

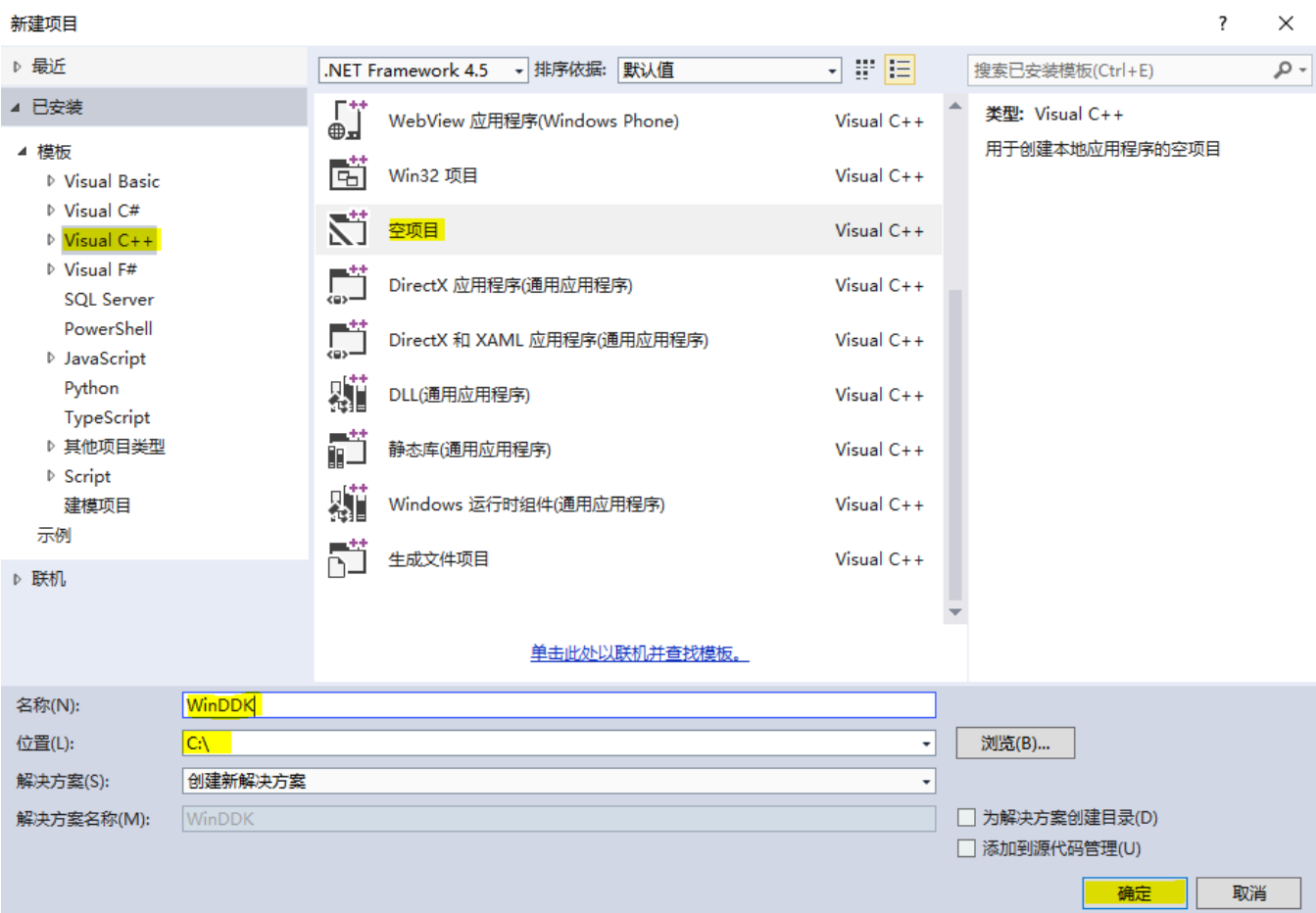
Windows Driver Kit 是一种完全集成的驱动程序开发工具包，它包含 WinDDK 用于测试 Windows 驱动器的可靠性和稳定性，本次实验使用的是 WDK8.1 驱动开发工具包，该工具包支持 Windows 7到 Windows 10 系统的驱动开发。

- 驱动WDK工具包推荐: Windows Driver Kit(WDK) v8.1 离线安装包
- 配置好的案例下载: <https://cdn.lyshark.com/code/WinDDK.zip>

