

fengMisaka

捧月华如胧
拾芬芳其中

Qt之程序发布以及打包成exe安装包

目录

- [一、简述](#)
- [二、设置应用程序图标](#)
- [三、发布程序](#)
- [四、打包程序](#)

[回到顶部](#)

一、简述

Qt 项目开发完成之后，需要打包发布程序，而因为用户电脑上没有 Qt 配置环境，所以需要将 release 生成的 exe 文件和所依赖的 dll 文件复制到一个文件夹中，然后再用 Inno Setup 打包工具打包成一个 exe 安装包，就可以发布了。

[回到顶部](#)

二、设置应用程序图标

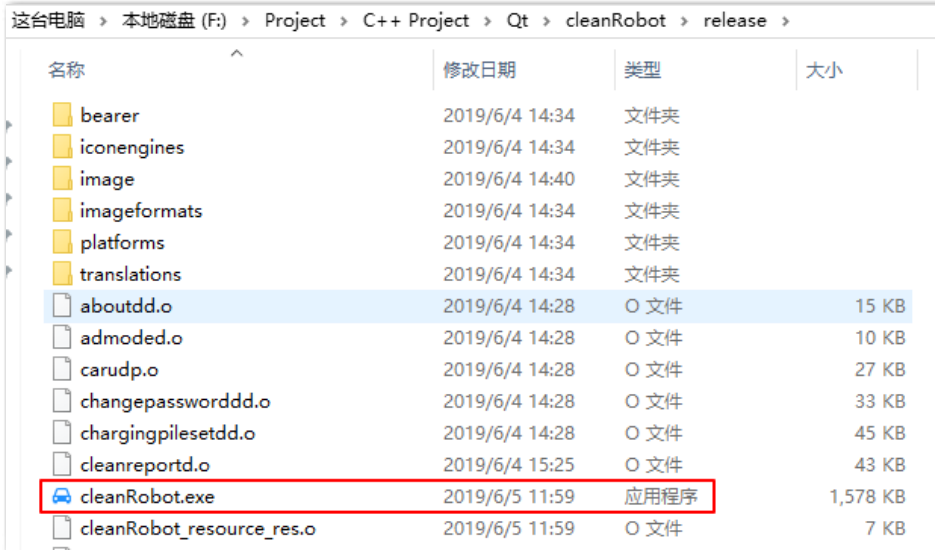
我们编译好生成的 exe 文件的图标是 Qt 自带的一个白色的框框，不太好看，怎么能够自己设置程序的图标呢。先创建一个 ico 格式的文件，网上有很多在线就可以把 png 或 jpg 的图片转成 ico 格式的图标文件，然后把 ico 文件放到源代码目录下，就是 .pro 文件的同一文件夹下。

这台电脑 > 本地磁盘 (F:) > Project > C++ Project > Qt > cleanRobot			
名称	修改日期	类型	大小
mainw.cpp	2019/6/5 11:59	C++ Source file	6 KB
Makefile	2019/6/5 11:57	文件	34 KB
Makefile.Debug	2019/6/5 11:57	DEBUG 文件	627 KB
Makefile.Release	2019/6/5 11:57	RELEASE 文件	629 KB
object_script.cleanRobot.Debug	2019/6/5 11:57	DEBUG 文件	2 KB
object_script.cleanRobot.Release	2019/6/5 11:57	RELEASE 文件	2 KB
cleanRobot.pro	2019/6/5 11:57	Qt Project file	4 KB
cleanRobot_resource.rc	2019/6/5 11:53	RC 文件	1 KB
car.ico	2019/6/5 11:52	图标	6 KB
cleanRobot.pro.user	2019/6/5 11:05	USER 文件	24 KB

在.pro项目文件中添加：

```
RC_ICONS = car.ico // 这里的car.ico是我的ico文件的名字，改成你自己的ico文件名
```

添加完后一定要重新 qmake 一遍，然后再运行一遍你的程序，这样再看你的 debug 或者 release 文件夹下的 exe 文件就会有设置好的图标了。

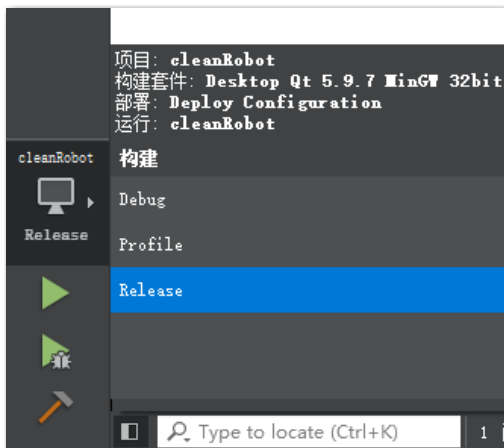


[回到顶部](#)

