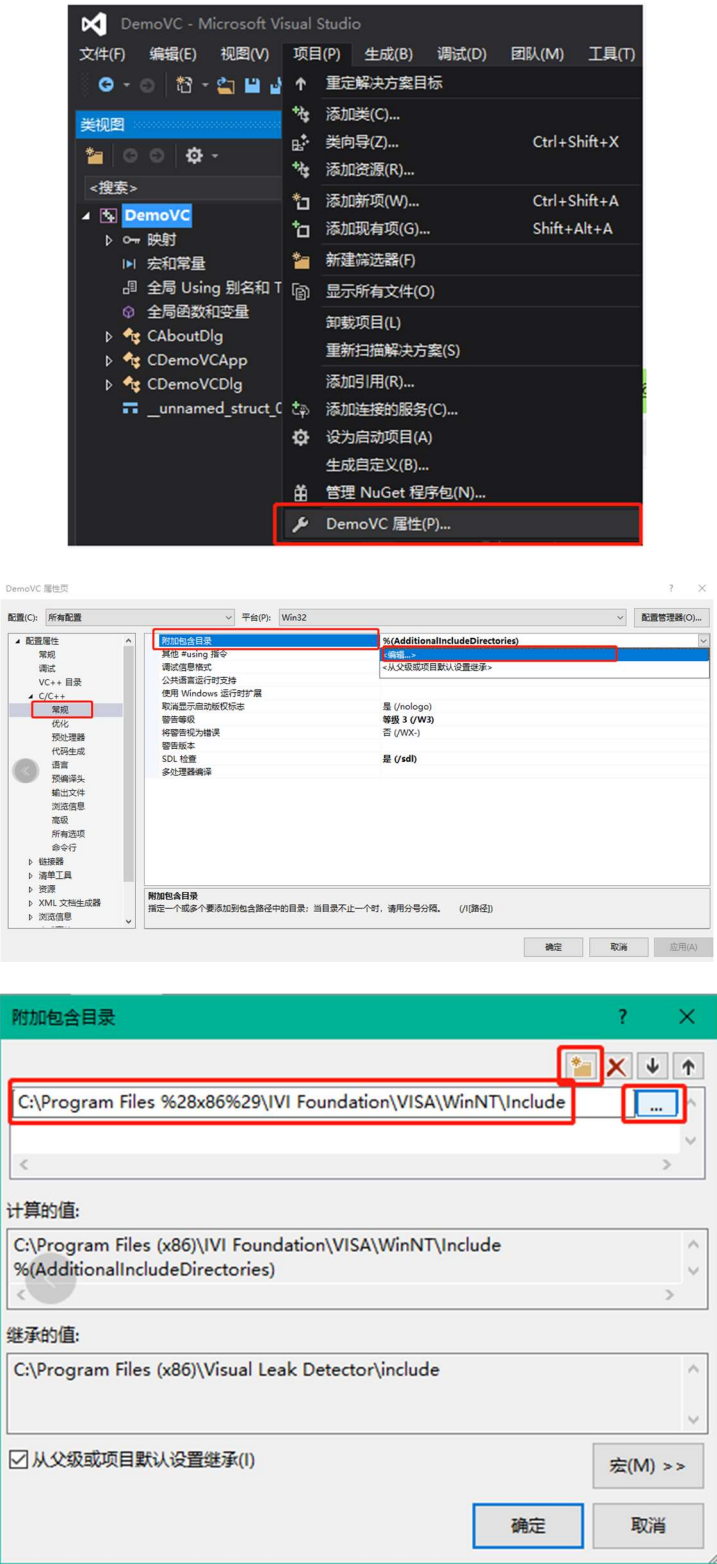


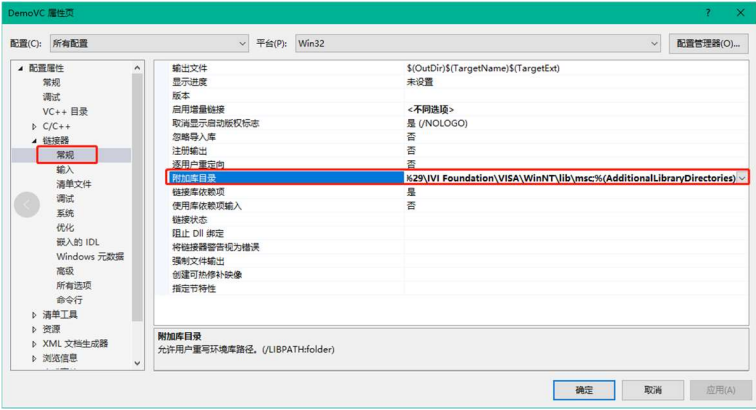
# Visual C++编程实例

本例使用的程序：Microsoft Visual C++ 2015

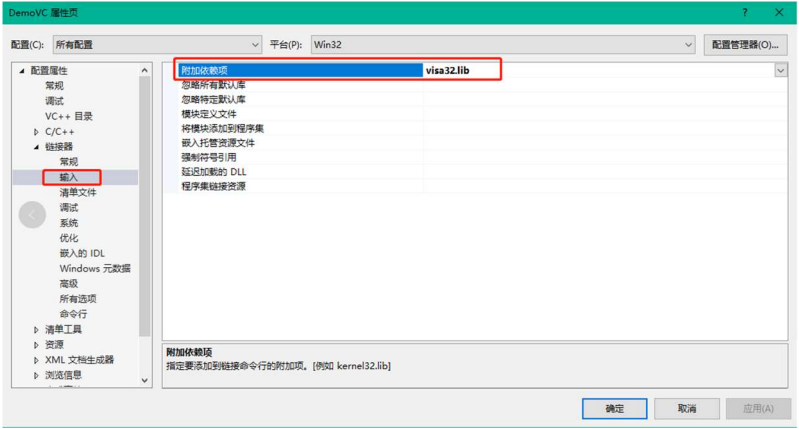
本例实现的功能：查找仪器地址、连接仪器、发送命令并读取返回值。

- 1. 运行Microsoft Visual C++ 2015，新建一个基于对话框的MFC工程，命名为DemoVC。
- 2. 点击菜单“项目->属性”，在弹出界面的“属性”选项卡中添加Include和Lib路径。  
注意： 此处添加的两个路径与您计算机上NI-VISA或Keysight-VISA的安装路径相关。此处默认为安装在C:\Program Files(x86)\IVI Foundation\VISA路径下。

