



# DragonFace 使用指南



版本号：4.0.7  
发布时间：2021-12-22

## 版本历史

版本	日期	责任人	版本描述
4.0.7	2021-12-22	KPA0398	Android 支持修改壁纸
4.0.6	2021-12-13	KPA0398	支持 android S
4.0.2	2021-09-01	KPA0398	修复删除 apk 不成功问题
4.0.1	2021-08-02	KPA0398	增加支持 android O&P 安全方案修改补丁说明
4.0.0	2021-06-15	KPA0398	界面重构版本, 支持多个模块功能配置化, 支持全平台
2.6.1	2020-11-27	KPA0398	1.更新文档模板和排版。 2.更新截图。
2.6.0	2020-06-01	KPA0398	增加支持 DTS 修改。
2.5.2	2020-01-02	KPA0398	支持 android P, Q。
2.3.0	2019-08-08	KPA0398	多个平台固件适配。
2.2.3	2018-06-05	KPA0398	增加 android 操作方式变更说明。
2.2.0	2017-04-09	KPA0398	初始版本。



# 目录

版本历史.....	i
目录.....	ii
图片目录.....	iii
1 前言.....	1
1.1 文档简介.....	1
1.2 目标读者.....	1
1.3 适用范围.....	1
1.4 环境.....	1
1.4.1 硬性要求.....	1
1.5 文档约定.....	1
1.5.1 标志说明.....	1
2 概述.....	2
2.1 工具简介.....	2
2.2 软件运行.....	2
2.3 软件界面.....	2
2.3.1 工具注册.....	2
2.4 规格简单介绍.....	4
3 功能详解.....	6
3.1 基本功能.....	6
3.1.1 配置以及导入固件.....	6
3.2 板卡配置.....	7
3.3 系统配置.....	8
3.3.1 基本设置.....	8
3.3.2 画面音乐.....	8
3.3.3 APK.....	15
3.3.4 高级设置.....	19
3.3.5 Rootfs.....	21
3.3.6 保存文件.....	22
3.4 预制 OEM 数据.....	22
3.4.1 oem 数据界面设置.....	23
3.4.2 打包 oem 数据.....	24
3.4.3 打包多用户固件配置.....	25
3.4.4 预制 oem 数据常见问题和注意事项.....	27
3.5 固件节点替换.....	27
3.5.1 固件节点替换界面设置.....	27
3.5.2 节点文件替换设置.....	28
3.5.3 打包固件节点替换.....	29
3.6 设置 key(保护固件).....	30
4 其他事项.....	35

# 图片目录

图 2-1	APST 直接执行示意图.....	2
图 2-2	工具文件夹直接执行软件示意图.....	2
图 2-3	工具注册界面.....	3
图 2-4	注册成功界面.....	3
图 2-5	软件界面示意图.....	4
图 3-1	软件配置界面示意图.....	6
图 3-2	导入固件示意图.....	7
图 3-3	板卡配置示意图.....	7
图 3-4	系统信息示意图.....	8
图 3-5	画面音乐示意图.....	9
图 3-6	替换 bootlogo 示例示意图.....	10
图 3-7	替换 Kernellogo 示例示意图.....	11
图 3-8	替换开机动画示例示意图.....	12
图 3-9	替换开机视频示例示意图.....	13
图 3-10	bootlogo 预览示意图.....	13
图 3-11	Kernellogo 预览示意图.....	14
图 3-12	开机动画示意图.....	14
图 3-13	添加/删除壁纸.....	14
图 3-14	APK 示意图.....	15
图 3-15	删除 APK 示意图.....	16
图 3-16	添加 APK 示意图.....	17
图 3-17	解压 APK 等待示意图.....	18
图 3-18	解压完 APK 显示示意图.....	18
图 3-19	高级设置示意图.....	19
图 3-20	build.prop 文件示意图.....	20
图 3-21	rc 文件示意图.....	20
图 3-22	键盘与遥控器 kl 文件示意图.....	21
图 3-23	oem 分区结构示意图.....	21
图 3-24	rootfs 目录结构示意图.....	21
图 3-25	保存文件示意图.....	22
图 3-26	预制 OEM 数据配置示意图.....	23
图 3-27	分区设置示意图.....	23
图 3-28	自定义分区设置示意图.....	24
图 3-29	打包 oem 数据示意图.....	25
图 3-30	增加 media_data 分区示意图.....	26
图 3-31	创建/mnt/extsd1 目录示意图.....	26
图 3-32	格式化 media_data 分区示意图.....	26
图 3-33	增加配置示意图.....	26
图 3-34	挂载外部 SD 卡示意图.....	26
图 3-35	固件不支持打包数据导致失败示意图.....	27

图 3-36	固件节点替换配置示意图.....	28
图 3-37	节点文件替换示意图.....	29
图 3-38	打包固件节点替换示意图.....	30
图 3-39	输入密码示意图.....	31
图 3-40	有密码的固件显示密码输入示意图.....	32
图 3-41	密码错误提示示意图.....	33
图 3-42	密码正确后的示意图.....	34



# 1 前言

## 1.1 文档简介

本档介绍了 DragonFace 的使用方法。

## 1.2 目标读者

DragonFace 的使用者。

## 1.3 适用范围

本档仅适用于运行在 Windows 系统中的 DragonFace 软件。

## 1.4 环境

### 1.4.1 硬性要求





操作系统	WindowXP、Vista (32 位、64 位)、Win7(32 位、64 位)、WIN10 (32 位、64 位)
PC 配置要求	C 盘剩余空间至少 5G 以上 工具安装所在盘至少 1G 以上剩余空间 内存要求 1G 以上
固件版本	Android4.0 及以上版本
运行要求	<b>工具必须在 NTFS 的磁盘上运行且是英文目录下</b>

## 1.5 文档约定

文档中给出以下约定。

### 1.5.1 标志说明

本档采用各种醒目的标志来表示在操作过程中应该特别注意的地方，这些标志的含义如下：

标识	说明
 <b>警告</b>	该标志后的说明应给予格外关注，如果不遵守，可能会导致人员受伤或死亡。
 <b>注意</b>	提醒操作中应注意的事项。不当的操作可能会损坏器件，影响可靠性、降低性能等。
 <b>说明</b>	为准确理解文中指令、正确实施操作而提供的补充或强调信息。
 <b>窍门</b>	一些容易忽视的小功能、技巧。了解这些功能或技巧能帮助解决特定问题或者节省操作时间。

## 2 概述

### 2.1 工具简介

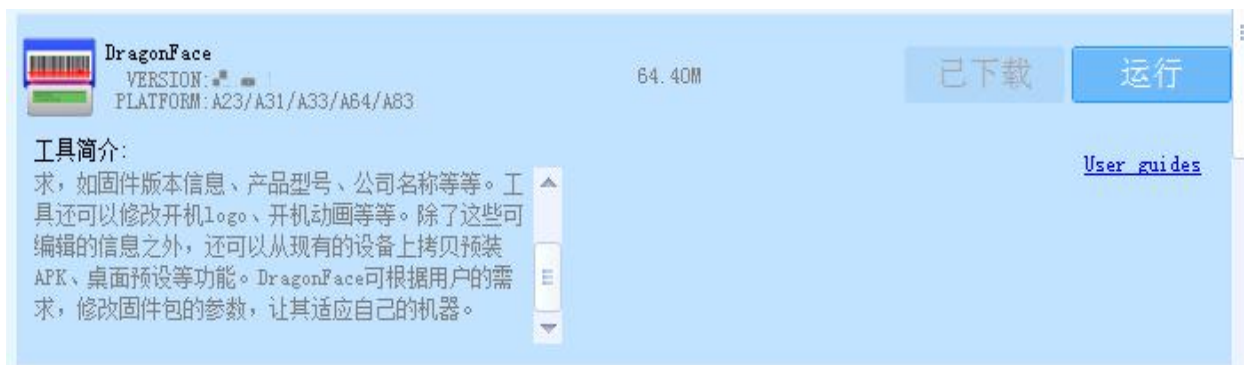
固件修改工具 (DragonFace) 是一个高效的固件修改——功能强大、所见即所得、操作便易的工具。只要是用于在某个发布固件中修改 OEM 厂家客户的需求，如固件版本信息、产品型号、公司名称等等。工具还可以修改开机 logo、开机动画等等。除了这些可编辑的信息之外，还可以预制 oem 数据如预装 APK、文件等功能。DragonFace 还可根据用户的需求，修改固件包的板卡参数，让其适应自己的机器。操作简洁方便。

### 2.2 软件运行

该软件的启动方式有两种。

(1) 在 APST (全志量产工具中心, 英文全称为 Allwinner Product Support Tool, 简称为 APST) 上找到 DragonFace, 点击“运行”按钮运行软件, 如图 2-1 所示。

图 2-1 APST 直接执行示意图



(2) 在工具文件夹中运行可执行文件“DragonFace.exe”，如图 2-2 所示。

图 2-2 工具文件夹直接执行软件示意图

cygwin1.dll	2016/11/22 15:32	应用程序扩展
cygz.dll	2016/11/22 15:31	应用程序扩展
Dbgview.exe	2017/12/15 10:57	应用程序
DragonFace.exe	2020/10/10 18:29	应用程序
Dragonface.ini	2017/9/14 10:23	Configuration Se...
dragonface.lan	2020/9/28 13:37	LAN 文件

### 2.3 软件界面

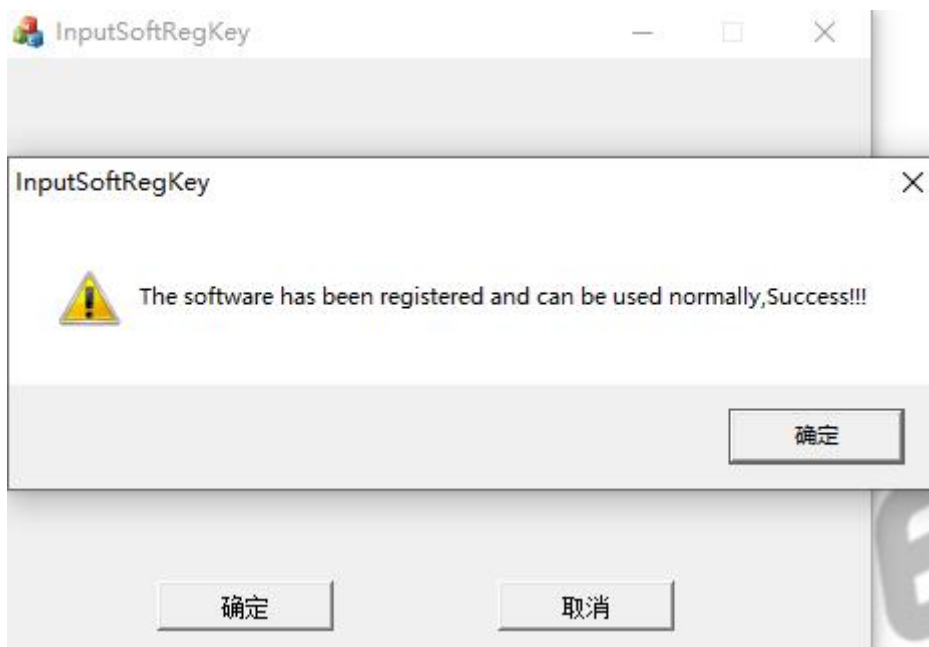
#### 2.3.1 工具注册

第一次使用工具需要注册，请打开 InputSoftRegKey.exe 进行注册，如图 2-3 所示。请复制 ID 向 FAE 获取 passwd，注册成功后如图 2-4 所示。

图 2-3 工具注册界面



图 2-4 注册成功界面



注册成功后就可以打开 DragonFace.exe, DragonFace 运行时的软件界面如图 2-5 所示。

图 2-5 软件界面示意图



## 2.4 规格简单介绍

(1) 支持修改的功能规格如表 2-1 所示。

表 2-1 功能规格示意图

列表	图标	功能介绍	备注
配置		1. 支持模块配置化	
板卡配置		1. Dts 2. Sys_config 3. sys_partition	
系统配置		基本设置: 1. 设置品牌 2. 设置设备 3. 设置制造商	

