



ED-IPC2000 系列

基于 Raspberry Pi CM4 的面向工业应用场景的计算机

上海晶珩电子科技有限公司

2023年6月

版权声明

ED-IPC2000 系列及其相关知识产权为上海晶珩电子科技有限公司所有。

上海晶珩电子科技有限公司拥有本文件的版权并保留所有权利。未经上海晶珩电子科技有限公司的书面许可，不得以任何方式和形式修改、分发或复制本文件的任何部分。

免责声明

上海晶珩电子科技有限公司不保证本手册中的信息是最新的、正确的、完整的或高质量的。上海晶珩电子科技有限公司也不对这些信息的进一步使用作出保证。如果由于使用或不使用本手册中的信息，或由于使用错误或不完整的信息而造成的物质或非物质相关损失，只要没有证明是上海晶珩电子科技有限公司的故意或过失，就可以免除对上海晶珩电子科技有限公司的责任索赔。上海晶珩电子科技有限公司明确保留对本手册的内容或部分内容进行修改或补充的权利，无需特别通知。

目 录

1	产品概述	1
1.1	目标应用	1
1.2	规格参数	1
1.3	系统框图	2
1.4	功能布局	3
1.5	包装清单	4
1.6	订购编码	5
2	快速启动	5
2.1	设备清单	5
2.2	硬件连接	5
2.3	首次启动	6
2.3.1	Raspberry Pi OS (Desktop)	6
2.3.2	Raspberry Pi OS (Lite)	9
2.3.3	使能 SSH 功能	10
2.3.4	查找设备 IP	10
3	接线指南	11
3.1	Panel I/O	11
3.1.1	micro-SD Card	11
3.2	Internal I/O	11
3.2.1	CSI	11
3.2.2	MIPI DSI	11
4	软件操作指引	12
4.1	USB	12
4.1.1	查看 USB 设备信息	12
4.1.2	USB 存储设备挂载	12
4.2	以太网配置	14
4.2.1	千兆以太网	14
4.2.2	使用 Network Manager 工具配置	14
4.2.3	使用 dhcpcd 工具配置	16
4.3	WiFi	17
4.3.1	使能 WiFi 功能	17
4.3.2	外置天线/内置 PCB 天线	18
4.3.3	AP 及桥接模式	18
4.4	蓝牙	18
4.4.1	基本用法	18
4.4.2	示例	19
4.5	RTC	19
4.6	LED 指示	20
4.7	Buzzer	21
4.8	加密芯片	21
4.9	摄像头	21
4.10	音频	21
4.10.1	查看声卡	21

4.10.2	录音	22
4.10.3	放音	22
4.11	串口通信	22
4.11.1	安装 picocom 工具	22
4.11.2	Debug UART	23
5	操作系统安装	24
5.1	镜像下载	24
5.2	eMMC 烧录	25
5.2.1	工具下载	25
5.2.2	烧录	25
5.3	基于原版 Raspberry Pi OS 在线安装 BSP	26
5.4	BSP 工具	26
5.4.1	配置设备 BSP	27
5.4.2	配置扩展板 BSP	27
5.4.3	报错信息	27
6	故障排除	28
7	FAQ	28
7.1.1	默认用户名密码	28
8	关于我们	28
8.1	关于 EDATEC	28
8.2	联系方式	28

1 产品概述

ED-IPC2000 系列是一款基于 Raspberry Pi CM4 的面向工业应用场景的计算机。ED-IPC2000 系列与 Raspberry Pi 软硬件兼容，外形尺寸比 Pi4 略大，对工业应用方面做了很多增强，在外部增加了铝合金外壳大大提高了散热性能，板上额外增加了加密芯片以及 RTC 等常用模块。

