



Orange Pi 4

用户手册





History

[illegible]



目录

一、 Orange Pi 4 的基本特性.....	1
1. 什么是 Orange Pi 4.....	1
2. Orange Pi 4 的用途.....	1
3. Orange Pi 4 是为谁设计的.....	1
4. Orange Pi 4 硬件参数.....	3
5. GPIO 规格.....	5
二、 开发板使用说明.....	6
1. 使用 OrangePi 4 要做哪些准备工作.....	6
2. 如何启动 OrangePi 4.....	6
3. 串口使用注意事项.....	7
三、 Android 系统的使用说明.....	8
1. 使用 MIPI 摄像头进行拍照和录像.....	8
2. 使用 MIPI 屏幕.....	8
3. 使用 adb.....	10
4. 双屏异显 demo 使用.....	10
5. PCIE 接口测试.....	12
四、 Linux 系统的使用说明.....	13
1. Linux 启动亮灯情况说明.....	13
2. 登录账号和密码.....	13
3. 扩展 rootfs 分区.....	13
4. 时区设置与中文支持.....	13
5. 测试录音与放音.....	15
6. HDMI 与 3.5 音频切换.....	15
7. WIFI 的配置方法.....	18
8. 使用蓝牙传输文件.....	19
9. 使用 OV13850 摄像头.....	19
10. 使用 MIPI 屏幕.....	20
11. PCIE 接口测试.....	21
12. 双屏显示说明.....	25
13. 测试 GPU.....	26
14. Chromium 网页浏览器.....	26



五、 Android 固件烧录.....	27
1. 进入 loader 模式烧录到 EMMC.....	27
2. 使用 SD 升级卡烧录到 EMMC.....	32
3. 使用 SD 卡启动安卓系统.....	33
六、 Linux 固件烧录.....	34
1. Etcher 的安装方式.....	34
2. 通过 Etcher 烧录 Linux 固件的方法.....	35
3. 通过脚本将 Linux 系统烧录到 EMMC Flash 芯片中.....	36
七、 Android 编译环境搭建.....	37
1. 获取 SDK 源码压缩包.....	37
2. 搭建编译环境.....	37
3. 编译 SDK 源码.....	38
八、 Linux 编译环境搭建.....	39
1. 获取 SDK 源码压缩包.....	39
2. 搭建编译环境.....	41
3. 编译 Linux 和 U-boot 源码.....	42
九、 串口调试工具介绍.....	44
1. 基于 Windows 平台的使用.....	45
2. 基于 Linux 平台的使用.....	49



一、Orange Pi 4 的基本特性

1. 什么是 Orange Pi 4

香橙派是一款开源的单板卡片电脑，新一代的 arm64 开发板，它可以运行 Android 8.1、Ubuntu 和 Debian 等操作系统。香橙派单板电脑使用瑞芯微 RK3399 芯片，同时拥有 4GB LPDDR4 内存。

2. Orange Pi 4 的用途

我们可以用它搭建：

- 一台计算机
- 一个网络服务器
- 游戏机
- 高清视频播放器
- 扬声器
- Android
-

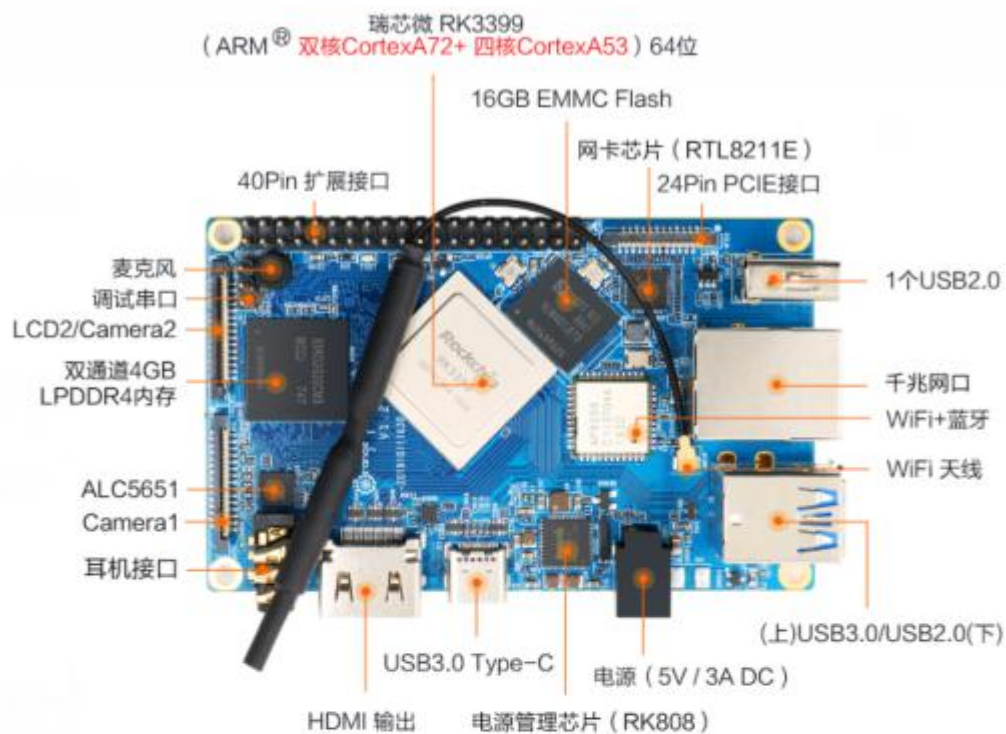
还有其他更多的功能，因为 Orange Pi 4 是开源的。

3. Orange Pi 4 是为谁设计的

Orange Pi 4 不仅仅是一款消费品，同时也是给任何想用技术进行创作创新的人设计的。它是一款非常简单、有趣、实用的工具，你可以用它去打造你身边的世界。



顶层视图



底层视图





4. Orange Pi 4 硬件参数

硬件参数	
CPU	Rockchip RK3399 (28 纳米 HKMG 制程) 6 核 ARM® 64 位处理器, 主频高达 2.0GHz 基于 big.LITTLE 大小核架构: 双核 Cortex-A72(大核)+四核 Cortex-A53(小核)
GPU	Mali-T864 GPU, 支持 OpenGL ES1.1/2.0/3.0/3.1, OpenVG1.1, OpenCL, DX11, 支持 AFBC
内存+板载存储	双通道 4GB LPDDR4 + 16GB EMMC flash 双通道 4GB LPDDR4 + EMMC (默认空贴)
板载 WIFI+蓝牙	AP6256, IEEE 802.11 a/b/g/n/ac, BT5.0
板载网络	10/100M/1000M 以太网 RJ45
网络芯片	RTL8211E
音频	输出: 3.5mm 耳机孔、HDMI2.0a、DP1.2 输入: MIC
视频输出	1 x HDMI 2.0 (Type-A), 支持 4K@60 帧输出 1 x DP 1.2 (DisplayPort), 支持 4K@60 帧输出 支持双通道 MIPI-DSI (每通道 4 线)
摄像头	2 x MIPI-CSI 摄像头接口 (MIPI_RX0、MIPI_TX1/RX1)
USB	2 x USB2.0 HOST, 1x USB3.0 HOST, 1 x USB3.0 Type-C
RTC	支持 RTC, 板载备用电池接口
调试	3pin 调试串口
预留接口	GPIO1 40pin 排针 (1 x I2S、2 x I2C、1 x SPI/UART、8 x GPIO) GPIO2 24pin PCIE 接口

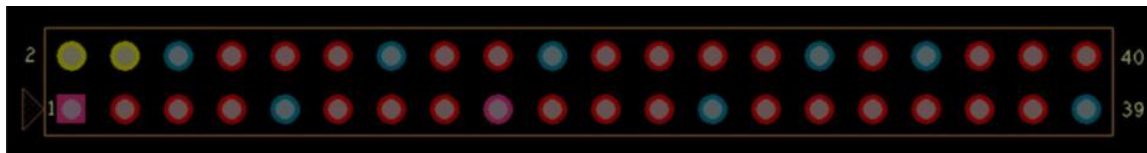


按键	1 x 升级键， 1x 复位键
电源	DC 5V/3A TYPE-C 5V/3A
LED 灯	电源指示灯、状态指示灯
支持的操作系统	Android8.1, Ubuntu, Debian 等操作系统
● Interface definition	
产品尺寸	91mm*55.7mm
OrangePi™是深圳市迅龙软件有限公司的注册商标	



5. GPIO 规格

下图是 Orange Pi 4 的 GPIO 引脚功能图：



PIN1	VCC3V3_SYS	PIN2	VCC5V0_SYS
PIN3	I2C2_SDA_3V0	PIN4	VCC5V0_SYS
PIN5	I2C2_SCL_3V0	PIN6	GND
PIN7	GPIO4_C6/PWM1	PIN8	I2C3_SCL
PIN9	GND	PIN10	I2C3_SDA
PIN11	GPIO1_A1	PIN12	GPIO1_C2
PIN13	GPIO1_A3	PIN14	GND
PIN15	GPIO2_D4	PIN16	GPIO1_C6
PIN17	GND	PIN18	GPIO1_C7
PIN19	UART4_TX	PIN20	GND
PIN21	UART4_RX	PIN22	GPIO1_D0
PIN23	SPI1_CLK	PIN24	SPI1_CS _n 0
PIN25	GND	PIN26	GPIO4_C5
PIN27	I2C2_SDA	PIN28	I2C2_SCL
PIN29	I2S0_LRCK_RX	PIN30	GND
PIN31	I2S0_LRCK_TX	PIN32	I2S_CLK
PIN33	I2S0_SCLK	PIN34	GND
PIN35	I2S0_SDIO	PIN36	I2S0_SD00
PIN37	I2S0_SDI1SD03	PIN38	I2S0_SDI2SD02
PIN39	GND	PIN40	I2S0_SDI3SD01



二、开发板使用说明

1. 使用 OrangePi 4 要做哪些准备工作

要启动你的OrangePi 4，请先准备好以下硬件：

- OrangePi 4 开发板
- TF 卡，最小 8GB 容量，class 10 级，建议使用品牌 TF 卡，如：闪迪 16G TF 卡（可选、Android系统无需TF卡）
- 一个 5V/3A或更大功率的电源适配器（DC和TYPEC接口的皆可）
- USB转TTL串口(可选，用于调试或PC上进行操作)
- 一台支持HDMI输入的显示器或者电视（或选购LCD配件）
- 一套USB键盘鼠标，同时连接更多USB的设备时还需要USB HUB
- 一台编译用的主机，配置最好满足以下条件：
 - 64 位 CPU
 - 8 GB 及以上内存
 - 100GB 以上的空闲磁盘空间
 - 操作系统可以是 Ubuntu14.04 Ubuntu18.04

2. 如何启动 OrangePi 4

将开发板的 HDMI 接口通过 HDMI 线与显示器相连，将电源适配器插入带电的插座上，电源线接口插入开发板，开发板 EMMC 预装有安卓系统，上电会自动开机。系统正常启动过程中，电源灯亮，状态灯闪烁。

如需使用 linux 系统，请参考 linux 系统烧录章节。

对于不带 EMMC 的 OrangePi 4 开发板，如需使用安卓需按照第五章第 3 小节的步骤烧录安卓系统到 SD 卡。



3. 串口使用注意事项

OrangePi 4 使用的波特率是 1500000，如果串口板使用的是cp210x 的芯片，在ubuntu18.04 会出现乱码的现象，ubuntu14.04 则不存在这个问题。经过验证，使用低版本内核的cp210x驱动可以解决这个问题，所以Ubuntu18.04 需要客户自己移植低版本的驱动。

下面给出参考方法。

```
git clone https://github.com/baiywt/cp210x.git
cd cp210x/
make
```

删除ubuntu 18.04 系统原本的cp210x驱动。

```
sudo rm -rf /lib/modules/$(uname -r)/kernel/drivers/usb/serial/cp210x.ko
```

拷贝编译好的cp210x驱动到系统目录

```
sudo cp ./cp210x.ko /lib/modules/$(uname -r)/kernel/drivers/usb/serial/
```

