

Visual C++, LabVIEW, LabWindows/CVI 与 MATLAB 接口技术的研究

王 华, 袁中凡

(四川大学 制造科学与工程学院, 四川 成都 610065)

摘 要: 详细阐述了当今最为流行的面向对象的可视化编程工具 Visual C++、虚拟仪器设计平台 LabVIEW、LabWindows/CVI 与计算及绘图功能强大的 MATLAB 之间的接口技术, 从而进行相互融合, 更好地体现各自的优越性。实例证明了接口技术的有效性和可行性。

关键词: VC++; LabVIEW; LabWindows/CVI; MATLAB; 混合编程

中图分类号: TP311 **文献标识码:** A **文章编号:** 1001-3695(2007)02-0281-05

Research on Interfacing Technology between Visual C++, LabVIEW, LabWindows/CVI and MATLAB

WANG Hua, YUAN Zhong-fan

(College of Manufacturing Science & Engineering, Sichuan University, Chengdu Sichuan 610065, China)

Abstract: An interfacing technology between Visual C++, LabVIEW, LabWindows/CVI and MATLAB is described in this article. Visual C++ is the most popular, OO, visual programming tool, and LabVIEW, LabWindows/CVI are two kinds of most excellent developing platform for virtual instrument at present, while MATLAB has powerful functions in calculating and drawing. Only by combining MATLAB with Visual C++, LabVIEW, LabWindows/CVI respectively it can make best of their own advantages. The examples can demonstrate the effectiveness and feasibility of the interfacing technology.

Key words: VC++; LabVIEW; LabWindow/CVI; MATLAB; Mixed Programming

MATLAB 是 MathWorks 公司开发的演算纸式的程序设计语言, 是一个跨平台的科学计算环境, 其具有强大的矩阵列运算、数值分析、信号处理、系统识别、图像语音处理等功能。应用 MATLAB 软件可以很方便地实现许多复杂的数学算法, 而这些复杂的算法在其他开发环境中需要大量代码才能实现。但 MATLAB 也有不足之处, 如在数据输入、硬件控制等方面相对较弱, 程序执行速度也相对较慢且一般需要在 MATLAB 环境下运行。所以在解决实际工程问题时往往将 MATLAB 与其他高级语言混合编程, 将 MATLAB 强大的数学功能服务于其他开发环境。Microsoft Visual C++ (VC++) 是一种面向对象的可视化高级编程工具, 它是微软公司推出的支持 32 位操作系统的开发工具, 提供了对 Microsoft 最新技术的集成, 以及一个可视化的编程模式, 是 Windows 平台上理想的开发工具; 但 VC++ 数学计算能力不强, 难以实现复杂的数学算法。LabVIEW 和 LabWindows/CVI 是目前最有影响的虚拟仪器开发平台, 它们将计算机软件设计平台与数据的采集、分析、处理及结果表达等仪器专业工具有机地结合起来, 作为交互式的高效率的集成开发环境; 但 LabVIEW 和 LabWindows/CVI 仍然无法满足复杂的数值计算和分析需求。本文结合上述各种开发环境的优缺点, 阐述了 VC++, LabVIEW, LabWindows/CVI 与 MATLAB 混合编程的接口技术, 并以实例成功证明了混合编程

接口技术的优越性。

1 Visual C++ 与 MATLAB 接口通信技术

VC++ 可开发基于 MFC 以及基于 Windows API 的 Win32 程序, 其特点是代码效率高、执行速度快, 可开发出风格多样的界面, 其不足是开发数字图像处理算法时代码编写工作量很大。如果将 MATLAB 强大的数值处理功能融合于 VC++ 开发环境, 将为工程项目开发提供更有力的技术支持。VC++ 和 MATLAB 的接口有四种途径。

1.1 MATLAB 自带编译器

通过 MATLAB 自带的编译器将 M 文件转换为 C, C++ 等各种不同类型的源代码, 并在此基础上根据应用需要生成 MEX 文件和独立可执行应用程序 (Stand-alone Applications), 大大提高了程序运行速度及代码执行效率。尤其是独立可执行应用程序可以完全脱离 MATLAB 环境独立运行, 大大扩展了程序的应用范围。但是, MATLAB 编译器不能编译脚本 M 文件、有 Object 的 M 文件、用 Input 或 Eval 语句操作工作空间变量的 M 文件, 同时 MATLAB 编译器也不能编译 MATLAB 的内建函数。

