

# OEMDataPack

使用说明/v1.0.9

## 文档履历

[illegible]

# 目 录

OEMDataPack.....	1
使用说明/v1.0.0.....	1
1. 概述.....	3
2. 界面说明.....	4
3. 操作流程.....	7
3.1 单用户固件打包数据的操作流程.....	7
3.2 多用户固件打包数据的操作流程.....	10
4. 常见问题和注意事项.....	15
5. 多用户固件的配置.....	16

## 1. 概述

OEMDataPack 后面简称 OEM，OEM 工具用于量产前把将要存放在设备的数据，压缩在固件文件中，在量产时将固件烧写到设备中，从而减少了升级设备完后再做数据拷贝的生产步骤，降低人力成本。

OEM 工具兼容多用户的固件和单用户的固件的数据打包（不兼容 A10, A13），如果要对多用户的固件进行打包数据，需要对 SDK 进行相应的配置，具体配置请详细看第 5 章，而且工具的操作上也有明显的区别，**请注意红色字体的说明**。

## 2. 界面说明

双击 OEMDataPack.exe 出现如下主界面

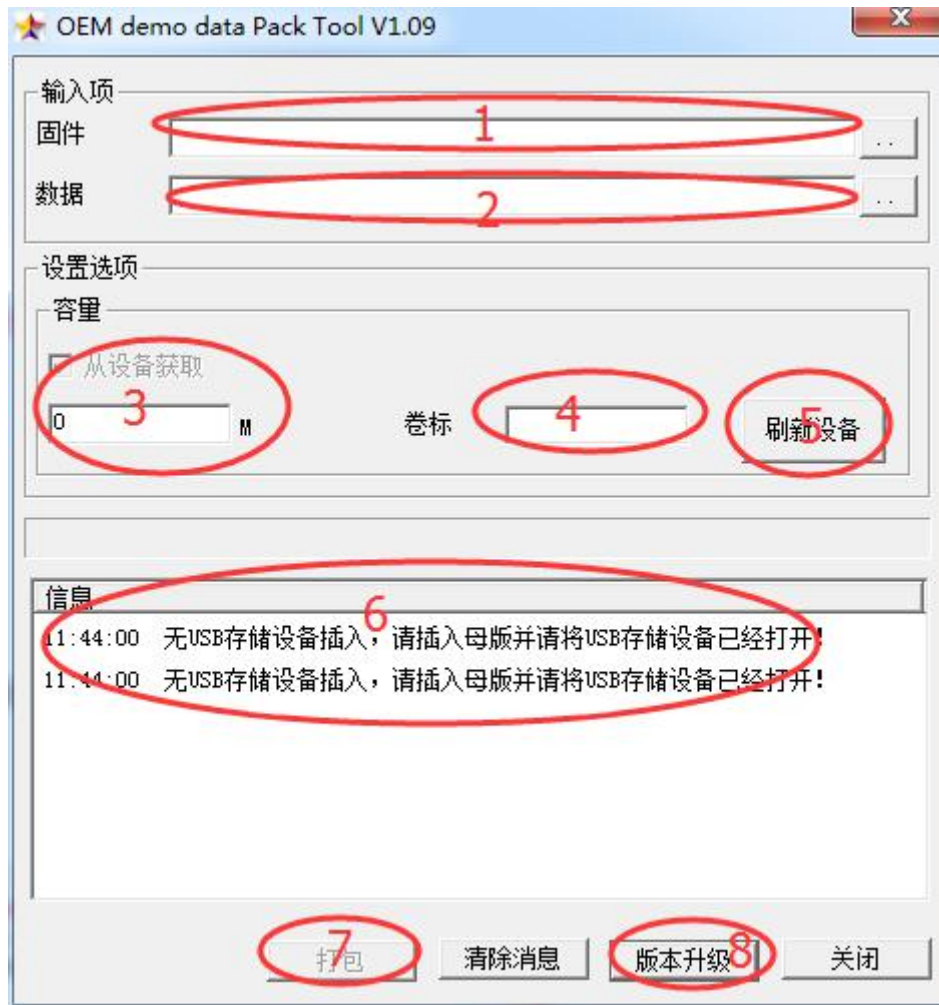


图 1：OEM 工具的界面

按图 1 标记的顺序进行功能解析如下：

### (1). 固件路径选择

选择后缀为 .img 的固件文件



图 2：固件选择按钮

### (2). 数据路径选择

选择要打包的数据所在的目录



图 3：数据选择按钮

### (3). 容量



图 4: U 盘的容量显示框

用来显示需要打包的磁盘容量。这里容量不能手动输入，只能通过插入设备获取。设备必须是已经升级过，并且是使用在上述的第 1 标号选择的固件文件。

