各平台编译发布指南

目录

[一、 Android版编译发布 1](#_Toc464741323)

[二、 Wince版编译发布 4](#_Toc464741324)

[三、 Windows版编译发布 9](#_Toc464741325)

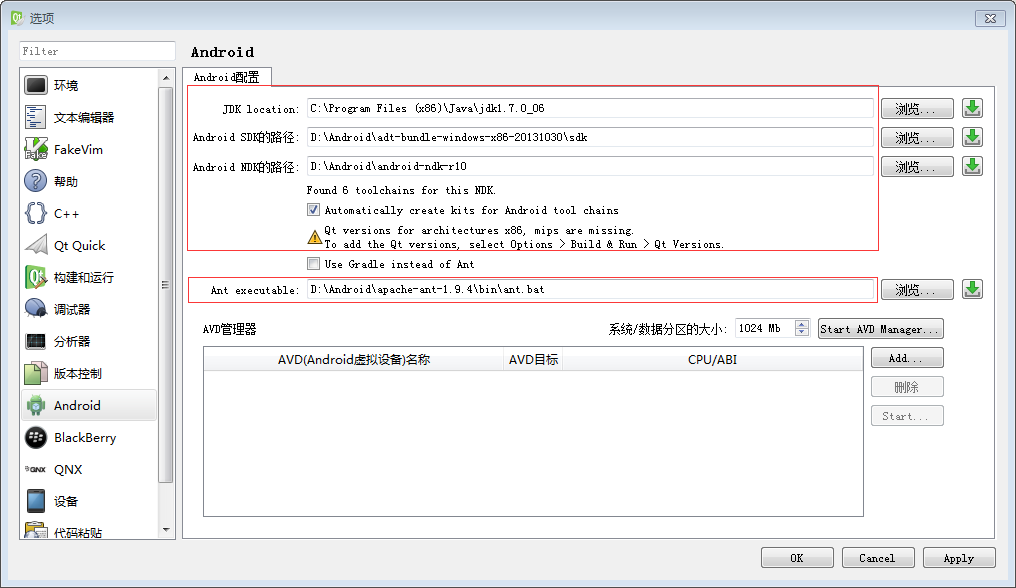
[四、 Linux版编译发布 12](#_Toc464741326)

1. Android版编译发布

这里介绍的是在windows上搭建的Android编译环境及发布。

1. 下载并安装QT for Android安装包qt-opensource-windows-x86-android-5.4.1
2. 安装完成之后，将形如C:\Qt\Qt5.4.1\5.4\android\_armv7\bin的路径加入到系统Path环境变量。
3. 配置编译器
4. 配置Android环境