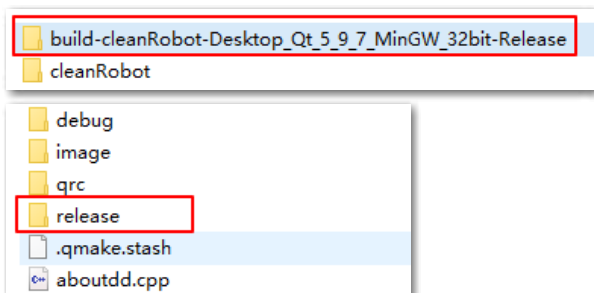
三、发布程序

发布程序首先要获得所依赖的 dll 文件，这里有两种方法。方法一就是去 Qt 安装目录的 bin 目录（笔者这里是 F:\technology\Qt5.9.7\5.9.7\mingw53_32）中找所依赖的 dll 文件。方法二就是使用 Qt 自带的发布程序工具 windeployqt.exe，可以生成所依赖的 dll 文件。这里着重讲解第二种方法，步骤如下：

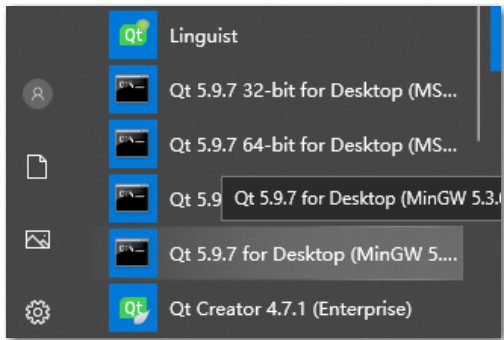
(1) 以 **Release 方式**编译生成 exe 程序,也就是调试运行的方式选择 **Release**。



运行成功后，如果勾选了“shadow build”将源码路径和构建路径分开，那么将在 build-cleanRobot-Desktop_Qt_5_9_7_MinGW_32bit-Release 文件夹下生成 exe 文件，如下图1所示；否则在源码工程目录下的 release 文件夹下生成 exe 文件，如下图2所示。



将 cleanRobot.exe 可执行文件复制到别的文件夹下, 这里我复制到 F:\cleanRobot 下, 然后在“开始菜单Qt5.9.7”下找到“Qt 5.9.7 for Desktop cmd”运行文件, 打开运行。



执行 `cd F:\cleanRobot` 命令进入exe文件所在目录下, 再执行 `dir` 命令查看目录, 最后执行 `windeployqt cleanRobot.exe` 命令, Qt 就会自动把该程序所需要的所有 dll 拷贝过来。

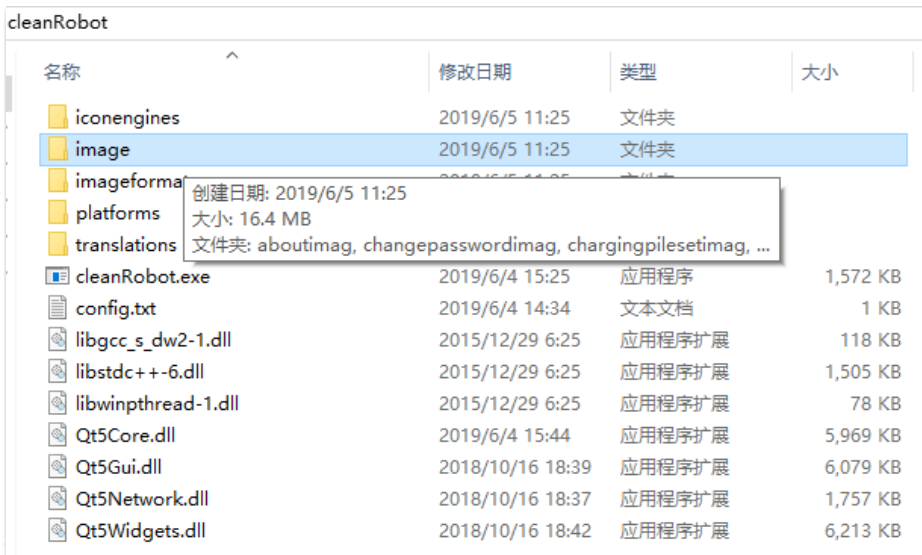
```
Qt 5.9.7 for Desktop (MinGW 5.3.0 32 bit)
Setting up environment for Qt usage...
F:\technology\Qt5.9.7\5.9.7\mingw53_32>cd F:\cleanRobot
F:\cleanRobot>dir
驱动器 F 中的卷是 本地磁盘
卷的序列号是 E46B-6CC0

F:\cleanRobot 的目录
2019/06/05  11:25    <DIR>          .
2019/06/05  11:25    <DIR>          ..
2019/06/04  15:25         1,609,216  cleanRobot.exe
               1 个文件          1,609,216  字节
               2 个目录 187,000,594,432  可用字节

F:\cleanRobot>windeployqt cleanRobot.exe
F:\cleanRobot\cleanRobot.exe 32 bit, release executable
Adding Qt5Svg for qsvgicon.dll
Skipping plugin qtvirtualkeyboardplugin.dll due to disabled dependencies (Qt5Qml Qt5Quick).
Direct dependencies: Qt5Core Qt5Gui Qt5Network Qt5Widgets
All dependencies   : Qt5Core Qt5Gui Qt5Network Qt5Widgets
To be deployed    : Qt5Core Qt5Gui Qt5Network Qt5Svg Qt5Widgets
Updating Qt5Core.dll.
Updating Qt5Gui.dll.
Updating Qt5Network.dll.
Updating Qt5Svg.dll.
Updating Qt5Widgets.dll.
Updating libGLESV2.dll.
Updating libEGL.dll.
```

