



台湾永宏 PLC、四方变频器 : 15021230050

www.91HMI.com

台湾永宏 PLC、四方变频器 : 15021230050

从入门到精通，从技巧到实际应用，完全掌握LabVIEW

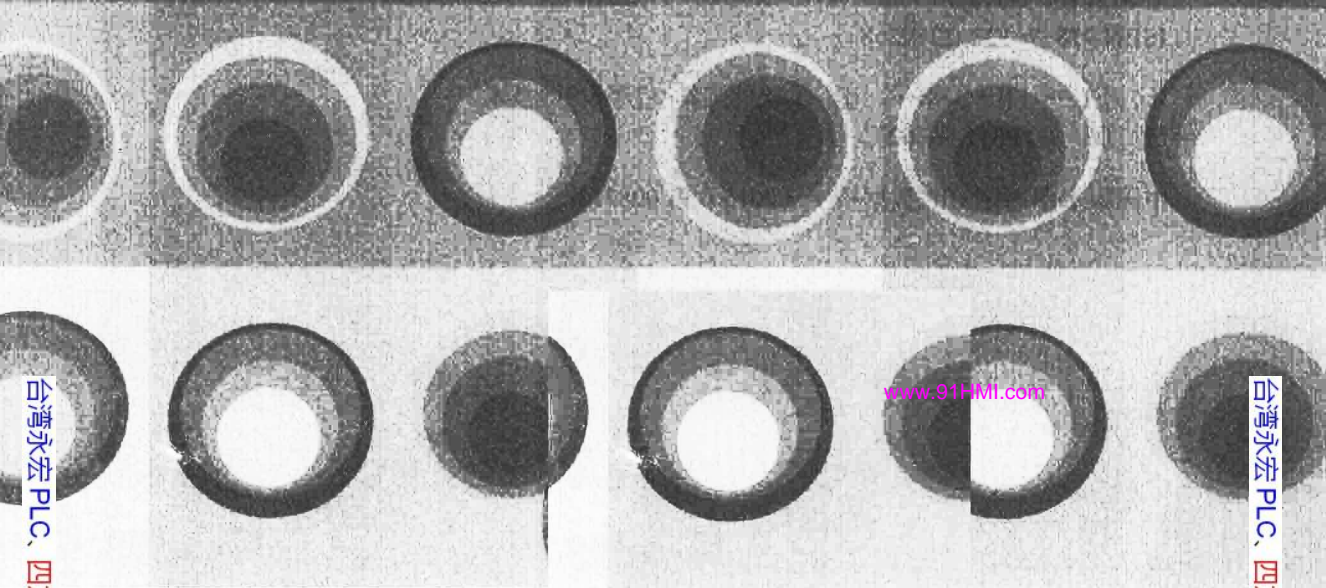
LabVIEW

2013

完全自学手册

李静 编著

 化学工业出版社



台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

LabVIEW

2013

完全自学手册

李静 编著

 化学工业出版社



图书在版编目 (CIP) 数据

LabVIEW 2013 完全自学手册/李静编著. —北京: 化学工业出版社, 2015.3

ISBN 978-7-122-22792-8

I. ①L… II. ①李… III. ①软件工具-程序设计-技术手册 IV. ①TP311.56-62

中国版本图书馆 CIP 数据核字 (2015) 第 008713 号

www.91HMI.com

台湾永宏 PLC、四方变频器 : 15021230050

www.PLCworld.cn

台湾永宏 PLC、四方变频器 : 15021230050

www.91HMI.com

www.PLCworld.cn

责任编辑: 宋 辉

装帧设计: 王晓宇

责任校对: 王素芹

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: 三河市延风印装厂

787mm×1092mm 1/16 印张 19½ 字数 514 千字 2015 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: 68.00 元

版权所有 违者必究



前言

LabVIEW编程技术如今已遍布各行各业，它以独特的图形化编程方式突破了传统编程方式的理念，使程序的开发与设计不再局限在专业人员的手中。只要你感兴趣，掌握LabVIEW同样可以编写出高质量的程序，且效率出众。

本书分为三部分：入门篇、提高篇和应用篇。

第1章至第7章为入门篇，主要涉及LabVIEW基本操作、基本数据类型及其相互转换、程序结构、数据运算函数、控件及其用法、数据表达与显示，使读者对LabVIEW有一个基本了解并掌握初级的编程技能。
www.91HMI.com

第8章至第18章为提高篇，主要涉及LabVIEW文件输入/输出、字符串操作、数学分析、信号处理、软件间通信、时间函数、应用程序控制、网络编程技术、总线技术与仪器控制、数据库访问及程序发布等，以利于读者对LabVIEW有更全面的理解，为解决专业性问题储备知识。

第19章和第20章为应用篇，通过两个实例来讲解如何运用已掌握的LabVIEW知识来解决实际问题。例如，该如何以实际问题为原型，通过提取、归纳来建立开发模型；如何在了解处理流程的基础上，将整体解决方案细分为相应的功能模块；以及如何通过控件、函数及节点等来实现设计目标。

本书由李静编著，感谢张阳、章佳荣、周天立、陈香凝、马惠来、李广鹏、马宏和杜强等对本书编写提供的帮助。此外，在成书过程中还得到了张铮博士的许多帮助及赵洁女士的大力支持，在此一并表示真诚的感谢。

由于时间所限，书中难免有疏漏和不足之处，敬请广大读者予以批评和指正。

李 静



第 1 部分 入门篇

第 1 章 初识 LabVIEW	1
1.1 LabVIEW 与虚拟仪器	1
1.2 LabVIEW 的安装与启动	2
1.3 VI 与子 VI	4
1.4 前面板与后面板	5
1.5 操作选板	6
1.6 综合实例：创建 LabVIEW 项目	9
第 2 章 LabVIEW 基本操作	11
2.1 创建 VI	11
2.2 编辑 VI	12
2.3 运行与调试 VI	15
2.4 子 VI 创建与调用	18
2.5 综合实例：制作弹出提示窗口	20
第 3 章 LabVIEW 基本数据类型及 其相互转换	22
3.1 基本数据类型	22
3.2 数据类型之间的转换	28
第 4 章 LabVIEW 程序结构	32
4.1 循环结构	32
4.2 顺序结构	35
4.3 条件结构	37
4.4 事件结构	38
4.5 定时结构	40
4.6 公式及 MATLAB 节点	41
4.7 反馈节点	43
4.8 变量	44
4.9 禁用结构	46
4.10 综合实例：数字平滑滤波	48
第 5 章 LabVIEW 数据运算函数	50
5.1 基本算术运算函数	50
5.2 布尔运算函数	57
5.3 关系运算函数和比较函数	60
5.4 其他函数	64
5.5 综合实例：逻辑芯片的功能 模拟	73
第 6 章 LabVIEW 控件及其使用方法	75
6.1 常用控件类型	75
6.2 特殊控件类型	80
6.3 常用控件属性设置	84

6.4 自定义控件	85
6.5 综合实例：虚拟仪表界面制作	86
第 7 章 LabVIEW 数据表达与显示	88
7.1 基本数据表达与显示	88
7.2 二维图形	88
7.3 三维图形	98
7.4 图形控件	108
7.5 其他图形展示	111
7.6 图片控件	115
7.7 综合实例：多曲线显示控制	121

第 2 部分 提高篇

第 8 章 LabVIEW 文件 I/O	123
8.1 常用文件操作函数	123
8.2 常用文件操作	126
8.3 特殊文件操作	130
8.4 其他文件操作	137
8.5 高级文件工具	143
8.6 综合实例：采集文件的 存储与读取	147
第 9 章 LabVIEW 中的字符串操作	149
9.1 常用的字符串常量	149
9.2 常用的字符串函数	149
9.3 “匹配模式字符串”和“匹配 正则表”函数	151
9.4 “格式化字符串”和“扫描字 符串”函数	152
9.5 数组与电子表格字符串	153
9.6 附加字符串函数	153
9.7 平化/还原字符串函数	156
9.8 创建文本	158
9.9 综合实例：电子表格中任意 数据的增删实例	159
第 10 章 LabVIEW 数学分析	161
10.1 基本数学分析	161
10.2 数理统计与最优化	169
10.3 曲线拟合与插值	174
10.4 其他操作	177
10.5 综合实例：基于最小二乘法的 温度传感器校准处理	180
第 11 章 LabVIEW 信号处理	182
11.1 信号发生	182



11.2	信号时域分析	186	第 16 章	LabVIEW 总线技术与 仪器控制	243
11.3	信号频域分析	190	16.1	LabVIEW 2013 支持的总线	243
11.4	信号调理	193	16.2	GPIB 总线	245
11.5	波形监测	196	16.3	VXI、PXI 和 LXI 总线	249
11.6	逐点分析	198	16.4	串行总线	256
11.7	综合实例：基于巴特沃斯滤波 的温度采集	200	16.5	VISA 的使用	261
第 12 章	LabVIEW 与其他软件通信	201	16.6	用 NI MAX 查找仪器	264
12.1	ActiveX 技术	201	16.7	综合实例：用 GPIB 总线 控制可编程滤波器	267
12.2	.NET 技术	203	第 17 章	LabVIEW 数据库的访问	269
12.3	LabVIEW 与 Microsoft 的通信	205	17.1	数据库连接工具包及其使用	269
12.4	动态链接库函数及调用	212	17.2	综合实例：采集数据的数据 库管理	276
12.5	执行.exe 命令	214	第 18 章	LabVIEW 程序发布	278
12.6	LabVIEW 与 MATLAB 混合编程	215	18.1	创建项目	278
12.7	综合实例：利用报表工具 制作报表	215	18.2	程序生成规范	280
第 13 章	LabVIEW 时间相关操作	217	18.3	发布前的准备	280
13.1	已用时间	217	18.4	源代码发布	282
13.2	时间延迟	218	18.5	创建应用程序	283
13.3	当前时间	219	18.6	创建 Zip 文件	283
13.4	时间格式转换函数	220	18.7	创建安装程序	284
第 14 章	LabVIEW 应用程序控制	223	第 3 部分 应用篇		
14.1	属性节点与调用节点	223	第 19 章	多通道温度检测记录系统	286
14.2	信息获取与提示	226	19.1	虚拟硬件环境	286
14.3	程序错误处理与控制	230	19.2	程序框架设计	287
14.4	综合实例：文件的任意 位置存储	231	19.3	功能模块设计	288
第 15 章	LabVIEW 网络编程技术	233	第 20 章	温湿度检测管理系统	294
15.1	TCP/IP 网络通信	233	20.1	硬件设计	294
15.2	DataSocket 通信	238	20.2	软件设计	299
15.3	远程面板	240	参考文献		306



第 1 部分

入门篇

第 1 章

初识LabVIEW

www.91HMI.com

LabVIEW 从诞生之初就致力于全面简化虚拟仪器系统的构建,使设计者能够以简单而有效的方式实现各种测控开发。其主要特点是采用图形化编程模式、使用灵活的数据表达方式、采用开放式开发环境并提供强大的分析处理能力。

【本章导读】

- ◆ LabVIEW 与虚拟仪器
- ◆ LabVIEW 的安装与启动
- ◆ VI 与子 VI
- ◆ 前面板与后面板
- ◆ 操作选板等

1.1 LabVIEW 与虚拟仪器

虚拟仪器是计算机技术与仪器仪表技术结合的产物,它通过应用软件将计算机与各种硬件设备结合在一起,使用时用计算机的操作界面就可对测试设备进行管控,从而完成各种物理信号的采集、分析、处理、显示及数据存储。

LabVIEW 是实验室虚拟仪器工程平台的英文缩写,由美国 NI 公司所开发。与传统的文本编程不同,LabVIEW 采用图形化编程,以程序框图来组织程序设计,由程序框图中的节点来传递数据的流动。

LabVIEW 提供了大量的与传统仪器(如表盘、按钮、示波器、万用表等)相类似的控件,便于高效地创建用户界面。此外,它有一个可以胜任各种编程任务的强大函数库,主要涉及数据采集、GPIB、串口控制、数据分析、数据显示及数据存储等。



LabVIEW 与虚拟仪器有着密切的关系，这主要涉及以下几个方面。

① 虚拟仪器是基于计算机的仪器，两者的结合代表了仪器发展的重要方向。其呈现方式有两种，一种是将计算机装入仪器，即所谓智能化仪器。另一种则是将仪器装入计算机。以通用计算机的硬件及操作系统为依托，实现各种仪器功能，虚拟仪器主要是采用这种方式。

② 采用常规微机构建虚拟仪器或计算机测试系统时，需要用到一些插卡式的仪器来扩展其性能。由于这些卡式仪器没有自身的面板，因此需要以虚拟面板的方式来呈现，使用 LabVIEW 可以快速而有效地完成此类开发。

③ 虚拟仪器的研究涉及各种标准仪器的互连以及同计算机的连接，目前使用较多的是 IEEE488 或 GPIB 协议，而未来的仪器则是向网络化发展。

④ 在硬件相同的条件下，软件的变化能够充分发挥计算机的强大潜能和数据处理能力优势，为构造各种性能强大的虚拟仪器提供了有利的支持。

LabVIEW 的应用领域十分广泛，除了测试测量领域，它还凭借其突出的特点在其他领域发挥着突出的作用。

在教学方面：由于 LabVIEW 具有极强的图形表现力，因而用它开发的教学软件，其界面效果非常美观极富吸引力。此外，图形及动画的讲述力也更容易被学习者接受和理解。

在仿真方面：LabVIEW 以它丰富的数学运算函数，为模拟、仿真及原型设计等工作提供了有力的支持。在设计前期，可利用 LabVIEW 在计算机上搭建出仿真模型，以验证设计的合理性。此外，使用 LabVIEW 设计的模拟软件还可进行虚拟实验，并可达到降低实验成本增强安全性的效果。

在控制方面：由于控制与测试是一对高度相关的领域，由测试测量领域做大的 NI 公司，其不但开发出了 LabVIEW 这样出色的开发平台，它在控制领域也推出了大量的为测控服务的硬件产品，LabVIEW 与它们可以便捷地构成各类控制系统。

1.2 LabVIEW 的安装与启动

本书以中文版 LabVIEW2013 为编程环境，其安装软件是一个名为 2013LV-WinChn 的压缩包，如图 1-1 所示。下载后，需进行解压缩才能安装，在解压后其如图 1-2 所示。

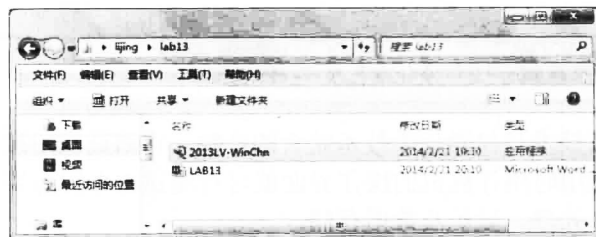


图 1-1 获取安装压缩包

在解压缩后的文件中，双击“setup.exe”即可执行安装，其初始界面如图 1-3 所示。通过单击“下一步”可展开安装。

在安装过程中，一旦出现图 1-4 的界面，则要决定是安装试用版还是正式版。若安装试用版，其无需序列号，但试用期比较短，到期后便无法继续使用。



图 1-2 解压安装软件

www.91HMI.com



图 1-3 安装初始界面

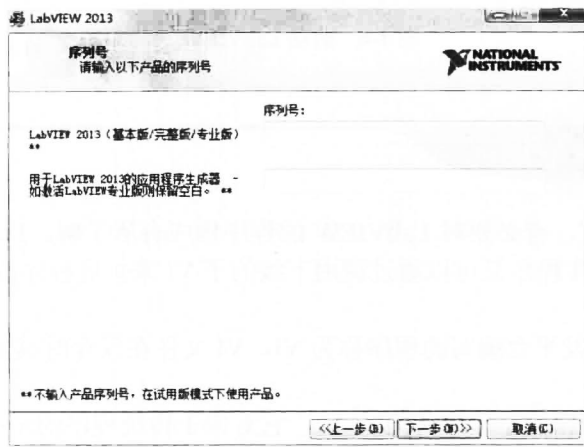


图 1-4 序列号限定


台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

www.91HMI.com

www.PLCworld.cn



在正常情况下，会在结束前出现图 1-5 的界面，单击“下一步”就可完成安装。此时，在开始菜单可见到  图标，单击它便可启动 LabVIEW。

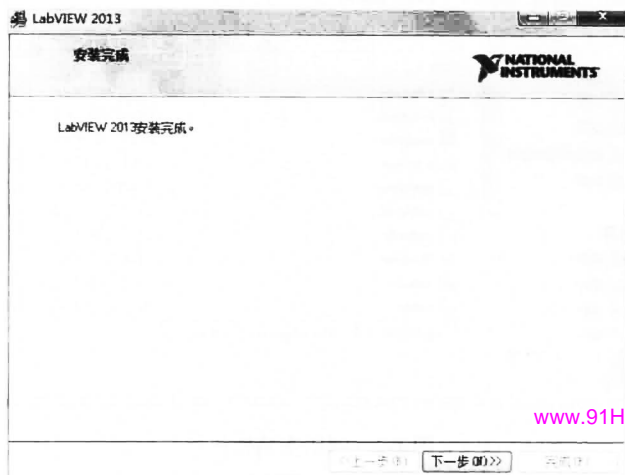


图 1-5 完成安装

启动 LabVIEW 后，其界面如图 1-6 所示。

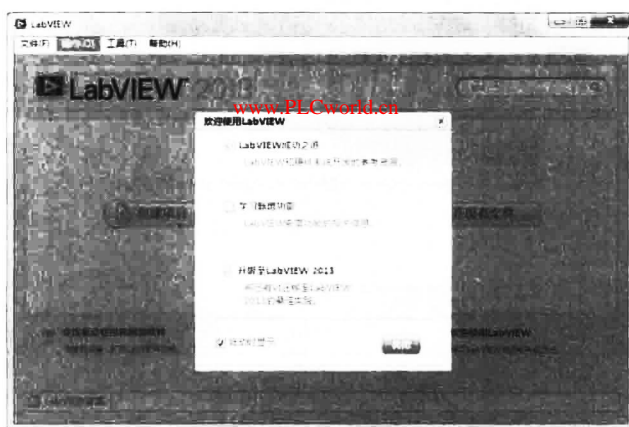


图 1-6 启动 LabVIEW

1.3 VI 与子 VI

在正式开始编程前，有必要对 LabVIEW 的程序构成有所了解。LabVIEW 是以 VI 文件为程序单位的，一个 VI 程序又可以通过调用下级的子 VI 来扩展程序规模。

(1) VI

使用 LabVIEW 开发平台编写的程序称为 VI，VI 文件在保存时其文件扩展名为 .VI。

(2) 子 VI

一个 LabVIEW 程序的子程序称为子 VI，它对应于传统程序的子程序或子函数。子 VI 的连接端口是其被调用时的输入/输出接口，用以传递信息参数。



(3) 图标与连线板

将 VI 作为子 VI 使用时, 要为其创建相应的图标和连线板。图标和连线板相当于文本程序中的函数原型, 它位于前面板或后面板窗口的右上角, 如图 1-7 所示。



图 1-7 图标与连线板

1.4 前面板与后面板

(1) 前面板

前面板是进行虚拟仪器设计的平台, 其形式如图 1-8 所示。在前面板上所使用的对象为输入和输出两类控件, 其中的输入控件是指开关、旋钮、转盘等输入装置, 而输出控件是指指示灯、图形等输出装置。

输入控件是通过模拟仪器的输入来向 VI 的程序框图提供数据, 而输出控件则是接收程序框图输出的数据并模拟仪器的形式来显示。

(2) 后面板

后面板是操纵和控制前面板上输入与输出功能编程平台, 与传统文本开发平台不同的是它采用图形化的编程, 即使用程序框图来编程。在运行时, 程序框图是不可见的, 显示的内容只有前面板的控件。后面板的形式如图 1-9 所示。

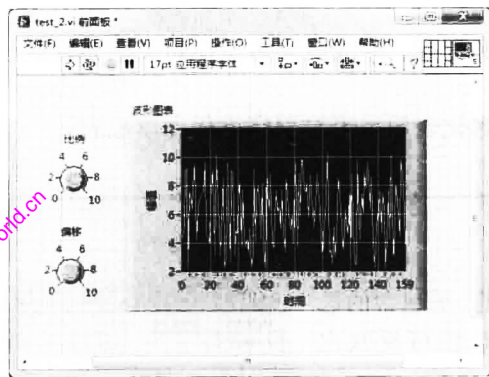


图 1-8 前面板

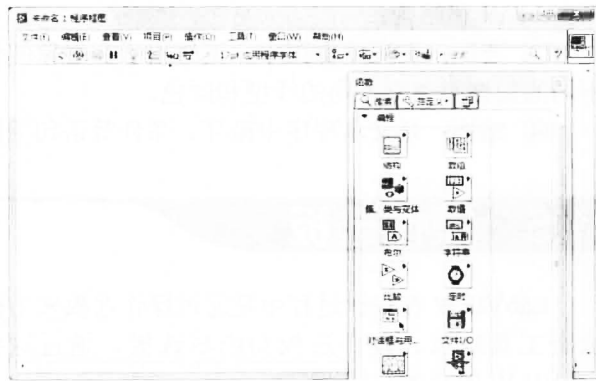


图 1-9 后面板

提示

在编写 LabVIEW 程序时, 经常需要在前后面板间进行切换, 切换最便捷的方法是使用快捷键 Ctrl+E。

(3) 程序框图

程序框图是图形化源代码的集合, 在执行时是按信号流过图标的顺序来接受数据流的控制, 故该编程方式被称为数据流编程。



程序框图由端点、节点、连线和结构这 4 类元素组成，其形式如图 1-10 所示。



图 1-10 程序框图

① 端点：当在前面板上添加某一控件时，在后面板上也会出现对应的图标，该图标就是此控件的端点，端点可以连线的方式来传递数据。输入控件接线端的箭头在右边，表示数据只能流出。输出控件接线端的箭头在左边，表示数据只能流入。正确区分端点的性质，利于提高连线的准确性。

② 节点：它类似于文本程序中的语句、运算符、函数和子程序，是程序框图上的对象，并带有输入输出端。当 VI 运行时，它起运算处理的作用。节点主要有 4 类，即函数、子 VI、Express VI 和结构。

③ 连线：在程序框图中，连线用于在对象间传递数据。连线能够传递不同类型的数据，不同的数据类型有不同的线型和颜色。

④ 结构：是文本程序中循环、条件等语句的图形化表示，可对程序流程加以控制。

1.5 操作选板

LabVIEW 在设计过程中是通过操作选板来工作的，所包含的选板有工具选板、控件选板和函数选板，通过这些选板可反映出 LabVIEW 的功能与特性。

1.5.1 控件选板

控件选板在前面板显示，它是进行界面设计的模块工具，其涵盖了前面板设计的全部对象。控件选板有多种模块功能，包括新式、银色、系统、经典、选择控件、Express、.NET 与 ActiveX 等，而每个模块又含有许多子功能模板。

显示控件选板时，可选取“查看”菜单中的“控件选板”选项，该选板如图 1-11 所示。

在控件选板中，各图标的功能如表 1-1 所示。

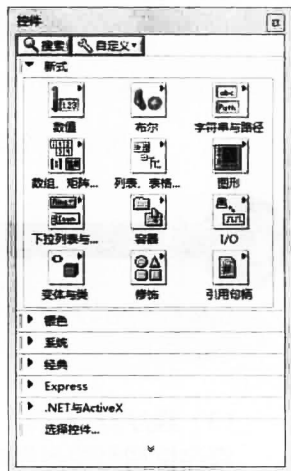


图 1-11 控件选板



表 1-1 控件选板图标功能表

序号	图标	名称	功能
1		数值量	数值的控制与显示, 包含数字式、指针式显示表盘及各种输入框
2		布尔量	逻辑值的控制与显示, 包含各种开关、按钮及指示灯等
3		字符串与路径	字符串与路径的控制与显示
4		数组、矩阵和簇	数组和簇的控制与显示
5		列表与表格	列表与表格的控制与显示
6		图形显示	显示数据结果的趋势图和曲线图 www.91HMI.com
7		下拉列表与枚举	环与枚举的控制与显示
8		容器	容器设计功能
9		输入/输出功能	输入/输出功能, 用于操作 OLE 和 ActiveX 等
10		变体与类	面向对象设计功能 www.PLCworld.cn
11		装饰	用于给前面板进行修饰的各种图形对象
12		引用句柄	参考数

1.5.2 函数选板

函数选板在后面板显示, 它是程序框图设计的模块工具, 其涵盖了创建框图程序的常用 VI 和函数对象。控件选板有多种模块功能, 包括编程、测量 I/O、仪器 I/O、数学、信号处理、数据通信、互连接口、Express、附加工具包、选择 VI 等, 而每个模块又含有许多子模板。

显示函数选板时, 可选取“查看”菜单中的“函数选板”选项, 该选板如图 1-12 所示。在函数选板中, 各图标的功能如表 1-2 所示。

表 1-2 函数选板图标功能表

序号	图标	名称	功能
1		结构	包含程序控制结构命令, 如循环控制等, 以及全局变量和局部变量
2		数值运算	包含各种常用的数值运算、数制转换、三角函数、对数复数运算, 以及数值常数等
3		布尔运算	包含各种逻辑运算符及布尔常数
4		字符串运算	包含各种字符串操作函数、数值与字符串转换函数及字符串常数等



续表

序号	图标	名称	功能
5		数组	包含数组运算函数、数组转换函数以及常数数组等
6		簇	包含簇的处理函数以及群常数等
7		比较	包含各种比较运算函数, 如等于、大于、小于等
8		定时	包含各种定时函数及时间转换函数
9		文件输入/输出	包含处理文件输入/输出的程序和函数
10		波形	包含各种波形处理工具
11		应用程序控制	包含动态调用 VI、标准可执行程序的功能模块
12		图形与声音	包含 3D、声音播放等功能模块
13		同步	包含各种同步处理函数
14		报表生成	包含各种报表生成模块
15		对话框与用户界面	包含各种对话框窗口处理函数

