

RP2_toolbox

toolbox for Retroid Pocket 2

关于挂载

目前用外置tf扩展内部存储空间有如下几种方式：

- 1. 用apps2sd链接app到外置tf卡，需要在tf卡建立第二个ext4分区，第一个分区exfat不受影响，前提是需要root，兼容性较好，也较稳定，可以恢复，缺点是app数据需要手动一个一个移动到tf卡上（apps2sd并没有扩展内部存储的空间）。
- 2. 融卡，我没弄过，但是我知道一旦tf卡有问题，可能需要刷机解决，据说速度也受影响，恢复也麻烦。群里大佬遇到过崩溃的情况。
- 3. 用tf卡的exfat分区的几个镜像文件（尺寸特别大，比如32G）虚拟ext4分区，挂载到内部存储的/data/media， /data/data， /data/app，第一个分区exfat不受影响，但是会被占用一部分空间，需要adb，有脚本自动化，不需要root，不需要给tf卡分区，可以恢复，扩展了内部存储空间，可以用mtp方式通过usb拷贝大文件，速度较其他拷贝方式快，缺点是读写性能受到一定影响，但是不挂载/data/app还可以接受。还有就是一旦镜像文件损坏，会丢失相关虚拟分区的数据。
- 4. 用tf卡新建的ext4分区挂载/data/media， /data/data， /data/app，类似上面第3种，但不是镜像文件虚拟的分区，是真实的ext4分区。第一个分区exfat不受影响。需要adb，挂载和恢复有脚本自动化，不需要root，可以恢复，性能比第3种方式好，也比较稳定，扩展了内部存储空间，可以用mtp方式通过usb拷贝大文件，速度较其他拷贝方式快，缺点是需要给tf卡新建分区，有点麻烦。

apps2sd方案

- 1. 首先需要root，群文件里有root工具包（教程）。 root工具需要adb，可以下载群文件里的adb工具包，解压缩后，在把root工具文件放到adb文件夹里，USB连接RP2后，运行root脚本。
- 2. pj版的apps2sd在群文件也有。 apps2sd需要tf卡的第二个分区是ext4格式。

挂载虚拟分区（镜像文件）

- 1. 首先保证tf卡足够大（至少128G），足够块（至少U1）。 分区格式为exfat，不需要多个分区。
- 2. 需要用adb工具，可以下载群文件里的adb工具包，解压缩后，再把批处理脚本文件以及其他文件全部放到adb文件夹里，USB连接RP2后，先运行001开头的文件，成功后，再运行041开头的脚本。脚本在Win10 64bit系统测试过。脚本包在群文件有下载。

- 3. 默认挂载/data/media和/data/data的空间分别是40G和9G，可以修改。 修改脚本中下面两行的数字就可以，数字单位GB。

```
set MEDIA_FILE_SIZE=40000  
set DATA_FILE_SIZE=9000
```

挂载tf卡的ext4分区

- 1. 首先保证tf卡足够大（至少128G），足够块（至少U1）。 第一个分区格式exfat，需要多个ext4分区。
- 2. 第一个分区必须是exfat格式，用来放月光的数据等，第2，3，4分区为ext4分区。第2个分区要分大一点，用来挂载到用户可访问的内部存储空间，建议32G以上，如果需要放类似天马的60GB的ROM那可能至少要70GB以上；第3，4个分区分别至少10GB以上。
- 3. 需要用adb工具，可以下载群文件里的adb工具包，解压缩后，再把批处理脚本文件以及其他文件全部放到adb文件夹里，USB连接RP2后，先运行001开头的文件，成功后，再运行031开头的脚本。脚本在Win10 64bit系统测试过。脚本包在群文件有下载。

其他

- 1. 不管上面那种方法，都是有损坏系统的风险的，而且高度依赖tf卡，对卡的速度和质量的要求很高。 RP2开机状态下，也绝对不可以弹出tf卡。
- 2. apps2sd方案和挂载tf卡的ext4分区方案有冲突，不可以共存。