

 中国电子教育学会高教分会推荐
普通高等教育电子信息类“十三五”课改规划教材

LabVIEW 2015 程序设计教程

主 编 向守超 侯从喜
副主编 张国军 谢钱涛



西安电子科技大学出版社
<http://www.xduph.com>

中国电子教育学会高教分会推荐

普通高等教育电子信息类“十三五”课改规划教材

LabVIEW 2015 程序设计教程

主 编 向守超 侯从喜

副主编 张国军 谢钱涛



西安电子科技大学出版社

内 容 简 介

本书按照软件开发的设计思想,介绍了虚拟仪器的基本概念和 LabVIEW 2015 应用软件开发的基础知识,主要讲解了 LabVIEW 2015 的安装和编程环境、数据类型与基本操作、程序编辑与调试、程序结构设计、图形显示、子 VI 与属性节点、人机界面设计、文件操作、面向对象编程、数据库操作、网络通信与编程等内容,并给出了综合项目实例。书中结合大量经典案例,将知识点融入案例中,深入浅出地讲解了 LabVIEW 2015 程序设计思想的重点和难点,使初学者能快速具备使用 LabVIEW 集成开发环境设计测量系统的能力。本书内容翔实,结构合理,语言浅显,实用性较强。

本书可作为高等院校物联网工程、电子技术、自动化工程、电气通信、测量控制等相关专业的教材或参考资料,也可供 LabVIEW 入门级读者以及从事相关专业的工程项目开发人员参考。

图书在版编目(CIP)数据

LabVIEW 2015 程序设计教程/向守超,侯从喜主编. —西安:西安电子科技大学出版社,2017.9
(普通高等教育电子信息类“十三五”课改规划教材)

ISBN 978-7-5606-4620-6

I. ① L… II. ① 向… ② 侯… III. ① 软件工具—程序设计 IV. ① TP311.56

中国版本图书馆 CIP 数据核字(2017)第 214128 号

策 划 刘玉芳

责任编辑 蔡雅梅 雷鸿俊

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2017 年 9 月第 1 版 2017 年 9 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 23

字 数 548 千字

印 数 1~3000 册

定 价 45.00 元

ISBN 978-7-5606-4620-6/TP

XDUP 4912001-1

如有印装问题可调换

前 言

LabVIEW 是一种业界领先的工业标准图形化编程工具，主要用于开发测试、测量与控制系统。它是专门为工程师和科学家而设计的直观图形化编程语言，可将软件和各种不同的测量仪器硬件以及计算机集成在一起，建立虚拟仪器系统，以形成用户自定义的解决方案。LabVIEW 大大缩小了软件易用性和强大功能之间的距离，为工程师提供了具有极佳效能的真正出色的开发环境。

LabVIEW 软件不但适用于各种测量和自动化领域，而且无论工程师是否有丰富的开发经验，都能顺利应用，因此，目前 LabVIEW 已经成为大学生必修的一门基础实验课程。本书以 LabVIEW 2015 为对象，通过理论与实例相结合的方式，结合编者多年的实践开发经验，深入浅出地介绍了该软件的使用方法和技巧，目的在于让读者能快速掌握这门功能强大的图形化编程语言。

本书以软件开发设计思想为主线，本着由易到难、深入浅出的原则，按照条理清晰、内容全面、实例经典、实用性强的要求，对 LabVIEW 2015 编程环境进行了全面详细的介绍，尤其对 LabVIEW 编程人员经常讨论的热点问题进行了重点介绍。此外，本书所有的知识点都配合了恰当的实例，读者可以快速掌握很多非常实用的编程技巧，例如图表自动图例、多面板程序设计、人机界面设计等。

在本书的编写过程中，编者参考了 LabVIEW 帮助文档几乎所有的内容以及大部分现有的 LabVIEW 书籍，搜索了 NI 网站中的大量网络资源，并且总结了编者多年的 LabVIEW 编程知识和编程技巧。通过本书，读者可以从入门开始，逐步深入地学习 LabVIEW，直到成为真正精通 LabVIEW 的编程高手。本书也可作为 LabVIEW 编程知识的必备参考书籍。

本书具有以下特点：

(1) 知识讲解扎实。本书全面详细地介绍了 LabVIEW 的基本概念以及程序开发的基础知识，内容讲解翔实，实例贴切，特别适合于从事 LabVIEW 软件设计的初学者。

