

Android系统源代码的下载与编译



Andy周 (/u/3ba3c3b59a87) [+ 关注](#)

2016.08.22 01:26* 字数 897 阅读 7794 评论 2 喜欢 19

(/u/3ba3c3b59a87)

目录

- 1.简介
- 2.官方同步源代码
- 3.镜像同步源代码
- 4.已有源代码更新
- 5.编译源代码
 - 5.1编译Android 4.1.2
 - 5.2编译Android 5.1.1
- 6.参考

1.简介

之前二次开发Launcher的时候有同步过官方Android 4.1.2的源代码，遗憾当时没有记录下载过程，现在重新温习一下，其实也比较简单。

2.官方同步源代码

官网网址 (<https://source.android.com/source/downloading.html>) 需要翻墙

2.1新建一个用于存放下载脚本文件的目录

```
mkdir ~/bin
PATH=~/bin:$PATH
```

2.2下载Repo脚本文件

```
curl https://storage.googleapis.com/git-repo-downloads/repo > ~/bin/repo
chmod a+x ~/bin/repo
```

2.3创建用于存放Android源代码的目录

```
mkdir android_source
cd android_source
```

2.4初始化

```
repo init -u https://android.googlesource.com/platform/manifest
```

上述命令会要求Repo下载最新的Android源代码，也就是master分支，如果想下载其他分支

```
repo init -u https://android.googlesource.com/platform/manifest -b android-4.0.1_r1
```



2.5同步Android源代码

```
repo sync
```

下载过程保持网络通畅，笔者网络较慢，同步了快一整天。

3.镜像同步源代码

- 对于没有翻墙的用户，可以使用清华大学的镜像。

<https://mirrors.tuna.tsinghua.edu.cn/help/AOSP/>

(<https://mirrors.tuna.tsinghua.edu.cn/help/AOSP/>)

3.1.1 同上述步骤，新建一个用于存放下载脚本文件的目录

```
mkdir ~/bin  
PATH=~:/bin:$PATH
```

3.1.2 下载Repo

```
git clone https://aosp.tuna.tsinghua.edu.cn/android/git-repo.git/  
cp git-repo/repo ~/bin/
```

3.1.3 修改Repo文件

~/bin/repo

```
REPO_URL = 'https://gerrit.googlesource.com/git-repo'  
改为  
REPO_URL = 'https://gerrit-google.tuna.tsinghua.edu.cn/git-repo'
```

3.1.4 创建用于存放Android源代码的目录

```
mkdir android_source  
cd android_source
```

3.1.5 同步源代码

```
repo init -u https://aosp.tuna.tsinghua.edu.cn/platform/manifest -b android-4.0.1  
repo sync -j4
```

- 由于首次同步需要下载 24GB 数据，过程中任何网络故障都可能造成同步失败，建议直接使用初始化包进行初始化。

3.2.1 下载初始包

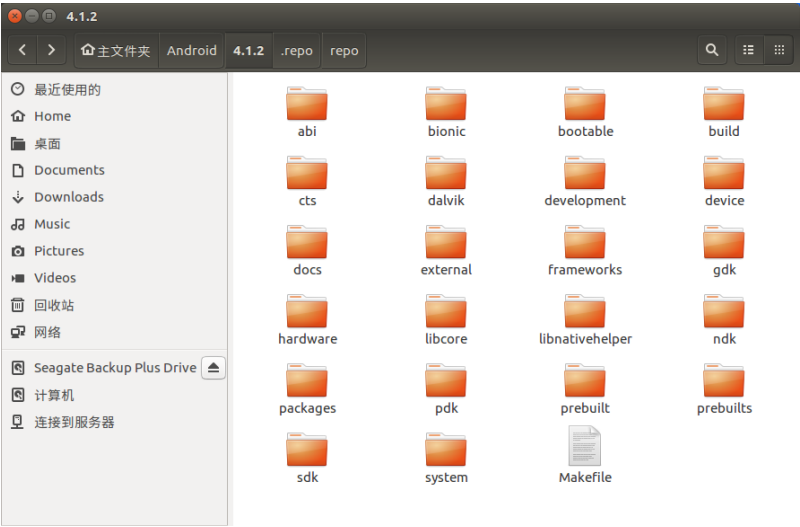
```
#下载重试不限次数，防止网络异常中断  
wget -c -t 0 https://mirrors.tuna.tsinghua.edu.cn/aosp-monthly/aosp-latest.tar  
tar -vxzf aosp-latest.tar  
cd aosp  
#这时ls的话什么也看不到，因为只有一个隐藏的.repo目录  
repo sync
```

