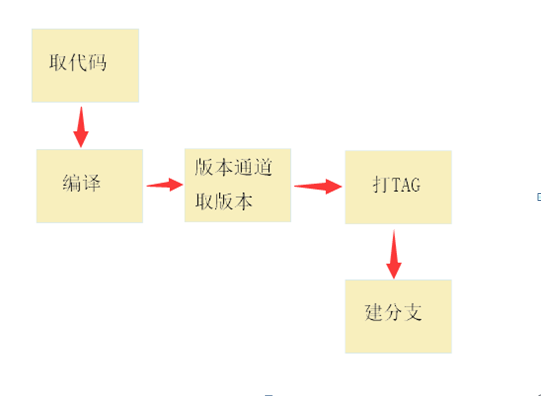
**工作指南**

[](http://10.0.10.6/windwiki/index.php/%E6%96%87%E4%BB%B6:GD05_SW01_C_048_1.png)

大体流程图如上

首先版本号按项目需求修改上传提交。

一、 取代码

1、 首次可以参考文件：/EXCHANGE/public/chusuxia/merge\_all.sh，获取对应项目代码

2、 非首次编译该项目，最好就是到vendor/wind/scripts获取相应项目的Auto脚本，省下更新GMS和找地址拖代码的麻烦，而且不易出错，版本号等常用的修改也很方便。

3、 如果已经有代码了，为了省时间，想直接在代码操作，可以执行如下操作

1、repo forall -c "git clean -df" ---------清除所有git管理的目录下新增的文件

2、repo forall -c "git co ." ---------清除所有git管理的目录下修改的代码

3、repo sync –j4 ---------更新代码

4、 此外，如果没用使用自动脚本，还应到对应的GMS文件夹更新一次

二、 编译

1、手动执行编译命令时，如果是user版本，后续的命令也要是user。例如，V200F的签名版本，编译到ota需执行四个命令，如果是eng，四个命令都得加上eng

2、编译时查看下空间大小，避免因空间问题多花时间。

3、已经编译过的代码想remake来编译版本，修改版本号需要删除根目录的VERSION文件和OUT路径下的build.pro，android5.0以上的，还需要到根目录下的device去修改对应项目的version。

4、android5.0以上，overlay以外的，执行remake快速编译出完成修改点的版本，需要将修改点同步修改到根目录里，因为remake无法执行拷贝命令。

5、单编模块验证修改的时候，需要在编过的版本根目录的device下，修改BoardConfig.mk

将WITH\_DEXPREOPT改为false。例V200F修改： device\ginreen\V200F\BoardConfig.mk

WITH\_DEXPREOPT := false

三、 版本通道取版本

1、 对比release出来文件的大小和编译代码里编译生成的大小是否一致，如果不一致，重新执行release脚本释放一次, 主要对比的是生成的大文件，例如：system.img、 target文件、vmlinux文件、SD卡升级 包。

2、 查看释放文件的时间，由于服务器拥堵或者一些意外原因，有可能会导致部分文件没有正常释放，所以必须要核对所有文件的释放时间。不同文件时间跨度可能会很大，比如时间跨度超过二十分钟，需要执行release 脚本重新释放一次。

:[](http://10.0.10.6/windwiki/index.php/%E6%96%87%E4%BB%B6:GD05_SW01_C_048_2.png)

图中的不同文件释放时间跨度有半小时，部分是ota前释放的，部分是ota后释放的，ota后部分文件没有释放出，这样的情况需要重新释放一次。

4、 编otapackage的时候，需要在otapackage编译结束正常释放后再拷贝各包，因为执行前后文件会有变化。

四、 打TAG

正式版本发布后，需要打上tag

$repo forall –c “git tag –a 版本号\_日期 –m ‘日期’”

1. 需要注意的是，版本号后需要跟上日期，如果不带日期，git tag 查询的时候就看不到具体时间了。

$repo forall -c "git push --tags" #提交tag

$repo forall -c "/EXCHANGE/public/hebin/scripts/repo\_check\_tag.sh tag名" #检查tag是否全部提交成功

需要注意的是，如果当前空间不够，导致打TAG失败，可以先删除已经提交的部分，释放空间，重新提交

可以通过如下命令删除已经提交的部分tag：

Repo forall –c “git tag –d tag名字”

Repo forall –c “git push origin :refs/tags/tag名字”

例如：删除这个tag：AZU\_A110F\_K14V1.0B08\_20150303

Repo forall –c “git push STS001 :refs/tags/ AZU\_A110F\_K14V1.0B08\_20150303”

五、 建新分支

版本稳定后，把稳定后打的tag拉一个新分支

首先，拖下代码

然后依次执行：

1、repo forall -c "git co -b 当前分支名 建立的分支名"

2、repo forall -c "git push STS001建立的分支名: 建立的分支名"

3、修改 .repo/manifests 里的default.xml

<default revision="refs/heads/当前分支名"

修改为：

<default revision="refs/heads/创建的服务器分支名"

4、上传default.xml

git add

git commit

git push origin default:创建的服务器分支名

这三个步骤缺一不可。

5、发邮件，收件人为该项目所有相关责任人，邮件告知新分支地址，相关负责人对项目的修改维护，都切换到新分支。