

LabVIEW与VB混合编程技术

刘广敏

(中北大学自动控制系 山西 太原 030051)

【摘要】针对LabVIEW与VB混合编程问题,提出了两种方法——VB直接调用LabVIEW生成的可执行文件和VB调用LabVIEW生成的动态链接库文件,并以具体的例子说明了两种方法的实现过程。结果表明这两种方法能有效地实现LabVIEW与VB混合编程。

【关键词】 LabVIEW (虚拟电子仪器) VB 混合编程 可执行文件 动态链接库文件

【中图分类号】 TP313 【文献标识码】 B 【文章编号】 1003-773X(2006)01-0092-03

引言

LabVIEW(Laboratory virtual instrument engineering workbench)是一种图形化虚拟仪器集成开发环境。所谓的虚拟仪器就是在以计算机为核心的硬件平台上,其功能由用户设计和定义,具有虚拟面板,其测试功能由测试软件实现的一种计算机仪器系统。LabVIEW最突出的特点就是能够实现仪器虚拟化和编程图形化,可以用软件来代替显示中的很多硬件,如示波器,波形发生器,各种控制开关等,也可以由这些虚拟子仪器(VI)再搭建高一层的仪器,这些虚拟仪器放在前面板上可以实现和实际的仪器相同的功能。尽管LabVIEW的功能强大,但是也难免有一些不足之处,不过通过和其他编程工具混合编程比如VB可以拟补这些不足。现在就针对LabVIEW与VB混合编程问题,提出两种实现的方法——VB直接调用LabVIEW生成的可执行文件和VB调用LabVIEW生成的动态链接库文件。

1 在VB中调用LabVIEW生成的可执行文件

1.1 LabVIEW中VI的设置

在编译之前先要在LabVIEW VI中做一些准备工作。

(1)打开要打包的VI,在后板中程序结束的地方调用Application Control\Quit LabVIEW.VI,使编译后执行文件在结束时能够退到Windows下。

(2)设置要生成的可执行文件的属性。在VI的右上角图标上点击右键,在下拉菜单中选择VI Properties...在弹出的对话框中的Category的下拉菜单中选择要配置参数进行配置。

1.2 把LabVIEW VI编译成可执行文件

属性设置完成之后,保存并关闭,就可以打开一个新的VI进行编译了。

打开一个新的VI进行编译。此时LabVIEW中必须

安装了application builder软件包,如果安装了就会在Tools菜单的下拉菜单中看到Build Application or Shared Library选项。

在弹出来的对话框中在Build Target项中选择Application(EXE)选项,表示生成的是可执行文件,在Installer Setting标签中可以设置生成Install(安装)。编译完成以后在相应的保存目录中就可以生成可执行文件的同时,生成了一个lvdaq.dll动态连接库文件。

1.3 在VB编程平台中调用可执行文件

在VB中可以通过shell函数来调用可执行程序。这种调用可执行文件的方式的特点是,在VB平台中可以看到LabVIEW所作的可执行文件的界面,并且可以在LabVIEW中一样在前面板中设置参数。调用LabVIEW的计算机中如果没有安装LabVIEW,则必须安装NI公司的Run time Engine软件包。

1.4 应用举例

(1)设计好的LabVIEW VI为Myfile.vi,为一个多通道数据采集VI,在VI结束的地方调用Application Control\Quit LabVIEW.vi,VI前面板如图1所示;

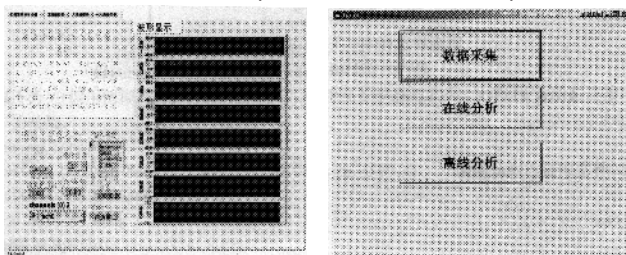


图1 LabVIEW 执行文件界面图

(2)设置要生成的可执行文件的属性;

