

# **Rockchip**

## **RK3399 软件开发指南**

发布版本:1.00

日期:2016.08

# 前言

## 概述

文档主要介绍 Rockchip RK3399 软件开发指南，旨在帮助软件开发工程师更快上手 RK3399 的开发及调试

## 产品版本

芯片名称	内核版本	Android 版本
RK3399	Linux4.4	Android6.0.1

## 读者对象

本文档（本指南）主要适用于以下工程师：  
技术支持工程师  
软件开发工程师

## 修订记录

日期	版本	作者	修改说明
2016-08-09	V1.00	YHX,HZB	构建初始版本 缺漏第 6、7 章节

# 目录

1	支持列表 .....	1-1
1.1	DDR 支持列表 .....	1-1
1.2	EMMC 支持列表 .....	1-1
1.3	WiFi/BT 支持列表 .....	1-2
1.4	SDK 软件包适用硬件列表 .....	1-2
1.5	多媒体编解码支持列表 .....	1-3
2	文档/工具索引 .....	2-1
2.1	文档索引 .....	2-1
2.2	工具索引 .....	2-2
3	SDK 编译/烧写 .....	3-1
3.1	SDK 获取 .....	3-1
3.2	SDK 编译 .....	3-1
3.3	固件烧写 .....	3-2
3.4	量产烧写 .....	3-3
4	U-Boot 开发 .....	4-1
4.1	Rockchip U-Boot 简介 .....	4-1
4.2	平台配置 .....	4-1
4.3	固件生成 .....	4-1
4.4	U-Boot 编译 .....	4-2
5	内核常见配置 .....	5-1
5.1	WiFi&BT 的配置 .....	5-1
5.2	U-Boot logo 相关的配置 .....	5-2
5.3	ARM、GPU 频率修改 .....	5-2
6	Android 常见配置 .....	6-1
6.1	Android 编译配置 .....	6-1
6.2	常用配置说明 .....	6-1
6.3	预制 APK .....	6-1
6.4	开/关机动画 .....	6-1
6.5	开/关机铃声 .....	6-1
6.6	Parameter 说明 .....	6-1
6.7	新增分区配置 .....	6-1
6.8	系统调试 .....	6-1
6.9	OTA 升级 .....	6-1
6.10	预制 Demo .....	6-1
6.11	系统灾难恢复 .....	6-1
7	常用工具说明 .....	7-1
7.1	StressTest .....	7-1
7.2	DeviceTest .....	7-1
7.3	PCBA 测试工具 .....	7-1
7.4	DDR 测试工具 .....	7-1
7.5	update.img 打包 .....	7-1
7.6	固件签名 .....	7-1

---

7.7	序列号/Mac/厂商信息烧写.....	7-1
7.8	量产工具使用.....	7-1

# 插图目录

图 1-1 EMMC Performance 示例..... 1-2

图 1-2 RK3399 目前大量测试的 Wifi/Bt 支持列表 ..... 1-2

图 3-1Android 开发工具烧写界面 ..... 3-3

# 表格目录

表 1-1 RK3399 DRAM Support Type ..... 1-1

表 1-2 RK3399 DDR Support Symbol..... 1-1

表 1-3 RK3399 EMMC Support Symbol ..... 1-1

表 1-4 RK3399 硬件说明列表..... 1-3

# 1 支持列表

## 1.1 DDR 支持列表

RK3399 DDR 目前选型列表支持双通道 DDR3、DDR3L、LPDDR3。

表 1-1 RK3399 DRAM Support Type

Chip	DRAM Support Type
RK3399	DDR3/DDR3L/LPDDR3

RK3399 DDR 颗粒支持程度列表，详见 RKDocs\Platform support lists 目录下《RK DDR Support List Ver2.18》，下表中所标示的 DDR 支持程度表，只建议选用√、T/A 标示的颗粒。

表 1-2 RK3399 DDR Support Symbol

Symbol	Description
√	Fully Tested and Mass production
T/A	Fully Tested and Applicable
N/A	Not Applicable

