EGE图形库配置教程

在程序设计中，常常需要进行图形处理操作。C语言标准库不带有图形处理的功能，如果想要绘制图形，需要调用操作系统提供的各种接口（操作系统本身大部分也是C语言编写的）。

目前使用率最高的桌面操作系统是Windows系统，此系统提供一系列C语言API函数，称为Win32 API，这些API声明在Windows.h中。通过Win32 API，可以用C语言实现操作窗口、处理图形、调用设备、访问服务和执行Shell命令。不过Win32 API比较复杂，涉及许多Windows程序设计的相关知识，比如窗口类、消息循环、消息映射等等，除非进行系统的学习，否则初学者难以学会使用Win32 API来进行开发。

为解决这个问题，一些公司或者开源组织对Win32 API进行封装，把这些API包装成更加便捷易用的程序框架，比如Qt、MFC等。Qt库功能强大，本身比较大型，是跨平台的应用程序开发框架，安装配置稍复杂。MFC(Microsoft Foundation Classes)库由于是通用框架，没有最好的针对性，丧失一些灵活性和效率，且只适合Windows环境。

另有更为强大的OpenGL和DirectX等。OpenGL(Open Graphics Library)是跨编程语言、跨平台的专业图形[程序接口](https://baike.baidu.com/item/%E7%A8%8B%E5%BA%8F%E6%8E%A5%E5%8F%A3" \t "https://baike.baidu.com/item/_blank)。DirectX(Direct eXtension)是多媒体编程接口，功能强大，只支持Windows平台。虽然OpenGL和DirectX强大，但是由于学习的门槛高，不适合初学者直接使用。

EGE(Easy Graphics Engine)图形库是最适合初学者的图形库之一，比起Qt和MFC，它的功能不算强大，但却有着体积小、上手简单的优点。EGE图形库可以作为控制台程序与GUI程序中间一个承上启下的过渡点。

**1 前期准备**

**1.1下载图形库**

通过以下任一链接地址下载EGE图形库：

EGE图形库主站下载地址：<http://xege.org/>

百度网盘分享链接：<https://pan.baidu.com/s/1nlG7BH_de5XFVsW64pHvnw>

密码：qzcg

这里主要下载EGE\_for\_Visual\_Studio\_2017和ege15.04两个压缩包，分别对应着Visual Studio 2017版本和Visual Studio 2015及以下版本的EGE图形库。

**1.2图形库目录的组织结构**

整个图形库的大小只有不到3M，解压之后，在EGE根目录会有以下文件夹：

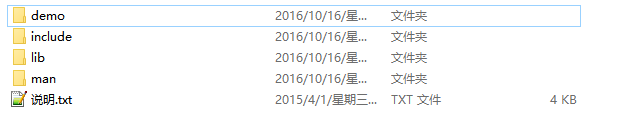


图1-1 EGE根目录组织结构

其中，demo文件夹下是EGE的作者提供的一些基于EGE图形库的演示程序，其中不乏很多高水平的作品。include下是EGE图形库的头文件，使用图形库，需要先包含头文件graphics.h。lib下是EGE的静态库，是整个图形库的具体功能实现。man下是说明文档，EGE提供的函数都可以在文档里面找到详细说明。

**2 配置方法**

**2.1创建项目**

打开Visual Studio，点击窗口左上角“菜单栏”中的“文件”，在弹出菜单中依次选择“新建->项目”。此时弹出“新建项目”窗口，窗口左边按照“已安装->模板->C/C++”依次展开，在窗口右边，Visual Studio 2017版本选择“空项目”，Visual Studio 2015及以下版本选择“Win32控制台应用程序”，接着为项目取名”Build”，写进“名称”，选择项目存放“位置”，最后点击“确定”。

到此，Visual Studio 2017自动开始创建项目，而Visual Studio 2015及以下版本则会弹出“Win32应用程序向导”窗口，点击“下一步”，勾选“附加选项”里面的“空项目”，去掉“安全开发生命周期(SDL)检查”的勾选，最后点击“完成”，开始自动创建项目。

