

# 实验一：安卓开发环境搭建

## 目录

一、Java 开发环境的搭建 .....	2
1.1 JDK（Java Development Kit）的下载与安装 .....	2
1.2 配置环境变量 .....	4
二、android studio 的安装 .....	9
2.1 Windows 下 android studio 的下载与安装 .....	9
2.2 更新 SDK .....	16
2.3 Intel Atom x86 模拟器的安装 .....	19
三、使用 android studio .....	25
3.1 第一个 Android 程序——Hello World .....	25
3.2 创建一个 AVD 以运行程序 .....	28
3.3 调试程序 .....	33
四、常见问题解答 .....	37
4.1 无法访问外网（android studio 无法下载及 SDK 更新无法联网） .....	37
4.2 Intel HAXM 安装不成功 .....	38
4.3 ADB 启动失败 .....	39
4.4 Failure [INSTALL_FAILED_OLDER_SDK] .....	41
4.5 android studio 安装卡在 SDK 下载界面不动（Setup Wizard - Downloading Components） .....	42
4.6 运行模拟器时错误 .....	44
4.7 高版本程序运行在低版本模拟器 .....	45
4.8 点击开始新 Android 项目无反应 .....	45
4.9 虚拟机启动后卡在开机界面 .....	45
4.10 SDK、AVD 闪退 .....	46
4.11 安卓虚拟机找不到 system image .....	46

# 一、Java 开发环境的搭建

## 1.1 JDK（Java Development Kit）的下载与安装

JDK（Java Development Kit）是程序员使用 java 语言编写 java 程序所需的开发工具包。大家可以在 oracle 官网上下载 JDK 安装包。

网址：（<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>）该网址打开后即为 JDK 的下载界面，根据自己的系统选择相应的版本，以 32 位 windows7 举例（如图 1.1.1）：

- 1、点击“Accept License Agreement”选项，以接受下载许可；
- 2、点击相应版本下载（本机为 windows 32 位系统）。

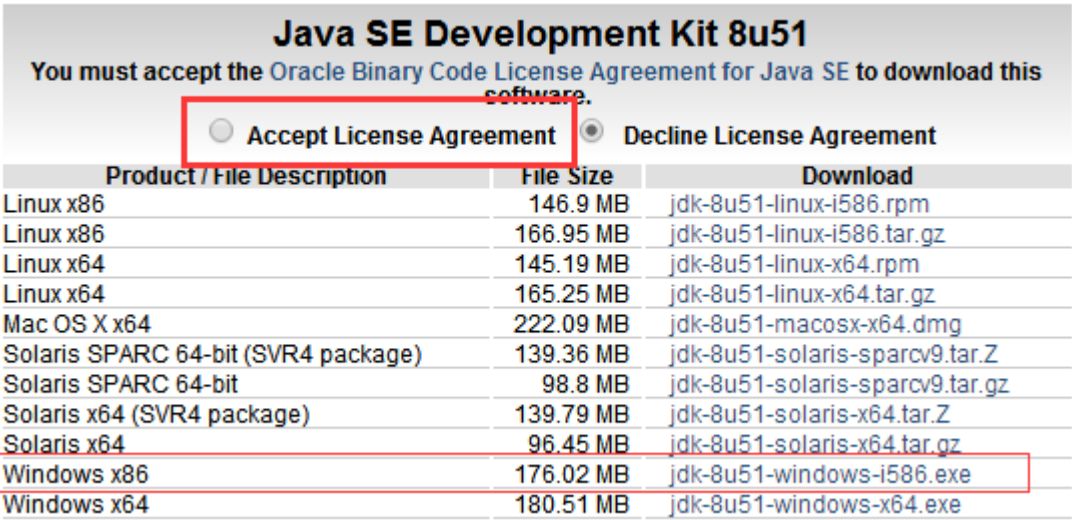


图 1.1.1

3、下载后，双击运行。在等待安装包采集完信息后出现以下界面（如图 1.1.2），点击下一步按钮继续。

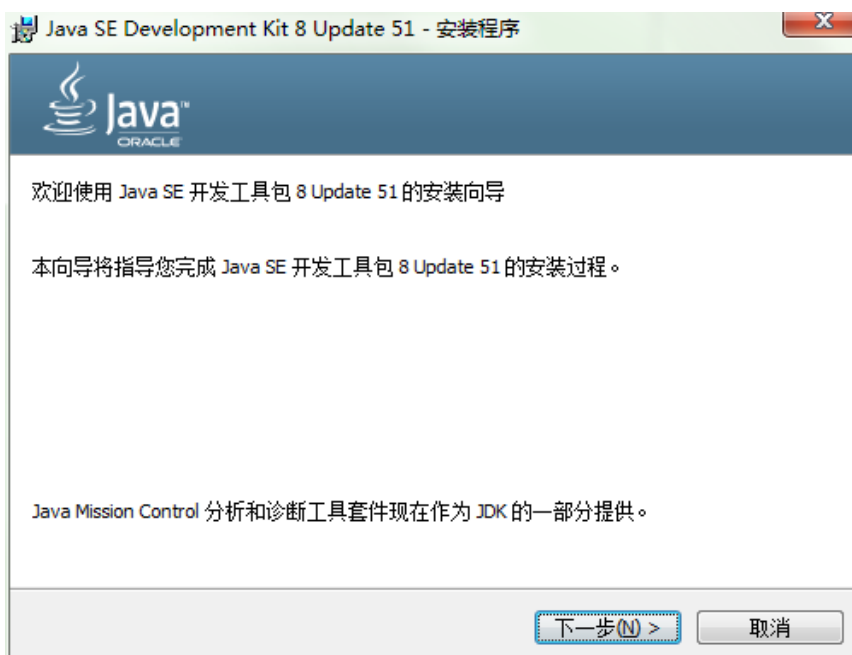


图 1.1.2

4、在安装路径界面（如图 1.1.3），你可以根据你的需要选择路径，如果非默认路径，需要记住该路径，以便以后配置环境变量时候用到。这里点击“下一步”继续。

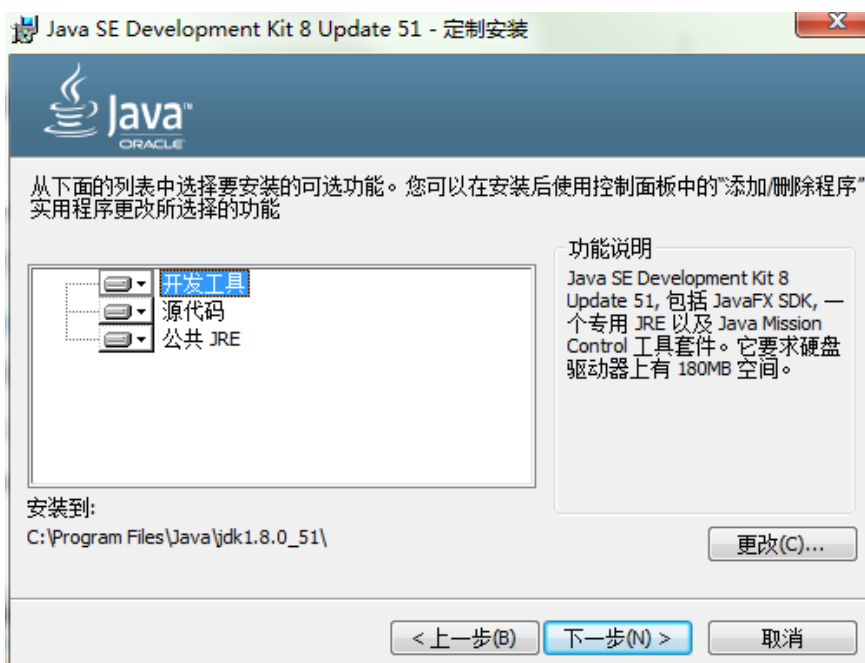


图 1.1.3

5、几分钟后，JDK 已经安装完毕，点击“关闭”按钮来结束安装。



图 1.1.4

6、为了验证 JDK 是否安装成功，我们可以打开 DOS 窗口，输入“java -version”。若出现 java 版本号（如图 1.1.5），即为安装成功。

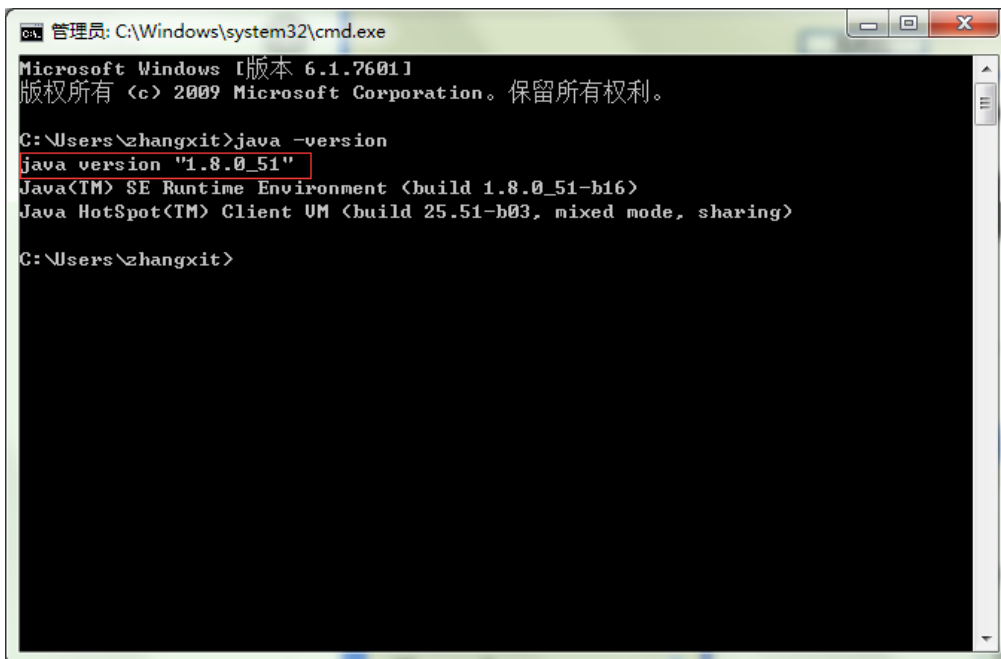


图 1.1.5

## 1.2 配置环境变量

JDK 安装完毕后我们需要配置环境变量。

1、右键点击【计算机】，选择【属性】，出现属性界面，点击【高级系统设置】（如图 1.2.1）。



图 1.2.1

2、弹出“高级系统设置”界面后，点击【环境变量】（如图 1.2.2）

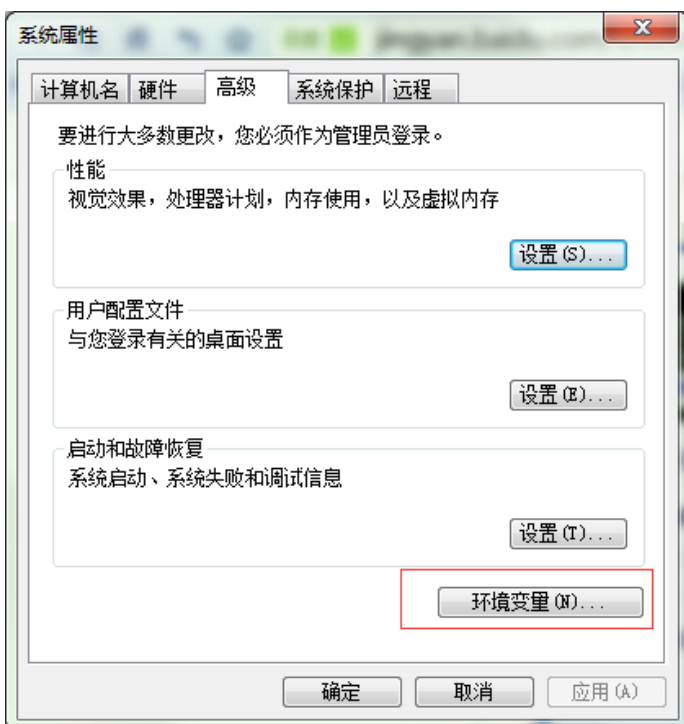


图 1.2.2

3、出现环境变量界面后我们需要配置系统变量中的三个变量，它们是“JAVA\_HOME 环境变量”、“PATH 环境变量”、“CLASSPATH 环境变量”。

#### （一）JAVA\_HOME 环境变量

首先，配置“JAVA\_HOME”。JAVA\_HOME 是 JDK 的安装目录。配置“JAVA\_HOME”我们就需要找到 JDK 的安装目录。右击地址栏上的地址，选择【复制地址】。如图（1.2.3）

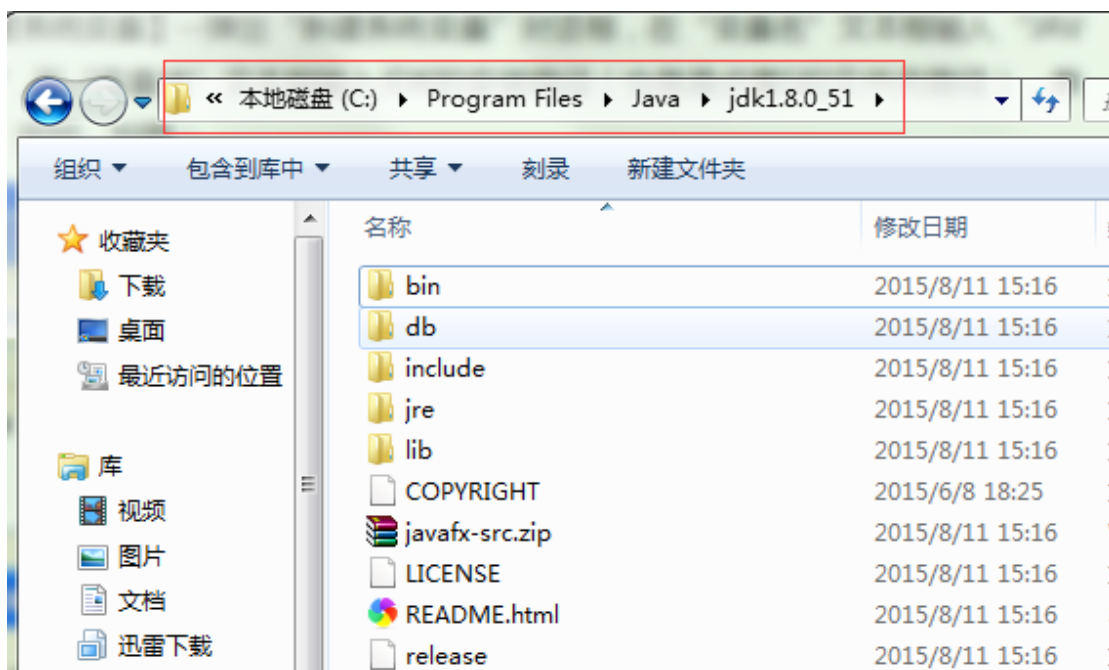


图 1.2.3

复制地址后，我们返回到环境变量配置界面，系统变量下面的【新建】。弹出新建窗口，变量名填入“JAVA\_HOME”，变量值填入你刚刚复制的 JDK 的地址。（如图 1.2.4）

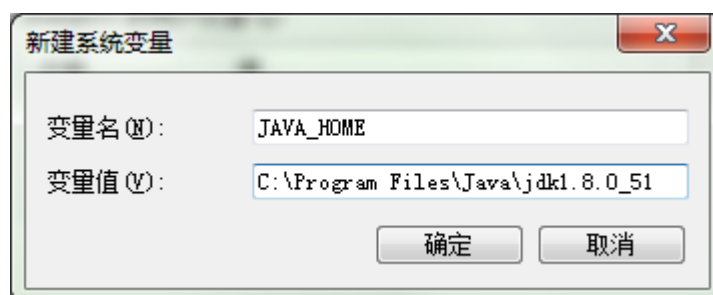


图 1.2.4

## （二）PATH 环境变量

在“系统变量”选项区域中查看 PATH 变量，如果不存在，则新建变量 PATH，否则选中该变量，单击“编辑”按钮，在“变量值”文本框的起始位置添加“%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin;”或者是直接“%JAVA\_HOME%\bin;”（不要落下分号，分号是用来与其他变量值隔开来的)单击确定按钮。（如图 1.2.5）。

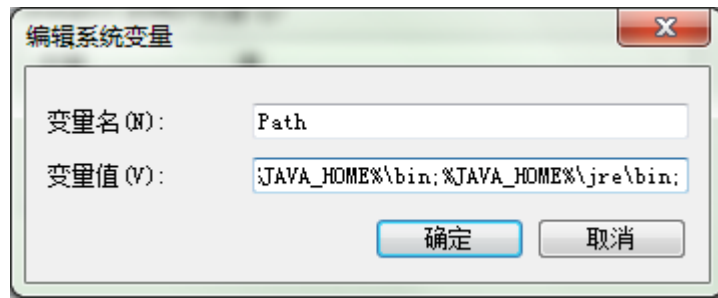


图 1.2.5

### (三) CLASSPATH 环境变量

在“系统变量”选项区域中查看 CLASSPATH 变量，如果不存在，则新建变量 CLASSPATH，否则选中该变量，单击“编辑”按钮，在“变量值”文本框的起始位置添加“.;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar;”。（不要落下前面的“.”，）（如图 1.2.6）

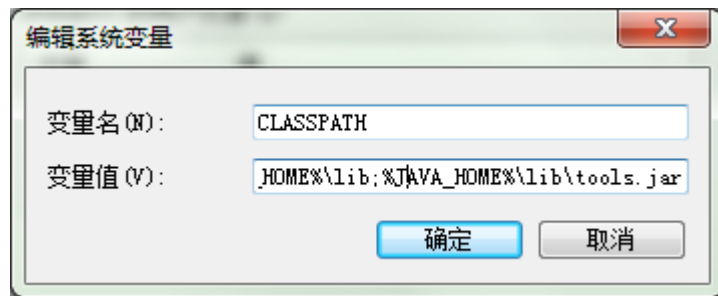
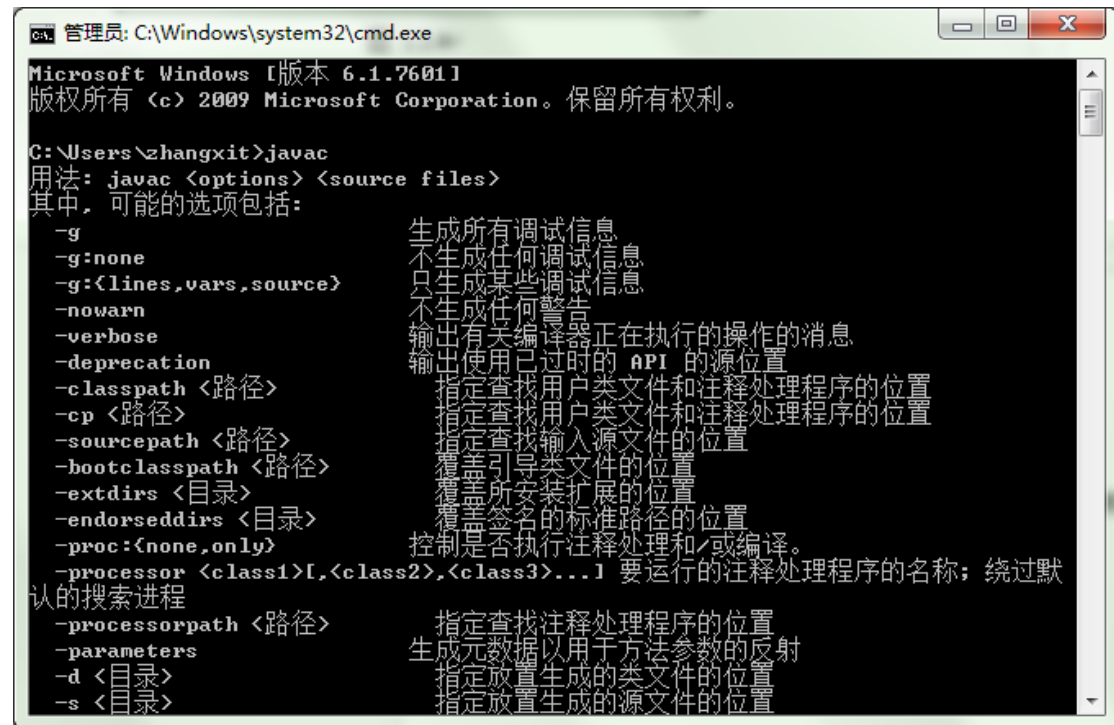