(2) 知识实用性强。本书中的全部实例均利用实验室内的环境进行设计与开发，且程序全部经过调试与验证。

(3) 知识覆盖面广。本书精选了若干个典型实例，内容新颖，反映了当前虚拟仪器的发展及时代的需求。

为了使初学者快速地具备使用 LabVIEW 设计测量系统的能力，本书从实用角度出发，将内容分为 12 章。

第 1 章 LabVIEW 概述：重点介绍虚拟仪器的概念和 LabVIEW 2015 集成开发环境的安装和配置。

第 2 章 数据类型与基本操作：全面而详细地介绍了 LabVIEW 2015 的数据类型和数据运算。

第 3 章 程序编辑与调试：详细介绍了 LabVIEW 2015 VI 的创建、编辑、运行、调试

和错误处理。

第4章 程序结构设计：重点介绍了顺序结构、条件结构、循环结构、公式节点、事件结构、使能结构、变量和定时结构等基本知识。

第5章 图形显示：重点介绍了波形显示、XY图与Express XY图、强度图形、数字波形图、三维图形和图形控件。

第6章 子VI与属性节点：重点介绍了子VI的创建和属性节点的应用。

第7章 人机界面设计：重点介绍了下拉列表、枚举、列表框、表格与树形、对话框、菜单、选项卡、多面板、光标等高级控件的应用和人机界面设计的基本技巧。

第8章 文件操作：重点介绍了文本文件、电子表格文件、二进制文件、波形文件、数据记录文件、测量文件、配置文件、XML文件和TDMS文件等基本文件的读/写操作技巧。

第9章 面向对象编程：重点介绍了对象的创建以及面向对象的继承、多态等特点的应用。

第10章 数据库操作：以ATM自动存取款机系统为例，通过LabVIEW 2015操作MySQL数据库，实现数据库的增、删、改、查等基本操作。

第11章 网络通信与编程：重点介绍了网络协议通信、DataSocket技术通信和远程访问技术。

第12章 综合项目实例：以双通道频谱滤波器设计为例，按照软件工程的思想，对从项目设计、开发到发布的整个过程进行了详细讲解。

本书在内容安排上循序渐进、深入浅出，力求重点突出、面向实际应用，提高读者的编程能力，并可用于解决实际问题。

本书由向守超(重庆工程学院)、侯从喜(四川省万源市罗文中心校)担任主编，张国军(重庆工程学院)、谢钱涛(重庆工程学院)担任副主编。其中，侯从喜负责第1章、第2章和第3章的编写，张国军负责第4章、第7章的编写，谢钱涛负责第5章、第6章的编写，向守超负责第8章至第12章的编写。本书的编写得到了重庆工程学院电子与物联网学院全体老师的鼓励和支持，在此向他们表示衷心的感谢！

由于编者水平有限，书中不当之处在所难免，敬请读者批评指正。

编 者

2017年6月

目 录

第 1 章 LabVIEW 概述..... 1	
1.1 虚拟仪器概述..... 1	
1.1.1 虚拟仪器的结构..... 1	
1.1.2 虚拟仪器的特点..... 2	
1.1.3 虚拟仪器的硬件..... 3	
1.1.4 虚拟仪器的软件..... 3	
1.2 LabVIEW 特点及功能..... 5	
1.2.1 LabVIEW 的特点..... 5	
1.2.2 LabVIEW 的功能..... 5	
1.2.3 LabVIEW 的应用..... 6	
1.3 LabVIEW 2015 的安装..... 7	
1.4 LabVIEW 2015 编程环境..... 11	
1.4.1 LabVIEW 2015 启动..... 11	
1.4.2 LabVIEW 2015 编程界面..... 12	
1.4.3 LabVIEW 2015 菜单栏..... 13	
1.4.4 LabVIEW 2015 工具栏..... 17	
1.4.5 LabVIEW 2015 选板..... 19	
1.4.6 LabVIEW 2015 帮助..... 22	
1.4.7 自定义编程环境..... 24	
1.5 LabVIEW 应用开发实例..... 26	
第 2 章 数据类型与基本操作..... 28	
2.1 基本数据类型..... 28	
2.1.1 数值型..... 28	
2.1.2 布尔型..... 34	
2.1.3 枚举类型..... 37	
2.1.4 时间类型..... 39	
2.1.5 变体类型..... 40	
2.2 数据运算..... 41	
2.2.1 数值函数选板..... 42	
2.2.2 布尔函数选板..... 44	
2.2.3 比较函数选板..... 45	
2.2.4 表达式节点..... 46	
2.3 数组..... 47	
2.3.1 创建一维数组..... 47	
2.3.2 创建多维数组..... 48	
2.3.3 循环结构创建数组..... 49	
2.3.4 数组函数..... 50	
2.4 字符串..... 55	
2.4.1 字符串与路径..... 55	
2.4.2 字符串数组控件..... 58	
2.4.3 字符串函数..... 59	
2.5 簇..... 68	
2.5.1 簇的创建..... 68	
2.5.2 簇函数..... 69	
2.6 矩阵..... 73	
2.6.1 矩阵的概念..... 73	
2.6.2 矩阵转置..... 74	
2.6.3 矩阵的基本运算..... 76	
2.7 波形..... 76	
2.7.1 波形数据..... 77	
2.7.2 波形操作函数..... 77	
第 3 章 程序编辑与调试..... 80	
3.1 VI 的创建..... 80	
3.1.1 前面板的创建..... 80	
3.1.2 程序框图的创建..... 82	
3.1.3 图标的创建..... 82	
3.2 VI 的编辑..... 83	
3.2.1 选择对象..... 83	
3.2.2 移动对象..... 84	
3.2.3 复制和删除对象..... 84	
3.2.4 对齐和分布对象..... 85	
3.2.5 调整对象大小..... 86	
3.2.6 重新排序..... 87	
3.2.7 修改对象外观..... 87	
3.2.8 连线..... 89	
3.3 VI 的运行和调试..... 90	