选择工具🡪选项🡪Android，配置jdk,sdk,ndk,ant，配置内容形如图中红框所示



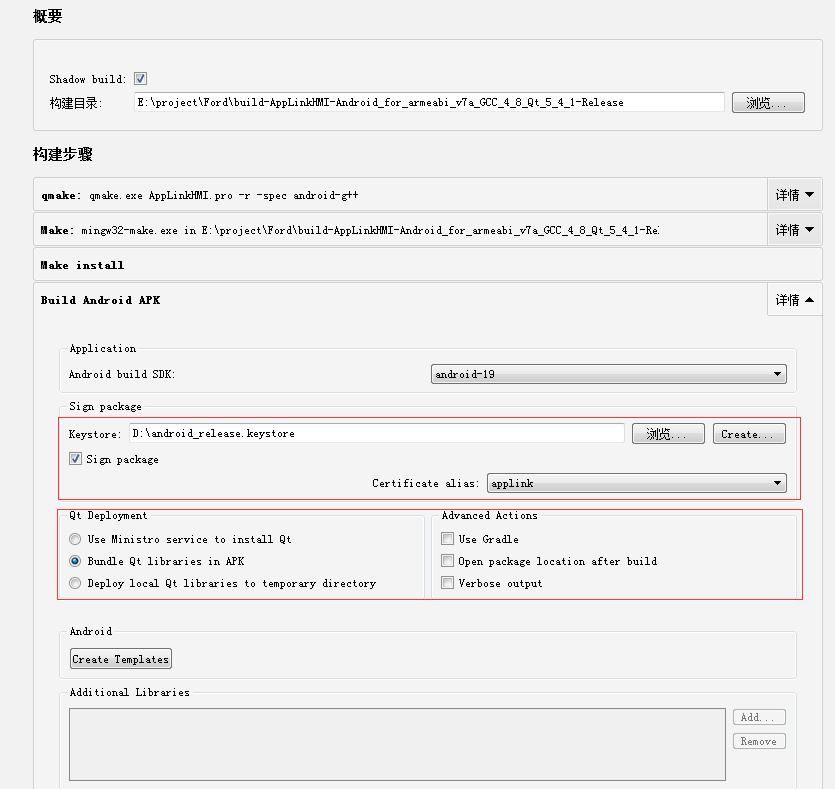
1. 配置QT编译器

选择工具🡪选项🡪构建运行，查看编译器配置情况（按照上面步骤执行成功后，QT会自动配置好Android编译器），配置内容形如图中红框所示



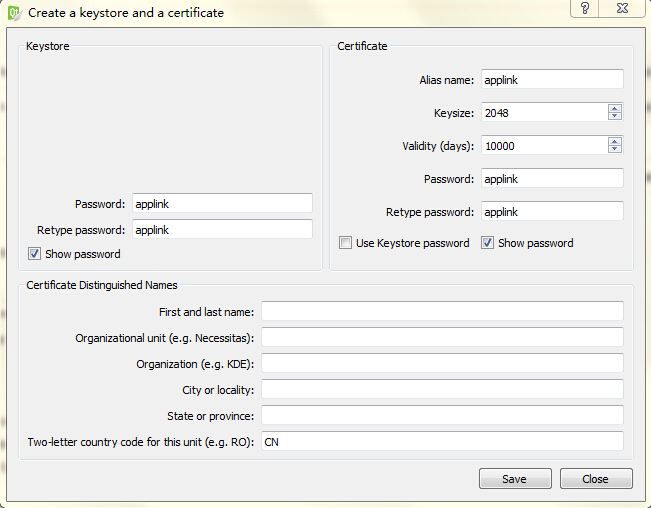
1. 配置项目选项

选择QT左边工具栏选项项目，首先添加编译套件（就是前面自动配置好的Android编译套件），之后对项目选项进行设置，配置内容形如图中红框所示



图中Sign package配置步骤如下，

如上图点击Sign Package 栏右侧Create，出现下图所示对话框，按下图设置完成后保存。其中Keysize 和Validity 保持默认值即可，Alias name和password 需要自己记住。完成后点击保存，即可使用。



1. 编译

用qt creator打开AppLink\_HMI.pro，点击Build->Build Project “AppLink\_HMI.pro”,等待编译完成。

1. 发布打包

编译成功后，在形如E:\project\Ford\build-AppLinkHMI-Android\_for\_armeabi\_v7a\_GCC\_4\_8\_Qt\_5\_4\_1-Release\android-build\bin该目录下会生成QtApp-release-signed.apk，将apk安装到Android车机上就可以运行。