		<p>4.设置模型</p> <p>6.设置时区国家语言</p> <p>5.保存电脑信息到固件中</p> <p>画面音乐:</p> <p>1.可替换 Bootlogo</p> <p>2.可替换 Kernellogo</p> <p>3.可替换开机动画</p> <p>4.可替换开机音乐</p> <p>5.可替换开机视频</p> <p>6.预览 BootLogo、kernelLogo、开机动画效果</p> <p>APK:</p> <p>预装用户不可卸载 APK, 并可删除或替换系统内置的 APK, 如 lacher.APK</p> <p>高级设置:</p> <p>1.修改 build.prop</p> <p>2.修改 Init.rc</p> <p>3.添加键盘与遥控器 kl 文件</p> <p>4.添加配置文件</p> <p>Rootfs:</p> <p>1.打开 rootfs 目录, 操作 rootfs</p>	
预制 oem 数据	预制OEM数据	<p>1.预制 oem 数据到 udisk 分区</p> <p>2.预制 oem 数据到 media_data 分区</p> <p>3.预制 oem 数据到自定义分区</p> <p>4.支持 file、fat、ext4、f2fs 文件系统</p>	
固件节点替换	固件节点替换	<p>1.支持替换固件节点</p>	
设置 key 保护固件	设置key(保护固件)	<p>1.保存固件时支持输入密码以保护固件</p>	
打包	打包	<p>1.修改完毕后保存新的 img 文件</p>	
使用手册	使用手册	<p>1.支持工具上打开使用手册</p>	
日志	工具详细日志	<p>1.支持工具上实时保存记录 log</p>	
关闭	关闭	<p>1.退出 DragonFace</p>	

## 3 功能详解

### 3.1 基本功能

#### 3.1.1 配置以及导入固件

(1) 先点击左上角的“配置”按钮，根据需要勾选需要修改的配置如图 3-1 所示，详细可修改的规格看表 2-1 所示。

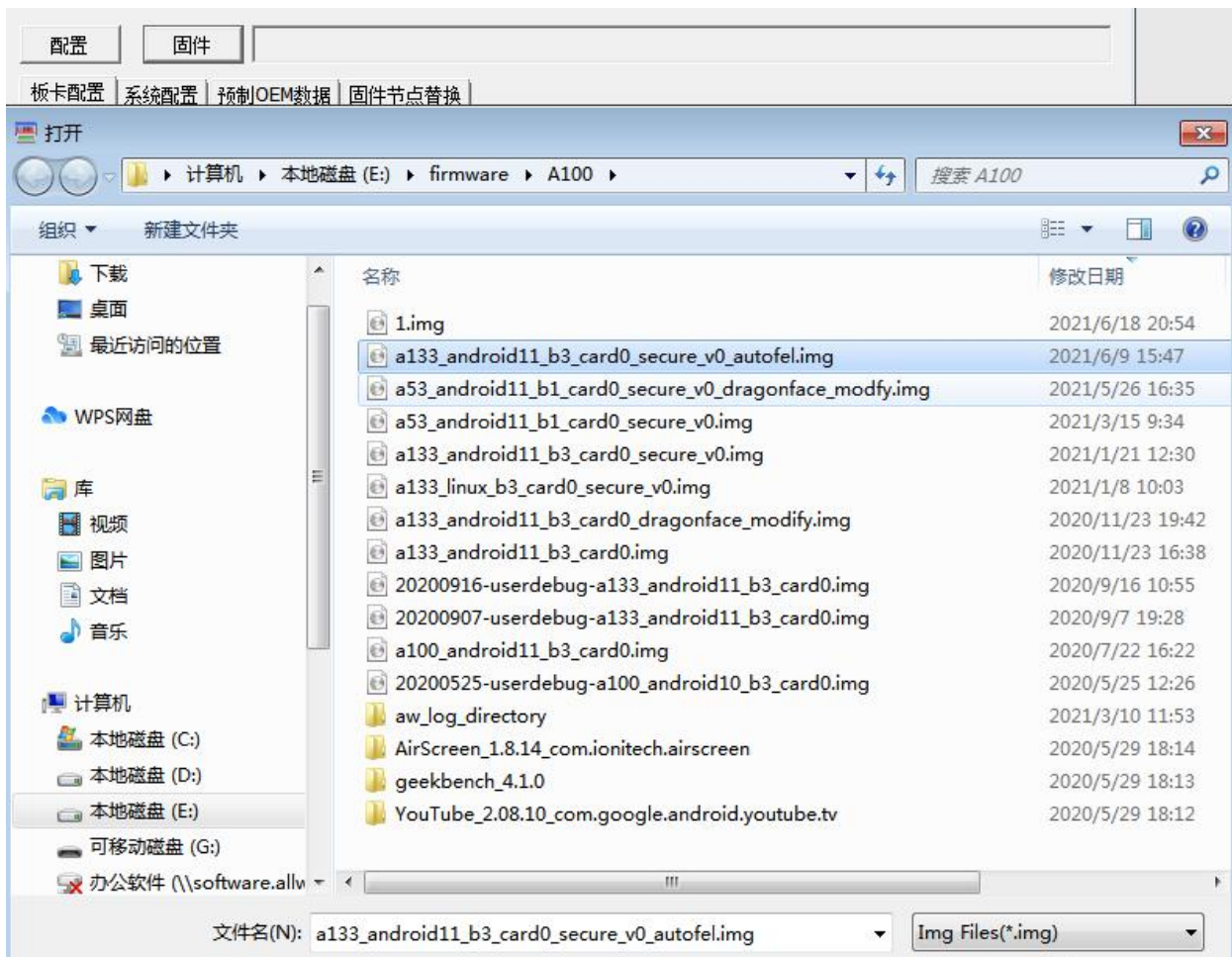
图 3-1 软件配置界面示意图



(2) 点击左上角“固件”按钮；

(3) 选择需要导入的 img 文件；

图 3-2 导入固件示意图

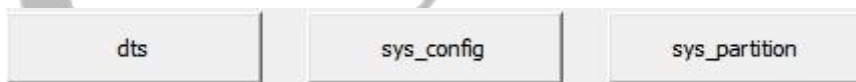


(3) 点击“打开”按钮；如图 3-2 所示。

## 3.2 板卡配置

当只需要修改板卡配置的时候，配置的时候可以只勾选板卡配置，这样可以非常快速的加载完固件以供修改。

图 3-3 板卡配置示意图



点击相应的按钮即会弹出相应的配置。

### 注意

① 不支持修改安全固件 uboot 中的 dts 和 sys\_config，安卓固件 android R 及以上安全和非安全固件修改的是 vendor\_boot 分区里的 dts，不支持修改 sys\_config，android R 以下平台非安全固件修改的是 uboot 中的 dts。

② 当修改的参数在 dts 和 sys\_config 都有时，需要同时修改。

如修改 sys\_config 里 platform 下的节点 eraseflag 改为 0 则同样在 dts 配置里面需要同时把这个参数改为 0。

```
[platform]
eraseflag = 1
debug_mode = 3
```

修改 dts 同理。

- ③修改完后请手动保存修改的文件。

## 3.3 系统配置

### 3.3.1 基本设置

图 3-4 系统信息示意图

Sysinfo			
品牌	Allwinner	设备	ceres-b3
分辨率	800 * 1280	制造商	QUAD-CORE A133 b3
模型	Allwinner	时区	
语言		国家	
指印	Allwinner/ceres_b3/ceres-b3: 11/RP1A.201005.006/1.0.1:userdebug/test-keys		
账号	ceres_b3-userdebug 11 RP1A.201005.006 1.0.1 test-keys		
PCInfo			
电脑名称		用户名	
物理地址		修改时间	

根据需求用户可以：

- (1) 设置品牌；
- (2) 设置设备；
- (3) 设置制造商；
- (4) 设置模型；
- (5) 设置时区国家语言。
- (6) 设置指印和账号。

PCInfo 会记录此次修改的 pc 信息，包括电脑名称、物理地址、用户名、修改时间。这些信息也会保存在固件的属性里面，在 adb 下通过 getprop 可以获取。

### 3.3.2 画面音乐

进入设置界面：如图 3-5 所示

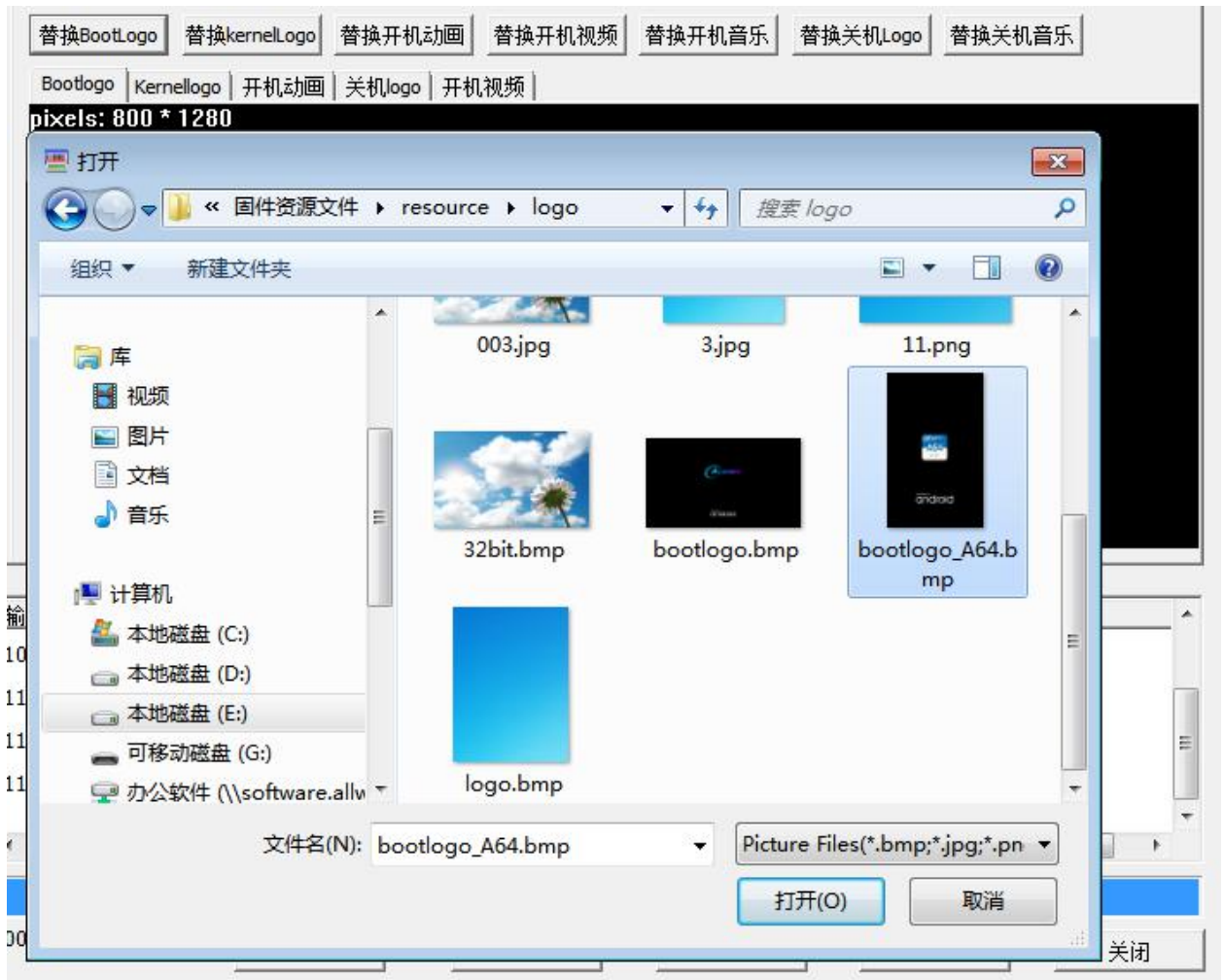
图 3-5 画面音乐示意图



替换功能:

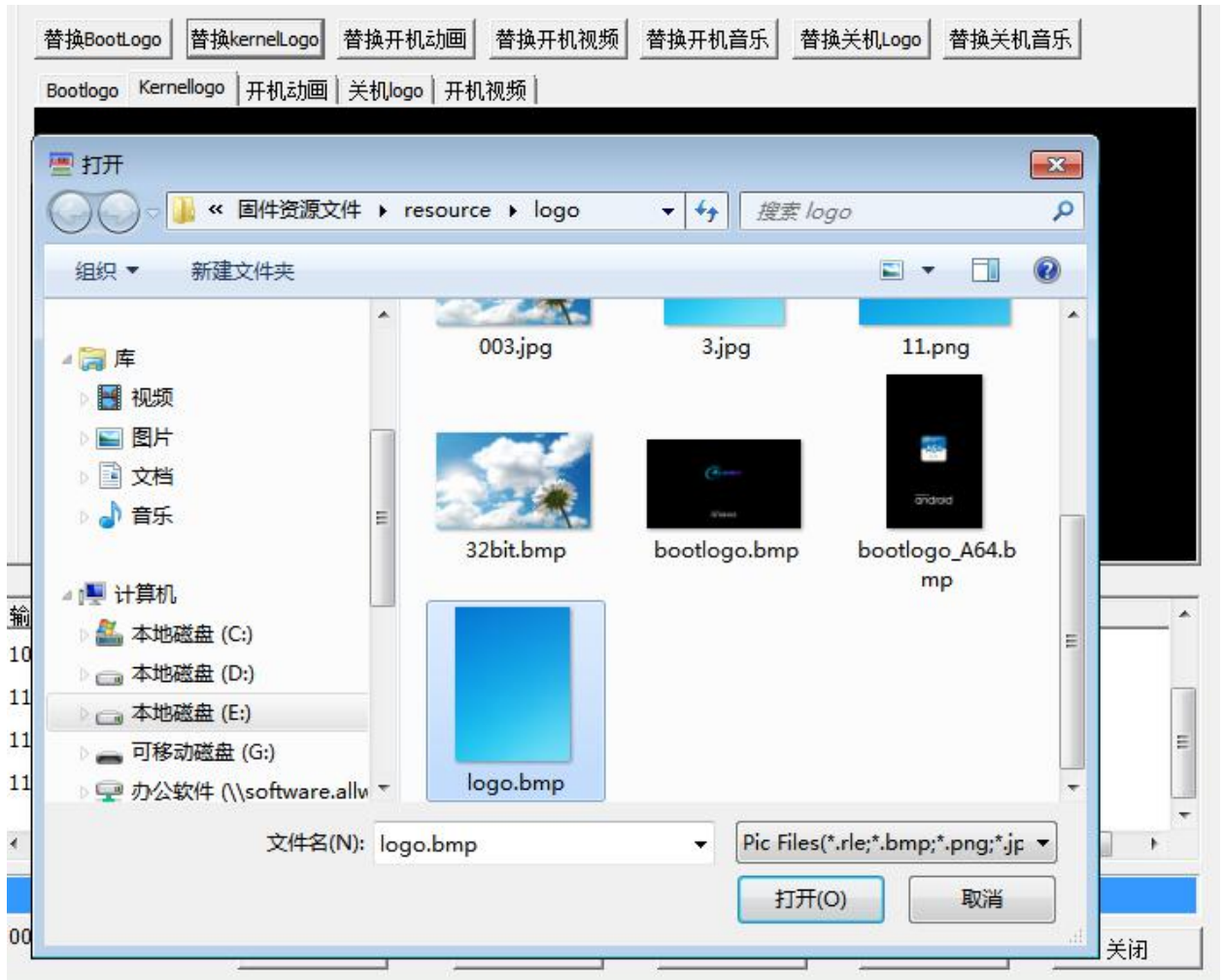
按钮名称	图标	功能呢	备注
替换 BootLogo		可替换固件的开机 BootLogo	图片文件尺寸在 1280*800 范围内
替换 kernel logo		可替换固件开机 kernel logo	图片文件尺寸在 1280*800 范围内
替换开机动画		可替换固件开机的动画 (ZIP 文件)	
替换开机视频		可替换固件的开机视频	需定制化
替换开机音乐		可替换固件开机音乐	
替换关机 logo		可替换固件关机 logo	需定制化
替换关机音乐		可替换固件关机音乐	暂未支持

图 3-6 替换 bootlogo 示例示意图



替换 Kernel logo 示例:

图 3-7 替换 Kernellogo 示例示意图

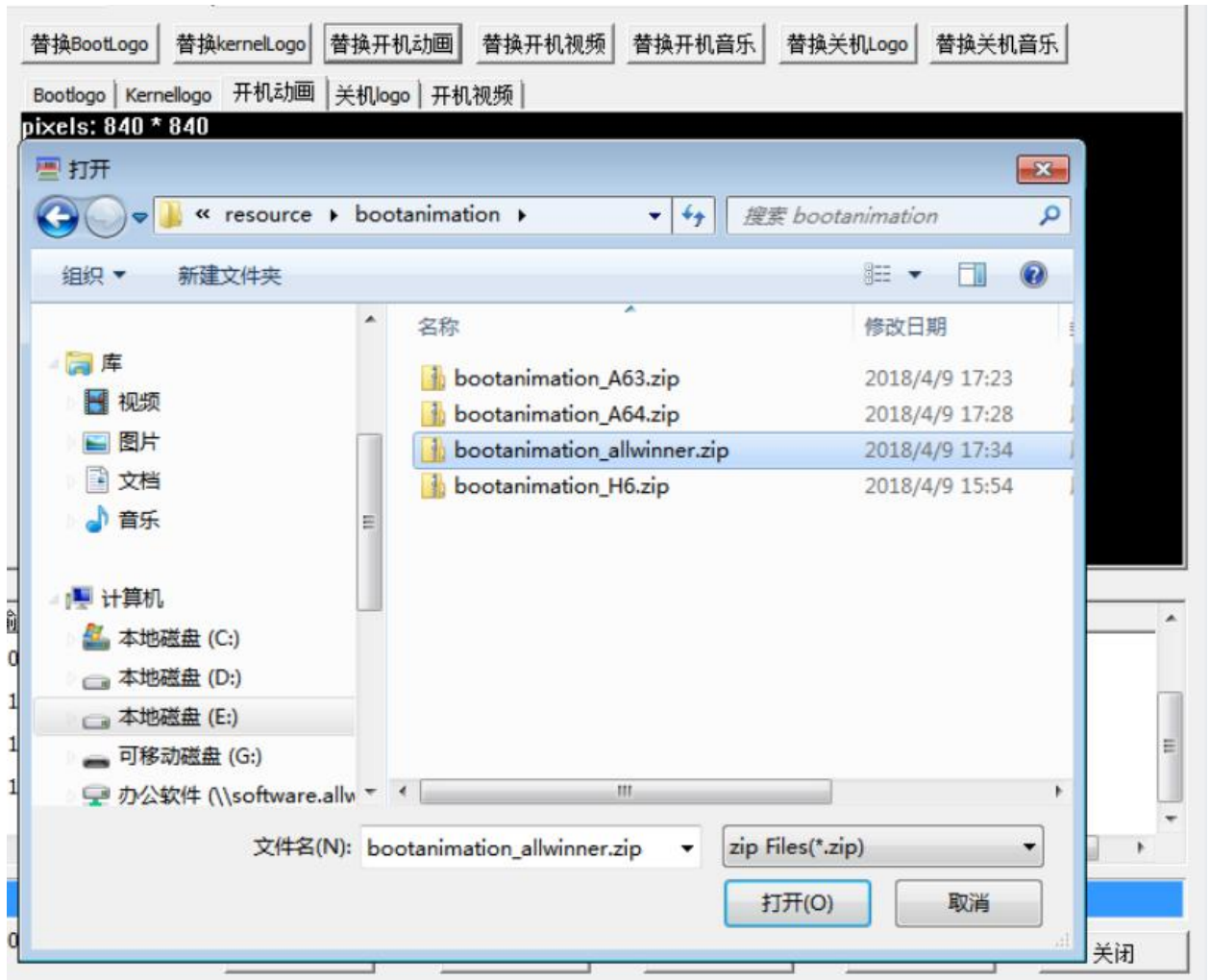


### ⚠ 注意

1. 替换 bootlogo 说明: bootlogo 存在于 boot-resource 分区, 里面包含了 bootlogo 图片。
2. 替换 kernellogo 说明: kernellogo 存在于 bootlogo 分区, 且是一张 jpg 或者 bmp 图片。

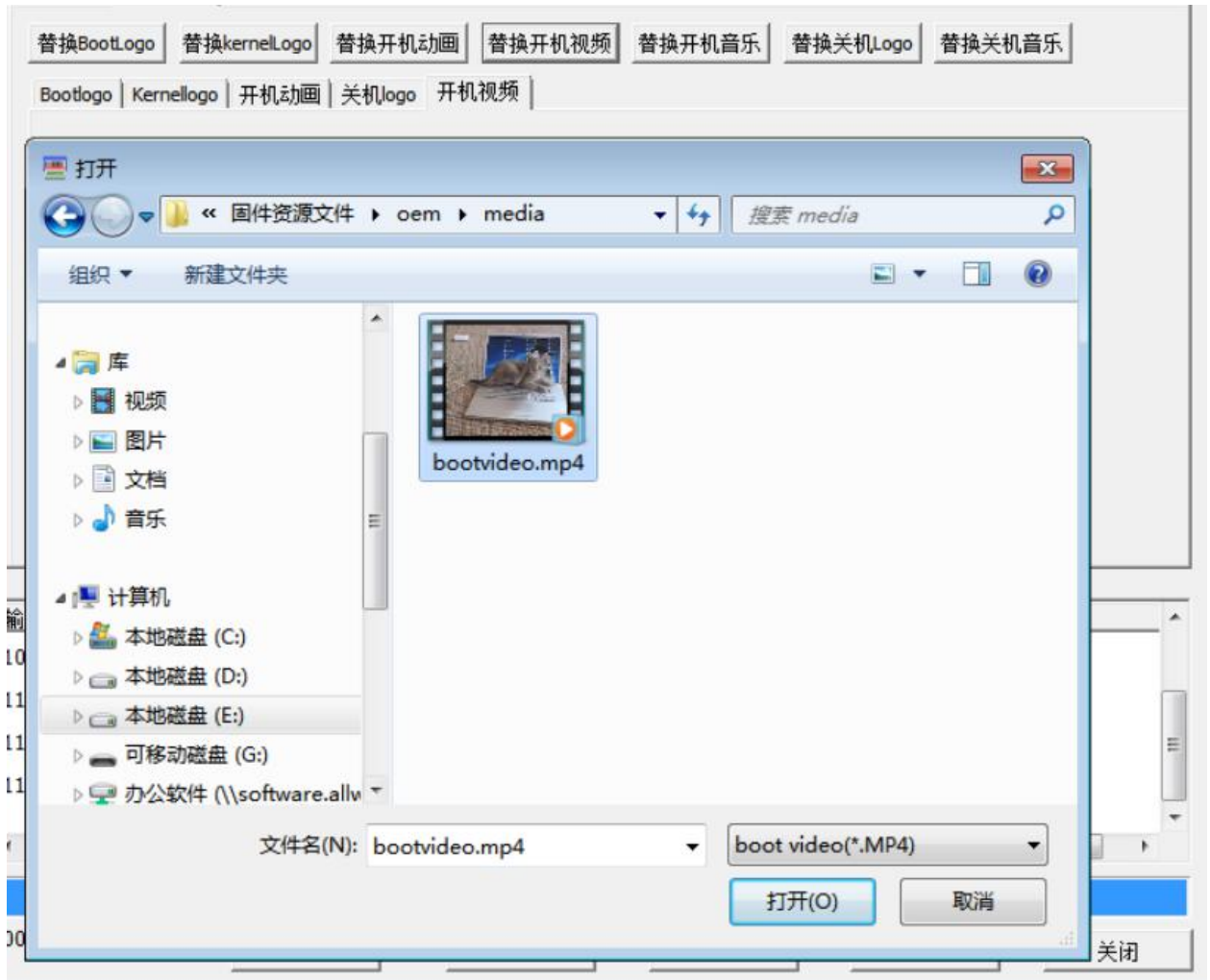
替换开机动画示例:

图 3-8 替换开机动画示例示意图



替换开机视频示例:

图 3-9 替换开机视频示例示意图



效果预览区，可预览固件的 Bootlogo、Kernellogo、开机动画、壁纸；

Boot logo 效果预览：

图 3-10 bootlogo 预览示意图



Kernel logo:

图 3-11 Kernellogo 预览示意图



开机动画:

图 3-12 开机动画示意图



壁纸:

图 3-13 添加/删除壁纸



添加壁纸：点击添加壁纸按钮，可以进行添加壁纸，壁纸在设备中存放路径为：/data/media/0/wallpaper

删除壁纸：点击删除壁纸按钮，可删除当前显示的壁纸。

显示：选择不同的项可以预览之前添加的壁纸。

设置当前图片为默认壁纸：勾选此按钮，可以设置当前显示的图片为开机默认壁纸。



### 注意

1.添加的壁纸存放在 data（udisk）分区，恢复出厂会被擦除，默认壁纸存放在 media\_data 分区，恢复出厂不会擦除。

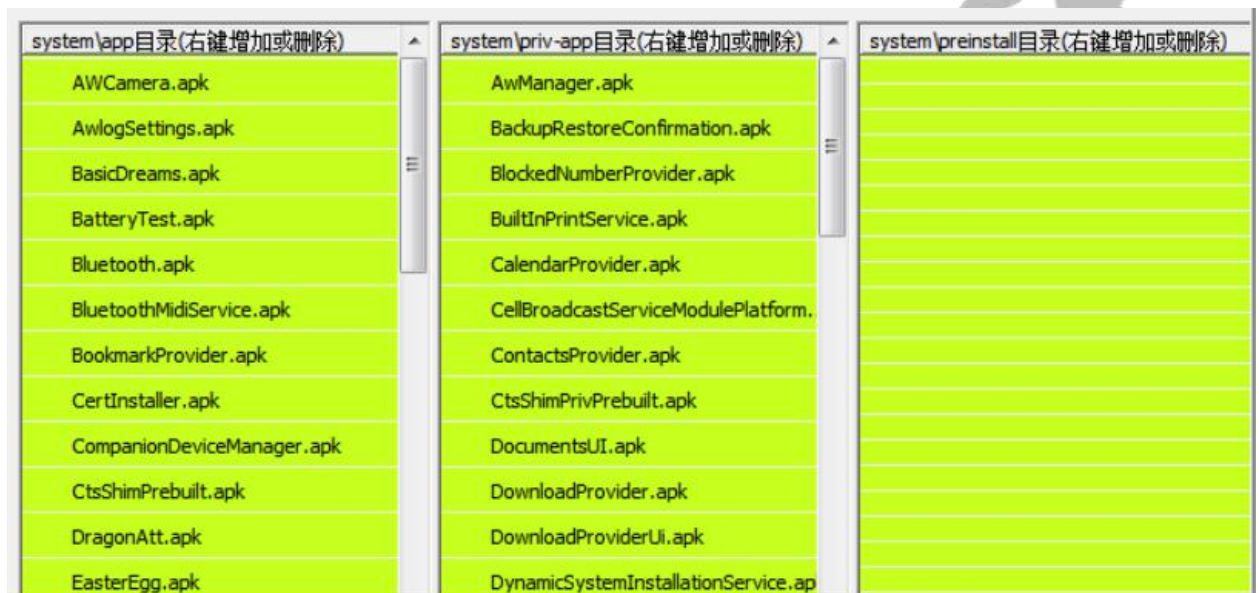
2.若要显示添加的壁纸，需要用 phoenixusbpro 多路量产工具进行固件烧写。

### 3.3.3 APK

预装用户不可卸载 APK，并可删除或替换系统内置的 APK，如 lacher. APK2. 6. 1

进入操作界面：

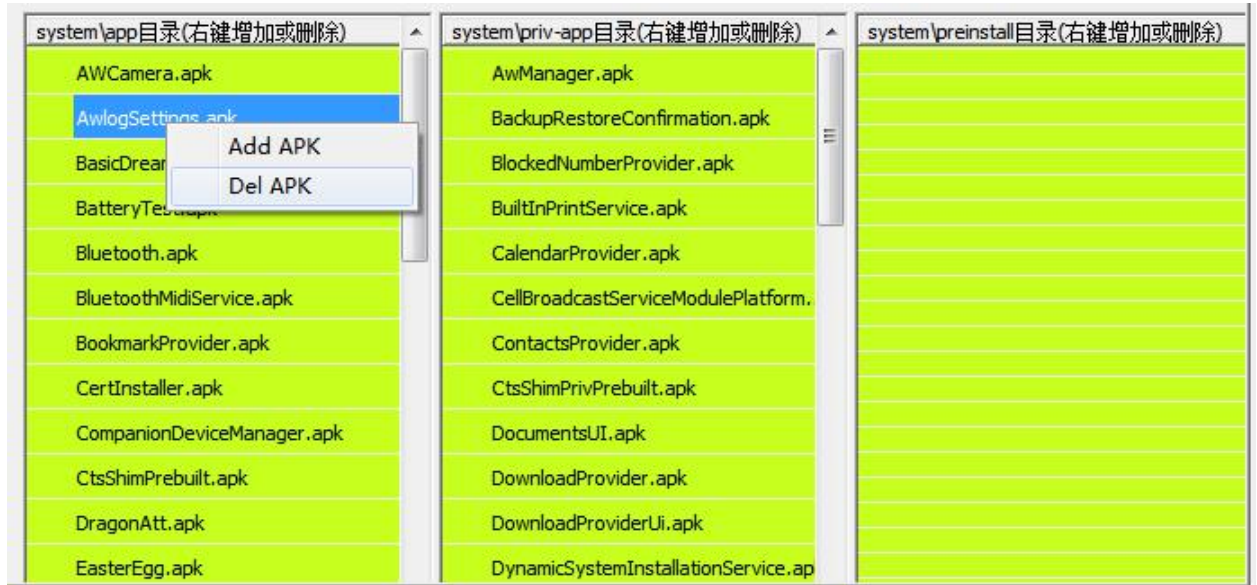
图 3-14 APK 示意图



删除 apk：

- (1) 右键选中需要删除的 APK；
- (2) 选择“Del APK”

图 3-15 删除 APK 示意图



(3) 弹出对话框，选择“是”

 **注意**

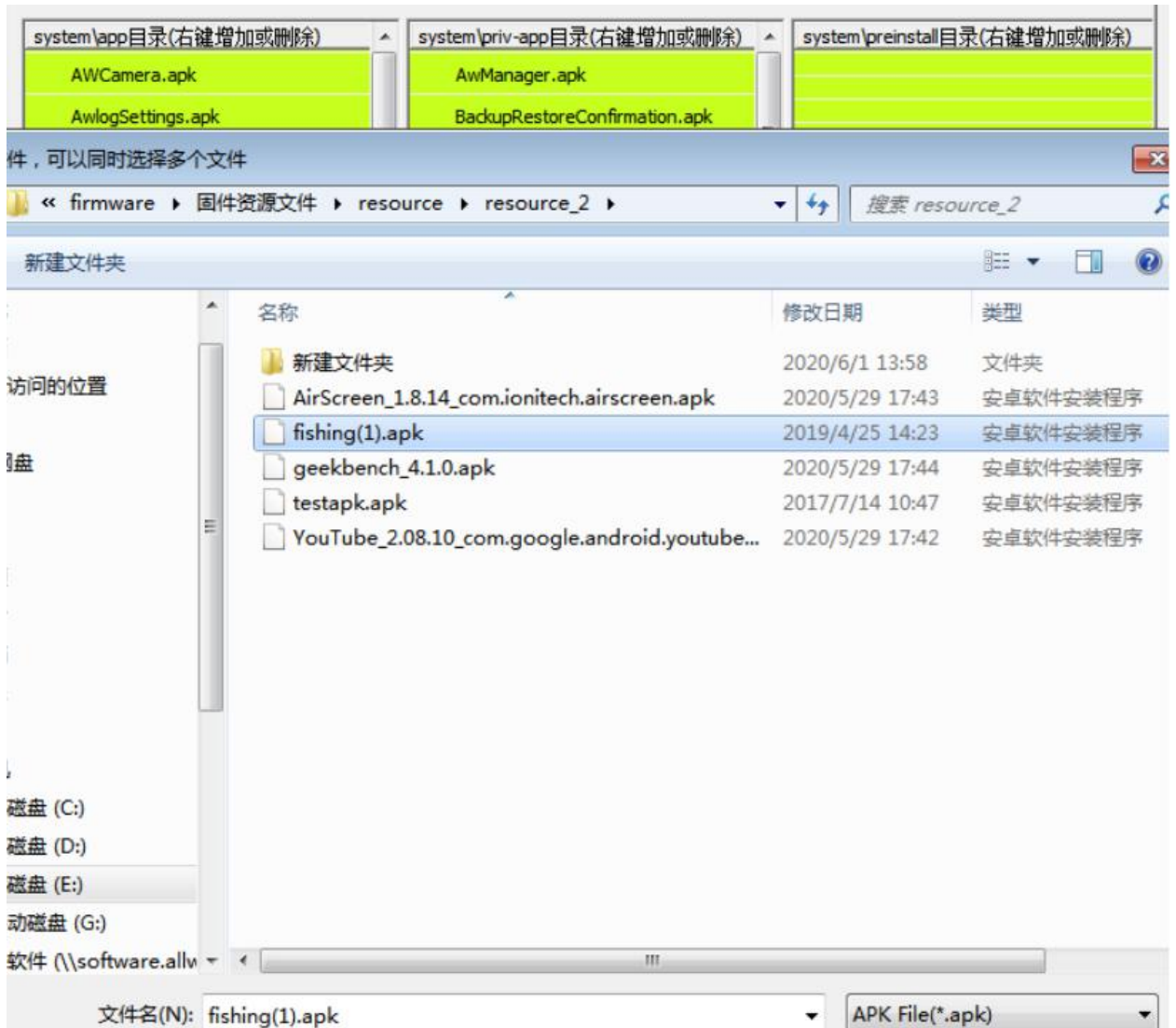
(1) 10.0 以及 10.0 以上的 android 固件，删除 apk 需要定制化，需要 sdk 打上删除 apk 的补丁。

(2) system\priv-app 目录下安装的 apk 属于特权应用，某些权限需要应用自身去申请白名单，若打包的固件无法启动先在 adb 下 logcat -b crash 看下是否是 apk 安装失败导致，若是请确认是跟白名单有关。

添加 apk:

- (1) 右键 APK 列表区域内;
- (2) 弹出“Add APK”，选择;
- (3) 选择需要添加 APK 文件的路径，并点击“打开”。

图 3-16 添加 APK 示意图



(4) 等待解压 apk (如图 3-17 解压 APK 等待示意图) 完后即可以看到添加的 apk 在列表中, 如图 3-18 所示。

图 3-17 解压 APK 等待示意图

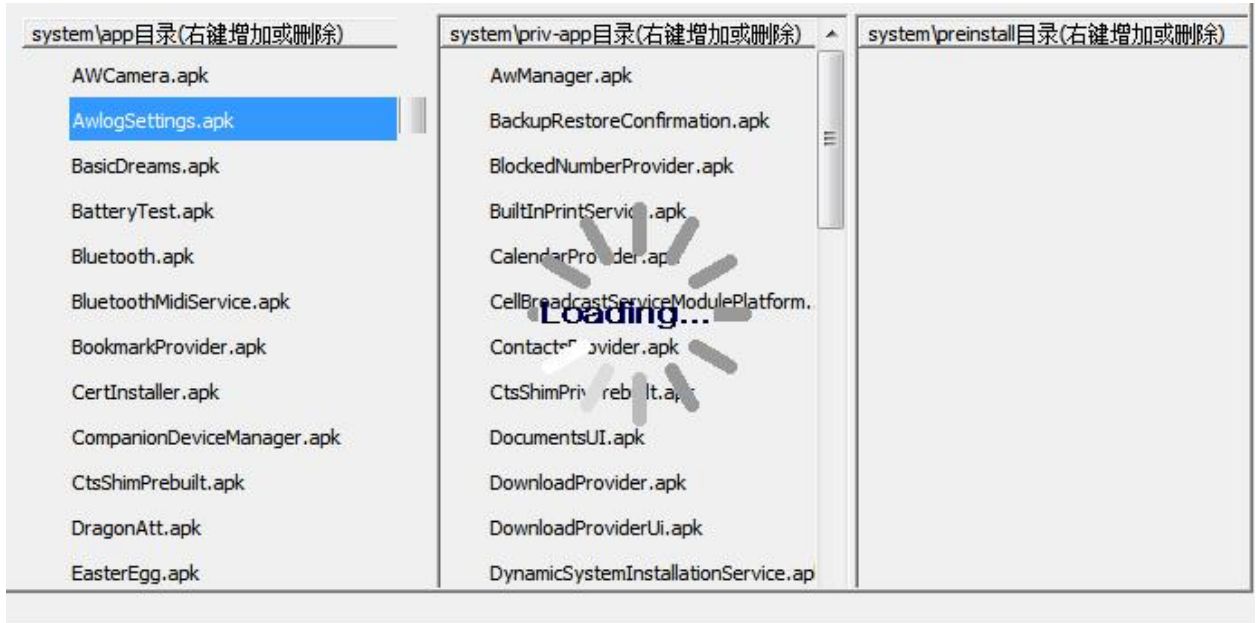


图 3-18 解压完 APK 显示示意图



### 说明

预装 apk 说明:

#### (1) 什么叫内置 APK、自定义 APK

内置 APK, 即安装在 system 分区的 APK, 用户不可删除。通过“章节 3.3.3APK”的方法添加内置 APK。

自定义 APK, 即安装在 data 分区的 APK, 用户可手动删除。直接手动在小机上安装, 然后通过工具系统克隆即可。

#### (2) 如何预装一次性 APK

一次性 APK, 即预装后恢复出厂设置 (或 recover 恢复) 时该 APK 不会再有。适合如 DragonFire 测试 APK 预装使用。使用一次性方式, 可在 APK 使用完后, 点击恢复出厂设置, 该 APK 就不存在。

预装一次性 APK 只能使用内置 APK 方式, 方法简单, 在 APK 文件名后面添加“646212C46B774F95BBBE2F9CFF32797”这一字符串, 然后使用内置 APK 定制预装, 则固件烧录进小机时安装会生效, 而在 recovery 或者恢复出厂设置的时候会把这类 APK 删除掉。

### (3) 不是所有的 APK 都可以内置

不是所有的 APK 都可以内置，这和 APK 本身有关。主要是因为有些带 lib 库的 APK，没有自动识别 lib 库存放路径。

### (4) 预装大型 APK 应用

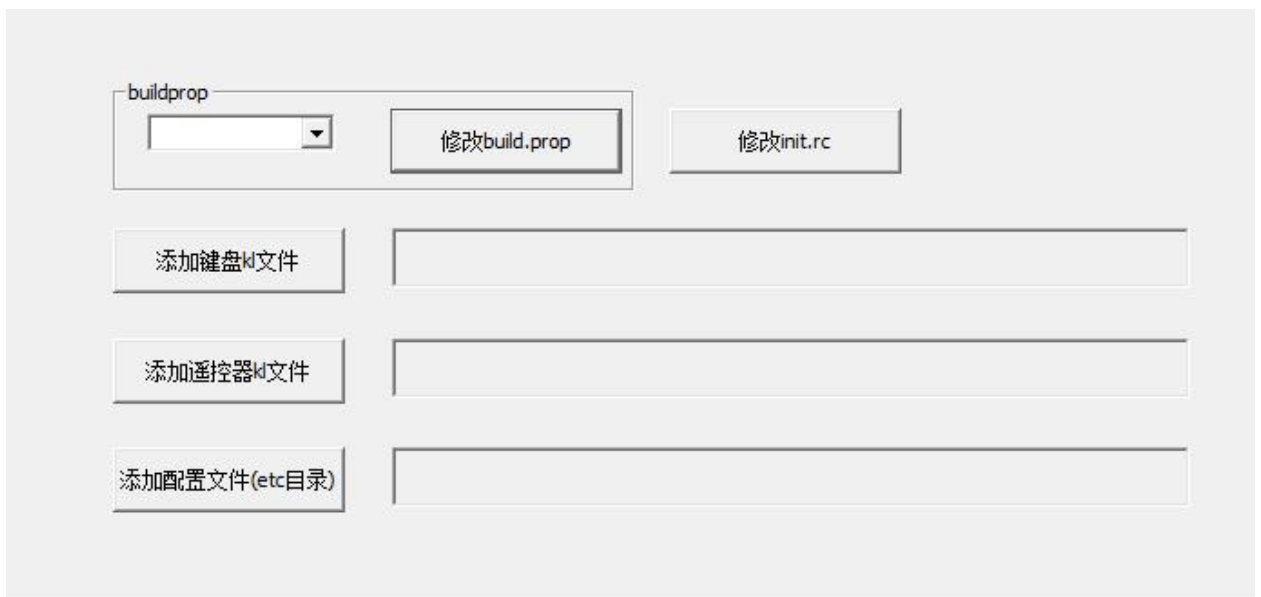
游戏、GPS 等大型 APK 应用，推荐预装为自定义 APK 方式，以免 system 分区空间不够。

带有独立数据包的 APK，需要把独立数据包放在 SD 卡里，可以通过配置 demo 文件的方式，把独立数据包添加进固件包，固件烧写进小机后，所有 demo 文件都会放置在 SD 卡内，所以独立数据包也会放置在卡里。但不是所有带独立数据包的游戏都可以通过这种方式预装，这和 APK 本身有关系，有的 APK 必须要自建存放独立数据包的文件夹。

## 3.3.4 高级设置

进入高级设置界面：

图 3-19 高级设置示意图



(1) 修改 build.prop: 对于 android 固件，系统信息在 build.prop 里面，不同版本的 android，build.prop 有多份，对于 10.0 以及以上的平台，build.prop 在 oem 分区，需打上 sdk 的补丁，然后直接修改 oem 分区的 build.prop 即可。10.0 以下固件，选择相应分区的 build.prop 进行修改即可。

图 3-20 build.prop 文件示意图

```

build.prop
1 #####system start#####
2
3
4 # begin common build properties
5 # autogenerated by build/make/tools/buildinfo_common.sh
6 ro.system.build.date=Thu Mar 25 17:00:32 CST 2021
7 ro.system.build.date.utc=1616662832
8 ro.system.build.fingerprint=Allwinner/ceres_b3/ceres-b3:11/RP1A.201005.006/DanielWei03251639:userdebug/t
9 ro.system.build.id=RP1A.201005.006
10 ro.system.build.tags=test-keys
11 ro.system.build.type=userdebug
12 ro.system.build.version.incremental=eng.Daniel.20210325.164143
13 ro.system.build.version.release=11
14 ro.system.build.version.release_or_codename=11
15 ro.system.build.version.sdk=30
16 ro.product.system.brand=Allwinner
17 ro.product.system.device=ceres-b3
18 ro.product.system.manufacturer=Allwinner
19 ro.product.system.model=QUAD-CORE A133 b3
20 ro.product.system.name=ceres_b3
21 # end common build properties
22 # begin build properties
23 # autogenerated by buildinfo.sh
24 ro.build.id=RP1A.201005.006
25 ro.build.display.id=ceres_b3-userdebug 11 RP1A.201005.006 eng.Daniel.20210325.164143 test-keys
26 ro.build.version.incremental=eng.Daniel.20210325.164143
27 ro.build.version.sdk=30
28 ro.build.version.preview_sdk=0
    
```

(2) 修改 init.rc: 可往 init.xxx.rc 添加命令, 同样 10.0 以上平台 rc 文件在 oem 分区, 目录是 etc/init\, 需打上 sdk 的补丁。如图 3-21 所示。

如在 on init 下新增语句: mkdir /mnt/test 0700 root root, 通过 adb 或串口 cd mnt 命令切换目录至 mnt 路径, ls 命令查看该目录 list, 可看到 test 文件夹。

图 3-21 rc 文件示意图

```

init.sun50iw10p1.rc
1 import /vendor/etc/init/hw/init.common.rc
2 import /vendor/etc/init/hw/init.ram.rc
3
4 # import module rc.
5 import /vendor/etc/init/hw/init.sun50iw10p1.usb.rc
6
7 on early-init
8     chown system system /proc/bootevent
9     chmod 0666 /proc/bootevent
10    write /proc/bootevent "INIT:early-init"
11
12 on init
13    # Set permissions for persist partition
14    mkdir /persist 0771 system system
15    mkdir /firmware 0771 system system
16
17    # Support legacy paths
18    symlink /sdcard /mnt/sdcard
19    symlink /sdcard /storage/sdcard0
    
```

(3) 键盘与遥控器 kl 文件

可添加键盘与遥控器 kl 文件到 oem 分区, 目录是 usr\keyboard, 需打上 sdk 的补丁。如图 3-22 所示。

图 3-22 键盘与遥控器 k1 文件示意图



## (4) 添加配置文件 (etc 目录)

可自行添加配置到 oem 分区, 目录下 etc\。

oem 分区结构如下:

图 3-23 oem 分区结构示意图

app	2021/6/19 10:02	文件夹	
etc	2021/6/19 9:21	文件夹	
media	2021/6/19 9:22	文件夹	
preinstall	2021/6/19 9:21	文件夹	
priv-app	2021/6/19 9:21	文件夹	
usr	2021/6/19 9:21	文件夹	
build.prop	2021/6/19 9:22	PROP 文件	10 KB

## 3.3.5 Rootfs

文件系统可通过“打开 rootfs 目录”按钮打开, 同时左边树形结构展开了目录结构, 如图 3-24 所示。

图 3-24 rootfs 目录结构示意图



选中左边树形结构中的文件, 在右边可显示文件大小和路径, 同时可点击 Replace 进行文件替换如替换 stalogo1.bin, 对于 bmp 图片文件会在右边预览显示出来。

