

LabVIEW

虚拟仪器程序设计与应用

刘其和 李云明 编著



化学工业出版社

刘其和 李云明 编著

LabVIEW

虚拟仪器程序设计与应用



化学工业出版社

· 北京 ·

图书在版编目 (CIP) 数据

LabVIEW 虚拟仪器程序设计与应用/刘其和, 李云明
编著. —北京: 化学工业出版社, 2011. 3
ISBN 978-7-122-10332-1

I. L… II. ①刘… ②李… III. 软件工具, LabVIEW-
程序设计 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2011) 第 001170 号



责任编辑: 宋 辉
责任校对: 吴 静

文字编辑: 徐娜华
装帧设计: 王晓宇

出版发行: 化学工业出版社 (北京市东城区青年湖南街 13 号 邮政编码 100011)
印 装: 大厂聚鑫印刷有限责任公司
787mm×1092mm 1/16 印张 18¼ 字数 454 千字 2011 年 4 月北京第 1 版第 1 次印刷

购书咨询: 010-64518888 (传真: 010-64519686) 售后服务: 010-64518899
网 址: <http://www.cip.com.cn>
凡购买本书, 如有缺损质量问题, 本社销售中心负责调换。

定 价: 48.00 元

版权所有 违者必究

FOREWORD

前言



LabVIEW

虚拟仪器程序设计与应用

虚拟仪器是当前测控领域的热点技术，它代表了未来仪器技术的发展方向。LabVIEW 是优秀的虚拟仪器软件开发平台，在 LabVIEW 环境下可高效地进行数据管理、科学计算等方面应用程序的开发，尤其是测控系统及虚拟仪器的开发。自美国 NI 公司于 1986 年推出 LabVIEW1.0 版后，经过版本的不断升级，现已推出 LabVIEW 2010 版。LabVIEW 2010 版是真正意义上的简体中文版。

LabVIEW 图形化的编程语言极大地提高了开发虚拟仪器的效率。据统计，相对于文本编程语言而言，使用 LabVIEW 开发虚拟仪器，开发效率可以提高 10~15 倍，丝毫不影响程序的执行速度，同时 LabVIEW 在信号处理等方面的强大功能是目前的组态软件不可比拟的。

全书共 13 章，内容深入浅出、图文并茂，各章之间既相互联系又相对独立，读者可根据自己需要选择阅读。考虑到实际应用的频度，本书对仪器控制未加介绍；在数据采集方面考虑到采集硬件的不断发展，舍弃了对传统 DAQ 的介绍；基于网络的测控是测控领域的发展方向，LabVIEW 提供了若干网络通信技术，本书第 8 章着重介绍了 LabVIEW 中的网络通信技术的应用。第 6 章介绍了 LabVIEW 的 I/O 文件，但在实际测量中测量数据往往使用数据库记录，因此本书的第 9 章介绍了在 LabVIEW 中使用数据库的技术。

本书主要面向 LabVIEW 的初、中级用户。

本书第 1、2、4 章由李云明编写，第 3、5~13 章由刘其和编写。全书由刘其和统稿。

在本书编写过程中，魏龙、冯秀、金良、闻雯等给予了大力指导和支持，在此一并表示感谢。

由于编著者水平有限，书中难免有疏漏和不足之处，恳请广大读者批评指正。

编著者

CONTENTS

目录



LabVIEW

虚拟仪器程序设计与应用

第1篇 入门知识和编程基本方法

第1章 LabVIEW 概述

2

- 1.1 LabVIEW 简介 2
- 1.2 G 语言与虚拟仪器 3
 - 1.2.1 虚拟仪器的构成 3
 - 1.2.2 虚拟仪器的特点 3
- 1.3 LabVIEW 2010 的系统配置 4
- 1.4 LabVIEW 2010 的安装与运行 4
- 1.5 LabVIEW 2010 软件的启动方法及步骤 5
- 1.6 LabVIEW 2010 编程环境 6
 - 1.6.1 项目浏览器 6
 - 1.6.2 LabVIEW 2010 的编辑界面 6
 - 1.6.3 LabVIEW 2010 工具栏 6
 - 1.6.4 LabVIEW 2010 菜单栏 7
- 1.7 工具模板 7
- 1.8 控件选板和函数模板 8
- 1.9 LabVIEW 2010 简体中文版的帮助系统 8
 - 1.9.1 使用即时帮助 8
 - 1.9.2 使用目录和索引查找在线帮助 9
 - 1.9.3 查找 LabVIEW 范例 9

