

MarsBoard RK3066 Pro 用户手册

版权声明

本手册所有权由深圳市微雪电子有限公司独家持有。未经本公司的书面许可，不得以任何方式或形式进行修改、分发或复制本文档的任何部分，否则一切后果由违者自负。

产品概述

MarsBoard RK3066 Pro 是基于瑞芯微 RockchipRK3066 处理器的卡片式电脑。Rockchip RK3066 采用 40nm 工艺打造，CPU 采用双核 ARM Cortex-A9 加四核 GPU 架构，主频高达 1.6GHz，支持 OpenGL ES 1.1/2.0 和 OpenVG 1.1。

板载资源

[核心板器件]

1. Rockchip RK3066
2. DDR3 1GB 内存
3. 4GB eMMC Flash 存储器
4. TPS659102 电源管理芯片
5. TX 调试 LED
6. 电源 LED
7. LAN8720A 10M/100M 以太网 PHY
8. J3 扩展接口 (20x2 pin 1.27mm)
9. J2 扩展接口 (50x2 pin 1.27mm)
10. J1 扩展接口 (50x2 pin 1.27mm)

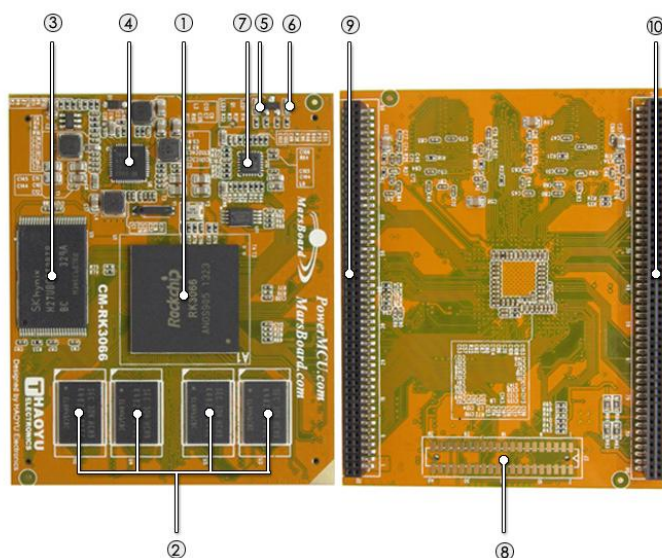


图 1. 核心板资源

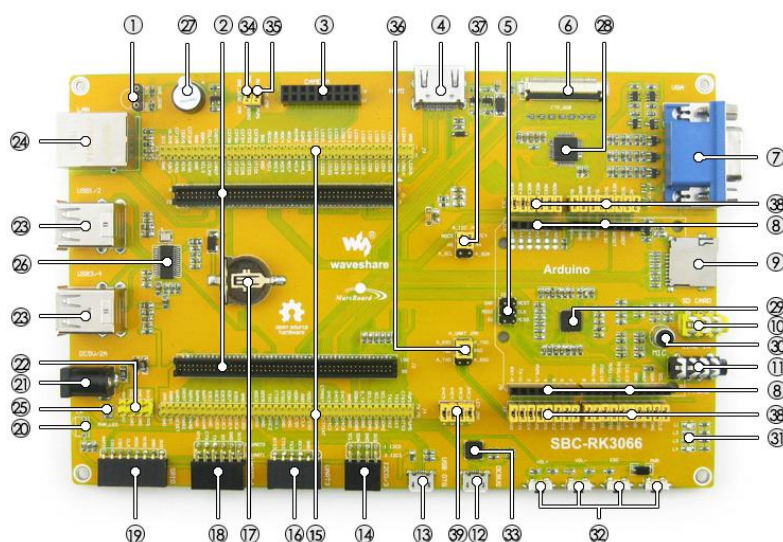


图 2. 底板资源

[接口简介]

1. 1-WIRE 接口
2. CM-RK3066 核心板接口
3. CSI 摄像头接口
4. HDMI 接口
5. ICSP 接口
6. LCD 接口
7. VGA 接口
8. Arduino 接口
9. Micro SD 卡接口
10. 3.5mm LINE IN 接口
11. 3.5mm 耳机接口
12. DEBUG 接口
13. OTG 接口
14. I2C 接口
15. 核心板扩展接口
16. UART 接口
17. 电池座
18. 带流控制的 UART 接口
19. SPI 接口
20. IR 接口
21. 电源接口
22. 扩展电源接口
23. USB 接口
24. LAN 接口

[器件简介]

25. 电源指示灯
26. FE1.1S
27. 蜂鸣器
28. GM7123C
29. ALC5631
30. 咪头
31. 用户 LED
32. AD 按键
33. CP2102

[跳线说明]

34. ONE-WIRE 接口跳线
35. 蜂鸣器跳线
36. Arduino UART 选择跳线
37. Arduino AD 选择跳线
38. Arduino Port 选择跳线
39. 用户 LED 跳线

目录

MarsBoard RK3066 Pro 用户手册	1
版权声明.....	1
产品概述.....	1
板载资源.....	1
第一章 Android 开发环境搭建与调试.....	5
1 Android 环境搭建.....	5
1.1 获取 Android 固件资源	5
1.2 烧录固件文件	5
2 Android 功能调试.....	9
2.1 LCD 与触摸屏测试	9
2.2 音频测试.....	10
2.3 MIC 测试.....	11
2.4 LINE IN 测试	12
2.5 视频测试.....	12
2.6 USB 摄像头测试.....	12
2.7 有线网络测试	13
2.8 USBWIFI 测试.....	14
2.9 SD 卡测试	14
2.10 USB HUB 测试	15
3 Android 源码编译.....	16
3.1 获取 Android 源码资源	16
3.2 Android 源码编译步骤	16
3.3 Android 源码下载步骤	16
4 Android 升级固件相关操作	19
4.1 升级固件的制作与烧录	19
4.2 升级固件的解压	20
第二章 Linux 开发环境搭建与调试.....	22
1 Linux 环境搭建.....	22
1.1 获取 Linux 固件资源	22
1.2 烧录镜像文件	22
2 Linux 功能调试.....	25
2.1 电脑端通过 SSH 方式对 SBC-RK3066 进行操作	25
2.2 LED 测试.....	28
2.3 PWM 测试	28
2.4 DS18B20 测试	28
2.5 RTC 测试	29
2.6 GPS 测试.....	29
2.7 MAG3110 测试.....	30
2.8 AT45DB E2PROM 测试	30
2.9 RS485 接收发送测试.....	31
2.10 USB CAMERA 测试.....	32
2.11 USB Wi-Fi 测试	34

2.12	USB HUB 测试	36
2.13	SD 卡测试	36
3	Linux 内核编译	37
3.1	获取 Linux 源码资源	37
3.2	Linux 内核编译	37
3.3	升级固件的制作与烧录	38

第一章 Android 开发环境搭建与调试

1 Android 环境搭建

SBC-RK3066 提供的提供 Android4.2.2 Android_4.4.2，两种版本的 Android 镜像与源码。本手册将简单讲述一下镜像下载和源码编译。出厂默认是 Android_4.4.2，显示分辨率为 1024*600。用户可以自行烧录其他版本的镜像。

注意：以下提到的电源为 5V/2A 的电源适配器。

1.1 获取 Android 固件资源

[MarsBoard_RK3066_PX2_Android_4.4.2_HDMI_720P_Nand_V1.0.img](#)

固件说明：产品出厂已经预装了搭配 7inch LCD HY070CTP-HD 屏幕的系统。如果用户需要外接 HDMI 显示器使用的话，需要重新烧写这个镜像。

[MarsBoard_RK3066_PX2_Android_4.4.2_LCD_HY070CTP-HD_1024x600_Nand_V1.0.img](#)

固件说明：出厂自带的系统镜像。适用于 SBC-RK3066 接入了 LCD(型号是 HY070CTP-HD, 分辨率为 1024x600) 的情况。

[RKBatchTool-MarsBoard_RK3066_Box_Android_4.2.2_HDMI_1080P_V1.0.img](#)

固件说明：用户如果需要用到较早的 Android 4.2.2 版本，并且是外接 HDMI 显示器的话，选择烧写这个镜像。

如需深度定制 Android 系统，请参考 3. Android 源码编译

1.2 烧录固件文件

产品出厂已经预装了搭配 7inch LCD HY070CTP-HD 屏幕的系统。如需外接 HDMI 显示器，或者需要重装系统的话，请参阅以下步骤。如果无需更换系统，可以直接阅读 2. Android 功能调试一节。

- 1) 连接 MicroUSB 线，并按下下图所示的按键 VOL+，直到电脑端发现新硬件。

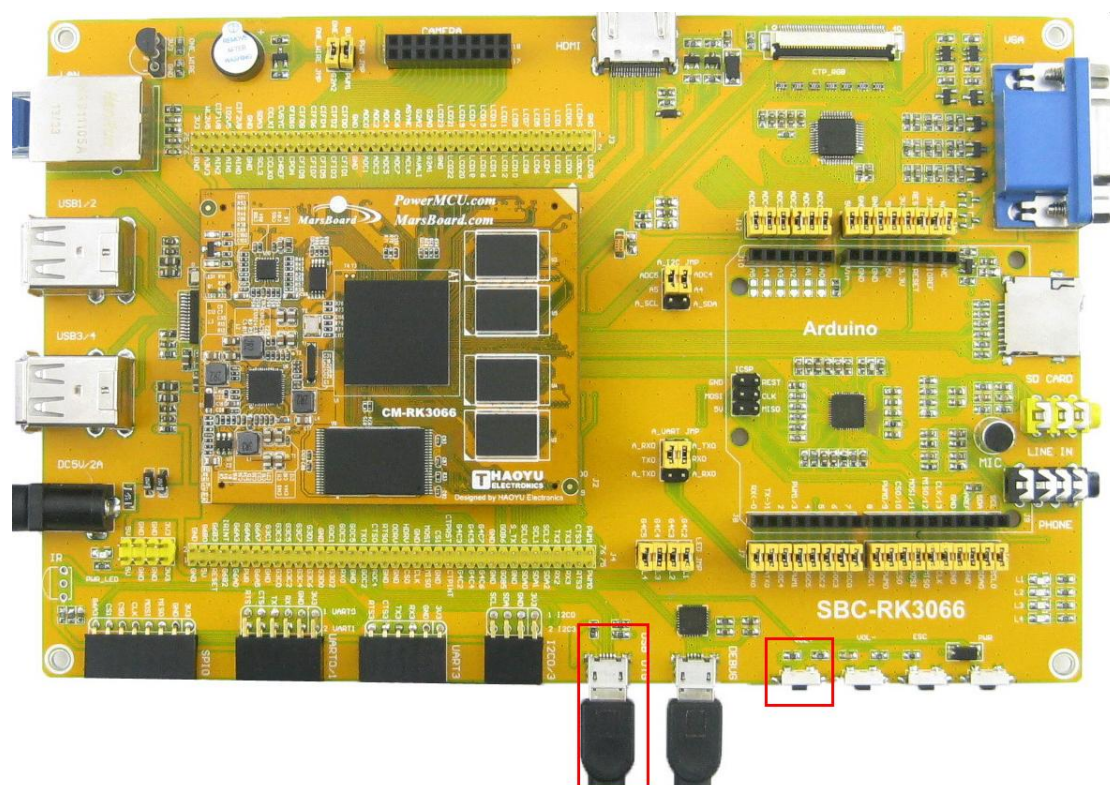


图 3. 通过 MicroUSB 线接入电脑

- 1) 安装瑞芯微驱动助手（[Release DriverAssitant\DriverInstall.exe](#)）。
- 2) 安装成功后在电脑的硬件管理界面可以看到如下的信息。

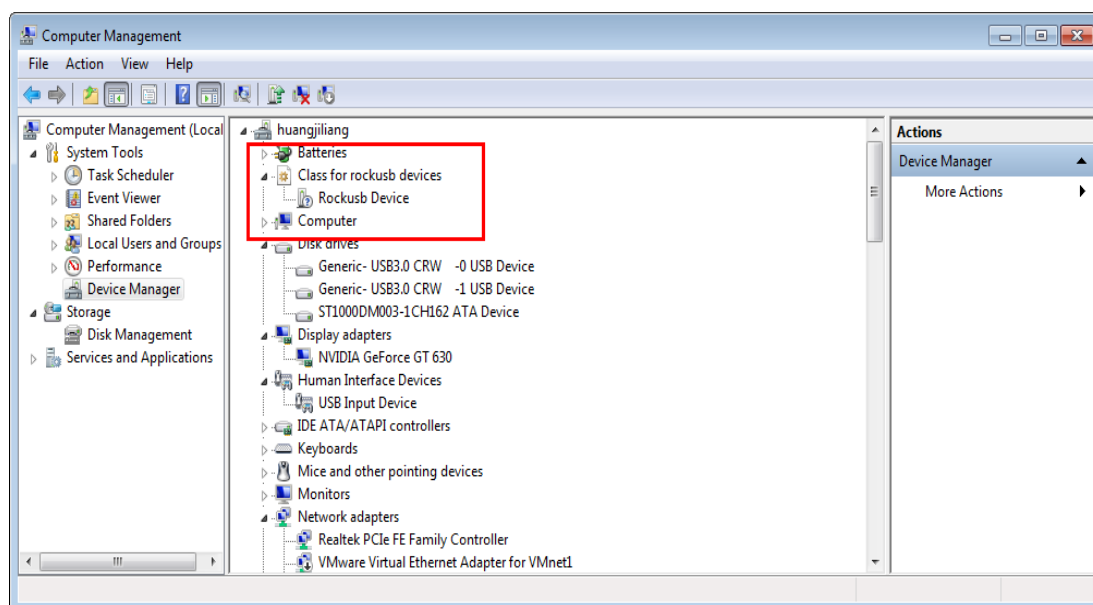


图 4. 硬件管理界查看

- 3) 将 AndroidTool_Release_v2.1 工具解压到 windows 下的 D 盘。
- 4) 打开 [AndroidTool_Release_v2.1\AndroidTool_Release\AndroidTool.exe](#) 界面如下图所示：

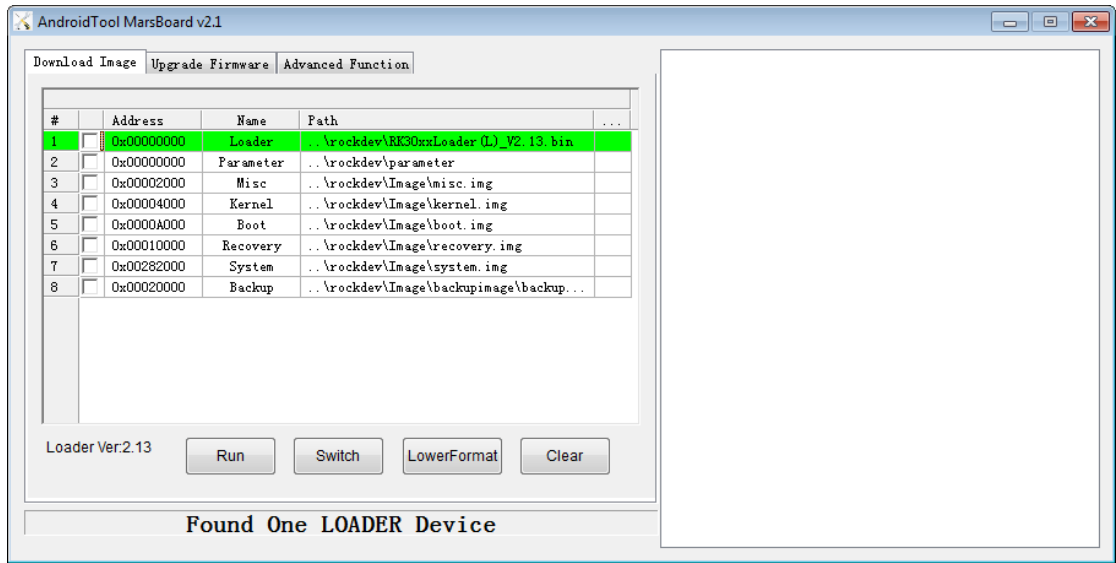


图 5. AndroidTool 界面

- 5) 解压下载的压缩包
MarsBoard_RK3066_PX2_Android_4.4.2_HDMI_720P_Nand_V1.0.img.7z
- 6) 打开 AndroidTool.exe, 选择 **Upgrade Firmware** 选项卡下的 **Firmware**, 选择需要下载的固件。点击 **Upgrade** 开始更新固件。

