

密级状态：绝密() 秘密() 内部() 公开(√)

Rockchip RK3399 软件开发指南

(技术部，第二系统产品部)

文件状态： [] 正在修改 [√] 正式发布	当前版本：	V1.00
	作 者：	吴良清
	完成日期：	2016-09-01
	审 核：	陈海燕、黄祖芳
	完成日期：	2016-09-05

福州瑞芯微电子股份有限公司

Fuzhou Rockchips Semiconductor Co., Ltd

(版本所有, 翻版必究)

版 本 历 史

版本号	作者	修改日期	修改说明	备注
V1.00	吴良清	2016.09.01	正式发布	

目录

概述	1
1 支持列表	2
1.1 DDR 支持列表.....	2
1.2 EMMC 支持列表	2
1.2.1 高性能 EMMC 颗粒的选取	2
1.3 WiFi/BT 支持列表	3
1.4 SDK 软件包适用硬件列表	3
1.5 多媒体编解码支持列表	4
2 文档/工具索引.....	5
2.1 文档索引	5
2.2 工具索引	6
3 SDK 编译/烧写.....	8
3.1 SDK 获取	8
3.1.1 SDK 下载链接	8
3.1.2 SDK 代码压缩包.....	8
3.2 SDK 编译	8
3.2.1 JDK 安装	8
3.2.2 编译模式	8
3.2.3 VR 一体机编译.....	9
3.2.4 VR 分体机编译.....	9
3.2.5 Tablet 编译.....	9
3.2.6 挖掘机编译	10
3.2.7 固件生成步骤	10
3.3 固件烧写	10
3.4 量产烧写	11
4 U-Boot 开发.....	11
4.1 Rockchip U-Boot 简介.....	12
4.2 平台配置	12

4.3	固件生成	13
4.3.1	一级 Loader 模式	13
4.3.2	二级 Loader 模式	13
4.4	U-Boot 编译.....	13
4.5	U-Boot 开启关机充电功能	14
4.6	U-Boot logo 相关的配置.....	15
4.6.1	U-Boot logo 开关配置	15
4.6.2	U-Boot logo 图片更换	15
5	Kernel 开发	15
5.1	DTS 介绍	15
5.2	USB 配置	16
5.3	WiFi 配置	17
5.4	BT 配置	18
5.5	GPIO.....	19
5.6	ARM、GPU 频率修改.....	20
5.7	温控配置	22
6	Android 常见配置	23
6.1	Android 产品配置	23
6.1.1	lunch 选项说明	23
6.1.2	添加一个新的产品	24
6.2	常用功能配置说明	26
6.2.1	常用配置宏说明	26
6.2.2	预装 APK	27
6.2.3	开/关机动画及铃声	27
6.3	Parameter 说明	27
6.4	新增分区配置	28
6.5	OTA 升级.....	28
7	常用工具说明	28
7.1	StressTest.....	28

7.2	DeviceTest.....	28
7.3	PCBA 测试工具	28
7.4	DDR 测试工具.....	28
7.5	update.img 打包.....	28
7.6	固件签名	28
7.7	序列号/Mac/厂商信息烧写	28
7.8	量产工具使用	28

前言

概述

本文档主要介绍 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

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 VR 一体机产品形态、VR 分体机产品形态、Tablet 产品形态、sapphire (蓝宝石) & excavator (挖掘机) 开发板、及其他基于 RK3399 平台开发的产品。