**注意**

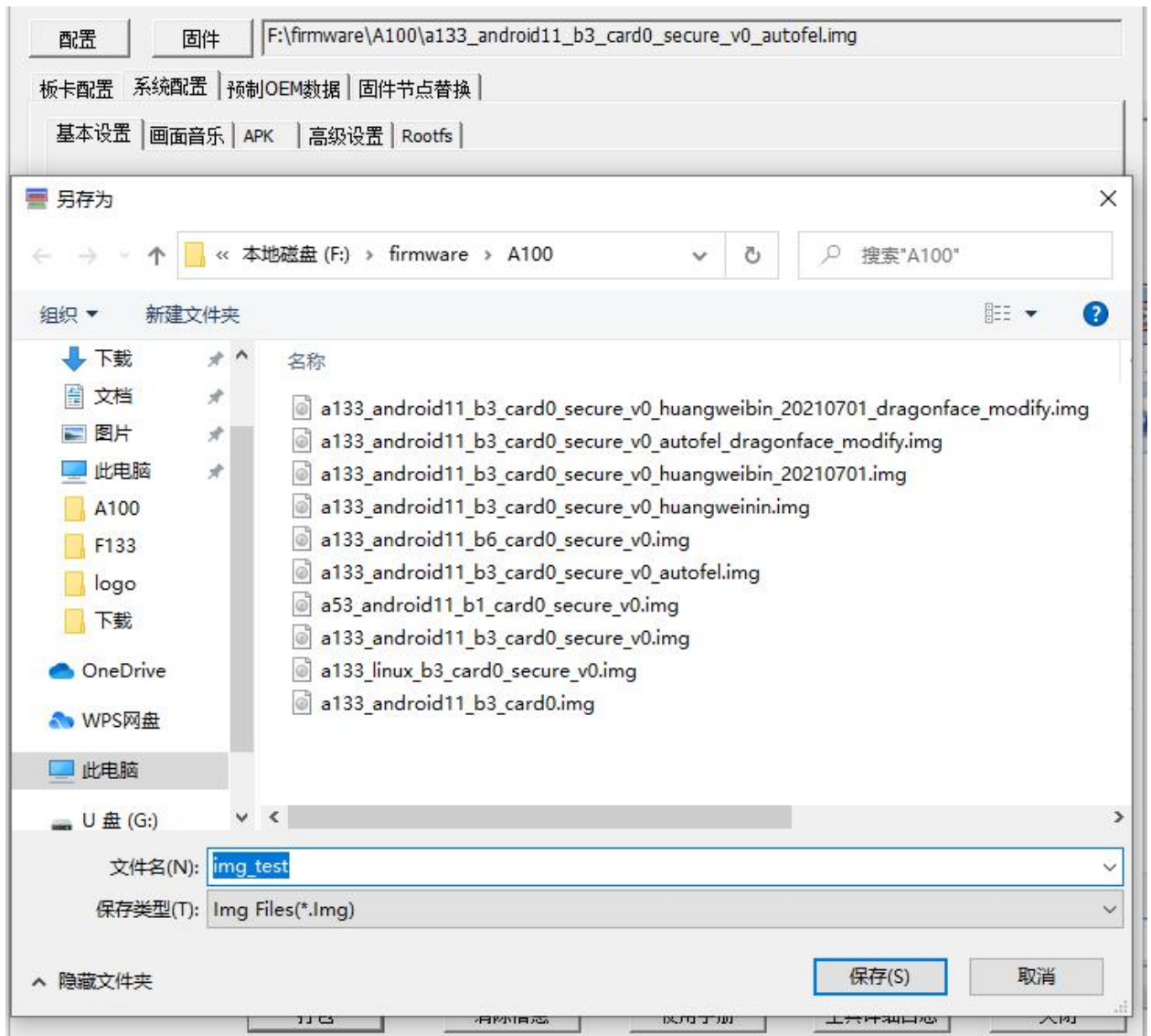
① 安卓固件: android Q 及以上平台请参考目录下 patch\androidQ&R 下 sdk 补丁, android Q 以下安全固件请参考目录下 patch\安全方案 androidO&P 下 sdk 补丁。

② 文件系统在“系统配置->Rootfs”。

### 3.3.6 保存文件

- (1) 确认修改完毕后, 点击工具的“打包”按钮;
- (2) 选择保存路径;
- (3) 输入新建 img 文件的名称, 如图 3-25 所示;

图 3-25 保存文件示意图



- (4) 点击“保存”按钮

## 3.4 预制 OEM 数据

此功能同 oem 工具的功能, 就是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。

### 3.4.1 oem 数据界面设置

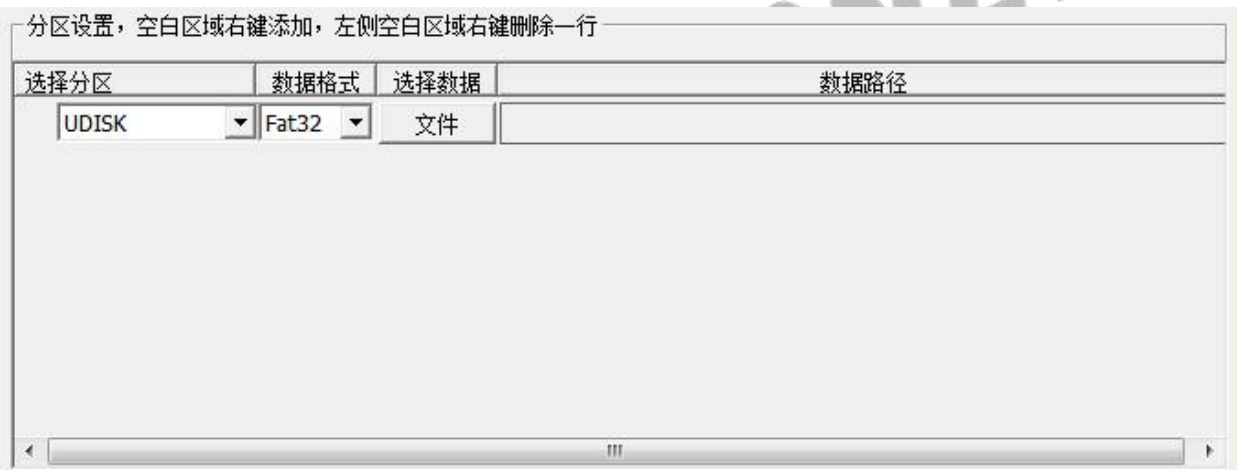
同样需要先配置勾选预制 OEM 数据，然后在选择固件进行加载。

图 3-26 预制 OEM 数据配置示意图



(1) 分区设置。如图 3-27 所示。

图 3-27 分区设置示意图



1、当选择了固件固件加载成功后分区设置列表才可以操作，默认添加了 udisk 分区。

#### 注意

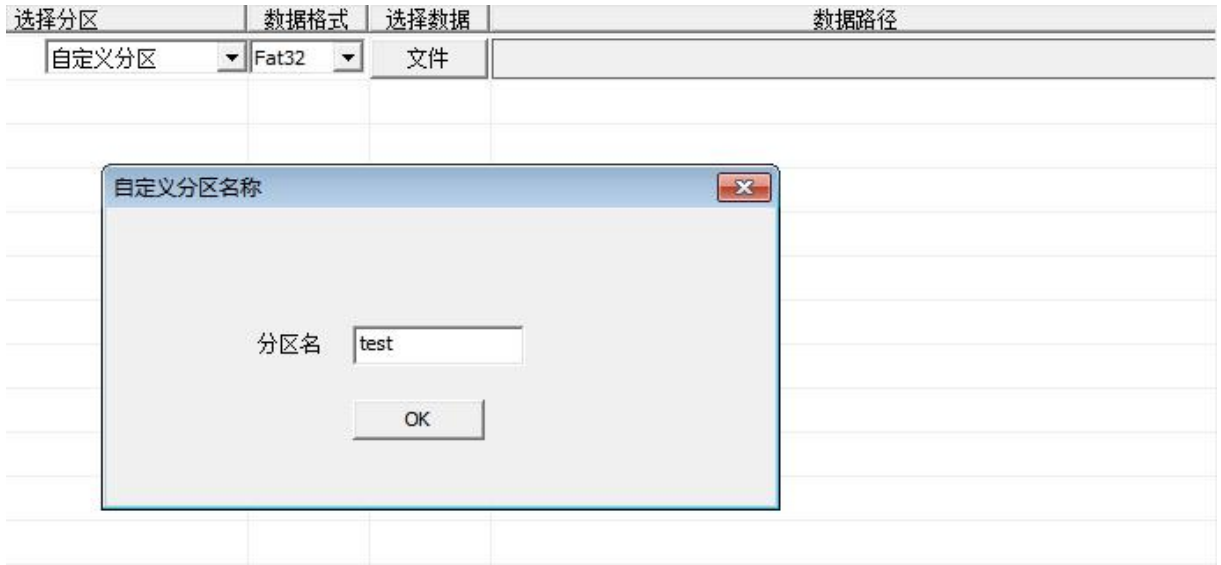
- android R 以及之后 UDISK 分区名称叫做 userdata 分区名，即 UDISK=userdata。
- userdata/udisk 分区打包的数据需要用 phoenixusbpro 多路量产工具进行烧写。

2、点击列表空白区域右键可以添加分区，点击左侧空白区域右键一行可以删除。

3、支持 UDISK、media\_data 分区以及自定义分区打包数据。

4、选择自定义分区时会弹出需要输入分区名称，注意分区名称不能是 UDISK 和 media\_data。

图 3-28 自定义分区设置示意图



5、打包时列表里面不能有重复的分区名称，如有两个 UDISK，工具会报错。

6、android Q 及以上平台 udisk 分区打包文件系统建议选择 f2fs，media\_data 分区建议选择 fat32

(2) 分区格式说明。

默认支持 file、fat32、ext4、f2fs 四种文件系统格式。

1、android Q 以前的平台 data 分区名称叫做 UDISK，打包到此分区文件系统默认是 fat32，系统启动后默认把此分区的数据挂载到 oem 目录，可以进入 oem 目录查看打包后的数据。

2、android Q 及以后的平台 data 分区名称叫做 userdata，打包到此分区文件系统默认是 f2fs，系统启动后默认把此分区的数据挂载到 data 目录，可以进入 data 目录查看打包后的数据。

3、也可以在 adb shell 下手动 mount 数据以验证正确性：

①当打包到 UDISK（android R 是 userdata）分区的时候

```
mount -t vfat dev/block/by-name/UDISK mnt/tmp
```

```
mount -t ext4 dev/block/by-name/UDISK mnt/tmp
```

```
mount -t f2fs dev/block/by-name/UDISK mnt/tmp
```

②当打包到 media\_data 分区的时候

```
mount -t vfat dev/block/by-name/media_data mnt/tmp
```

```
mount -t ext4 dev/block/by-name/media_data mnt/tmp
```

```
mount -t f2fs dev/block/by-name/media_data mnt/tmp
```