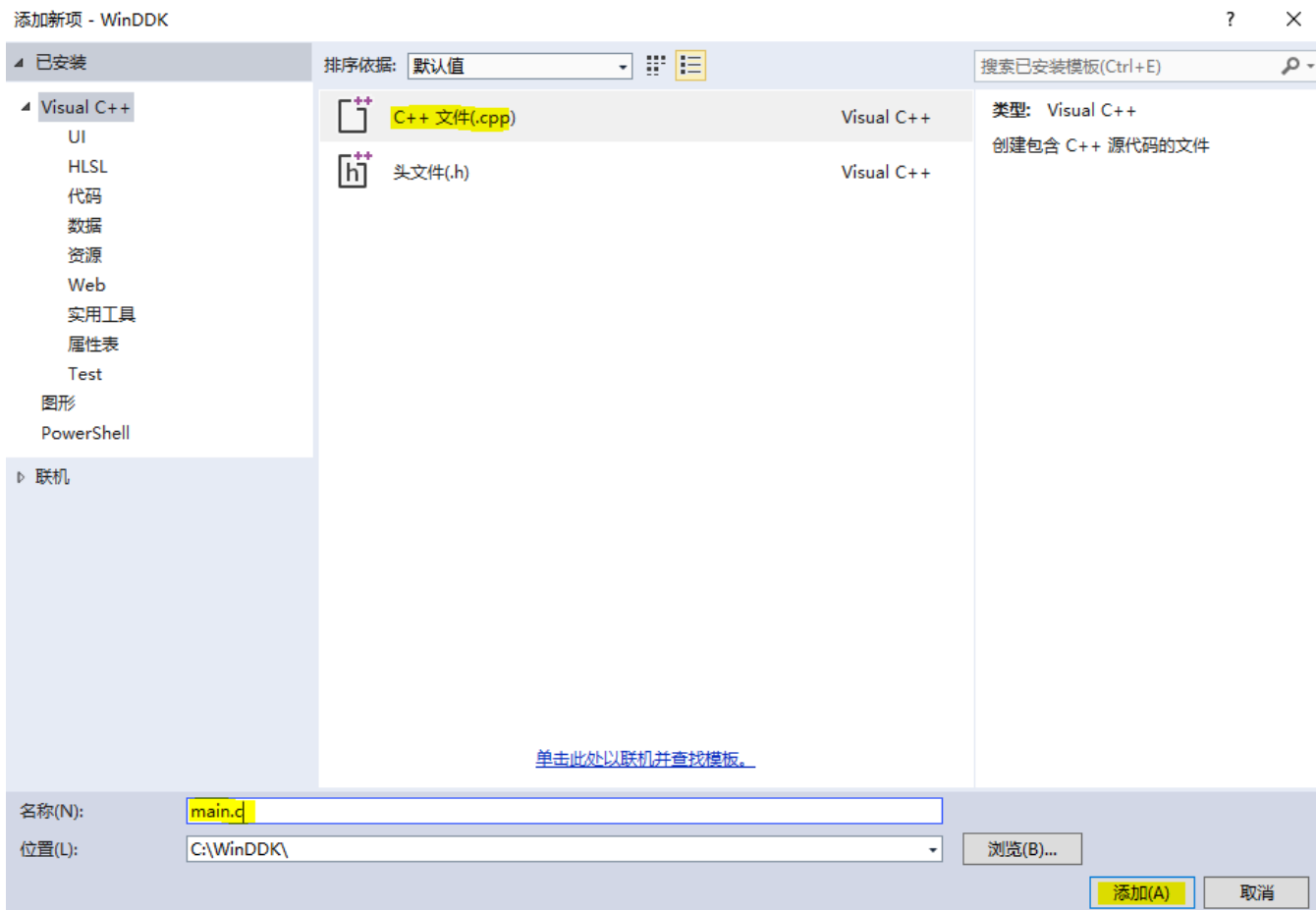
首先你需要先安装好 Visual Studio 2013 的开发环境，然后再安装 Windows Driver Kit 8.1 的驱动开发工具包，这个工具包安装好以后1.5G左右，不过我已经把这个安装包中的关键库文件提取出来了，提取出的文件只有 80MB，直接将提取好的 WinDDK.zip 解压缩到指定文件中，并配置环境即可使用，省去了安装WDK的麻烦。

为了方便开发，您可以使用 Visual Studio 提供的驱动开发模板，这些模板可以快速创建驱动程序项目，同时包含了一些常用的代码和设置。在创建项目时，您可以选择相应的模板，以便快速开始开发工作。同时，您还需要设置正确的编译和构建选项，以确保生成的驱动程序可以正确运行。