1.2 在 VC++ 中调用 MATLAB 数学库函数

在 VC++ 的集成环境下进行程序设计, MATLAB C++ 数学库最大的应用价值就在于脱离 MATLAB 环境运行。VC++

调用 MATLAB 数学库和图形库有几个明显的优点:执行速度快、内存需求小,可以发布给没有 MATLAB 的用户使用。但它也有几个明显的缺点:不能调用图形句柄系统的函数;不能调用 MATLAB 工具箱中的函数;MATLAB 中的一些方法在 C/C++ 中得不到支持。使用 MATLAB 数学函数库的环境设置步骤如下(假设 MATLAB 安装在 E 盘根目录下):

(1) 在 VC++ 开发环境中由 Tools->Options...->Directories 在“Show directories for”栏中首先选择 Includefiles, 再在“Show directories for”中加入 E:\MATLAB\extern\include 和 E:\MATLAB\extern\include\cpp, 然后在“Show directories for”栏中选择 Library files, 再在“Directories”中分别加入 E:\MATLAB\bin, E:\MATLAB\extern\lib。

(2) 在 VC++ 开发环境中由 Project->Settings...->C/C++ 在 C/C++ 的“Category”栏中选择 Preprocessor, 然后在出现的“Preprocessor definition”栏中加入 MSVC, MSWIND。

(3) 设置运行时动态链接库。由 VC++ 开发环境中的 Project->Settings...->C/C++, 在 C/C++ 的“Category”栏中选择 Code Generation, 然后在出现的“User run-time library”栏中选择 Multithreaded DLL。

(4) 在完成(1)~(3)后就可以建立工程了。在源程序中必须加上#include "MATLAB.hpp", 但在工程编译前要把 libmx.lib, libmatlb.lib, libmatpm.lib 和 libmmfile.lib 等文件添加到工程中。其方法是在由 VC++ 开发环境中 Project->Add to Project->Files...弹出的“Insert Files to Project”对话框中, 从目录 E:\MATLAB\extern\lib 中选择上述文件后按“OK”按钮即可。

1.3 用 Matcom 软件实现 VC++ 与 MATLAB 的接口通信

Matcom 是 MathTools 公司推出的一个能将 M 文件转换成相同功能 C++ 代码的工具。相比 MATLAB 自带的编译器 Compiler, Matcom 可生成不需 MATLAB 环境支持的 EXE 文件, 从而脱离 MATLAB 环境对 MATLAB 函数和过程进行有效调用。Matcom 的应用主要有三个方面:①利用 Matcom 生成独立执行的 EXE 文件;②利用 Matcom 生成动态链接库(DLL)文件;③利用 Matcom 提供的 Visual Matcom 功能使 Matcom 工具条显示在 VC++ 集成环境中。启动 VC++ 6.0, 由 Tools->Customize 选择“Add-ins and Macro Files”, 点击“Browse”, 选择文件类型为 lib, 在 Matcom\bin 下找到 MVCIDE.dll 文件并点击“OK”, 这样 Matcom 工具条就出现在 VC++ 集成环境下了, 可在 VC++ 中直接编译 M 文件。

使用 Matcom 时的 Visual VC++ 环境设置如下:

(1) 在 VC++ 开发环境中由 Tools->Options...->Directories->Show Directories for, 在 Include files 和 Library files 两项中分别加入 Matcom 下的 lib 目录, 如 D:\MATCOM45\lib。

(2) 在 VC++ 开发环境中由 Project->Add To Project->Files 将 D:\MATCOM45\lib\v4501.lib 加入工程, 在 VC++ 源文件的头文件处应包含 matlib.h。VC++ 源文件中调用由 M 文件转换的 VC++ 代码之前应用 initM(Matcom_VERSION) 函数来初始化类库调用, 并用 exitM() 函数来结束类库调用。因此在 Main.cpp 文件中加入下列代码:

```
int main()
{
    initM(Matcom_VERSION);
    ...//转换成功的C++代码
    exitM();
    return 0;
}
```

1.4 在 VC++ 中调用 MATLAB 引擎实现接口通信

与其他三种接口方式相比, MATLAB 引擎所提供的 MATLAB 功能支持是最全面的。通过引擎, 应用程序实际上会打开一个新的 MATLAB 进程, 可以控制它完成任何计算和绘图操作, 对所有的数据结构都提供 100% 的支持。MATLAB 引擎采用客户机+服务器模式, 其实质是用 ActiveX 自动化 DDE 技术来实现 VC++ 与 MATLAB 之间的数据通信。在具体应用程序中, VC++ 作为客户端负责用户接口并提出数据请求, 而 MATLAB 则属于服务端, 向客户端提供数据服务。MATLAB 提供了以下几个引擎函数与客户端进行交互:

- (1) engOpen, 启动 MATLAB 引擎;
- (2) engClose, 关闭 MATLAB 引擎;
- (3) engEvalString, 向 MATLAB 发送一个字符串, 让 MATLAB 执行;
- (4) engOutputBuffer, 确定存放 MATLAB 输出结果缓冲区域;
- (5) engGetArray, 从 MATLAB 工作空间获取一个变量;
- (6) engPutArray, 将 mxArray 结构体变量输送到 MATLAB 工作空间。

在 VC++ 集成开发环境中建立和调试 MATLAB 引擎的基本步骤如下:

(1) 启动 VC++ 6.0, 建立 MFC AppWizard(exe) 类型工程。

(2) 项目工程创建完毕后, 在 VC++ 开发环境中由 Tools->Options...->Directories->Show Directories for, 在 Include Files 和 Library Files 两项中分别加入 MATLAB 根目录\extern\include, MATLAB 根目录\extern\lib\win32\microsoft\msvc60, 然后选择“OK”按钮。

(3) 在 VC++ 开发环境中, 选择下拉式菜单 Project 中的菜单项 Settings, 打开工程设置属性页, 选择 Link 页面, 在“Object/Library modules”编辑框中添加文件名 libmx.lib。

步骤(2)只需设置一次, 而步骤(3)则每个工程都要单独设定, 因为每个工程使用的 MATLAB 函数可能不一样。

经上述三个步骤, 用户便完成了对 VC++ 6.0 开发环境 MATLAB 引擎函数的调用、设置以及 MATLAB 引擎程序的建立, 经过编译、链接后便可执行。

1.5 应用举例

前面主要基于 VC++ 与 MATLAB 各自的优缺点, 提出了将它们混合编程接口的四种途径。与其他三种方法相比较, 用 MATLAB 引擎实现接口节省了大量的系统资源。该应用程序整体性能好, 引擎提供的 MATLAB 支持功能最全面, 可充分发挥 MATLAB 的数值计算和图形处理的功能, 对所有数据结构提供完全支持。在实际工程问题中, 用户可根据具体情况选择适合自己的接口技术。笔者针对每一种接口技术, 一一在计算机上进行了调试、验证。事实证明上面的四种 VC++ 与 MATLAB 的接口技术都完全正确、有效。限于篇幅, 在此只对用 MATLAB 引擎实现接口详细举例。

MATLAB 在信号处理和图形显示方面具有 VC++ 无法比

拟的优势。在 MATLAB 中实现 FFT 算法只需要一个函数 $\text{fft}()$ 就可实现,如果采用 VC++ 将需要很长的代码,并且还需要理解 FFT 的原理。如果采用 VC++ 调用 MATLAB 引擎函数来实现 FFT 算法,就大大减少了编程的工作量,提高了编程效率,可以很方便地实现信号的 FFT 变换。

现有模拟信号 $x(t) = \sin(2\pi ft) + 3\sin(4\pi ft)$, 求 N 点 DFT 的幅值谱。其主要步骤如下:

(1) 用 VC++ 生成一个 Engdemo 的单文档 MFC 应用程序。

(2) 根据第 1.4 节所介绍的方法设置编译环境。

(3) 添加一个新的 CPP 文件进入工程,其文件名为 matlab

.cpp,代码如下:

```
#include "stdafx.h"
#include "engine.h"
void MatlabDraw()
{
    Engine *ep; //打开 MATLAB 引擎,建立与本地 MATLAB 的连接
    if(! (ep = engOpen(NULL)))
        MessageBox(NULL, "不能打开 MATLAB 引擎", "engine demo",
        MB_OK); //向 MATLAB 传送数据,分析、运算数据
    engEvalString(ep, "N = 1024;");
    engEvalString(ep, "t = 0:0.001:1.023;");
    engEvalString(ep, "f = 100;");
    engEvalString(ep, "x = sin(2 * pi * f * t) + 3 * sin(2 * pi * 2 * f *
    t);");
    //进行 FFT 变换,并将运算结果以图像形式显示出来返回 VC++
    engEvalString(ep, "y = fft(x, N);");
    engEvalString(ep, "fre = abs(y);");
    engEvalString(ep, "subplot(2, 1, 1);");
    engEvalString(ep, "plot(t, x);");
    engEvalString(ep, "xlabel('时间');");
    engEvalString(ep, "subplot(2, 1, 2);");
    engEvalString(ep, "plot(fre(1:600));");
    engEvalString(ep, "xlabel('频率');");
    MessageBox(NULL, "please any key", "engine demo", MB_OK);
    //关闭引擎
    engClose(ep);
}
```

