

应用型本科 电气工程及其自动化专业“十三五”规划教材

LabVIEW 编程 与项目开发实用教程

Practical Programming and Development of LabVIEW

陈飞 陈奎 谢启 南江丽 编著

- 内容新颖：新知识、新技术、新工艺
- 特色鲜明：突出“应用、实践、创新”
- 定位准确：面向工程技术型人才培养
- 质量上乘：应用型本科专家全力打造



西安电子科技大学出版社

<http://www.xduph.com>

应用型本科 电气工程及其自动化专业“十三五”规划教材

LabVIEW 编程与项目开发实用教程

Practical Programming and Development of LabVIEW

陈飞 陈奎 谢启 南江丽 编著

西安电子科技大学出版社

内 容 简 介

本书以 LabVIEW 2014 简体中文版为对象, 通过理论、实例和项目应用相结合的方式, 深入浅出地介绍了 LabVIEW 的使用方法、应用技巧和项目实践应用。全书分为上、下两篇。上篇(第 1~7 章)偏重于介绍 LabVIEW 软件编程, 内容包括虚拟仪器概述、程序结构、数据类型、数据输出与图形控件、文件与数据库、LabVIEW 高级编程、通信与远程测控等。下篇(第 8~18 章)以精心编排的实践创新项目为主线, 侧重于讲述数据采集(DAQ)系统的应用和开发, 内容包括数据采集系统设计基础、数据采集软件基础、温度预警系统、交通灯控制系统、智能电子秤的设计与应用、电动自行车模拟系统、自动门控制仿真系统、数字存储式录音系统、CPU 智能散热模拟系统、油门控制系统、室内自动照明系统等, 涉及常用的六类传感器信号的采集及处理: 温度、基于电桥测量、测频、编码器、压电式集成电路和光敏传感器。本书所有例程和项目程序都经过精心设计、反复测试, 方便读者掌握 LabVIEW 编程技术与项目开发方法。

本书可作为普通高等学校相关课程的教材或教学参考书, 也可供相关领域的工程技术人员学习与参考。

图书在版编目(CIP)数据

LabVIEW 编程与项目开发实用教程/陈飞等编著. —西安: 西安电子科技大学出版社, 2016.8

应用型本科电气工程及其自动化专业“十三五”规划教材

ISBN 978-7-5606-4160-7

I. ① L… II. ① 陈… III. ① 软件工具—程序设计—高等学校—教材

IV. ① TP311.56

中国版本图书馆 CIP 数据核字(2016)第 159870 号

策 划 高 樱

责任编辑 马武装 杨 璠

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 xdupfb001@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 2016 年 8 月第 1 版 2016 年 8 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 19

字 数 447 千字

印 数 1~3000 册

定 价 34.00 元

ISBN 978-7-5606-4160-7/TP

XDUP 4452001-1

如有印装问题可调换

西安电子科技大学出版社

应用型本科电气工程及其自动化专业系列教材

编审专家委员名单

主任：汪志锋(上海第二工业大学 电子与电气工程学院 院长/教授)

副主任：罗印升(江苏理工学院 电气信息工程学院 院长/教授)

刘 燕(常熟理工学院 电气与自动化工程学院 教授)

成 员：(按姓氏拼音排列)

陈 桂(南京工程学院 自动化学院 副院长/副教授)

邓 琛(上海工程技术大学 电子电气工程学院 副院长/教授)

杜逸鸣(三江学院 电气与自动化工程学院 副院长/副教授)

高 亮(上海电力学院 电气工程学院 副院长/教授)

胡国文(盐城工学院 电气工程学院 院长/教授)

姜 平(南通大学 电气工程学院 副院长/教授)

王志萍(上海电力学院 自动化工程学院 副院长/副教授)

杨亚萍(浙江万里学院 电子信息学院 副院长/副教授)

郁有文(南通理工学院 机电系 教授)

张宇林(淮阴工学院 电子与电气工程学院 副院长/教授)

邹一琴(常州工学院 电子信息与电气工程学院 副院长/副教授)

周渊深(淮海工学院 电子工程学院 副院长/教授)

前 言

虚拟仪器是一种全新的仪器系统概念，它是计算机技术、现代测控技术、现代通信技术以及信号处理技术完美结合的产物，代表着测控技术的发展方向。LabVIEW 是美国 NI 公司推出的一种图形化虚拟仪器软件开发环境，具有简单直观、易于理解和开发效率高等特点。LabVIEW 的应用范围从简单仪器控制与数据采集到尖端测试和工业自动化，从大学课堂与实验室到工业、产业现场，从探索与研究到技术集成。近年来，我国对虚拟仪器应用的需求日益增长，虚拟仪器的应用范围也不断扩大，对掌握虚拟仪器技术与开发的毕业生需求也越显迫切。目前高校中的很多科研项目都是基于虚拟仪器平台实现的，而且很多高校都开设了虚拟仪器或 LabVIEW 课程，建立了基于虚拟仪器的实验室。