卸载原来的驱动

```
sudo rmmod cp210x
```

安装新的驱动

```
sudo modprobe cp210x
```

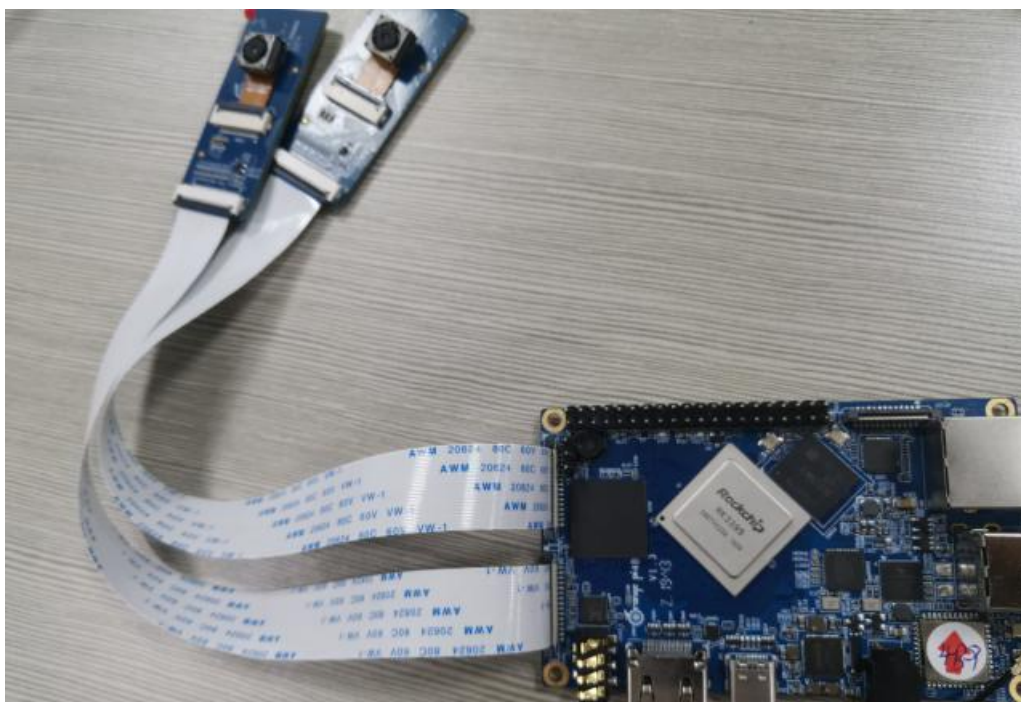


三、Android 系统的使用说明

1. 使用 MIPI 摄像头进行拍照和录像

OrangePi 4 在 android 系统下，可以搭配 OV13850 摄像头进行拍照和录像，连接摄像头到 OrangePi4 的 MIPI 接口，开机进入 Android 系统，用系统自带的 Camera 应用即可完成拍照和录像，操作跟 Android 手机是一样的。

OrangePi 4 最多可同时连接两个 MIPI 摄像头，一个为前置、另一个为后置。
连接方法如下图所示：



2. 使用 MIPI 屏幕

官网释放安卓镜像支持 MIPI 屏幕与 HDMI 输出，将屏连接到 LCD1 接口即可。

如需连接两个 MIPI 屏幕，需要关闭 camera2 的支持，打开 LCD2 的配置。具体可参考安卓源码中 RKDocs 文件夹下的文档。



连接方法如图：





3. 使用 adb

第一次启动如需更改系统文件，需要关闭安全验证（注：安装 apk 不需要关闭），关闭安全验证后需要重启。

```
adb root
adb disable-verity
adb reboot
```

重新挂载

```
adb root
adb remount
```

传送文件

```
adb push example.txt /system/
```

注意事项: 有的 USB Type-C 线不支持 adb ,所以遇到这种情况可以换根 USB Type-C 线

4. 双屏异显 demo 使用

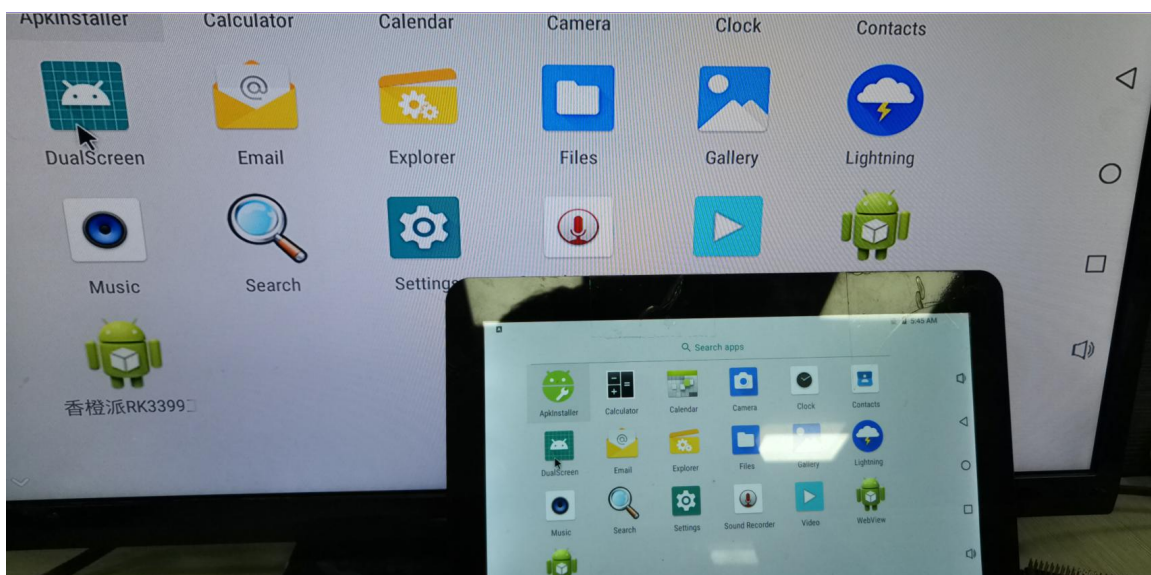
- 1) OrangePi 4 连接 HDMI 显示器、LCD1 连接 mipi 屏幕
- 2) 安装 dualscreen.apk、拷贝 test.mp4 文件到安卓系统的/sdcard/目录，可用以下命令完成

```
adb install dualscreen.apk
adb push test.mp4 /sdcard/
```

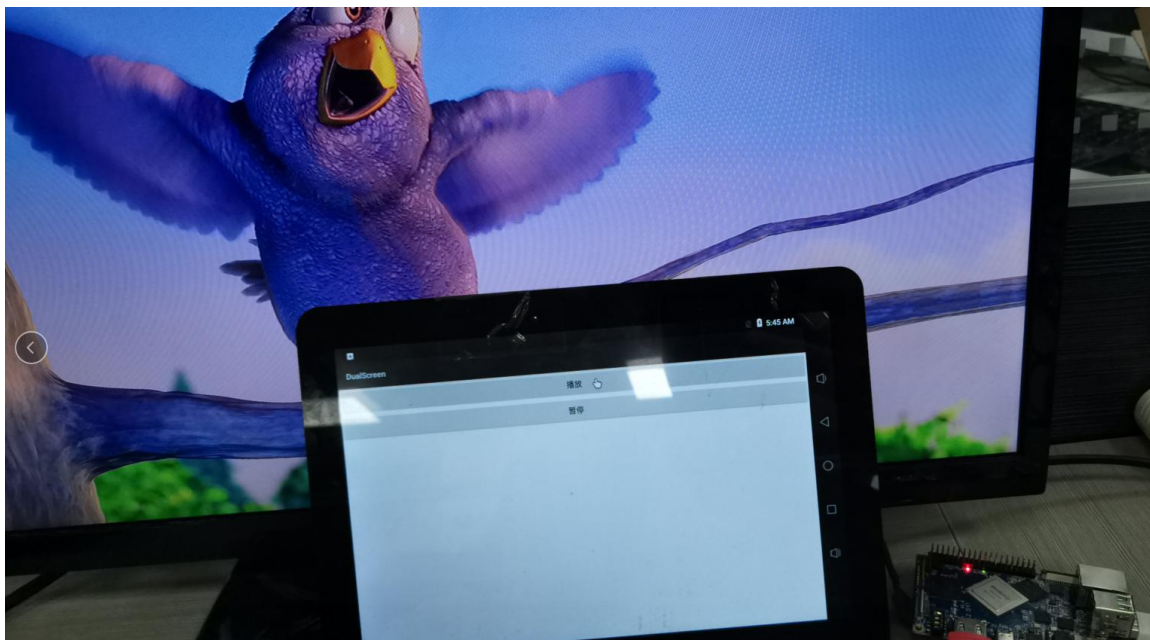
- 3) 在桌面点击 Settings-» Apps & notifications-» App info-» DualScreen-» Permissions-» 打开 Storage 权限



4) 点击 DualScreen 图标



5) 点击播放开始播放 test.mp4





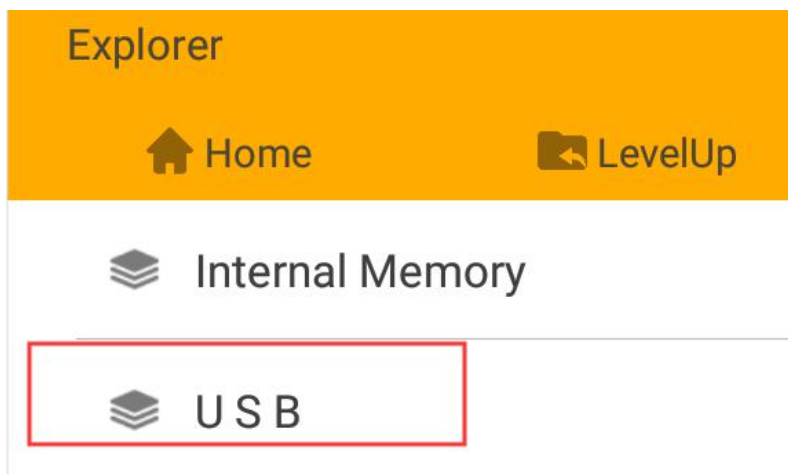
5. PCIE 接口测试

目前 Android 8.1 系统只适配了 PCIE 转 SATA（ASM1062）。其他 PCIE 器件的驱动暂时还没有适配。

将 PCIE 小板通过 24pin 反向排线接到板子接口上。再将 ASM1062 小板接到 PCIE 小板上，将硬盘接到 ASM1062，启动板子。系统会识别到设备，并自动挂载硬盘。