(4) 添加一个“Engine”菜单,在“Engine”菜单下添加一个“demo”菜单项。

(5) 实现“demo”菜单项的消息函数。

```
void CEngdemoView::OnEngineDemo()
```

```
{ MatlabDraw(); }
```

(6) 在文件 EngdemoView.cpp 的头部添加函数声明:

```
Extern void MATLABDraw();
```

编译运行程序,单击“Engine”菜单上的“demo”菜单项,后台启动 MATLAB,并显示如图 1 所示的结果。

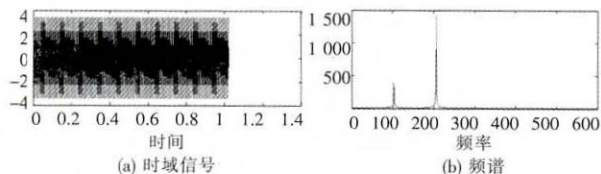


图 1 时域信号及其频谱

2 虚拟仪器开发平台 LabVIEW, LabWindows/CVI 与 MATLAB 接口技术

LabVIEW 是目前国际上唯一的编译型图形化编程语言,它提供了丰富的数据采集、分析及存储的库函数,是一个开放

型的开发环境,拥有大量与其他应用程序通信的 VI 库。例如 LabVIEW 使用自动化 ActiveX, DDE 和 SQL, 可与其他 Windows 应用程序集成;在对硬件的支持方面, LabVIEW 集成了与 GPIB, VXI, PXI, RS-232/485, PLC 和插入式数字采集设备等进行数据通信的全部功能。LabVIEW 还直接支持动态数据交换 (DDE)、结构化查询语言 (SQL)、TCP 和 UDP 网络协议等。LabWindows/CVI 以标准 C 语言为基础,将计算机软件设计平台与数据的采集、分析、处理及结果表达等仪器专业工具有机地结合起来,作为交互式的集成开发环境, LabWindows/CVI 提供大量与外部代码或软件进行接口的机制,如 DLL (动态链接库), DDE (共享库), ActiveX 控件等。LabWindows/CVI 是熟悉 C 语言的开发设计人员开发检测、数据采集、信号处理、监控等虚拟仪器的理想工具。尽管如此,但 LabVIEW 和 LabWindows/CVI 仍然无法满足复杂的数值计算和分析需求,所以在大型的测试系统和虚拟仪器开发中, LabWindows/CVI 就显得力不从心,而 MATLAB 是一种高效率的数学计算工具,将它与 LabVIEW, LabWindows/CVI 混合起来编程是一种非常有效的开发途径。下面就 LabVIEW, LabWindows/CVI 与 MATLAB 之间的接口技术进行详细介绍,并举例验证。

2.1 LabVIEW 与 MATLAB 的接口

LabVIEW 可通过 MATLAB Script 节点与 MATLAB 通信。

2.1.1 MATLAB Script 节点

MATLAB Script 节点使用户既可以在 MATLAB 下编写的 M 程序导入到流程图中,又可在流程图中根据 MATLAB 程序的语法编辑 M 程序。在 LabVIEW 中选择该节点的操作为:在流程图编辑窗口中由 Function→Mathematics→Formula→MATLAB Script 将该节点添加到流程图中。打开 MATLAB Script 节点之后,可直接在其框图内编写 MATLAB 程序代码。如果事先已经将 MATLAB 程序编辑好,则可直接将其导入到节点中,可通过单击“Import”完成导入;还可以通过在节点边框上右击鼠标右键,在弹出式菜单中选择“Add Import”或“Add Output”增加输入、输出变量,起到在 LabVIEW 与 MATLAB 之间传递参数的作用。

2.1.2 MATLAB Script 应用示例

前面详细介绍了 MATLAB Script 节点的使用方法,下面通过虚拟信号提取的仿真设计加以演示。

(1) 小波分析在信号提取中的应用。在信号的采集、传输与处理过程中,由于外界或电路内部因素的干扰,使得信号被噪声污染,所处理的噪声或高频信号中夹杂着一些有用的缓变信号,通过小波分析可将缓变的有用信号从噪声或其他高频信号中识别出来。在实际工程中,有用信号通常表现为频率较低、平稳的信号,而噪声信号或其他高频信号则表现为频率较高的信号。应用小波分析进行信号提取主要涉及到小波的分解与重构。

