



普通高等教育测控信息技术规划教材

虚拟仪器 实用编程技术

Practical Programming of Virtual Instruments

童刚 主编



TP216/52

2008

普通高等教育测控信息技术规划教材

虚拟仪器实用编程技术

主 编 童 刚
副主编 樊春玲 崔凤英
参 编 孙四通

机械工业出版社

本书从实用角度出发,以 LabVIEW7.1 为基础,介绍了 LabVIEW 程序设计的基本内容及虚拟仪器开发的实用方法和技术。

全书共分 10 章,前 5 章介绍了入门知识和基本编程方法,主要包括 VI 程序的建立、结构、数组和簇、图形显示、文件读写等,后 5 章从程序开发的实际应用角度介绍了以 LabVIEW 作为开发平台的实用技术,主要包括局部变量和全局变量的定义与使用、外部应用程序接口、分析软件、数据采集和远程测控等。全书编写侧重于基础,力求简单实用,以方便读者学习使用。

本书可作为高等院校测控技术与仪器、自动化等专业的教材或参考书,也可供相关专业的工程技术人员和软件工程师参考。

本书配有电子课件,欢迎选用本书作教材的老师索取,索取邮箱:wbj@mail.machineinfo.gov.cn。

图书在版编目(CIP)数据

虚拟仪器实用编程技术/童刚主编. —北京:机械工业出版社,2008.1

普通高等教育测控信息技术规划教材
ISBN 978-7-111-22902-5

I. 虚… II. 童… III. 智能仪器-程序设计-高等学校-教材 IV. TP216

中国版本图书馆 CIP 数据核字(2007)第 182401 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:王保家 责任校对:袁凤霞

封面设计:张 静 责任印制:洪汉军

北京铭成印刷有限公司印刷

2008 年 1 月第 1 版第 1 次印刷

184mm×260mm·11.5 印张·278 千字

标准书号:ISBN 978-7-111-22902-5

定价:18.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

销售服务热线电话:(010)68326294

购书热线电话:(010)88379639 88379641 88379643

编辑热线电话:(010)88379727

封面无防伪标均为盗版

前 言

虚拟仪器技术是电子测量技术与计算机技术相结合的产物,是这两门学科的最新技术的结晶,它融合检测理论、仪器原理和技术、计算机接口技术、高速总线技术及图形化软件编程技术于一身,实现了测量仪器的智能化、多样化、模块化和网络化。虚拟仪器有功能多、成本低、应用灵活、操作方便等优点,在很多领域已大有取代传统仪器的趋势,可以说,它代表了未来仪器发展的方向。

LabVIEW 是与虚拟仪器相关的软件,是国内外测控技术和仪器科学中的通用语言之一。近年来,LabVIEW 在国内普及和应用的速度不断加快,许多理工科院校已开设了 LabVIEW 与虚拟仪器的课程,并建立了相关的虚拟仪器实验室,虚拟仪器技术在教学实验和科研中起着越来越重要的作用。

由于多年从事虚拟仪器及 LabVIEW 的教学和研究应用工作,深感有必要将 LabVIEW 的实用编程技术及应用介绍给读者,因此,笔者结合自身体会和经验编写了此书,希望有更多读者能通过对本书的学习,快速、系统、全面地掌握 LabVIEW。

为了帮助读者轻松地进入 LabVIEW 编程世界,充分享受 LabVIEW 的图形化编程语言给用户应用程序的开发带来的灵活性和快捷性,本书由浅入深、循序渐进,对图形化编程语言的基本概念、基本结构进行了透彻的讲解,并在各章节适时插入图形,力求做到图文并茂。同时,书中还提供了大量的应用实例和分析,便于读者学习。全书共分 10 章,前 5 章介绍了入门知识和基本编程方法,主要包括 VI 程序的建立、结构、数组和簇、图形显示、文件读写等内容;后 5 章从程序开发的实际应用角度,介绍了以 LabVIEW 作为开发平台的实用技术,主要包括局部变量和全局变量的定义与使用、外部应用程序接口、分析软件、数据采集和远程测控等。全书各章开头有该章的重点内容和学习目标。部分章后附有习题,以便于读者学习和掌握该章的重点内容。本书内容力求简单实用,以方便读者对 LabVIEW 快速入门和使用。

本书由童刚、樊春玲、崔凤英、孙四通编写,可作为高等院校测控技术与仪器、自动化等专业的教材或参考书,也可供相关专业的工程技术人员和软件工程师参考。

在本书的编写过程中得到了青岛科技大学相关部门的领导和同事们的支持,申庆花、宋蓓、崔凯、张城、童思博等对书稿进行了校对。在此,对上述提到的机构和人员以及对本书中所引用的参考文献的作者一并表示真诚的感谢。