**注：固件为多用户的时候忽略此项**

#### (4). 卷标

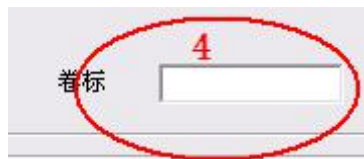


图 5: U 盘卷标设置输入框

用来设置 U 盘的卷标。

**注：固件为多用户的时候忽略此项**

#### (5). 刷新设备



图 6: 刷新设备按钮

有些设备需要在设备端进行连接 U 盘才能有效提供 PC 端进行识别，当用户确定连接好 U 盘之后，再点击刷新设备才能有效获取容量。

#### (6). 信息提示

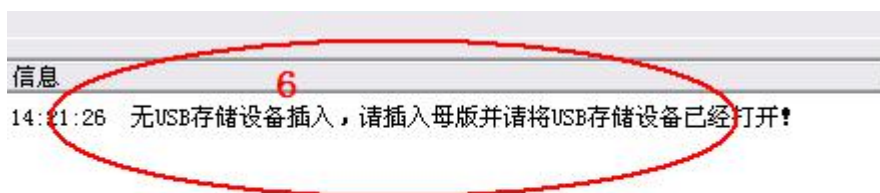


图 7: 信息交互显示框

此栏用来提示用户操作和显示操作过程中的信息。

**注：固件为多用户的时候忽略此项**

#### (7). 打包



图 8: 打包按钮

当用户选择完固件文件、数据目录路径、**如果是单用户的固件需要插入已经升级过的设备**后才可以点击该按钮进行打包。

#### (8). 版本升级



图 9：版本升级按钮

检测本工具的版本是否能升级，若能升级，用户可以选择升级。

### 3. 操作流程

#### 3.1 单用户固件打包数据的操作流程

(1). 制作设备母版

使用 PhoneixUSBPro 或者 Livsuit 升级将要生产的设备，升级过程中必须使用强制擦除的方式升级。

(2). 打开 OEM 工具（双击 OEMDataPack.exe）

选择固件文件和数据路径。数据路径的选择方法是，指定一个本地目录，该目录下的所有文件和目录将会存放在烧录好的设备的 U 盘的根目录下，如图 10 所示的选择了：F:\测试文件\Music 这个目录。