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

www.PLCworld.com

www.PLCworld.cn

提示

只有使用前面板时才能对控件选板进行调用, 同样只有使用后面板时才能对函数选板进行使用。

1.5.3 工具选板

工具选板既可在前面板显示, 也可在后面板显示, 它是创建、修改和调试 VI 程序的工具。在使用工具选板时, 依据工具的不同, 光标会呈现出不同的形状以便于操作。

显示工具选板时, 可选取“查看”菜单中的“工具选板”选项, 该选板如图 1-13 所示。

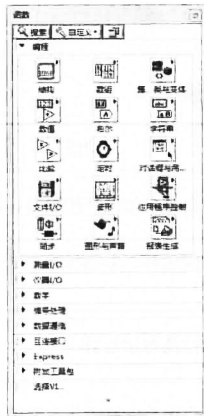


图 1-12 函数选板

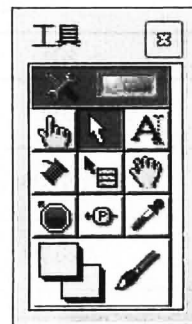


图 1-13 工具选板



在工具选板中，各图标的功能如表 1-3 所示。

表 1-3 工具选板图标功能表

序号	图标	名称	功能
1		操作值	操作前面板的控制与显示
2		定位/调整大小/选择	选择、移动或改变对象的大小
3		编辑文本	输入标签文本或创建自由标签
4		进行连线	在程序框图上连接对象
5		对象快捷菜单	单击左键可弹出对象的快捷菜单
6		滚动窗口	在窗口中漫游 www.91HMI.com
7		设置/清除断点	在 VI 程序框图的对象上设置断点
8		数据探针	在程序框图的数据流线上设置探针
9		获取颜色	提取颜色
10		设置颜色	给对象定义前、背景颜色

1.6 综合实例：创建 LabVIEW 项目

使用项目可对程序进行有效管理，创建项目的操作如下：

- ① 打开 LabVIEW，单击“创建项目”，将出现创建项目界面，如图 1-14 所示。
- ② 选取“项目模板”，单击“完成”，则弹出图 1-15 所示的项目浏览界面。



图 1-14 创建项目

- ③ 单击“保存全部”图标按钮，则弹出图 1-16 所示命名项目对话框。指定文件名（即



项目名), 然后单击“确定”, 则完成项目的创建及命名保存。在创建项目后, 若新建 VI 程序 (如 test1.vi 和 test2.vi), 则被保存到项目中, 如图 1-17 所示。

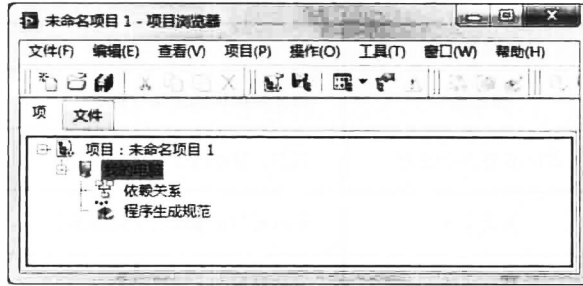


图 1-15 项目浏览界面

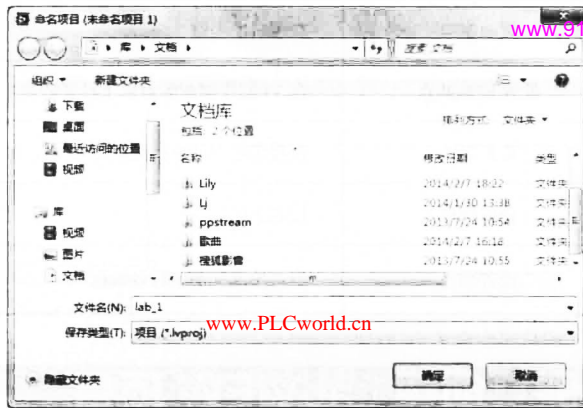


图 1-16 命名项目对话框

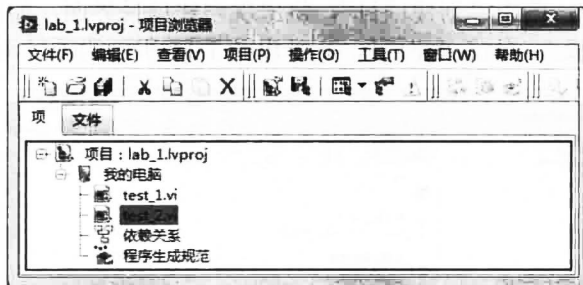


图 1-17 VI 程序纳入项目管理



LabVIEW 不仅用于虚拟仪器开发, 它还是仿真软件开发的利器, 这一切都建立在它强大



第 2 章

LabVIEW基本操作

要进行 LabVIEW 编程就离不开基本操作，即创建、编辑和调试 VI。通过对基本操作的学习，则可展开 LabVIEW 的设计历程，并可体会到图形化程序设计的快乐。

【本章导读】

- ◆ 创建 VI
- ◆ 编辑 VI
- ◆ 运行与调试 VI
- ◆ 子 VI 的创建与调用等

与其他编程语言不同，LabVIEW 编程是采用图形化程序设计，它具有编程简单、设计直观、开发效率高的特点。使用 LabVIEW 设计的程序被称为 VI，它由三部分构成，即前面板、后面板和图标/连接板。

2.1 创建 VI

在 LabVIEW 中，VI 是程序设计的舞台。创建 VI 需包括新建一个 VI 并对其进行命名保存。

新建 VI 的方法是：选取“文件”菜单中的“新建 VI”选项或按快捷键“Ctrl+N”，其菜单如图 2-1 所示。

命名并保存 VI 的方法是：选取“文件”菜单中的“保存”选项或按快捷键“Ctrl+S”，其菜单如图 2-2 所示。再由图 2-3 所示“命名 VI”对话框，输入文件名并点击“确定”即可。

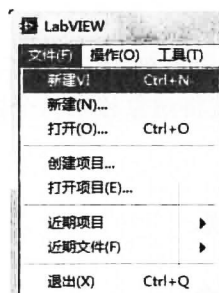


图 2-1 创建 VI

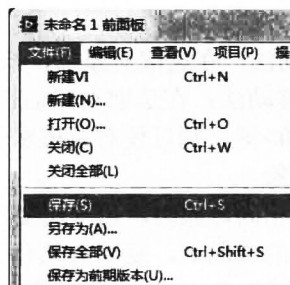


图 2-2 保存 VI



图 2-3 命名 VI

www.91HMI.com

台湾永宏 PLC、四方变频器 : 15021230050

2.2 编辑 VI

2.2.1 对象的基本操作

(1) 选择对象

要操作对象必须先选择对象，使用图标选择对象的方法有两种。

累积选择法：按下<Shift>键，通过移动鼠标来逐个拾取选择对象。选择后的对象周围会出现流动的虚线选取框，如图 2-4 所示。

拖拽选择法：用鼠标左键拖拽出一个选择框，则会选中选择框内的对象，图 2-5 所示。若配合<Shift>键的使用，则可产生多个选择框。

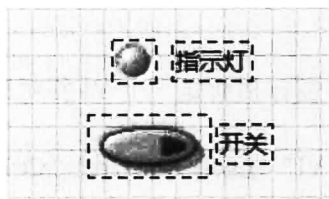


图 2-4 累积选择法

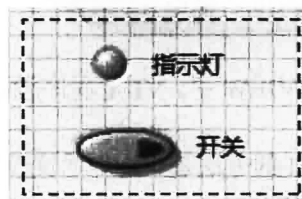


图 2-5 拖拽选择法

(2) 移动对象

移动对象的方法有两种：

- ① 鼠标移动法：在选取对象后，按住鼠标左键进行移动，到指定位置后再松开鼠标。
- ② 方向键移动法：在选取对象后，每按一次方向键就将对象移动一个像素。在移动对象时若按住<Shift>键，则可使对象按水平或垂直方向移动。

(3) 复制对象

在使用多个相同对象时，可采用复制的方法来快速实现。

在已选对象的情况下，复制对象有三种方法：

- ① 使用快捷键<Ctrl+C>复制对象，用<Ctrl+V>粘贴对象。
- ② 使用编辑菜单中的“复制”和“粘贴”选项。

台湾永宏 PLC、四方变频器 : 15021230050

www.PLCworld.cn



③ 按下<Ctrl>键，在选取对象后按下鼠标左键进行拖拽，直到指定位置再释放鼠标。

(4) 删除对象

当出现不使用的对象后，可采用删除的方法予以去除。

在已选对象的情况下，删除对象有三种方法：

- ① 使用快捷键<Ctrl+X>。
- ② 使用编辑菜单中的“删除”选项。
- ③ 使用<Delete>键或<Backspace>键。

若在操作中出现误删现象，可用编辑菜单中的“恢复”选项来找回。

(5) 对象值操作

对于控件的操作值，可在运行前和运行中对其进行修改。

根据控件的不同，修改操作值的方法如下：

- ① 对于带文本框的控件，用操作值工具点击该控件并键入相应数值。
 - ② 对于含“增、减量”标识的控件，用操作值工具点击增减量符号来增减数值。
 - ③ 对于旋钮和滑动条类控件，用操作值工具点击或拖动指针或滑动条来修改数值。
- 修改对象值的方法如图 2-6 所示。

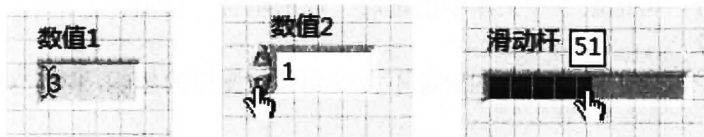



图 2-6 修改对象值

2.2.2 美化布局

排列与布局，其创建的方法有两种，一种是重新创建，另一种是利用已有 VI 来生成。

(1) 调整对象大小

前面板的控件是一种可视化的对象，其大小是可以按照设计的需要予以调整。调整大小的方法如下。

① 使用图标，当鼠标移动到被调整对象时，对象的边缘就会出现尺寸控制点，将鼠标移到控制点上，然后按下鼠标左键进行拖动，当调节对象的虚线框达到需要后松开鼠标即可。调整对象大小的过程如图 2-7 所示。

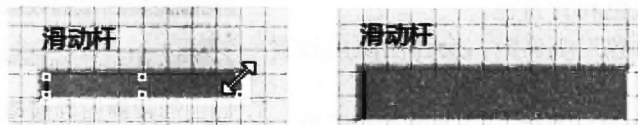




图 2-7 调整对象大小

② 若选取多个对象，再用工具栏中的“调整对象大小”图标，在出现的子选项中选择一种，即可调整它们的大小。


(2) 对齐对象

为了界面的美观，可使用相应的工具使多个对象整齐划一的排列，其操作方法如下。选取多个调整对象，然后在工具栏选取“对齐对象”图标，在出现的子选项中，按照需要予以选取即可。



(3) 分布对象


对象的分布处理可以使对象的分布更均匀，其操作方法如下：

选取多个调整对象，然后在工具栏选取“分布对象”图标，在出现的子选项中，按照需要予以选取即可。

(4) 重新排序

当前面板有多个对象出现重叠以形成特殊显示效果时，可使用重新排序工具来排列对象的前后次序。


重新排序的操作方法是：

选取多个调整对象，然后在工具栏选取“重新排序”图标，在出现的子选项中，按照需要予以选取即可。

重新排序工具有 4 个子选项，分别是向前移动、向后移动、移至前面和移至后面。

(5) 组合对象

使用对象组合功能可使操作对象成一个整体来同步操作，**组合对象的操作**方法如下：

选取多个操作对象，然后在工具栏选取“重新排序”图标，在下拉菜单中选取“组”选项即可。

若想取消组合，可选取下拉菜单中的选取“取消组合”选项。

(6) 锁定对象

在操作过程中，处于锁定状态的对象不会受到任何操作的影响，锁定对象的操作方法如下：

选取多个操作对象，然后在工具栏选取“重新排序”图标，在下拉菜单中选取“锁定”选项即可。

若想取消锁定，可选取下拉菜单中的选取“解锁”选项。

2.2.3 修饰对象

修饰对象的操作主要包括修改颜色与修改字体等。



(1) 修改颜色

使用颜色修改功能，可以改变大多数前面板对象、前面板窗格和程序框图工作区的颜色。但不能改变系统控件的颜色，因为这些对象的颜色是与系统的颜色设置相一致的。

修改颜色的操作方法如下：


在工具选板上选取“设置颜色”工具，通过点击对象可以设置前景色和背景色。

当选取“设置颜色”工具时，会弹出颜色设置面板，如图 2-8 所示。

在颜色设置面板中，图标用于设置透明度，图标用于自行定义颜色。

(2) 修改文本属性

对于所有的文本对象，均可进行字体颜色、大小和字体的修改，具体操作方法如下：

① 选取文本对象，点击文本设置图标，则弹出菜单中选取“字体对话框”。

② 在“选项字体”对话框，选取“颜色”可改变文本的颜色。

③ 在“选项字体”对话框，选取“大小”可改变文本的大小。



图 2-8 颜色设置面板



④ 在“选项字体”对话框，选取“字体”可改变文本的字体。

2.2.4 VI 编辑示例

编辑 VI 离不开前面板和后面板的联合使用，下面以指示灯控制为示例来介绍 VI 编辑的基本操作。

(1) 编辑 VI 前面板

① 新建一个 VI，并将该 VI 命名为 test.vi 加以保存。

② 为了用开关来控制指示灯，需在前面板放置一个圆形指示灯和一个开关按钮。具体操作是：选择“控件→新式→布尔→圆形指示灯”，然后在前面板的指定单击以放置圆形指示灯。再选择“控件→新式→布尔→开关按钮”，向前面板放置一个开关按钮。此时，前面板如图 2-9 所示。

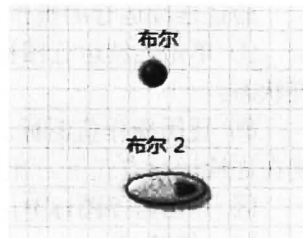



图 2-9 前面板效果

(2) 编辑 VI 程序框图

为了实现指示灯的控制，需在后面板对程序框图进行编辑。

按快捷键“Ctrl+E”可显示出后面板的程序框图，其内容如图 2-10 所示。用进行连线图标  将指示灯和按钮的端口连接起来，这样按钮的控制信号就可以控制指示灯的亮灭，连接后程序框图如图 2-11 所示。

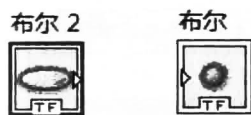


图 2-10 连线前程序框图

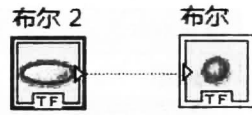




图 2-11 连线后程序框图


(3) 编辑工具使用

使用编辑工具能对已有的设计成果进行修改，为获得图 2-12 的效果可采用如下操作：

① 选取编辑文本图标 ，点击指示灯的标签“布尔”将其改为“指示灯”。

② 再将按钮的标签“布尔 2”改为“开关”。

③ 使用选择定位图标 ，将两个控件的标签调整到所需的位置。

④ 选取设置颜色图标 ，将前景色设为红色，然后点击圆形指示灯，这样指示灯的关闭状态就被设为红色。

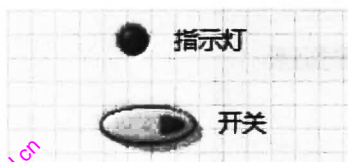


图 2-12 编辑后的效果


2.3 运行与调试 VI

在完成 VI 设计后，验证设计正确性的最好方法就是运行 VI，通过测试来发现设计中的问题。


(1) 运行 VI

运行的方式有两种，一种是单次运行，另一种是连续运行。

验证单次运行的操作方法如下：

① 单击运行图标 ，会发现界面只是快速地一闪，并未发生任何变化。这说明指示灯 VI 被执行了一次，之所以未出现变化是由于开关处于断开状态。





② 选取操作值图标, 并点击开关, 会发现开关的状态出现改变, 说明开关被闭合。


③ 再次使用运行图标, 会发现指示灯被点亮, 说明开关已对指示灯起到了控制作用, 其效果如图 2-13 所示。

经实践你会发现, 以单次运行的方式来验证开关作用是很不方便的, 此时可改用连续运行方式。

验证连续运行的操作方法如下:


① 单击连续运行图标, 使 VI 处于不停地运行过程中。此时, 终止运行图标呈点亮状态。

② 使用操作值图标, 不断变换开关的状态, 可观察到指示灯也同步地跟着变化, 这说明在连续运行方式下, 开关的控制信号可以及时的传送给指示灯, 起着同步的作用。

③ 点击暂停图标, 可暂停 VI 运行。若点击终止运行图标, 则可退出 VI 运行状态。

(2) 调试 VI

调试是查找错误、修改和优化程序的有效手段, LabVIEW 为设计者提供了多种 VI 调节工具, 如错误列表、单步执行、断点及探针等。

① 错误列表 当程序框图中出现错误时, 单次运行图标会变为, 单击该图标会弹出“错误列表”窗口以供修改, 该窗口如图 2-14 所示。

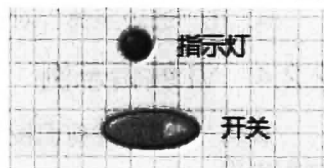


图 2-13 指示灯运行效果

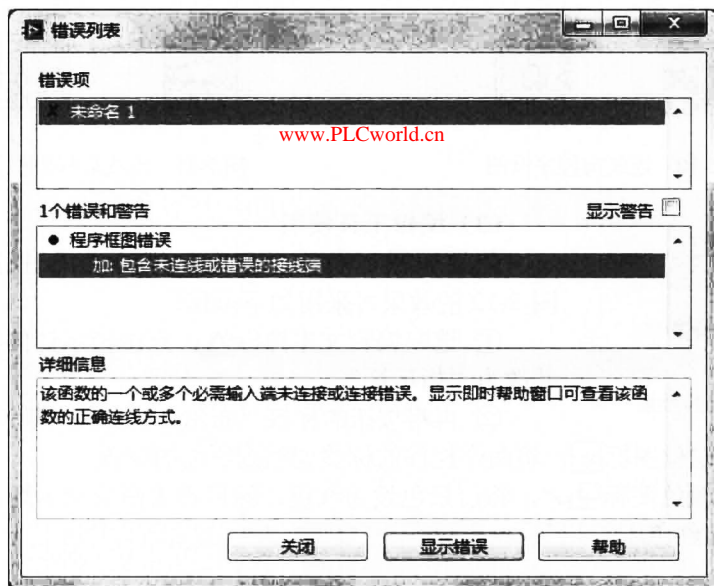




图 2-14 错误列表窗口

② 断点 使用工具选板的断点图标, 可在程序框图的 VI、节点或连线上放置断点, 此时会在连线的中央出现一个小红点, 当程序运行到断点时将暂停执行, 执行效果如图 2-15 所示。

再次单击设置的断点, 可取消该断点的设置。

③ 探针 使用工具选板的探针图标, 可检查 VI 运行时流过连线的的数据。使用探针工具在数据连线查看处单击, 其设置效果如图 2-16 所示。在程序运行时, 若有数据流过该探针位置, 则会弹出探针监视窗口并自动显示出流过探针的数据, 如图 2-17 所示。

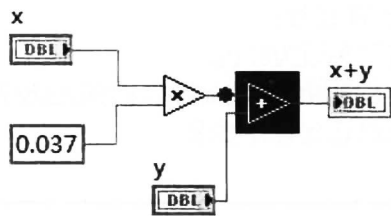


图 2-15 断点执行效果

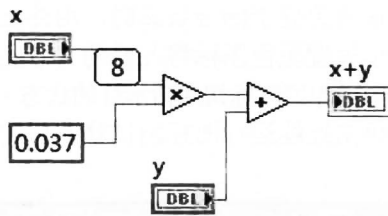


图 2-16 设置探针

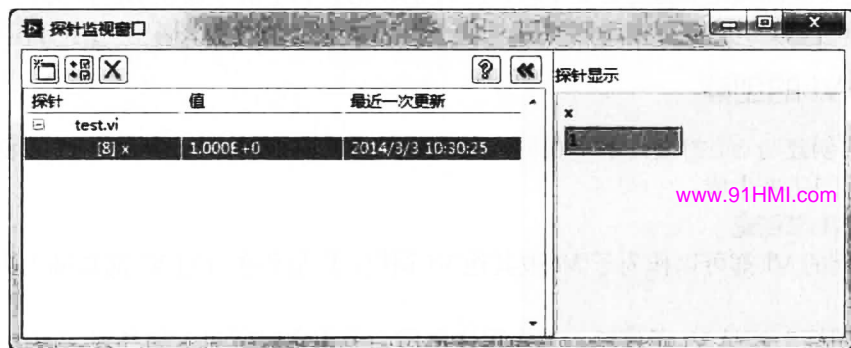



图 2-17 探针监视窗口

④ 单步执行 使用工具栏中的单步执行按钮,可查看 VI 运行时程序框图上的执行步骤,


www.91hmi.com


www.PLCworld.cn

单步步入按钮,表示单步进入循环或子 VI, 点击一次该按钮, 则程序执行一步;

单步步过按钮,表示单步执行整个循环或子 VI,

点击一次该按钮, 则程序执行一步;

单步步出按钮,表示单步进入某循环或子 VI, 点击该按钮则执行完循环或子 VI 并跳出。

⑤ 高亮显示 使用工具栏中的高亮运行按钮,可进入高亮执行状态。在执行过程中, 通过移动的圆点来指示连线上的数据由一个节点移动到另一个节点的全过程。

高亮显示与单步执行相配合的执行效果如图 2-18 所示。

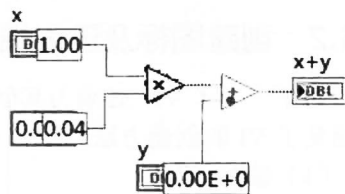


图 2-18 高亮显示运行

提示

错误列表只能指示出连线类的错误, 如缺少连线或出现断线和漏线等。对于编程中出现的逻辑性错误它是无能为力的, 只能通过运行结果和数据探针来分析。

(3) 调整 VI

通过运行 VI 可以发现设计中的一些不足, 对发现的问题需要边调试边修改的方法来解决。

对指示灯控制 VI, 经测试发现它并非是设计时所设想的: 在开关闭合时指示灯呈红色点亮状态, 在开关断开时指示灯呈灰色熄灭状态。

调试并修改 VI 的方法如下:



- ① 在开关处于闭合状态时，用终止运行图标停止 VI 运行；
- ② 用设置颜色图标将此时的（点亮状态）指示灯颜色设为红色；
- ③ 用操作值图标切换指示灯的状态（熄灭状态），再用设置颜色图标将指示灯颜色设为灰色；
- ④ 单击连续运行图标测试修改后的效果，经验证已达到设计要求。

2.4 子 VI 创建与调用

子 VI 是实现模块化编程的必要手段，子 VI 的使用有利于清晰设计结构并实现开发成果的重复使用。但子 VI 不能被直接递归调用，即一个 VI 不能在程序框图中调用本身作为子 VI。

2.4.1 子 VI 的创建

子 VI 的创建与 VI 的设计基本相同，其创建的方法有两种，一种是由 VI 来创建，另一种是利用已有 VI 来生成。

(1) 由 VI 来创建

其实任何的 VI 都可以作为子 VI 被其他 VI 调用，只要要在普通 VI 的基础上再定义连接端口和图标。

所谓“图标”就是 VI 子程序，当在程序框图中调用它时所显示的外观。而“连接端口”就是子 VI 与调用它的 VI 进行数据交换的端口。

(2) 用已有 VI 改造

使用已有的 VI 也可改造出子 VI，其方法是用选取工具在已有的 VI 中框定有用的部分程序框图，然后在“编辑”菜单中执行“创建子 VI”命令，这样所选的部分就变成系统默认的程序图标，进而实现子 VI 的创建。

2.4.2 创建图标及连线端口

对于一个子 VI，必须为其创建图标和连接端口。下面以求取半径为 R 的圆面积 S 为例来介绍其子 VI 的创建方法。

(1) 编写子 VI

根据圆的面积公式： $S = \pi R^2$ ，编写子 VI 的操作如下：

- ① 新建一个 VI，并命名为 S.VI 加以保存。
- ② 在前面板放置一个数值输入控件和一个数值显示控件，然后将它们的标签分别改为“半径：”和“面积：”，其界面如图 2-19 所示。
- ③ 在后面板，选取“函数”选板→“编程→数值”选项，从而放置 2 个乘运算符。
- ④ 选取“函数”选板→“编程→数值→数学与科学常量”选项，放置常数 π 。
- ⑤ 使用连线工具进行连线，完成后的程序框图如图 2-20 所示。