本书配有电子课件,欢迎选用本书作教材的老师索取,索取邮箱:wbj@mail.machineinfo.gov.cn。

由于水平有限,再加上时间仓促,书中难免有疏漏和错误之处,恳请广大读者给予批评指正。

编 者

目 录

前言

第1章 绪论	1
1.1 虚拟仪器	1
1.1.1 虚拟仪器起源	1
1.1.2 虚拟仪器构成	2
1.1.3 虚拟仪器现状及应用	3
1.1.4 虚拟仪器C语言特点	3
1.2 LabVIEW 简介	3
1.3 启动 LabVIEW	4
1.4 创建虚拟仪器	6
1.4.1 创建虚拟仪器程序	6
1.4.2 调试虚拟仪器程序	7
1.4.3 创建和调用虚拟仪器子程序	8
1.5 虚拟温度测量仪实例	9
1.5.1 原理	9
1.5.2 设计步骤	10
习题	13
第2章 结构	14
2.1 循环结构	14
2.1.1 While 循环结构	14
2.1.2 For 循环结构	16
2.1.3 定时循环	22
2.2 选择结构	24
2.2.1 Case 结构的建立和组成	24
2.2.2 Case 结构分支的添加、 删除与排序	25
2.2.3 数据的输入和输出通道	26
2.3 顺序结构	26
2.3.1 顺序结构的创建与组成	27
2.3.2 顺序结构局部变量的创建	27
2.4 公式节点	30
2.4.1 公式节点的创建	30
2.4.2 公式节点语法	31
2.5 事件结构	33

2.5.1 事件结构的创建与组成	33
2.5.2 事件结构的设置	34
习题	34
第3章 数组、簇	36
3.1 数组	36
3.1.1 创建数组	36
3.1.2 利用循环的自动索引功能 创建数组	40
3.1.3 数组函数	41
3.2 簇	49
3.2.1 簇的创建	50
3.2.2 簇函数	51
习题	55
第4章 图形控件和图形数据显示	56
4.1 图线显示	57
4.1.1 Waveform Chart 组件及功能	57
4.1.2 Waveform Chart 的设置	60
4.1.3 Waveform Chart 的数据类型	63
4.2 图形显示	64
4.2.1 Waveform Graph 的组件及功能	64
4.2.2 Waveform Graph 的数据类型	66
4.3 XY Graph 和 Express XY Graph	68
4.4 Digital Waveform Graph	68
4.5 三维图形显示	69
4.5.1 Intensity Chart	69
4.5.2 Intensity Graph	71
4.5.3 3D Surface Graph	71
4.5.4 3D Parametric Graph	72
4.5.5 3D Curve Graph	72
4.6 其他图形的表达与显示	72
习题	73
第5章 字符串和文件存取	75
5.1 字符串	75

5.1.1 字符串控件的创建	75	习题	118
5.1.2 字符串控件的属性设置	77	第8章 分析软件	119
5.1.3 字符串函数	80	8.1 概述	119
5.2 文件存取	84	8.2 信号的产生	121
5.2.1 文件 I/O 子模板	85	8.2.1 标准频率	122
5.2.2 高层文件操作节点及应用	86	8.2.2 信号产生子模板简介	122
5.2.3 中层文件操作节点及应用	90	8.2.3 采样与混频	123
5.2.4 数据记录文件 I/O	92	8.2.4 波形产生子模板简介	124
5.2.5 波形文件 I/O	94	8.3 信号的频域分析	125
5.2.6 使用文件 I/O 功能函数的 几点说明	96	8.3.1 FFT 变换	126
习题	96	8.3.2 窗函数	129
第6章 实用编程技术	98	8.3.3 频谱分析	130
6.1 局部变量和全局变量	98	8.3.4 谐波分析	131
6.1.1 局部变量的建立	98	8.4 数字滤波	133
6.1.2 局部变量的使用方法	99	8.5 曲线拟合	136
6.1.3 全局变量的建立	100	习题	139
6.1.4 全局变量的使用方法	100	第9章 数据采集	140
6.1.5 使用局部变量和全局变量时需要 注意的问题	102	9.1 数据采集基础知识	140
6.2 属性节点	102	9.1.1 信号类型	140
6.2.1 属性节点的创建	102	9.1.2 采样频率、抗混叠滤波器和 样本数	141
6.2.2 属性节点的使用	104	9.1.3 数据采集系统构成	142
6.2.3 属性节点的特点	105	9.1.4 信号调理	143
6.3 菜单的编辑	105	9.1.5 数据采集卡及配置	144
6.3.1 通过菜单编辑器创建菜单	105	9.2 模拟输入	147
6.3.2 在框图程序中响应菜单操作	108	9.2.1 模拟输入的相关参数	147
习题	110	9.2.2 模拟输入模块分类及简介	148
第7章 外部应用接口	111	9.3 模拟输出	155
7.1 CIN	111	9.3.1 基本参数与术语	155
7.1.1 CIN 的概念	111	9.3.2 模拟输出模块介绍	156
7.1.2 CIN 的调用步骤	111	习题	159
7.2 Matlab	113	第10章 远程测控	160
7.2.1 Matlab 节点	113	10.1 远程测控系统的组成	160
7.2.2 LabVIEW 和 Matlab 数据类型	114	10.2 利用 DataSocket 技术实现测 控数据实时通信	161
7.2.3 Matlab 节点应用	115	10.2.1 DataSocket 的组成	161
7.3 DLL	116	10.2.2 DataSocket 子模板	162
7.3.1 DLL 的概念	116	10.2.3 DataSocket 应用实例	164
7.3.2 调用 DLL	117	10.3 利用网络协议进行测控	