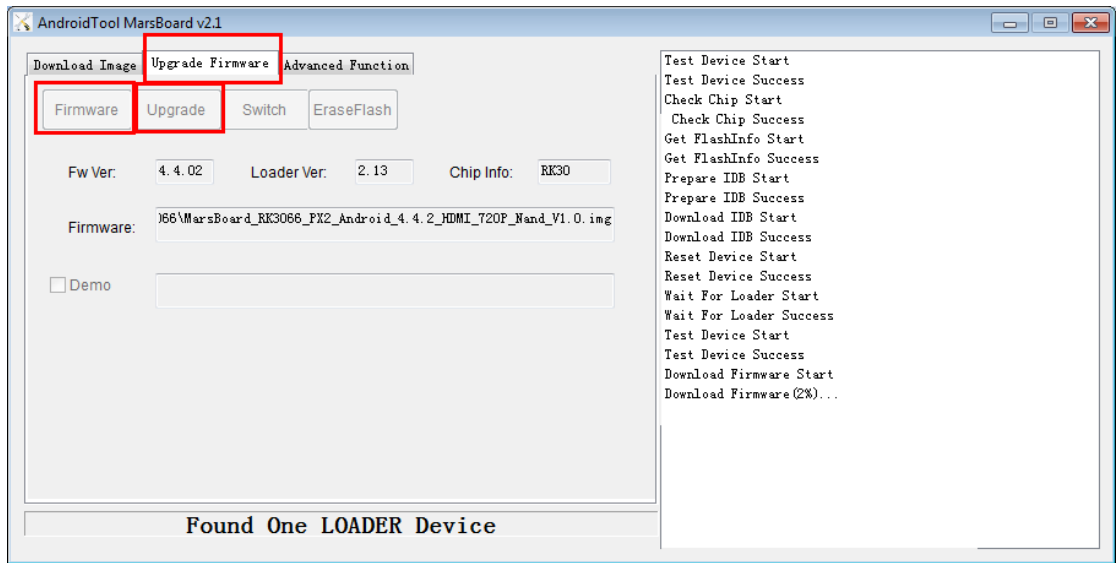


图 6. 升级固件

注意：加载固件需要的时间较长，请耐心等待。

- 7) 显示下图红框中的信息时说明下载完成

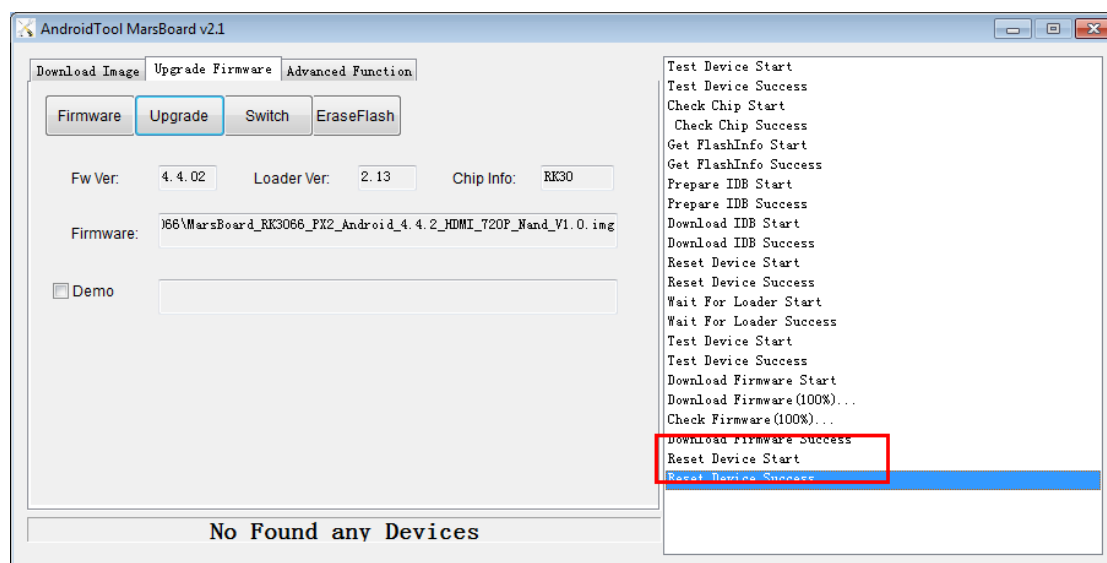


图 7. 下载完成

2 Android 功能调试

2.1 LCD 与触摸屏测试

- 1) 断开电源，使用 LCD 的排线连接底板的 CTP_RGB 接口和 LCD 屏的 RGB 接口。

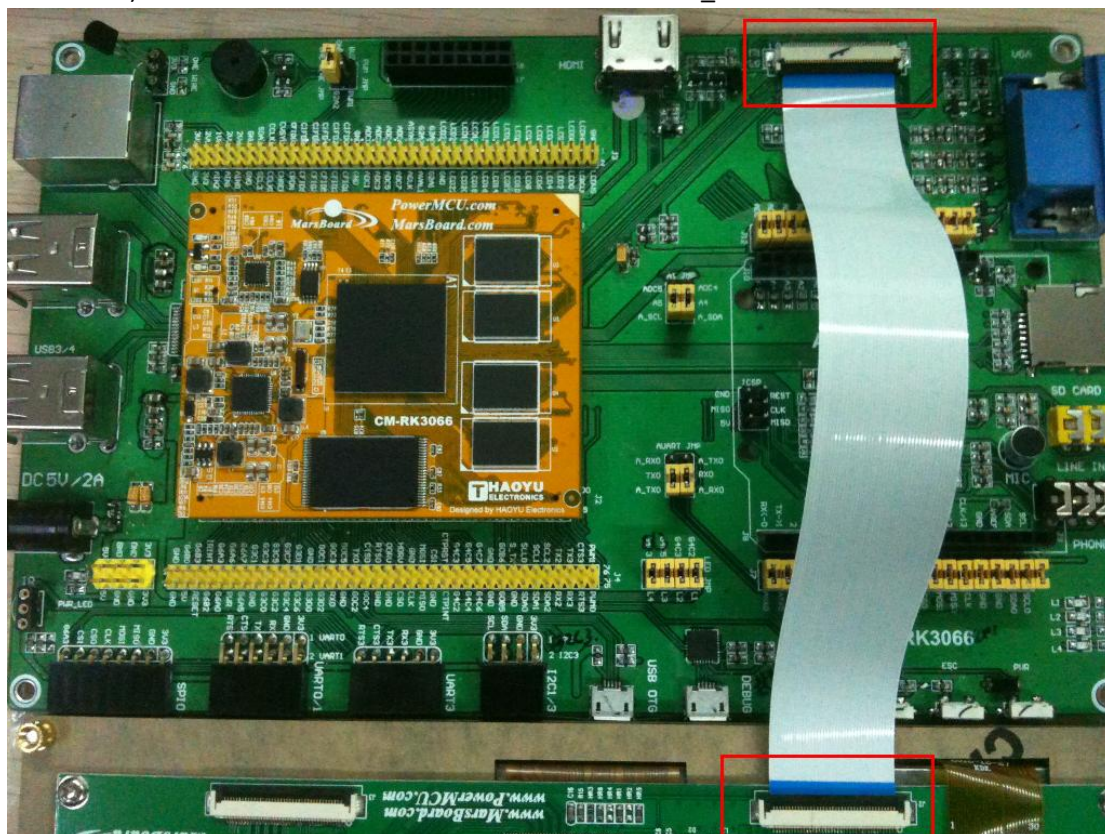


图 8. LCD 硬件连接

- 2) 接通电源，Android 正常启动，正常显示 Android 系统的界面。
- 3) 随意点击自带 APP，可以启动 APP。

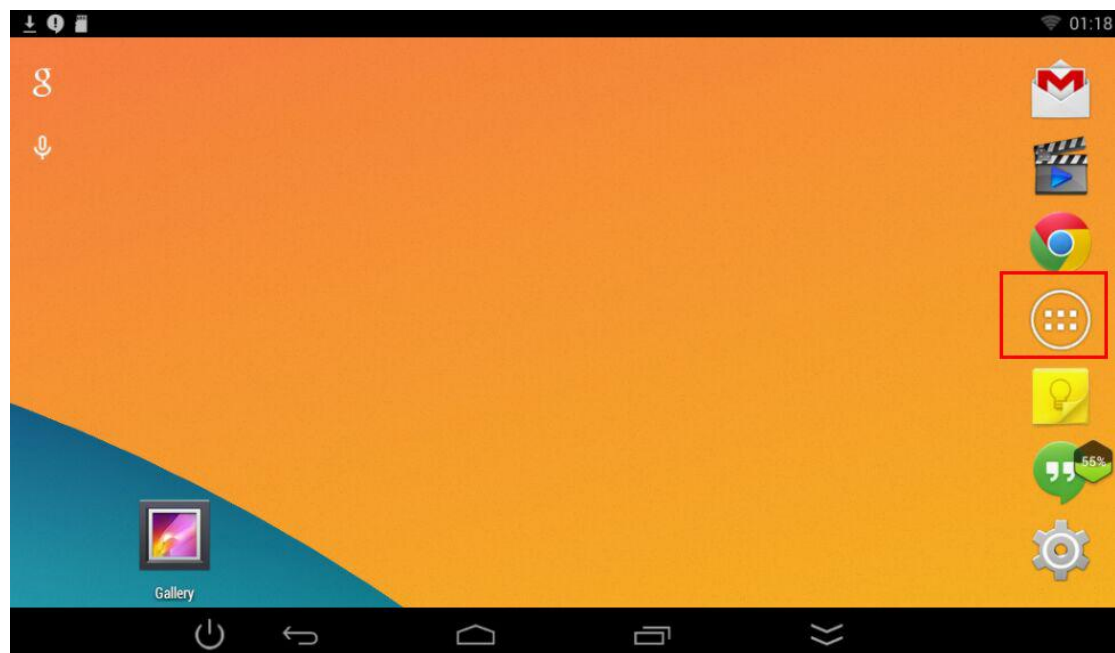


图 9. 显示界面

2.2 音频测试

- 1) 下载谷歌播放器或者其他播放器。
- 2) 下载音乐到开发板中。

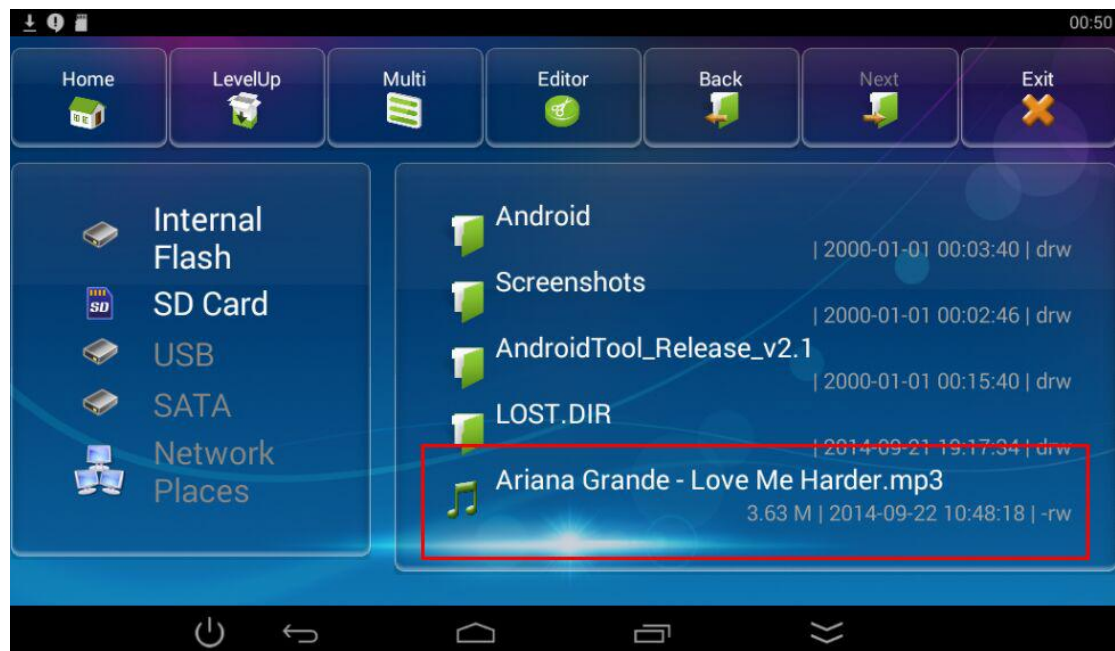


图 10. 下载音乐到开发板中

- 3) 找到该音乐文件，点击使用谷歌播放器打开。

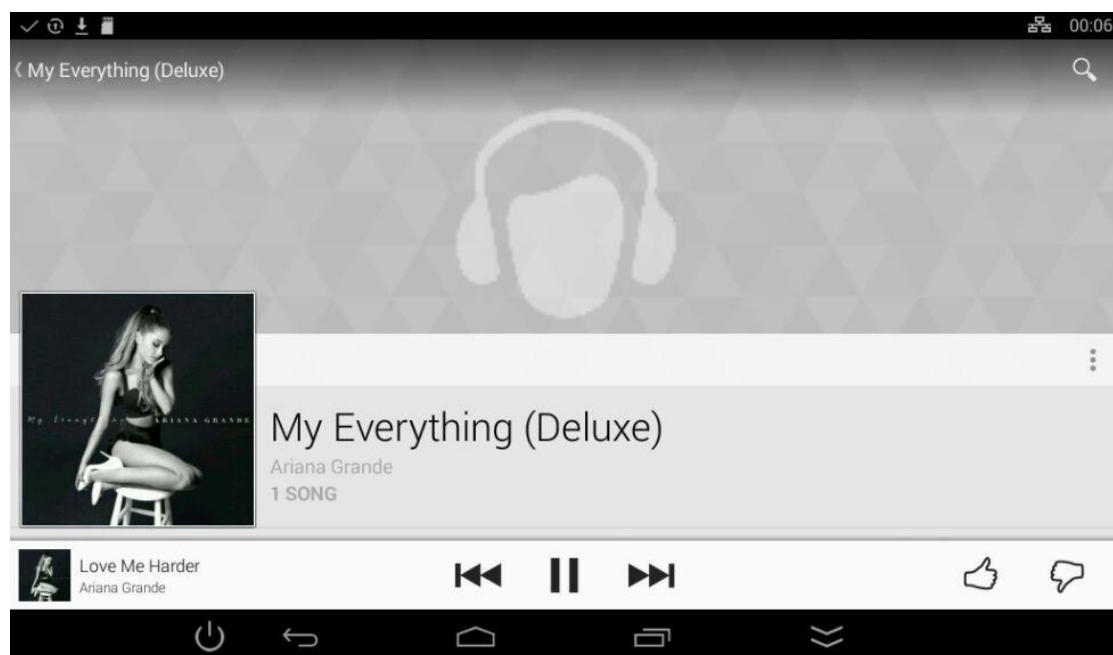


图 11. 音频测试

- 4) 接入耳机到 PHONE 接口即可听到音乐。

2.3 MIC 测试

- 1) 打开 Android 中的 SoundRecorder APP。
- 2) 点击 APP 开始录音，对着咪头讲话可以看到指针在左右的摆动。

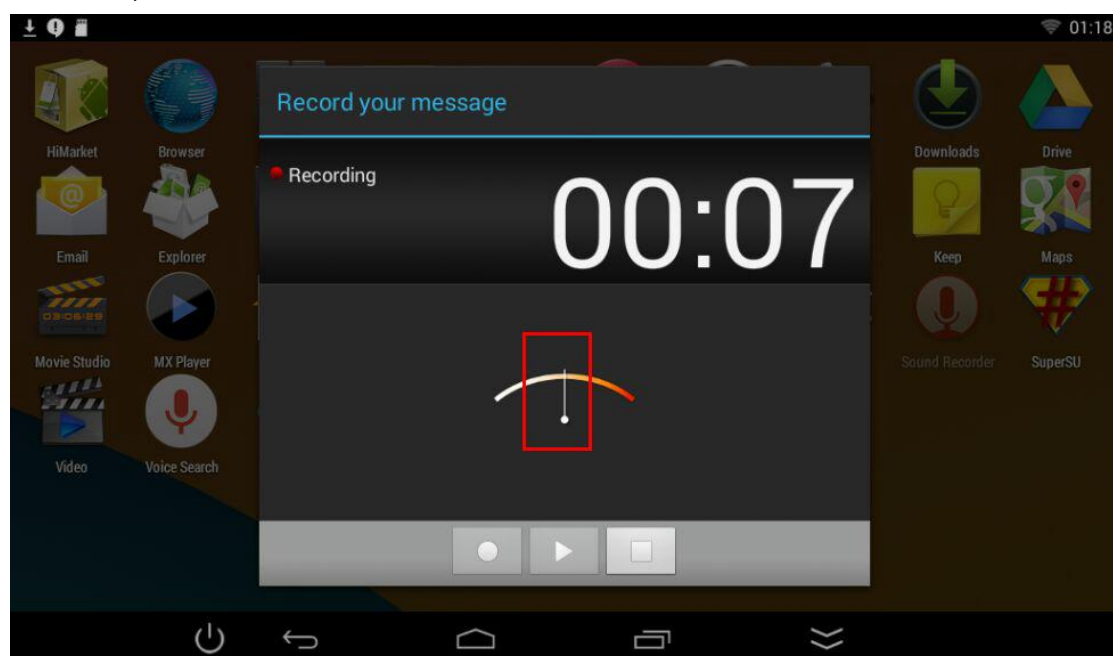


图 12. MIC 测试

2.4 LINE IN 测试

- 1) 使用音频线连接带有 LINE OUT 功能设备的（如 CD 机）LINE OUT 接口与开发板的 LINE IN 接口。
- 2) 打开 Android 中的 SoundRecorder APP。
- 3) 点击 APP 开始录音。使用带有 LINE OUT 功能的设备输出音频信号，可以看到指针在左右的摆动。参见图 12. MIC 测试