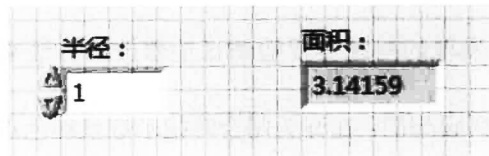


图 2-19 圆面积计算界面

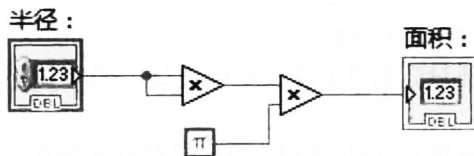



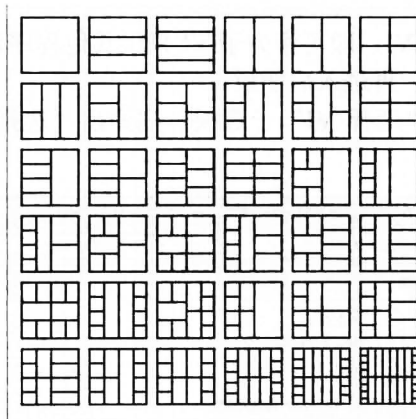
图 2-20 计算圆面积程序框图



(2) 创建连线端口


在创建好 VI 后，需要为其创建连线端口，具体的操作步骤是：


① 切换到前面板，使用鼠标右键点击图标，由快捷菜单中选取“模式”选项，则弹出连线端口样式表，如图 2-21 所示。



www.91HMI.com

图 2-21 连线端口样式表


② 由于该 VI 只有两个端口，因此选取 2 端口样式，以对应前面板的 2 个控件。

③ 用连线工具单击图标左侧的端口，再单击“半径:”控件，则端口变为，表示已定义了一个数据端口。

④ 用同样方法定义好右侧端口，此时端口变为，表示 2 个端口都已完成定义。

(3) 创建图标

为了更好地辨识 VI，需要为其创建一个图标，其方法是：

① 使用鼠标右键点击图标，由快捷菜单中选取“编辑图标...”选项，则弹出“图标编辑器”界面，如图 2-22 所示。

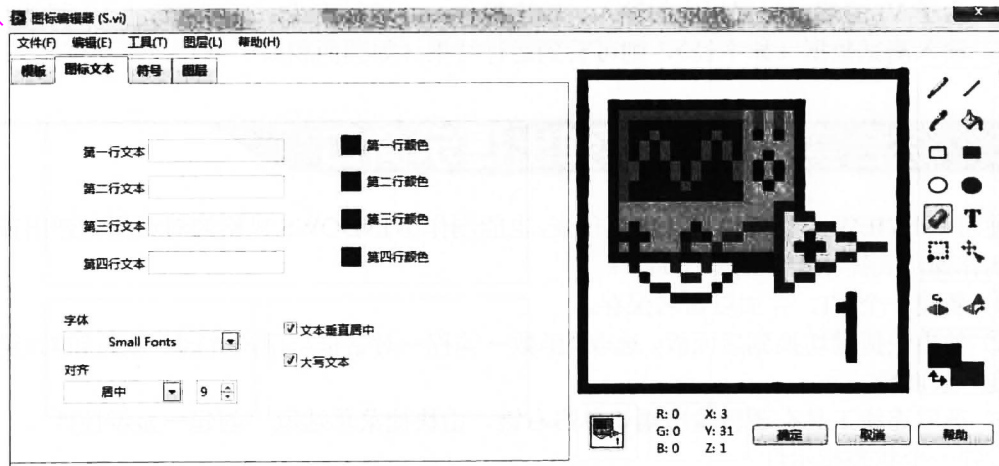



图 2-22 图标编辑器界面

② 使用选取工具，选取图标区域内黑色框线中的系统默认图标，并按<Delete>键予以删除。



③ 将字体大小设为“16”，用文本工具 **T** 在图标区域单击，然后输入“面积”2 字作为图标，则图标变为

④ 单击“确定”按钮，则退出编辑状态。

2.4.3 子 VI 调用

创建了图标和连线端口的 VI，即可作为子 VI 来供其他程序调用。具体的调用方法是：

① 新建或打开一个主 VI，再进入后面板。

② 从“函数”选板选取“选择 VI”子选板，则弹出“选择需打开的 VI”对话框，如图 2-23 所示。

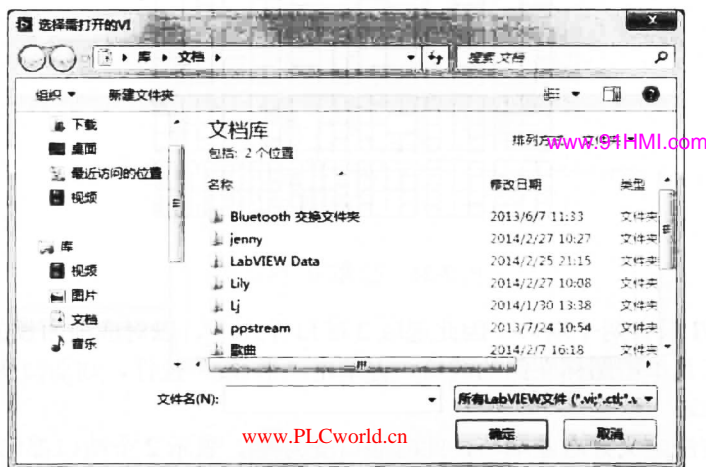


图 2-23 选择 VI 对话框

③ 在对话框中选择需要的 VI（如 S.VI），则被调用的子 VI 图标就被放置到主 VI 的程序框图中，便实现了调用子 VI。

④ 为子 VI 的端口配置相应的输入、输出控件，并进行连线，则可进行主 VI 测试。

⑤ 输入测试数据（如半径），则可以看到运行结果（如圆面积）。

2.5 综合实例：制作弹出提示窗口

使用 LabVIEW 不仅可以设计虚拟仪表，也能制作 WINDOWS 风格的对话框和弹出窗口。制作弹出式提示窗口的操作如下：

① 新建一个 VI，并加以命名保存。

② 使用快捷键切换到后面板，选取“函数→编程→对话框与用户界面→双按钮对话框”，将其拖入后面板。

③ 选取连线工具在逻辑输出端口单击右键，由快捷菜单选取“创建→显示控件”，为对话框设置一个状态指示灯。

④ 选取连线工具在消息输入端口单击右键，由快捷菜单选取“创建→常量”，来为对话框设置提示信息。

⑤ 用文字工具来选取常量，将常量值改为“你的时间都去哪了？”。

⑥ 将指示灯的标签改为“状态指示”，此时程序框图如图 2-24 所示。



⑦ 执行单次运行，其运行效果如图 2-25 所示。注意不要使用连线运行方式，否则难以退出运行。

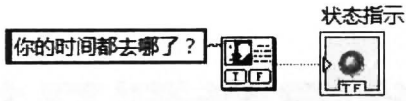


图 2-24 弹出式提示窗程序框图

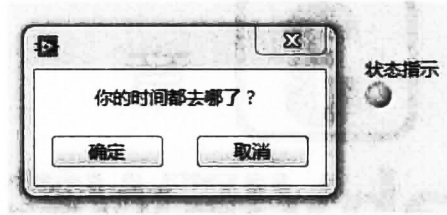
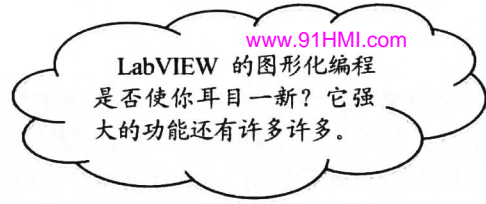


图 2-25 运行效果



www.91HMI.com
LabVIEW 的图形化编程
是否使你耳目一新？它强
大的功能还有许许多多。

www.PLCworld.cn



第 **3** 章

LabVIEW基本数据类型及其相互转换

www.91HMI.com

LabVIEW 是一种编程语言，其任务的核心就是处理数据。LabVIEW 所支持的数据类型十分丰富，其不但包括数值型、布尔型及字符串型，还包括波形数据、路径数据和 I/O 数据等，数据类型及其转换对 LabVIEW 编程有着重要的作用。

【本章导读】

- ◆ 基本数据类型
- ◆ 数据类型之间的转换

www.PLCworld.cn

3.1 基本数据类型

3.1.1 数值型

数值型数据是 LabVIEW 中最简单也是最常用的数据类型，它分为整型、浮点型及复数等类型，具体分类如表 3-1 所示。

表 3-1 数值型数据分类表

类型	输入控件 接线端	图标	颜色	存储 位数	数值范围	
整型	无符号 8 位整型			蓝色	8	0~255
	无符号 16 位整型			蓝色	16	0~65535
	无符号 32 位整型			蓝色	32	0~4294967295
	无符号 64 位整型			蓝色	64	0~18446744073709551615
	有符号 8 位整型			蓝色	8	-127~127
	有符号 16 位整型			蓝色	16	-32768~32768



续表

类型	输入控件 接线端	图标	颜色	存储 位数	数值范围	
整型	有符号 32 位整型			蓝色	32	-2147483648~2147483648
	有符号 64 位整型			蓝色	64	0~18446744073709551615
浮点型	单精度浮点数			橙色	32	最小正数 1.40e-45 最大正数 3.40e+38 最小负数 -1.40e-45 最大负数 -3.40e+38
	双精度浮点数			橙色	64	最小正数 4.94e-324 最大正数 1.79e+308 最小负数 -4.94e-324 最大负数 -1.79e+308
	扩展精度浮点数			橙色	128	最小正数 6.48e-4966 最大正数 1.19e+4932 最小负数 -6.48e-4966 最大负数 -1.19e+4932
复数	单精度浮点复数			橙色	64	实部与虚部分别与 单精度浮点数相同
	双精度浮点复数			橙色	128	实部与虚部分别与 双精度浮点数相同
	扩展精度浮点复数			橙色	256	实部与虚部分别与 扩展精度浮点数相同
定点数	定点数			灰色	任意	任意

数值型数据的创建包括变量与常数的创建。

(1) 数值型变量的创建

在前面板空白处单击右键，选取“控件→新式→数值”或“控件→经典→经典数值”，就可选取相应的数值型输入控件或显示控件。

(2) 数值型常量的创建

在 LabVIEW 内部集成了一些常用的高精度数学、物理和化学常量，如圆周率、重力常数、阿伏伽德罗常数等。

在后面板空白处单击右键，选取“编程→数学→数值→数学与科学常量”，就可选取相应的数值型常数。

提示

数值型变量是在程序运行时通过键盘或鼠标来设定，而常量只能在编程时设定，在程序运行时常数就不能再改变。

(3) 设置数值类型

在前面板，选取数值型输入控件或显示控件并单击右键，从快捷菜单中选取“表示法”，



则可设置所需类型。

3.1.2 布尔型

布尔型数据也称逻辑型数据，它只有真和假两个值，以对应布尔型控件的“开”和“关”的状态。

布尔型数据的创建有三种方法：

- ① 在前面板由菜单栏选取“查看→控件选板”→“新式→布尔”。
- ② 在前面板由菜单栏选取“查看→控件选板”→“经典→经典布尔”。
- ③ 在后面板由菜单栏选取“查看→函数选板”→“编程→布尔”。

布尔型控件按外形可分为按钮式、摇杆式、开关式、指示灯式及滑杆式等，如图 3-1 所示，凭此可设计出逼真的虚拟仪表界面。

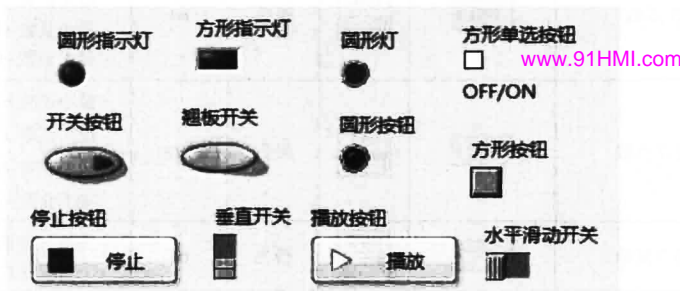


图 3-1 布尔型控件外观

3.1.3 数组

数组是相同数据元素组成的大小可变的集合，数组由两个要素组成，即元素和维数。数组元素的类型可以是数值型、布尔型、字符串型和波形等，但数组、表单及图形等不能作为数组元素。

数组可以是一维或多维的，每一维最多有 2^{31} 个元素。

数组的创建有 3 种，分别是由前面板创建、由后面板创建及用程序创建。

(1) 由前面板创建数组

在前面板，选取“控件→新式→数组、矩阵与簇→数组”。再向数组内放入一个数值输入控件，就创建了一个一维数组。若要显示多个数组成员，可选取定位工具，在数组的边框处拖动出现的网状拐角以增减显示区域，如图 3-2 所示。

若增加数组的维数，可在索引框处单击鼠标右键，并从快捷菜单选取“添加维度”。

为了显示更多的数组成员，可选取定位工具，将鼠标移动到数组上，会出现调节控制点，拖动调节控制点就可增加显示成员，其效果如图 3-3 所示。



图 3-2 数组添加输入控件前后

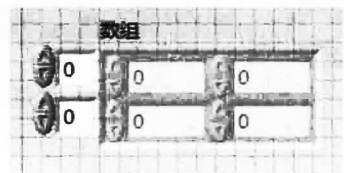


图 3-3 增添维数和显示成员



(2) 由后面板创建数组

在后面板, 选取“函数→编程→数组→数组常量”, 来放置一个数组常量, 如图 3-4 所示。然后再将一个数值常量放入数值常量框架中, 就创建了一个数值, 如图 3-5 所示。

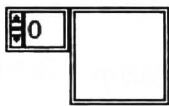


图 3-4 添加数组常量

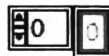


图 3-5 添加数值常量

若要添加多个数组成员, 可选取定位工具, 在数组的边框处拖动出现的调节控制点, 以增减数组成员, 如图 3-6 所示。若修改数值常量值, 可用文本工具点击相应的数值常量加以修改即可, 如图 3-7 所示。



图 3-6 增减数组成员



图 3-7 修改数值常量值

此外, 也可仿照前面板的方法来增添数组的维数等。

(3) 用程序创建数组

除了使用前、后面板来创建数组, 还可以通过 For 循环来创建数组, 具体的操作是:

① 选取“函数→编程→结构→循环”, 在后面板放置一个循环结构。

② 用连接工具点击变量 i , 以建立首个连接点。移动鼠标到循环结构的右边框, 如图 3-8 所示。

③ 单击鼠标, 以结束连线。

④ 在连线结束位置, 单击鼠标右键, 由快捷菜单选取“创建→显示控件”, 以存储和显示数值内容。

⑤ 在循环次数输入端点, 单击鼠标右键, 由快捷菜单选取“创建常量”, 并对循环次数加以修改, 如图 3-9 所示。这样就创建了一个数组元素分别为 0、1、2 的一维数组。

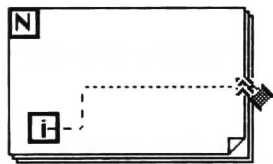


图 3-8 循环结构内连线

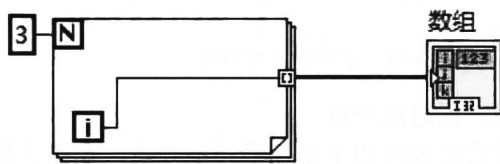


图 3-9 以程序建数组

3.1.4 字符串

字符串是一系列可显或不可显 ASCII 码字符的集合, 它提供了一种独立于操作平台的信息和数据格式。在 LabVIEW 中, 数据采集和仪器控制基本是以字符串的形式来传递数据。

字符串控件分为字符串输入控件、字符串显示控件和组合框控件。

(1) 创建字符串输入控件

字符串输入控件是以文本框的形式来输入字符串, 其创建方法是:

在前面板, 选取“控件→新式→字符串与路径→字符串输入控件”来添加一个字符串输



入控件，如图 3-10 所示。

(2) 创建字符串显示控件

字符串显示控件用于以文本框的形式来显示字符串，字符串有 4 种显示类型，即正常显示、“\”代码显示、密码显示和十六进制显示。

① 正常显示：是指显示可打印的字符串，对于不可打印的字符则以一个小方框替代。

② “\”代码显示：是指将所有不可显示的字符以反斜杠方式来显示，对于中文字符所显示的是其内码。

③ 密码显示：是指为了保护输入信息而以星号（*）来代替输入内容的显示方式，多用于密码输入等场合。

④ 十六进制显示：是指以十六进制的 ASCII 值来替代原内容的显示方法。

字符串显示控件的创建方法是：

在前面板，选取“控件→新式→字符串与路径→字符串显示控件”来添加一个字符串显示控件。

为了设置不同的显示方式，使用鼠标右键单击相应的字符串显示控件，再由快捷菜单选取相应选项即可。各显示方式的效果如图 3-11 所示，其程序框图如图 3-12 所示。

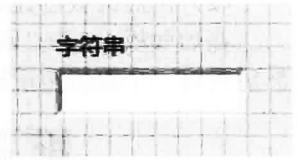


图 3-10 字符串输入控件

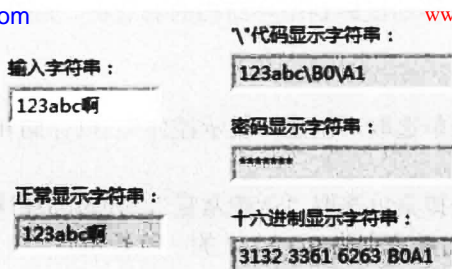


图 3-11 字符串显示效果

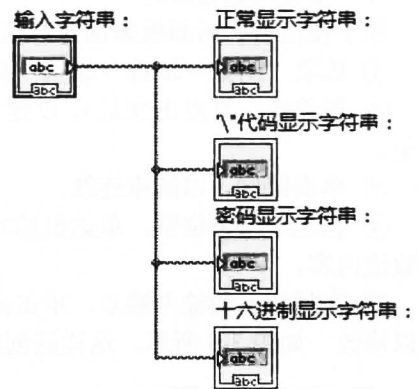


图 3-12 字符串显示效果程序框图

(3) 创建组合框

组合框控件用于创建字符串列表，在组合框中有多个字符串，每个字符串都是组合框的一个选项，并对应一个键值。

创建组合框的操作是：

① 在前面板，选取“控件→新式→字符串与路径→组合框”来添加一个组合框控件。为了添加各选项，使用鼠标右键单击组合框控件，由快捷菜单选取“编辑项...”选项，则弹出“组合框属性”对话框，如图 3-13 所示。

② 单击“插入”按钮，为对话框添加选项“电风扇”。若需要独立设置键值，可取消“值与项值匹配”前的对钩，然后修改值的内容。

③ 使用“插入”按钮，添加更多选项。

④ 单击“确定”按钮，退出组合框编辑。

⑤ 运行程序，其效果如图 3-14 所示。



图 3-13 组合框属性对话框

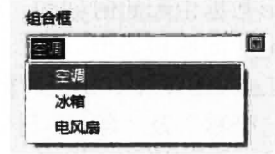


图 3-14 组合框运行效果
www.91HMI.com

3.1.5 簇

簇是由不同数据类型的元素所组成的集合，簇的成员可以是任意的数据类型，但不能同时将输入控件和输出控件放在一起。

创建簇有两种方法，分别是由前面板创建和由后面板创建。

(1) 由前面板创建簇

在前面板，选取“控件→新式→数组、矩阵与簇→簇”，再向簇的框架内放入相关的输入控件以构成一条记录信息，如图 3-15 所示。

(2) 由后面板创建簇

在后面板，选取“函数→编程→簇、类与变体→簇常量”，再向簇的框架内放入相关的数据常量以构成一条记录信息，如图 3-16 所示。

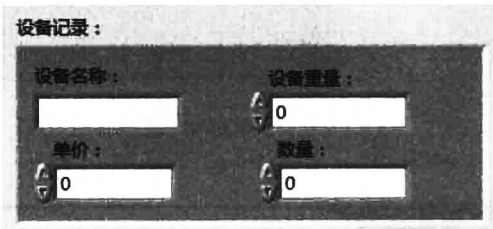


图 3-15 前面板创建簇

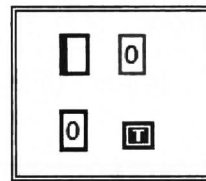


图 3-16 后面板创建簇

对于簇内对象的大小是可以自主调整的，右击簇边界，在弹出的快捷菜单中可选取相应的调节操作，如图 3-17 所示。

3.1.6 路径

路径是进行文件操作的关键参数之一，在 LabVIEW 中，涉及路径的控件有两种，即文件路径输入控件和文件路径显示控件。

创建路径的操作是：

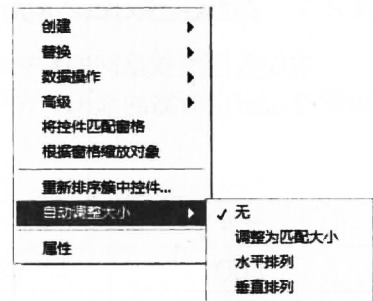


图 3-17 调整大小



在前面板，选取“控件→新式→字符串与路径”，然后视具体需要选取文件路径输入控件或文件路径显示控件，如图 3-18 所示。

3.1.7 波形数据

波形数据是 LabVIEW 特有的一种数据类型，它类似于簇，是由一系列不同数据类型的数据构成。

波形数据由起始时刻 t_0 、波形采样时间 dt 、波形数据 Y 以及波形属性所组成。

(1) 时间标识类型

在 LabVIEW 中，记录时间的专用数据类型是时间标识类型，时间标识的内容包括“时：分：秒：毫秒”及“年-月-日”。

创建时间标识的方法是：

在前面板，选取“控件→新式→数值”，然后视具体需要选取时间标识输入控件或时间标识显示控件，如图 3-19 所示。

(2) 波形显示控件

波形显示控件可用于波形数据的显示，其创建方法是：

在前面板，选取“控件→新式→I/O→波形”，如图 3-20 所示。在默认状态下，波形显示控件是不显示属性的。若要显示属性，可在控件上单击右键，再由快捷菜单选择“显示项→属性”。波形属性主要用于存储波形的长度、类型等参数信息。

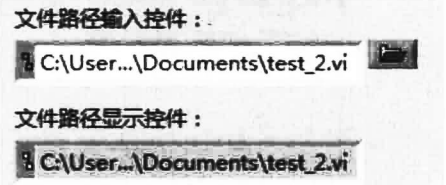


图 3-18 路径控件



图 3-19 时间标识控件

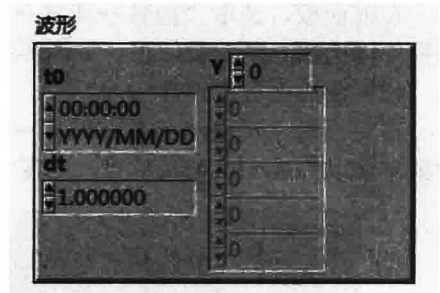


图 3-20 波形显示控件

3.2 数据类型之间的转换

3.2.1 数值型数据之间的转换

实现数值型数据间的转换可采用的方法是：在后面板，选取“函数→编程→数值→转换”，再按要求指定所需的选项。在转换子模板中，各图标的具体功能如表 3-2 所示。

表 3-2 数值型数据转换图标功能表

图标	功能
	将输入量转换为 8 位有符号整型
	将输入量转换为 16 位有符号整型



续表

图标	功能
	将输入量转换为 32 位有符号整型
	将输入量转换为 32 位有符号整型
	将输入量转换为 8 位无符号整型
	将输入量转换为 16 位无符号整型
	将输入量转换为 32 位无符号整型
	将输入量转换为 32 位无符号整型
	将输入量转换为单精度浮点数
	将输入量转换为双精度浮点数
	将输入量转换为扩展精度浮点数
	将输入量转换为单精度复数
	将输入量转换为双精度复数
	将输入量转换为扩展精度复数

www.91HMI.com

将一个单精度浮点型数据 3.1415 转换为 8 位无符号整数的程序框图和转换结果如图 3-21 所示。

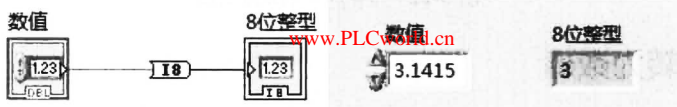


图 3-21 数值型数据转换

3.2.2 数值与字符串之间的转换

在 LabVIEW 中，数值可以转换为字符串，同样字符串也可以转换为数值，采用的方法是：在后面板，选取“函数→编程→字符串→数值/字符串转换”，再按要求指定所需的选项。在转换子模板中，各图标的具体功能如表 3-3 所示。

表 3-3 数值型数据转换表

图标	功能
	将数值转换为十进制数值字符串，“数字”端口用于输入转换数值，由“宽度”端口指定字符串长度，“十进制整形字符串”端口用于输出结果。当宽度不足时，则自动予以补齐。示例：15→“15”
	将数值转换为十六进制字符串，“十六进制整形字符串”端口用于输出结果，其他端口含义同上。示例：15→“F”
	将数值转换为八进制字符串，“八进制整形字符串”端口用于输出结果，其他端口含义同上。示例：15→“17”
	将数值转换为十进制小数字符串，“数字”端口用于输入转换数值，由“宽度”端口指定字符串长度，由“精度”端口指定小数点后位数，“F-格式字符串”端口用于输出结果，“使用系统小数点”端口用于决定是否使用系统小数点。当宽度不足时，则自动予以补齐。示例：15.9→“15.900000”
	将数值转换为十进制指数字符串，“E 格式字符串”端口用于输出结果，其他端口含义同上。示例：15.9→“1.590000E+1”



续表

图标	功能
	将数值转换为十进制工程型字符串，“工程字符串”端口用于输出结果，其他端口含义同上。示例：15.9→“15.900000E+0”
	将十进制数值字符串转换为数，“字符串”端口用于输入转换内容，“偏移量”端口指定读取字符串的位置，“默认”端口为0时标志数据类型为I32，“数字”端口用于输出结果，“数字后偏移量”端口用于输出整数部分的宽度。示例：“15”→15
	将十六进制字符串转换为数值，其他端口含义同上。示例：“F”→15
	将八进制字符串转换为数值，其他端口含义同上。示例：“17”→15
	将分数/指数字符串转换为数值，“默认”端口为0时标志数据类型为DBL，其他端口含义同上。示例：“1234.567”→1234

