“葫芦娃”项目的客制化以及版本发布

1. 一般项目客制化

这里所说的一般项目客制化是指PRODUCT\_NAME\PRODUCT\_DEVICE不相同的版本客制化。

因为MTK的客制化文件夹名是用PRODUCT\_NAME来命名的，所以PRODUCT\_NAME不一样就需要重新创建新的目录。

参考文档： 

1. 葫芦娃项目客制化

葫芦娃是我们对这类项目的一个戏称，这类项目的PRODUCT\_NAME的是一样，但是需求却多有不同，由于我们无法通过PRODUCT\_NAME来区分项目，所以针对PRODUCT\_NAME一样，但是需求不一样的项目衍生了现在的定制方式。

1. 由于mediatek/config/$(PRODUCE\_NAME)/ProjectConfig.mk可以控制整个版本的各种配置，所以我们同一目录下创建ProjectConfigXX文件，在编译之前做替换。

我们还同时替换了2个文件：

一个是mediatek/config/$(PRODUCT\_NAME)/下的custom.conf文件，用于配置UA等信息，

另一个是mediatek/custom/$(PRODUCT\_NAME)/下的version，替换这个文件主要因为版本号比较长，防止写错。

1. 创建新项目的经验总结

欧标双卡（ProjectConfigEur）：欧标、双卡、有DTV、无ATV

美标单卡（ProjectConfigUsa）：美标、单卡、无DTV、有ATV

一般创建一个新项目，首先判断以上四个配置，选择一个相似的作为模板，拷贝一份。然后再比较修改其他配置项。

（一）单/双卡：

|  |  |
| --- | --- |
| 双卡 | 单卡 |
| GEMINI=yes | GEMINI=no |
| MTK\_GEMINI\_ENHANCEMENT=yes | MTK\_GEMINI\_ENHANCEMENT=no |
| MTK\_SHARE\_MODEM\_CURRENT=2 | MTK\_SHARE\_MODEM\_CURRENT=1 |

（二）DTV/ATV/无TV

支持DTV：(具体参考欧标的配置)

在AUTO\_ADD\_GLOBAL\_DEFINE\_BY\_VALUE最后增加CUSTOM\_KERNEL\_DTV

增加：CUSTOM\_KERNEL\_DTV=nm32x

DTV\_SUPPORT=yes

支持ATV：

|  |  |
| --- | --- |
| 支持ATV | 不支持ATV |
| MTK\_MATV\_SERIAL\_IF\_SUPPORT=yes | MTK\_MATV\_SERIAL\_IF\_SUPPORT=no |
| MTK\_MAV\_PLAYBACK\_SUPPORT=yes | MTK\_MAV\_PLAYBACK\_SUPPORT=no |
| HAVE\_MATV\_FEATURE=yes | HAVE\_MATV\_FEATURE=no |
| MTK\_ATV\_CHIP=MTK\_MT5193 | MTK\_ATV\_CHIP= |
| CUSTOM\_HAL\_MATV=matv | CUSTOM\_HAL\_MATV= |
| CUSTOM\_KERNEL\_MATV=mt5193 | CUSTOM\_KERNEL\_MATV= |
| MTK\_MATV\_ANALOG\_SUPPORT=no | MTK\_MATV\_ANALOG\_SUPPORT=yes |

（三）欧标/美标

欧标：UMTS：900/2100 MHz

美标：UMTS：850/1900 MHz

四频：GSM: 850/900/1800/1900 MHz

三频（去除850频段）：GSM: ~~850~~/900/1800/1900MHz

三频（去除900频段）：GSM: 850/~~900~~/1800/1900MHz

在modem的代码中修改射频参数（硬件同事会提供），修改射频参数之后，编译生成modem文件放到AP侧对应的文件夹中。

Modem侧的编译参数：

1. 如果该项目需要锁网，由于不同项目需要支持的MCCMNC号不一样，就会设置单独的项目宏。如（MX、DO、ESY、CL、MXI等）
2. 非锁网项目，则需要判断是美标还是欧标，四频还是三频的问题。（EUR4、EUR3、USA4、USA3）

|  |  |
| --- | --- |
| 编译参数 | 频段信息 |
| EUR4 | GSM: 850/900/1800/1900 MHz  UMTS: 900/2100 MHz |
| EUR3 | GSM: ~~850~~/900/1800/1900MHz  UMTS: 900/2100 MHz |
| USA4 | GSM: 850/900/1800/1900MHz  UMTS: 850/1900MHz |
| USA3 | GSM: 850/~~900~~/1800/1900MHz  UMTS: 850/1900MHz |