2.5 视频测试

- 1) 下载视频播放器，此处以 MX Player 为例。

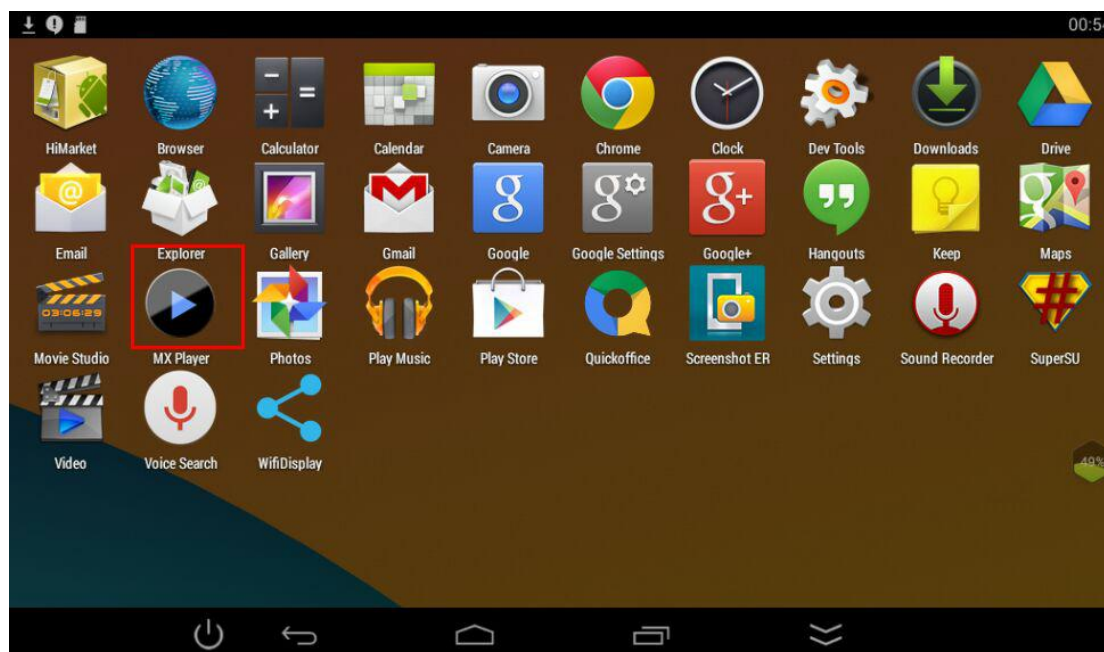


图 13. 使用 MX Player 播放视频文件

- 2) 下载视频文件（该视频格式应该是 MX Player 支持的格式）。
- 3) 找到该视频文件，点击该视频文件，使用 MX Player 打开。

2.6 USB 摄像头测试

- 1) 接入 USB 摄像头。
- 2) 打开 Android 自带的摄像程序（Camera）。
- 3) 可以看到摄像头采集到的图像显示出来。



图 14. 摄像头测试

2.7 有线网络测试

- 1) 接入网线到开发板的 LAN 接口。
- 2) 打开 Android 的设置（Settings）。
- 3) 将 Ethernet 设置为 ON。
- 4) 打开 Android 的自带浏览器（Browser）。
- 5) 正常情况下可以浏览网页。

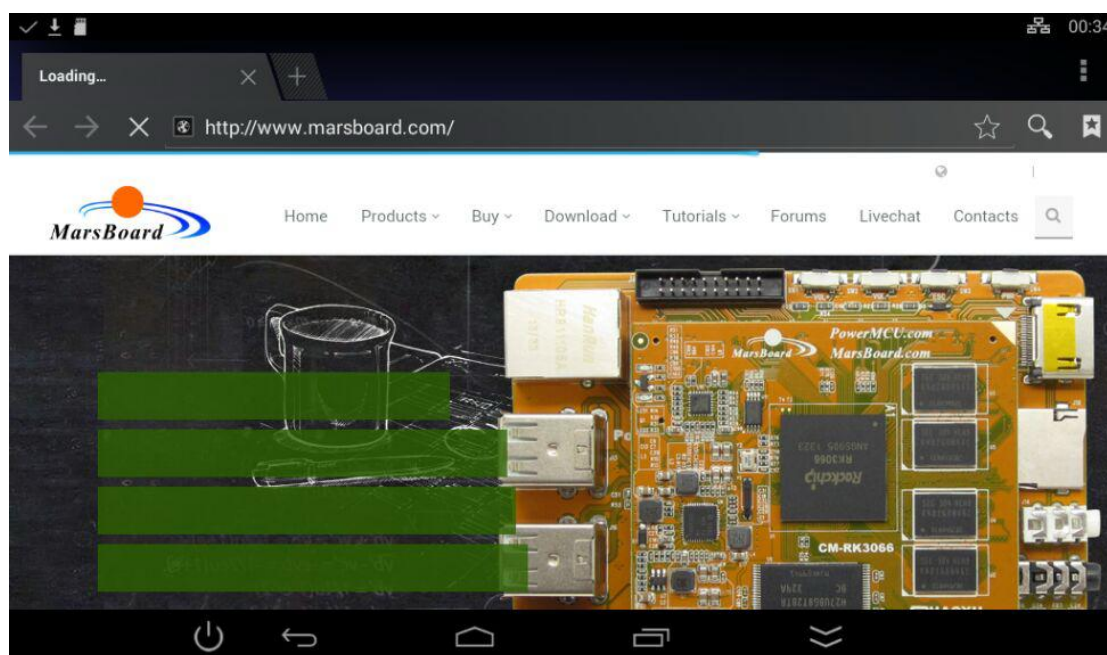


图 15. 用 Android 自带浏览器浏览网页

2.8 USBWIFI 测试

- 1) 接入 USBWIFI 到开发板的 USB 接口。
- 2) 打开 Android 的设置 (Settings)。
- 3) 将 Wi-Fi 设置为 ON。
- 4) 找到合适的 WIFI 热点。

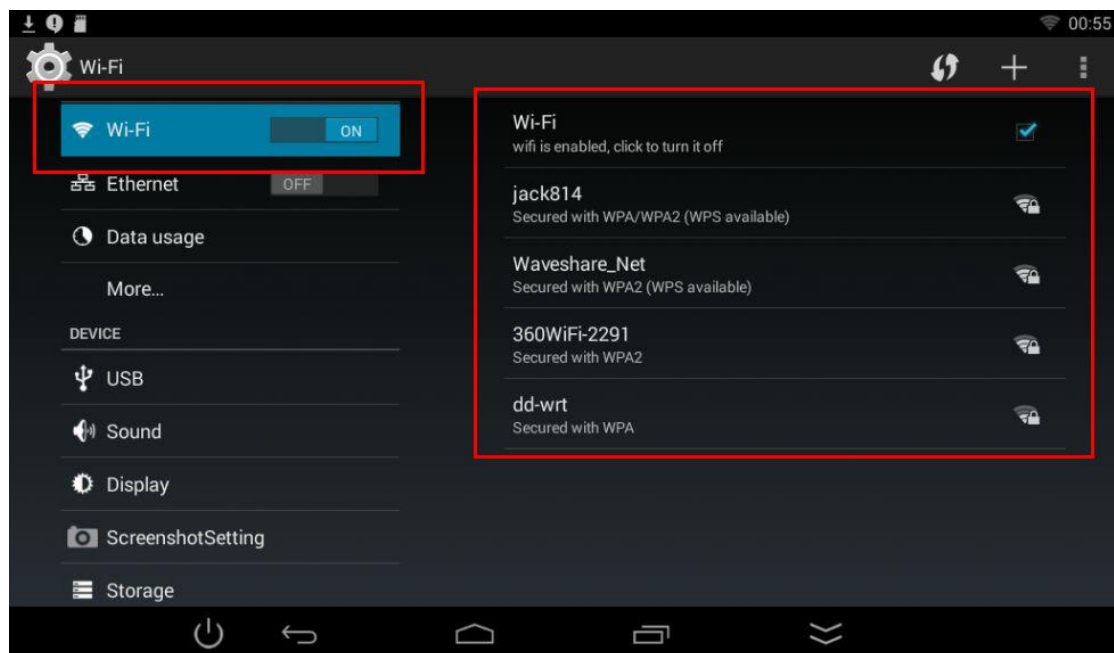


图 16. 连接 Wi-Fi 热点

- 5) 打开 Android 的自带浏览器 (Browser)
- 6) 正常情况下可以浏览网页, 参见图 15. 用 Android 自带浏览器浏览网页。

2.9 SD 卡测试

- 1) 接入 Micro TF 卡到 SD CARD 接口。
- 2) 打开 Android 中的文件管理器 (Explorer)。
- 3) 点击 SD Card, 即可看到 SD 卡的内容。

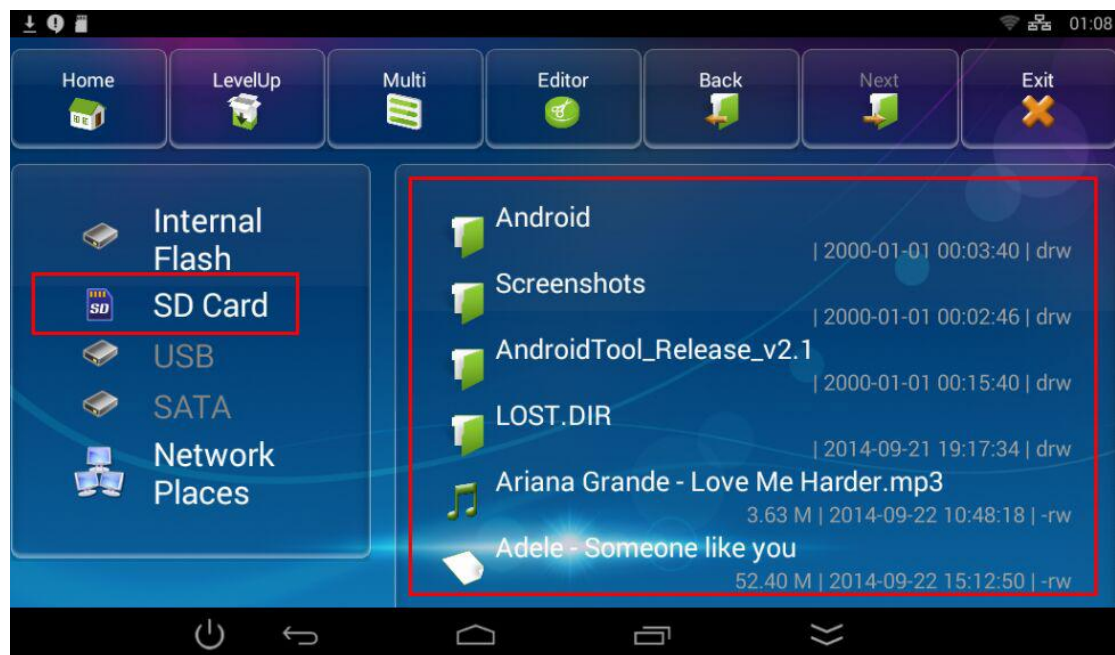


图 17. 用文件管理器查看 SD 卡内容

2.10 USB HUB 测试

接入鼠标，键盘，USBWIFI，USB 摄像头等设备，正常情况下可以识别到。

注意：建议不要接移动硬盘，以免应供电不足造成移动硬盘的损坏。

3 Android 源码编译

源码编译适用于需要深度定制 Android 系统的用户，轻度用户可以跳过阅读本节。

3.1 获取 Android 源码资源

[MarsBoard-RK3066-PX2-Android-4.4.2-SDK.tar.gz](#)

[MarsBoard_RK3066_Box_Android_4.2.2_SDK_V1.0.tar.gz](#)

[MarsBoard_RK3066_Tablet_Android_4.1.1_SDK_V1.0.tar.gz](#)

注意：

- 三个版本的编译的步骤是一样的。只是加载内核的配置文件不同。
- 编译的环境是在桌面版的 **ubuntu12.04**。
- 在编译的时候确保有 **root** 权限。

3.2 Android 源码编译步骤

- 1) 在 PC 端的 ubuntu 系统中新建文件夹

```
#mkdir waveshareRK3066
```

- 2) 复制源码到 waveshareRK3066 目录中，解压源码

```
# tar xvf MarsBoard-RK3066-PX2-Android-4.4.2-SDK.tar.gz
```

- 3) 编译 Android 源码下的 linux 内核

```
#cd /MarsBoard-RK3066-PX2-Android-4.4.2-SDK/ kernel
```

```
#make marsboard_rk3066_px2_hy070ctp_hd_defconfig
```

```
#make kernel.img
```

- 4) 编译 Android 源码

```
#cd ..
```

```
#source build/envsetup.sh
```

```
# lunch marsboard-eng
```

```
#make -j4
```

- 5) 生成镜像文件

```
#./mkimage.sh
```

```
#cp kernel/kernel.img rockdev/Image
```

```
#cd rockdev/Image
```

```
#ls
```

- 完成上述步骤之后，即可看到编译后生成的可下载的 Android 镜像文件。

3.3 Android 源码下载步骤

- 1) 将 AndroidTool_Release_v2.1 工具解压到 windows 下的 D 盘。

- 2) 打开 [AndroidTool Release v2.1\AndroidTool Release\AndroidTool.exe](#) 界面如下图所示:

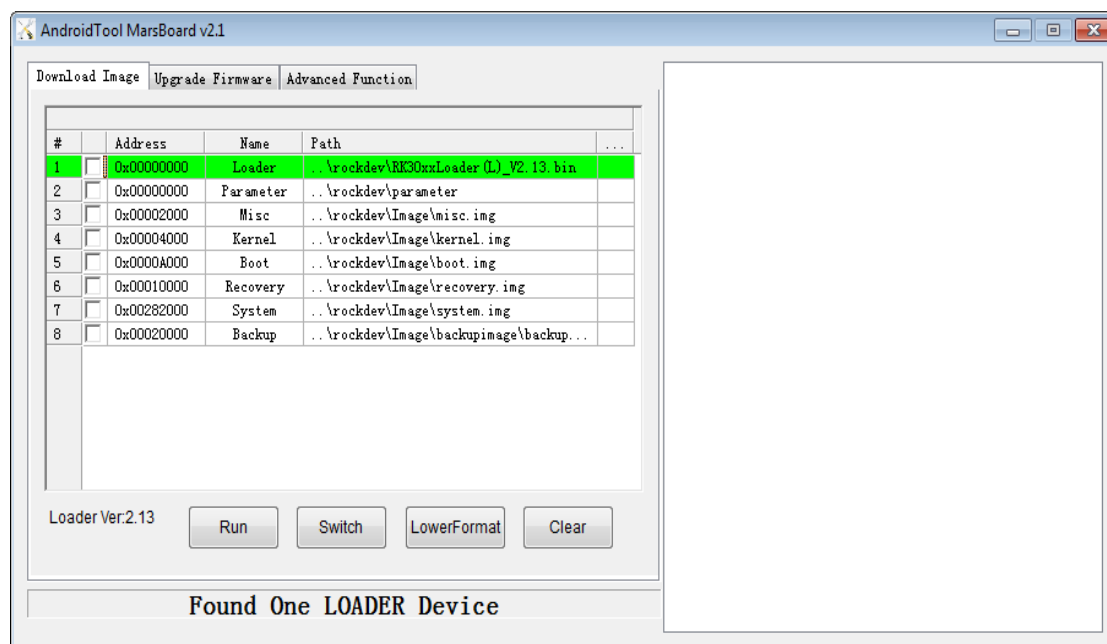


图 18. AdroidTool 界面

- 3) 将内核源码中的镜像文件 (rockdev\Image) 拷贝到 [AndroidTool Release v2.1\rockdev\Image](#)
- 4) 按照下图所示选择要下载的文件, 点击 **Run** 开始下载。

