

Rockchip Parameter File Format

Ver 1.3

2013/4/15

版本	日期	描述	作者	审核
Ver 1.0	2011-4-11	parameter 文件说明	赵仪峰	
Ver 1.1	2011-9-5	完善功能文档	赵仪峰	
Ver 1.2	2012-10-16	增加 RK30 和 RK292X 配置	赵仪峰	
Ver 1.3	2013-4-15	增加 gpio 控制定义配置等	赵仪峰	

概述

Rockchip android 系统平台使用 parameter 文件来配置一些系统参数，比如定义串口号，固件版本，nand flash 分区信息等等。

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。

由于 Parameter 的参数是由 BootLoader 解析的，所以本文列出来的参数不一定适用于旧版本的 boot loader。

在文档发布时，各个平台的 loader 版本如下：

RK29xx: RK29xxLoader(L)_V2.38.bin

RK30xx: RK30xxLoader(L)_V1.30.bin

RK292x: RK292xLoader(L)_V1.30.bin

RK3168: RK3168Loader(L)_V1.14.bin

RK3188: RK3188Loader(L)_V1.14.bin

parameter文件预览:

下面是一个 parameter 文件的内容，不同的项目，不同的平台，parameter 文件的内容会有一些差异。下例是 RK30 平台的 parameter 文件。

```

1 FIRMWARE_VER:4.0.4
2 MACHINE_MODEL:rk30sdk
3 MACHINE_ID:007
4 MANUFACTURER:RK30SDK
5 MAGIC: 0x5041524B
6 ATAG: 0x60000800
7 MACHINE: 3066
8 CHECK_MASK: 0x80
9 KERNEL_IMG: 0x60408000
10 COMBINATION_KEY: 0,4,C,5,0
11 COMBINATION_KEY: 1,1,8,20,0
12 CMDLINE:console=ttyFIQ0 androidboot.console=ttyFIQ0 init=/init in
. itrd=0x62000000,0x00800000 mtdparts=rk29xxnand:0x00002000@0x00002
. 000(misc),0x00004000@0x00004000(kernel),0x00008000@0x00008000(boo
. t),0x00008000@0x00010000(recovery),0x000C0000@0x00018000(backup),
. 0x00040000@0x000D8000(cache),0x00100000@0x00118000(userdata),0x00
. 002000@0x00218000(kpanic),0x000C0000@0x0021A000(system),-@0x002DA
. 000(user)
13
  
```

文件内容说明

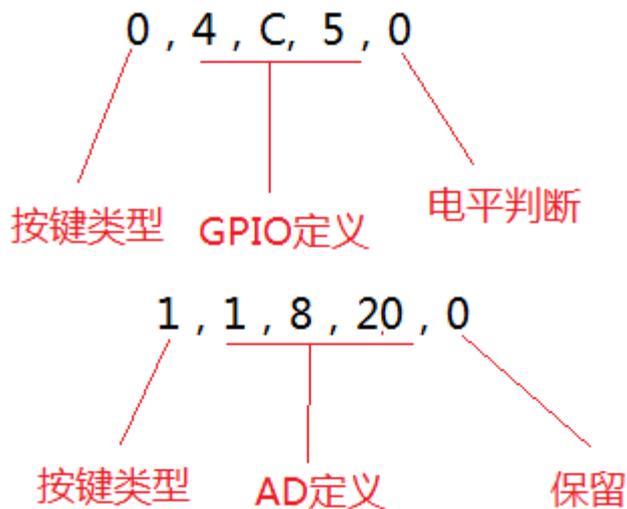
1. FIRMWARE_VER:4.0.4
固件版本，打包 updata.img 时会使用到，升级工具会根据这个识别固件版本。Boot loader 会读取这个版本传递给 kernel 使用。
2. MACHINE_MODEL:rk30sdk
机型，打包 updata.img 使用，不同的项目，可以自己修改，用于升级工具显示。
3. MACHINE_ID:007
产品开发 ID，可以为字符和数字组合，打包 updata.img 使用，不同的项目使用不同的 ID，可以用于识别机器机型。
4. MANUFACTURER:RK30SDK
厂商信息，打包 updata.img 使用，可以自己修改，用于升级工具显示。
5. MAGIC: 0x5041524B
MAGIC，不能修改
6. ATAG: 0x60000800
ATAG，不能修改