在安卓系统中可打开文件管理器看到挂载的设备。





四、Linux 系统的使用说明

1. Linux 启动亮灯情况说明

- 启动后板载 LED 灯会亮红灯，绿灯闪烁

2. 登录账号和密码

- 用户名 orangepi, 密码: orangepi

3. 扩展 rootfs 分区

系统第一次启动后会自动扩容，无需手动扩容。如果用户发现并没有扩容成功，可运行以下命令扩容。

```
resize-helper
```

4. 时区设置与中文支持

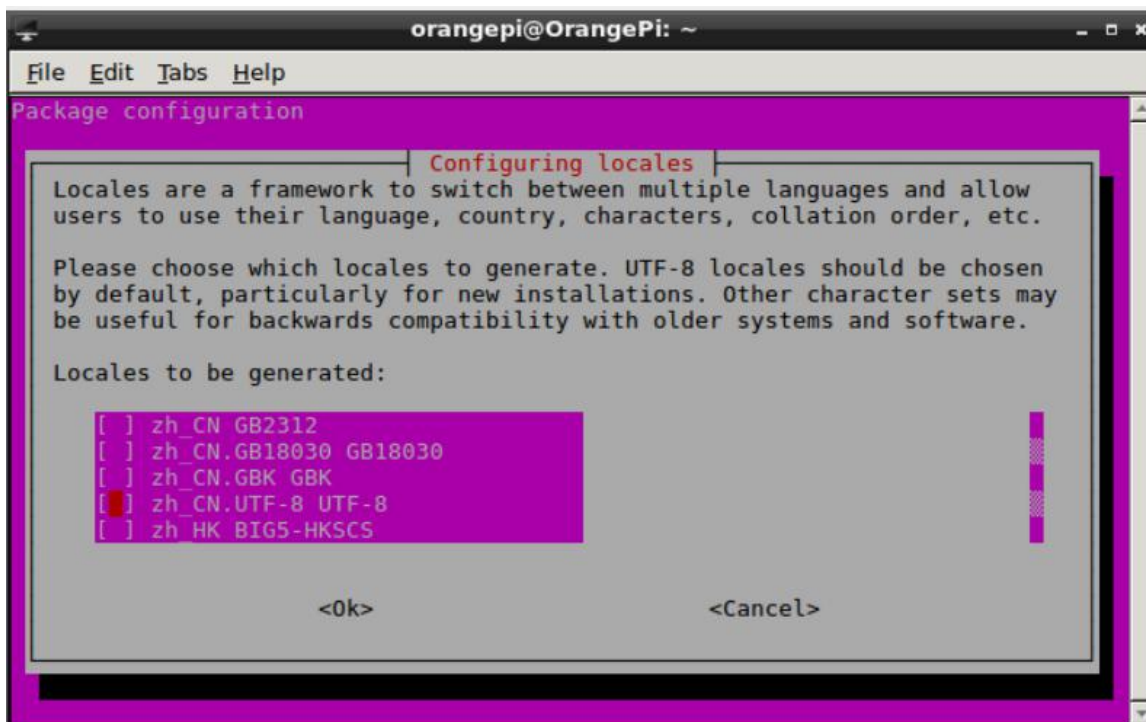
设置上海时间

```
cp /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
```

中文设置，打开终端，输入以下命令

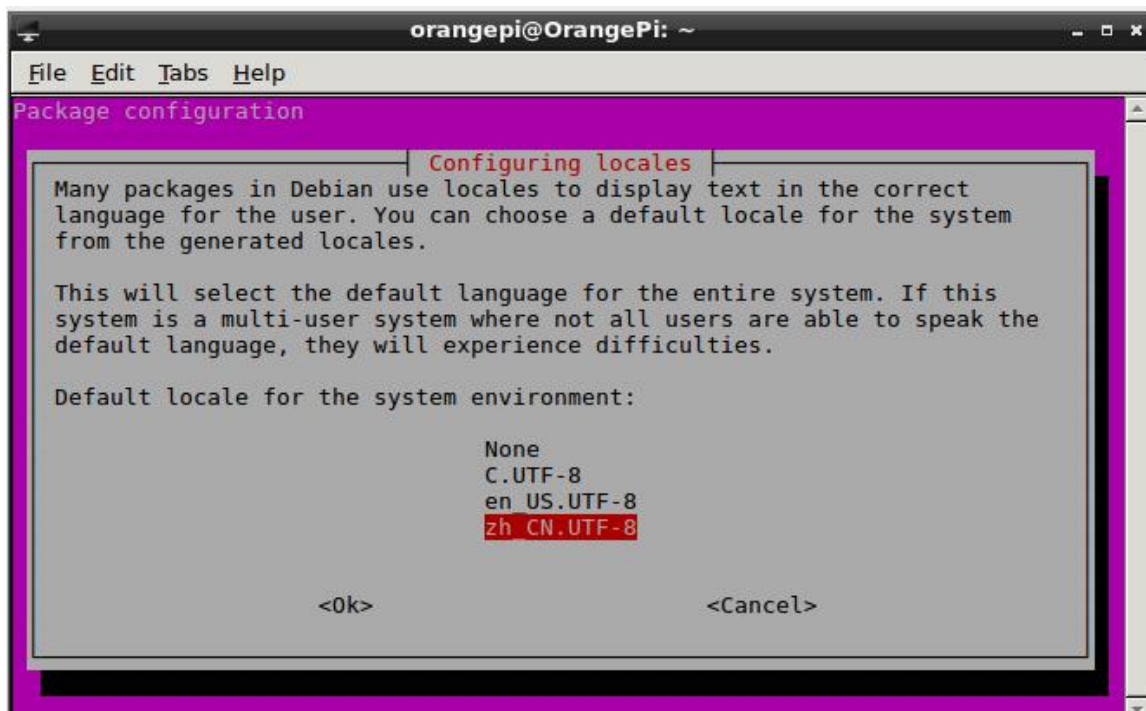
```
sudo dpkg-reconfigure locales
```

进入语言配置界面



然后往下找（键盘-按下键），在较后面，找到 zh_CN.UTF-8 UTF-8，如上图所示按空格选中，按回车确定。

然后来到如下界面，选择 zh_CN.UTF-8，确定，按下回车键。





出现如下信息，配置完成

```
orangePi@OrangePi:~$ sudo dpkg-reconfigure locales
Generating locales (this might take a while)...
 en_US.UTF-8... done
 zh_CN.UTF-8... done
Generation complete.
orangePi@OrangePi:~$
```

重启系统即可。

中文输入法安装

参考文档 <https://github.com/baiywt/docs>

5. 测试录音与放音

输入以下命令测试麦克风录音

```
arecord -Dhw:0,0 -r8000 -f cd audio.wav
```

3.5 耳机孔输出声音

```
aplay -Dhw:0,0 audio.wav
```

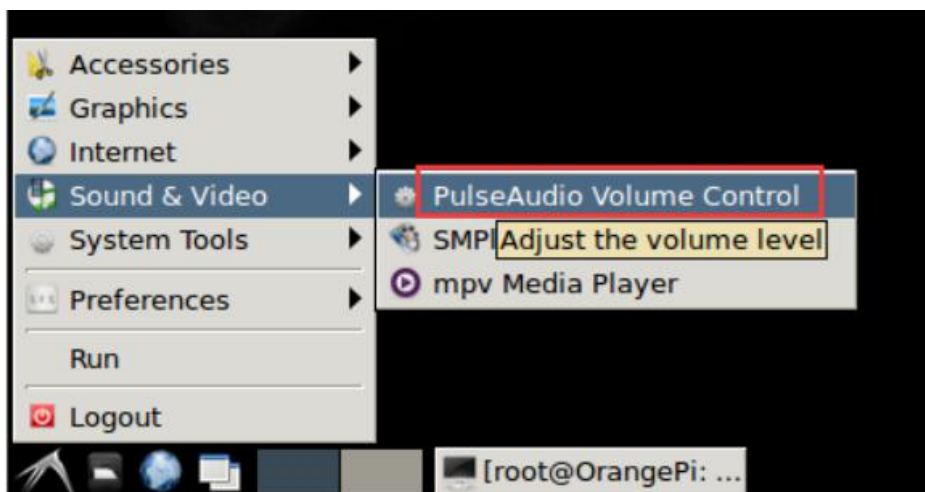
HDMI 输出声音

```
aplay -Dhw:1,0 audio.wav
```

也可修改/etc/asound.conf 文件，配置默认输出。

6. HDMI 与 3.5 音频切换

使用 smplayer 播放一段音频，打开 PulseAudio Volume Control 应用。

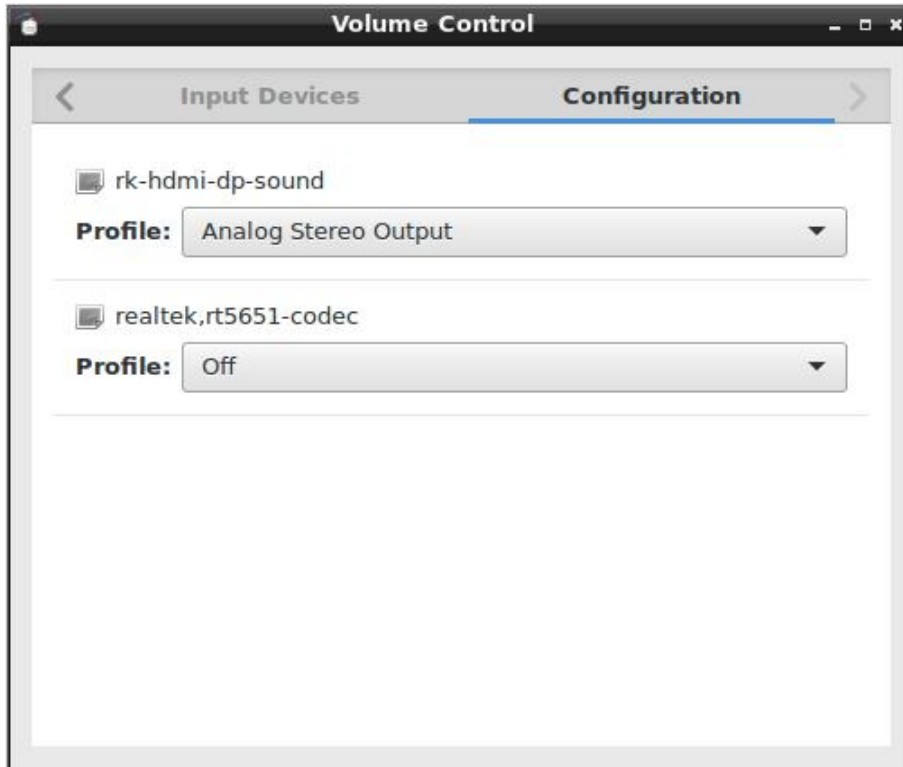


点击下图箭头，切换到 Configuration 一栏

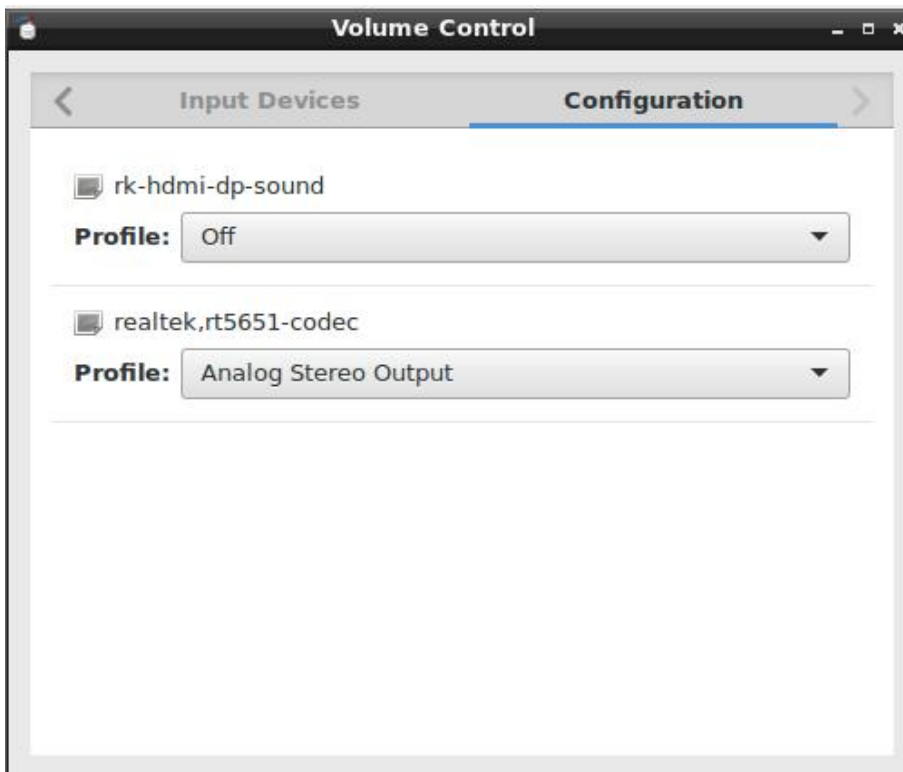




如下图配置，声音从 HDMI 输出。



声音从耳机输出。





7. WIFI 的配置方法

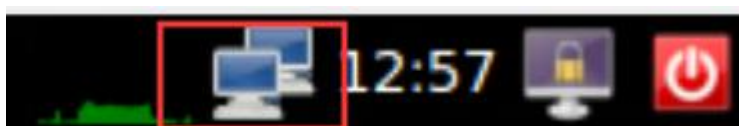
- 修改配置文件（适用于 server 版本）

在 `/etc/network/interface` 中加入下面的配置，然后重启即可

```
auto wlan0
iface wlan0 inet dhcp
wpa-ssid orangepi          //此处填入 WIFI 账号（现在是 orangepi）
wpa-psk orangepi           //此处填入 WIFI 密码（现在是 orangepi）
```