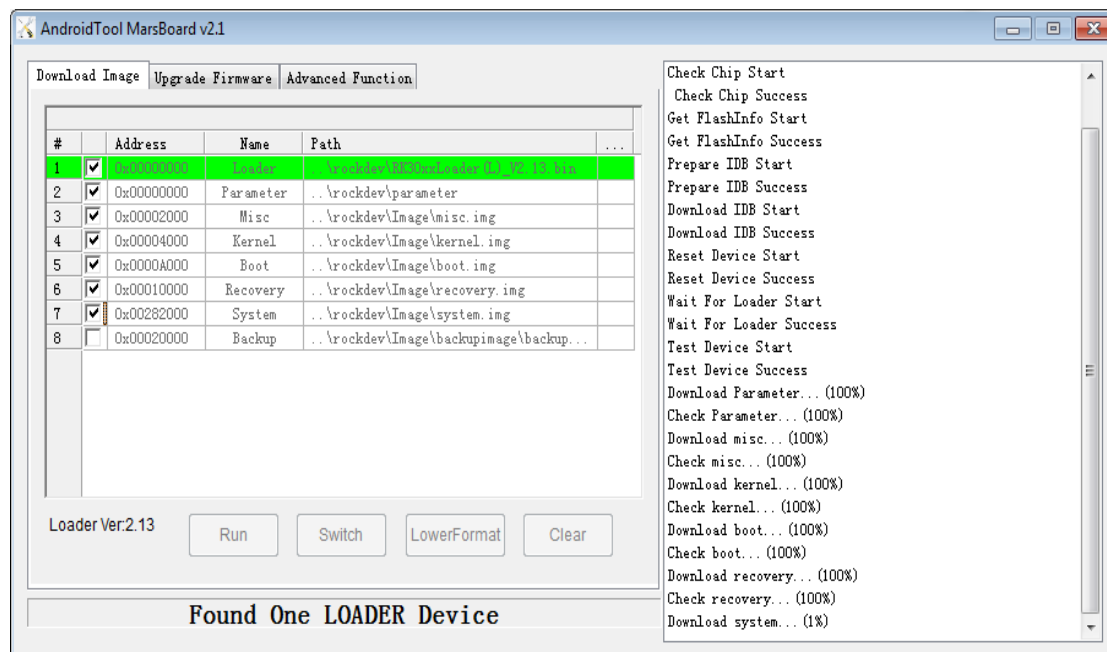


图 19. 点击 Run 开始下载

- 5) 显示下图红框中的信息时说明下载完成

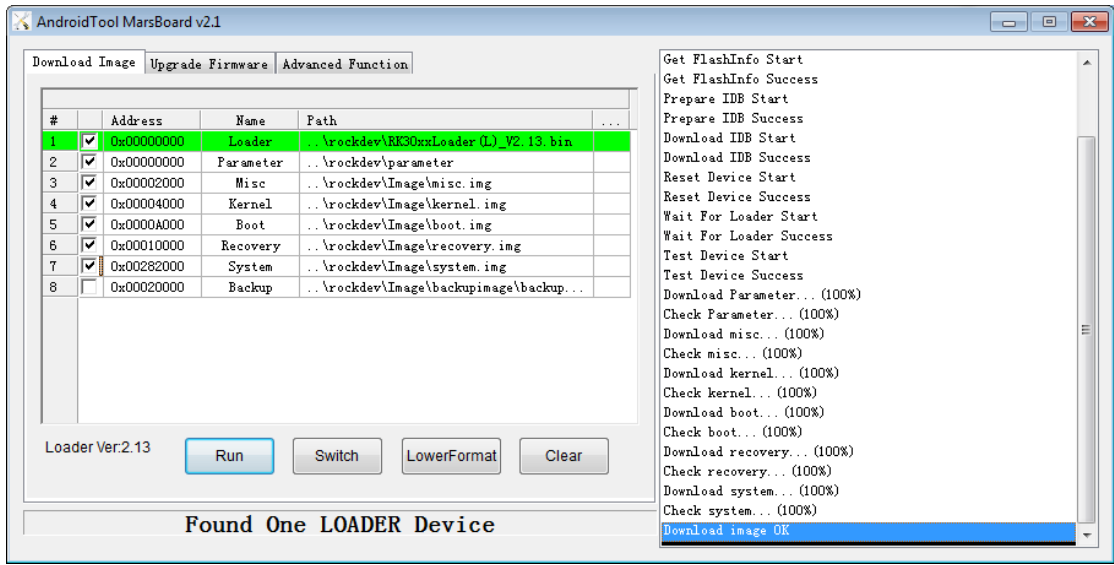


图 20. 下载完成

4 Android 升级固件相关操作

AndroidTool 软件提供了固件升级的功能，用户可以把多个文件打包到同一个固件中，便于量产。

4.1 升级固件的制作与烧录

- 1) 在 windows 下使用 winRAR 解压下载工具 AndroidTool_Release_v2.1.7z
- 2) 运行 CMD，在命令提示符下打开 mkupdate.bat 批处理脚本。

本例是把文件夹放在 D: 盘，所以执行如下命令：

```
D:
cd \AndroidTool_Release_v2.1\rockdev
mkupdate.bat
```

执行效果如下：

```
Administrator: C:\Windows\system32\cmd.exe
D:\AndroidTool_Release_v2.1\rockdev>
D:\AndroidTool_Release_v2.1\rockdev>mkupdate.bat

D:\AndroidTool_Release_v2.1\rockdev>Afptool -pack .\backupimage backupimage\backup.img
Android Firmware Package Tool v1.62
----- PACKAGE -----
Add file: .\backupimage\package-file
Add file: .\backupimage\package-file done,offset=0x800,size=0x281,userspace=0x1
Add file: .\backupimage\...\RK30xxLoader(L)_U2.13.bin
Add file: .\backupimage\...\RK30xxLoader(L)_U2.13.bin done,offset=0x1000,size=0x3014e,userspace=0x61
Add file: .\backupimage\...\parameter
Add file: .\backupimage\...\parameter.tmp done,offset=0x31800,size=0x286,userspace=0x1
Add file: .\backupimage\...\Image/misc.img
Add file: .\backupimage\...\Image/misc.img done,offset=0x32000,size=0xc000,userspace=0x19
Add file: .\backupimage\...\Image/kernel.img
Add file: .\backupimage\...\Image/kernel.img done,offset=0x3e800,size=0x766030,userspace=0xecd
Add file: .\backupimage\...\Image/boot.img
Add file: .\backupimage\...\Image/boot.img done,offset=0x7a5000,size=0x21771d,userspace=0x42f
Add file: .\backupimage\...\Image/recovery.img
Add file: .\backupimage\...\Image/recovery.img done,offset=0x9bc800,size=0xb58000,userspace=0x16b1
Add CRC...
Make firmware OK?
----- OK -----

D:\AndroidTool_Release_v2.1\rockdev>Afptool -pack ./ Image\update.img
Android Firmware Package Tool v1.62
----- PACKAGE -----
Add file: .\package-file
Add file: .\package-file done,offset=0x800,size=0x243,userspace=0x1
Add file: .\RK30xxLoader(L)_U2.13.bin
Add file: .\RK30xxLoader(L)_U2.13.bin done,offset=0x1000,size=0x3014e,userspace=0x61
Add file: .\parameter
Add file: .\parameter.tmp done,offset=0x31800,size=0x286,userspace=0x1
Add file: .\Image/misc.img
Add file: .\Image/misc.img done,offset=0x32000,size=0xc000,userspace=0x19
Add file: .\Image/kernel.img
Add file: .\Image/kernel.img done,offset=0x3e800,size=0x766030,userspace=0xecd
Add file: .\Image/boot.img
Add file: .\Image/boot.img done,offset=0x7a5000,size=0x21771d,userspace=0x42f
Add file: .\Image/recovery.img
Add file: .\Image/recovery.img done,offset=0x9bc800,size=0xb58000,userspace=0x16b1
Add file: .\Image/system.img
Add file: .\Image/system.img done,offset=0x1515000,size=0x17501c00,userspace=0x2ea04
Add file: .\backupimage\backup.img
Add file: .\backupimage\backup.img done,offset=0x18a17000,size=0x1515004,userspace=0x2a2b
Add file: .\update-script
Add file: .\update-script done,offset=0x19f2c800,size=0x3a5,userspace=0x1
Add file: .\recover-script
Add file: .\recover-script done,offset=0x19f2d000,size=0x10a,userspace=0x1
Add CRC...
```

图 21. 执行 mkupdate.bat

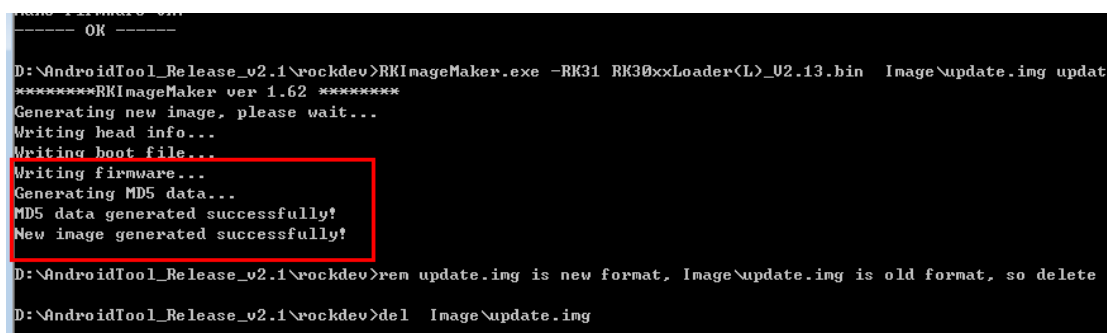


图 22. 生成固件成功

- 3) 执行完毕后在, 在 [AndroidTool Release v2.1\rockdev](#) 文件夹会生成 update.img 固件。
- 4) 升级固件的烧录方法和镜像的烧录方法大同小异, 步骤如下:
打开 AndroidTool.exe, 选择 **Upgrade Firmware** 选项卡下的 **Firmware**, 选择需要下载的固件。点击 **Upgrade** 开始更新固件。

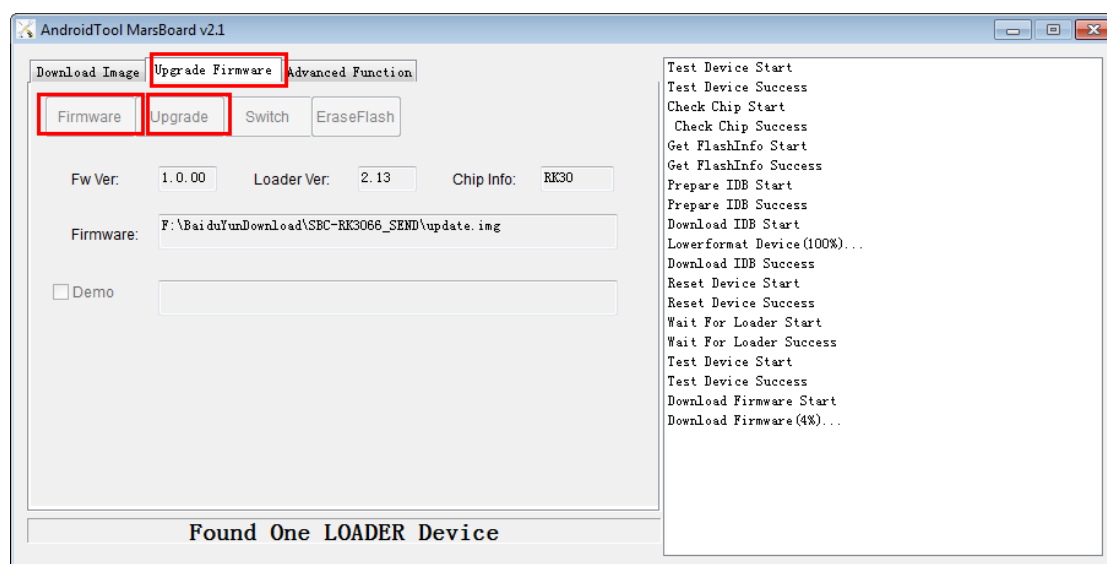


图 23. 升级固件

注意: 加载固件需要的时间较长, 请耐心等待。

4.2 升级固件的解压

- 1) 在命令提示符中定位到 [AndroidTool Release v2.1\rockdev](#) 所在目录, 解压 update.img。执行:
`RKImageMaker.exe -unpack ./update.img ./`
解压得到 boot.bin 和 firmware.img 两个文件。

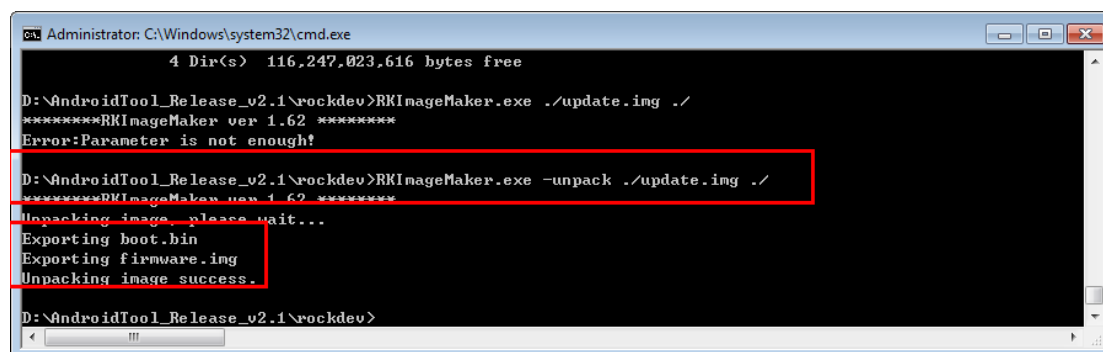


图 24. 解压 update.img

2) 解压 firmware.img。执行:

```
AFPTool.exe -unpack firmware.img ./
```

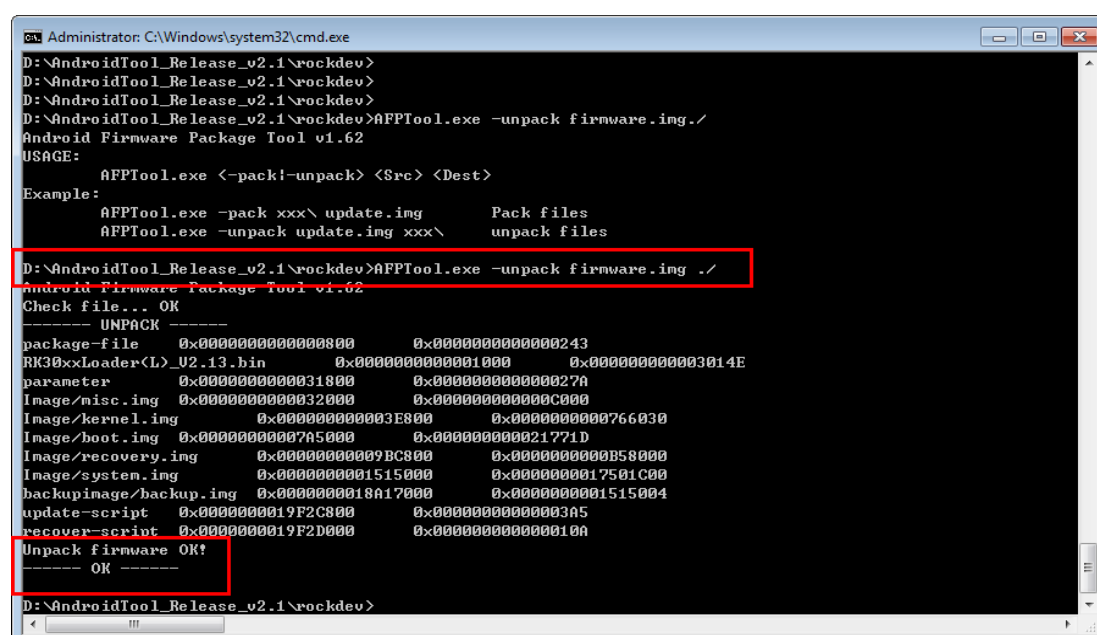


图 25. 解压 firmware.img

解压成功后的文件在 AndroidTool_Release_v2.1\rockdev\Image 目录中。

3) 更详细的说明请参考开发文档目录下的

- Android 固件生成用户手册
- Android 开发工具手册
- Rockchip Parameter File Format
- 瑞芯微 Image 打包工具使用说明

第二章 Linux 开发环境搭建与调试

1 Linux 环境搭建

SBC-RK3066 提供 Ubuntu Trusty 14.04 LTS LXDE Desktop 的固件。本手册将简单讲述一下如何获取方式和源码编译下载。出厂默认是 Android_4.4.2，显示分辨率为 1024*600。用户可以自行烧录其他版本的镜像。

