**Android编译及烧录说明书**

1.0.2

上海未来伙伴机器人有限公司

进化部辅助学习部

**2017-01-01**

**版本说明**

| 版本号 | 日期 | 编写 | 审核 | 说明 |
| --- | --- | --- | --- | --- |
| 1.0.0 | 2018-03-20 | 王冬冬 |  | 初稿 |
| 1.0.1 | 2018-04-16 | 王冬冬 |  | 添加编译内容 |
| 1.0.2 | 2018-04-17 | 王冬冬 |  | 添加差分包制作过程 |
|  |  |  |  |  |
|  |  |  |  |  |

**目录**

[一、 简要介绍 5](#_Toc511750087)

[二、系统类别及烧录原理简要说明 5](#_Toc511750088)

[1 、mtk6580平台 5](#_Toc511750089)

[2 、瑞芯微rk3128平台 5](#_Toc511750090)

[三、烧录工具及驱动 6](#_Toc511750091)

[1、mtk6580平台 6](#_Toc511750092)

[2、瑞芯微rk3128平台 6](#_Toc511750093)

[3、所需账号及地址 6](#_Toc511750094)

[四、 详细步骤介绍 6](#_Toc511750095)

[1、mtk6580平台 6](#_Toc511750096)

[A、 6](#_Toc511750097)

[B、 7](#_Toc511750098)

[C、 8](#_Toc511750099)

[D、 9](#_Toc511750100)

[E、 11](#_Toc511750101)

[2、瑞芯微rk3128平台 11](#_Toc511750102)

[A、 11](#_Toc511750103)

[B、 12](#_Toc511750104)

[C、 13](#_Toc511750105)

[D、 14](#_Toc511750106)

[E、 15](#_Toc511750107)

[五、常见问题分析与解决 16](#_Toc511750108)

[1、驱动安装 16](#_Toc511750109)

[2、 烧录问题 16](#_Toc511750110)

[3、出现其他的问题，请首先尝试做一些对比实验，比如同一天PC更换机器、同一台PC更换USB口、同一台机器更换PC等等。如果多次对比后仍有问题，请检查硬件或联系android系统组伙伴。 17](#_Toc511750111)

[六、MTK AOSP Build 18](#_Toc511750112)

[1、获取代码 18](#_Toc511750113)

[2、替换apk 19](#_Toc511750114)

[3、编译工程项目 20](#_Toc511750115)

[4、打包 23](#_Toc511750116)

[5、发布 24](#_Toc511750117)

[6、差分包制作 24](#_Toc511750118)

[7、升级后台管理方式 25](#_Toc511750119)

# 简要介绍

该文档的目的在于让操作者快速准确的掌握android系统的烧录方法，以及了解一些常见烧录问题的分析与解决。

此处介绍的是调试烧录模式，与工厂烧录模式不同，请区分！！！

# 二、系统类别及烧录原理简要说明

## 、mtk6580平台

Mtk平台在bootloader之前有一个DA文件，该文件是mtk平台的一个二进制的文件，我们无法修改。该文件的作用是在板子刚上电的时候，等待USB口的握手动作。而烧录软件在执行了下载按钮以后，会一直尝试与新插入的USB设备进行握手，该握手会互通设备的型号、平台、flash信息等。握手成功后进行bootloader、kernel、system等分区的烧录

## 、瑞芯微rk3128平台

RK平台烧录一般是原本中已经有系统，待开机进入android系统后，插入USB线，此时识别到的是一个MSC设备、ADB设备，该两种模式表示android系统已经进入normal模式、recovery模式，接下来在烧录软件上点击切换，软件会下发命令让设备重启进入LOADER模式，此模式下就可以烧录了。