3.3.1 运行调试工具	90	4.8 定时结构	142
3.3.2 高亮显示执行过程	91	4.8.1 定时循环	142
3.3.3 探针与断点诊断	92	4.8.2 定时顺序	144
3.3.4 VI 程序调试技巧	96	4.8.3 定时 VI	145
3.4 错误处理	97	第 5 章 图形显示	146
3.4.1 错误簇	98	5.1 波形显示	146
3.4.2 错误捕获	100	5.1.1 波形图	147
3.4.3 错误报告	102	5.1.2 波形图显示实例	152
3.4.4 错误处理	103	5.1.3 波形图表	155
第 4 章 程序结构设计	106	5.1.4 波形图表实例	158
4.1 顺序结构	106	5.2 XY 图与 Express XY 图	160
4.1.1 平铺式顺序结构	107	5.2.1 XY 图	160
4.1.2 层叠式顺序结构	108	5.2.2 利萨如图的绘制	161
4.1.3 顺序结构之间的转换	108	5.2.3 XY 图实例	162
4.2 条件结构	110	5.2.4 Express XY 图	164
4.2.1 条件选择器标签	110	5.3 强度图形	165
4.2.2 分支选择器	112	5.3.1 强度图	165
4.2.3 分支子程序框	113	5.3.2 强度图表	167
4.3 循环结构	114	5.4 数字波形图	168
4.3.1 For 循环	115	5.5 三维图形	170
4.3.2 While 循环	116	5.5.1 三维曲面图	171
4.3.3 循环结构数据通道与自动索引	117	5.5.2 三维参数图	172
4.3.4 移位寄存器	118	5.5.3 三维曲线图	174
4.3.5 反馈节点	119	5.6 图形控件	175
4.3.6 循环结构应用实例	119	5.6.1 极坐标图	176
4.4 公式节点	122	5.6.2 最小-最大曲线显示控件	177
4.4.1 公式节点变量	122	第 6 章 子 VI 与属性节点	179
4.4.2 公式节点运算符和函数	123	6.1 子 VI	179
4.5 事件结构	124	6.1.1 创建子 VI	179
4.5.1 事件选择器标签	125	6.1.2 定义子 VI 属性	181
4.5.2 事件数据节点	126	6.2 属性节点	184
4.5.3 事件结构的设置	127	6.3 调用节点和引用句柄	187
4.5.4 在 LabVIEW 中使用事件	130	6.4 属性节点应用实例	189
4.6 使能结构	132	6.4.1 进度条	189
4.6.1 程序框图禁用结构	132	6.4.2 单选按钮	189
4.6.2 条件禁用结构	133	第 7 章 人机界面设计	192
4.7 变量	135	7.1 下拉列表控件和枚举控件	192
4.7.1 局部变量	136	7.2 列表框控件	194
4.7.2 全局变量	139	7.2.1 单列列表框	194

7.2.2 多列列表框	196
7.3 表格与树形控件	198
7.3.1 表格	198
7.3.2 树形控件	199
7.4 VI 属性设置	202
7.5 对话框	208
7.5.1 普通对话框	209
7.5.2 自定义对话框	210
7.6 菜单	211
7.6.1 菜单函数	211
7.6.2 运行主菜单	212
7.6.3 右键快捷菜单	217
7.7 选项卡	219
7.8 多面板	221
7.9 光标	223
7.10 自定义控件和数据类型	224
7.10.1 自定义控件	224
7.10.2 自定义数据类型	225
7.10.3 自定义控件选板和函数选板	226
7.11 用户界面设计	227
7.11.1 修饰静态界面	227
7.11.2 动态交互界面	228
7.12 VI 程序设计规则	229
7.12.1 关于前面板的设计	230
7.12.2 关于程序框图的设计	232
7.12.3 关于 VI	232
第 8 章 文件操作	234
8.1 文件操作基本术语	234
8.1.1 文件的基本概念	234
8.1.2 文件的基本类型	235
8.1.3 文件操作基本函数	236
8.2 文本文件	238
8.3 电子表格文件	240
8.4 二进制文件	242
8.5 波形文件	244
8.6 数据记录文件	246
8.7 测量文件	248
8.8 配置文件	251
8.9 XML 文件	253

