

# A83T

## A83T NAND 分区分配说明书



## 1. 概述

### 1.1. 编写目的

本文档用于帮助相关人员理清 A83T 平台 NAND 分区的实际占用情况, 及如何根据实际需要提高实际可用容量;

### 1.2. 适用范围

本说明文档适用于 A83T 平台, 同样适用于采用 NAND2.0 架构的其他全志平台;

Confidential

## 2. NAND 空间占用分析

### 2.1. 计算公式

用户可使用分区容量(即设置->存储->内存设备->总容量) = nand 标称总容量 - 各个分区容量和 - nand 驱动保留容量(包含 nand 管理容量, nand 物理坏块及物理数据占用容量等)

nand 驱动管理容量占用的计算方法如下:

(1) UDISK (即用户可使用的挂载分区) 作为裸机管理需要使用其约七分之一的空间; 如用户看到挂载到电脑的 UDISK 盘总容量为 4.12GB, 则其额外占用了  $4.12 \times 1024 / 7 = 603\text{MB}$

(2) NAND 为了提高 IO 性能, 对 UDISK 分区之外的分区额外分出一部分空间作为逻辑管理, 空间大约占用空间的四分之一; 如在 sys\_partition.fex 文件中各分区容量 (boot-res/env/boot/system/data/misc/recovery/cache/databk) 总和为 2544MB, 则其额外占用了  $2544\text{MB} / 4 = 636\text{MB}$

### 2.2. 举例说明

举例来说明使用标称 8GB 的 flash 容量占用情况:

各分区容量如下 (在 sys\_partition.fex 文件中查看):

boot-resource: 32MB

env: 16MB

boot: 16MB

system: 768MB

data: 1024MB (设置->存储->内部存储空间->总容量)

misc: 16MB

recovery: 32MB

cache: 512MB

databk: 128MB

UDISK: 剩余容量

按照以上 nand 驱动管理容量算法, 计算<UDISK>及<其他分区>占用 nand 容量的情况如下:

(1) UDISK, 即用户可使用容量(即设置->存储->内存设备->总容量)显示为 4.12GB, 其占用 flash 大小为:  $4.12 \times 1024\text{MB} \times 8 / 7 = 4218.88\text{MB} \times 8 / 7 = 4821.6\text{MB}$

(2) 其他各分区容量为 (32+16+16+768+1024+16+32+512+128) MB=2544MB, 其占用 flash 大小为:  $2544\text{MB} \times 5 / 4 = 3180\text{MB}$

nand 总容量为 8G, 即  $8 \times 1024\text{MB} = 8192\text{MB}$ , UDISK 占用 flash 4821.6MB, 其他分区占用 flash 3180MB,  $8192\text{MB} - 4821.6\text{MB} - 3180\text{MB} = 190.4\text{MB}$

即剩余 190.4MB 容量‘不知所踪’, 但其实我们以上容量计算中并没有包含 nand 物理坏块、物理数据占用的情况, 其解释如下:

(1) 物理数据占用，目前 boot0、uboot 都是直接放在 nand 物理块上的，并没有放在逻辑分区；

(2) nand 物理坏块，不同的 nand 可能不同

如果按照以上算法，计算出来的‘不知所踪’的容量超出了 250MB，则就要检查是不是使用的 nand 太次了，物理坏块太多导致的；‘不知所踪’的容量小于 250MB，则在正常范围内；

其他标称的容量可以按照以上同样的计算方法即可；

### 2.3. 特殊说明

以上计算方法并不完全适用所有情况，因为 nand 驱动内对容量的保留是基于超级块计算的，而不同 flash 的规格不同、所贴 flash 数目不同、对 flash 的操作配置不同都会影响到单位超级块的大小，从而影响实际容量大小。

Confidential

### 3. 提高可用容量的方法

#### 3.1. 合理配置分区大小

从第二节的计算方法中可以看出，为了提高用户可使用的容量，除了使用质量好的 flash 外，最重要的就是合理配置各个逻辑分区的大小；

其实在公版中给出的方案中，为了让方案商改动较小快速实现方案，各个分区大小留有比较大的阈值，如果想提高用户可用容量，可以参考如下：

(1) system 分区预留了 768MB，但实际 system 分区编译出来的大小为 454M，配置成标准的 512MB 是足够的，这样可以省下  $768\text{MB}-512\text{MB}=256\text{MB}$

```
lgm@SzExdroid2:~/polaris/jb42/android/out/target/product/polaris-p4$ du -sh system.img
454M    system.img
```

(2) cache 分区大小预留了 512MB，该分区主要用于 recovery 或者 OTA 升级，其容量最好大于完整包的大小；

但其实对国内大部分无需过 cts 的平台来讲，使用 recovery 及 ota 的机会基本没有，用于升级完整包的机会更没有，一般最多用于升级差分包，预留 128MB 足够，这样可以省下  $512\text{MB}-128\text{MB}=384\text{MB}$

(3) databk 分区预留了 128MB，该分区主要用于固件修改工具克隆功能使用的，如果用不到该功能，配置 16MB 即可，可以省下  $128\text{MB}-16\text{MB}=112\text{MB}$

其他分区占用空间较小，且一般不要随意改动

总和以上，可以多分出  $(256\text{MB}+384\text{MB}+112\text{MB})\times 5/4=940\text{MB}$ ， $940\text{MB}\times 6/7=806\text{MB}$ ，即用户可用空间可以多分出 806MB 空间

以上只是针对公版 P4 机型各分区的优化方案，具体分区大小配置需要根据自己方案的实际情况及功能需求来定义

#### 3.2. 更改 nand 驱动配置

1. 在对应的 sys\_config.fex 中，在主键 nand0\_para 下添加子键 nand\_capacity\_level = 1，这样可以进一步压缩保留空间的比例。

2. 在对应的 sys\_config.fex 中，在主键 nand0\_para 下添加子键 id\_number\_ctl = 0x2 和 nand\_p1 = 0xeeeeee，减少超级块的大小，从而减少保留空间。

#### 3.3. 联系 FAE 修改驱动

1. 若以上方法采用后，仍不能满足客户对可用容量的需求，请联系 FAE 对驱动进行修改。