当然有可能有些 dll 不是我们需要的, 例如下面的 D3Dcompiler_47.dll, 为了使发布程序尽可能的小, 可以把这些 dll 一个一个删掉, 再打开 exe 程序, 看哪些 dll 是删掉后就打开不了程序的, 保留下来, 剩下的删除即可。

再将自带文件夹, 例如下图的 image 文件夹复制过来, 删掉不必要的 dll, 之后的 exe 安装包文件夹如下图所示。



其实到这一步，就可以将整个文件夹发布出去了，但如果想打包成一个 exe 安装包，就要进行下面的步骤了。

[回到顶部](#)

四、打包程序

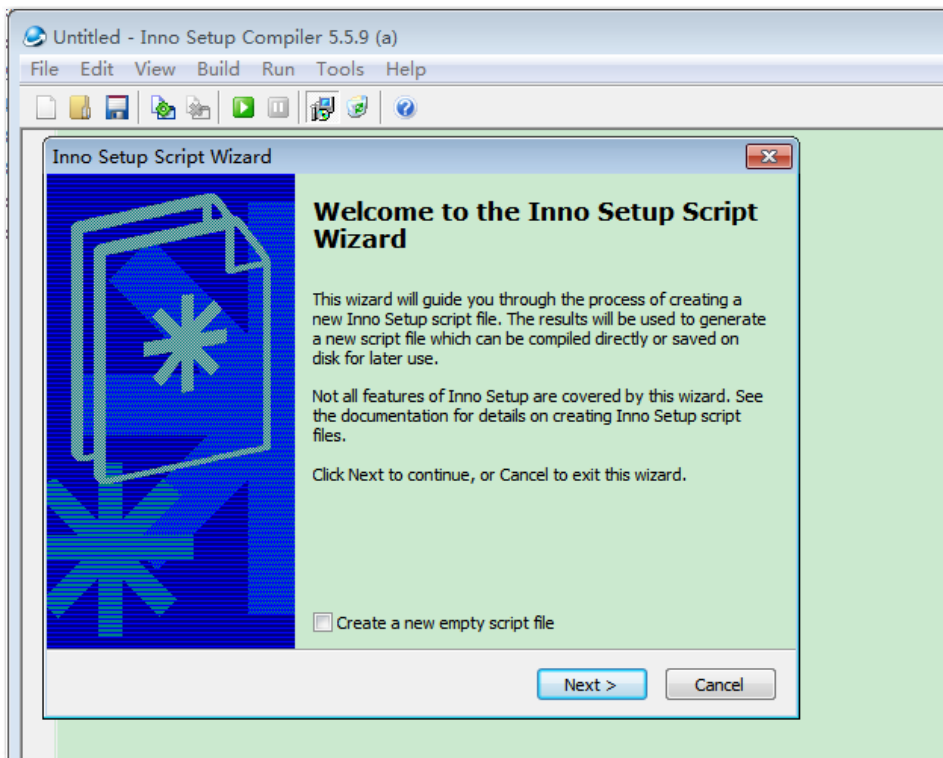
Inno Setup 是一个免费的 Windows 安装程序制作软件，小巧、简便、精美是其最大特点，支持 pascal 脚本，能快速制作出标准 Windows2000 风格的安装界面，足以完成一般安装任务。

