support 4K 10bits H265/H264 video decoding

— Support 1080P video encoding: H.264, VP8

Storage

—Running memory: 2GB DDR3 onboard (up to 4GB)

—Storage: 8GB eMMC onboard (up to 128GB)

Network

—Ethernet: Support Ethernet 10/100/1000Mbps

—WIFI: Support 2.4G WiFi, support 802.11b/g/n protocol (optional 5G dual-band WIFI)

—Bluetooth: support Bluetooth function, V2.1+EDR/Bluetooth 3.0/3.0+HS/4.0

—Mobile network: Support MiniPCIe expansion 4G module

Peripheral interface:

-Dual 8-bit LVDS*1

—MIPI interface*2, can support dual-channel MIPI screen

—EDP interface*1, support 4K@30Hz

—HDMI interface*1: Support 1080P@120Hz or 4K@60Hz

—Standard USB interface*2, standard OTG interface*1, USB pitch2.0 socket*5

—Dual channel audio output, single mic input

—Micro SD (TF) slot*1

—RS232 serial port*4, one of which can be multiplexed as RS485

—Mini PCIe slot*1

—SIM card (nano) slot*1

—I2C bus*2

—CAN bus*1

—GPIO*6

operating system: support Android or Linux

1.4 Electrical characteristics

Power adapter: 12V DC-IN RTC

Battery: 3V button battery

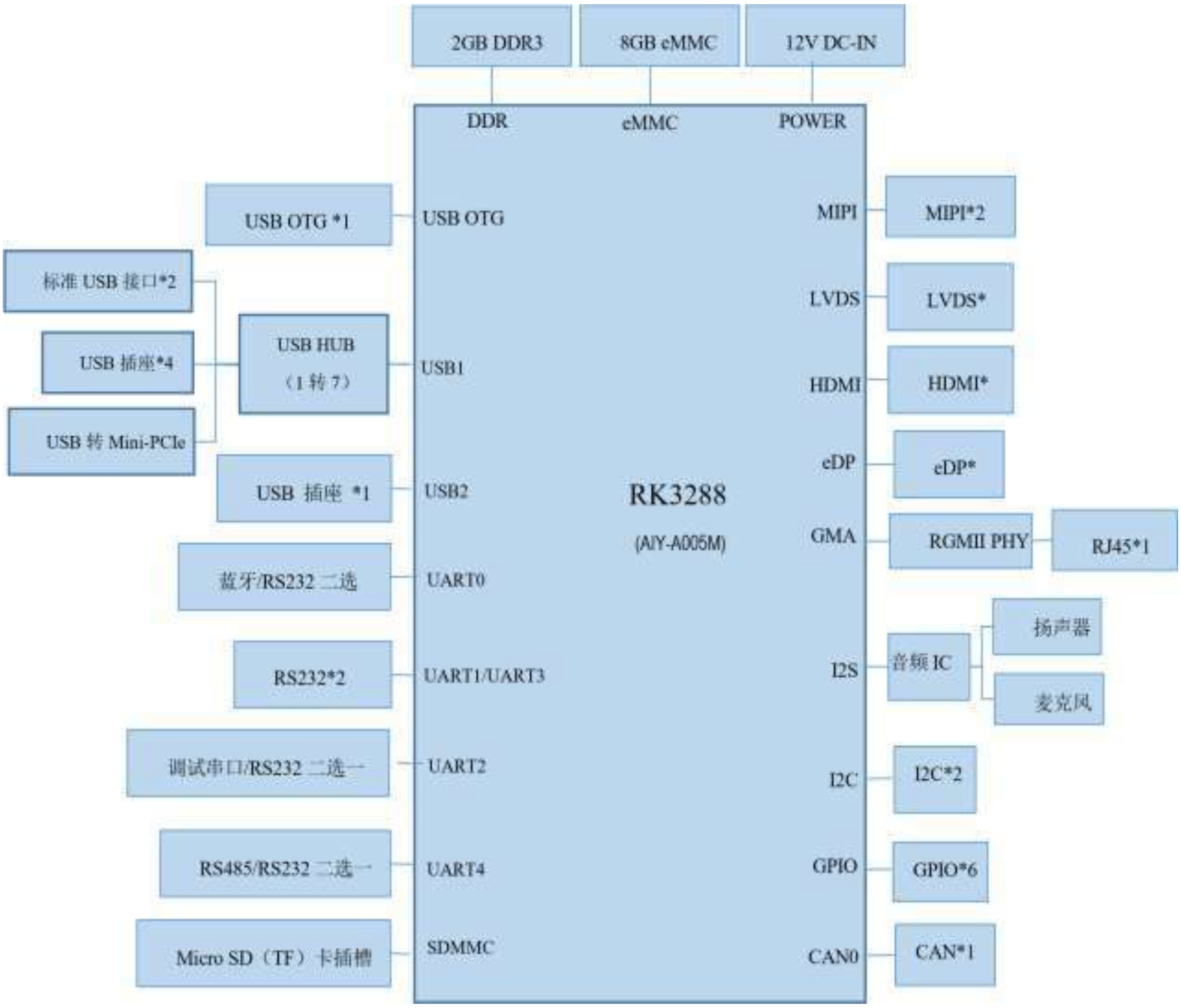
1.5 Environmental characteristics

Working temperature: -20~75℃

Working humidity: 5%~95%

Storage temperature: -30~85℃

1.6 System Block Diagram



Chapter 2

Hardware Function Description

This chapter mainly introduces

- Jumper description
- DIP switch description
- Connector description
- Structure description

2.1 Jumper

The jumper is a small switch on the control circuit board, and its function is to adjust the on-off relationship of different electrical signals on the main board. Connecting different jumper PINs through the jumper cap can change the motherboard circuit and adjust the working status of the device to achieve different purposes, such as determining the peripheral power supply voltage, switching multiplexing functions, and so on.

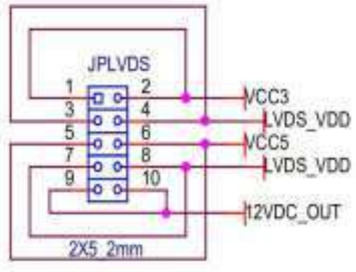
The jumper cap is a movable part. The outer layer is made of insulating plastic and the inner layer is made of conductive material. It can be inserted on the jumper pins to connect the two jumper pins. When the jumper cap is buckled on the two jumper pins, it is in the ON state, and there is current passing through it, which is called the ON state; otherwise, when the jumper cap is not buckled, it means it is off, which is called the OFF state.