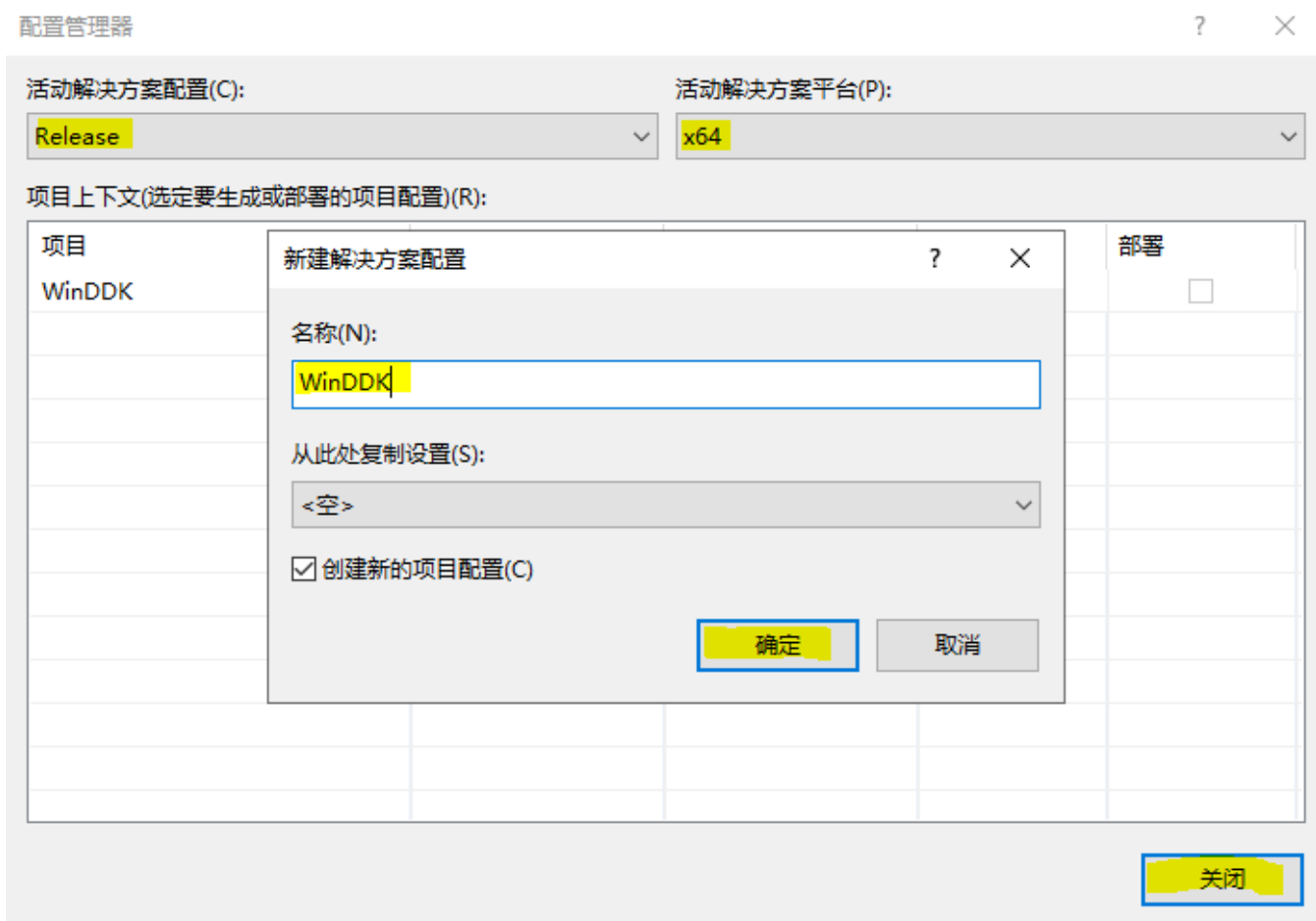
1.打开 Visual Studio 开发工具，然后选择 文件 菜单新建项目，并在已安装模板中选中 Visual C++ 新建空项目，并将项目名称命名为 WinDDK 点击确定。