官方下载地址：<http://www.jrsoftware.org/isdl.php#stable>

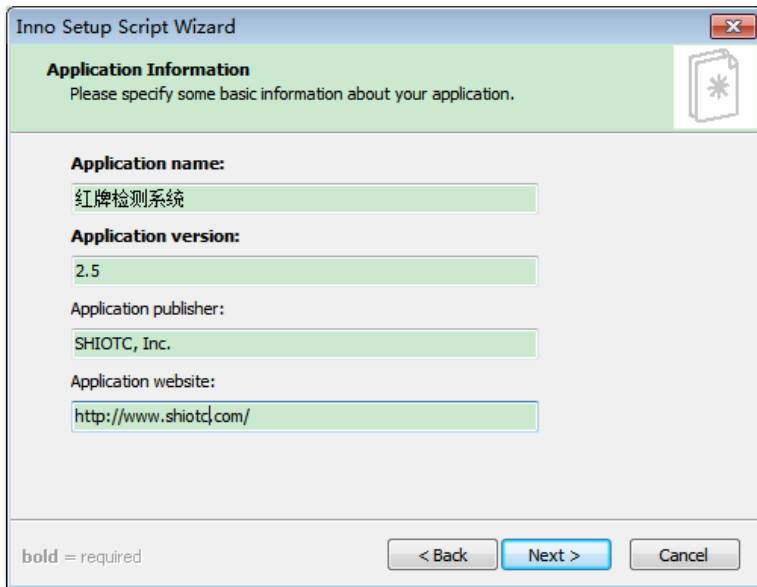
下载后直接安装即可。

项目开发完成之后，需要打包程序发行，可以使用 Inno Setup 软件制作安装 exe。以下详细讲解如何打包应用程序：

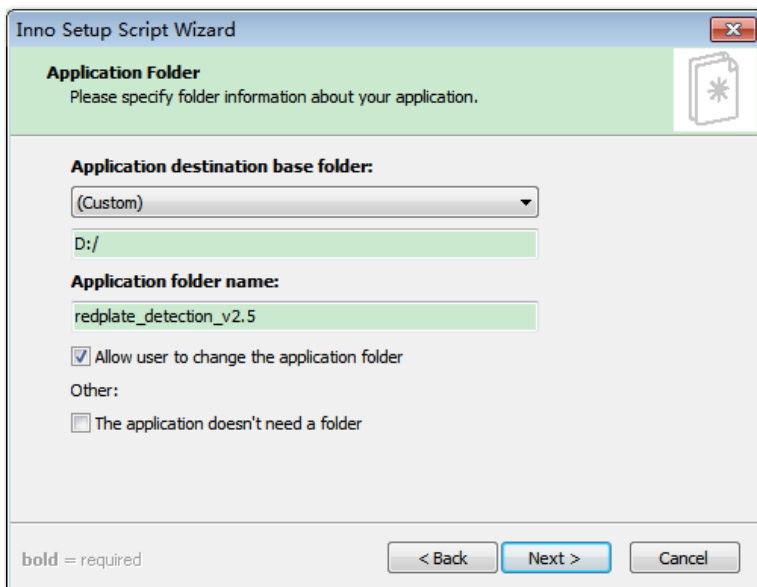
(1) 打开 Inno Setup Compiler，点击【file】→【new】，新建一个脚本向导程序，然后点击【next】；



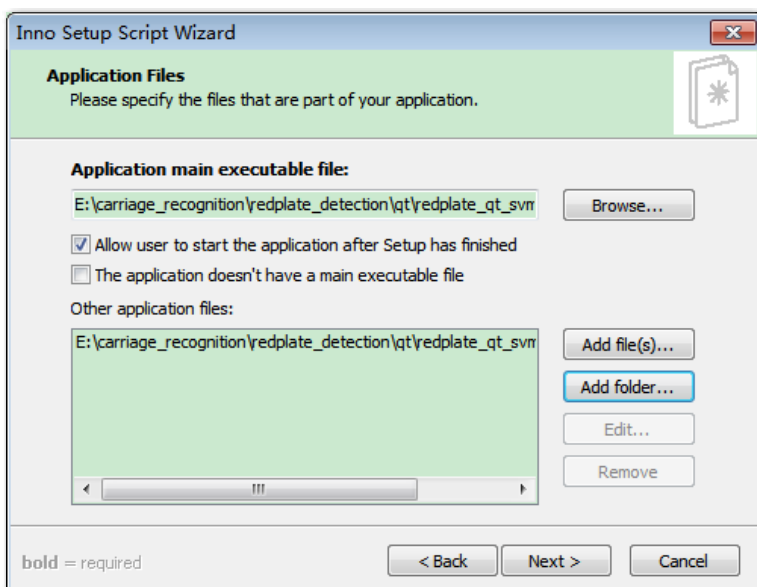
(2) 在应用程序信息中输入“应用程序名称”、“应用程序版本”、“应用程序发布者”、“应用程序网站”，这些信息会自动在脚本中生成，然后点击下一步；