为项目添加一个源文件。用鼠标右键点击“解决方案资源管理器”中”Build”项目下的“源文件”文件夹，弹出右键菜单，依次选择“添加->新建项”，如图2-1。

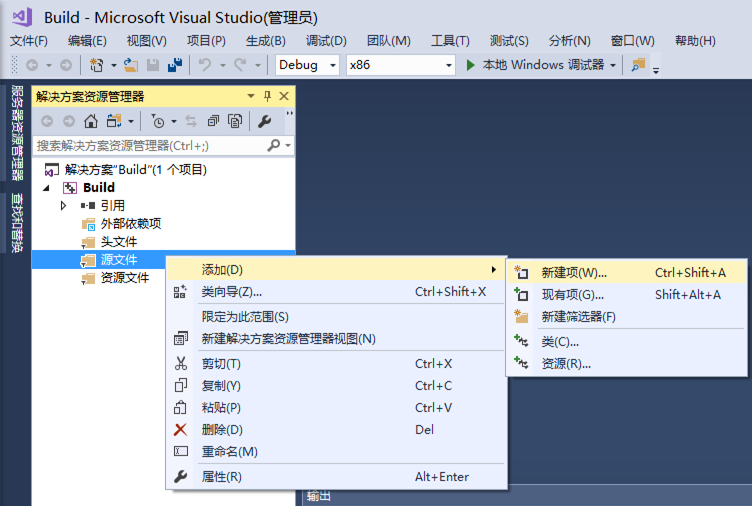


图2-1 添加源文件

此时会弹出“添加选项”窗口，在窗口左边按照“已安装->Visual C++”依次展开，在窗口右边选择“C++ 文件(.cpp)”，为文件取名，在“名称”输入框中写”main.cpp”，最后点击“添加”。

**2.2构建项目环境**

此处使用相对路径的配置方法，虽然操作复杂，但是具有可移植性，即在另一台计算机的相同开发环境下不需改动（或者做少许改动）就可以编译运行。

在Build项目的源代码目录下创建include和lib文件夹，我创建好的Build项目位于"E:\Microsoft Visual Studio Projects\Build"，如图2-2。

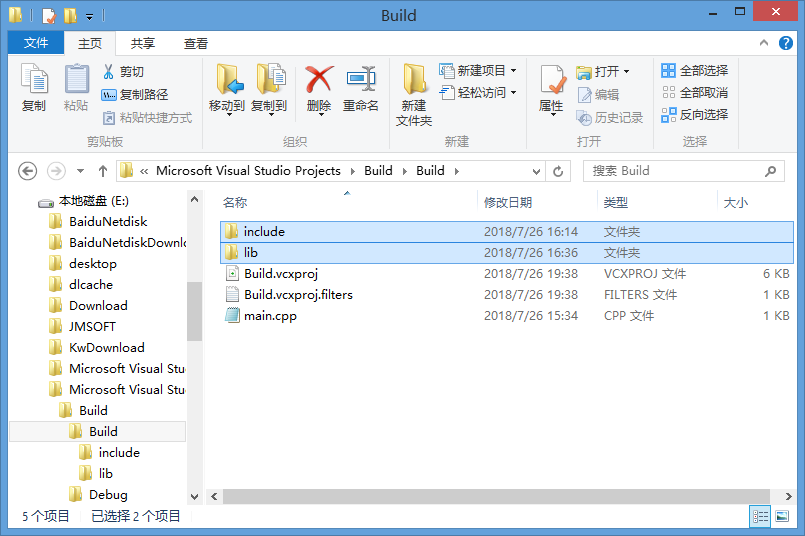


图2-2 Build项目源代码目录下的include和lib文件夹

接着解压EGE压缩包，把位于EGE根目录的include文件夹复制到源代码目录的include文件夹中，并重命名为”ege”。进入EGE根目录的lib文件夹（lib文件夹中有不同的子文件夹，分别对应不同的开发环境），把对应当前开发环境的文件夹复制到源代码目录的lib文件夹中，并重命名为”ege”。比如开发环境是Visual Studio 2017，就复制vc2017文件夹。如果开发环境是Visual Studio 2015，就复制vc2015文件夹。