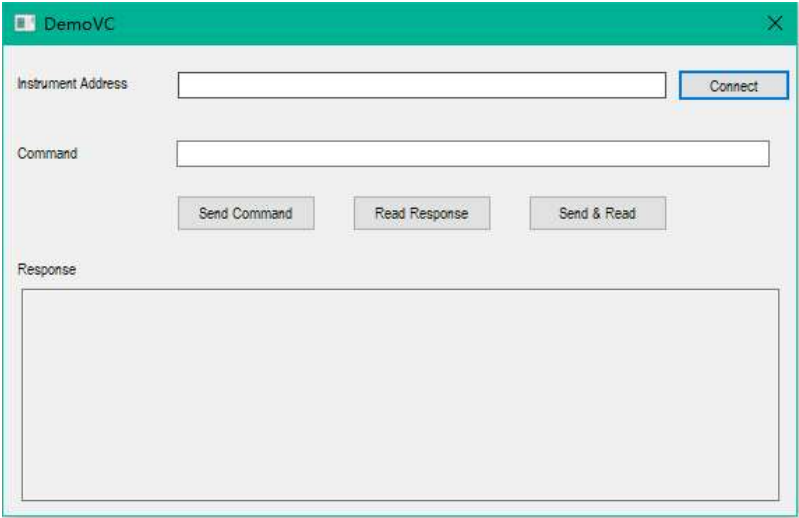




3. 在属性页界面，手动添加“visa32.lib” 文件

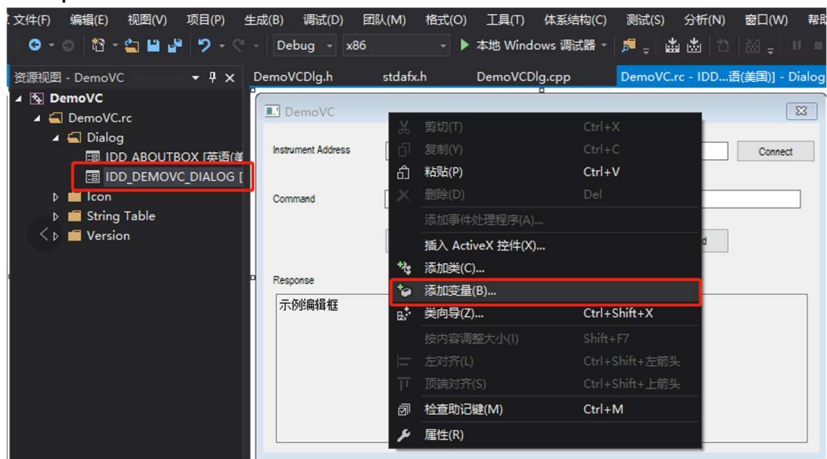


4. 添加Text、Edit和Button控件，布局如下图所示，其中Instrument Address、Command和Response是Text控件，Connect、Send Command、Read Response和Send & Read是Button控件，3个Edit均为Edit控件，且第3个Edit控件为只读（Read-only）控件。



5. 在头文件中手动添加3个CString 类型变量：
- 仪器地址 CString m\_strInstrAddr
  - 命令 CString m\_strCommand
  - 返回值 CString m\_strResult
6. 在资源视图中，右击对话框，手动添加3个控件变量：
- CEdit m\_editInstrAddr;
  - CEdit m\_editCommand;

CEdit m\_editResponse;



7. 封装VISA的读和写操作。
- 1) 对VISA的写操作进行封装便于操作。

```
bool CDemoVCDlg::InstrWrite(CString strAddr, CString strContent)
{
    USES_CONVERSION;

    ViSession defaultRM, instr;
    ViStatus status;
    ViUInt32 retCount;

    bool bWriteOK = false;
    CString str;
    strContent += _T("\n");
    char * SendBuf = T2A(strContent);
    //open a VISA resource
    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        AfxMessageBox(_T("No VISA resource was opened!"));
        return false;
    }

    status = viOpen(defaultRM, T2A(strAddr), VI_NULL, VI_NULL, &instr);

    //Write command to the instrument
    status = viWrite(instr, (unsigned char *)SendBuf, strlen(SendBuf), &retCount);
    if (VI_SUCCESS == status)
    {
        bWriteOK = true;
    }
    //Close the system
    status = viClose(instr);
    status = viClose(defaultRM);
    return bWriteOK;
}
```

- 2) 对VISA的读操作进行封装便于操作。

```

bool CDemoVCDlg::InstrRead(CString strAddr, CString *pstrResult)
{
    USES_CONVERSION;
    ViSession defaultRM, instr;
    ViStatus status;
    ViUInt32 retCount;
    unsigned char RecBuf[MAX_REC_SIZE];
    bool bReadOK = false;
    CString str;

    memset(RecBuf, 0, MAX_REC_SIZE);
    //Open a VISA resource
    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        // Error Initializing VISA...exiting
        AfxMessageBox(_T("No VISA resource was opened!"));
        return false;
    }
    //Open the instrument
    status = viOpen(defaultRM, T2A(strAddr), VI_NULL, VI_NULL, &instr);
    //Read from the instrument
    status = viRead(instr, RecBuf, MAX_REC_SIZE, &retCount);
    if (status == VI_SUCCESS || status == VI_SUCCESS_TERM_CHAR || status == VI_SUCCESS_MAX_CNT)
        bReadOK = true;
    //close the system
    status = viClose(instr);
    status = viClose(defaultRM);
    (*pstrResult).Format(_T("%s"), A2T((char*)RecBuf));
    return bReadOK;
}