第2章 LabVIEW 编程初步

10

- 2.1 项目 10
 - 2.1.1 创建项目文件 10
 - 2.1.2 在项目中新建或添加编程元素 10
 - 2.1.3 在项目中添加编程元素 11
- 2.2 VI 前面板设计 11
 - 2.2.1 控件的类型 11

2.2.2	控件模板	11
2.2.3	控件的设置	13
2.3	程序框图	16
2.3.1	函数节点	16
2.3.2	端口	16
2.3.3	数据连线	17
2.3.4	函数模板	18
2.4	创建 VI	20
2.4.1	创建前面板	20
2.4.2	创建框图程序	22
2.4.3	创建 VI 图标	22
2.4.4	保存 VI	23
2.5	编辑 VI	23
2.6	子 VI	26
2.6.1	创建子 VI	26
2.6.2	调用子 VI	28
2.7	快速 VI	29
2.7.1	快速 VI 的特点	29
2.7.2	由快速 VI 创建子 VI	30
2.7.3	动态数据类型	30
2.7.4	Express VI 的使用方法	31
2.8	多态	31
2.8.1	函数的多态	32
2.8.2	多态 VI	32
2.9	运行和调试 VI	33
2.9.1	运行 VI	33
2.9.2	调试 VI	33
2.10	数据操作	35
2.10.1	数据类型	36
2.10.2	基本数学运算	38
2.10.3	布尔运算	41
2.10.4	比较运算	42

第3章 控制程序运行的结构

45

3.1	For 循环	45
3.1.1	For 循环的建立	45
3.1.2	For 循环的时间控制与数据传递	46
3.1.3	For 循环对数组的自动索引	47
3.2	移位寄存器和反馈节点	48
3.2.1	移位寄存器	48
3.2.2	反馈节点	49

3.3 While 循环	50
3.4 条件结构 (Case Structure)	51
3.4.1 条件结构的建立	51
3.4.2 条件结构的设置	52
3.4.3 条件结构的应用示例	53
3.5 顺序结构	54
3.5.1 顺序结构的建立	54
3.5.2 顺序结构中数据输入、输出与传递	55
3.5.3 顺序结构应用示例	55
3.5.4 顺序结构的缺陷与人为的数据依从关系	56
3.6 事件结构 (Event Structure)	57
3.6.1 事件驱动的概念	57
3.6.2 事件结构的建立	58
3.6.3 用户界面事件的分类与注册	58
3.6.4 事件结构的设置	58
3.6.5 通知事件和过滤事件的区别	59
3.6.6 其他事件应用示例	60
3.7 程序框图禁用结构	62
3.8 条件禁用结构	62
3.9 公式节点	65
3.9.1 公式节点的用途	65
3.9.2 公式节点的建立	66
3.9.3 公式节点的语法	66
3.9.4 公式节点在数值计算中的应用	68
3.10 表达式节点	68
3.11 属性节点	69
3.11.1 属性节点的创建	69
3.11.2 属性节点的使用	71
3.11.3 属性节点的特点	73
3.11.4 属性节点的应用	73

第4章 集合类型的数据与变量

75

4.1 数组	75
4.1.1 数组的概念	75
4.1.2 创建数组的方法	76
4.1.3 数组函数	77
4.2 簇	82
4.2.1 簇的概念	82
4.2.2 簇的创建	82
4.2.3 簇函数	83
4.3 字符串	85

4.3.1 字符串的概念	85
4.3.2 字符串控件	85
4.3.3 字符串函数	89
4.4 波形	94
4.4.1 波形的概念	94
4.4.2 波形的创建	94
4.4.3 波形的属性	95
4.4.4 数字波形	95
4.5 本地变量和全局变量	97
4.5.1 本地变量	97
4.5.2 全局变量	100