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

参考《RK3399_VR&Tablet_V10_20160620》硬件设计的 VR/TABLET 样机，kernel 配置可以参考：

VR:rk3399-vr-android.dts

TABLET:rk3399-mid-818-android.dts

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。

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

RK3399 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
|   |—— 分体机
|       |——RK3399_VR 分体机_KEY_修改说明文档.pdf
|       |——RK3399_VR 分体机_NANOC_sensor 方向配置.pdf
|       |——RK3399_VR 分体机_NANOC_编译和烧写说明文档.pdf
|       |——RK3399_VR 分体机_NANOC_编译工具安装说明文档.pdf
|       |——RK3399_VR 分体机_显示屏_参数修改说明文档.pdf
|       |——RockChip_DSS Development Guide v1.2.pdf
|       |——
TC358860XBG_eDP-DSI_Tv32p_AUO1080x1200_DPmode_2lane_20160831.xlsm
|   |——
TC358870XBG_HDMI-DSI_Tv28p_HDMI_1440x1280_60fps_DSI0_Audio_Enable.xlsm
|   |—— RK3288 VR 整机测试工具使用说明.pdf
|   |—— RK3399_VR 分体机_软件开发指南.pdf
|   |—— 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
- | |—— 压力测试 Stresstest 文档 forVR_ver3.0.pdf
- |—— Platform support lists
- | |—— RK3399 Multimedia Codec Benchmark v1.0.pdf
- | |—— RK3399_WiFi_Situation_20160808.pdf
- | |—— RK3399 挖掘机蓝宝石用户指南_20160815.pdf
- | |—— RK3399 硬件使用说明-BOX 样机.pdf
- | |—— RK DDR Support List Ver2.18.pdf
- | |—— RKeMMCSupportList Ver 1.29_2016_08_05.pdf
- | |—— RKISPV1_Camera_Module_AVL_v1.3.pdf
- | |—— RKISPV1_Camera_User_Manual_v2.0.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
- |—— Rockchip RK3399VR-TABLET 软件开发指南 V1.00.pdf
- |—— Rockchip RK3399 软件开发指南 V1.00-20160809.pdf

2.2 工具索引

RK3399 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
- └── UpgradeDlIITool_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-VR-TABLET-SDK_V1.00 发布说明.pdf》。

3.1.1 SDK 下载链接

RK3399_ANDROID6.0_SDK 下载地址如下：

```
repo init --repo-url=ssh://git@www.rockchip.com.cn:2222/repo-release/tools/repo.git -u  
ssh://git@www.rockchip.com.cn:2222/rk3399/manifests.git -m Rk3399_Android6.0_sdk.xml
```

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
```

3.1.2 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_VR_Tablet_sdk.tar.gz 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 zxvf Rk3399_Android6.0_VR_Tablet_sdk.tar.gz -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 VR 一体机编译

uboot 编译:

```
cd u-boot
make rk3399_defconfig
make ARCHV=aarch64
```

kernel 编译:

```
cd kernel
make ARCH=arm64 rockchip_defconfig -j8
make ARCH=arm64 rk3399-vr-android.img -j12
```

android 编译:

```
source build/envsetup.sh
lunch rk3399_64_vr-userdebug
make -j12
./mkimages.sh
```

3.2.4 VR 分体机编译

uboot 编译:

```
cd u-boot
make rk3399_defconfig
make ARCHV=aarch64
```

kernel 编译:

由于分体设备只是作为一个热插拔的配件，所以根据主设备（RK3399 端）产品形态不同，可对应选择自己适合的配置，并且做对应的修改，比如这里我们选择 **vr** 一体机的配置：

```
cd kernel
make ARCH=arm64 rockchip_defconfig -j8
make ARCH=arm64 rk3399-vr-android.img -j12
```

然后根据自己的板子硬件修改了 **dts** 文件以后，分体机有几个需要注意的配置事项，请参考文档：
RKDoc/分体机/RK3399_VR 分体机_软件开发指南.pdf

android 编译:

```
source build/envsetup.sh
lunch rk3399_disvr-userdebug
make -j12
./mkimages.sh
```

3.2.5 Tablet 编译

uboot 编译:

```
cd u-boot
make rk3399_defconfig
make ARCHV=aarch64
```

kernel 编译:

```
cd kernel
make ARCH=arm64 rockchip_defconfig -j8
make ARCH=arm64 rk3399-mid-818-android.img -j12
```

android 编译:

```
source build/envsetup.sh
lunch rk3399_mid-userdebug
make -j12
./mkimages.sh
```

3.2.6 挖掘机编译

uboot 编译:

```
cd u-boot
make rk3399_defconfig
make ARCHV=aarch64
```

kernel 编译:

```
cd kernel
make ARCH=arm64 rockchip_defconfig -j8
make ARCH=arm64 rk3399-sapphire-excavator-edp.img -j12
```

android 编译:

```
source build/envsetup.sh
lunch rk3399_64-userdebug
make -j12
./mkimages.sh
```

3.2.7 固件生成步骤

执行./mkimage.sh 后, 在 rockdev/Image-xxx/ 目录生成完整的固件包(xxx 是具体 lunch 的产品名)