图 1.2.6

#### (四) 验证环境变量是否配置成功

打开 DOS 窗口，输入“javac/java”等 java 代码编辑程序。若出现其用法，说明已经配置成功。如图（1.2.7）



```
ca. 管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Users\zhangxit>javac
用法: javac <options> <source files>
其中, 可能的选项包括:
    -g                  生成所有调试信息
    -g:none             不生成任何调试信息
    -g:<lines,vars,source> 只生成某些调试信息
    -nowarn             不生成任何警告
    -verbose            输出有关编译器正在执行的操作的消息
    -deprecation        输出使用已过时的 API 的源位置
    -classpath <路径>   指定查找用户类文件和注释处理程序的位置
    -cp <路径>          指定查找用户类文件和注释处理程序的位置
    -sourcepath <路径>  指定查找输入源文件的位置
    -bootclasspath <路径> 覆盖引导类文件的位置
    -extdirs <目录>     覆盖所安装扩展的位置
    -endorseddirs <目录> 覆盖签名的标准路径的位置
    -proc:<none,only>   控制是否执行注释处理和/或编译。
    -processor <class1>[,<class2>,<class3>... ] 要运行的注释处理程序的名称; 绕过默
    认的搜索进程
    -processorpath <路径> 指定查找注释处理程序的位置
    -parameters        生成元数据以用于方法参数的反射
    -d <目录>           指定放置生成的类文件的位置
    -s <目录>           指定放置生成的源文件的位置
```

图 1.2.7

至此，Java 开发环境搭建完毕。



## 二、android studio 的安装

### 2.1 Windows 下 android studio 的下载与安装

1、下载地址：<http://developer.android.com/sdk/index.html>（若无法下载请查看常见问题解答）

2、点击“Develop”在Tools菜单下选择“Download”，网站可以识别到当前电脑的系统，如实

验电脑是 Windows 系统，所以提供了 Windows 系统的下载链接。如下图所示：

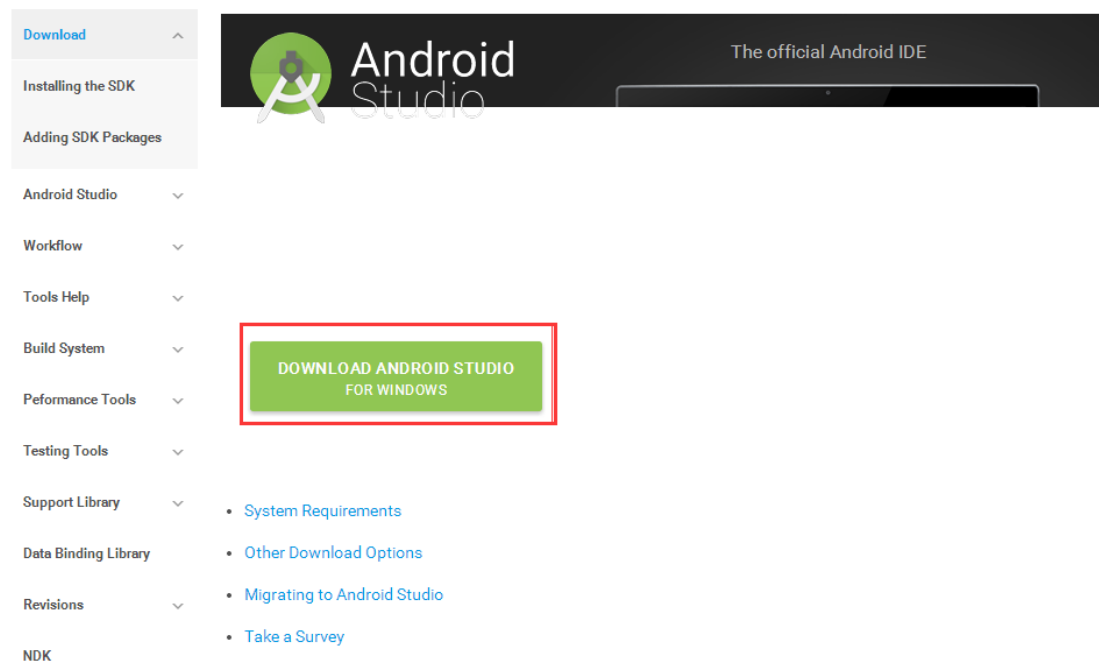


图 2.1.1

3、点击“Download Android Studio for Windows”后，勾选“I have read and agree with the above terms and conditions”，开始下载。如图2. 1. 2所示：

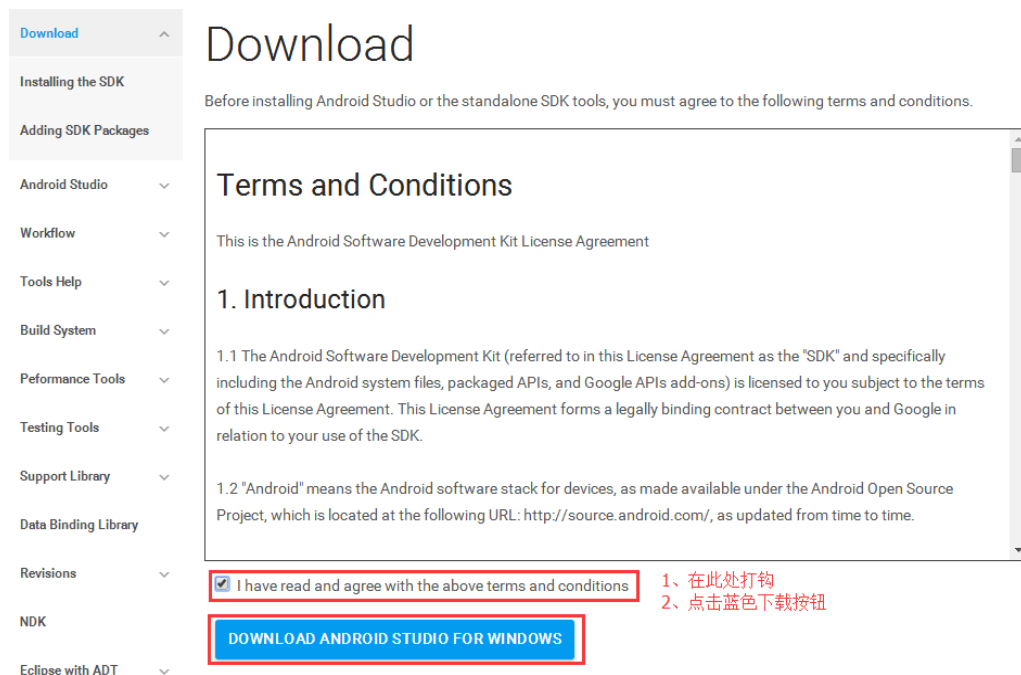


图 2.1.2

如果需要安装其他版本可以选择“Other Download Options”，如图2. 1. 3所示：

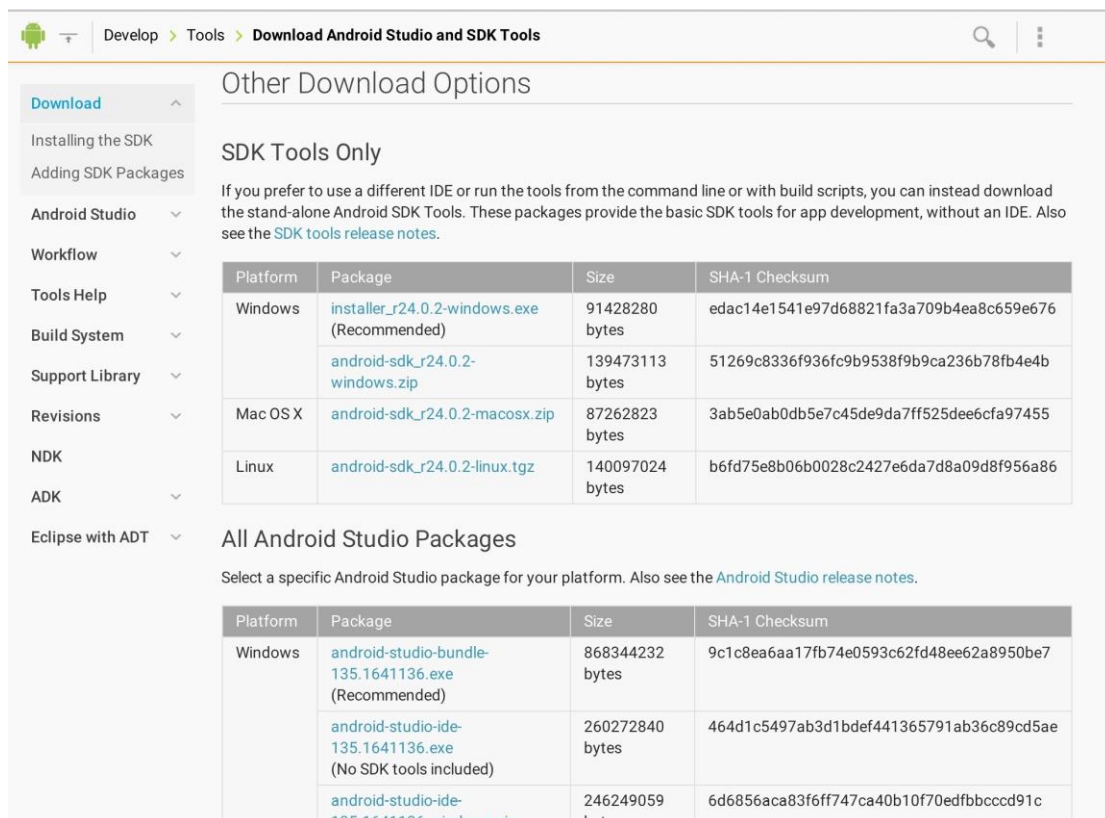


图 2.1.3

下载后，双击下载完成的安装包，进行安装，点击“Next”继续。如图 2.1.4 所示：

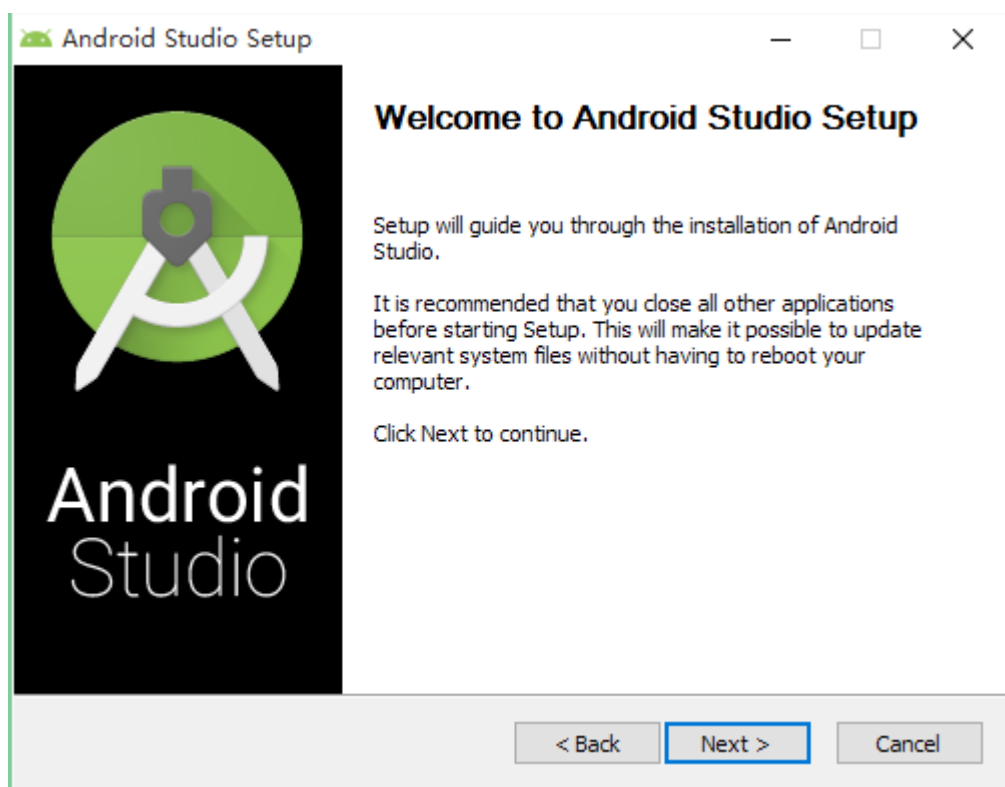


图 2.1.4

出现以下界面，继续点击“Next”继续，如图 2.1.5所示。

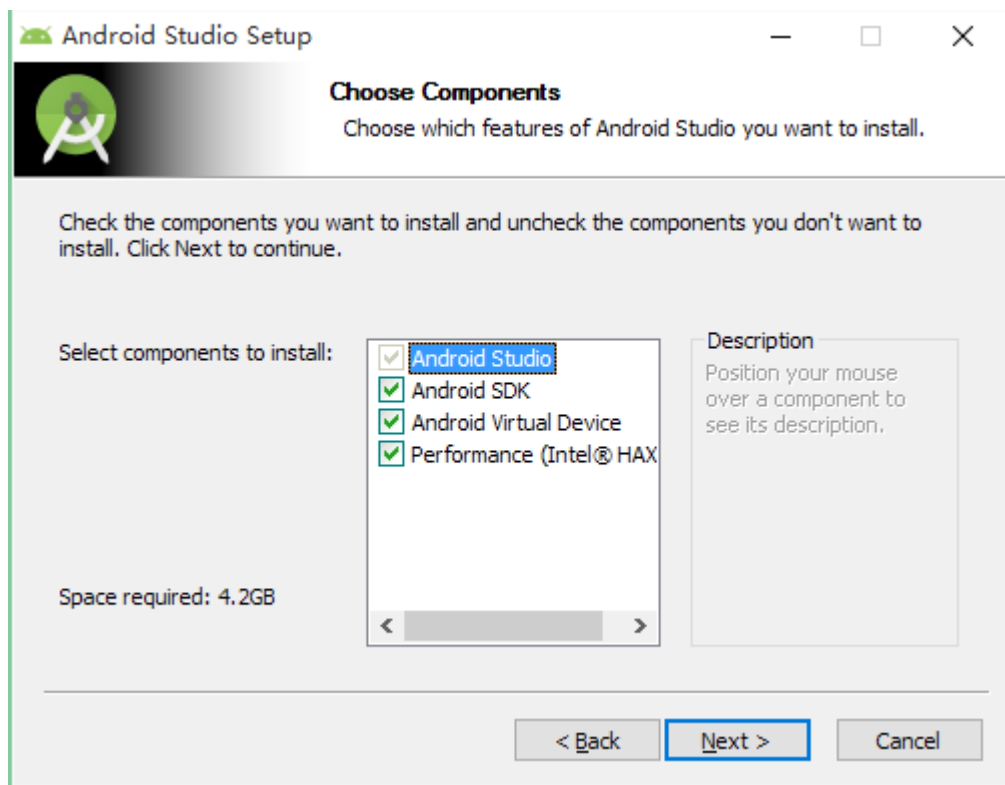


图 2.1.5

点击“I Agree”继续安装，如图 2.1.6 所示。

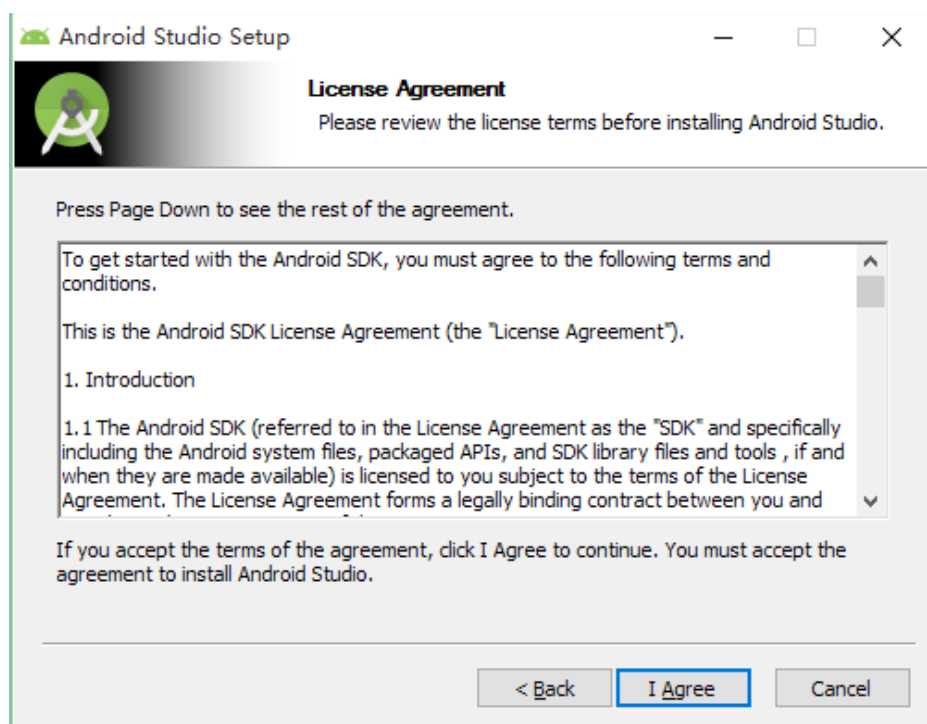


图 2.1.6

在这里，根据你的硬盘容量进行安装。但是，除非你的C盘有足够大的空间，否则我不建议你安装在C盘。图中是我为自己选择的安装路径。分别为android studio和SDK选择好路径后，点击“Next”继续，如图 2.1.7所示。

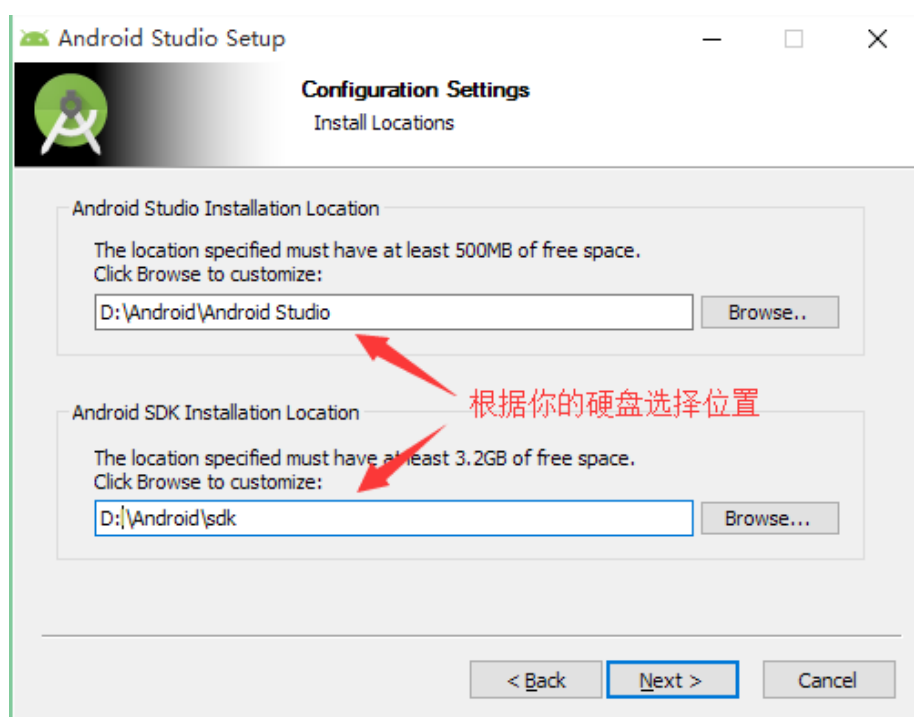


图 2.1.7

继续点击“Next”执行下一步，如图 2.1.8所示。

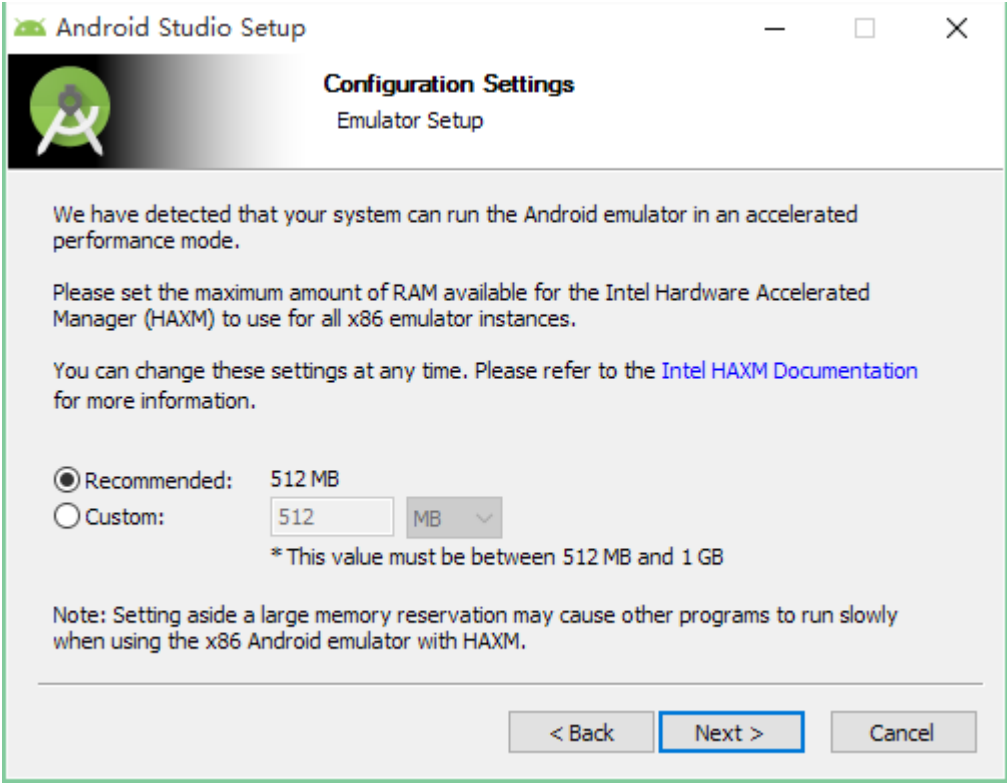


图 2.1.8

点击“Install”执行安装命令，这里可能需要耗费一小会儿的时间，如图 2.1.9所示。

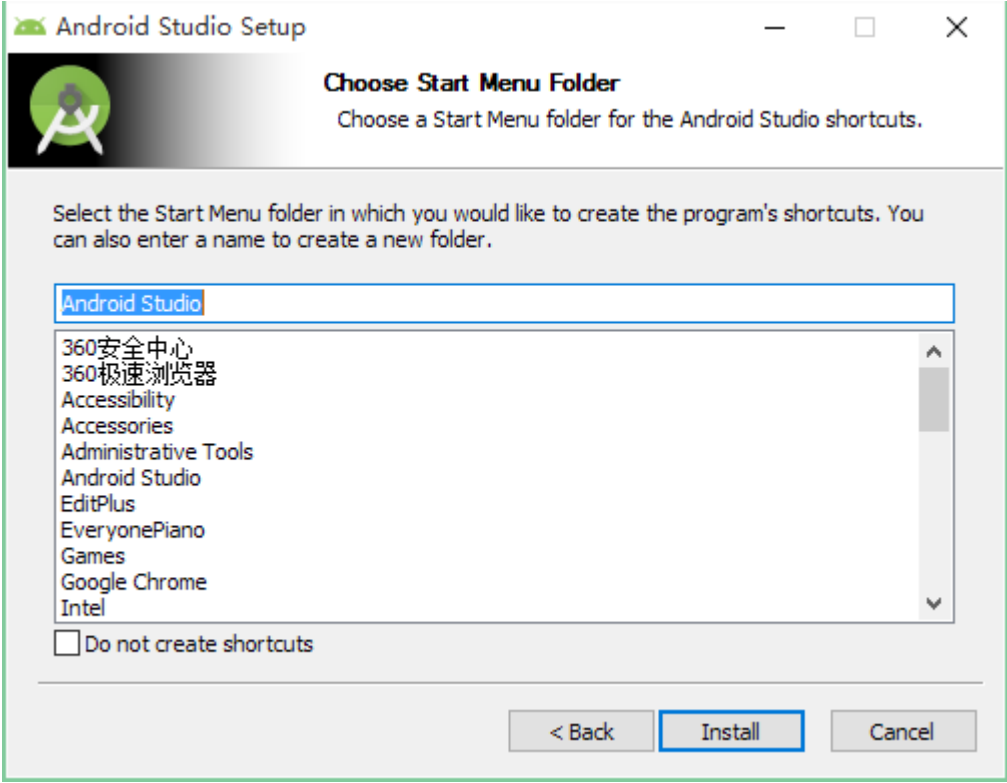


图 2.1.9

点击“Next”继续，如图 2.1.10所示。

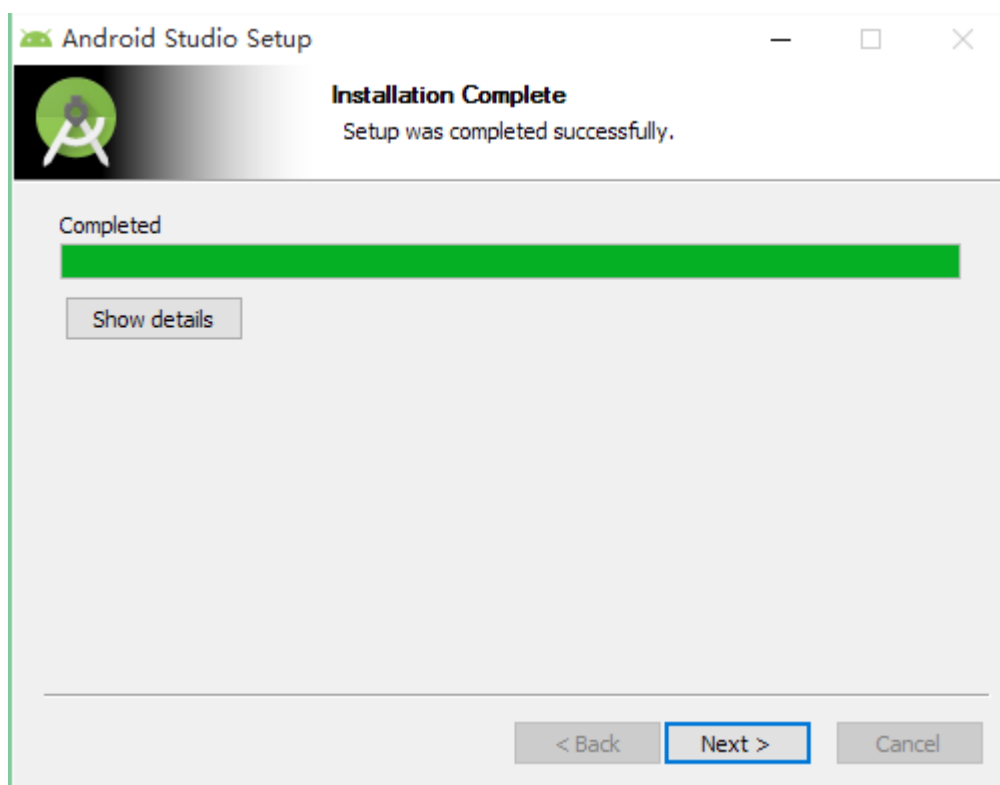


图 2.1.10

点击“finish”按钮来完成安装，如图 2.1.11所示。

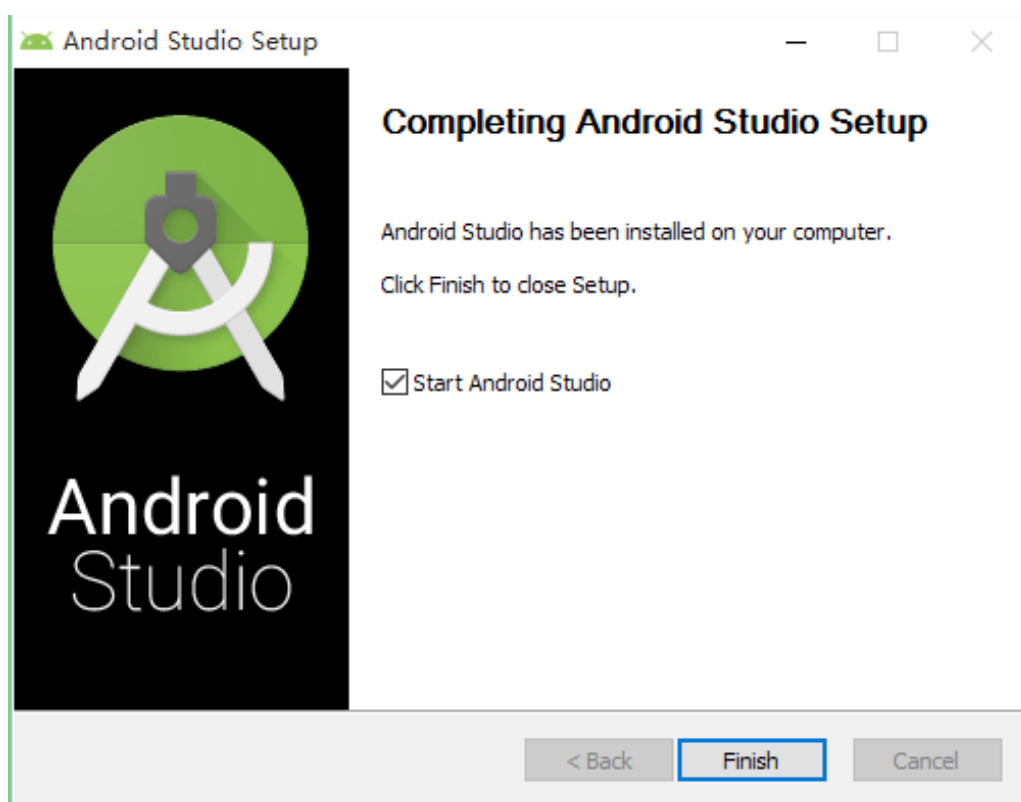


图 2.1.11

如果之前有安装过Android Studio 的其他版本，可以选择导入设置（第一个选项），否则选择第二个选项。然后点击“OK”按钮，如图 2.1.12所示。

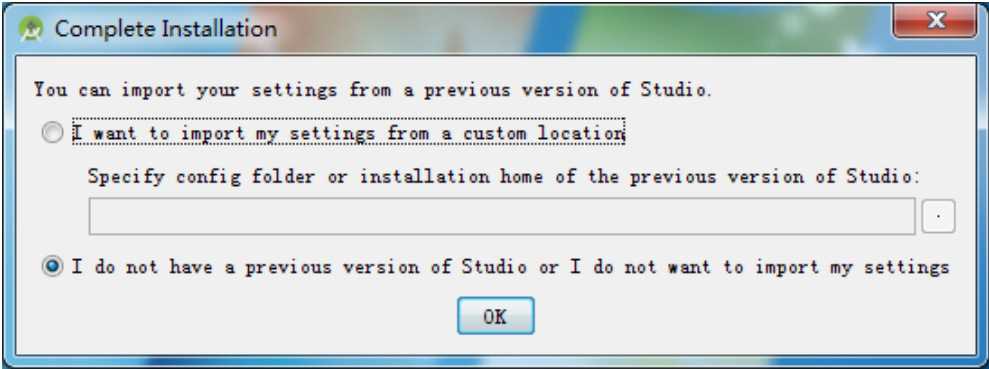


图 2.1.12

接下来是选择主题界面，如图2.1.13，选择你喜欢的主题后，点击“Next”继续，。

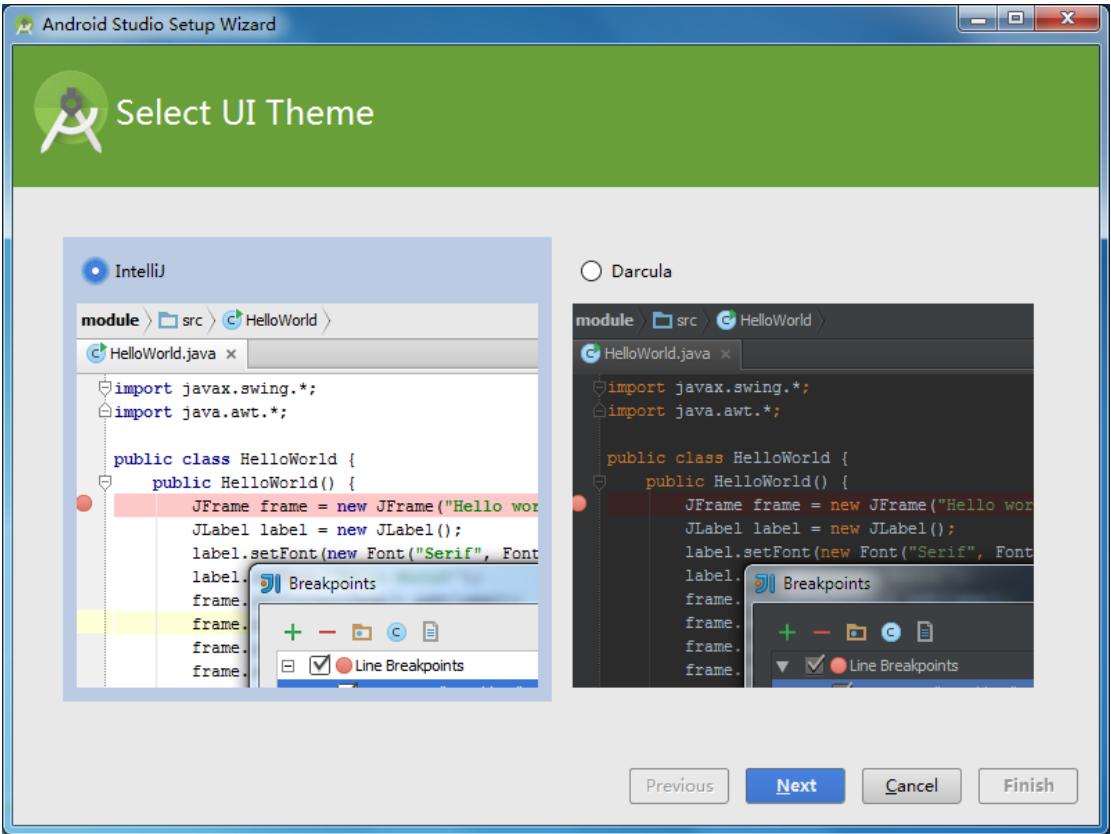


图 2.1.13

接下来的界面，点击“finish”来完成安装（这一步如果遇到问题请参照4.5小节解决问题）。如图 2.1.14

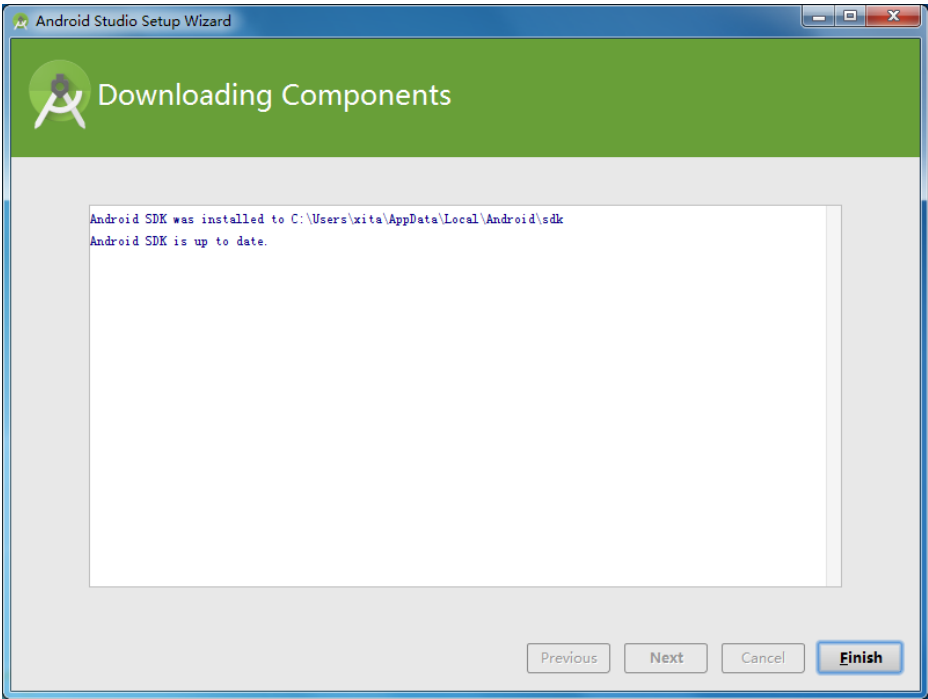


图 2.1.14

## 2.2 更新 SDK

点击“Finish”后，会出现下面这个界面，点击Configure。如图2.1.15所示：

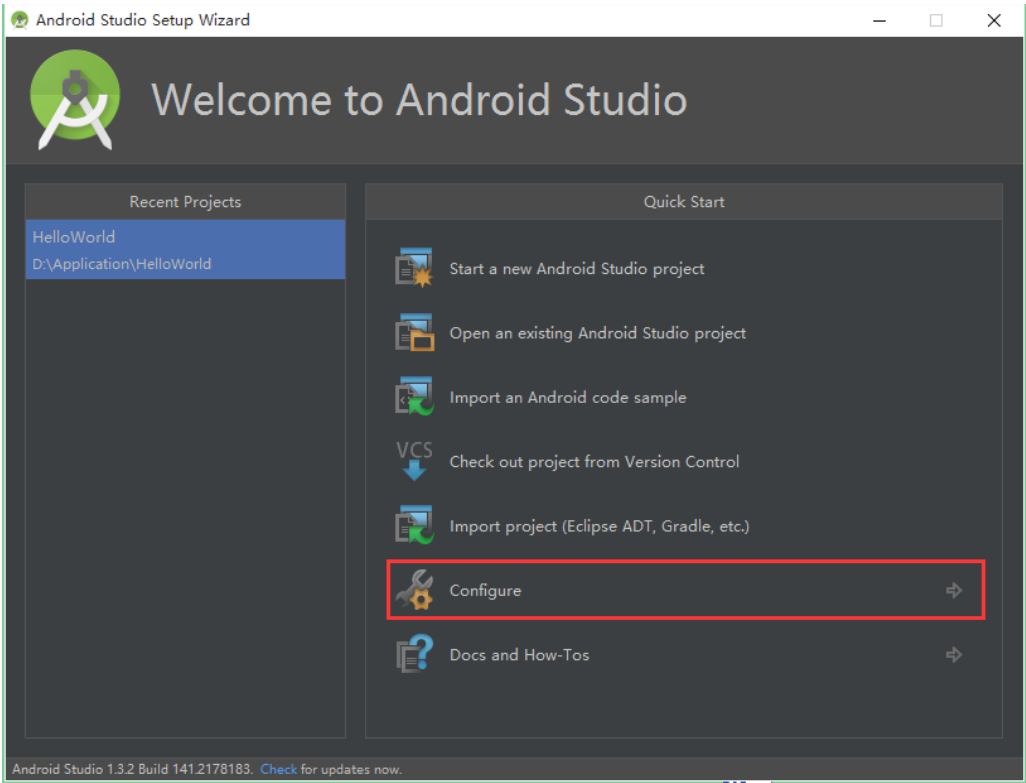


图 2.2.1



然后再次点击 “SDK Manager”，进行SDK更新，如图2. 2. 2：

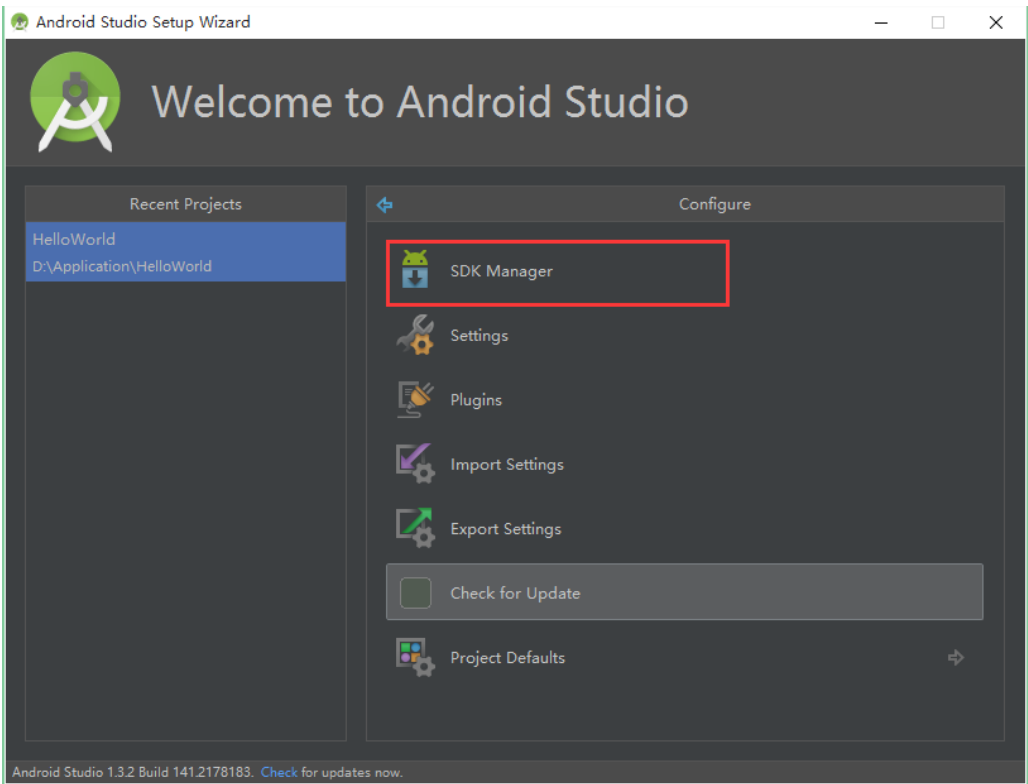


图 2. 2. 2

来到SDK Manager界面（如图 2. 2. 3，图2.2. 4），我们可以看到有各种版本的开发包，勾选“Tools”“Extras”后，再选择课程需要版本的开发包（课程需要Android 5. 0. 1 API 21），下载后即可编写安卓程序。

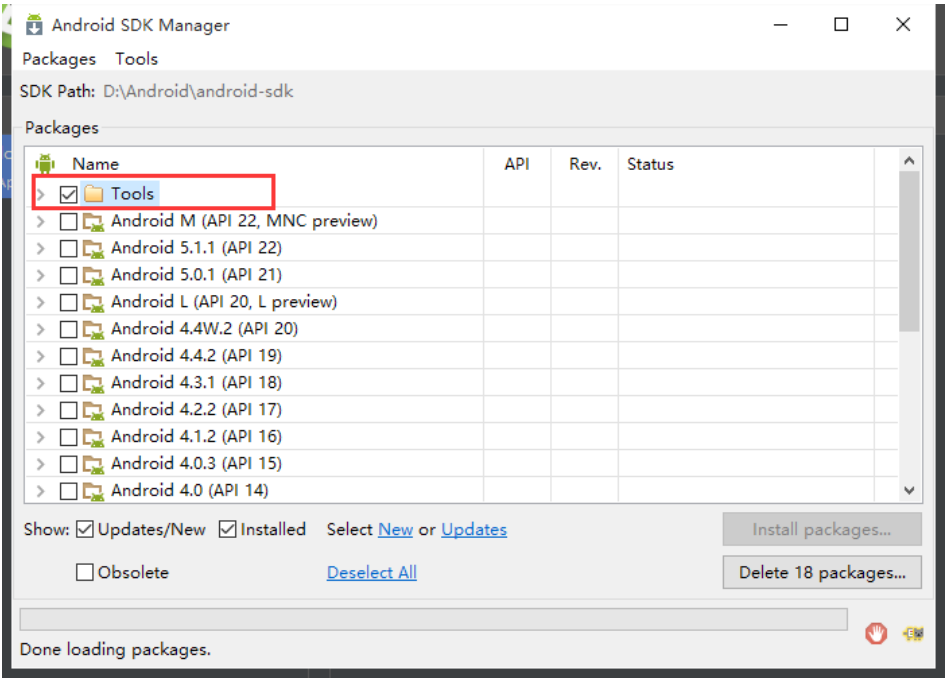


图 2. 2. 3

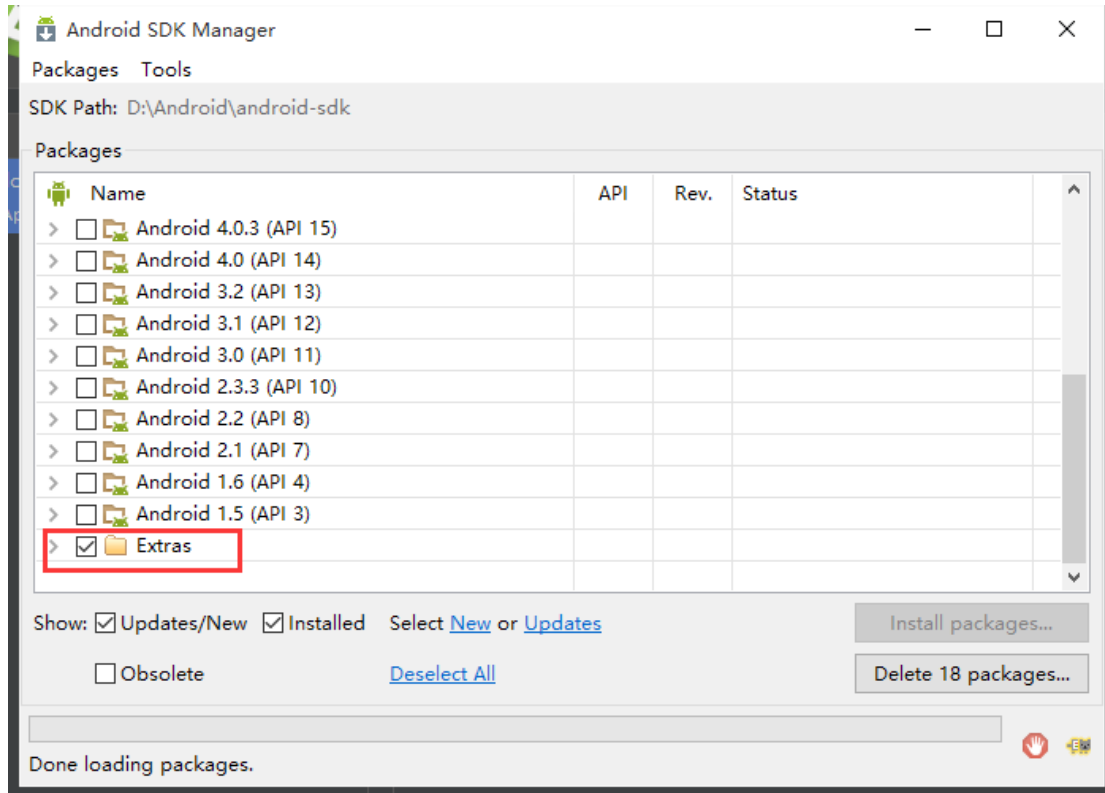


图 2.2.4

2.3 Intel Atom x86 模拟器的安装

Intel Atom x86 的模拟器比标准的 Android 模拟器具有更高的执行效率，所以推荐安装 Intel Atom x86 模拟器。

首先要下载 Intel 硬件加速执行管理器加速模拟器。

下载地址：<https://software.intel.com/zh-cn/>

1、点击“工具”菜单中的“安卓”选项，如图 2.3.1 所示：



图 2.3.1

2、点击“英特尔 HAXM”选项，如图 2.3.2 所示。

使用英特尔工具最大限度地提高应用的性能



使用英特尔工具最大限度地提高应用的性能

**新功能**

**英特尔® INDE >**

一组跨平台的原生开发工具套件，用于创建面向安卓\*和 Microsoft Windows\* 的应用

**新功能**

**英特尔® Mobile Dev Kit >**

获得一台 Root 的设备和英特尔 System Studio，以用于系统和中间件安卓\* 开发。

**已更新**

**英特尔® XDK >**

开发、模拟、设备上测试及构建跨平台应用。

**英特尔® HAXM >**

也是英特尔® INDE 的一个特性。在开发用计算机上加速安卓应用的模拟。

**英特尔® C++ 编译器 (适用安卓) >**

英特尔® INDE 的一个特性。提高应用的响应性，延长电池续航时间。

**Project Anarchy\* >**

利用 Havok™ 开发的这款免费跨平台移动游戏引擎来推进您的应用。.

图 2.3.2

3、找到适合自己的操作系统。（本机为 windows）然后点击相应的下载包。本机如图 2.3.3 所示：

Microsoft Windows\*

匹配自己的操作系统

Windows 8 和 8.1（32/64 位）、Windows 7 32/64 位）、Windows Vista\*（32/64 位）

安装指南和系统要求 - Windows

haxm-windows\_r05.zip (1.1.1)

也是英特尔® Integrated Native Developer Experience（英特尔® INDE/英特尔® 集成原生开发人员体验）的一个特性

描述：  
系统驱动程序  
(2014 年 10 月 24 日)

大小：1.7MB

校验和：  
(MD5)  
8305796fb4b414dfc196517d4771dbb5  
(SHA-1)  
feccd79e6a2a1f3dd65db5f4bfe04a9b7f09b481

图 2.3.3

4、点击“接受许可”进行下载，如图 2.3.4 所示：

免责声明。  
此软件由版权持有者和献助者按“原样”提供，绝不提供其他任何明确或隐含的担保（包括，但不限于，商品适销性和或适用于特定目的适用性的担保）。在任何情况下，版权持有者或 献助者对使用本软件而以任何方式产生之直接的、间接的、事故性的、特殊的、惩罚性的或后果性的损失（包括，但不限于，购买替代产品或服务，使用、数据或利润的减少，或者业务中断）概不承担责任，不论损失是如何造成的及根据任何责任理论（无论是按合同法、严格责任或侵权 - 包括疏忽或其它 -），即使事先被告知这种损失的可能性。

点击以接受《最终用户许可协议》并下载 haxm-windows\_r03.zip

有关编译器优化的更完整信息，请参阅[优化通知](#)。

类别: 英特尔硬件加速执行管理器（英特尔 HAXM），安卓\*，电话，开发人员，安卓\*，Microsoft Windows\* (XP, Vista, 7)

图 2.3.4

5、下载后进行安装（首次安装可能会出现安装失败）如图 2.3.5:

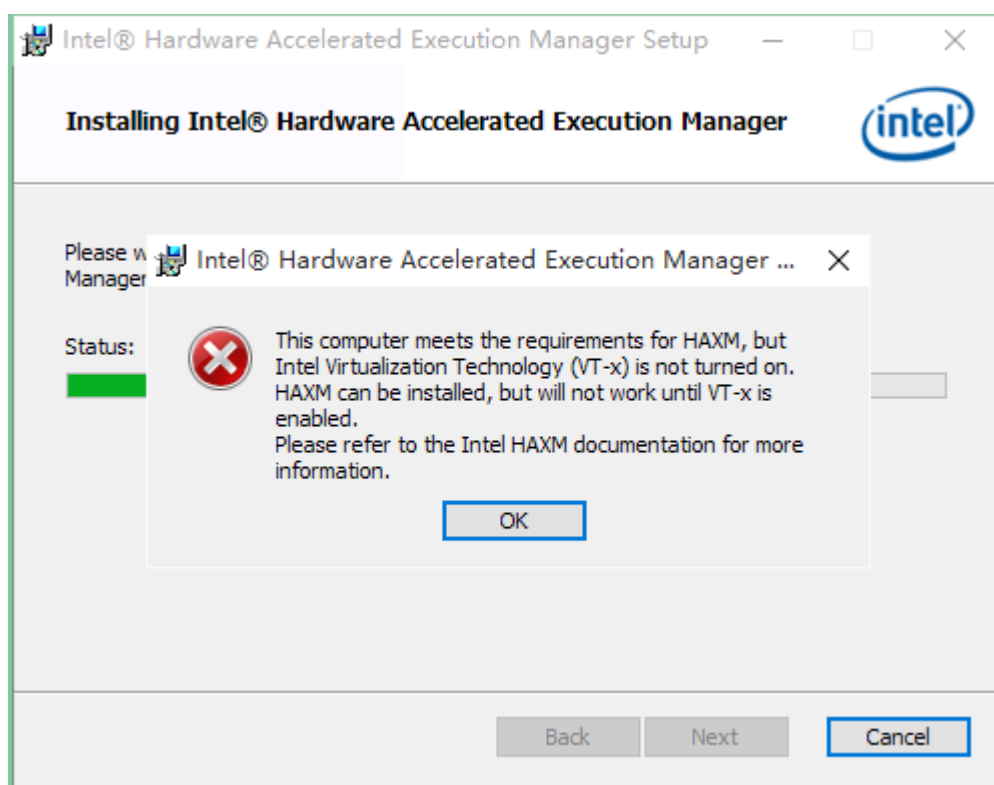


图 2.3.5

这是因为电脑的主板未开启“Virtualization”选项。

解决办法：进入你的 BIOS。找到 Virtualization。将他打开。

由于这一步不同的品牌机操作步骤不同。可以自己摸索。

如图 2.3.6:

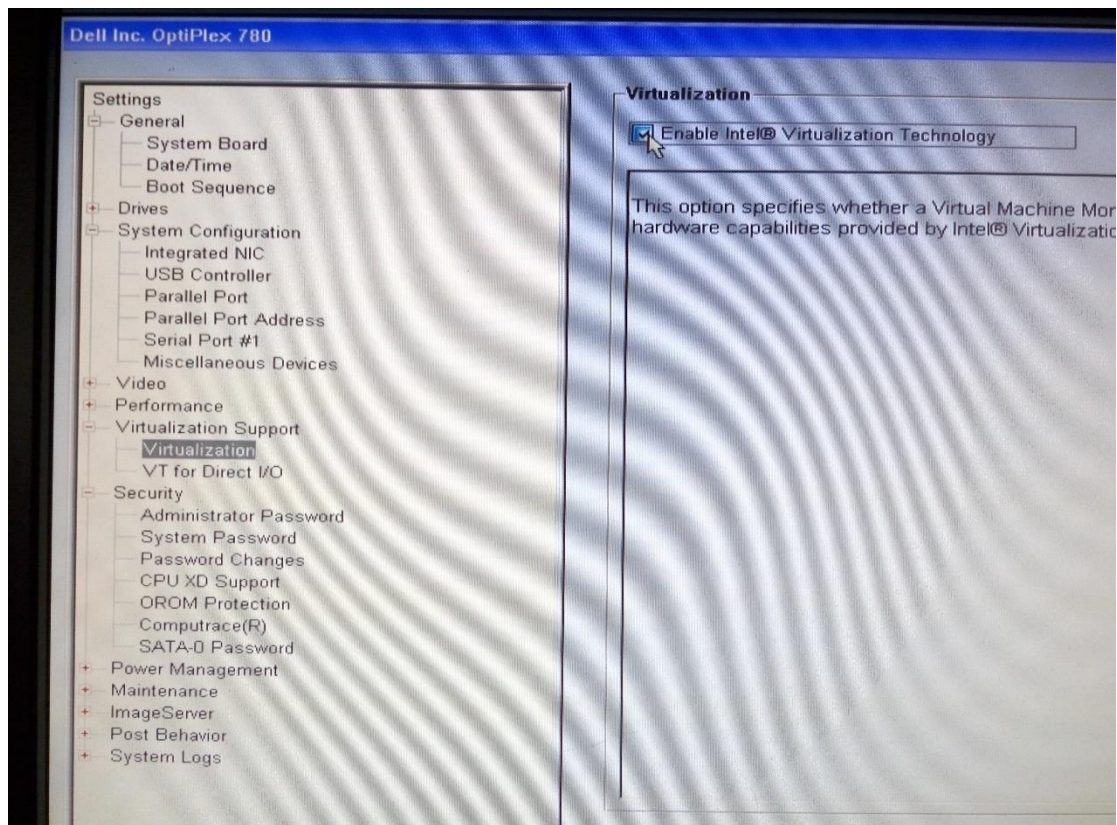


图 2.3.6

6、重启后继续安装，双击安装包，点击“Next”继续，如图 2.3.7。



图 2.3.7

7、继续点击“Next”继续，如图 2.3.8。

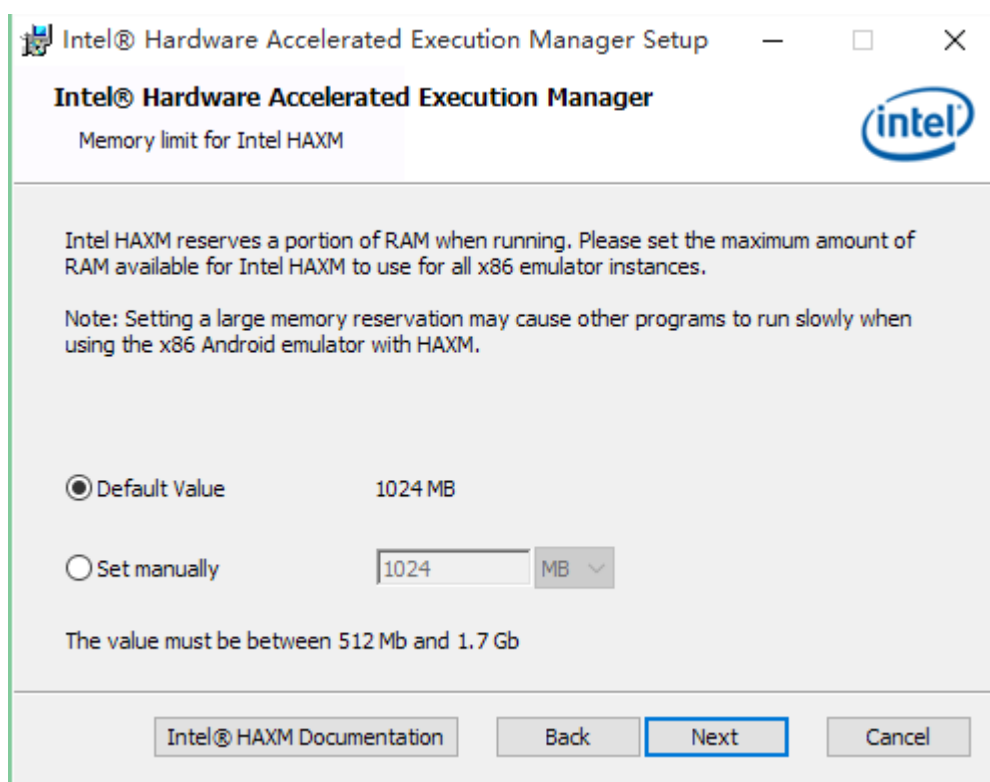


图 2.3.8

8、点击“Install”执行安装命令，如图 2.3.9。

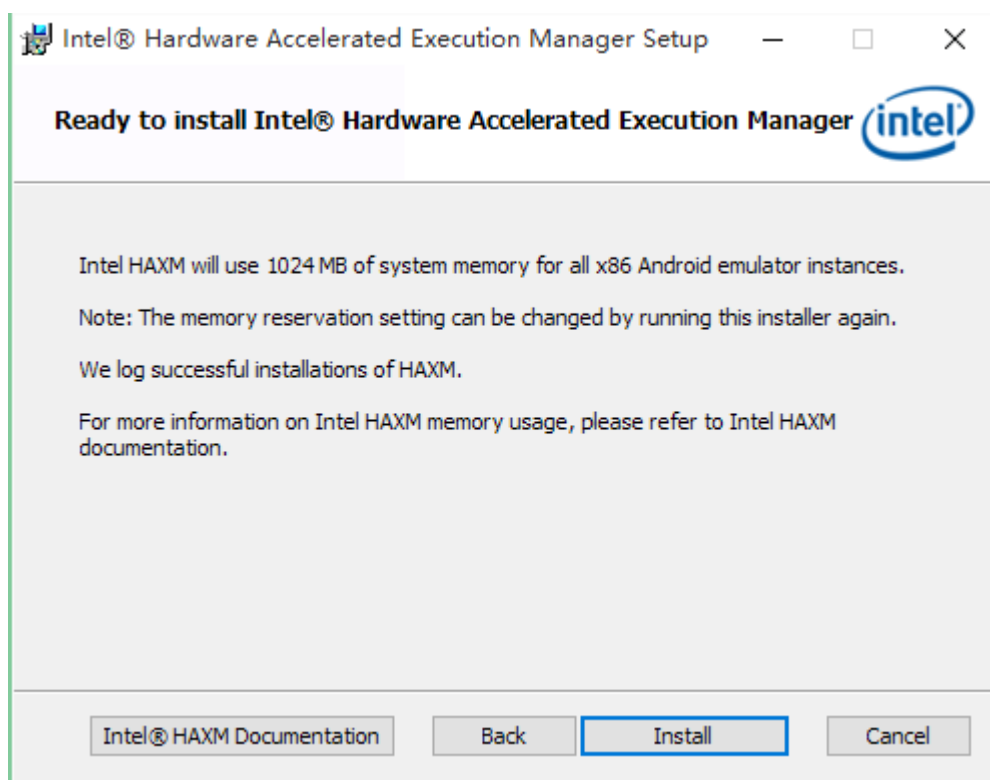


图 2.3.9

9、点击“Finish”以完成安装，如图 2.3.10。

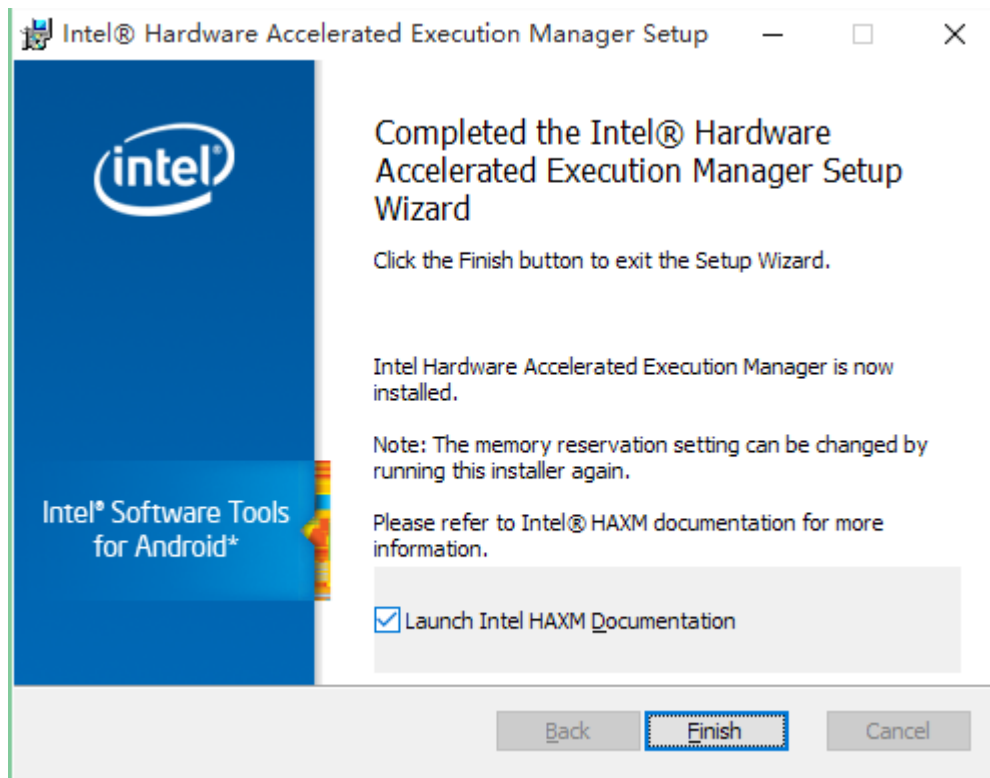


图 2.3.10



## 三、使用 android studio

### 3.1 第一个 Android 程序——Hello World

点击“Start a new Android Studio project”来开始创建一个android程序。如图

3.1.1所示：

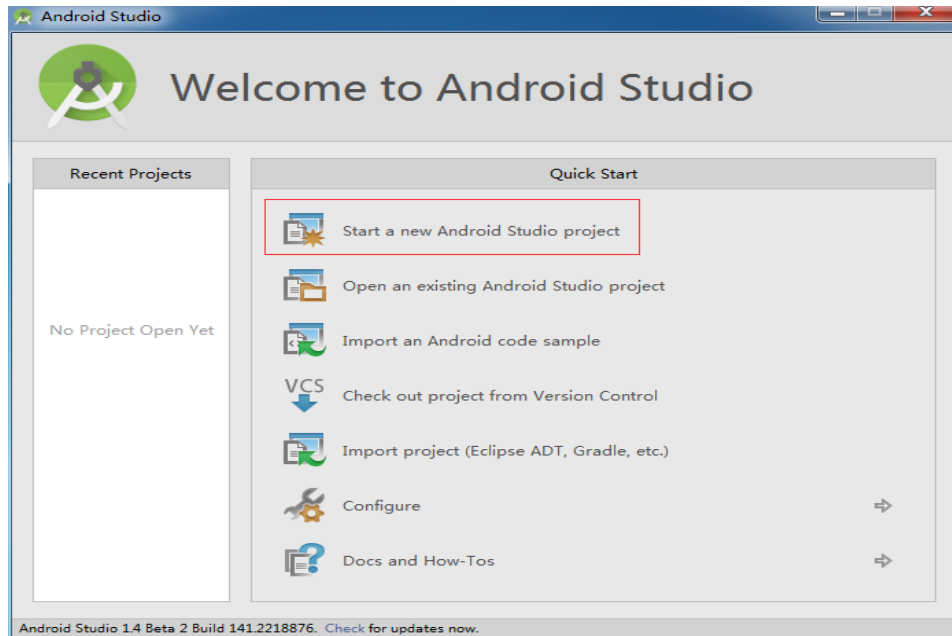


图 3.1.1

为你的程序起一个名字，“Application name”就是将来你做的这个 App 的名字，本例命名为“Hello World”如图 3.1.2。

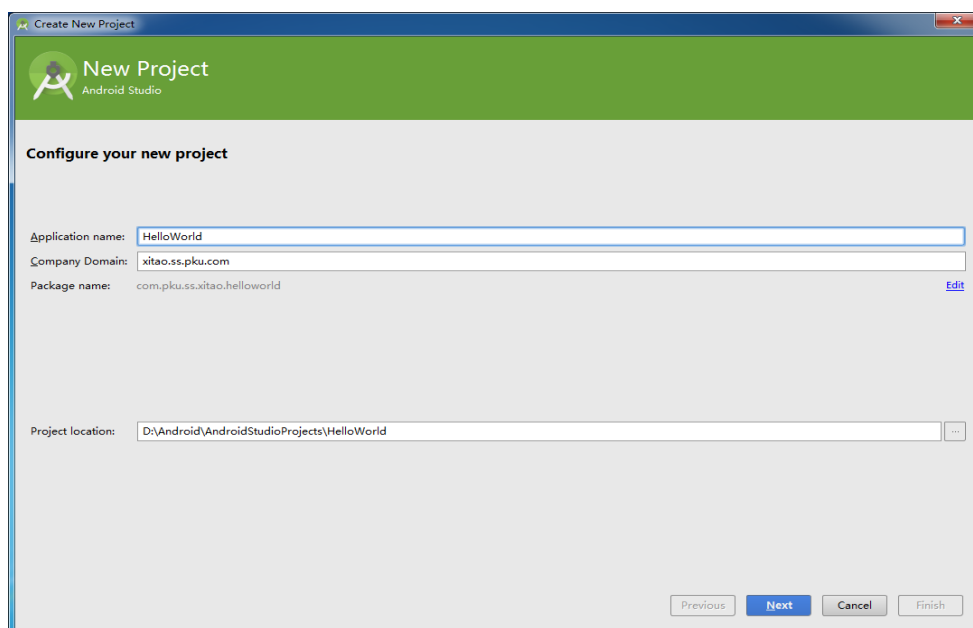


图 3.1.2

图 3.1.3 这里选择你的代码服务与什么类型的安卓设备和最小支持的SDK版本。

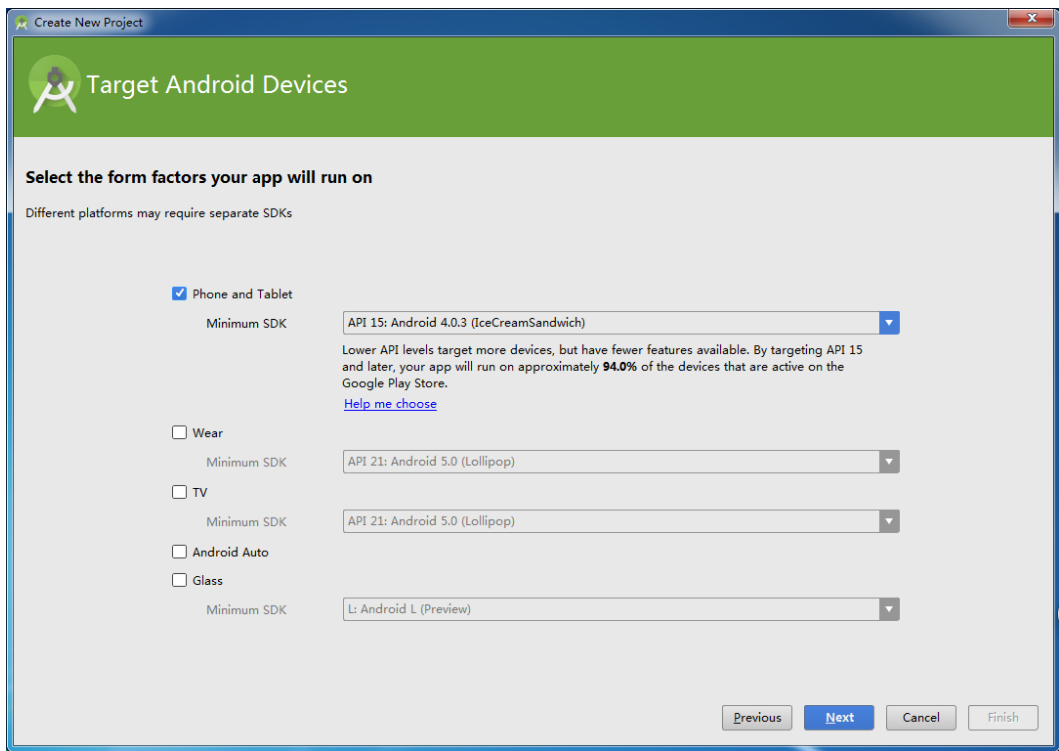


图 3.1.3

这里我们选择“Blank Activity”（如图 3.1.4），这个选项，android studio 会帮我们自动建一个“Hello World”的 Activity 以及一些默认的设置。我们可以在这个程序上进行修改。如果你想自己创建这些东西，你应该选择“Add No Activity”。

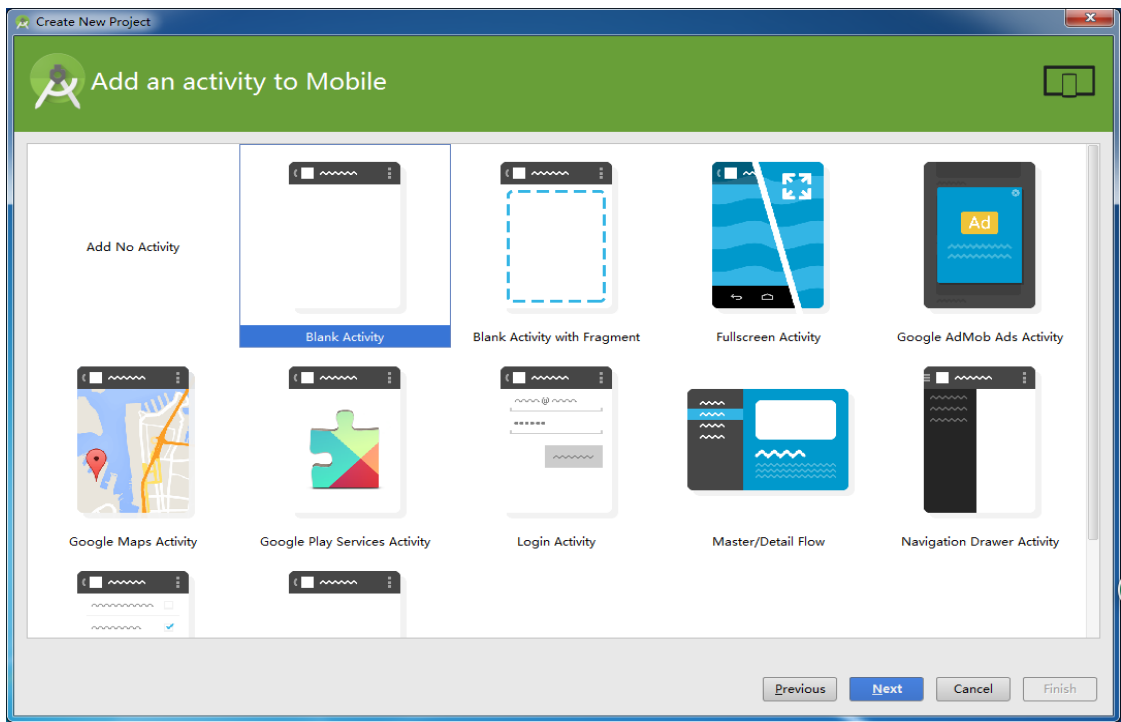


图 3.1.4

在这里你可以修改这些文件的名字，然后点击“finish”来完成工程的创建，如图 3.1.5所示。

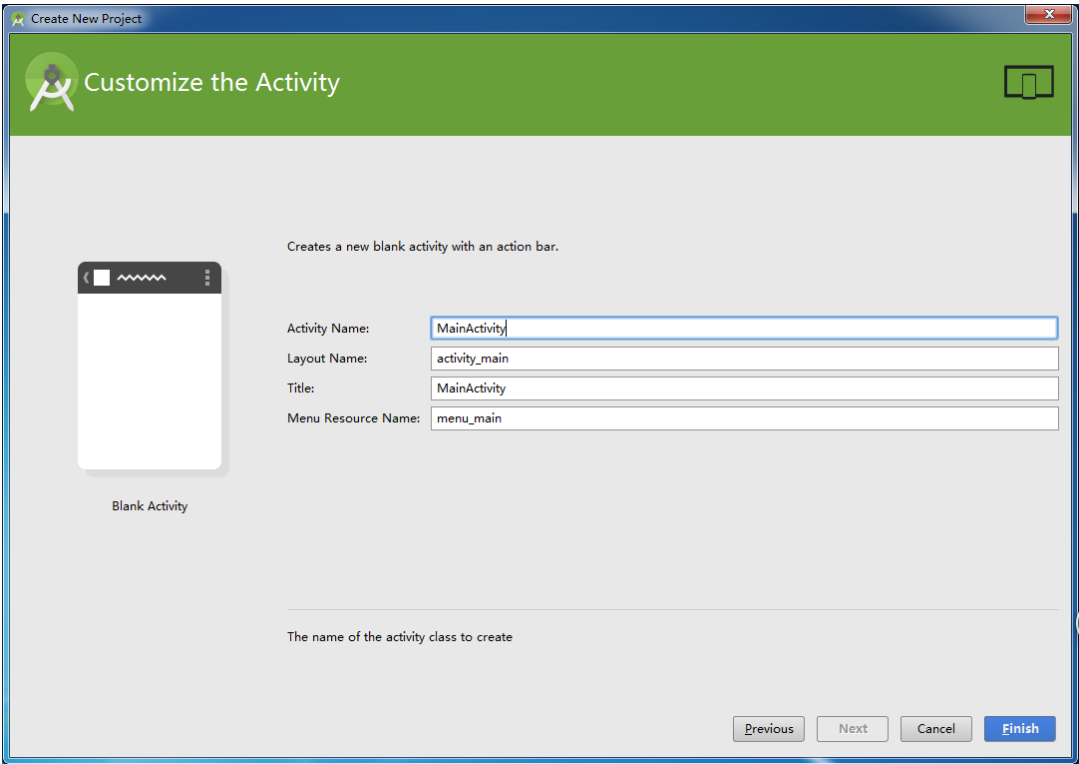


图 3.1.5

至此，你已经创建了一个“Hello World”工程并且可以对这个工程进行修改，界面图如图 3.1.6 所示。

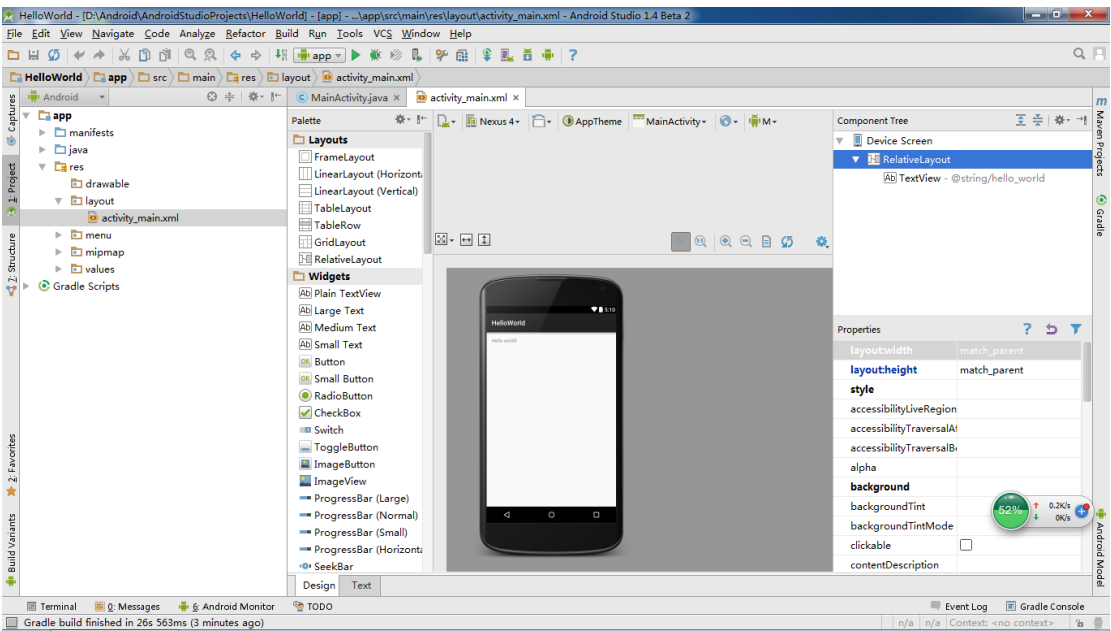


图 3.1.6

### 3.2 创建一个 AVD 以运行程序

AVD（Android Virtual Device）是一个可以安卓虚拟设备。我们可以用它来当做测试机来调试我们的程序。

同更新SDK一样，你有两种方式去创建一个AVD。你可以点击android studio面板上的AVD图标，如图 3.2.1，也可以直接找到SDK安装路径下的“AVD Manager”。点击运行。