3.2.2 选择版本同步

```
repo init -u https://aosp.tuna.tsinghua.edu.cn/platform/manifest -b android-4.0.1  
repo sync -j4
```



下载好了就是下图



源代码目录含义:

目录名	描述
abi	应用程序二进制接口
bionic	C/C++运行时库，在NDK程序中很大一部分调用就是这里的程序
bootable	用于Android装载和启动程序，其中就包括bootloader和recovery。bootloader是Android中唯一在Linux内核之前执行的程序。通过这段程序可以初始化硬件，建立内存控件的映射图等，总之，bootloader就是为Linux内核准备合适的运行环境。
build	用于编译Android源代码以及建议system.img，ramdisk.img等文件的工具
cts	用于兼容性测试的工具
dalvik	Dalvik虚拟机的源代码
development	高层的开发和调试工具
device	与设备相关的代码
docs	包含与Android源代码项目的文档和工具，如Dalvik虚拟机格式文档等
external	扩展工具的源代码
framworks	Android框架层源代码。也就是Android SDK的源代码
hardware	硬件层接口和库
libcore	Java核心库
ndk	NDK相关的源代码
packages	与Android系统一同发布的应用程序的源代码
prebuilts	Android在各种平台下编译之前要使用的工具
sdk	在开发环境中使用的工具，如ddms，draw9path，sdkmanager等
system	Android的基本系统

注意：
查看所有分支

```
cd .repo/manifests
git branch -a
```



```
andy@andy: ~/android-source/aosp.repo/manifests
remotes/origin/android-6.0.1_r43
remotes/origin/android-6.0.1_r45
remotes/origin/android-6.0.1_r46
remotes/origin/android-6.0.1_r47
remotes/origin/android-6.0.1_r48
remotes/origin/android-6.0.1_r49
remotes/origin/android-6.0.1_r5
remotes/origin/android-6.0.1_r50
remotes/origin/android-6.0.1_r51
remotes/origin/android-6.0.1_r52
remotes/origin/android-6.0.1_r53
remotes/origin/android-6.0.1_r54
remotes/origin/android-6.0.1_r55
remotes/origin/android-6.0.1_r56
remotes/origin/android-6.0.1_r57
remotes/origin/android-6.0.1_r58
remotes/origin/android-6.0.1_r59
remotes/origin/android-6.0.1_r60
remotes/origin/android-6.0.1_r61
remotes/origin/android-6.0.1_r62
remotes/origin/android-6.0.1_r63
remotes/origin/android-6.0.1_r7
remotes/origin/android-6.0.1_r8
remotes/origin/android-6.0.1_r9
```

如果想切换到其它分支可以重新执行

```
repo init -b android-6.0.1_r63
repo sync
```

4.已有源代码更新

如果手上已有Android系统源代码，可以通过代理远程更新，不过速度可能会比较慢。另外还可以

4.1修改~/bin/repo文件

```
REPO_URL = 'https://gerrit.googlesource.com/git-repo'
改为
REPO_URL = 'https://gerrit-google.tuna.tsinghua.edu.cn/git-repo'
```

4.2修改源代码目录.repo/manifests.git/config

```
url = https://android.googlesource.com/platform/manifest
改为
url = https://aosp.tuna.tsinghua.edu.cn/platform/manifest
```

