Git 推送编译说明文档

目录

[一、新增仓库说明 2](#_Toc46493169)

[1、device 目录说明 2](#_Toc46493170)

[2、Modem 目录说明 3](#_Toc46493171)

[3、Script 目录说明 3](#_Toc46493172)

[4、工程目录说明 4](#_Toc46493173)

[5、工程目录下的bin和配置文件目录说明 4](#_Toc46493174)

[6、flash.cfg文件说明 5](#_Toc46493175)

[二、脚本使用说明 6](#_Toc46493176)

[1、修改Modem配置文件 7](#_Toc46493177)

[2、编译打包脚本build\_pac.sh 7](#_Toc46493178)

[3、签名脚本avb\_sign\_modem\_v3.sh 8](#_Toc46493179)

[4、编译执行案例 9](#_Toc46493180)

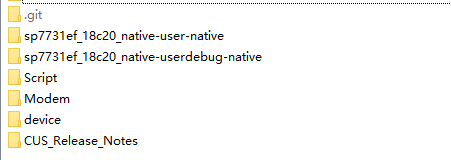
[附录 10](#_Toc46493181)

# 一、新增仓库说明

git 推送会比原代码新增一个仓库，用于存放提供给客户的 Modem 文件、编译脚本、AP属性包、和编译需要用到的一些配置文件。

仓库地址：./vendor/sprd/release/IDH

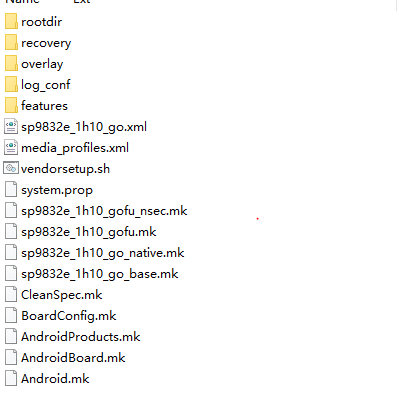
仓库下内容截图举例：



## 1、device 目录说明

目录地址：./vendor/sprd/release/IDH/device

目录截图示例：



* 这个目录不是必有的，如果device仓库在manifest下是开源的，那么Git推送代码根目录下就不存在device目录。
* 如果Git推送代码根目录下存在device目录：

（1）这个目录和代码根目录下的 device 文件是一样。

（2）如果需要修改device 下内容请修改./vendor/sprd/release/IDH/device 下的文件。

（3）device 目录是AP工程编译的一些mk等配置文件，具体见上图示例。我们的编译脚本会自动copy 此文件夹到code 根目录下。

* 如果Git推送代码根目录下不存在device目录：

1. 不影响编译。
2. 如果需要修改device 下内容请修改根目录下的device文件。

## 2、Modem 目录说明

目录地址：./vendor/sprd/release/IDH/Modem

● 展锐推送的Modem文件，以分支版本号区分，版本号下面以工程区分。

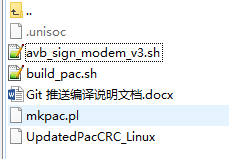
● 展锐推送的Modem文件均未签名。

● 该目录下包含tools 工具，和对应Modem 的说明文档。

## 3、Script 目录说明

目录地址：./vendor/sprd/release/IDH/Script

截图举例：

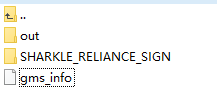


* .unisoc 文件存放的是git 推送我们内部获取的版本地址，和commit 信息。
* build\_pac.sh ，文件是编译和打包脚本。
* mkpac.pl & UpdatedPacCRC\_Linux 是打pac 包用的脚本和校验文件。
* Git 推送编译说明文档.docx 是本文档。
* avb\_sign\_modem\_v3.sh 是modem 签名脚本。

## 4、工程目录说明

目录地址：./vendor/sprd/release/IDH/”SP\*”

截图举例：

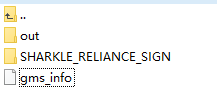


* SP\* 这一类是推送的AP工程目录。
* out是我们闭源代码编译生产的属性包文件，编译脚本运行时会拷贝到根目录下。
* 上面截图 SHARKLE\_RELIANCE\_SIGN 是打pac包需要用到的bin和配置文件的目录。
* 打包的配置文件是 SHARKLE\_RELIANCE\_SIGN/flash.cfg 用途：打包的分区和加载的img 文件名。
* SHARKLE\_RELIANCE\_SIGN目录下还包含有打包用到的GPS bin、WCN bin、开机图片。
* gms\_info 是我们编译脚本设置gms size、指定gms .mk(只有在GMS工程才会有)，编译脚本会自动解析文件。

