

Rockchip 分区介绍

ID: RK-SM-YF-015

发布版本: V1.4.1

日期: 2020-02-21

文件密级: ☐绝密 ☐秘密 ☐内部资料 ☒公开

免责声明

本文档按“现状”提供, 福州瑞芯微电子股份有限公司(“本公司”, 下同)不对本文档的任何陈述、信息和内容的准确性、可靠性、完整性、适销性、特定目的性和非侵权性提供任何明示或暗示的声明或保证。本文档仅作为使用指导的参考。

由于产品版本升级或其他原因, 本文档将可能在未经任何通知的情况下, 不定期进行更新或修改。

商标声明

“Rockchip”、“瑞芯微”、“瑞芯”均为本公司的注册商标, 归本公司所有。

本文档可能提及的其他所有注册商标或商标, 由其各自拥有者所有。

版权所有 © 2019 福州瑞芯微电子股份有限公司

超越合理使用范畴, 非经本公司书面许可, 任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部, 并不得以任何形式传播。

福州瑞芯微电子股份有限公司

Fuzhou Rockchip Electronics Co., Ltd.

地址: 福建省福州市铜盘路软件园A区18号

网址: www.rock-chips.com

客户服务电话: +86-4007-700-590

客户服务传真: +86-591-83951833

客户服务邮箱: fae@rock-chips.com

前言

概述

Rockchip android系统平台使用parameter文件来配置一些系统参数, 比如固件版本, 存储器分区信息等。

Parameter文件是非常重要的系统配置文件, 最好在能了解清楚各个配置功能时再做修改, 避免出现parameter文件配置异常造成系统不能正常工作的问题。

Parameter文件大小有限制, 最大不能超过64KB。

产品版本

芯片名称	SDK版本
全系列	全部版本

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

版本号	作者	修改日期	修改说明
V1.0.0	赵仪峰	2011-04-11	parameter文件说明
V1.1.0	赵仪峰	2011-09-05	完善功能文档
V1.2.0	赵仪峰	2012-10-16	增加RK30和RK292X配置
V1.3.0	赵仪峰	2013-04-15	增加gpio控制定义配置等
V1.4.0	赵仪峰	2018-01-23	删除过时内容，增加GPT等
V1.4.1	赵仪峰	2020-02-21	修改格式

Rockchip 分区介绍

1. Parameter文件预览
2. 文件内容说明
 - 2.1 FIRMWARE_VER:9.0
 - 2.2 MACHINE_MODEL:RK3326
 - 2.3 MACHINE_ID:007
 - 2.4 MANUFACTURER:rk3326
 - 2.5 MAGIC:0x5041524B
 - 2.6 ATAG:0x60000800
 - 2.7 MACHINE:3226
 - 2.8 CHECK_MASK:0x80
 - 2.9 TYPE:GPT
 - 2.10 CMDLINE:

1. Parameter文件预览

Parameter文件主要用于定义分区表，可以支持两种分区格式：一个是GPT格式，另一个是传统CMDLINE分区。不同的项目，不同的平台，parameter文件的内容会有一些差异。本文档以RK3326平台的parameter文件进行举例说明。

RK3326 GPT分区格式parameter定义：

```
1  FIRMWARE_VER:9.0
2  MACHINE_MODEL:RK3326
3  MACHINE_ID:007
4  MANUFACTURER: RK3326
5  MAGIC: 0x5041524B
6  ATAG: 0x00200800
7  MACHINE: 3326
8  CHECK_MASK: 0x80
9  PWR_HLD: 0,0,A,0,1
10 TYPE: GPT
11 CMDLINE:mtddparts=rk29xxnand:0x00002000@0x00004000 (uboot),0x00002000@0x00006000 (trust),0x00002000@0x00008000 (misc),0x00002000@0x0000a000 (dtb),0x00002000@0x0000c000 (dtbo),0x00000800@0x0000e000 (vbmeta),0x00010000@0x0000e800 (boot),0x00030000@0x0001e800 (recovery),0x00028000@0x0004e800 (backup),0x00002000@0x00076800 (security),0x00070000@0x00078800 (cache),0x002d0000@0x000e8800 (system),0x00008000@0x003b8800 (metadata),0x00070000@0x003c0800 (vendor),0x00020000@0x00430800 (oem),0x00000400@0x00450800 (frp),-@0x00450c00 (userdata:grow)
12 uuid:system=af01642c-9b84-11e8-9b2a-234eb5e198a0
```