第5章 图表和图形

103

5.1 实时趋势图控件	104
5.1.1 波形图表 (Waveform Chart)	104
5.1.2 实时趋势图控件的功能	104
5.1.3 实时趋势图控件的应用	110
5.2 事后记录波形控件	111
5.2.1 波形图	111
5.2.2 事后记录波形控件的功能	112
5.2.3 事后记录波形控件的应用	112
5.3 XY 图	112
5.4 强度图形显示控件 (Intensity Graph)	114
5.4.1 使用强度图	114
5.4.2 定义强度图的颜色	115
5.4.3 设置强度图的外观	116
5.5 强度图表	116
5.5.1 强度趋势图控件的功能	116
5.5.2 强度图表控件的应用	117
5.6 三维图形显示控件	117
5.6.1 三维曲面图形	118
5.6.2 三维参数曲面图	122
5.6.3 三维曲线图	123

第6章 文件输入输出

124

6.1 基本概念	124
6.1.1 路径	124
6.1.2 引用句柄	124
6.1.3 文件 I/O 的出错管理	125
6.1.4 LabVIEW 的文件格式	125
6.1.5 文件 I/O 操作流程控制	126

6.2 文件操作.....	126
6.2.1 文件的基本操作.....	126
6.2.2 文件输入输出函数.....	126
6.3 文件输入输出.....	127
6.3.1 文本文件的输入输出.....	127
6.3.2 二进制文件的输入输出.....	129
6.3.3 数据记录文件的读写.....	131
6.3.4 电子表格格式文件的输入输出.....	133
6.3.5 波形文件输入输出.....	134

第7章 程序的动态控制

137

7.1 VI 服务器概述.....	137
7.2 VI 服务器技术简介.....	137
7.3 配置 VI 服务器.....	139
7.4 如何获得对象的引用句柄.....	140
7.4.1 获得应用程序的引用句柄.....	140
7.4.2 获得 VI 的引用句柄.....	141
7.4.3 获得控件的引用句柄.....	141
7.4.4 引用句柄控件子模板.....	142
7.4.5 动态加载 VI.....	142
7.5 动态加载 VI 的程序.....	142
7.6 动态控制 VI 运行.....	144
7.6.1 动态刷新被控 VI 前面板控件值.....	144
7.6.2 选择性打开 VI 前面板.....	147
7.6.3 子面板设计.....	148
7.6.4 动态控制 VI 属性.....	149
7.7 运行菜单控制.....	150
7.7.1 运行菜单的设置.....	150
7.7.2 用程序代码进行运行菜单设置.....	151

第2篇 网络及数据库的应用

第8章 网络通信技术

158

8.1 DataSocket 技术.....	158
8.1.1 概述.....	158
8.1.2 DataSocket 的构成.....	158
8.1.3 DataSocket 节点.....	160
8.1.4 DataSocket 的数据传输.....	163
8.2 TCP 协议的应用.....	169
8.2.1 TCP 节点.....	169

8.2.2 利用 TCP 协议进行双机通信	170
8.3 在 Web 上发布程序	173
8.3.1 在 Web 上发布 LabVIEW 程序的设置	173
8.3.2 在 Web 上发布 HTML 文件	176
8.4 前面板的远程连接	179
8.5 共享变量	182
8.5.1 创建共享变量	183
8.5.2 在程序框图上放置共享变量的方法	184
8.5.3 单进程共享变量的使用	184
8.5.4 共享变量用于网络通信	185
8.5.5 共享变量的网络应用	186

第 9 章 LabVIEW 中数据库访问技术

191

9.1 LabVIEW 中数据库访问途径	191
9.2 数据库操纵语言 SQL	192
9.3 ADO 编程基础	195
9.3.1 ADO 模型	195
9.3.2 ADO 编程过程	196
9.4 LabVIEW 中 ADO 编程	197
9.4.1 LabVIEW 中 ADO 编程使用的前面板对象及函数	197
9.4.2 生成 Connection、Recordset 和 Command 对象	198
9.4.3 打开、关闭数据库连接	199
9.4.4 记录集赋值	201
9.4.5 在数据源中创建表	202
9.4.6 读取当前记录的数据	202
9.4.7 查询记录	204
9.4.8 添加新记录	204
9.4.9 修改记录	205
9.4.10 删除记录	205

第 3 篇 测控基础知识

第 10 章 数据采集

208

10.1 数据采集系统 (DAQ) 的构成	208
10.2 信号类型	209
10.2.1 数字信号	210
10.2.2 模拟直流信号	210
10.2.3 模拟时域信号	210
10.2.4 模拟频域信号	211
10.2.5 一个信号的五种测量角度	211