4.3修改源代码目录.repo/manifest.xml

```
<manifest>

  <remote name="aosp"
-      fetch="https://android.googlesource.com"
+      fetch="https://aosp.tuna.tsinghua.edu.cn"
      review="android-review.googlesource.com" />

  <remote name="github"
```

最后直接同步即可

```
repo sync -j4
```

5.1编译Android 4.1.2

笔者下载的是Android 4.1.2源代码。

默认的源代码仅能在64位机器上编译。编译过程有很多坑，要有心理准备。

5.1.1进入源码目录初始化编译环境



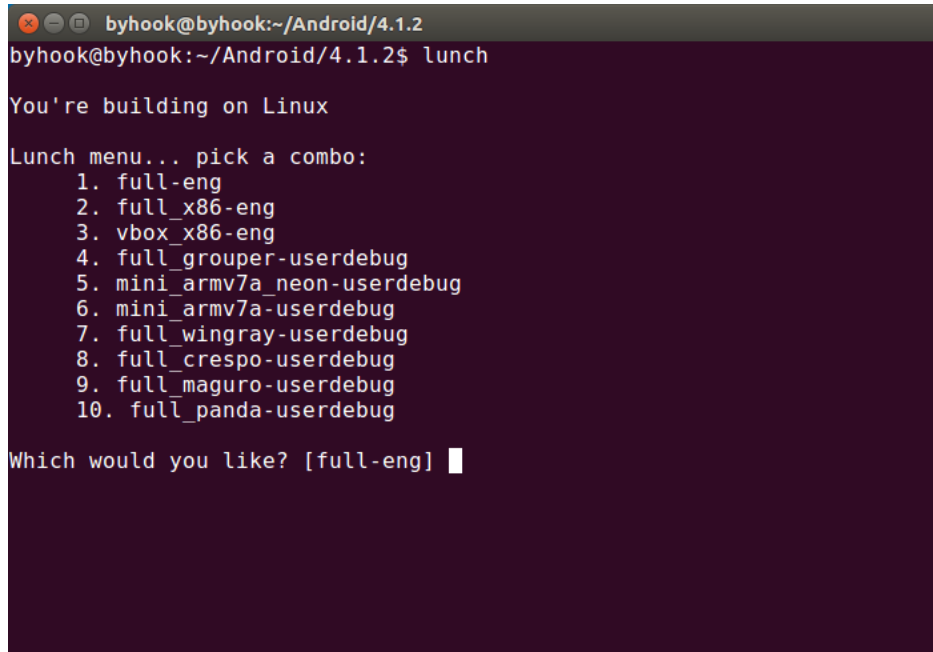
```
source build/envsetup.sh
```

5.1.2选择目标

```
lunch full-eng
```

设置编译目标为full-eng，表示正对所有的移动设备，Android模拟器有效，并打开所有的调试选项。

只执行lunch命令，会出现对应的选项



```
byhook@byhook:~/Android/4.1.2
byhook@byhook:~/Android/4.1.2$ lunch

You're building on Linux

Lunch menu... pick a combo:
 1. full-eng
 2. full_x86-eng
 3. vbox_x86-eng
 4. full_grouper-userdebug
 5. mini_armv7a_neon-userdebug
 6. mini_armv7a-userdebug
 7. full_wingray-userdebug
 8. full_crespo-userdebug
 9. full_maguro-userdebug
10. full_panda-userdebug

Which would you like? [full-eng]
```

5.1.3编译Android源代码

make命令只会利用一个CPU核进行编译，如果是多核CPU，为了缩短时间，可以加上-jn参数。

注意:

```
#Android 4.1.2源代码编译要求
笔记本OS:  Ubuntu 16.04 x64
GNU Make版本: 3.8.1或者3.8.2(笔者用的是3.8.2)
JDK 版本:  JDK 1.6
```

编译过程你很可能会碰到如下问题

```
1. /bin/bash: xsltproc: 未找到命令
2. /bin/bash: flex: 未找到命令
3. /bin/bash: bison: 未找到命令
4. sh: 1: gperf: not found
5. /bin/bash: xmllint: 未找到命令
6. failed--compilation aborted at external/webkit/Source/WebCore/make-hash-tools.pl
7. /usr/include/stdlib.h:759:34: fatal error: bits/stdlib-bsearch.h: No such file or
```