1.1 目标应用

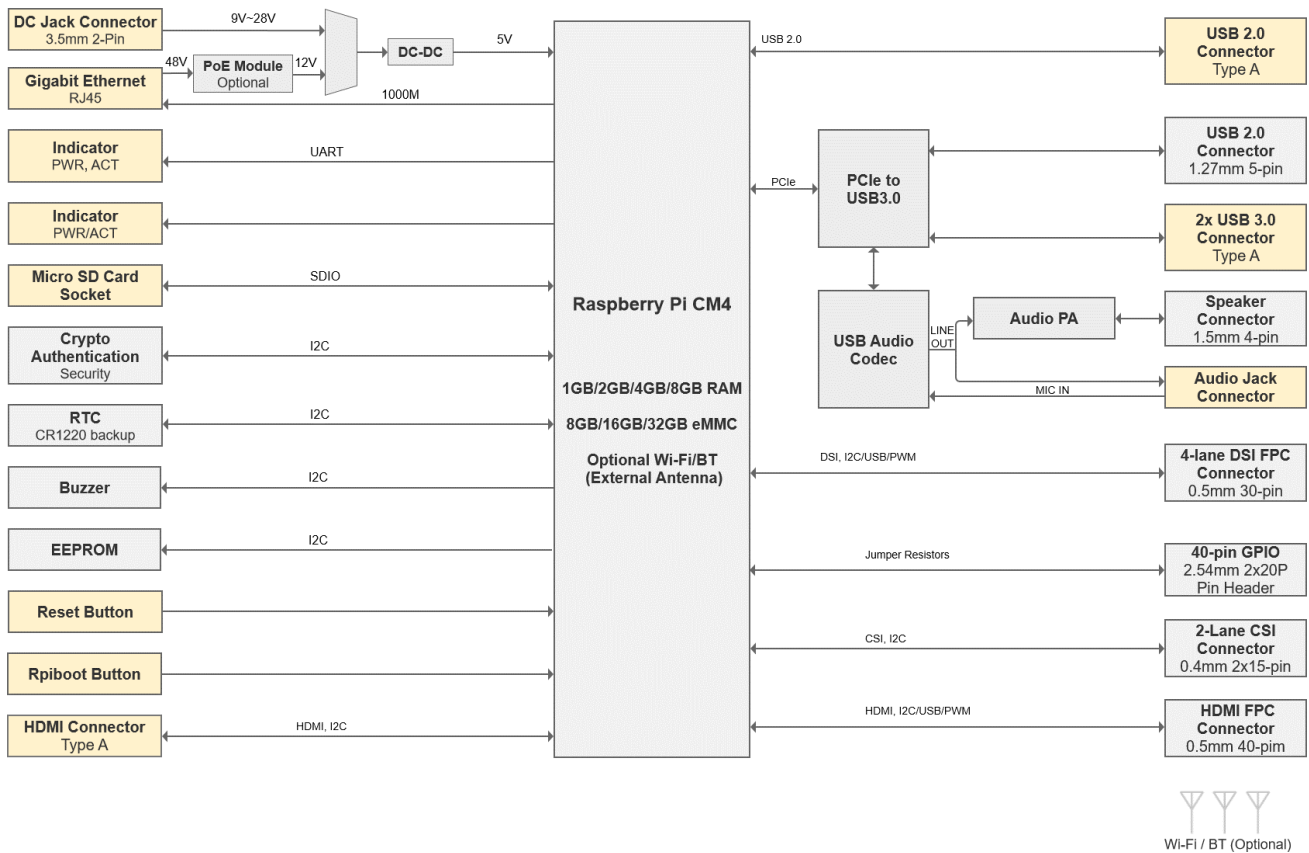
- 多媒体娱乐
- 人工智能
- 智能仪表
- 全景显示
- 智慧生活

1.2 规格参数

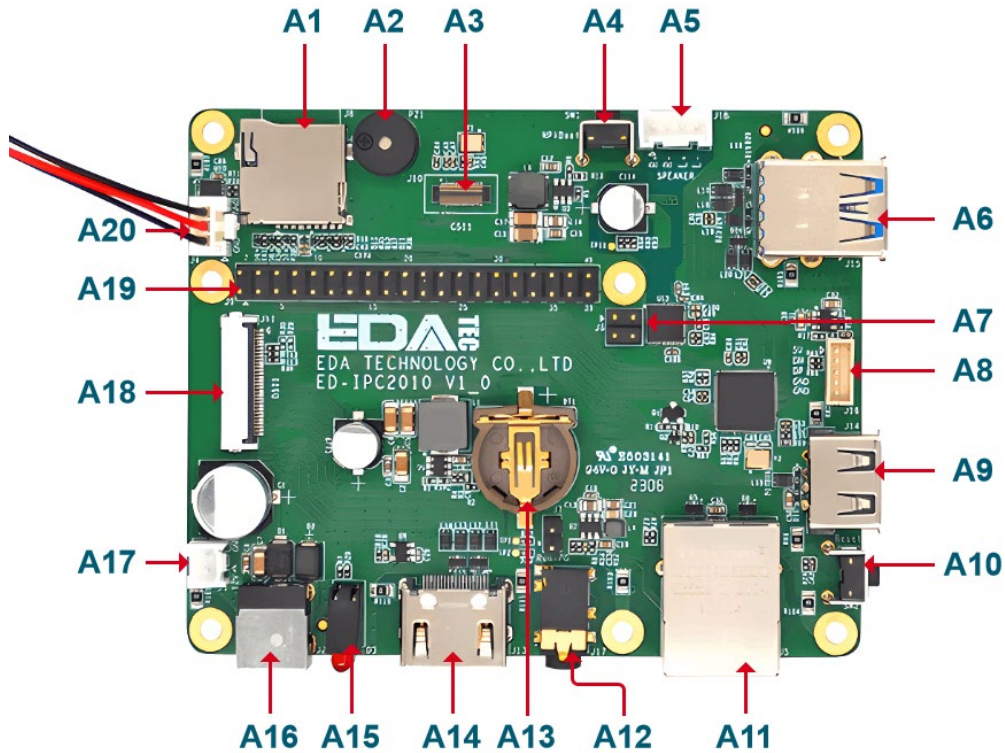
功能	参数
CPU	Broadcom BCM2711 4 核 Cortex A72 1.5GHz (ARM v8) 64-bit CPU
RAM	1GB/2GB/4GB/8GB 可选
eMMC	8GB/16GB/32GB 可选
WiFi/BT	2.4G/5.8G WiFi, Bluetooth BLE5.0
以太网接口	10/100/1000M 以太网, 支持 PoE
SD 卡	支持 micro SD 卡作为扩展存储使用
HDMI	1x 标准 HDMI Type A
HDMI FPC 接口(可选)	1x FPC HDMI + 1x USB 触摸, 支持 12 寸、15 寸、19 寸触摸屏 注: 仅 ED-IPC2020 包含此接口
USB	2x 标准 USB 3.0 接口 1x 标准 USB 2.0 接口
DSI(可选)	1x DSI 接口, 支持 7 寸、10 寸触摸屏 注: 仅 ED-IPC2020 包含此接口
CSI(可选)	1x CSI 接口, 支持摄像头功能 注: 仅 ED-IPC2020 包含此接口
实时时钟	支持 RTC 功能
安全芯片	板载加密芯片 ATECC608
蜂鸣器	1x 蜂鸣器
输出电源	5V@1A 线对板连接器输出 12V@1A 线对板连接器输出
音频(可选)	3.5mm 耳机接口, 支持立体声播放, 耳机麦克风录音 注: 仅 ED-IPC2020 包含此接口

功能	参数
扬声器(可选)	两对线对板连接器喇叭接口 注：仅 ED-IPC2020 包含此接口
40PIN 接口	支持 Raspberry Pi 标准 40PIN 接口
PoE	支持 PoE 接口
Rpiboot 按键	设置 CM4 进入烧录模式
复位按键	CM4 复位按键
LED	绿(系统状态), 红(电源)
输入电源	9 ~ 28V 支持 DC 接口, 选配凤凰端子接口
尺寸	103(长) x 80(宽) x 35(高) mm
外壳	全金属外壳, 支持 DIN 导轨安装
天线	PCB 天线/外置天线
工作温度范围	-25~60°C
OS	兼容 Raspberry Pi OS 官方标准系统

1.3 系统框图

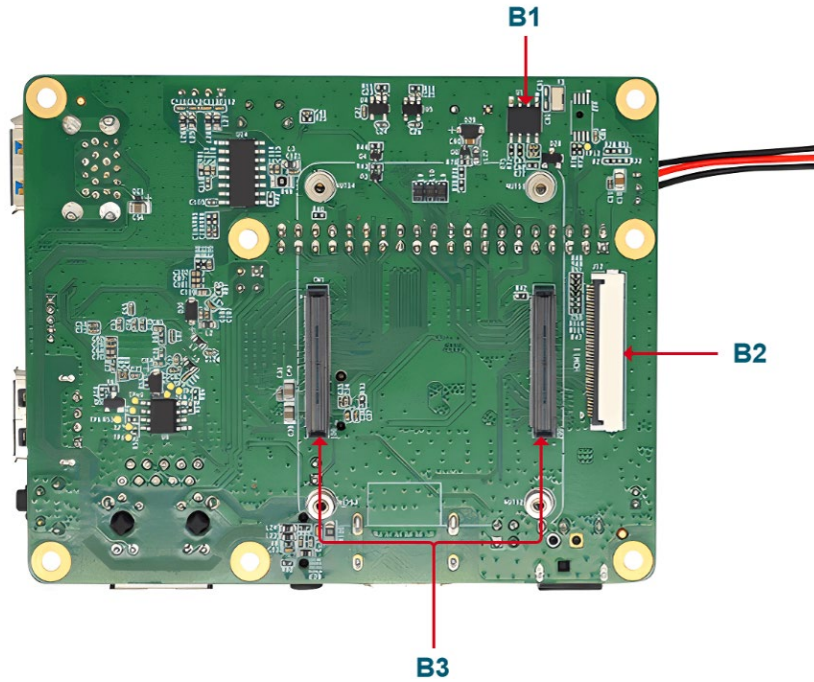


1.4 功能布局



编号 ^注	功能说明	编号	功能说明
A1	Micro SD Card	A11	网口
A2	蜂鸣器	A12	3.5mm 耳机接口
A3	CSI	A13	CR1220
A4	rpiboot 按键	A14	HDMI
A5	扬声器	A15	指示灯
A6	USB 3.0	A16	电源
A7	PoE	A17	12V 输出
A8	USB 2.0	A18	DSI
A9	USB 2.0	A19	40PIN
A10	复位按键	A20	5V 输出

注：仅 ED-IPC2020 包含 A3、A5、A12 和 A18 接口

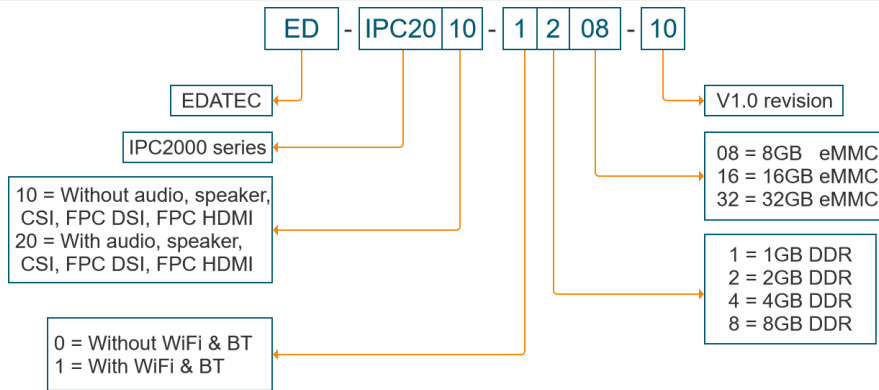


编号 ^注	功能说明	编号	功能说明
B1	RTC	B3	CM4
B2	FPC HDMI		
注：仅 ED-IPC2020 包含 B2 接口。			

1.5 包装清单

- 1x ED-IPC2000 系列主机
- [选配 WiFi/BT 版本]1x WIFI/BT 天线

1.6 订购编码



Example

Part# : ED-IPC2010-1208-10

Configuration : CM4 Industrial Computer V1.0 version
 ED-IPC2010-1208-10 Module with Wireless, 2GB DDR & 8GB eMMC
 1pcs Raspberry Pi certified WiFi/Bluetooth Antenna
 Assembled by a metal case

