

IBOX3588 边缘 AI 嵌入式工控机



用户手册

版本：1.5

© 成都万创科技股份有限公司 版权所有

版本记录：

编号	版本	说明	日期
1	V1.0	首次发布	2022 年 9 月 26 日
2	V1.1	增加 Ubuntu 系统手册，并对应调整排版	2022 年 11 月 28 日
3	V1.2	更新部分接口描述	2023 年 2 月 13 日
4	V1.3	根据设计更改，更新串口说明	2023 年 5 月 18 日
5	V1.4	将 Debian 系统串口调试命令更新为与其他系统一致	2023 年 12 月 8 日
6	V1.5	将 Ubuntu 和 Debian 系统的串口调试命令与 Android 系统的调试命令加以区分	2023 年 12 月 15 日

目录

前言	1
第 1 章 引言	5
1.1 产品概述	6
1.2 开箱	7
1.3 规格	8
1.4 产品布局	9
1.5 操作系统	10
1.6 机械尺寸	10
1.7 电源及功耗	10
1.8 环境参数	10
第 2 章 快速开始	11
2.1 硬件连接	12
2.2 使用设备	15
2.3 打开/关闭设备	16
第 3 章 硬件说明	17
3.1 电源输入	18
3.2 开/关按钮	18
3.3 网口	19
3.4 USB 2.0 + 网口	19
3.5 USB 3.0	19
3.6 USB Type-C	20
3.7 HDMI 接口	20
3.8 Micro SD & Micro SIM 卡槽	20
3.9 串口接头	20
3.10 CAN	21
3.11 GPIO	22
3.12 M.2 插槽	22
3.13 天线接头	22
第 4 章 首次使用调试	23
4.1 串口接头	24
4.1.1 Android 系统	25
4.1.2 Ubuntu 和 Debian 系统	27
4.2 CAN	29
4.3 GPIO	31
4.3.1 Android 系统	31
4.3.2 Ubuntu 和 Debian 系统	32
4.4 看门狗计时器	33
4.5 音频播放	34
第 5 章 Android 系统手册	37
5.1 启用开发者选项	38
5.2 在 Windows 主机上设置 ADB	39

5.3	通过 ADB 命令安装应用程序.....	41
5.4	Windows 环境下升级固件	42
第 6 章	Debian 系统手册	45
6.1	系统介绍.....	46
6.1.1	用户名与密码	46
6.1.2	系统信息	46
6.2	系统设置.....	47
6.2.1	语言设置	48
6.2.2	时间和日期设置	50
6.2.3	网络配置	51
6.3	Ubuntu 环境升级镜像	52
6.3.1	环境准备	52
6.3.2	烧写镜像	52
第 7 章	Ubuntu 系统手册	54
7.1	系统介绍.....	55
7.1.1	用户名与密码	55
7.1.2	系统信息	55
7.2	系统设置.....	56
7.2.1	语言	56
7.2.2	日期和时间	56
7.2.3	网络/Wi-Fi	56
7.2.4	用户	56
7.3	Ubuntu 环境升级镜像	57
7.3.1	环境准备	57
7.3.2	烧写镜像	57
7.4	ROS 教程.....	58
第 8 章	废弃处理与质保	59
8.1	废弃处理.....	60
8.2	质保	61
附录	合规声明	62

前言

感谢购买 IBOX3588 边缘 AI 嵌入式工控机（“设备”或“产品”）。本手册旨在就产品的设置、操作及维护提供必要的指导和帮助。请仔细阅读本手册，并确保您在使用产品前已理解产品的结构和功能。

目标用户

本手册旨在提供给：

- 设备使用人
- 技术支持工程师
- 其他用户

版权说明

成都万创科技股份有限公司（“万创”）保留本手册的所有权利，包括随时更改内容、形式、产品功能和规格的权利，恕不事先另行书面通知。您可访问 www.vantrontech.com.cn 获取本手册最新版本。

本手册中的商标和注册商标均为其各自所有者的财产。本手册的任何部分均不得复制、翻印、翻译或出售。未经万创事先书面同意，不得对本手册进行任何更改或将其用于其他用途。万创保留对本手册所有公开发布副本的权利。

免责声明

尽管已对本手册包含的所有信息进行了仔细检查，以确保其技术细节和印刷的准确性，但万创对本手册的任何错误或特性，或由于本手册或软件的不当使用造成的后果不承担任何责任。

产品额定功率或者特性发生变化时，或者发生重大结构变更时，我们会更换配件编号。产品规格如有变更，我们或不会另行通知。

技术支持与帮助

如您遇到本手册未曾提及的情况，请联系您的销售代表了解相关解决方案。请在来函中附上以下信息：

- 产品名称和订单编号；
- 关于相关问题的描述；
- 收到的报错信息，如有。

美国：Vantron Technology, Inc.

地址：48434 Milmont Drive, Fremont, CA 94538

电话：(650) 422-3128

邮箱：sales@vantrontech.com

中国：成都万创科技股份有限公司

地址：四川省成都市武侯区武科东三路9号1号楼6楼 610045

电话：86-28-8512-3930/3931, 86-28-8515-7572/6320

邮箱：sales@vantrontech.com.cn

法规信息

产品符合：

- FCC 认证
- 中国强制性认证（CCC）
- UL 认证

请查阅附录的合规声明。

符号约定

本手册使用以下符号，提醒用户注意相关信息。

	提醒可能会造成潜在的系统损坏或人员伤害。
	提示重要信息或法规。

一般安全说明

产品应当由合格熟练的技术人员按照当地及/或国际电气规范和法规进行安装。为保证人身安全并防止产品及其所连接设备发生损坏，请于产品安装和运行前，仔细阅读并遵守以下安全说明。请保留本手册，以供将来查阅。

- 请勿拆卸或以其他方式改装产品。此类行为可能造成发热、起火或人身伤害等其他损害，且导致产品保修失效。
- 保持产品远离加热器、散热器、发动机机壳等热源。
- 请勿将任何物品塞入产品，否则可能导致产品故障或烧坏。
- 为确保产品正常运行，防止产品过热，请勿阻挡产品通风口。
- 请使用提供或推荐的安装工具并遵守安装说明。
- 作业工具的使用或放置应当遵守此类工具的实施规程，避免产品短路。
- 检查产品前，请切断电源，避免出现人身伤害或产品损坏。

电缆和配件安全说明

- ⚠ 仅使用满足条件的电源。确保使用符合手册规定范围的供电电压。
- ⚠ 请确保合理放置电缆，避免受到挤压。
- ⚠ 产品包含纽扣电池，为实时时钟提供备用电源。因此，请在搬运或高温操作过程中避免电池短路。
- ⚠ 清洁说明：
 - 清洁前请关闭产品电源
 - 请勿使用喷雾清洁剂
 - 使用湿布进行清洁
 - 除非使用除尘器，否则请勿清洁裸露的电子组件
- ⚠ 出现以下故障时，请关闭电源并联系万创技术支持工程师：
 - 产品损坏
 - 温度过高
 - 根据手册检修后，故障仍然无法解决
- ⚠ 请勿在易燃易爆环境中使用：
 - 远离易燃易爆环境
 - 远离通电电路
 - 未经授权，不得拆开产品外壳
 - 拔掉电源之前，请勿更换零件
 - 某些情况下，拔掉电源后，产品仍有余电。因此，更换零件前，必须停止充电并等待产品完成放电。

第 1 章 引言

1.1 产品概述

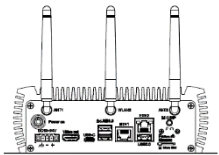
万创 IBOX3588 边缘 AI 嵌入式工控机由瑞芯微（Rockchip）RK3588 全新一代旗舰 AIoT 芯片驱动，搭载八核六十四位 CPU、ARM Mali-G610 MP4 四核 GPU，内置 AI 加速器 NPU，能够提供 6 TOPS 算力，支持主流深度学习框架。随着行业技术的发展，工业机器人、自动化控制、无人机等工控市场无疑将对具有 AI 算力的产品产生旺盛需求。

IBOX3588 提供两个千兆网口，支持 2.4GHz/5GHz Wi-Fi 6 和蓝牙 5.1，且支持 M.2 扩展 4G/5G 通信，保证通信不受阻碍。支持 8K 视频编码解码，增强显示性能。

此外，IBOX3588 提供丰富的外设接口，可接入大量外围设备，灵活用于 ARM PC、边缘计算、云服务器、智能 NVR 等领域。另外，产品提供不同的操作系统，可以保证用户系统环境的稳定性和安全性。

1.2 开箱

本产品包装细致，质量严格把关。但是，若您发现任何损坏或遗失，请立即联系您的销售代表。

标准配件		可选配件	
	1 x IBOX3588 工控机		1 x 12V DC Power adapter & power cord
	2 x Wi-Fi & 蓝牙 天线		1 x DC 转接线
\	\		2 x 4G LTE 天线 / 4 x 5G 天线