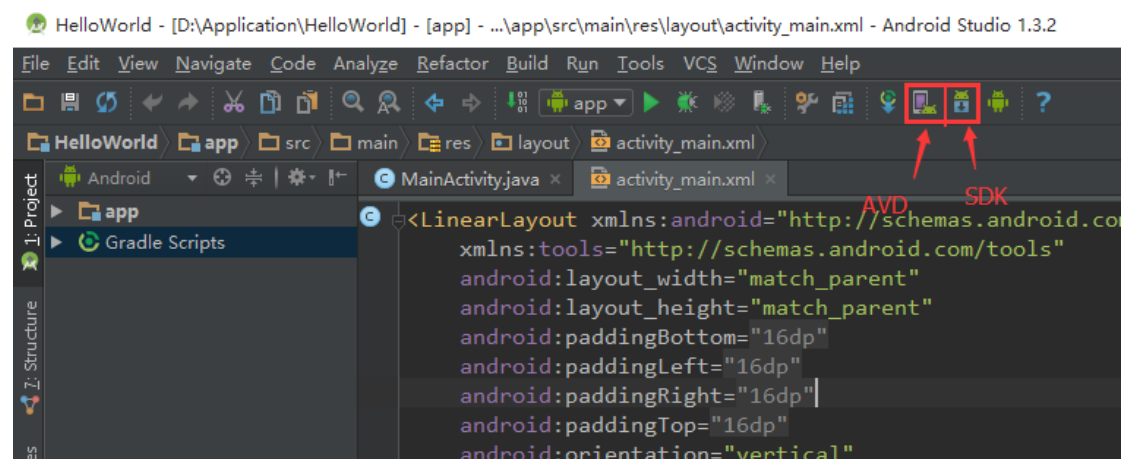


图 3.2.1

在android studio里面点击AVD Manager图标会出现图 3.2.2，这时点击“Create Virtual Device”. 来创建一个AVD。

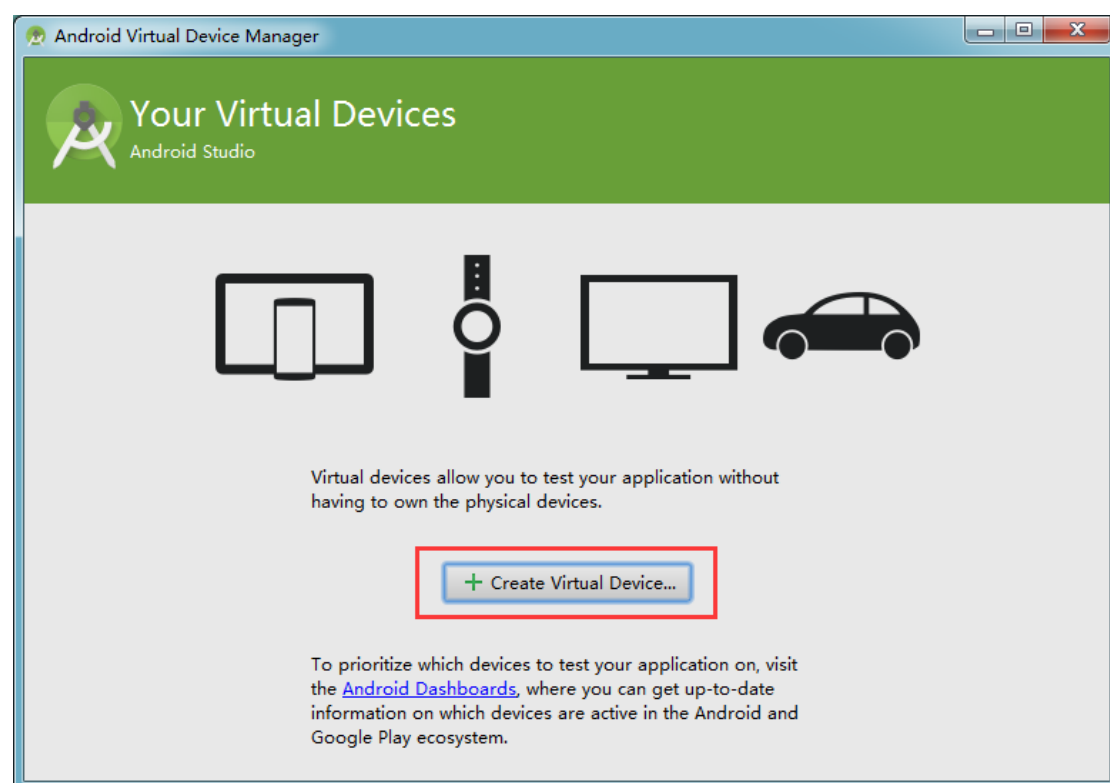


图 3.2.2

接下来会“Select Hardware”界面（如图 3.2.3），选择合适的硬件后，点击“Next”。

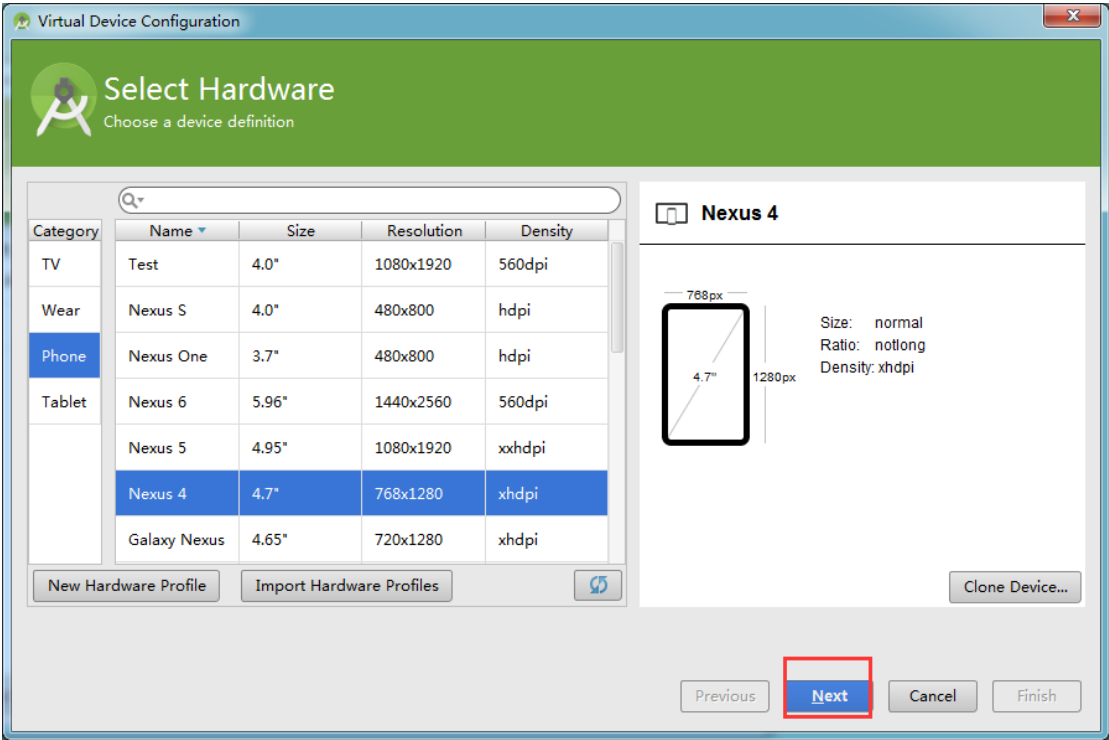


图 3.2.3

这里为“System Image”界面（如图 3.2.4），是让我们选择一个合适的安卓系统镜像。由于我们已经安装Intel HAXM（Intel Atom x86模拟器）。这里我们选择一个X86的ABI。点击“Next”继续。

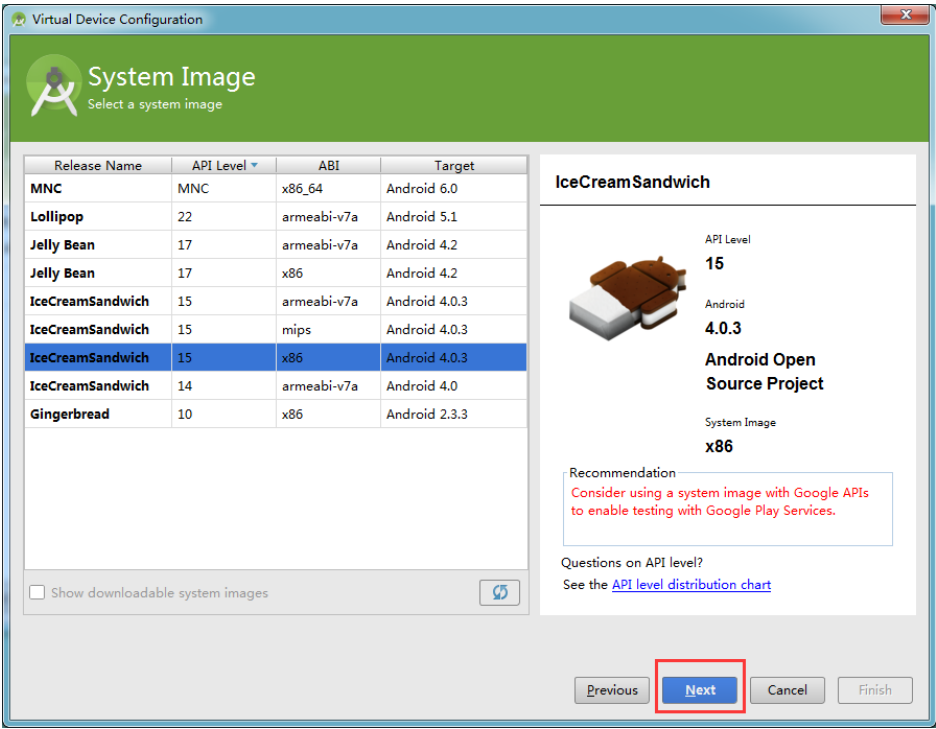


图 3.2.4

这里你可以为你的AVD命名，并且设置它的一些参数。然后点击Finish完成（如图 3.2.5）。

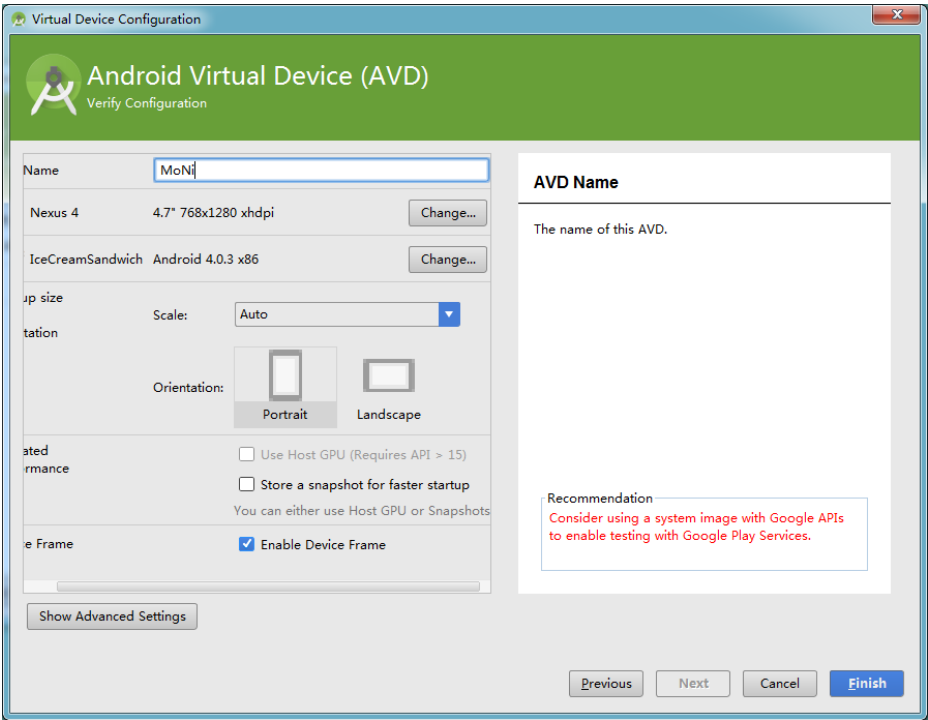


图 3.2.5

当你创建一个或几个不同的AVD时候，你需要启动并使用它。点击右方“Action”中的绿色三角形来运行它（如图 3.2.6）。

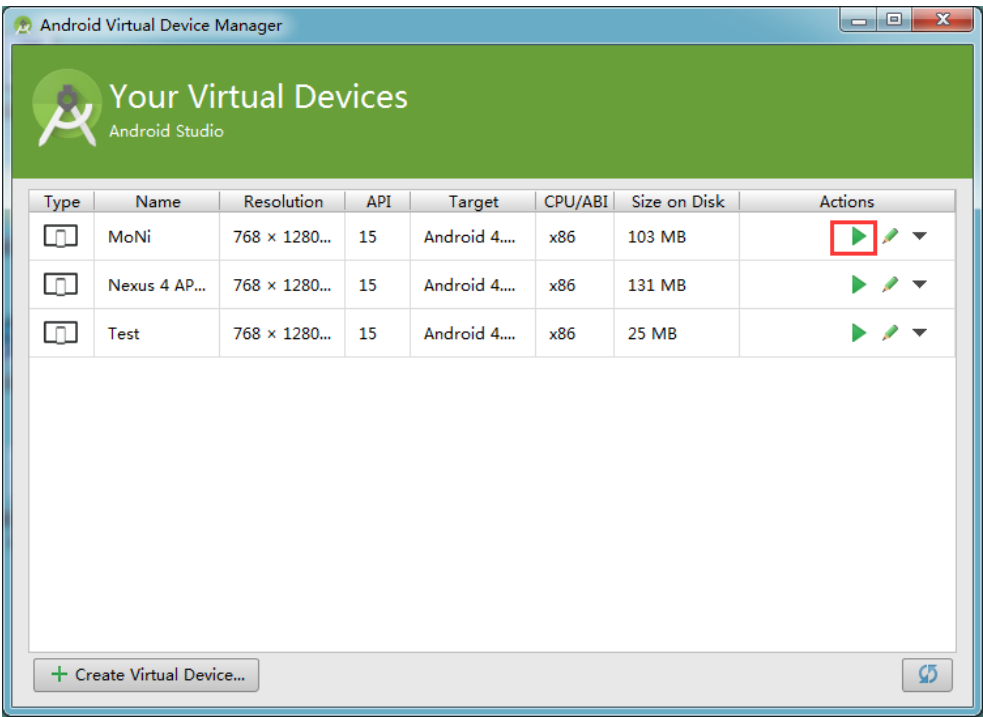


图 3.2.6

根据PC的不同，启动时间会有所不同，有些机器第一次启动会失败，这时候我们可以先关闭，再次启动。（首次启动时间比较长）

启动完毕后，我们可以点击android studio中的运行按钮，如图3.2.7：

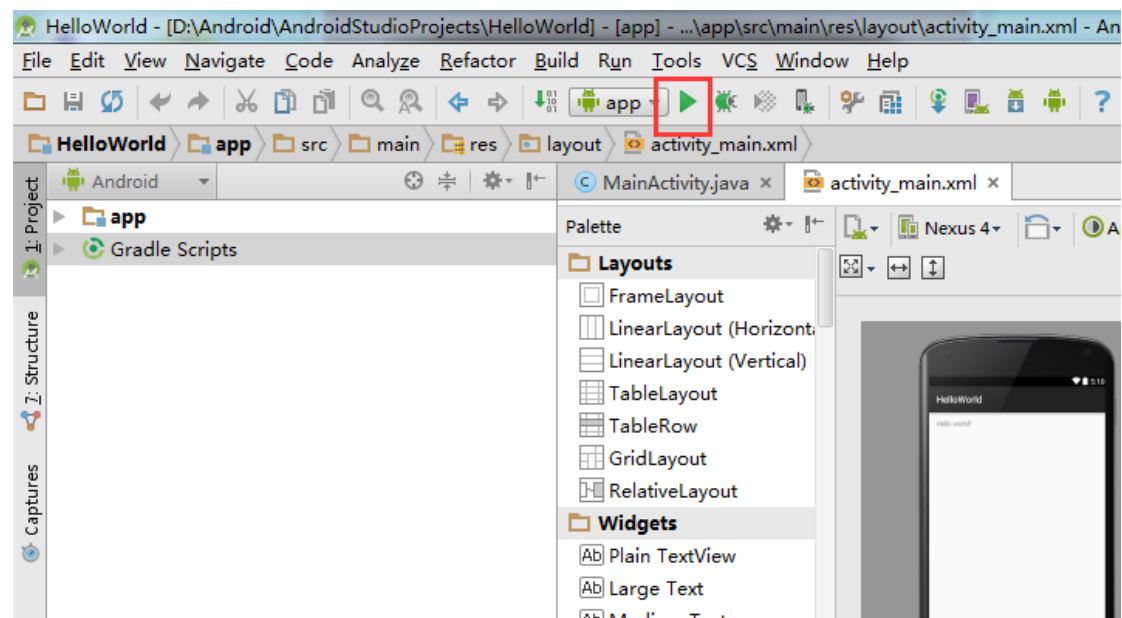


图 3.2.7

选择一个可以利用的设备，点击OK，如图 3.2.8

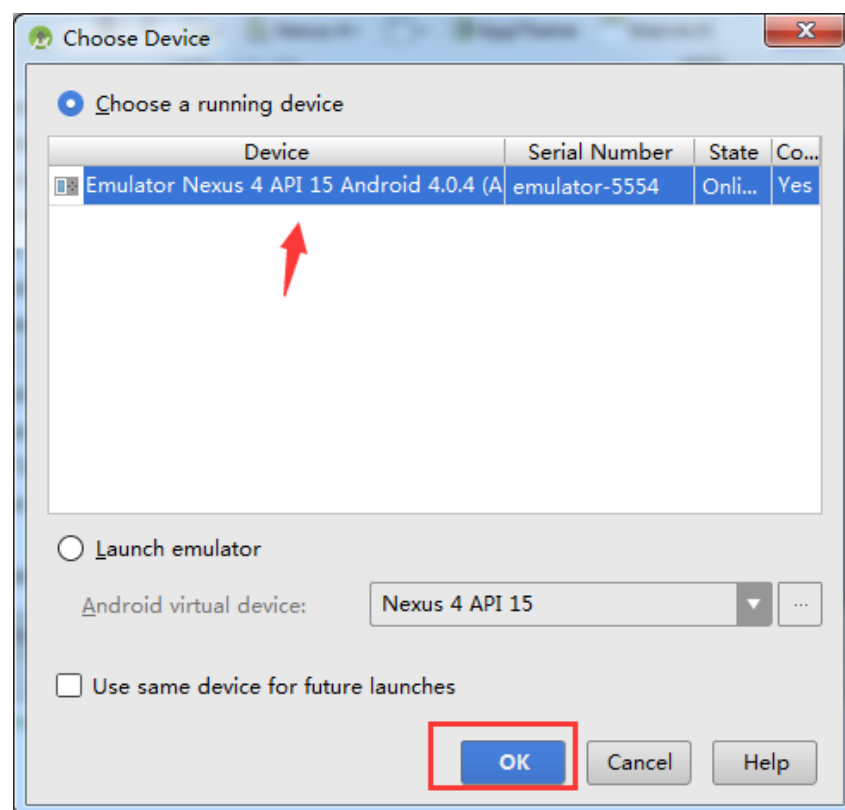


图 3.2.8

我们的HelloWorld可以运行在设备上，如图 3.2.9。



图 3.2.9



### 3.3 调试程序

调试程序设备选择分为两种：虚拟设备调试、真机调试。

#### 1、虚拟设备

在上一节，我们讲过利用AVD调试程序，这里不再赘述。大家也可以去网上下载一些目前比较流行的虚拟设备软件来使用。

#### 2、真机调试

- （一）将符合你所开发程序API等级的测试机利用数据线连接USB接口。
- （二）安装驱动（可以利用一些像豌豆荚、91手机助手等手机管理辅助工具）
- （三）进入手机开发者选项，打开手机的USB调试功能，如图3. 3. 1



图3. 3. 1

- （四）点击“一直允许用这台计算机进行调试”如图3. 3. 2



图 3. 3. 2

此时android studio可以识别安卓手机，如果你的程序出现“no debuggable applications”问题，请点击“Tools→Android→Enable ADB Integration”然后再运行程序，就好了，如图3.3.3。