# 三、烧录工具及驱动

## 1、mtk6580平台

存放目录：

\\10.100.2.28\android\_readonlyRom\_Images\shares\Tools\MTK6580-tools-driver

## 2、瑞芯微rk3128平台

存放目录：

\\10.100.2.28\android\_readonlyRom\_Images\shares\Tools\MTK6580-tools-driver

## 3、所需账号及地址

\\10.100.2.28\android\_readonly

账号：readrom

密码：123456

建议使用磁盘驱动的方式访问该地址

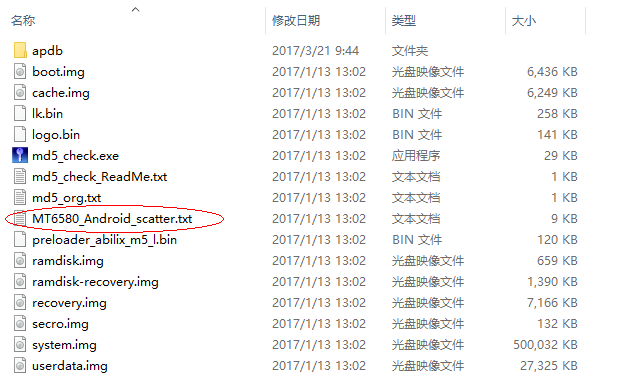
# 详细步骤介绍

**（重点步骤已经红色圈出！！！）**

## 1、mtk6580平台

### A、

首先你需要先拿到一份烧录的文件包，类似下图，其中注意红色圈出的文件



### B、

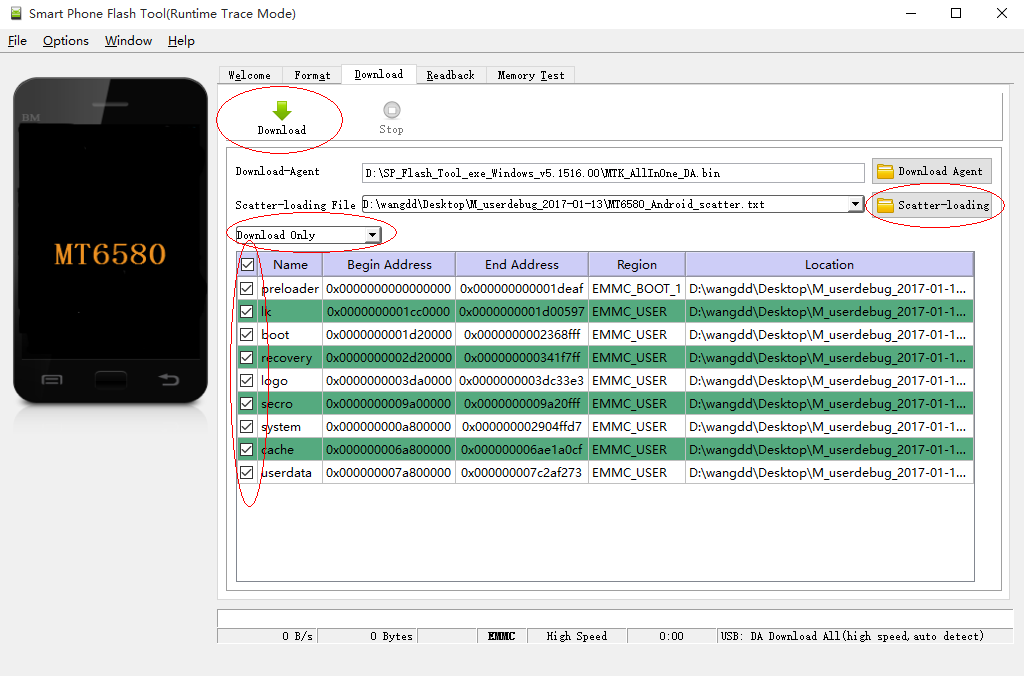