- 图形界面配置

点击桌面右下角的网络管理器图标



点击名为 `xunlong_orangepi_5G` 的热点。





输入密码即可连接



8. 使用蓝牙传输文件

执行如下命令初始化蓝牙

```
$ sudo bluetooth.sh
```

可下载 **blueman** 工具进行测试

```
sudo apt-get -y install blueman
```

9. 使用 OV13850 摄像头

启动系统后，打开命令行终端

执行以下命令，如出现如下信息说明摄像头工作正常，如没有此信息，请检查摄像头是否连接正确。

```
$ dmesg |grep Async  
[ 1.378648] rkisp1: Async subdev notifier completed
```

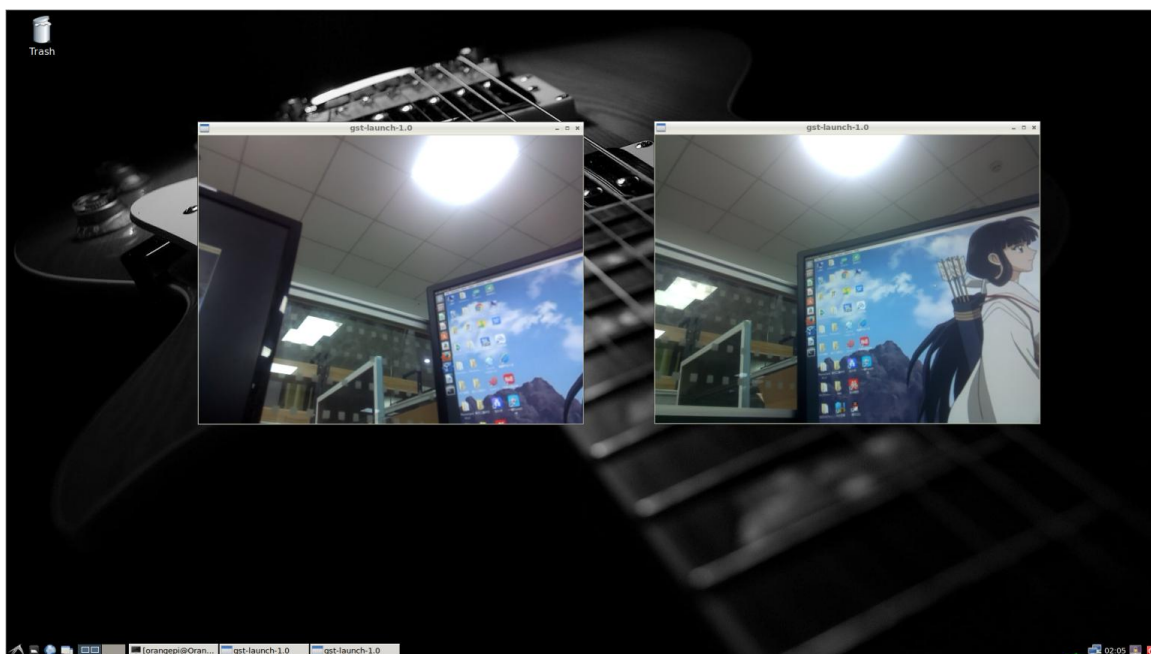
执行如下命令打开摄像头

```
$ test_camera-gst.sh
```



同时打开两个摄像头

```
$ test_camera-dual.sh
```



用命令拍照，在/home/orangepi 目录下生成 jpg 图片

```
$ test_camera-capture.sh
```

用命令录像，录像时使用了硬件编码。在/home/orangepi 生成视频文件 output.ts

```
$ test_camera-record.sh
```

10.使用 MIPI 屏幕

下载 linux 源码，在 dts 中做如下修改

```
--- a/arch/arm64/boot/dts/rockchip/rk3399-orangepi-lcd.dtsi
+++ b/arch/arm64/boot/dts/rockchip/rk3399-orangepi-lcd.dtsi
@@ -40,12 +40,12 @@
         max-x = <800>;
         max-y = <1280>;
         tp-size = <101>; // <911> for 8 inch // <101>

for 10 inch
-         status = "disable";
```




```
+             status = "okay";
                };
        };
        &dsi {
-             status = "disable";
+             status = "okay";
            panel@1 {
                compatible = "simple-panel-dsi";
                reg = <0>;
```

重新编译替换内核后，LCD1 接 MIPI 屏幕测试。

默认是竖屏，可使用 `xrandr -o left` 翻转屏幕。

11.PCIE 接口测试

目前 Linux 系统（v1.2 以上版本）适配了 PCIE 转 SATA（ASM1062）、RTL8822BE 网卡。其他 PCIE 器件的驱动暂时还没有适配。

● RTL8822BE 网卡测试

根据下图所示的方法插入 RTL8822BE 无线网卡模块后启动系统,系统将会自动识别并加载 8822be.ko 内核模块。



通过 `lsmod` 命令可以查看驱动是否成功加载，通过 `ifconfig` 命令可以查看 PCIE 无线网卡对应的网络节点。

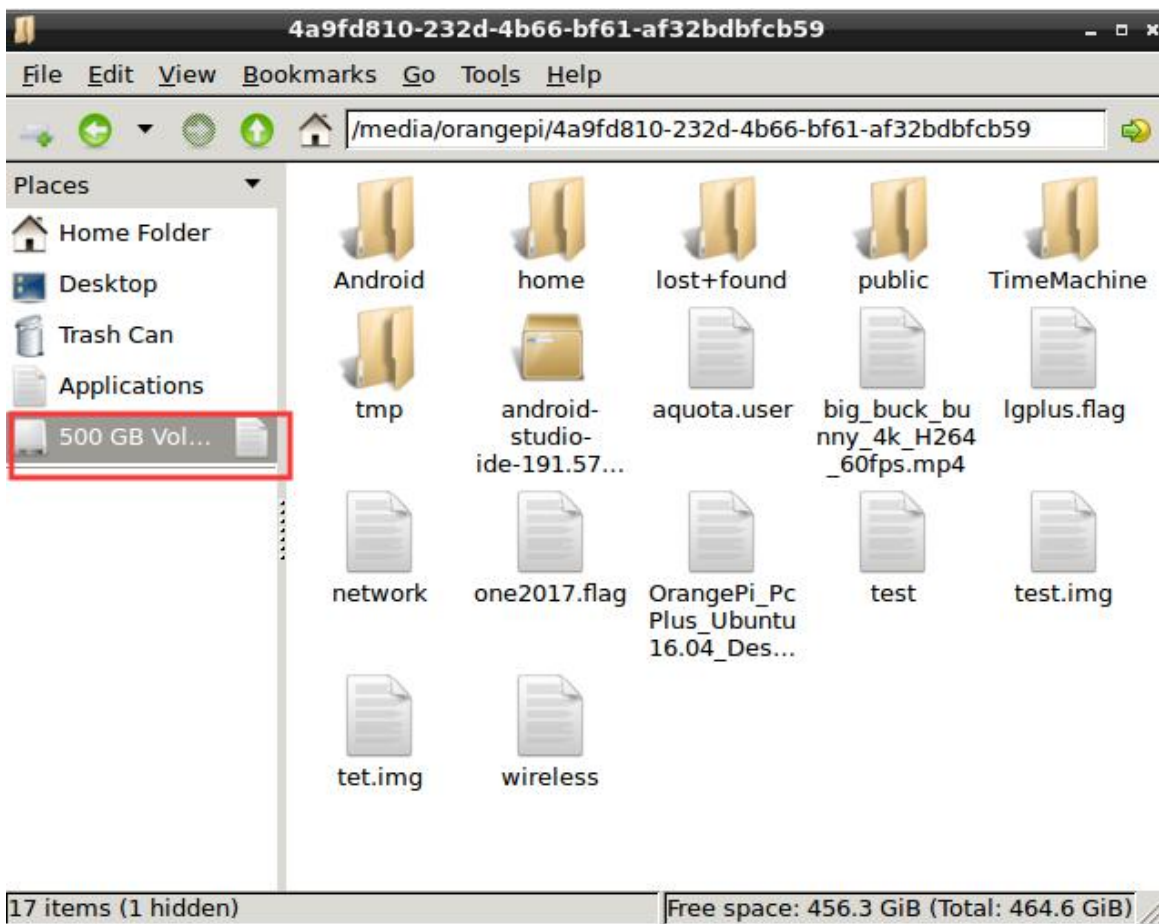
```
root@OrangePi:~# lsmod
Module                Size  Used by
8822be                 2420736  0

root@OrangePi:~# ifconfig wlp1s0
wlp1s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether f8:da:0c:5a:00:6f txqueuelen 1000  (Ethernet)
    RX packets 0  bytes 0 (0.0 B)
    RX errors 0  dropped 83  overruns 0  frame 0
    TX packets 0  bytes 0 (0.0 B)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions
```

● PCIE 转 SATA 测试

将 PCIE 小板通过 24pin 反向排线接到板子接口上。再将 ASM1062 小板接到 PCIE 小板上，将硬盘接到 ASM1062，启动板子。系统会识别到硬盘设备。





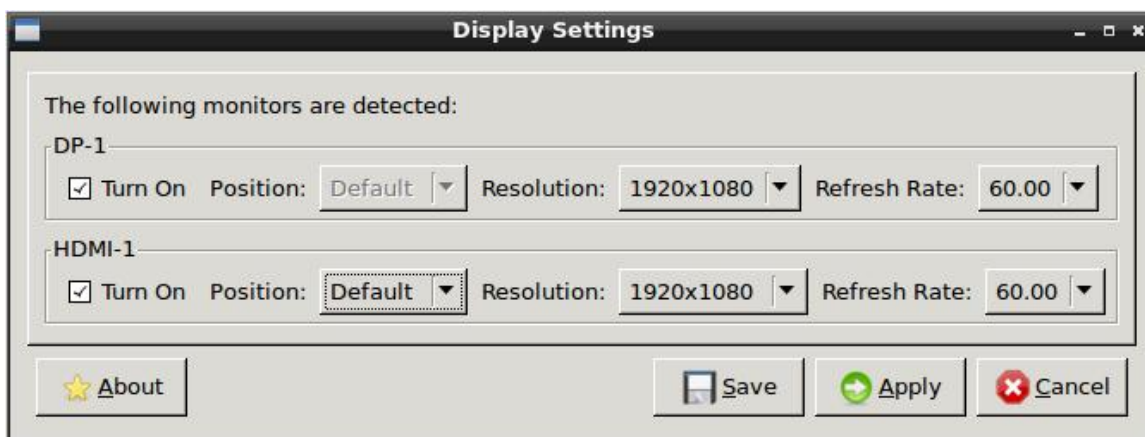


12. 双屏显示说明

OrangePi 4 除了 HDMI 接一个显示屏外，还可以支持 typeC 转 HDMI 接第二个显示屏。

在 Linux 系统中可以通过界面下对双屏显示模式进行设置。

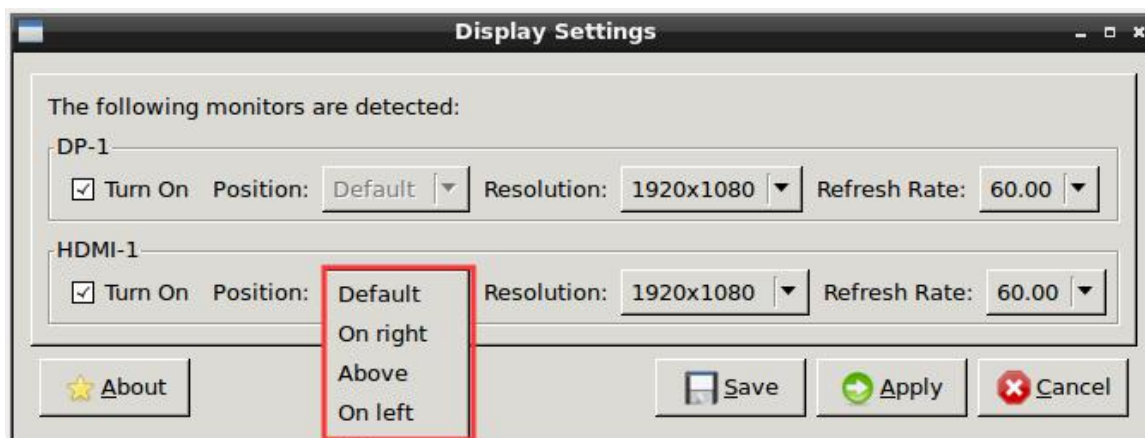
使用鼠标点击 menu-》 Preferences-》 Monitor Settings