8.10 TDMS 文件	254
第 9 章 面向对象编程	260
9.1 面向对象的基本概念	260
9.2 类的创建	261
9.3 对象的创建	264
9.4 继承	266
9.5 多态	268
9.6 动态方法	269
9.7 LabVIEW 面向对象的其他特点	272
9.7.1 构造函数与析构函数	272
9.7.2 对象数据文件操作	272
9.8 两种编程方法的比较	273
9.8.1 测试目的	274
9.8.2 面向过程的方法	274
9.8.3 面向对象的方法	275
9.8.4 两种方法的比较	276
第 10 章 数据库操作	277
10.1 MySQL 基础	277
10.1.1 MySQL 下载与安装	277
10.1.2 启动 MySQL 服务	281
10.1.3 MySQL 常用图形管理工具	283
10.2 LabSQL 简介	284
10.3 数据源配置	285
10.3.1 ODBC 简介	285
10.3.2 ADO 简介	286
10.3.3 建立数据源	287
10.4 数据库操作	289
10.4.1 ATM 登录功能	290
10.4.2 ATM 查询功能	292
10.4.3 ATM 注册功能	295
10.4.4 ATM 存取款功能	297
10.4.5 修改密码功能	299
10.4.6 注销用户功能	301
第 11 章 网络通信与编程	304
11.1 网络协议通信	304
11.1.1 TCP 通信	305
11.1.2 UDP 通信	310
11.1.3 SMTP Email	313
11.1.4 IrDA 技术	314

11.1.5 蓝牙技术	316	12.2 项目设计目的	340
11.2 DataSocket 技术通信	318	12.3 项目功能模块介绍	341
11.2.1 DataSocket 服务管理器	318	12.4 项目功能设计实现	342
11.2.2 DataSocket 服务器	319	12.4.1 滤波信号的产生	342
11.2.3 DataSocket 应用程序接口	320	12.4.2 添加滤波器 VI 控件	343
11.3 远程访问	323	12.4.3 双通道谱测量	344
11.3.1 配置服务器	323	12.4.4 检测信号	346
11.3.2 远程面板控制	324	12.4.5 各 VI 控件之间数据流的连线	347
11.3.3 浏览器访问	325	12.4.6 各 VI 信号的图形显示控件	348
第 12 章 综合项目实例	328	12.4.7 创建上下截止频率调节控件	349
12.1 软件工程基础	328	12.4.8 信号掩区和边界测试输出信号的 显示设置	350
12.1.1 软件工程的概 念	328	12.4.9 创建截止频率变化子 VI	353
12.1.2 软件生命周期	329	12.5 发布应用程序	355
12.1.3 软件开发模型	330	12.5.1 独立可执行程序(EXE)	355
12.1.4 软件需求分析	334	12.5.2 安装程序	357
12.1.5 软件设计	335	参考文献	360
12.1.6 程序编码	337		
12.1.7 软件测试	338		
12.1.8 软件实施和维护	339		

第1章

LabVIEW 概述

本章首先从虚拟仪器引出 LabVIEW 软件，然后逐一介绍 LabVIEW 语言的特点及其功能、LabVIEW 的安装及编程环境、LabVIEW 应用开发实例，使得读者能从整体上认识并学会应用 LabVIEW 软件。

1.1 虚拟仪器概述

虚拟仪器(VI)的概念是美国国家仪器公司(National Instruments, NI)1986 年提出的，是现代计算机技术和仪器技术深层次结合的产物，也是计算机辅助测试领域的一项重要技术。虚拟仪器引发了传统仪器领域的一场重大变革，使计算机和网络技术得以深入仪器领域，虚拟仪器和仪器技术的结合，开创了“软件就是仪器”的先河。

1.1.1 虚拟仪器的结构

虚拟仪器是基于计算机的仪器，是计算机与仪器密切结合的产物。粗略地说，这种结合有两种方式：一种是将计算机装入仪器，其典型的例子就是智能化仪器；另一种是将仪器装入计算机，以通用的计算机硬件和操作系统为依托，实现各种仪器功能，虚拟仪器主要指这种结合方式产生的仪器。从实质上讲，虚拟仪器就是利用硬件系统完成信号的采集、测量与调理，利用计算机强大的软件功能实现信号数据的运算、分析和处理，利用计算机的显示器模拟传统仪器的控制面板，以多种形式输出检测结果，从而完成所选的各种测试功能。

虚拟仪器可以通过多种接口(GPIB、VXI、PXI 等)或具有这些接口的仪器来连接被测控对象和计算机。虚拟仪器结构图如图 1.1 所示。

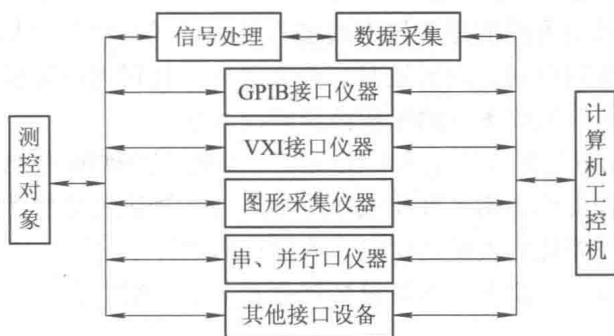


图 1.1 虚拟仪器结构图

虚拟仪器系统包括仪器硬件和应用软件两大部分。仪器硬件是计算机的外围电路，与计算机共同构成了虚拟仪器系统的硬件环境，是应用软件的基础；应用软件则是虚拟仪器的核心，在基本硬件确定后，软件通过不同功能模块(即软件模块)的组合构成了多种仪器，赋予系统特定的功能，以实现不同的测量功能。

