**编译环境和编译过程手册**

**第一步：安装UBUNTU 64bits**

1 先在ubuntu的网站下载对应的Ubuntu版本: xxx.iso, Ubuntu 16.04再下载Universal-USB-Installer-x.x.x.exe（务必64bits的Ubuntu, 因为Android 2.3以后的版本Google建议在64位系统编译）

2, 用universal-usb将iso写入到U盘中, 创建usb启动盘. 注意要选择正确的ubuntu版本, usb必须格式化为fat32

3. 分三个区出来, 一个是swap区,大概8G; 另一个是ext4的分区, 用于装ubuntu的操作系统.建议40 GB；最后一个ext4, 100 GB，建议mount到/home目录，这个用于代码存放、项目开发以及用于编译。

4, 重启电脑，通过usb启动，按提示安装。

5，安装完后，装依赖软件列表：

sudo apt-get update

sudo apt-get install -y git flex bison gperf build-essential libncurses5-dev:i386

sudo apt-get install libx11-dev:i386 libreadline6-dev:i386 libgl1-mesa-dev g++-multilib

sudo apt-get install tofrodos python-markdown libxml2-utils xsltproc zlib1g-dev:i386

sudo apt-get install dpkg-dev libsdl1.2-dev libesd0-dev

sudo apt-get install git-core gnupg flex bison gperf build-essential

sudo apt-get install zip curl zlib1g-dev gcc-multilib g++-multilib

sudo apt-get install libc6-dev-i386

sudo apt-get install lib32ncurses5-dev x11proto-core-dev libx11-dev

sudo apt-get install lib32z-dev ccache

sudo apt-get install libgl1-mesa-dev libxml2-utils xsltproc unzip m4

sudo apt install liballegro4-dev

sudo apt-get install libX11-dev libxext-dev libxrender-dev libxtst-dev libxt-dev

sudo apt-get install git-core gnupg flex bison gperf build-essential zip curl zlib1g-dev gcc-multilib g++-multilib libc6-dev-i386 lib32ncurses5-dev x11proto-core-dev libx11-dev lib32readline-gplv2-dev lib32z-dev libxml2-utils libswitch-perl gawk

u-boot-tools:i386 u-boot-tools

sudo apt-get install gawk

Android 7 要求JDK为openjdk-8

$ sudo apt-get install openjdk-8-jdk

$sudo apt-get install rar

5、把

6

Android 7.1编译到33%时出现JDK内存溢出的错误：

Out of memory error (version 1.2-rc4 'Carnac' (298900 f95d7bdecfceb327f9d201a1348397ed8a843843 by android-jack-team@google.com)).

GC overhead limit exceeded.

Try increasing heap size with java option '-Xmx<size>'.

Warning: This may have produced partial or corrupted output.

错误日志里边列出了问题并且已经给出了解决方案 - 增加Java虚拟机的-Xmx大小，即设置一个较大的堆内存上限。

所以，解决办法是：

1、修改Jack的配置文件prebuilts/sdk/tools/jack-admin

　　这是一个管理Jack的shell脚本，找到start-server函数，直接修改其启动参数，由原来的

JACK\_SERVER\_COMMAND="java -XX:MaxJavaStackTraceDepth=-1 -Djava.io.tmpdir=$TMPDIR $JACK\_SERVER\_VM\_ARGUMENTS -cp $LAUNCHER\_JAR $LAUNCHER\_NAME"

更改为

JACK\_SERVER\_COMMAND="java -XX:MaxJavaStackTraceDepth=-1 -Djava.io.tmpdir=$TMPDIR $JACK\_SERVER\_VM\_ARGUMENTS -Xmx4096m -cp $LAUNCHER\_JAR $LAUNCHER\_NAME"

　　此时Jack服务器仍然在后台执行，所以我们需要将其停止，然后重启启动(make会自动启动Jack服务器)才能使得修改后的参数生效。

　　我们执行下面的命令此时Jack服务器仍然在后台执行，所以我们需要将其停止，然后重启启动(make会自动启动Jack服务器)才能使得修改后的参数生效。

　　我们执行下面的命令：

#先停止服务

./prebuilts/sdk/tools/jack-admin stop-server

#重新开始服务

./prebuilts/sdk/tools/jack-admin start-server