可以看到当前系统下有两个显示设备，分别是 DP-1 和 HDMI-1。

双屏显示支持双屏同显模式，双屏异显模式。异显模式 支持 On right、Above、On Left、Below 四种模式。

在 Display Settings 中设置 Position，则可以切换双屏的显示模式。



除此之外，linux 系统还支持双 MIPI DSI 输出（官网镜像不支持，需要修改软件）。



13. 测试 GPU

安装测试软件

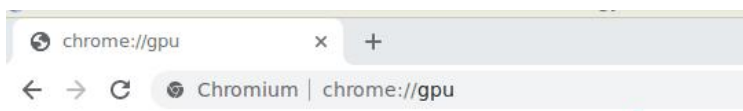
```
$ sudo apt install glmark2-es2
```

运行测试，可看到 GPU 跑分情况。

```
glmark2-es2
```

14. Chromium 网页浏览器

系统预装的 Chromium 网页浏览器已经默认启用硬件加速，支持 WebGL，可以输入网址 `chrome://gpu` 了解硬件加速情况。如下图所示。



Copy Report to Clipboard

Graphics Feature Status

- Canvas: **Hardware accelerated**
- Flash: **Hardware accelerated**
- Flash Stage3D: **Hardware accelerated**
- Flash Stage3D Baseline profile: **Hardware accelerated**
- Compositing: **Hardware accelerated**
- Multiple Raster Threads: **Enabled**
- Out-of-process Rasterization: **Disabled**
- Hardware Protected Video Decode: **Hardware accelerated**
- Rasterization: **Software only. Hardware acceleration disabled**
- Skia Renderer: **Disabled**
- Video Decode: **Hardware accelerated**
- Viz Display Compositor: **Enabled**
- Viz Hit-test Surface Layer: **Enabled**
- Vulkan: **Disabled**
- WebGL: **Hardware accelerated**
- WebGL2: **Hardware accelerated**



五、Android 固件烧录

准备工作：TYPE-C 数据线；android 固件、烧录工具及驱动（在官网百度云下载）
按照如下方式将系统烧录至 EMMC。

1. 进入 loader 模式烧录到 EMMC

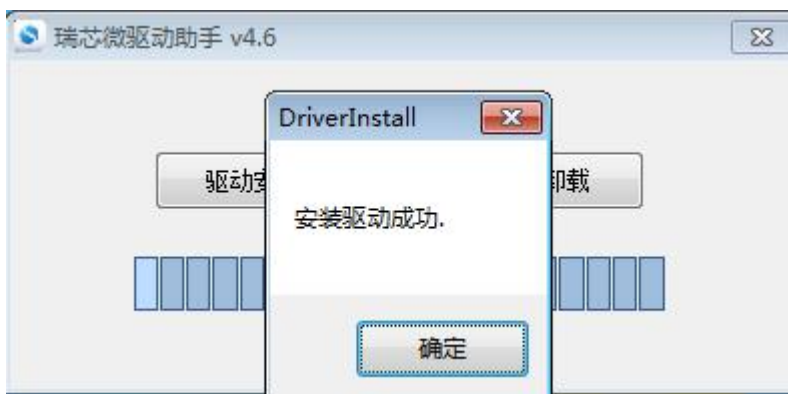
- 主机是 win7 系统

(1) 安装瑞芯微驱动



ADBDriver	2019/9/16 20:01	文件夹	
bin	2019/9/16 20:01	文件夹	
Driver	2019/9/16 20:01	文件夹	
Log	2019/9/16 20:02	文件夹	
config	2014/6/3 15:38	配置设置	1 KB
DriverInstall	2017/11/24 9:13	应用程序	490 KB
Readme	2018/1/31 17:44	文本文档	1 KB



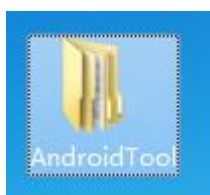


(2) OrangePi 4 先接上 DC 电源适配器，再通过 Type-C 数据线连接好 OrangePi 4 与 PC。

(3) 按住 rescover 键不放，轻按下 reset 键，此时 OrangePi 4 进入 loader 模式。
此时如连接串口 会看到如下信息：

```
#Boot ver: 0000-00-00#0.00  
empty serial no.  
normal boot.  
checkKey  
vbus = 1  
rockusb key pressed.
```

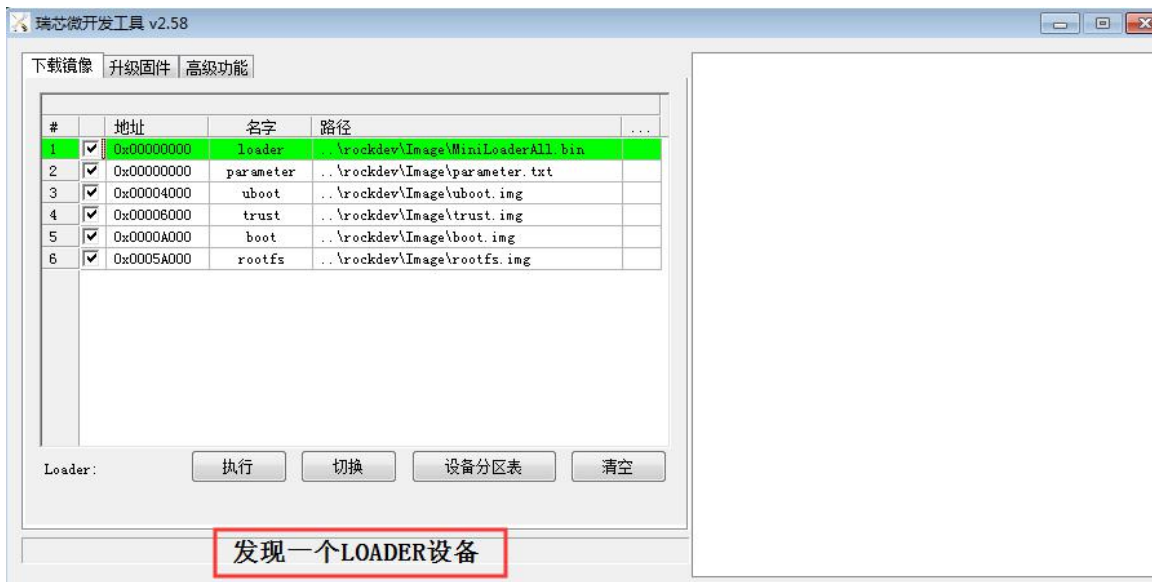
(4) 打开 AndroidTool 工具



rockdev	2019/9/16 13:58	文件夹	
AndroidTool_Release	2019/9/16 13:58	文件夹	
<hr/>			
名称	修改日期	类型	大小
bin	2019/9/16 13:58	文件夹	
Language	2019/9/16 13:58	文件夹	
Log	2019/11/21 12:26	文件夹	
AndroidTool	2019/7/4 13:59	应用程序	1,149 KB
Android开发工具手册_v1.2	2019/7/4 13:59	WPS PDF 文档	579 KB
confia.cfa	2019/7/4 13:59	CFG 文件	7 KB



此时工具下方会显示,识别到 Loader 设备,如下图所示。



(5) 点击升级固件, 选择固件路径后, 先擦除 Flash, 然后点击升级进行烧录。注意下图红色标记部分的顺序。





烧录完成



- 主机是 linux 系统

- (1) 准备 upgrade_tool 工具

```
$ unzip Linux_Upgrade_Tool_v1.39.zip
$ cd Linux_Upgrade_Tool_v1.39
$ sudo chmod +x ./upgrade_tool
```

- (2) OrangePi 4 接上电源适配器，Type-C 数据线连接好 OrangePi 4 与 PC。

- (3) 按住 rescover 键不放，轻按下 reset 键，此时 OrangePi 4 进入 loader 模式。

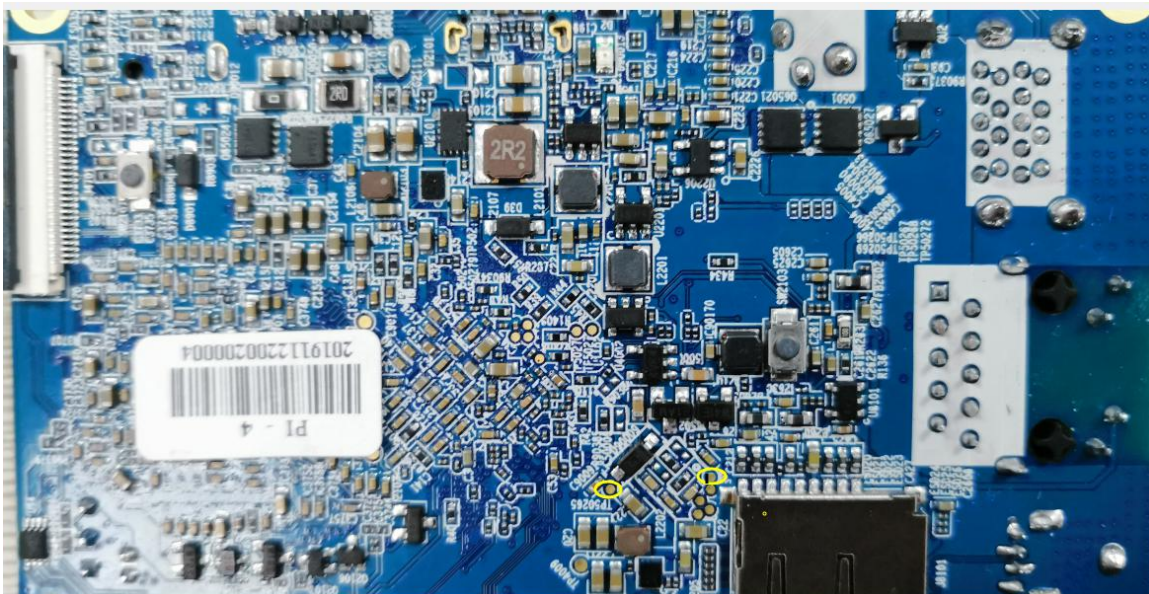
- (4) 烧录安卓固件

```
sudo ./upgrade_tool ef OrangePi_4_Android8.1_v1.0.img 擦除 EMMC
sudo ./upgrade_tool uf OrangePi_4_Android8.1_v1.0.img 烧录到 EMMC
```

注：如遇到无法进入 loader 模式的情况，可尝试 MaskRom 模式烧录。方法如下：



- 1、设备断开所有电源
- 2、拔出 SD 卡
- 3、用 USB Type-C 线连接好设备和主机
- 4、用金属镊子接通 OrangePi 4 预留的焊点（注意下图黄圈），并保持。



- 5、设备插入电源。
- 6、稍候片刻，之后松开镊子，设备应该就会进入 MaskRom 模式。





2. 使用 SD 升级卡烧录到 EMMC

准备工作：android 固件、一张 tf 卡、读卡器

该方式只能在 windows 系统中使用。

将读卡器插入主机，在 OrangePi 4 官方工具板块下载 SDDiskTool_v1.56，打开软件。

勾选固件升级，选择固件，最后点击开始创建。

请注意下图红色标记的地方。



完成后将 TF 卡插入 OrangePi 4 的卡槽，上电启动，启动代码会将固件烧写到 EMMC 中。如需看到烧写进度，可准备一根 HDMI 线，连接板子与显示器。烧写过程大约是 3~4 分钟。