接下来打开烧录软件，按如下步骤选择烧录内容

第一步先选择scatter文件，该文件就是上一步骤中的MT6580\_Android\_scatter.txt文件

第二步选择烧录方式为Download Only

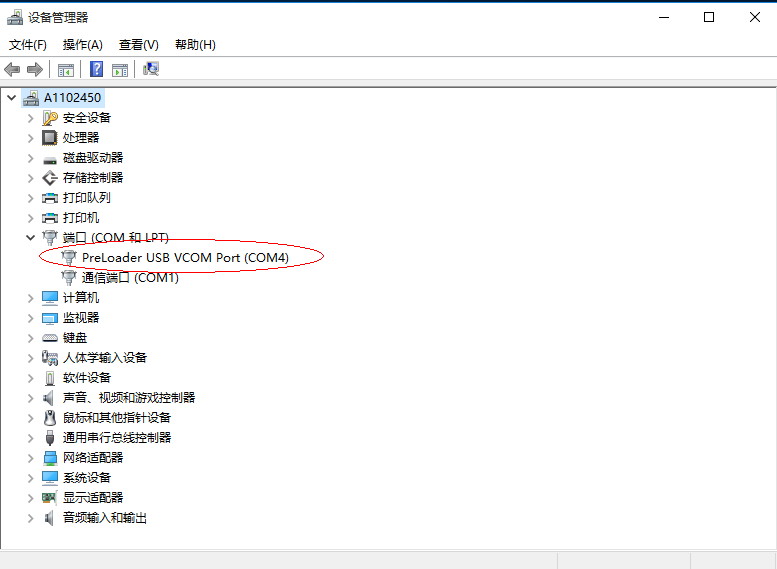
第三部选择你需要烧录的分区，一般情况下请全选（全部烧录）

第四部选择Download，然后等待设备插入



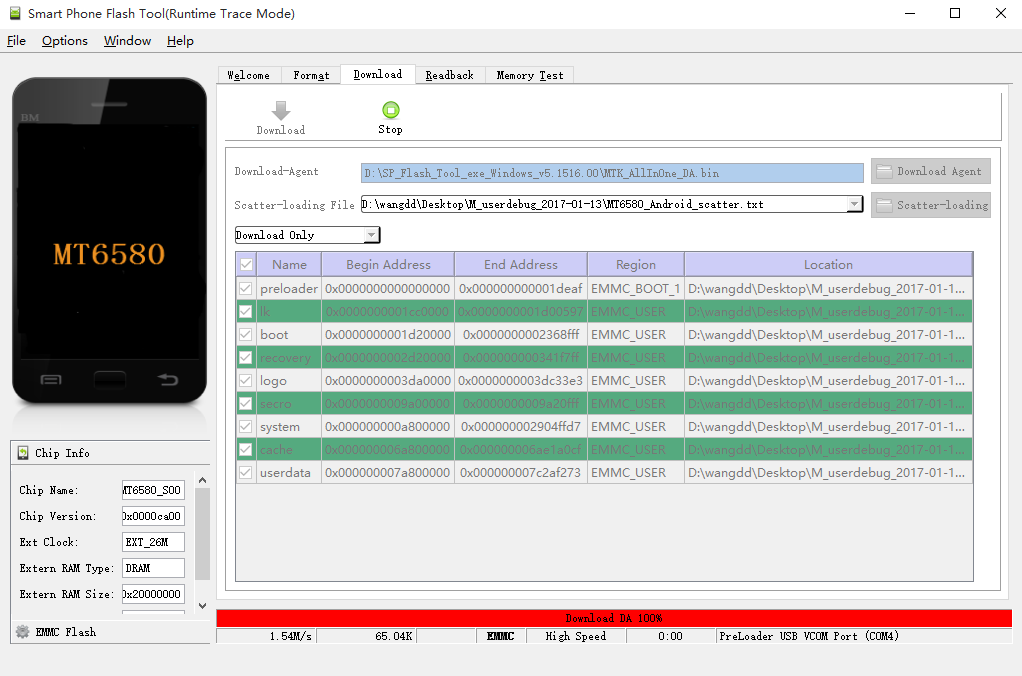
### C、

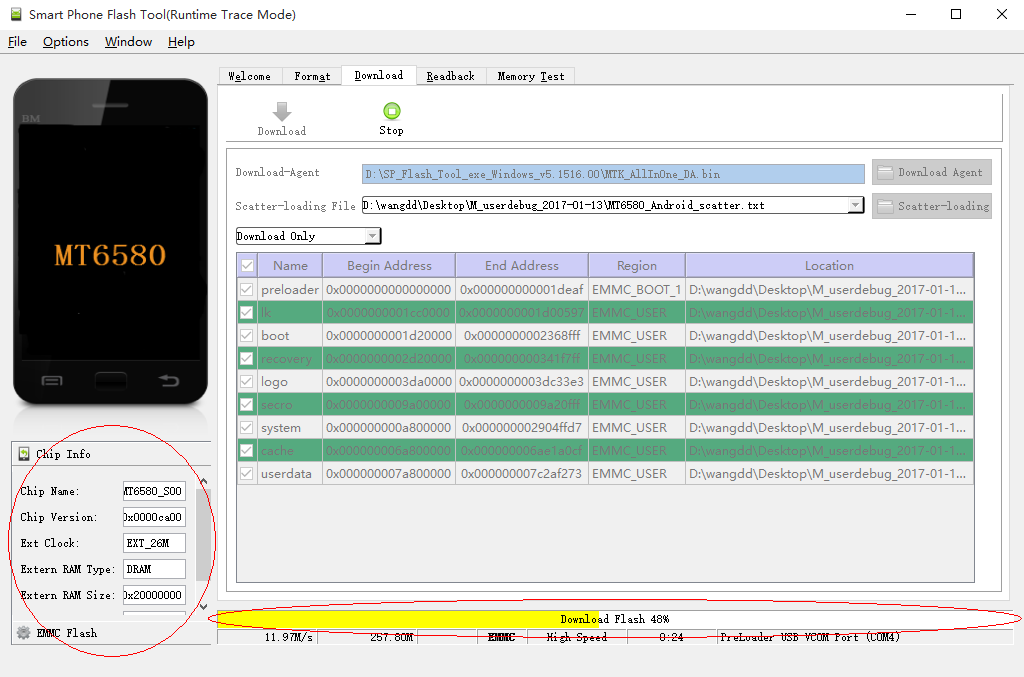
给设备掉电（拿下电池、关机或其他），然后放上电池/加电（此时android不要开机），然后插入USB，在设备管理器中出现一个Preloader设备，烧录就开始了。



### D、

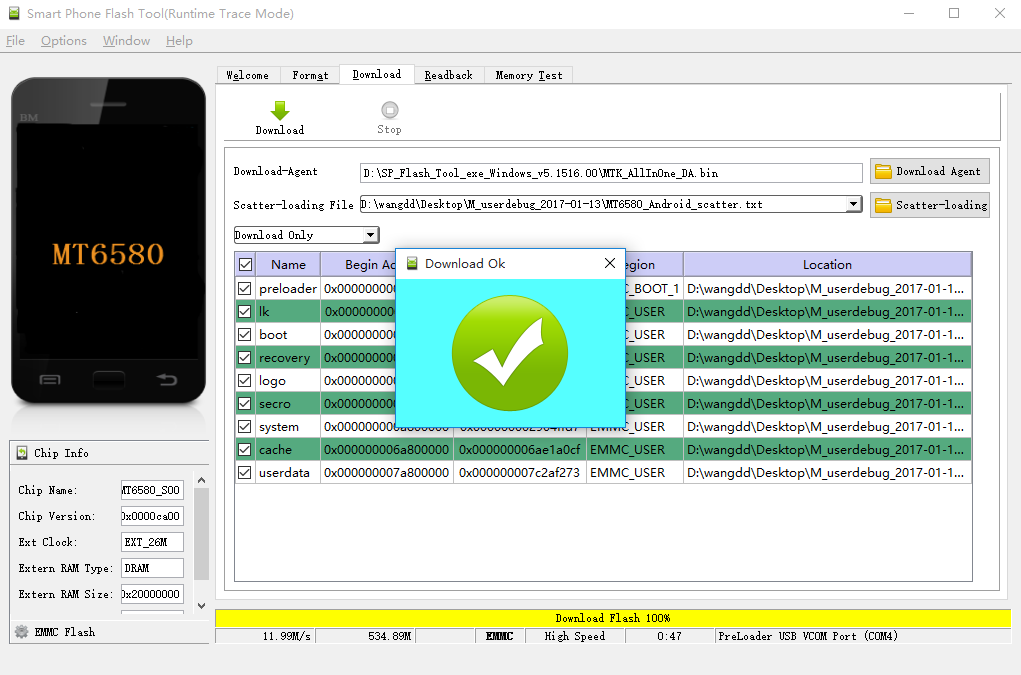
首先是烧录DA文件，然后是Bootloader，然后是kernel、system、recovery等