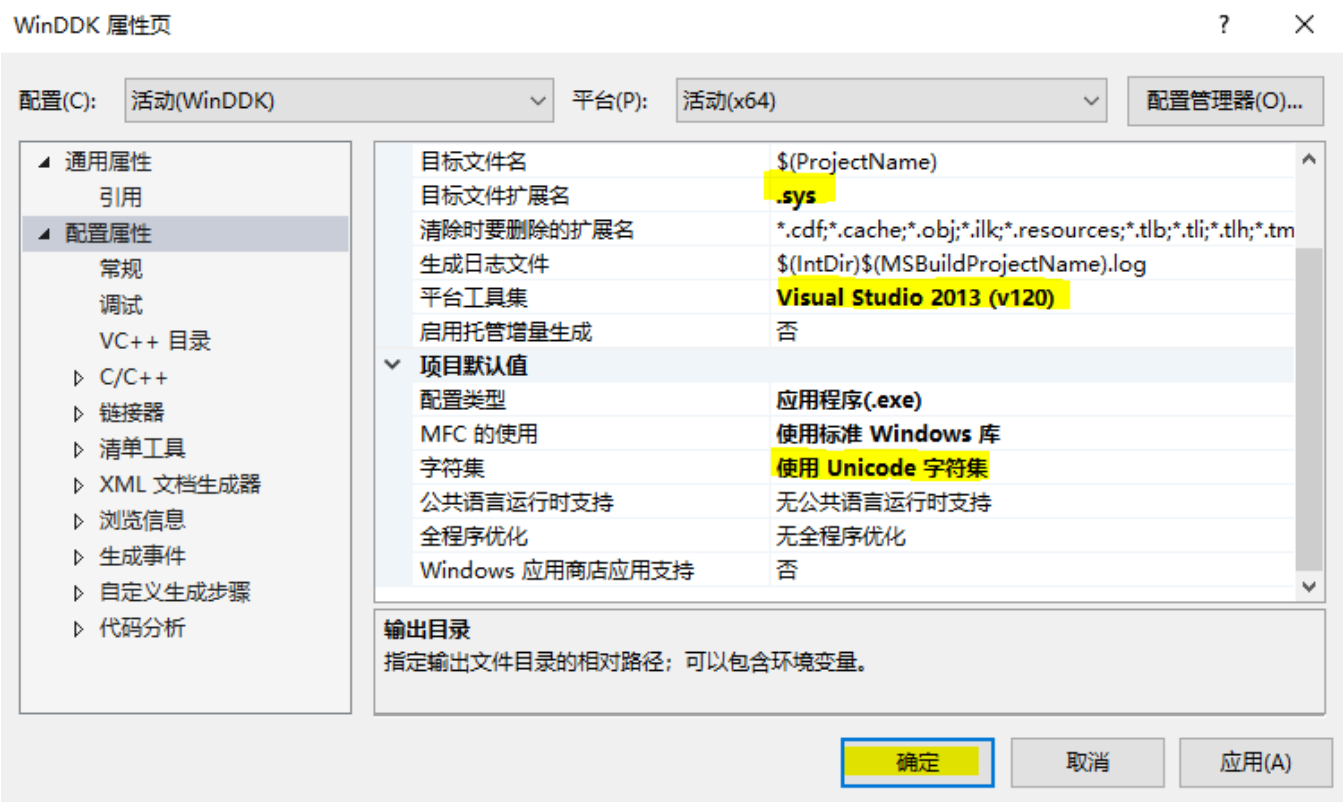
2.依次选择 解决方案视图-源文件-添加新建项 选项卡，或者直接按下 Ctrl + Shift + A 快捷打开菜单，并创建 main.c 文件。



3.接着需要修改配置管理器，添加自定义配置管理，选择生成-配置管理器-新建 选项卡，此处我们命名为 WinDDK 即可。



4.修改配置属性中的 常规 属性，点击菜单栏中的调试，选择 WinDDK属性-配置-常规 修改为标黄处所示内容即可。



5.配置可执行文件路径与导入库路径，这里我们选择 配置属性-VC++目录 依次将如下信息填入配置项。

可执行目录

C:\Program Files (x86)\windows kits\8.1\bin\x64
C:\Program Files (x86)\windows kits\8.1\bin

包含目录

C:\Program Files (x86)\windows kits\8.1\Include\km
C:\Program Files (x86)\windows kits\8.1\Include\shared
C:\Program Files (x86)\windows kits\8.1\Include\um
C:\Program Files (x86)\windows kits\8.1\Include\wdf\kmdf\1.13
C:\Program Files (x86)\windows kits\8.1\Include\wdf\umdf\2.0
C:\Program Files (x86)\windows kits\8.1\Include\winrt