3. 使用 SD 卡启动安卓系统

此方法适用于没有带 EMMC 的 OrangePi 4 开发板。

准备工作：带 SD 字样的 android 固件、一张 tf 卡、读卡器

从官网下载带 SD 字样的 android 固件（与 EMMC 启动的固件不同），以及 SDDiskTool_v1.59 烧录工具，注意版本是 1.59。

点击恢复磁盘格式化 TF 卡，勾选固件升级，选择固件，最后点击开始创建。注意下图红色标记部分。



将烧录好的 tf 卡插入 OrangePi 4 开发板，接通电源启动 android 系统。



六、Linux 固件烧录

我们可以通过 **Etcher** 将 Orange Pi 4 的 Linux 固件烧录到 TF 卡中，如果您购买的是没有贴 EMMC Flash 芯片的 Orange Pi 4 开发板，则只能通过 TF 卡来启动系统。Etcher 支持如下的操作系统：

- Linux（大多数发行版，如 Ubuntu）
- MacOS 10.9 和后续的版本
- Windows 7 和后续的版本

Etcher 软件安装包可以在其官网 <https://etcher.io/> 中下载，也可以从 Orange Pi 4 的官网下载页的**官方工具**中下载

1. Etcher 的安装方式

- Etcher 在 Windows 系统中的安装方式和普通软件的安装方式一样，这里不再赘述。
- Etcher 在 Ubuntu 和 Debian 系统中的安装方式如下

1. 增加 Etcher Debian 仓库：

```
$ echo "deb https://dl.bintray.com/resin-io/debian stable etcher" | sudo tee /etc/apt/sources.list.d/etcher.list
```

2. 下载 key

```
$ sudo apt-key adv --keyserver hkp://pgp.mit.edu:80 --recv-keys 379CE192D401AB61
```

3. 更新和安装

```
$ sudo apt-get update && sudo apt-get install etcher-electron
```

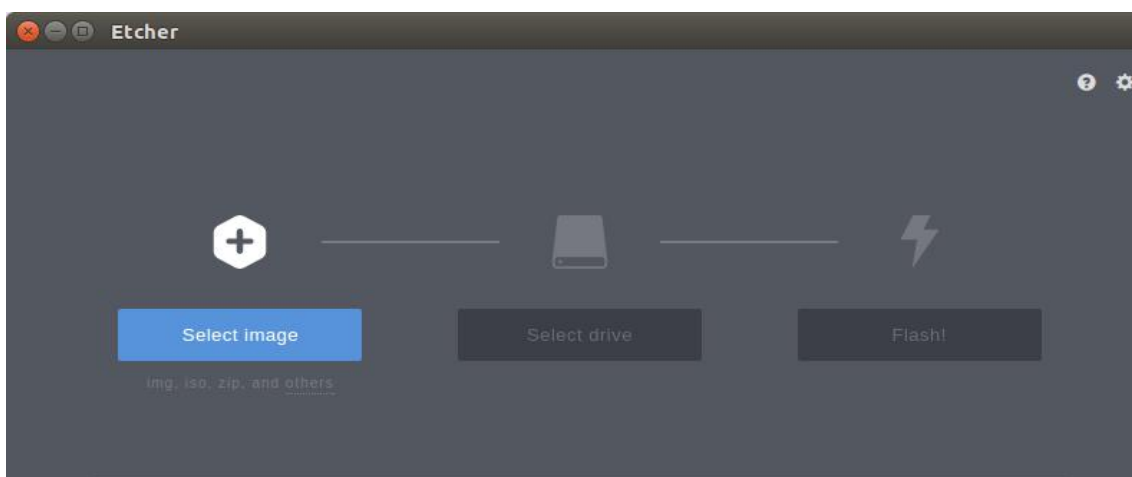



4. 卸载方式

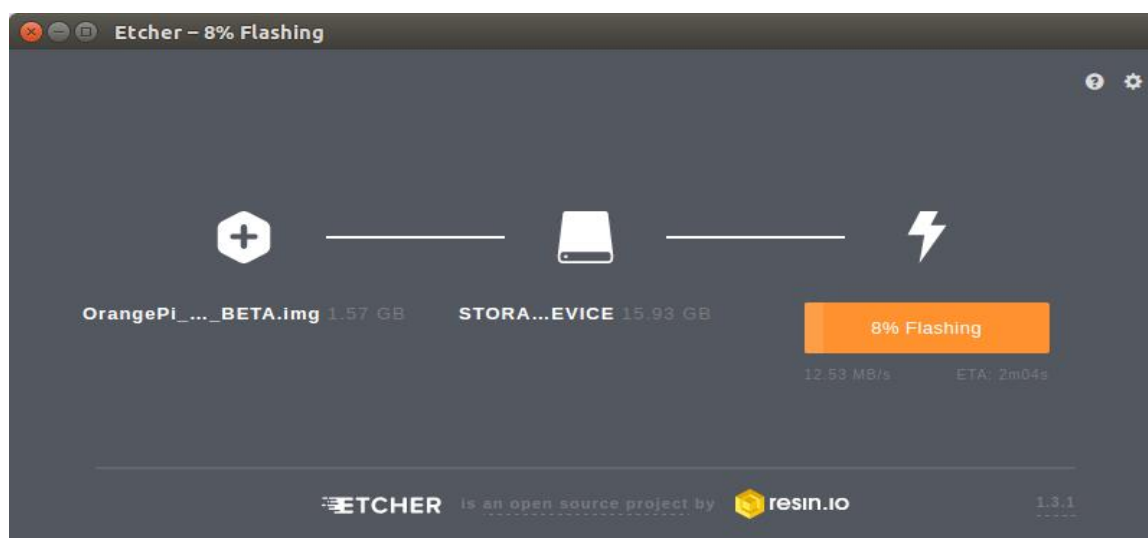
```
$ sudo apt-get remove etcher-electron  
$ sudo rm /etc/apt/sources.list.d/etcher.list && sudo apt-get update
```

2. 通过 Etcher 烧录 Linux 固件的方法

- 首先打开 Etcher，其界面如下图所示



- 然后通过 “Select image” 选择需要烧录的 Linux 固件
- 接着插入 TF 卡，Etcher 会自动识别相应驱动器
- 最后点击 “Flash!” 开始烧录，烧录完后，即可插入开发板启动系统





3. 通过脚本将 **Linux** 系统烧录到 **EMMC Flash** 芯片中

如果您购买的是贴有 EMMC Flash 芯片的 Orange Pi 4 开发板，通过 TF 卡启动 Linux 系统后，还可以通过 **install_to_emmc** 脚本将 Linux 系统烧录到 EMMC Flash 中。

在 Linux 终端中输入 **install_to_emmc** 命令，然后根据提示输入 y，就会开始自动烧录 Linux 系统到 EMMC Flash 中。烧录完成后，关闭电源，取出 TF 卡，再次上电后就会自动运行 EMMC Flash 中的 Linux 系统。

```
root@OrangePi:~# install_to_emmc

WARNING: EMMC WILL BE ERASED !, Continue (y/N)?  y
Erasing EMMC ...
Creating new filesystem on EMMC ...
  New filesystem created on /dev/mmcblk0.
Partitioning EMMC ...
  Creating boot & linux partitions
  OK.
Formatting fat partition ...
  fat partition formatted.
Formatting linux partition (ext4), please wait ...
  linux partition formatted.

Instaling u-boot to EMMC ...

Mounting EMMC partitions...
FAT partitions mounted to /tmp/_fatdir
linux partition mounted to /tmp/_extdir

Copying file system to EMMC ...

  Creating "fstab"

*****
Linux system installed to EMMC.
*****
```




七、Android 编译环境搭建

编译环境建议 Ubuntu 14.04 或者 Ubuntu 18.04。不建议用虚拟机。

1. 获取 SDK 源码压缩包

Android 源码包下载完后，首先需要将多个压缩文件合并成一个，然后进行解压。

```
$ mkdir OrangePi_4
$ cat rk3399-android-8.1.tar.gz* > rk3399-android-8.1.tar.gz
$ tar xvf rk3399-android-8.1.tar.gz -C OrangePi_4
```

2. 搭建编译环境

● 安装 JDK

Android 8.1 开发只能使用 `openjdk8` 的版本，高于或低于此版本以及 `oracle` 的 JDK 都会导致编译失败。`openjdk-8` 的安装命令如下：

```
$ sudo add-apt-repository ppa:openjdk-r/ppa
$ sudo apt-get update
$ sudo apt-get install openjdk-8-jdk
```

● 安装平台支持软件

对于 Ubuntu14.04:

```
$ sudo apt-get update
$ sudo apt-get install git gnupg flex bison gperf build-essential \
zip curl zlib1g-dev gcc-multilib g++-multilib libc6-dev-i386 \
lib32ncurses5-dev x11proto-core-dev libx11-dev lib32z1-dev ccache \
libgl1-mesa-dev libxml2-utils xsltproc unzip
```



```
$ sudo apt-get install u-boot-tools
```

3. 编译 SDK 源码

清除中间文件

```
cd rk3399-android-8.1
cd uboot && make distclean
cd kernel && make distclean
```

使用以下命令编译完整镜像

```
./make.sh -F -M -u
```

编译完成后，在以下目录生成镜像文件。

```
cd rockdev/Image-rk3399_mid/
tree -L 2
├── boot.img
├── kernel.img
├── MiniLoaderAll.bin
├── misc.img
├── oem.img
├── parameter.txt
├── pcba_small_misc.img
├── pcba_whole_misc.img
├── recovery.img
├── resource.img
├── system.img
├── trust.img
├── uboot.img
├── update.img
└── vendor.img
```

0 directories, 15 files

其中 `update.img` 是完整固件，可参照《Android 固件烧录》章节烧录至 EMMC。



八、Linux 编译环境搭建

推荐使用主机环境为 64 位ubuntu18.04 。OrangePi 4 开发板 ubuntu 18.04 镜像需要主机环境是ubuntu18.04 才能编译成功, debian9 和ubuntu16.04 则可以在ubuntu 14.04 上编译。

1. 获取 SDK 源码压缩包

● Orange Pi Linux 源码下载器

Orange Pi 4 的 Linux 源码已经上传到 GitHub, 内核版本为 Linux 4.4, 我们可以使用 OrangePi Linux 源码专用的下载器进行下载, 获取下载器源码的方式如下所示:

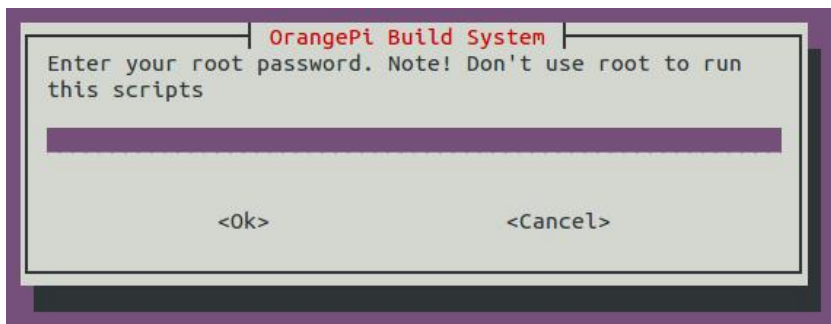
```
$ sudo apt-get install git
$ git clone https://github.com/orangepi-xunlong/OrangePi_Build.git
$ cd OrangePi_Build
$ ls
Build_OrangePi.sh  lib  README.md
```