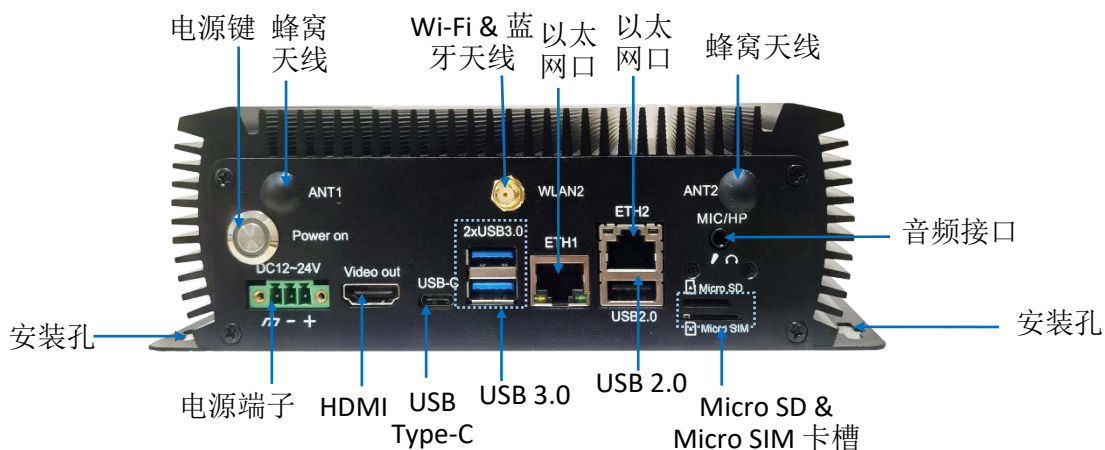
 以上配件取决于用户的选配规格，实际情况可能略有不同。

1.3 规格

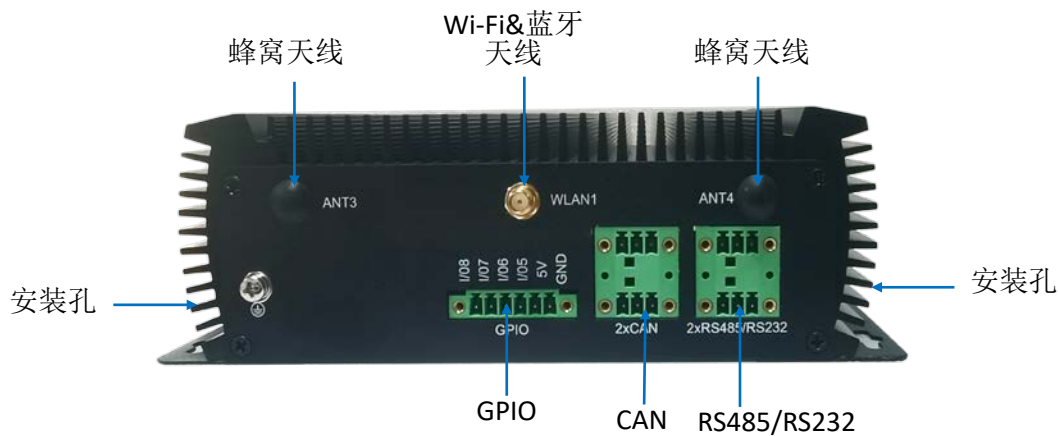
IBOX3588		
系统	CPU	Rockchip RK3588, 四核 Cortex-A76 + 四核 Cortex-A55, 最高频率 2.4GHz
	GPU	ARM Mali-G610 MC4, 最高频率 1GHz
	NPU	6 TOPS
	内存	8GB LPDDR4, 最大 32GB
	存储	32GB eMMC, 最大 128GB 支持 M.2 M-Key/M.2 B-Key 扩展 SSD (256GB ~ 1TB)
通信	以太网	2 x RJ45, 1000Mbps
	蜂窝网络	4G/5G (由 M.2 B-Key 扩展)
	Wi-Fi 及蓝牙	Wi-Fi 802.11 a/b/g/n/ac/ax + 蓝牙 5.1
多媒体	显示	1 x HDMI (4096 x 2160 @60Hz)
	视频	1 x 视频输出接口
	音频	1 x 二合一音频接口
输入/输出	USB	2 x USB 3.0 Host, Type-A 1 x USB 2.0 Host, Type-A 1 x USB 3.0 Type-C (支持 OTG)
	串口	2 x RS232/RS485 (2 x 3 x 3.81mm, 波特率: 115200)
	SIM 卡槽	1 x SIM 卡槽
	Micro SD 卡槽	1 x Micro SD 卡槽
扩展	RTC	支持
	看门狗	支持
	M.2 插槽	1 x M.2 M-Key 2260/2280, PCIe 3.0 x 4, 用于扩展 SSD 1 x M.2 B-Key 2242/3052, USB 3.0/SATA 3.0, 用于扩展 4G/5G/SSD
系统控制	GPIO	4 x GPIO (凤凰端子)
	CAN	2 x CAN (凤凰端子)
	按钮	1 x 开/关按钮 (长按复位)
软件	LED	1 x LED 指示灯 (电源和系统状态)
	操作系统	Android12, Debian, Ubuntu
	语言	英语 (默认), 中文
电源	设备管理平台	BlueSphere MDM (仅适用安卓版本)
	OTA 平台	BlueSphere OTA
	输入	1 x 电源端子 (12V/5A, 24V/3A)
机械	尺寸	200mm x 143mm x 63.2mm
	外壳	铝+钣金, 黑色
	重量	1200 克
	冲击测试	IEC 60068-2-27
环境	防水防尘	IP40
	温度	工作温度: -20°C ~ +70°C 存储温度: -40°C ~ +85°C
	湿度	相对湿度 0~95% (无凝露)
	认证	FCC, CCC, UL

1.4 产品布局

1.4.1 前视图



1.4.2 后视图

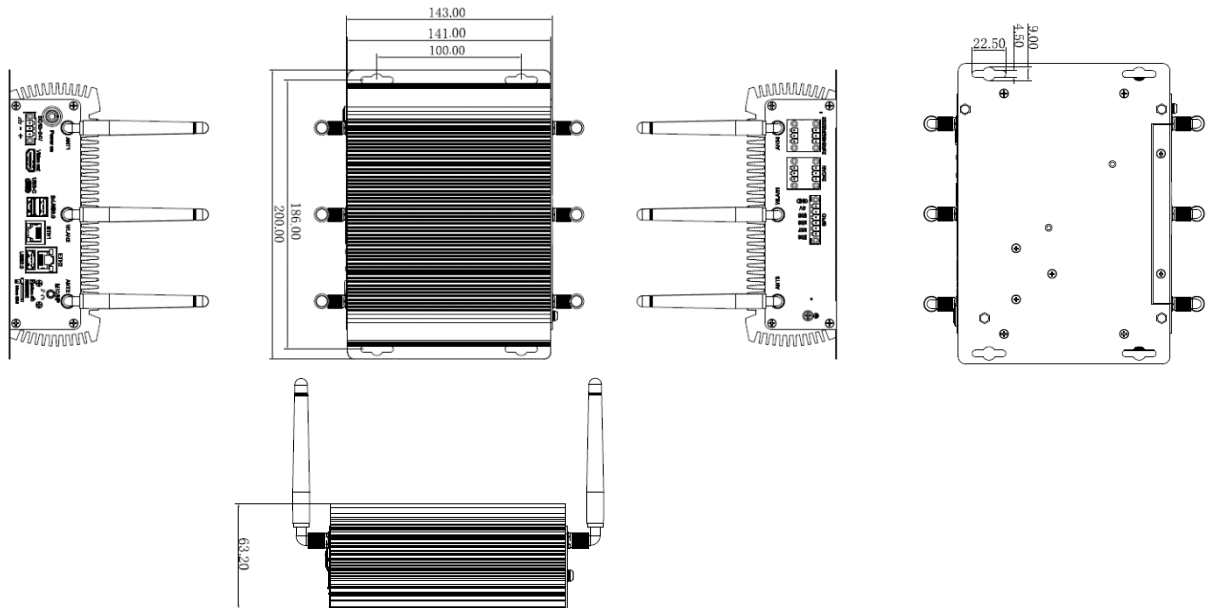


1.5 操作系统

IBOX3588 支持 Debian、Android 及 Ubuntu 操作系统。

1.6 机械尺寸

- 200mm x 143mm x 63.2mm



1.7 电源及功耗

IBOX3588 的工作电源为 12V/24V DC，通过设备上的电源连接头供电。

产品功耗在很大程度上是由产品的内存、存储容量、外接设备和其他配置决定的。

1.8 环境参数

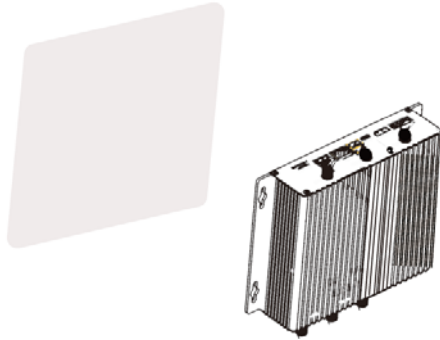
IBOX3588 的工作温度范围为 0°C至+60°C，其存储温度范围为-40°C至+85°C，相对湿度范围不超过 95%（无凝露）。

第 2 章 快速开始

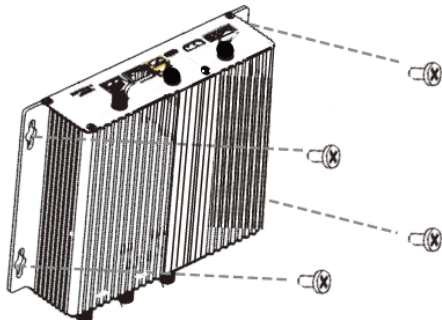
2.1 硬件连接

配置 IBOX3588 前，请根据以下步骤完成产品硬件连接。

1. 将设备置于目标位置，如墙壁或桌面上；



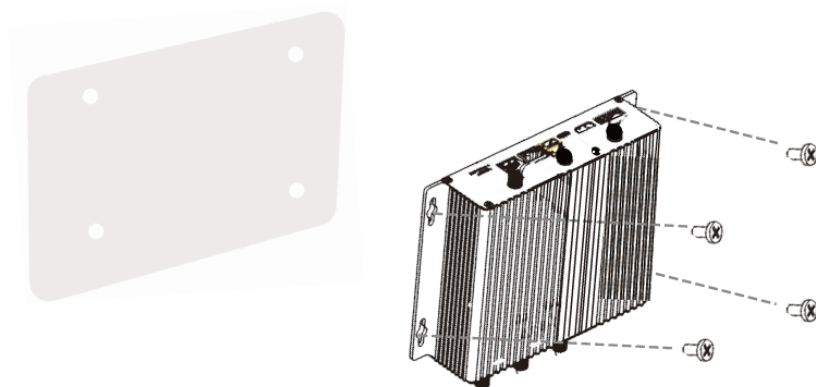
2. 将螺栓（推荐使用 M4 x 6 或 M4 x 8）与安装支架上的螺栓孔对齐，并在墙壁或桌面标记螺栓的位置；



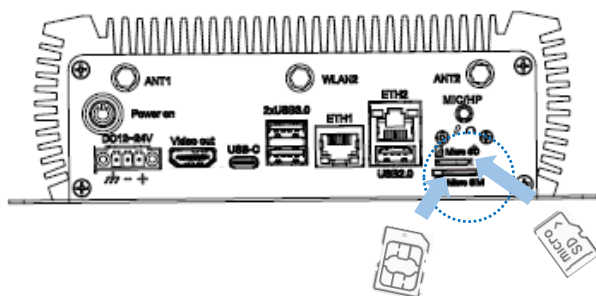
3. 在前述标记处钻孔；



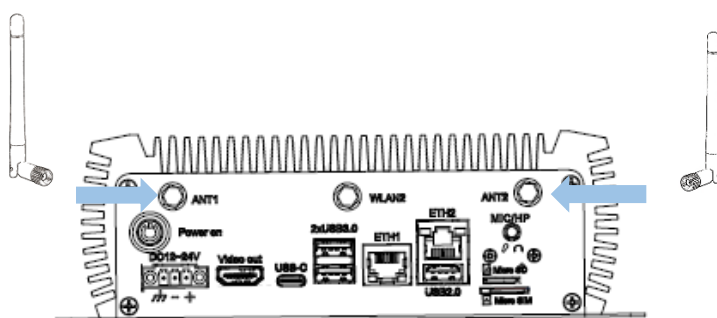
4. 将设备放回目标位置，如有需要，调节支架的位置。然后使用螺丝刀紧固螺栓，将设备固定；



5. 金属芯片朝上，将激活的 SIM 卡插入 Micro SIM 卡槽，同时将 SD 卡金属排针朝下插入 Micro SD 卡槽；

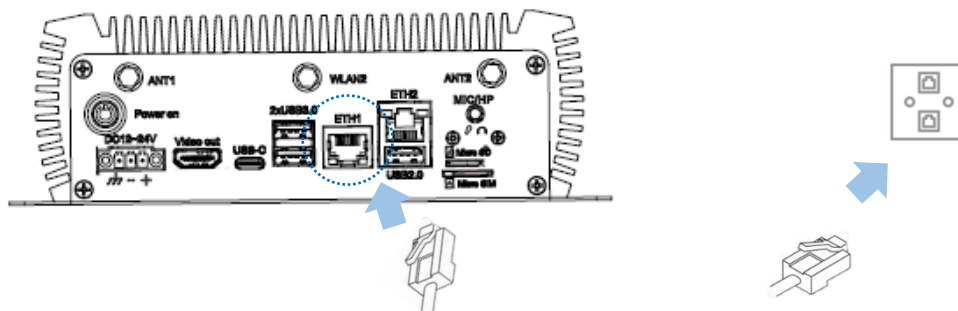


6. 将天线安装到天线接头上，并紧固接头；

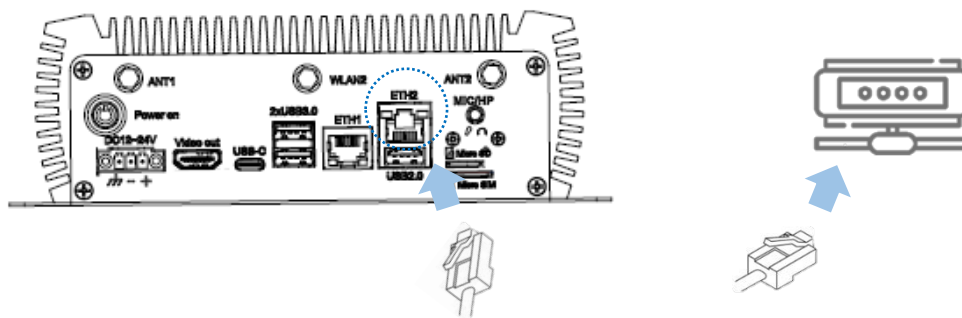


▶ 蜂窝天线接口未区分专门的 4G 或 5G 天线接口，4G 通信需安装 2 只天线，5G 通信需安装 4 只天线。

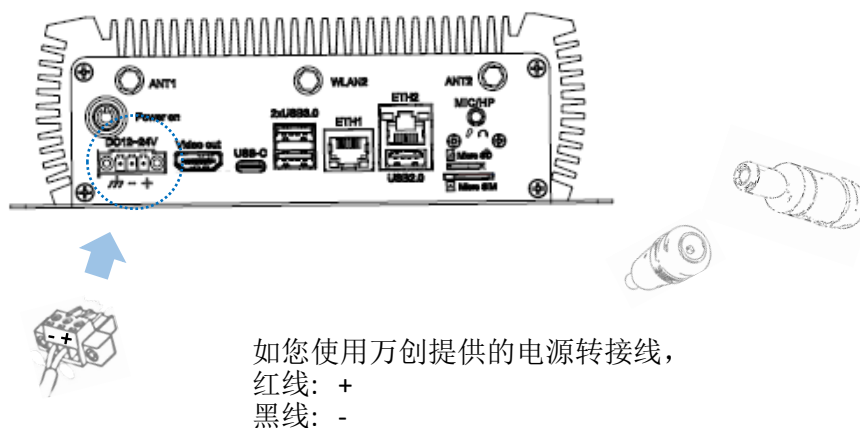
7. 如有需要，将网线的一端连接至 IBOX3588 的网口，另一端连接至外网；