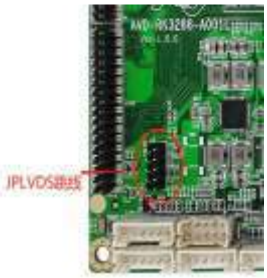
2.1.1 Jumper list

Jumper list	
JPLVDS	LVDS screen supply voltage selection
JPLCD	LVDS backlight supply voltage selection
JPEDP	EDP screen power supply voltage selection

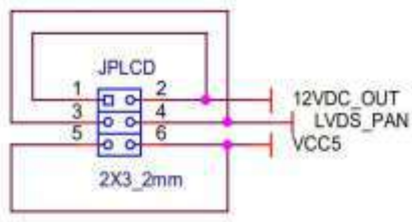
2.1.2 Jumper Setting Introduction

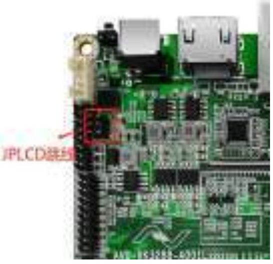
JPLVDS：LVDS screen power supply options	
Jumper pin arrangement	5X2PIN 2.0mm pitch
3.3V power supply	1-3 (indicating that pin 1 is connected to pin 3, the same below) or 2-4
5V power supply	3-5 or 4-6 or 5-7 or 6-8
12V power supply	7-9 or 8-10
Note: The PIN with a square pin pad is the first pin	



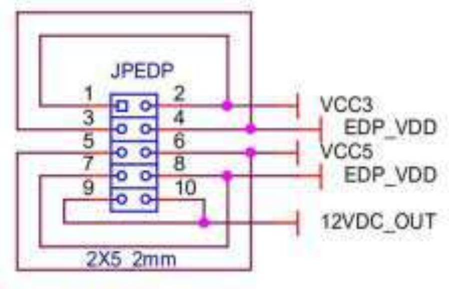


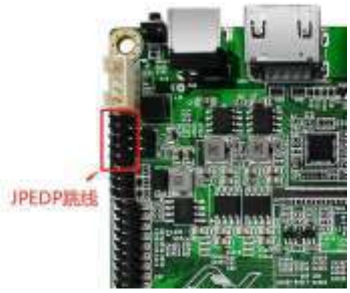
JPLCD：LVDS backlight supply voltage selection	
Jumper pin arrangement	3X2PIN 2.0mm pitch
12V power supply	1-3 (indicating that pin 1 is connected to pin 3, the same below) or 2-4
5V power supply	3-5 or 4-6
Note: The PIN with a square pin pad is the first pin	





JPEDP：EDP screen power supply voltage selection	
Jumper pin arrangement	5X2PIN 2.0mm pitch
3.3V power supply	1-3 (indicating that pin 1 is connected to pin 3, the same below) or 2-4
5V power supply	3-5 or 4-6 or 5-7 or 6-8
12V power supply	7-9 or 8-10
Note: The PIN with a square pin pad is the first pin	





2.2 Dip switch

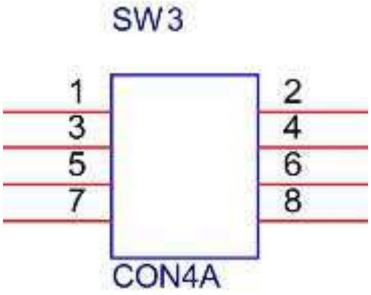
DIP switch (also called DIP switch, toggle switch, overclocking switch, address switch, pull switch, digital switch, thumb switch) is an address switch for operation and control, using the principle of 0/1 binary coding.

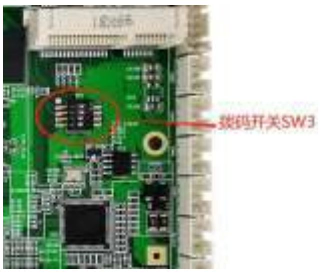
Each key of the DIP switch has two pins on the top and bottom of the back. Turn it to the ON side, and the two pins below are connected to the channel; otherwise, it is disconnected. Each key of the multi-digit DIP switch is independent and not related to each other.

2.2.1 DIP switch list

DIP switch list	
SW3	COM4 RS232/RS485 level switch

SW3：COM4 RS232/RS485 level switch	
Number of switches	4 BIT
RS232 level	1100 (1-2, 3-4 are in ON state, 5-6, 7-8 are in OFF state)
RS485	0011 (1-2, 3-4 are in OFF state, 5-6, 7-8 are in ON state)





2.3 Connector

Connector list	
CON3	MIPI camera FPC connector
CON4/CON5	MIPI screen FPC connector
LVDS	LVDS screen connection header
EDP	EDP screen connection header
TOUCH1	Touch screen connector 1
TOUCH2	Touch screen connector 2

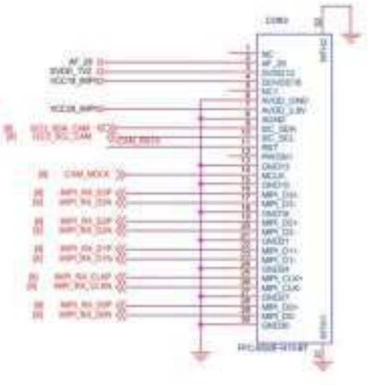
MINI PCIE CONN	4G network card socket
SIM	SIM socket
TF	TF card socket
ETH0	Ethernet RJ45 socket
OTG	Micro USB socket
USB1/UBS2	Universal USB socket
USB3~USB7	Pitch2.0 USB socket
SPK	Pitch2.0 audio interface socket
JPIO	GPIO socket
COMDB	RS232 socket (debug serial port/COM2 multiplexing, the default is debug serial port)
COM1/COM3/COM4	RS232 socket
RS485	RS485 socket (choose one from COM4 of RS232 level)
CAN	CAN bus socket

2.3.1 Connector Instructions

2.3.1.1 Camera connector（CON3）

The AIY-A005M motherboard supports MIPI CSI interface cameras, which can support up to 13 million pixels. It can be connected to the motherboard MIPI camera connector through FPC. The following is the connector PIN definition:

Definition	Pin	Pin	Definition
NC	1	16	MIPI_RX_D3P
AF_28	2	17	MIPI_RX_D3N
DVDD_1V2	3	18	GND
VCC1V8_MIPI	4	19	MIPI_RX_D2P
NC	5	20	MIPI_RX_D2N
GND	6	21	GND
AVDD2V8	7	22	MIPI_RX_D1P
GND	8	23	MIPI_RX_D1N
IIC_SDA_CAM	9	24	GND
IIC_SCL_CAM	10	25	MIPI_RX_CLKP
CAM_RST	11	26	MIPI_RX_CLKN
NC	12	27	GND
GND	13	28	MIPI_RX_D0P
CAM_MCLK	14	29	MIPI_RX_D0N
GND	15	30	GND

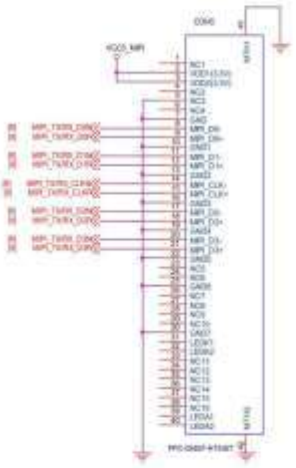


2.3.1.2 MIPI Screen Connector（CON4/CON5）

The AIY-A005M motherboard provides two MIPI outputs, which can be used as two separate MIPI outputs or one dual-channel MIPI output. The following is the PIN definition:

Definition	Pin		Definition
NC	1	21	MIPI_TX_D3P
VCC3_MIPI	2	22	GND
VCC3_MIPI	3	23	NC
NC	4	24	NC
GND	5	25	GND
NC	6	26	NC
GND	7	27	NC
MIPI_TX_D0N	8	28	NC
MIPI_TX_D0P	9	29	NC
GND	10	30	GND
MIPI_TX_D1N	11	31	NC
MIPI_TX_D1P	12	32	NC

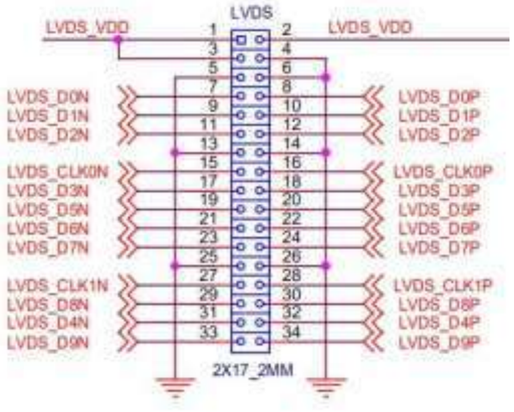
GND	13	33	NC
MIPI_TX_CLKN	14	34	NC
MIPI_TX_CLKP	15	35	NC
GND	16	36	NC
MIPI_TX_D2N	17	37	NC
MIPI_TX_D2P	18	38	NC
GND	19	39	NC
MIPI_TX_D3N	20	40	NC

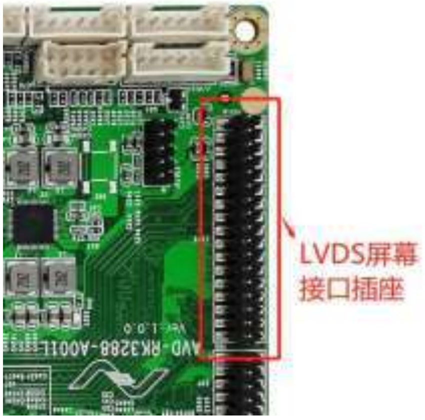


2.3.1.3 LVDS screen connection（LVDS）

The AIY-A005M motherboard provides a 34PIN (17X2) LVDS connection header, which can support dual 8-bit LVDS, with a maximum resolution of 1920X1080. The VDD voltage is determined by the LVDS screen power supply jumper (JPLVDS) (for details, please refer to Rely on chapter 2.1.2), the PIN of the LVDS screen connection interface is defined as:

Definition	Pin		Definition
VDD	1	2	GND
VDD	3	4	GND
GND	5	6	GND
LVDS_D0N	7	8	LVDS_D0P
LVDS_D1N	9	10	LVDS_D1P
LVDS_D2N	11	12	LVDS_D2P
GND	13	14	GND
LVDS_CLK0N	15	16	LVDS_CLK0P
LVDS_D3N	17	18	LVDS_D3P
LVDS_D5N	19	20	LVDS_D5P
LVDS_D6N	21	22	LVDS_D6P
LVDS_D7N	23	24	LVDS_D7P
GND	25	26	GND
LVDS_CLK1N	27	28	LVDS_CLK1P
LVDS_D8N	29	30	LVDS_D8P
LVDS_D4N	31	32	LVDS_D4P
LVDS_D9N	33	34	LVDS_D9P

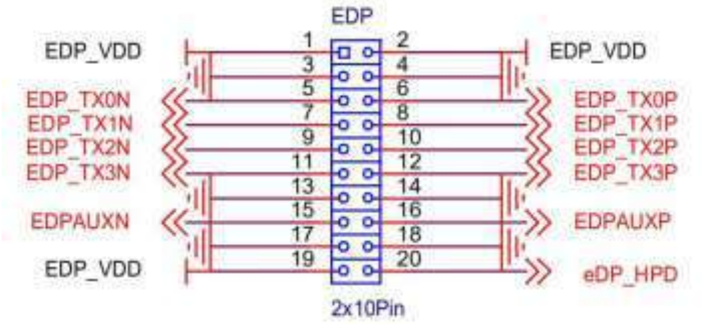


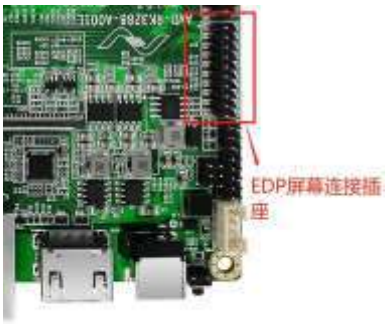


2.3.1.4 EDP screen connection（EDP）

The AIY-A005M motherboard provides a 20PIN (10X2) EDP connection header, which can support 4Lane eDP screens, and the highest resolution can reach 4K (3840*2160). The EDP_VDD voltage is determined by the EDP screen power supply jumper JPEDP (specifically Please refer to chapter 2.1.2), the PIN of the EDP screen connection interface is defined as:

Definition	Pin		Definition
EDP_VDD	1	2	EDP_VDD
GND	3	4	GND
TX0N	5	6	TX0P
TX1N	7	8	TX1P
TX2N	9	10	TX2P
TX3N	11	12	TX3P
GND	13	14	GND
AUXN	15	16	AUXP
GND	17	18	GND
VDD	19	20	HPD

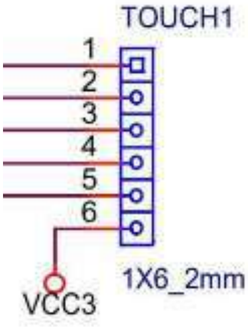


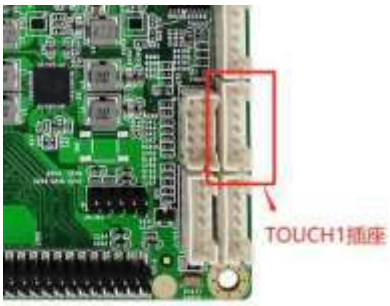


2.3.1.5 Touch Screen Connector（TOUCH1/TOUCH2）

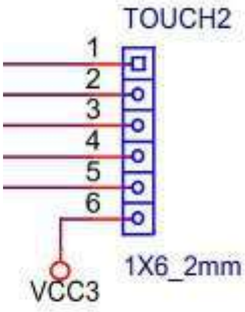
The touch screen connection socket is a 6PIN (6X1) connection pin socket, including a set of power supply, a set of I2C bus, interrupt and reset GPIO, suitable for touch screens using I2C protocol, the PIN is defined as:

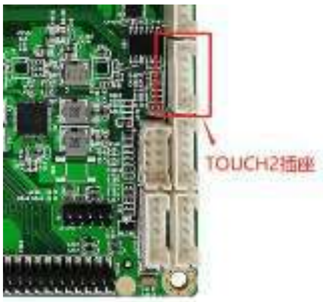
Definition	Pin		Definition
IIC_SCL	1	4	TP1_IRQ
IIC_SDA	2	5	TP1_RST
GND	3	6	VCC





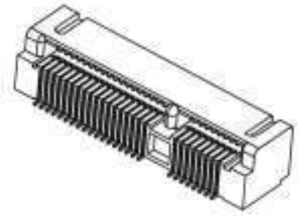
Definition	Pin		Definition
IIC_SCL	1	4	TP2_IRQ
IIC_SDA	2	5	TP2_RST
GND	3	6	VCC

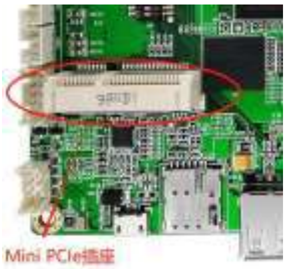




2.3.1.6 4G Network Card Socket（MINI PCIE CONN）

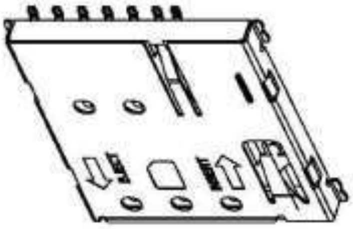
The AIY-A005M motherboard supports Mini PCIe interface, which can be used to support 4G network card. Mini PCIe is a general standard interface, and there is no PIN definition description。

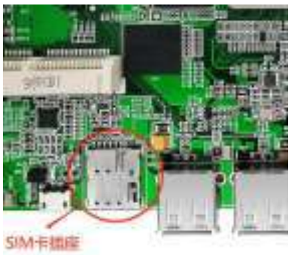




2.3.1.7 SIM Socket（SIM）

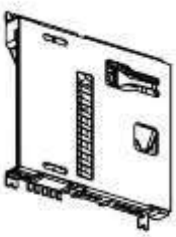
AIY-A005M motherboard supports standard Nano SIM card socket, which can be used to support 4G network, Nano SIM is a general standard socket:

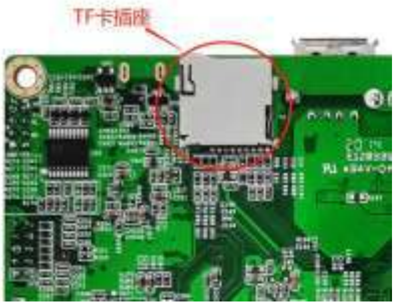




2.3.1.8 TF card socket（TF）

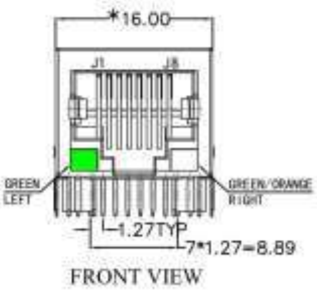
AIY-A005M motherboard supports TF (MICRO SD) card socket, which can be used for storage expansion:





2.3.1.9 Ethernet RJ45 Socket（ETH0）

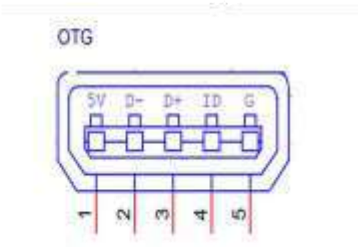
AIY-A005M motherboard provides one RJ45 Ethernet interface, which can adapt to 10/100/1000Mbps, RJ45 includes two LED indicators to indicate the network connection status and network speed status:



2.3.1.10 Micro USB Connector OTG）

The AIY-A005M motherboard provides an OTG interface, using a Micro USB port, which can be used to connect to a PC as an adb debugging interface, and can also be used to connect OTG peripherals.

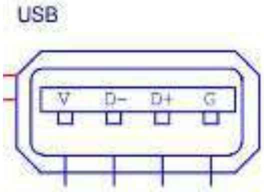
Definition	Pin		Definition
VCC 5V	1	4	ID
D-	2	5	GND
D+	3		



2.3.1.11 Universal USB Socket（USB1/USB2）

The AIY-A005M motherboard provides two universal USB 2.0 interfaces (TYPE-A).

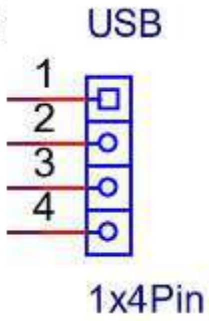
Definition	Pin		Definition
VCC 5V	1	3	D+
D-	2	4	GND



2.3.1.12 Pitch2.0 USB socket（USB3~USB7）

In order to meet the needs of users for multi-channel USB, the AIY-A005M motherboard also provides 5-channel USB sockets, which are suitable for Pitch2.0 cables.

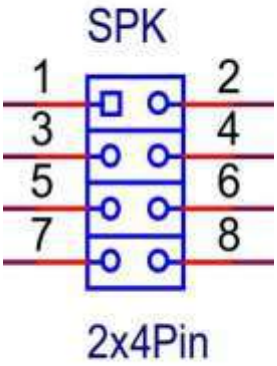
Definition	Pin		Definition
VCC 5V	1	3	D+
D-	2	4	GND

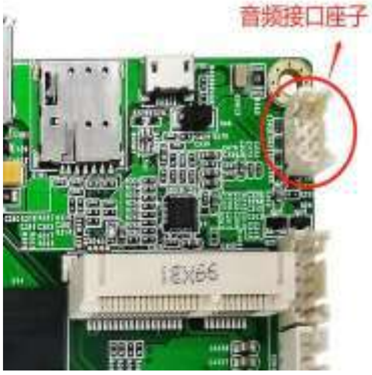


2.3.1.13 Pitch2.0 audio interface socket (SPK)

AIY-A005M motherboard audio interface includes dual-channel output and MIC input, which can be connected to 5W dual-channel output.