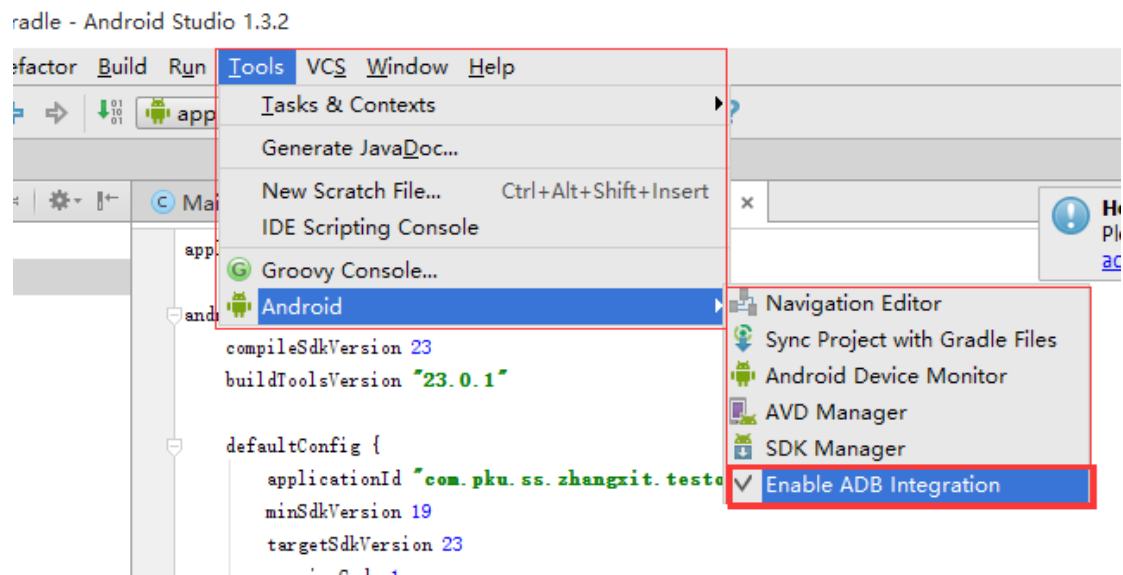


图 3.3.3

（五）点击运行按钮，出现设备选择界面，如图 3.3.4

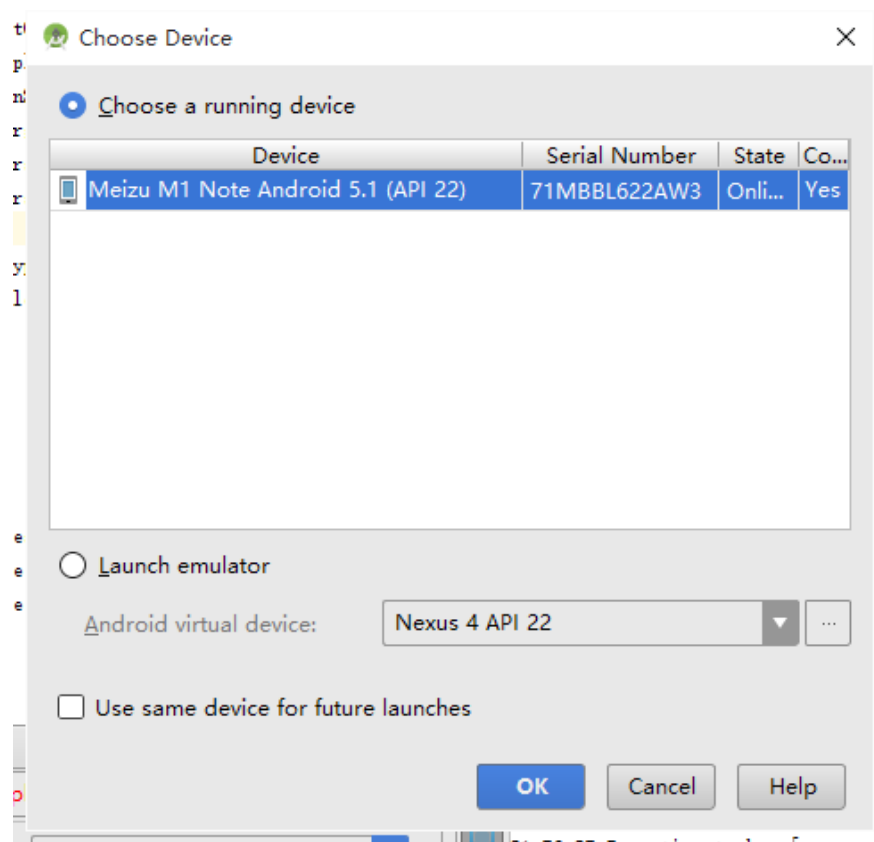


图 3.3.4

点击OK后程序开始运行，如图 3.3.5

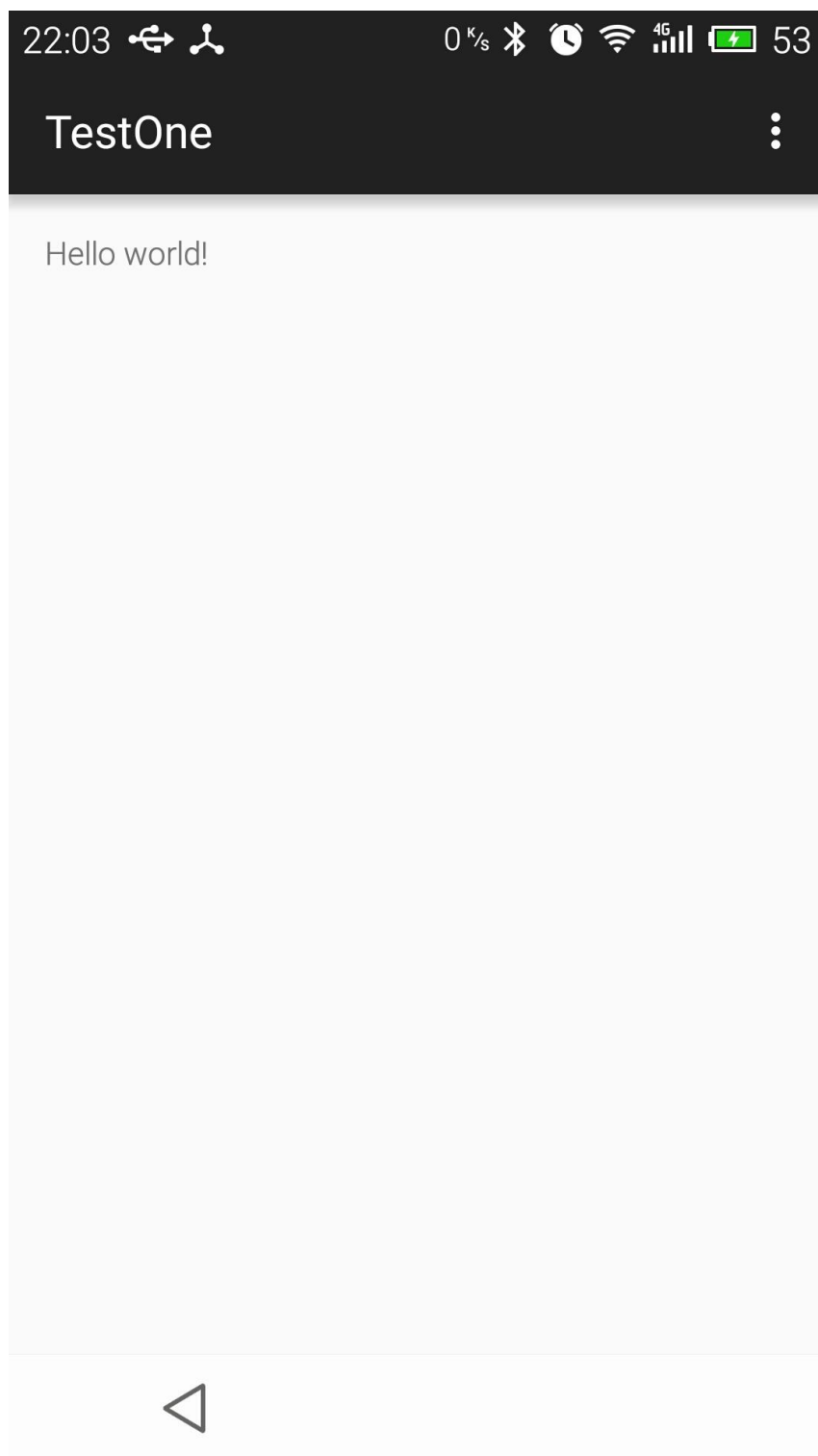


图 3.3.5

如果你想将你的设备设为默认(即点击运行按钮后,不出现设备选择按钮,直接运行程序),你可以点击“run→Edit configurations”(如图 3.3.6),然后选择“USB device”即可(如图 3.3.7)

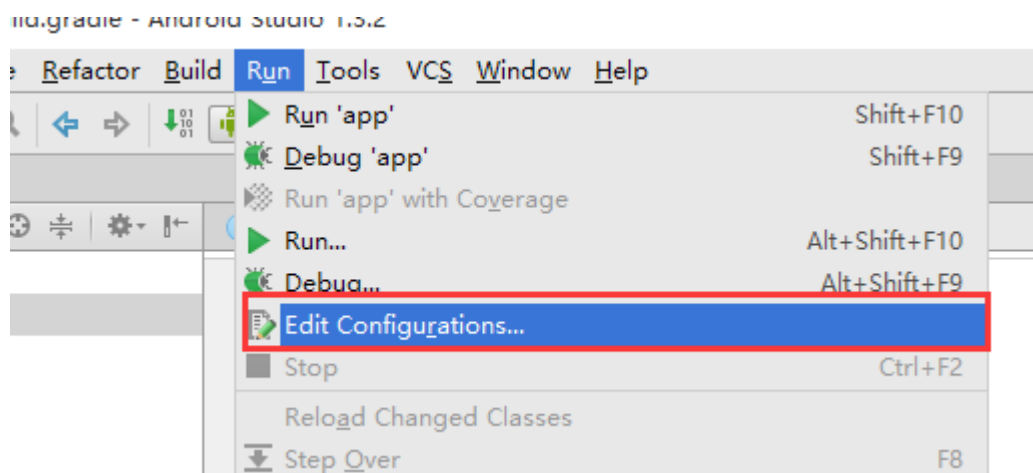


图 3.3.6

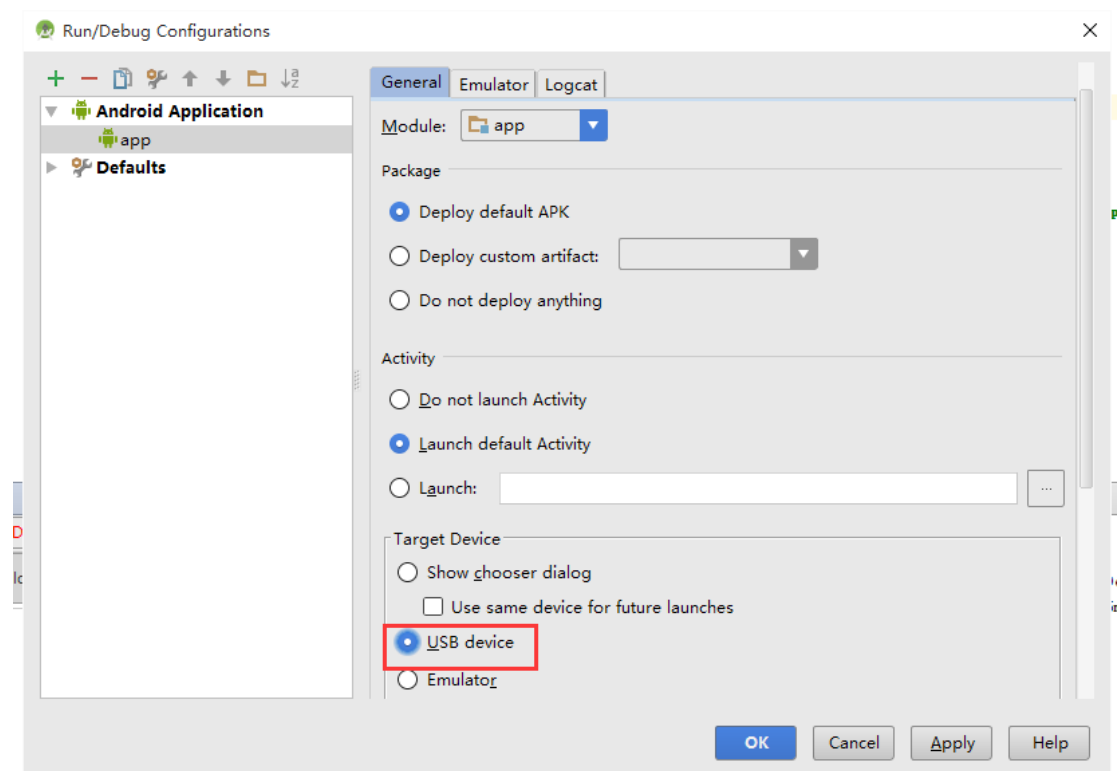


图 3.3.7

至此, 安卓环境搭建完毕。谢谢

## 四、常见问题解答

### 4.1 无法访问外网（android studio 无法下载及 SDK 更新无法联网）

由于大陆地区正常情况下无法访问 google 网站。然而，SDK 更新需要链接 google 服务器。所以，这对于我们来说是一个难题。下面我来介绍一些方法去解决。当然，大家可能有更好的方法，你可以通过我们的微信群或者 QQ 群来分享你的方法。

1、启动 Android SDK Manager，打开主界面，依次选择「Tools」、「Options...」，弹出『Android SDK Manager - Settings』窗口，对话框如图 4.1.1 所示。

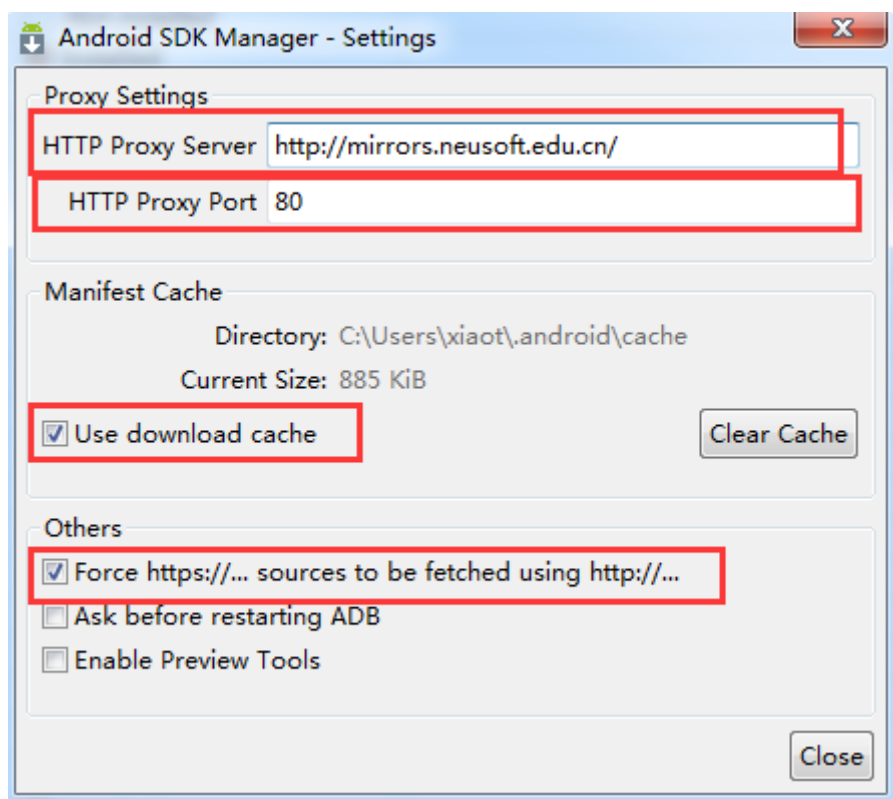


图 4.1.1

方法一：使用国内某些开源软件镜像站，域名 mirrors.neusoft.edu.cn，端口号为 80。

方法二：使用已有的代理，这是可以使用自己设置的代理，域名和端口号视情况而定。

2、购买 VPN 代理实现获得访问外网的权限。

3、可以通过一些论坛去下载使用。

例如：<http://www.androiddevtools.cn/>

这个网站有你需要的安卓开发几乎所有的开发工具。

## 4.2 Intel HAXM 安装不成功

在首次安装 Intel HAXM 的时候可能会出现失败的情况，如图 4.2.2：

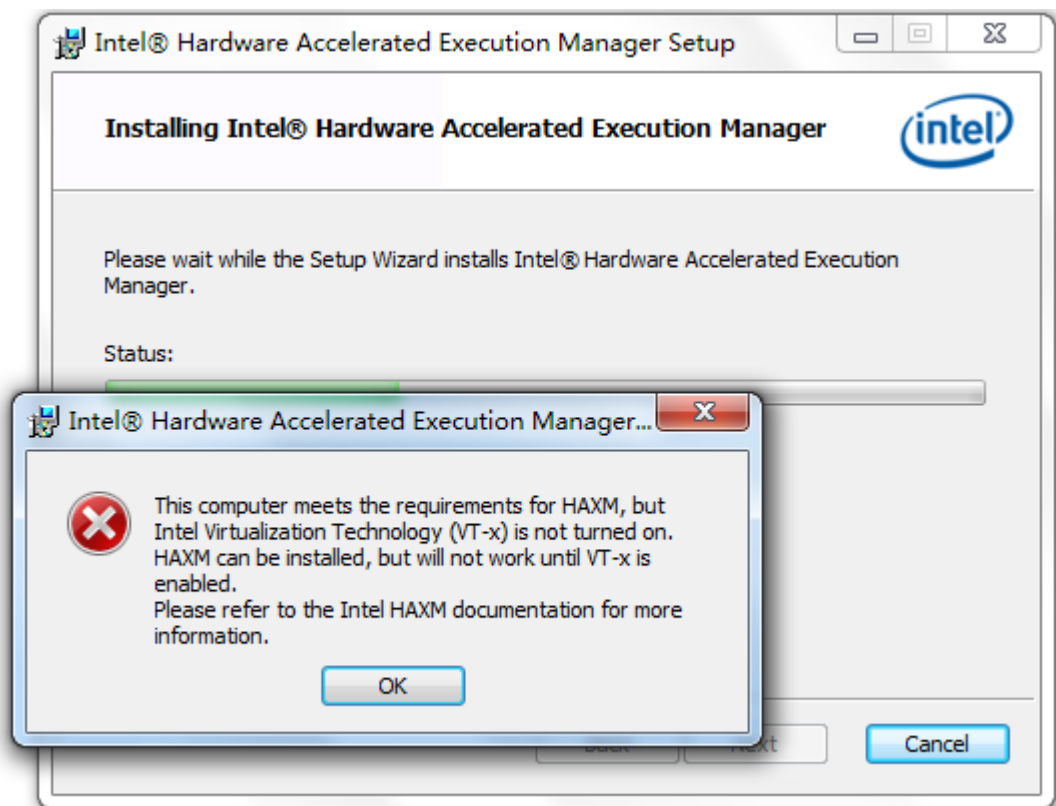


图 4.2.2

这是因为 Intel HAXM (Hardware Accelerated Execution Manager) 使用基于 Intel(R) Virtualization Technology (VT) 的硬件加速，因此需要 CPU 支持 VT，所以是 BIOS 里面的“Virtualization Technology”选项没有打开，去 BIOS 里面打开就可以了。

英特尔官方给出的解决办法如下：

### Intel VT-x not enabled

In some cases, Intel VT-x may be disabled in the system BIOS and must be enabled within the BIOS setup utility. To access the BIOS setup utility, a key must be pressed during the computer's boot sequence. This key is dependent on which BIOS is used but it is typically the F2, Delete, or Esc key. Within the BIOS setup utility, Intel VT may be identified by the terms "VT", "Virtualization Technology", or "VT-d." Make sure to enable all of the Virtualization features.