```
rockdev/Image-xxx/
├── 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。

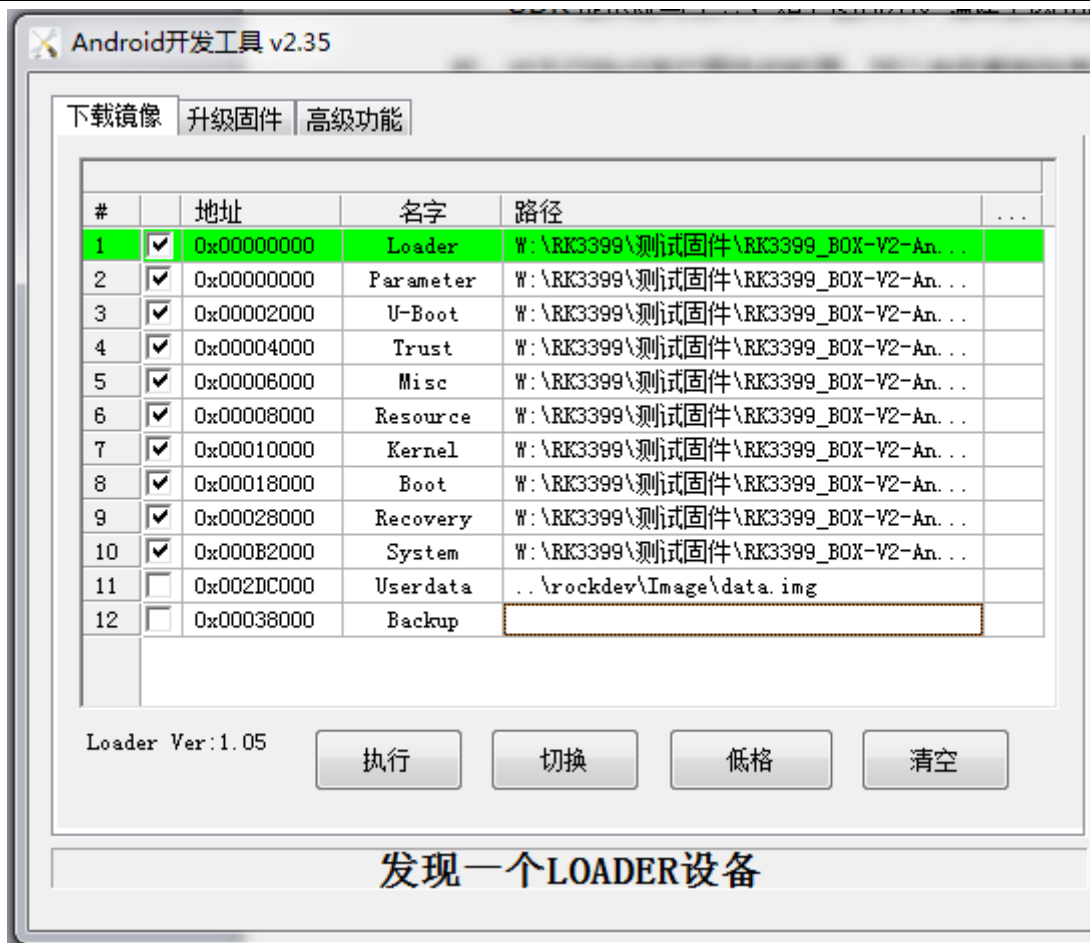


图 3-1Android 开发工具烧写界面

注：烧写前，需安装最新的 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

U-boot 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 VR/Tablet 选用的是 rk3399_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 rk3399 编译生成的镜像为例：

```
RK3399MiniLoaderAll_V1.05.bin  
UBOOT.img  
trust.img
```

其中 V1.05 是发布的版本号，rockchip 定义 U-Boot loader 的版本，其中 1.05 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 SDK 编译使用的是如下配置：

```
make rk3399_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

4.5 U-Boot 开启关机充电功能

为了实现充电动画，需要在 `uboot/include/configs/rk33plat.h` 中打开如下开关,默认该功能是关闭的。