Definition	Pin		Definition
OUTPL	1	2	MIC_IN1L
OUTNL	3	4	MIC_IN1R
OUTNR	5	6	MIC_IN2L
OUTPR	7	8	GND_signal

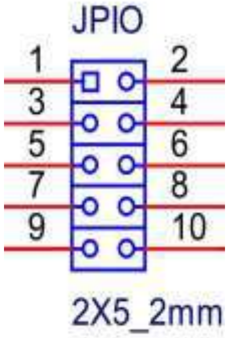


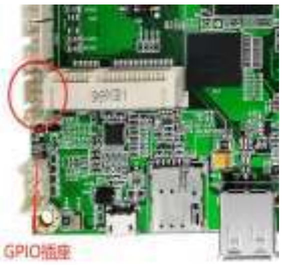


2.3.1.13 GPIO Socket（JPIO）

The AIY-A005M motherboard provides 6 GPIO interfaces:

Definition	Pin		Definition
VCC	1	2	VCC
GPIO5_C0	3	4	GPIO8_A1
GPIO5_C1	5	6	GPIO8_A0
GPIO5_C2	7	8	GPIO5_C3
GND	9	10	GND

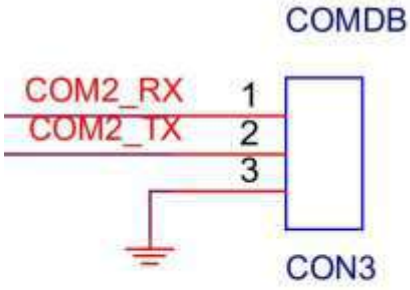


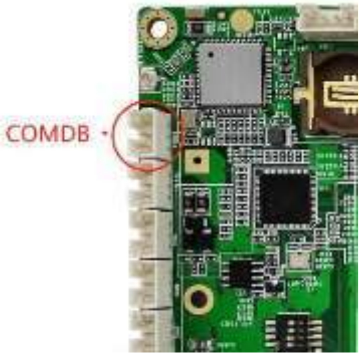


2.3.1.14 Debug serial socket（COMDB）

The debugging serial port of AIY-A005M motherboard is RS232 level, and can be reused as COM2, that is, COMDB can be cancelled and it becomes a normal RS232 serial port (you need to modify the system software for configuration):

Definition	Pin		Definition
RX	1	3	GND
TX	2		

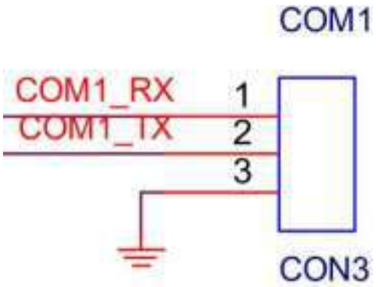


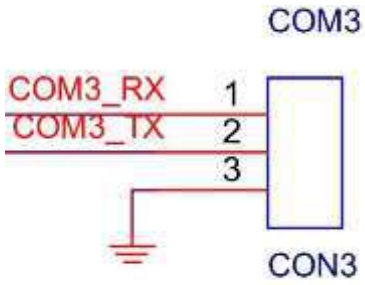


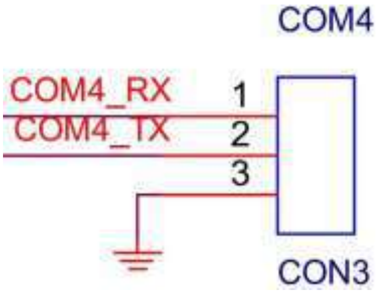
2.3.1.15 RS232 Socket（COM1/COM3/COM4）

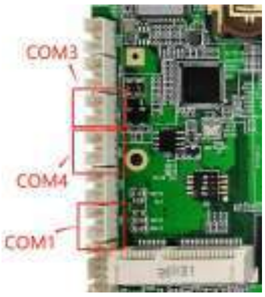
In addition to COMDB, the AIY-A005M motherboard also provides 3 RS232 serial ports:

Definition	Pin		Definition
RX	1	3	GND
TX	2		





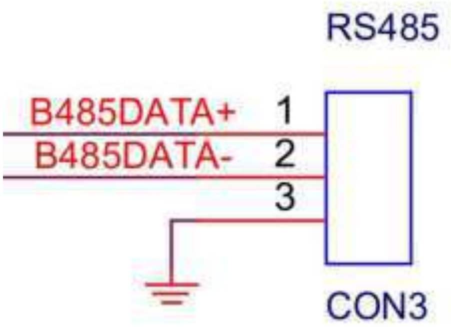


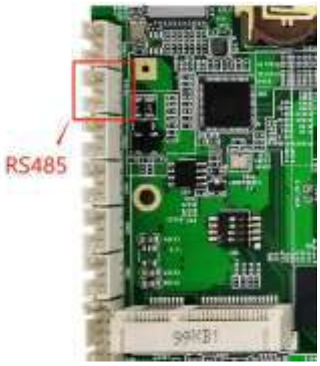


2.3.1.16 RS485 Socket (RS485)

The AIY-A005M mainboard provides a RS485 interface (multiplexed with COM4), and the level of the serial port can be controlled to RS485 or RS232 through the dial switch (see chapter 2.2.1 for details)

Definition	Pin		Definition
DATA+	1	3	GND
DATA-	2		

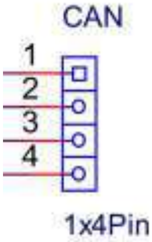


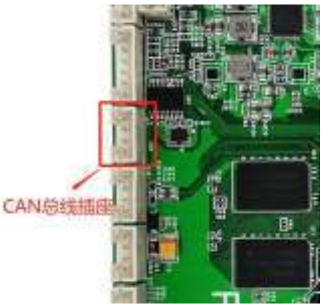


2.3.1.17 CAN bus socket（CAN）

The AIY-A005M motherboard provides a CAN bus output, which can be used to communicate with external CAN devices:

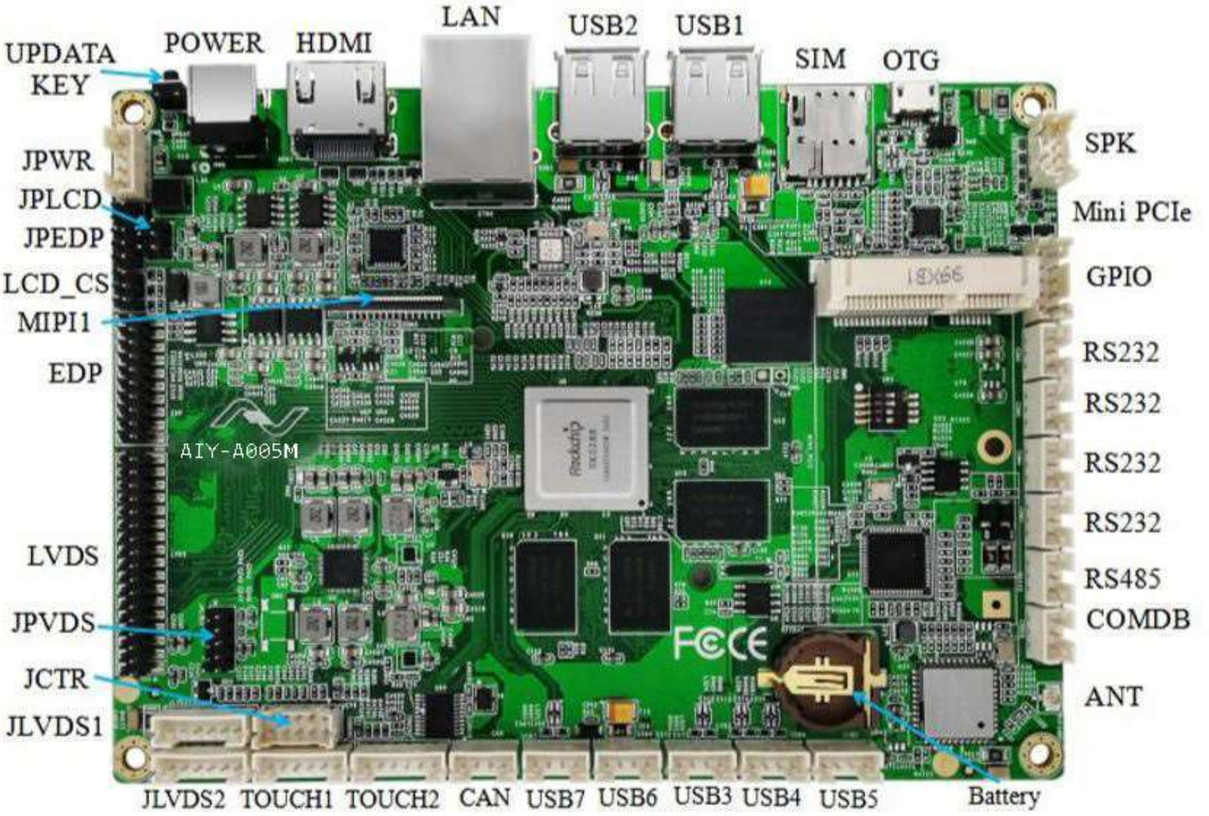
Definition	Pin		Definition
VCC	1	3	CAN_L
CAN_H	2	4	GND