本书通过精心设计编排例程及实践项目，深入浅出地介绍了 LabVIEW 的基本编程技术与 DAQ 系统开发的基本方法，以便读者尽快地掌握 LabVIEW 应用和开发的一般方法及技巧，能够尽快上手开发自己的测控系统，做到理论联系实际。

本书分为上、下两篇。上篇(第 1~7 章)介绍 LabVIEW 基本编程方法，包括虚拟仪器概述、程序结构、数据类型、数据输出与图形控件、文件与数据库、LabVIEW 高级编程、通信与远程测控等内容。下篇(第 8~18 章)介绍数据采集系统设计基础、数据采集软件基础、温度预警系统、交通灯控制系统、智能电子秤的设计与应用、电动自行车模拟系统、自动门控制仿真系统、数字存储式录音系统、CPU 智能散热模拟系统、油门控制系统、室内自动照明系统等内容。通过对项目目标、项目分析、项目内容、项目实现的详细讲解，培养读者运用虚拟仪器技术解决工程实践问题的能力。

编者结合多年教学经验和学生的学习特点，对书中例程进行精心设计、反复测试，尤其对文件 I/O、数据通信、数据库应用等方面的例程进行了补充或加强。本书可作为高等学校电子、通信、仪器、测量及计算机等专业相关课程的教材，也可供相关领域的工程技术人员学习与参考。

本书上篇由陈奎编写，下篇由陈飞、谢启等编写。在此感谢 NI 公司、上海泛华测控技术有限公司技术人员提供的帮助，感谢黄为勇、徐惠钢教授和其他老师在编写过程中给予的支持和帮助。同时还要感谢参与程序测试与文字校正的老师和学生。

由于编者水平有限，书中难免有疏漏之处，敬请读者指正。如需与作者联系，可发邮件至 kuirs@163.com、cfmonday@163.com。

编者

2016 年 7 月

目 录

上 篇

第 1 章 虚拟仪器概述	2	2.3 顺序结构.....	32
1.1 虚拟仪器.....	2	2.3.1 顺序结构及创建方法.....	32
1.1.1 虚拟仪器的定义.....	2	2.3.2 顺序结构的组成与操作.....	33
1.1.2 虚拟仪器的历史与发展.....	3	2.3.3 层叠式顺序结构局部变量的创建....	34
1.1.3 虚拟仪器的组成.....	4	2.3.4 顺序结构应用实例.....	35
1.2 图形化编程与 LabVIEW.....	7	2.4 事件结构.....	36
1.2.1 NI 公司与 LabVIEW.....	7	2.4.1 事件结构的创建和编辑.....	36
1.2.2 图形化 G 语言.....	7	2.4.2 事件的种类.....	37
1.3 LabVIEW 2014 简介.....	8	2.4.3 事件结构应用实例.....	38
1.3.1 LabVIEW 操作界面.....	8	习题.....	40
1.3.2 LabVIEW 选板.....	10	第 3 章 数据类型	41
1.3.3 LabVIEW 工具条.....	13	3.1 LabVIEW 数据类型及其操作.....	41
1.4 LabVIEW 虚拟仪器设计.....	14	3.1.1 数值型.....	41
1.4.1 LabVIEW 程序设计流程.....	14	3.1.2 布尔型.....	42
1.4.2 创建 VI 和调用子 VI 实例.....	18	3.1.3 枚举型.....	43
1.5 LabVIEW 的学习方法和学习资源.....	19	3.1.4 数值运算.....	44
1.5.1 学习方法.....	19	3.2 数组.....	45
1.5.2 学习资源.....	20	3.2.1 数组的创建.....	45
习题.....	21	3.2.2 数组控件的操作与设置.....	46
第 2 章 程序结构	22	3.2.3 循环索引创建数组.....	47
2.1 循环结构.....	22	3.2.4 数组函数.....	48
2.1.1 While 循环.....	22	3.3 字符串.....	53
2.1.2 For 循环.....	24	3.3.1 常用字符串控件.....	53
2.1.3 隧道与数据交换.....	24	3.3.2 常用字符串函数.....	56
2.1.4 移位寄存器和反馈节点.....	25	3.3.3 数值与字符串转换.....	61
2.1.5 循环结构应用实例.....	27	3.4 簇.....	62
2.2 条件结构.....	28	3.4.1 簇控件的创建.....	62
2.2.1 条件结构.....	28	3.4.2 簇函数.....	64
2.2.2 条件结构操作.....	30	3.4.3 簇与数组.....	66
2.2.3 选择结构应用实例.....	31	3.5 波形数据.....	66