## 5、工程目录下的bin和配置文件目录说明

目录地址：./vendor/sprd/release/IDH/”SP\*”/

截图举例：



即上面截图中的“SHARKLE\_RELIANCE\_SIGN”，进入到截图中的“SHARKLE\_RELIANCE\_SIGN”中，截图示例如下：



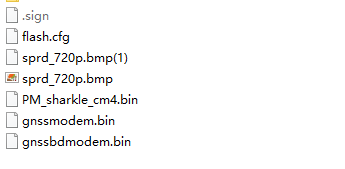
* .sign文件是打包时自动签名Modem用的配置文件，具体配置方法见“修改Modem配置文件”章节。
* flash.cfg文件是打包的分区和加载的img 文件名。
* sprd\_720.bmp(1)是fast boot Logo，sprd\_720p.bmp是boot Logo，即开机图片。
* PM\_sharkle\_cm4.bin是WCN bin。
* gnssmodem.bin是GPS GL bin，gnssbdmodem.bin是GPS BD bin。

## 6、flash.cfg文件说明

文件地址：./vendor/sprd/release/IDH/”SP\*”/“SH\*”/

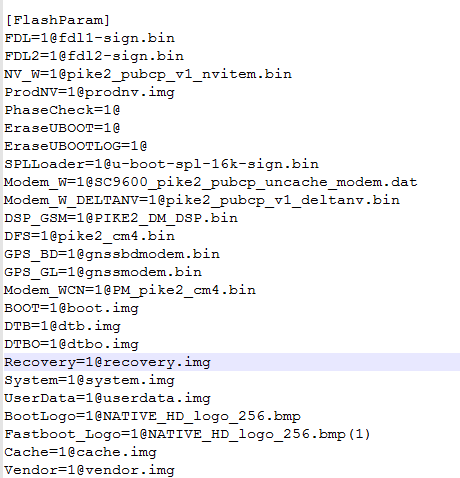
Eg：./vendor/sprd/release/IDH/sp9832e\_1h10\_oversea-user-native/SHARKLE\_9\_LL\_SIN/flash.cfg

截图举例：



* 此文件的用途是打包的分区和加载的img 文件名。
* 如果修改了make编译文件，或者修改了生成的AP img名称，或者一个分区打包2个img，出现上述情况，就需要重新配置flash.cfg，配置规则如下：

1. 打开flash.cfg文件，截图举例：

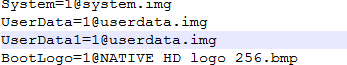


找到上图中的[FlashParam]位置。

（1）如果修改了make编译文件，或者修改了生成的AP img名称，则修改对应的“=”后面的img名称为本地实际的img名称。

（2）如果一个分区打包2个img，则在其中一个img的“=”前加1，

截图举例，截图中的UserData和UserData1



# 二、脚本使用说明

我们推送包含编译打包脚本和Modem bin 签名脚本，编译打包脚本可以支持只编译，只打包，只OTA编译，编译+打包，编译+OTA编译。

脚本地址：./vendor/sprd/release/IDH/Script

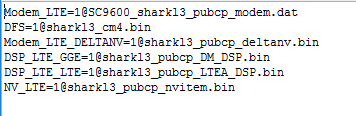
## 1、修改Modem配置文件

* 配置文件名称为.sign，位置在./vendor/sprd/release/IDH/ "sp\*" 目录/”SH\*”目录/.sign
* 此配置文件作用是打包时自动签名使用。

Eg: ./vendor/sprd/release/IDH/s9863a3h10\_Natv-user-native/SHARKL3\_9863A3\_9/.sign

● 配置方式如下：

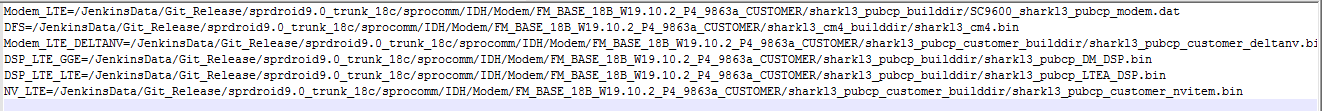
(1)打开.sign文件，看到如下图展示



注意：文件里的Modem类别因项目而异。

(2) 将文件中各个“=”后面的名称改为对应的Modem所在位置的绝对路径，

举例示例(配置有点长，请放大图片查看)：



上图中“=”后面改为对应的modem、LTE DSP、GMS DSP、AG DSP、cm4 bin等在代码中的绝对路径。

● 打包时脚本会从配置文件中配置的绝对路径中拷贝各个Modem并自动签名，OTA编译时也会用到此配置文件。

● 只需要配置.sign中列出了的Modem，没有的就不需要配置。

* 如果包含多个工程，请将所有工程下的.sign文件都配置好。

## 2、编译打包脚本build\_pac.sh

* 脚本需要在 ./vendor/sprd/release/IDH/Script 目录下执行，并加执行权限。
* 脚本执行方式：./build\_pac.sh -a "参数a" -b "参数b" -c "参数c"
* -a & -b 是必填项，-c 可选。-c 参数只对编译有影响，对pac 没有影响。
* 参数-a 是 ./vendor/sprd/release/IDH 目录下面 "sp\*" 目录名称