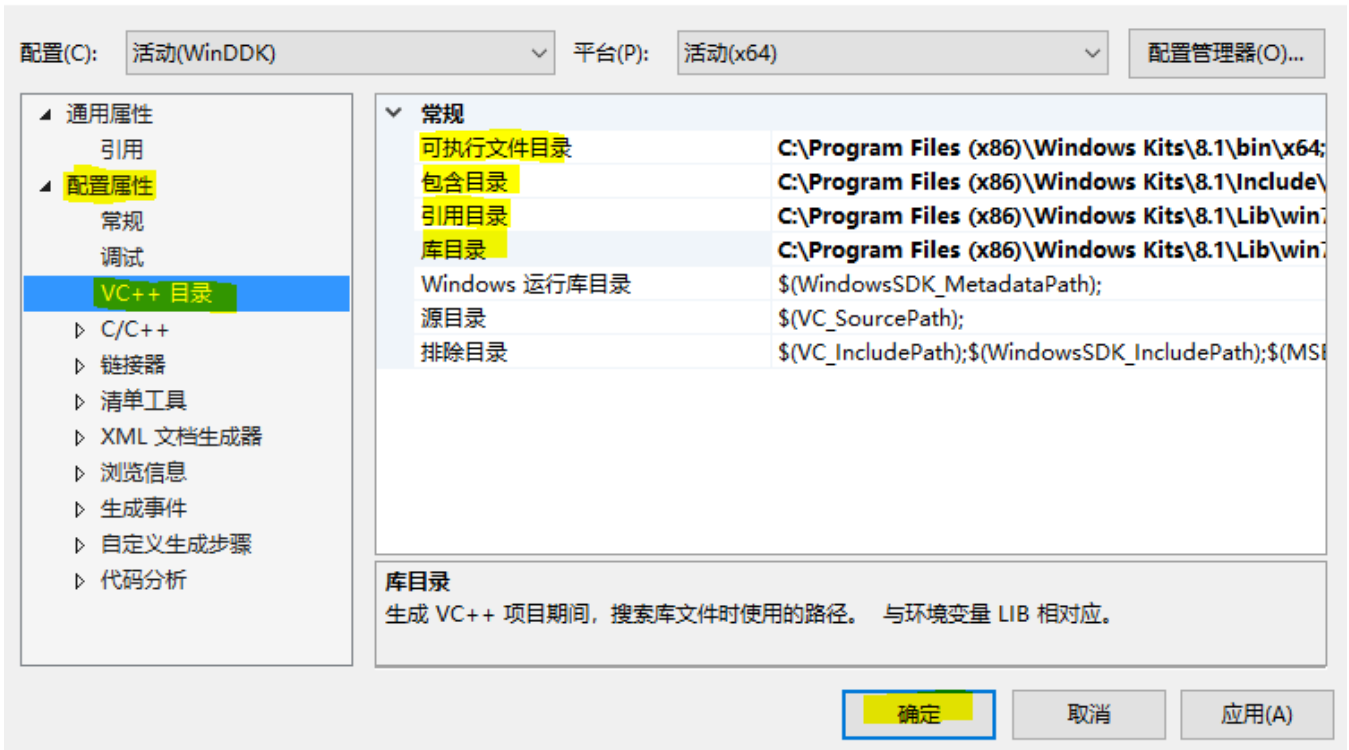
引用目录

C:\Program Files (x86)\windows kits\8.1\Lib\win7\km\x64

库目录

C:\Program Files (x86)\windows kits\8.1\Lib\win7\km\x64
C:\Program Files (x86)\windows kits\8.1\Lib\wdf\kmdf\x64\1.13
C:\Program Files (x86)\windows kits\8.1\Lib\wdf\umdf\x64\2.0
C:\Program Files (x86)\windows kits\8.1\Lib\winv6.3\um\x64
C:\Program Files (x86)\windows kits\8.1\Lib\winv6.3\km\x64