**注意**

拷贝include文件夹和vc文件夹时需按照开发环境版本找对应的压缩包，EGE\_for\_Visual\_Studio\_2017和ege15.04两压缩包分别对应着Visual Studio 2017版本和Visual Studio 2015及以下版本。

解压之后EGE\_for\_Visual\_Studio\_2017根目录的lib文件夹如图2-3所示：

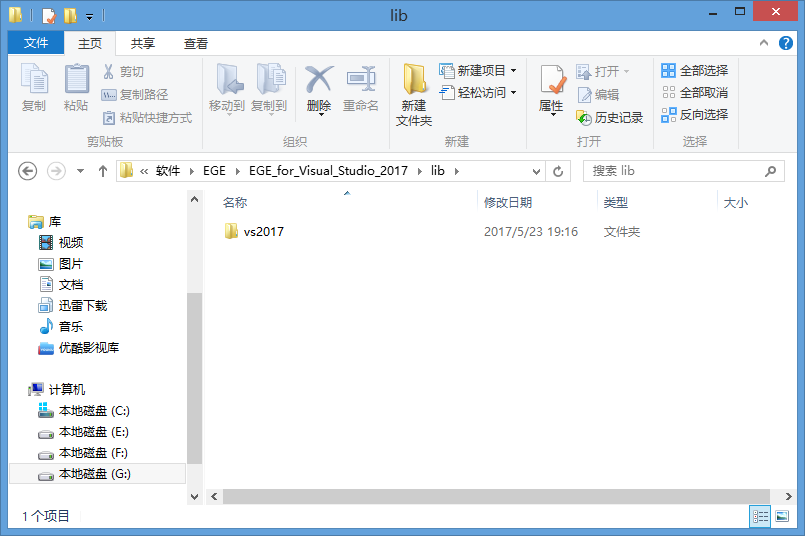


图2-3 EGE\_for\_Visual\_Studio\_2017根目录的lib文件夹

解压之后ege15.04根目录的lib文件夹如图2-4所示：

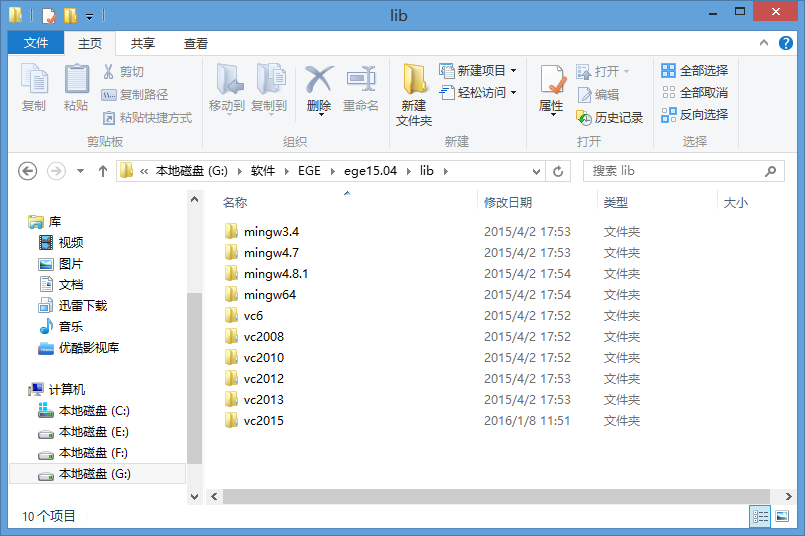


图2-4 ege15.04根目录的lib文件夹

**2.3配置项目属性**

用鼠标右击“解决方案资源管理器”中的”Build”项目，如图2-5所示：