```
#define CONFIG_UBOOT_CHARGE
```

```
#define CONFIG_SCREEN_ON_VOL_THRESD 3400//3.4v
```

```
#define CONFIG_SYSTEM_ON_VOL_THRESD 3500//3.5v
```

其中 `CONFIG_SCREEN_ON_VOL_THRESD` 是系统点亮屏幕的电压门限，低于这个电压，禁止系统亮屏。`CONFIG_SYSTEM_ON_VOL_THRESD` 是系统正常启动的电压门限，低于这个电压，禁止 `uboot` 启动内核。这两个电压可以根据具体的产品设计灵活调整。

产品板级 `dts` 中如下节点进行充电模式开关配置，可以灵活配置使用 `uboot` 还是 `Android` 的关机充电模式：

```
uboot-charge {  
    compatible = "rockchip,uboot-charge";  
    rockchip,uboot-charge-on = <0>;  
    rockchip,android-charge-on = <1>;  
};
```

`rockchip,uboot-charge-on` 开关 `uboot` 阶段的充电动画，`rockchip,android-charge-on` 开关 `android` 充电动画。内核常见配置

如果选择使用 uboot 阶段的充电动画, 即 `rockchip,uboot-charge-on = <1>` 时, 还需要将动画图片资源文件打包在 `resource.img`, 方法如下: `pack_resource.sh` 脚本可以新增资源文件到现有的镜

像: `./pack_resource <resources dir> <old image> <dst image><resource_tool path>` 如在 uboot 根目录下执行:

```
sudo ./tools/resource_tool/pack_resource.sh
```

```
tools/resource_tool/resources/ ../kernel/resource.img resource.img
```

`tools/resource_tool/resource_tool` 将 `tools/resource_tool/resources/` 目录下的动画图片资源打包在 `../kernel/resource.img`, 生成新的 `resource.img` 在 uboot 根目录。

4.6 U-Boot logo 相关的配置

4.6.1 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

4.6.2 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 Kernel 开发

RK3399 kernel 版本是 4.4, config 配置文件统一为 `arch/arm64/configs/rockchip_defconfig`。

5.1 DTS 介绍

RK3399 的 `dts` 文件在 `kernel/arch/arm64/boot/dts/rockchip/` 下, 其中 `rk3399.dtsi` 是核心配置文件定义了平台相关的内容; `RK3399-android.dtsi` 是产品级配置文件定义了一些外围设备; 具体的产品 `dts` 需要 `include` 这两个文件, 如 VR 产品的 `dts` 文件

rk3399-vr-android.dts, Tablet 产品的 dts 文件 rk3399-mid-818-android.dts。产品的 dts 里面根据具体的产品需求配置 CPU、GPU、DDR 的频率和电压表；配置 io、屏、wifi、bt、sensor、温控、背光、电池、系统供电配置等等。

5.2 USB 配置

RK3399 Type-c 模块需要外挂一个 fusb302 逻辑检测芯片来识别接入设备类型及 USB 的正反插。Fusb302 的软件驱动在 dts 里面的配置如下：

```
fusb0: fusb30x@22 {
    compatible = "fairchild,fusb302";
    reg = <0x22>;
    pinctrl-names = "default";
    pinctrl-0 = <&fusb0_int>;
    int-n-gpios = <&gpio1 1 GPIO_ACTIVE_HIGH>;
    status = "okay";
};

中断脚配置:
&pinctrl {
    fusb30x {
        fusb0_int: fusb0-int {
            rockchip,pins = <1 2 RK_FUNC_GPIO &pcfg_pull_up>;
        };
    };
}
```

在 fusb302 及 usb phy 检测区分接入 type-c 口的是哪一类设备（充电器、USB、OTG、DP 等）之后，通知系统，所以相关联的模块代码需要注册 fusb302 的 extcon notifier 来接收，需要在模块 dts 配置加入 extcon = <&fusb0>。

如 rk818 dts 节点中加入 extcon = <&fusb0>, 通过 fusb302 及 usb phy 检测区分充电器、USB、OTG 的拔插后, rk818 模块决定相关的充电电流配置及 OTG 的开关。

目前 sdk 参考 dts 中默认 enable 了 fusb302 的配置, 如果产品未使用 type-c 接口、未使用 fusb302, 产品 dts 中请 disabled 节点 tcphy0 及 fusb0, 并将 USB 相关联的模块 dts 中 extcon = <&fusb0> 改为 extcon = <&u2phy0>。

5.3 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。

5.4 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";  
};
```