将一个字符串“1234.567”转换为整数的程序框图和转换结果如图3-22所示。

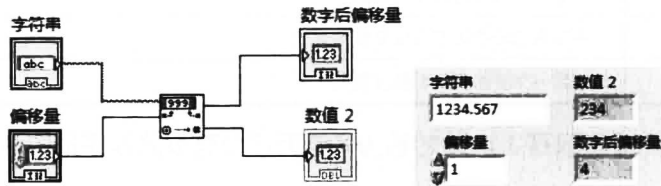


图 3-22 字符串转换为整数

www.PLCworld.cn

3.2.3 布尔值转为数值

布尔型与数值间也能够转换，在转换中 T 对应 1，F 则对应 0。将布尔型转换为数值的方法是：在后面板，选取“函数→编程→数值→转换→布尔值至(0,1)转换”，为“布尔”端口添加输入控件，为“0,1”端口添加输出控件，则程序框图和前面板如图 3-23 所示。



图 3-23 布尔值转为数值

3.2.4 路径与字符串之间的转换

路径可以转换成字符串，字符串也能够转换成路径，使用的方法是：在后面板，选取“函数→编程→字符串→路径/数组/字符串转换”，再按要求选取“路径至字符串转换”或“字符串至路径转换”。

将字符串转换为路径的程序框图和前面板如图 3-24 所示，将路径转换为字符串的程序框图和前面板如图 3-25 所示。

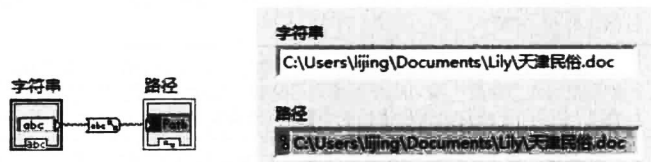


图 3-24 字符串转换为路径

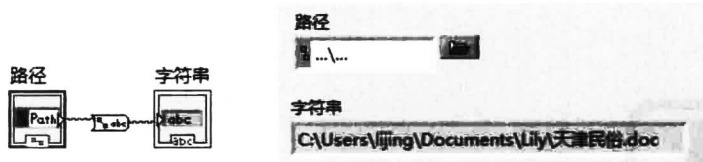


图 3-25 路径转换为字符串

3.2.5 数值与时间的转换

数值可以被转换成时间格式的数据，采用的方法是：在后面板，选取“函数→编程→数值→转换→转换为时间标识”，再为其两端添加输入和输出控件即可。将数值转换为时间的程序框图和前面板如图 3-26 所示。

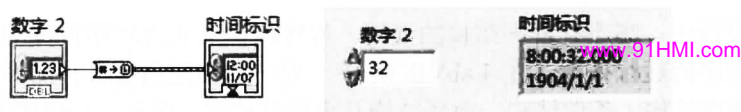


图 3-26 数值转换成时间



www.PLCworld.cn

合理地使用数据类型，
并加以适当地转换，有利
于灵活地进行数据处理。



第 4 章

LabVIEW程序结构

在程序设计中，离不开程序结构的使用。程序结构是实现程序设计的重要组成部分，它直接关系到程序的运行效率。在 LabVIEW 中，程序结构是通过各种图形化的程序结构来控制流程，如顺序结构、条件结构、循环结构及事件结构等。此外，LabVIEW 还通过节点、变量等来辅助程序设计。

【本章导读】

- ◆ 循环结构
- ◆ 顺序结构
- ◆ 条件结构
- ◆ 事件结构
- ◆ 定时结构
- ◆ 公式及 MATLAB 节点
- ◆ 反馈节点
- ◆ 变量
- ◆ 禁用结构等

www.PLCworld.cn

台湾永宏 PLC、四方变频器：15021230050

台湾永宏 PLC、四方变频器：15021230050

4.1 循环结构

循环结构有两种类型，即 For 循环和 While 循环。两种结构的差别在于，For 循环必须指定循环次数，当执行完指定次循环后就自动退出循环。While 循环则按条件来判别是否退出循环，无需指定循环次数。

4.1.1 For 循环

在后面板，建立 For 循环的方法是：选取“函数→编程→结构→For 循环”，然后在窗口空白区单击鼠标左键，向右下角拖动鼠标使虚线框达到相应大小，并单击鼠标左键结束放置。

For 循环含有两个重要的参数，即总循环次数 N 和当前循环次数 i 。设置总循环次数 N 的方法是，在“ N ”上单击鼠标右键，由快捷菜单选取“创建常量”，再将默认值“0”改为指定值。此时，For 循环如图 4-1 所示。

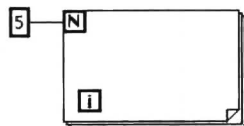


图 4-1 For 循环

值得指出的是，当前循环次数 i 是从“0”开始计数的，若循环 n



次, 则 i 的值为 $n-1$ 。For 循环在执行过程中是不能中断的, 必须在完成循环后才能停止。

循环程序的所有对象都应放置在循环框内, 实现的方法既可以是在循环框内直接添加, 也可以将事先建立的循环对象拖入循环框内。

4.1.2 While 循环

在后面板, 建立 While 循环的方法是: 选取“函数→编程→结构→While 循环”, 然后在窗口空白区单击鼠标左键, 向右下角拖动鼠标使虚线框达到相应大小, 并单击鼠标左键结束放置。

为了设置循环条件, 将鼠标移到循环条件输入端单击右键, 由快捷菜单选取“创建输入控件”即可, 此时, While 循环如图 4-2 所示。在执行循环时, 单击添加的“停止”按钮就可退出循环。

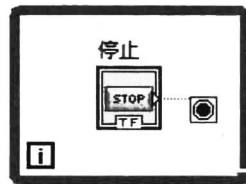


图 4-2 While 循环

对于循环程序内的所有对象, 其添加方法同 For 循环。

4.1.3 移位寄存器

www.91HMI.com

移位寄存器可理解为数据缓冲存储器, 通过快捷菜单可添加或删除移位寄存器数量, 它在循环结构中多用于实现迭代运算。

移位寄存器可对运算结构进行保存, 以便在下次运算中使用该运算结果。移位寄存器表现为循环框架左右两侧的向上、向下的图标, 它分别表示移位寄存器的前后两个状态。其中右侧的图标用来保存数据, 它在下次运行后会被移动到右侧的图标上保存, 新数据将继续存入右侧图标。以此循环往复执行, 直到满足条件结束循环。

下面以计算 $N!$ 的阶乘为例, 对移位寄存器的使用方法加以说明。

为了能自由地计算 $N!$, 必须在前面板放置一个数值输入控件和一个数值输出控件, 以输入 N 值和显示 $N!$ 的结果, 其界面如图 4-3 所示。

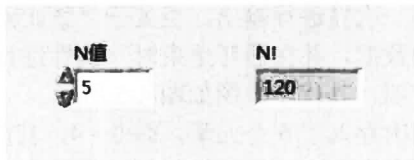


图 4-3 计算 $N!$ 的界面

计算 $N!$ 的程序框图如图 4-4 所示, 其操作步骤如下:

- ① 在后面板添加一个 While 循环, 并在 While 循环的边框上单击鼠标右键, 由快捷菜单中选取“添加移位寄存器”。
- ② 在 While 循环的内部添加一个加 1 函数, 并与循环变量 i 相连, 从而产生起始值为 1, 增量值为 1 的计数器。
- ③ 在循环外部, 为左侧的移位寄存器添加一个取值为 1 的常量, 作为阶乘的初始值。
- ④ 将数值输入控件的图标拖入循环内部, 并添加一个乘函数。
- ⑤ 用常量 1、乘函数、加 1 函数和移位寄存器构造一个计算连乘的功能体, 并由数值输出控件来显示计算结果。
- ⑥ 为了限定阶乘的范围, 向循环内部添加一个小于的判别函数, 由加 1 函数、值输入控件和它一起构成结束循环的条件, 并与循环条件端口相连。
- ⑦ 选取循环条件端口, 单击鼠标右键, 由快捷菜单中选取“真 (T) 时继续”。

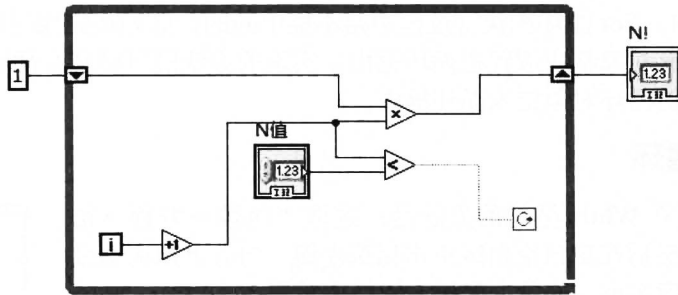


图 4-4 计算 N! 的程序框图

为验证设计的功能,在前面板由数值输入控件输入 5,则会在数值输出控件看到结果 120,即 $5! = 1 \times 2 \times 3 \times 4 \times 5 = 120$, 由此证明设计正确。

提示

移位寄存器在进行迭代运算中是一种有效的工具,不使用移位寄存器会很难实现设计目标。

4.1.4 框架通道

框架通道是循环结构内部与外部进行数据交换的通道,它位于循环结构的边框上,显示为小方格。对于输入而言,它是将循环外的数据保存到框架通道内以供输入。而对于输出来讲,它是将数据保存在框架通道内以供输出。

框架通道的属性分为“启用索引”和“禁止索引”两种状态,当处于“启用索引”状态时,框架通道充当数据缓冲器,每次循环的结果在通道内会按次序的先后而构成一个数组,在循环结束时可将数组的内容一次性进行输出。当处于“禁止索引”状态时,框架通道所存储的是最后一次循环所产生的数据,并在循环结束时一次性进行输出。

下面以一个示例来加以说明,其程序框图如图 4-5 所示。

运行该程序,可看到数组内存入了 5 个元素,即 0~4,其前面板如图 4-6 所示。若禁用索引,可用鼠标右键在框架通道上单击,再从框架菜单中选取“隧道模式”下的“最终值”。此时,会发现通向数组显示控件的连线被断开,说明连线错误。删去数组显示控件,为其添加一个新的显示控件,可以发现该显示控件是一个数值显示控件,说明其结果只是一个数值,而非原来的数值了。

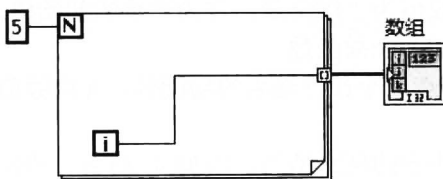


图 4-5 验证框架通道作用的程序框图

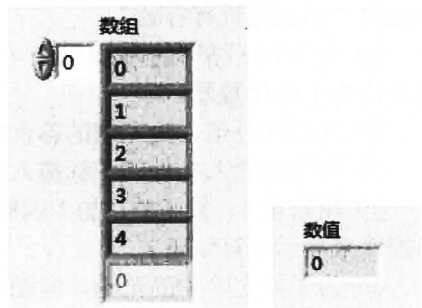


图 4-6 启用与禁止框架通道的效果



4.2 顺序结构

在 LabVIEW 中, 为了使程序能够按指定顺序执行, 必须要用顺序结构来控制运行流程。顺序结构分为平铺式顺序结构和层叠式顺序结构, 在顺序结构中, 是以帧划分来执行单元的, 每一帧就如同电影胶片的每一幅图像, 它涵盖了一个执行单元的所有内容。

4.2.1 平铺式顺序结构

平铺式顺序结构的特点是形式直观, 在结构内容较多时平铺式顺序结构会较占屏幕空间, 其框图如图 4-7 所示。

平铺式顺序结构的创建方法是, 在后面板选取“函数→编程→结构→平铺式顺序结构”, 然后在空白位置单击鼠标并进行拖动, 直至大小合乎要求再单击鼠标结束放置。

若添加或删除帧, 可在帧上单击鼠标右键, 由快捷菜单中选取“在前面添加帧”、“在后面添加帧”或“删除本帧”、“删除顺序”。

4.2.2 层叠式顺序结构

层叠式顺序结构的特点是节省屏幕空间占用, 其框图如图 4-8 所示。

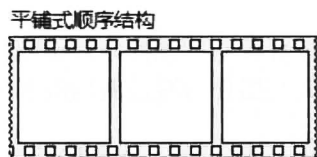


图 4-7 平铺式顺序结构

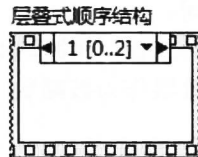


图 4-8 层叠式顺序结构

层叠式顺序结构的创建方法是, 在后面板选取“函数→编程→结构→层叠式顺序结构”, 然后在空白位置单击鼠标并进行拖动, 直至大小合乎要求再松开鼠标。

为层叠式顺序结构添加或删除帧的方法同平铺式顺序结构相同。

由于层叠式结构的每一帧是叠层放置的, 因此在浏览帧内容时需要使用标签选择器, 左侧三角表示显示前一帧, 右侧三角表示显示后一帧。点向下的三角, 可通过下拉列表快速跳转到指定帧。

提示

由于平铺式顺序结构与层叠式顺序结构具有相同的逻辑含义, 因此在后面板空间紧张的情况下, 使用层叠式顺序结构替代平铺式顺序结构是一种有效的方法。

在空间允许的条件下, 平铺式顺序结构则具有很好的可读性。

4.2.3 顺序结构使用

使用顺序结构不仅能够限定程序执行的流程, 在与计时器配合的情况下, 还能够测量流程中两个节点间的时间差。图 4-9 所示的程序框图是一个检测操作者反应时间的功能体, 多用于回答测试题的时间测量, 其制作过程如下。

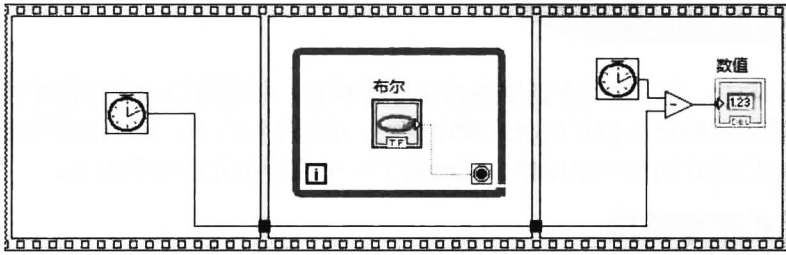


图 4-9 时间测试程序框图

① 新建一个 VI，然后在前面板放置一个数值输出控件和一个布尔按钮，如图 4-10 所示。

② 在后面板放置一个平铺式顺序结构，并添加 2 个新帧。www.91HMI.com

③ 选取“函数→编程→定时→时间计数器”，然后在第 1 帧的空白位置单击鼠标进行放置，该时间计数器的作用是获取测试的起始时间。

④ 在第 3 帧再放置一个时间计数器，该时间计数器的作用是获取测试的结束时间。

⑤ 在第 3 帧放置一个减函数，并将数值输出控件移动到该帧，以作为减函数的输出。将该帧的时间计数器作为被减数，由第 1 帧的时间计数器作为减数就构成了一个计算时间差的功能体。
www.PLCworld.cn

⑥ 为产生时间差，在第 2 帧放置一个循环结构，并将布尔按钮移至该帧，通过它来控制循环的结束。运行该功能体，则可获取毫秒级的时间差。

除此之外，若想动态显示时间差，按图 4-11 来设计前面板，并将程序框图修改为图 4-12 的内容即可。

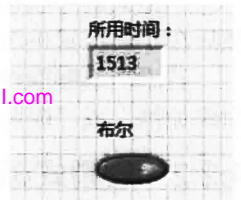


图 4-10 时间检测界面



图 4-11 动态显示的时间检测界面

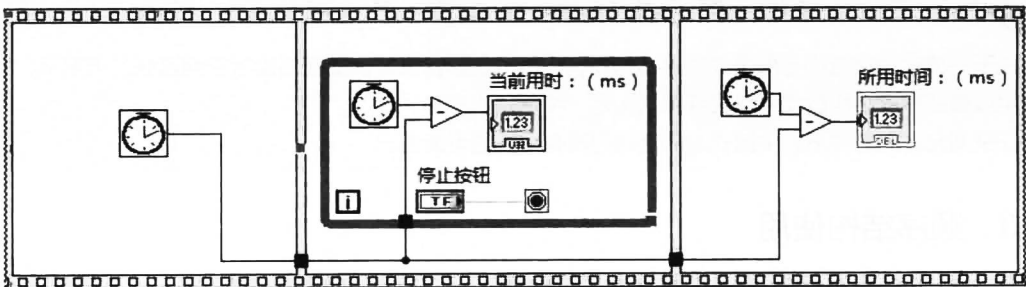


图 4-12 动态显示的时间检测程序框图



4.3 条件结构

条件结构的用途在于按执行条件控制程序的运行方向，它所支持的控制条件有 3 种数据类型，即布尔型、数值型和字符串型。控制条件的数据类型必须与框图标识符中的数据类型相匹配，否则无法执行。

(1) 布尔型条件

布尔型条件的取值为“真”与“假”，故它只有两个分支，实现其测试的制作步骤如下：

① 新建一个 VI，在前面板放置一个布尔按钮和一个数字输出控件，有数字输出控件来指示不同条件下的输出结果，其界面如图 4-13 所示。

② 在后面板，选取“函数→编程→结构→条件结构”，然后在空白位置单击鼠标并进行拖动，直至大小合乎要求再松开鼠标。

③ 将布尔按钮和条件结构的输入端连接起来，以提供条件值。

④ 在真假两个条件下分别添加 2 个数值常量，并赋予不同的值以区分变化。

⑤ 分别将 2 个数值常量与数字输出控件连接起来以供显示，其程序框图如图 4-14 所示。运行该程序即可观察到不同条件下的变化结果。



图 4-13 测试界面

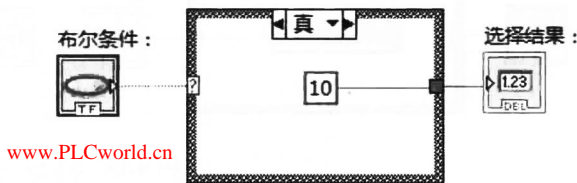


图 4-14 测试程序框图

(2) 数值型条件

数值型条件的取值为 0, 1, 2, ..., 故它可以有多个分支，实现其测试的制作步骤如下：

① 新建一个 VI，在前面板放置一个数值输入控件和一个数字输出控件，其界面如图 4-15 所示。

② 在后面板，添加一个条件结构，并将数值输入控件与条件结构的输入条件连接起来，此时可发现框图标识符由“真”、“假”变为 0、1。

③ 在框图标识符上单击鼠标右键，由快捷菜单选取“在前面添加分支”或“在后面添加分支”，则可添加分支；选取“删除本分支”或“删除空分支”，则可删除分支。此外，选取“复制分支”，则可以采用复制本分支的方法来添加分支。

④ 分别为每个分支添加一个数值常量，并与数字输出控件连接起来，则程序框图如图 4-16 所示。

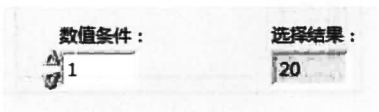


图 4-15 测试界面

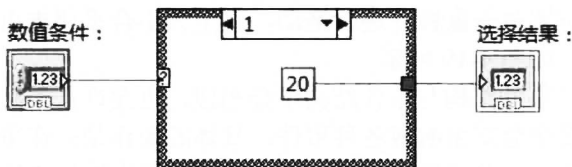


图 4-16 测试程序框图



运行该程序，则可观察到多分支处理的变化结果。

(3) 字符串条件

字符串条件的取值为相应的字符串，所以它也可以有多个分支，实现其测试的制作步骤如下：

① 新建一个 VI，在前面板放置一个字符串输入控件和一个字符串输出控件，其界面如图 4-17 所示。

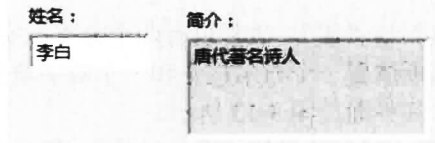


图 4-17 测试界面

② 在后面板，添加一个条件结构，并将字符串输入控件与条件结构的输入端连接起来。

③ 增添分支并修改框图标识符上的条件值，如图 4-18 所示。

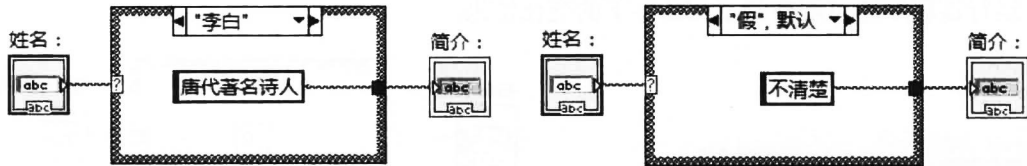


图 4-18 测试程序框图

④ 为每个分支添加一个字符串常量，并与字符串输出控件相连。

运行该程序，则可观察到相应的处理结果。

4.4 事件结构

在计算机的使用过程中，离不开键盘和鼠标的操作，这些操作可以事件的形式来通知系统进行响应。LabVIEW 虽属图形化编程，但它同样支持事件驱动。在事件发生前，程序一般是处于循环等待状态。一旦事件被触发，则立即响应事件并执行相应的功能代码，处理后再重新返回等待状态。

4.4.1 创建事件结构

要响应事件，必须创建事件结构，其操作方法是：在后面板，选取“函数→编程→结构→事件结构”，然后在空白位置单击鼠标并进行拖动，直至大小合乎要求再松开鼠标，如图 4-19 所示。

事件结构与条件结构十分相似，也允许在其结构下添加多个分支来响应各种事件，具体的操作是：在事件结构的边框上单击鼠标右键，再由快捷菜单中选择“添加事件分支”或“复制事件分支”。

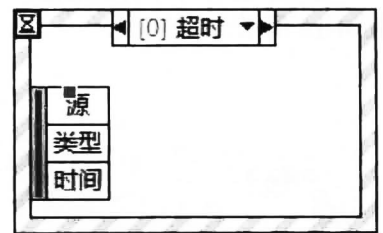


图 4-19 事件结构图标



4.4.2 事件结构应用

在程序设计中，为增强系统的支持性会设计各种在线帮助功能，使用事件结构则可以实现这种目标。下面设计一个事件，在点击鼠标时即弹出提示对话框，其操作步骤如下。

- ① 在后面板，添加一个事件结构。
- ② 为事件结构添加一个事件分支，则弹出编辑事件对话框，如图 4-20 所示。

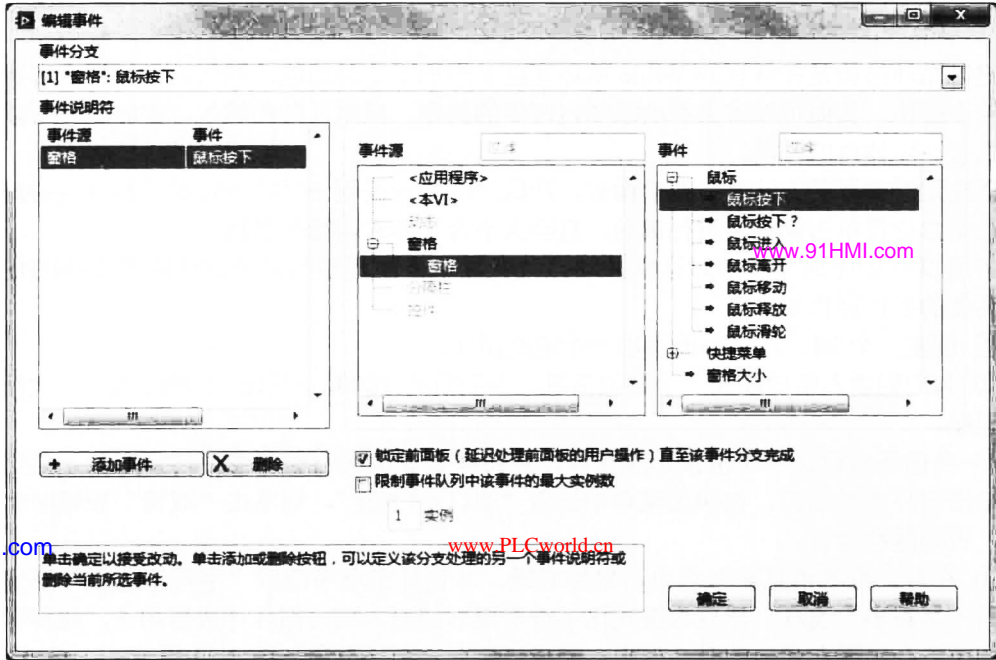


图 4-20 编辑事件对话框

③ 在事件源中选取“窗格”作为对象，在事件中选取“鼠标按下”，然后点击“确定”按钮。若使用“添加事件”和“删除”按钮，可对事件进行增删处理。

④ 在“鼠标按下”分支，添加一个单按钮对话框控件，并为对话框的消息端口创建一个常量，以提供显示信息，完成后如图 4-21 所示。

运行该程序，则在按下鼠标时会出现图 4-22 所示的提示对话框。

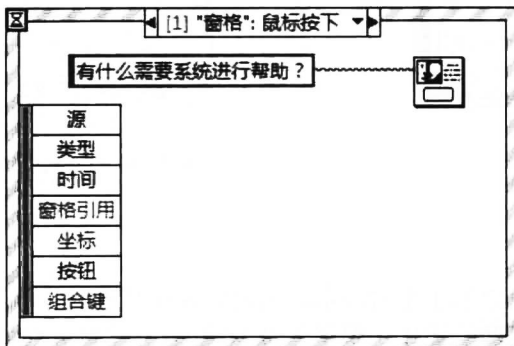


图 4-21 事件处理程序框图

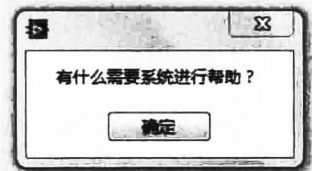


图 4-22 事件响应效果



4.5 定时结构

定时结构相对于常规的循环结构具有循环执行时间更精确的功效，定时结构分为定时顺序和定时循环。此外，还包含创建定时源、清除定时源、同步定时结构开始、定时结构停止和创建定时源层次结构等，其内容都较为简单。