虚拟仪器的“虚拟”有以下两个层面的意思：

(1) 虚拟的控制面板。传统仪器通过设置在面板上的各种“控件”来完成一些操作和功能，如采用各种开关、按键、滑动调节杆、显示器等实现仪器电源的“通”或“断”，被测信号“输入通道”、“放大倍数”、“滤波特性”等的参数设置，测量结果的“数值显示”、“波形显示”等。

传统仪器面板上的“控件”都是实物，而且是用手动或触摸等方式进行操作的；而虚拟仪器面板上的各种“控件”都是虚构的，其外形是与传统仪器控件的实物外形相像的图标，其实际功能是通过相应的软件程序来实现的。

(2) 虚拟的测量、测试与分析。传统仪器是通过设计具体的模拟或数字电路来实现仪器的测量、测试及分析功能的；而虚拟仪器是利用软件程序实现这些功能的。

可见，虚拟仪器是由计算机硬件资源、模块化仪器硬件以及用于数据分析、过程通信和图形用户界面的软件组成的测控系统，是一种由计算机操纵的模块化仪器系统。

1.1.2 虚拟仪器的特点

虚拟仪器技术利用高性能的模块化硬件，结合高效灵活的软件完成各种测试、测量和自动化功能的应用。灵活高效的软件能创建完全自定义的用户界面，模块化的硬件能方便地提供全方位的系统集成，标准的软/硬件平台能满足用户对同步和定时应用的需求。与传统仪器相比，虚拟仪器具有以下四个特点：

(1) 性能高。虚拟仪器技术是在 PC 技术的基础上发展起来的，因此完全“继承”了以现成即用的 PC 技术为主导的最新商业技术的优点，包括功能卓越的处理器和文件 I/O，使用户在高速导入数据的同时就能实时地进行复杂的分析。此外，不断发展的互联网技术和越来越快的计算机网络传输速度，使虚拟仪器技术展现出了更强大的优势。

(2) 扩展性强。由于虚拟仪器应用软件灵活的扩展性，使得工程师和科学家们在应用及开发软/硬件工具的过程中，不再局限于当前的技术。只需要在已有的应用工程中更新计算机或测量硬件，就能以最少的硬件投资和极少的甚至无需升级软件即可改进整个系统。

(3) 开发时间短。虚拟仪器在驱动和应用两个层面上，直接把目前高效的软件架构与计算机、仪器仪表和通信方面的最新技术通过软件的形式结合在一起，大大降低了驱动和应用两个层面的开发调试时间。同时还具有强大功能，使得用户能轻松地配置、创建、分布、维护和修改高性能、低成本的测量和控制解决方案。

(4) 无缝集成。虚拟仪器技术从本质上说是一个集成的软/硬件概念，随着产品在功能上不断趋于复杂，工程师们通常需要集成多个测量设备来满足完整的测试需求，而连接和集成这些不同设备则需要耗费大量的时间。虚拟仪器软件平台为所有的 I/O 设备提供了标准的接口，帮助用户轻松地将多个测量设备集成到单个系统中，降低了任务的复杂性。

1.1.3 虚拟仪器的硬件

随着测试、测量应用的日益复杂，目前市场上提供的模块化硬件产品也非常丰富。例如，总线类型包括 PCI、PXI、PCMCIA、USB 和 1394 等，产品种类从数据采集、信号调理、声音和振动测量、视觉、运动、仪器控制、分布式 I/O 等常用的设备到 CAN 接口等工业通信设备。

按照硬件接口的不同，虚拟仪器可分为基于 PC 总线、GPIB 总线、VXI 总线和 PXI 总线的四种标准体系结构。

(1) 基于 PC 总线的虚拟仪器。由于个人计算机的用户量大、通用性强，基于 PC 总线的虚拟仪器成为了人们的首选。这类虚拟仪器一般采用基于 PC 总线的通用型 DAQ(数据采集卡)，可以充分利用计算机的资源，大大增加了测试系统的灵活性和扩展性。利用通用型 DAQ 可方便快捷地组建基于计算机的仪器，易于实现一机多型和一机多用。随着 A/D 转换技术、精密放大技术、滤波技术与数字信号调制技术等迅速发展，DAQ 的采用速率已达到 2 GB/s，精度高达 24 位，通道数高达 64 个，并能任意组合数字 I/O、模拟 I/O、计数器/定时器等通道，大大扩展了仪器的功能。

(2) 基于 GPIB 总线的虚拟仪器。如今已有的专业仪器多数配有 GPIB(General Purpose Interface Bus，通用接口总线)，所以，一般采用 GPIB 实现利用此类仪器构建基于计算机的虚拟仪器。基于 GPIB 的虚拟仪器充分利用了现有条件，可以实现测量、检测等功能，但其数据传输速度一般低于 500 KB/s，不适合对系统速度要求较高的应用。