(3) 自定义打包文件的输出地址以及输出文件夹的名称，然后点击下一步；

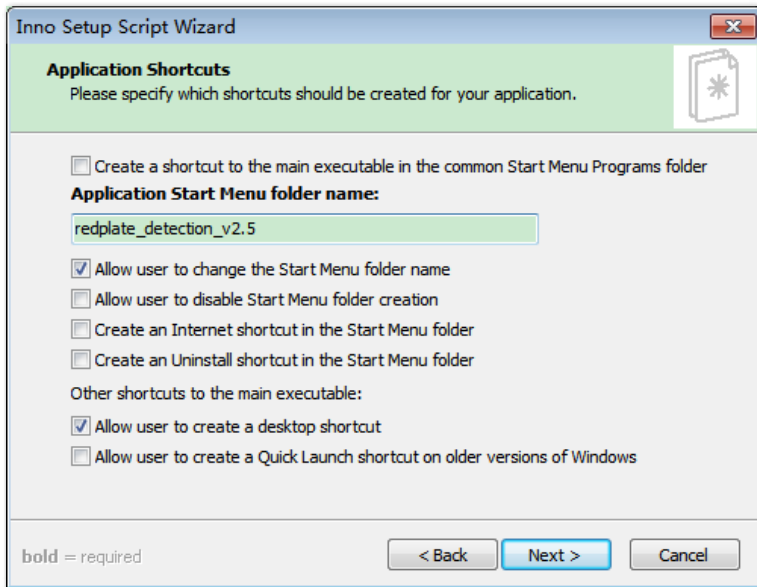


(4) 选择应用程序的主执行文件，也就是你编译生成的 exe 文件，下面点击 “Add folder”，把所有你的程序运行需要的 dll 和资源以及配置文件放在一个文件夹里，选择这个文件夹，然后点击下一步；

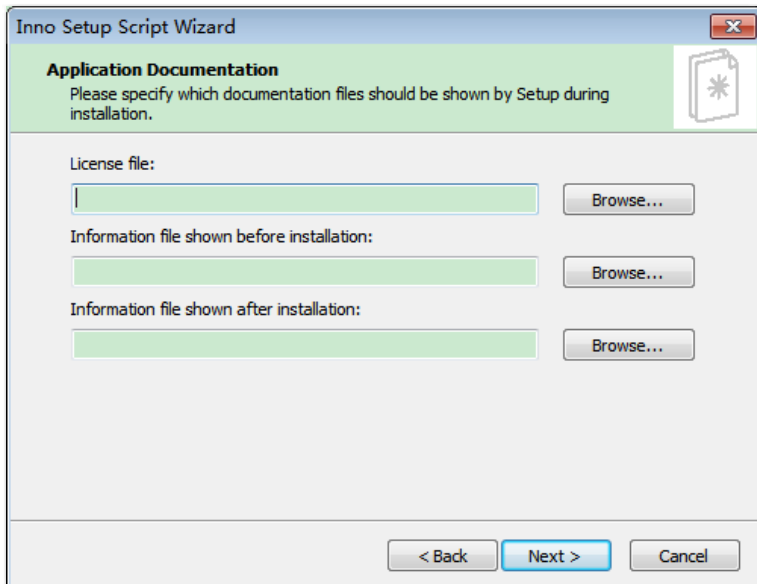


注：Add folder 添加 .exe 的父文件夹，将依赖的 dll 等文件全部包含在内。

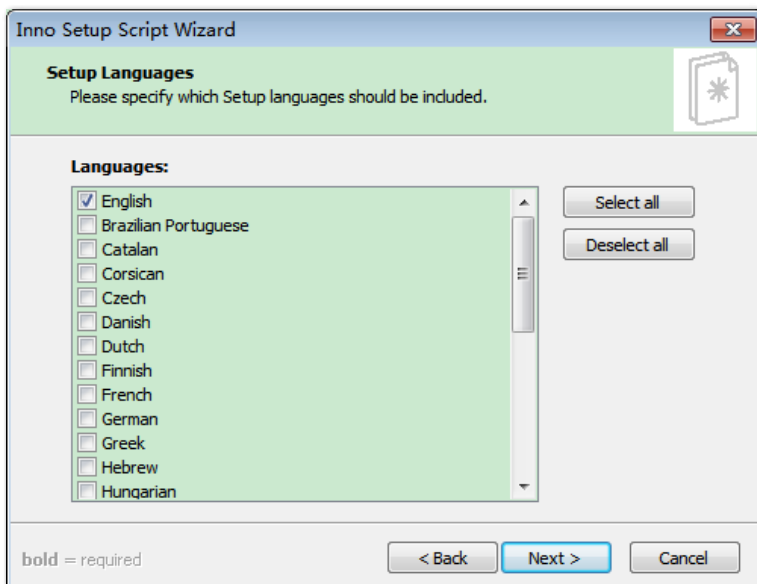
(5) 输入应用程序在开始菜单中显示的名字，需要哪些功能，在前边打上对勾，一般默认即可，然后点击下一步；