### E、

烧录最后显示如下，表示烧录成功，此时就可以拔下设备，进行开机

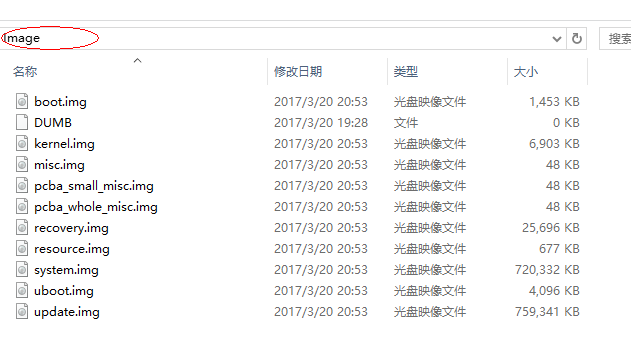


## 2、瑞芯微rk3128平台

### A、

首先拿到如下的第一张image烧录包截图，其中Image文件夹如第二张截图

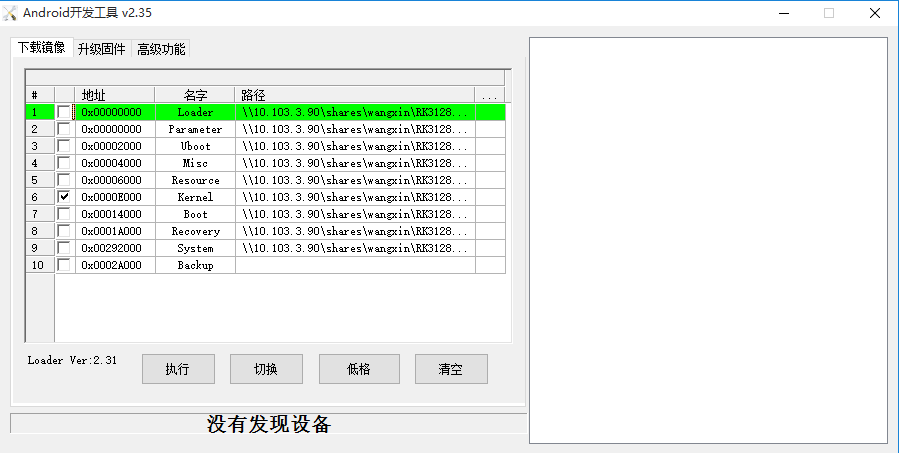


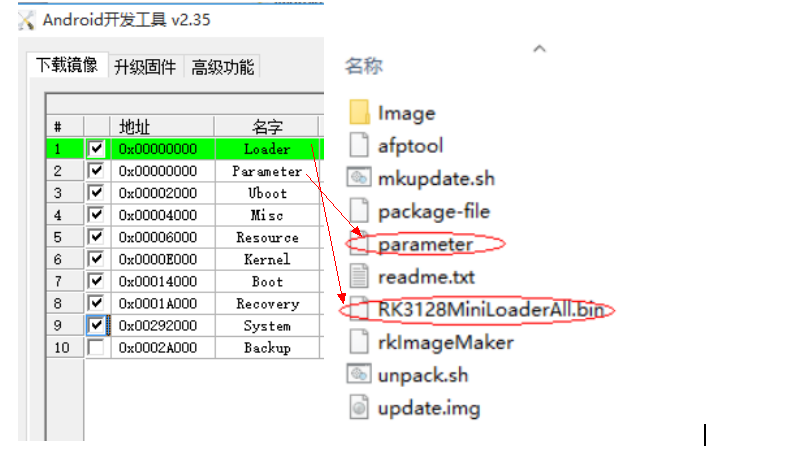


### B、

我们打开烧录软件，然后根据烧录包的路径，对应选取烧录的bin文件

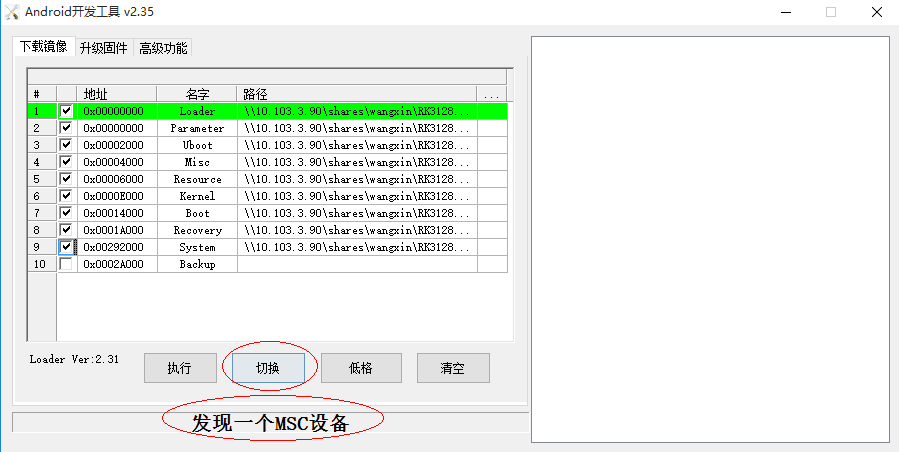
文件对应关系一定要正确，其中的Loader和Parameter按照下图中的红色箭头所示，其他文件按照名称和Image文件夹中的img文件一一对应。