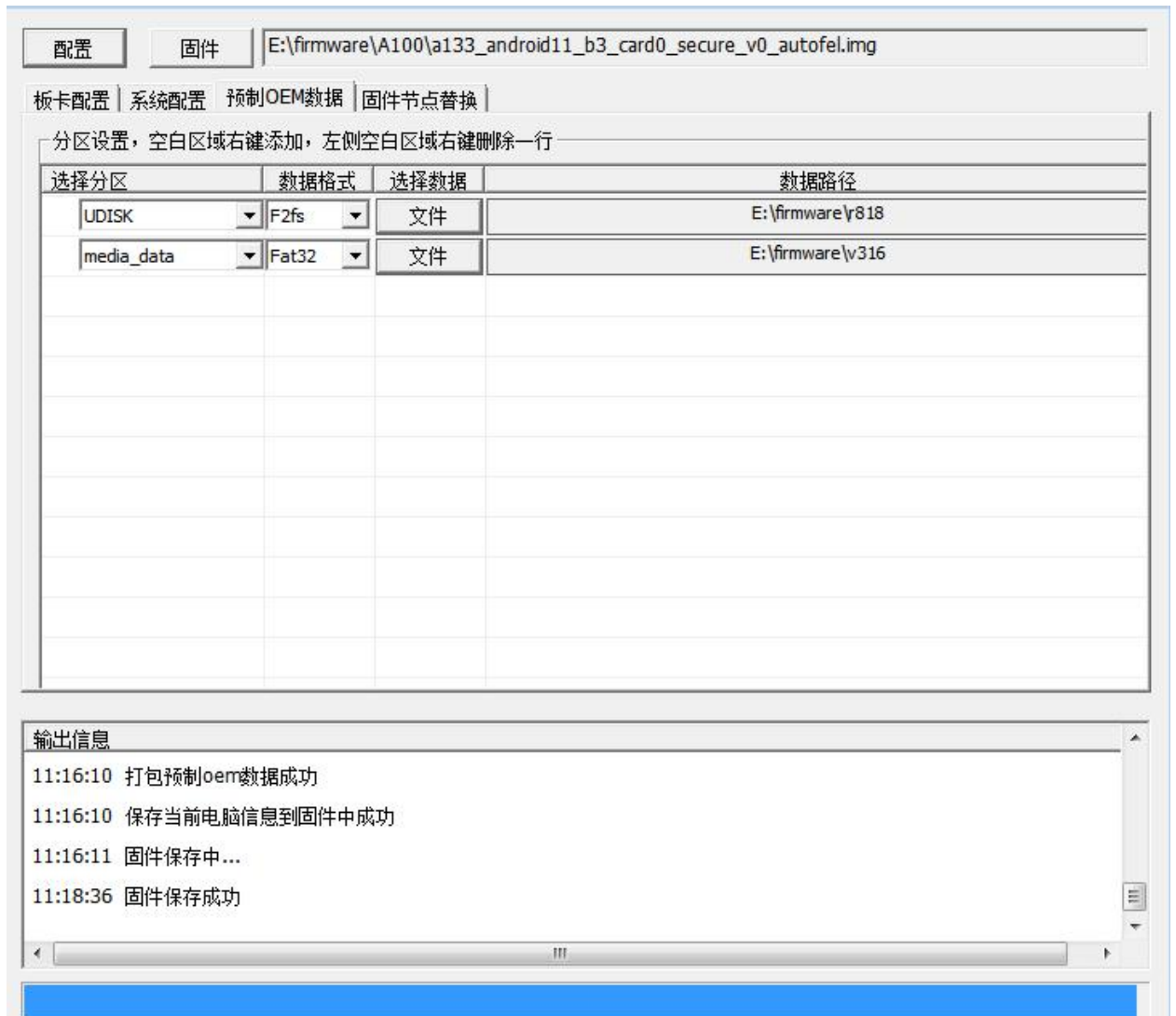
③自定义分区也可以在 adb shell 下验证数据正确性：

```
mount -t vfat dev/block/by-name/分区名 mnt/tmp
```

### 3.4.2 打包 oem 数据

(1) 打包，点击打包按钮，进行打包。当提示信息如下时，表示打包结束，如图 3-29 所示界面。

图 3-29 打包 oem 数据示意图



### 3.4.3 打包多用户固件配置

若多用户固件数据打包到 media\_data 分区，需要在 SDK 里进行相应的配置，具体步骤如下。

#### 1、增加 media\_data 分区

修改 `lichee/linux-3.4/pack/chips/sun8iw3p1/configs/android/polaris-xx/sys_partition.fex`, 如图 3-30 所示。

图 3-30 增加 media\_data 分区示意图

```

;----->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**

请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` 以创建 `/mnt/extsd1` 目录, 如图 3-31、图 3-32 所示。

图 3-31 创建/mnt/extsd1 目录示意图

```

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
    
```

图 3-32 格式化 media\_data 分区示意图

```

# 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/fsab.sun8i`

挂载 `media_data` 分区。在后面增加此行, `/devices/virtual/block/nandj` 是步骤 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, 如图 3-33 所示。

图 3-33 增加配置示意图

```

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

4、修改 `android/devices/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` 挂载到外部 SD 卡 1, 可以通过文件管理器看到此目录, 插入外置的 SD 卡, 默认为 `mnt/extsd`, 如图 3-34 所示。

图 3-34 挂载外部 SD 卡示意图

```

<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:mtpRootPath=""
    android:maxFileSize="0"/>
    