注意事项: 源码要放到本地硬盘上, 不要使用共享目录, 或者另外挂载进来的硬盘, 否则影响正常编译。

● 运行下载器

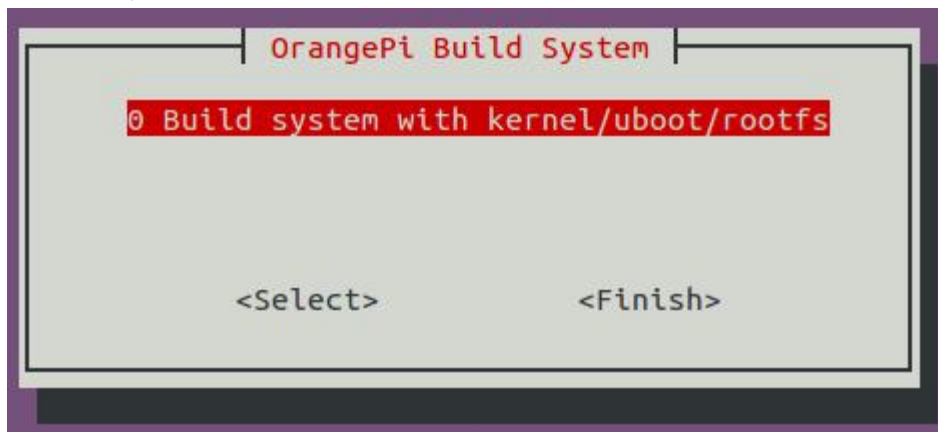
```
$ ./Build_OrangePi.sh
```

输入 root 密码, 然后回车

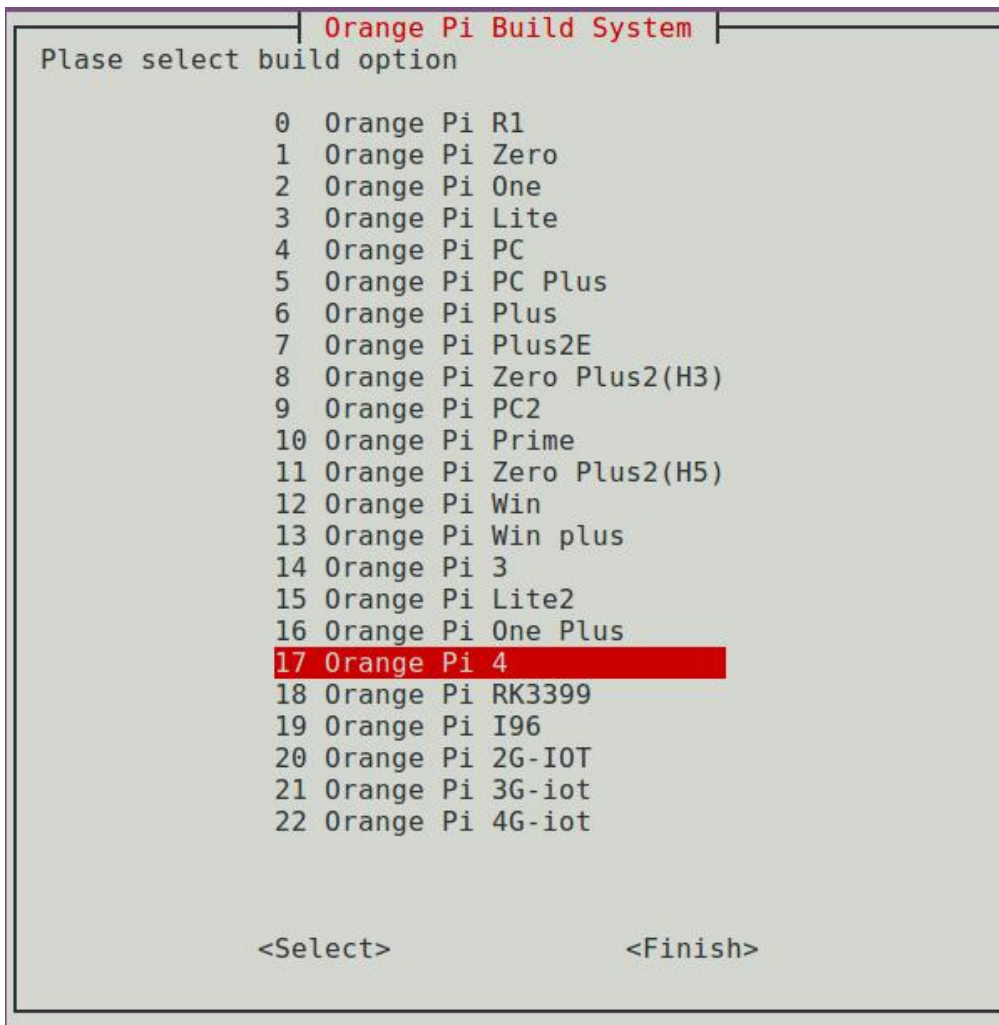




选择 0 Build system with kernel/u-boot/rootfs 进入开发板型号选择的界面



选择 17 OrangePi 4，回车后将会开始下载 Orange Pi 4 的 Linux SDK 源码





下载的源码会存放在 `OrangePi_Build` 的同级目录下

```
$ ls ../OrangePi_Build
OrangePi_Build OrangePiRK3399_Pi4
```

2. 搭建编译环境

OrangePi4 的 Linux 源码目录结构如下所示

```
$ cd OrangePiRK3399_Pi4
$ tree -L 1
.
├── build.sh -> scripts/build.sh    编译启动脚本
├── external                        存放额外的配置文件
├── kernel                         Linux 内核源码
├── output                        存放输出文件，编译源码后才会生成
├── scripts                       编译过程使用的脚本文件
├── toolchain                     内核和 u-boot 使用的交叉编译工具链
└── uboot                         u-boot 的源码

6 directories, 1 file
```

其中交叉编译工具链的目录结构如下所示，如果下载的文件目录和其有区别，或者 `toolchain` 目录下为空，说明下载过程有问题，请尝试重新使用 `OrangePi_Build` 下载器下载。

```
$ cd toolchain
$ tree -L 2
├── gcc-linaro-6.3.1-2017.05-x86_64_aarch64-linux-gnu
│   ├── aarch64-linux-gnu
│   ├── bin
│   ├── gcc-linaro-6.3.1-2017.05-linux-manifest.txt
│   ├── include
│   ├── lib
│   ├── libexec
│   └── share
└── README.md

7 directories, 2 files
```

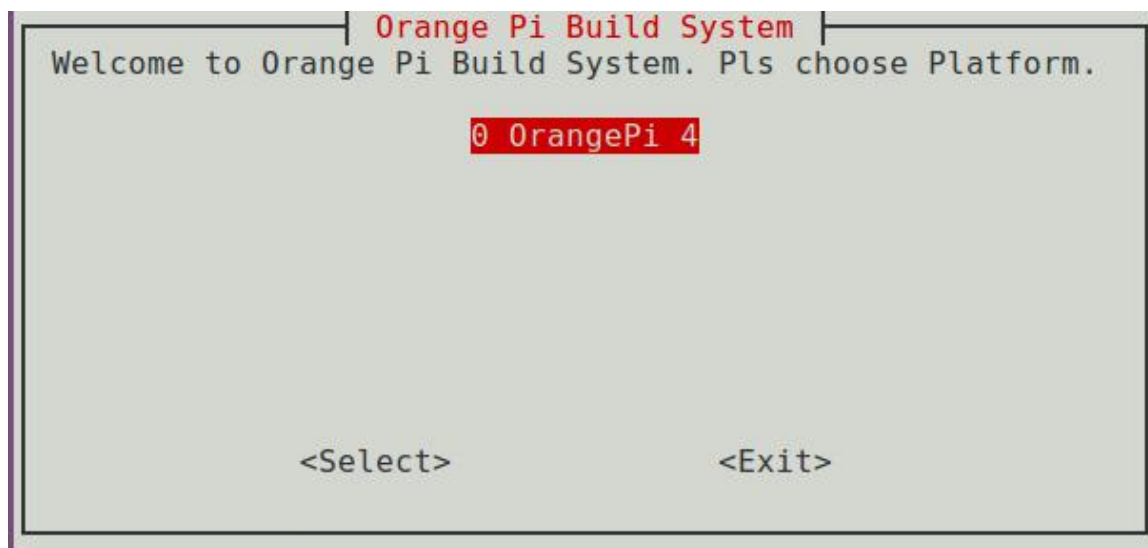


3. 编译 Linux 和 U-boot 源码

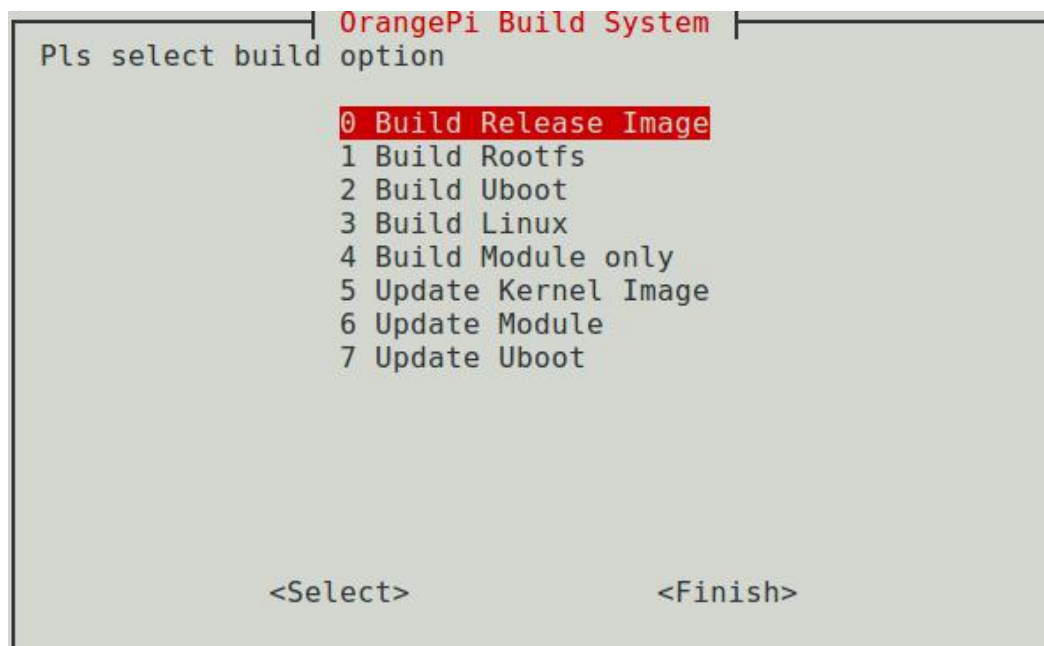
● 执行编译启动脚本

```
$ cd OrangePiRK3399_Pi4  
$ ./build.sh
```

选择 **0 OrangePi 4** 并回车



然后选择需要执行的功能。





其中各选项的功能如下:

- **0 Build Release Image** —— 编译 Ubuntu 或 Debian 的发行版镜像
- **1 Build Rootfs** -----编译Rootfs
- **2 Build Uboot** —— 编译 u-boot 的源码
- **3 Build Linux** —— 编译 Linux 内核源码
- **4 Build Module only** -----只编译内核模块
- **5 Update kernel Image** —— 更新 SD 卡 Linux 系统中的内核boot.img
- **6 Update Module** —— 更新 SD 卡 Linux 系统中的内核模块
- **7 Update Uboot** —— 更新 SD 卡 Linux 系统的 u-boot

编译完 u-boot 和内核源码后生成的最终文件会保存在 output 目录下

```
$ cd output
$ tree -L 2
.
├── kernel
│   └── boot.img
├── lib
│   └── modules
└── uboot
    ├── idbloader.img
    ├── rk3399_loader_v1.22.119.bin
    ├── trust.img
    └── uboot.img

4 directories, 5 files
```

编译发行版镜像后, 生成镜像保存在output/images目录下

```
$ tree -L 2 output/images/
output/images/
├── OrangePi_4_ubuntu_bionic_desktop_linux4.4.179_v1.0.img
├── OrangePi_4_ubuntu_bionic_desktop_linux4.4.179_v1.0.tar.gz
├── OrangePi_4_ubuntu_xenial_desktop_linux4.4.179_v1.0.img
└── OrangePi_4_ubuntu_xenial_desktop_linux4.4.179_v1.0.tar.gz

0 directories, 4 files
```