### 4.3 ADB 启动失败

安装 AndroidStudio 的时候，当启动虚拟机的时候出现了 ADB 没有反应的问题，ADB not responding. You can wait more, or kill "adb.exe" process manually and click 'Restart'

截图如图 4.1.3:

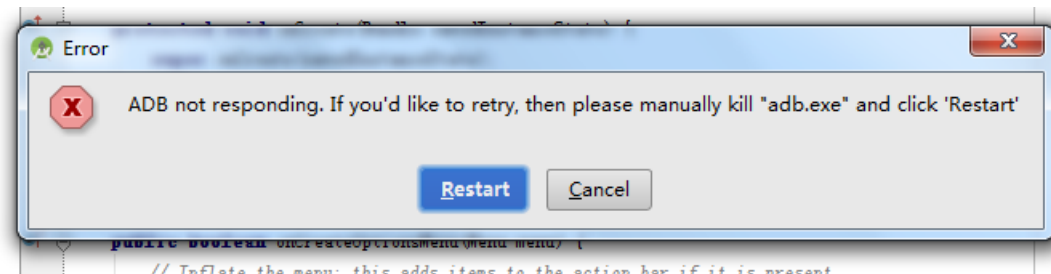


图 4.3.3

出现这个提示的时候，事实上任务管理器不存在一个"adb.exe"的进程，也就没法 kill "adb.exe"，这时，应该查下 adb.exe 使用的端口 5037 被“谁”占用，然后 kill 这个进程，再重新编译就行了。

步骤 1: 查询端口占用的方法：开始——运行——cmd: 打开命令窗口，输入：netstat -aon|findstr “5037” 回车，正常的话会显示出来，如图 4.1.4:

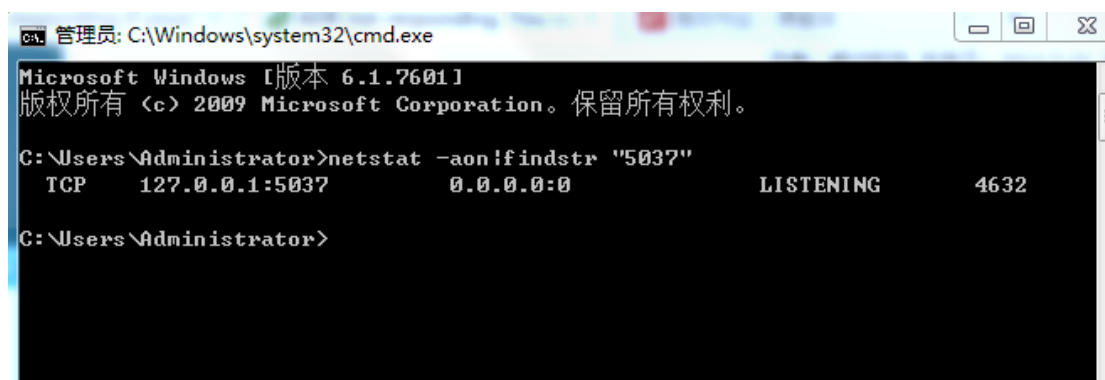


图 4.3.4

表明 PID = 4632 的进程占用了这个端口号

打开任务管理器（如图 4.3.5），显示出 PID 列，找到这个 4632 的进程，kill 掉，再在 Android Studio 重新编译即可。

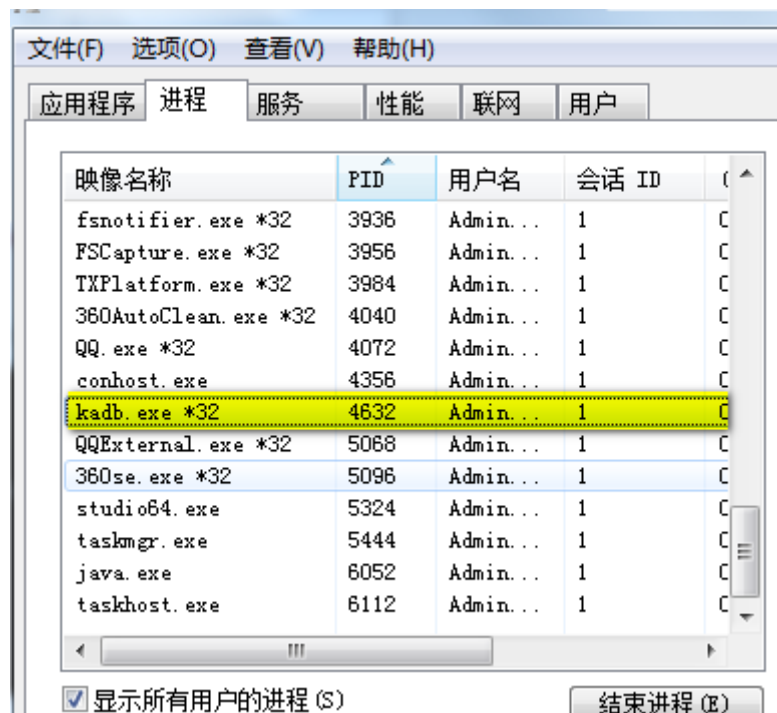


图 4.3.5



#### 4.4 Failure [INSTALL\_FAILED\_OLDER\_SDK]

解决方法：在工程的 `build.gradle` 文件中设置安卓 `sdk` 的最小版本 `minSdkVersion`，不能高于目前模拟器所支持的 `sdk` 版本，如图 4.4.1 。

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 19
    buildToolsVersion '19.1.0'

    defaultConfig {
        applicationId "pku.ss.liudong.myweather"
        minSdkVersion 12
        targetSdkVersion 19
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
}
```

图 4.4.1

#### 4.5 android studio 安装卡在 SDK 下载界面不动 (Setup Wizard - Downloading Components)

安装后第 1 次启动的时候会首先显示 “Fetching Android SDK component information” 如图 4.5.1

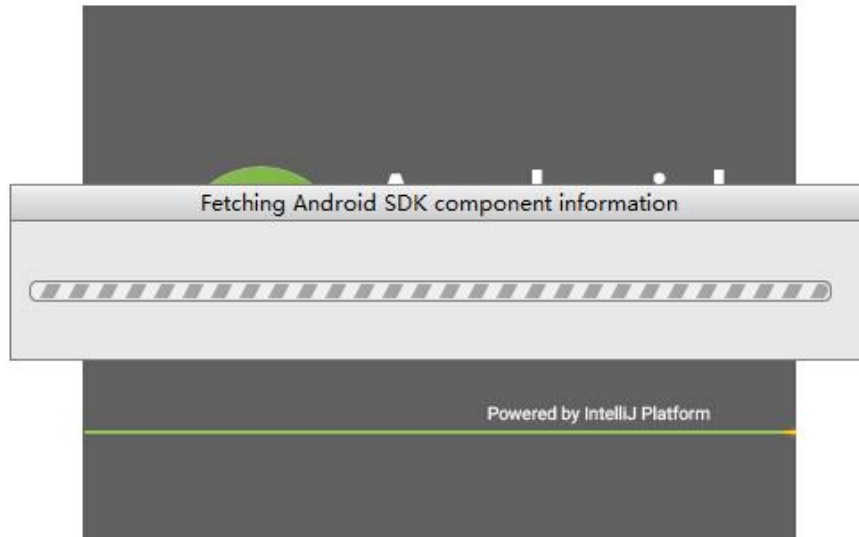


图 4.5.1

等一会儿在 Setup Wizard - Downloading Components 界面下面开始下载 Android SDK，如图 4.5.2

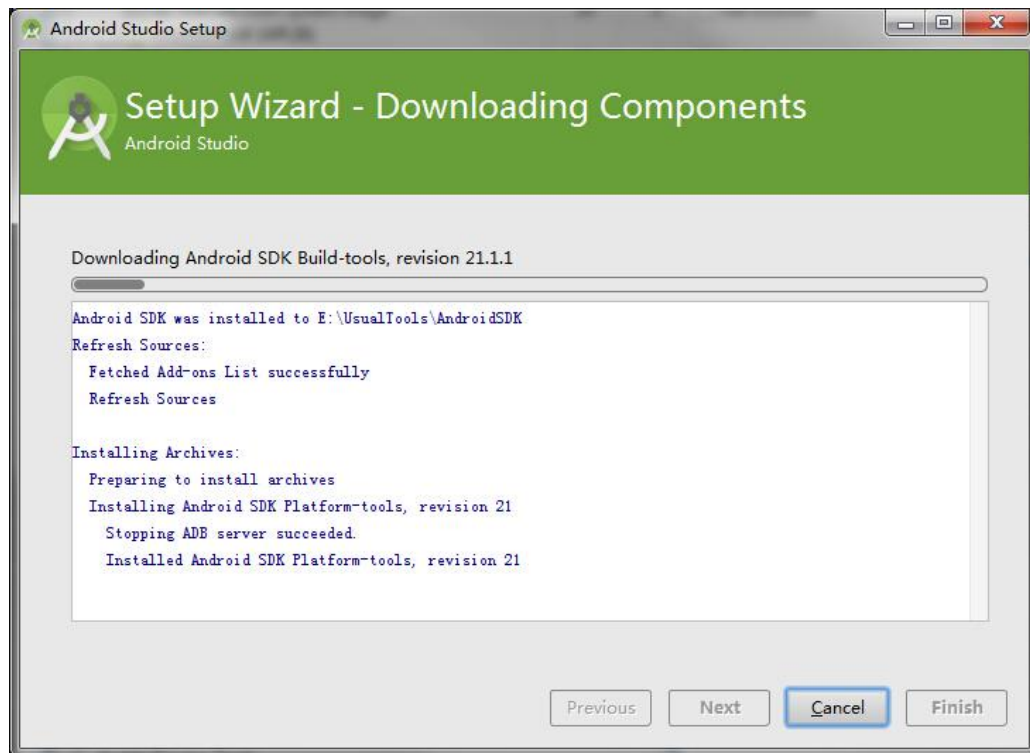


图 4.5.2

还是由于被墙的缘故，下载不了。

解决方法：

- 1、关闭安装向导。
- 2、打开 Android Studio 安装目录的 bin 目录下面的 idea properties 文件，添加一条禁用开始运行向导的配置项：`disable.android.first.run=true`。
- 3、启动程序打开项目向导界面，此时点击 Start a new Android Studio project 没有反应，并且在 Configure 下面的 SDK Manager 也是灰色的（如图 5.4.3），这是因为没有安装 Android SDK 的原因。

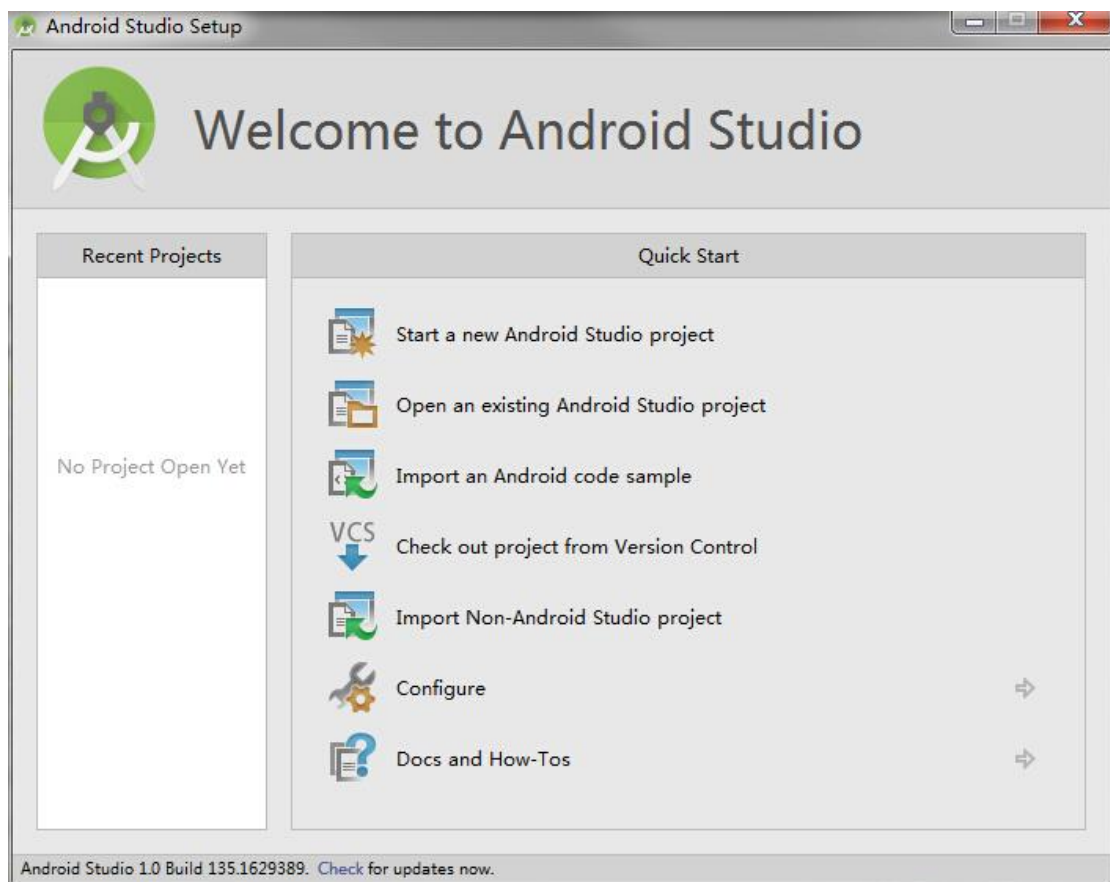


图 5.4.3

- 4、这个时候需要打开 Configure-Settings，在查找框里面输入 proxy，找到下面的 HTTP Proxy，设置代理服务器，然后退出将上面在 idea.properties 配置文件中添加的那条配置项注释掉重新打开 Android Studio 等刚开始的向导把 Android SDK 下载安装完成就可以了。

#### 4.6 运行模拟器时错误

当我们启动一个选择 x86 的虚拟机的会后，有可能会出现这个界面，如图 4.6.1

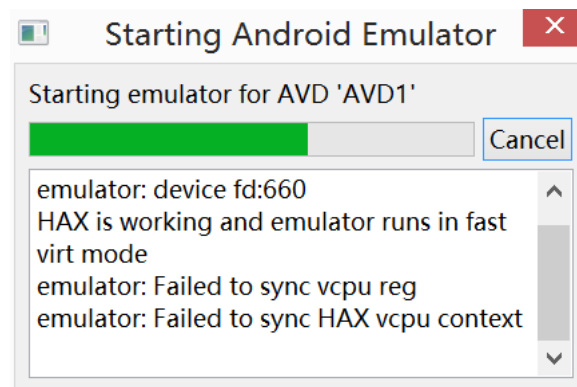


图 4.6.1

这是由于安装 Intel HAXM 版本过低。

解决方法一：去引文官网去下载新版本。

解决方法二：进入“<Android

SDK>\extras\intel\Hardware\_Accelerated\_Execution\_Manager\”这个文件夹，运行 intelhaxm-android.exe 来更新。

4.7 高版本程序运行在低版本模拟器

创建“Hello world”程序出错！如图 4.7.1、图 4.7.2

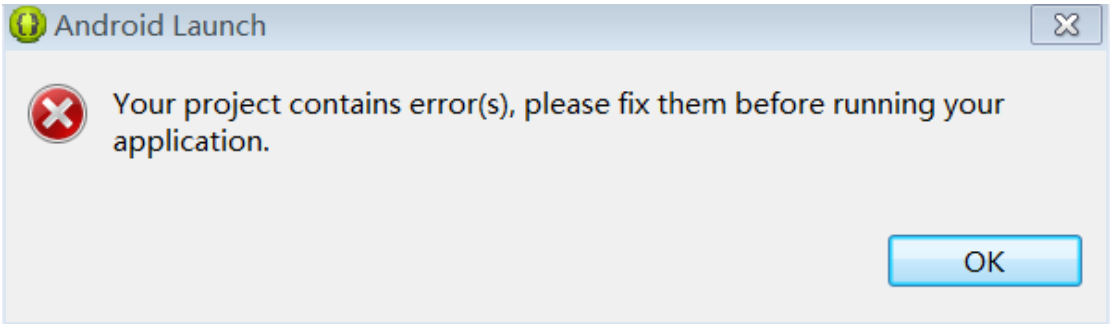


图 4.7.1

3 errors, 0 warnings, 0 others			
Description	Resource	Path	L
⌵ ✖ Errors (3 items)			
💡 ✖ The container 'Android Dependencies' i	HelloWorld		B
✖ The project cannot be built until build p	HelloWorld		U
✖ The project cannot be built until build p	appcompat...		U

图 4.7.2

原因分析以及解决方法:jar 包的原因，之前创建的模拟器是基于 Android 4.4.2 的，但是创建的 HelloWorld 默认是基于 Android 5.0 重新添加 jar 包，在工程区域右键->Build Path ->Configure Build Path->Add External JARs...选定 jar 包后点击“OK”即可。

4.8 点击开始新 Android 项目无反应

在欢迎界面的面板，点击 Configure Project Defaults Project Structure，然后会发现 Android SDK location 的路径是空的。点击右边的浏览按钮，然后找到相应的 SDK 路径，并保存。回到欢迎界面，单击 Start a new Android Studio project，会发现可以弹出创建工程的界面了，问题得到解决。

4.9 虚拟机启动后卡在开机界面

重启电脑后进行一下操作：

- 1) 将 system-images 里的 x86 从 android-19→default→x86 目录，移动到了 android-19→x86。

2) 修改了虚拟机的配置信息，RAM 设置从 1536 改为 512（设置大的想法是想让虚拟机快一些，多点内存资源），SD card 从 0 修改为 Size 100，如图 4.9.1 所示。

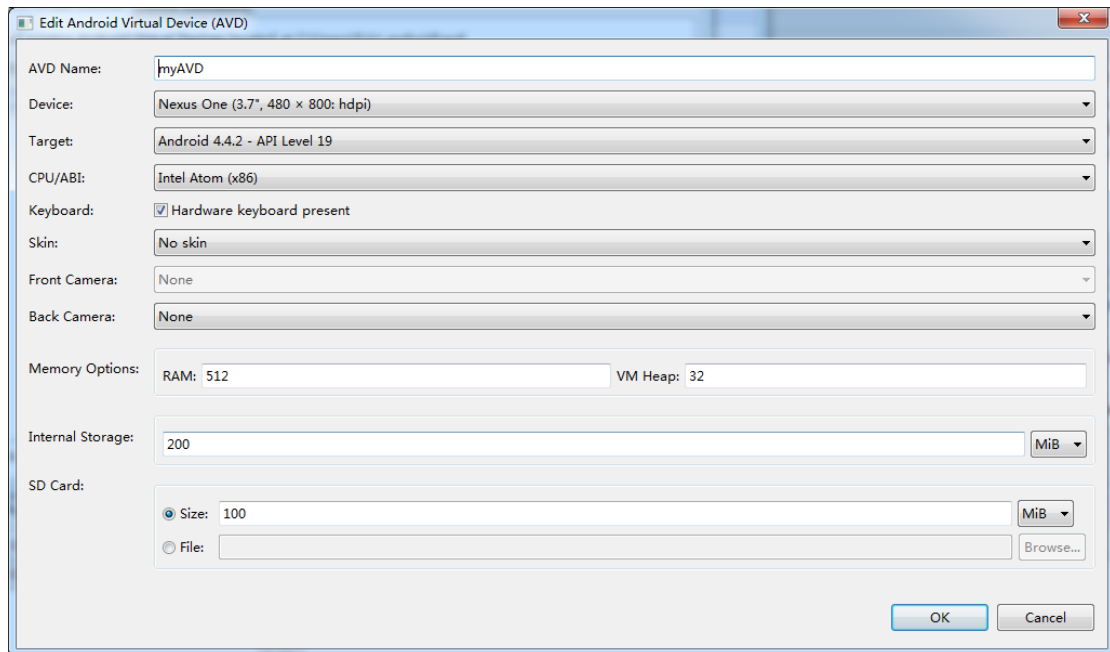


图 4.9.1

#### 4.10 SDK、AVD 闪退

压缩包解压之后，SDK Manager 和 AVD Manager 文件点击会发生闪退无法打开  
解决方案：

- 1、右键以管理员身份运行：失败，依旧闪退；
- 2、打开压缩包解压后 tools 文件夹里的 android.bat 文件：打开后会自动打开 SDK Manager；

#### 4.11 安卓虚拟机找不到 system image

更新 SDK 相应镜像。