注意：以下提到的电源为 5V/2A 的电源适配器。

1.1 获取 Linux 固件资源

[MarsBoard_SBC-RK3066_Ubuntu_Trusty_14.04_LTS_LXDE_Desktop_HDMI_720P_Nand_V2.0.img](#)

固件说明：适用于外接 HDMI 显示器的 Ubuntu 系统。

[MarsBoard_SBC-RK3066_Ubuntu_Trusty_14.04_LTS_LXDE_Desktop_LCD_HY070CTP-HD_1024x600_Nand_V2.0.img](#)

固件说明：适用于 SBC-RK3066 接入了 LCD（型号是 HY070CTP-HD，分辨率为 1024x600）时的 Ubuntu 系统。

1.2 烧录镜像文件

- 1) 连接 MicroUSB 线，并按下图所示的按键 VOL+，直到电脑端发现新硬件。

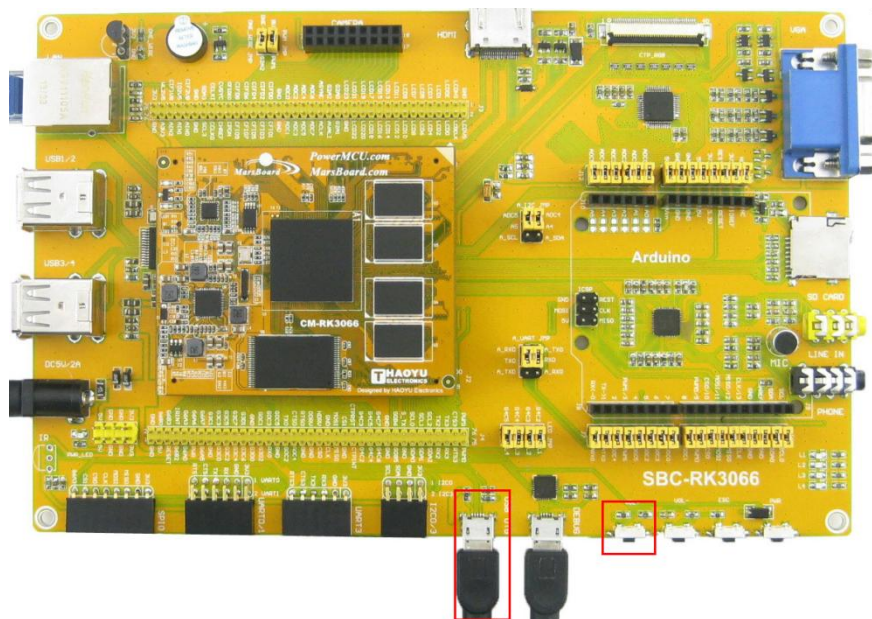


图 26. 通过 MicroUSB 线接入电脑

- 2) 安装瑞芯微驱动助手 ([Release DriverAssistant\DriverInstall.exe](#))。
- 3) 安装成功后在电脑的硬件管理界面可以看到如下的信息。

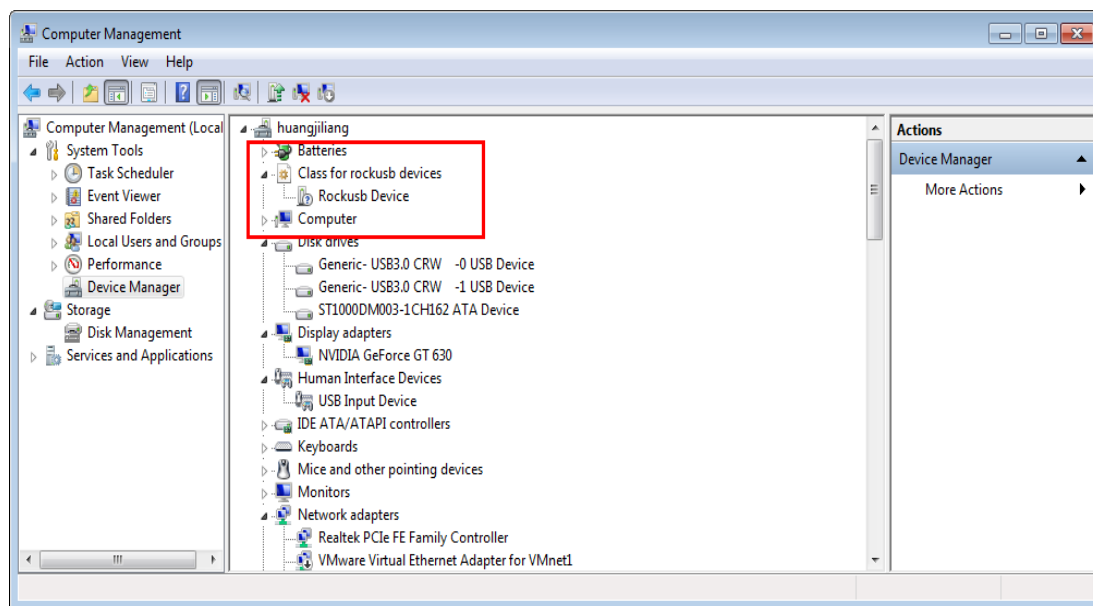


图 27. 硬件管理器查看

- 4) 在 Windows 下使用 winRAR 解压 [AndroidTool_Release_v2.1.7z](#)
- 5) 执行 `AndroidTool_Release_v2.1\AndroidTool_Release\AndroidTool.exe`

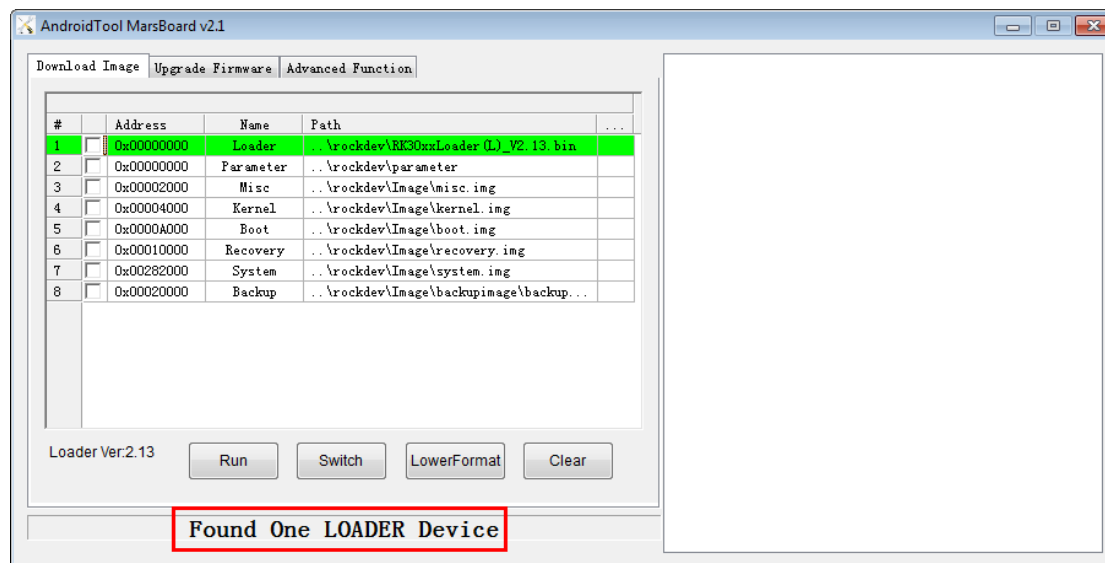


图 28. AndroidTool 界面

- 6) 解压镜像：
[MarsBoard_SBC-RK3066_Ubuntu_Trusty_14.04_LTS_LXDE_Desktop_HDMI_720P_Nand_V2.0.img.7z](#)
 选择 **Upgrade Firmware** 选项卡下的 **Firmware**，选择需要下载的固件。点击 **Upgrade** 开始更新固件

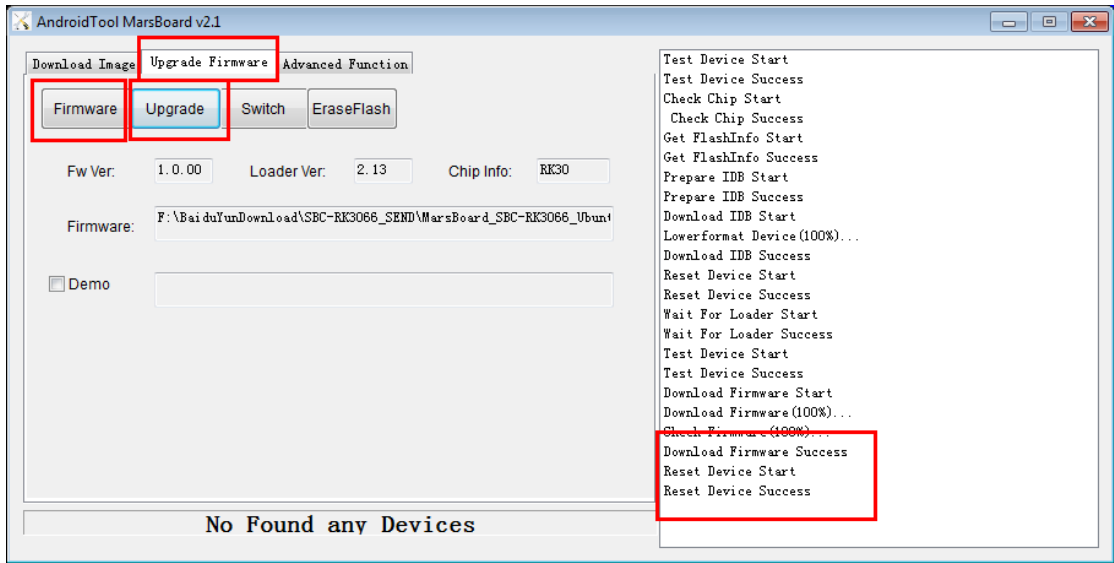


图 29. 固件下载成功

图 30. AndroidTool 参数修改框框中的参数的修改，可以参考网络资源：

- http://www.hotmcu.com/wiki/RK3066_update.img
- http://www.hotmcu.com/wiki/How_to_make_a_update.img_booting_from_SD_Card_or_Nand

以及开发文档：

- Android 固件生成用户手册
- Android 开发工具手册
- Rockchip Parameter File Format
- 瑞芯微 Image 打包工具使用说明

#		Address	Name	Path	...
1	☒	0x00000000	Loader	..\rockdev\RK30xxLoader (L)_V2.13. bin	
2	☒	0x00000000	Parameter	..\rockdev\parameter	
3	☒	0x00004000	kernel	..\rockdev\Image\linux-kernel. img	
4	☒	0x0000A000	boot	..\rockdev\Image\linux-boot. img	
5	☒	0x00012000	rootfs	..\rockdev\Image\linux-rootfs	

图 30. AndroidTool 参数修改

- 7) 点击 **Run** 下载。（也可以下载其中的某个文件）。
- 8) 注意：下载如果出现错误，请重新点击 **Run** 下载。若还是不能下载，请断开 USB OTG 并重新接上，重试下载。

2 Linux 功能调试

2.1 电脑端通过 SSH 方式对 SBC-RK3066 进行操作

- 1) 把 SBC-RK3066 接上电源，鼠标，键盘，通过 HDMI 或者 VGA 连接到电脑显示器，或者连接 LCD 显示屏。
- 2) 把 SBC-RK3066 和电脑接入同一台路由器。执行以下步骤之前，需确保 SBC-RK3066 和电脑在局域网内可以互相发现。

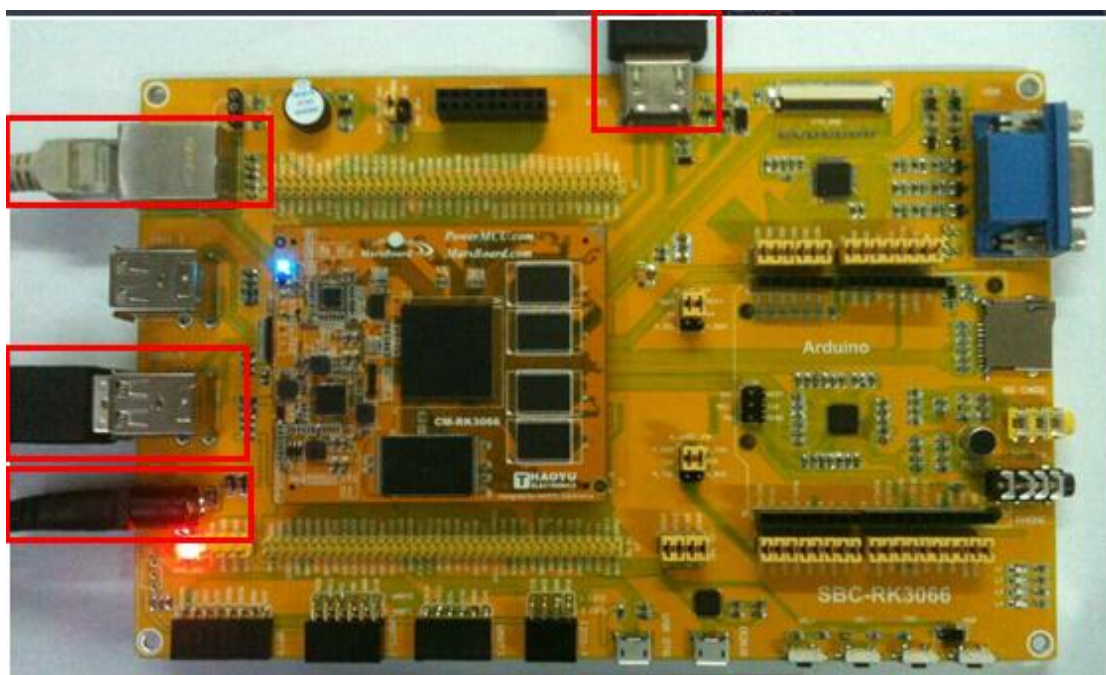


图 31. 把 SBC-RK3066 接入路由器

- 3) 在 SBC-RK3066 的登陆界面输入密码：marsboard（输入密码是不显示字符的，输入完成之后直接按回车确认即可）
- 4) 进入 SBC-RK3066 的 ubuntu 桌面后按 Ctrl+Alt+T 进入终端。
- 5) 输入 ifconfig 获取 SBC-RK3066 的 IP 地址，本例获取到的 IP 地址是：192.168.1.108（inet addr: 192.168.1.108）。