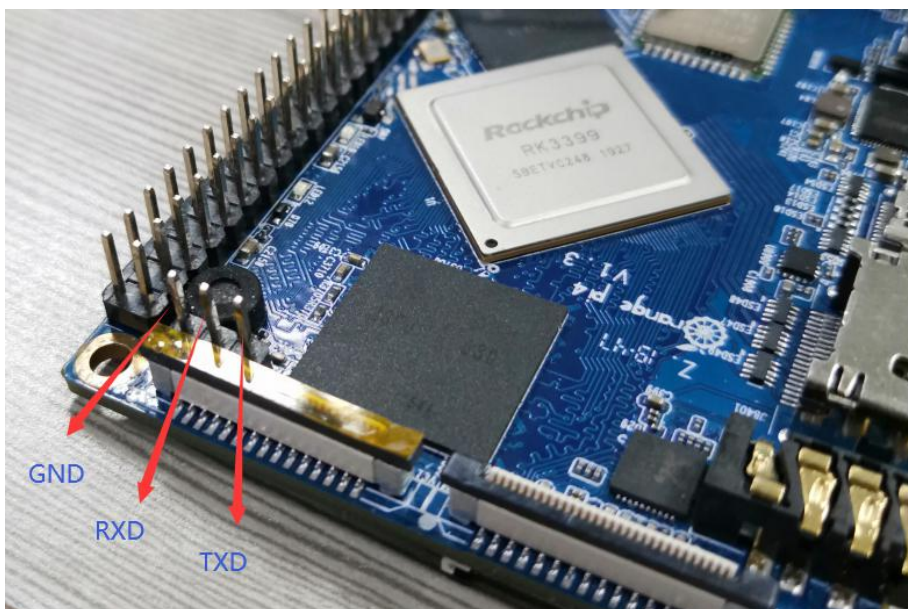

九、 串口调试工具介绍

首先需要准备一根和下图类似的 USB 转 TTL 串口线：



按下图接好串口线，不同颜色的线对应的功能如下：

- 黑色——GND
- 绿色——RX
- 白色——TX



注意：板子的 RXD 接 USB 转 TTL 串口的 TXD
板子的 TXD 接 USB 转 TTL 串口的 RXD



1. 基于 Windows 平台的使用

在使用 OrangePi 做项目开发过程中，为了获得更多的调试信息，OrangePi 默认支持串口信息调试。对于开发者而言，只需准备上面提到的材料，即可简单的获得串口调试信息。不同的上位机使用的串口调试工具大同小异，基本可以参考下文的方法进行部署。使用 Windows 平台进行串口调试的工具很多，通常使用的工具是 putty。本节以 putty 作为例子进行部署讲解。

● 安装 USB 驱动

下载最新版的驱动 PL2303_Prolific_DriverInstaller_v130.zip，下载解压。

PL2303_Prolific_DriverInstaller_v130	2010/7/15 10:41	应用程序	3,099 KB	← 解压之后的应用程序
PL2303_Prolific_DriverInstaller_v130	2016/8/3 9:20	WinRAR ZIP 压缩...	2,316 KB	← 下载的压缩包
releasenote	2010/7/22 10:14	文本文件	2 KB	

以管理员身份选择应用程序安装



等待安装完成



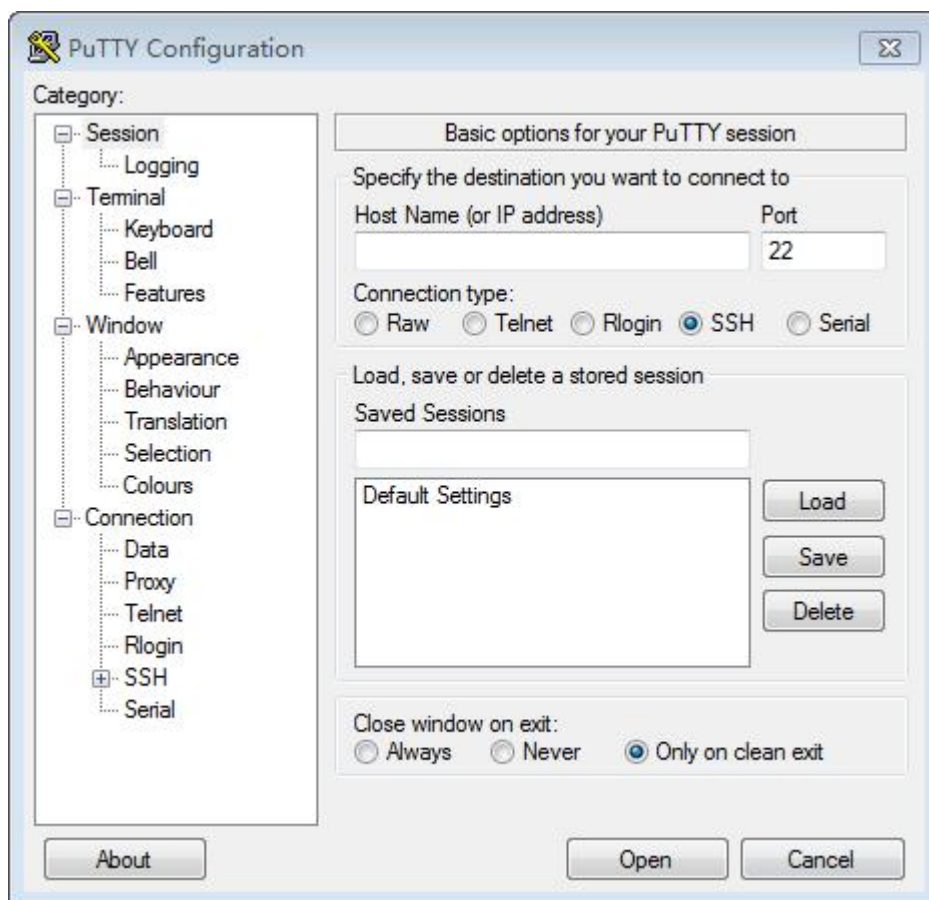


● 下载安装 Putty

Putty 可从下面的地址下载，请选择适合自己开发环境的版本。

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

直接双击下载的 putty.exe 即可打开 putty，软件界面如下图所示。





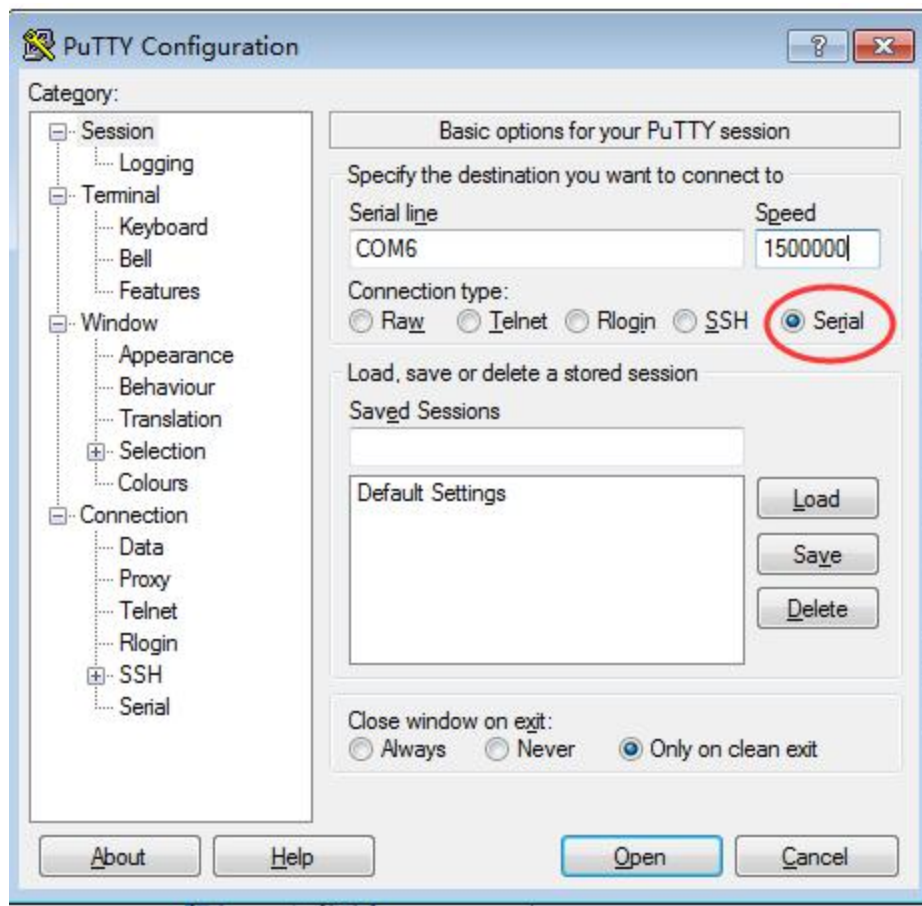
● 设备信息的获取

在 Windows7 中，我们可以通过设备管理器查看串口连接是否正常以及串口的设备号。如果设备没有正常识别，请检查驱动是否安装成功。如果驱动安装有问题，可以尝试使用 360 驱动大师扫描安装驱动。



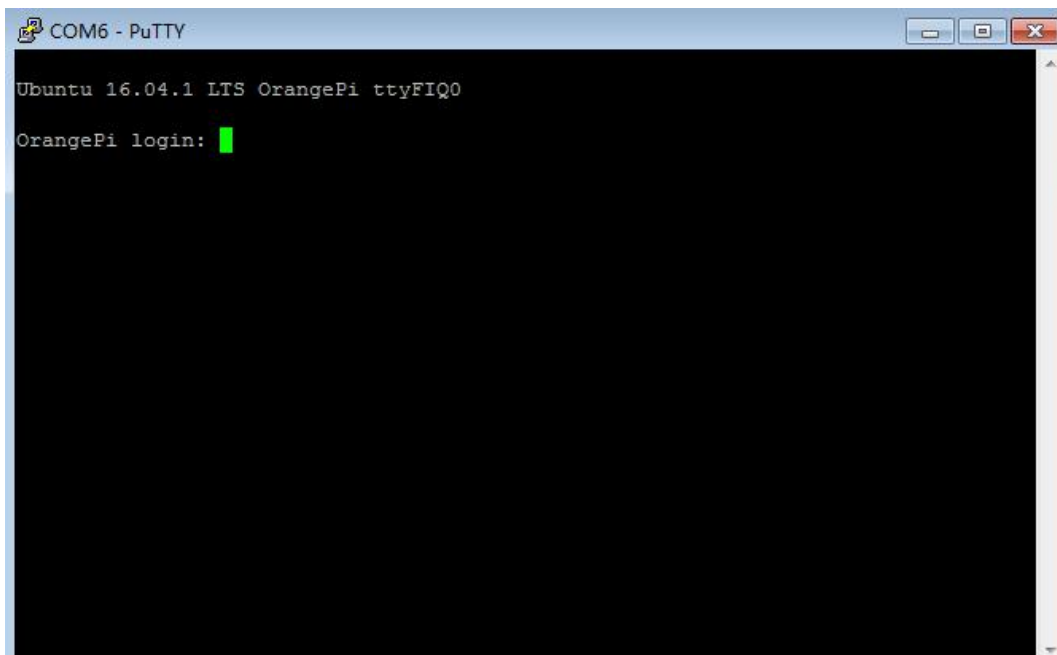
● Putty 配置

串行口设置成相应的端口号 (COM6)，关闭流控，速度设置成 **1500000**



● 启动调试串口输出

OrangePi 上电开机，putty 将会自动打印串口 log 信息



2. 基于 Linux 平台的使用

在 Linux 平台使用 putty 和 Windows 平台区别不太,下面主要说明有差异地方的操作步骤。所有操作都是基于 Ubuntu 14.04 系统。

● 安装并启动 Putty

```
$ sudo apt-get install putty  
$ sudo putty
```

● 配置 Putty

串口号可以通过 `ls /dev/ttyUSB*` 查看
波特率需要设置为 1500000
并且关闭流控