注：Android平台下，SDL为动态库方式加载，并且Android打包工具会自动将所有库文件打包到apk中

1. Wince版编译发布

这里介绍的是在windows上搭建的Wince6编译环境及发布。

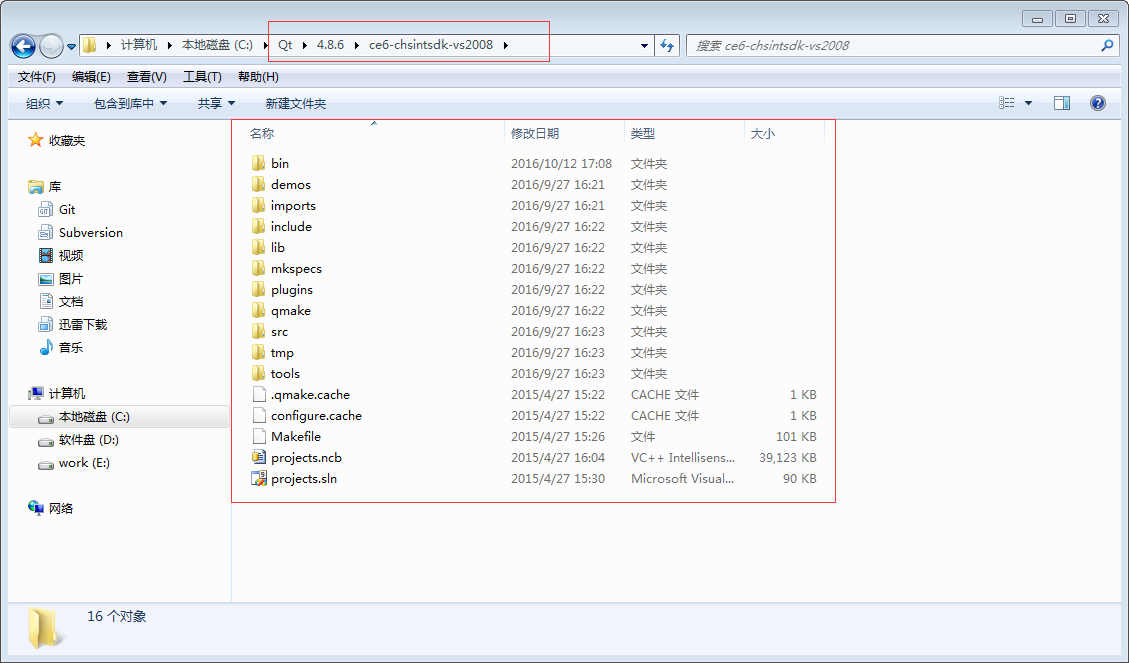
1. 按照如下顺序安装工具包

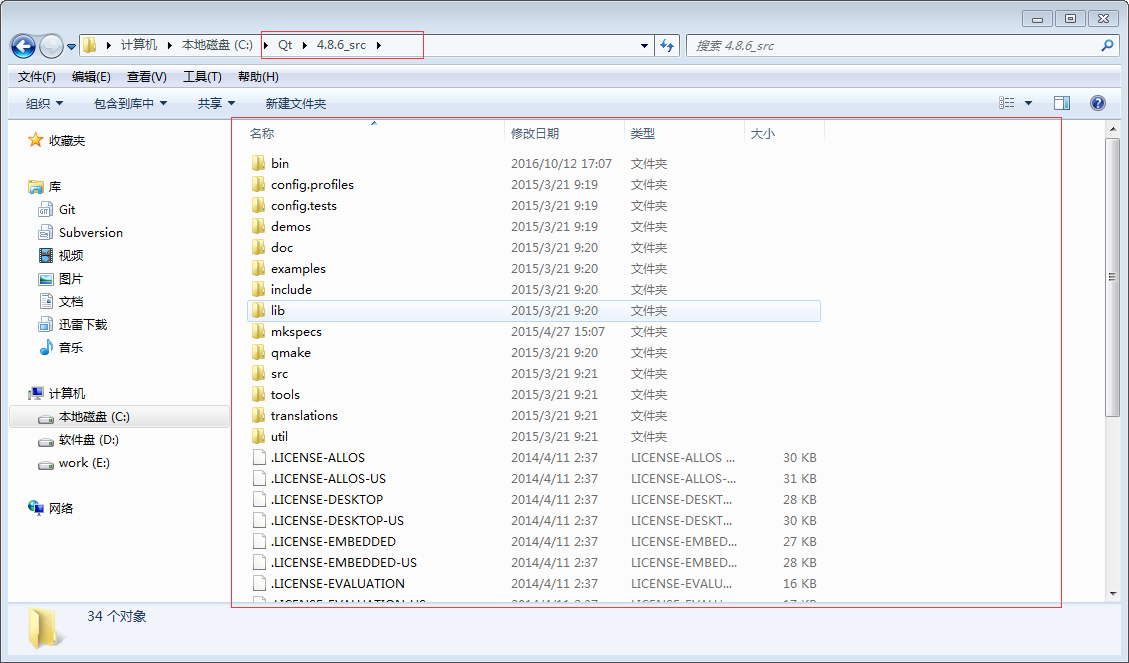
Windows Mobile 6 Standard SDK Refresh.msi

WINCE6.0\_CHS\_SDK.msi

qt-opensource-windows-x86-msvc2013\_opengl-5.3.1.exe（由于还要之后还要配置Windows平台的编译环境，所以安装该版本的QT）