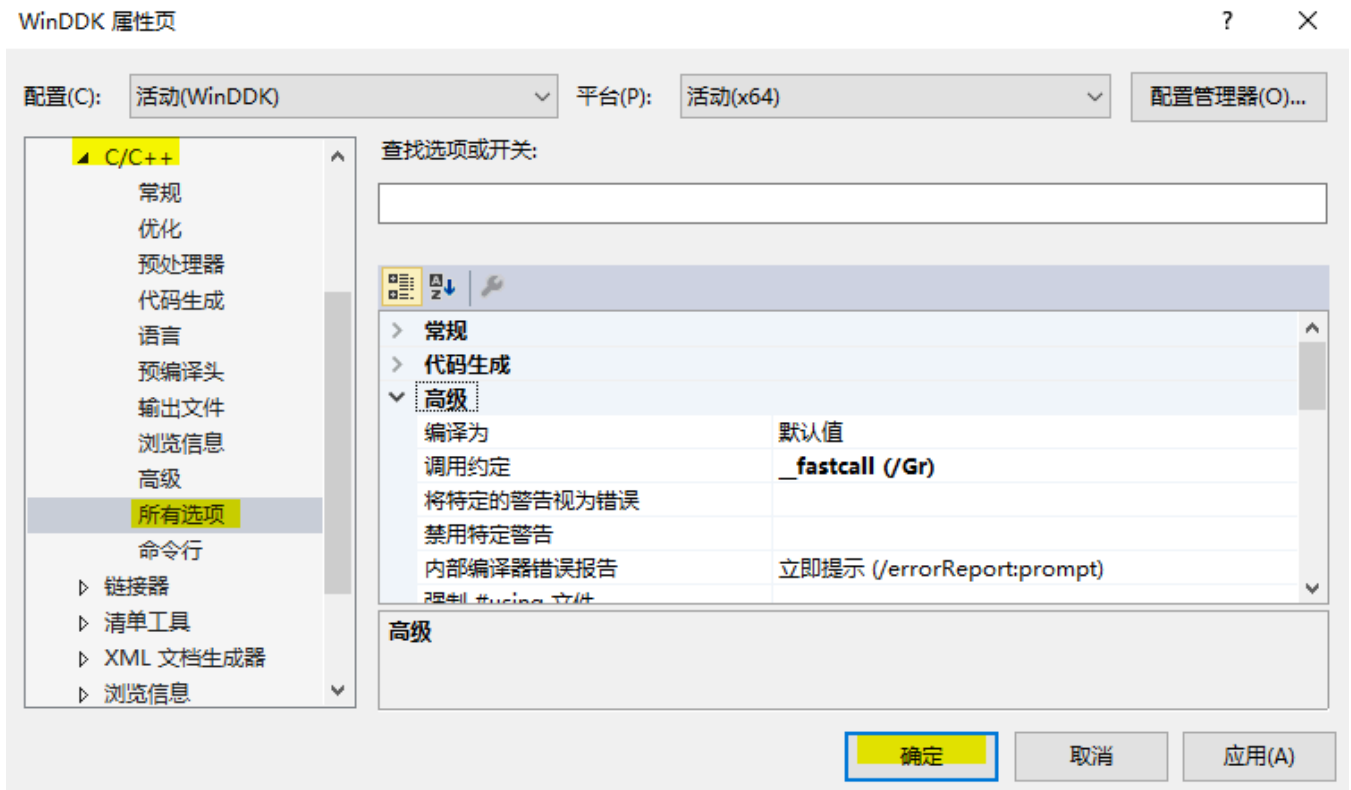
当如上文件配置完成后，最终效果如下图所示；



6.配置C/C++优化选项, 在配置属性中找到 C/C++-所有选项 并依次修改下方几个关键位置。

安全检查	禁用安全检查 (/GS-)
将警告视为错误	否 (/WX-)
警告等级	关闭所有警告
启用C++异常	否
调用约定	__fastcall (/Gr)
优化	已禁用 (/Od)
运行库	多线程 (/MT)
预处理器定义 (PreprocessorDefinitions)	_AMD64_;_DDK_;_WIN32_WINNT=0x0501;WINVER=0x0501;_NDEBUG;DBG=0;%