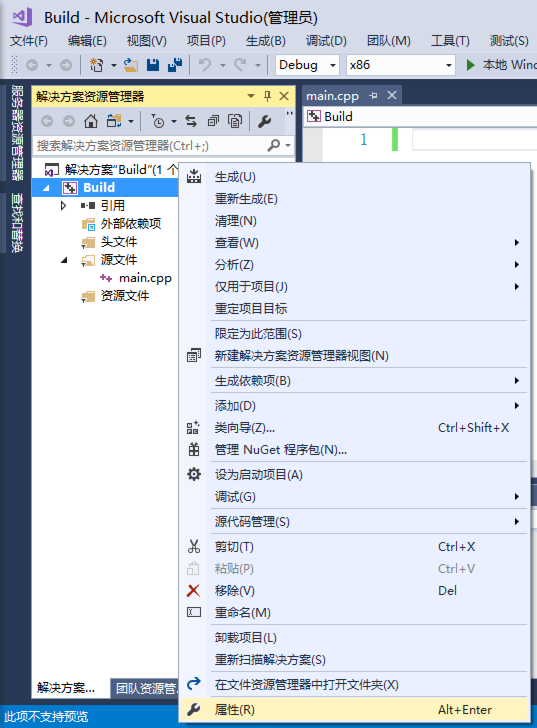


图2-5 解决方案资源管理器中项目的右键菜单

选择右键菜单最下面的“属性”选项，弹出项目属性页，点开左边“属性配置”中的“C/C++”，点击“常规”，在右边的“附加包含目录”中输入”include/ege”。如图2-6所示：

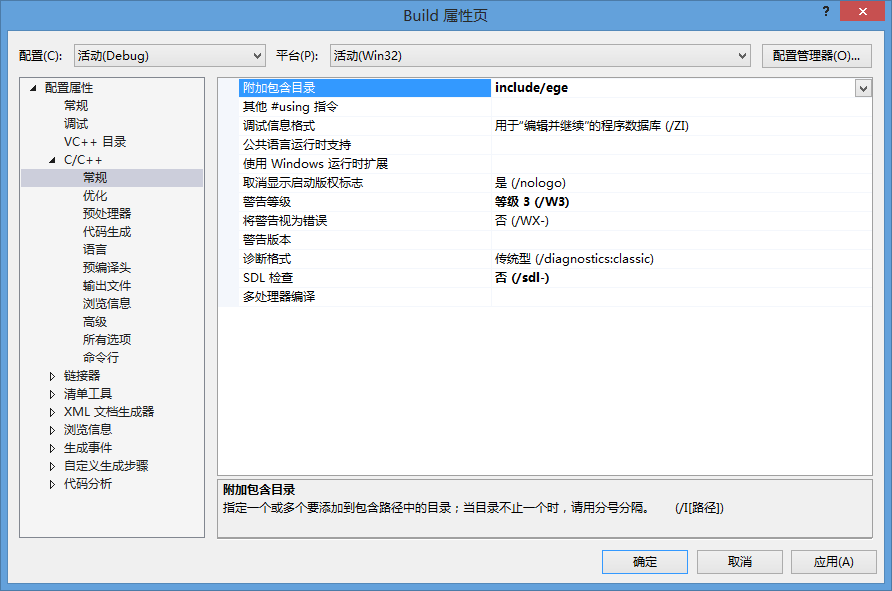


图2-6 以相对路径方式配置附加包含目录

再点开左边的“链接器”，点击“常规”，若在构建项目环境时拷贝的是vc2017文件夹，由于在此文件夹中还有x64和x86两个文件夹，这里编写32位应用程序，使用x86文件夹中的32位图形库，因此在右边的“附加库目录”输入”lib/ege/x86”，如图2-7所示。如果拷贝的是低版本的vc文件夹，则在“附加库目录”输入”lib/ege”即可。