2 快速启动

本章介绍 ED-IPC2000 系列的启动及部分开机设置。

2.1 设备清单

- 1x ED-IPC2000 系列主机
- 1x WIFI/BT 天线
- 1x 鼠标
- 1x 键盘
- 1x HDMI 显示器
- 1x 网线
- 1x 12V@2A DC 电源

2.2 硬件连接

1. 将天线安装到设备顶部天线连接器处。
2. 插入网线、键盘、鼠标
3. 插入 HDMI，另一端连接到显示器
4. 给显示器上电
5. ED-IPC2000 系列没有电源开关，插入电源线，系统将会开始启动。
6. 给 12V@2A 电源适配器供电，并插入 ED-IPC2000 系列的 DC 电源输入口(标注 +12V DC)

7. 红色 LED 灯亮起，代表电源正常供电
8. 绿灯开始闪烁，表示系统正常启动，然后屏幕的左上角会出现 Raspberry 的 logo.
9. 系统启动完成后，进入到 Desktop，若提示登录，请输入用户名: pi, 默认密码:raspberrypi

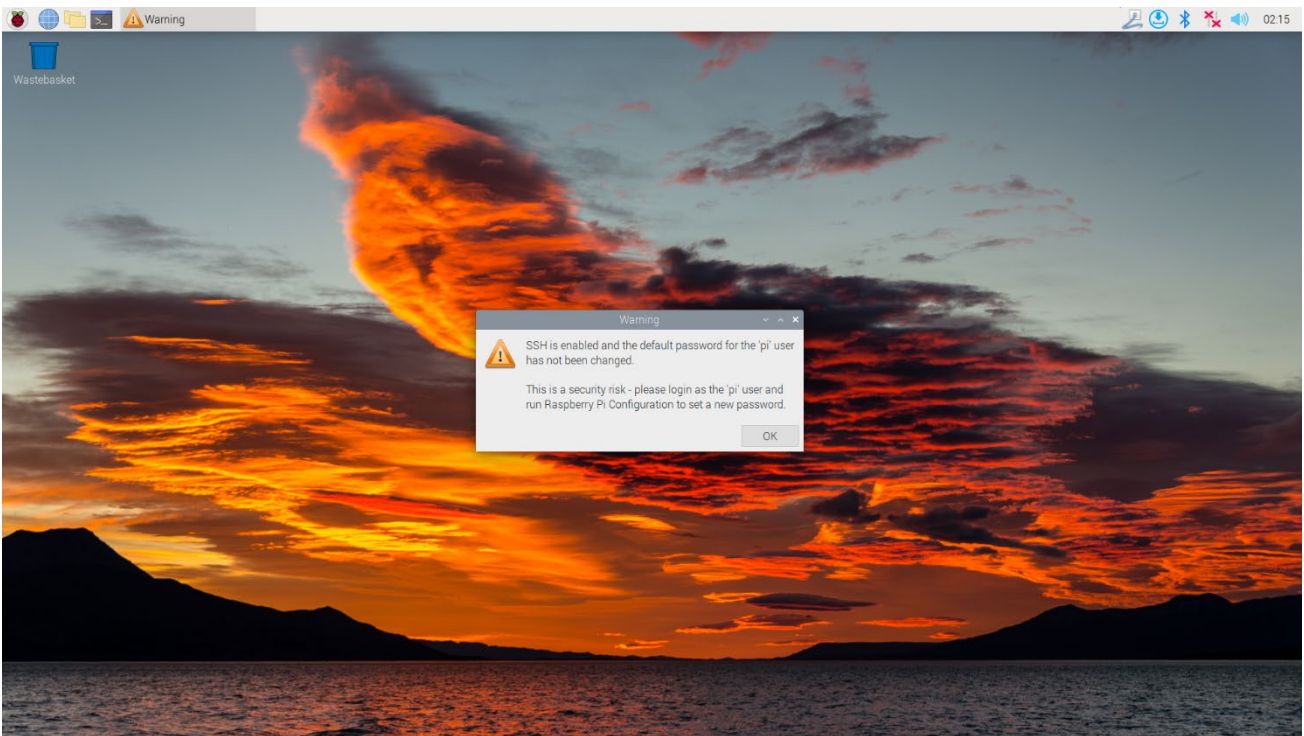
2.3 首次启动

ED-IPC2000 系列没有电源开关，插入电源线，系统将会开始启动。

- 1.给 12V@2A 电源适配器供电，并插入 ED-IPC2000 系列的 DC 电源输入口(标注 +12V DC)
- 2.红色 LED 灯亮起，代表电源正常供电
- 3.绿灯开始闪烁，表示系统正常启动，然后屏幕的左上角会出现 Raspberry 的 logo.

2.3.1 Raspberry Pi OS (Desktop)

Desktop 版系统启动完成后，直接进入桌面。



如果您使用官方系统镜像，并且烧录前没有配置镜像，首次启动时，Welcome to Raspberry Pi 应用会弹出并且指导完成初始化设置。



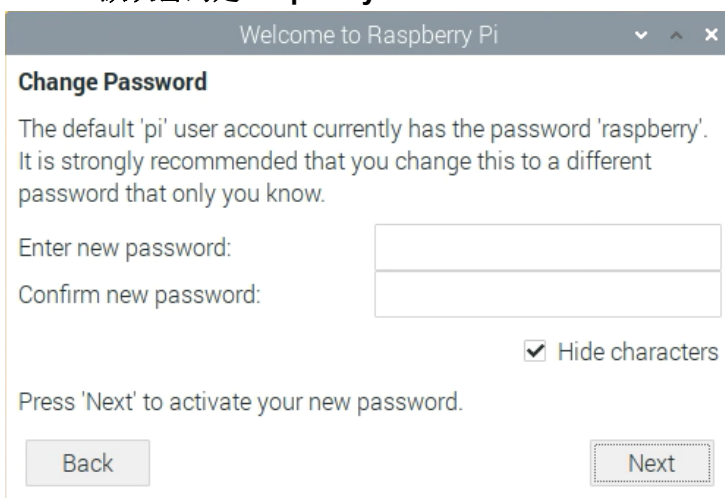
- 点击 **Next** 开始进行设置。
- 设置 **Country**, **Language** 和 **Timezone**, 再次点击 **Next**。