建议提前安装好下列必要依赖

```
sudo apt-get install xsltproc flex bison gperf libxml2-utils libswitch-perl gcc-multilib
```

最后开始编译

```
make -j4
```



笔者笔记本编译花了接近3个小时

```
byhook@byhook:~/Android/4.1.2
kage.apk
Done!
Install: out/target/product/generic/system/app/SystemUI.odex
Install: out/target/product/generic/system/app/SystemUI.apk
'out/target/common/obj/APPS/Settings_intermediates/classes.dex' as '
classes.dex'...
Processing target/product/generic/obj/APPS/Settings_intermediates/pac
kage.apk
Done!
Install: out/target/product/generic/system/app/Settings.odex
Install: out/target/product/generic/system/app/Settings.apk
build/tools/generate-notice-files.py out/target/product/generic/obj/
NOTICE.txt out/target/product/generic/obj/NOTICE.html "Notices for f
iles contained in the filesystem images in this directory:" out/targe
t/product/generic/obj/NOTICE_FILES/src
Combining NOTICE files into HTML
Combining NOTICE files into text
Installed file list: out/target/product/generic/installed-files.txt
Target system fs image: out/target/product/generic/obj/PACKAGING/syst
emimage_intermediates/system.img
Running: mkyaffs2image -f out/target/product/generic/system out/targ
et/product/generic/obj/PACKAGING/systemimage_intermediates/system.img
Install system fs image: out/target/product/generic/system.img
byhook@byhook:~/Android/4.1.2$
```

5.2编译Android 5.1.1

源代码的下载参考上述步骤

环境配置

```
#公司的台式机
本机OS: Ubuntu 14.04 x64
JDK版本: openjdk 1.7
```

配置过程同上，编译过程中如果出现

```
You asked for an OpenJDK 7 build but your version is
java version "1.7.0_79" Java(TM) SE Runtime Environment (build 1.7.0_79-b15) Java Ho
```