3.5.1 时标	66	5.3.3 LabVIEW 与数据库应用	108
3.5.2 波形数据	68	5.3.4 LabSQL 应用实例	112
3.5.3 波形函数与波形处理	68	5.3.5 LabVIEW DCT 应用实例	114
习题	70	习题	116
第 4 章 数据输出与图形控件	71	第 6 章 LabVIEW 高级编程	118
4.1 波形图表	71	6.1 公式节点	118
4.1.1 波形图表的创建	71	6.1.1 公式节点	118
4.1.2 波形图表外观定制	71	6.1.2 公式	120
4.1.3 波形图表的输入数据类型	74	6.1.3 公式实例	121
4.1.4 波形图表实例	75	6.2 MATLAB 脚本节点	121
4.2 波形图	76	6.2.1 MATLAB Script 节点	122
4.2.1 Graph 的基本操作	76	6.2.2 MATLAB Script 节点实例	123
4.2.2 波形图的输入数据类型	78	6.3 局部变量与全局变量	123
4.2.3 波形图实例	80	6.3.1 局部变量	123
4.3 XY 图	80	6.3.2 全局变量	125
4.3.1 XY 图概述	80	6.4 属性节点	126
4.3.2 XY 图绘图	81	6.4.1 属性节点的创建	127
4.3.3 Express XY 图	82	6.4.2 属性节点的使用	128
4.4 强度图	82	6.5 子 VI	130
4.5 三维图形显示控件简介	84	6.5.1 VI 图标的创建与编辑	130
4.5.1 三维图形控件	84	6.5.2 连接器端口的设置	131
4.5.2 三维图片	87	6.5.3 子 VI 的创建	132
习题	88	6.5.4 添加子 VI 至用户库	133
第 5 章 文件与数据库	89	6.5.5 子 VI 的调用和 VI 的层次结构	134
5.1 LabVIEW 与文件类型	89	习题	134
5.1.1 文件类型	89	第 7 章 通信与远程测控	136
5.1.2 LabVIEW 文件操作的基本概念	90	7.1 远程测控系统概述	136
5.1.3 LabVIEW 文件选板	92	7.2 DataSocket 通信技术	137
5.2 常用文件类型	93	7.2.1 DataSocket 技术	137
5.2.1 文本文件	93	7.2.2 DataSocket 组成	138
5.2.2 电子表格文件	97	7.2.3 DataSocket 应用实例	140
5.2.3 二进制文件	100	7.3 网络通信	143
5.2.4 数据记录文件	102	7.3.1 TCP	143
5.2.5 波形文件	103	7.3.2 TCP 应用实例	145
5.3 数据库应用	105	7.3.3 UDP	148
5.3.1 SQL 语言基础	105	7.3.4 UDP 与 TCP 的区别	149
5.3.2 常用 SQL 编程语句	107	7.4 串口通信	149

7.4.1 串口通信	149
7.4.2 LabVIEW 串口编程	151
7.4.3 串口通信实例	153

7.5 Web 发布	156
习题	158

下 篇

第 8 章 数据采集系统设计基础

8.1 NI 数据采集系统的基本组成	160
8.2 信号类型	161
8.2.1 数字信号	162
8.2.2 模拟信号	162
8.3 数据采集设备与指标	163
8.3.1 数据采集设备	163
8.3.2 数据采集设备的主要性能指标	164
8.4 模拟测试信号的连接	165
8.4.1 信号的连接	165
8.4.2 信号的测量系统类型	166
8.4.3 模拟输入信号连接	168
8.4.4 模拟输出信号连接	169
8.5 数字信号连接	170
8.6 数据采集卡 PCI-6221	170
习题	173

第 9 章 数据采集软件基础

9.1 数据采集软件的安装	174
9.2 配置管理软件 MAX	174
9.2.1 测试功能	175
9.2.2 创建任务	178
9.3 NI-DAQmx 重要概念	181
9.3.1 NI-DAQmx 通道和任务	181
9.3.2 虚拟通道的类型	182
9.3.3 物理通道语法	182
9.4 NI-DAQmx 常用函数	183
习题	190

第 10 章 温度预警系统

10.1 项目目标	191
10.2 项目分析	191
10.3 项目内容	192

10.3.1 温度检测电路设计	192
10.3.2 温度信号采集程序设计	195
10.3.3 温度分析处理程序设计	202
10.4 项目实现	206
习题	210

第 11 章 交通灯控制系统

11.1 项目目标	211
11.2 项目分析	212
11.3 项目内容	212
11.3.1 交通灯模拟系统硬件电路设计	212
11.3.2 模拟简易交通灯	212
11.3.3 数字信号输入/输出程序设计	215
11.4 项目实现	217
习题	222

第 12 章 智能电子秤的设计与应用

12.1 项目目标	223
12.2 项目分析	224
12.3 项目内容	224
12.3.1 称重传感器电路设计	224
12.3.2 数码管显示控制	226
12.3.3 电子秤的程序设计	228
12.4 项目实现	229
习题	237