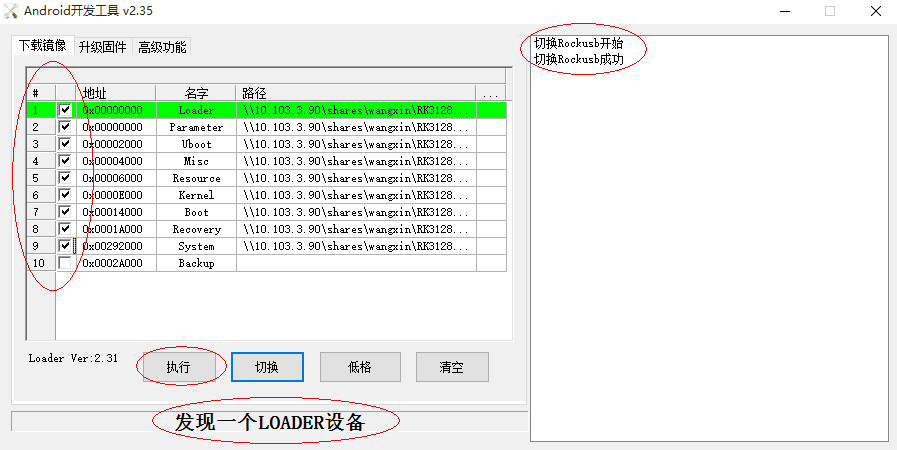
### C、

将设备通过USB线插入电脑，当出现MSC设备后，点击切换按钮



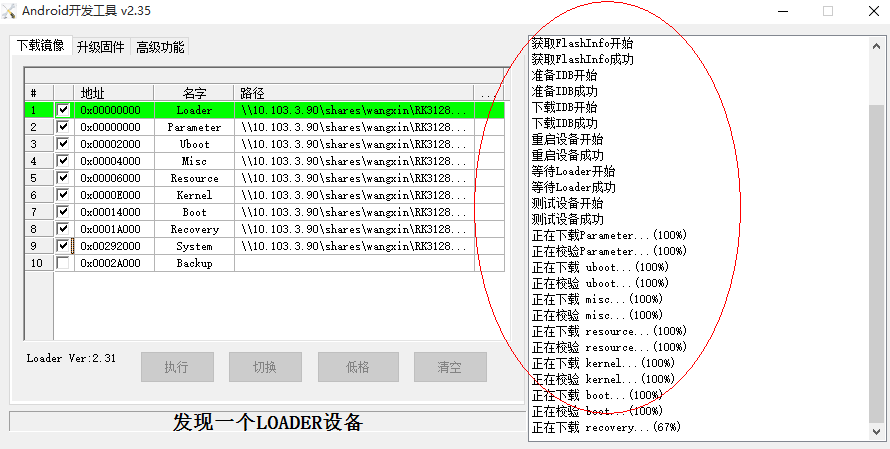
### D、

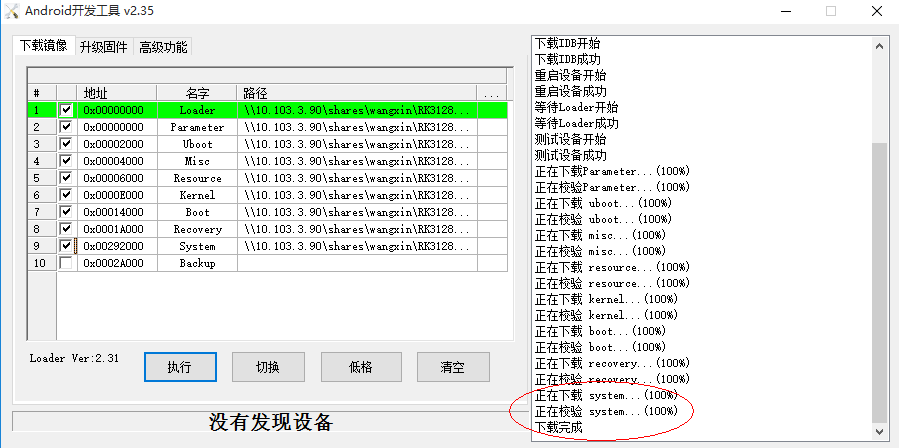
然后待切换成功，设备显示“发现一个LOADER设备”后，可以选择需要烧录的内容，该步骤不清楚的建议选择1-9，第10项不要选择，然后点击执行按钮



### E、

然后信息框会有烧录信息提示，直到最后一步全部完成，会显示下载完成，然后就可以拔下设备了





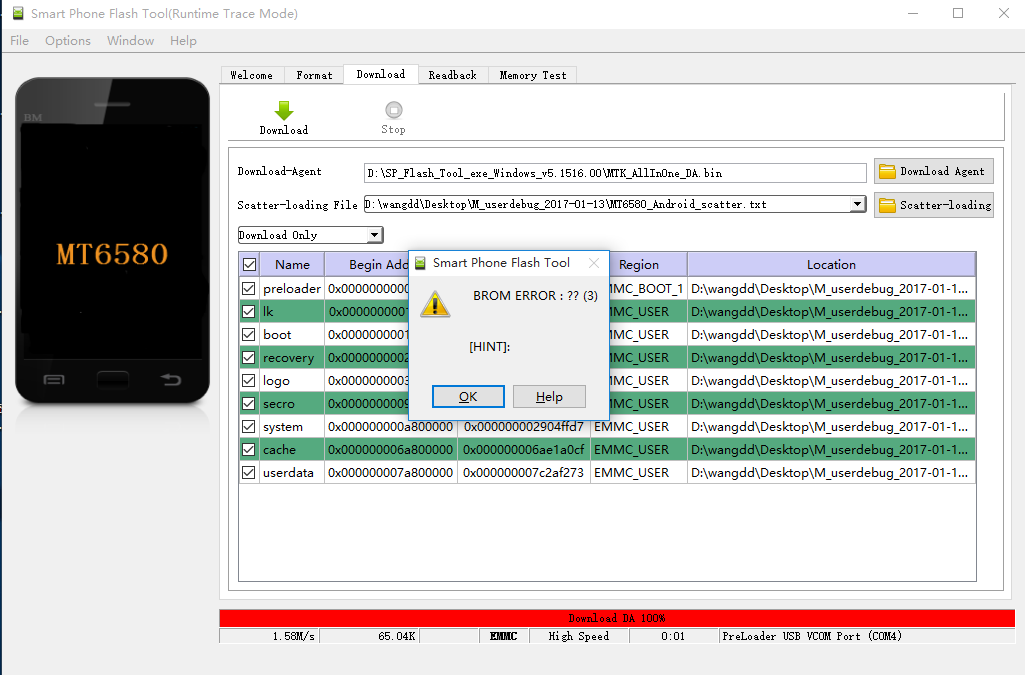
# 五、常见问题分析与解决

## 1、驱动安装

Mtk平台的驱动安装在windows 7上兼容不好，请联系IT然后开通PC网络，通过驱动精灵、豌豆荚等第三方设备完成安装。Windows 10系统可以正确安装，所以也可以让IT换系统