数据通信	165	10.4.1 服务器端的 Web 发布配置	170
10.3.1 TCP 通信	165	10.4.2 客户端的远程测控	171
10.3.2 UDP 通信	167	习题	172
10.4 在 Web 上发布程序	169	参考文献	173

第 1 章 绪 论

本章首先简要介绍了虚拟仪器的起源、发展、现状和特点,其次介绍了 LabVIEW 的开发环境,有关的模板、控件和虚拟仪器的创建与调试,最后采用虚拟温度测试仪的例子详细介绍了创建过程。学习目标如下:

- 1) 了解虚拟仪器的有关知识。
- 2) 熟悉 LabVIEW 开发环境。
- 3) 掌握虚拟仪器程序 VI 的创建、编辑和调试过程。
- 4) 掌握虚拟仪器子程序 SubVI 的创建。

1.1 虚拟仪器

虚拟仪器是基于计算机的仪器。简单地说,虚拟仪器就是在通用计算机上加上软件和(或)硬件,使得使用者在操作这台计算机时,就像是在操作一台他自己设计的专用的电子仪器。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,任何一个用户都可以通过改写软件的办法,方便地改变和增减仪器系统的功能,即“软件就是仪器”。虚拟仪器技术的出现,打破了传统仪器由厂商定义功能,用户无法改变的固定模式,给了用户一个充分发挥自己才能和想象力的空间。用户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。同时,虚拟仪器在数据处理、显示、传送等方面的特点,也使用户可以方便地对其进行维护、扩展和升级等。

1.1.1 虚拟仪器起源

虚拟仪器是以计算机为核心的,是仪器系统与计算机软件技术的紧密结合。这种结合有两种方式,一种是将计算机装入仪器,就是所谓的智能仪器,随着计算机功能的日益强大及体积的日趋缩小,这类仪器功能也越来越强大,目前已经出现嵌入式系统的仪器。另一种方式是将仪器装入计算机,以通用的计算机硬件及操作系统为依托,实现各种仪器功能。虚拟仪器主要是指第二种方式。

美国国家仪器(NI, National Instrument)公司于 20 世纪 80 年代中期首先提出了“软件就是仪器”(The Software Is The Instrument)这一基于计算机技术的虚拟仪器概念。这个概念为用户定义、构造自己的仪器系统提供了完美的解决途径。虚拟仪器通过软件将计算机硬件资源与仪器硬件有机地融合为一体,从而把计算机强大的计算功能和仪器硬件的测量、控制能力结合在一起,大大缩小了仪器硬件的成本和体积,并通过软件实现对数据的显示、存储及分析处理。在硬件平台 I/O 接口设备与计算机确定之后,再编制上某种测量功能的软件,计算机就成为该种功能的测试仪器。即用户只需改变软件程序就可以不断赋予它或扩展增强它的测量功能。这就是说,仪器的设计制造不再是厂商的专利。虚拟仪器开创了仪器使用者可以成为仪器设计者的时代,这给仪器使用者带来了无尽的收益。

1.1.2 虚拟仪器构成

与传统仪器一样,虚拟仪器同样被划分为数据采集与控制、数据分析处理、结果表达3大功能模块,如图1-1所示。虚拟仪器以透明的方式把计算机资源和仪器硬件的测试能力结合起来,实现了仪器的功能运作。

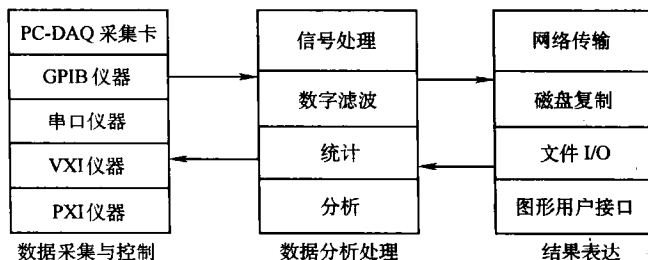


图 1-1 虚拟仪器的功能模块

虚拟仪器用各种图表或控件来虚拟传统仪器面板上的各种器件。由各种开关图标来实现仪器电源的通断;由各种按钮图标来设置被测信号的“放大倍数”、“通道”等参数;由各种显示控件以数值或波形的方式显示测量或分析结果;由计算机的鼠标和键盘操作来模拟传统仪器面板上的实际操作;用图形化的语言编程,实现各种信号的测量和数据分析。

虚拟仪器由硬件平台和应用软件两大部分构成。本节将分别从硬件和软件两个方面来介绍虚拟仪器的工作原理及具体构成。

虚拟仪器的组成如图1-2所示。

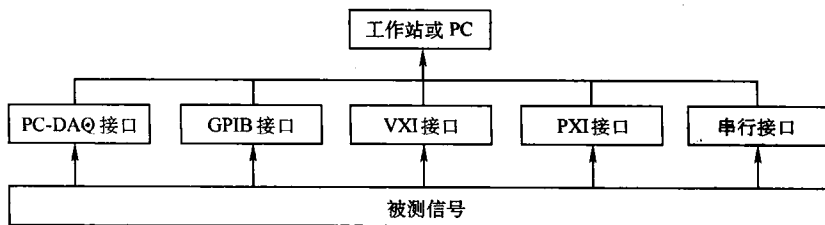


图 1-2 虚拟仪器的组成

1. 硬件平台

虚拟仪器的硬件平台由计算机和 I/O 接口设备组成,计算机是硬件平台的核心,一般是工作站,也可用普通的 PC。

I/O 接口设备负责被测信号的采集、调整、放大、模/数转换。常用的 I/O 接口设备有以下 5 种:

- 1) PC-DAQ 接口。PC-DAQ 接口采用计算机本身的 PCI 总线或 ISA 总线,将数据采集卡(DAQ),包括信号调整电路,插入计算机主板的空槽中即可。
- 2) GPIB 接口。GPIB 接口采用 GPIB 标准总线连接仪器与计算机。
- 3) VXI 接口。VXI 接口采用 VXI 标准总线连接仪器与计算机。
- 4) PXI 接口。PXI 接口采用 PXI 标准总线连接仪器与计算机。
- 5) 串行接口。串行接口采用 Serial 标准总线连接仪器与计算机。

以上5种接口中PC-DAQ接口是最基本、最常用的接口方式。

2. 软件平台

虚拟仪器软件将可选硬件(如DAQ、GPIB、RS-232、VXI、PXI)和可以重复使用的源码库函数等软件结合起来,实现模块间的通信、定时与触发,源码库函数为用户构造自己的虚拟仪器系统提供了基本的软件模块。当用户的测试要求变化时,可以方便地由用户自己来增减软件模块,以满足现有系统的测试要求。

虚拟仪器软件包括应用程序和I/O接口设备驱动程序。

应用程序又由实现虚拟仪器前面板功能的软件程序和定义测试功能流程图的软件程序两部分构成。

I/O接口设备驱动程序实现对特定外部硬件设备的控制。

1.1.3 虚拟仪器现状及应用

目前流行的虚拟仪器软件开发工具有两类。文本式编程语言有C、C++、VB、Lab windows/CVI等;图形化编程语言有LabVIEW、AgilentVEE等。其中LabVIEW最流行,是目前应用最广、发展最快、功能最强的图形化软件。

虚拟仪器作为新兴的仪器仪表,用户可以定义其结构和功能,构建灵活,转变容易,因此它在各个领域尤其是在超大规模集成电路测试、工厂测试、现代家用电器测试,以及军事、航空、航天、通信、汽车、半导体和生物医学等领域得到了广泛应用。

1.1.4 虚拟仪器G语言特点

G语言是一种图形化的程序语言。使用这种语言编程时,基本上不写程序代码,取而代之的是框图程序或流程图。它尽可能利用技术人员、科学家、工程师所熟悉的术语、图表和概念,因此,LabVIEW是一个面向终端用户的工具。它既可以增强用户构建工程系统的能力,又提供了实现仪器编程和数据采集的便捷途径。使用它进行原理研究、设计、测试并实现仪器系统功能时,可以大大提高工作效率。

G语言是一种通用编程语言,具有通用函数库。它和常规的文本式编程语言一样,定义了数据类型、结构类型和模块调用等规则。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,并且还包括各种常用的程序调试工具,如支持断点设置、单步调试、动态显示程序执行流程、数据变化监视等功能。

G语言具有较好的模块化性能。G语言编写的程序被称为VI,由交互式用户接口、数据流框图和图标连接端口构成。一个VI既可以作为独立程序,也可以作为其他程序的子程序。VI作为子程序时被称为SubVI。VI与SubVI之间的参数传递由图标连接端口来实现。

1.2 LabVIEW 简介

LabVIEW是Laboratory Virtual Instrument Engineering Workbench(实验室虚拟仪器集成环境)的简称,是美国国家仪器公司的创新软件产品,被誉为“科学家与工程师”的语言。LabVIEW是一个软件开发环境,它是一种不同于常规文本式编程语言的图形化编程工具,它为不熟悉文本语言编程的设计者在测控领域建立计算机仪器系统提供了便捷、轻松的图形

化设计开发集成环境。正是因为 LabVIEW 的成功,才使虚拟仪器的概念为学术界和工程界广泛接受;反过来也正是因为虚拟仪器概念的延伸与扩展,才使 LabVIEW 的应用更加广泛。目前,已经发展到以最新版本 LabVIEW 8.0 为核心,包括控制与仿真、高级数字信号处理、统计过程控制、模糊控制和 PID 控制等众多附加软件包,运行于 Windows、Linux、Macintosh、Sun 和 Unix 等多种平台的工业标准软件开发环境。

LabVIEW 已经广泛地被工业界、学术界和研究实验室所接受,被公认为是标准的数据采集和仪器控制软件。LabVIEW 不仅提供了遵从 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,还内置了支持 TCP/IP、ActiveX 等软件标准的库函数,而且其图形化的编程界面使编程过程变得生动有趣。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。

以 LabVIEW 为代表的图形化程序语言,又称为 G 语言。使用这种语言编程时,基本上不需要编写程序代码,而是“绘制”程序流程图。利用 LabVIEW,可产生独立运行的可执行文件。