(3) 基于 VXI 总线的虚拟仪器。VXI 系统最多可包含 256 个装置，主要由主机箱、控制器、具有多种功能的模块仪器和驱动软件、系统应用软件等组成，具有即插即用的特性，所以，系统中各功能模块可随意更换构成新系统。基于 VXI 总线的虚拟仪器具有模块化、系统化、通用化、互换性和互操作性的特征，VXI 的价格相对较高，适用于尖端的测试领域。

(4) 基于 PXI 总线的虚拟仪器。PXI(PXI eXtensions for Instrumentation)总线整合了台式计算机高速 PCI 总线所具有的 80 MB/s 的优势，借鉴 VXI 总线中先进的仪器技术，如同步触发、板间总线、星形触发总线、板载时钟等的特性，兼容 Compact PCI 机械规范，并增加了主动冷却、环境测试(温度、湿度、振动和冲击试验)等要求。

1.1.4 虚拟仪器的软件

虚拟仪器框架从底层到顶层，由 VISA(Virtual Instrumentation Software Architecture)库、仪器驱动程序和应用软件三部分组成。

1. VISA 库

VISA 库即虚拟仪器软件体系结构库，实质上是标准 I/O 函数库及相关规范的总称，一般将该 I/O 函数库称为 VISA 库。VISA 库驻留于计算机系统中，执行仪器总线的特殊功能，起着连接计算机与仪器的作用，以实现对仪器的程序控制。

1) VISA 库的作用

对于虚拟仪器驱动程序的开发编程者来说，VISA 库是一个可调用的操作函数集。作为

标准化的 I/O 接口软件规范, VISA 库的作用有以下四点:

(1) 为所有使用者提供统一的软件编程基础, 对于驱动程序、应用程序不必考虑接口均可使用。

(2) 仅规定为用户提供标准函数, 不对具体实现做任何说明。

(3) 用于编写符合 VPP 规范的仪器驱动程序, 完成计算机与仪器之间的命令和数据传输, 实现对仪器的控制。

(4) VISA 库作为底层 I/O 接口软件, 一般运用于计算机系统中。

2) VISA 库的特点

VISA 库的主要特点一般有以下四点:

(1) 适用于各类仪器, 如 VXI、PXI、GPIB、RS-232 和 USB 仪器等。

(2) 与硬件接口无关。

(3) 既适用于单处理器结构, 又适合于多处理器或分布式结构。

(4) 适用于各种网络机制。

VISA 库只解决了仪器接口的可互换性(即改变接口或总线方式时, 不必修改测试程序), 但并没有解决更高层次的针对不同仪器的可互换性。

2. 仪器驱动程序

所谓仪器驱动程序, 是指能实现某一仪器系统控制与通信的软件程序集, 是应用程序实现仪器控制的桥梁。仪器的驱动程序由仪器生产商以源代码形式提供给用户使用, 每个仪器模块都有各自的仪器驱动程序。

常用的虚拟仪器设计软件集成了大量常用仪器的驱动程序, 以方便编程者使用。经常使用的测量仪器有几十大类、上万种型号, 各种仪器的驱动程序各不相同, 为使同类功能的仪器可以互换而不必修改测试软件, 即实现仪器的可互换性, 世界各大仪器公司都在为仪器驱动程序研究和制定统一的标准和规范而努力。

仪器驱动程序又称为驱动器。目前广泛使用的驱动器规范有 VPP(VXI Plug & Play, 即插即用型驱动器)规范和 IVI(Interchangeable Virtual Instruments, 互换型驱动器)规范两种。

3. 应用软件

应用软件是直接面向操作用户的程序。该软件建立在仪器驱动程序之上, 通过提供的测控操作界面、丰富的数据分析及处理功能等完成自动测试任务。尤其是通用数字处理软件, 集中体现了虚拟仪器的优点。

虚拟仪器应用软件的开发工具包括通用编程软件和专业图形化编程软件两类。

(1) 通用编程软件: 主要包括 Microsoft 公司 Visual Basic 中的 Visual C++、Borland 公司的 Delphi、Sybase 公司的 PowerBuilder 等。这类软件功能强大, 但并非专门为虚拟仪器而设计。利用通用编程软件开发虚拟仪器, 需要开发者具备较高的软件编程技术, 同时对虚拟仪器技术也应有相当的了解。

(2) 专业图形化编程软件: 主要包括 HP 公司的 VEE、NI 公司的 LabVIEW 和 Lab Windows/CVI 等。这类软件是针对虚拟仪器的开发而设计的, 对开发者的要求较低, 只要了解软件的总体功能以及所要设计的虚拟仪器的功能, 即可快捷方便地进行虚拟仪器的开发。

1.2 LabVIEW 特点及功能

LabVIEW 是一种图形化的编程语言，广泛地应用于工业界、学术界和研究实验室，可被视为一个标准的数据采集和仪器控制软件。LabVIEW 集成了满足 GPIB、VXI、RS-232 和 RS-485 协议的硬件及数据采集卡通信的全部功能，还内置了便于应用 TCP/IP、ActiveX 等软件标准的库函数，是一个功能强大且灵活的软件。利用 LabVIEW 可以方便地建立自己的虚拟仪器，其图形化的界面使得编程及使用过程都生动有趣。