8. 如有需要，使用另外一条网线将设备另一个网口和外围设备相连；



9. 将电源转接线的端子头插入 IBOX3588 的电源端子座，另一端与电源适配器连接；



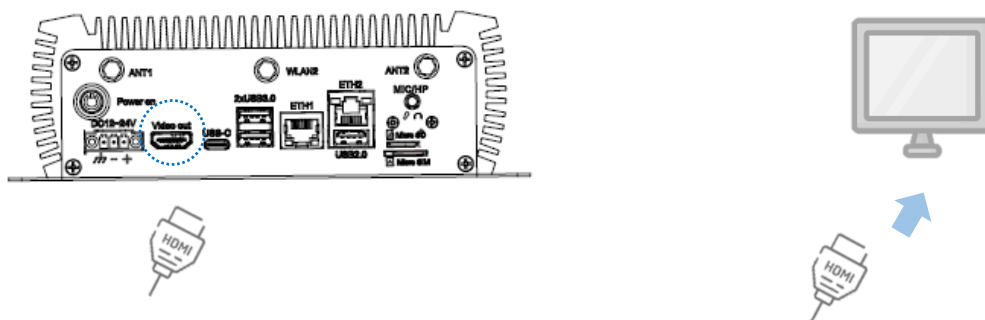
10. 使用 12V/24V DC 电源适配器使设备通电；

11. 通电后，电源按钮上的 LED 指示灯将变为绿色常亮。

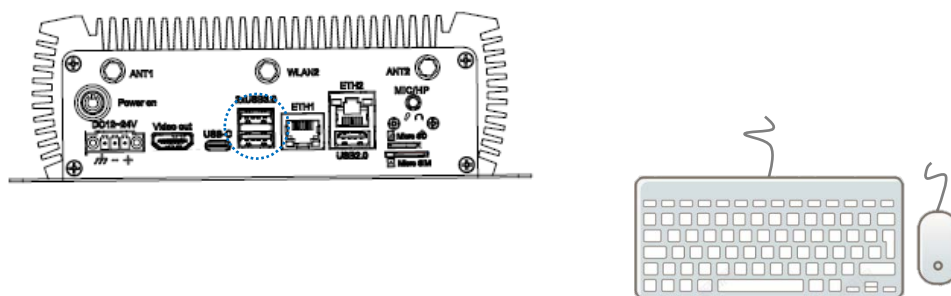
2.2 使用设备

请根据以下步骤，将鼠标、键盘、显示器与设备相连，方便进入并使用设备图形交互界面。

1. 将 HDMI 视频线的一端插入 IBOX3588 的 HDMI 接口，另一端与显示器相连；

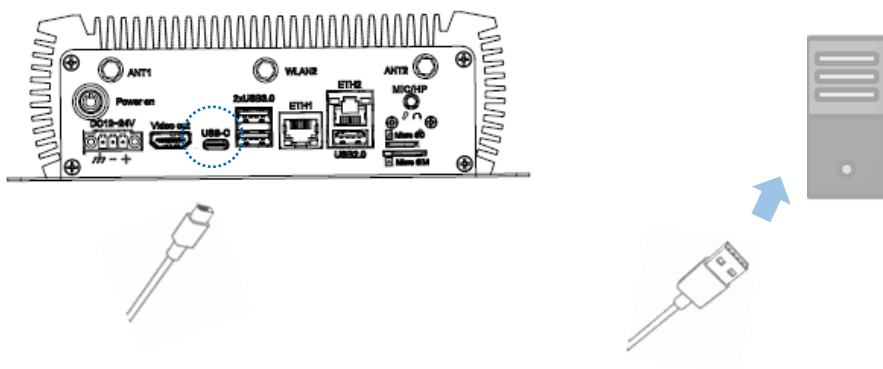


2. 通过设备的 USB 接口分别连接鼠标与键盘；



3. 使用 12V/24V DC 电源适配器使设备通电；

4. 如果需要远程调试设备，请使用 USB Type-A 转 Type-C 数据线连接主机和设备上的 USB Type-C 接口；



2.3 打开/关闭设备

根据前述步骤完成 IBOX3588 的设置后，用户可以通过以下步骤操作设备：

- 接通电源后，设备将自动启动；
- IBOX3588 启动后，用户可以：
 1. 短按开/关按钮一次，关闭显示器/从休眠模式唤醒设备；
 2. 按下开/关按钮 2 秒左右，调用电源菜单，然后点击**关机**，有序关闭设备；
 3. 按下开/关按钮并保持约 5 秒钟，直接关闭设备；
 4. 必要时，从 IBOX3588 的电源接口或插座上拔下电源适配器，可以强制关机。

 上述操作或因操作系统的不同而有所不同。

第 3 章 硬件说明

本节将简要介绍设备硬件定义以及连接头/跳线的引脚分布。

3.1 电源输入

根据设计，IBOX3588 支持 12V/24V 直流电源输入（推荐 12V 5A），设备带过压和过流保护。

电源连接头规格：1 x 3 针, 3.81mm



引脚 1

引脚说明：

引脚编号	信号	说明
1	GND	接地
2	-VDC	电源输入-
3	+VDC	电源输入+

3.2 开/关按钮



按钮定义（Android 系统）：

动作（通电状态）	说明
短按	唤醒设备/进入睡眠模式 打开设备（如果设备为关闭状态）
按下 1~2 秒	打开电源菜单，选择关机/重启设备/ 拨打紧急电话
按下 5 秒	关机

按钮定义（Debian/Ubuntu 系统）：

动作（通电状态）	说明
短按	唤醒设备/进入睡眠模式
	退出账号/ 重启/关闭/锁定设备
按下 1~2 秒	开启设备
	退出账号/ 重启/关闭/锁定设备
按下 5 秒	关机

开/关按钮上有一个 LED 指示灯。当设备接入带电电源时，指示灯将打开，并保持绿色常亮。

3.3 网口

IBOX3588 配置了两个以太网口。

规格：RJ45，支持 10M/100M/1000M Base-T4。每个网口上有两个 LED 指示灯，用于指示网络的连接/活动状态。

下图为网口 1。



3.4 USB 2.0 + 网口

第二个以太网口与 USB 2.0 Type-A 集成在一起，成堆叠状态。该 USB 接口最大输出为 5V/0.5A。



3.5 USB 3.0

IBOX3588 配置了两个 USB 3.0 Type-A 接口，可以连接外围设备。规格：最大输出 5V/1A。

3.6 USB Type-C

IBOX3588 配置了一个 USB Type-C 接口，该接口支持 USB 3.0 OTG 和 DP 1.4 视频显示输出。规格：最大输出 5V/1.5A。

3.7 HDMI 接口

IBOX3588 配置一个 HDMI 接口（Type-A），支持 HDMI 2.1 协议，分辨率可达 4096 x 2160 @60Hz。

3.8 Micro SD & Micro SIM 卡槽

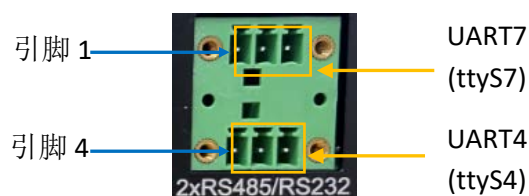
IBOX3588 提供一个 Micro SD 卡槽，最大支持 128GB 的存储设备。

SIM 卡槽支持 SIM 卡热插拔。

3.9 串口接头

IBOX3588 提供一个 6 针凤凰端子，配置为两个 RS232/RS485 复用串口接头。

凤凰端子上的两个复用接头为 UART4 和 UART7，分别对应/dev/ttyS7 和 ttyS4 节点。



RS232/RS485 引脚说明：

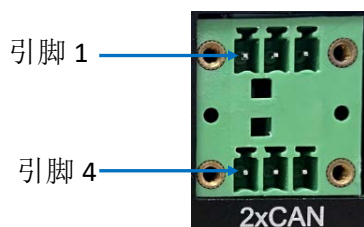
引脚编号	信号	说明
1	RS485_A_2/ RS232_TX2	RS485 I/O, A / RS232 数据输出
2	RS485_B_2/ RS232_RX2	RS485 I/O, B / RS232 数据输入
3	ISO GND	隔离接地
4	RS485_A_1/ RS232_TX1	RS485 I/O, A / RS232 数据输出
5	RS485_B_1/ RS232_RX1	RS485 I/O, B / RS232 数据输入
6	ISO GND	隔离接地

请参考 4.1，了解串口接头的功能测试。

3.10 CAN

IBOX3588 上还提供一个 6 针凤凰端子，配置为两个 CAN 连接头。

规格：2 x 3 x 3.81mm。

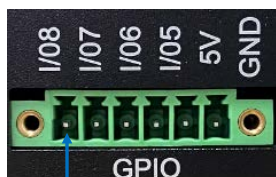


引脚说明：

引脚编号	信号	说明
1	CAN0_H	CAN0 高电平
2	CAN0_L	CAN0 低电平
3	GND	接地
4	CAN1_H	CAN1 高电平
5	CAN1_L	CAN1 低电平
6	GND	接地

3.11 GPIO

IBOX3588 另外提供一个 6 针凤凰端子，配置为 4 个 GPIO，便于用户进行设备开发。
规格：1 x 6 x 3.81mm。



引脚 1

引脚说明：

引脚编号	信号	说明
1	GPIO8_5V	GPIO 5V 电平
2	GPIO7_5V	GPIO 5V 电平
3	GPIO6_5V	GPIO 5V 电平
4	GPIO5_5V	GPIO 5V 电平
5	VCC_5V0	5V 电源输出
6	GND	接地

3.12 M.2 插槽

IBOX3588 上配置两个 M.2 插槽，包括一个 M.2 M-Key 插槽（2260/2280），支持 PCIe 3.0 x 4，用于 SSD 扩展，以及一个 M.2 B-Key 插槽（2242/3052），支持 SATA 3.0，用于 SSD 扩展，或者 USB 3.0，用于连接 4G/5G 模块。

3.13 天线接头

机壳上标注的 ANT1、ANT2、ANT3、ANT4 天线接头用于连接 4G（任意两个接头）/5G（四个接头）天线。WLAN1 和 WLAN2 天线接头用于连接 Wi-Fi 和蓝牙天线。上述天线无主从之分。

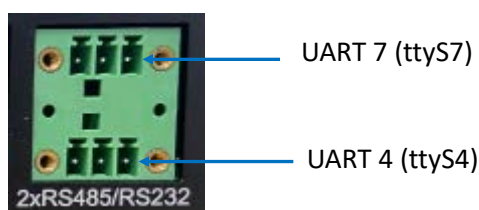
第 4 章 首次使用调试

本章主要针对接口或软件应用的首次使用调试进行说明。您可以使用鼠标、键盘和显示器连接 IBOX3588。如果设备运行 Linux 系统，您可以直接在设备控制台中进行调试，或者，也可以使用 USB Type-A 转 Type-C 数据线将 IBOX3588 和主控电脑连接，实现远程调试。