RK3326 传统分区格式（CMDLINE）parameter定义：

```
1  FIRMWARE_VER:9.0
2  MACHINE_MODEL:RK3326
3  MACHINE_ID:007
4  MANUFACTURER: RK3326
5  MAGIC: 0x5041524B
6  ATAG: 0x00200800
7  MACHINE: 3326
8  CHECK_MASK: 0x80
9  PWR_HLD: 0,0,A,0,1
10 CMDLINE:console=ttyFIQ0 androidboot.console=ttyFIQ0
    initrd=0x62000000,0x00800000
    mtdparts=rk29xxnand:0x00002000@0x00004000 (uboot),0x00002000@0x00006000 (trust),0x00002000@0x00008000 (misc),0x00002000@0x0000a000 (dtb),0x00002000@0x0000c000 (dtbo),0x00000800@0x0000e000 (vbmeta),0x00010000@0x0000e800 (boot),0x00030000@0x0001e800 (recovery),0x00028000@0x0004e800 (backup),0x00002000@0x00076800 (security),0x00070000@0x00078800 (cache),0x002d0000@0x000e8800 (system),0x00008000@0x003b8800 (metadata),0x00070000@0x003c0800 (vendor),0x00020000@0x00430800 (oem),0x00000400@0x00450800 (frp),-@0x00450c00 (userdata)
```

GPT分区相对传统分区，parameter文件主要差异：

- 定义 **TYPE: GPT**
- 最后一个分区增加grow标识，如：**userdata:grow**
- 指定system或者rootfs的uuid，如：**uuid:system=af01642c-9b84-11e8-9b2a-234eb5e198a0**
- parameter文件不会烧录到NVM（EMMC，NAND等）里面，会使用到的信息只有mtdparts的分区定义和UUID，其他信息只是为了兼容升级工具而定义。

2. 文件内容说明

2.1 FIRMWARE_VER:9.0

条目	FIRMWARE_VER
类型	十进制数，格式: X.X
数值	0 - 255
描述	打包update.img时会使用到，升级工具会根据这个识别固件版本号。

2.2 MACHINE_MODEL:RK3326

条目	MACHINE_MODEL
类型	字符串
长度	255（最大）
描述	机器型号，打包 update.img 使用，不同的项目，可以修改，用于升级工具显示。 在 recovery 里面升级固件时可以用于判断固件和机器是否匹配。

2.3 MACHINE_ID:007

条目	MACHINE_ID
类型	字符串
长度	255（最大）
描述	产品开发ID，可以为字符和数字组合，打包update.img使用，不同的项目使用不同的ID，可以用于识别机器机型。 在 recovery 里面升级固件时可以用于判断固件是否匹配。

2.4 MANUFACTURER:rk3326

条目	MANUFACTURER
类型	字符串
长度	255（最大）
描述	厂商信息，打包 update.img 使用，可以修改，用于升级工具显示。

2.5 MAGIC:0x5041524B

条目	MAGIC
类型	十六进制数
数值	0x5041524B（固定）
描述	魔数MAGIC，不能修改， 一些新的 AP 使用 DTS，这一项没有用，为了兼容，不要删除或修改。

2.6 ATAG:0x60000800

条目	ATAG
类型	十六进制数
数值	32bits DDR地址
描述	ATAG DDR存放地址， 一些新的AP使用DTS，这一项没有用，为了兼容，不要删除或修改。