## 烧录问题

烧录中常见问题，主要以下几点

1. mtk平台烧录不跑红条（就是烧录的第一步DA下载），可能是机器由于smt32或其他原因，没有给android核心板彻底掉电造成
2. Mtk烧录中在某一个百分比停下来，对比烧录其他机器可以。建议硬件查看板子的供电稳定性
3. Mtk平台烧录中，红条跑完以后，出现类似一下界面，错误信息有多种。建议彻底掉电后重新尝试
4. RK平台有时无法切换成功，检查当前板子是否可以开机，如果无法开机，则需要拆机短接ddr flash管脚烧录，具体操作请联系android系统的伙伴或其他知道如何操作的伙伴。
5. RK平台优势在校验阶段停止，请更换PC重新尝试，也可更换其他机器进行对比实验。

## 3、出现其他的问题，请首先尝试做一些对比实验，比如同一天PC更换机器、同一台PC更换USB口、同一台机器更换PC等等。如果多次对比后仍有问题，请检查硬件或联系android系统组伙伴。

# 六、MTK AOSP Build

以下主要讲解如何获取代码、替换apk、编译、打包、发布等一个步骤。让系统版本的发布成为一个标准化的工作。

## 1、获取代码

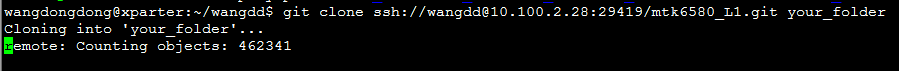
获取代码的前提是你有一个android代码的git/gerrit账号（如果是新开的账号，需要做

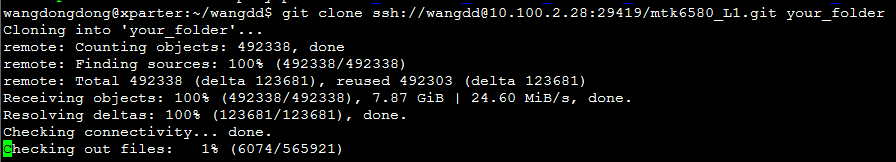
一些ssh key的设置及邮箱账户认证，该内容本文暂不讲述），然后就可以在服务器上获取

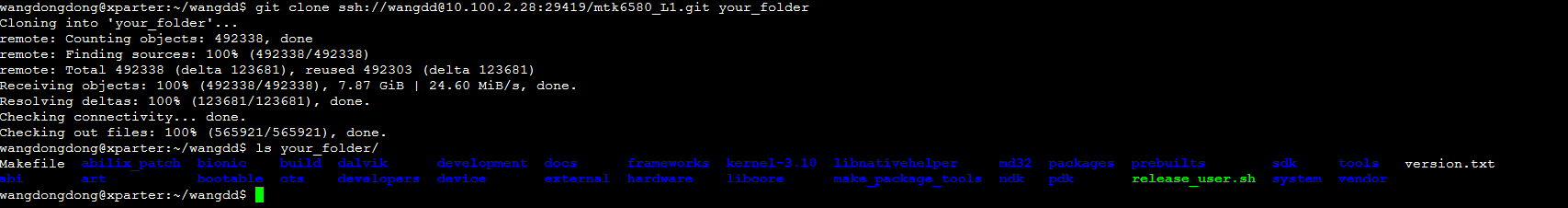
Android系统的代码。地址是如下的命令的后面部分，整个的命令如下，其中将wangdd替换为你的账号。后面的your\_folder是你自己取得放置代码的目录，没有的话就是默认创建一个mtk6580\_L1文件夹

git clone ssh://wangdd@10.100.2.28:29419/mtk6580\_L1.git your\_folder

如下表示已经可以正常获取代码





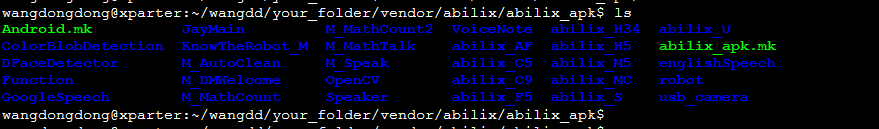


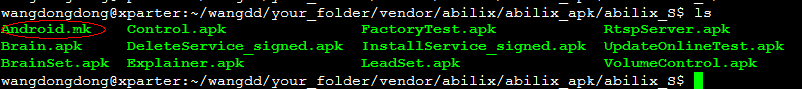
## 2、替换apk

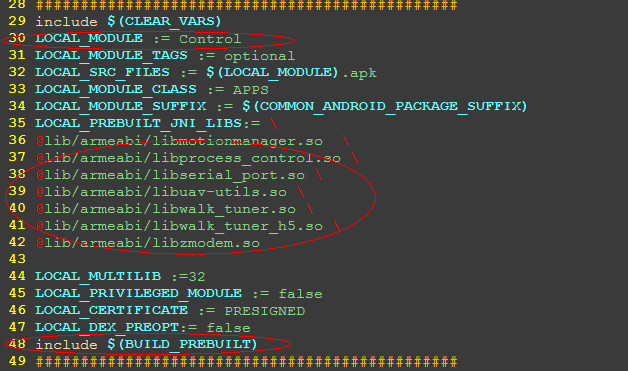
此处讲解替换apk的目的，完全是为了我们发布版本的需求。首先我们将我们需要植入的apk，弄到外面的编译服务器。然后使用拷贝/剪切的方式，将apk放置到如下目录中（该目录是我们专门放置植入apk的位置，其中的Android.mk文件是与植入apk相匹配的。其中的编译方式、取名、lib库等，都是事先配置好的，如需改动请自行处理）