## 1.2 EMMC 支持列表

RK3399 支持 eMMC 5.1, SDIO3.0, 可运行 HS200,HS400 模式，详见 RKDocs\Platform support lists 目录下《RKeMMCSupportList Ver1.29\_2016\_08\_05》，下表中所标示的 DDR 支持程度表，只建议选用√、T/A 标示的颗粒。

表 1-3 RK3399 EMMC Support Symbol

Symbol	Description
√	Fully Tested , Applicable and Mass Production
T/A	Fully Tested , Applicable and Ready for Mass Production
D/A	Datasheet Applicable,Need Sample to Test
N/A	Not Applicable

### 1.2.1 高性能 EMMC 颗粒的选取

为了提高系统性能，选取高性能的 EMMC 颗粒也是需要的。请在挑选 EMMC 颗粒前，参照我们的支持列表的型号，对应的研究下厂商提供的 Datasheet，重点关注下厂商标注的 performance 一章节。

参照厂商大小、读写的速率进行筛选。建议选取顺序读速率>200Mb/s、顺序写速率>40Mb/s。如有选型上的疑问，也可直接联系我们的 Fae 窗口。

6.1.5 Performance

[Table 23] Performance

Density	Partition Type	Performance	
		Read(MB/s)	Write (MB/s)
16GB	General	285	40
32GB		310	70
64GB		310	140
128GB		310	140
16GB	Enhanced	295	80
32GB		320	150
64GB		320	245
128GB		320	245

图 1-1 EMMC Performance 示例

1.3 WiFi/BT 支持列表

RK3399 内核运行 Linux4.4，WiFi/BT 支持列表，详见 RKDocs\Platform support lists 目录下《RK3399\_WiFi\_Situation\_20160808》，下表中所标示为目前 RK3399 上大量测试过的 Wifi/Bt 芯片列表，建议按照列表上的型号进行选型。如果有其他 WiFi/BT 芯片调试，可先与 WiFi/BT 芯片原厂沟通，是否有可以稳定在 Linux4.4 运行的驱动程序，并能提供调试帮助。

另外后续我们会不断更新支持列表，如果疑问和建议可以与我们的 Fae 窗口联系。

RK3399 Wi-Fi Situation										
WiFi Chip	IFACE	IEEE 802.11 Standard	2.4GHz Band	5.0GHz Band	BT	GPS	NFC	SoftA P	P2P①	4.4 kernel ②
AP6354	SDIO	IEEE 802.11A/B/G/N/AC	✓	✓	✓	×	×	✓	✓	✓
Realtek RTL8188EUS	USB	IEEE 802.11B/G/N	✓	×	×	×	×	✓	✓	✓
①. 支持P2P功能，支持WiFi Display。										
②. ✓：表示驱动支持kernel 4.4, 后续验证更多模块后会更新WIFI在kernel 4.4支持列表										

图 1-2 RK3399 目前大量测试的 Wifi/Bt 支持列表

1.4 SDK 软件包适用硬件列表

本 SDK 是基于谷歌 Android6.0 64bit 系统，适配瑞芯微 RK3399 芯片的软件包，适用于 RK3399 Box 开发板、sapphire（蓝宝石）& excavator（挖掘机）开发板及基于其上所有的开发产品。

若参考《RK3399\_BOX\_V02\_20160628》硬件设计的 Box 样机板，软件上面，内核配置可参考 rk3399-box-rev1.dts 进行改动。

若使用的是 sapphire（蓝宝石） & excavator（挖掘机）开发板的，内核配置可直接使用 rk3399-excavator-sapphire-box.dts 进行配置。

另外随 SDK 发布，附带了 Box 样机板，sapphire（蓝宝石） & excavator（挖掘机）开发板的硬件使用说明。



表 1-4 RK3399 硬件说明列表

硬件板	对应文档说明
Box 样机板	RKDocs\Platform support lists\RK3399 硬件使用说明-BOX 样机.pdf
sapphire（蓝宝石） & excavator（挖掘机）开发板	RKDocs\Platform support lists\RK3399 硬件使用说明-蓝宝石&挖掘机.pdf