NOTE: 您需要国家区域, 否则系统默认的键盘布局是英国键盘布局 (我们国内的键盘一般是美国键盘布局), 有一些特殊符号可能打不出来。

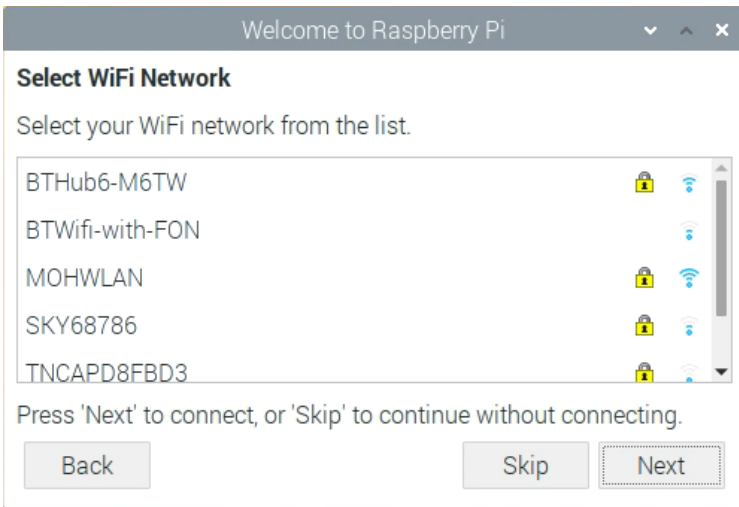


- 输入新的密码, 用于默认账户 **pi**,再次点击 **Next**。

NOTE: 默认密码是 **raspberrry**



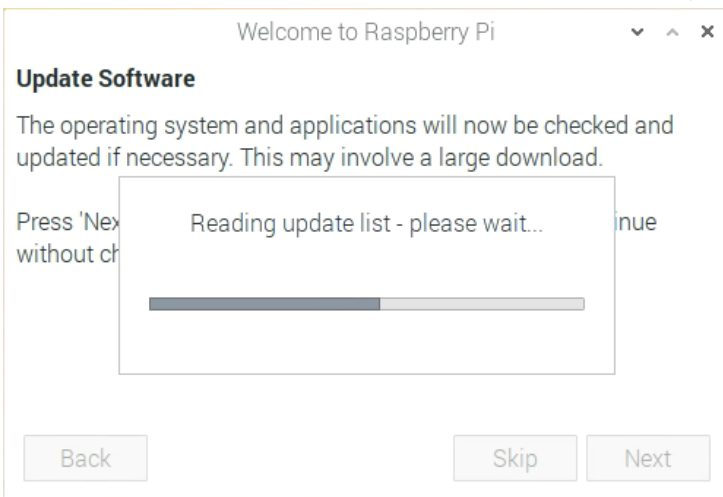
- 选择需要连接的无线网络，输入密码后，再次点击 **Next**。



注: 如果你的 **CM4** 模组不带 **WIFI** 模块，将没有此步骤。

注: 升级系统之前，需要等待 **wifi** 连接正常（右上角 **wifi** 图标出现）。

- 点击 **Next**，向导程序将会自动检查和更新 **Raspberry Pi OS**。



- 点击 **Restart** 完成系统更新。



2.3.2 Raspberry Pi OS (Lite)

如果您使用我们提供的系统镜像，系统启动后会使用用户名 pi 自动登入，默认密码为 raspberry。

```
[ OK ] Started User Login Management.
[ OK ] Finished Permit User Sessions.
[ OK ] Started Getty on tty1.
[ OK ] Reached target Login Prompts.
[ OK ] Started OpenBSD Secure Shell server.
[ OK ] Started Modem Manager.
[ OK ] Started Hostname Service.
      Starting Network Manager Script Dispatcher Service...
[ OK ] Started Network Manager Script Dispatcher Service.
[ OK ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
      Starting Load/Save RF Kill Switch Status...
[ OK ] Started LSB: Switch to on@ (unless shift key is pressed).
[ OK ] Started Load/Save RF Kill Switch Status.
      Starting Save/Restore Sound Card State...
[ OK ] Finished Save/Restore Sound Card State.
[ OK ] Reached target Sound Card.

Debian GNU/Linux 11 raspberrypi tty1
raspberrypi login: pi (automatic login)

Linux raspberrypi 5.15.32-v8+ #1538 SMP PREEMPT Thu Mar 31 19:40:39 BST 2022 aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

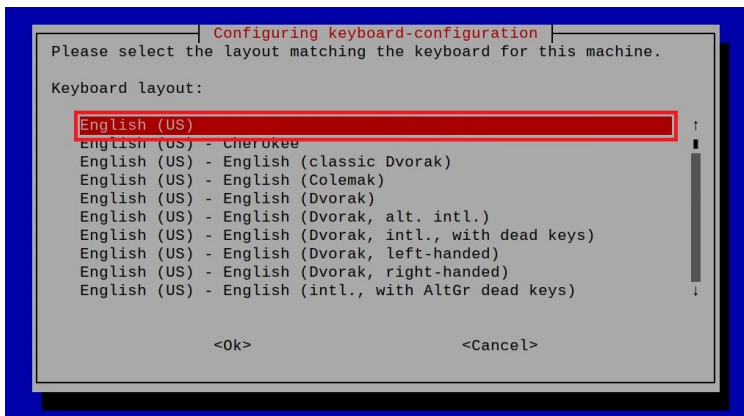
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Jan 31 03:52:21 GMT 2023 from 192.168.168.211 on pts/0

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

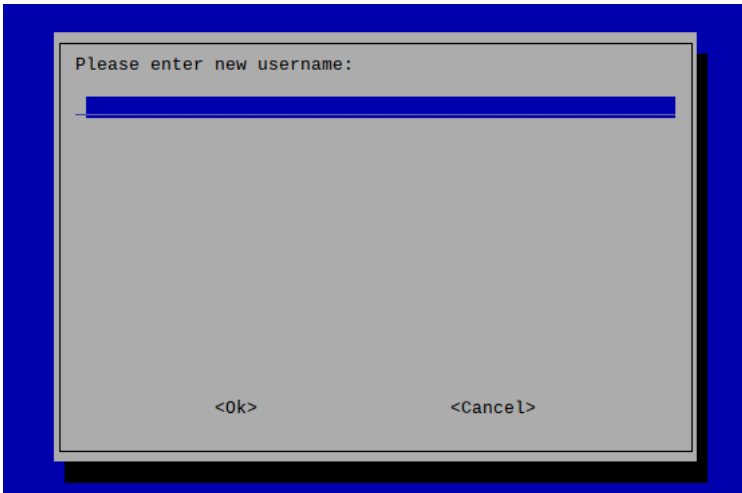
pi@raspberrypi:~$
```

如果您使用官方系统镜像，并且烧录前没有配置镜像，首次启动时，会出现配置窗口，需要依次配置键盘布局，设置用户名及对应密码。

- 设置配置键盘布局



- 创建新用户名



然后按提示设置用户对应的密码，并再次输入密码进行确认。至此您就可以使用刚才设置的用户名及密码进行登录了。

2.3.3 使能 SSH 功能

我们提供的镜像都已经打开了 SSH 功能，如果使用官方镜像则需要使用一下方法打开 SSH 功能。

2.3.3.1 raspi-config

```
sudo raspi-config
```

1. 选择 3 Interface Options
2. 选择 I2 SSH
3. Would you like the SSH server to be enabled? 选 Yes
4. 选择右下角 Finish

2.3.3.2 添加空文件使能 SSH

在 boot 分区中放入一个名为 ssh 的空文件，设备上电后将会自动使能 SSH 功能。

2.3.4 查找设备 IP

- 设备开启如果接有显示屏可以使用 ifconfig 命令查看当前设备 IP
 - 如果没有显示屏，则可以通过路由器查看分配的 IP
 - 如果没有显示屏，则可以下载 Nmap 工具扫描当前网络下的 IP
- nmap 支持 Linux、macOS、Windows 等多个平台。如果希望使用 nmap 扫描 192.168.3.0~255 的网段，则可以使用以下命令：

```
nmap -sn 192.168.3.0/24
```

等待一段时间后即会输出结果，类似与下方输出：

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-12-30 21:19 中国标准时间
Nmap scan report for 192.168.3.1 (192.168.3.1)
Host is up (0.0010s latency).
MAC Address: XX:XX:XX:XX:XX:XX (Phicomm (Shanghai))
Nmap scan report for DESKTOP-FGEOUUK.lan (192.168.3.33)
```