(2) 虚拟信号提取仪的设计。本例中含有噪声的一维信号为

$$s(i) = f(i) + e(i), i = 0, 1, 2, \dots, n-1$$

其中, $f(i)$ 为真实的低频缓变信号——Sine Wave; $e(i)$ 为噪声信号——Uniform White Noise。

在该仿真过程中,LabVIEW 通过数据采集或仿真生成含有噪声或高频信号的低频缓变信号,通过虚拟仪器前面板设置信号提取处理的参数,将参数通过 LabVIEW 与 MATLAB 的接口传递给 MATLAB 相应的功能函数,完成信号分析与处理功能,最后将处理结果回传给 LabVIEW 进行显示。MATLAB 小波工具箱提供了许多小波分析功能函数,包括小波分析通用函数、小波函数、一维小波变换、二维小波变换、小波包算法等。在该仿真过程中主要应用了 WEDN 函数,该函数可直接对一维信号消噪,其调用格式为 $[XD, CXD, LXD] = WEDN(X, TPTR, SORH, SCAL, N, 'wname')$,函数的返回值是经过对原始信号 X 进行消噪处理后的信号 XD 及其分解结构 $[CXD, LXD]$ 。式中, $SORH$ 指定软阈值 ($SORH = s$) 或硬阈值 ($SORH = h$) 的选择; $TPTR$ 指定阈值的选取规则,它有四种选择: $Rigrsure, Heursure, Sqtwolog, Minimaxi$,本程序中选择 $Heursure$,即首次选择的启发式变量作为阈值; N 为小波分解的层数; $wname$ 指定分解时所用的小波。小波信号提取仿真过程的设计步骤如下:

(1) 打开 LabVIEW,在前面板开发窗口中放置两个波形显示控件(Waveform Graph),即用于显示含噪声的波形和用于显示消除噪声后的波形。放置三个数值输入控件,分别用于设置低频缓变信号频率、消噪层数(X)和采样频率。

(2) 在流程图窗口中放置正弦波形发生器(Sine Wave.vi)、均匀分布的随机信号发生器(Uniform White Noise.vi),分别用来产生正弦波序列和均匀分布的随机序列,它们分别用来仿真低频缓变信号和噪声信号;添加加法器(Add)将低频缓变信号和噪声信号叠加;由 Function->Mathematics->Formula 操作选择 MATLAB Script 放到流程图编辑框中,添加输入变量 XX, X 和输出变量 XD ,对应数据类型分别为 Real Vector, Real, Real Vector。

(3) 按照图 2 所示的连线图进行数据流向的连接,在 MATLAB Script 节点编辑框中编写 M 程序。

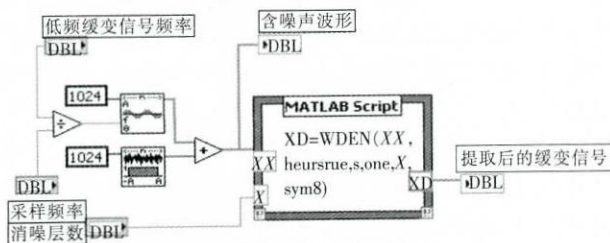


图 2 虚拟信号仿真仪器流程图

(4) 运行程序。单击前面板上的“低频缓变信号频率”按钮,选择仿真正弦信号的频率;单击“消噪层数”按钮,确定消噪层数;单击采样频率按钮,选择仿真正弦信号的采样频率;最后运行仪器 VI 文件。其结果如图 3 所示。

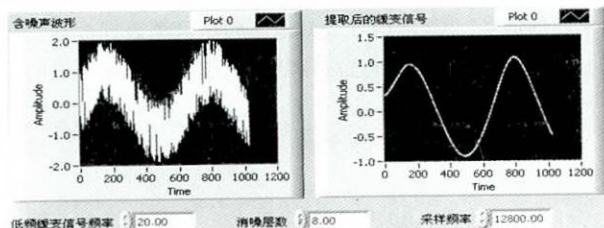


图 3 运行结果

由此可以看出,运用 LabVIEW 和 MATLAB 软件的接口技

术,通过小波分析成功地实现了信号提取的仿真过程。通过此例可以看到,基于 LabVIEW 和 MATLAB 接口技术的虚拟仪器的开发是完全可行的。LabVIEW 是图形化仪器编程语言,它方便易用,尽管其提供了大量的信息功能处理函数,但毕竟功能有限;而 MATLAB 有强大的矩阵列运算、数值分析、信号处理、系统识别、图像语音处理等功能。通过接口技术在 LabVIEW 中调用 MATLAB 是开发虚拟仪器的有效途径。

