**安装流程**

1． 安装JDK.

2． 安装Eclipse(eclipse-cpp-indigo-SR1-incubation-win32.zip，解压即可使用。注：该版本集成CDT)

3． 搭建Android开发环境（参考文档：<http://developer.android.com/sdk/installing.html>）

  3.1  安装Android SDK（installer\_r16-windows）

  3.2  安装Eclipse的ADT插件.

           3.2.1  启动Eclipse, 选择菜单项的Help->Install New Software

           3.2.2  点击Add, 输入名称，然后在URL中输入：

<http://dl-ssl.google.com/android/eclipse/>，OK！

           3.2.3  选择需要更新的工具，Finish, 安装完成后重启Eclipse。

3.3  配置ADT插件

       3.3.1  启动Eclipse，选择菜单项的Window -> Preferences

       3.3.2  选择Android，在SDK Location中选择SDK的安装目录，OK！

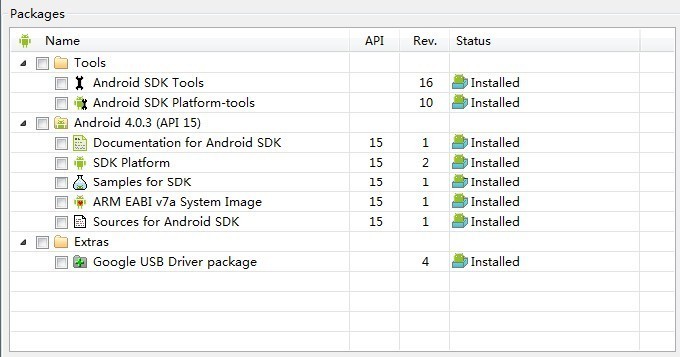
完成后重新启动Eclipse。

       3.4  添加平台和其他组件

              3.4.1  启动Eclipse，选择菜单项的Window -> Android SDK Manager

              3.4.2  根据实际需要选择相应组件，然后点击install。

                       默认可安装以下组件，如图一所示：



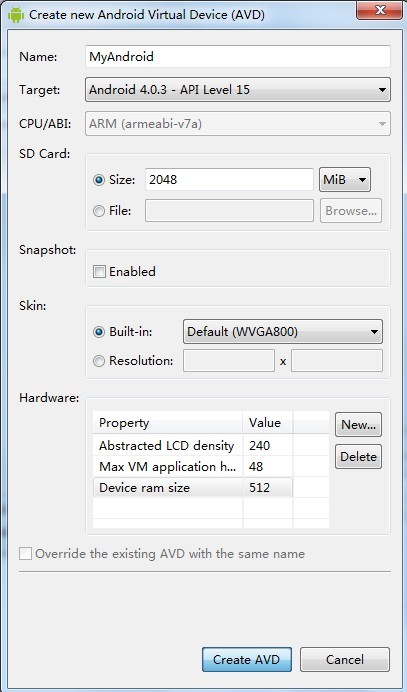
                                                                （图一）

              3.4.3  创建一个Android 虚拟设备。选择菜单项的Window ->AVD Manager.

              New ,新建一个虚拟设备，具体参数请自行设置。如图二所示：

4． 安装NDK（R7, android-ndk-r7b-windows.zip, 解压即可使用）

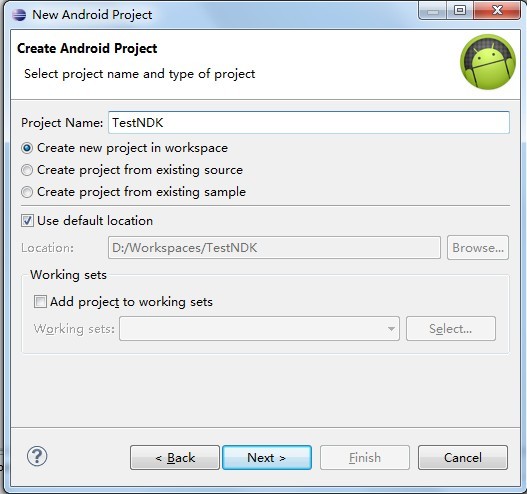
注：R7之前的版本，必须要安装Cygwin才能使用NDK。从R7开始，Windows版本的NDK提供了一个ndk-build.cmd脚本，可以直接利用这个脚本编译。想要做到这一点，只要为Eclipse Android工程添加一个Builder，就能实现Eclipse的自动编译NDK。



　　　　　　　　　　　　（图二）

5．  NDK实例的实现

5.1   启动Eclipse，新建Android工程（此处名为TestNDK），如图三所示：



                                              （图三）

5.2  在TestNDK工程的根目录下新建文件夹jni，之后找到NDK的安装目录（\samples\hello-jni\jni）下的 Android.mk和hello-jni.c 这两个文件，将其拷贝到TestNDK->jni文件夹下面。