```

## 8. 增加控件消息响应代码。

### 1) 连接仪器

```

void CDemoVCDlg::OnBnClickedBtnConnect()
{
    // TODO: 在此添加控件通知处理程序代码
    USES_CONVERSION;
    ViStatus status;
    ViSession defaultRM;
    ViString expr = "?*";
    ViFindList findList = new unsigned long;
    ViPUInt32 retcnt = new unsigned long;
    ViChar instrDesc[1000];
    CString strSrc = L"";
    CString strInstr = L"";
    unsigned long i = 0;
    bool bFind = false;

    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        // Error Initializing VISA...exiting
        AfxMessageBox(L"No VISA instrument was opened !");
        return;
    }

    memset(instrDesc, 0, 1000);

    //Find resource
    status = viFindRsrc(defaultRM, expr, findList, retcnt, instrDesc);
    for (i = 0; i < (*retcnt); i++)
    {
        //Get instrument name
        strSrc.Format(L"%s", A2T(instrDesc));
        InstrWrite(strSrc, L"*IDN?");
        ::Sleep(200);
        InstrRead(strSrc, &strInstr);

        //If the instrument(resource) belongs to the XXXX series then
        strInstr.MakeUpper();
        if (strInstr.Find(L"HDM") >= 0)
        {
            bFind = true;
            m_strInstrAddr = strSrc;
            break;
        }
        //Find next resource
        status = viFindNext(*findList, instrDesc);
    }
    if (bFind == false)
    {
        AfxMessageBox(L"Didn't find any device!");
        m_editInstrAddr.SetWindowText(_T(""));
    }
    else
    {
        m_editInstrAddr.SetWindowText(m_strInstrAddr);
    }
}

```

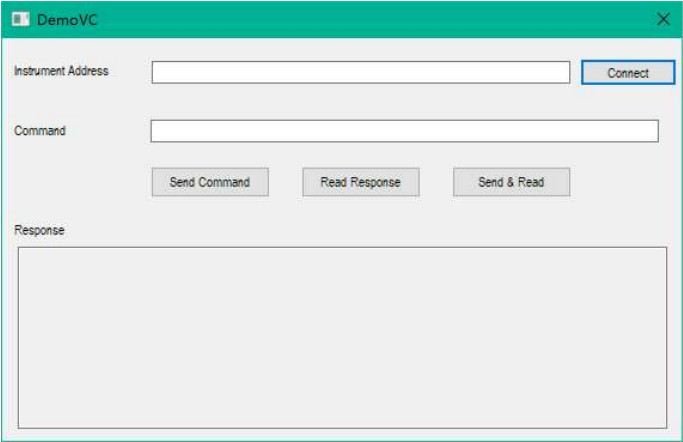
### 2) 写操作

```
void CDemoVCDlg::OnBnClickedBtnSendCommand()
{
    // TODO: 在此添加控件通知处理程序代码
    if (m_strInstrAddr.IsEmpty())
    {
        AfxMessageBox(L"Please connect the instrument first!");
    }
    m_editCommand.GetWindowText(m_strCommand);
    if (!m_strCommand.IsEmpty())
    {
        InstrWrite(m_strInstrAddr, m_strCommand);
    }
    m_strResponse.Empty();
}
```

3) 读操作

```
void CDemoVCDlg::OnBnClickedBtnReadResponse()
{
    // TODO: 在此添加控件通知处理程序代码
    if (m_strInstrAddr.IsEmpty())
    {
        AfxMessageBox(L"Please connect the instrument first!");
    }
    InstrRead(m_strInstrAddr, &m_strResponse);
    m_editResponse.SetWindowText(m_strResponse);
}
```

9. 运行程序将进入如下操作界面。



执行以下步骤：

- 1) 点击Connect按钮查找并连接设备；
- 2) 在Command编辑框中输入命令，如“\*IDN?”；
- 3) 点击Send Command按钮发送命令；
- 4) 点击Read Response按钮读取返回值。

运行结果如下图所示：