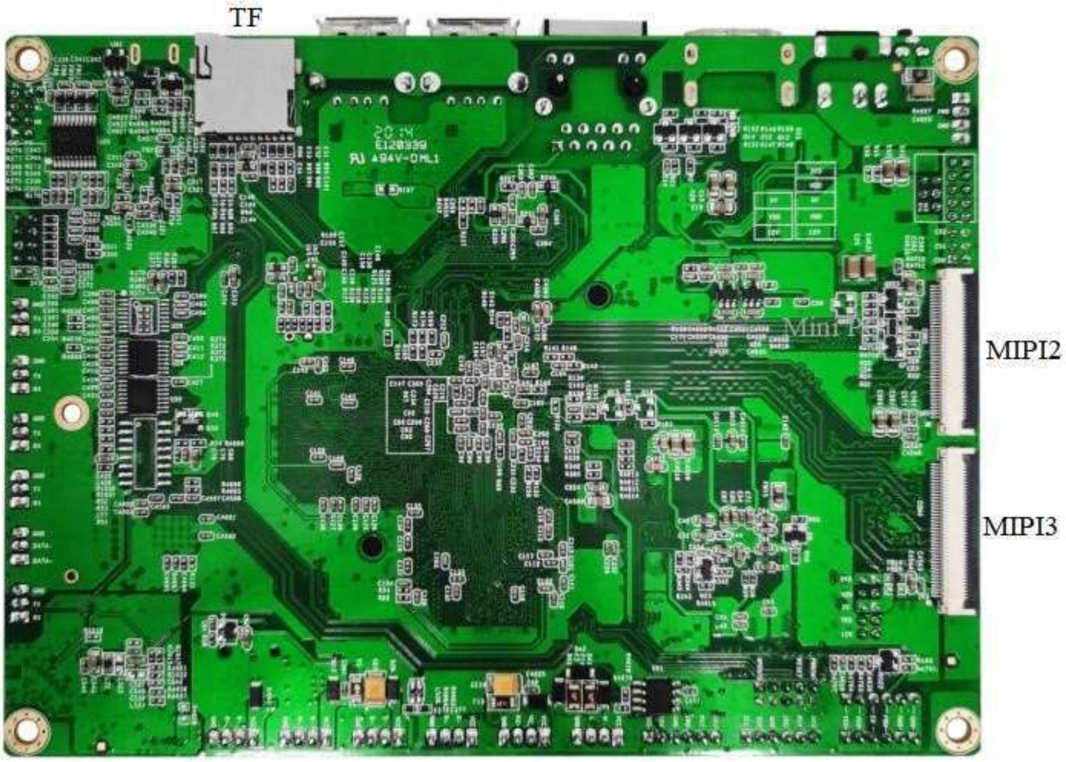




2.4 Structure description

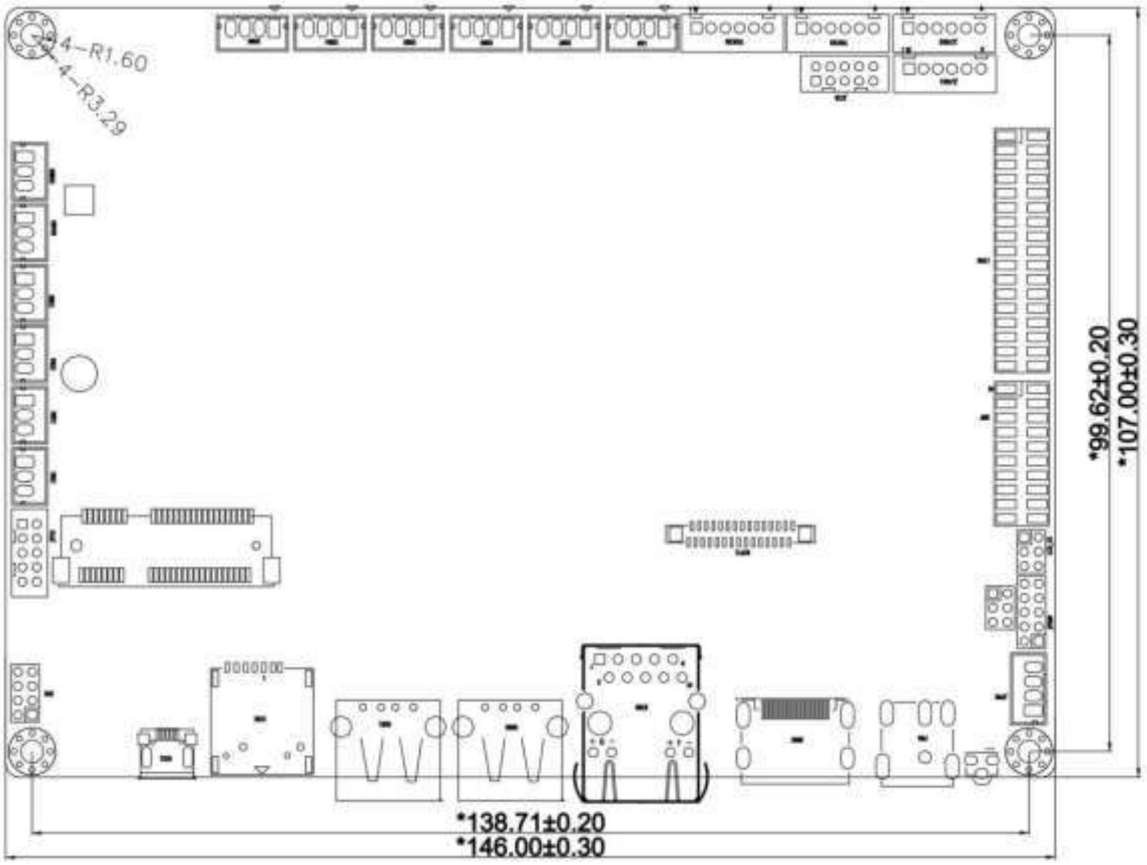
2.4.1 Structure and connector location diagram



