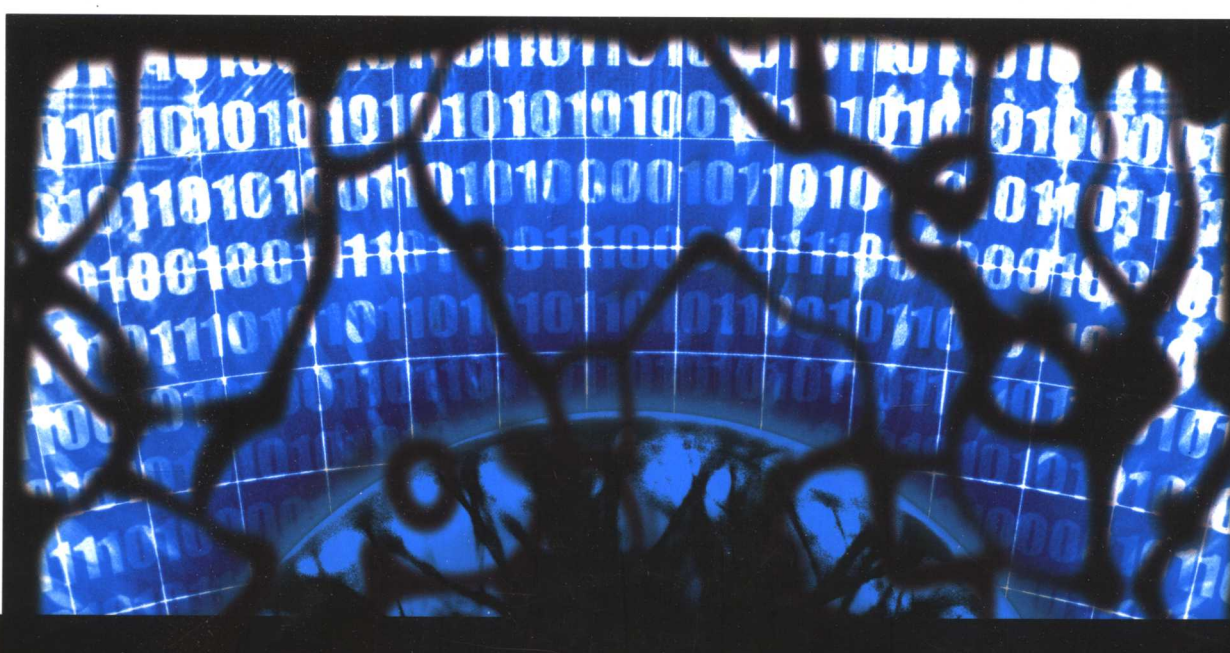




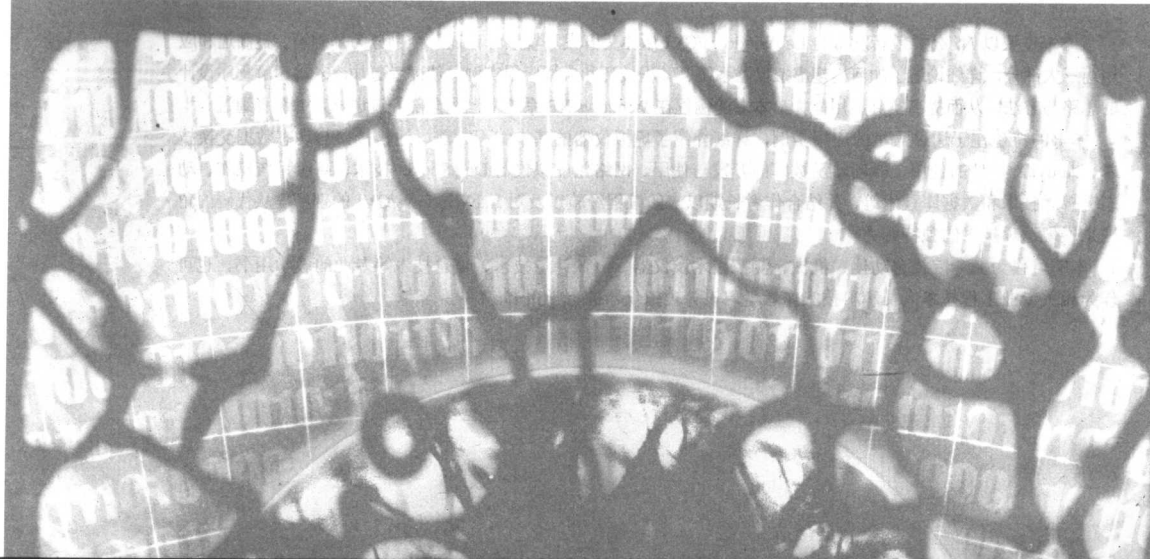
LabWindows/CVI

测试技术**及**工程应用

王建新 杨世凤 隋美丽 主编



化学工业出版社



LabWindows/CVI

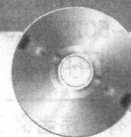
测试技术及工程应用

王建新 杨世凤 隋美丽 主编



化学工业出版社

·北京·



附光盘

本书主要以 NI 公司最新发布的 LabWindows/CVI 版本为开发环境展开论述, 全书分为基础篇、提高篇和应用篇三大部分, 详细论述了如何构建虚拟仪器系统、编程技巧和典型应用实例, 使读者更加深入地理解程序设计的思路与方法, 从而提高实际应用能力。本书的例程在 Windows XP 系统下已经全部经过调试, 并编译通过。

本书的特色在于通过大量的应用实例, 详细介绍了虚拟仪器的基本编程方法、通信接口与动态数据交换技术、ActiveX 控件的使用、驱动器函数库的开发与使用、网络化虚拟仪器 DataSocket 技术、高级系统操作、动态链接库技术和数据库技术等, 并结合作者长期以来的编程经验以及对 LabWindows/CVI 的理解, 从代码的角度来展现这项技术的独特魅力。

本书可作为大专院校自动化、机械、测控技术及仪器、测试计量技术、信息技术等专业的教学用书, 也可作为相关专业工程技术人员和科技工作者学习设计虚拟仪器的自学用书。

图书在版编目 (CIP) 数据

LabWindows/CVI 测试技术及工程应用/王建新, 杨世凤,
隋美丽主编. —北京: 化学工业出版社, 2006.8
ISBN 7-5025-9234-2

I. L… II. ①王…②杨…③隋… III. 软件工具,
LabWindows/CVI-程序设计 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2006) 第 102643 号

LabWindows/CVI 测试技术及工程应用

王建新 杨世凤 隋美丽 主编

责任编辑: 周 红 宋 辉

责任校对: 洪雅姝

封面设计: 尹琳琳

化学工业出版社出版发行

(北京市朝阳区惠新里 3 号 邮政编码 100029)

购书咨询: (010) 64982530

(010) 64918013

购书传真: (010) 64982630

<http://www.cip.com.cn>

新华书店北京发行所经销

大厂聚鑫印刷有限责任公司印刷

三河市延风装订厂装订

开本 787mm×1092mm 1/16 印张 28 $\frac{1}{4}$ 字数 779 千字

2006 年 10 月第 1 版 2006 年 10 月北京第 1 次印刷

ISBN 7-5025-9234-2

定 价: 59.00 元

版权所有 违者必究

该书如有缺页、倒页、脱页者, 本社发行部负责退换

前言

随着微电子技术、计算机技术和通信技术的迅速发展,测控技术与仪器科学领域发生了巨大的变化,美国国家仪器公司(National Instruments Corporation,简称NI)于20世纪80年代中期首先提出了基于计算机技术的虚拟仪器概念,随后研制和推出了基于多种总线系统和开发平台的虚拟仪器,把现代测试及仪器科学技术带入了新的发展时期。

虚拟仪器技术是日益发展的计算机技术、仪器技术、信息技术密切结合的产物,虚拟仪器就是在通用计算机上加上软件和(或)硬件,使用者在操作这台计算机时,就像是在操作一台用户自己设计的专用电子仪器。在虚拟仪器系统中,硬件仅仅是为了解决信号的输入、输出,软件才是整个仪器系统的关键,任何一个使用者都可以通过修改软件的方法,很方便地改变、增减测控仪器系统的功能与规模,所以有“软件就是仪器”之说。虚拟仪器技术的出现,彻底打破了传统仪器由厂家定义,用户无法改变的模式,虚拟仪器技术给用户一个充分发挥自己的才能、想像力的空间。用户(而不是厂家)可以随心所欲地根据自己的需求,设计自己的仪器系统,满足多种多样的应用需求。它利用计算机系统的强大功能,结合相应的硬件,大大地突破传统仪器在数据处理、显示、传送等方面的限制,使用户可以方便地对其进行维护、扩展和升级。

虚拟仪器系统可以广泛地应用在通信、自动化、半导体、航空、电子、电力、生化、制药和工业生产的各个领域。虚拟仪器发展到现在,实际上它不仅仅局限在测控仪器的范畴,在很多时候、很多场合,它其实是一种思维方式、一种概念化的抽象形式。

本书主要以NI公司最新发布的LabWindows/CVI版本为开发环境展开论述,全书分为基础篇、提高篇和应用篇三大部分,详细论述了如何构建虚拟仪器系统、编程技巧和典型应用实例,使读者更加深入地理解程序设计的思路与方法,提高实际应用能力。本书的例程在Windows XP系统下已经全部经过调试,并编译通过。

本书的特色在于通过大量的应用实例,详细介绍了虚拟仪器的基本编程方法、通信接口与动态数据交换技术、ActiveX控件的使用、驱动器函数库的开发与使用、网络化虚拟仪器Data Socket技术、高级系统操作、动态链接库技术和数据库技术等,并结合作者长期以来的编程经验以及对LabWindows/CVI的理解,从代码的角度来展现其独特的魅力。

本书由天津科技大学王建新博士、杨世凤教授、中国农业大学隋美丽博士主编,郭雅萌为副主编,参与本书编写工作的还有钱东平、张磊伟、陈浩、童官军、赵继民、李志广等。

由于本书内容涉及面比较广,加上作者水平有限、时间仓促,书中难免有不妥之处,恳请广大读者批评指正。

编者
2006年7月

目 录

第1篇 基础篇

第1章 虚拟仪器及其开发语言 LabWindows/CVI	1
1.1 虚拟仪器	1
1.1.1 虚拟仪器的基本概念	1
1.1.2 虚拟仪器的结构	2
1.2 LabWindows/CVI 编程语言	3
1.2.1 LabWindows/CVI 特性	3
1.2.2 LabWindows/CVI 的工作空间	4
1.2.3 LabWindows/CVI 的文件类型	5
1.2.4 LabWindows/CVI 中的对象编程	5
1.2.5 LabWindows/CVI 的基本编程窗口	6
第2章 构建一个简单程序	12
2.1 建立工程文件	12
2.2 创建用户界面文件	12
2.2.1 面板的设置	12
2.2.2 向面板中添加控件	13
2.3 生成源代码文件	15
2.3.1 生成全部源代码框架	15
2.3.2 主函数及回调函数框架分步生成	17
2.3.3 源代码框架主函数	18
2.3.4 向源代码框架中添加回调函数	19
2.4 运行和调试程序	23
2.4.1 程序单步调试	23
2.4.2 程序断点调试	23
2.4.3 程序变量和内存观察	24
2.5 生成可执行文件和发布文件	25
2.5.1 生成可执行文件	25
2.5.2 生成发布文件	26
2.6 LabWindows/CVI 编程的基础知识	27
2.6.1 控件模式	27

2.6.2 用户交互事件	27
2.6.3 回调函数的模式	29
第3章 基本控件的使用与编程	31
3.1 文本框、列表框、树控件和表格控件	31
3.1.1 文本框	31
3.1.2 列表框、树控件和表格控件	33
3.1.3 编程实例	37
3.2 定时器控件	41
3.2.1 定时器相关函数	41
3.2.2 编程实例——板球游戏	42
3.3 创建菜单	48
3.3.1 菜单编辑器	48
3.3.2 菜单项函数	49
3.3.3 编程实例——计算器制作	50
3.4 图表控件	52
3.4.1 Graph 控件	52
3.4.2 Strip Chart 控件	54
3.4.3 Digital Graph 控件	54
3.4.4 绘图控件编程	55
3.5 弹出式面板	58
3.5.1 弹出面板类型	58
3.5.2 弹出式面板编程	61
3.6 Splitter 控件	66
3.6.1 Splitter 控件设置	66
3.6.2 Splitter 控件函数	67
3.7 滚动条及仪器驱动的使用	68
3.7.1 仪器驱动的装载	68
3.7.2 滚动条控件	68
3.7.3 滚动条编程	69
3.8 文件的读写和格式化	73
3.8.1 文件读写函数	73
3.8.2 字符串操作函数	75
3.8.3 数据格式化函数	75
3.8.4 定制用户工具	75

第2篇 提高篇

第4章 Toolslib 扩展控件编程	77
4.1 Hot Ring 控件	77
4.2 3D Graph 控件	79

4.3	Animation Control 控件	85
4.4	Combo Box 与 Password 控件	89
4.5	File Browser 与 Path Control 控件	95
4.6	Radio Group 控件	100
4.7	Graph Cursor 控件	103

第5章 数据采集技术 109

5.1	数据采集卡 (DAQ)	109
5.1.1	数据采集卡的组成	109
5.1.2	数据采集卡的选择	110
5.2	数据采集卡测试	110
5.2.1	数据采集卡的参数设置	110
5.2.2	数据采集卡属性设置	111
5.2.3	数据采集卡的测试	112
5.3	基于 NI-DAQ 数据采集	113
5.3.1	Easy I/O for DAQ 函数库	113
5.3.2	Traditional NI-DAQ 函数库	114
5.3.3	利用 Easy I/O for DAQ 实现数据采集	119
5.3.4	利用 DAQ 助手创建测试任务	122
5.4	基于非 NI 数据采集卡的应用	128
5.4.1	ISA 数据采集卡	128
5.4.2	PCI 数据采集卡	133

第6章 信号的分析处理技术 141

6.1	信号的分类	141
6.1.1	信号的描述与分析	141
6.1.2	采样技术	142
6.2	信号处理编程	143
6.2.1	信号的产生	143
6.2.2	信号时域分析	149
6.2.3	信号频域分析	156
6.2.4	信号滤波	165

第7章 通信接口与动态数据交换技术 177

7.1	RS-232 串口通信	177
7.2	串口数据收发过程	179
7.2.1	RS-232 文件发送程序	179
7.2.2	RS-232 文件接收程序	183
7.2.3	RS-232 与其他设备间的通信	187
7.3	DDE 动态链数据交换技术	193

7.3.1 动态数据交换技术	193
7.3.2 LabWindows/CVI 动态数据交换的一般步骤	193
7.3.3 DDE 与 Excel 间的通信实例	194

第 8 章 网络化虚拟仪器设计 202

8.1 网络测控系统	202
8.2 TCP/IP 技术	202
8.2.1 OSI 体系结构及协议	202
8.2.2 TCP/IP 协议	204
8.2.3 TCP/IP 参考模型	205
8.3 TCP 程序设计	206
8.3.1 TCP 服务器程序设计	206
8.3.2 TCP 客户机程序设计	215
8.4 DataSocket 技术	220
8.4.1 DataSocket 技术特性	220
8.4.2 DataSocket 服务器和管理器	221
8.4.3 DataSocket 程序设计	223
8.5 Polling Queue 技术	233
8.5.1 Polling Queue 服务器程序设计	233
8.5.2 Polling Queue 客户机程序设计	237

第 3 篇 应用篇

第 9 章 Windows SDK 程序设计 243

9.1 SDK 和 Windows API 函数简介	243
9.1.1 电脑驱蚊程序制作	244
9.1.2 特殊图形面板的制作	246
9.1.3 面板的拖拽操作	253
9.2 注册表操作	255
9.2.1 定制 IE 工具栏	256
9.2.2 开机启动程序制作	263

第 10 章 动态链接库技术 266

10.1 基本外部模块的编程	266
10.1.1 创建无界面动态链接库	267
10.1.2 创建静态链接库	274
10.1.3 创建目标代码文件	275
10.1.4 各种库函数的产生	276
10.2 创建有界面的动态链接库	277
10.3 外部模块的显式调用方式	283

第 11 章 高级系统操作	289
11.1 系统托盘设计	289
11.2 工具栏设计	295
11.3 ini 文件读取操作	303
11.4 异步定时器	311
11.5 多线程编程	316
11.6 文件拖拽操作	320
第 12 章 ActiveX 技术	325
12.1 ActiveX 简介	325
12.2 LabWindows/CVI 中 ActiveX 的调用	326
12.2.1 LabWindows/CVI 中 ActiveX 服务器的创建	326
12.2.2 注册一个外部 ActiveX 服务器	327
12.2.3 生成仪器驱动	327
12.3 ActiveX 应用实例	330
12.3.1 TimePicker 控件的使用	330
12.3.2 Calendar 控件的使用	338
12.3.3 Chart 控件的使用	343
12.3.4 ProgressBar 控件的使用	350
12.3.5 MCI 控件的使用	352
12.3.6 RichBox 控件的使用	358
12.3.7 ShockwaveFlash 控件的使用	364
第 13 章 驱动器函数库开发	375
13.1 驱动器函数库开发实例	375
13.1.1 驱动器函数库的结构规划	375
13.1.2 驱动器函数库的生成	377
13.1.3 其他文件的生成	380
13.2 利用已开发好的工具	382
13.3 驱动器函数库的装载	383
第 14 章 数据库技术	386
14.1 信息、数据与数据处理	386
14.1.1 基本概念	386
14.1.2 数据处理技术的发展	386
14.1.3 数据模型	387
14.1.4 数据层次	387
14.2 ODBC 数据源管理器	388
14.2.1 ODBC 配置	388
14.2.2 数据库会话	389

14.3 SQL Toolkit.....	389
14.3.1 函数详解.....	389
14.3.2 基本 SQL 语句.....	411
14.4 数据库编程举例.....	413
14.4.1 表格编程举例.....	413
14.4.2 常规数据库操作编程举例.....	422
附录 A 函数 Fmt 标识符参考.....	435

附录 B SQL 命令.....	436
------------------	-----

参考文献.....	441
-----------	-----

1.1.2 虚拟仪器的结构

一个典型的数据采集控制系统由传感器、信号调理电路、数据采集卡(板)、计算机、控制执行设备五部分组成。一个好的数据采集产品不仅应具备良好的性能和高可靠性,还应提供高性能的驱动程序和简单易用的高层语言接口,使用户能较快速地建立可靠的应用系统。近年来,由于多层电路板、可编程仪器放大器、即插即用、系统定时控制器、多数据采集板实时系统集成总线、高速数据采集的双缓冲区以及实现数据高速传送的中断、DMA(直接存储器存取)等技术的应用,使得最新的数据采集卡能保证仪器级的高准确度与可靠性。

软件是虚拟仪器测控方案的关键。虚拟仪器的软件系统主要分为四层结构:系统管理层、测控程序层、仪器驱动层和I/O接口层。

I/O接口驱动程序完成特定外部硬件设备的扩展、驱动和通信。DAQ(数据采集卡)硬件是离不开相应软件的,大多数的DAQ应用都需要驱动软件。驱动软件直接编制DAQ硬件的登录、操作管理和集成系统资源,如处理器中断、DMA和存储器等的软件层管理。驱动软件隐含了低级、复杂的硬件编程细节,而提供给用户的是容易理解的界面。控制DAQ硬件的驱动软件按功能可分为模拟I/O、数字I/O和定时I/O。驱动软件有如下的基本功能。

- ① 以特定的采样频率获取数据。
- ② 在处理器运算的同时提取数据。
- ③ 使用编程的I/O、中断和DMA传送数据。
- ④ 在磁盘上存取数据流。
- ⑤ 同时执行几种功能。
- ⑥ 集成一个以上的DAQ卡。
- ⑦ 同信号调理器结合在一起。

虚拟仪器硬件系统包括GPIB(IEEE 488.2)、VXI、插入式数据/图像采集板、串行通信与网络等几类I/O接口。虚拟仪器测试系统构成方案如图1-1所示。

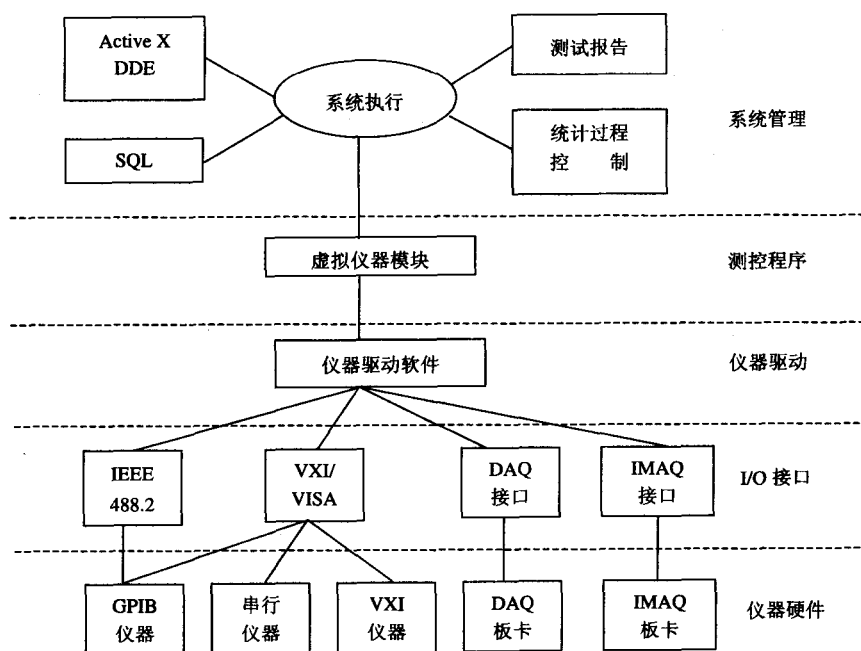


图 1-1 虚拟仪器测试系统构成方案

GPIB (General Purpose Interface Bus) 是目前使用最为广泛的仪器接口, IEEE 488.2 标准使基于 GPIB 的计算机测试系统进入了一个新的发展阶段。GPIB 总线的出现, 提高了仪器设备的性能指标。利用计算机对带有 GPIB 接口的仪器实现操作和控制, 可实现系统的自动校准、自诊断等要求, 从而提高了测量精度, 便于将多台带有 GPIB 接口的仪器组合起来, 形成较大的自动测试系统, 高效地完成各种不同的测试任务, 而且组建和拆散灵活, 使用方便。

VXI 总线是 VMEbus eXtension for Instrumentation 的缩写, 即 VME 总线在测量仪器领域中的扩展。它能够充分利用最新的计算机技术来降低测试费用, 增加数据吞吐量和缩短开发周期。VXI 系统的组建和使用越来越方便, 其应用面也越来越广, 尤其是在组建大、中规模自动测量系统以及对精度、可靠性要求较高的场合, 有着其他仪器系统无法比拟的优势。

PCI (Peripheral Component Interconnect Special Interest Group, PCISIG 简称 PCI), 即外部设备互连。PCI 总线是一种即插即用 (PnP, Plug-and-Play) 的总线标准, 支持全面的自动配置, PCI 总线支持 8 位、16 位、32 位、64 位数据宽度, 采用地址/数据总线复用方式。其主要特点有: 突发传输, 多总线主控方式, 同步总线操作, 自动配置功能, 编码总线命令, 总线错误监视, 不受处理器限制, 适合多种机型, 兼容性强, 高性能价格比, 预留了发展空间等。PC-DAQ 测试系统是以数据采集卡、信号调理电路及计算机为硬件平台组成的测试系统, 如图 1-2 所示。这种方式借助于插入 PC 中的数据采集卡和专用的软件, 完成具体的数据采集和处理任务。由于系统的组建方便, 数据采集效率高, 成本低廉, 因而得到广泛的应用。

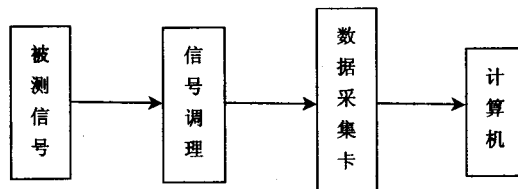


图 1-2 PC-DAQ 数据采集系统

串行总线, RS-232 总线是最早采用的通用串行总线, 最初用于数据通信上, 但随着工业测控行业的发展, 许多测量测试仪器带有 RS-232 串口总线接口。

1.2

LabWindows/CVI 编程语言

LabWindows/CVI 是一个完全的 ANSI C 开发环境, 用于仪器控制、自动检测、数据处理的应用软件。它以 ANSI C 为核心, 将功能强大、使用灵活的 C 语言平台与用于数据采集、分析和显示的测控专业工具有机结合起来。它的交互式开发平台、交互式编程方法、丰富的功能面板和函数库大大增强了 C 语言的功能, 为熟悉 C 语言的开发人员建立自动化检测系统、数据采集系统、过程控制系统等提供了一个理想的软件开发环境。

LabWindows/CVI 软件把 C 语言的有力与柔性同虚拟仪器的软件工具库结合起来, 包含了各种总线、数据采集和分析库。同时, LabWindows/CVI 软件提供了国内外知名厂家生产的三百多种仪器的驱动程序。LabWindows/CVI 软件的重要特征就是在 Windows 和 Sun 平台上简化了图形用户接口的设计, 使用户很容易地生成各种应用程序, 并且这些程序可以在不同的平台上移植。

使用 LabWindows/CVI 设计的应用程序可脱离 LabWindows/CVI 开发环境独立运行, 并可以打包生成.msi 安装文件。LabWindows/CVI 主要采用事件驱动与回调函数方式, 编程方法简单易学。

1.2.1 LabWindows/CVI 特性

(1) LabWindows/CVI 的一般特性。

① 提供了标准函数库和交互式函数面板。

② 利用便捷的用户界面编辑器，代码创建向导及函数库，实现可视化用户界面的建立、显示和控制。

- ③ 利用向导和函数库开发 IVI 驱动程序和控制 ActiveX 服务器。
- ④ 提供了部分特定仪器的驱动。
- ⑤ 可创建和编辑 NI-DAQmx 任务。

(2) LabWindows/CVI 8.0 的新特性。

① 优化的集成编译器。使用外部优化编译器可在 LabWindows/CVI 环境中编译 LabWindows/CVI 代码，还可以使用专用的 Microsoft Visual C++、Borland 和 Intel 编译器预先配置好的模板，或创建自定义模板。

② 便捷的程序发布功能。LabWindows/CVI 8.0 重新设计了应用程序的分布方式，可创建高级程序发布，其中不仅包含 LabWindows/CVI 应用程序，还包括其支持程序，如 NI-DAQmx、NI-VISA 和 NI-SCOPE 等驱动程序。

③ 调用.NET 程序集。使用.NET 库调用方法，从 Microsoft .NET 处设置/获取属性，如记录错误和监视 CPU 使用情况。LabWindows/CVI 还具有创建.NET 控制器的功能，可使用该功能生成一个仪器驱动程序，作为.NET 程序集的载体。

④ Tab 控件。LabWindows/CVI 8.0 增加了 Tab 控件，可将用户界面分别添加到多个 Tab 中。与其他控件类似，用户可在用户界面编辑器或通过编程方式创建并修改 Tab 控件。

⑤ 全新分析函数。新的高级分析库包含全新改写的曲线拟合和加窗函数、高效线性代数函数和各种特殊函数，同时对快速傅里叶（FFT）变换进行了改进。

⑥ 表格控件中增加了新的单元格类型。在 LabWindows/CVI 前期版本中，表格控件支持数字、图片和字符串单元格。在 LabWindows/CVI 8.0 中，表格控件还支持下拉列表控件、组合框和按钮单元格。

⑦ 图像文件中增加了 Alpha 通道。可以在 LabWindows/CVI 使用的位图中设置 Alpha 通道，创建半透明图像。

⑧ 工具栏的增强性能。在工具栏中可添加下拉菜单、字符串控件、颜色数值控件等。在一个面板上还可添加多个工具栏。升级的工具栏仪器驱动程序提供了一个移动句柄，借助该句柄，用户可以在面板间移动工具栏。

⑨ 升级的图表控件特性。可以为某数据点添加注释、使用坐标轴的绝对和相对时间模式、通过图例交互地改变图表的设置、将像素转换为图表单位以及在自定义坐标轴标签字符串中使用多行文本。

⑩ 其他新增特性。源文件自动换行、简化的仪器 I/O 助手代码生成、增强的 NI DIAdem 文件 I/O 性能、工作区窗口的源代码浏览器集成、在应用程序菜单栏中添加自定义图像的功能、为用户定义函数添加说明信息提示等。

1.2.2 LabWindows/CVI 的工作空间

工作空间窗口如图 1-3 所示，包括以下内容。

(1) 工程目录区 工程目录区中包含当前工作空间中所有工程的目录，位于界面的左上角。粗体的工程名表示该工程当前为激活状态。工程人员可以对这个激活的工程构建、调试、修改等。如果一个工程名后有*号，则表明工程已经被修改需要保存。

(2) 函数目录区 函数目录区包括 LabWindows/CVI 的函数库和仪器库目录，位于界面的左下角。当装载一个仪器驱动时，仪器文件夹中包含了仪器函数面板目录。双击 LabWindows/CVI 目录中的函数名，即可打开相关的函数面板。

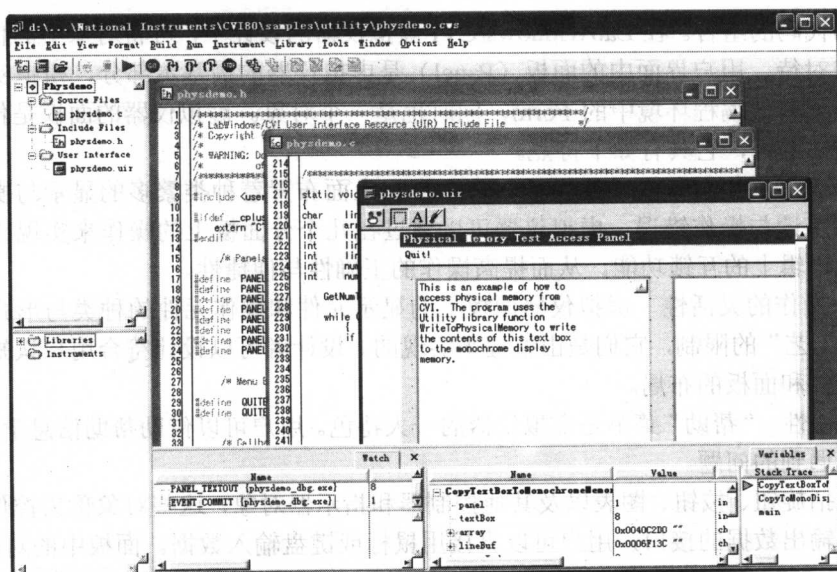


图 1-3 工作空间窗口

(3) 窗口区 在窗口区可以打开源代码编辑器、用户界面编辑器、函数面板编辑器等，位于界面的右半部分。当在打开任一窗口时，菜单栏和工具栏就会随着编辑界面的不同发生相应的改变。

(4) 输出区 输出区域包括编译错误、运行时错误和源代码错误信息窗口等，一般位于界面的右下角。这些输出窗口中包括错误、提示、程序变量列表等。在错误信息列表中双击一条错误信息，则会在源文件处与错误对应的地方加亮显示。

(5) 运行区 运行区包括变量、监视、内存和堆栈窗口，一般位于界面的右下角，可以在这些窗口中编辑变量和观察程序运行状况。

1.2.3 LabWindows/CVI 的文件类型

用 LabWindows/CVI 编写的虚拟仪器程序，其工作空间文件 (*.cws) 通常包含的文件类型有如下六种。

(1) *.prj 文件 工程文件是程序文件的主体框架，主要由*.uir 文件、*.c 文件、*.h 文件组成。程序调试运行后，可以生成可执行文件 (*.exe)。

(2) *.c 文件 C 源程序。它主要包含头文件、主程序文件和回调函数，其结构和 C 语言结构一致。

(3) *.uir 文件 用户界面文件，即面板文件。该文件中包括菜单和各种控件资源。

(4) *.h 头文件 在 LabWindows/CVI 中，头文件是由系统自动生成的。它的作用一方面是便于打开和编辑，另一方面是确保编译器在编译时能引用它们。

(5) *.fp 文件 当打开工程的仪器驱动函数面板文件时，LabWindows/CVI 自动加载仪器驱动文件。

(6) *.lib 文件 这类文件可能是 DLL 导入库文件，也可以是静态库文件。

1.2.4 LabWindows/CVI 中的对象编程

对象编程是 LabWindows/CVI 编程的核心概念。虚拟仪器的面板和面板中的控件都是对象。

对象是数据和代码的组合。在 LabWindows/CVI 虚拟仪器的设计中,可将对象中的代码和数据当成一个整体来对待。用户界面中的面板 (Panel) 是虚拟仪器的最基本部分,模拟实际仪器的面板,类似 VB 或其他编程环境中的 Form,同时也是一个对象。虚拟仪器的面板是传统仪器的面板和软件界面的融合,它具有如下特点。

(1) 面板互锁性 传统仪器的面板只有一个,上面布置着种类繁多的显示与操作元件,由此可能导致许多读与操作错误。虚拟仪器可以通过在几个分面板上的操作来实现比较复杂的功能,并且设置逻辑上的互锁功能,从而提高操作的正确性与便捷性。

(2) 控件操作的灵活性 虚拟仪器面板上的显示元件和操作元件的种类与形式不受“标准件”和“加工工艺”的限制,它们是由编程来实现的。设计者可以设计符合用户认知要求的显示元件、操作元件和面板的布局。

(3) 帮助特性 “帮助”菜单是虚拟仪器的一大特色。用户可以借助帮助信息学会操作仪器,解决使用时所遇到的问题。

面板中包括旋钮、按钮、图表以及其他控制器和指示器对象,这些对象称为控件。面板是虚拟仪器输入和输出数据的接口,用户可以直接用鼠标或键盘输入数据。面板中的对象是可视的,有一个图标 (Icon) 和它相对应。

对象的两个基本元素是属性和事件。在 LabWindows/CVI 中可以通过对象的这两个元素来操纵和控制对象。

(4) 对象的属性 属性是反映对象特征的参数,例如仪器面板中旋钮的大小、位置、刻度等。在 LabWindows/CVI 中,可通过控件属性对话框来设置属性。

对于 LabWindows/CVI 中的大部分控件,都有如下的属性设置。

- ① 控件名称的设置。
- ② 控件事件响应函数的设置。
- ③ 控件外观的设置。
- ④ 文字标签的设置。

(5) 对象的事件和回调函数 每一个控件对象都有其相应的响应事件,如双击鼠标、拖动窗口、点击按钮等。在 LabWindows/CVI 中,每个事件对应一个回调函数,当事件发生时,相应的回调函数被激活,由回调函数来完成控件相应的功能,从而达到预定的结果。

1.2.5 LabWindows/CVI 的基本编程窗口

LabWindows/CVI 开发平台是交互式集成开发平台,图形化用户界面。其编程环境主要由源代码窗口、用户界面编辑窗口以及函数面板这三部分组成。

(1) 用户界面编辑窗口 用户界面编辑窗口是用来创建、编辑 GUI (用户界面) 的面板、控件和菜单。一般情况下,一个用户界面至少要有有一个面板。用户界面编辑窗口可以创建面板和控件及设置各种属性,在短时间里建立符合要求的高质量图形界面。

在用户界面编辑窗口,右击鼠标将出现弹出式菜单。当鼠标点击在用户界面编辑窗口的背景上时,则弹出式菜单中包含创建面板和菜单的菜单项;当鼠标点击在面板背景上时,弹出式菜单将出现创建控件的菜单项;当鼠标点击在控件上时,将出现生成和查看控件回调函数的菜单项。图形用户界面编辑窗口如图 1-4 所示。

① File 菜单。File 菜单用于完成对工作空间文件、工程文件、C 源代码文件、头文件、用户界面文件及函数面板文件的新建、打开、保存、另存等功能,同时还具有保存全部文件、自动保存工作空间、设置当前工程、最近打开文件及退出环境的功能,其菜单如图 1-5 所示。

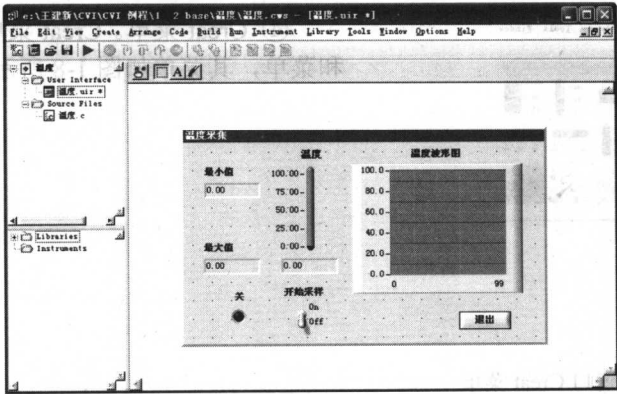


图 1-4 用户界面编辑窗口

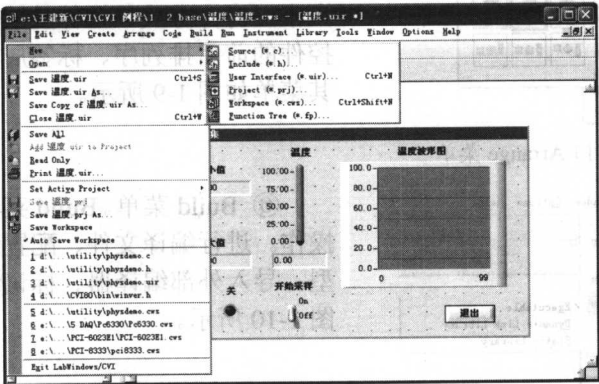


图 1-5 用户界面窗口 File 菜单

② Edit 菜单。Edit 菜单用于完成对工作空间的编辑、工程编辑、向当前工程添加文件、撤消操作、重复操作、剪切、复制、粘贴、删除/删除面板、复制面板、控件编辑、菜单编辑等，其菜单如图 1-6 所示。

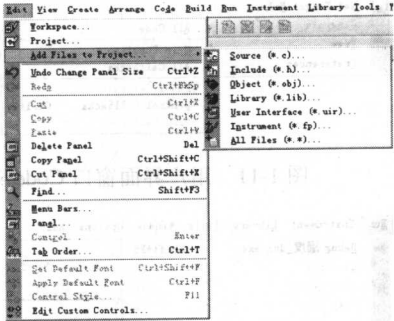


图 1-6 用户界面窗口 Edit 菜单

③ View 菜单。View 菜单用于定制是否显示工程目录区、函数目录区、工具栏及窗口排列方式，并且对于不同的编辑窗口，其他菜单项会有所不同。在用户界面编辑窗口为当前激活窗口的状态下，其菜单如图 1-7 所示。

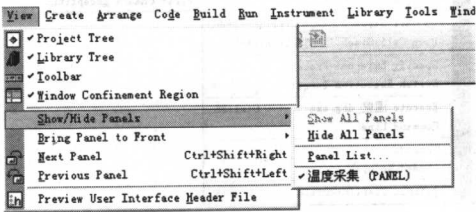


图 1-7 用户界面窗口 View 菜单

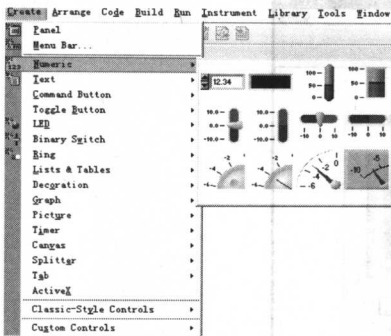


图 1-8 用户界面窗口 Create 菜单

④ Create 菜单。Create 菜单用于创建面板、控件和菜单，其菜单如图 1-8 所示。

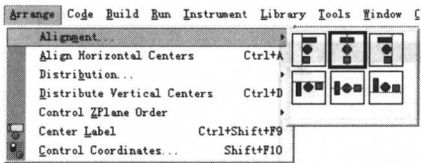


图 1-9 用户界面窗口 Arrange 菜单

⑤ Arrange 菜单。Arrange 菜单用于控件位置、大小、对齐方式、控件叠放顺序的调节。能实现对控件的前后排序、标签居中、对齐控件的功能，其菜单如图 1-9 所示。

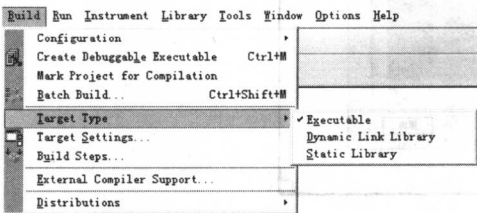


图 1-10 用户界面窗口 Build 菜单

⑥ Build 菜单。Build 菜单用于完成相关的编译操作。进行编译文件、配置编译文件、配置编译类型、导入外部编译器、标记编译文件等，其菜单如图 1-10 所示。

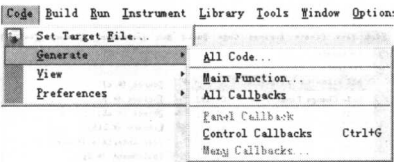


图 1-11 用户界面窗口 Code 菜单

⑦ Code 菜单。Code 菜单用于程序源代码的产生，选择所需的事件消息类型，查看控件的回调函数及事件设置。利用 LabWindows/CVI 的代码编辑器，可以根据创建的用户界面文件自动产生 C 源代码。选择 Code→Generate→All Code，LabWindows/CVI 将在 C 源文件中写入头文件、变量声明、回调函数框架及主函数。每个控件函数框架中含有一个 switch 结构，在每个结构中都包含指定默认事件的 case 声明，其菜单如图 1-11 所示。

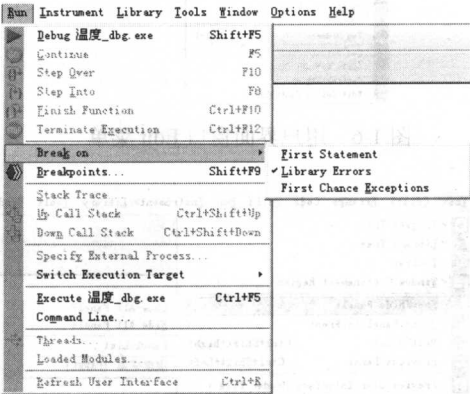


图 1-12 用户界面窗口 Run 菜单

⑧ Run 菜单。Run 菜单用于执行程序、调试程序、设置断点、单步执行、终止执行等，而且可以设置错误的中断方式，其菜单如图 1-12 所示。