第 13 章 电动自行车模拟系统

13.1 项目目标	238
13.2 项目分析	238
13.3 项目内容	239
13.3.1 电动自行车模拟系统硬件设计	239
13.3.2 电动自行车车把信息采集	241
13.3.3 电动自行车转速测量	243

13.3.4 电动自行车轮速控制	244	15.3.4 声音文件播放程序设计	264
13.4 项目实施	245	15.4 项目实施	265
习题	247	习题	268
第 14 章 自动门控制仿真系统	248	第 16 章 CPU 智能散热模拟系统	270
14.1 项目目标	248	16.1 项目目标	270
14.2 项目分析	248	16.2 项目分析	270
14.3 项目内容	249	16.3 项目实施	271
14.3.1 自动门控制系统硬件设计	249	习题	276
14.3.2 步进电机正反转控制程序设计	252	第 17 章 油门控制系统	277
14.3.3 编码器的角度测量	253	17.1 项目目标	277
14.3.4 仿真自动门的动画控件设计	255	17.2 项目分析	277
14.4 项目实施	257	17.3 项目实施	278
习题	259	习题	283
第 15 章 数字存储式录音系统	260	第 18 章 室内自动照明系统	284
15.1 项目目标	260	18.1 项目目标	285
15.2 项目分析	260	18.2 项目分析	285
15.3 项目内容	261	18.3 项目实施	285
15.3.1 录音播放系统硬件电路设计	261	习题	292
15.3.2 声音文件的存储和读取	262		
15.3.3 声音信号采集程序设计	263		
参考文献	293		

上 篇

- 第 1 章 虚拟仪器概述
- 第 2 章 程序结构
- 第 3 章 数据类型
- 第 4 章 数据输出与图形控件
- 第 5 章 文件与数据库
- 第 6 章 LabVIEW 高级编程
- 第 7 章 通信与远程测控

第 1 章 虚拟仪器概述

1.1 虚拟仪器

1.1.1 虚拟仪器的定义

虚拟仪器(Virtual Instrument)是基于计算机的仪器。将计算机和仪器密切结合是目前仪器技术发展的一个重要方向。粗略地说,这种结合有两种方式。一种方式是将计算机装入仪器,其典型的例子就是智能化仪器。随着计算机功能的日益强大以及其体积的日趋缩小,这类仪器的功能也越来越强大。目前,内嵌高性能嵌入式系统的仪器仪表就是其典型代表。另一种方式是将仪器装入计算机,以通用计算机硬件及操作系统为依托,实现各种仪器的功能,即虚拟仪器的实现方式。

一般来说,仪器系统的基本功能主要包括信息的采样与控制、信息的分析与处理、信息的表达与指示。如图 1-1 所示,虚拟仪器实现方式中的数据采样、分析和处理由通用计算机的软硬件完成。插入计算机的数据采集卡实现信号的采样、调理与传输;应用软件实现被测信号的分析、处理,并按需要输出结果,包括图表显示、数据传输、数据存储和报表打印,整个测控过程仅通过对软件界面操作即可完成。

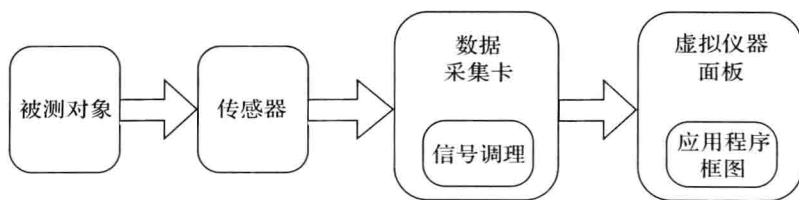


图 1-1 虚拟仪器实现方式

虚拟仪器充分利用计算机软件与硬件的优势,实现数据自动采集、分析和处理,既达到了传统仪器的效果,又突破了传统仪器在测量速度、测量功能以及适应范围的局限,并能拓展用户创新想象的空间,从而在仪器功能、处理算法、信息表达形式上灵活创新。根据测控项目的不同需求,通过必要的软件设计与硬件增减,用户就可以设计出全新的、独有的测控系统。在虚拟仪器系统中,硬件仅负责信号的输入与输出,软件才是整个仪器系统的关键和灵魂,所以有“软件就是仪器”的说法。

表 1-1 列出了虚拟仪器与传统仪器的各自特点。相比传统仪器,虚拟仪器的最大特点是其功能由软件定义,用户可按照不同应用要求进行选择和调整。而传统仪器的功能是由厂商事先定义好的,用户无法变更其功能。正是由于这个显著特点,虚拟仪器具有功能多样、开放灵活、面向应用、重复使用、价格便宜等特点。