```

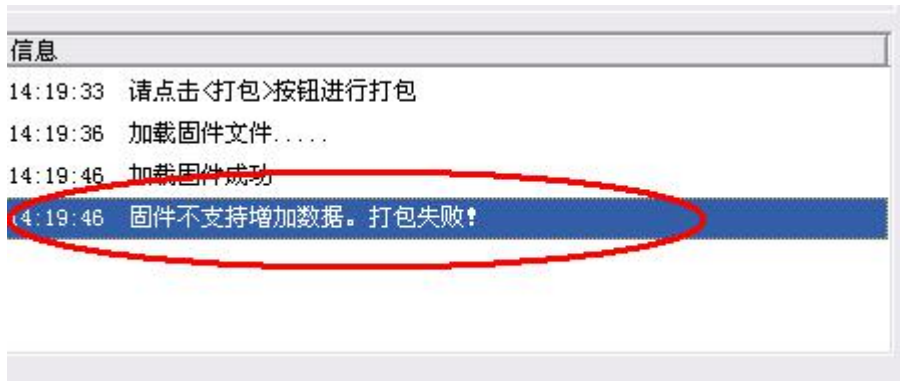
#### 5、恢复出厂设置时选择是否格式化打包进去的数据（此功能非必须，根据需求）

如果要添加此功能，请咨询开发或者 fae 工程师。

#### 3.4.4 预制 oem 数据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图 3-35 所示。

图 3-35 固件不支持打包数据导致失败示意图



#### ⚠ 注意

- 1、打包数据时必须确认待打包的数据文件夹内没有被打开的文件或者被其他程序占用，否则可能出现目标固件无法烧录或者丢失数据等问题。
- 2、请预先预留足够的 C 盘空间以防止数据打包失败，C 盘可用空间至少是打包数据的两倍，如打包 5G 数据，那么 C 盘剩余容量至少 10G。

## 3.5 固件节点替换

此功能是方便替换固件里面的节点生成所需的新固件。

### 3.5.1 固件节点替换界面设置

同样配置里面只勾选“固件节点替换”，然后在选择固件进行加载。

图 3-36 固件节点替换配置示意图



### ⚠ 注意

替换 sys\_config、sys\_partiton、sunxi、uboot、fes1、bootpkg、sunxi\_mbr 这些跟启动相关的节点时，请只打开固件节点替换配置，其他配置不要打开，否则数据会被覆盖。

### 3.5.2 节点文件替换设置

加载完固件后在列表空白区域右键添加然后设置需要替换的节点，如图 3-37 所示。



图 3-38 打包固件节点替换示意图

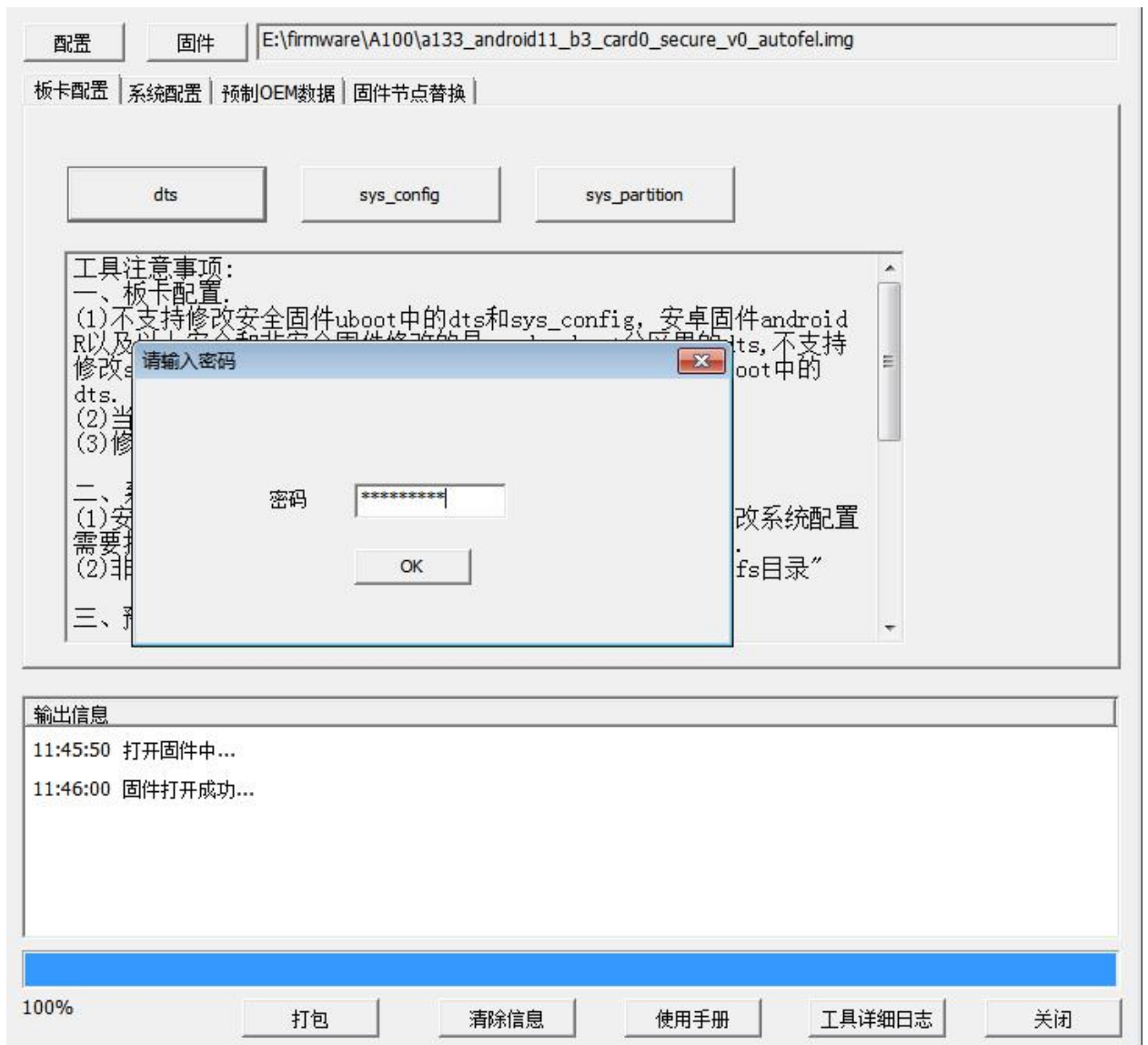


### 3.6 设置 key(保护固件)

当配置里面选择了"设置 key 保护固件", 保存固件时, 需要输入密码, 下次加载新固件时需要输入保存的密码, 以达到保护固件的目的。

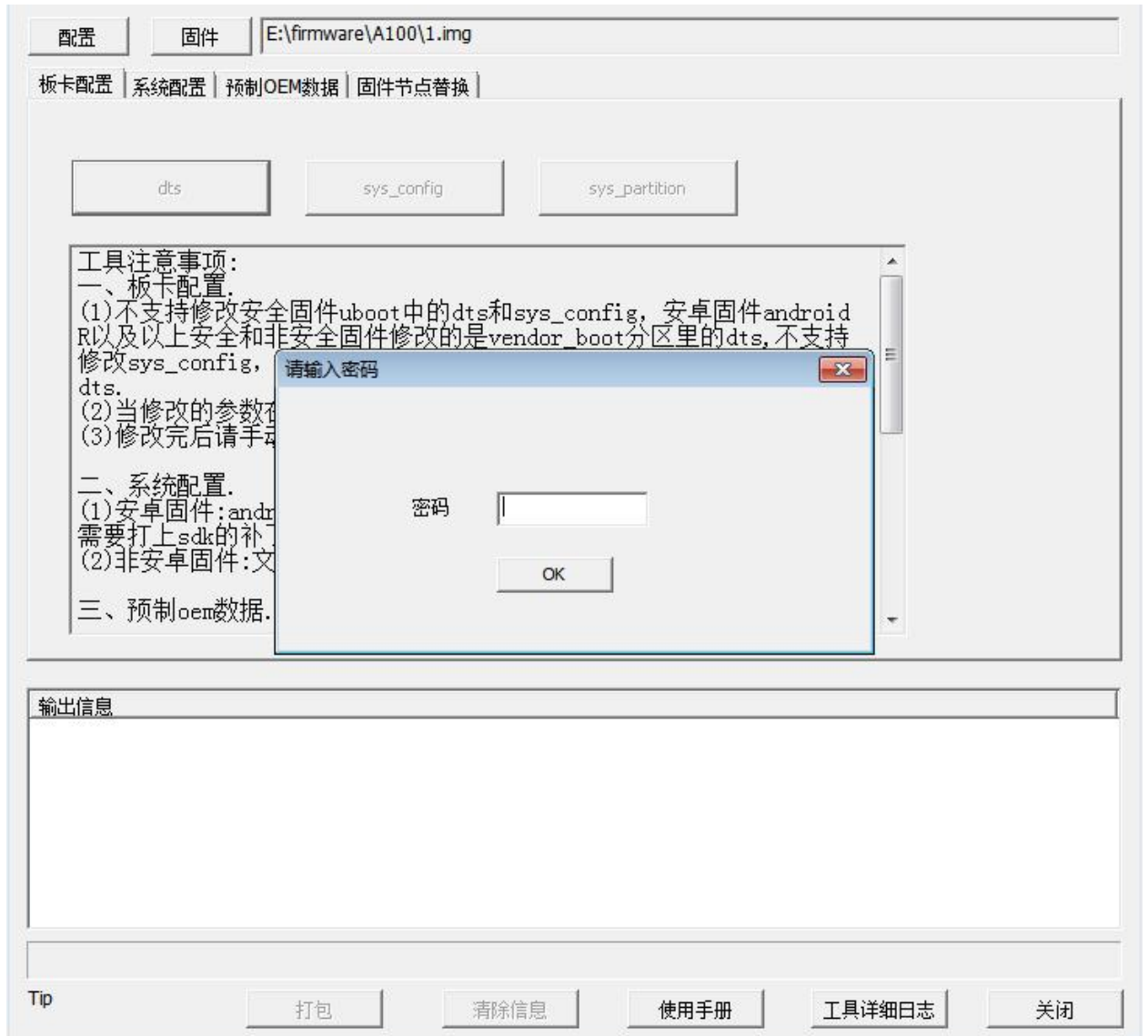
(1) 设置 key 可以跟任何其他的模块一起使用或者单独使用。当勾选了这个的时候, 打包的时候会弹出输入密码框。如图 3-39 所示。

图 3-39 输入密码示意图



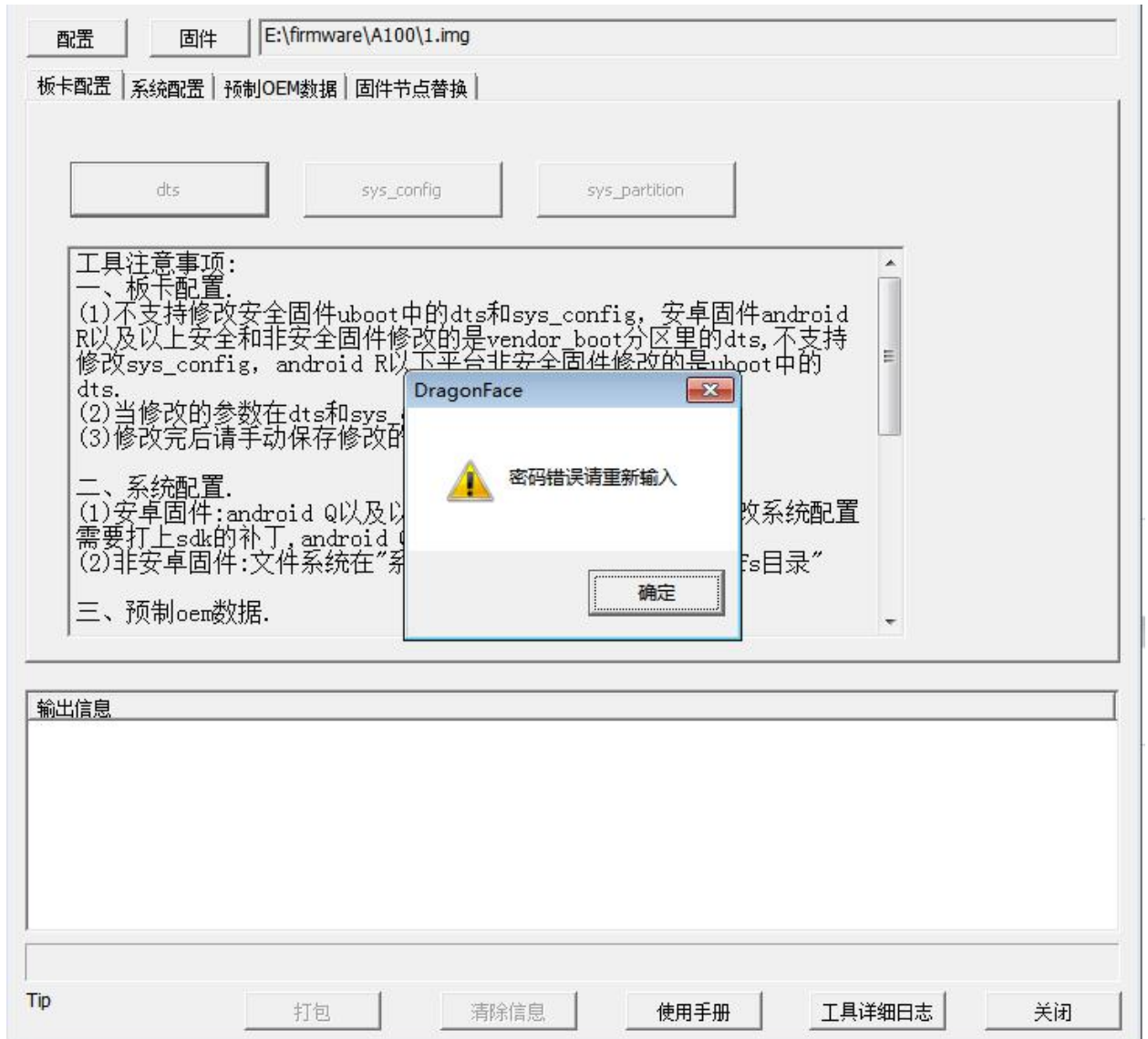
(2) 导入有密码的固件的时候同样会弹出输入密码的显示框。如图 3-40 所示。

图 3-40 有密码的固件显示密码输入示意图



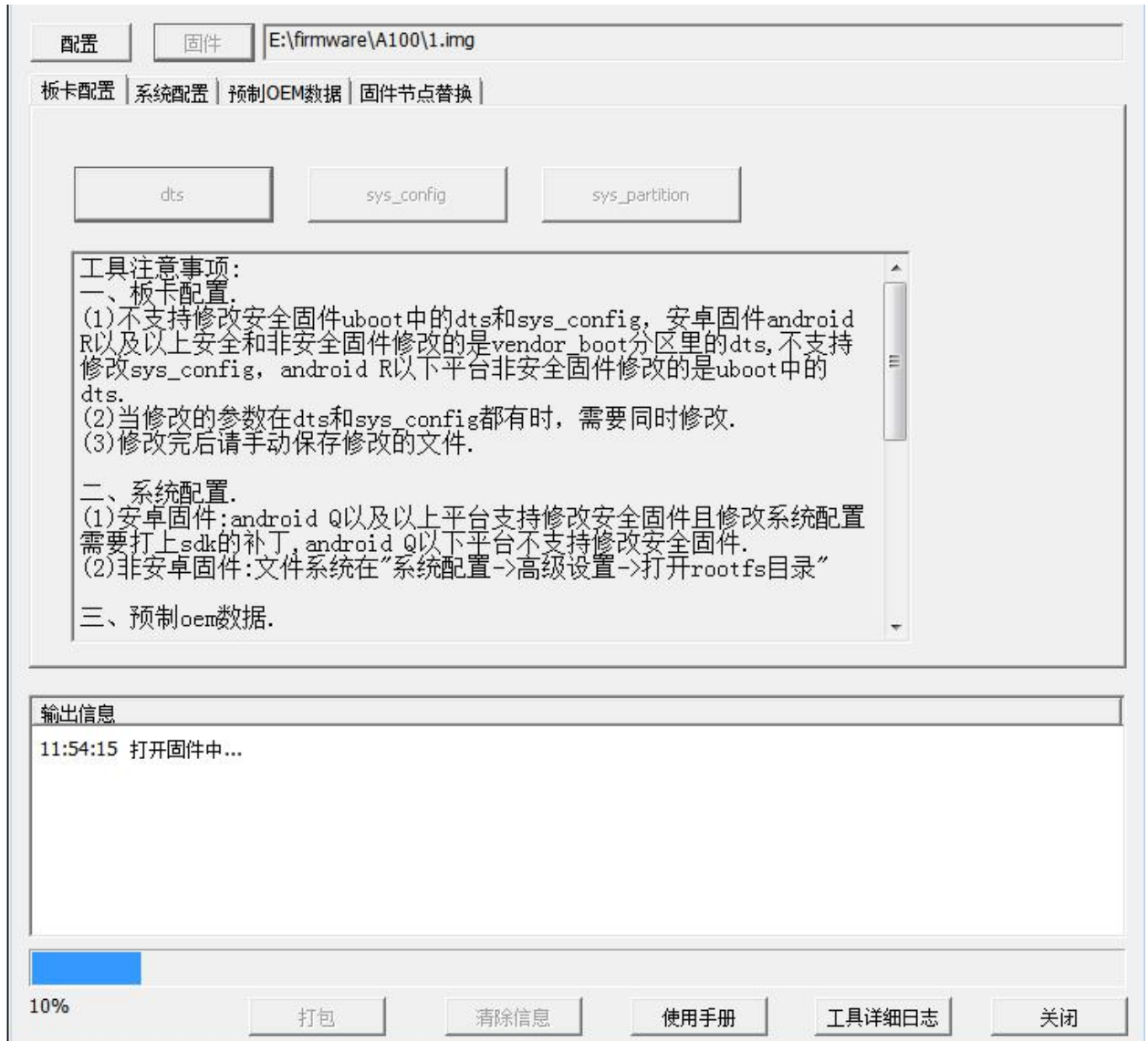
(3) 当密码错误的时候将无法使用工具进行修改。如图 3-41 所示，密码错误时的提示。

图 3-41 密码错误提示示意图



密码正确的时候才可以修改，如图 3-42 所示。

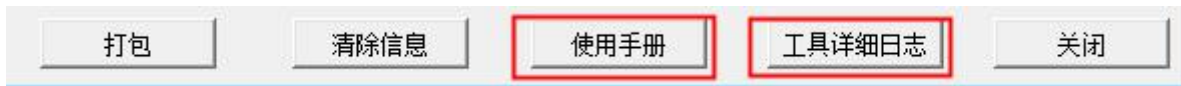
图 3-42 密码正确后的示意图



## 4 其他事项

(1) 点击工具界面上的“使用手册”按钮可以直接打开使用手册。

(2) 如果使用过程中碰到工具报错或者无法进行修改时, 请点击界面上的“工具详细日志”, 在目录下找到当前最新的日志发给工程师进行分析。



## 著作权声明

版权所有©2020 珠海全志科技股份有限公司。保留一切权利。

本档及内容受著作权法保护，其著作权由珠海全志科技股份有限公司（“全志”）拥有并保留一切权利。

本档是全志的原创作品和版权财产，未经全志书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本档内容的部分或全部，且不得以任何形式传播。

## 商标声明

、、、 Allwinner Technology（不完全列举）均为珠海全志科技股份有限公司的商标或者注册商标。在本档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

## 免责声明

您购买的产品、服务或特性应受您与珠海全志科技股份有限公司（“全志”）之间签署的商业合同和条款的约束。本档中描述的全部或部分产品、服务或特性可能不在您所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，全志概不负责。

本档作为使用指导仅供参考。由于产品版本升级或其他原因，本档内容有可能修改，如有变更，恕不另行通知。全志尽全力在本档中提供准确的信息，但并不确保内容完全没有错误，因使用本档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，全志概不负责。本档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本档未以明示或暗示或其他方式授予全志的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。全志不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。全志不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。