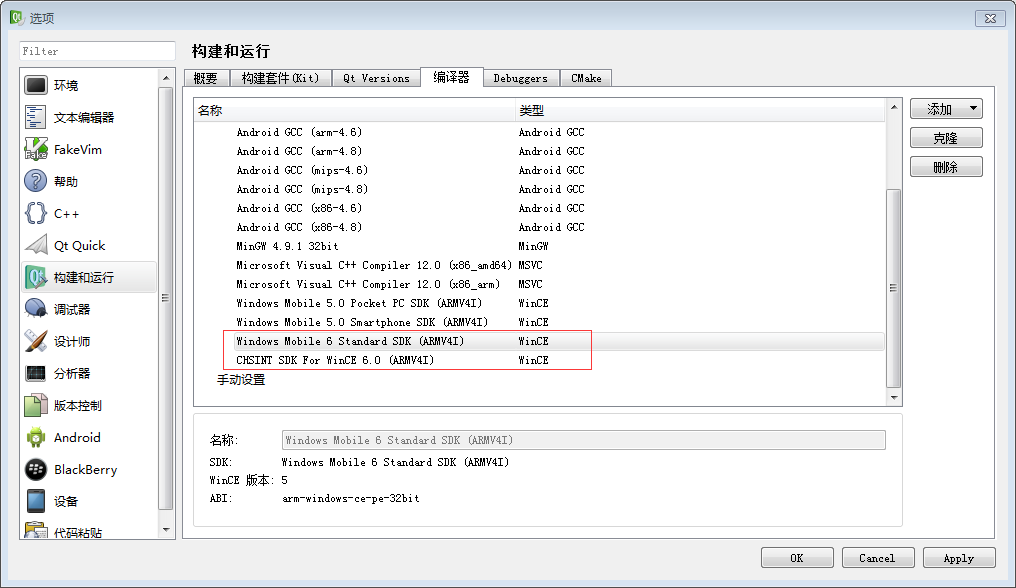
1. 安装完成之后，将如下压缩包解压到该目录下面，内容形如图中红框所示





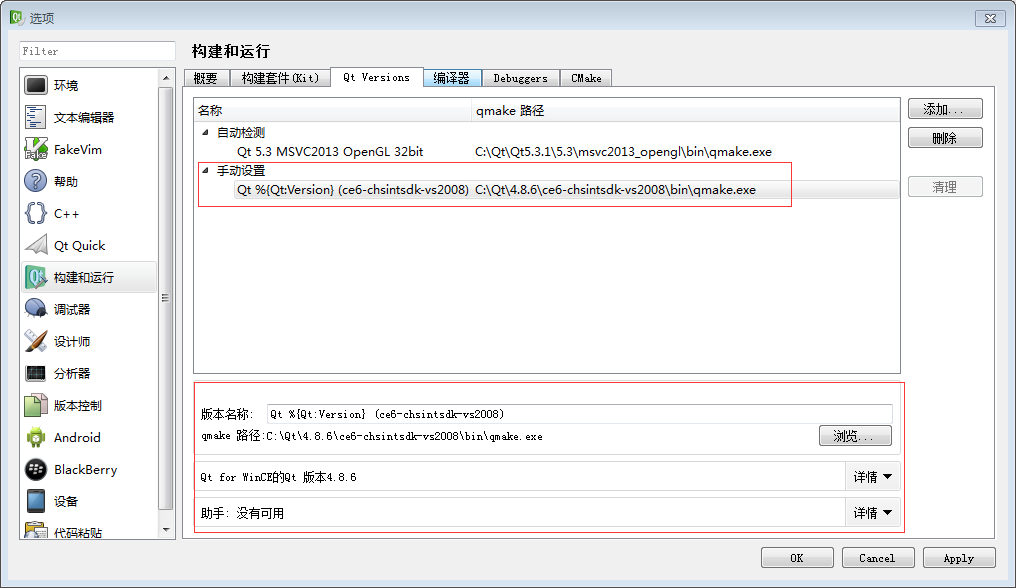
1. 配置编译器
2. 配置编译器环境

选择工具🡪选项🡪构建运行，进入编译器页面，配置内容形如图中红框所示



1. 配置QT版本

选择工具🡪选项🡪构建运行，进入QT Versions页面，配置内容形如图中红框所示



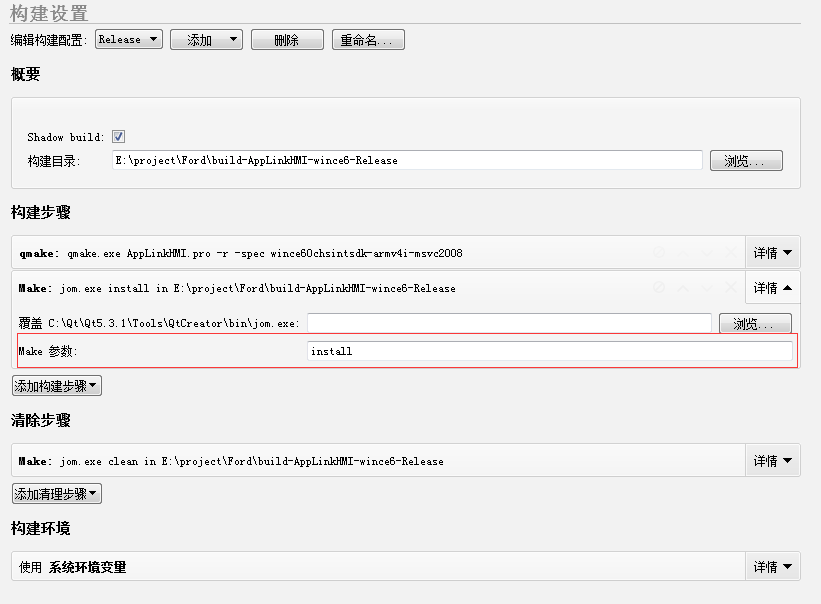
1. 配置构建套件

选择工具🡪选项🡪构建运行，进入QT Versions页面，配置内容形如图中红框所示



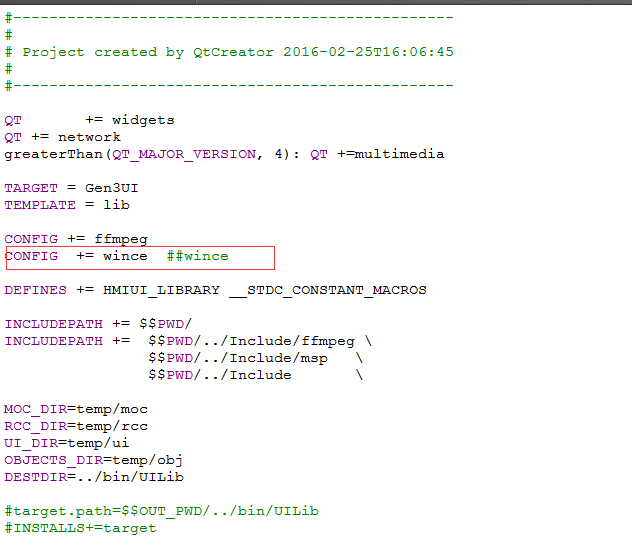
1. 配置项目选项

选择QT左边工具栏选项项目，首先添加编译套件（就是前面自动配置好的Wince编译套件），之后对项目选项进行设置，配置内容形如图中红框所示