表 1-1 传统仪器与虚拟仪器对比

传统仪器	虚拟仪器
功能由仪表厂家定义	功能由用户自定义
功能确定, 与其他设备的连接受到限制	面向应用, 可以方便地连接其他设备
关键为硬件	关键为软件
价格昂贵	价格便宜, 可再利用
封闭, 功能固定、单一, 不能更改	以计算机为支撑, 开放性好, 功能灵活、多样
技术更新慢	技术更新快
开发和维护费用高	开发和维护费用低

虚拟仪器是计算机技术与仪器仪表技术深层次结合的产物, 是对传统仪器概念的重大突破, 是仪器仪表领域内的一次革命。虚拟仪器是继模拟化仪表、智能化仪器之后的新一代仪器, 代表了当前测控仪器发展的方向之一。

1.1.2 虚拟仪器的历史与发展

1. 虚拟仪器的历史回顾

虚拟仪器的起源可以追溯到 20 世纪 70 年代, 那时计算机测控系统在国防、航天等领域已经有了相当的发展。20 世纪 80 年代, 美国 NI(National Instruments)公司提出“仪器的计算机化”, 即虚拟仪器的概念。从此虚拟仪器技术便成为自动测控领域的研究热点和应用前沿。虚拟仪器在国外已发展成为一种新的产业。美国是虚拟仪器的诞生地, 目前也是全球最大的虚拟仪器制造国。

虚拟仪器的发展大致分为以下三个阶段:

第一阶段: 基于计算机技术来提升传统仪器功能阶段(PC + 数据采集卡 + 开发软件)。

第二阶段: 内在标准统一阶段(硬件标准化、软件标准化)。

第三阶段: 虚拟仪器软件封装及组合阶段。

近年来, 世界各国的许多大型自动测控和仪器公司均相继研制了为数不少的虚拟仪器开发平台, 但最早和最具影响力的是 NI 公司的 LabVIEW 图形化开发平台。NI 公司早在 Windows 诞生之前, 就已经在 Macintosh 计算机上推出了它的 LabVIEW 2.0 及之前的版本。由于对虚拟仪器和 LabVIEW 长期、系统、有效的研究开发, NI 公司已成为业界公认的权威。

LabVIEW 的发展历程如下:

20 世纪 70 年代末期: 美国应用研究实验室产生 VI 概念的雏形。

1986 年: 发布 Macintosh 平台下的 LabVIEW 1.0。

1988 年: 发布 Macintosh 平台下的 LabVIEW 2.0。

1990 年: 虚拟仪器面板和结构化数据流图获两项美国专利。

1994 年: 发布 LabVIEW 3.0, 带有附加工具包。

1996 年: 发布 LabVIEW 4.0, 增加了自定义界面和 Application Builder。

1998 年: 发布 LabVIEW 5.0, 支持多线程。

2000年：发布 LabVIEW 6i，集成因特网功能。

2001年：发布 LabVIEW 6.1，实现远程控制和增加事件结构等重要功能。

2003年：发布 LabVIEW 7 Express，增加了 Express VI。

2004年：发布 LabVIEW 7.1 Express，增加了许多全新的功能。

2005年：发布 LabVIEW 8.0，增加了许多全新的功能。

2006年：发布 LabVIEW 8.2，有了第一个中文版的开发环境。

LabVIEW 8.X 版本中引入了面向对象(OOP)的程序设计概念，使 LabVIEW 更接近一个完整的编程语言。2010年以来，NI 公司相继发布了 LabVIEW 2010、LabVIEW 2011，直至最新的 LabVIEW 2016。LabVIEW 是向上兼容的，最近几年的 LabVIEW 各版本在操作方法、基本界面和功能上没有区别，有条件的用户可下载较新版本学习。

2. 虚拟仪器的发展和展望

虚拟仪器技术是计算机技术与测试技术相结合的产物，融合了多个学科、多种技术，如测试技术、计算机技术、软件技术、数字信号处理技术、总线与接口技术、网络与通信技术、传感技术、光电技术、微机械技术等。

对于虚拟仪器来说，计算机是载体，软件是核心，高质量的 A/D 采集及信号调理是关键。虚拟仪器是基于计算机的仪器系统，必然与计算机相关技术的进展有密切的关系。虚拟仪器又是测控仪器，必然带有浓厚的仪器系统特征。因此，虚拟仪器的发展取决于以下三个重要因素：

- (1) 虚拟仪器的发展必须依赖于计算机的进步和发展。
- (2) 虚拟仪器的发展必须借助各种硬件技术和强大的软件资源。
- (3) 虚拟仪器的发展必须制定开放式的虚拟仪器 I/O 总线标准。