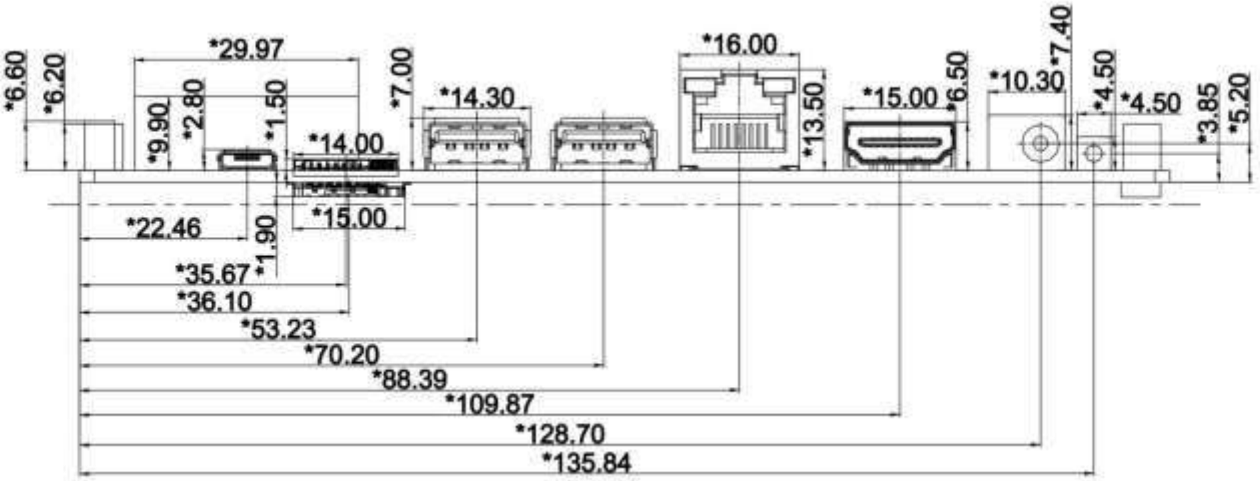


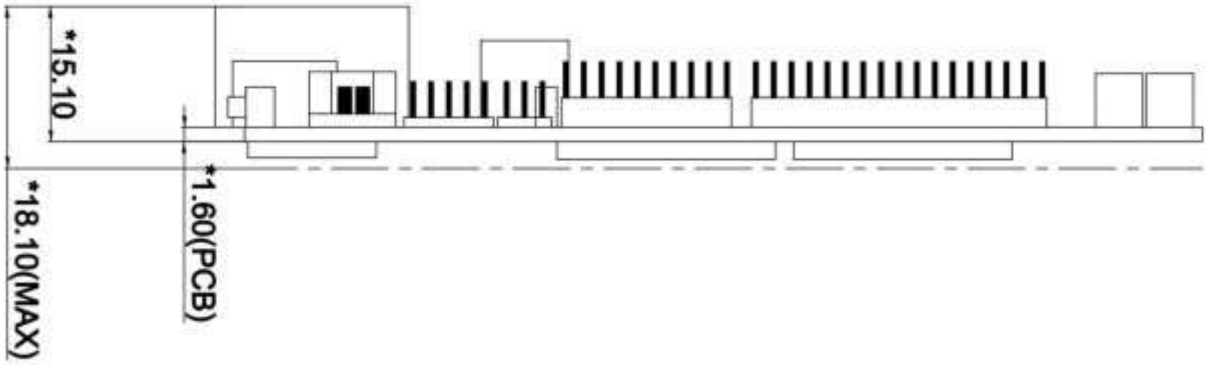
2.4.2 Structure size chart

2.4.2.1 Front view



2.4.2.2 Side View





Chapter 3

Software Function Description

This chapter mainly introduces

- Debug interface description
- Hardware programming instructions

3.1 Debug interface

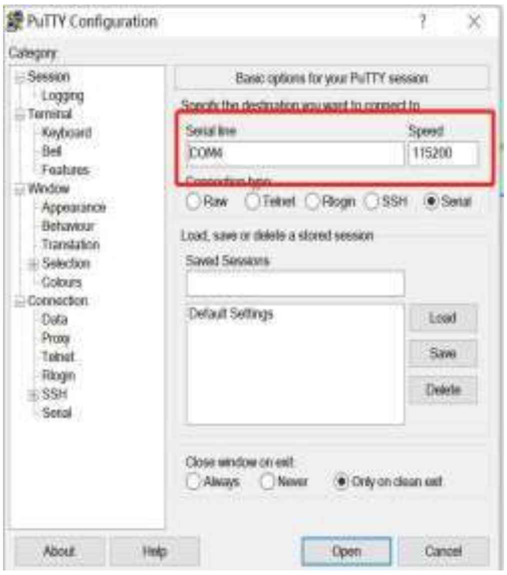
3.1.1 Serial debugging

3.1.1.1 Debug serial port connection

The COMDB pin definition of the motherboard debugging serial port can refer to 2.3.1.14 or check the motherboard silkscreen. The debugging serial port is RS232 level. If the debugging computer is equipped with a standard serial port (usually RS-232 level), you can use the serial cable to connect the computer to the AIY-A005M, if you are using a portable computer (usually without a standard serial port) or a desktop computer that does not provide a standard serial port, you need a USB to RS-232 serial cable to connect the computer and the motherboard, and you are running Windows When using a USB to RS-232 serial cable on a computer with an operating system, you need to install the corresponding driver first (Provided by the manufacturer of the conversion chip).

3.1.1.2 Serial port parameter configuration

After connecting the motherboard to the PC, use the serial communication tool on the PC to communicate with the motherboard, print the motherboard serial information or input debugging commands. The serial communication tool used in this article is PUTTY, and the configuration is as shown in the figure below:



Specify the destination you want to connect to

Serial line: COM4

Speed: 115200

Connection type: ○ Raw ○ Telnet ○ Rlogin ○ SSH ● Serial

Close window on exit: ○ Always ○ Never ● Only on clean exit

As shown in the red box, confirm the serial number of the PC port (can be viewed through the device manager), fill in the serial number of the connected serial port in "Serial line", and set the baud rate "Speed" to 115200, and then click OPEN You can enter the debug serial port command terminal.