4.1 串口接头

IBOX3588 提供一个凤凰端子，配置为两个 RS232/RS485 复用串口接头，名为 UART 7 和 UART 4，在软件系统中的设备名称分别对应/dev/ttyS7 和 ttyS4。

请参考 3.9，了解接头引脚定义。



用户可以使用适当的 USB 转串口适配器连接设备串口和主控电脑（RS232 接线：TX-RX, RX-TX, GND-GND; RS485 接线：A-A, B-B, GND-GND），然后在主控电脑上使用串口工具进行串口通信，测试串口接头功能是否正常。

或者，也可以交叉连接两个串口，如果设备运行 Debian 或 Ubuntu 系统，则直接在设备控制台执行命令，并查看命令运行结果；如果设备运行 Android 系统，则请连接设备和主控电脑并在主控电脑上执行命令。

4.1.1 Android 系统

设备运行 Android 系统时，两个串口在不同模式下(RS232、RS485)的收发命令汇总如下：

RS232 模式		
UART4	发送数据	<pre># echo 0 > /sys/class/gpio/gpio497/value # echo 1 > /sys/class/gpio/gpio498/value # echo "Test string" > /dev/ttyS4</pre>
	接收数据	<pre># echo 0 > /sys/class/gpio/gpio497/value # echo 1 > /sys/class/gpio/gpio498/value # cat /dev/ttyS4</pre>
UART7	发送数据	<pre># echo 0 > /sys/class/gpio/gpio499/value # echo 1 > /sys/class/gpio/gpio500/value # echo "Test string" > /dev/ttyS7</pre>
	接收数据	<pre># echo 0 > /sys/class/gpio/gpio499/value # echo 1 > /sys/class/gpio/gpio500/value # cat /dev/ttyS7</pre>

RS485 模式		
UART4	发送数据	<pre># echo 0 > /sys/class/gpio/gpio497/value # echo 0 > /sys/class/gpio/gpio498/value # echo "Test string" > /dev/ttyS4</pre>
	接收数据	<pre># echo 0 > /sys/class/gpio/gpio497/value # echo 0 > /sys/class/gpio/gpio498/value # cat /dev/ttyS4</pre>
UART7	发送数据	<pre># echo 0 > /sys/class/gpio/gpio499/value # echo 0 > /sys/class/gpio/gpio500/value # echo "Test string" > /dev/ttyS7</pre>
	接收数据	<pre># echo 0 > /sys/class/gpio/gpio499/value # echo 0 > /sys/class/gpio/gpio500/value # cat /dev/ttyS7</pre>

下文的示例说明了如何在 RS232 模式下测试串口的功能是否正常：

1. 按照下图所示，交叉连接两个串口（TX-RX, RX-TX, GND-GND）；



2. 打开一个终端，使用其中一个串口（如 UART 7）接收数据；

```
# echo 0 > /sys/class/gpio/gpio499/value  
# echo 1 > /sys/class/gpio/gpio500/value  
# cat /dev/ttyS7
```

3. 打开另一个终端，并使用另一个串口（UART 4）发送数据；

```
# echo 0 > /sys/class/gpio/gpio497/value  
# echo 1 > /sys/class/gpio/gpio498/value  
# echo "Test string" > /dev/ttyS4 •
```

4. 第一个终端将打印接收到的数据；
5. 参考上页的表格，了解不同模式下，不同数据传输动作所使用的命令。

本例中，用户也可以使用单个串口进行数据收发。

4.1.2 Ubuntu 和 Debian 系统

设备运行 Ubuntu 或 Debian 系统时，两个串口在不同模式下(RS232、RS485)的收发命令汇总如下：

RS232 模式		
UART4	发送数据	<pre># gpioset gpiochip7 4=0 # gpioset gpiochip7 5=1 # echo "Test string" > /dev/ttyS4</pre>
	接收数据	<pre># gpioset gpiochip7 4=0 # gpioset gpiochip7 5=1 # cat /dev/ttyS4</pre>
UART7	发送数据	<pre># gpioset gpiochip7 6=0 # gpioset gpiochip7 7=1 # echo "Test string" > /dev/ttyS7</pre>
	接收数据	<pre># gpioset gpiochip7 6=0 # gpioset gpiochip7 7=1 # cat /dev/ttyS7</pre>

RS485 模式		
UART4	发送数据	<pre># gpioset gpiochip7 4=1 # gpioset gpiochip7 5=0 # echo "Test string" > /dev/ttyS4</pre>
	接收数据	<pre># gpioset gpiochip7 4=0 # gpioset gpiochip7 5=0 # cat /dev/ttyS4</pre>
UART7	发送数据	<pre># gpioset gpiochip7 6=1 # gpioset gpiochip7 7=0 # echo "Test string" > /dev/ttyS7</pre>
	接收数据	<pre># gpioset gpiochip7 6=0 # gpioset gpiochip7 7=0 # cat /dev/ttyS7</pre>

下文的示例说明了如何在 RS232 模式下测试串口的功能是否正常：

1. 按照下图所示，交叉连接两个串口（TX-RX, RX-TX, GND-GND）；



2. 打开一个终端，使用其中一个串口（如 UART 7）接收数据；

```
# gpioset gpiochip7 6=0
# gpioset gpiochip7 7=1
# cat /dev/ttyS7
```

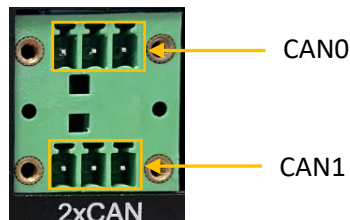
3. 打开另一个终端，并使用另一个串口（UART 4）发送数据；

```
# gpioset gpiochip7 4=0
# gpioset gpiochip7 5=1
# echo "Test string" > /dev/ttyS4
```

4. 第一个终端将打印接收到的数据；
5. 参考上页的表格，了解不同模式下，不同数据传输动作所使用的命令。

4.2 CAN

设备配置了两个 CAN 总线接头（CAN0 和 CAN1）。用户可以交叉连接两个接头（H-H, L-L, GND-GND），然后使用其中一个接头发送数据，另外的接头接收数据，测试接头功能是正常。请参考 3.10，了解接头引脚定义。



- 使用 CAN0 接收数据，CAN1 发送数据：

1. 打开终端设置 CAN0 的波特率，并输入命令接收数据；

```
# ip link set can0 up type can bitrate 125000  
# candump can0
```

2. 打开另外一个终端，设置 CAN1 的波特率，并输入命令发送数据。

```
# ip link set can1 up type can bitrate 125000  
# cansend can1 500#11.22.33.44
```

- 使用 CAN1 接收数据，CAN0 发送数据：

1. 打开终端设置 CAN1 的波特率，并输入命令接收数据；

```
# ip link set can1 up type can bitrate 125000 （若之前已设置，则略过此步）  
# candump can1
```

2. 打开另外一个终端，设置 CAN0 的波特率，并输入命令发送数据。

```
# ip link set can0 up type can bitrate 125000 （若之前已设置，则略过此步）  
# cansend can0 500#11.22.33.44
```

- 查看 CAN0 的状态:

```
# ip -details link show can0
```

```
root@linaro-alip:/# ip -details link show can0
2: can0: <NOARP,UP,LOWER_UP,ECHO> mtu 16 qdisc pfifo_fast state UP mode DEFAULT
group default qlen 10
    link/can  promiscuity 0 minmtu 0 maxmtu 0
    can state ERROR-PASSIVE (berr-counter tx 0 rx 0) restart-ms 1
    bitrate 125000 sample-point 0.871
    tq 60 prop-seg 57 phase-seg1 57 phase-seg2 17 sjw 1
    rockchip_canfd: tseg1 1..128 tseg2 1..128 sjw 1..128 brp 1..256 brp-i
nc 2
    clock 99000000 numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max
_segs 65535
root@linaro-alip:/#
```

- 查看 CAN1 的状态:

```
# ip -details link show can1
```

4.3 GPIO

设备配置了四个 GPIO 接口 (GPIO5 ~ GPIO8)，在软件系统中的编号分别为 gpio489、gpio490、gpio491、gpio492。用户可以参考 3.11 的描述，了解每个接头的引脚定义。

输入以下命令，查看 GPIO 状态：

```
# cat /sys/kernel/debug/gpio
```

Android 系统下设置 GPIO 管脚高/低电平的命令与 Ubuntu 和 Debian 系统中使用的命令不同。用户可以参考后文总结的命令，设置管脚的高/低电平，然后使用电压表测试设置是否有效。

4.3.1 Android 系统

下表列出了 Android 系统下各 GPIO 接口管脚的高/低电平设置命令。

GPIOs		
GPIO5	高电平	# echo 1 > sys/class/gpio/gpio489/value
	低电平	# echo 0 > sys/class/gpio/gpio489/value
GPIO6	高电平	# echo 1 > sys/class/gpio/gpio490/value
	低电平	# echo 0 > sys/class/gpio/gpio490/value
GPIO7	高电平	# echo 1 > sys/class/gpio/gpio491/value
	低电平	# echo 0 > sys/class/gpio/gpio491/value
GPIO8	高电平	# echo 1 > sys/class/gpio/gpio492/value
	低电平	# echo 0 > sys/class/gpio/gpio492/value

切换 GPIO 接口的方向：

```
# echo in > sys/class/gpio/gpioxxx/direction  
# echo out > sys/class/gpio/gpioxxx/direction
```

4.3.2 Ubuntu 和 Debian 系统

下表列出了 Ubuntu 或 Debian 系统下各 GPIO 接口管脚的高/低电平设置命令。

GPIOs		
GPIO5	高电平	# gpioset gpiochip8 4=1
	低电平	# gpioset gpiochip8 4=0
GPIO6	高电平	# gpioset gpiochip8 5=1
	低电平	# gpioset gpiochip8 5=0
GPIO7	高电平	# gpioset gpiochip8 6=1
	低电平	# gpioset gpiochip8 6=0
GPIO8	高电平	# gpioset gpiochip8 7=1
	低电平	# gpioset gpiochip8 7=0

4.4 看门狗计时器

看门狗计时器默认被禁用。开启后，看门狗计时器会在每 44 秒内与系统通信一次（“喂狗”）。如果通信中断，系统将在看门狗计时结束时重启，这也意味着系统或者应用出现故障。

1. 写入除大写字母 ‘V’ 以外的任意字符，开启看门狗;

```
# echo A > /dev/watchdog
```

2. 设置喂狗频率（如，22 秒）以及设备异常时自动重启的时间间隔（如，2 分钟）;

```
[Manager]
LogLevel=warning
LogTarget=journal
#LogColor=yes
#LogLocation=no
#DumpCore=yes
#ShowStatus=yes
#CrashChangeVT=no
#CrashShell=no
#CrashReboot=no
#CtrlAltDelBurstAction=reboot-force
#CPUAffinity=1 2
#NUMAPolicy=default
#NUMAMask=
RuntimeWatchdogSec=22s
RebootWatchdogSec=2min
#ShutdownWatchdogSec=10min
#KExecWatchdogSec=0
#WatchdogDevice=
```

3. 禁用看门狗;

```
# echo V > /dev/watchdog
```

4. 用户也可以通过“systemd”命令开启硬件看门狗，并按照上述说明进行配置。

```
# /etc/systemd/system.conf
```

4.5 音频播放

设备上的 HDMI 接口和 3.5mm 音频插孔都支持音频输出，而且音频插孔还支持音频录制。对于 Android 用户，您只需插入一条 3.5 毫米音频线即可使用这些功能。对于 Debian 和 Ubuntu 用户，您需要手动打开耳机功能，才能播放已录制的音频文件。

请使用下文所述命令，在 Debian 和 Ubuntu 系统上启用 3.5mm 音频插孔录制和播放音频的功能。

1. 查看设备上的声卡；

```
$ aplay -l

linaro@linaro-alip:~$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchip-hdmi1 [rockchip-hdmi1], device 0: rockchip-hdmi1 i2s-hifi-0 [rockchip-hdmi1 i2s-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: rockchip-es8388 [rockchip-es8388], device 0: dailink-multicodecs ES8323.7-0011-0 [dailink-multicodecs ES8323.7-0011-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
linaro@linaro-alip:~$
```