1.5 多媒体编解码支持列表

RK3399 多媒体方面支持强大，支持 4K VP9 and 4K 10bits H265/H264 视频解码，高达 60fps，1080P 多格式视频解码 (WMV, MPEG-1/2/4, VP8)，1080P 视频编码，支持 H.264，VP8 格式，视频后期处理器：反交错、去噪、边缘/细节/色彩优化。

具体的编解码支持列表，详见 RKDocs\Platform support lists 目录下《RK3399 Multimedia Codec Benchmark v1.0》

# 2 文档/工具索引

## 2.1 文档索引

随 RK3399 Box SDK 发布文档旨在帮助开发者快速上手开发及调试，另由于 RK3399 内核运行版本为 Linux4.4，有些新的知识请大家也多自己补充，文档中涉及的并不能涵盖所有的知识和问题。文档列表也正在不断更新，如有文档上的疑问及需求，请联系我们的 Fae 窗口。

RK3399 SDK 中在 RKDocs 目录下附带了 Develop reference documents(开发指导文档)、Platform support lists（支持列表）、RKTools manuals（工具使用文档）。

```
RKDocs/
├── Develop reference documents
│   ├── RK USB Compliance Test Note V1.2.pdf
│   ├── Rockchip Audio 开发指南 V1.0-20160606.pdf
│   ├── Rockchip DEVFreq 开发指南 V1.0-20160701.pdf
│   ├── Rockchip I2C 开发指南 V1.0-20160629.pdf
│   ├── Rockchip IO-Domain 开发指南 V1.0-20160630.pdf
│   ├── Rockchip Pin-Ctrl 开发指南 V1.0-20160725.pdf
│   ├── Rockchip RK818 电量计 开发指南 V1.0-20160725.pdf
│   ├── Rockchip SDMMC SDIO eMMC 开发指南 V1.0-20160630.pdf
│   ├── Rockchip SPI 开发指南 V1.0-20160629.pdf
│   ├── Rockchip Thermal 开发指南 V1.0-20160701.pdf
│   ├── Rockchip UART 开发指南 V1.0-20160629.pdf
│   ├── Rockchip U-Boot 开发指南 V3.7-20160708.pdf
│   ├── Rockchip USB 开发指南 V1.0-20160704.pdf
│   ├── Rockchip 以太网 开发指南 V2.3.1-20160708.pdf
│   ├── Rockchip 休眠唤醒 开发指南 V0.1-20160729.pdf
│   ├── Rockchip 时钟子模块 开发指南 V1.0-20160630.pdf
│   ├── Rockchip 背光控制 开发指南 V0.1-20160729.pdf
│   └── Rockchip 量产烧录 指南 V1.0-20160718.pdf
├── Platform support lists
│   ├── RK3399 Multimedia Codec Benchmark v1.0.pdf
│   ├── RK3399_WiFi_Situation_20160808.pdf
│   ├── RK3399 硬件使用说明-BOX 样机.pdf
│   ├── RK3399 硬件使用说明-蓝宝石&挖掘机.pdf
│   ├── RK DDR Support List Ver2.18.pdf
│   └── RKeMMCSupportList Ver 1.29_2016_08_05.pdf
└── RKTools manuals
    ├── Android 增加一个分区配置指南 V1.00.pdf
    ├── Android 开发工具手册.pdf
    ├── Recovery 升级相关文档.rar
    ├── RK SDK OTA 包生成方法.pdf
    ├── RK 固件升级失败原因分析_V1.1 20121122.pdf
    ├── RK 平台 apache_tomcat_ota 服务器搭建说明.rar
    └── rk 平台量产升级指导文档 V1.1.pdf
```

- └── Rockchip Parameter File Format Ver1.3.pdf
- └── Rockchip PCBA 模块 开发指南--20160808.pdf
- └── 压力测试 Stresstest 文档\_ver1.1.pdf
- └── 量产工具升级及相关问题处理.pdf

## 2.2 工具索引

随 RK3399 Box SDK 发布工具，用于开发调试阶段及量产阶段使用。工具可能随 SDK 更新不断更新，如有工具上的疑问及需求，请联系我们的 **Fae** 窗口。