5.5 GPIO

RK3399 提供 5 组 GPIO(GPIO0~GPIO4)共 122 个，所有的 GPIO 都可以用作中断，GPIO0/GPIO1 可以作为系统唤醒脚，所有 GPIO 都可以软件配置为上拉或者下拉，所有 GPIO 默认为输入，GPIO 的驱动能力软件可以配置

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；

GPIO 的使用请参考 RKDocs\Develop reference documents 目录下《RKDocs\Develop reference documents 目录下《Rockchip Pin-Ctrl 开发指南 V1.0-20160725.pdf》

5.6 ARM、GPU 频率修改

以下以 VR 一体机的 rk3399-vr-android.dts 为例进行说明。

CPU A53 支持的频率有：

408MHZ/600MHZ/816MHZ/1008MHZ/1200MHZ/1416MHZ/1512MHZ

频率和电压的 dts 配置如下：

```
&cluster0_opp {
    opp@408000000 {
        opp-hz = /bits/ 64 <408000000>;
        opp-microvolt = <800000>;
        clock-latency-ns = <40000>;
    };
    opp@1512000000 {
        opp-hz = /bits/ 64 <1512000000>;
        opp-microvolt = <1100000>;
        status="disabled";
    };
};
```

CPU A72 支持的频率有：

408MHZ/600MHZ/816MHZ/1008MHZ/1200MHZ/1416MHZ/1608MHZ/1800MHZ/1992MHZ

频率和电压的 dts 配置如下：

```
&cluster1_opp {
    opp@408000000 {
        opp-hz = /bits/ 64 <408000000>;
        opp-microvolt = <800000>;
```

```

        clock-latency-ns = <40000>;

};

opp@1992000000 {
    opp-hz = /bits/ 64 <1992000000>;
    opp-microvolt = <1225000>;
};

};

```

GPU 支持的频率有;

200MHZ/300MHZ/400MHZ/500MHZ/600MHZ/800MHZ

频率和电压的 dts 配置如下:

```

&gpu_opp_table {
    compatible = "operating-points-v2";
    opp-shared;
    opp@200000000 {
        opp-hz = /bits/ 64 <200000000>;
        opp-microvolt = <825000>;
    };
    opp@800000000 {
        opp-hz = /bits/ 64 <800000000>;
        opp-microvolt = <1125000>;
    };
};

```

A53/A72/GPU 分别有对于的调试接口，可以通过 ADB 命令进行操作，对于的接口目录如下:

A53: /sys/devices/system/cpu/cpu0/cpufreq/

A72: /sys/devices/system/cpu/cpu4/cpufreq/

GPU: /sys/class/devfreq/ff9a0000.gpu/

这些目录下有如下节点：

- available_frequencies: 显示支持的频率
- available_governors: 显示支持的变频策略
- cur_freq: 显示当前频率
- Governor: 显示当前的变频策略
- max_freq: 显示当前最高能跑的频率
- min_freq: 显示当前最低能跑的频率

以 GPU 为例进行定频操作，流程如下：

- 查看支持哪些频率

```
cat /sys/class/devfreq/ff9a0000.gpu/available_frequencies
```

- 切换变频策略

```
echo userspace > /sys/class/devfreq/ff9a0000.gpu/governor
```

- 定频

```
echo 400000000 > /sys/class/devfreq/ff9a0000.gpu/userspace/set_freq
```

- 设置完后，查看当前频率

```
cat /sys/class/devfreq/ff9a0000.gpu/cur_freq
```

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

5.7 温控配置

RK3399 芯片的 ARM 核和 GPU 核分别带有温控传感器，可以实时监控 cpu 和 gpu 的温度，并通过算法来控制 cpu 和 gpu 的频率从而控制 cpu 和 gpu 的温度。每个产品的硬件设计和模具不同对应的散热情况也不同，可以通过 dts 中的如下配置进行适当的调整温控参数来适配产品：

设置温控开启的温度：

```
&threshold {
```

```
    temperature = <85000>; /* millicelsius */
```

有限公司

```
};
```

设置温控上限温度:

```
&target {
```

```
    temperature = <100000>; /* millicelsius */
```

```
};
```

设置软件关机温度:

```
&soc_crit {
```

```
    temperature = <105000>; /* millicelsius */
```

```
};
```