(3)编译VI,生成Application .exe文件的同时,生成了一个lvdaq.dll动态连接库文件;

(4)在VB中设计调用LabVIEW数据采集VI程序界面如图2所示,调用Application .exe文件,需要把Appli-

ation .exe文件Ivdaq.dll动态连接库文件同时拷贝到VB的执行目录下。

调用代码如下：

```

' 执行file.exe文件
' lngTask = Shell("Application.exe", vbNormalFocus)
lngTask = Shell("Application.exe", vbHide) ' 调用可执行文件 '
lngHandle = OpenProcess(1, False, lngTask)
' 等待进程执行完毕
lngret = WaitForSingleObject(lngHandle, 0)
lngret = CloseHandle(lngHandle)

```

在VB程序中点击数据采集按钮时调用Application.exe会出现LabVIEW的界面,如图1所示,效果和直接在LabVIEW中执行的效果一样,可以直接在调用的前面板中设置参数。

2 VB调用LabVIEW生成的动态连接库文件

如果想在VB编程平台中调用LabVIEW,而又不想出现LabVIEW的界面的话,这就需要用到这种调用方式了。

2.1 VB和LabVIEW之间的参数传递

在进行调用动态连接库文件之前,同样需要在LabVIEW进行一些必要的设置,主要是需要设置生成的动态连接库中的函数的参数,这是通过设置要编译的VI的输入输出参数来实现的。但是考虑到在调用动态连接库的时候参数传递比较复杂,因而就不用参数传递,干脆没有参数。解决的办法有两种:(1)在LabVIEW的VI中用到的参数为固定值,用鼠标右键单击设置参数的子VI在下拉菜单中选中Data Operation,再选定Make Current Value Default(目前数值为默认值),这种方法要求所设定的参数固定不变就可以实现预期的功能;(2)利用LabVIEW与数据库的接口,把要改变的参数放到数据库中,LabVIEW程序不断动态地扫描对应的数据库文件,这样就可以改变参数,甚至可以动态地改变参数。这样就解决了参数传递的问题。

2.2 把LabVIEW VI编译成动态连接库文件

在编译时会弹出一个要求设置函数原型的对话框,可以看到生成的动态连接库中的函数名称,可以添加参数,可以看到参数的数据类型,用Build打包一下就可以了。这样生成一个你设定名称的动态连接库文件和一个自动生成的系统Ivdaq.dll文件,在VB中调用这个动态连接库文件时,要把这个动态连接库文件也拷贝到VB程序运行目录下。

以这种方法调用LabVIEW程序和调用执行文件的效果是一样的,区别之处在于这种方式调用时在应用程序中不会出现LabVIEW程序的界面。

2.3 在VB中调用动态连接库文件

在VB中可以用Declare语句声明对动态链接库(DLL)中的外部过程的引用,在声明了这个动态连接库后,就可以直接调用这个动态连接库中的函数,注意没有参数传递。调用LabVIEW的计算机中如果没有安装LabVIEW,则必须安装NI公司的Run time Engine软件包。

2.4 应用举例

在VB中调用动态连接库文件的总体流程如图3所示,设计的程序界面如图2所示,为了区别两种调用的不同,与1.4中用同样的Myfile.vi,区别是编译生成DLL文件。



图 3 在 VB 中调用动态连接库文件的总体流程框图

以下为VB调用例程,其中SharedLib.dll为打包生成的动态连接库,MyFile为动态连接库中的函数。

```

Option Explicit
Private Declare Function MyFile Lib "SharedLib.dll" () ' 调用动态连接库文件
Private Sub Command1_Click()
Dim dblC As Long
dblC = MyFile()
End Sub

```

把生成的两个动态连接库文件都拷贝到VB运行目录下,运行VB程序,点击数据采集按钮,调用LabVIEW VI生成的动态连接库文件,不会出现LabVIEW的界面(见图2)。在调用过程中不用声明Ivdaq.dll文件,在执行MyFile()函数时自动连接。