RK3399 SDK 中在 RKTools 目录下附带了 linux(Linux 操作系统环境下使用工具)、windows (Windows 操作系统环境下使用工具)。

### RKTools

- └── linux
  - | └── Linux\_Pack\_Firmware (Linux 固件打包工具)
  - | └── Linux\_SecureBoot (Linux 固件签名工具)
  - | └── Linux\_Upgrade\_Tool (Linux 开发工具)
- └── windows
  - └── AndroidTool (开发工具)
    - | └── AndroidTool\_Release\_v2.38
    - | └── rockdev (固件打包工具)
  - └── DriverAssitant\_v4.5 (驱动安装助手)
  - └── Efuse\_Tool\_V1.36 (Efuse 烧写工具)
  - └── FactoryTool-v1.42e.rar (工厂量产工具)
  - └── FWFactoryTool-5.3.rar (固件工厂工具)
  - └── OemTool\_v1.2.rar (Demo 镜像制作工具)
  - └── SD\_Firmware\_Tool\_v1.46.zip (SD 卡升级固件制作工具)
  - └── SecureBootTool\_v1.83\_foruser.rar (固件签名工具)
  - └── SpiImageTools\_v1.36.zip
  - └── UpgradeDllTool\_v1.35.zip (厂商信息烧写工具—待更新版本)

# 3 SDK 编译/烧写

## 3.1 SDK 获取

SDK 通过瑞芯微代码服务器对外发布。客户向瑞芯微技术窗口申请 SDK，需同步提供 SSH 公钥进行服务器认证授权，获得授权后即可同步代码。关于瑞芯微代码服务器 SSH 公钥授权，请参考《RK3399\_ANDROID6.0-BOX-SDK\_V1.00\_20160809 发布说明.pdf》。

### 3.1.1 SDK 下载链接

RK3399\_ANDROID6.0\_BOX\_SDK 下载地址如下：

```
repo init --repo-url ssh://git@www.rockchip.com.cn/repo/rk/tools/repo -u  
ssh://git@www.rockchip.com.cn/gerrit/rk/platform/manifest -b android-6.0 -m  
rk3399_box_marshmallow_release.xml
```

### 3.1.2 repo

repo 是 google 用 Python 脚本写的调用 git 的一个脚本，主要是用来下载、管理 Android 项目的软件仓库，其下载地址如下：

```
git clone ssh://git@www.rockchip.com.cn/repo/rk/tools/repo
```

### 3.1.3 SDK 代码压缩包

为方便客户快速获取 SDK 源码，瑞芯微技术窗口通常会提供对应版本的 SDK 初始压缩包。以 rk3399\_android6.0\_box\_v1.00\_20160809.tgz 为例，拷贝到该初始化包后，通过如下命令可检出源码：

```
mkdir rk3399  
tar xvf rk3399_android6.0_v1.00_20160809.tgz -C rk3399  
cd rk3399  
.repo/repo/repo sync -l  
.repo/repo/repo sync
```

## 3.2 SDK 编译

### 3.2.1 JDK 安装

Android6.0 系统编译依赖于 JAVA 7。编译之前需安装 OpenJDK。

安装命令如下。

```
sudo apt-get install openjdk-7-jdk
```

配置 JAVA 环境变量，例如，安装路径为/usr/lib/jvm/java-7-openjdk-amd64，可在终端执行如下命令配置环境变量。

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64  
export PATH=$JAVA_HOME/bin:$PATH  
export CLASSPATH=.:$JAVA_HOME/lib:$JAVA_HOME/lib/tools.jar
```

### 3.2.2 编译模式

SDK 默认以 userdebug 模式编译。

使用 adb 时，需要先执行 adb root 使 shell 获取 root 权限，进而执行其他像 adb remount 等操作。

### 3.2.3 uboot 编译步骤

```
make rk3399_box_defconfig  
make ARCH=aarch64
```

编译完，会生成 trust.img、RK3399MiniLoaderAll\_V1.05.bin、uboot.img 三个文件。

### 3.2.4 kernel 编译步骤

Box 样机配置与编译如下：