1. 工程文件修改

打开文件Gen3UI.pro、hmi\_sdk.pro修改配置选项，设置标记为Wince平台，配置内容形如图中红框所示

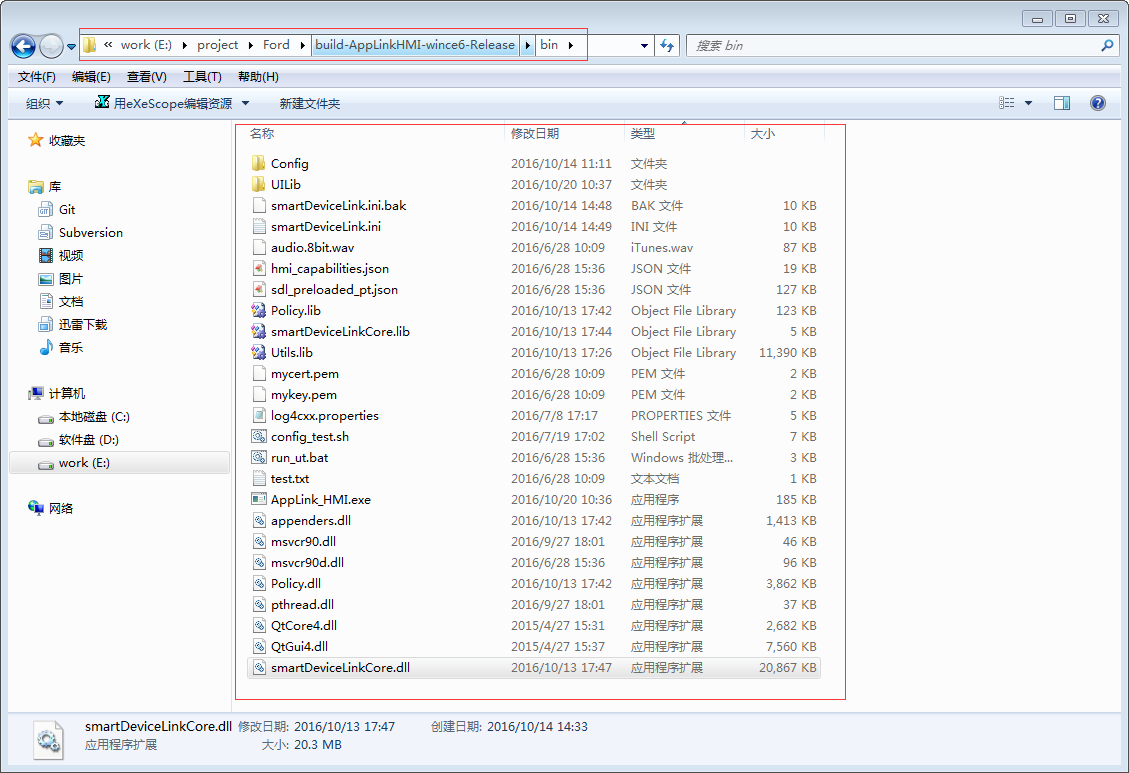


1. 编译

用qt creator打开AppLink\_HMI.pro，点击Build->Build Project “AppLink\_HMI.pro”,等待编译完成。

1. 发布打包

编译成功后，手动将SDL的发布包释放到该目录下面，形如E:\project\Ford\build-AppLinkHMI-wince6-Release\bin，该目录内容如图所示，然后将该目录整个打包发布（图中QT相关库文件，可以从QT安装目录项目拷贝，形如C:\Qt\4.8.6\ce6-chsintsdk-vs2008\lib）



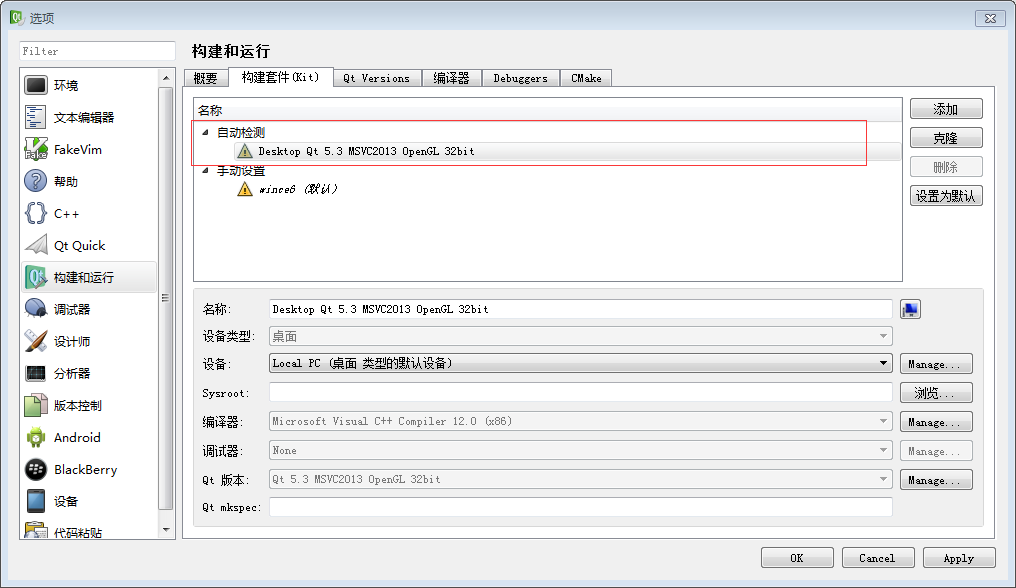
注：Wince平台下，SDL为动态库方式加载到HMI中，所以需要SDL发布库的支持