1.3 启动 LabVIEW

从“开始”菜单中选择程序 National Instruments LabVIEW 7.1 运行,在计算机屏幕上将出现如图 1-3 所示的 LabVIEW 欢迎窗口。

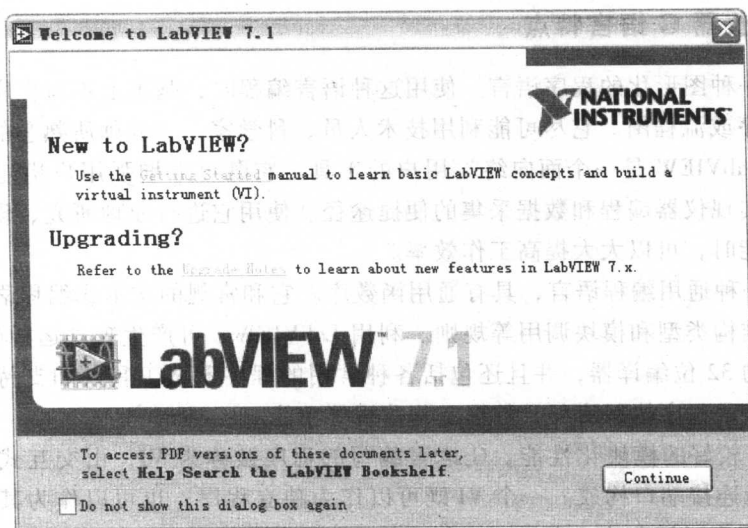


图 1-3 LabVIEW 欢迎窗口

单击 Continue 按钮之后进入如图 1-4 所示的 LabVIEW 对话框,在这个对话框中可以直接访问某些 LabVIEW 资源和工具,而不需要打开程序窗口。

在 LabVIEW 对话框中单击 New 按钮,将弹出如图 1-5 所示的 New 对话框。

在 New 对话框左边的 Create New 区中,显示出树形控件,用于选择新建文档类型。其中,Blank VI 用于建立一个新程序;VI from Template 按类型列出 LabVIEW 系统提供的程序模板,用户可以以这些模板为基础,建立自己的程序;Other Document Types 列出其他文件

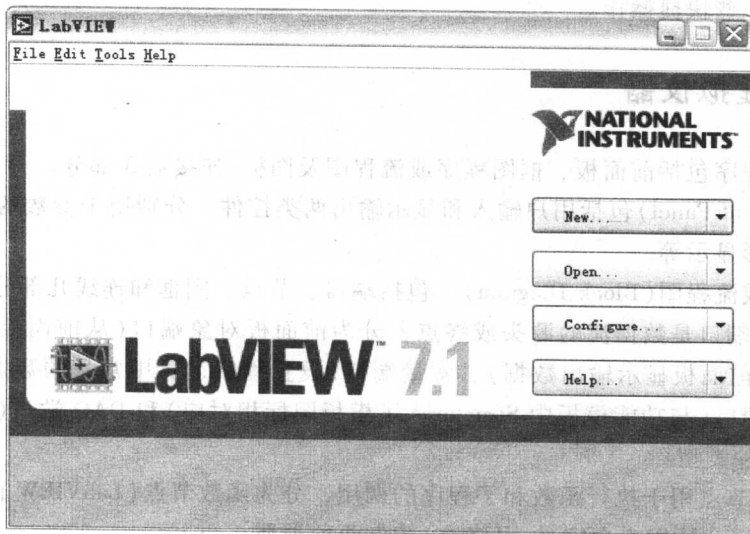


图 1-4 LabVIEW 对话框

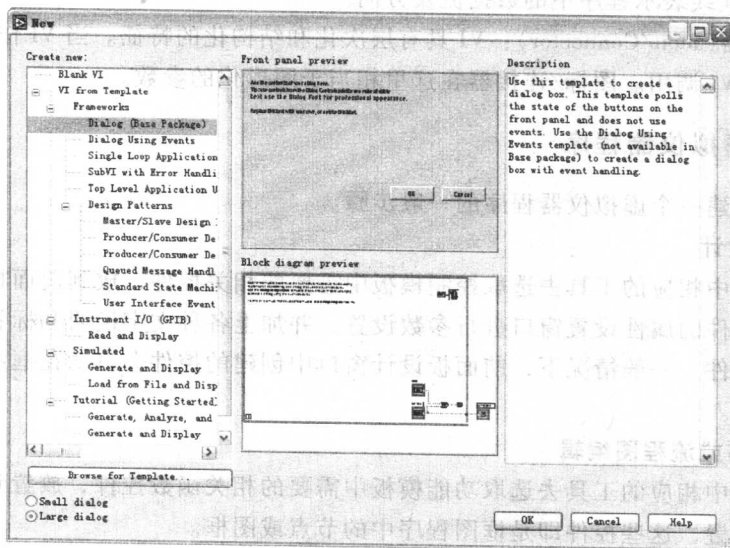


图 1-5 New 对话框

类型，比如全局变量、自定义控件、运行时菜单等。当选中某一个模板 VI 时，Front panel preview 和 Block diagram preview 子窗口给出其前面板和框图预览。Description 子窗口给出所选文档类型的简单描述。选中文档类型后，单击 OK 按钮，将打开对应的新文档窗口。