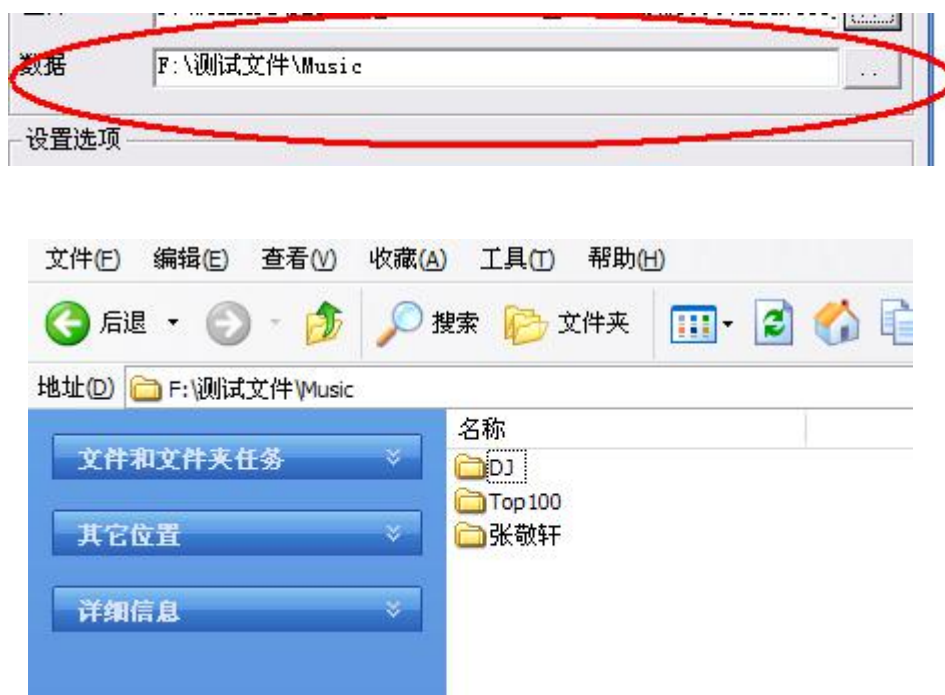


图 10：选择打包数据

(3). 插入步骤(1)中制作的母版设备

插入 USB 存储设备后，出现如下界面：





图 11: 插入 U 盘后

#### (4). 打包

点击打包按钮，进行打包。等提示信息如下时，打包结束。

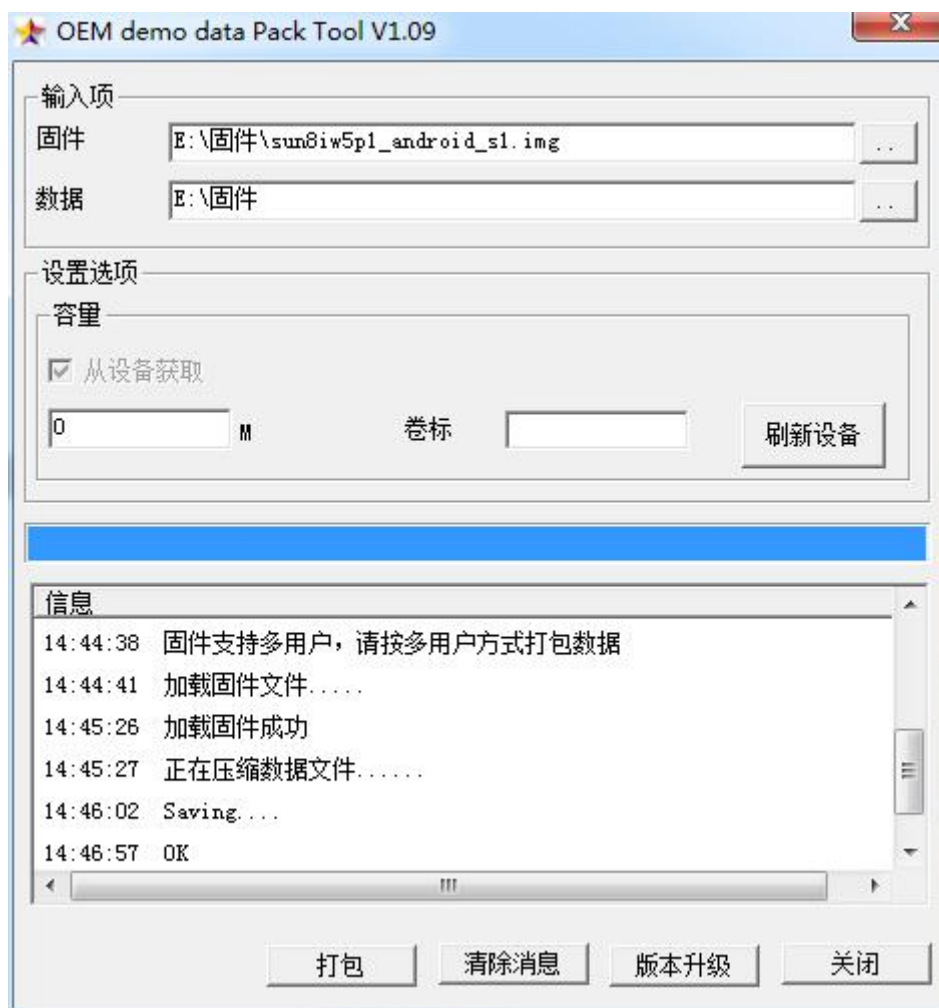


图 12: 打包结束

## (5). 查看输出文件

打包完成后，新的固件文件会在原来固件的同一级目录下，并在原来文件名的基础上用日期和时间作为文件名修饰。如图 13 所示，sun8iw5p1\_android\_s1.img 是源文件，sun8iw5p1\_android\_s1\_20160816\_144602.img 打包数据后生成的新固件。