1. Windows版编译发布

这里介绍的是在windows上搭建的Windows编译环境及发布。

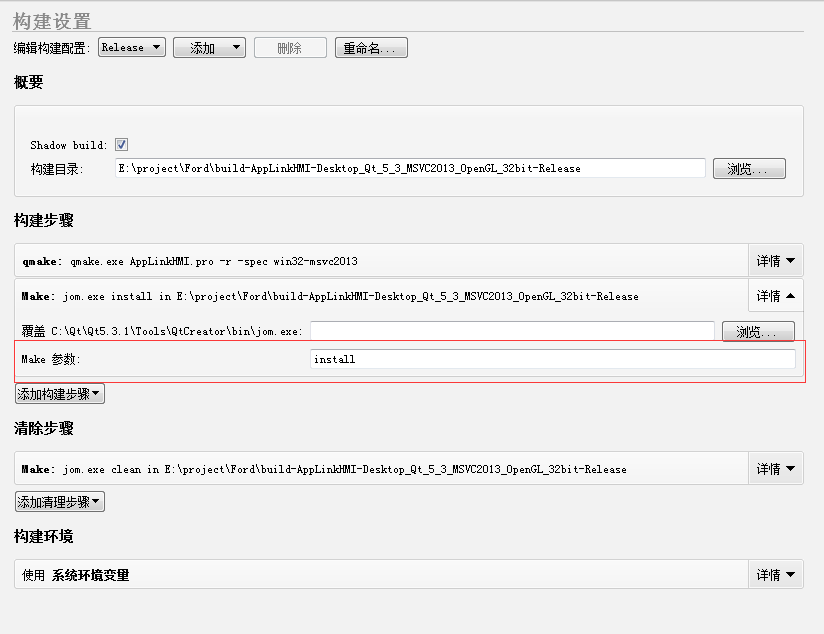
1. 下载并安装QT for Windows安装包qt-opensource-windows-x86-msvc2013\_opengl-5.3.1.exe，如前面配置过Wince环境，则此处不用安装
2. 安装完成之后，将形如C:\Qt\Qt5.3.1\5.3\msvc2013\_opengl\bin的路径加入到系统Path环境变量。
3. 配置编译器
4. 配置QT编译器