```
Host is up (0.0029s latency).
MAC Address: XX:XX:XX:XX:XX:XX (Dell)
Nmap scan report for 192.168.3.66 (192.168.3.66)
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 11.36 seconds
```

3 接线指南

3.1 Panel I/O

3.1.1 micro-SD Card



3.2 Internal I/O

3.2.1 CSI(可选)

J10 为 CSI 接口，可使用我司摄像头直接安装。

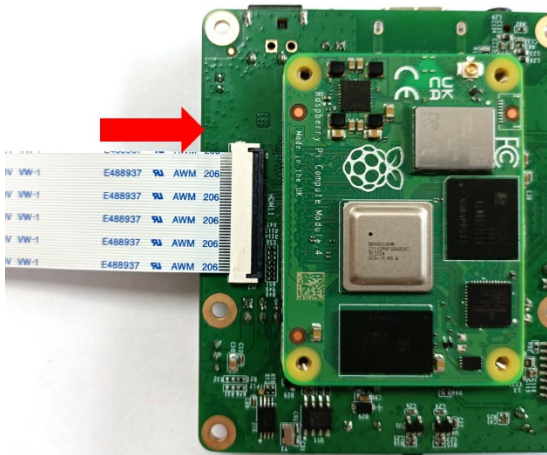


提示：仅 ED-IPC2000 包含此接口。

3.2.2 MIPI DSI(可选)

J11 为 MIPI DSI 接口。

DSI 接口请使用 15pin 1mm 间距单面 FPC 连接线进行连接，金属触点面朝上，沿垂直于 FPC 连接器方向插入，如下图所示。



提示：仅 ED-IPC2000 包含此接口。

4 软件操作指引

4.1 USB

4.1.1 查看 USB 设备信息

显示 USB 设备

```
lsusb
```

显示信息如下：

```
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 005: ID 1a2c:2d23 China Resource Semico Co., Ltd Keyboard
Bus 001 Device 004: ID 30fa:0300 USB OPTICAL MOUSE
Bus 001 Device 003: ID 0424:9e00 Microchip Technology, Inc. (formerly SMSC)
LAN9500A/LAN9500Ai
Bus 001 Device 002: ID 1a40:0201 Terminus Technology Inc. FE 2.1 7-port Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

4.1.2 USB 存储设备挂载

您可以将外部硬盘、SSD 或 USB 棒连接到 Raspberry Pi 上的任何 USB 端口，并挂载文件系统以访问存储在其上的数据。

默认情况下，您的 Raspberry Pi 会在 `/media/pi/HARD-DRIVE-LABEL` 位置自动挂载一些流行的文件系统，如 FAT、NTFS 和 HFS+。

对于一般情况，您可以直接使用如下命令挂载或卸载外置存储设备。


```
lsblk
```

```
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda            8:0    1 29.1G  0 disk
├─sda1         8:1    1 29.1G  0 part
mmcbk0        179:0    0 59.5G  0 disk
├─mmcbk0p1    179:1    0 256M  0 part /boot
└─mmcbk0p2    179:2    0 59.2G  0 part /
```

使用 `mount` 命令来挂载 `sda1` 到 `/mnt` 目录，挂载完成后用户可以直接在 `/mnt` 目录下操作存储设备。

```
sudo mount /dev/sda1 /mnt
```

使用完成以后使用命令 `umount` 卸载存储设备。

```
sudo umount /mnt
```

4.1.2.1 挂载

您可以将存储设备安装在特定的文件夹位置。通常在 `/mnt` 文件夹中进行，例如 `/mnt/mydisk`。请注意，文件夹必须是空的。

1. 将存储设备插入设备上的 USB 端口。
2. 使用以下命令列出 Raspberry Pi 上的所有磁盘分区：

```
sudo lsblk -o UUID,NAME,FSTYPE,SIZE,MOUNTPOINT,LABEL,MODEL
```

Raspberry Pi 使用挂载点 `/` 和 `/boot`。您的存储设备将显示在此列表中，以及任何其他连接的存储设备。

3. 使用“大小”、“标签”和“型号”列来标识指向您的存储设备的磁盘分区的名称。例如，`sda1`。
4. `FSTYPE` 列包含文件系统类型。如果您的存储设备使用 `exFAT` 文件系统，请安装 `exFAT` 驱动程序：

```
sudo apt update
sudo apt install exfat-fuse
```

5. 如果您的存储设备使用 `NTFS` 文件系统，您将对其拥有只读访问权限。如果要写入设备，可以安装 `ntfs-3g` 驱动程序：

```
sudo apt update
sudo apt install ntfs-3g
```

6. 运行以下命令获取磁盘分区的位置：

```
sudo blkid
```

比如显示，`/dev/sda1`

7. 创建一个目标文件夹作为存储设备的装载点。本例中使用的挂载点名称是 `mydisk`。您可以指定自己选择的名称：

```
sudo mkdir /mnt/mydisk
```

8. 在您创建的装载点装载存储设备：

```
sudo mount /dev/sda1 /mnt/mydisk
```

9. 通过列出以下内容来验证存储设备是否已成功装载：

```
ls /mnt/mydisk
```

WARN: 无桌面系统将不会自动挂载外置存储设备

4.1.2.2 卸载

当设备关闭时，系统会负责卸载存储设备，以便安全地将其拔出。如果您想要手动卸载设备，可以使用以下命令：

```
sudo umount /mnt/mydisk
```

如果您收到“目标繁忙”的错误，这意味着存储设备未卸载。如果没有显示错误，您现在可以安全地拔出设备。

4.1.2.3 命令行中设置自动挂载

可以通过修改 `fstab` 设置自动挂载。

1. 首先需要获取磁盘 UUID

```
sudo blkid
```

2. 找到挂载设备的 UUID，例如 `5C24-1453`
3. 打开 `fstab` 文件

```
sudo nano /etc/fstab
```

4. 添加以下内容到 `fstab` 文件中

```
UUID=5C24-1453 /mnt/mydisk fstype defaults,auto,users,rw,nofail 0 0
```

将 `fstype` 替换为您的文件系统的类型，您可以在上面的“挂载存储设备”的步骤 2 中找到，例如：`ntfs`。

5. 如果文件系统类型是 FAT 或 NTFS，则在 `nofail` 之后立即添加 `umask = 000` 这将允许所有用户对存储设备上的每个文件进行完全读/写访问。

关于更多 `fstab` 命令的信息可以使用 `man fstab` 来查看。

4.2 以太网配置

4.2.1 千兆以太网

ED-IPC2000 系列上有一路自适应 10 / 100 / 1000Mbps 以太网接口，与 DC 电源座子相邻，建议使用 Cat6(六类)网线配合使用。系统默认使用 DHCP 自动获取 IP。


4.2.2 使用 Network Manager 工具配置

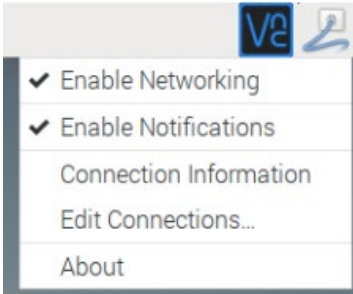
如果使用桌面版镜像，推荐安装 NetworkManager 插件 `network-manager-gnome`，安装后即可通过桌面图标直接配置网络。

```
sudo apt update
sudo apt install network-manager-gnome
sudo reboot
```

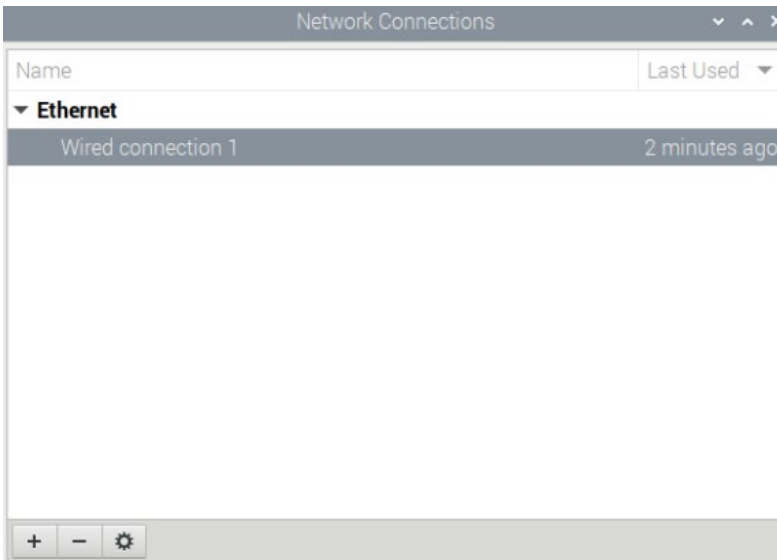
NOTE: 若使用我们的出厂镜像，默认已安装 `network-manager` 工具及 `network-manager-gnome` 插件。