2.2 LabWindows/CVI 与 MATLAB 的接口通信

通过在 LabWindows/CVI 环境下建立一个数据交换 ActiveX 服务控件可以实现与 MATLAB 之间的接口通信。建立 ActiveX 服务的方法有两种:

(1) 在 LabWindows/CVI 6.0 窗口中,用户自行创建 ActiveX 服务函数,但这种方法要求先装 MATLAB,后装 LabVIEW/CVI。其步骤如下:

① 打开 LabWindows/CVI 窗口中选择 Tools->Create ActiveX Controller,弹出 ActiveX Controller Wizard-Welcome 对话框,单击“Next”,会在 ActiveX Controller Wizard-Choose Server 中列表显示出系统已安装的 ActiveX Server,在列表中选择“Matlab Automation Server Type Library”选项。

② 单击“Next”后,弹出 ActiveX Controller Wizard-Configure 对话框,需要对目标 FP 文件命名并设定存放路径,假设文件名为 MATLABatvx.fp,存放路径为 D:\CVI6.0\Work,单击“Next”。

③ 在弹出的 ActiveX Controller Wizard-Advanced Options 对话框中单击“Advanced Options...”,按下“Check All”按钮,其余按照默认方式。单击“Next”后,系统将自动生成 MATLABatvx.fp, MATLABatvx.c, MATLABatvx.h, MATLABatvx.obj 和 MATLABatvx.sub 五个文件。在目录 D:\CVI6.0\Work 下找到如图 4 所示的五个图标。



图 4 ActiveX 服务函数图标

(2) 利用 LabWindows/CAI 自带的 ActiveX 服务函数进行接口通信。5.0 以上版本的 LabWindows/CAI 的 CVI\samples\activex\matlab 中直接提供有 LabVIEW/CVI 和 MATLAB 接口,通信的 ActiveX 服务函数有 MATLABsrrvr.c, MATLABsrrvr.h, MATLABsrrvr.obj, MATLABsrrvr.fp, MATLABsrrvr.sub。但是由于各个用户所装的 MATLAB 版本的不同,在实现混合编程时,并不能直接应用 CVI\samples\activex\MATLAB 下的 ActiveX 服务函数,因为不同版本 MATLAB 软件的 MATLAB Application 的注册码是不同的,这就需要修改 MATLAB 的注册码。经笔者悉心研究,总结其方法及其步骤如下:

① 打开注册表编辑器,查到用户安装的相应版本 MATLAB Application 的注册码,以 version 6.5 为例,其注册码是 {554F6053-79D4-11D4-B067-009027BA5F81}。

② 由 CVI\samples\activex\matlab 打开 ActiveX 服务函数 matlabssrvr.c,将 matlabssrvr.c 中自带的 MATLAB 注册码 5.01 版本:GUID clsid = {0x669CEC94, 0x6E22, 0x11CF, 0xA4, 0xD6, 0x0, 0xA0, 0x24, 0x58, 0x3C, 0x19} 改写为

```
GUID clsid = {0x554F6053, 0x79D4, 0x11D4, 0xB0, 0x67, 0x0, 0x90,
0x27, 0xBA, 0x5f, 0x81};
```

以上两种方法得到的 ActiveX 服务函数的功能是一样的,即包含了 LabWindows/CVI 与 MATLAB 的混合编程所要用的最基本的功能函数,包括打开 MATLAB 函数、关闭 MATLAB 函数、发送矩阵函数、接收矩阵函数、运行 MATLAB 功能的函数及 MATLAB 窗口最大/最小化函数等。为了方便开发者实现 LabWindows/CVI 环境下与 MATLAB 混合编程,NI 公司还专门配置了一个更加友好的接口函数文件 `matlabutil.c`。

在 LabWindows/CVI 中实现与 MATLAB 混合编程的具体方法如下:ActiveX 服务函数创建完毕后,就可实现混合编程了。启动 LabWindows/CVI,选择 File->New->UserInterface (*.uir) 创建仪器面板文件,面板设计完毕后保存该文件,取名为 `Example.uir`;选择 Code->Generate->All Code,编写源代码,并以 `Example.c` 名保存。经过面板设计和源代码编写后就可创建并保存工程文件了。在工程文件窗口中选择 Edit->Add Files to Project,添加 `Example.c`, `Example.h`, `Example.uir`,并添加上文提到的 ActiveX 服务函数;在工程文件、用户面板编辑或代码窗口中操作 Run->Run Project,就可调试、运行工程文件了。由于篇幅所限,笔者在此就不举例了。相信只要熟悉 LabWindows/CVI 的读者都会根据上述方法很顺利地实现在 LabWindows/CVI 环境下与 MATLAB 混合编程。