单击图 1-4 中 LabVIEW 对话框的 Open 按钮，将弹出标准的 Windows 打开文件对话框，在这里可以打开各种 LabVIEW 文件。Configure 按钮用于打开 Measurement & Automation Explorer，进行系统软硬件的配置，详见后面章节。Help 按钮用于打开 LabVIEW Help 窗口，可以在其中查阅详细的帮助信息。

LabVIEW 对话框的 4 个按钮右侧都有黑色向下的三角形，单击这些三角形可以打开下

拉列表,完成一些快捷操作。

1.4 创建虚拟仪器

虚拟仪器程序包括前面板、框图程序或流程图及图标/连接器 3 部分。

前面板(Front Panel)包括用户输入和显示输出两类控件,分别用于参数的设置和测量结果的数值、波形显示等。

框图程序或流程图(Block Diagram),包括端口、节点、图框和连线几部分,简介如下。

1) 端口。端口是数据流的源头或终点,分为前面板对象端口(从前面板用户输入控件获得数据,或向前面板显示输出数据)、常量端口(设置程序运算中常量的数据源端口)、全局与局部变量端口(与功能模板中 Structures 子模板图标相对应)和 DAQ 端口(从数据采集卡中获得数据)4 种类型。

2) 节点。节点用于执行函数和子程序的调用,分为函数节点(LabVIEW 函数库提供,用户不可修改)和子 VI(用户自定义,可修改)节点两种类型。

3) 图框。图框用于执行结构化程序控制命令,如 Case 等。

4) 连线。连线表示程序中的数据流及方向。

图标/连接器(Icon/Connector): VI 具有层次化和结构化的特征。当 VI 作为子程序(Sub-VI)时,被其他 VI 调用。图标/连接器在这里相当于图形化的参数。

1.4.1 创建虚拟仪器程序

本节介绍创建一个虚拟仪器程序的一般步骤。

1. 前面板设计

用工具模板中相应的工具去选取控制模板中需要的相关控件,放到前面板设计窗口的合适位置,打开控件的属性设置窗口进行参数设置,并加上各种文字说明和标签,也可以加入一些装饰用的控件。一般情况下,前面板设计窗口中创建的控件会自动在框图程序编辑窗口创建相应的端口。

2. 框图程序或流程图编辑

用工具模板中相应的工具去选取功能模板中需要的相关函数控件,放置到框图程序编辑窗口中的合适位置,这些控件即是框图程序中的节点或图框。

3. 数据流的编辑

数据流的编辑就是连线操作。连线也是程序设计中较为复杂的问题。框图程序上的每一个对象都带有自己的连线端子,连线将构成对象之间的数据通道。因为这不是几何意义上的连线,并非任意两个端子间都可连接。连线类似于高级文本语言程序中的变量,数据单向流动,从源端口向一个或多个目的端口流动。不同的线型代表不同的数据类型:数值的标量用普通细实线表示,一维数组用粗实线表示,二维数组用两根平行的细实线表示;逻辑量、字符串及文件路径的标量用普通波浪线表示,一维数组用两根方向相对且平行的波浪线表示,二维数组用两根方向相同且平行的波浪线表示。其中整型数用蓝色表示,浮点数用橙色表示,逻辑量用绿色表示,字符串用粉色表示,文件路径用青色表示。一些常用数据类型所对应的线型和颜色如图 1-6 所示。

	标量	一维数组	二维数组	
整型数				蓝色
浮点数				橙色
逻辑量				绿色
字符串				粉色
文件路径				青色

图 1-6 一些常用数据类型所对应的线型和颜色

当需要连接两个端点时，在第一个端点上单击连线工具(从工具模板中调用)，然后移动到另一个端点，再单击第二个端点。端点的先后次序不影响数据流动的方向。

当把连线工具放在端点上时，该端点区域将会闪烁，表示连线将会接通该端点。当把连线工具从一个端口接到另一个端口时，不需要按住鼠标键。当需要连线转弯时，单击鼠标左键，即可正交垂直方向地弯曲连线。按空格键可以改变转角的方向。

接线头是为了帮助正确连接端口的连线。当把连线工具放到端口上，接线头就会弹出。接线头还有一个黄色的小标识框，以显示该端口的名字。

线型为波折型的连线表示坏线。出现坏线的原因有很多，例如连接了两个控制对象，源端口和终点端口的数据类型不匹配等。可以通过使用定位工具单击坏线，再按下 Delete 键来删除它。当 VI 无法运行，或者显示 Signal has loose ends (信号丢失终端)的错误信息时，选择 Edit→Remove Broken Wires 菜单命令或者按下 Ctrl + B 键可以一次性删除框图程序中的所有坏线。