图形化的程序语言又称为“G”语言，使用这种语言编程时，基本上不用写入程序代码，只需输入流程图即可。“G”语言最大限度地利用了技术人员、科学家、工程师所熟悉的术语、图标和概念。因此，LabVIEW 是一个面向最终用户的工具，它可以增强用户构建自己的科学和工程系统的能力，提供实现仪器编程和数据采集系统的便捷途径。使用 LabVIEW 进行原理研究、设计、测试并实现仪器系统，可以大大提高工作效率。

1.2.1 LabVIEW 的特点

LabVIEW 的出现大大提高了虚拟仪器的开发效率，降低了软件操作对开发人员的要求。LabVIEW 交互式的图形化开发环境，彻底颠覆了以往开发工具在拥有强大开发功能的同时使用却比较困难的状况。LabVIEW 所包含的各种特性使其成为开发测试、测量、自动化及控制应用的理想工具。

作为基于图形化编程语言的开发环境，LabVIEW 自然、直观、便捷的程序开发方式大大降低了学习难度。用户可以通过各种交互式的控件、对话框、菜单及函数模块进行编程。

LabVIEW 的优势主要体现在以下八个方面：

(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，实现了自动多线程处理，从而能充分利用处理器的处理能力。

(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) 内建有 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、动态链接库和安装包。

(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 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、数据记录与监控模块、机器视觉模块与触摸屏模块。

1.2.2 LabVIEW 的功能

LabVIEW 结合了简单易用的图形式开发环境与强大的图形化编程语言，提供了一个非

常直观的编程环境。LabVIEW 拥有专为大型应用开发、集成开发及应用配置设计的附加开发工具,包括应用程序生成器、图形比较、源代码控制、程序码编写指导及复杂矩阵运算等功能。

LabVIEW 不仅是一种编程语言,还是一种可用于测量和自动化的特定应用程序开发环境,也是一种可用于快速设计工业原型和应用程序的高度交互式的开发环境。同时,LabVIEW 还实现了对 FPGA 等硬件的支持,实际上也是一种硬件设计工具。在 LabVIEW 应用过程中,其测量和自动化程序在处理与通用程序相同的问题(如数据结构和算法、文件 I/O、网络 I/O 和数据库存取、打印等)时,还要处理额外的问题(如物理 I/O、实时性约束和硬件配置等)。

LabVIEW 最初是为测试测量而设计的,因而测试测量是 LabVIEW 最广泛的应用领域。经过多年的发展,LabVIEW 在测试测量领域获得了广泛的认可。如今,大多数主流的测试仪器、数据采集设备都拥有专门的 LabVIEW 驱动程序,使用 LabVIEW 可以非常便捷地控制这些硬件设备。LabVIEW 还包含了多种多样的数学运算函数,特别适合进行模拟、仿真、原型设计等工作。例如在设计机电设备之前,可以先在计算机上用 LabVIEW 搭建仿真原型,验证设计的合理性,找到潜在的问题,最后形成一个合理的机电设备设计方案。在高等教育领域,使用 LabVIEW 进行软件模拟,也可以达到同样的效果,使学生能更深刻地体会所学知识。

1.2.3 LabVIEW 的应用

电子信息类专业以及控制技术专业的专业课程,如模拟和数字电子、电路、数字信号处理、信号与系统、自动控制等,这些课程讲述的是抽象的电学原理以及控制学原理等内容,需要通过反复的计算分析、实验研究甚至实践来帮助学生理解和掌握,因此,计算和实验是学习此类课程非常重要的环节,在培养学生的实际动手能力和创新能力方面有着不可取代的作用。计算过程通常比较复杂,需要进行大量的手工运算或者编程来实现;而实验练习通常需要多种仪器,不同的实验所用的仪器也不尽相同,同时,实验仪器准备工作繁琐,实验过程复杂,投入资金较高,容易出现错误,较难熟练掌握,在有些情况下还会存在危险。

通过 LabVIEW 简化计算过程,模拟一些功能各异的实验,让读者把更多的时间用在理解原理和掌握设计上,而不用在编制复杂的程序和准备实验上花费大量的精力,大大提高了学习效率。利用 LabVIEW 辅助学习,能够更好地使理论学习与实践紧密结合,学习更生动、更形象,达到事半功倍的效果。LabVIEW 软件的诸多特点决定了它将成为未来实验教学、实践练习的重要工具。

LabVIEW 作为一种可视化工具更易被广大师生接受,建议将其贯穿到大学的辅助学习中。LabVIEW 在实验教学以及课程学习中的优势有:

(1) LabVIEW 是一个开放的开发平台,使用“所见即所得”的可视化技术建立人机界面,并且针对测试、测量和过程控制等电子信息技术与控制技术等应用领域,使用大多数学生、工程师和科学家熟悉的数据流程图式的语言书写程序代码,编程过程和思维过程非常相似,使编写程序变得简单,而且编写出来的程序更易懂。