3 结束语

本文系统地介绍了 Visual C++, LabVIEW, LabWindows/

CVI 与 MATLAB 之间的接口技术,它们的实现极大地缩短了程序开发周期,减轻了编程工作量,提高了工作效率。在工程研究中,这种混合编程的接口技术将会有很大的应用。本文所举例程在 VC++ 6.0, MATLAB 6.5, LabVIEW 6.1 中调试全部通过,验证了上述接口技术的正确性。

参考文献:

- [1] [美] David J Kruglinski, et al. Programming Visual C++ 6.0 技术内幕[M]. 北京:北京希望电子出版社,1999. 321-322.
- [2] MATLAB C++ Math Library Reference [M]. MathWorks Inc., 2000. 150-152.
- [3] NI 公司. LabWindows/CVI Programmer Reference Manual [M]. 2001. 160-161.
- [4] 何强. MATLAB 扩展编程[M]. 北京:清华大学出版社,2002. 170.
- [5] 刘君华. 基于 LabVIEW 的虚拟仪器设计[M]. 北京:电子工业出版社,2003. 127-128.
- [6] 刘君华. 基于 LabWindows/CVI 的虚拟仪器设计[M]. 北京:电子工业出版社,2003. 87-89.
- [7] 陶红,袁金荣. MATLAB 与 C/Visual C++ 混合编程的实现[J]. 计算机工程与应用,2000,36(10):100-101,104.
- [8] 郑力新,周凯汀,龚勇,等. LabWindows/CVI 与 MATLAB 软接口技术及应用[J]. 自动化仪表,2004,25(4):14-16.
- [9] 聂开俊,赵转萍. LabWindows/CVI 中 ActiveX 服务器的开发[J]. 计算机仿真,2004,21(12):232-234.

作者简介:

王华(1981-),男,甘肃临泽人,硕士研究生,主要研究方向为机械电子工程;袁中凡(1950-),男,教授,博导,主要研究方向为机械电子工程。

(上接第 280 页)

2.3 基于 CF 卡的嵌入式系统平台构建

在完成操作系统定制及优化的基础上,需要进一步对嵌入式软硬件进行整合。笔者通过以下步骤完成了所定制嵌入式软件系统在目标嵌入式硬件平台的部署:创建文件系统;部署定制的嵌入实时操作系统;架构精简的嵌入式图形库;安装启动装载程序;修改启动配置文件;部署应用软件。