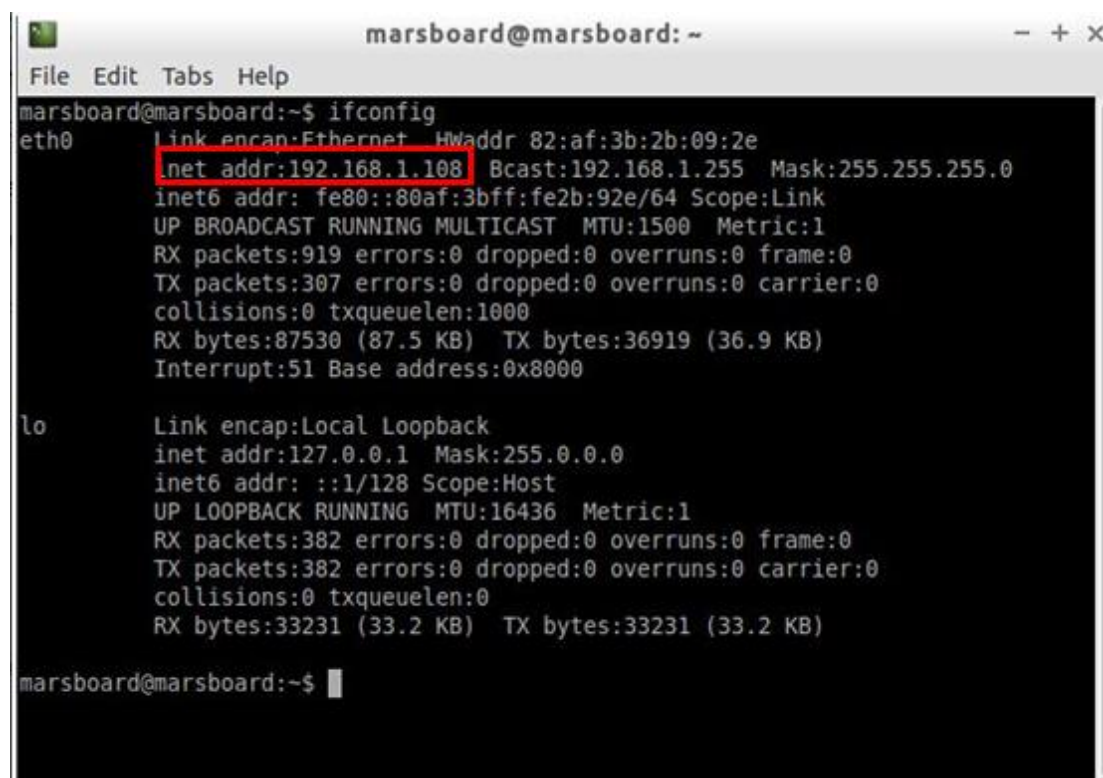


图 32. 获取 SBC-RK3066 的 IP 地址

- 6) 在电脑端打开 PuTTY 软件。
- 7) 输入板的 IP 地址，选择 SSH，点击 **open**。

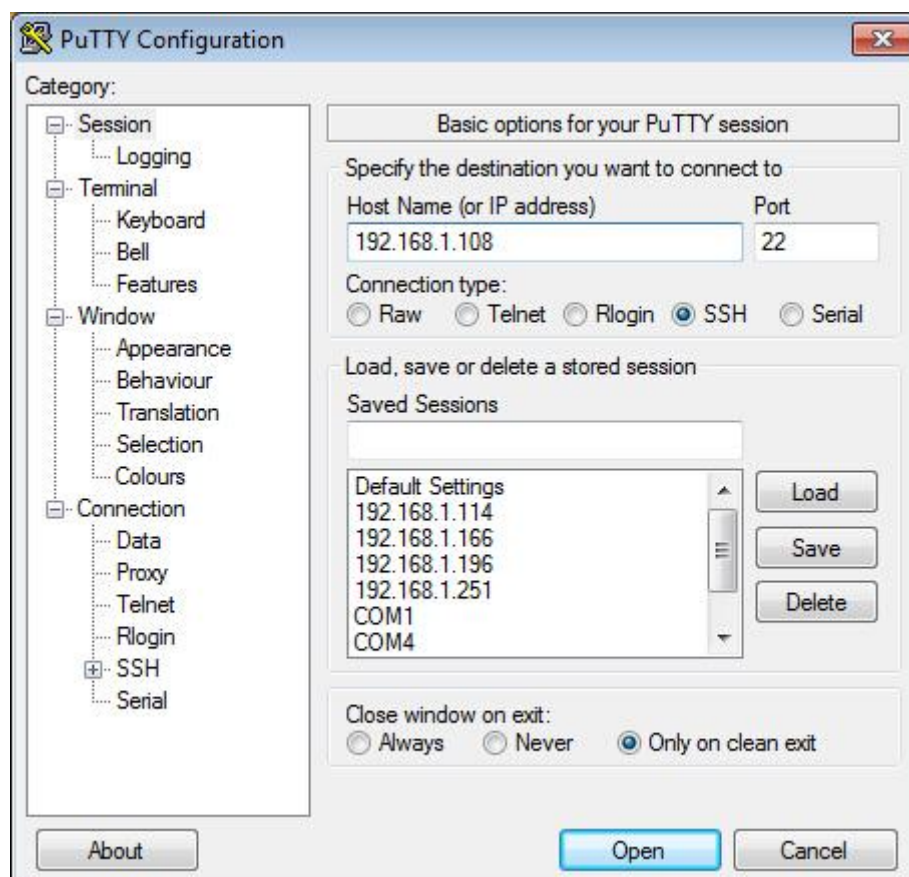


图 33. 电脑端打开 PuTTY 软件

- 8) 如果连接成功,可以直接从电脑端操作 SBC-RK3066。登陆之后需要输入账户: marsboard 密码 marsboard 回车。

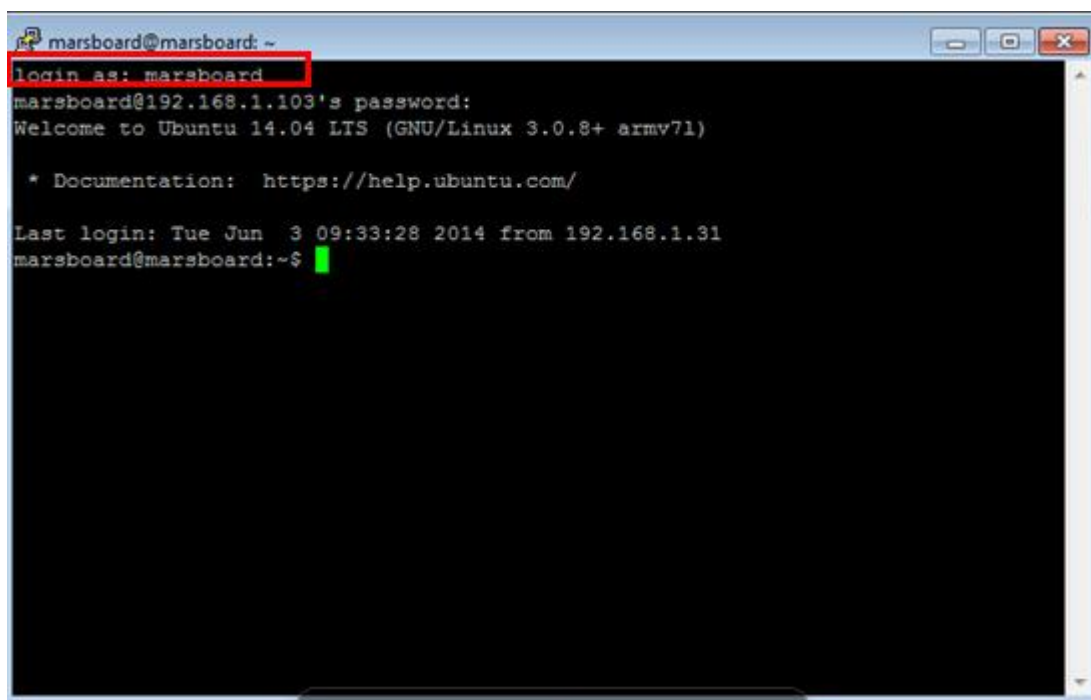


图 34. SSH 方式登陆 SBC-RK3066

- 9) 开启 root 权限, 输入以下命令:

```
sudo -s
```

密码: marsboard (输入密码是不显示字符的, 输入完成之后直接按回车确认即可)

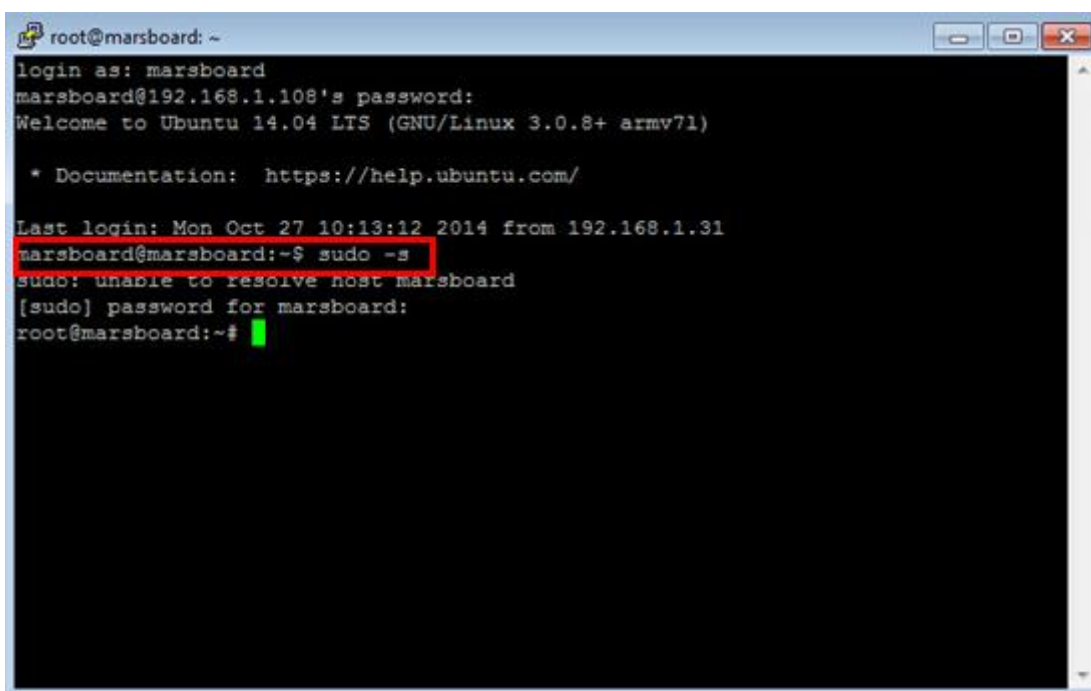


图 35. 开启 root 权限

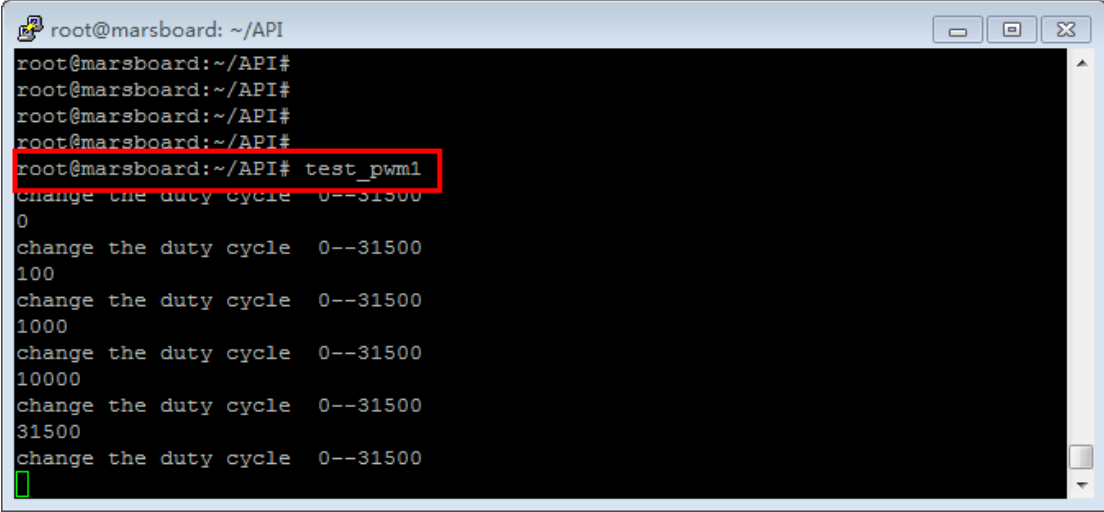
注意：以下步骤除非特别说明，均通过 SSH 操作，

2.2 LED 测试

- 1) 确保四个 LED JMP 跳线帽短接。
- 2) 开发板终端下输入：
`test_led`
- 3) SBC-RK3066 上 led 灯闪烁变化。

2.3 PWM 测试

- 1) 确保 BUZ JMP 跳线帽短接。
- 2) 开发板终端下输入：
`test_pwm1`
- 3) 输入数值，对 PWM 信号脉宽进行调整。
- 4) 可以听到蜂鸣器响声不同的变化。



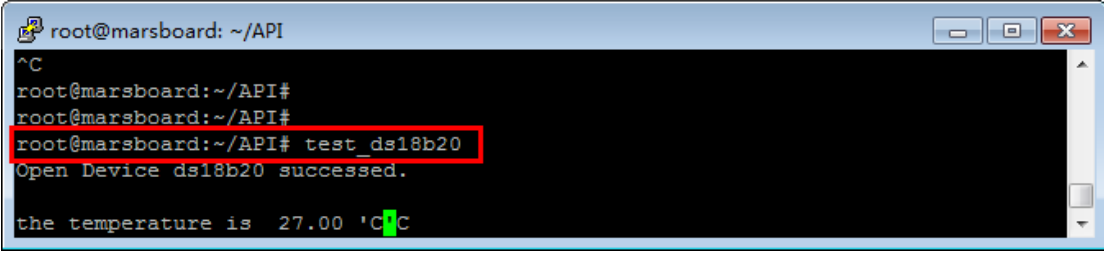
```
root@marsboard: ~/API
root@marsboard:~/API#
root@marsboard:~/API#
root@marsboard:~/API#
root@marsboard:~/API# test_pwm1
change the duty cycle 0--31500
0
change the duty cycle 0--31500
100
change the duty cycle 0--31500
1000
change the duty cycle 0--31500
10000
change the duty cycle 0--31500
31500
change the duty cycle 0--31500
```

图 36. PWM 测试

- 5) 按键盘 ctrl+c 结束测试。

2.4 DS18B20 测试

- 1) 确保 ONE-WIRE JMP 跳线短接。
- 2) 接入 ds18b20 温度传感器到 ONE-WIRE 接口。
- 3) 开发板终端输入：
`test_ds18b20`
- 4) 终端下显示获取的当前环境温度。



```

root@marsboard: ~/API
^C
root@marsboard:~/API#
root@marsboard:~/API#
root@marsboard:~/API# test_ds18b20
Open Device ds18b20 succeeded.
the temperature is 27.00 'C

```

图 37. DS18B20 测试

- 5) 按键盘 ctrl+c 结束测试。

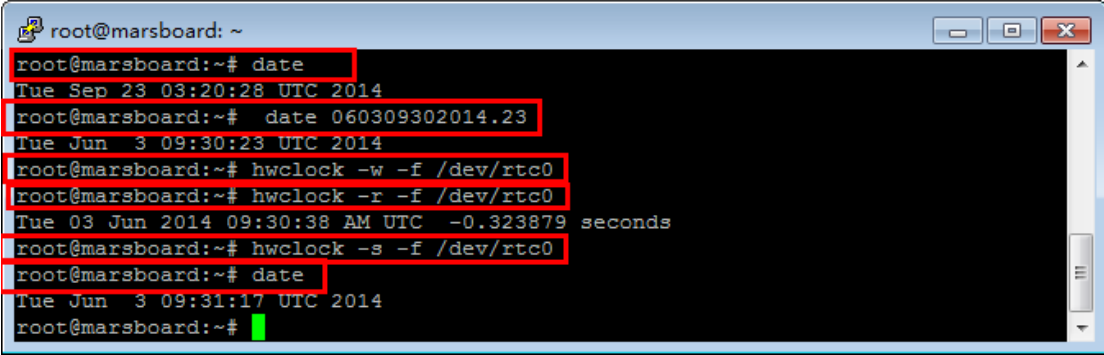
2.5 RTC 测试

- 1) 确保 RTC 跳线帽短接。

注意：将纽扣电池接上。不然时间不能保存，掉电丢失。

- 2) 开发板终端下输入命令：

<code>date</code>	读取系统时间
<code>date 060309302014.23</code>	设置系统时间
<code>hwclock -w -f /dev/rtc0</code>	设置 RTC 硬件时间
<code>hwclock -r -f /dev/rtc0</code>	读取 RTC 时钟模块的硬件时间
<code>hwclock -s -f /dev/rtc0</code>	RTC 时钟模块的硬件时间同步到系统时间



```

root@marsboard: ~
root@marsboard:~# date
Tue Sep 23 03:20:28 UTC 2014
root@marsboard:~# date 060309302014.23
Tue Jun 3 09:30:23 UTC 2014
root@marsboard:~# hwclock -w -f /dev/rtc0
root@marsboard:~# hwclock -r -f /dev/rtc0
Tue 03 Jun 2014 09:30:38 AM UTC -0.323879 seconds
root@marsboard:~# hwclock -s -f /dev/rtc0
root@marsboard:~# date
Tue Jun 3 09:31:17 UTC 2014
root@marsboard:~#