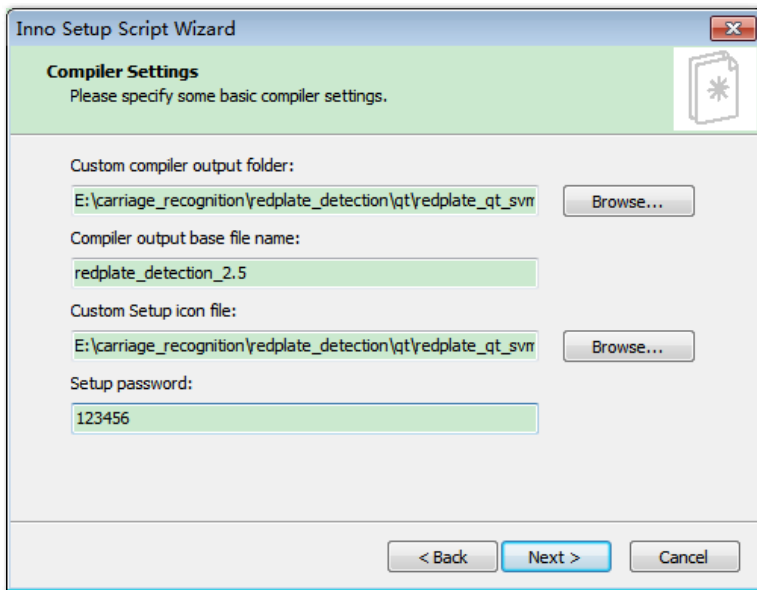
(6) 这里可以选择“许可文件”，“安装前的文件”（如 Readme 文件等），“安装后显示信息”等，然后点击下一步；



(7) 选择“安装语言”，然后点击下一步；

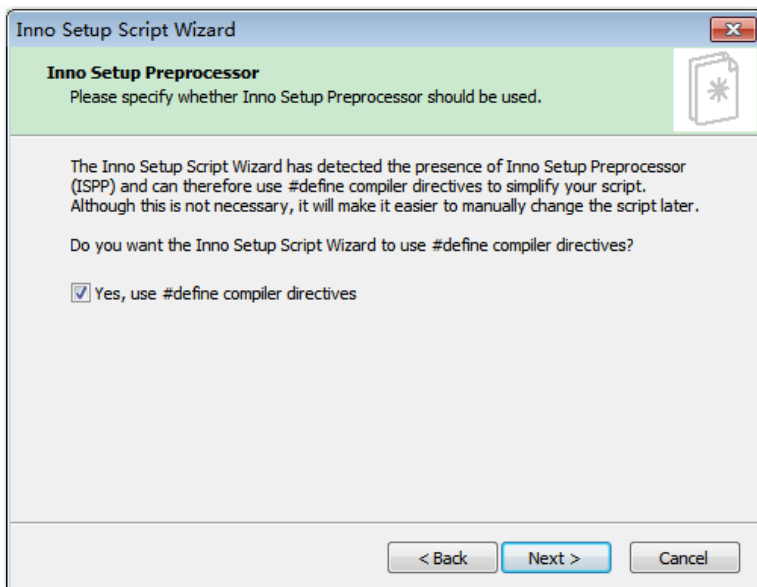


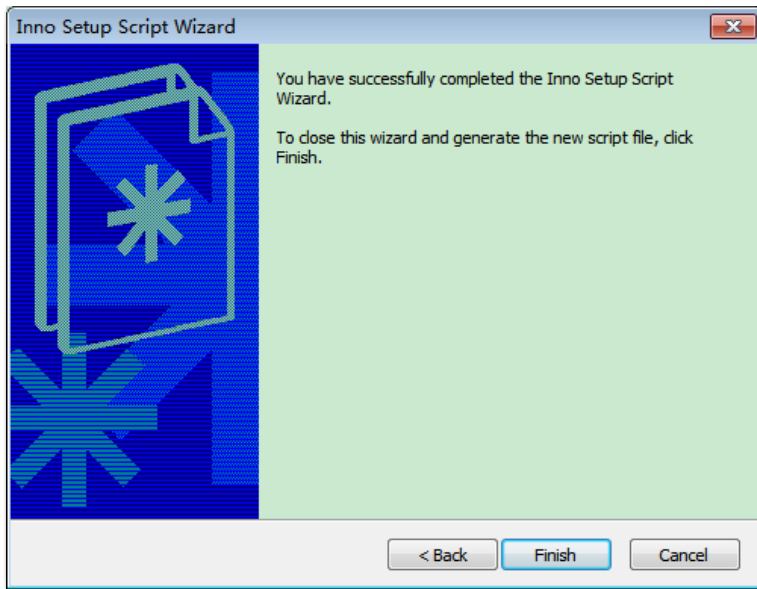
(8) 选择“自定义编译器输出文件夹”、“编译器输出基本文件名”、“自定义的安装程序图标文件”及“安装密码”，然后点击下一步；