目前，虚拟仪器的发展趋势表现在以下几个方面：

(1) 高性能数字信号处理芯片、大规模可编程逻辑器件的发展，提高了信号采集和处理的速度，缩短了虚拟仪器系统的开发时间，提高了系统的扩展性。

(2) 智能化、模块化、集成化是硬件发展的主流。

(3) 软件技术、软件标准化得到发展，制定虚拟仪器行业标准和相关产业政策，保护自主的虚拟仪器知识产权。

(4) 智能化软件开发平台是虚拟仪器一个重要的发展方向。用人工智能技术提高虚拟仪器软件系统的可重构能力，降低虚拟仪器设计难度，真正实现用户自己定义仪器的目标。

(5) 新的总线技术应用(如 HS488、1394b 等)。

(6) 仪器系统网络化发展。

1.1.3 虚拟仪器的组成

虚拟仪器是以计算机为载体的自动化测控系统，包括高性能模块化的硬件系统、高效灵活的软件系统。

1. 虚拟仪器硬件系统

图 1-2 所示为虚拟仪器硬件系统，包括计算机(PC、工作站)、各种 I/O 接口及设备(如 RS-232 或 IEEE1394 串行仪器、USB 仪器、GPIB 接口卡与仪器、VXI 或 PXI 仪器，以及

基于上述标准的数据采集卡和信号调理电路等)。

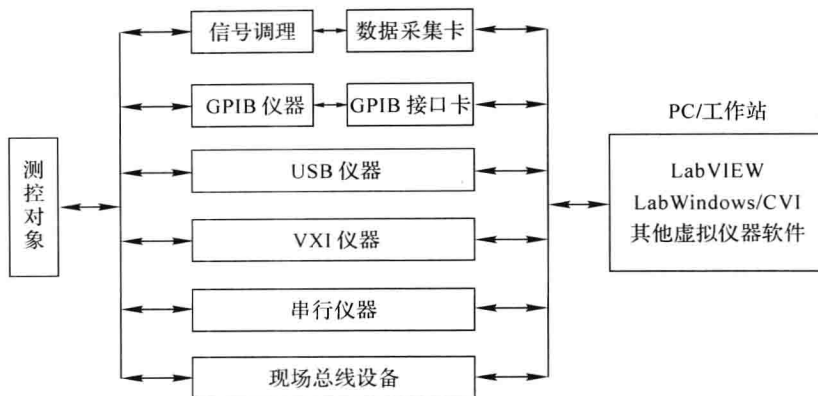


图 1-2 虚拟仪器硬件组成

I/O 接口或设备的产品种类涵盖了数据采集、信号调理、声音和振动测量、视觉、运动、仪器控制、分布式 I/O、工业通信等方面。针对不同应用目的、应用环境和性能要求，I/O 接口及设备可采用不同的总线结构，包括 PCI 结构、PXI 总线结构、USB 总线结构、分布式系统结构、GPIB 或串口、RIO 系统结构等。下面对几种常用总线结构进行简要介绍。

RS-232C 串行接口是计算机与外部设备之间以及计算机与测试系统之间进行数据传输的一种通信接口。RS-232C 串行接口的引脚有 9 针和 25 针两类，目前一般使用 9 针接口。RS-232C 能够可靠进行数据传输的最大通信距离为 15 m，若需要远程传输，则必须通过调制解调器，或改用差分传输方式的 RS-485 进行通信。计算机串行端口的传输速率为 1200~115 200 b/s，在仪器仪表或工业控制场合，9600 b/s 是最常见的传输速率。近距离串行传输可采用较高的传输速率，远距离传输时通过降低传输速率可延长传输距离、提高通信稳定性。

USB(Universal Serial Bus)是一种高性能串行总线规范，具有传输速率高、即插即用、热切换(带电插拔)和可利用总线传送电源等特点，目前已是 PC 标准配置之一。USB 电缆中只有一对信号线和一对电源线，最高传输速率为 480 Mb/s。USB 轻巧便宜，适用于传递文件数据和音响信号。NI 公司的部分外置式数据采集卡就采用 USB 接口。目前常用的是 USB 2.0、USB 3.0 等。

IEEE1394 串行总线(俗称火线(FireWire))是目前应用前景非常广阔的串行总线，最高传输速率为 400 Mb/s，传输距离为 72 m，以后还要按 800 Mb/s、1.6 Gb/s 及 3.2 Gb/s 分段提高。IEEE1394 有两对信号线和一对电源线，在无 HUB 时可用任何方式连接 63 个装置，且支持即插即用、带电插拔。

VXI 总线(VME Bus Extension for Instrumentation)是高速计算机总线 VME(Versa Module Eurocard)在仪器领域的扩展。由于 VXI 标准开放、传输速率高、数据吞吐能力强、定时和同步精确、模块化设计、结构紧凑、使用方便灵活，已越来越受到重视。VXI 便于组织大规模、集成化系统，是仪器硬件发展的一个方向。

GPIB(General Purpose Interface Bus)又称 IEEE 488 标准、IEC 652 标准，是 HP 公司推出的台式仪器接口总线，也被称为 HP-IB。GPIB 有 8 根数据线，最高传输速率为 1 Mb/s，传输距离为 20 m(加驱动器可达 500 m)。在微机中插入一块 GPIB 接口卡，通过 24 或 25 线