配置硬件关机温度:

```
&tsadc {
```

```
    rockchip,hw-tshut-mode = <1>; /* tshut mode 0:CRU 1:GPIO */
```

```
    rockchip,hw-tshut-polarity = <1>; /* tshut polarity 0:LOW 1:HIGH */
```

```
    rockchip,hw-tshut-temp = <110000>;
```

```
    status = "okay";
```

```
};
```

温控的具体说明可以参考 RKDocs\Develop reference documents 目录下《Rockchip Thermal 开发指南 V1.0-20160701.pdf》

6 Android 常见配置

6.1 Android 产品配置

6.1.1 lunch 选项说明

rk3399_box-userdebug: //rk3399 平台 box 产品 userdebug (64 位)

rk3399_box-user: //rk3399 平台 box 产品 user (64 位)

rk3399_64-userdebug:	//rk3399 平台 EVB 开发板 userdebug (64 位)
rk3399_64-user:	//rk3399 平台 EVB 开发板 user (64 位)
rk3399_64_vr-userdebug:	//rk3399 平台 VR 产品 userdebug (64 位)
rk3399_64_vr-user:	//rk3399 平台 VR 产品 user (64 位)
rk3399_64_discrete_vr-userdebug:	//rk3399 平台分体机 VR 产品 userdebug (64 位)
rk3399_64_discrete_vr-user:	//rk3399 平台分体机 VR 产品 user (64 位)
rk3399_mid-userdebug:	//rk3399 平台平板产品 userdebug (64 位)
rk3399_mid-user:	//rk3399 平台平板产品 user (64 位)

6.1.2 添加一个新的产品

rk3399 平台支持平板、VR、分体机 VR、Box 等产品形态，当需要添加一个新的产品时，可以基于已有的 rk3399_mid、rk3399_64_vr、rk3399_64_discrete_vr、rk339_box 来建立，如下以建立一个新的平板产品为例进行说明，具体步骤为：

- 产品命令规则：
 - VR** 产品名中需带有“**vr**”字样；
 - Box** 产品名中需带有“**box**”字样；
 - 平板产品名中需带有“**tablet**”字样；
 - 请务必遵守以上规则，否则系统会异常。
- 新增文件夹 device/rockchip/rk3399/rk3399_mid_000，基于 rk3399_mid.mk 创建 rk3399_mid_000.mk，将 rk3399 目录下的所有文件拷贝至 rk3399_mid_000 目录下。


```
cd device/rockchip/rk3399
mkdir rk3399_mid_000
cp rk3399_mid.mk ./rk3399_mid_000.mk
cp rk3399_mid/* rk3399_mid_000/
```
- 在 device/rockchip/rk3399/AndroidProducts.mk 中添加：


```
PRODUCT_MAKEFILES := \
    $(LOCAL_DIR)/rk3399.mk \
```

```
$(LOCAL_DIR)/rk3399_64.mk \  
$(LOCAL_DIR)/rk3399_64_vr.mk \  
$(LOCAL_DIR)/rk3399_64_discrete_vr.mk \  
$(LOCAL_DIR)/rk3399_32.mk \  
$(LOCAL_DIR)/rk3399_box.mk \  
$(LOCAL_DIR)/rk3399_mid.mk \  
$(LOCAL_DIR)/rk3399_mid_000.mk \  
$(LOCAL_DIR)/rk3399_mid_.mk \  
$(LOCAL_DIR)/rk3399_64_only.mk
```

- 在 vendorsetup.sh 中添加产品对应的 lunch 选项:

```
add_lunch_combo rk3399-userdebug  
add_lunch_combo rk3399_32-userdebug  
add_lunch_combo rk3399_box-userdebug  
add_lunch_combo rk3399_box-user  
add_lunch_combo rk3399_64-userdebug  
add_lunch_combo rk3399_64-user  
add_lunch_combo rk3399_64_vr-userdebug  
add_lunch_combo rk3399_64_vr-user  
add_lunch_combo rk3399_64_discrete_vr-userdebug  
add_lunch_combo rk3399_64_discrete_vr-user  
add_lunch_combo rk3399_mid-userdebug  
add_lunch_combo rk3399_mid-user  
add_lunch_combo rk3399_mid_000-userdebug  
add_lunch_combo rk3399_mid_000-user
```