```
make ARCH=arm64 rockchip_defconfig  
make ARCH=arm64 rk3399-box-rev1.img
```

sapphire（蓝宝石） & excavator（挖掘机）开发板配置与编译如下：

```
make ARCH=arm64 rockchip_defconfig  
make ARCH=arm64 rk3399-excavator-sapphire-box.img
```

编译完成后，kernel 根目录，生成 kernel.img，resource.img 两个镜像文件。

### 3.2.5 Android 编译及固件生成步骤

客户按实际编译环境配置好 JDK 环境变量后，按照以下步骤配置完后，执行 make 即可。

```
$ source build/envsetup.sh  
$ lunch
```

选择 rk3399\_box-userdebug

```
$ make -j4
```

完成编译后，执行 SDK 根目录下的 mkimage.sh 脚本生成固件，所有烧写所需的镜像将都会拷贝于 rockdev/Image-rk3399\_box 目录。

```
rockdev/Image-rk3399_box  
├── boot.img  
├── kernel.img  
├── misc.img  
├── parameter.txt  
├── recovery.img  
├── resource.img  
├── RK3399MiniLoaderAll_V1.05.bin  
├── system.img  
├── trust.img  
└── uboot.img
```

## 3.3 固件烧写

刷机说明详见 RKDocs\RKTools manuals 目录下《Android 开发工具手册.pdf》。

SDK 提供烧写工具，如下图所示。编译生成相应的固件后，进入烧写模式，即可进行刷机。对于已烧过其它固件的机器，可以选择重新烧录固件，或是选择低格设备，擦除 idb，然后进行刷机。