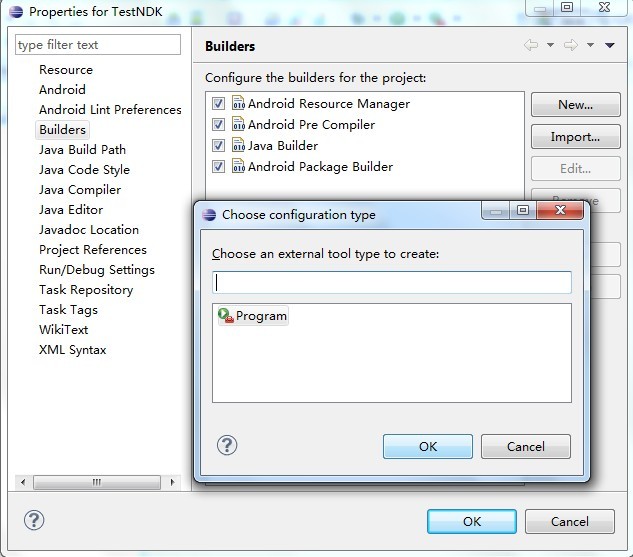
5.3   找到NDK的安装目录（\samples\hello-jni\src\com\example\hellojni）下的HelloJni.java文件，将其拷贝到TestNDK工程src文件夹对应的包下。

5.4  修改项目文件AndroidManifest.xml，将activity标签下的 android:name =“.TestNDKActivity”修改为 android:name =“.HelloJni”。（注：此处的

TestNDKACtivity为项目自动生成的，项目不同，此处也会不同）

5.5   新建并配置一个Builder

5.5.1        点击Project->Properties->Builders->New，新建立一个Builder。在弹出的对话框上面点击Program,OK！如图四所示：



                                                     （图四）

5.5.2       在弹出的对话框【Edit Configuration】中，配置选项卡【Main】：

Location中需要填入nkd-build.cmd的路径（NDK安装目录下）。

Working Diretcoty中需要填入TestNDK的工程根目录。

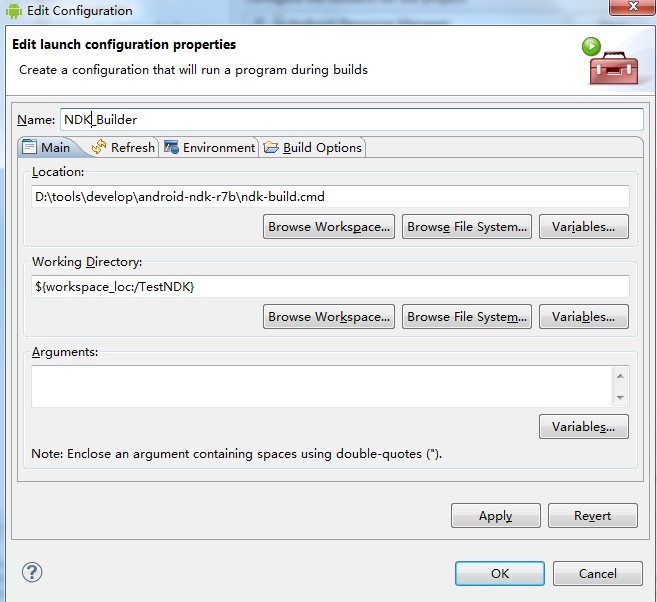
如图五所示：

5.5.3      在弹出的对话框【Edit Configuration】中，配置选项卡【Refresh】：

勾选“Refresh resources upon completion”，

勾选“The entire workspace”，

勾选“Recuresively include sub-folders”。



                                                         （图五）

5.5.4       在弹出的对话框【Edit Configuration】中，配置选项卡【Build Options】：

勾选“After a “Clean””，

勾选“During manual builds”，

勾选“During auto builds”,

勾选“Specify working set of relevant resources”。

点击“Specify Resources…”勾选TestNDK工程的“jni“目录 ，Finish！

保存设置，OK！

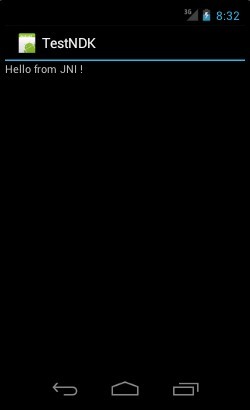
5.5.5       右键点击工程TestNDK，Build Project。当在console里面看到如图六所示的信息，则表示一切正常,此时就已经是自动编译代码了。



　　　　　　　　　　　　　　　　　　（图六）

5.5.6      右键点击工程TestNDK，Run As -> Android Application, 运行程序。当出

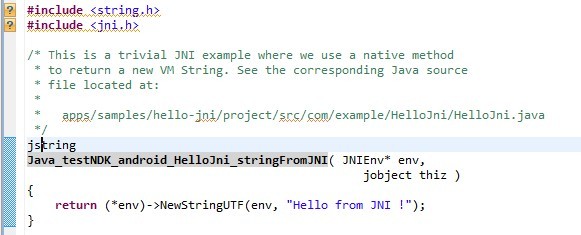
现图七所示信息时，NDK实例运行成功！



                   （图七）

6．   可能出现的问题

      如图八所示：



                               （图八）

在 c 文件中，函数名这样定义：

Java\_testNDK\_android\_HelloJni\_stringFromJNI ，是因为这个是 JNI 的标准，定义需要按照如下格式：

       Java\_packagename\_classname\_methodname ,

       例如： Java\_testNDK\_android\_HelloJni\_stringFromJNI