NOTE: 若使用我们的出厂镜像，默认已自动启动 `NetworkManager` 服务并禁用了 `dhcpcd` 服务。

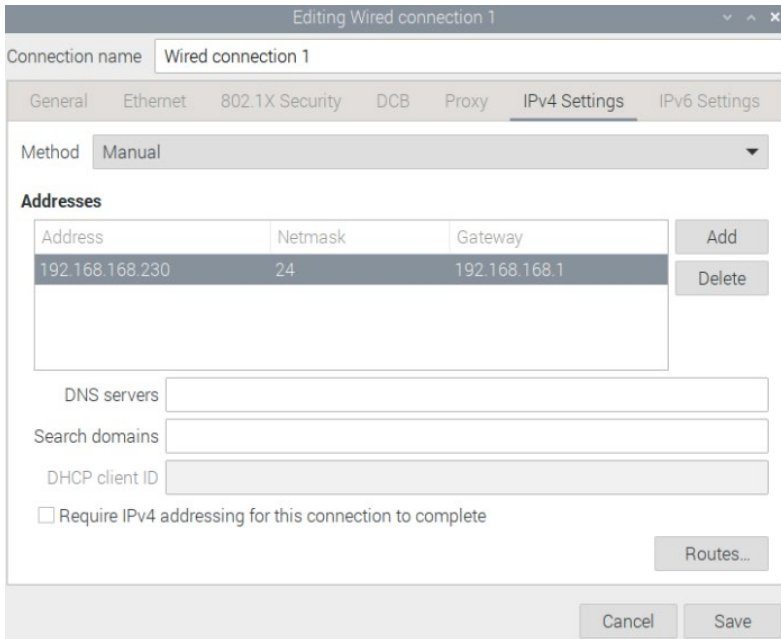
安装完成后，在系统桌面状态栏会看到 NetworkManager 图标 。
右键单击 NetworkManager 图标，选择 Edit Connections。



选中要修改的连接名称，然后点击下方齿轮。



切换至 IPv4 Settings 配置页，如果要设置静态 IP，Method 选择 Manual，将 Addresses 您要配置的 IP，如果要设置为动态获取 IP，只需配置 Method 为 Automatic(DHCP)，重启设备后即可。



如果使用 Lite 版系统，可以通过命令行方式进行配置。

如果希望使用命令为设备设置静态 IP，则可以参考以下方法。

设置静态 IP

```
sudo nmcli connection modify <name> ipv4.addresses 192.168.1.101/24 ipv4.method manual
```

设置网关地址

```
sudo nmcli connection modify <name> ipv4.gateway 192.168.1.1
```

设置动态获取 IP

```
sudo nmcli connection modify <name> ipv4.method auto
```

4.2.3 使用 dhcpcd 工具配置

Raspberry Pi 官方系统默认使用的 dhcpcd 作为网络管理工具。

如果您使用的我们提供的出厂镜像并希望从 NetworkManager 切换为 dhcpcd 网络管理工具，首先需要停止并禁用 NetworkManager 服务并启用 dhcpcd 服务。

```
sudo systemctl stop NetworkManager
sudo systemctl disable NetworkManager
sudo systemctl enable dhcpcd
sudo reboot
```

系统重启后即可使用 dhcpcd 工具。

设置静态 IP 通过修改/etc/dhcpd.conf 来设置，示例设置 eth0，用户可以根据自己的不同需要设置 wlan0 等网络接口。

```
interface eth0
static ip_address=192.168.0.10/24
static routers=192.168.0.1
static domain_name_servers=192.168.0.1 8.8.8.8 fd51:42f8:caae:d92e::1
```

4.3 WiFi(可选)

客户可选购带 WiFi 版本的 ED-IPC2000 系列，支持 2.4 GHz，5.0 GHz IEEE 802.11 b/g/n/ac 双频 WiFi。我们提供双频外置天线，已与 Raspberry Pi CM4 一起通过无线认证。

4.3.1 使能 WiFi 功能

WiFi 功能默认是屏蔽的，您首先需要设置国家区域后才可使用。若使用桌面版系统，可参考章节：初始化设置配置 WiFi，若使用 Lite 版系统，请使用 raspi-config 设置 WiFi 国家区域，参考文档：["Raspberry Pi 官方文档 - Using the Command Line"](#)

4.3.1.1 使用 Network Manager 工具配置

安装桌面插件后可以直接通过桌面图标连接到 WIFI 网络。

也可以使用命令行方式，执行如下命令：

扫描附近的 WIFI

```
sudo nmcli device wifi
```

连接带密码的 WIFI

```
sudo nmcli device wifi connect <SSID> password <password>
```

设置 WIFI 自动连接

```
sudo nmcli connection modify <name> connection.autoconnect yes
```

4.3.1.2 使用 dhcpd 工具配置

Raspberry Pi 官方系统默认使用的 dhcpd 作为网络管理工具。

```
sudo raspi-config
```

1. 选择 1 System Options
2. 选择 S1 Wireless LAN
3. 在 Select the country in which the Pi is to be used 窗口中选择您的国家，然后选择 OK，此提示仅在第一次设置 WIFI 时出现
4. Please enter SSID，输入 WIFI SSID 名称
5. Please enter passphrase. Leave it empty if none，输入密码，然后重启设备即可

4.3.2 外置天线/内置 PCB 天线

可以通过软件配置，来切换使用外置天线还是内置 PCB 天线。考虑到兼容性和最广泛的支持性，出厂默认系统是选用的内置 PCB 天线，若客户选配带外壳的整机，并配置了外置天线，可以通过以下操作来切换：

编辑/boot/config.txt

```
sudo nano /boot/config.txt
```

选择外置天线，加入

```
dtparam=ant2
```

然后重启设备后生效。

4.3.3 AP 及桥接模式

ED-IPC2000 系列上的 WiFi 还支持配置为:AP 路由器模式、桥接模式或者混合模式。

请参考开源项目 [github: garywill/linux-router](https://github.com/garywill/linux-router) 了解如何配置。

4.4 蓝牙(可选)

ED-IPC2000 系列可选配是否集成了蓝牙功能，如果带蓝牙的话，此功能默认是开启的。

可使用 bluetoothctl 扫描，配对，连接蓝牙设备，请参考 [ArchLinux-Wiki-Bluetooth](#) 指引配置和使用蓝牙。

4.4.1 基本用法

扫描：

```
bluetoothctl scan on/off
```

发现：

```
bluetoothctl discoverable on/off
```

信任设备：

```
bluetoothctl trust [MAC]
```

连接设备：

```
bluetoothctl connect [MAC]
```

断开设备：

```
bluetoothctl disconnect [MAC]
```

4.4.2 示例

进入 bluetooth shell

```
sudo bluetoothctl
```

使能 Bluetooth

```
power on
```

扫描设备

```
scan on
```

```
Discovery started
```

```
[CHG] Controller B8:27:EB:85:04:8B Discovering: yes
```

```
[NEW] Device 4A:39:CF:30:B3:11 4A-39-CF-30-B3-11
```

查找开启的蓝牙设备名称，这里测试开启的蓝牙设备名称是 test

```
devices
```

```
Device 6A:7F:60:69:8B:79 6A-7F-60-69-8B-79
```

```
Device 67:64:5A:A3:2C:A2 67-64-5A-A3-2C-A2
```

```
Device 56:6A:59:B0:1C:D1 Lefun
```

```
Device 34:12:F9:91:FF:68 test
```