10.3	数据采集的过程	211
10.4	数据采集卡的主要指标	212
10.4.1	采样率	212
10.4.2	分辨率	213
10.4.3	合理设置设备量程范围	213
10.4.4	合理进行信号极限设置	214
10.5	被测试信号的连接方式	214
10.5.1	信号的参考点	214
10.5.2	信号的连接方式	214
10.5.3	模拟输入连接方式小结	215
10.6	基于 LabVIEW 的数据采集系统总体结构	216
10.7	数据采集设备的设置与测试	217
10.7.1	测试与自动化资源管理器	217
10.7.2	在 NI-DAQmx 系统中进行设备设置与测试	218

第 11 章 模拟信号的输入输出

220

11.1	设置数据采集任务和采集通道	220
11.1.1	全局虚拟物理通道的建立	220
11.1.2	数据采集任务的建立	222
11.2	DAQmx 采集数据的组织	225
11.3	在 DAQmx 系统中创建应用程序	226
11.3.1	在 LabVIEW 环境中打开数据采集助手	226
11.3.2	数据采集助手转换为任务	226
11.4	生成图形代码	227
11.4.1	通过任务生成程序代码	227
11.4.2	将采集助手快速 VI 转换为程序代码	229
11.5	DAQmx 数据采集函数	230
11.6	使用 DAQmx 测量直流电压信号	234
11.7	使用 DAQmx 的方法采集波形	235
11.7.1	采集有限数量的数据	235
11.7.2	连续采集数据	236
11.8	模拟信号频率测量	236
11.8.1	模拟频率、采样率和奈奎斯特 (Nyquist) 定理	236
11.8.2	模拟信号频率的测量	237
11.9	模拟输出概述	238
11.9.1	输出直流信号	238
11.9.2	模拟波形输出	238
11.9.3	模拟输出的信号连接	238
11.10	输出直流信号	239
11.11	输出波形信号	239
11.11.1	有限数量波形数据输出	239

11.11.2 连续波形数据输出.....	240
-----------------------	-----

第 12 章 数字输入输出与计数器	242
--------------------------	------------

12.1 数字信号输入输出概述.....	242
12.1.1 数字信号的概念.....	242
12.1.2 数字端口与数字线.....	242
12.1.3 数字信号输入输出设备.....	242
12.1.4 数字信号输入输出方式.....	242
12.2 DAQmx 的数字信号输入输出.....	243
12.2.1 无条件方式数字输入输出.....	243
12.2.2 握手方式数字输入输出.....	244
12.3 DAQmx 的计数器输入输出.....	245
12.3.1 计数器基础知识.....	245
12.3.2 用计数器输出脉冲信号.....	247
12.3.3 测量 TTL 信号频率.....	248
12.3.4 边沿计数.....	250

第 13 章 LabVIEW 的应用	252
---------------------------	------------

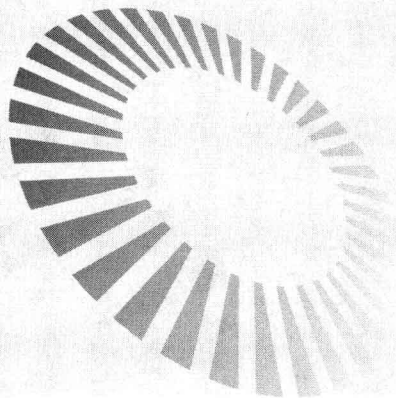
13.1 基于 LabVIEW 的机械密封数据采集系统的设计.....	252
13.1.1 系统需求分析.....	252
13.1.2 系统的硬件构成.....	252
13.1.3 系统采集软件的设计.....	253
13.1.4 采集程序设计.....	256
13.2 基于 LabVIEW 的机械密封端面接触特性数值模拟软件的开发.....	267
13.2.1 数学模型：基于分形的旋转机械密封端面接触模型.....	267
13.2.2 程序设计.....	269
13.2.3 在 LabVIEW 中如何进行数值逼近求解.....	273
13.2.4 数据输出.....	276

参考文献	277
-------------	------------



第 1 篇

入门知识和编程基本方法



第 1 章

LabVIEW概述

1.1 LabVIEW 简介

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。LabVIEW 已广泛地被工业界、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 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 采用的是数据流编程方式,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。