1、如果新增的项目不是锁网版本，则可以使用现有modem，判断好频段信息，选择对应的modem路径。

2、如果新增的项目是锁网版本，则需要在modem中添加单独的编译命令。

比如添加的新项目是AR，其频段信息同USA3

Modem代码里需要修改3个文件，T651W使用的modem代码是如下这个目录的

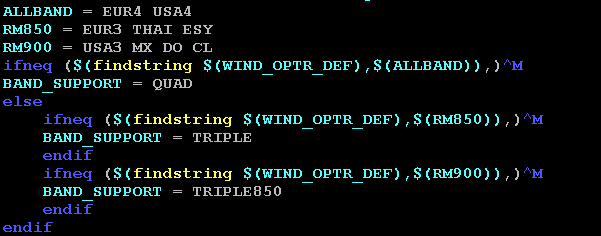


1. makeHSPA.sh



在其中添加AR

1. make/WIND82\_WET\_JB5\_HSPA.mak



在RM900这项后添加AR

1. custom/modem/ul1\_rf/CUSTOM\_MT6166\_SKY\_18/ul1d\_custom\_rf.h

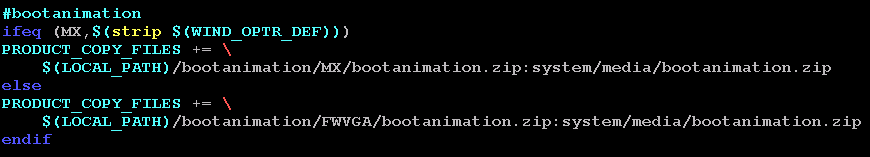
这个文件修改的地方较多，



在出现上述字段的判断语句最后添加 || defined (\_\_OPTR\_AR\_\_)

如果新增项目同EUR4或者EUR3则这个文件不需要修改。

（四）开关机动画以及铃声定制



如果项目需要定制关机铃声以及动画，则需要在mediatek/config/P182A10/system.prop添加

ro.operator.optr=CUST，这个添加后，没有添加关机动画的项目，关机时会显示android默认的关机动画。

考虑到项目客制化，我们新增了一个开关SHOW\_SHUTDOWN\_ANIMATION共同控制。

需要关机动画的项目则需要在mediatek/config/P182A10/ProjectConfigXX

中添加SHOW\_SHUTDOWN\_ANIMATION=yes

这个值默认为no。

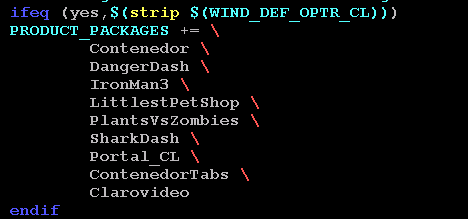
（五）集成第3方应用

1、集成apk不可卸载

在alps/mediate/binary/package/P182A10创建运营商目录XX，将需要集成的apk放在该目录下，添加Android.mk

Android.mk最好用项目开关控制好。

修改alps/build/target/product/common.mk文件



2、集成apk需要可以卸载，并且恢复出厂设置需要能够恢复

之前的项目在verdor目录下直接添加对应目录及文件即可，但是如果在vendor目录增加，T651W的不同项目将无法区分。

我们目前还是放在alps/mediate/binary/package/P182A10这个目录下。

以西班牙Yoigo为例，在上述目录添加ESY目录。

在ESY目录下创建目录data/app，system/app，system/appbackup,一般还需要创建system/lib

在data/app目录下放置需要集成的apk，以及隐藏文件.keep\_list该文件为空。

在system/appbackup目录下放置需要集成的apk

在system/lib目录放置apk需要的库文件

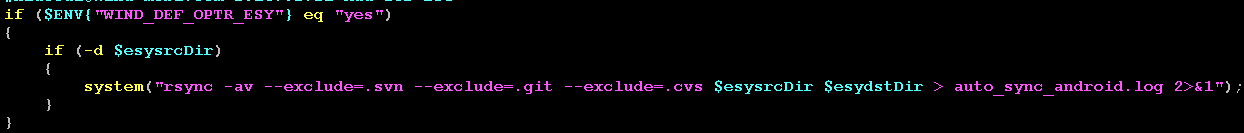
在system/app放置隐藏文件.restore\_list，该文件列出需要回复的apk名称。