配对 test 设备

```
pair 34:12:F9:91:FF:68
```

```
Attempting to pair with 34:12:F9:91:FF:68
```

```
[CHG] Device 34:12:F9:91:FF:68 ServicesResolved: yes
```

```
[CHG] Device 34:12:F9:91:FF:68 Paired: yes
```

```
Pairing successful
```

连接蓝牙

```
trust 34:12:F9:91:FF:68
```

```
[CHG] Device 34:12:F9:91:FF:68 Trusted: yes
```

```
Changing 34:12:F9:91:FF:68 trust succeeded
```

4.5 RTC

ED-IPC2000 系列上集成有 RTC，对于国内销售的版本，我们出货会默认附带安装 CR1220 纽扣电池 (RTC 备份电源)。这样，可以保障系统有一个不间断的可靠的时钟，不受设备下电等因素影响。

默认出货系统镜像，会集成我们编写的 RTC 自动同步服务，客人无需设置，即可自动同步时钟，可无感使用 RTC。大概的原理是：

- 系统开机时，服务自动从 RTC 读出保存的时间，并同步到系统时间。
- 若有连接互联网，系统会自动从 NTP 服务器同步时间，使用互联网时间更新本地系统时间。
- 系统关机时，服务自动把系统时间写入 RTC，更新 RTC 的时间。
- 因为有安装纽扣电池，尽管 ED-IPC2000 系列下电，但是 RTC 仍在工作计时。

这样，可以保证我们的时间是准确可靠的。

WARN: 若是第一次开机，因为 RTC 中无有效时间，可能会同步失败，直接重启即可。重启的时候，会把系统时间写入 RTC，后续正常使用。

若您不想用此服务，可手动关闭：

```
sudo systemctl disable rtc
sudo reboot
```

重新使能此服务：

```
sudo systemctl enable rtc
sudo reboot
```

手动读取 RTC 的时间：

```
sudo hwclock -r
2022-11-09 07:07:30.478488+00:00
```

手动同步 RTC 时间到系统：

```
sudo hwclock -s
```

把系统时间写入 RTC：

```
sudo hwclock -w
```

问题排查：

请首先看是否有 rtc 设备(/dev/rtc0)加载：

```
ls /dev/rtc0
```

如果没有，可能是您使用了官方标准系统，但是没有安装我们的 BSP 包，请参考章节：["基于原版 Raspberry Pi OS 在线安装 BSP"](#)，另外，您同样需要安装 ed-rtc 包使能 RTC 自动同步功能。

其他可能的检查点：

- CR1220 纽扣电池有没有安装
- NTP 网络时间协议，需要连接互联网才可自动同步时间，另外，需要开放端口(UDP, 123)，否则同步失败

4.6 LED 指示

ED-IPC2000 系列有两个指示灯，红色 LED 与 CM4 的 LED_PI_nPWR 管脚相连，为电源指示灯，绿色

LED 与 CM4 的 LED_PI_nACTIVITY 管脚相连，为运行状态指示灯。

4.7 Buzzer

蜂鸣器使用 GPIO6 来控制。

打开蜂鸣器：

```
raspi-gpio set 6 op dh
```

关闭蜂鸣器：

```
raspi-gpio set 6 op dl
```

4.8 加密芯片

安装 `atecc` 工具：[atecc-util](#)

关于更多加密芯片用法请参考链接中 README 文档。

4.9 摄像头(可选)

如果设备安装了摄像头则可以使用命令：

检查摄像头

```
libcamera-hello
```

摄像头拍照

```
libcamera-jpeg -o test.jpg
```

录制 10s 的视频

```
libcamera-vid -t 10000 -o test.h264
```

播放刚刚录制的视频

```
vlc test.h264
```

提示：仅 ED-IPC2020 才会安装摄像头。

4.10 音频(可选)

提示：仅 ED-IPC2020 包含此接口。

4.10.1 查看声卡

用户可以使用以下命令来查看声卡设备：

```
aplay -l
```

4.10.2 录音

用户可以使用以下命令录音：

```
arecord -fcd test.mp3
```

当然也支持用户指定声卡设备录音：

```
arecord -fcd -Dhw:<声卡标号> test.mp3
```

对于 arecord 的更多用法，用户可以用以下命令查看：

```
man arecord
```

4.10.3 放音

用户可以使用以下命令播放声音：

```
aplay test.mp3
```

当然也支持用户指定声卡设备录音：

```
aplay -Dhw:<声卡标号> test.mp3
```

对于 aplay 的更多用法，用户可以用一下命令查看：

```
man aplay
```

4.11 串口通信

4.11.1 安装 picocom 工具

picocom 串口终端可以在 Linux 环境下十分方便的进行串口调试。

首先安装 picocom

```
sudo apt-get install picocom
```

您可以先键入 Ctrl+a，然后键入 Ctrl+h 以查看可用命令。

```
*** Picocom commands (all prefixed by [C-a])

*** [C-x] : Exit picocom
*** [C-q] : Exit without resetting serial port
*** [C-b] : Set baudrate
*** [C-u] : Increase baudrate (baud-up)
*** [C-d] : Decrease baudrate (baud-down)
*** [C-i] : Change number of databits
*** [C-j] : Change number of stopbits
```

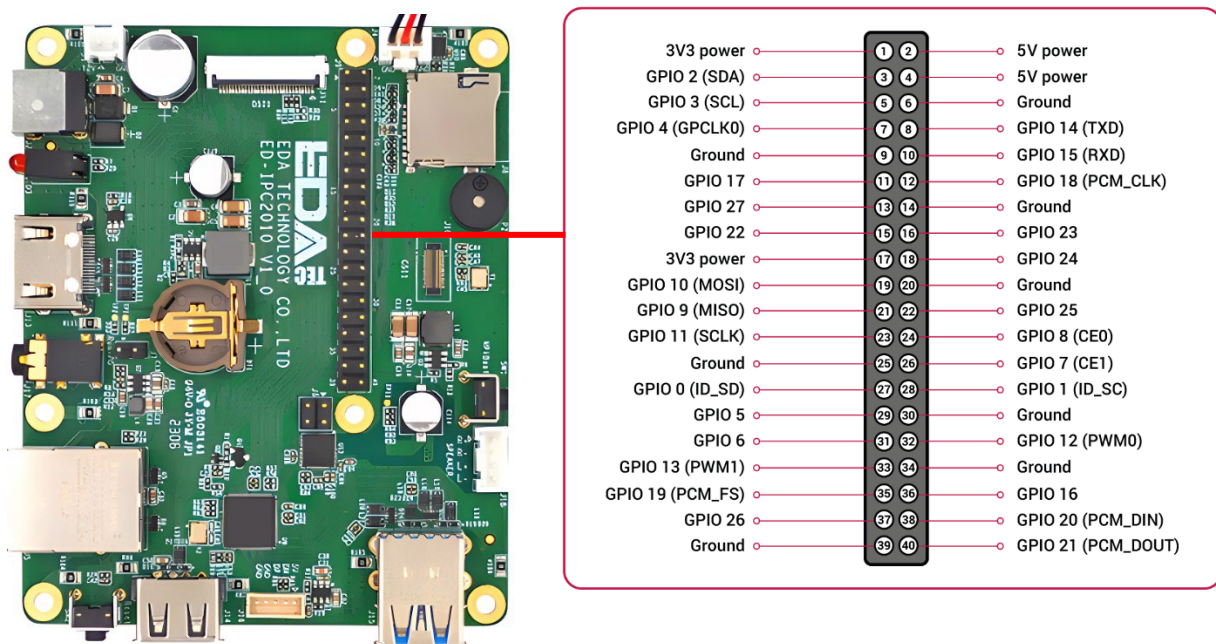
```

*** [C-f] : Change flow-control mode
*** [C-y] : Change parity mode
*** [C-p] : Pulse DTR
*** [C-t] : Toggle DTR
*** [C-g] : Toggle RTS
*** [C-] : Send break
*** [C-c] : Toggle local echo
*** [C-w] : Write hex
*** [C-s] : Send file
*** [C-r] : Receive file
*** [C-v] : Show port settings
*** [C-h] : Show this message
    
```

先键入 **Ctrl+a**，然后键入 **Ctrl+c** 以切换本地回显模式。

先键入 **Ctrl+a**，然后键入 **Ctrl+q** 即可退出 **picocom**。

4.11.2 Debug UART



ED-IPC2000 系列 40Pin 中拥有一个调试串口，GPIO14 GPIO15 分别为读写引脚。
此外用户还需要使用 **rspi-config** 工具设置为

1. 打开 **rspi-config**: `sudo rspi-config`.
2. 选择 **option 3 - Interface Options**.
3. 选择 **option P6 - Serial Port**.