7. MACHINE: 3066
内核识别用，不能修改。
RK29xx 识别码：MACHINE: 2929
RK292x 识别码：MACHINE: 2928

8. CHECK_MASK: 0x80
保留，不能修改。

9. KERNEL_IMG: 0x60408000
内核地址，boot loader 会将内核加载到这个地址，kernel 编译地址改变时，这个值需要相应修改。

- 10 COMBINATION_KEY: 0,4,C,5,0
按键定义说明：



按键类型为：0 = 普通按键 ,1=AD 按键

GPIO 定义：上例中定义的是 GPIO 4 C5

判断电平：0= 低电平 ， 1=高电平

AD 定义（通道，下限值，上限值）：上例中，**1** 表示 ADC 通道 1，**8** 表示下限值为 80，**20** 表示上限值为 200，也就是 AD 值在 80~200 内的按键都认为是 COMBINATION_KEY。

Combination 按键定义，可以定义多个，用户可以根据实际机型定义按键。

功能说明：

- 1、按住 recovery 按键并接 USB 开机，进 loader rockusb 升级模式。
- 2、按住 recovery 按键不接 USB 开机，3S 左右会引导 recovery.img。
- 3、按住 combination 按键开机，会引导 recovery.img，进 android 的 recovery 模式，用户可以根据菜单选择操作。

11. CMDLINE:
console=ttyFIQ0 androidboot.console=ttyFIQ0，串口定义。

`initrd=0x62000000,0x00800000`，第一个参数是 `boot.img` 加载到 `sdram` 的位置，第二个参数为 `ramdisk` 的大小，目前 `ramdisk` 大小没有限制。

MTD 分区：

```
mtdparts=rk29xxnand:0x00002000@0x00002000(misc),0x00004000@0x00004000(kernel),0x00002000@0x00008000(boot),0x00004000@0x0000A000(recovery),0x00080000@0x0000E000(system),0x00082000@0x0008E000(backup),0x0003a000@0x00110000(cache),0x00100000@0x0014a000(userdata),0x00002000@0x0024a000(kpanic),-@0x0024c000(user)
```

分区定义说明：

为了兼容性，RK30xx、RK29xx 和 RK292x 都是用 `rk29xxnand` 做标识。

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)。

目前 flash 最大的 block 是 4MB (0x2000 sectors)，所以每个分区需要 4MB 对齐，也就是每个分区都必须为 4MB 的整数倍大小。

Backup 及之前的分区为固件区，在 linux 系统里是只读的，在确定大小后，后续升级时不能修改分区大小。

Backup 之后的分区在 linux 系统里是可读写的。在后续升级时可以调整分区大小，调整分区大小后需要进 android 的 recovery 系统格式化 cache 等分区。

12. GPIO 控制信息

例如：

```
PWR_HLD:0,0,C,7,1 //控制 GPIO0C7 输出高电平
PWR_HLD:0,0,C,7,2 //控制 GPIO0C7 输出低电平
PWR_HLD:0,0,A,0,3 //配置 PWR_HLD 为 GPIO0A0,在 Loader 需要锁定电源时，输出高电平
锁定电源
```

GPIO 控制信息和 PWR_HLD 管脚定义信息复用，因此定义 GPIO 控制信息后，最后一条需要配置真正的 PWR_HLD 引脚，不然 loader 不会锁定电源。

代码不会配置 IOMUX，被配置为特殊功能的 IO 是不可以控制的，这些 IO 有：SD0, SD1, NAND FLASH（不包含 D8-D15），SPI0，UART2 等使用到的 IO。

定义格式和 COMBINATION_KEY 的 GPIO 定义类似，只有最后一个字节（电平判断）不同。

电平判断解释：

- 1: = 解析 parameter 时，输出高电平
- 2: = 解析 parameter 时，输出低电平
- 3: = 在 Loader 需要控制电源时，输出高电平
- 0: = 在 Loader 需要控制电源时，输出低电平

常见问题和解决办法

1、system 分区修改为 ext3 格式时，parameter 的 mtd 分区怎么定义？

Ext3 为可写文件系统，那么 system 分区需要定义在 backup 分区之后。

2、怎么增加一个 mtd 分区？

在 parameter MTD 分区中按住分区定义格式增加分区，如果在系统中需要可写的，那么要把分区定义在 backup 分区之后，只读的，那么分区定义在 backup 分区之前。