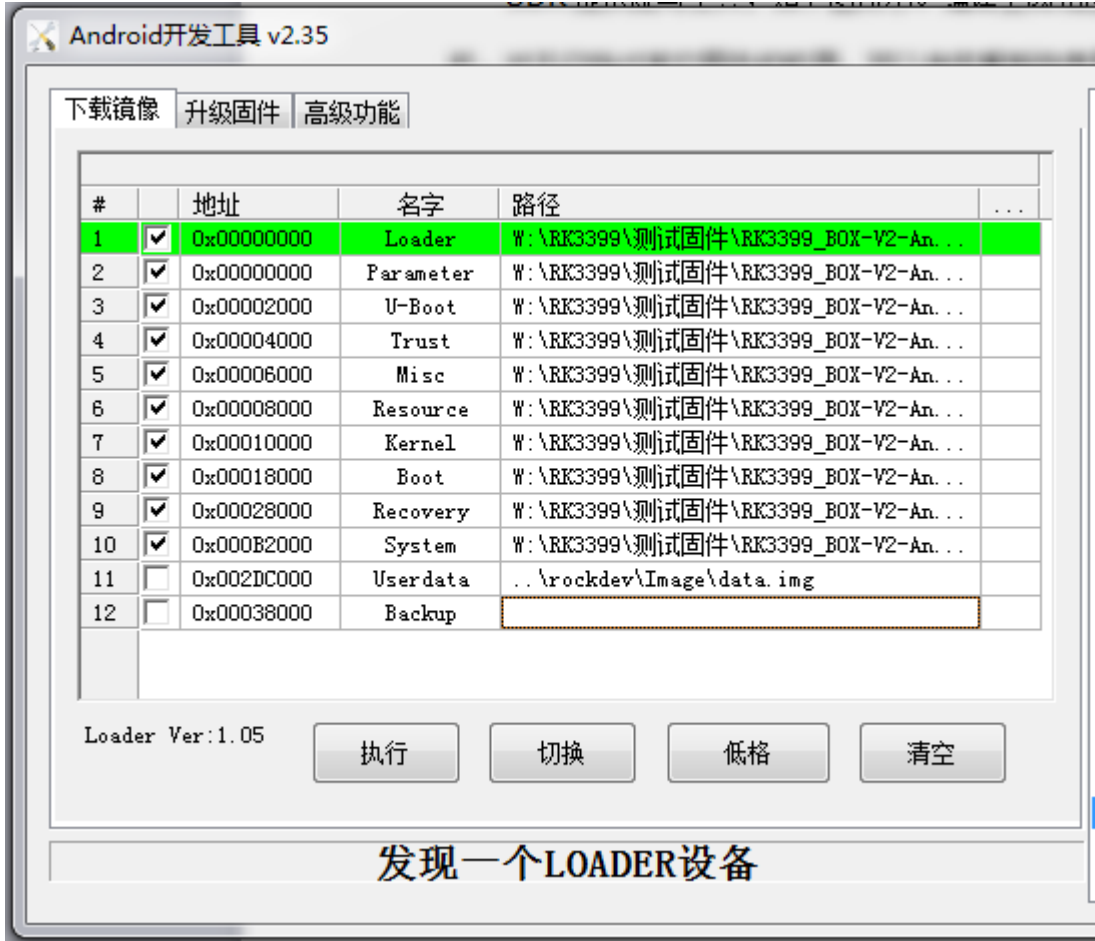


图 3-1Android 开发工具烧写界面

注：烧写前，需安装最新的的 USB 驱动，驱动详见

RKTools/windows/  
|—— DriverAssitant\_v4.5

3.4 量产烧写

量产上考虑到生产效率及工厂工位安排，量产烧写说明详见 RKDocs\Develop reference document 目录下《Rockchip 量产烧录 指南 V1.0-20160718》。

在量产过程中如涉及到工具上的问题，可以联系我们的 Fae 窗口。

# 4 U-Boot 开发

本节简单介绍 U-Boot 基本概念和编译的注意事项，帮助客户了解 RK 平台 U-Boot 框架，具体 U-Boot 开发细节可参考 RKDocs\Develop reference documents 目录下《Rockchip USB 开发指南 V1.0-20160704.pdf》。

## 4.1 Rockchip U-Boot 简介

Rockchip U-Boot 是基于开源的 UBoot 2014.10 正式版进行开发的，主要支持：

- 支持芯片：rk3288、rk3036、rk312x、rk3368、rk322x、rk3366、rk3399 等；
- 支持 Android 平台的固件启动；
- 支持 ROCKUSB 和 Google Fastboot 两种方式烧写；
- 支持 secure boot 固件签名加密保护机制；
- 支持 LVDS、EDP、MIPI、HDMI、CVBS 等显示设备；
- 支持 SDCard、Emmc、Nand Flash、U 盘等存储设备；
- 支持开机 logo 显示、充电动画显示，低电管理、电源管理；
- 支持 I2C、SPI、PMIC、CHARGE、GUAGE、USB、GPIO、PWM、DMA、GMAC、EMMC、NAND 中断等驱动；

## 4.2 平台配置

平台配置文件位于 U-Boot 根目录下的 configs 文件夹下，其中 Rockchip 相关的以 RK 开头，并根据产品形态分为 MID 和 BOX 两种配置：