电缆连接到仪器端的 GPIB 接口。当微机的总线变化时，例如采用 ISA 或 PCI 等不同总线，接口卡也随之变更，从而使 GPIB 系统能适应微机总线的快速变化。GPIB 仍是仪器、仪表及测控系统与计算机互连的主流并行总线。装有 GPIB 接口的台式仪器的品种和数量都明显超过 VXI 仪器，目前应用的 VXI 系统、PXI 系统中与 GPIB 混合应用的比例很大。

目前，基于以上各种总线标准的硬件产品层出不穷，大大拓宽了用户对硬件系统的选择面。NI 公司推出的数据采集设备 M 系列 DAQ 卡、CompactDAQ 控制器，可简化用户的测量系统，为数据采集领域设定了全新标准。

2. 虚拟仪器软件系统

“软件即仪器”，应用软件是虚拟仪器的核心。一般来说，虚拟仪器硬件系统中功能模块的生产商会提供示波器、数字万用表、逻辑分析仪等常用虚拟仪器应用程序。但对于用户的特殊应用需求，则可利用虚拟仪器开发软件平台来实现。

VISA(Virtual Instrumentation Software Architecture)体系结构是标准的 I/O 函数库及相关规范的总称。一般称这个 I/O 函数库为 VISA 库。它驻留于计算机系统之中，执行仪器总线的特殊功能，是实现计算机系统与仪器之间命令与数据传输的桥梁和纽带。许多仪器生产厂家推出硬件接口电路的同时，也推出不同结构的 I/O 接口软件，如 GPIB 仪器、VXI 仪器等。计算机与仪器之间的软件层连接，用于实现对仪器的程序控制。对于仪器驱动程序开发者来说，VISA 库是一个个可调用的操作函数。

如图 1-3 所示，VISA 体系结构从底到顶包括：VISA/O 库与 VXI 总线接口软件、仪器驱动层、应用软件开发环境(LabVIEW 等)和应用程序(VI)。

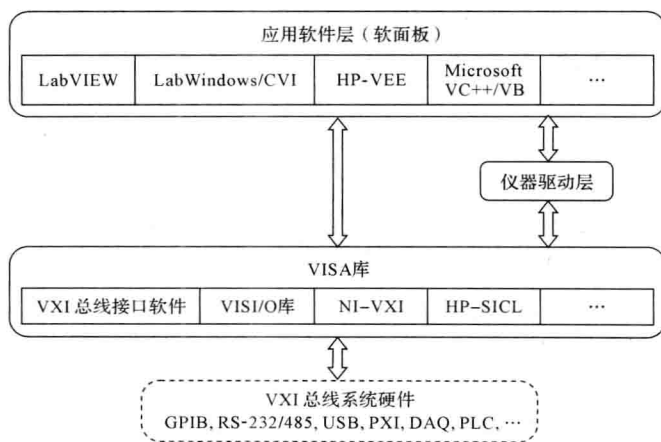


图 1-3 VISA 体系结构

仪器驱动层是连接虚拟仪器系统硬件(如系统前端电路、数据采集器及其接口)和虚拟仪器应用软件之间的软件中间层，它由函数库、实用程序、工具套件等组成，用于实现对仪器的控制。必须指出，每个仪器模块都有自己的仪器驱动程序，仪器厂商以源码的形式提供给用户。

应用软件开发环境建立在仪器驱动程序之上，直接面对开发用户，通过提供直观、友好的测控设计界面，丰富的数据分析与处理功能，完成测控系统的软件开发。应用软件开发环境分为通用软件开发环境和专用软件开发环境，专用软件开发环境还可细分为基于图形化编程的开发环境和基于文本编程的开发环境。

(1) 通用软件开发环境: Microsoft 公司的 VC++、VB, Inprise 公司的 Delphi 和 C++ Builder。

(2) 专用软件开发环境如下:

① 基于图形化编程的开发环境: NI 公司的 LabVIEW, Agilent 公司的 VEE, HP 公司的 HP-VEE, Tektronix 公司的 Ez-Test、Tek-TNS, HEM Data 公司的 Snap-Master。

② 基于文本编程的开发环境: NI 公司的 LabWindows/CVI。

其中, 基于图形化编程的开发环境是随虚拟仪器的发展而设计与发展起来的, 是一种虚拟仪器专用软件开发平台。它具有方便、便捷的可视化界面, 具有类似流程图的编程风格, 是目前应用最广、发展最快、功能最强的虚拟仪器开发环境之一。

基于文本编程的开发环境是一个更底层、更基本和更开放的平台, 它开发的程序小, 执行效率高, 特别适合于底层驱动开发或复杂的大系统、通用的测试测量仪器系统以及高性能要求的特定测试系统。