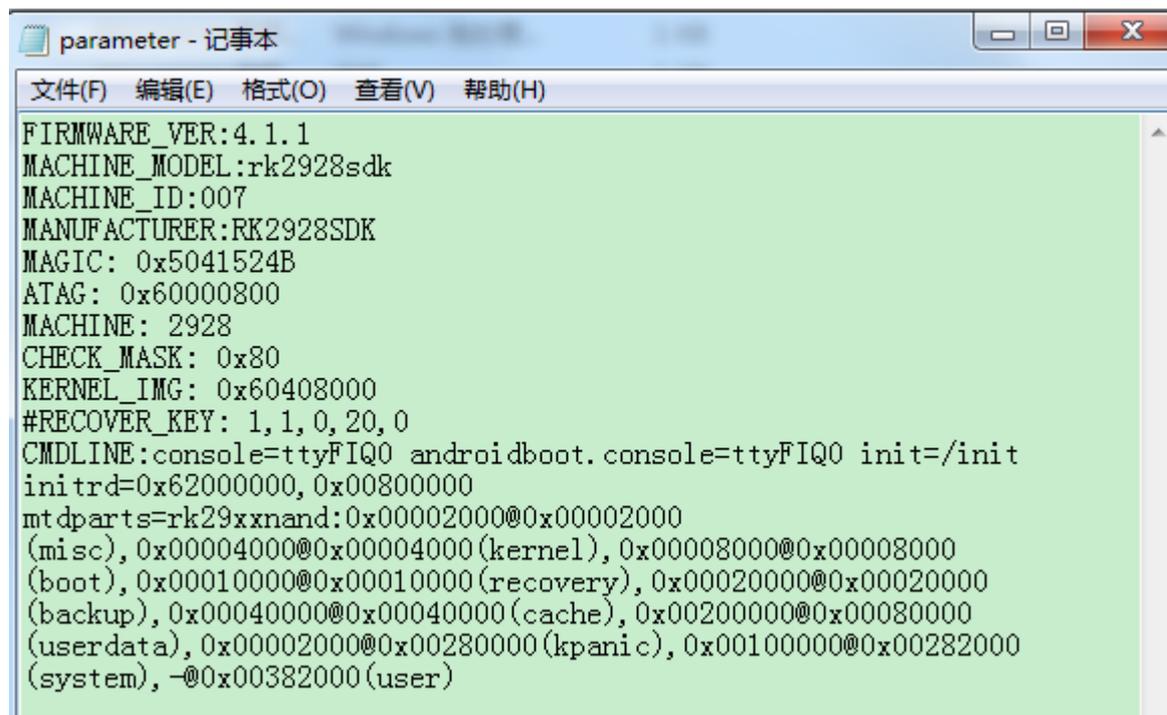
3、由于系统固件变大，backup 分区起始位置和大小变大后整个系统不正常怎么处理？

Backup 分区之前的区域为只读区域，为保证系统固件可靠性和稳定性，这个区域有做特殊处理，在第一次升级固件后就不可以把分区改大（改小不会有问题），所以在开发阶段定义分区大小时尽量预留足够空间。

出现问题的机器，需要按住 recovery 按键进 loader rockusb 升级模式，等待 10 秒后，用量产工具或者用户工具的修复模式升级固件，也可以用开发工具的擦除 idb 功能低格 flash 后再升级固件。

针对这个问题，目前新发布的 SDK 使用新的解决方案，buckup 分区不再备份 system.img，当 system.img 分区变大时，就不会有这个问题。

下例是 2928 的定义：



```
parameter - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
FIRMWARE_VER:4.1.1
MACHINE_MODEL:rk2928sdk
MACHINE_ID:007
MANUFACTURER:RK2928SDK
MAGIC: 0x5041524B
ATAG: 0x60000800
MACHINE: 2928
CHECK_MASK: 0x80
KERNEL_IMG: 0x60408000
#RECOVER_KEY: 1, 1, 0, 20, 0
CMDLINE:console=ttyFIQO androidboot.console=ttyFIQO init=/init
initrd=0x62000000,0x00800000
mtdparts=rk29xxnand:0x00002000@0x00002000
(misc), 0x00004000@0x00004000(kernel), 0x00008000@0x00008000
(boot), 0x00010000@0x00010000(recovery), 0x00020000@0x00020000
(backup), 0x00040000@0x00040000(cache), 0x00200000@0x00080000
(userdata), 0x00002000@0x00280000(kpanic), 0x00100000@0x00282000
(system), -@0x00382000(user)
```

Recovery 是 32MB，backup 分区是 64MB，data 分区是 1GB，system 分区是 512MB。