我们将需要内置的apk放置在如下的目录中，我们已经根据项目进行了区分，一般情况下无需修改Android.mk文件，但文件名比如替换为下图的apk的名字。

比如：你需要将S\_Factory.apk ,替换为FactoryTest.apk,否则编译不成功。注意大小写



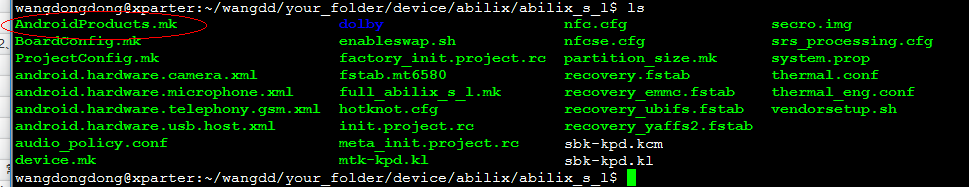


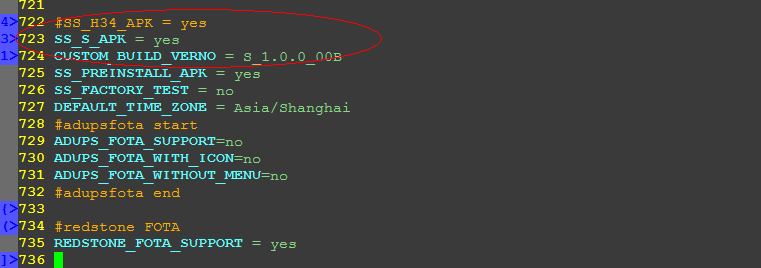


接下来一个重要的改动，是需要进行版本号的修改，请打开如下的文件，查看最底部的位置，有一个宏控，控制使用刚才apk目录下的哪个项目目录，其次就是一个版本号。

如果是再次编译，需要修改版本号，请删除

rm out/target/product/abilix\_c9\_l/system/build.prop ，其中的abilix\_c9\_l替换为你正进行的项目名。然后再进行编译，否则版本号将不会被更新





## 3、编译工程项目

a.先加载编译环境

source build/envsetup.sh

b.选择要编译的项目

lunch full\_$project-eng（这里有eng、debug和user版本可选择）

c. 编译

* Build

$ make –j$(processes number) 2>&1 | tee build.log （其中build.log是编译过程的log，可用可不用，debug编译过程用，以下略去）

* Build preloader

$ make pl –j$(processes number)

* Build lk  
   $ make lk –j$(processes number)
* Build kernel  
   $ make kernel –j$(processes number)
* Make bootimage package  
   $ make bootimage –j$(processes number)
* Build android and make system image  
   $ make systemimage –j$(processes number)

d. 编译模块说明：

采用下面命令编译，必须确保有对整个project做过编译

* 采用mmm命令：mmm <directory>

example: mmm framework/base

* 采用mmm -B <directory>

example: mmm -B framework/base

区别在于：-B 代表always rebuilt , 不加-B，make会根据依赖的时间去比较。

* 采用make -j24 <module name>

example: make -j24 libjpeg ,

在编译过程中的log ，可以看到此module的android.mk 有被include ，否则此module 不会被编译到。

1. 出正式版本

大致和上述编译过程一样

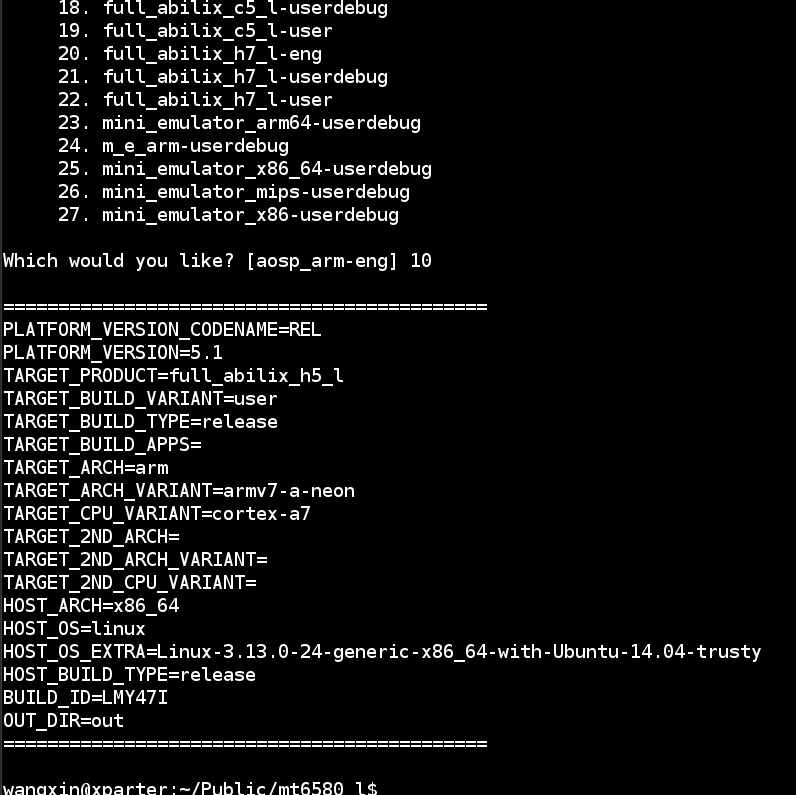
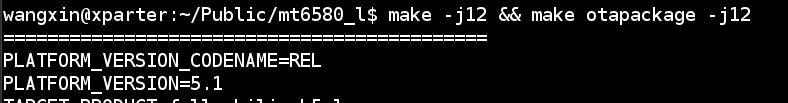
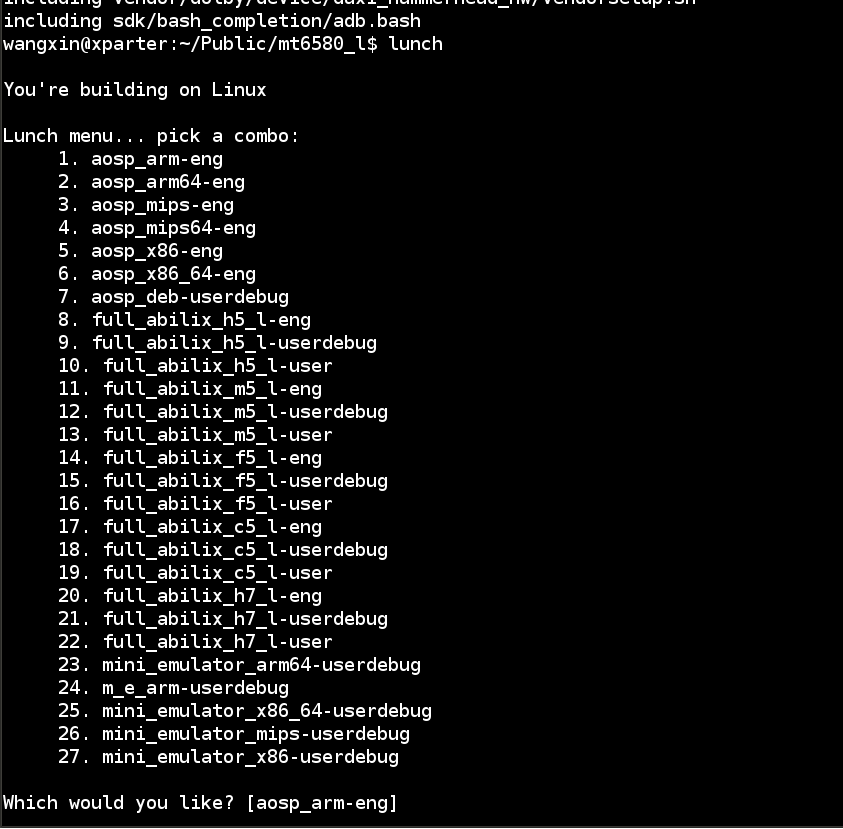
* 先进入工程目录下执行此脚本来加载编译环境 source build/envsetup.sh；
* 执行lunch，选择要出版本的Project；
* 执行make -j12 && make otapackage -j12来编译整个工程；（注：第一次编译时最好采用make -j12 && make otapackage -j12，非第一次的编译可以直接make otapackage -j12，这样能节省点时间。）
* 以上完成之后，会在生成的out的目录下生成对应的项目的redstone 开头的zip的压缩文件，example:redstone\_M\_CN\_1.0.0.\*\_targetfile\_\*.zip。然后把以往的项目的redstone的压缩文件copy到同一目录下，回到主目录下执行./buildota.sh mtk 旧项目的redstone 开头的zip文件 新项目的redstone 开头的zip文件。