4. 功能检验

功能检验包括如下两种检验方式：

1) 仿真检验。不使用外部接口的硬件设备采集数据，用数组和信号发生函数产生的信号数据作为信号源来检验虚拟仪器程序的功能。

2) 实测检验。用外部接口的硬件设备采集的数据作为信号源检验 VI 的功能。

5. 保存文件

将设计完成的虚拟仪器程序命名并保存为 VI 文件。至此即完成了虚拟仪器程序的创建工作。

1.4.2 调试虚拟仪器程序

虚拟仪器程序的一般调试步骤如下。

1. 清除语法错误

如果程序中存在语法错误，单击  按钮运行程序时，该按钮会变成 ，程序无法执行；再单击  按钮，则弹出一个窗口，显示语法错误清单；单击其中的错误，再单击 Show Error 按钮，则出错的对象或端口将会闪烁。

2. 跟踪程序的执行

单击按钮, 该按钮将会变成; 再单击按钮, 程序将慢速地以动画的方式执行, 并显示数据流上的数值, 这样即可跟踪程序的执行。

3. 设置断点单步调试

用工具模板中的工具单击希望设置或清除断点的地方, 即完成了断点的设置和清除。流程图中的节点或图框设为断点后就变成红框, 连线设为断点就加上红点, 当 VI 程序执行到断点时, 程序暂停, 断点处的节点变为闪烁; 单击按钮, 则执行该节点, 下一个节点变为闪烁; 单击按钮, 则连续执行到下一个断点。

4. 设置探针

用工具模板中的工具单击希望设置探针的连接线, 或者用工具模板中的工具或工具单击希望设置探针的连接线, 将弹出一个对话框, 选择 Probe 选项, 即完成了探针的设置。随后会出现一个探针窗口, 显示被设置探针连接线上的数据值。

5. 观察数据

单击按钮, 即可观察数据流中各个节点的数值。

1.4.3 创建和调用虚拟仪器子程序

虚拟仪器子程序(SubVI)相当于普通编程语言中的子程序, 也就是被其他 VI 调用的子 VI。子程序的创建有两种方法: ①任何一个定义了图标和连接口的 VI 都可以作为另一个 VI 的子程序; ②用鼠标选择工具选择程序中的一部分作为子程序, 方法是选择要作为子程序的部分, 然后在 Edit 菜单中选择 Create SubVI 菜单命令即可。

构造一个子 VI 的主要工作就是定义它的图标和连接口。每个 VI 在前面板和框图程序编辑窗口的右上角都显示了一个默认的图标。用鼠标右键单击前面板窗口右上角的默认图标, 在弹出快捷菜单中选择 Edit Icon 菜单命令, 即弹出如图 1-7 所示的图标编辑器窗口, 可以用窗口左边的各种工具设计图标编辑区中的图标形状。在主 VI 的框图程序中用该图标代替子 VI。

连接口是 VI 的数据输入/输出接口。若调用该程序时, 前面板输入控件或者输出控件从子 VI 中输入或者输出数据, 则这些控件都需要在连接口面板中设置一个连线端子。用户可以通过选择 VI 的端子数并为每个端子指定对应的前面板对象以定义连接口。

用鼠标右键单击前面板窗口右上角的图标窗口, 在快捷菜单中选择 Show Connector 菜单命令。则连接器窗口会取代前面板窗口右上角的图标。LabVIEW 自动选择的端子连接模式是控制对象的端子位于连接器窗口的左边, 显示对象的端子位于连接器窗口的右边。选择的端子数取决于前面板中控制对象和显示对象的个数。

连接器中的各个矩形表示各个端子, 可以用它们从 VI 中输入或者输出数据。也可以选择另外一种端子连接模式。在图标上单击鼠标右键弹出快捷菜单, 选择 Show Connector 菜单命令, 再次弹出快捷菜单, 选择 Patterns 菜单命令即可设置用于输入和输出的端子数及选择各种连接模式。

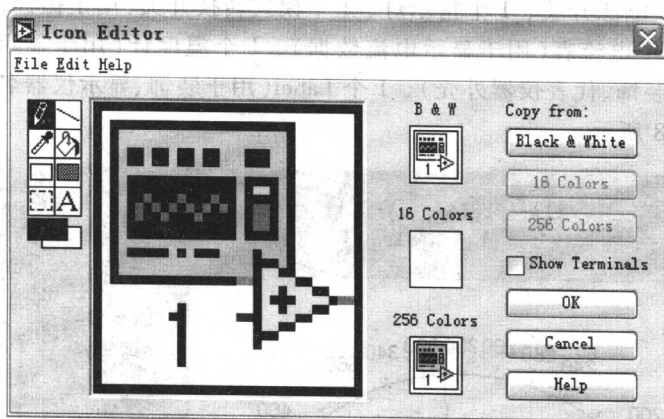


图 1-7 图标编辑器窗口

用鼠标单击连接器窗口中的一个矩形(代表一个端子),光标自动变成连线工具;同时端子变成黑色,再单击需要连接的前面板对象,此时该前面板对象被虚线框包围,选中的端子的颜色变为与该前面板对象的数据类型一致的颜色,并有阴影;再单击前面板中的任何空白区域以后,虚线消失,选中端子的阴影消失,这表示该前面板对象和端子连接成功。如果端子是白色的,则表示没有连接成功。待所有端子都与相关对象连接成功后,用鼠标右键单击连接器,在快捷菜单中选择 Show Icon 菜单命令,则连接器窗口恢复成先前编辑的 VI 图标。