HDMI interface

Audio jack

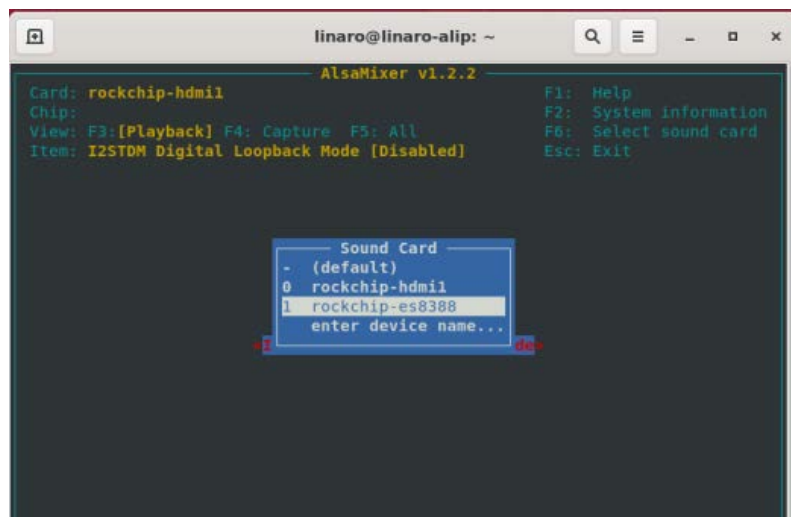
2. 将同时支持录音和播放功能的 3.5mm 音频线插入音频插孔；
3. 录制音频文件并保存，可以将该文件命名为“test.wav”；

```
$ arecord -D plughw:1,0 -f S16_LE -c 2 -r 44100 test.wav
```

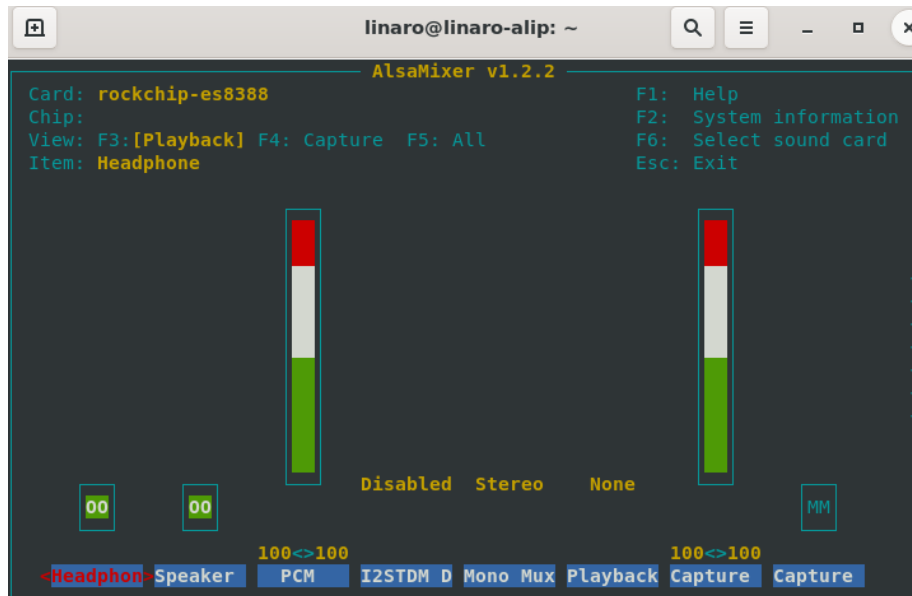
4. 打开 alsamixer 程序并按 F6 修改音频设置；

```
$ alsamixer
```

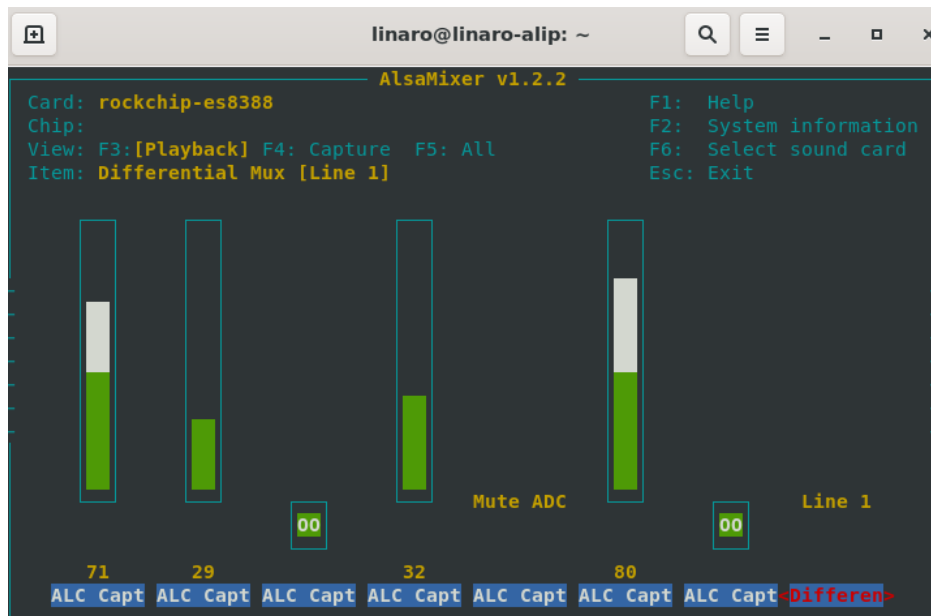
5. 使用键盘上的上/下箭头选择“1 rockchip-es8388”并按回车键；



6. 使用键盘上的左/右箭头导航到<Headphone>选项，并按“m”键启用耳机功能；



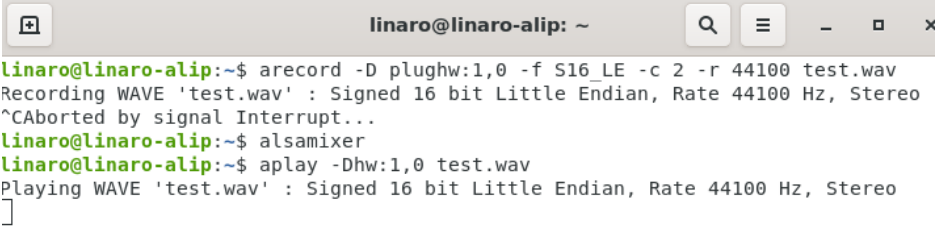
7. 使用键盘上的左/右箭头导航到<differential>选项，并使用上/下箭头选择线路 1；



8. 按 ESC 键退出设置；

9. 输入以下命令播放录制的音频。

```
Aplay -Dhw:1,0 test.wav
```



A terminal window titled 'linaro@linaro-alip: ~' showing a sequence of commands and their outputs. The commands are: 'arecord -D plughw:1,0 -f S16_LE -c 2 -r 44100 test.wav', 'alsamixer', and 'aplay -Dhw:1,0 test.wav'. The outputs show the recording process being interrupted and then the audio file being played back through the hardware interface.

```
linaro@linaro-alip:~$ arecord -D plughw:1,0 -f S16_LE -c 2 -r 44100 test.wav
Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
^CAborted by signal Interrupt...
linaro@linaro-alip:~$ alsamixer
linaro@linaro-alip:~$ aplay -Dhw:1,0 test.wav
Playing WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
]
```

要切换到 HDMI 并从该接口播放音频文件，请使用以下命令：

```
Aplay -Dhw:0,0
```


第 5 章 Android 系统手册

前提条件：

- IBOX3588
- Windows 主机
- IBOX3588 软件发布包
- USB 鼠标、键盘和显示器，用于连接 IBOX3588，实现便捷操作
- 12V/24V 适配器，用于启动设备
- USB Type-A 转 Type-C 数据线，用于连接设备和 Windows 主机

5.1 启用开发者选项

请根据以下步骤启用 IBOX3588 的开发者选项：

1. 将鼠标、键盘、显示器与 IBOX3588 相连，便于后续操作；
2. 系统启动后，上划进入应用抽屉；
3. 依次点击**设置 > 关于平板 (About tablet)**；
4. 向下滚动鼠标至**版本号**，连续点击版本号七次，打开**开发者选项**；
5. 回到**设置 > 系统 > 高级 > 开发者选项**，打开设备的 **USB 调试**，之后即可自定义 IBOX3588 的设置。

 根据 Android 版本的不同，入口或许略有差异。

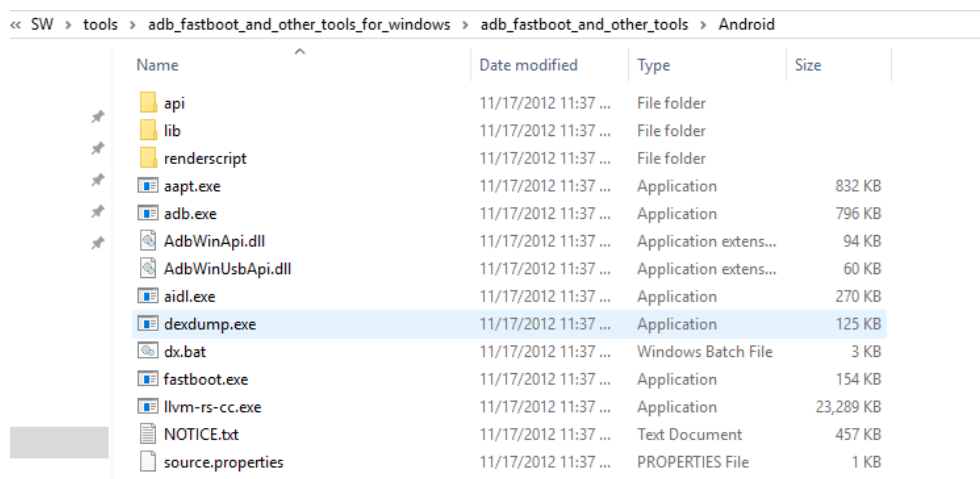
5.2 在 Windows 主机上设置 ADB

安卓调试桥（ADB）是一种用于直接连接用户开发工作站与其安卓设备的工具，用户可以通过该工具完成调试、设备升级、应用安装等。

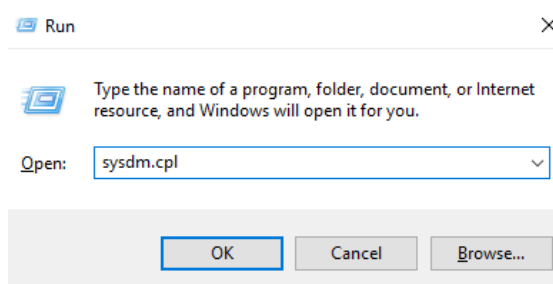
将 ADB 可执行文件添加至系统的环境变量后，无论用户当前工作路径如何，均可以直接在任意位置使用 ADB 工具。