4.5.1 定时循环

定时循环结构是在常规的 While 循环基础上添加了定时功能，因而它常被用来解决周期性的轮询操作，从而可以省去手动添加定时器的操作。根据外部的输入，定时循环可以周期性地执行循环体内的程序。

创建定时循环的方法是：在后面板，选取“函数→编程→结构→定时结构→定时循环”，然后在空白位置单击鼠标并进行拖动，直至大小合乎要求再松开鼠标。

下面以一个周期性弹出对话框并动态改变显示值的程序为例来说明定时循环的使用方法，具体的操作过程如下。

① 创建一个 VI，在后面板添加一个定时循环。

② 为周期输入端口添加一个数值常量，并取值为 1000，则每过 1 秒就执行一次结构内部的程序。

③ 在循环内添加一个双按钮对话框，并将按钮输出端与循环条件端口连接。再用鼠标右键单击循环条件端口，在快捷菜单中选取“真(T)时继续”，则单击“取消”按钮时将退出循环，否则继续运行。

④ 在对话框的消息输入端单击鼠标右键，从快捷菜单中选取“字符串选板”下的“格式化写入字符串”选项。然后将格式化字符串图标的输入端与循环计数器相连，这样循环计数器的值将会出现在弹出的对话框中。完成连接后，程序框图如图 4-23 所示。

运行该程序，其界面如图 4-24 所示。每隔 1 秒就会弹出一次对话框，其内容是从 1 开始自动进行加 1 运算。单击“确定”键将连续执行，按“取消”键将退出循环。

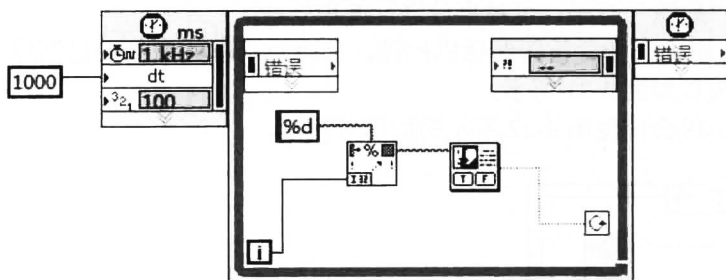


图 4-23 定时循环程序框图

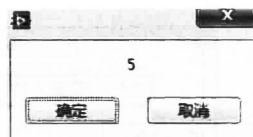


图 4-24 对话框形式

4.5.2 定时顺序

定时顺序结构是在常规的顺序结构基础上添加了定时功能，因而它常被用在解决周期性的顺序操作方面，从而可以省去手动添加定时器的操作。根据外部的输入，定时顺序结构可以周期性地执行顺序结构体内的程序。



创建定时循环的方法是：在后面板，选取“函数→编程→结构→定时结构→定时顺序”，然后在空白位置单击鼠标并进行拖动，直至大小合乎要求再松开鼠标。

同顺序结构一样，定时顺序结构也可以添加多个帧，以便执行多种处理。

下面以一个周期性顺序弹出对话框为例来说明明定时顺序结构的用法，具体的操作过程如下。

① 创建一个 VI，在后面板添加一个定时顺序结构。

② 为偏移输入端口添加一个数值常量，并取值为 1000，则每过 1 秒就执行一次结构内部的程序。

③ 在循环内添加一个单按钮对话框，并为消息输入端口添加一个字符串常量，将字符串常量的值设为“采集温度”。

④ 在本帧后面添加两个新帧，并将第 1 帧的内容复制到第 2 和第 3 帧。再将字符串常量的值分别改为“采集压力”和“采集扭矩”。

⑤ 在每帧的边框处单击鼠标右键，由快捷菜单取消“显示左数据节点”和“显示右数据节点”，则程序框图如图 4-25 所示。

运行该程序，其界面如图 4-26 所示。每隔 1 秒就按顺序弹出一轮对话框，其内容依次是“采集温度”、“采集压力”和“采集扭矩”。

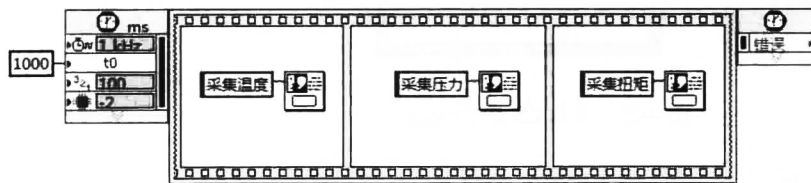


图 4-25 定时顺序程序框图

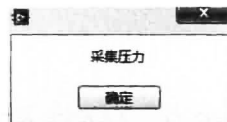


图 4-26 对话框形式

4.6 公式及 MATLAB 节点

LabVIEW 在程序方面是面向数据流的，因此它对于复杂的数学计算，如数学公式计算等是较为麻烦的。在这种情况下，就需要借助 LabVIEW 的公式节点和 MATLAB 节点来解决。

4.6.1 公式节点

公式节点的作用相对于数值运算子程序，因此它需要创建输入变量和输出变量来进行参数的传递。在公式节点的内部是采用类似 C 语言的语句来编写，每条语言用分号来隔开。公式节点所能包含的变量数和内部的语句个数没有限制。在输入或输出变量中不允许有重名变量，但在输入和输出变量间可以有同名变量出现。

创建公式节点的方法是：

① 在后面板，选取“函数→编程→结构→公式节点”，然后在空白位置单击鼠标并进行拖动，直至大小合乎要求再松开鼠标。

② 在公式节点框架上单击鼠标右键，由快捷菜单中选取“添加输入”或“添加输出”，可增添输入或输出变量，添加变量前后如图 4-27 所示。

下面以计算 $y = x1 + x2$ 为例来介绍公式节点的使用方法，其制作步骤如下。

① 新建一个 VI，在前面板放置两个数值输入控件和一个数值输出控件，并将其标签改为 $x1$ 、 $x2$ 和 y ，则界面如图 4-28 所示。



图 4-27 添加变量前后变化

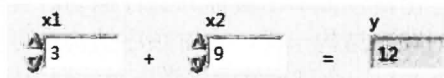


图 4-28 计算界面

- ② 在后面板, 添加一个公式节点。
- ③ 在框架的左侧添加两个输入变量, 并将其命名为 x1 和 x2。
- ④ 在框架的右侧添加一个输出变量, 并命名为 y。
- ⑤ 将控件与同名变量相连, 并在节点内部输入计算语句, 完成后如图 4-29 所示。通过运行程序, 可验明程序设计的正确性。

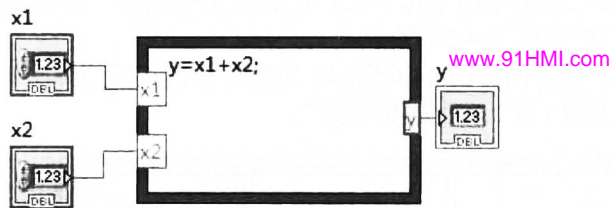


图 4-29 计算处理程序框图

4.6.2 MATLAB 节点

Math Script 是一种面向数学的文本编辑语言, 其核心是高级文本化编程语言, 具有对信号处理、分析和数学计算等复杂性问题进行抽象化处理的语法和功能。Math Script 与 MathWorks 公司出品的 MATLAB 软件具有兼容性, 并支持 m 文件的脚本语法, 从而对 m 文件的开发成果加以继承。

LabVIEW 是通过 MATLAB 节点来调用 Math Script 编程的, 从而增强程序设计的灵活性。下面以绘制正弦波为例来介绍 MATLAB 节点的使用方法, 其制作步骤如下。

- ① 新建一个 VI, 在前面板放置一个垂直指针滑动杆控件和一个波形图控件, 如图 4-30 所示。

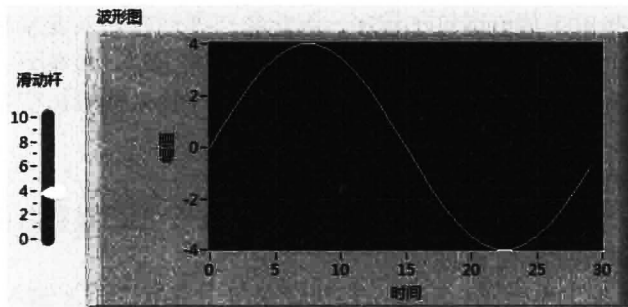


图 4-30 测试界面

- ② 在后面板, 选取“函数→数学→脚本与公式→脚本节点→MATLAB 脚本”, 然后在空白位置单击鼠标并进行拖动, 直至大小合乎要求再松开鼠标。



③ 在脚本框架的左侧添加一个输入变量 a ，再在框架右侧添加一个输出变量 b 。为了使变量 B 的值能够绘制出波形，必须将其数据类型改为 DBL 型的一维数组。

④ 在节点内部输入语句“ $x=\text{linspace}(0,2*\pi,30);$ ”和“ $b=a*\sin(x);$ ”。语句“ $x=\text{linspace}(0,2*\pi,30);$ ”的作用是在 $0\sim 2*\pi$ 间均匀取 30 个值存入变量 x ，语句“ $b=a*\sin(x);$ ”的作用是按变量 x 的角度值求出幅值为 a 的正弦值，进而通过连线绘制出波形曲线。

⑤ 将滑动杆与变量 a 相连，将波形图与变量 b ，则程序框图如图 4-31 所示。

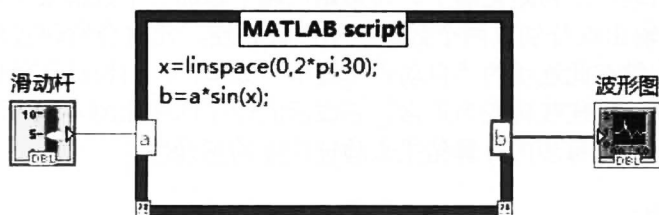


图 4-31 测试程序框图

在安装服务器软件的前提下，运行该程序则可看到正弦曲线。调节滑动杆的位置，可改变波形的幅值。

4.7 反馈节点

反馈节点的作用是在循环结构间传递数据，它与移位寄存器的工作机制相同，也是在循环结束的时候进行数据存储，并将数据传送到下一个循环。但它只相当于有一个左侧端子的移位寄存器，不能任意增添缓冲的容量。当把一个函数对象的输出端连接到其输入端时，就会自动形成一个反馈节点。

反馈节点所支持的数据类型有数字、字符串和数组等，反馈节点的数据传送方向按图标上箭头的指示方向。

创建反馈节点的操作是：在后面板，选取“函数→编程→结构→反馈节点”，然后在空白位置单击鼠标，其形式如图 4-32 所示。

反馈节点也可以进行初值的初始化，方法是在标有“*”的输入端为其连接一个常量。

下面以计算 1 到 N 的累加和为例，介绍反馈节点的使用方法，其制作步骤如下。

① 创建一个 VI，在前面板放置一个数值输入控件、一个数值输出控件和一个数值显示控件，并将控件标签修改为图 4-33 的形式。



图 4-32 反馈节点图标

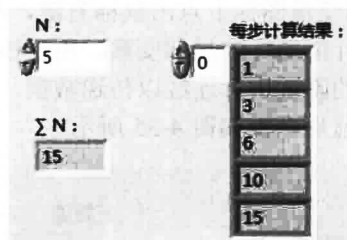


图 4-33 计算累加和界面

② 在后面板，放置一个 For 循环结构，将循环次数输出端与数值输入图标相连接。



③ 在循环结构框架内放置一个加 1 函数，并与循环计数器 i 相连，从而实现从 1 开始计算累加和。

④ 在循环结构框架内放置一个加函数和一个反馈节点，将加 1 函数的输出端与加函数的一个输入端相连，再将反馈节点的输出端与加函数的另一个输入端相连，然后将加函数的输出端与反馈节点的输入端连在一起，从而实现 1 到 N 的累加计算。

⑤ 用鼠标右键单击反馈节点，从快捷菜单中选取“将初始化器移出一层循环”，则发现在循环框架的左侧出现一个初始化端子。为初始化端子添加一个数值常量，并取值为零。

⑥ 将加函数的输出端分别以两个数值显示图标相连，此时会发现数值输出图标的连线出现错误。用鼠标右键在此连线的“自动索引隧道”处单击，由快捷菜单选取“隧道模式”下的“最终值”选项，则连线转变为正常，完成后的程序框图如图 4-34 所示。

运行该程序，可通过每步的计算结果来验证程序的正确性。

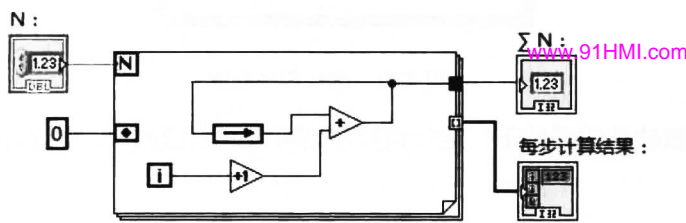


图 4-34 计算累加和程序框图

4.8 变量

4.8.1 局部变量

局部变量是指处在同一个 VI 程序中并能够在不同位置被访问的变量，它可实现程序内的数据传递。创建局部变量的方法有两种，一种是直接在后面板通过函数模板中的创建局部变量，然后再指向某个对象；另一种是直接针对已有的对象创建其局部变量。

方法一的操作过程：

① 在前面板放置一个水平指针滑动杆，以便作为局部变量的主体对象。

② 在后面板，选取“函数→编程→结构→局部变量”，然后在空白位置单击，则呈现一个局部变量图标。

③ 在局部变量图标上点击鼠标右键，从快捷菜单中选取“选择项→滑动杆”，则该变量就成为水平指针滑动杆的局部变量。由于该变量具有水平指针滑动杆的数值属性，因此它可放置在使用者的附近进行连线以传递数据，避免了凌乱的图标间连线。

完成后的前后面板如图 4-35 所示。

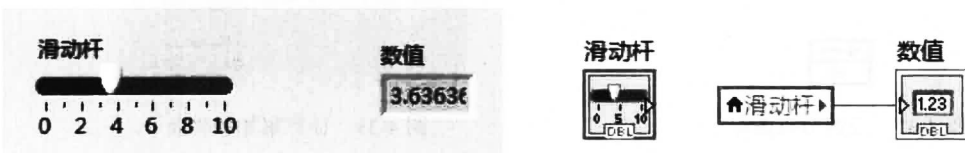


图 4-35 局部变量使用



方法二的操作过程:


在前面板放置一个水平指针滑动杆,然后在后面板用鼠标右键单击滑动杆图标,由快捷菜单中选取“创建→局部变量”即可。

局部变量既可以从所指向的对象读取数据,也可以向其写入数据。在修改局部变量的类型时,只需在局部变量的图标单击鼠标右键,然后视需要选择“转换为读取”或“转换为写入”即可。

4.8.2 全局变量

全局变量是指在多个 VI 间能够实现数据传递的变量,它可使多个程序间相互并行。

创建全局变量的操作过程如下。

① 在后面板,选取“函数→编程→结构→全局变量”,然后在空白位置单击,则呈现一个全局变量图标 。

② 在全局变量图标上点击鼠标右键,从快捷菜单中选取“打开前面板”,则新创建了一个 VI 文件。

③ 在新建的 VI 文件中,由前面板放置一个水平指针滑动杆。

④ 返回到原程序框图,选取全局变量图标,单击鼠标右键,由快捷菜单中选取“选择项→滑动杆”。再次激活快捷菜单,从中选取“转换为读取”。

⑤ 为了显示滑动杆的数值,在前面板放置一个数值输出控件。

⑥ 在后面板,将数值输出图标与全局变量进行连接,则可显示滑动杆的变化情况。

⑦ 运行该程序,则可发现拖动另一文件中的水平滑动杆,在本 VI 文件中可读取到变化的数值,说明全局变量发挥了作用。

www.91HMI.com 对两个 VI 程序进行命名保存。 www.PLCworld.cn

完成后的程序框图如图 4-36 所示,运行时的界面效果如图 4-37 所示。

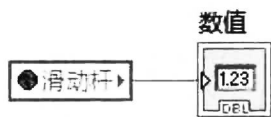


图 4-36 全局变量使用的程序框图

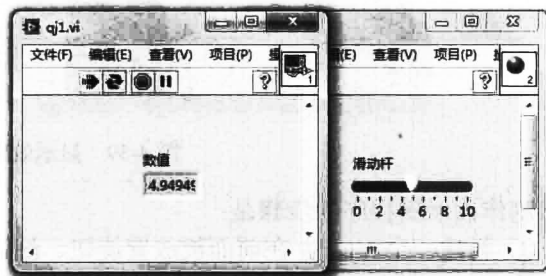


图 4-37 全局变量使用效果

在建有全局变量 VI 的情况下,在后面板使用“函数→选择 VI…”也可选取已有的全局变量来进行使用。

全局变量同局部变量一样,也可修改变量的类型,其方法是在全部变量的图标单击鼠标右键,然后视需要选择“转换为读取”或“转换为写入”。

提示

尽管 LabVIEW 通过了局部变量和全局变量的用法,但并不提倡更多地使用它们。因为更多地使用局部变量和全局变量会打乱 LabVIEW 以数据流为主的驱动方式,降低程序的可读性和可维护性。此外,全局变量会长期占用系统的内存占用,降低程序的执行效率。



4.9 禁用结构

禁用结构包括程序框图禁用结构和条件禁用结构，其主要作用就是限定已有程序片段的运行，并起到一定的注释用途。

4.9.1 程序框图禁用结构

面对一段辛辛苦苦设计的程序，一旦有了新的设计思路就将其全部删除显然不是一个聪明的选择。此时，若使用程序框图禁用结构，则可以立即限定它的执行并保留这份劳动成果。

程序框图禁用结构通常有两个分支，一个是禁用分支，另一个是启用分支。当需要再次使用禁用的程序片段时，只要把此程序片段放入启用分支即可恢复其作用。

创建程序框图禁用结构的方法如下。

① 在后面板，选取“函数→编程→结构→程序框图禁用结构”，然后在空白位置单击鼠标并进行拖动，直至大小合乎要求再松开鼠标，其形式如图 4-38 所示。



图 4-38 程序框图禁用结构

② 将禁用的程序片段拖放到禁用分支框架内，此时禁用的程序片段呈灰色显示状态，表示其已被禁用。

下面以一个选择界面显示效果作例子来说明程序框图禁用结构的作用，其界面如图 4-39 所示。该程序的作用是通过旋钮来为显示控件提供数值，对比显示效果。

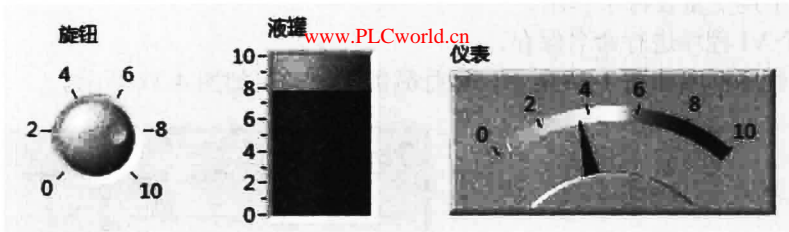


图 4-39 显示效果选择界面

制作该示例程序的步骤是：

- ① 新建一个 VI，并在前面板放置旋钮、液罐和仪表这三个控件。
- ② 在后面板放置一个程序框图禁用结构，然后将液罐和仪表两个控件分别拖放到启用分支和禁用放置。
- ③ 将旋钮控件依次与液罐和仪表控件相连，则程序框图如图 4-40 所示。

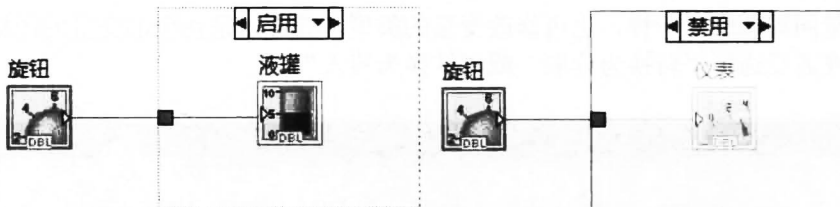


图 4-40 程序框图禁用结构使用

运行该程序，通过旋转到可发现液罐的液位一起随着变化，而仪表的指针却一丝也没



有反应，说明仪表已被禁用。

4.9.2 条件禁用结构

如果程序框图禁用结构的作用是完全禁用某一程序片段，那条件禁用结构的作用就是视具体的运行环境来禁止程序的运行。它与条件结构的差别在于条件结构是在运行时动态决定程序执行哪一支，而条件禁用结构是在编译时就选择好要禁用哪个分支，禁用的程序代码在运行时根本就不会放入内存。

创建条件禁用结构的方法如下。

① 在后面板，选取“函数→编程→结构→条件禁用结构”，然后在空白位置单击鼠标并进行拖动，直至大小合乎要求再松开鼠标，其形式如图 4-41 所示。



图 4-41 条件禁用结构

② 在条件禁用结构的框架上单击鼠标右键，由快捷菜单中可选取“在前面添加子程序框图”、“在后面添加子程序框图”或“复制子程序框图”来添加分支，此时将弹出的“配置条件”对话框如图 4-42 所示。



图 4-42 配置条件对话框

配置条件对话框用于设置执行某一支的条件，它由“条件表达式”和“设置为默认”选项构成，其中：符号列表框提供了一些条件禁用结构所预定义的符号，具体如图 4-43 所示。如 TARGET_TYPE 就指目标代码在何种操作系统下运行，可供选择的值如图 4-44 所示。

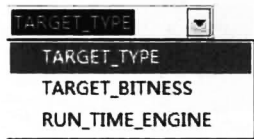


图 4-43 预定义条件符号

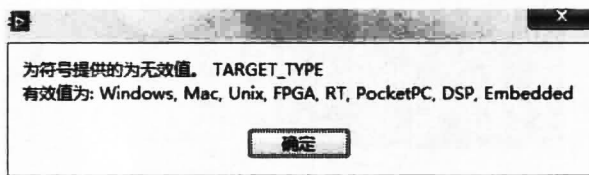


图 4-44 条件符号的取值范围

判别的条件有“=”（相等）和“!=”（不等）。

“值”是指选定符号下的取值，不同的符号有不同的取值范围。

“-”和“+”按钮分别用于移除表达式或添加表达式。

“与”和“或”连接符用于对多个条件表达式进行连接，“与”是指多个条件同时满足，“或”是指多个条件中有一个满足即可。

“设置为默认”选项是指将当期分支的程序框图是否设置为默认的处理分支。

③ 使用快捷菜单中的“编辑本子程序框图的条件”，可对已设置的条件加以修改。

为说明条件禁用结构的使用，下面制作一个验证程序，其制作步骤如下。

① 新建一个 VI，在前面板放置一个水平进度条和一个水平指针滑动杆控件，如图 4-45



所示。

② 在后面板放置一个条件禁用结构，将水平进度条图标拖入默认分支，并由随机函数为其提供数值。此时，可发现分支内的图标已变为灰色，表示不可执行。

③ 为条件禁用结构添加一个新分支，将条件设置为“TARGET_TYPE==Windows”。

④ 将水平指针滑动杆图标拖入新的分支，并为其连接一个随机函数来提供变化数值，完成后的程序框图如图 4-46 所示。

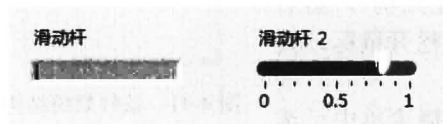


图 4-45 界面布局

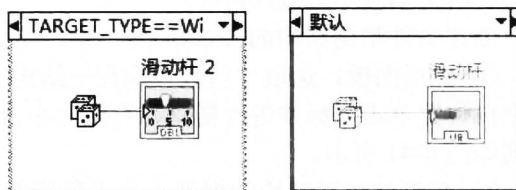


图 4-46 验证程序框图

运行程序可发现水平指针滑动杆在不停摆动，说明运行环境符合设置条件的要求，是 Windows 环境。

4.10 综合实例：数字平滑滤波

在虚拟仪表设计中，对于采集的数据需要进行必要的数据平滑处理，否则脉动的数据不利于数据的准确分析。实现数据平滑处理的设计思想是：利用移位寄存器的缓冲作用对若干个连续的数据求取均值来替代原始采集值，从而起到平滑数据的效果。在平滑处理前后，其的波形如图 4-47 和图 4-48 所示。

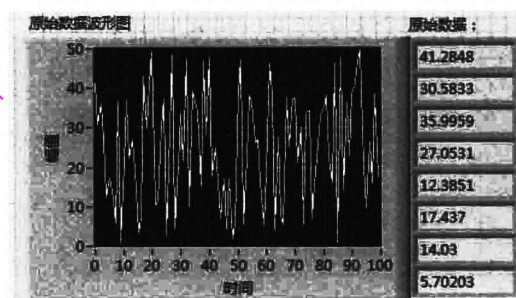


图 4-47 原始数据波形图

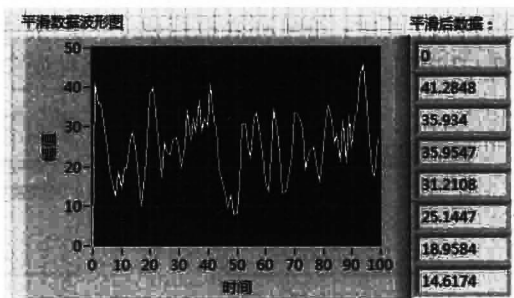


图 4-48 平滑数据波形图

使用移位寄存器实现数字平滑滤波的操作步骤如下。

- ① 新建一个 VI，并加以命名保存。
- ② 在前面板放置两个波形图控件和两个数组控件，再向两个数组控件中分别添加一个数值显示控件以显示数组元素值。
- ③ 按照用途，对波形图和数组的标签加以更改，以便说明作用。
- ④ 为了模拟采集的数据，在后面板添加一个 For 循环，并将循环次数 N 设置为 100。
- ⑤ 在循环内部添加一个随机数函数和一个乘法函数，通过放大产生的模拟数据在 0~50 之间。