```

图 38. RTC 测试

- 3) 按键盘 ctrl+c 结束测试。

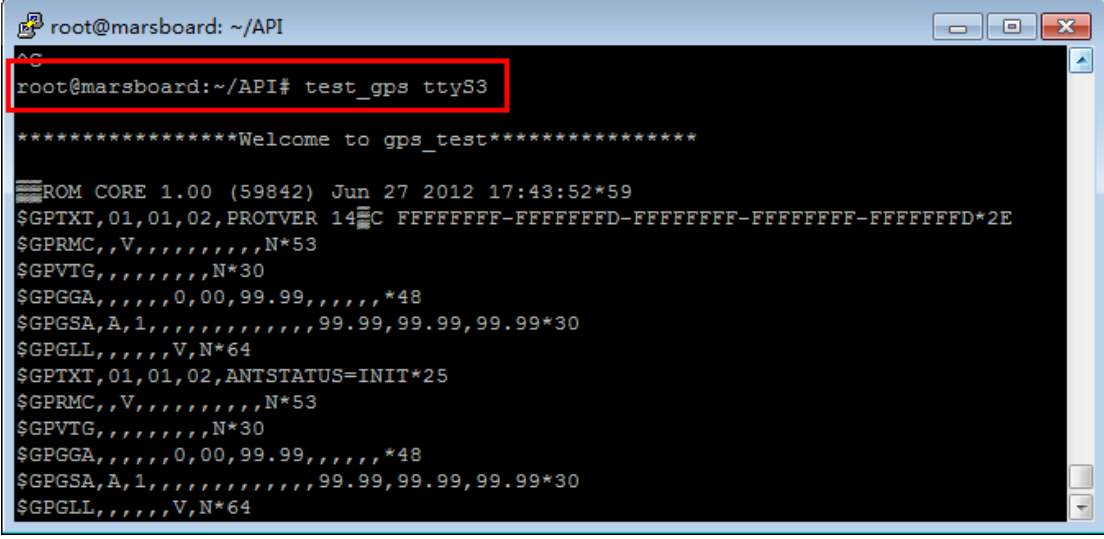
2.6 GPS 测试

- 1) 接入 UART GPS NEO-7M 模块到 UART3 接口。

- 2) 开发板终端下输入：

```
test_gps ttyS3
```

- 3) 终端显示定位状态及经纬度信息。

A terminal window titled 'root@marsboard: ~/API' showing the execution of the 'test_gps ttyS3' command. The command is highlighted with a red box. The output displays a 'Welcome to gps_test' message followed by various NMEA-style strings such as '\$GPRMC', '\$GPVTG', '\$GPGGA', '\$GPGSA', and '\$GPGLL', along with system information like 'ROM CORE 1.00 (59842) Jun 27 2012 17:43:52*59'.

```
root@marsboard: ~/API
root@marsboard:~/API# test_gps ttyS3

*****Welcome to gps_test*****

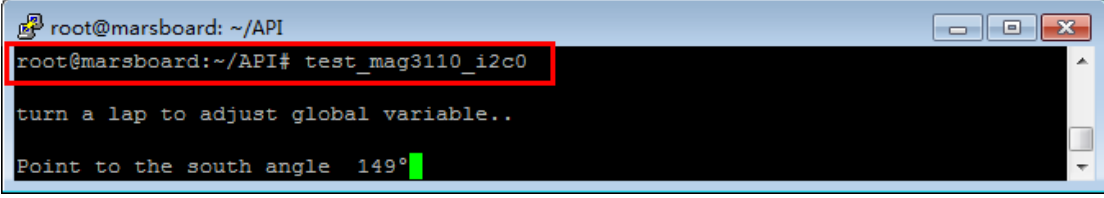
ROM CORE 1.00 (59842) Jun 27 2012 17:43:52*59
$GPTXT,01,01,02,PROTVER 14C FFFFFFFF-FFFFFFFD-FFFFFFF-FFFFFFF-FFFFFFFD*2E
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,,,,,,,,N*30
$GPGGA,,,,,0,00,99.99,,,,,*48
$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30
$GPGLL,,,,,V,N*64
$GPTXT,01,01,02,ANTSTATUS=INIT*25
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,,,,,,,,N*30
$GPGGA,,,,,0,00,99.99,,,,,*48
$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30
$GPGLL,,,,,V,N*64
```

图 39. GPS 测试

- 4) 按键盘 ctrl+c 结束测试。

2.7 MAG3110 测试

- 1) 接入 MAG3110 Board 到 I2C0 接口。
- 2) 开发板终端下输入：
`test_mag3110_i2c0`
- 3) 旋转传感器来校准参数，校准完成后终端下显示获取的地磁信息。

A terminal window titled 'root@marsboard: ~/API' showing the execution of the 'test_mag3110_i2c0' command. The command is highlighted with a red box. The output shows instructions to 'turn a lap to adjust global variable..' and 'Point to the south angle 149°' with a green cursor at the end.

```
root@marsboard: ~/API
root@marsboard:~/API# test_mag3110_i2c0

turn a lap to adjust global variable..

Point to the south angle 149°
```

图 40. MAG3110 测试

- 4) 按键盘 ctrl+c 结束测试。

2.8 AT45DB E2PROM 测试

- 1) 接入 AT45DBXX DataFlash Board 到 SPI0 接口。
- 2) 开发板终端下输入：
`test_at45db`
- 3) 3) 终端下显示写入和读出数据。

```
root@marsboard: ~/API
Point to the south angle 0°0'0"
root@marsboard:~/API# test_at45db
spi mode: 1
bits per word: 8
max speed: 50000 Hz (50 KHz)
>>>>>>>Start to write [0-255]<<<<<<<<
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62
63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82
83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101
102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117
118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133
134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149
150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165
166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181
182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197
198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213
214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229
230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245
246 247 248 249 250 251 252 253 254 255
>>>>>>>Start to read [0-255]<<<<<<<<
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62
63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82
83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101
102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117
118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133
134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149
150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165
166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181
182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197
198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213
214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229
230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245
246 247 248 249 250 251 252 253 254 255
root@marsboard:~/API#
```

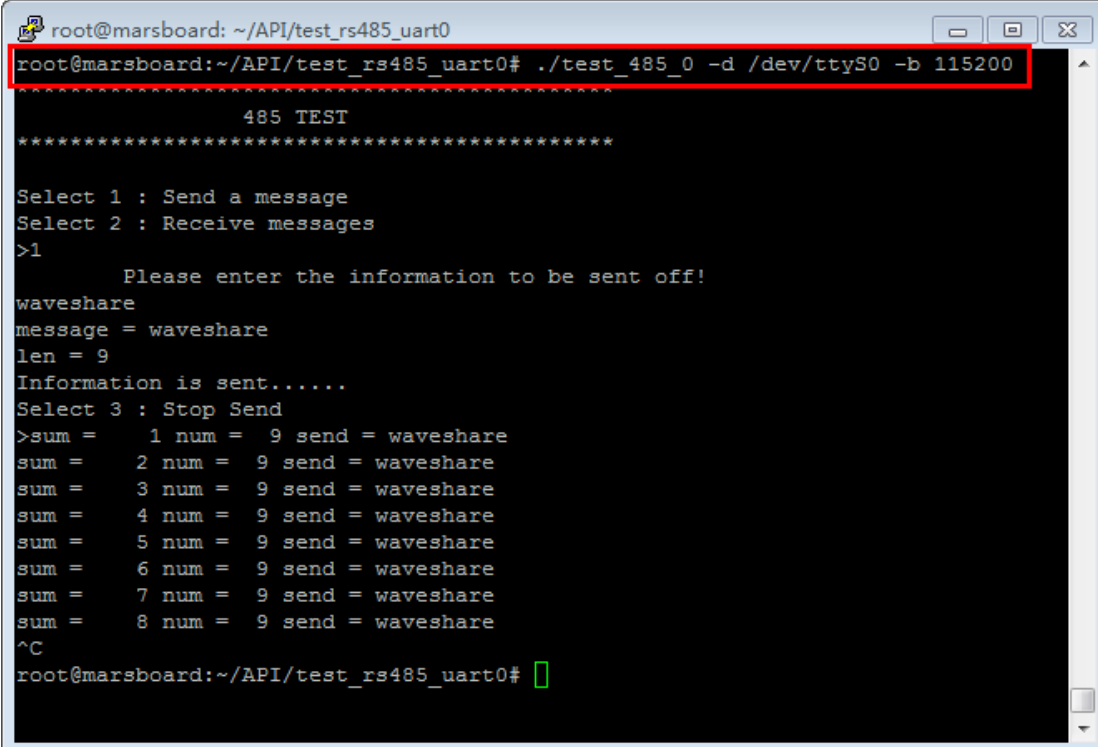
❄ 41. AT45DB E2PROM 测试

- 4) 按键盘 ctrl+c 结束测试。

2.9 RS485 接收发送测试

- 1) 分别接入两个 RS485 Board 到 UART0 和 UART3 接口,并用杜邦线把模块上的 A 和 B 对应连接。
- 2) 打开两个终端(都通过网络访问),分别输入命令:

```
cd /home/marsboard/API/test_rs485_uart0  
./test_485_0 -d /dev/ttyS0 -b 115200  
cd /home/marsboard/API/test_rs485_uart3  
./test_485_3 -d /dev/ttyS3 -b 115200
```
- 3) 按照提示选择相应的功能,实现一个终端发一个终端收。
- 4) 分别查看发送数据的终端和接收数据的终端。
 - 发送数据的送端:

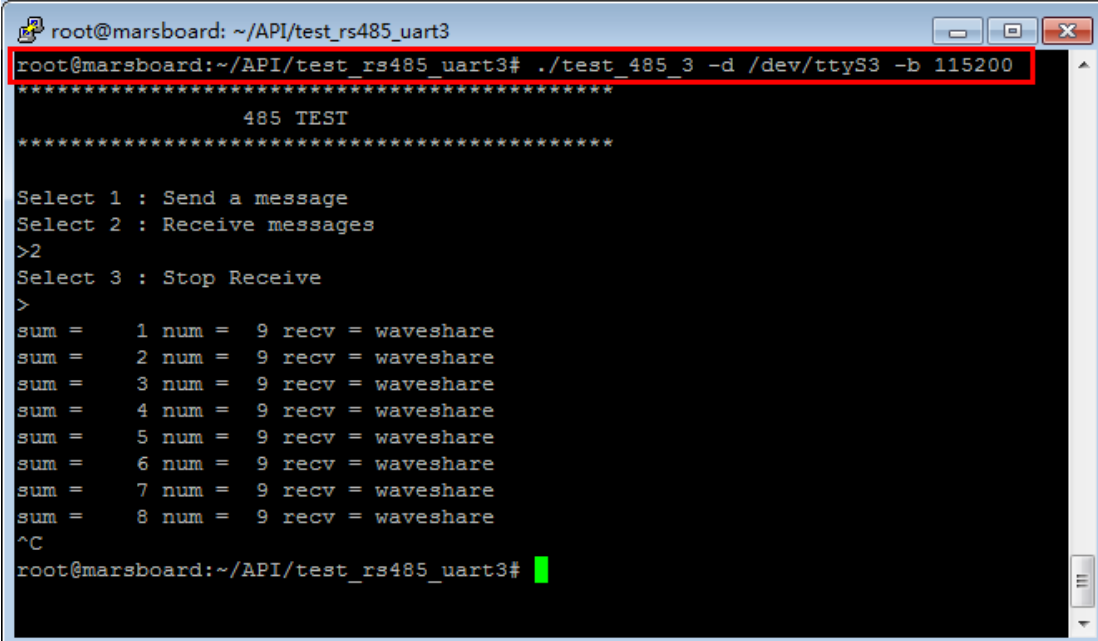


A terminal window titled 'root@marsboard: ~/API/test_rs485_uart0'. The command line is highlighted with a red box: 'root@marsboard:~/API/test_rs485_uart0# ./test_485_0 -d /dev/ttyS0 -b 115200'. The output shows a menu with '485 TEST' and options to send or receive messages. Option 1 (Send a message) is selected. The program prompts for a message, and 'waveshare' is entered. It then displays a loop of 8 iterations, each showing 'sum =', 'num = 9', and 'send = waveshare'. The terminal ends with a green cursor at the prompt 'root@marsboard:~/API/test_rs485_uart0#'.

```
root@marsboard: ~/API/test_rs485_uart0
root@marsboard:~/API/test_rs485_uart0# ./test_485_0 -d /dev/ttyS0 -b 115200
*****
485 TEST
*****
Select 1 : Send a message
Select 2 : Receive messages
>1
Please enter the information to be sent off!
waveshare
message = waveshare
len = 9
Information is sent.....
Select 3 : Stop Send
>sum = 1 num = 9 send = waveshare
sum = 2 num = 9 send = waveshare
sum = 3 num = 9 send = waveshare
sum = 4 num = 9 send = waveshare
sum = 5 num = 9 send = waveshare
sum = 6 num = 9 send = waveshare
sum = 7 num = 9 send = waveshare
sum = 8 num = 9 send = waveshare
^C
root@marsboard:~/API/test_rs485_uart0#
```

图 42. RS485 发送数据终端

- 接收数据的收端:



A terminal window titled 'root@marsboard: ~/API/test_rs485_uart3'. The command line is highlighted with a red box: 'root@marsboard:~/API/test_rs485_uart3# ./test_485_3 -d /dev/ttyS3 -b 115200'. The output shows a menu with '485 TEST' and options to send or receive messages. Option 2 (Receive messages) is selected. The program prompts for a message, and 'waveshare' is entered. It then displays a loop of 8 iterations, each showing 'sum =', 'num = 9', and 'recv = waveshare'. The terminal ends with a green cursor at the prompt 'root@marsboard:~/API/test_rs485_uart3#'.

```
root@marsboard: ~/API/test_rs485_uart3
root@marsboard:~/API/test_rs485_uart3# ./test_485_3 -d /dev/ttyS3 -b 115200
*****
485 TEST
*****
Select 1 : Send a message
Select 2 : Receive messages
>2
Select 3 : Stop Receive
>
sum = 1 num = 9 recv = waveshare
sum = 2 num = 9 recv = waveshare
sum = 3 num = 9 recv = waveshare
sum = 4 num = 9 recv = waveshare
sum = 5 num = 9 recv = waveshare
sum = 6 num = 9 recv = waveshare
sum = 7 num = 9 recv = waveshare
sum = 8 num = 9 recv = waveshare
^C
root@marsboard:~/API/test_rs485_uart3#
```

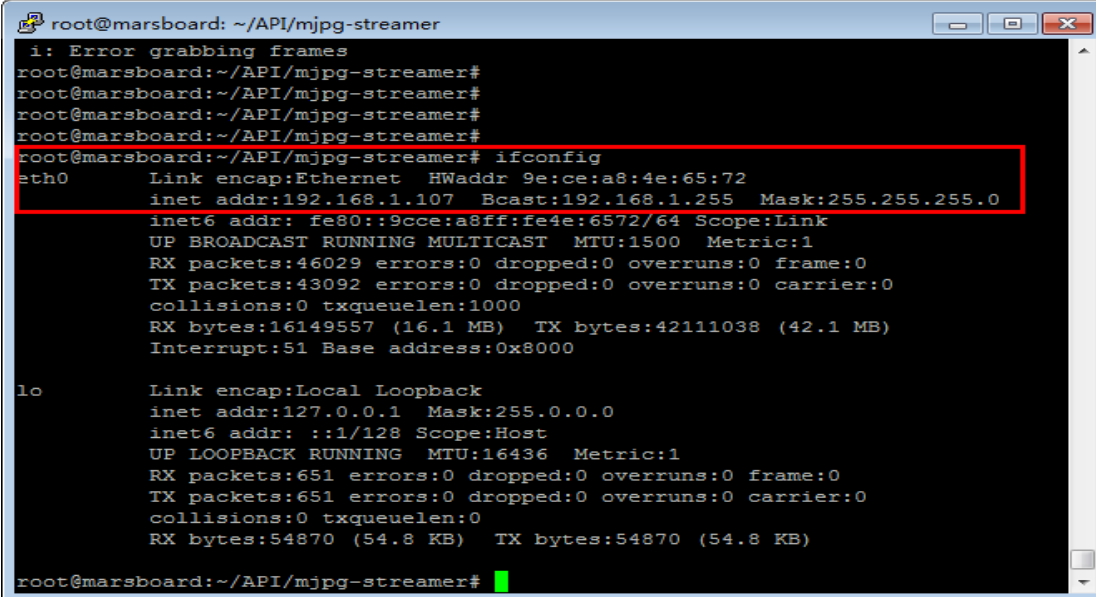
图 43. RS485 接收数据终端

2.10 USB CAMERA 测试

- 1) 接入 USB Camera 到 USB Host 接口，并插入网线。

- 2) 输入以下命令查看 ip:

```
ifconfig eth0
```



```

root@marsboard: ~/API/mjpg-streamer
i: Error grabbing frames
root@marsboard:~/API/mjpg-streamer#
root@marsboard:~/API/mjpg-streamer#
root@marsboard:~/API/mjpg-streamer#
root@marsboard:~/API/mjpg-streamer# ifconfig
eth0      Link encap:Ethernet  HWaddr 9e:ce:a8:4e:65:72
          inet addr:192.168.1.107  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::9cce:a8ff:fe4e:6572/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:46029 errors:0 dropped:0 overruns:0 frame:0
          TX packets:43092 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:16149557 (16.1 MB)  TX bytes:42111038 (42.1 MB)
          Interrupt:51 Base address:0x8000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:651 errors:0 dropped:0 overruns:0 frame:0
          TX packets:651 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:54870 (54.8 KB)  TX bytes:54870 (54.8 KB)

root@marsboard:~/API/mjpg-streamer#

```

图 44. USB CAMERA 测试

可以看到 IP 地址为 192.168.1.107

- 3) 开发板输入:

```
cd /home/marsboard/API/mjpg-streamer
./start.sh
```

- 4) 在局域网内的电脑上打开浏览器，输入地址:

<http://192.168.1.107:8080/javascript.html>

注意: ip 地址要以开发板的 ip 地址一致，根据实际修改!