注：第一个选项表示的是生成的 安装包*.exe 的输出目录。icon文件是 安装包*.exe 的图标文件。

(9) 在这里选择是否使用 #define 编译指令，选择 yes 也就是在生成的脚本里使用 #define 宏定义一些常量，方便脚本编写，勾不勾选随意，然后点击下一步；





前面最后一步中间会询问你是否立即编译新的脚本，这个脚本是根据刚才我们前面填写的那些内容自动生成的。可以直接使用编译，也可以自己重新编写，下面贴出生成的脚本代码。

```
; Script generated by the Inno Setup Script Wizard.
; SEE THE DOCUMENTATION FOR DETAILS ON CREATING INNO SETUP SCRIPT FILES!
```

```
#define MyAppName "红牌检测系统"
#define MyAppVersion "2.5"
#define MyAppPublisher "SHIOTC, Inc."
#define MyAppURL "http://www.shiotc.com/"
#define MyAppExeName "redplate_qt.exe"
```

```
[Setup]
```

```
; NOTE: The value of AppId uniquely identifies this application.
; Do not use the same AppId value in installers for other applications.
; (To generate a new GUID, click Tools | Generate GUID inside the IDE.)
AppId={{D21B4F6B-401E-4575-B6E1-612A9CA50D58}}
AppName={#MyAppName}
AppVersion={#MyAppVersion}
;AppVerName={#MyAppName} {#MyAppVersion}
AppPublisher={#MyAppPublisher}
AppPublisherURL={#MyAppURL}
AppSupportURL={#MyAppURL}
AppUpdatesURL={#MyAppURL}
DefaultDirName=D:/redplate_detection_v2.5
DefaultGroupName=redplate_detection_v2.5
OutputDir=E:\carriage_recognition\redplate_detection\qt\redplate_qt_svm2\x64\Release
OutputBaseFilename=redplate_detection_2.5
;SetupIconFile=E:\carriage_recognition\redplate_detection\qt\redplate_qt_svm2\x64\Release\ICON.ico
Password=123456
Compression=lzma
SolidCompression=yes
```

```
[Languages]
```

```
Name: "english"; MessagesFile: "compiler:Default.isl"
```

```
[Tasks]
```

```
Name: "desktopicon"; Description: "{cm:CreateDesktopIcon}"; GroupDescription: "
{cm:AdditionalIcons}"; Flags: unchecked
```

```
[Files]
```

```
Source:
```

```
"E:\carriage_recognition\redplate_detection\qt\redplate_qt_svm2\x64\Release\redplate_qt.exe";
```



```

DestDir: "{app}"; Flags: ignoreversion
Source: "E:\carriage_recognition\redplate_detection\qt\redplate_qt_svm2\x64\Release\*"; DestDir: "
{app}"; Flags: ignoreversion recursesubdirs createallsubdirs
; NOTE: Don't use "Flags: ignoreversion" on any shared system files

```

[Icons]

```

Name: "{group}\{#MyAppName}"; Filename: "{app}\{#MyAppExeName}"
Name: "{commondesktop}\{#MyAppName}"; Filename: "{app}\{#MyAppExeName}"; Tasks: desktopicon

```

[Run]