example:

./buildota.sh mtk /home/wangxin/Success/mt6580\_l/out/target/product/abilix\_c5\_l/redstone\_C\_1.0.0.1\_targetfile\_1476438761.zip /home/wangxin/Success/mt6580\_l/out/target/product/abilix\_c5\_l/redstone\_C\_1.0.0.1\_targetfile\_1476438761.zi

稍等一段时间后，就会得到example:C\_CN\_1.0.0.1-to-C\_CN\_1.0.0.\*.zip的文件包，上传至redstone即可。此后内容请参阅：fota测试说明.docx

1. 编译版本的执行命令大致过程如下图：



## 4、打包

./release\_user.sh -M 20171219\_C\_1.2.0\_004

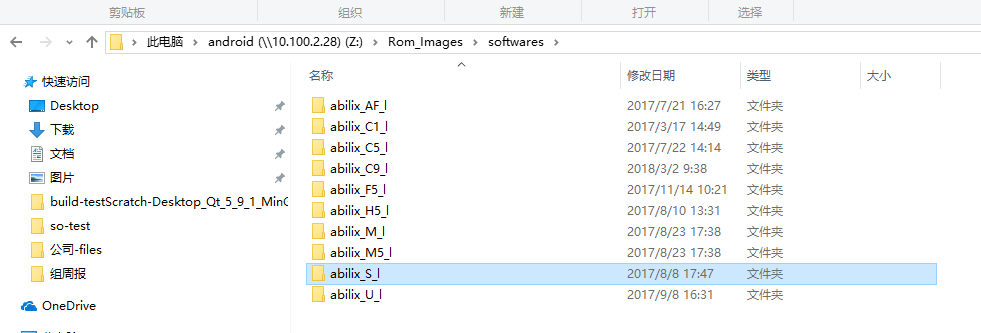
使用如上命令，后面的20171219\_C\_1.2.0\_004是文件名，一般按照这个格式 日期+版本号的格式。该命令会将out目录下的image文件以及相关的zip文件打包到一个文件夹中。并将一个md5的校验码工具一起打包到问价中，除此还会有一个image文件及相应的校验码。这些都是在生产中需要用到的文件。其中两个zip文件是在升级/制作差分包需要用到的。

制作差分包会在后面的篇幅中用到

## 5、发布

这一步就是用上面4中提到的文件夹，放入到指定的目录下，具体目录按项目划分

[\\10.100.2.28\android](file:///\\\\10.100.2.28\\android) 需要使用盘符映射的方式登陆，账户和密码需要向我要。



## 6、差分包制作

1、首先代码根目录有一个问价夹make\_package\_tools，进入该文件夹下cd make\_package\_tools，有一个脚本文件buildota.sh

2、在执行脚本文件之前需要做两件事，一个是创建一个文件夹mkdir update\_files，一个是将需要用到的red\*\*\*.zip 升级包文件拷贝到一个目录下（该目录随意，但不要直接放在make\_package\_tools文件夹下）

3、然后执行如下命令，注意三个参数分别是mtk、需要升级的版本zip、升级到的目标版本zip

./buildota.sh mtk …xxx/redstone\_C\_1.0.0.1\_targetfile\_1476438761.zip …xxx/redstone\_C\_1.0.0.2\_targetfile\_1476438761.zip

4、不出错的情况下，会在update\_files目录下生成需要的差分包，该差分包需要上传到image发布的服务器，以及上传到红石后台供测试使用

## 7、升级后台管理方式

后续讲解