选择 LabVIEW 开发测量应用程序的一个决定性因素是它的开发效率,与一般文本语言相比开发效率至少提高 90%。

与文本编程语言相比 LabVIEW 的主要优势体现在如下几个方面。

- 提供了丰富的图形化编程控件,采用图形化编程方法,把开发者从复杂枯燥的文本编程工作中解脱出来。
- 采用数据流编程,自动实现了多线程处理,充分利用了多处理器的处理能力。
- 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,能立即在显示器上显示出来。
- 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。
- 内建了大量的分析函数用于数据分析和信号处理。
- 通过应用程序生成器可以很轻松将所编写的程序以可执行应用程序、动态链接库或安装包的形式发布。
- 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进行轻松连接。

NI 提供了丰富的附加模块,用于扩展 LabVIEW 在不同领域中的应用。如实时模块、PDA 模块、运动模块、机器视觉模块、DAQmx 等。



1.2 G 语言与虚拟仪器

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,取而代之的是流程图或框图。它尽可能利用了工程技术人员所熟悉的术语、图标和概念,因此,LabVIEW 是一个面向最终用户的工具。它可以增强构建自己的科学和工程系统的能力,提供了实现仪器编程和数据采集系统的便捷途径。使用它进行原理研究、设计、测试并实现仪器系统时,可以大大提高工作效率。

虚拟仪器实际上是一个按照仪器需求组织的数据采集系统。虚拟仪器的研究中涉及的基础理论主要有计算机数据采集和数字信号处理。目前在这一领域内,使用较为广泛或者说首选的 G 语言是美国 NI 公司的 LabVIEW。

虚拟仪器(virtual instrumentation)是基于计算机的仪器。计算机和仪器的密切结合是目前仪器发展的一个重要方向。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的仪器。随着计算机功能的日益强大以及其体积的日趋缩小,这类仪器功能也越来越强大,目前已经出现含嵌入式系统的仪器。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。虚拟仪器主要是指这种方式。

1.2.1 虚拟仪器的构成

虚拟仪器是计算机化的仪器,由计算机、模块化功能硬件和应用软件三大部分组成。一般而言,虚拟仪器所用的计算机是通用的计算机,虚拟仪器根据其模块化功能硬件的不同而有多种构成方式。

(1) PC-DAQ 测试系统 是以数据采集卡、信号调理电路及计算机为仪器硬件平台组成的测试系统。

(2) GPIB 系统 是以 GPIB 标准总线仪器与计算机为硬件平台组成的测试系统。

(3) VXI 系统 是以 VXI 标准总线仪器与计算机为硬件平台组成的测试系统。

(4) 串口系统 是以 RS-232 标准串行总线仪器与计算机为硬件平台组成的测试系统。

(5) 现场总线系统 是以 FieldBus 标准总线仪器与计算机为硬件平台组成的测试系统。

无论上述哪种形式的虚拟仪器系统,都是通过应用软件将仪器的模块化功能硬件与各类计算机相连接,其中 PC-DAQ 测试系统是构成虚拟仪器 VI 的最常用最基本的方式,因为一般而言,这种类型的虚拟仪器成本比较低,它能充分利用计算机的计算能力。

目前为不同的应用目的和环境,已设计了多种性能和用途的数据采集板卡,从低速采集板卡到高速同步采集板卡,直至图像采集卡都有成熟的产品,为虚拟仪器系统的建立提供了物质基础。

1.2.2 虚拟仪器的特点

在通用硬件平台确定后,可由软件而不是硬件来决定仪器的功能。

仪器的功能是用户根据需要由应用软件来定义的,而不是事先由厂家定义好了的。因此用户不必购买多台不同功能的传统仪器,也不必购买昂贵的多种功能的传统仪器,因为在这种仪器中常常只有一种或几种功能是常用的,而其他功能都是不需要的。

为提高仪器性能或需要构造新的仪器功能时,可由用户自己改变软件来实现,而不必重新购买新的仪器。



虚拟仪器灵活、开放，因此可与计算机同步发展，可与网络及其他周边设备互连。

决定虚拟仪器具有上述传统仪器可能具备的特点的根本原因在于虚拟仪器的关键是软件，即仪器。

1.3 LabVIEW 2010 的系统配置

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 的系统配置。NI 公司为不同层次用户提供了 3 种系统配置。