4 结束语

可以在VB中通过可执行文件和动态连接库文件调用LabVIEW程序,前者调用时VB程序中会出现LabVIEW程序的界面,后者不会出现LabVIEW程序的界面,两种调用方法中执行的效果是一样的。这样就实现了LabVIEW和VB的结合,可以相互弥补不足,在VB中就可以充分利用LabVIEW做数据采集的优势,又可以利用其本身的灵活优势进行数据分析,故障诊断等,达到编程的最佳效果。

参考文献

- [1] 杨乐平,李海涛,赵勇,等.LabVIEW高级程序设计.北京:清华大学出版社,2003.
- [2] 刘君华,郭会军,等.基于LabVIEW的虚拟仪器设计.北京:电子工业出版社,2003.
- [3] 史惠康,曹琦,戴剑彬.Visual Basic 5.0实用编程技术.中国水利水电出版社,1998.

(收稿日期:2005-07-20)
(下转第95页)

最基本的是以链表进行存储,通过遍历和多关键字遍历来进行的,其中的基本的数据结构有以下两种:

3.1 双链表数据结构

双向链表中两条方向不同的链,即每个结点中除next域存放后继结点地址外,还增加一个指向其直接前趋的指针域prior。(见图2)

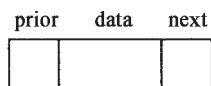


图 2 结点结构

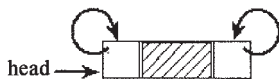


图 3 空的双循环链表

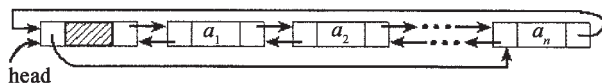


图 4 非空的双循环链表

双链表类型描述:

```
typedef struct dlstnode{
    DataType data;
    struct dlstnode *prior,*next;
}DLstNode;
typedef DLstNode *DLinkedList;
DLinkedList head;
```

注意 (1)双链表由头指针head唯一确定的。(2)带头结点的双链表的某些运算变得方便。(3)将头结点和尾结点链接起来,为双向循环链表(见图3、图4)。

3.2 单链表数据结构^[4]

由于我们常常只注重结点间的逻辑顺序,不关心每个结点的实际位置,可以用箭头来表示链域中的指针,线性表(bat,cat,fat,hat,jat,mat)的单链表就可

以表示为图5形式。



图 5 单链表图示法

单链表类型描述

```
typedef char DataType; //假设结点的数据域类型为字符
typedef struct node{ //结点类型定义
    DataType data; //结点的数据域
    struct node *next; //结点的指针域
}ListNode;
typedef ListNode *LinkedList;
ListNode *p;
LinkedList head;
```

注意 (1)LinkedList和ListNode *是不同名字的一个指针类型(命名的不同是为了概念上更明确)。(2)LinkedList类型的指针变量head表示它是单链表的头指针。(3)ListNode *类型的指针变量p表示它是指向某一结点的指针。

4 结束语

通过以上的分析,对于中国铸造网的电子商务系统经过了大量的数据分析后和实践的调试,最后终于成功的完成了该网站电子商务系统的开发。

参考文献

[1] 赵吉兴.电子商务基础.2003:1-2.
[2] 曹桂琴,郭芳.数据结构学习指导.2003:1-5.
[3] 丁贵广.ASP编程基础与实例.2002:233-248.
[4] 张绍民,李淑华.数据结构教程(C语言版).1999:8-16.

(收稿日期:2005-09-08)

China Founding Web Electronic Commerce and Its Data Structure Analysis

Chen Jianfeng Ren Yifeng

[Abstract] By analysis of electronic commerce system design plan and data structure, fulfilled EC system design of the web station.

[Key words] Electronic commerce(EC) Data structure Internet

(上接第 93 页)

Mixed Programming Technology between LabVIEW and VB

Liu Guangmin

[Abstract] In order to solve the problem of implementation of mixed programming between LabVIEW and VB, both methods are introduced: directly invoke LabVIEW to generate executive files and applying VB invoke LabVIEW to generate DLL files, whose implementing procedure are described by detailed examples. The result shows that both method can effectively implement the mixed programming between LabVIEW and VB.

[Key words] LabVIEW VI (Virtual electronic instrument) VB Mixed programming Executive files DLL