- ⑥ 在循环控件上添加一个移位寄存器，并在控件左侧单击鼠标右键，由快捷菜单中选取“添加元素”来增加缓冲器的容量。
- ⑦ 将模拟数据的输出连向移位寄存器的右端子、原始数据波形图及原始数据数组。若此时运行程序，则可看到原始数据和波形。
- ⑧ 将移位寄存器的左端子连在一起，并赋初值零。
- ⑨ 按图 4-49 完成全部程序框图。

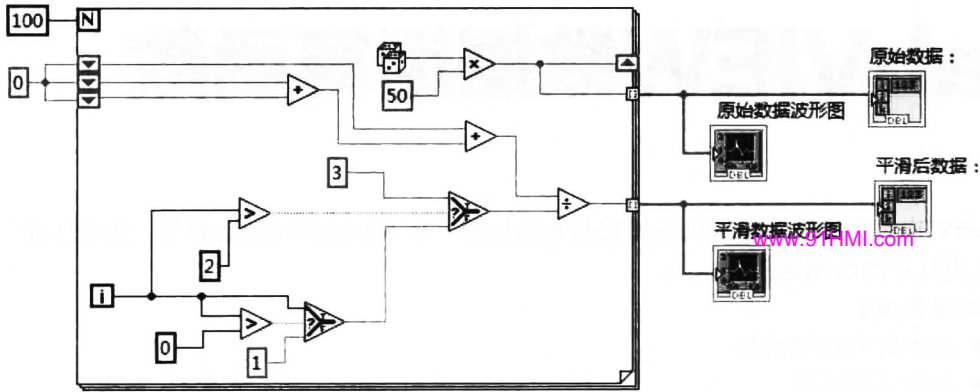


图 4-49 平滑处理程序框图

经程序运行，可发现它具有设计精巧的特点，通常的平滑处理只是进行简单的算术平均，在移位寄存器未被采集数据完全填充前是不合理的，而该程序却是按照采集数据的个数进行平均处理，从程序运行的前几拍即可起到平滑作用。

在数字平滑处理中，缓冲器的容量越大，平滑效果越明显，但有时会错过关键数据的捕捉使之淹没在平滑中，因此对缓冲器的容量要合理进行设置。



程序结构是组织程序的有效工具，没有它的存在，程序就成了一个流水账或放映机。



第 5 章

LabVIEW数据运算函数

LabVIEW 提供的基本数据运算包括数值运算、布尔运算和比较运算等。在此基础上还可构建更为复杂的数据运算。

【本章导读】

- ◆ 基本算术运算函数
- ◆ 布尔运算函数
- ◆ 关系运算函数和比较函数等

5.1 基本算术运算函数

5.1.1 基本运算函数

基本运算函数是所有函数中的最基本运算，涉及的函数包括加、减、乘、除等，详细内容如表 5-1 所示。

表 5-1 基本运算函数

函数名称	图标	函数功能	使用示例
加		对两个输入值求和	设 $x=5$, $y=6$, 则结果为 11
减		对两个输入值求差	设 $x=9$, $y=3$, 则结果为 6
乘		对两个输入值求积	设 $x=5$, $y=6$, 则结果为 30
除		对两个输入值求商	设 $x=9$, $y=3$, 则结果为 3
商与余数		对两个输入值求商和余数	设 $x=11$, $y=6$, 则商为 1, 余数为 5
加 1		对输入值加 1	设 $x=5$, 则结果为 6
减 1		对输入值加 1	设 $x=6$, 则结果为 5
数组元素相加		对数组中的所以元素求和	设数组元素为 1、2、3、4, 则输出为 10



续表

函数名称	图标	函数功能	使用示例
数组元素相乘		对数组中的所有元素求积	设数组元素为 1、2、3、4，则输出为 24
复合运算		对多个输入值求和、求积	设输入 3、6 和 2，则积为 36
绝对值		取绝对值	设 $x=-5$ ，则结果为 5
最近数取整		对输入值取整获得最接近的整数	设 $x=5.6$ ，则结果出为 6
向下取整		对输入取下一级临近值	设 $x=5.6$ ，则结果为 5
向上取整		对输入取上一级临近值	设 $x=5.6$ ，则结果为 6
按 2 的幂缩放		将输入值与 2^n 相乘	设 $x=2$ ， $n=3$ ，则输出为 16
平方根		求输入值的平方根	设 $x=16$ ，则结果为 4
平方		求输入值的平方	设 $x=4$ ，则输出为 16
取负数		对输入值取反	设输入 5，则结果为 -5
倒数		对输入值求倒数	设 $x=5$ ，则结果为 0.2
符号		求输入值得符号，正值为 1，零为 0，负值为 -1	设 $x=-5$ ，则结果为 -1
随机数 (0~1)		产生一个 0~1 间的随机数	无需输入，可随机输出 0~1 间的数值

以复合运算为例，使用的方法如下。

- ① 新建一个 VI，并在前面板放置 3 个数值输入控件和 1 个数值显示控件。
- ② 在后面板，选取“函数→编程→数值→复合运算”，然后在空白位置单击。
- ③ 在默认状态，复合运算图标只要 2 个输入端，为了扩展输入端，将光标移到复合运算图标上，此时会出现调节柄，向下拖动则可增加输入端。
- ④ 将复合运算图标的输入端分别与输入图标连接，并将其输出端与显示图标相连。
- ⑤ 为实现连乘运算，在复合运算图标上单击鼠标右键，从快捷菜单中选取“更改模式→乘”。完成后的前面板和程序框图如图 5-1 所示。

运行程序并设置输入值，则可观察到相应的结果。

数组元素相加与数组元素相乘函数的使用方法如下。

- ① 新建一个 VI，并在前面板放置 2 个数值显示控件和一个数值输入控件。
- ② 选取“控件→新式→数组、矩阵与簇→数组”，并在空白位置单击鼠标予以放置。
- ③ 将数值输入控件拖入到数组控件内部，用以输入数据。为了增添数组的元素个数，将鼠标移动数组控件的右下角，当出现图 5-2 所示的网状光标时，将鼠标向右拖动可增添数



组元素的数量。

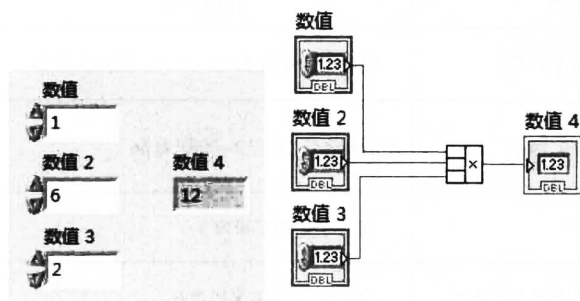


图 5-1 复合运算使用



图 5-2 增添数组元素时的光标形式

④ 在后面板，选取“函数→编程→数值”下的“数组元素相加”和“数组元素相乘”，并整理成图 5-3 所示的形式。

运行该程序，则可验证函数的功能，其运行效果如图 5-4 所示。

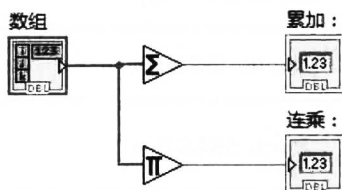


图 5-3 数组元素函数的使用

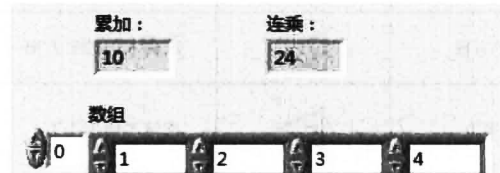


图 5-4 数组元素函数的验证

5.1.2 表达式节点

表达式节点用于计算含有单个变量的表达式，可供表达式节点使用的内置函数有：三角函数、反三角函数、对数、指数等，具体内容如表 5-2 所示。

表 5-2 关系运算函数

函数名称	函数功能	使用示例
Abs	计算绝对值函数	设 $x=-5$ ，则 $\text{abs}(x)=5$
acos	计算反余弦函数，以弧度为单位	设 $x=0.5$ ，则 $\text{acos}(x)=1.0472$
acosh	计算反双曲余弦函数	设 $x=5$ ，则 $\text{acosh}(x)=2.29243$
asin	计算反正弦函数，以弧度为单位	设 $x=0.5$ ，则 $\text{asin}(x)=0.523599$
asinh	计算反双曲正弦函数	设 $x=5$ ，则 $\text{asinh}(x)=2.31244$
atan	计算反正切函数，以弧度为单位	设 $x=1$ ，则 $\text{atan}(x)=0.785398$
atanh	计算反双曲正切函数	设 $x=0.5$ ，则 $\text{atanh}(x)=0.549306$
ceil	向上取整	设 $x=6.4$ ，则 $\text{ceil}(x)=7$
cos	计算余弦函数， x 以弧度为单位	设 $x=1.0472$ ，则 $\text{cos}(x)=0.5$
cosh	计算双曲余弦函数	设 $x=1.0472$ ，则 $\text{cosh}(x)=1.60029$
cot	计算余切函数， x 以弧度为单位	设 $x=0.785$ ，则 $\text{cot}(x)=1.0008$



续表

函数名称	函数功能	使用示例
csc	计算余割函数, x 以弧度为单位	设 $x=0.5$, 则 $\text{csc}(x)=2.08583$
exp	计算 e 的 x 次幂	设 $x=1$, 则 $\text{exp}(x)=2.71828$
expm1	计算 $(e^x) - 1$	设 $x=1$, 则 $\text{expm1}(x)=1.71828$
floor	向下取整	设 $x=6.4$, 则 $\text{floor}(x)=6$
getexp	返回 x 的指数	设 $x=8.71828$, 则 $\text{getexp}(x)=3$
getman	返回 x 的尾数	设 $x=0.5$, 则 $\text{getman}(x)=1$
int	四舍五入求最近的整数	设 $x=6.4$, 则 $\text{int}(x)=6$
intrz	求 x 至 0 之间的最近的整数	设 $x=1.4$, 则 $\text{intrz}(x)=1$
ln	计算以 e 为底的自然对数	设 $x=2.71828$, 则 $\text{ln}(x)=1$ www.91HMI.com
lnp1	计算 $(x+1)$ 的自然对数	设 $x=1$, 则 $\text{lnp1}(x)=1.09861$
log	计算以 10 为底的对数	设 $x=10$, 则 $\text{log}(x)=1$
log2	计算以 2 为底的对数	设 $x=2$, 则 $\text{log2}(x)=1$
rand	产生随机数	如 $\text{rand}()=0.776158$
sec	计算正割函数, x 以弧度为单位	设 $x=1$, 则 $\text{sec}(x)=1.85082$
sign	求符号函数	设 $x=6$, 则 $\text{sign}(x)=1$
sin	计算正弦函数, x 以弧度为单位	设 $x=0.523599$, 则 $\text{sin}(x)=0.5$ www.PLCworld.cn
sinc	计算 $\text{sin}(x)/x$, x 以弧度为单位	设 $x=0.5$, 则 $\text{sinc}(x)=0.958851$
sinh	计算双曲正弦函数	设 $x=2.31244$, 则 $\text{sinh}(x)=5$
sqrt	计算平方根	设 $x=4$, 则 $\text{sqrt}(x)=2$
tan	计算正切函数, x 以弧度为单位	设 $x=1$, 则 $\text{tan}(x)=1.55741$
tanh	计算双曲正切函数	设 $x=0.549306$, 则 $\text{tanh}(x)=0.5$

以正弦函数为例, 使用方法如下。

- ① 新建一个 VI, 并在前面板放置一个数值输入控件和一个数值显示控件。
- ② 切换到后面板, 选取“函数→编程→数值→表达式节点”, 并在空白位置单击鼠标予以放置。
- ③ 将表达式节点图标的输入端与输出端分别与输入图标和显示图标相连, 完成后的程序框图和界面如图 5-5 所示。

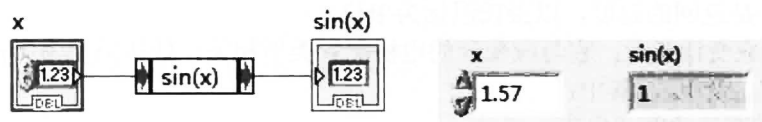


图 5-5 表达式节点验证

运行该程序, 改变 x 值可得到相应的正弦值。注意此时 x 值是以弧度为单位, 由于 90 度角的弧度等于 $90 \times 3.1415926 / 180 = 1.5707963$, 而 $\text{sin}(1.5707963) = 1$, 说明验证正确。



5.1.3 数值缩放运算

数值缩放功能主要包括转换 RTD 读数、转换应变计、转换热敏电阻、转换热电偶、弧度转换至角度、角度转换至弧度、RPM 转换至弧度/秒、弧度/秒转换至 RPM，具体内容如表 5-3 所示。

表 5-3 基本运算函数

函数名称	图标	函数功能	说明
转换 RTD 读数		将 RTD 读取的电压值转换为摄氏温度值	函数功能依照公式： $R_t = R_0[1 + A_t + B_t^2 t C(t-100)^2]$
转换应变计读数		将应变计的电压读数转换为应变单位	函数功能依照公式： $V_r = (V_{sg} - V_{init})/V_{ex}$
转换热敏电阻读数		将热敏电压转换为温度	电压激励： $R_t = R_0 \left(\frac{V_0}{V_{REF} - V_0} \right)$ 电流激励： $R_t = \frac{V_0}{I_{EX}}$
转换热电偶读数		将热电偶读取的电压值转换为温度值	适用的热电偶类型可以是 B、E、J、K、R、S、T 或 N 型
弧度转换至角度		将弧度转换至角度	设 $x=3.14159$ ，则输出为 180
角度转换至弧度		将角度转换至弧度	设 $x=180$ ，则输出为 3.14159
RPM 转换至弧度/秒		将 RPM 转换至弧度/秒	设 $x=30$ ，则输出为 3.14159
弧度/秒转换至 RPM		将弧度/秒转换至 RPM	设 $x=3.14159$ ，则输出为 30

对于转换 RTD 读数函数，其涉及的参数有 R_0 、RTDvolts、 I_{ex} 、 A 、 B 及 RTDtemp，其中：

R_0 是 0°C 时的 RTD 电阻，默认值为 100Ω ；

RTDvolts 是从 RTD 读取的电压；

I_{ex} 是 RTD 的激励电流，默认值为 0.15mA ；

A 是 RTD 的 Callendar Van-Dusen 方程的系数，默认值为欧洲曲线（DIN 43760 标准）的系数；

B 是 RTD 的 Callendar Van-Dusen 方程的系数，默认值为欧洲曲线（DIN 43760 标准）的系数；

RTDtemp 是返回的温度，以摄氏温标为单位。

对于转换应变计函数，它与应变计的电桥配置类型相关，默认类型为半桥 II。所涉及的参数是 V_{sg} 、 V_{init} 及 V_{ex} ，其中：

V_{sg} 是从应变计读取的电压；

V_{ex} 是激励电压，默认值为 3.333V ；

V_{init} 是应变计的无应变电压。

$$V_r = (V_{sg} - V_{init})/V_{ex}$$

而电桥配置又与 V_r 、 R_g 、 v 及 GF 相关，并可求出应变值。相关参数的含义如下：



R_e 是应变计的额定电阻值，默认值为 120Ω ；

ν 是泊松比，仅在某些电桥配置中使用；

GF 是应变计的应变计因子。

以弧度转换至角度为例，其使用方法如下：

- ① 新建一个 VI，并在前面板放置一个数值输入控件和一个数值显示控件。
- ② 切换到后面板，选取“函数→编程→数值→缩放→弧度转换至角度”，并在空白位置单击鼠标予以放置。
- ③ 将角度转换至弧度图标的输入端、输出端分别与输入图标和显示图标相连，完成后的程序框图和界面如图 5-6 所示。

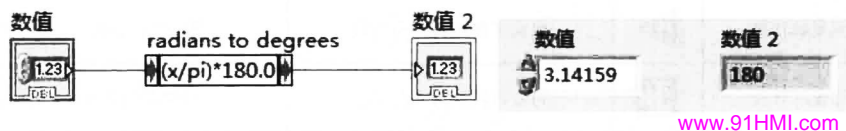


图 5-6 表达式节点验证

运行该程序，改变 x 值可得到相应的角度值。注意此时 x 值以弧度为单位，当输入 3.1415926 时，角度值为 180 度，说明验证正确。

以转换热电偶读数为例，其使用方法如下：

- ① 新建一个 VI，并在前面板放置一个波形图表控件。
- ② 切换到后面板，选取“函数→编程→数值→缩放→转换热电偶读数”，并在空白位置单击鼠标予以放置。
- ③ 将转换热电偶读数的输出端与波形图表的输入端相连，然后按图 5-7 所示完成程序框图。

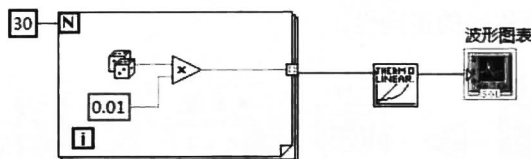


图 5-7 转换热电偶读数程序框图

运行该程序，则可得到图 5-8 所示的波形图。

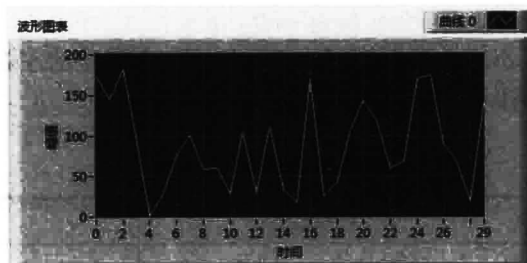


图 5-8 转换热电偶读数波形图

5.1.4 复数运算函数

复数运算可实现各种复数的运算，它包括求共轭复数、将幅角和幅值转换为复数、将复



数转换为幅角和幅值、将两个实数组成复数等，具体内容如表 5-4 所示。

表 5-4 复数运算函数

函数名称	图标	函数功能	说明
复共轭		求 $x + iy$ 的复共轭	设输入为 $4 + 3i$ ，则输出为 $4 - 3i$
极坐标至复数转换		将极坐标转换为复数	设 $r=1$ ， $\theta=1$ ，则输出为 $0.540302 + 0.841471i$
复数至极坐标转换		将复数转换为极坐标	设 $0.540302 + 0.841471i$ ，则 $r=1$ ， $\theta=1$
实部虚部至复数转换		将实部虚部转换为复数	设 $x=2$ ， $y=1$ ，则输出为 $2 + 1i$
复数至实部虚部转换		将复数转换为实部虚部	设输入为 $2 + 1i$ ，则 $x=2$ ， $y=1$ www.91HML.com
实部虚部至极坐标转换		将实部虚部转换为极坐标	设 $x=2$ ， $y=1$ ，则 $r=1.41421$ ， $\theta=0.785398$
极坐标至实部虚部转换		将极坐标转换为实部虚部	设 $r=1.41421$ ， $\theta=0.785398$ ，则 $x=2$ ， $y=1$

以复共轭函数为例，其使用方法如下。

① 新建一个 VI，切换到后面板，选取“函数→编程→数值→缩放→转换 RTD 读数”，并在空白位置单击鼠标予以放置。

② 为复共轭函数添加一个数值输入控件和一个数值显示控件，则程序框图和界面如图 5-9 所示。

运行该程序，可验证程序的正确性。

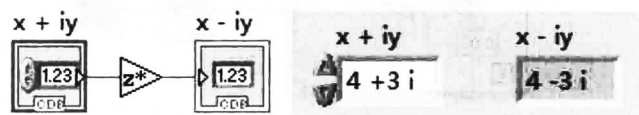


图 5-9 复共轭函数验证

5.1.5 数据操作

数据操作功能可实现不同类型间的相互转换、带进位的左右移动、逻辑位移、循环位移、拆分数字、整数拼接等，常用的内容如表 5-5 所示。

表 5-5 数据操作函数

函数名称	图标	函数功能	说明
尾数与指数		依照数字=尾数*2 ^{指数} ，求输入数值的尾数和指数	设输入值为 9，则指数值为 3、尾数值为 1.125
带进位的左移		将输入值向左移动一位	设输入值为 1、进位位为 1，则输出值为 3、最高有效位进位为 0
带进位的右移		将输入值向右移动一位	设输入值为 1、进位位为 0，则输出值为 0、最低有效位进位为 1



续表

函数名称	图标	函数功能	说明
逻辑移位		将输入值 x 移动 y 位数	设 $x=1$, $y=3$, 则输出值为 8
循环移位		将输入值 x 移动 y 位数	设 $x=8$, $y=-1$, 则输出值为 1
拆分数字		对基于字节或字长的数字进行拆分	设输入值为 65536, 则 $hi=1$, $lo=0$
整数拼接		对基于字节或字长的数字进行拼接	设 $hi=1$, $lo=1$, 则输出值为 65537
交换字节		对字的高 8 位和低 8 位进行数据交换	设输入值为 1, 则输出值为 256
交换字		对长整型数的高 16 位和低 16 位数据进行交换	设输入值为 1, 则输出值为 65536

以交换字节函数为例, 其使用方法如下。

www.91HMI.com

① 新建一个 VI, 切换到后面板, 选取“函数→编程→数值→数据操作→交换字节”, 并在空白位置单击鼠标予以放置。

② 为交换字节函数添加一个数值输入控件和一个数值显示控件, 则程序框图和界面如图 5-10 所示。

运行该程序, 输入数值 1, 由于低 8 位为 1, 高 8 位为 0, 交换后低 8 位为 0, 高 8 位为 1 (等于 256), 故显示结果为 256。



图 5-10 交换字节函数验证

5.2 布尔运算函数

5.2.1 布尔运算函数

在 LabVIEW 中, 布尔运算主要完成布尔型数据的逻辑操作, 因此它也被称为逻辑运算, 布尔运算涉及与、或、非及异或等函数, 详细内容如表 5-6 所示。

表 5-6 布尔运算函数

函数名称	图标	函数功能	使用示例
与		两个布尔量相与	设 $x=T$ (真), $y=F$ (假), 则结果为 F (假)
或		两个布尔量相或	设 $x=T$, $y=F$, 则结果为 T
异或		两个布尔量相异则或	设 $x=T$, $y=F$, 则结果为 T
非		对输入值取反	设 $x=T$, 则结果为 F



续表

函数名称	图标	函数功能	使用示例
与非		两个布尔量相与非	设 $x=T$, $y=T$, 则结果为 F
或非		两个布尔量相或非	设 $x=F$, $y=F$, 则结果为 T
同或		两个布尔量相同则或	设 $x=F$, $y=F$, 则结果为 T
蕴含		判别 x 是否蕴含 y , 它是 y 和取反后的 x 的逻辑或	设 $x=T$, $y=F$, 则结果为 F
数组元素与操作		对数组中的所以元素求和	设数组元素为 1、2、3、4, 则输出为 10
数组元素或操作		对数组中的所以元素求和	设数组元素为 1、2、3、4, 则输出为 10
数值至布尔数组转换		使整数或定点数转换为布尔数组	www.91HMI.com 设输入为 6, 则布尔数组为 011
布尔数组至数值转换		使用布尔数组作为数字的二进制表示	设布尔数组为 101, 则输出为 5
布尔值至 (0, 1) 转换		将布尔值转换为 0 或 1	设 $x=T$, 则结果为 1
复合运算		对输入的布尔值进行与、或、异或等布尔复合运算	设输入为 T、F、T, 则执行与运算后结果为 F

数组元素与函数的使用方法是:

- ① 新建一个 VI, 并进入后面板。
- ② 选取“函数→编程→布尔→数组元素与”, 并在空白位置单击鼠标予以放置。
- ③ 在布尔数组输入端单击鼠标右键, 由控件菜单选取“创建→输入控件”。再在布尔数组输出端单击鼠标右键, 由控件菜单选取“创建→显示控件”。
- ④ 切换到前面板, 将光标移到布尔数组上, 当控件四周出现调节柄后, 向右拖动鼠标, 直至出现 3 个开关按钮, 此时界面和程序框图则如图 5-11 所示。



图 5-11 数组元素与函数的验证

运行该程序, 通过设置布尔数组中的开关则可观察到函数的功能, 当所有开关处于点亮状态时, 逻辑与的指示灯才点亮。否则, 指示灯将处于熄灭状态。

布尔数组至数值函数的使用方法如下。

- ① 新建一个 VI, 并进入后面板。
- ② 选取“函数→编程→布尔→布尔至数值”, 并在空白位置单击鼠标予以放置。
- ③ 在布尔数组输入端单击鼠标右键, 由控件菜单选取“创建→输入控件”。再在布尔数组输出端单击鼠标右键, 由快捷菜单选取“创建→显示控件”。
- ④ 切换到前面板, 将光标移到布尔数组上, 当控件四周出现调节柄后, 向右拖动鼠标, 直至出现 4 个开关按钮, 此时界面和程序框图则如图 5-12 所示。