然后修改mediatek/build/tools/mtkBegin.pl文件

添加//大家可以查看代码





这个主要是将ESY中的内容拷贝到vendor目录下。

（六）上层overlay定制开关添加

以添加添加开关WIND\_DEF\_OPTR\_XX

1. 在mediatek/config/common/ProjectConfig.mk

添加WIND\_DEF\_OPTR\_XX=no

1. 在mediatek/config/P182A10/ProjectConfigXX

添加WIND\_DEF\_OPTR\_XX=yes

1. 在mediatek/build/tools/javaoption.pm添加WIND\_DEF\_OPTR\_XX

在上层java代码中通过FeatureOption. WIND\_DEF\_OPTR\_XX进行客制化，注意导包：



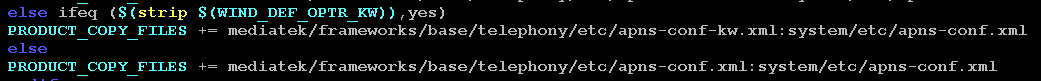
注：FeatureOption中能够添加的开关只能是boolean类型

（七）apn定制

在mediatek/frameworks/base/telephony/etc目录新增apns文件

例如增加T651W基线版本apn文件：apns-conf-zte.xml

修改build/target/product/common.mk文件



（八）overlay资源控制

修改对应项目mediatek/config/P182A10/ProjectConfigXX文件中的

RESOURCE\_OVERLAY\_SUPPORT值，其中的generic修改为XX（项目参数）

（九）开机第一屏定制

开机第一屏时通过ProjectConfig.mk中的BOOT\_LOGO=XXX来定义用哪个的目录的开机第一屏图片。

我们新增一个开机第一屏定制目录：BOOT\_LOGO=T651W

在mediatek/custom/common/lk/logo目录增加t651w目录，注意图片的分辨率要和ProjectConfig.mk中的宽高值一致，不然开机第一屏不会显示。

$ cp cmcc\_fwvga t651w -r

将t651w目录中所有文件前缀修改为t651w，替换开机第一屏图片。

另需要在



文件

之后添加。

如果定制的第一屏是QHD的则需要在

后添加对应路径名称。

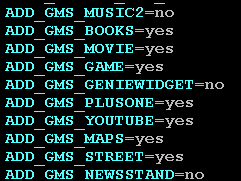
（十）GMS包应用集成



我们的编译脚本中集成GMS的代码已经写好。大家只要本地同步GMS包，编译的版本中就会有GMS应用。

不同项目集成的GMS应用不同，所以我们有很多开关来控制GMS中apk的集成。

下列开关是在mediatek/config/common/ProjectConfig.mk中设置的默认值，不同项目根据需求不同，修改对应开关的值。



1. 客制化注意事项

（一）、利用PRODUCT\_PROPERTY\_OVERRIDES设置默认属性时，如来电铃声、闹钟声、时区时

注意使用PRODUCT\_PROPERTY\_OVERRIDES += \

不要使用PRODUCT\_PROPERTY\_OVERRIDES := \ 不然会覆盖之前添加的属性

（而）、在4.0后的版本中添加资源文件(字符串、图片、布局等)，

必须在alps\frameworks\base\core\res\res\values\下的Symbols.xml或Public.xml中声明，才会生成resource ID，否则编译会报错。

比如，要添加一个drawable，

1. 在Symbols.xml中，先找到<java-symbol type="drawable" name="xxx" />的区域，然后在其最后面仿照此格式添加一行代码，name即为要添加的图片的名字。

此处声明的资源为framework internal的资源，不会build到Android SDK中。

2. 在Public.xml中，可用<public  type="drawable" name="xxx"/>的方式声明为framework public资源，编译时，需要手动update api，这样可使新添加的资源能被第三方apk引用到。

（三）AUTO\_ADD\_GLOBAL\_DEFINE\_BY\_NAME是编译系统开出可以将ProjectConfig.mk中的宏(注意添加的宏必须大写)添加到C/C++源文件的接口，在增加时需要注意：

AUTO\_ADD\_GLOBAL\_DEFINE\_BY\_NAME中添加的宏定义开关的值只能是yes和no，不能为其他值，否则会编译不过，而且这个错误会很难定位。

（四）之前T651W遇到版本号过长导致进入modem meta模式需要2分钟以上，GPS校准不过的问题。

主要是因为



这个字符串的长度超过64位引起。之后T651W是将该字段中的外部版本号去除。

六、版本中已有开关

文档中从1078行开始为我们自己添加的开关，已经注明开关的用途。



1. 版本发布



1、代码下载好之后，修改版本号 mediatek/custom/P182A10/versionXX;修改好之后及时提交到服务器

2、编译版本： ./quick\_build.sh T651W\_XX n user

3、编译结束后将版本刷到机器中，大致验证一下，没有问题则开始编译update包，注意正式发布的版本不是这时候释放的版本

4、编译update包：

source change\_java.sh 1.6

./makeMtk -opt=TARGET\_BUILD\_VARIANT=user P182A10 otapackage

执行 ./release\_version.sh T651W，这时释放的版本才是用于正式版本发布的，压缩包命名为：版本号\_DL.zip

out/target/product/P182A10/P182A10-ota-\*\*\*\*\*.zip 这个为SD卡升级包，正式发布时需命名为:版本号\_SD.zip

out/target/product/P182A10/obj/PACKAGING/target\_files\_intermediates/P182A10-target\_file-\*\*\*\*.zip 这个用于生成fota升级包

5、编译fota升级包

将前一个版本(B03)的P182A10-target\_file-\*\*\*\*.zip拷贝至代码根目录，重命名为update\_a.zip

将这个版本(B04)的P182A10-target\_file-\*\*\*\*.zip也拷贝至代码根目录，重命名为update\_b.zip

执行：source change\_java.sh 1.6

./build/tools/releasetools/ota\_from\_target\_files -i update\_a.zip update\_b.zip updateB05ToB06.zip