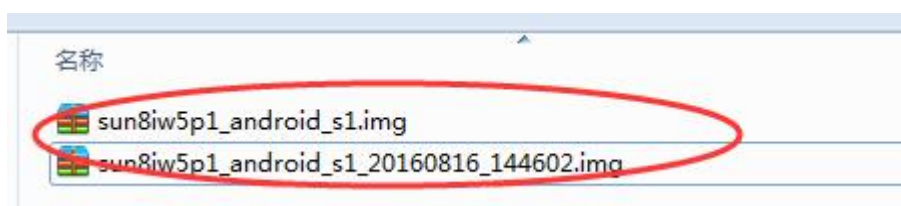


图 13: 源文件与输出文件

## (6). 使用量产

采用新生成的固件 sun8iw5p1\_android\_s1\_20160816\_144602.img 进行升级，**必须要用量产工具升级**，否则无法实现 OEM 数据在烧录过程中烧到设备上。

## (7). 连接 PC 查看打包进去的数据

升级完成后可以在 U 盘的根目录下查看打包进去的文件，如图 14 所示：

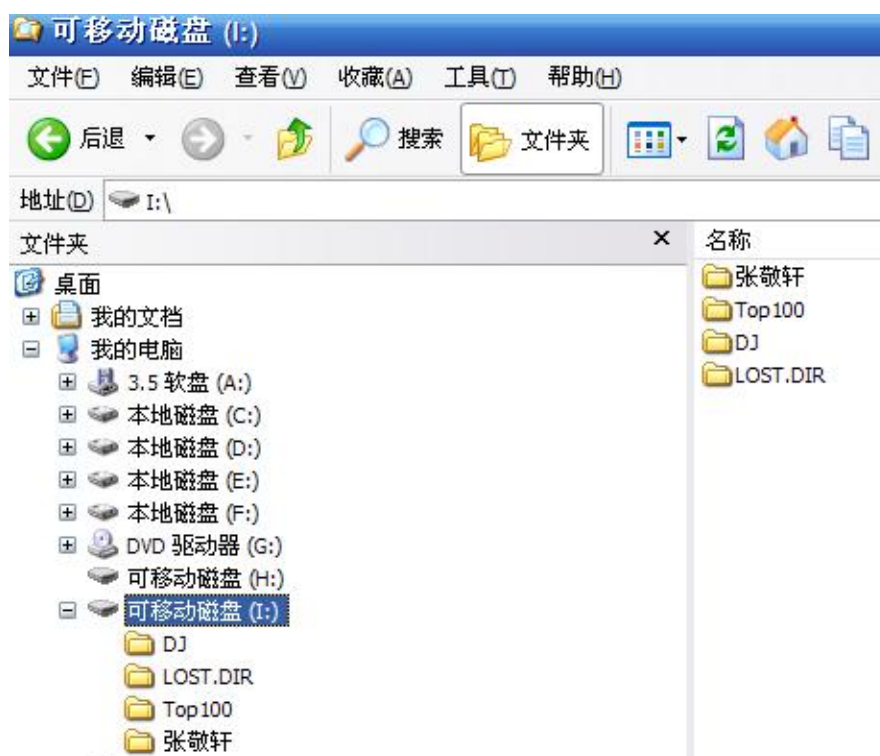


图 14：查看被打包进去的数据

## 3.2 多用户固件打包数据的操作流程

(1). 双击 OEMDataPack.exe 运行打包工具

运行打包工具，显示界面如图 15 所示。



图 15：OEM 工具的界面

(2)．选择固件

单击固件路径显示框右边的按钮选择固件。

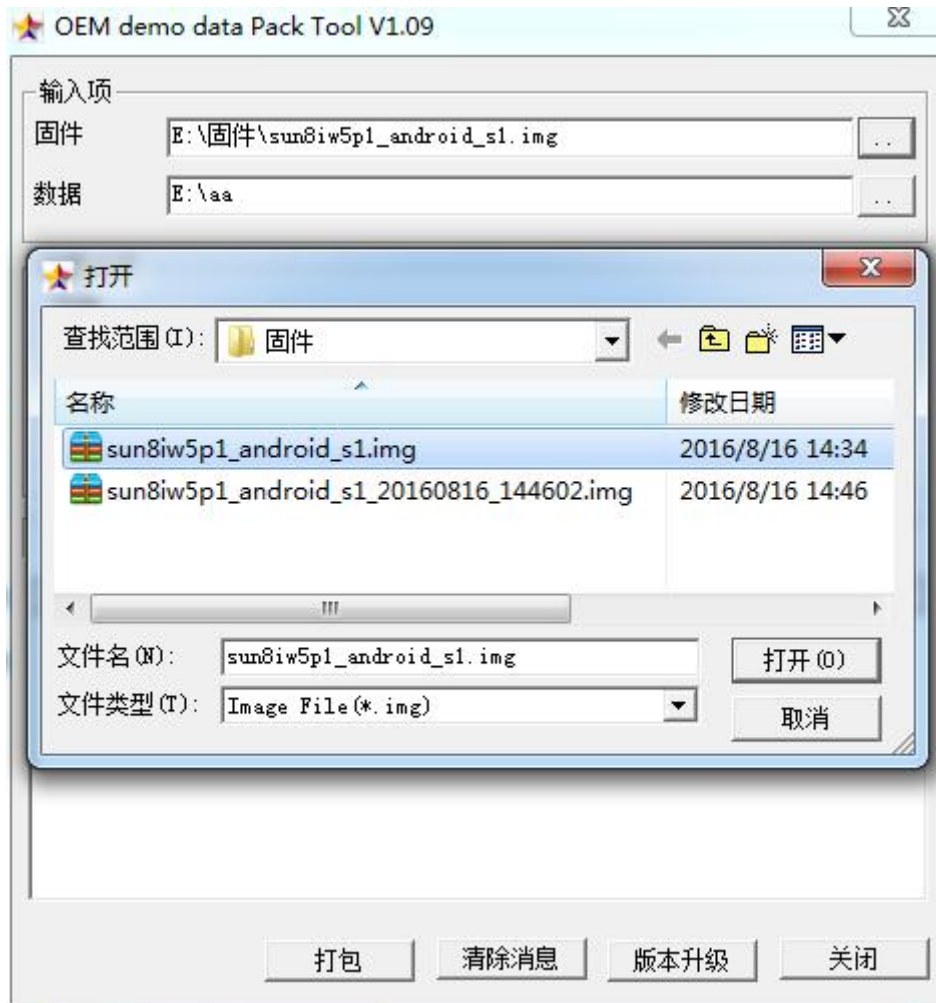


图 16：选择固件

(3). 选择需要打包进去的数据

单击数据路径显示框右边的按钮选择需要打包进去的数据。

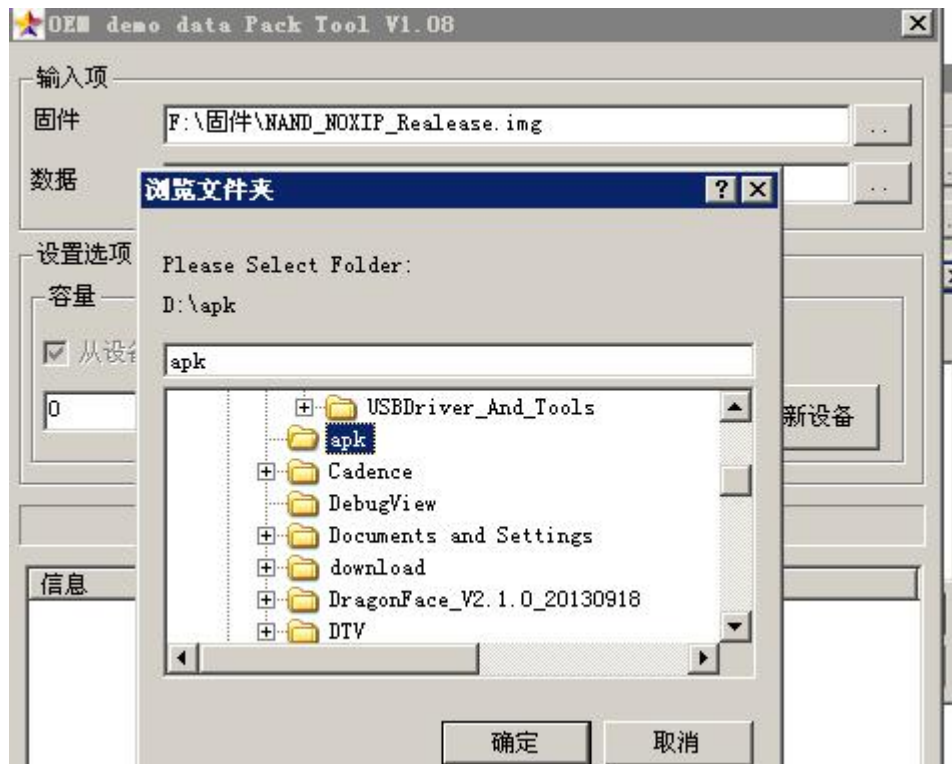


图 17：选择被打包的数据

#### (4) . 打包

点击打包按钮把数据打包进固件。

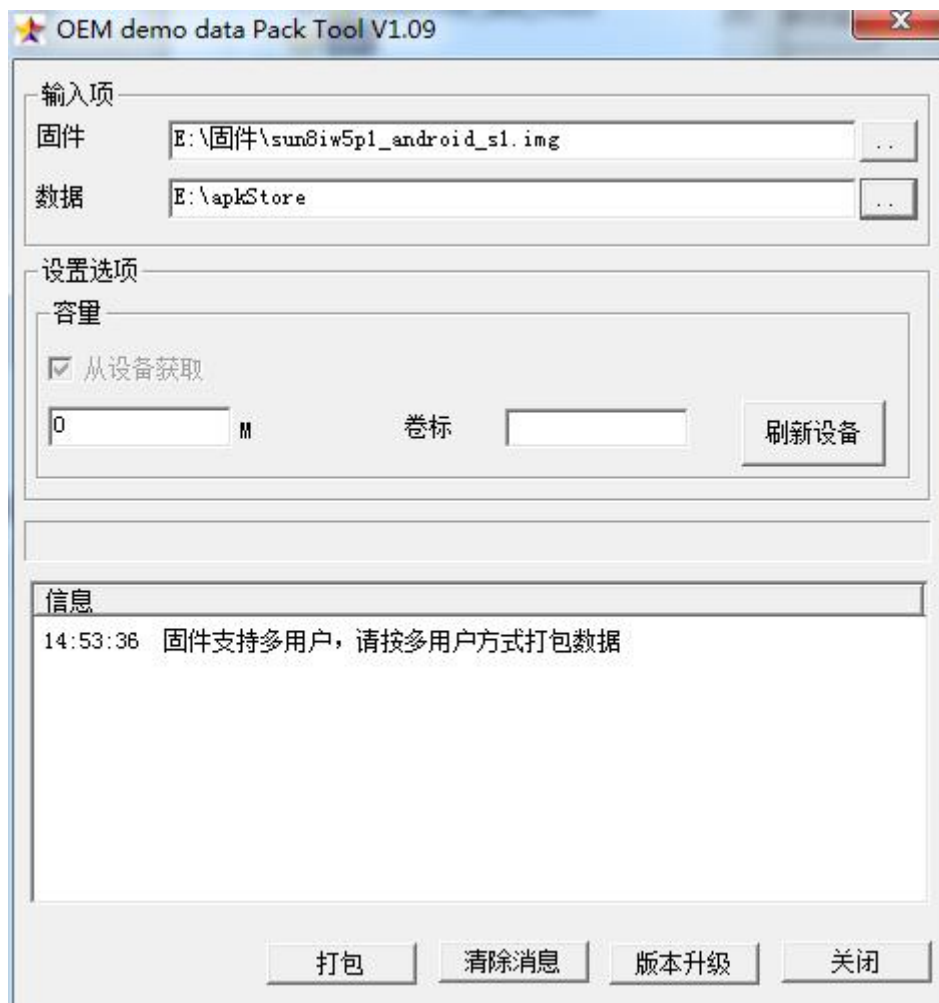


图 18: 正在打包



图 19: 打包完成

(5). 查看生成的新固件

打包后生成的新固件会与源固件在同一目录下，生成的固件的固件名字会在源固件名字的基础上增加时间修饰。

名称	修改日期	类型	大小
 sun8iw5p1_android_s1.img	2016/8/16 14:34	好压 IMG 压缩文件	737,222 KB
 sun8iw5p1_android_s1_20160816_144602.img	2016/8/16 14:46	好压 IMG 压缩文件	739,159 KB

图 20: 生成的固件

(6). 升级

不一定要采用量产工具进行升级，与单用户的不同，单用户的必须要用量产工具升级才能把打包到固件的数据烧写进去。

## 4. 常见问题和注意事项

(1). 如果固件不支持数据打包则提示如下

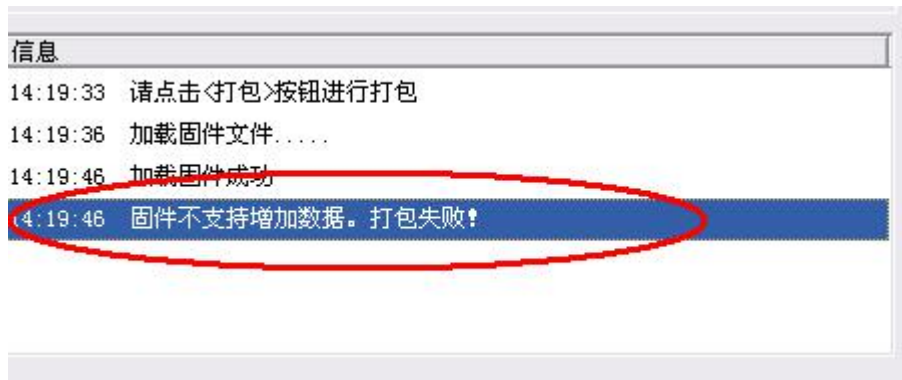


图 21: 固件不支持打包数据导致打包失败

(2). 打包数据时必须确认待打包的数据文件夹内没有问题被打开或者被其他程序占用，  
否则可能目标固件无法烧录或者丢失数据等问题。



## 5. 多用户固件的配置

如果要对多用户的固件进行打包数据，需要在 SDK 里进行相应的配置，以下具体说明。

- (1). 增加 media\_data 分区

修改 `lichee/linux-3.4/pack/chips/sun8iw3p1/configs/android/polaris-xx/sys_partition.fex`

```
;----->nandj,media_data
[partition]
    name          = media_data  //分区名
    size          = 2097152     //分区大小，会被工具根据打包数据的大小调整
    user_type     = 0x8000
```

注意添加分区的顺序，它会对应实际分区的 `/dev/block/nanda`、`/dev/block/nandb` ...，  
Size 的单位扇区（512 字节/扇区）需要 16M 对齐

- (2). 修改 `android/device/softwinner/polaris-xx/init.sun8i.rc`

创建 `/mnt/extsd1` 目录

```
mkdir /mnt/usbhost1 0000 system system
mount debugfs debugfs /sys/kernel/debug
mkdir /databk 0700 root system
mkdir /mnt/extsd1 0000 system system //增加此行
```

格式化 media\_data 分区

```
#    format_userdata /dev/block/by-name/UDISK polaris
    format_userdata /dev/block/by-name/media_data polaris //增加此行
# radio related, such as imei.conf
    mkdir /data/misc/radio 0777 system radio
```

- (3). 修改 `android/device/softwinner/polaris-xx/fstab.sun8i`

挂载 media\_data 分区。在后面增加此行，`/devices/virtual/block/nandj` 是（1）中创建的分区，  
不一定是 nandj，与 `sys_config.fex` 里的分区表顺序有关，根据实际配置。

```
/devices/virtual/block/nandj /mnt/extsd1 vfat rw,relatime,fmask=0000,dmask=0000
```

## (4). 修改

android/device/softwinner/polaris-xx/overlay/frameworks/base/core/res/res/xml/storage\_list.xml

在文件后面加上以下的内容，把/mnt/extsd1 挂成外部 SD 卡 1，可以通过文件管理器看到此目录，外置插入的 SD 卡，默认为/mnt/extsd

```
<storage android:mountPoint="/mnt/extsd1"    //(2)中创建的目录
    android:storageDescription="@string/storage_sd_card"
    android:primary="false"
    android:removable="true"
    android:emulated="false"
    android:mtpReserve="0"
    android:maxFileSize="0"/>
```

## (5). 恢复出厂设置选择是否格式化打包进去的数据（此功能非必须，跟据需求）

如果要添加此功能需要打上 OEM 工具目录下的 patch 文件夹里面提供的相关 patch（[注意阅读 Readme.txt 文件](#)），添加此功能后，在机器的：设置 -> 备份和重置 -> 恢复出厂设置，在这个界面下会看到标识为“格式化 SD 卡 1”的复选框，如果勾选上，在恢复出厂设置时，会格式化 media\_data 分区，即会清空/mnt/extsd1 目录下的所有数据，否则不会。