2.7 MACHINE:3226

条目	MACHINE
类型	字符串
长度	255（最大）
描述	内核识别用，不能修改，这个定义和内核匹配。

下表列出几个平台的值：

芯片	MACHINE
RK29xx	2929
RK292X	2928
RK3066	3066
RK3126C	3126c
RK3326	3326
RK3399	3399
RK3308	3308

2.8 CHECK_MASK:0x80

条目	CHECK_MASK
类型	十六进制数
数值	0x80(固定)
描述	保留，不能修改。

2.9 TYPE:GPT

GPT分区表标识，烧录工具会根据 CMDLINE 里面定义的分区创建GPT分区表，paramter文件不会烧录到 NVM（NAND，EMMC等）存储器件里面。

2.10 CMDLINE:

`console=ttyFIQ0 androidboot.console=ttyFIQ0`，串口定义。

`initrd=0x62000000,0x00800000`，第一个参数是 RAMDISK 加载到 DDR 的位置，第二个参数为 RAMDISK 的大小，目前 RAMDISK 大小没有限制。

androidboot.xxx 的定义在 android 启动时使用，有些平台会在 kernel 的 dts 里面定义，这部分定义一般不用修改，只需用发布 SDK 默认的了。

MTD 分区定义说明：

```
1 | mtdparts=rk29xxnand:0x00002000@0x00002000 (uboot),0x00002000@0x00004000 (trust)
  | ,0x00002000@0x00006000 (misc),
2 | 0x00008000@0x00008000 (resource),0x00010000@0x00010000 (kernel),0x00010000@0x00
  | 020000 (boot),0x00020000@0x00030000 (recovery),
3 | 0x00038000@0x00050000 (backup),0x00002000@0x00088000 (security),0x00100000@0x00
  | 08a000 (cache),0x00400000@0x0018a000 (system),
4 | 0x00008000@0x0058a000 (metadata),0x00080000@0x00592000 (vendor),0x00080000@0x00
  | 612000 (oem),0x00000400@0x00692000 (frp),-@0x00692400 (userdata)
```

分区定义说明：

1、为了兼容性，目前所有 AP 都是用 rk29xxnand 做标识。

2、单个分区说明：

例如：`0x00002000@0x00008000(boot)`，@符号之前的数值是分区大小，@符号之后的数值是分区的起始位置，括号里面的字符是分区的名字。所有数值的单位是 sector，1个 sector 为512Bytes.上例中，boot 分区起始位置为0x8000 sectors位置，大小为0x2000 sectors(4MB)。

3、为了性能，每个分区起始地址需要32KB（64 sectors）对齐，大小也需要32KB的整数倍。

4、如果使用 sparse 格式的镜像，升级时会擦除数据，为了兼容性更好，对应的分区最好按4MB对齐，大小也按4MB整数倍配置。

5、使用GPT分区时，parameter 里面定义的地址，都是真实的逻辑地址（LBA），例如 uboot 定义在 0x4000，那么烧录到 EMMC 和 NAND 里面时，逻辑地址也是0x4000。

名称	Parameter 定义地址	EMMC 逻辑地址	NAND 逻辑地址	大小
GPT	--	0	0	32KB
LOADER	--	0x40	0x40	4MB-32KB
保留	--	0x2000	0x2000	4MB

名称	Parameter 定义地址	EMMC 逻辑地址	NAND 逻辑地址	大小
UBOOT	0x4000	0x4000	0x4000	4MB
TRUST	0x6000	0x6000	0x6000	4MB

最后一个分区需要指定 grow 参数，工具会把剩余的空间都分配给最后一个分区。

6、使用传统 cmdline 分区时，如果是EMMC颗粒，0-4MB的空间是保留存放loader的，parameter 里面定义的分区都需要加上4MB，例如 uboot 定义在0x2000，实际烧录到 EMMC 里面时，和使用 GPT 分区时烧录的逻辑地址是一样的，也是0x4000。如果是NAND颗粒，为了和原来产品兼容，所有地址都是真实逻辑地址，例如uboot定义在0x2000，实际烧录到NAND里面是，逻辑地址也是0x2000，和使用GPT时不一样。

名称	Parameter 定义地址	EMMC 逻辑地址	NAND 逻辑地址	大小
保留	--	0	0	32KB
LOADER	--	0x40	0x40	4MB-32KB
parameter	--	0x2000	0x0	4MB
UBOOT	0x2000	0x4000	0x2000	4MB
TRUST	0x4000	0x6000	0x4000	4MB

注：NAND FLASH 的机器，0x40有可能会写 loader 的镜像，和 parameter 在同一个4MB空间内，有效的数据是相互错开存放的，不会覆盖。