最后选择 File→Save 菜单命令,保存该 VI,即完成了子 VI 的创建。在流程图中选择工具模板中的  (Select a VI),就可以选择需要的子 VI 供调用。

1.5 虚拟温度测量仪实例

本节将以虚拟温度测量仪为例介绍 LabVIEW7.1 开发虚拟仪器的方法。

1.5.1 原理

实际的温度测量仪有多种测量温度的方法,如使用红外温度传感器、红外摄像头等。本例中采用最常用的温度传感器——AD590 集成温度传感器。AD590 在一定的温度范围内,可以将温度数据线性变换为电流信号,其转换公式为

$$I = k \cdot \text{temp} \quad (1-1)$$

式中, I 为电流; temp 为温度; k 为温度系数。

整个温度测量仪的原理为: AD590 将温度数据转换为电流信号,电流信号经过模/数转换变为计算机可以识别的数字信号,再由本例实现的虚拟温度测量仪来显示电流数据,计算出温度数据并显示出来。

为了设计方便,用一个随机数据代替温度传感器输出的电流数据,同时假设 $k = 1\mu\text{A}/\text{K}$,假定 AD590 的线性温度范围为 $0 \sim 100^\circ\text{C}$,即 $273.15 \sim 373.15\text{K}$,则随机数据的范围应该为 $273.15 \sim 373.15\text{K}$,电流数据的范围也应为 $273.15 \sim 373.15\mu\text{A}$ 。在本例中共有 7

个前面板控件：1 个电源开关、1 个指示灯、1 个模式转换开关(用于摄氏温度和华氏温度显示方式的切换)、1 个电流表(用于显示电流数据)、1 个温度计(用于显示温度数据)、1 个 Raised Frame(用于装饰、代表仪器外壳)、1 个 Label(用于装饰、显示仪器名称)。虚拟温度测量仪前面板如图 1-8 所示。

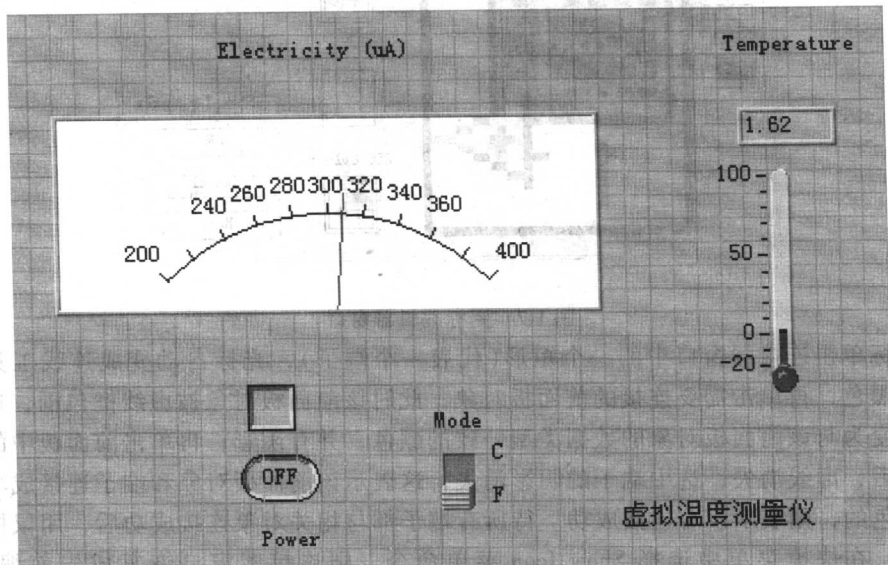


图 1-8 虚拟温度测量仪前面板

1.5.2 设计步骤

1. 前面板设计

在主窗口中选择 New→Blank VI 命令，新建一个空白 VI，然后开始如下操作。

(1) 放置电源开关 用工具模板中的  工具选择控制模板中的 All Controls→Classic Controls→Classic Boolean→Labeled Oblong Button 控件，放到前面板设计窗口中合适的位置。用工具模板中的  工具将控件的标签(Label)改为 Power，并将标签移动到合适的位置。

(2) 放置电源指示灯 用工具模板中的  工具选择控制模板中的 All Controls→Classic Controls→Classic Boolean→Square Light 控件，放到前面板设计窗口合适的位置。在控件上单击鼠标右键，在弹出的快捷菜单中选择 Visible Items→Label 命令，去掉控件的标签。

(3) 放置模式转换开关 用工具模板中的  工具选择控制模板中的 All Controls→Classic Controls→Classic Boolean→Vertical Switch 控件，放到前面板设计窗口中合适的位置。用工具模板中的  工具将控件的标签 Label 改为 Mode，并将标签移到合适的位置。用工具模板中的  工具在控件右侧添加两个标签，分别为 C 和 F，代表摄氏温度和华氏温度方式。

(4) 放置电流表 用工具模板中的  工具选择控制模板中的 All Controls→Classic Controls→Classic Numeric→Meter 控件，放到前面板设计窗口中合适的位置。用工具模板中的