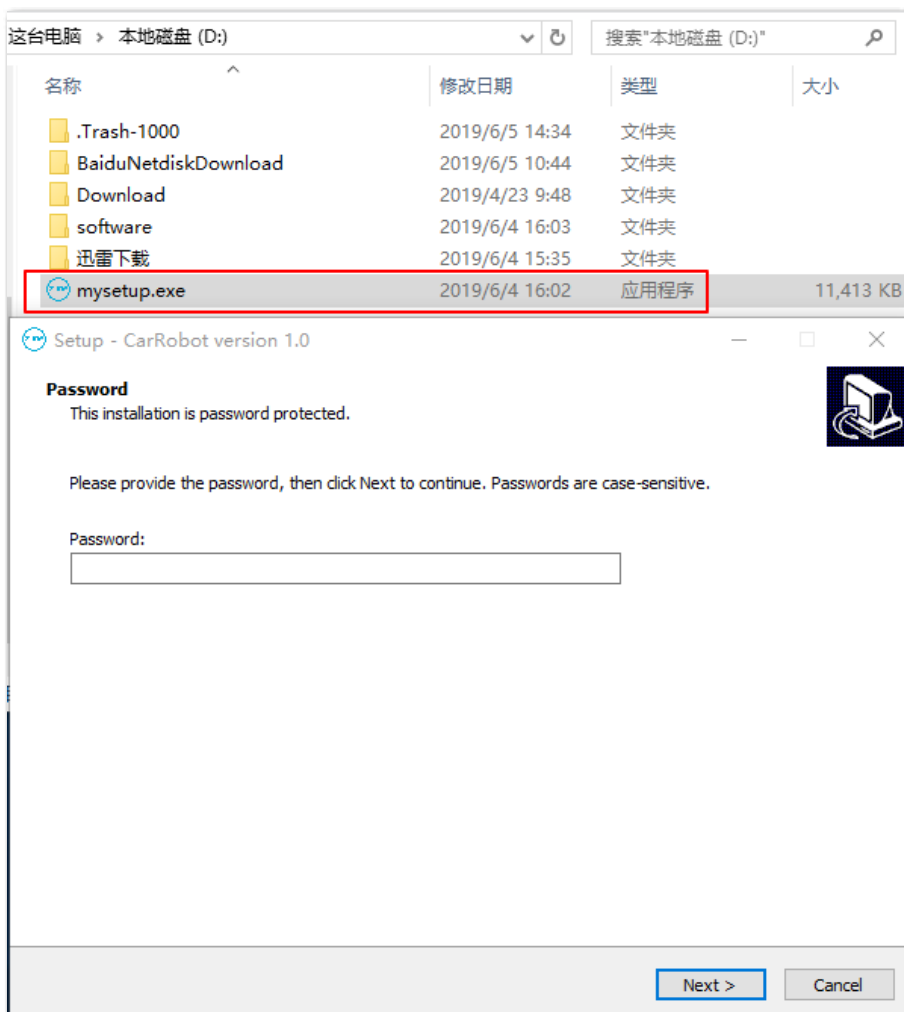
```

Filename: "{app}\{#MyAppExeName}"; Description: "{cm:LaunchProgram,{#StringChange(MyAppName, '&', '&&')}}"; Flags: nowait postinstall skipifsilent

```

- 运行过程中出现错误，可以直接在脚本中进行修改。
- 如果要打包出新版本的 exe 安装包文件，不需要再将上面流程走一遍，只需要将原先的 exe 可执行文件替换掉，再打开原先生成的脚本文件运行，即可生成新版本的 exe 安装包文件。

最后就生成了 exe 安装包文件，点击打开exe文件即可进行安装，如下图所示。



参考:

[Qt使用自带的windeployqt 生成exe来发布软件](#)

[QT 程序发布以及用Inno Setup打包成安装包](#)

[使用Inno Setup Compiler制作安装软件包](#)

分类: [Qt应用](#)

好文要顶

关注我

收藏该文

fengMisaka

关注 - 10

粉丝 - 70

+加关注

« 上一篇： Qt下存储读写应用程序设置的三种方法

» 下一篇： Qt事件系统之一：Qt中的事件处理与传递

posted @ 2019-06-04 17:33 fengMisaka 阅读(2187) 评论(2) 编辑 收藏

评论列表

#1楼 2019-10-14 13:37 糖炒浆糊

您好，这篇文章可以转载吗

支持(0) 反对(0)

#2楼 [楼主] 2019-10-14 13:51 fengMisaka

同学你好，没事，随便转载。

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)， [访问](#) 网站首页。

- 相关博文：
- [Qt之新手打包发布程序](#)
 - [Qt 程序打包发布总结](#)
 - [winform打包发布安装包详解..](#)
 - [Qt 程序打包发布总结](#)
 - [Qt开发北斗定位系统融合百度地图API及Qt程序打包发布](#)
- » 更多推荐...
- [12知识点+20干货案例+110面试题，助你拿offer！ | Python工程师面试宝典](#)

搜索

找找看

随笔分类

- C++(17)
- C++小项目(4)
- cmake(4)
- C语言(7)
- C语言小项目(4)
- Linux(19)
- Qt Quick(18)
- Qt学习(41)
- Qt应用(12)
- ROS(12)

[shell脚本\(9\)](#)[STL\(8\)](#)[常见笔试面试题\(7\)](#)[读书笔记\(5\)](#)[前端\(5\)](#)[数据结构\(23\)](#)[数据库\(2\)](#)[杂项\(12\)](#)

Copyright © 2019 fengMisaka
Powered by .NET Core 3.0.1 on Linux