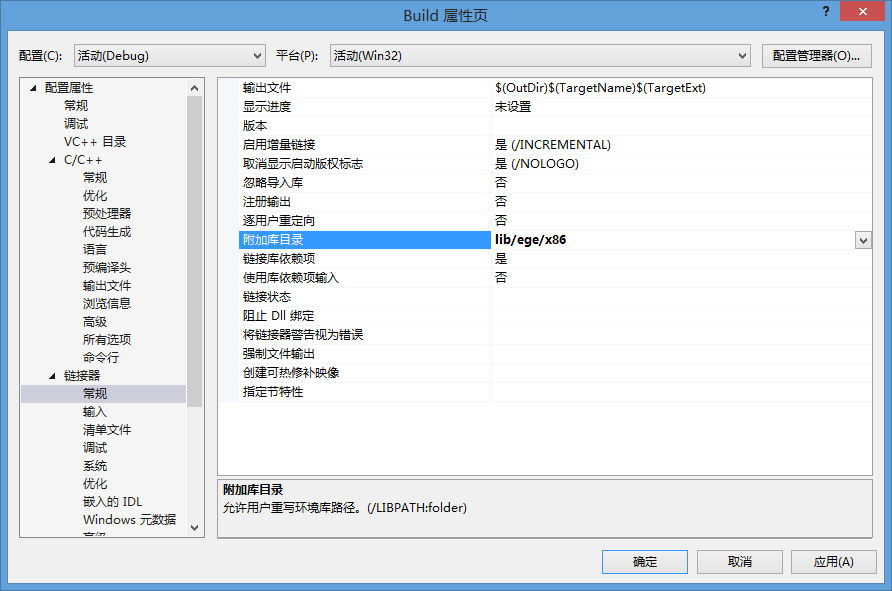


图2-7 以相对路径方式配置附加库目录

最后点击“确定”，完成VS开发环境以相对路径引用EGE图形库的步骤。

**说明**

一般编写的应用程序是32位x86应用程序，较少编写64位的x64应用程序。一般需要使用大内存的应用程序才编译成64位，使用内存较小的应用程序编译成64位反而浪费资源。

**3 测试图形库**

在配置完开发环境之后，编写以下代码，测试EGE图形库：

1. #include <graphics.h>
2. int main()
3. {
4. initgraph(640, 480);//初始化一个640\*480的绘图窗口
5. setcolor(RED);//设置绘图前景色为绿色
6. line(64, 48, 576, 432);//绘制一条直线
7. getch();//等待按键
8. closegraph();//关闭绘图窗口
9. return 0;
10. }

**说明**

如果使用了预编译头，需要在文件最上面加上一行代码，引用stdafx.h头文件。

#include "stdafx.h"

按F5或者点击“工具栏”中的“本地Windows调试器”进行调试，运行结果如图3-1所示：

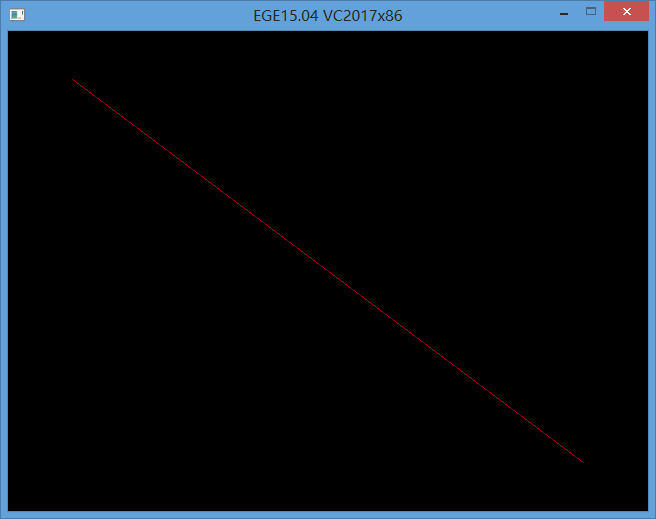


图3-1 测试程序运行结果

**注意**

由于之前按照32位配置图形库，需保证“工具栏”中的“解决方案平台”是”x86”。若按照64位配置图形库，先在“工具栏”中的“解决方案平台”选择”x64”，然后再按照配置项目属性的步骤重新配置。