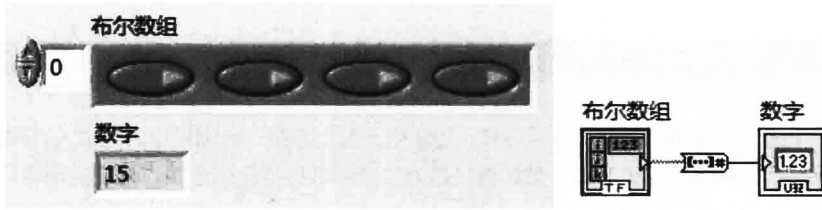


图 5-12 布尔数组至数值的转换

运行该程序，通过设置布尔数组中的开关则可观察到函数的功能，在所有开关处于点亮状态时，则数字输出控件显示 15，即二进制 1111 对于十进制的 15。

采用相同的方法，可验证数值至布尔数组的转换功能，其界面和程序框图则如图 5-13 所示。

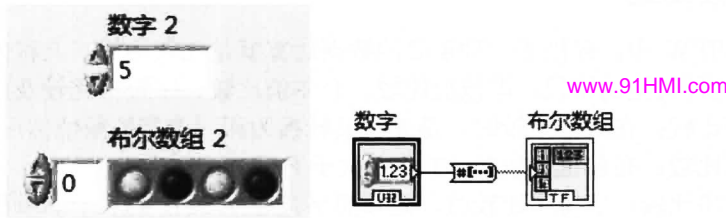


图 5-13 数值至布尔函数的验证

5.2.2 位运算

布尔运算不仅能够进行逻辑运算，也能够对整数进行位运算。为验证位运算功能，可通过下面的示例加以说明，其制作过程如下。

- ① 新建一个 VI，并在前面板放置 2 个数值输入控件和 4 个显示控件。
 - ② 对控件的标签予以修改，形成图 5-14 的形式。
 - ③ 选取标签为“数值 X 的二进制”的显示控件，单击鼠标右键，由快捷菜单将其表示法改为“U16”，再通过快捷菜单将其显示格式改为“二进制”。
 - ④ 将标签为“数值 Y 的二进制”和“结果的二进制”的显示控件也改为二进制显示格式。
 - ⑤ 选取“函数→编程→布尔→与”，并在空白位置单击鼠标予以放置。
 - ⑥ 按照图 5-15 所示完成程序框图连线。
- 运行该程序，则验证位运算的功能。

数值 X	数值 X 的二进制	结果
5	101	4
数值 Y	数值 Y 的二进制	结果的二进制
6	110	100

图 5-14 位运算验证界面

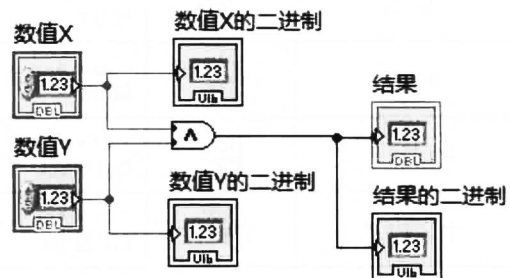


图 5-15 位运算程序框图



提示

布尔运算不仅适用于布尔型数据、数组或簇，还适用于整型数据、数组和簇。在输入为整型数据时，是先将数据转换为相应的二进制数，然后再按位对二进制数进行逻辑运算，最后的结果再转换为十进制数输出。

5.3 关系运算函数和比较函数

5.3.1 比较模式

在 LabVIEW 中，提供了一种重要的数据运算就是比较运算，又称为关系运算。该运算可实现以下几种类型的比较，即数值比较、布尔值比较、字符串比较及簇的比较。

- ① 数值比较：在数值比较时，是先将其转换为相同类型的数值然后再进行比较。
- ② 布尔比较：布尔值比较时，T 值要大于 F 值。
- ③ 字符串比较：字符串比较时，是按照字符在 ASCII 码表中的编码值进行逐个字符的比较。
- ④ 簇的比较：簇的比较类似于字符串比较，是从簇的第 0 个元素开始比较，直至有一个元素不相等为止。在簇比较时，簇的元素个数必须相同，且元素的数据类型和顺序也必须相同。

www.PLCworld.cn

5.3.2 基本关系运算函数

基本关系运算函数包括等于、不等于、大于、小于、大于等于及小于等于等，具体内容如表 5-7 所示。

表 5-7 基本关系运算函数

函数名称	图标	函数功能	使用示例
等于否		若 x 等于 y ，则输出 T。否则，输出 F	设 $x=5$ ， $y=5$ ，则结果为 T
不等于否		若 x 不等于 y ，则输出 T。否则，输出 F	设 $x=9$ ， $y=6$ ，则结果为 T
大于否		若 x 大等于 y ，则输出 T。否则，输出 F	设 $x=5$ ， $y=6$ ，则结果为 F
小于否		若 x 小于 y ，则输出 T。否则，输出 F	设 $x=5$ ， $y=6$ ，则结果为 T
大于等于否		若 x 大于等于 y ，则输出 T。否则，输出 F	设 $x=9$ ， $y=6$ ，则结果为 T
小于等于否		若 x 小于等于 y ，则输出 T。否则，输出 F	设 $x=6$ ， $y=6$ ，则结果为 T

以布尔值比较为例，其具体操作如下。

- ① 新建一个 VI，并在前面板放置一个圆形指示灯。



- ② 切换到后面板, 选取“函数→编程→比较→大于?”, 并在空白位置单击鼠标予以放置。
- ③ 在比较图标的 X 输入端添加一个真常量, 在 Y 输入端添加一个假常量。
- ④ 将比较图标的输出端与圆形指示灯的输入端相连, 完成后的程序框图如图 5-16 所示。

运行该程序, 则可观察到圆形指示灯是处于点亮状态, 说明真常量比假常量大。

以字符串比较为例, 其具体操作如下:

- ① 新建一个 VI, 并在前面板放置一个圆形指示灯。
- ② 切换到后面板, 选取“函数→编程→比较→大于?”, 并在空白位置单击鼠标予以放置。
- ③ 在比较图标的 X 输入端添加一个字符串常量“abcd”, 在 Y 输入端添加一个字符串常量“aacd”。
- ④ 将比较图标的输出端与圆形指示灯的输入端相连, 完成后的程序框图如图 5-17 所示。

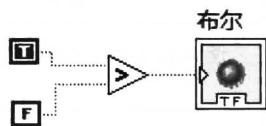


图 5-16 布尔值比较程序框图

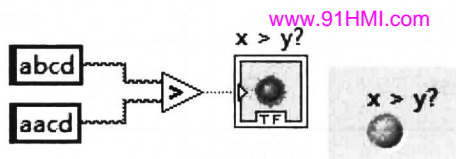


图 5-17 字符串比较程序框图

运行该程序, 则可观察到字符串常量比较的结果。由于字符 b 的 ASCII 码为 98, 而字符 a 的 ASCII 码为 97, 所以字符串“abcd”比字符串“aacd”要大, 故指示灯被点亮。

5.3.3 “比较 0”关系运算函数

在关系比较中, 数值零是一个比较特殊的值, 为了必要使用在 LabVIEW 中专门设置了“比较 0”关系运算函数, 主要包括等于 0、不等于 0、大于 0、小于 0、大于等于 0 及小于等于 0 函数, 具体内容如表 5-8 所示。

表 5-8 “比较 0”关系运算函数

函数名称	图标	函数功能	使用示例
等于零否		若 x 等于零, 则输出 T。否则, 输出 F	设输入 2, 则结果为 F
不等于零否		若 x 不等于零, 则输出 T。否则, 输出 F	设输入 6, 则结果为 T
大于零否		若 x 大于零, 则输出 T。否则, 输出 F	设输入 6, 则结果为 T
小于零否		若 x 小于零, 则输出 T。否则, 输出 F	设输入 6, 则结果为 F
大于等于零否		若 x 大于等于零, 则输出 T。否则, 输出 F	设输入 6, 则结果为 T
小于等于零否		若 x 小于等于零, 则输出 T。否则, 输出 F	设输入 6, 则结果为 F

以大于等于 0 比较为例, 其具体操作如下。

- ① 新建一个 VI, 并在前面板放置一个数值输入控件和一个圆形指示灯。



② 切换到后面板，选取“函数→编程→比较→大于等于 0?”，并在空白位置单击鼠标予以放置。

③ 在比较图标的 x 输入端与输入图标相连，将比较图标的输出端与圆形指示灯相连，完成后的程序框图如图 5-18 所示。

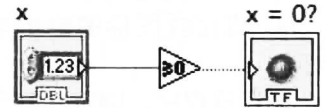


图 5-18 大于等于 0 比较程序框图

运行该程序，则可观察到当数值大于等于 0 时，圆形指示灯处于点亮状态；否则，指示灯处于熄灭状态。

5.3.4 复杂关系运算函数

除了前面两类关系运算外，还有其他一些关系运算函数，其具体内容如表 5-9 所示。

表 5-9 复杂关系运算函数

函数名称	图标	函数功能	www.91HMI.co 使用示例
选择		依据 s 的值，返回连线至 t 输入或 f 输入的值	设 $s=T$, $t=100$, $f=500$ ，则输出值为 100
最大值与最小值		判别两个数的最大值和最小值	设 $x=100$, $y=60$ ，则 \min 端输出为 60， \max 端输出为 100
判定范围并强制转换		判别一个数是否在给定数值区域内	设 $x=50$ ，上限=60，下限=42，则范围内输出端为 T，已强制转换端输出为 50
非法数字/路径/引用句柄		判别输入量是否为非法数值/路径/引用句柄，若非法则输出 T，否则输出 F	设输入端为 5，则输出端为 F
空数组否		判别输入是否为空数组，若为空数组则输出 T，否则输出 F	设输入端为空数组，则输出端为 T
空字符串/路径		判别输入是否为空字符串或空路径，若为空字符串或空路径则输出 T，否则输出 F	设输入端为空字符串常量，则输出端为 T
十进制数否		判别输入的字符串是否为十进制数，是则为 T，否则为 F	设输入值为 A56，则输出端为 F
十六进制数否		判别输入的字符串是否为十六进制数，是则为 T，否则为 F	设输入值为 A56，则输出端为 T
八进制数否		判别输入的字符串是否为八进制数，是则为 T，否则为 F	设输入值为 86，则输出端为 F
可打印否		判别输入的字符串中首个字符是否为可打印的 ASCII 码，是则为 T，否则为 F	设输入为回车键常量，则输出端为 F
空白否		判别输入的字符串中首个字符是否为空格字符，是则为 T，否则为 F	设输入空格和字符 ab，则输出端为 T
字符类否		对输入的字符串进行分类，按字符串的首个字符，输出其类别号	设输入值为 86，则类别号为 3
定点溢出否		判别定点数是否包含溢出状态且是否为溢出运算的结果	设输入值为 0.5，则输出为 F

以空数组函数为例，其使用方法如下。

① 新建一个 VI，并在前面板放置一个圆形指示灯。



② 切换到后面板，选取“函数→编程→比较→空数组？”，并在空白位置单击鼠标予以放置。

③ 将比较图标的输出端与圆形指示灯相连，再为比较图标的输入端创建一个输入控件，完成后的程序框图如图 5-19 所示。

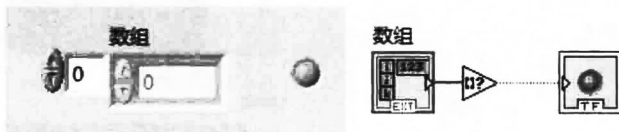


图 5-19 空数组函数验证

运行该程序，则可观察到圆形指示灯处于点亮状态，修改数组元素值，则指示灯处于熄灭状态。

判定范围并强制转换函数的使用方法如下。

① 新建一个 VI，并在前面板放置一个圆形指示灯、一个数值输入控件和一个数值显示控件。

② 切换到后面板，选取“函数→编程→比较→判定范围并强制转换”，并在空白位置单击鼠标予以放置。

③ 比较图标的 x 输入端与输入图标相连，范围内输出端与圆形指示灯相连，已强制值端与显示图标相连。

④ 为比较图标的上限和下限输入端分别添加数值常量 60 和 42，完成后的程序框图如图 5-20 所示。

运行该程序，通过调节输入值可观察到输出的变化情况，其界面如图 5-21 所示。

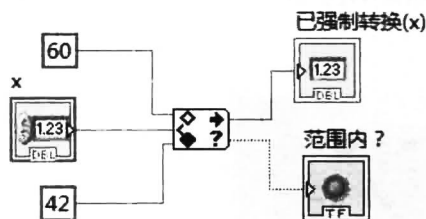


图 5-20 判定范围并强制转换程序框图

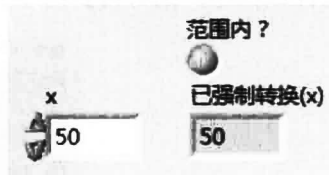


图 5-21 判定范围并强制转换界面

5.3.5 Express 比较函数

Express 比较函数的功能是对输入项进行指定的比较，以决定数值间的等于、大于或小于关系，并可以图表的方式予以显示。

Express 比较函数的使用方法如下。

① 新建一个 VI，然后切换到后面板，选取“函数→编程→比较→比较”，并在空白位置单击鼠标予以放置。

② 在弹出的配置比较对话框中，按图 5-22 所示对比较项加以设置。

③ 单击“确定”，退出配置对话框。

④ 在比较图标的结果输出端单击鼠标右键，由快捷菜单选取“创建→图形显示控件”。

⑤ 为了生成比较的数据以显示比较图形，按图 5-23 所示完成程序框图。

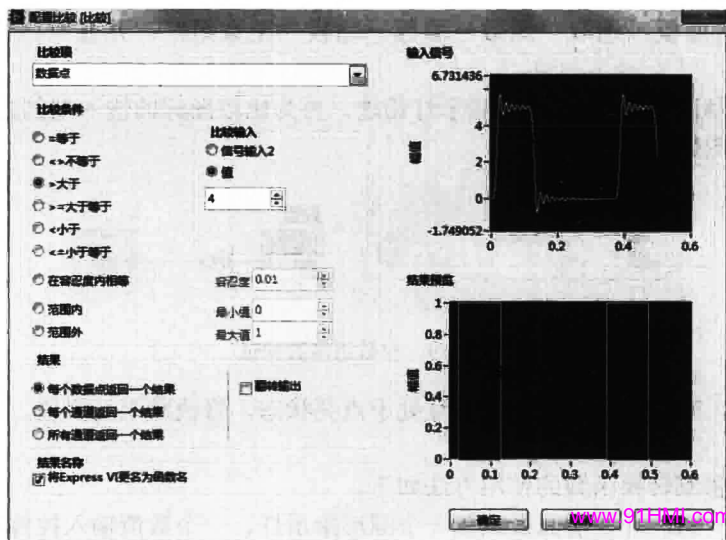


图 5-22 配置比较对话框

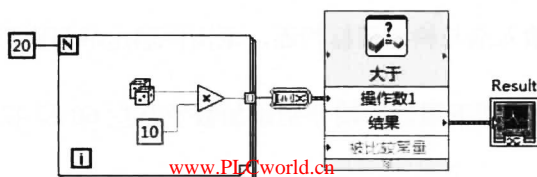


图 5-23 Express 比较函数程序框图

运行该程序，则可得到图 5-24 所示的波形图。

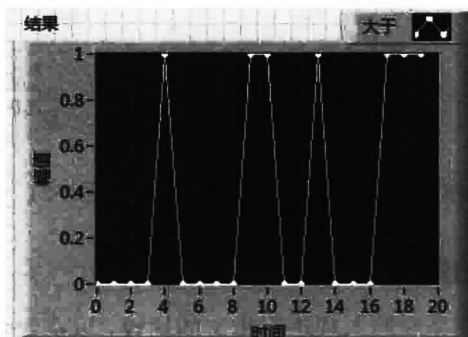


图 5-24 比较判别波形图

5.4 其他函数

5.4.1 数组函数

数组函数的内容十分丰富，主要包括数组大小、索引数组、替换数组子集、数值插入、



删除数组元素、初始化数组、创建数组、数组最大值与最小值等。

选取数组函数的方法是：在后面板，选取“函数→编程→数组”下的相应函数，并在空白位置单击鼠标予以放置。

(1) 数组大小函数

该函数的功能是获取给定数值的各维长度，若输入的是一维数值，则输出该数组的元素个数；若输入的是多维数组，则输出的是一个一维数组，此数组的每个元素对应每一维的元素个数。

图 5-25 是获取一维数组大小的示例，图 5-26 是获取二维数组大小的示例。



图 5-25 获取一维数组大小

www.91HMI.com



图 5-26 获取二维数组大小

www.PLCworld.cn

(2) 索引数组函数

该函数的功能是访问数组中的指定元素，由索引值来指定要访问的数组元素，对于一维数组，得到是某一元素；对于多维数值，则获得数组中的某行或某列。

图 5-27 是索引数组获取一个元素，图 5-28 是索引数组获取一行元素。

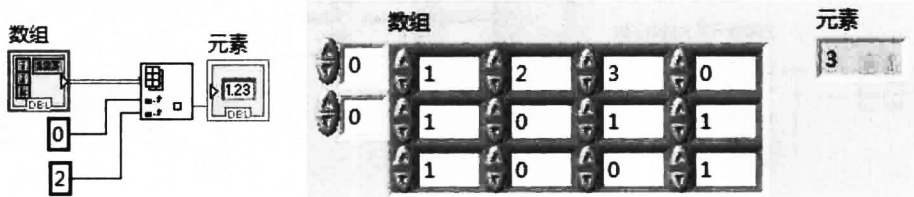


图 5-27 索引数组获取一个元素

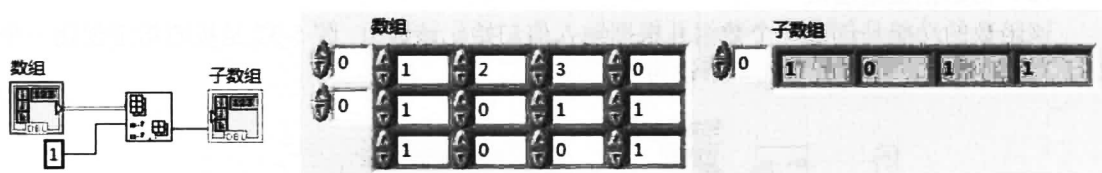


图 5-28 索引数组获取一行元素

(3) 替换数组子集函数

该函数的功能是将原数组中由索引所指定的元素或子数组进行替换。图 5-29 是按索引值



将第 2 行用子数组进行替换。

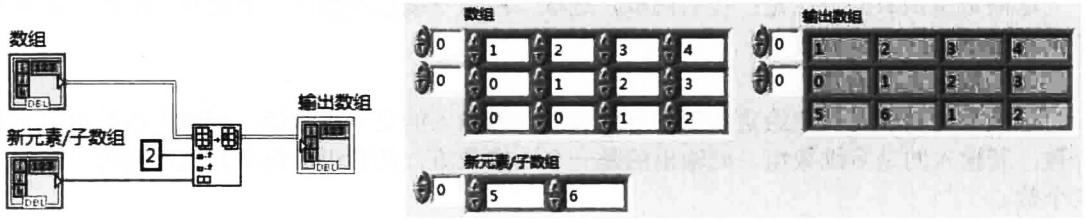


图 5-29 替换数组子集

(4) 数值插入函数

该函数的功能是向数组加入新元素，图 5-30 是按索引值在第 2 行插入子数组。



图 5-30 数值插入处理

(5) 删除数组元素函数

该函数的功能是按索引的指定从数组中删除单个元素或子数组，图 5-31 是在原数组中从第 0 行开始删除一行子数组。

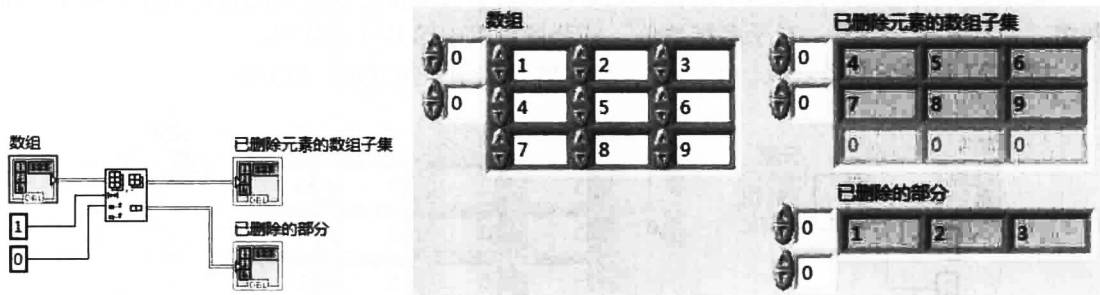


图 5-31 删除数组元素

(6) 初始化数组函数

该函数的功能是创建一个数组并根据输入值初始化该数组，图 5-32 是按照指定创建一个 2 行 3 列的数组，并将所有元素都设为 0。

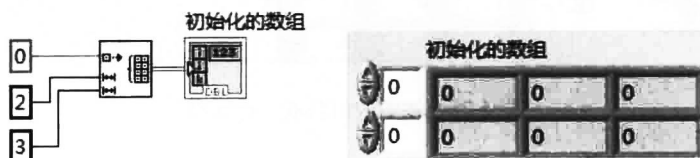


图 5-32 初始化数组



(7) 创建数组函数

该函数的功能是创建一个数组并根据输入值初始化该数组,图 5-33 分别是向数组添加元素和用一维数组来创建 2 行 3 列的数组。

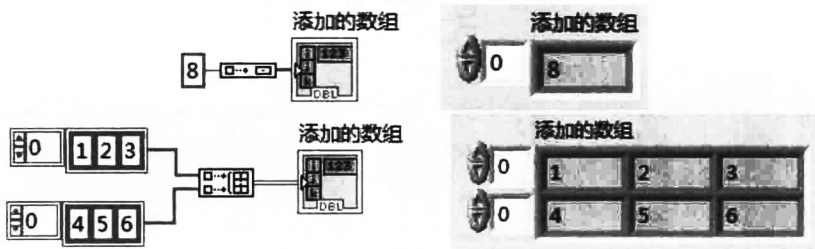


图 5-33 创建数组

(8) 数组子集函数

该函数的功能是在数组中由索引指定来提取指定长度的子数组并予以输出,图 5-34 是对一个 3 行 5 列数组提取由 0 行 1 列开始的 2 行 3 列元素构成一个 2 行 3 列的新数组。

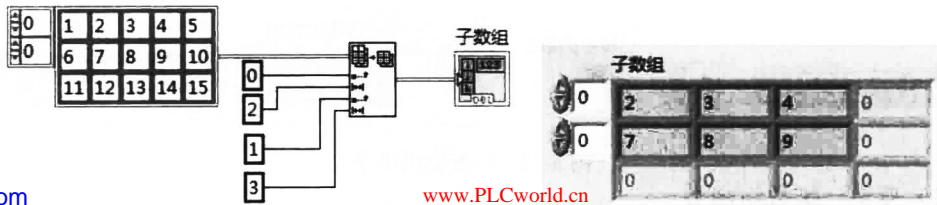


图 5-34 求数组子集

(9) 数组最大值与最小值函数

该函数的功能是获取数组的最大值、最小值及相应的索引值,图 5-35 是对一个 3 行 3 列的二维数组求最大、最小值。

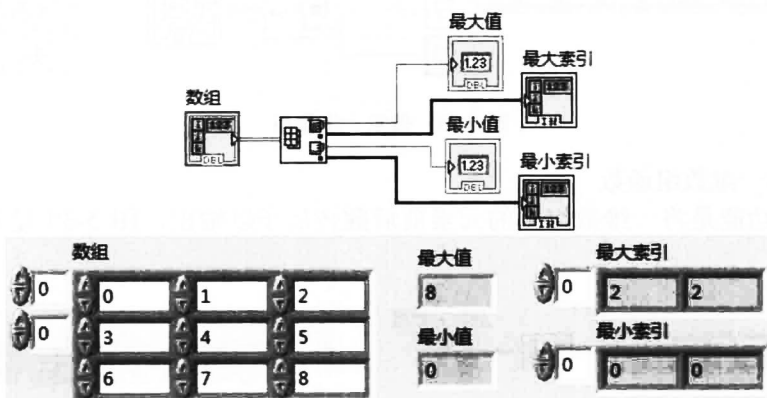


图 5-35 求数组最大值和最小值

(10) 重排数组维数函数

该函数的功能是对多维数组进行重构,并将数组中元素按顺序分派到新构建的数组内。当元素不足时,以 0 填补;若原数组中有未使用的元素则被忽略。图 5-36 是将一个 3 行 3 列



的数组重构成一个 2 行 4 列的数组。

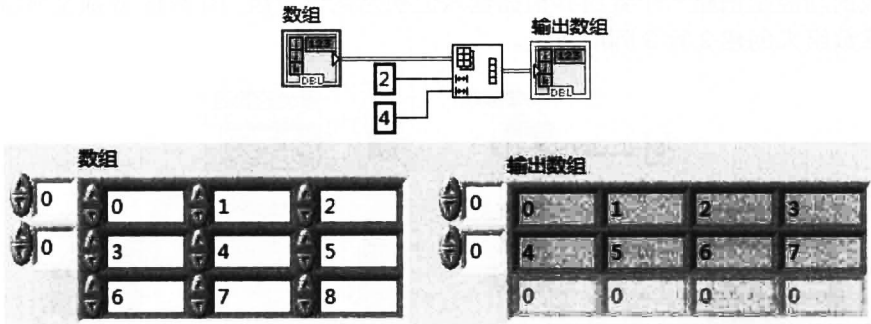


图 5-36 重排数组维数

(11) 一维数组排序函数

该函数的功能是对一维数组按升序进行排序，图 5-37 是对有 6 个元素的一维数组进行排序。

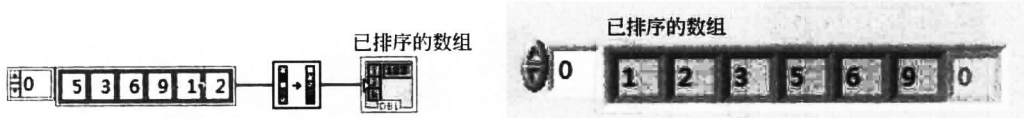


图 5-37 一维数组排序

(12) 搜索一维数组函数

该函数的功能是在一维数组中从索引位置开始搜索指定值的元素并返回结果，若找到该元素，则输出其索引值；否则输出-1。图 5-38 是对 1 个一维数组，从索引值 3 开始搜索元素值为 6 的索引值。