请根据以下步骤在 Windows 主机上完成 ADB 设置。

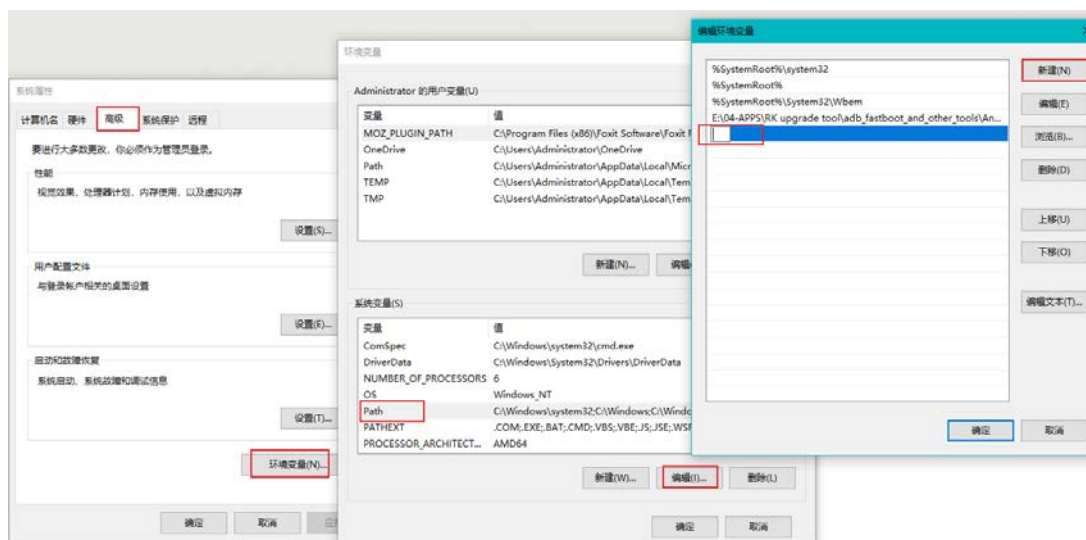
1. 将软件发布包解压缩，然后进入以下目录：\SW\tools；
2. 解压 adb_fastboot_and_other_tools_for_windows 文件夹；
3. Android 文件夹包含了 ADB 工具套件，导航至该文件夹并复制其路径；



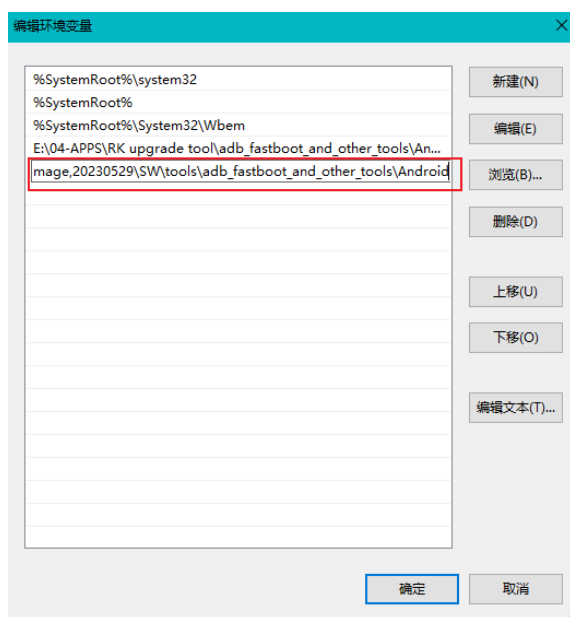
4. 按下“Win + R”键，并在对话框中输入 sysdm.cpl，打开设置界面；



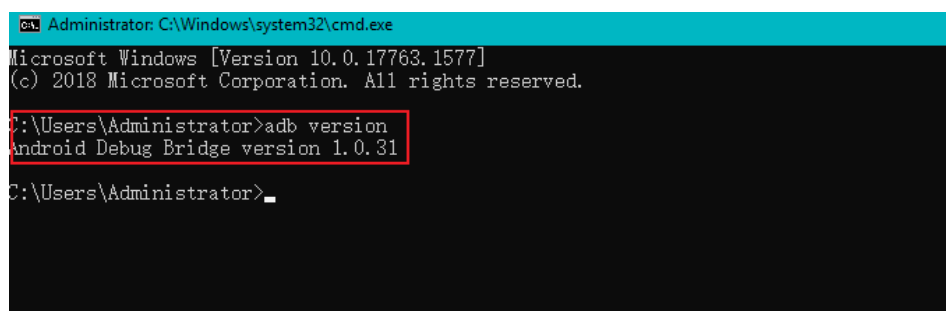
5. 依次点击**高级 > 环境变量 > 系统变量 > Path > 编辑**，然后在弹窗内选择**新建**；



6. 将复制的 Android 文件夹路径粘贴至环境变量对话框中，然后逐一点击**确定**关闭对话框；



7. 按下“Win + R”键，并在对话框中输入 **cmd**；
8. 在命令提示框内输入 **adb version**，查看 ADB 工具是否已安装。



5.3 通过 ADB 命令安装应用程序

针对运行 Android 操作系统的设备，除了标准预装的 Android 应用程序外，用户还可以在设备上安装自己的应用。只要 Windows 主机上安装了 ADB 工具套件，并且用户具有访问权限，即可以在 Windows 环境下安装上述应用程序。

1. 请根据 5.2 中的步骤完成 Windows 主机的 ADB 设置；
2. 使用 USB Type-A 转 Type-C 数据线连接设备和控制电脑；
3. 按下“Win + R”键，并在对话框中输入 `cmd`；
4. 在命令提示框内输入 `adb devices -l`，查看设备是否已连接控制电脑；

```
C:\Users\Administrator>adb devices -l
List of devices attached
674cc0aaede7d049    device product:occam model:Nexus_4 device:mako transport_id:1
```

5. 如果命令下方显示设备信息，则说明设备已经通过 ADB 与控制电脑连接，用户可以复制设备序列号（上图标识的区域），便于下一步使用；
6. 在命令提示框内输入以下命令安装 apk 应用程序：

```
adb -s <序列号> install <APK 文件路径>
```

7. 输入命令行后即执行安装，安装的结果将显示于命令下方；

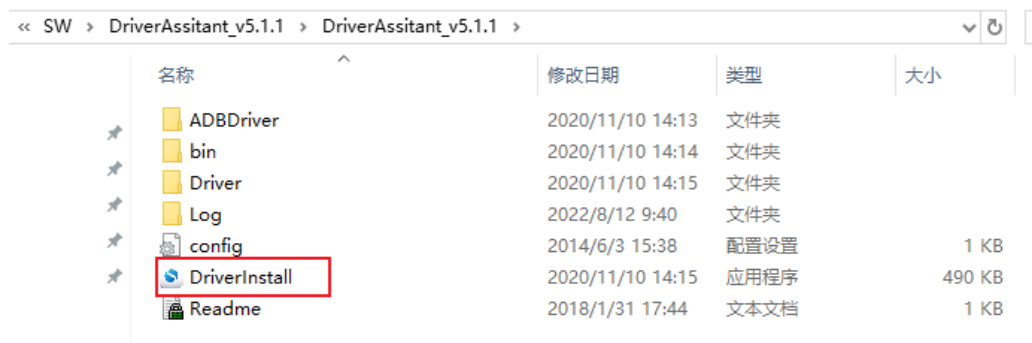
```
C:\Users\Administrator>adb -s 674cc0aaede7d049 install "C:\Users\Administrator\Desktop\Libraries for developers_v3.83_apkpure.com.apk"
Performing Streamed Install
Success
```

8. 新安装的应用程序将按照字母顺序在应用抽屉上显示。

- ▶ 以上截图仅作参考，并不代表您当前所持设备的编号及其他设备信息。
- ▶ 上述第 5 步中，用户也可以直接将本地的 apk 文件拖拽至命令行，替换手动输入的应用程序路径。
- ▶ 请确保 apk 文件路径中没有特殊字符。如果安装失败，请尝试添加双引号将 apk 文件的路径转换为绝对路径。

5.4 Windows 环境下升级固件

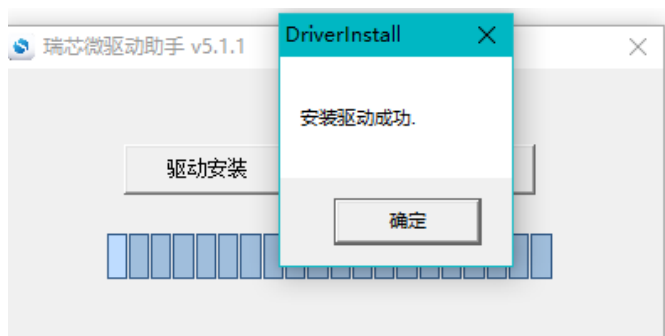
1. 请根据 5.2 中的步骤完成 Windows 主机的 ADB 设置；
2. 将软件发布包解压缩，打开升级驱动目录（\SW\DriverAssitant_vxxx），找到 **DriverInstall.exe**；



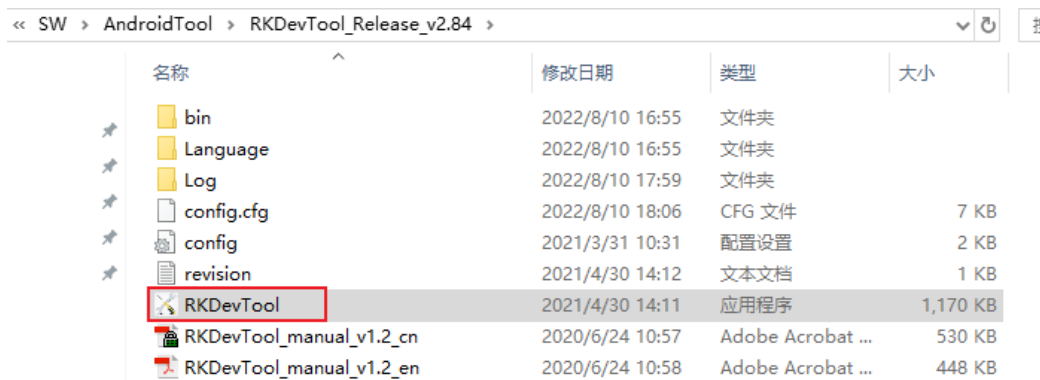
3. 点击鼠标右键，以管理员的身份运行驱动程序；
4. 首先点击**卸载驱动**，卸载之前安装的驱动版本（如有），然后点击**安装驱动**并等待；



5. 数秒后，将出现安装成功的弹窗；



6. 回到 SW 文件夹，打开升级工具目录（\SW\AndroidTool\RKDevTool_Release_vxxx\RKDevTool）；



7. 双击升级驱动程序 RKDevTool.exe, 打开升级窗口；



8. 使用 USB Type-A 转 Type-C 数据线连接 IBOX3588 和 Windows 主机；
9. 同时按下 “Windows + R” 键并在对话框中输入 `cmd`，打开命令提示框；
10. 在命令提示框中输入 `adb devices` 查看设备是否与 Windows 主机相连；
11. Windows 主机识别到设备后，在命令提示框内输入 `adb reboot loader` 命令，重启设备并进入 bootloader 模式；

12. 之后，升级窗口将提示发现一个 LOADER 设备，表示升级准备就绪。



13. 在 RK 开发工具窗口，依次点击**升级固件 > 固件**；

14. 打开升级文件 **update.img** 的路径（\SW\Image），固件的详细信息将自动填充到固件信息框内；

15. 点击**升级**按钮，系统将自动开始下载镜像并升级；



16. 升级完成后，系统将自动重启。

第 6 章 Debian 系统手册

6.1 系统介绍

Debian 是 Linux 系统的一个发行版本。IBOX3588 基于 Debian GNU/Linux 11 操作系统创建。

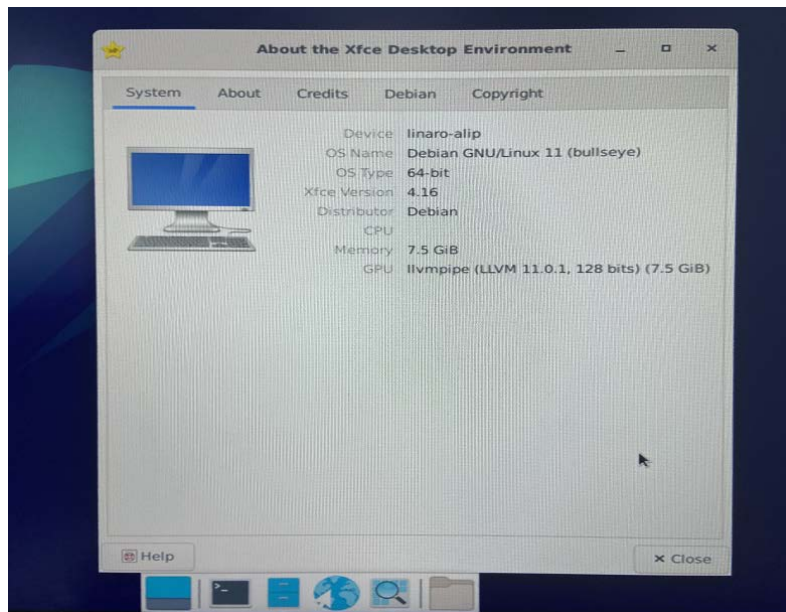
6.1.1 用户名与密码

系统开机后自动默认登陆 **linaro** 账号。

- 用户：linaro
- 密码：linaro

6.1.2 系统信息

1. 为方便操作，请使用键盘、鼠标和显示器连接设备；
2. 设备通电，系统将默认登录 linaro 用户；
3. 鼠标右键点击屏幕空白处，打开设备属性对话框；
4. 依次点击**应用程序 > 关于 Xfce**，进入桌面环境，然后可以查看系统信息，包括设备名称、操作系统、版权声明等。