```
rk3288_defconfig
rk3126_defconfig
rk3128_defconfig
rk3368_defconfig
rk3399_defconfig

rk3288_box_defconfig
rk3128_box_defconfig
rk3036_box_defconfig
rk3368_box_defconfig
rk322x_box_defconfig
rk3399_box_defconfig
```

RK3399 Box 选用的是 rk3399\_box\_defconfig 配置。

## 4.3 固件生成

Rockchip 平台 Loader 分为一级模式和二级模式，根据不同的平台配置生成相应的 Loader 固件。通过宏 CONFIG\_SECOND\_LEVEL\_BOOTLOADER 的定义二级 Loader 模式。

### 4.3.1 一级 Loader 模式

U-BOOT 作为一级 Loader 模式，那么仅支持 EMMC 存储设备，编译完成后生成的镜像：

```
RK3399MiniLoaderAll_V1.05.bin
```

其中 V1.05 是发布的版本号。

### 4.3.2 二级 Loader 模式

U-Boot 作为二级 Loader 模式，那么固件支持所有的存储设备，该模式下，需要 MiniLoader 支持，通过宏 CONFIG\_MERGER\_MINILOADER 进行配置生成。同时引入 Arm Trusted Firmware 后会生成 trust image，这个通过宏 CONFIG\_MERGER\_TRUSTIMAGE 进行配置生成。

以 rk322x 编译生成的镜像为例：

```
RK322XMiniLoaderAll_V2.31.bin
UBOOT.img
trust.img
```

其中 V2.31 是发布的版本号，rockchip 定义 U-Boot loader 的版本，其中 2.31 是根据存储版本定义的，客户务必不要修改这个版本。

UBOOT.img 是 U-Boot 作为二级 loader 的打包。

trust.img 是 U-Boot 作为二级 loader 的打包。

RK3036、RK3126、RK3128、RK322x、RK3368、RK3366、RK3399 等采用二级 loader 模式。

## 4.4 U-Boot 编译

RK3399 Box SDK 编译使用的是如下配置：

```
make rk3399_box_defconfig
make ARCHV=aarch64
```

编译完，会生成 trust.img、RK3399MiniLoaderAll\_V1.05.bin、uboot.img 三个文件。

目前编译出来的 RK3399MiniLoaderAll\_V1.05.bin DDR 为定频 666Mhz 版本，为了调试方便，发布 SDK 的同时，我们也在烧写工具目录下提供了不同频率版本的 Loader。

路径：RKTools\windows\AndroidTool\rockdev

```
DDR 运行 200Mhz: RK3399MiniLoaderAll_V1.05_DDR200MHz.bin
DDR 运行 400Mhz: RK3399MiniLoaderAll_V1.05_DDR400MHz.bin
DDR 运行 666Mhz: RK3399MiniLoaderAll_V1.05_DDR666MHz.bin
DDR 运行 800Mhz: RK3399MiniLoaderAll_V1.05_DDR800MHz.bin
```



# 5 内核常见配置

本节简单介绍内核一些常见配置的修改，主要是 dts 的配置，帮助客户更快更方便的进行一些简单的修改。

## 5.1 WiFi&BT 的配置

### 5.1.1 WiFi 配置

```
wireless-wlan {  
    compatible = "wlan-platdata";  
    rockchip,grf = <&grf>;  
    wifi_chip_type = "ap6354";  
    sdio_vref = <1800>;  
    WIFI,host_wake_irq = <&gpio0 3 GPIO_ACTIVE_HIGH>; /* GPIO0_a3 */  
    status = "okay";  
};
```

上面部分内容是 WiFi 的 dts 配置内容，主要包括电源控制、中断等功能脚的配置。下面将对各个配置项（一般客户只需要修改下面红色标出部分参数）的功能进行详细描述：

**wifi\_chip\_type = " ap6354";**

用来确认 WiFi 芯片型号,实际使用什么型号的 WiFi 需要在这里指定

**sdio\_vref = <1800>; //1800mv or 3300mv**

这个配置项配置 WiFi 模组的 IO 参考电压值，根据实际硬件设计中提供给 WiFi 模组参考电压输入的电压值来进行设定，参考电压设置错误会导致 WiFi 通信异常引起 WiFi 打不开或者工作不稳定。

**WIFI,host\_wake\_irq = <&gpio0 3 GPIO\_ACTIVE\_HIGH>;**

这个配置项是 WiFi 中断脚的配置，某些 WiFi 模组没有这个脚可以不用配置直接将此配置项注释掉。使用 Broadcom 的 WiFi 比如 AP6xxx 以及 RK90x 等模组都需要正确配置这 GPIO。

Broadcom wifi AP6xxx 系统会使用此中断脚作为 WiFi 数据中断脚，此中断脚有异常将会导致 WiFi 无法正常工作。其它 WiFi，例如 RTL8723BS，在机器进入休眠时，如果有 WiFi 数据到来时此中断用来唤醒机器。此中断脚有异常并不会造成 WiFi 无法正常工作。

### 5.1.2 BT 配置