建议更换JDK为openjdk 1.7

```
apt-get install openjdk-7-jdk
```

如果出现

```
/bin/bash: build/tools/findleaves.py: 权限不够
```

直接给文件加上执行权限，笔者是直接源码目录

```
chmod a+x -R *
```

笔者编译完大概也是3个多小时，过程跟4.12编译差不多



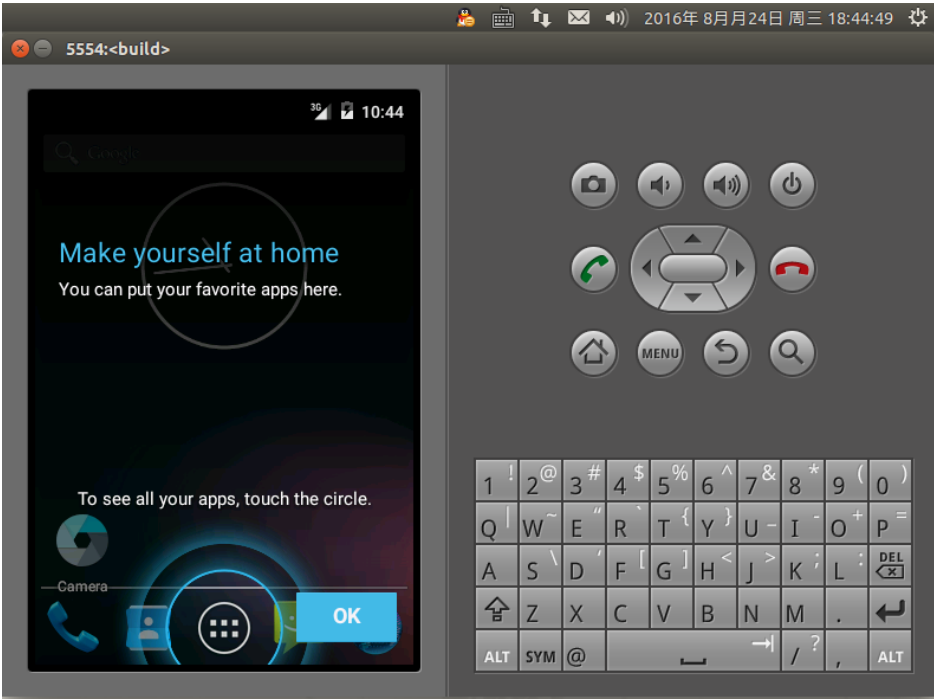
```
andy@andy: ~/android-source/5.1.1
intermediates/system.img
Running: mkuserimg.sh out/target/product/generic/system out/target/product/generic/obj/PACKAGING/systemimage_intermediates/system.img ext4 system 576716800
out/target/product/generic/root/file_contexts
make_ext4fs -T -1 -S out/target/product/generic/root/file_contexts -l 576716800
-a system out/target/product/generic/obj/PACKAGING/systemimage_intermediates/system.img out/target/product/generic/system
Creating filesystem with parameters:
  Size: 576716800
  Block size: 4096
  Blocks per group: 32768
  Inodes per group: 7040
  Inode size: 256
  Journal blocks: 2200
  Label:
  Blocks: 140800
  Block groups: 5
  Reserved block group size: 39
Created filesystem with 1467/35200 inodes and 109511/140800 blocks
Install system fs image: out/target/product/generic/system.img
out/target/product/generic/system.img+ maxsize=588791808 blocksize=2112 total=576716800 reserve=5947392

#### make completed successfully (01:59:32 (hh:mm:ss)) ####
andy@andy:~/android-source/5.1.1$
```

启动模拟器

```
emulator &
```

如图



注意：几个很有用的命令。

```
make -k 继续编译
m 编译完整的Android源代码
mm 此命令必须进入指定的工程的目录进行编译
mmm 可以在任何一级目录编译任意指定的工程
```

6.参考

- <https://source.android.com/source/downloading.html>
- (<https://source.android.com/source/downloading.html>)
- <https://mirrors.tuna.tsinghua.edu.cn/help/AOSP/>



(<https://mirrors.tuna.tsinghua.edu.cn/help/AOSP/>)
<http://blog.csdn.net/michaelpp/article/details/22801953>
(<http://blog.csdn.net/michaelpp/article/details/22801953>)
<http://blog.csdn.net/ambulong/article/details/51627115>
(<http://blog.csdn.net/ambulong/article/details/51627115>)
《Android 深度探索(卷1)：HAL与驱动开发》

📖 Android源代码 (nb/5332137)

举报文章 © 著作权归作者所有



Andy周 (/u/3ba3c3b59a87)

写了 31964 字，被 70 人关注，获得了 125 个喜欢

(/u/3ba3c3b59a87)

+ 关注

不以物喜，不以己悲！





♡ 喜欢 (/sign_in?utm_source=desktop&utm_medium=not-signed-in-like-button) | 19



更多分享

被以下专题收入，发现更多相似内容

(<http://cwb.assets.jianshu.io/notes/images/5373334>)

-  Android... (/c/ddfd0f9bb992?utm_source=desktop&utm_medium=notes-included-collection)
-  Android... (/c/433b3efdb6c9?utm_source=desktop&utm_medium=notes-included-collection)
-  Android... (/c/58b4c20abf2f?utm_source=desktop&utm_medium=notes-included-collection)
-  Android... (/c/ba7defc6f489?utm_source=desktop&utm_medium=notes-included-collection)
-  Android... (/c/644451dfa3ee?utm_source=desktop&utm_medium=notes-included-collection)
-  Android... (/c/3e8d3b8d757c?utm_source=desktop&utm_medium=notes-included-collection)
-  Android... (/c/d8da4555603a?utm_source=desktop&utm_medium=notes-included-collection)