6.2 系统设置

点击屏幕右上角的 **linaro**，用户可以选择：

- 锁屏；
- 切换用户；
- 关机/挂起设备；
- （点击“注销”后）注销/重启/关机/挂起设备或者切换用户，同时选择保存当前会话，用于将来登录。

用户可以通过屏幕上方的菜单栏：

- 修改设备亮度和电源设置；
- 调节系统音量和声音设置；
- 将设备添加至 **Wi-Fi** 网络并与蓝牙设备配对；
- 切换不同的工作区域。

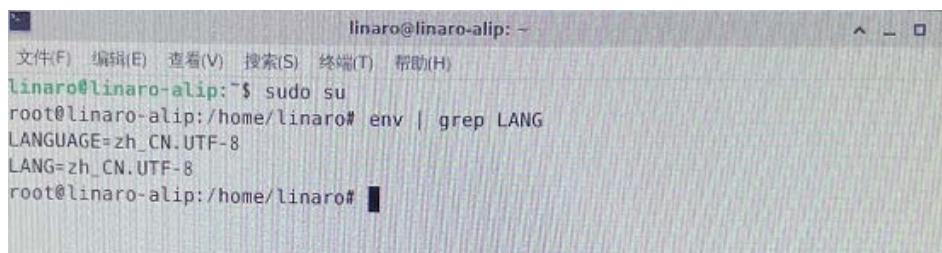
用户可以点击屏幕左上角的**所有应用程序**或者右键点击屏幕空白处，在弹出的设备属性对话框中点击**应用程序**，然后进行系统其他设置。

6.2.1 语言设置

假设当前系统语言为中文，下文的例子将说明如何将系统语言更改为英文（English US）。

1. 从桌面底部打开终端并输入 `sudo su`，切换至 `root` 权限；
2. 运行下列命令查看系统当前使用的语言（系统默认使用 `zh_CN.UTF-8` 区域设置）；

```
# env | grep LANG
```



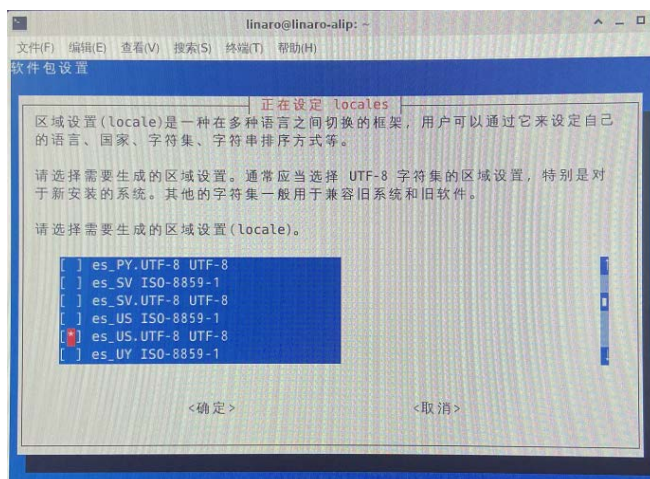
3. 输入下列命令调用区域设置的配置页面（`en_US.UTF-8` 为用户希望切换的目标语言所在区域）；

```
# export LANG=en_US.UTF-8
```

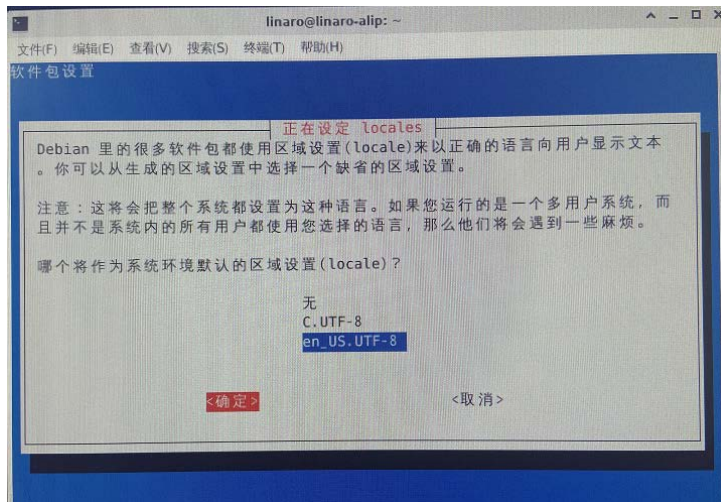
4. 输入下列命令重新设定区域设置；

```
# dpkg-reconfigure locales
```

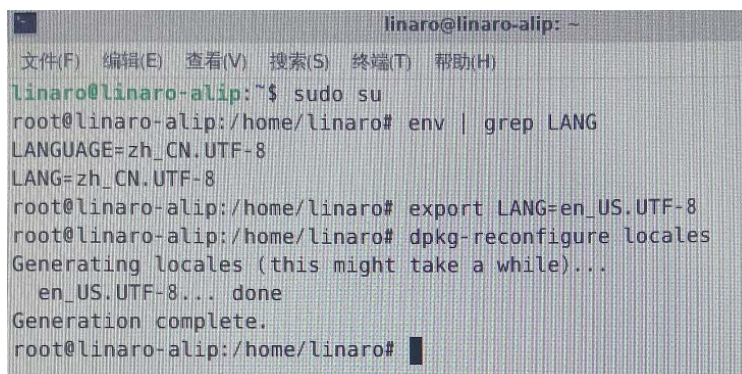
5. 使用键盘的上、下箭头移动至目标区域设置（`en_US.UTF-8`）；
6. 按下空格键，选择目标区域设置；



7. 移动至当前区域设置(zh_CN.UTF-8)，按空格键取消选择；
8. 然后按 Tab 键将鼠标移动至“确定”选项，再按回车键确认选择；
9. 使用上、下箭头在列表中选择需要设定为默认值的区域设置(en_US.UTF-8)，然后按 Tab 键将鼠标移动至“确定”选项，再按回车键确认选择；



10. 等待几秒钟，待所选的区域设置生效；



11. 输入 **reboot** 重启设备；
12. 设备重启后选择“更新名称”，将文件更新至当前语言；
13. 输入 **locale-a** 查看所选区域设置是否为系统默认设定。

6.2.2 时间和日期设置

用户可以点击屏幕左上角的**所有应用程序**或者右键点击屏幕空白处，在弹出的设备属性对话框中点击**应用程序**，调出系统菜单：

1. 点击**系统 > 时间和日期**，打开时间和日期设置；
2. 点击**解锁**并输入密码（linaro）进行设置；
3. 点击时区，并从列表中选择用户所在地时区；
4. 选择与互联网服务器保持同步或者手动输入时间；



5. 点击**锁定**，等待设置生效。

6.2.3 网络配置

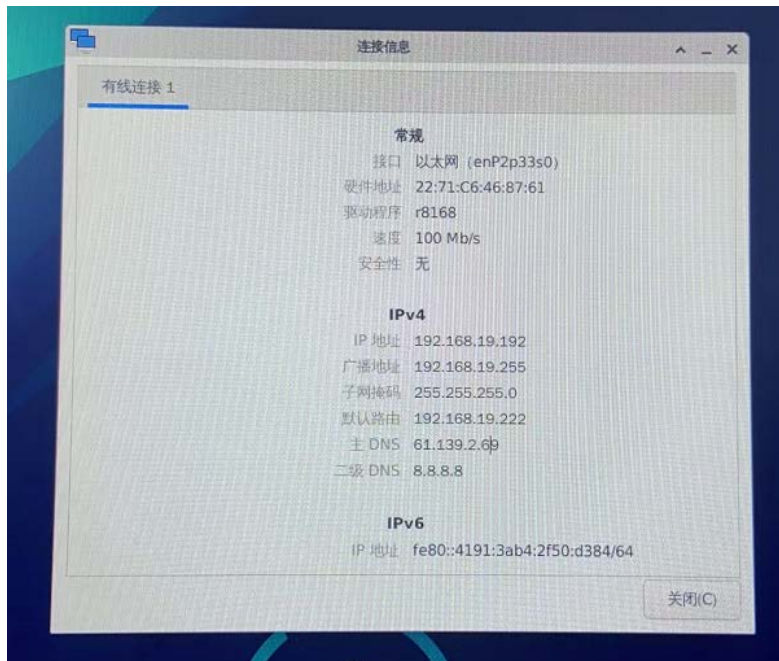
IBOX3588 配置为连接外部以太网或 Wi-Fi 无线网络后即可上网。

连接无线网络前，请确保设备已安装 Wi-Fi 天线。

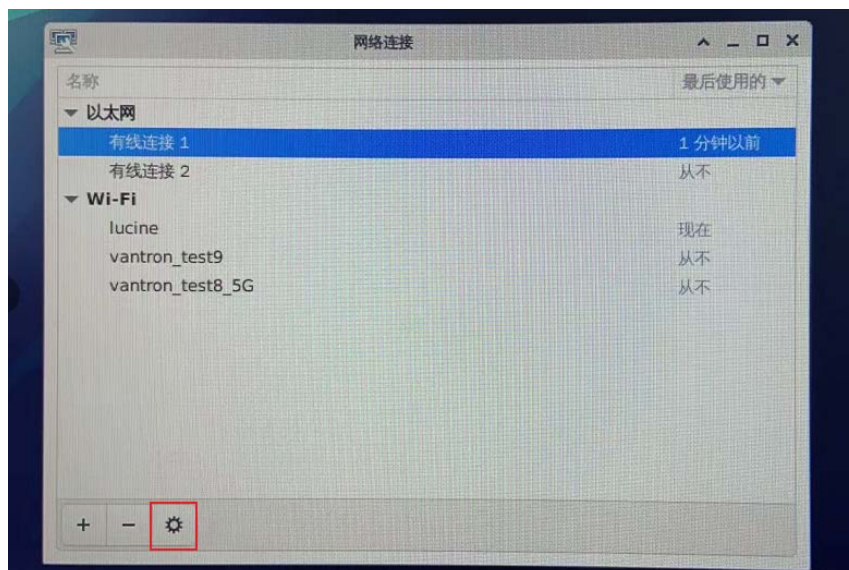
1. 点击菜单栏上的网络连接图标（双电脑/Wi-Fi 图标），从列表中搜索目标网络；



2. 如需查看当前的网络连接，用户可以将鼠标移动至网络连接图标处，然后点击鼠标右键，选择**连接信息**：



3. 如需编辑网络配置，用户可以在网络连接图标处点击鼠标右键，选择**编辑连接**，然后选择一个连接点，再点击齿轮图标编辑网络设置。



6.3 Ubuntu 环境升级镜像

6.3.1 环境准备

- IBOX3588
- 运行 Ubuntu 的主机（推荐 Ubuntu 18.04 或以上系统）
- USB Type-A 转 Type-C 数据线
- IBOX3588 设备软件发布包

6.3.2 烧写镜像

1. 使用 USB Type-A 转 Type-C 数据线连接设备和主机；
2. 打开终端，输入以下命令，安装 ADB 工具；

```
$ sudo apt-get install adb -y
```