```
wireless-bluetooth {  
    compatible = "bluetooth-platdata";  
    //wifi-bt-power-toggle;  
    uart_rts_gpios = <&gpio2 19 GPIO_ACTIVE_LOW>; /* GPIO2_C3 */  
    pinctrl-names = "default", "rts_gpio";  
    pinctrl-0 = <&uart0_rts>;  
    pinctrl-1 = <&uart0_gpios>;  
    //BT,power_gpio = <&gpio3 19 GPIO_ACTIVE_HIGH>; /* GPIOx_xx */  
    BT,reset_gpio = <&gpio0 9 GPIO_ACTIVE_HIGH>; /* GPIO0_B1 */  
    BT,wake_gpio = <&gpio2 26 GPIO_ACTIVE_HIGH>; /* GPIO2_D2 */
```

```
BT,wake_host_irq = <&gpio0 4 GPIO_ACTIVE_HIGH>; /* GPIO0_A4 */
status = "okay";

};
```

以上是 BT 在 dts 里面的配置，下面对常见可能需要修改的部分进行简单的说明

**BT,reset\_gpio = <&gpio0 9 GPIO\_ACTIVE\_HIGH>;**

这个配置项是关于 BT 的 RESET 脚配置，这个脚不同的 BT 模组不一定都有，具体以实际原理图为准。

**BT,power\_gpio = <&gpio3 19 GPIO\_ACTIVE\_HIGH>**

这个配置项是关于 BT 的电源控制 GPIO 配置，高电平有效，具体以实际原理图为准。

**BT,wake\_gpio = <&gpio2 26 GPIO\_ACTIVE\_HIGH>;**

这个配置项是关于 BT 的 WAKE 脚配置，对应原理图中的 BT\_WAKE 管脚，高电平有效。

**BT,wake\_host\_irq = <&gpio0 4 GPIO\_ACTIVE\_HIGH>**

这个配置项是关于 BT 的中断脚配置，对应原理图中的 BT\_HOST\_WAKE 管脚，高电平有效。

默认 BT 使用 uart0 接口连接，uart0 的配置如下

```
&uart0 {
    pinctrl-names = "default";
    pinctrl-0 = <&uart0_xfer &uart0_cts>;
    status = "okay";
};
```

### 5.1.3 GPIO 对应关系注意

关于原理图上的 **gpio** 跟 **dts** 里面的 **gpio** 的对应关系，例如 GPIO4c0，那么对应的 dts 里面应该是“gpio4 16”。因为 GPIOA 有 8 个 pin，GPIOB 也有 8 个 pin，以此计算可得 c0 口就是 16，c1 口就是 17，以此类推；

## 5.2 U-Boot logo 相关的配置

### 5.2.1 U-Boot logo 开关配置

Sdk 默认开启 U-Boot logo 功能，以达到更快显示开机 logo 的目的

```
rockchip,uboot-logo-on = <1>;
```

如果需要关闭这个功能的，设置 rockchip,uboot-logo-on = <0>;即可

### 5.2.2 U-Boot logo 图片更换

U-boot logo 显示的两张图片是 kernel 根目录下的 logo.bmp 和 logo\_kernel.bmp，如果需要更换，用同名的 bmp 替换掉，重新编译 resource.img 即可。

Ps: 不一定要两张图片，可以只要一张，如果只有一张就保留 logo.bmp 这一张

## 5.3 ARM、GPU 频率修改

参考 RKDocs\Develop reference documents 目录下《Rockchip DEVFreq 开发指南 V1.0-20160701.pdf》

# 6 Android 常见配置

6.1 Android 编译配置

6.2 常用配置说明

6.3 预制 APK

6.4 开/关机动画

6.5 开/关机铃声

6.6 Parameter 说明

6.7 新增分区配置

6.8 系统调试

6.9 OTA 升级

6.10 预制 Demo

6.11 系统灾难恢复

# 7 常用工具说明

7.1 StressTest

7.2 DeviceTest

7.3 PCBA 测试工具

7.4 DDR 测试工具

7.5 update.img 打包

7.6 固件签名

7.7 序列号/Mac/厂商信息烧写

7.8 量产工具使用