- 修改 rk3399_mid_000.mk 及 rk3399_mid_000 目录下的新产品所需要修改的配置。

6.2 常用功能配置说明

6.2.1 常用配置宏说明

宏配置	功能说明
BUILD_WITH_GOOGLE_MARKET	若为 true 则集成 GMS 包，false 不集成
BUILD_WITH_GOOGLE_MARKET_ALL	若为 true 集成 full 的 GMS 包，false 集成 mini 的 GMS 包
BUILD_WITH_GOOGLE_FRP	使能恢复出厂设置保护 FRP 功能
BUILD_WITH_FORCEENCRYPT	使能默认全盘加密
BUILD_WITH_GMS_CER	GMS 认证配置选项
BUILD_WITH_WIDEVINE	集成 Widevine level3 插件库
BOARD_NFC_SUPPORT	使能 NFC 功能
BOARD_SENSOR_ST	选用 ST 的 sensor 框架
BOARD_SENSOR_MPU	选用 MPU 的 sensor 框架
BOARD_SENSOR_MPU_VR	选用 MPU_VR 的 sensor 框架
BOARD_GRAVITY_SENSOR_SUPPORT	使能 G-Sensor
BOARD_COMPASS_SENSOR_SUPPORT	使能 Compass
BOARD_GYROSCOPE_SENSOR_SUPPORT	使能陀螺仪 Gyroscope
BOARD_PROXIMITY_SENSOR_SUPPORT	使能距离感应器
BOARD_LIGHT_SENSOR_SUPPORT	使能光感应器
BOARD_PRESSURE_SENSOR_SUPPORT	使能压力感应器
BOARD_TEMPERATURE_SENSOR_SUPPORT	使能温度传感器
BOARD_ENABLE_3G_DONGLE	使能 3G Dongle 功能
TARGET_ROCKCHIP_PCBATEST	使能 PCBA 测试
BOOT_SHUTDOWN_ANIMATION_RINGING	使能开关机动画+铃声
BOARD_SYSTEMIMAGE_PARTITION_SIZE	System 分区最大容量

6.2.2 预装 APK

Android 上的应用预安装功能，主要是指配置产品时，根据厂商要求，将事先准备好的第三方应用预制进 Android 系统。预安装分为可卸载预安装和不可卸载预安装，本文主要阐述的是**可卸载预安装**的功能。配置步骤如下：

- 新增文件夹 device/rockchip/rk3399/preinstall
- 拷贝需要预制的第三方应用到上述文件夹，注意 apk 文件名尽量使用英文，避免空格。

编译结束后会在 out/target/product/rk3399_mid/system/ 目录，生成 preinstall 文件夹，文件夹内包含了预制的第三方应用。烧录后，系统会自动安装这些应用到 data/app 目录。因此他们是可卸载的。需要注意的是，在 preinstall 目录中的应用，即使用户在使用过程中将其卸载，但在恢复出厂设置后，应用又会自动安装。如果希望恢复出厂设置后不再恢复预安装应用，可以将上述文件夹名字改为 preinstall_del 即可实现。

6.2.3 开/关机动画及铃声

需要在产品的 makefile 中配置 BOOT_SHUTDOWN_ANIMATION_RINGING := true，并且准备如下相应资源文件，编译结束后对应的资源文件会拷贝到相应的 out 目录下。

将开机铃声 复制到 device/rockchip/common/startup.wav (源码路径)

将关机铃声 复制到 device/rockchip/common/startup.wav (源码路径)

将开机动画 复制到 device/rockchip/common/bootanimation.zip (源码路径)

将关机动画 复制到 device/rockchip/common/shutdownanimation.zip (源码路径)

6.3 Parameter 说明

rk3399 平台有平板、VR、分体机、Box 等产品形态，不同的产品形态可能需要不同的 parameter 参数，请参考 device/rockchip/rk3399/ 下子目录 rk3399_mid、rk3399_box、rk3399_64_vr 中的来相应修改配置，关于 parameter 中各个参数、分区情况细节，请参考 \RKDocs\RKTools manuals\ Rockchip Parameter File Format Ver1.3.pdf。

6.4 新增分区配置

请参考\RKDocs\RKTools manuals\Android 增加一个分区配置指南 V1.00.pdf

6.5 OTA 升级

请参考\RKDocs\RKTools manuals\ RK SDK OTA 包生成方法.pdf

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