选择工具🡪选项🡪构建运行，查看编译器配置情况（按照上面步骤安装成功后，QT会自动配置好Windows编译器），配置内容形如图中红框所示



1. 配置项目选项

选择QT左边工具栏选项项目，首先添加编译套件（就是前面自动配置好的Windows编译套件），之后对项目选项进行设置，配置内容形如图中红框所示

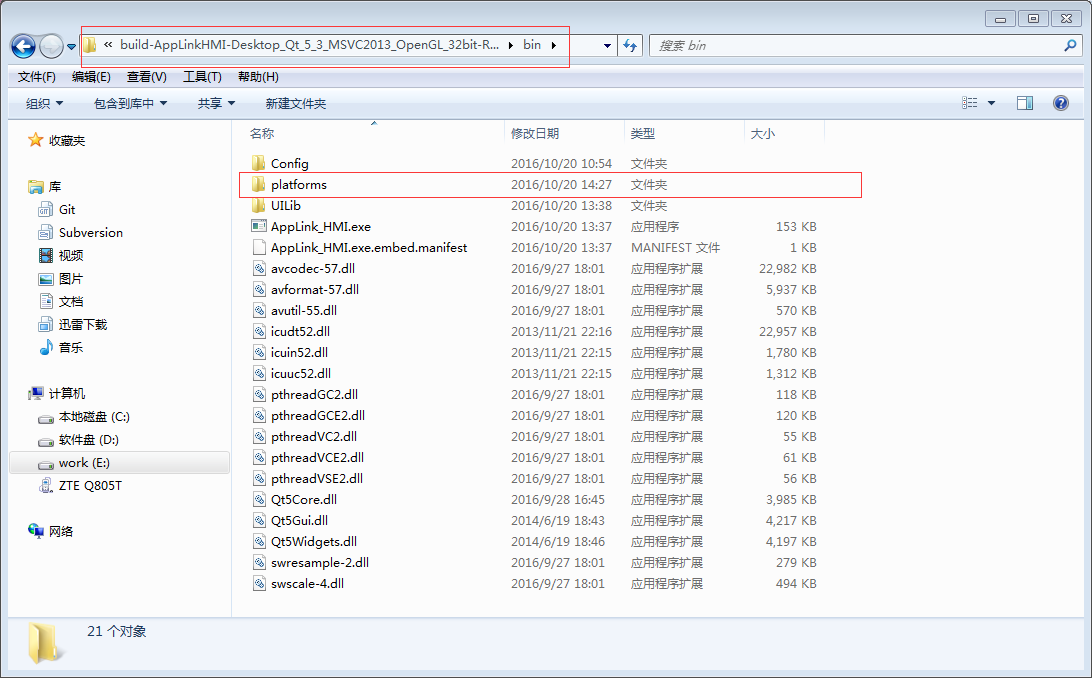


1. 编译

用qt creator打开AppLink\_HMI.pro，点击Build->Build Project “AppLink\_HMI.pro”,等待编译完成。

1. 发布打包

编译成功后，在形如E:\project\Ford\build-AppLinkHMI-Desktop\_Qt\_5\_3\_MSVC2013\_OpenGL\_32bit-Release\bin目录下会生成发布包文件，然后将该目录整个打包发布（图中QT相关库文件，可以从QT安装目录项目拷贝，形如C:\Qt\Qt5.3.1\5.3\msvc2013\_opengl\bin以及platforms相关库文件，形如C:\Qt\Qt5.3.1\5.3\msvc2013\_opengl\plugins\platforms）



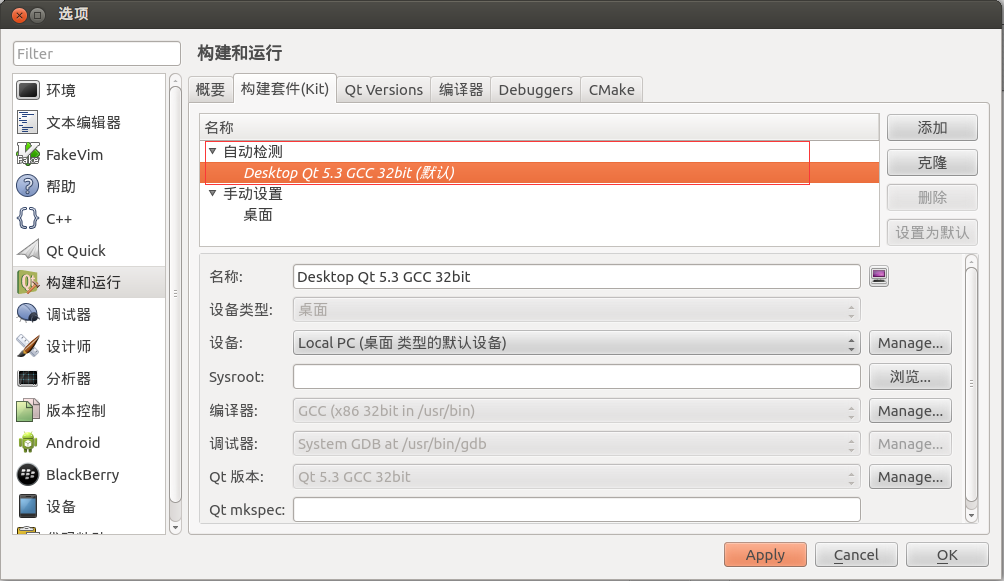
注：Windows平台下，SDL为独立可执行程序

1. Linux版编译发布

这里介绍的是在Ubuntu虚拟机环境上搭建的Linux编译环境及发布。

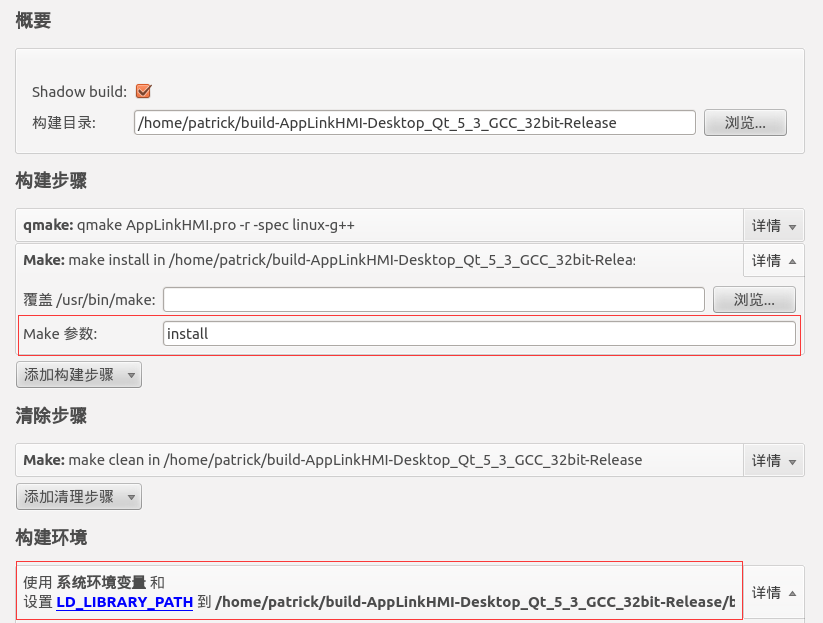
1. 下载并安装QT for Linux安装包qt-opensource-linux-x86-5.3.1.run
2. 安装完成之后，将形如/home/patrick/build-AppLinkHMI-Desktop\_Qt\_5\_3\_GCC\_32bit-Release/bin的路径加入到系统LD\_LIBRARY\_PATH环境变量。
3. 配置编译器
4. 配置QT编译器