(2) LabVIEW 软件本身提供了丰富而实用的函数库以及硬件驱动程序库,用户可以在极短的时间内开发出满足自己需要的虚拟仪器,并且可以通过修改软件的方法很方便地改变或增减仪器的功能和规模,以满足新的需求。

(3) LabVIEW 提供了许多仪器面板包含的控制对象,如表头、旋钮、开关、坐标图等,用户还可以通过控制编辑器,将现有的控制对象修改为适合自己工作领域的控制对象,可以方便直观地进行计算和检验查看实验结果。

LabVIEW 平台除了可以独立地完成电子信息类与控制类专业的基础课和专业课的计算和实验练习外,还可以将信号与系统、数字信号处理、数字图像处理等课程有机地结合起来,甚至可以完成电子信息综合课程设计、毕业设计等任务,加深学生对抽象专业知识的理解,培养学生分析问题和解决问题的能力。

利用 LabVIEW 软件建立的学习和实验平台,使学生能在所设计的 LabVIEW 程序中,通过修改参数、不断凑试和调试、反复实验、对比分析等过程,在实践中理解和掌握专业知识,为今后的工作奠定坚实的基础。

LabVIEW 软件目前已在电子测量、物理探伤、电子工程、振动分析、声学分析、地球物理勘探、故障分析、医学信息处理、射频信号处理等领域得到了广泛的应用。

1.3 LabVIEW 2015 的安装

LabVIEW 可以安装在 Mac OS、Linux 和 Windows 2000/XP/Vista 等不同的操作系统上,在安装 LabVIEW 时,对不同的操作系统的配置要求也不同。用户在安装 LabVIEW 软件之前,需要对个人计算机的软/硬件作一定的了解。

安装 LabVIEW 的最低配置要求如下:

(1) 处理器:最低配置为 Pentium III 或 Celeron 866 MHz 处理器,推荐配置 Pentium 4 或类似处理器。

(2) 内存:最小内存为 1G,推荐内存配置为 2G。

(3) 分辨率:1024 × 768 像素。

(4) 硬盘空间:最少需要 2G 的硬盘空间。

LabVIEW 2015 的安装包可以通过网络直接下载,这里建议安装 LabVIEW 2015 32 位的中文版软件。安装之前,建议先关闭杀毒软件,因为有些杀毒软件会干扰 LabVIEW 2015 的安装。具体的安装步骤如下:

步骤一:双击下载的 LabVIEW 2015 安装包,会弹出如图 1.2 所示的界面,可以通过“Browse...”按钮选择解压文件的存放位置,然后点击“Unzip”按钮对安装包进行解压。之后软件即可进入解压进度界面,如图 1.3 所示。解压完成后,会弹出如图 1.4 所示的界面,点击“确定”按钮。



图 1.2 安装包解压开始界面

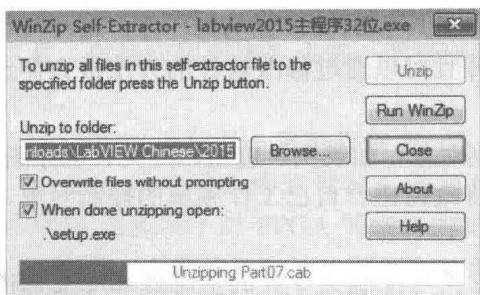


图 1.3 安装包解压进度界面



图 1.4 安装包解压成功界面

步骤二：直接弹出安装界面，点击“下一步”按钮，即可进入如图 1.5 所示的安装程序初始化界面。



图 1.5 安装程序初始化界面

步骤三：安装程序初始化完成以后，会进入如图 1.6 所示的用户信息输入界面。直接点击“下一步”按钮，进入序列号输入界面，如图 1.7 所示。这时可以直接点击“下一步”按钮，暂时不输入序列号。

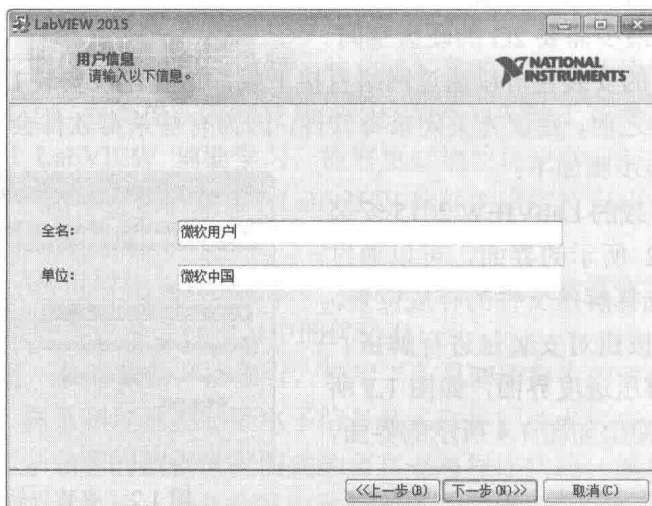


图 1.6 用户信息输入界面