(1) LabVIEW 基本版 是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 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 除了具备完整版的功能外，还具有专业程序员开发时所需要的全部工具。

对一般用户而言，购买 LabVIEW 完整版，根据实际需要选购专用的 LabVIEW 工具套件是最佳选择。

表 1-1 列出了三种版本的 LabVIEW 的配置。

表 1-1 LabVIEW 三种版本对照

项 目	完整版	专业版	开发者套件
用户界面开发	√	√	√
数据采集函数和向导	√	√	√
仪器控制函数和向导	√	√	√
报告生成和数据存储	√	√	√
调用外部代码	√	√	√
模块化和面向对象的开发	√	√	√
网络通信	√	√	√
随附的 NI LabVIEW SignalExpress	√	√	√
数学、分析和信号处理	√	√	√
事件驱动型编程	√	√	√
应用发布	—	√	√
软件工程工具	—	√	√
随附的生产力工具包(Productivity Toolkit)	—	—	√

1.4 LabVIEW 2010 的安装与运行

LabVIEW 可以安装在 Mac Os、Linux、Windows 2000/XP/Vista 等不同的操作系统或同一操作系统的不同版本上。不同的操作系统在安装 LabVIEW 2010 时对系统的配置要求也不同，用户在安装 LabVIEW 2010 前需对计算机系统的软硬件环境配置作一定的了解。

LabVIEW 的安装十分简单，只需运行安装光盘中的 setup 程序，按照屏幕提示，一步步选择必要的安装选项即可完成。整个系统安装时间取决于硬件平台和选择的安装选项。LabVIEW 所有文件约占 900MB 的硬盘空间。



为了控制 VXI, GPIB 和 DAQ 设备, 在 LabVIEW 2010 系统安装完成后, 如果需要还必须安装专门的仪器驱动和 VISA 库函数。一般不随 LabVIEW 2010 系统光盘, 例如 DAQ 卡的驱动程序, 对应于 LabVIEW 2010 的 DAQmx 为 NIDAQ921-1, 需从 NI 网站免费下载或购买 DAQ 采集卡时 NI 公司提供的数张驱动程序光盘, 需单独安装。

1.5 LabVIEW 2010 软件的启动方法及步骤

单击开始→程序→National Instruments LabVIEW 即可启动 LabVIEW, 启动后最终显示在屏幕上的启动界面如图 1-1 所示。图 1-1 是启动方式选择对话框。该对话框由左右两部分组成, 左侧包含新建和打开两个大的选项。在新建选项下可以新建 VI、新建项目和新建基于模板的 VI, 如果单击更多会出现新建类型目录树, 从中选中需创建的文件类型, 如图 1-2 所示。打开选项可以打开项目或 VI。

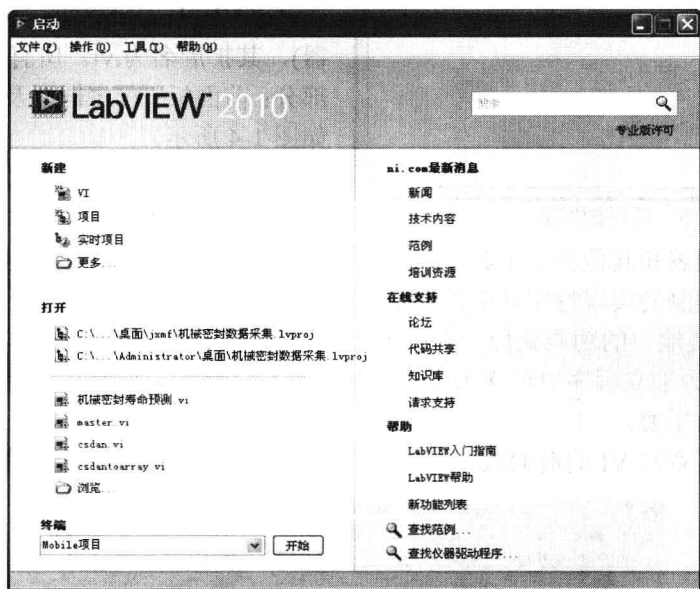


图 1-1 启动方式选择对话框



图 1-2 新建树型结构