- 5) 可以看到开发板作为 web 服务器上传的视频流。

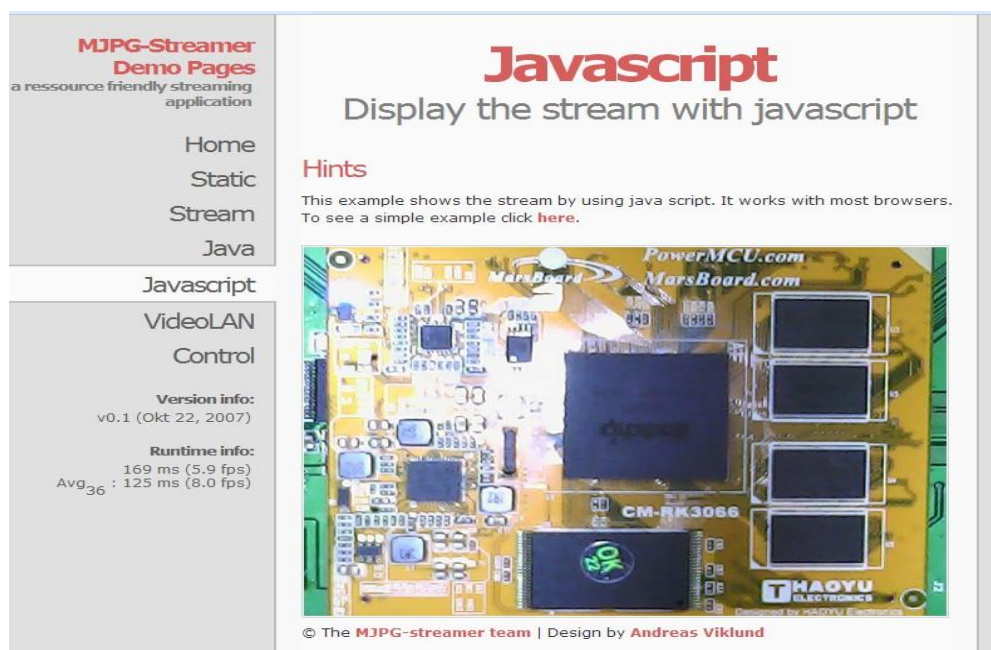


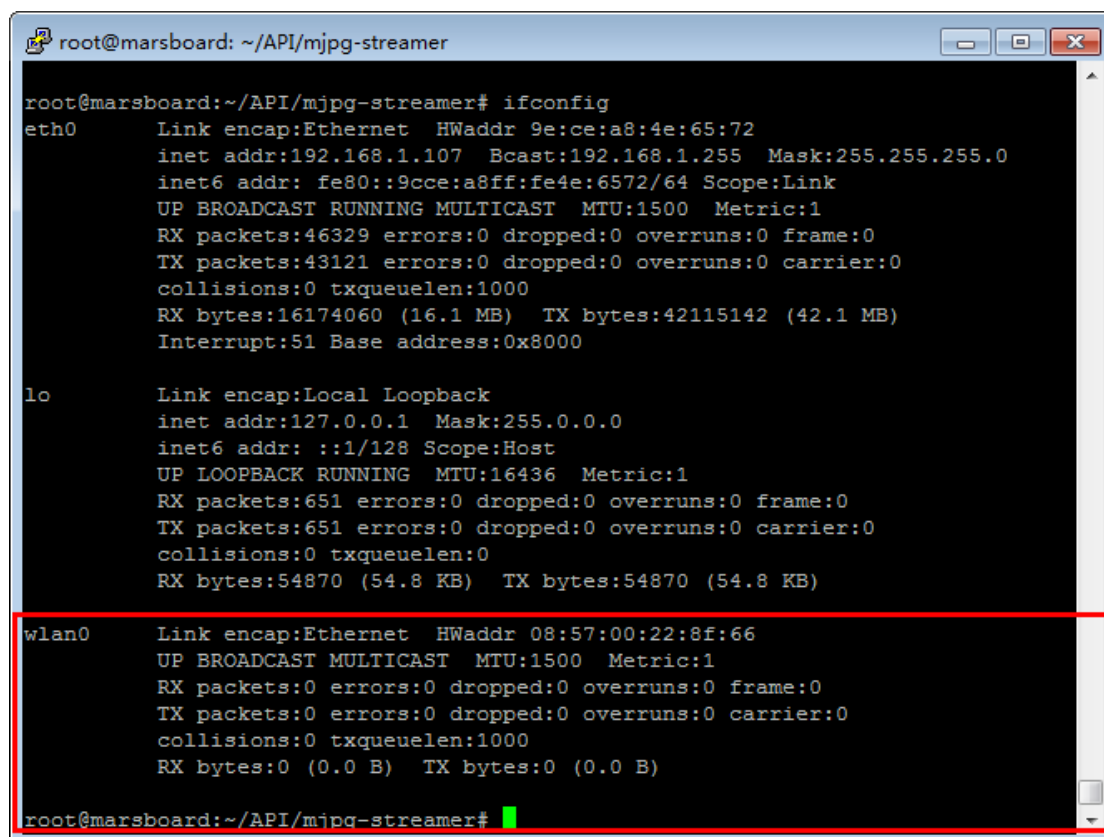
图 45. USB CAMERA 测试

- 6) 按键盘 ctrl+c 结束测试。

2.11 USB Wi-Fi 测试

- 1) 接入 USB 无线网卡，在终端下输入相应命令：
- 2) 查看网络状态：

```
ifconfig -a
```



```
root@marsboard: ~/API/mjpg-streamer
root@marsboard:~/API/mjpg-streamer# ifconfig
eth0      Link encap:Ethernet  HWaddr 9e:ce:a8:4e:65:72
          inet addr:192.168.1.107  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::9cce:a8ff:fe4e:6572/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:46329 errors:0 dropped:0 overruns:0 frame:0
          TX packets:43121 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:16174060 (16.1 MB)  TX bytes:42115142 (42.1 MB)
          Interrupt:51 Base address:0x8000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:651 errors:0 dropped:0 overruns:0 frame:0
          TX packets:651 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:54870 (54.8 KB)  TX bytes:54870 (54.8 KB)

wlan0     Link encap:Ethernet  HWaddr 08:57:00:22:8f:66
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

root@marsboard:~/API/mjpg-streamer#
```

图 46. 查看网络状态

可以看到 wlanx 的网络节点，则说明 usb 无线网卡驱动成功。

- 3) 通过命令行连接 wifi 热点：

```
iwlist wlan0 scanning
```

```

root@marsboard: ~
root@marsboard:~# iwlist wlan0 scanning
wlan0 Scan completed :
      Cell 01 - Address: 5C:63:BF:46:EA:B8
                ESSID:"Waveshare Net"
                Protocol:IEEE 802.11bgn
                Mode:Master
                Frequency:2.427 GHz (Channel 4)
                Encryption key:on
                Bit Rates:300 Mb/s
                Extra:rsn_ie=30140100000fac040100000fac040100000fac020000
                IE: IEEE 802.11i/WPA2 Version 1
                  Group Cipher : CCMP
                  Pairwise Ciphers (1) : CCMP
                  Authentication Suites (1) : PSK
                IE: Unknown: DD810050F204104A0001101044000102103B00010310470
0100000000000000100000005C63BF46EAB81021000754502D4C494E4B10230009544C2D575238343
04E10240003312E3010420003312E301054000800060050F204000110110019576972656C6573732
0526F7574657220544C2D57523834304E100800020086103C000101
                Quality=64/100 Signal level=-60 dBm
      Cell 02 - Address: 08:10:78:86:00:A2
                ESSID:"360WiFi-2291"
                Protocol:IEEE 802.11bgn
                Mode:Master
                Frequency:2.427 GHz (Channel 4)

```

图 47. 连接 Wi-Fi 热点

- 4) 连接热点（WaveshareNet 即为热点名称，根据实际修改）：
 - a) 禁用有线网络：


```
wpa_cli -p/var/run/wpa_supplicant remove_network 0
```
 - b) 扫描热点：


```
wpa_cli -p/var/run/wpa_supplicant ap_scan 1
```
 - c) 添加热点：


```
wpa_cli -p/var/run/wpa_supplicant add_network
```
 - d) 设置连接的热点的 SSID：


```
wpa_cli -p/var/run/wpa_supplicant set_network 0 ssid "WaveshareNet"
```
 - e) 设置热点的密码（根据实际修改）：


```
wpa_cli -p/var/run/wpa_supplicant set_network 0 psk "12345678"
```
 - f) 设置热点为首选的连接网络：


```
wpa_cli -p/var/run/wpa_supplicant select_network 0
```
 - g) 设置开发板的 IP 地址和子网掩码：


```
ifconfig wlan0 192.168.1.196 netmask 255.255.255.0
```
 - h) 设置默认网关：


```
route add default gw 192.168.1.1
```
- 5) 测试网络情况，终端下输入：


```
ping www.wvshare.com
```

查看连接情况。

注意：

- 确保要 ping 的网站通过电脑可以正常访问。如果 ping 不通，可能是没有拔掉网线。
- 可以通过执行 [/home/marsboard/API/](#) 中的 net_connect_config.sh 来配置无线网络，注意要修改一些必要的参数（无线网络的设备号，SSID，PSK 等）

```
cd /home/marsboard/API  
./net_connect_config.sh
```

2.12 USB HUB 测试

- 1) 接入 USB 键盘或 USB 鼠标等 USB 设备到 SBC-RK3066 的 USB 接口。
- 2) 系统可以识别鼠标和键盘等 USB 设备。

注意：USB 接口在不直接移动硬盘，容易损坏。

2.13 SD 卡测试

- 1) 插入 SD 卡
- 2) 终端输入：

```
cd /media/marsboard/D678-00FB
```

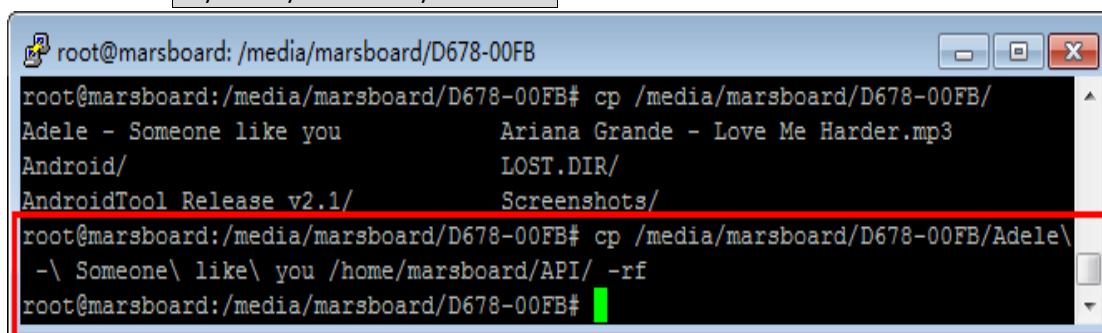


图 48. 定位到 SD 卡

可以通过 ls 命令查看到 SD 卡里面的内容：

```
ls
```

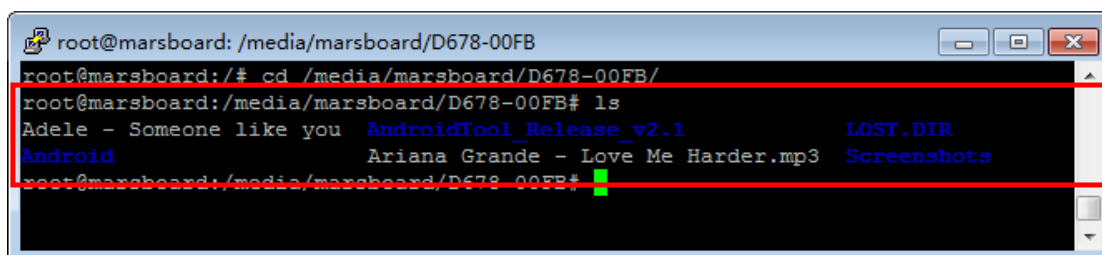


图 49. 查看 SD 卡内容

- 3) 可以通过 cp 命令复制 SD 卡里面的内容：

```
cp /media/marsboard/D678-00FB/ /home/marsboard/API/ -rf
```

```
ls
```

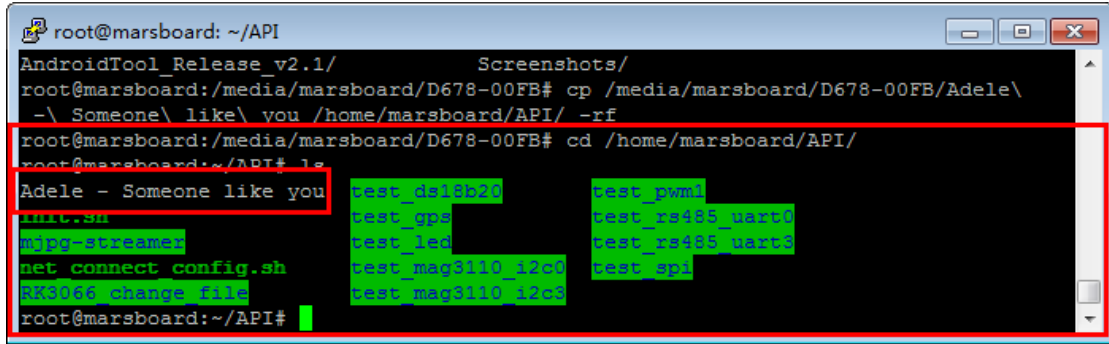


图 50. 复制 SD 卡内容

- 4) 成功复制，SD 卡测试结束。

3 Linux 内核编译

源码编译适用于需要深度定制 Linux 系统的用户，轻度用户可以跳过阅读本节。

3.1 获取 Linux 源码资源

电脑端必须安装 Linux 系统以便编译。本手册以 Ubuntu12.04 环境为例。

- 1) 更新与下载必要的包，电脑端终端执行：

```
sudo apt-get install git-core gnupg flex bison gperf libssl-dev libbsd-dev
libxgtk2.8-dev build-essential zip curl libncurses5-dev zlib1g-dev ia32-libs
lib32z1-dev lib32ncurses5-dev gcc-multilib g++-multilib sharutils lzop
```

- 2) [点击下载测试镜像](#)
- 3) [点击下载源码](#)

注意：以下步骤如果没有特别说明，均是在电脑版的 ubuntu12.04 操作。

3.2 Linux 内核编译

- 1) 解压源码：

```
unzip rockchip-marsboard-rk3066-linux-3.0.8-master.zip
```

- 2) 编译 SD 卡 Recovery 模式启动内核：

```
./build_marsboard_rk3066_recovery_sdcard
```

- 3) 编译 SD 卡启动内核：

```
./build_marsboard_rk3066_sdcard
```

- 4) 编译 Nand 启动内核：

```
./build_marsboard_rk3066_mtd
```

- 5) 每执行完一次以上的脚本，系统都会清理内核，并重新进行编译，内核的所有关联的代码也会重新编译，会浪费较多的时间。为了解决这个问题，用户需要修改一下脚本：

以修改 build_marsboard_rk3066_mtd 脚本为例。

```
vim build_marsboard_rk3066_mtd
```

注释该脚本第 3 行和第 7 行的代码（在代码前面加#即可注释）

```
#make mrproper
```

```
#ARCH=arm CROSS_COMPILE=./toolchain/arm-eabi-4.6/bin/arm-eabi-make marsboard_rk3066_mtd_defconfig
```

- 完成上述步骤之后，即可看到编译后生成的可下载的镜像文件。

注意：内核编译时如果遇到 `linux/in_route.h: No such file or directory` 错误，通常是因为缺少 `in_route.h`，可以在相关源码/`in_route.h` 复制 `in_route.h` 到内核源码的根目录下，再复制到 `include/linux` 目录下。

3.3 升级固件的制作与烧录

AndroidTool 软件提供了固件升级的功能，用户可以把多个系统打包到同一个固件中，便于量产。

- 1) 下载 [rk-tools.zip](#) 工具
- 2) 解压到 ubuntu 系统中

```
unzip rockchip-pack-tools-master.zip
cd rockchip-pack-tools-master
sudo apt-get install libssl-dev libcrypto++-dev
```
- 3) 复制需要制作成固件的文件到 `mkupdate.sh` 所在的目录中。（例如： `./RK3066/rockchip-pack-tools/linux`）
- 4) 在 shell 终端上执行：

```
./mkupdate.sh
```
- 5) 最终会在该目录下生成 `update.img` 的文件。
- 6) 升级固件的烧录方法和镜像的烧录方法大同小异，步骤如下：
打开 `AndroidTool.exe`，选择 `Upgrade Firmware` 选项卡下的 `Firmware`，选择需要下载的固件。点击 `Upgrade` 开始更新固件。
注意：加载固件需要的时间较长，请耐心等待。