本研究所形成的嵌入式软件平台现已在新型智能缝制设备嵌入式系统产品 NPU-ECS-36-68(图 3)中得到应用。

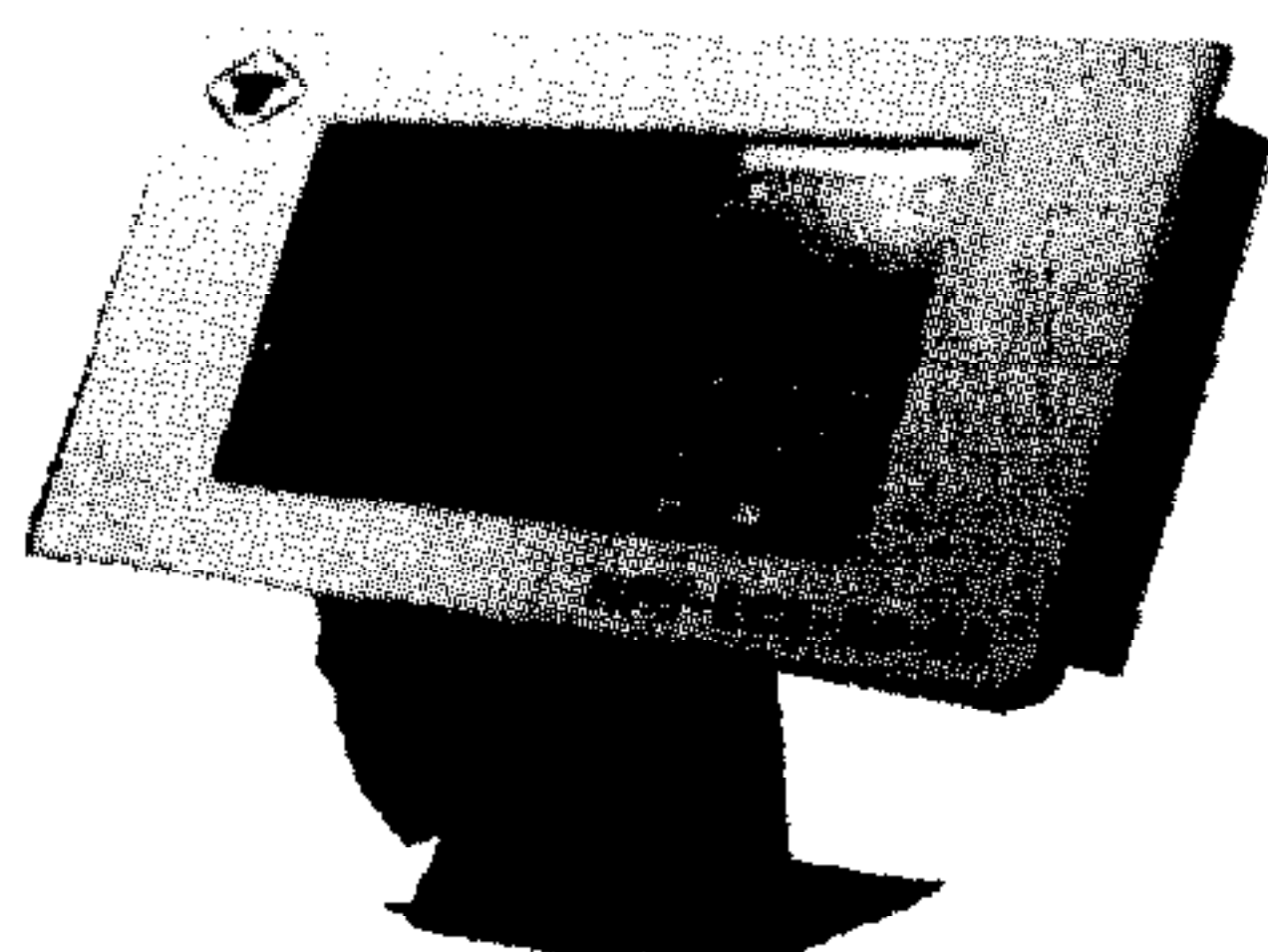


图 3 缝制设备嵌入式系统

3 结论与展望

嵌入式系统技术是传统缝制设备信息化改造的技术关键,同时也是知识产权的核心。本文重点研究了基于 Linux 构建面向领域的嵌入实时软件系统的关键技术和方法,并进一步实现了一个缝制设备嵌入式系统,这也说明所研究技术路线及所采用方法是可行的。在下一步的研究工作中,我们将结合嵌入

式软构件等技术开展可定制嵌入式软件技术研究,进而形成满足特定领域系列化产品需求的可定制嵌入式软件平台。

参考文献:

- [1] Zhang Kailong, Zhou Xingshe, Liang Ke, et al. The Customizable Embedded System for Seriate Intelligent Sewing Equipment [C]. ICES, LNCS 3605, 2004. 143-149.
- [2] P Mendoza, J Vila, S Terrasa. Using RT-Linux for Developing Real-time Embedded Systems [C]. IFAC Conference on New Technologies for Computer Control, 2001. 68-73.
- [3] 赵雁南,贾培发,张玉彬. 基于 RT-Linux 的串行实时通信系统 SERCOS 的设计[J]. 计算机工程与应用,2002,38(19):153-155.
- [4] 朱达宇,李彦,吉华,等. 基于 RT-Linux 的全软件数控系统[J]. 计算机集成制造系统,2004,10(10):1571-1575.
- [5] 林洪,蔡光起,李凤究,等. 实时化的 Linux 系统及其实时性能的研究[J]. 小型微型计算机系统,2004,25(8):1454-1457.
- [6] 刘峥嵘,张智超,等. 嵌入式 Linux 应用开发详解[M]. 北京:机械工业出版社,2004.

作者简介:

张凯龙(1977-),男,陕西凤翔人,讲师,博士研究生,主要研究方向为网络化嵌入式计算、嵌入式软件测试技术;梁克(1982-),男,河南南阳人,博士研究生,主要研究方向为嵌入式中间件技术;王博伟(1981-),男,陕西乾县人,硕士研究生,主要研究方向为网络化嵌入式计算技术;杨志义(1952-),男,陕西蒲城人,教授,主要研究方向为嵌入式系统体系结构技术。