选择工具🡪选项🡪构建运行，查看编译器配置情况（按照上面步骤安装成功后，QT会自动配置好Linux编译器），配置内容形如图中红框所示



1. 配置项目选项

选择QT左边工具栏选项项目，首先添加编译套件（就是前面自动配置好的Linux编译套件），之后对项目选项进行设置，配置内容形如图中红框所示

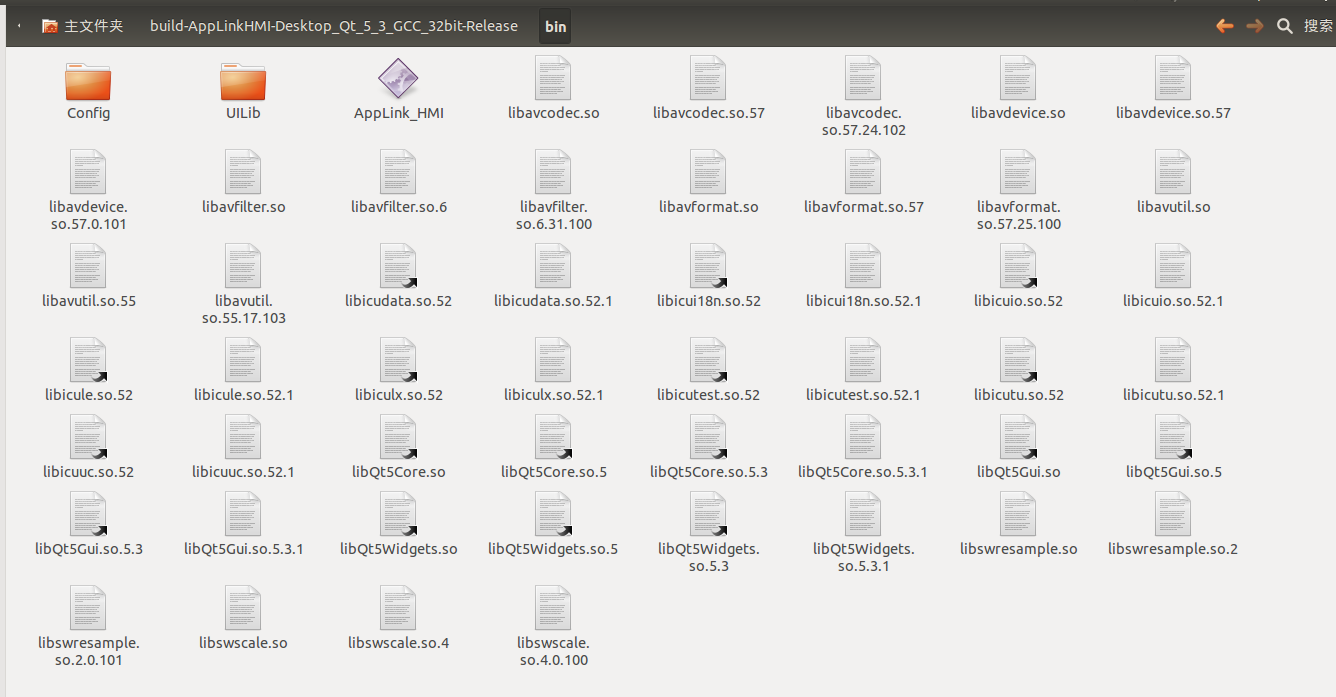


1. 编译

用qt creator打开AppLink\_HMI.pro，点击Build->Build Project “AppLink\_HMI.pro”,等待编译完成。

1. 发布打包

编译成功后，在形如/home/patrick/build-AppLinkHMI-Desktop\_Qt\_5\_3\_GCC\_32bit-Release/bin目录下会生成发布包文件，然后将该目录整个打包发布（图中QT相关库文件，可以从QT安装目录项目拷贝，形如/home/patrick/Qt5.3.1/5.3/gcc/lib）



注：Linux平台下，SDL为独立可执行程序