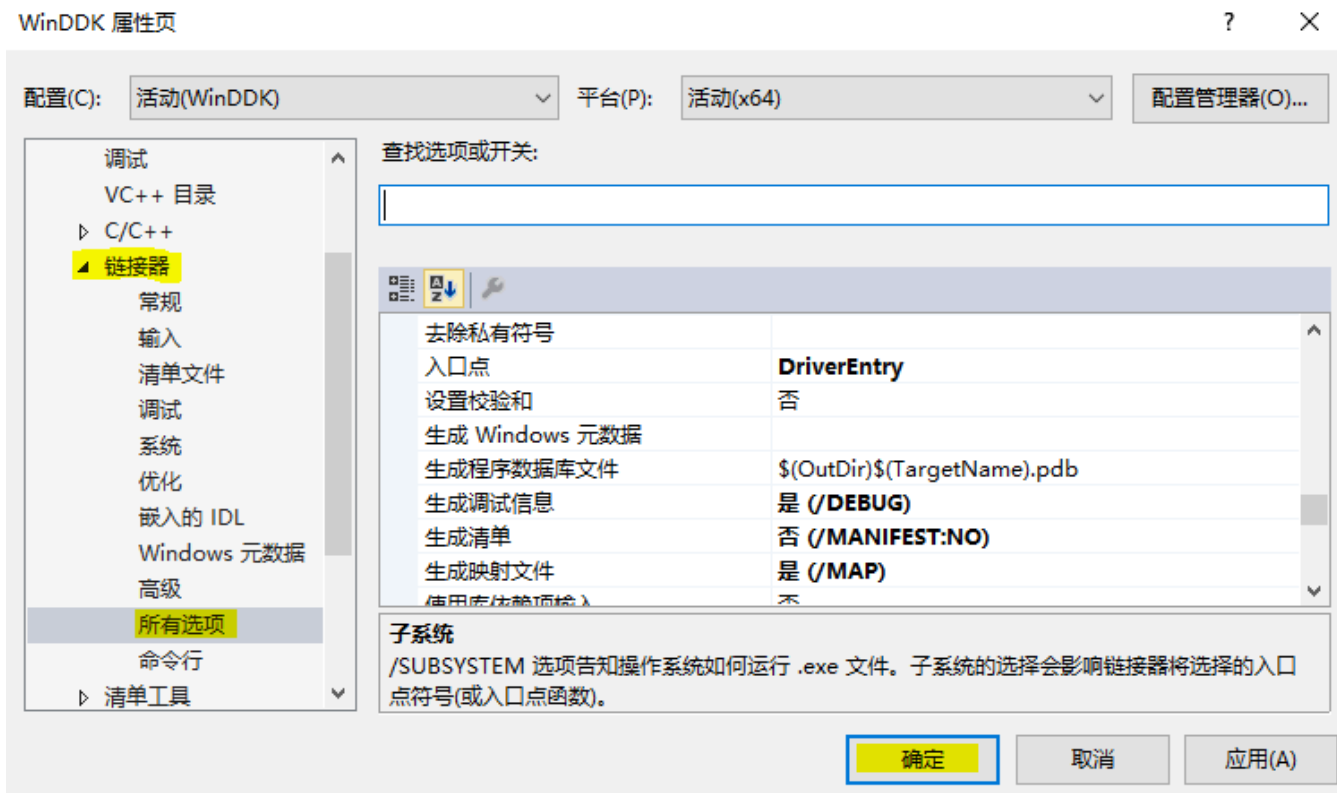
当如上文件配置完成后, 最终效果如下图所示;



7.配置连接器选项，选择 连接器-所有选项 依次修改下方几个关键位置。

附加选项	<code>/IGNORE:4078 /safeseh:no</code>
附加依赖项	<code>ntoskrnl.lib;ndis.lib;Hal.lib;wdm.lib;wdmsec.lib;wmilib.lib</code>
固定基址	此处需要清空
忽略所有默认库	是 (<code>/NODEFAULTLIB</code>)
启用增量链接	否 (<code>/INCREMENTAL:NO</code>)
驱动程序	驱动程序 (<code>/Driver</code>)
入口点	<code>DriverEntry</code>
生成清单	否 (<code>/MANIFEST:NO</code>)
生成调试信息	是 (<code>/DEBUG</code>)
生成映射文件	是 (<code>/MAP</code>)
数据执行保护	是 (<code>/NXCOMPAT</code>)
随机基址	此处需要清空
子系统	本机 (<code>/SUBSYSTEM:NATIVE</code>)