4. 提示 Would you like a login shell to be accessible over serial? 回答 'Yes'
5. 退出 raspi-config
6. 重启设备生效: sudo reboot

要启用调试串口，需要修改 config.txt 配置文件。

```
sudo nano /boot/config.txt
```

在最后面添加

```
[all]
enable_uart=1
```

调试串口默认波特率为 115200，您可以通过 cmdline.txt 文件查看当前调试串口波特率

```
sudo nano /boot/cmdline.tx
```

5 操作系统安装

5.1 镜像下载

在出厂时我们已经在 eMMC 中烧录了系统，用户可以跳过本节直接使用。
我们提供了出厂镜像，如果系统恢复出厂设置，请点击以下链接下载出厂镜像。

默认登录信息：

- 用户名: pi
- 密码: raspberry
- SSH: 默认使能，端口使用默认 22

Raspberry Pi OS with desktop, 32-bit

- Release date: July 08nd 2022
- System: 32-bit
- Kernel version: 5.10
- Debian version: 11 (bullseye)
- Release notes

Raspberry Pi OS Lite, 32-bit

- Release date: July 08nd 2022
- System: 32-bit
- Kernel version: 5.10
- Debian version: 11 (bullseye)
- Release notes

Raspberry Pi OS with desktop, 64-bit

- Release date: July 08nd 2022
- System: 32-bit
- Kernel version: 5.10
- Debian version: 11 (bullseye)
- Release notes

Raspberry Pi OS Lite, 64-bit

- Release date: July 08nd 2022
- System: 32-bit
- Kernel version: 5.10
- Debian version: 11 (bullseye)
- Release notes

5.2 eMMC 烧录

5.2.1 工具下载

推荐烧录使用 Raspberry Pi 官方烧录工具：

Raspberry Pi Imager (https://downloads.raspberrypi.org/imager/imager_latest.exe)

SD Card Formatter (<https://www.sdcardformatter.com/download/>)

rpiboot(https://github.com/raspberrypi/usbboot/raw/master/win32/rpiboot_setup.exe)

烧录需要一根 USB-USB 转接线。一台已经安装烧录工具的电脑，以下以 Windows 系统为例。

5.2.2 烧录

1. 下载希望烧录的镜像
2. ED-IPC2000 系列断电，按下 rpiboot 按键，连接 USB2.0 到电脑，然后给设备上电（注意 USB2.0 旁边是复位按键不是 rpiboot 按键）



靠近 USB 2.0 的按键为复位按键，在 SD 卡一侧的按键为 rpiboot 按键



3. 打开 `rpiboot` 工具，等待盘符化完成，完成后会弹出盘符
4. 打开 `SD Card Formatter`，选择格式化的盘符，点击下方 `Format` 进行格式化
5. 格式化完成后打开 `Raspberry Pi Imager`，在第一项选择希望烧录的镜像，本地镜像选择自定义镜像，第二个选择存储卡
6. 设置完成后点击烧录，等待烧录结束
7. 烧录完成后取下 `USB` 连接线，重新上电即可

5.3 基于原版 Raspberry Pi OS 在线安装 BSP

如果使用 `Raspberry Pi` 官方镜像则需要安装产品 `BSP` 来使用其特有功能，使用我们提供的镜像则可以跳过这步。添加仓库及 `BSP` 的方式如下：

```
curl -s https://apt.edatec.cn/pubkey.gpg | sudo apt-key add -  
echo "deb https://apt.edatec.cn/raspbian stable main" | sudo tee /etc/apt/sources.list.d/edatec.list
```

安装 `rtc` `BSP`：

```
sudo apt update  
sudo apt install ed-rtc
```

安装 `NetworkManager` 网络管理工具[可选]

`NetworkManager` 工具可以更容易的配置路由规则和设置优先级

```
# 如果使用的 Raspberry Pi OS Lite 版系统  
sudo apt install ed-networkmanager  
  
# 如果您使用的系统带有桌面，我们推荐您安装插件 ed-networkmanager-gnome  
sudo apt install ed-networkmanager-gnome
```

5.4 BSP 工具

从 `ED-IPC2000` 系列开始，我们提供了一个新的 `BSP` 配置工具，此工具可以帮助用户一键配置 `dtoverlay`，而且在未来扩展板中，也支持一键配置。

安装 `BSP` 工具。

```
sudo apt update
```

```
sudo apt install ed-bsp-tool
```

5.4.1 配置设备 BSP

```
sudo eda-bsp-tool -i 10 -a 80 start
```

5.4.2 配置扩展板 BSP

```
# sudo eda-bsp-tool -i "i2c-bus" -a "address" start
```

用户可根据不同扩展板配置 BSP，-i 后边根据挂载的 I2C 总线更改，一般挂载在 I2C-10，假设地址为 0x50，转变为十进制则是 80，所以配置命令为以下写法：

```
sudo eda-bsp-tool -i 10 -a 80 start
```

5.4.3 报错信息

5.4.3.1 GPIO-x occupied

GPIOx 被占用，请确认是否有 GPIO 冲突。

5.4.3.2 Reuse i2c:x failed

i2c 引脚已经被使用，无法将对应引脚复用为 I2C。

5.4.3.3 Unable to reuse i2c x – addr x

I2C 地址已经被占用，无法再将设备设置为 x 地址。

5.4.3.4 Not found i2c x

无法找到/dev/i2c-x，可能是 I2C 功能未打开或 I2C 引脚被占用无法设置为 I2C。

5.4.3.5 Uart x cannot reuse

Uart x 无法复用，可能引脚已经被占用。

5.4.3.6 GPIO x has already been configured

GPIO x 已经被配置，无法再对 GPIO x 进行配置。

5.4.3.7 i2c-y 0x{:x} not exist

指定挂载在 I2c-y 总线上地址为 x 的设备不存在，请检查设备是否存在。

5.4.3.8 Not found x.dtbo file

此问题为 dtbo 文件缺失，如发现此问题请与我司售后同事联系。

5.4.3.9 x not contain uuid information

此问题为配置信息丢失无法完成自动配置 BSP，如发现此问题请与我司售后同事联系。

5.4.3.10 x not contain product information

此问题为配置信息丢失无法完成自动配置 BSP，如发现此问题请与我司售后同事联系。

6 故障排除

1. 为什么烧写官方的系统，但有些功能用不了？

您需要使用我们提供的系统镜像，或者在官方系统上安装我们的 BSP 包才可以使能，请参照安装 BSP 章节。

7 FAQ

7.1.1 默认用户名密码

用户名： pi

密码： raspberry

8 关于我们

8.1 关于 EDATEC

EDATEC 位于上海，是 Raspberry Pi 的全球设计合作伙伴之一。我们的愿景是提供基于 Raspberry Pi 技术平台的物联网、工业控制、自动化、绿色能源和人工智能的硬件解决方案。

我们提供标准的硬件解决方案，定制设计和制造服务，以加快电子产品的开发和上市时间。

8.2 联系方式

邮箱 - sales@edatec.cn / support@edatec.cn

手机 - +86-18621560183

网站 - <https://www.edatec.cn>

地址 - 上海市嘉定区嘉罗公路 1661 号 24 栋 301 室