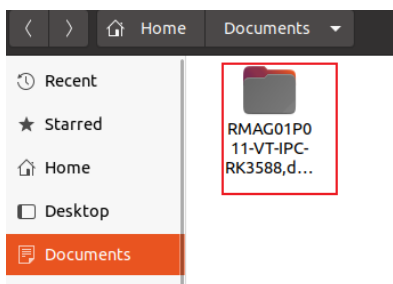
3. 检查设备是否通过 ADB 连接至 Ubuntu 主机；

```
$ adb devices -l
```

4. 在终端中输入 `adb shell`，便于在设备上执行 shell 命令；
5. 输入 `reboot loader`，设备将自动重启并进入 bootloader 模式；

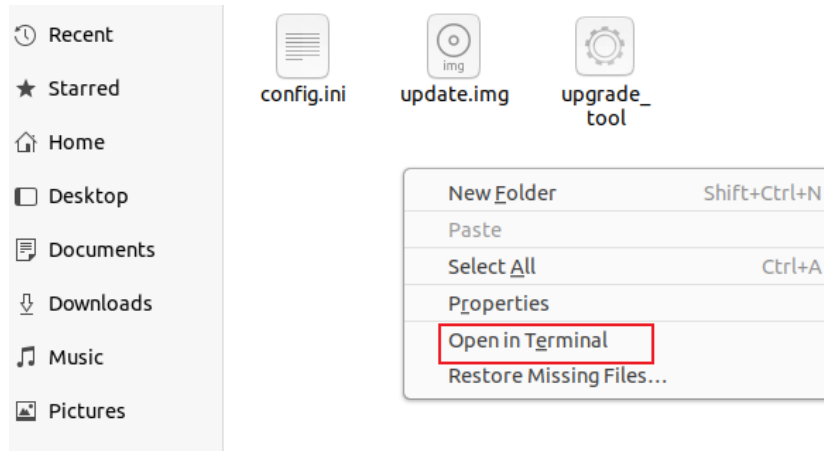
```
e@ubuntu:~$ adb shell
root@linaro-alip:/# reboot loader
root@linaro-alip:/# @ubuntu:~$
```

6. 将万创提供的升级镜像包拷贝至 Ubuntu 主机（例如，拷贝至文档文件夹）；



7. 将镜像包解压缩并打开 `img` 文件夹；

8. 在空白处点击鼠标右键并选择在**终端中打开**，然后在新的终端内执行后续命令；



9. 输入以下命令下载升级镜像并开始升级；

```
sudo ./upgrade_tool uf update.img
```

10. 输入 sudo 密码后开始下载系统镜像；

```
root@ubuntu: ~/Documents/RMAG01P011-VT-IPC-RK3588,debian,V1.0.0,2022-08-01/img
$ sudo ./upgrade_tool uf update.img
[sudo] password for root:
Using /home/root/Documents/RMAG01P011-VT-IPC-RK3588,debian,V1.0.0,2022-08-01/img/config.ini
Program Log will save in the /root/upgrade_tool/log/
Loading firmware...
Support Type:RK3588      FW Ver:1.0.00   FW Time:2022-07-30 16:22:22
Loader ver:1.0b Loader Time:2022-07-30 16:02:36
Download Image Total(5829691K),Current(2798535K)
```

11. 下载完成后，系统将立即开始升级，升级完成后，设备将自动重启。

```
Program Log will save in the /root/upgrade_tool/log/
Loading firmware...
Support Type:RK3588      FW Ver:1.0.00   FW Time:2022-07-30 16:22:22
Loader ver:1.0b Loader Time:2022-07-30 16:02:36
Upgrade firmware ok.
```

第 7 章 Ubuntu 系统手册

7.1 系统介绍

Ubuntu 是 Linux 系统的一个发行版本。IBOX3588 当前运行的 Ubuntu 版本为 Ubuntu 20.04. LTS。

7.1.1 用户名与密码

开机后，用户需登陆账号进入系统。

- 用户名：linaro
- 密码：linaro

7.1.2 系统信息

1. 为方便操作，请使用键盘、鼠标和显示器连接设备；
2. 设备通电，输入密码后登录默认用户（linaro）；
密码：linaro
3. 点击屏幕左下角**显示应用程序**图标 ，进入应用程序；
4. 依次点击**设置 > 关于**，可以查看设备系统的相关信息。

7.2 系统设置

点击**进入应用程序**，用户可以点击或搜索单个应用，使用相关功能。也可以进入**设置**页面对系统基本设置进行修改。

7.2.1 语言

用户可以通过**设置 > 区域和语言**，设置系统的显示语言和格式，也可以配合**语言支持**应用添加或删除输入语言。所有更改将在系统重启后生效。

如果使用命令更改系统语言，操作步骤与 6.2.1 中的步骤一致。

7.2.2 日期和时间

用户可以点击**设置 > 日期和时间**，打开设置日期和时间页面：

1. 点击**解锁**并输入密码进行设置；
2. 打开**自动设置日期和时间**，保持设备时间与网络时间一致，或者，关闭该选项，手动输入日期和时间；
3. 选择/搜索目标时区并设置时间格式；
4. 点击**锁定**，保存设置；
5. 退出并等待设置生效。

7.2.3 网络/Wi-Fi

IBOX3588 配置为连接外部以太网或 Wi-Fi 无线网络后即可上网。用户也可以在**设置 > 网络**页面，添加 VPN，使用虚拟网络。

连接 Wi-Fi 前，请确保设备已安装 Wi-Fi 天线。

用户可以输入命令，查看网络接口信息：`ip link show`。

7.2.4 用户

输入密码解锁当前用户后，用户可以对该账号进行编辑，也可以添加或删除新的用户。

7.3 Ubuntu 环境升级镜像

7.3.1 环境准备

- IBOX3588
- 运行 Ubuntu 系统的主机（推荐 Ubuntu 18.04 或以上系统）
- USB Type-A 转 Type-C 数据线
- 设备软件发布包

7.3.2 烧写镜像

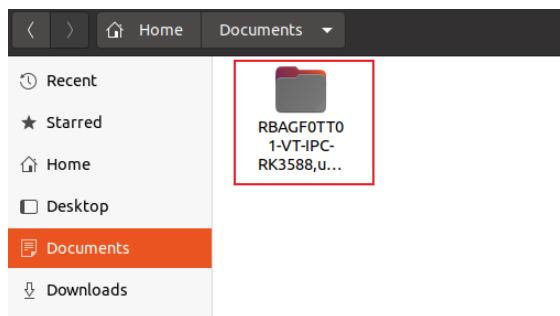
1. 使用 USB 数据线连接设备和主机；
2. 打开终端，输入以下命令，安装 ADB 工具；

```
$ sudo apt-get install adb -y
```

3. 检查设备是否通过 ADB 连接至 Ubuntu 主机；

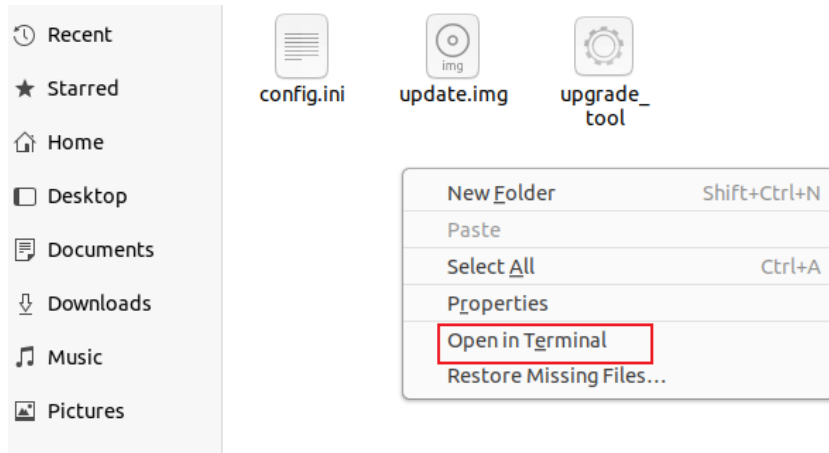
```
$ adb devices -l
```

4. 在终端中输入 `adb shell`，便于在设备上执行 shell 命令；
5. 输入 `reboot loader`，设备将自动重启并进入 bootloader 模式；
6. 将万创提供的升级镜像包拷贝至 Ubuntu 主机（例如，拷贝至文档文件夹）；



7. 将镜像包解压缩并打开 `img` 文件夹；

8. 在空白处点击鼠标右键并选择在**终端中打开**，然后在新的终端内执行后续命令；



9. 输入以下命令下载升级镜像并开始升级：

```
sudo ./upgrade_tool uf update.img
```

10. 输入 `sudo` 密码后开始下载系统镜像；
11. 下载完成后，系统将立即开始升级，升级完成后，设备将自动重启。

7.4 ROS 教程

请参考 <http://wiki.ros.org/ROS/Tutorials>，了解 ROS 系统详细信息。

第 8 章 废弃处理与质保

8.1 废弃处理

当设备到了使用期限，为了环境 and 安全，建议您适当地处理设备。

处理设备前，请备份您的数据并将其从设备中删除。

建议在处理前拆解设备，以符合当地法规。请确保废弃的电池已按照当地关于废物处理的规定进行处理。电池具有爆炸性，请勿将其扔进火中或放入普通垃圾桶中。标有 "爆炸性" 标志的产品或产品包装不应该按照家庭垃圾处理，应当送到专门的电气和电子垃圾回收/处理中心。

妥善处理这类废物有助于避免对周围环境和人们的健康造成伤害和不利影响。请联系当地机构或回收/处理中心，了解更多相关产品的回收/处理方法。

8.2 质保

产品质保

万创向客户保证，万创或万创分包商制造的产品从万创发运时将严格符合双方商定的规格，不存在工艺和材料上的缺陷（由客户提供的除外）。万创的质保义务限于产品的更换或维修（由其自行决定）。如果出现质量问题，产品发货后，客户应当自开具发票之日起 **24 个月** 内，自付运费将产品返回万创工厂。经检查后，万创合理确认产品具有缺陷的，由万创承担质保责任。之后，由万创承担将产品发运给客户的运输费用。

保修期外的维修

万创将按照当时的服务费率为已过保修期的产品提供维修服务。只要市场有售，万创将根据客户要求向客户提供非保修期内的维修部件，但客户需提前下达采购订单。维修部件有 3 个月的延长保修期。

产品退回

任何根据上述条款被认定为有缺陷并在保修期内的产品，只有在客户收到并参照万创提供的退货授权（RMA）号码后，才能退回给万创。万创应在客户提出要求后的 3（三）个工作日内提供 RMA。万创应在向客户发出退货产品后，向客户提供新的发票。在客户因拒收或保修期内的缺陷而退回任何产品之前，应向万创提供在客户所在地检查该产品的机会。除非拒收或缺陷的原因被确定为万创的责任，否则经检查的产品不得退回万创。万创应在收到产品后的 14（十四）个工作日内，向客户发出缺陷产品的替换。如果万创由于其无法控制的原因而不能提供上述服务，万创应记录这种情况并立即通知客户。

附录 合规声明

FCC 声明

此设备经检测，符合 FCC 规则第 15 部分中关于 B 级数字设备的限制规定。这些限制的目的是为了在居住区中安装此设备时，可以提供合理的保护以防止有害干扰。此设备会产生、使用和辐射射频能量，如果未遵照制造商的使用手册安装和使用，可能会对无线电通信产生有害干扰。但是，这并不能确保在某些特定安装中绝不会产生干扰。如果此设备确实对无线电或电视机接收信号造成有害干扰，而这一点可以通过关闭和打开设备来确定，那么建议用户尝试使用以下一种或多种措施来消除干扰：

- 调整接收天线的方向或重新放置。
- 扩大设备与接收器之间的距离。
- 将设备连接至与接收器不同的电路。
- 请与代理商或有经验的无线电/电视技术人员联系获得帮助。

此设备符合 FCC 规则的第 15 部分。操作应符合以下两个条件：（1）该设备不会产生有害干扰，以及（2）本设备必须承受收到的任何干扰，包括可能导致意外操作的干扰。

注意：制造商对未经授权改装本设备而造成的任何无线电或电视干扰不承担任何责任。改装后，用户或将无权操作本设备。