当如上文件配置完成后，最终效果如下图所示：



8.上方的配置已经基本完成了，接着我们编写一段驱动初始化代码，然后按下 F7 即可完成驱动的编译。

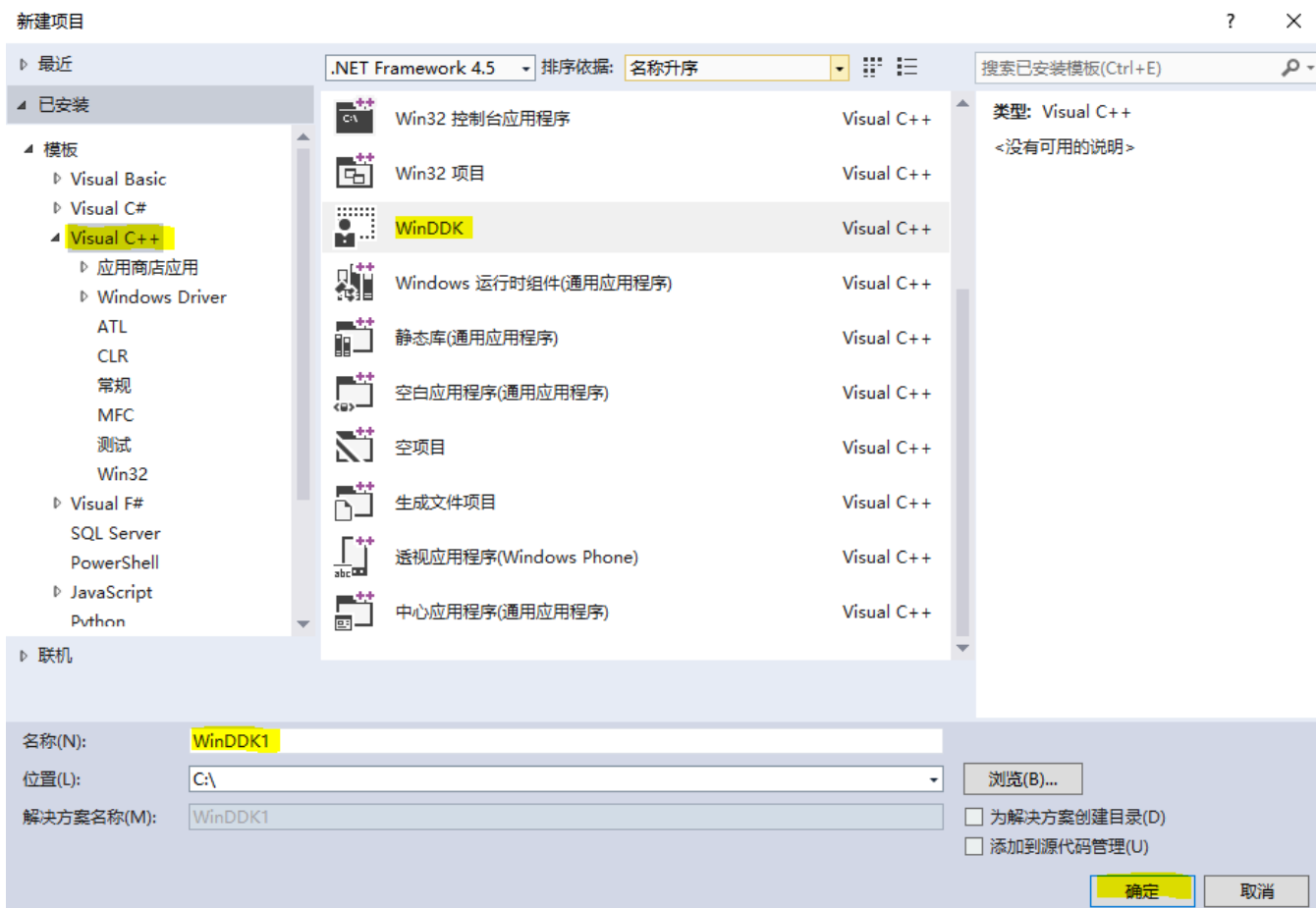
```
#include <ntifs.h>

// 卸载驱动
NTSTATUS UnDriver(PDRIVER_OBJECT driver)
{
    DbgPrint("Uninstall Driver Is OK \n");
    return STATUS_SUCCESS;
}

// 驱动入口地址
NTSTATUS DriverEntry(IN PDRIVER_OBJECT Driver, PUNICODE_STRING RegistryPath)
{
    DbgPrint("Hello LyShark \n");
    Driver->DriverUnload = UnDriver;
    return STATUS_SUCCESS;
}
```

9.最后生成一个驱动开发模板，依次选择 文件-导出模板-项目模板-下一步-完成 即可完成模板的导出，此时关闭VS工具并再次打开，就能直接使用我们的模板来开发驱动了，当用户需要使用时，不需要每次都配置。

- 模板位置：C:\Users\admin\Documents\Visual Studio 2013\My Exported Templates



读者也应注意，如果用户通过模板创建驱动开发项目则需要手动在配置菜单中切换到 WinDDK 选项的 x64 模式下。