Eg：sp9832e\_1h10\_reliance-user-native

* 参数-b ALL|all:编译和打包； BUILD|build:只编译；PAC|pac:只打包；OTA|ota:只OTA编译；BOTA|bota:编译和OTA编译。
* 参数-c 是编译运营商工程，如果有多个用英文","隔开。
* 生成的pac 包在"参数-a"目录内除out目录外的子目录下。
* 编译OTA生成的target\_files压缩包和ota压缩包在./out/target/product/项目工程/”OTA\_\*”目录下。

Eg：./out/target/product/sp7731e\_1h10/OTA\_userdebug\_gms/sp7731e\_1h10\_native-ota-PIKE2\_9\_Uncache.zip

./out/target/product/sp7731e\_1h10/OTA\_userdebug\_gms/sp7731e\_1h10\_native-target\_files-PIKE2\_9\_Uncache.zip

举例调用方式：./build\_pac.sh –a sp9832e\_1h10\_reliance-user-native –b BUILD –c \*\*\*

## 3、签名脚本avb\_sign\_modem\_v3.sh

● build\_pac.sh打包已实现Modem自动签名，如果想要手动签名，按下面步骤手动签名：

（1）脚本需要在 ./vendor/sprd/release/IDH/Script 目录下执行，并加执行权限。

（2）签名调用方式 ./vendor/sprd/release/IDH/Script/avb\_sign\_modem\_v3.sh \

-c $IMG\_DIR/out/target/product/\*\*\*/\*\*\*.xml \

-m ${MODEM\_MODEM} \

-l ${MODEM\_DSP} \

-g ${MODEM\_G\_DSP} \

-a ${MODEM\_DSP\_AG} \

-d ${MODEM\_DFS} \

-x ${DSP\_LTE\_CDMA}\

-t ${MODEM\_DSP\_W\_G}\

-w ${MODEM\_DSP\_W\_L}

● -c 是必填项，且必须是out 地址下.xml 文件的绝对地址。

● -m 是必填项，是modem .bin 文件或者是 .dat 文件。

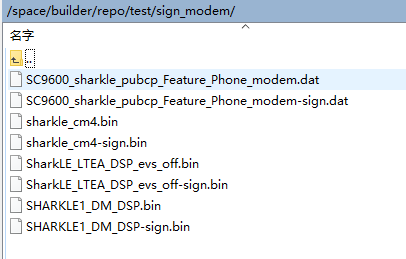
● –m –l –g –a –d -x –t -w分别是：modem、LTE DSP、GMS DSP、AG DSP、cm4 bin、CDMA、Modem\_NR\_V3PHY、Modem\_NR\_PHY。

● 如果没有 AG DSP 那么-a 就不填，其他-x –t -w的也是一样，没有就不填。

● 签名好的img 在./sign\_modem 目录下。

举例调用方式：./avb\_sign\_modem\_v3.sh -c /space/builder/repo/test/out/target/product/sp7731ef\_19c10/sp7731ef\_19c10.xml -m /sharkle\_pubcp\_Feature\_Phone\_builddir/SC9600\_sharkle\_pubcp\_Feature\_Phone\_modem.dat -g /sharkle\_pubcp\_Feature\_Phone\_builddir/SHARKLE1\_DM\_DSP.bin -l /sharkle\_pubcp\_Feature\_Phone\_builddir/SharkLE\_LTEA\_DSP\_evs\_off.bin -d /sharkle\_cm4\_builddir/sharkle\_cm4.bin

举例生成的签名bin：



## 4、编译执行案例

* 目前的脚本流程如下：

1. 修改Modem配置文件，即修改.sign文件。  
2. 执行./build\_pac.sh -a sp9832e\_1h10\_gofu-userdebug-gms –b ALL 进行编译和打包。

3．如果需要编译OTA，有如下方式：

(1)编译+OTA编译: 执行./build\_pac.sh -a sp9832e\_1h10\_gofu-userdebug-gms –b BOTA

(2) 已编译，现只需要OTA编译: 执行./build\_pac.sh -a sp9832e\_1h10\_gofu-userdebug-gms –b OTA

# 附录

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Owner | Notes |
| 1.0 | 2019-01-28 | fufa.li@unisoc.com | 创建 |
| 1.1 | 2019-07-31 | Apple.wu@unisoc.com | 修订 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |