



21 世纪高等院校电气工程与自动化规划教材
21 century institutions of higher learning materials of Electrical Engineering and Automation Planning

The Program Design and Application
of LabVIEW (2nd Edition)

LabVIEW 虚拟仪器 程序设计及应用 (第2版)

孙秋野 吴成东 黄博南 编著

 人民邮电出版社
POSTS & TELECOM PRESS



21 世纪高等院校电气工程与自动化规划教材
21 century institutions of higher learning materials of Electrical Engineering and Automation Planning

The Program Design and Application
of LabVIEW (2nd Edition)

LabVIEW 虚拟仪器 程序设计及应用 (第2版)

孙秋野 吴成东 黄博南 编著

人民邮电出版社
北京

图书在版编目(CIP)数据

LabVIEW虚拟仪器程序设计及应用 / 孙秋野, 吴成东, 黄博南编著. — 2版. — 北京: 人民邮电出版社, 2015.6

21世纪高等院校电气工程与自动化规划教材

ISBN 978-7-115-38784-4

I. ①L… II. ①孙… ②吴… ③黄… III. ①软件工具—程序设计—高等学校—教材 IV. ①TP311.56

中国版本图书馆CIP数据核字(2015)第066917号

内 容 提 要

本书以 LabVIEW 2013 简体中文版为对象, 通过理论与实例相结合的方式, 深入浅出地介绍了 LabVIEW 的使用方法及应用技巧。全书共分为 14 章, 内容包括 LabVIEW 概述、LabVIEW 2013 开发环境、LabVIEW 程序对象的基本操作、LabVIEW 的数据类型与程序控制、创建子 VI、操作界面的设计、字符串的实现、文件 I/O、图形与图表、访问数据库、数据采集、仪器控制与网络通信、LabVIEW 常用外部接口和上机练习。为使读者更加方便快捷地掌握 LabVIEW 的编程方法, 本书各章都附有一定数量的示例程序。

本书可作为高等院校相关课程的教材, 也可作为相关应用和技术研发人员的参考用书。

-
- ◆ 编 著 孙秋野 吴成东 黄博南
责任编辑 邹文波
执行编辑 税梦玲
责任印制 沈 蓉 彭志环
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京隆昌伟业印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 15.25 2015 年 6 月第 2 版
字数: 398 千字 2015 年 6 月北京第 1 次印刷
-

定价: 36.80 元

读者服务热线: (010) 81055256 印装质量热线: (010) 81055316

反盗版热线: (010) 81055315

第2版 前言

本书第一版出版至今已过去六个年头，在这几年里，电子技术、计算技术和网络技术得到了高速发展。LabVIEW 2013 中文版是 NI 发布的最新中文版本。相对于第一版中的 LabVIEW 8.5 中文版来说，LabVIEW 2013 拥有更加强大的功能。LabVIEW 2013 的发布大大缩短了软件易用性和强大功能之间的差距，为工程师提供了效率与性能俱佳的出色的开发平台。为了能够使读者更好地学习和掌握 LabVIEW 2013 中文版，我们决定对《LabVIEW 虚拟仪器程序设计及应用》一书进行再版修订。此次修订同样从入门的角度循序渐进地讲解 LabVIEW 2013 的基本操作，通过理论与实例相结合的方式，深入浅出地介绍 LabVIEW 的使用方法和技巧，并通过大量翔实的例子给出具体的创建过程和程序运行过程。

由于软件版本的重大革新，软件功能有很多提升，因此本次修订与第一版会有很多不同，主要体现在以下两个方面。

第一，章节安排做了重大调整，章节内容进行了一定程度上的删减。作为一本入门级教材，为了便于读者更好地学习和掌握 LabVIEW 2013 中文版，我们特意对原有的章节安排进行了调整。与此同时，相比于上一版，LabVIEW 2013 中文版在创建、调试和发布 Web 服务虚拟仪器等很多方面做出了改进，增加了很多新特性，使得操作更加简便易行、更加人性化。

第二，为了使读者能够更好地理解和掌握所讲解的内容，本版特意在上一版的基础上，增加和更改了大量实例。同时，考虑到本书为入门级教材，因此在实例选择上也秉持着基础性原则，主要目的仍是希望能够通过实例让读者更好地理解和掌握基本操作及应用。

在本书的修订过程中，杨珺、刘鑫蕊、马大中、王智良等老师给予了很多帮助。博士研究生滕菲、周建国、李玉帅等提供了软件及硬件方面的很多资料。硕士研究生沈政委、谢志远等也参与了大量的工作，提出了很多宝贵的建议，并帮助我们解决了很多具体问题。张艺缤、韩仁科、李大双、刘玲、王冰玉、陈思、甘俞乾、陈磊、赵霖等硕士研究生对全稿进行了仔细的校对，并测试了书中的示例程序。在此，一并向他们表示感谢。

最后，借此修订版出版之机，编者谨向本书第一版广大读者，尤其是对其提出批评建议的读者致以衷心的感谢。

由于时间仓促，加上编者水平有限，修订后的本书仍难以尽善，错误和不妥之处仍在所难免，恳请广大读者批评指正。

编者

2014年11月

目 录

第 1 章 LabVIEW 概述	1
1.1 LabVIEW 的起源与发展.....	1
1.2 LabVIEW 概述.....	2
1.3 LabVIEW 的工程应用.....	3
1.3.1 LabVIEW 的优势.....	3
1.3.2 LabVIEW 的应用.....	3
本章小结.....	4
第 2 章 LabVIEW 2013 开发环境 ... 5	5
2.1 LabVIEW 系统安装.....	5
2.2 LabVIEW 启动.....	7
2.3 LabVIEW 编辑界面及系统菜单.....	9
2.3.1 LabVIEW 的编辑界面.....	9
2.3.2 LabVIEW 菜单栏.....	9
2.4 LabVIEW 选板.....	15
2.4.1 控件选板.....	15
2.4.2 函数选板.....	17
2.4.3 工具选板.....	18
2.5 LabVIEW 帮助系统.....	19
本章小结.....	20
习题.....	20
第 3 章 基本操作—— 创建、编辑与调试	21
3.1 LabVIEW 的基本概念.....	21
3.2 VI 的创建.....	21
3.2.1 VI 的创建方法.....	21
3.2.2 VI 的创建实例.....	22
3.3 VI 的编辑.....	24
3.3.1 对象的放置与选择.....	25
3.3.2 对象的移动、复制与删除.....	25
3.3.3 对象大小的调整.....	26
3.3.4 对象的对齐、分布.....	26
3.3.5 控件属性的设置.....	27
3.3.6 连线的编辑.....	28
3.3.7 对象的着色与字体选择.....	31
3.4 VI 的调试.....	32
3.4.1 程序的运行.....	32
3.4.2 错误信息.....	32
3.4.3 程序的加亮执行.....	33
3.4.4 程序的单步执行.....	33
3.4.5 设置断点与探针.....	33
3.4.6 VI 程序调试技巧.....	34
本章小结.....	35
习题.....	35
第 4 章 数据类型与程序控制	36
4.1 数据类型及其操作.....	36
4.1.1 数值型.....	36
4.1.2 布尔型.....	41
4.1.3 字符串与路径.....	42
4.2 LabVIEW 的数据结构.....	44
4.2.1 数组.....	44
4.2.2 簇.....	51
4.3 程序控制.....	55
4.3.1 循环控制.....	55
4.3.2 顺序结构.....	57
4.3.3 条件结构.....	60
4.3.4 Case 条件结构示例.....	62
4.3.5 事件结构.....	62
4.3.6 自动索引、移位寄存器和 反馈节点.....	67
4.3.7 局部变量与全局变量.....	68
4.3.8 公式节点与脚本.....	72
4.4 不同类型函数的综合应用.....	76
本章小结.....	77
习题.....	77
第 5 章 创建子 VI	79
5.1 子 VI 的概念.....	79
5.2 连接器和图标.....	79

5.2.1 图标的创建和编辑.....	79	7.2 字符串函数.....	114
5.2.2 连接器端口的设置.....	80	本章小结.....	118
5.3 子VI的创建.....	82	习题.....	118
5.3.1 现有VI创建成子VI.....	83	第8章 文件I/O..... 119	
5.3.2 选定内容创建成子VI.....	84	8.1 文件的类型.....	119
5.4 添加子VI至用户库.....	85	8.2 文件I/O选板.....	120
5.5 子VI的调用.....	86	8.3 常用文件类型.....	122
5.6 VI的层次结构.....	87	8.3.1 文本文件.....	122
本章小结.....	88	8.3.2 电子表格文件.....	124
习题.....	89	8.3.3 二进制文件.....	125
第6章 操作界面的设计..... 90		8.3.4 波形文件.....	126
6.1 VI属性的设置.....	90	8.3.5 数据记录文件.....	127
6.1.1 常规属性页.....	90	8.3.6 配置文件.....	129
6.1.2 内存属性页.....	91	8.3.7 XML文件.....	129
6.1.3 说明信息属性页.....	91	8.3.8 图形文.....	131
6.1.4 修订历史属性页.....	91	8.3.9 文件I/O Express VI.....	131
6.1.5 编辑器选项属性页.....	92	本章小结.....	132
6.1.6 保护属性页.....	92	习题.....	132
6.1.7 窗口外观属性页.....	92	第9章 图形与图表..... 133	
6.1.8 窗口大小属性页.....	93	9.1 图形控件的分类.....	133
6.1.9 窗口运行时位置属性页.....	94	9.2 波形图表.....	134
6.1.10 执行属性页.....	94	9.2.1 波形图表外观与属性的设置.....	134
6.1.11 打印选项属性页.....	95	9.2.2 单曲线波形图表.....	139
6.2 用户菜单的设计.....	95	9.2.3 多曲线波形图表.....	139
6.2.1 菜单编辑器.....	95	9.3 波形图.....	140
6.2.2 菜单函数选板.....	97	9.3.1 单曲线波形图.....	140
6.2.3 用户菜单编程实例.....	99	9.3.2 多曲线波形图.....	140
6.3 属性节点.....	102	9.3.3 游标图例的设置.....	142
6.3.1 属性节点的直接创建法.....	102	9.4 XY图.....	143
6.3.2 属性节点的编程创建法.....	105	9.4.1 XY图绘制曲线.....	143
6.3.3 属性节点使用实例.....	106	9.4.2 Express XY图.....	144
6.4 控件定制方法.....	107	9.5 强度图.....	144
本章小结.....	110	9.6 数字波形图.....	146
习题.....	110	9.7 三维图形.....	147
第7章 字符串的实现..... 111		9.7.1 三维曲面图.....	147
7.1 字符串型数据.....	111	9.7.2 三维参数图.....	149
7.1.1 字符串控件.....	111	9.7.3 三维曲线图.....	150
7.1.2 列表与表格控件.....	113	本章小结.....	150
		习题.....	150

第 10 章 访问数据库	152	12.5.1 TCP	200
10.1 LabVIEW 访问数据库的途径	152	12.5.2 UDP	204
10.2 SQL 数据库语言	152	12.5.3 UDP 和 TCP 的比较	206
10.2.1 SQL 基础知识	153	12.6 DataSocket 通信	206
10.2.2 常用 SQL 编程语句	153	12.6.1 概述	206
10.3 Microsoft ADO	156	12.6.2 Datasocket 技术在 LabVIEW 中的实现	209
10.3.1 Microsoft ADO 基础知识	156	本章小结	211
10.3.2 Microsoft ADO 的对象模型	157	习题	211
10.4 通过 LabSQL 访问数据库	162	第 13 章 LabVIEW 常用外部 接口	212
10.4.1 LabSQL 工具包概述	162	13.1 调用库函数	212
10.4.2 数据源的两种创建方法	162	13.1.1 DLL 简介	212
10.4.3 数据源的三种连接方式	164	13.1.2 API 简介	213
10.4.4 LabSQL 中的数据操作	166	13.1.3 库函数的调用	213
10.5 通过 ADO 控件访问数据库	168	13.1.4 调用库函数示例	215
本章小结	170	13.2 Matlab 接口	215
习题	170	13.2.1 Matlab Script 节点	215
第 11 章 数据采集	171	13.2.2 Matlab Script 节点示例	216
11.1 DAQ 系统概述	171	13.3 代码接口节点 (CIN)	216
11.1.1 DAQ 系统的构成	171	13.4 ActiveX	220
11.1.2 DAQ 系统的功能	172	13.4.1 ActiveX 简介	220
11.2 数据采集卡的安装	175	13.4.2 ActiveX 控件	221
11.3 DAQ VI 的组织结构	179	13.4.3 ActiveX 自动化	222
11.4 数据采集助手	180	本章小结	223
本章小结	184	习题	224
习题	184	第 14 章 上机练习	225
第 12 章 仪器控制与网络通信	185	练习一	225
12.1 常用总线及总线模块	185	练习二	227
12.1.1 GPIB	185	练习三	228
12.1.2 VXI	187	练习四	229
12.1.3 PXI	190	练习五	231
12.1.4 PCI 技术	193	练习六	232
12.1.5 总线平台的比较	195	练习七	232
12.2 仪器驱动程序	196	练习八	233
12.3 串口通信	197	参考文献	235
12.4 VISA 编程	199		
12.5 TCP 与 UDP 通信	200		

第 1 章

LabVIEW 概述

1.1 LabVIEW 的起源与发展

LabVIEW 的全称为 Laboratory Virtual Instrument Engineering Workbench (实验室虚拟仪器集成环境), 是由美国国家仪器公司 (National Instruments, NI) 创立的一种功能强大而又灵活的仪器和分析软件应用开发工具。它是一种基于图形化的、用图标来代替文本行创建应用程序的计算机编程语言。在以 PC 为基础的测量和工控软件中, LabVIEW 的市场普及率仅次于 C++/C 语言。LabVIEW 已经广泛地被工业界、学术界和研究实验室所接受, 并被公认为是标准的数据采集和仪器控制软件。

LabVIEW 使用的编程语言通常称为 G 语言。G 语言与传统文本编程语言的主要区别在于: 传统文本编程语言是根据语句和指令的先后顺序执行, 而 LabVIEW 则采用数据流编程方式, 程序框图中节点之间的数据流向决定了程序的执行顺序。G 语言用图标表示函数, 用连线表示数据流向。

在 20 世纪 80 年代初个人计算机出现之前, 几乎所有拥有程控仪器的实验室都采用贵重的仪器控制器来控制测试系统。这些功能单一、价格昂贵的仪器控制器通过一个集成通信口来控制 IEEE-488 总线仪器 (也称为 GPIB 程控仪器)。后来随着 PC 的出现, 通过性价比比较高的通用 PC 控制台式仪器逐渐成为行业主流, 各种基于 PC 的接口板卡产品也逐渐占据市场, NI 公司也应运而生。

LabVIEW 的概念来源于特鲁查德和柯德斯凯两人于 20 世纪 70 年代末期在应用研究实验室 (Applied Research Laboratory, ARL) 完成的一个大型测试系统, 该系统主要用于测试美国海军的声纳探测器。通过几年的时间, 柯德斯凯把从该测试系统得到的启示应用到测试系统软件从而引入了多层虚拟仪器 (Virtual Instruments, VI) 构成的新概念。1984 年, NI 公司投资启动 LabVIEW 软件工程项目, 经过两年的努力, 于 1986 年 10 月正式发布了 LabVIEW 1.0 版。为解决内存复用问题, NI 在 1988 年推出了 LabVIEW 2.0。该版本采用了最新的面向对象编程 (OOP) 技术, 程序的执行速度和应用的灵活性达到了一个新的高度。LabVIEW 2.0 以前的版本都是运行在 Macintosh 平台上, 当 Windows 3.0 操作系统出现, 32 位的 Windows 应用程序设计成为可能后, LabVIEW 实现了从 Macintosh 平台到 Windows 平台的移植。

1992 年 8 月, 跨平台的 LabVIEW 2.5 问世。1993 年 1 月, 增加了大量新特性的 LabVIEW 3.0 正式发行, 这些新特性包括全局与局部变量、属性节点和执行动画等。此后, 每一个重大版本的

发布都包括了里程碑意义的特性和功能上的飞跃。1998 年发布的 LabVIEW 5.0 已经提供了多线程支持功能,为现在的多处理器技术打下了基础。1999 年 LabVIEW 的首个实时系统版本诞生;2003 年 LabVIEW 7 Express 引入了波形数据类型以及一些交互性更强的、基于配置的函数;2005 年推出的 LabVIEW 8 实现了分布式智能;2006 年发布的 LabVIEW 8.2 提供了仿真框图和 MathScript 节点功能,同时第一次推出了简体中文版本;2007 年 NI 发布了 LabVIEW 8.5 版本;2013 年 8 月推出了最新的版本 LabVIEW 2013。2013 版最新的 LabVIEW 主要关注以下三个方面:集成最新技术帮助用户开发更高性能的系统,改善开发环境帮助开发者提高效率,提供系统的培训和众多联盟商工具。

1.2 LabVIEW 概述

LabVIEW 以图形语言 (G 语言),用图标和连线代替文本的形式编写程序。和 VC、VB 等高级语言一样,LabVIEW 也是一种带有扩展库函数的通用程序开发系统。LabVIEW 的库函数包括数据采集、通用接口总线 (General Purpose Interface Bus, GPIB) 和串口仪器控制,数据显示、分析与存储等。为了便于程序调试,LabVIEW 还带有传统的程序开发调试工具,例如可以设置断点,可以单步执行,也可以激活程序的执行过程,以动画方式查看数据在程序中的流动。

LabVIEW 是一个通用编程系统,它不但能够完成一般的数学运算与逻辑运算和输入输出功能,还带有专门的用于数据采集和仪器控制的库函数和开发工具,尤其带有专业的数学分析程序包,基本上可以满足复杂的工程计算和分析要求。LabVIEW 环境下开发的程序称为虚拟仪器 (Virtual Instruments, VI),因为它的外型与操作方式可以模拟实际的仪器。实际上,VI 类似于传统编程语言的函数或子程序。

VI 由一个用户界面、图标代码和一个接口板组成。接口板用于上层 VI 调用该 VI。VI 具有以下特点。

(1) 用户界面由于类似于仪器的面板也叫作前面板。前面板包括旋钮、按钮、图形和其他控制元件与显示元件以完成用鼠标、键盘向程序输入数据或从计算机显示器上观察结果。

(2) VI 用图标代码和连线来完成算术和逻辑运算。图标代码是对具体编程问题的图形解决方案。图标代码即 VI 的源代码。

(3) VIs 具有层次结构和模块化的特点。它们可以作为顶层程序,也可以作为其他程序的子程序。VI 代码内含的 VI 叫子程序 subVI。

(4) VI 程序使用接口板来替代文本编程的函数参数表,每个输入和输出的参数都有自己的连接端口,其他的 VIs 可以由此向 subVI 传递数据。

由于这些特色,LabVIEW 符合模块化的程序设计概念并对这种概念起到了推进作用。我们把一个复杂的应用程序逐步划分为一系列简单的子任务,为每一个子任务创建一个 VI,再把它们装配到另一个图标代码中完成一个复杂的任务。最终,顶层的 VI 包含着一系列 VIs,它们分别代表着应用程序的功能。

由于每一个 subVI 都可以单独执行,使得程序调试非常方便。此外,许多低层 subVIs 可以完成不同应用软件的通用功能,所以可以为将要构建的应用软件开发一系列适用的 subVIs。这些 subVIs 作为可重复利用的资源大大提高了开发效率。

总之,LabVIEW 建立在易于使用的图形数据流编程语言 G 语言上。G 语言大大简化了科学

计算、过程监控和测试软件的开发，并可以在更广泛的范围内得以应用。

1.3 LabVIEW 的工程应用

1.3.1 LabVIEW 的优势

选择 LabVIEW 进行开发测试和测量应用程序的一个决定性因素是它的开发速度。一般来说，用 LabVIEW 开发应用系统的速度要比其他的编程语言快 4~10 倍。造成这种巨大差距的主要原因在于 LabVIEW 易用易学，上手很快。

LabVIEW 的优势主要体现在以下几个方面。

(1) 提供了丰富的图形控件，采用了图形化的编程方法，把工程师从复杂枯燥的文件编程工作中解放出来；

(2) 采用数据流模型，实现了自动的多线程，从而能充分利用处理器（尤其是多处理器）的处理能力；

(3) 内建有编译器，能在用户编写程序的同时自动完成编译，因此如果用户在编写程序的过程中有语法错误，就能立即在显示器上显示出来；

(4) 通过 DLL、CIN 节点、ActiveX、.NET 或 MATLAB 脚本节点等技术，能够轻松实现 LabVIEW 与其他编程语言的混合编程；

(5) 内建了 600 多个分析函数用于数据分析和信号处理；

(6) 通过应用程序生成器可以轻松地发布可执行程序、动态链接库或安装包；

(7) 提供了大量的驱动和专用工具，几乎能够与任何接口的硬件轻松连接；

(8) NI 同时提供了丰富的附加模块，用于扩展 LabVIEW 在不同领域的应用，如实时模块、PDA 模块、数据记录与监控（DSC）模块、机器视觉模块与触摸屏模块。

1.3.2 LabVIEW 的应用

LabVIEW 在测试与测量、过程控制、工业自动化和实验室研究等方面都得到了广泛的应用。在世界范围内，汽车、通信、航空、半导体、电子设计生产、过程控制和生物医学等各领域均通过 LabVIEW 提高了应用开发的效率，LabVIEW 的应用涵盖了从研发、测试、生产到服务的产品开发所有阶段。使用 LabVIEW 可以实现和完成在任何平台上进行数据采集、仪器控制和连接、机器视觉、运动控制、模块化仪器、工业监控等众多方面的功能和任务。

LabVIEW 带有超过 450 个内置函数，专门用于从采集到的数据中挖掘有用的信息，用于分析测量数据及处理信号。LabVIEW 提供了一系列工具用于数据显示、用户界面设计、Web 信息发布、报告生成、数据管理及软件连接。只需简单地从控件选板中拖放内置的控制件和显示件，然后单击鼠标即可利用交互式的属性页面轻松地定义它们的功效和外观。

LabVIEW 将广泛的数据采集、分析与显示功能集中在了同一个环境中，可以在自己的平台上无缝地集成一套完整的应用方案。LabVIEW 是一个开放式的开发环境，用户可以将其与任何测量硬件轻松连接。LabVIEW 的交互式测量助手、自动代码生成以及与成千上万个设备的简易连接功能能够轻而易举地完成数据采集。LabVIEW 简化了与数百家仪器厂商的数千种仪器设备的连接和通信。使用 LabVIEW 中的仪器驱动程序、交互式仪器 I/O 助手（Instrument I/O Assistant）和内置

仪器 I/O 函数库, 可以从 GPIB、串口、以太网、PXI、USB 接口仪器及 VXI 仪器中快速采集数据。

LabVIEW 是一个具有高度灵活性的开发系统, 用户可以根据自己的应用领域和开发要求选择 LabVIEW 系统配置。NI 公司为不同层次的用户提供了如下 3 种系统配置。

(1) LabVIEW 基本版。LabVIEW 基本版是指用于开发数据采集和仪器控制系统的最小 LabVIEW 配置, 包括 VISA、GPIB、RS-232、DAQ 和基本分析库, 同时还包括支持 ActiveX、TCP/IP 和 DDE 等标准程序的接口。

(2) LabVIEW 完整版 (FDS)。除了基本版的功能外, FDS 还包括完整的高级分析库。

(3) LabVIEW 专业版 (PDS)。PDS 除了 FDS 功能外, 还具有专业程序员开发时所需要的全部工具, 包括: 可执行文件生成工具、源代码控制、复杂矩阵分析、软件工程文档管理、质量控制标准文档、图形差异比较和大型软件项目管理文档工具等。对一般用户而言, 采购 LabVIEW 完整版, 并根据实际应用选取专门的 LabVIEW 工具套件是最佳选择。

本章小结

LabVIEW 是一种功能强大的软件, 其图形化编程语言的出现将人们从复杂的编程工作中解放出来。本章作为 LabVIEW 的入门, 主要介绍了 LabVIEW 的起源与发展和工程应用, 使读者对 LabVIEW 有了一个基本的认识。

第 2 章

LabVIEW 2013 开发环境

2.1 LabVIEW 系统安装

LabVIEW 2013 可以安装在 Windows XP SP3 平台和 Windows 7/8/8.1 等不同的操作系统上,不同的操作系统在安装 LabVIEW 2013 时对系统配置的要求也不同。用户在安装 LabVIEW 2013 软件之前,需要对个人计算机的软硬件配置作一定的了解,该版本不支持 Windows XP SP2 版本!

对于常用的 Windows 操作系统,安装 LabVIEW 2013 的硬件配置必须满足:①处理器: Pentium III/Celeron 866 MHz 或同等性能以上处理器;②内存: 最小内存为 256MB,推荐配置为 1GB 及以上;③屏幕分辨率: 1024 像素×768 像素;④硬盘空间: 最小安装需要至少 900MB 磁盘空间,完整安装时至少需要留出 3.5GB 磁盘空间。

LabVIEW 的安装十分简单,只要运行安装光盘中的 Setup 程序,按照屏幕提示,每一步选择必要的安装选项即可。

选择安装程序后,屏幕上将会出现初始化界面,如图 2-1 所示。



图 2-1 初始化 LabVIEW 2013 的安装程序

初始化 LabVIEW 2013 的安装程序运行完毕后,系统会提示用户输入产品序列号,如图 2-2 所示。

安装 LabVIEW 2013 的试用版不需要输入序列号,试用期为 7 天,在程序编译完成后,不能打包生成独立可执行应用程序 (EXE) 和安装程序 (Installer)。若使用正版软件,输入正确序列号后,单击下一步,即可进入如图 2-3 所示的安装路径对话框。

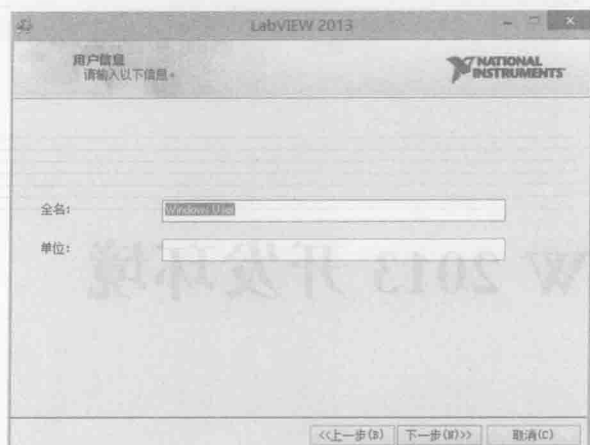


图 2-2 输入 LabVIEW 2013 用户信息的界面

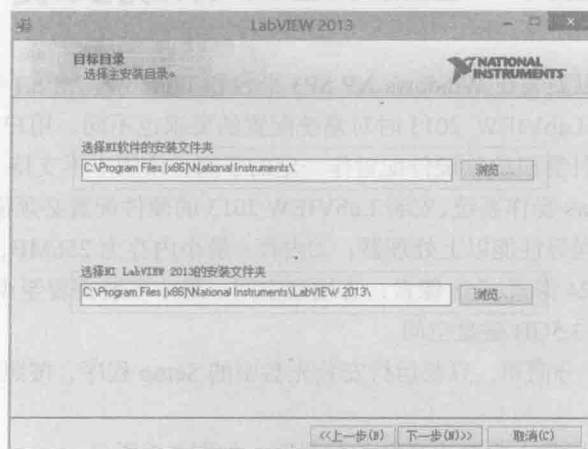


图 2-3 LabVIEW 2013 的安装路径

图 2-3 中默认的是安装路径为 C 盘，用户可以单击“浏览”按钮选择其他安装路径，然后单击“下一步”按钮进入模块安装界面，如图 2-4 所示。

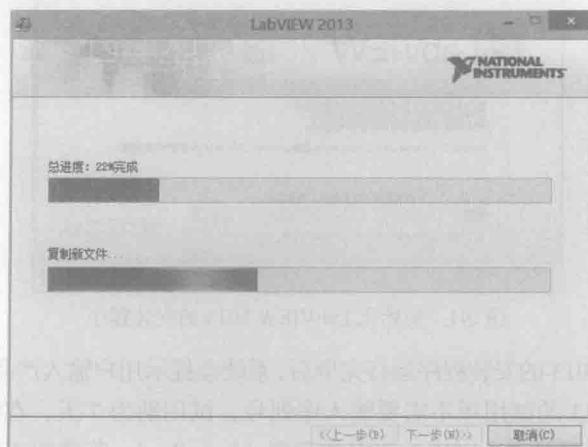


图 2-4 模块安装进度界面

安装完成后，出现如图 2-5 所示的界面，单击“完成”按钮，则完成 LabVIEW 2013 简体中文版的安装。

当完成软件的安装后，就可以使用 LabVIEW 软件了。另外，若装有 VXI、GPIB 和 DAQ 等设备，则需在 LabVIEW 系统软件安装完成后，运行专门的仪器驱动程序和 VISA 库函数的安装程序。

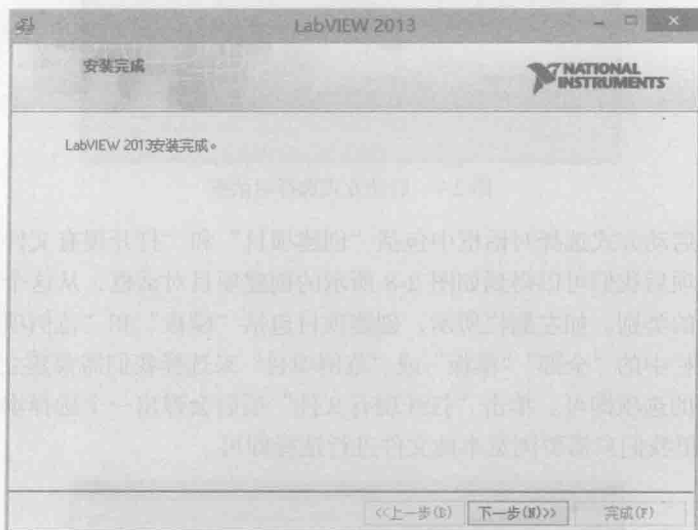


图 2-5 安装完成界面

2.2 LabVIEW 启动

我们从“开始”菜单中运行“National Instruments LabVIEW 2013”或直接在桌面上双击快捷方式图标，便可运行程序。随后计算机屏幕上将出现如图 2-6 所示的启动界面，几秒钟后跳转为如图 2-7 所示的启动方式选择对话框。

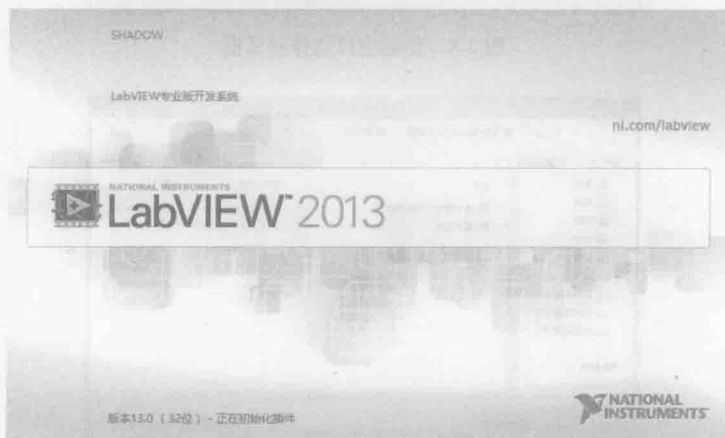


图 2-6 启动画面

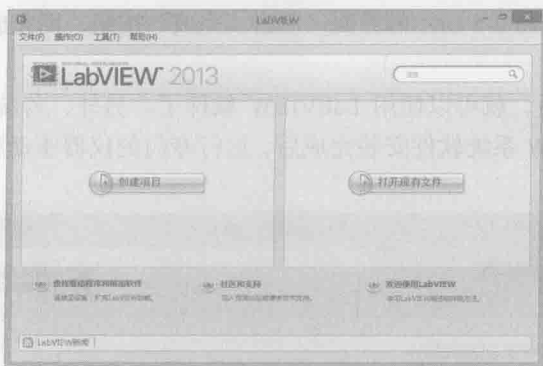


图 2-7 启动方式选择对话框

图 2-7 所示的启动方式选择对话框中包括“创建项目”和“打开现有文件”两种启动方式。单击“创建项目”项后我们可以得到如图 2-8 所示的创建项目对话框，从这个对话框里我们可以选择需要创建项目的类别，如左侧栏所示，创建项目包括“模板”和“范例项目”两大类，我们可以通过单击左侧栏中的“全部”“模板”或“范例项目”来选择我们需要建立的项目板块，然后单击右侧栏中相应的选项即可。单击“打开现有文件”项后会弹出一个选择本地文件的对话框，如图 2-9 所示，这里我们只需要浏览本地文件进行选择即可。

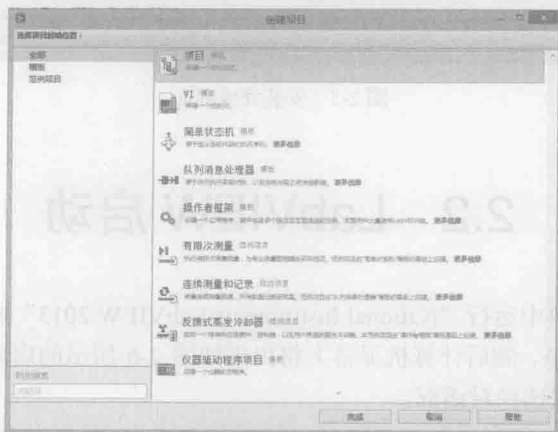


图 2-8 创建项目选择对话框

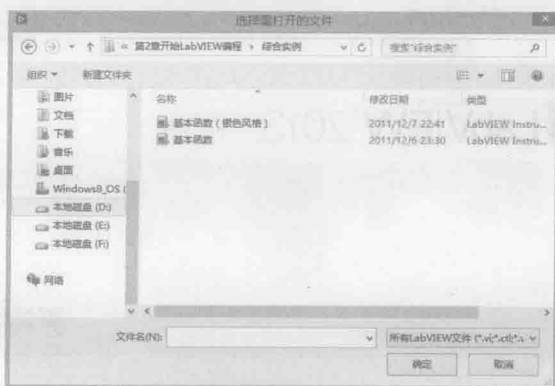


图 2-9 打开现有文件选择对话框

2.3 LabVIEW 编辑界面及系统菜单

2.3.1 LabVIEW 的编辑界面

LabVIEW 与虚拟仪器有着紧密的联系, 在 LabVIEW 中开发的程序都被称为 VI (虚拟仪器), 其扩展名默认为 .vi。所有的 VI 都包括以下 3 个部分: 前面板、程序框图和图标, 如图 2-10 所示。

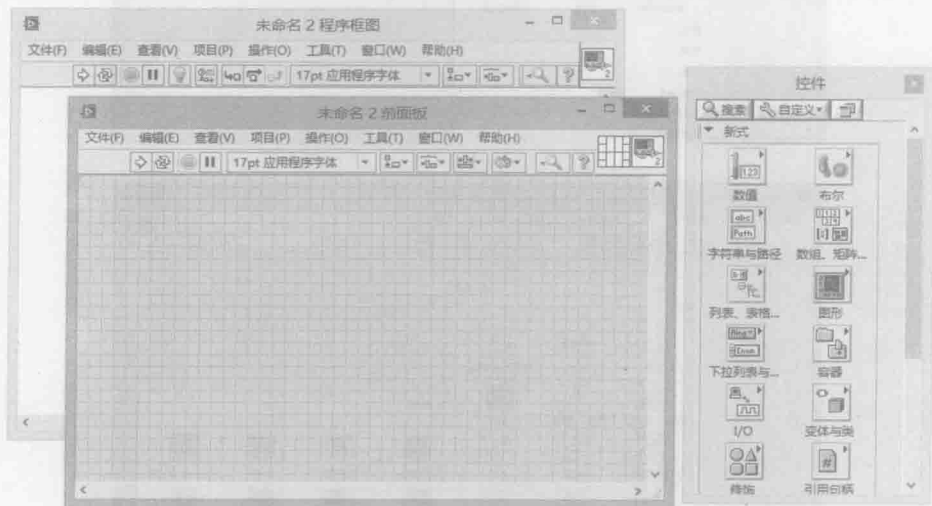


图 2-10 LabVIEW 前面板和程序框图

前面板就是图形化用户界面, 也是 VI 的前面板。该界面上有交互式的输入和输出两类对象, 分别称为控制器 (Controller) 和显示器 (Indicator)。控制器包括开关、旋钮、按钮和其他各种输入设备; 指示器包括图形 (Graph 和 Chart)、LED 和其他显示输出对象。该界面可以模拟真实仪器的前面板, 用于设置输入数值和观察输出量。

程序框图是定义 VI 逻辑功能的图形化源代码。框图中的编程元素除了包括与前面板上的控制器和显示器对应的连线端子 (Terminal) 外, 还有函数、子 VI、常量、结构和连线等。在程序框图中对 VI 编程的主要工作是从前面板上的控制器获得用户输入信息, 并进行计算和处理, 最后在显示器中反馈给用户处理结果。只要在前面板中放入输入或显示控件, 用户就可以在程序框图中看到相应的图表函数等内容。

如果将 VI 与标准仪器相比较, 那么前面板就相当于仪器面板, 而程序框图则相当于仪器箱内的功能部件。在许多情况下, 使用 VI 可以仿真标准仪器。

2.3.2 LabVIEW 菜单栏

LabVIEW 有两种类型的菜单栏: 快捷菜单和下拉菜单, 如图 2-11 和图 2-12 所示。要访问快捷菜单, 可通过在前面板或框图中任何对象上单击鼠标右键操作。这个过程也称之为“弹出”, 因此快捷菜单又可称为“弹出菜单”。

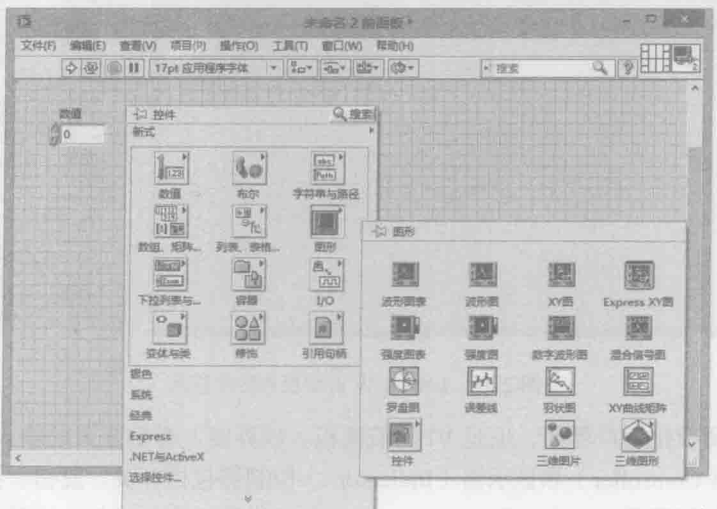
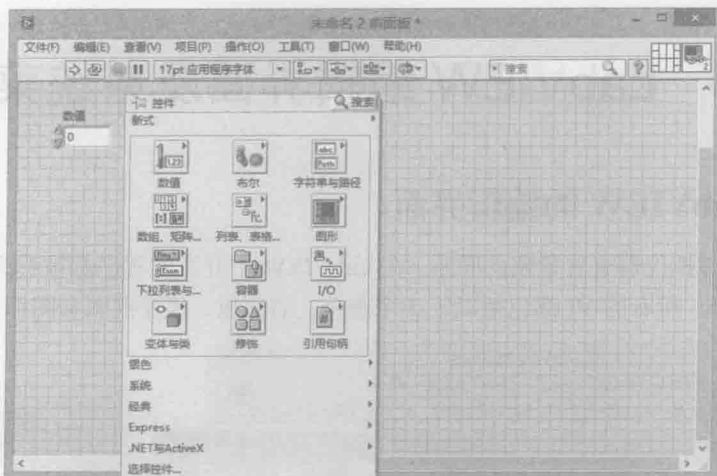


图 2-11 快捷菜单示例



图 2-12 下拉菜单及扩展子菜单示例