总之, 虚拟仪器的崛起是测控仪器技术的一次革命, 是仪器领域的一个新的里程碑, 它使现代测控系统更灵活、更紧凑、更经济、功能更强。无论是测量、测试、计量或是工业过程控制和分析处理, 还是涉及其他更为广泛的测控领域, 虚拟仪器都是理想的选择。随着电子技术、计算机技术、网络技术等相关技术的发展, 虚拟仪器将会逐步取代传统的测试仪器而成为测试仪器的主流。

1.2 图形化编程与 LabVIEW

1.2.1 NI 公司与 LabVIEW

目前, 虚拟仪器软件开发环境使用最为广泛的是 NI 公司的 LabVIEW(Laboratory Virtual Instrument Engineering Workbench), 类似于 C 语言和 BASIC 语言开发环境。LabVIEW 是一种程序开发环境, 是一种图形化的编程语言。LabVIEW 集成了工程师和科学家快速构建各种应用所需的所有工具, 旨在帮助工程师和科学家解决问题、提高生产力和不断创新, 广泛地被工业界、学术界和研究实验室所接受, 被视为一个标准的数据采集和仪器控制软件。

LabVIEW 软件是 NI 公司设计平台的核心, 也是开发测量或控制系统的理想选择, 它集成与满足 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, 内置了便于应用 TCP/IP、ActiveX 等软件标准的库函数, 还有丰富的工具包软件可供开发更加专一的虚拟仪器系统使用。LabVIEW 是一个功能强大且灵活的软件, 利用它可以方便地建立自己的虚拟仪器系统, 其图形化界面使得编程及使用过程都生动有趣、易于上手。

利用 LabVIEW 可产生独立运行的可执行文件, 它是一个真正的 32 位编译器。像许多重要的软件一样, LabVIEW 提供了 Windows、UNIX、Linux、Macintosh 的多种版本。

1.2.2 图形化 G 语言

图形化编程语言又称为 G 语言(Graphical Programming Language)。采用 G 语言编程时,

基本不用写程序代码，取而代之的是绘制流程图，又称程序框图。

G 语言是基于数据流驱动的，与一般文本编程语言的过程驱动机制有很大差别，在程序设计的思路上也与文本编程语言有所区别。因此编程者需掌握 G 语言编程思想，按照图形化程序设计思想来解决问题。表 1-2 对常用编程方式做了简单对比。

表 1-2 编程方式比较

编程方式	界面	编程	举例
文本编程	运行时才可见文本行和图形	文本行	Turbo C
可视化编程(OOP)	图形，编程时编辑	文本行	VC、VB
图形化编程(G 语言)	图形，编程时编辑	框图程序	LabVIEW

由于 LabVIEW 的影响力，它已成为 G 语言的主要代表之一。LabVIEW 是一个开发环境(类似的如 Visual Studio)，在这个环境下编写的代码就是 G 语言代码(类似的如在 Visual Studio 下写出的 C 代码)。目前，很多工程师认为 LabVIEW 是一个应用在工业测控领域的应用软件，并不理解它是一个编程语言。其原因有两个：首先是因为它和以往其他的编程语言差距太大，第一次看到它的人更容易联想到电路板布线、工业总线配置软件等；其次是因为 LabVIEW 在中国使用的年头不多，大多数用户仅用到了 LabVIEW 的一小部分功能，还没有真正体验到 LabVIEW 的强大。

LabVIEW 虽然不能覆盖所有文本语言的优点，但它具有自己的特色。在编写与工业领域设计、测量、控制等相关的程序或系统时，LabVIEW 的开发效率大大高于其他语言。LabVIEW 图形化编程的特点如下：

(1) 编程简单，不需要记忆编程语言，只要通过交互式图形前面板进行系统控制和结果显示，再通过后面板进行常用的符号和功能模块的组合操作来指定各种功能，即可完成软件编程。

(2) 开发周期短，只需通过交互式图形前面板进行系统控制和结果显示，可省去硬件面板的制作。

(3) 功能性强，可省去硬件面板的制作，具有多达 20 种的功能模块。

(4) 高效性，这主要是靠软件来保证的。以功能强大的 LabVIEW 作为软件开发平台，诸如数据采集、数据分析、文件处理、波形处理、数学运算等都能轻而易举地解决。

(5) 开放性，可根据实际情况进行更新扩展，发展迅速。

(6) 自定义性，可在非常广泛的测量和控制应用中自定义芯片级硬件功能。

(7) 性价比高，能一机多用。

1.3 LabVIEW 2014 简介

1.3.1 LabVIEW 操作界面

LabVIEW 2014 可到 NI 公司官网下载。安装并启动 LabVIEW 2014 后出现开始界面，如图 1-4 所示。窗口界面用于界面设计、图形化程序设计和程序运行与调试。