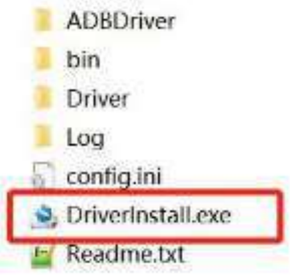
3.1.2 adb debugging

3.1.2.1 adb debug interface connection

The adb debugging port of AIY-A005M is an OTG port (for detailed definition, please refer to 2.3.1.10). Developers can use a Micro USB cable to connect the motherboard to a PC.

3.1.2.1 adb driver installation

Generally, to debug the motherboard through the adb interface, you may need to install the adb driver on the PC. Developers can download common mobile assistants (such as 360 mobile assistants) and other software. Such software will automatically adapt to the devices connected to the PC. Download the relevant driver, or install the driver software DriverAssitant provided by Rockchip to support adb debugging, as shown in the following figure:



ADBDriver
bin
Driver
Log
config.ini
DriverInstall.exe
Readme.txt

3.1.2.2 adb Installation

Developers can download the compressed package of adb kits and get adb.exe after decompression. Add the decompressed directory of adb directly to the environment variable, and then run the adb command directly in the windows command prompt (cmd).

ADB Kits (1364 KB) (adb.exe AdbWinApi.dll AdbWinUsbApi.dll)

3.1.2.3 Use of adb

After the driver is installed successfully, open cmd and run the adb devices command to see the adb devices connected to the PC:

```
C:\Users\admin>adb devices
List of devices attached
AUK0BFEX73      device
```

In this case, you can enter the adb command line of the device by running the adb shell command (more adb commands can be searched online by yourself).

3.1.2.4 Switch to root user

After logging in to the command line of the motherboard through the serial port or adb, the default is a normal user (shell). For security reasons, some high-privileged commands of the Android system require the user to switch to root to execute. When using the serial port for debugging, the user can directly use the su command Switch to the root user. After switching, you can confirm the current user through whoami.

```
rk3288:/ Whoami

shell

rk3288:/ $ su

rk3288:/ # whoami

root
```

When debugging with adb, the user can also switch to root through the su command, or switch to root authority through the adb root command before running adb shell, and then log in to adb using adb shell:

```
C:\Users\admin>adb root
restarting adbd as root
```

3.2 GPIO

3.2.1 gpio operation interface

This motherboard builds the operating system and provides a convenient and quick operation interface for GPIO. The GPIO subsystem directory is /sys/class/gpio. There are the following file nodes in the directory:

```
rk3288:/ $ ls /sys/class/gpio/ export gpiochip0 gpiochip120 gpiochip152 gpiochip184
gpiochip216 gpiochip24 gpiochip248 gpiochip56 gpiochip88 unexport
```

Among them, export and unexport are the attribute files of the GPIO subsystem. Export is used to export gpio. After gpio is exported, related nodes will appear in the /sys/class/gpio/ directory. Operating related nodes can realize the direction and level of gpio. Read and write status, etc.

Write the sequence number of the GPIO to the export file to export its device catalog. The calculation formula of the sequence number is as follows:

GPIO sequence number = 32 x BANK+ N – 8

In the formula, BANK is the gpio group where the GPIO pin is located, and N is the serial number of the pin in the group. If the pin is A0~A7, N is 0~7, B0~B7 is 8~15, and C0~C7 It is 16~23, D0~D7 is 24~31. Take GPIO5_C0 on the motherboard as an example, its BANK value is 5 and N value is 16, so the sequence number is 32*5+16-8=168. Then the operation command to write the sequence number (requires root authority) is as follows:

```
rk3288:/ # echo 168 > /sys/class/gpio/export
```

After the above command is executed, the gpio168 directory will be generated under the /sys/class/gpio directory. The purpose of operating this GPIO can be achieved by reading and writing the device files in this directory. By analogy, you can also export other GPIO device catalogs.

After the device directory of GPIO168 is generated, you can see that it contains the following property files:

```
rk3288:/ # ls /sys/class/gpio/gpio168/ active_low
device direction edge power subsystem uevent value
```

Among them, direction and value are commonly used, direction is used to configure gpio as input or output, and value is used to output level (when used as output) or read level (when used as input).

3.2.2 Operate GPIO from the command line

After exporting, GPIO defaults to input function. You can view the current operation direction of the GPIO by reading the direction file:

```
rk3288:/ # cat /sys/class/gpio/gpio168/direction

in
```

Write the "out" string to the direction file to set GPIO as output:

```
rk3288:/ # echo out > /sys/class/gpio/gpio168/direction
```

Similarly, you can write "in" string to set GPIO back to input

When GPIO is set as input, the value file records the input level status of the GPIO pin: 1 means high level input; 0 means low level input. The input level of GPIO can be read by viewing the value file:

```
rk3288:/ # echo in > /sys/class/gpio/gpio168/direction
rk3288:/ # cat /sys/class/gpio/gpio168/value
1
```

When the GPIO is set to output, the state of the output level can be controlled by writing 0 or 1 to the value file (0 means low output, 1 means high output):

```
rk3288:/ # echo out > /sys/class/gpio/gpio168/direction
rk3288:/ # echo 1 > /sys/class/gpio/gpio168/value
rk3288:/ # echo 0 > /sys/class/gpio/gpio168/value
```

3.3 Serial port

3.3.1 Serial file node

Like most other devices, the serial port of the android system appears as a device file. AIY-A005M motherboard serial device file is /dev/ttySn (n=0~4), there are 5 devices in total. Corresponding to COM0~COM4, serial port 0 is used as a Bluetooth data transmission channel and cannot be used as a normal serial port, and serial port 2 is used as a debugging serial port and cannot be used as a normal serial port. Therefore, the actual serial ports that users can use are COM1, COM3, and COM4 (if the user There is no need to debug the serial port, you can use 4 COM1~COM4 serial ports), where the ttySn file node corresponds to the physical serial port COMn.

3.3.2 Serial port read and write permissions

When the Android system operates a file node, it needs corresponding permissions. The same is true for reading and writing serial ports. The corresponding node needs to have read and write permissions. You can view the file permissions of ttySn through the command:

```
rk3288:/ # ls -l /dev/ttyS*

crw-rw-rw- 1 bluetooth net_bt_stack 4,          64 2017-01-01 20:41 /dev/ttyS0

crw-rw-rw- 1 bluetooth net_bt_stack 4,          65 2017-01-01 20:41 /dev/ttyS1

crw------- 1 root         root          4,      66 2017-01-01 20:41 /dev/ttyS2

crw-rw-rw- 1 root        root           4,      67 2017-01-01 20:41 /dev/ttyS3

crw-rw-rw- 1 root        root           4,      68 2017-01-01 20:41 /dev/ttyS4
```

You can see that there are read and write permissions for com1, com3, and com4. If there is no corresponding permission, you can use the chmod command (requires root permissions) to modify the file permissions

```
rk3288:/ # chmod 777 /dev/ttyS3
rk3288:/ # ls -l /dev/ttyS3
crwxrwxrwx 1 root root 4,      67 2017-01-01 20:41 /dev/ttyS3
```