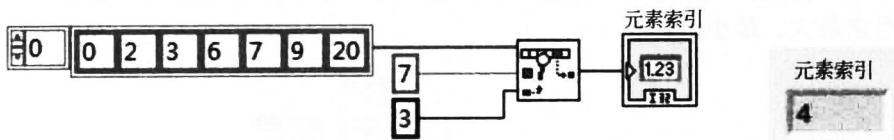


图 5-38 搜索一维数组

(13) 翻转一维数组函数

该函数的功能是将一维数组中的元素前后翻转后予以输出，图 5-39 是对一维数组进行翻转。



图 5-39 反转一维数组

(14) 拆分一维数组函数

该函数的功能是根据给定的索引值把一维数组拆分成两部分，图 5-40 是把一维数组按索引值 3 拆分成两个数组。

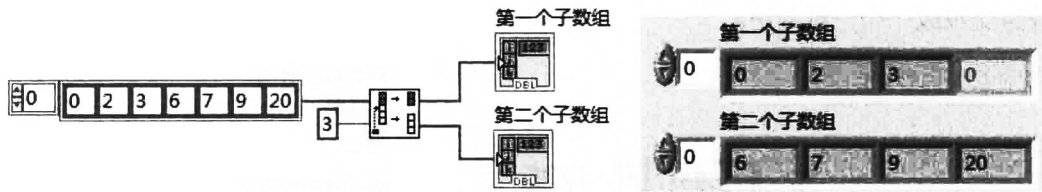


图 5-40 拆分一维数组

(15) 一维数组移位函数

该函数的功能是将一维数组中的元素循环右移若干位并予以输出, 图 5-41 是将一维数组移动 1 位。

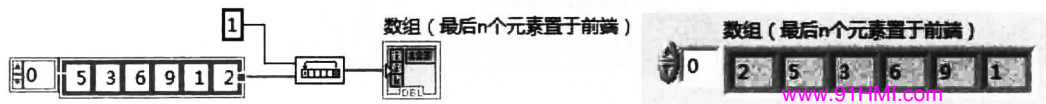


图 5-41 一维数组移位

(16) 一维数组插值函数

该函数的功能是计算一维数组指定位置处的线性插值, 图 5-42 是在一维数组中按参数 1, 在数组中取索引值为 1 的元素 4, 和它的下一个元素是 5, 取平均, 即 $4+(5-4)/2=4.5$, 取整后返回插值 5。

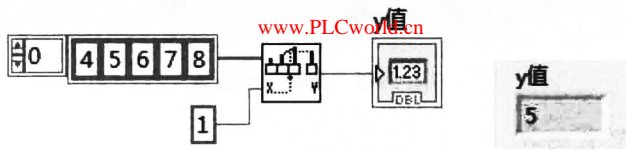


图 5-42 一维数组插值

(17) 以阈值插值一维数组函数

该函数的功能是求一维数组的门限值, 相对于是一维数组插值函数的逆运算。图 5-43 是按照给定的阈值 5, 在数组中找到它的索引值。

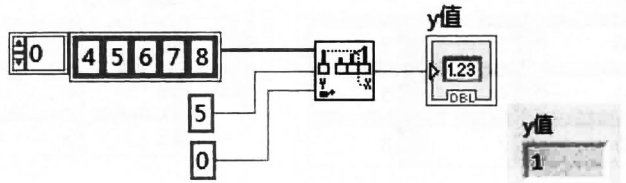


图 5-43 以阈值插值一维数组

(18) 抽取一维数组函数

该函数的功能是一维数组插值函数的逆操作, 是将一维数组中的元素按顺序逐个分派到输出端的数组中, 从而形成多个一维数组, 图 5-44 是将有 5 个元素的一维数组抽取成 2 个一维数组。

(19) 交织一维数组函数

该函数的功能是多个一维数组按序进行插接重新构成一个一维数组, 图 5-45 是将 2 个一

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



维数组交织成一个一维数组。

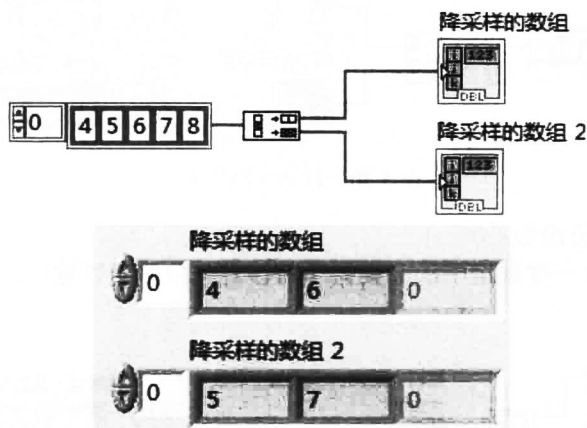


图 5-44 抽取一维数组

www.91HMI.com

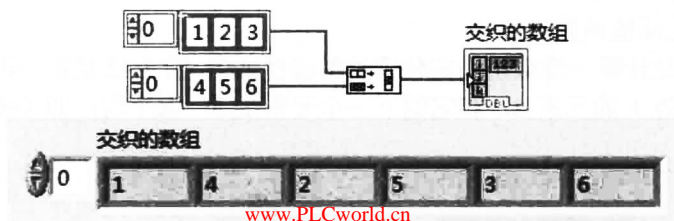


图 5-45 交织一维数组

(20) 二维数组转置函数

该函数的功能是二维数组的各行变为各列，图 5-46 是将一个 3 行 3 列的数组进行转置。

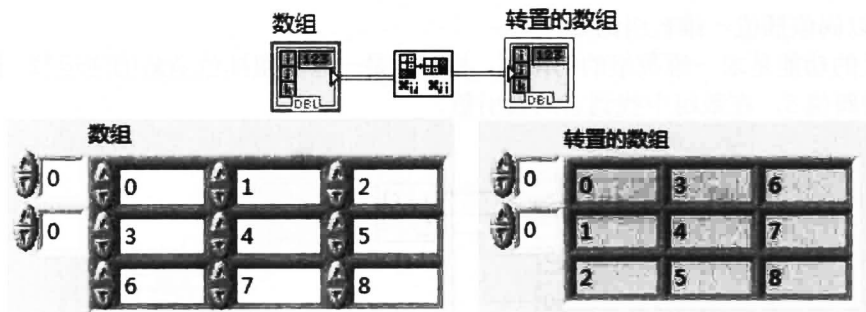


图 5-46 二维数组转置

5.4.2 簇函数

簇函数主要包括捆绑、解除捆绑、按名称捆绑、按名称解除捆绑、创建簇数组及簇与数组间转换函数等，其中捆绑函数和解除捆绑函数在它们中的地位最为重要。

选取数组函数的方法是：在后面板，选取“函数→编程→簇、类与变体”下的相应函数，并在空白位置单击鼠标予以放置。



(1) 捆绑函数

该函数的功能是将不同数据类型的数据组合成一个新簇，或为簇中的某个元素进行赋值。图 5-47 是将姓名、年龄、婚否 3 个数据捆绑成一个簇，以便作为一个整体进行使用。

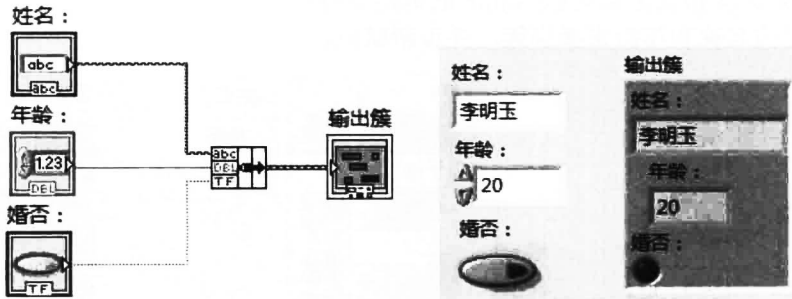


图 5-47 捆绑数据生成簇

使用簇函数不仅能够将数据捆绑在一起来构成簇，还能够用来替换簇中的元素，如图 5-48 就是此类的示例。

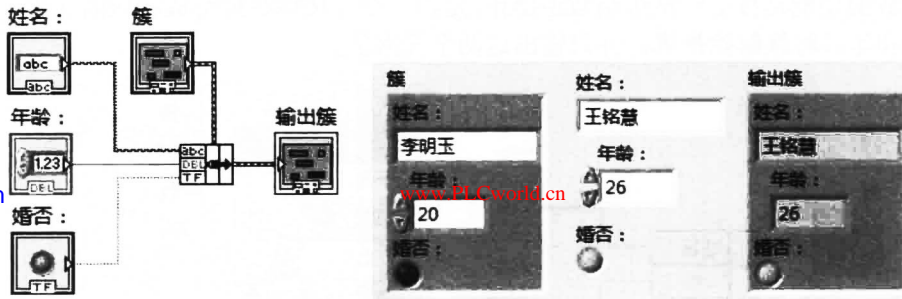


图 5-48 用捆绑函数替代簇中的元素

提示

在簇的使用中，不能将一个簇既当输入元素，又当输出元素来使用。在一个簇与另一个簇相连接时，两个簇的顺序和类型必须完全相同。

(2) 解除捆绑函数

该函数的功能是捆绑函数逆操作，是将一个簇拆解成相应的元素值。图 5-49 是将簇还原成 3 个基本构成元素，即姓名、年龄和婚否，以便单独进行处理。

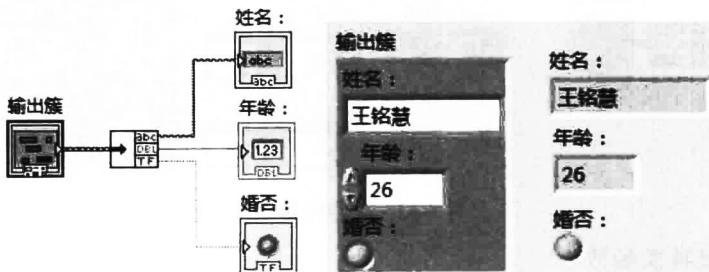


图 5-49 解除捆绑



(3) 按名称捆绑函数

该函数的功能是按照元素标签来捆绑元素以形成一个新簇，函数中的输入簇必须被接入，而且要求其元素至少有一个标签。用于该函数是按元素名称进行捆绑，因此端口的摆放顺序无要求，只要按元素名称接入相应的数据即可。

图 5-50 是按名称和年龄来捆绑簇，并重新赋值。

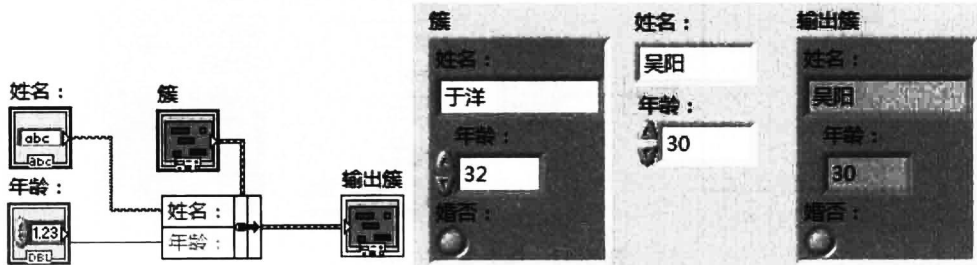


图 5-50 按名称捆绑

www.91HMI.com

(4) 按名称解除捆绑函数

该函数的功能是按名称捆绑函数逆操作，是将一个簇按标签拆解成相应的元素值。图 5-51 是按名称和年龄将簇解除捆绑，并只输出这两个元素值。

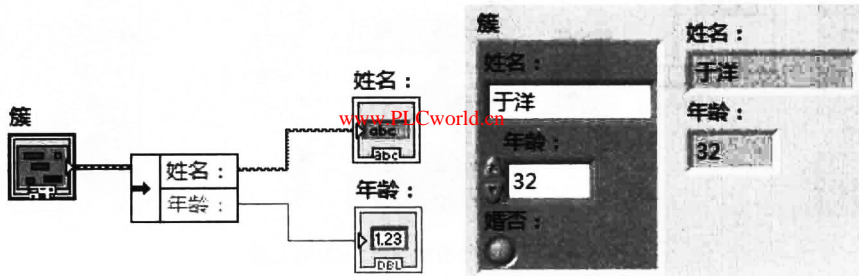


图 5-51 按名称解除捆绑

(5) 创建簇数组函数

该函数的功能是把多个具有相同类型的数组捆绑成簇数组，图 5-52 是将 3 个一维数组构成簇数组。

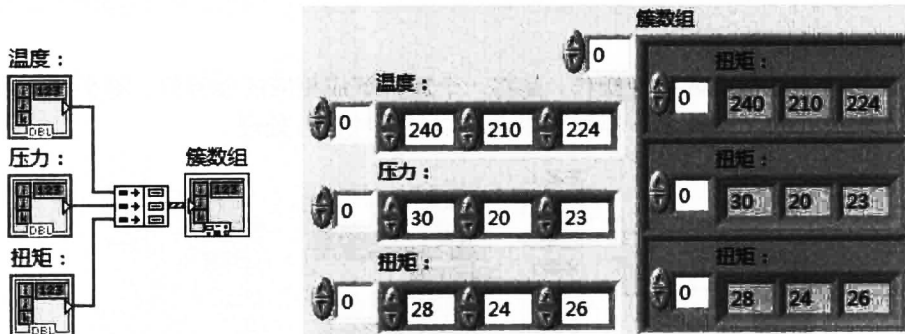


图 5-52 创建簇数组

(6) 数组至簇转换函数

该函数的功能是将一维数组的元素按顺序组合成簇，其中簇元素的类型与数组元素的类



型是相同的。图 5-53 是将一维数组转换为簇。

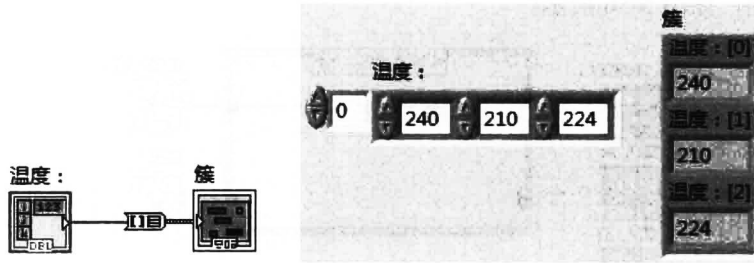
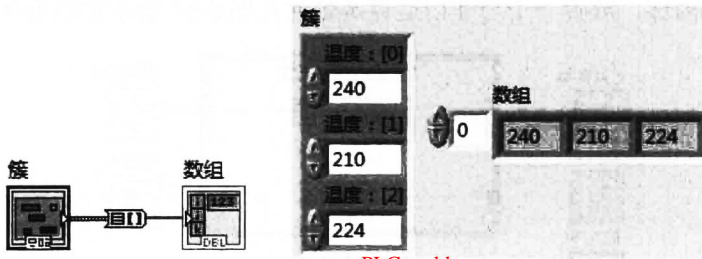


图 5-53 数组至簇转换

(7) 簇至数组转换函数

该函数的功能是将所有簇元素按顺序组合成一维数组并予以输出，它要求输入簇的所有元素具有相同的类型。图 5-54 是将簇转换为一维数组。

www.91HMI.com



www.PLCworld.cn

图 5-54 簇至数组转换

5.5 综合实例：逻辑芯片的功能模拟

LabVIEW 不仅是虚拟仪器的开发平台，也是仿真实验系统开发的有效工具。图 5-55 是一个逻辑芯片功能模拟的界面，通过前面的学习可以快速地实现这一功能，它涉及芯片名称的输入、芯片功能的选择及芯片功能的模拟。

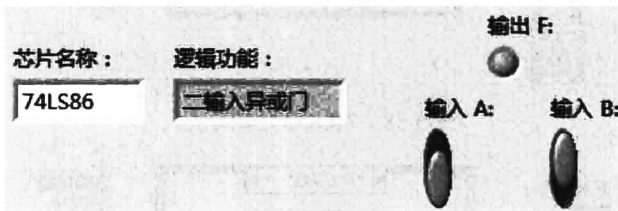


图 5-55 逻辑芯片功能模拟界面

实现逻辑芯片功能模拟的制作步骤如下。

① 新建一个 VI，在前面板放置一个字符串输入控件、一个字符串显示控件、两个垂直滑动杆开关和一个圆形指示灯，并按图 5-55 所示对控件标签加以修改。

② 为实现芯片功能的选择，在后面板添加一个条件结构，并将芯片名称图标与条件输入端口相连。



③ 选取条件假的分支，在其内部添加一个“暂不支持此芯片”的字符串常量，并与逻辑功能图标相连，如图 5-56 所示。

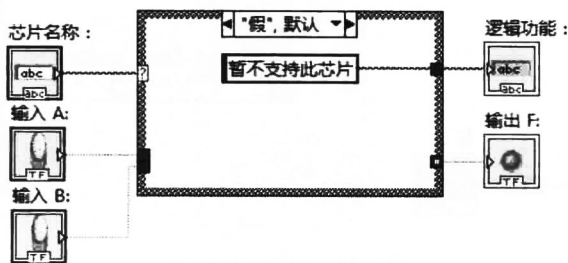
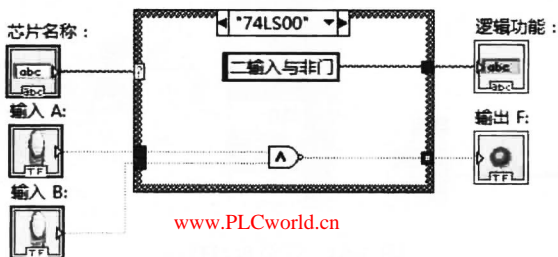


图 5-56 条件假分支处理

④ 选取条件真的分支，将条件值改为“74LS00”，并在其内部添加一个“二输入与非门”的字符串常量，并与逻辑功能图标相连。

www.91HMI.com

⑤ 在此分支内部，放置一个与非门运算函数并按图 5-57 所示加以连接。



www.PLCworld.cn

图 5-57 74LS00 功能分支

⑥ 复制此分支 2 次，并按图 5-58 和图 5-59 所示加以修改即可实现制作功能。

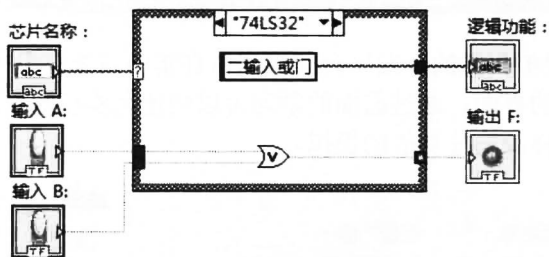


图 5-58 74LS32 功能分支

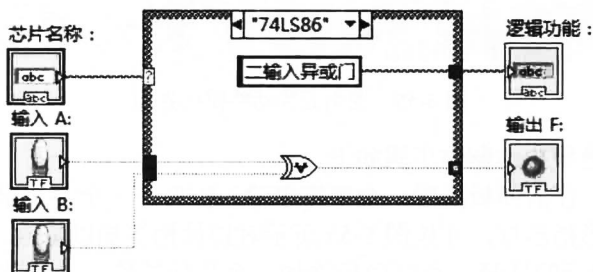


图 5-59 74LS86 功能分支



第 6 章

LabVIEW 控件及其使用方法

LabVIEW 程序是控件和程序框图相结合的产物，作为控件它是人机交互的桥梁，是界面设计的主要元素。控件可分为两类，即输入型控件和显示型控件，至于它们的具体形式则是多种多样。

【本章导读】

- ◆ 常用控件类型
- ◆ 特殊控件类型
- ◆ 常用控件属性设置
- ◆ 自定义控件等

6.1 常用控件类型

6.1.1 数值控件

数值控件包括输入控件和显示控件，主要用于单值的数值型数据。在一定情况下，通过快捷菜单对控件的输入输出性进行转换。

选取数值控件的方法如下。

① 在前面板“查看”下拉菜单中选取“控件选板”，或在前面板的空白处单击鼠标右键，将会见到图 6-1 所示的选板内容。

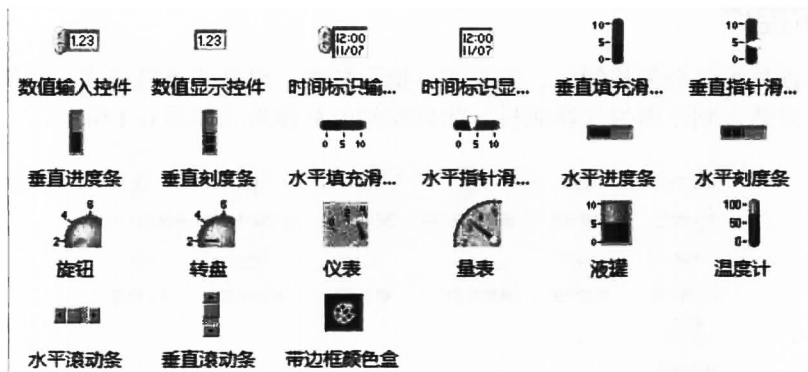


图 6-1 数值型控件



② 单击控件选板中的“新式→数值”，或选取“银色”、“系统”、“经典”及“Express”选板均可以找到数值型控件。

数值控件具有多种属性，通过属性对话框可进行设置，涉及内容有外观、数据类型、数据输入、显示格式等，其形式如图 6-2 所示。



图 6-2 数值属性设置对话框

以制作一个进程条为例，其具体操作如下。

- ① 选取空白 VI 的前面板，添加一个水平刻度条控件。
- ② 切换到后面板，添加一个 For 循环图标，并将循环次数设为“100”。
- ③ 将水平刻度条图标移入循环体，并连接循环变量 i 与进程图标输入端。
- ④ 为了增强可视性，在循环体内再添加一个等待 (ms) 图标和一个数值常量“100”。

进度条的程序框图和运行效果如图 6-3 所示。

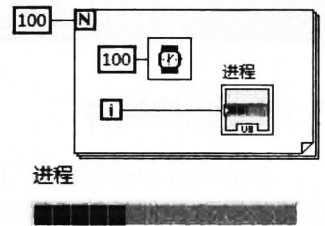


图 6-3 进程条程序框图及运行效果

6.1.2 布尔控件

布尔控件按外形可分为按钮式、开关式、指示灯式、滑杆式及摇杆式，它们均具有逼真的外观，故非常适合制作虚拟仪器面板。此类控件的具体形式如图 6-4 所示。



图 6-4 布尔型控件

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



选取布尔控件的常用方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→布尔”即可。

布尔型控件不仅具有外观特征，还具有相应的属性特征以配合对应的机械动作。此类机械动作有单击时转换、释放时转换、保持转换直到释放、单击时触发、释放时触发及保持触发直到释放，在使用时可视具体的需要而设置。

除了通过控件面板可以创建布尔控件，使用相关的布尔函数也能达到创建布尔控件的作用。

以制作一个流水灯为例，其具体操作如下。

① 选取空白 VI 的后面板，选取“函数→编程→布尔→数值至布尔数组转换”来添加一个转换图标，再为其创建一个显示控件并将标签改为“流水灯”。

② 添加一个 For 循环图标，并将循环次数设为“8”。

③ 添加一个乘函数图标和一个数值常量“2”，然后将全部图标移入循环体内。

④ 为了增强可视性，在循环体内再添加一个等待 (ms) 图标和一个数值常量“200”。

⑤ 用鼠标右键点击循环图标的右边框，从快捷菜单中选取“添加移位寄存器”来添加移位寄存器。

⑥ 经连线其程序框图和运行效果如图 6-5 所示。

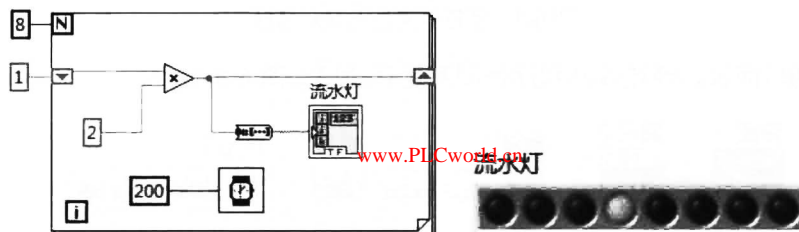


图 6-5 流水灯程序框图及运行效果

6.1.3 字符串与路径控件

字符串与路径控件包括字符串、组合框及文件路径控件，主要用于处理单一的字符型数据，其外观如图 6-6 所示。

选取字符串与路径控件的常用方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→字符串与路径”即可。



图 6-6 字符串与路径型控件

字符串与路径控件有多种属性，通过属性对话框可进行设置，所涉及的主要内容是外观选项卡内的显示样式，字符串对话框的形式如图 6-7 所示。

以文件路径输入控件的使用为例，其具体的操作方法如下。

① 选取空白 VI 的前面板，选取“控件→新式→字符串与路径→文件路径输入控件”来添加该控件。

② 再添加一个文件路径显示控件。



图 6-7 字符串属性设置对话框

③ 切换到后面板，经连线其程序框图和运行效果如图 6-8 所示。



图 6-8 文件路径控件使用

注：在运行前，将光标移到文件夹图标处单击，则会弹出文件对话框，选取所需的文件后，单击“确定”按钮予以确认退出。再运行程序即可获取文件的路径和文件名。

6.1.4 下拉列表与枚举控件

下拉列表与枚举控件包括文本下拉列表、菜单下拉列表、图片下拉列表、文本与图片下拉列表及枚举，其外观如图 6-9 所示。

选取下拉列表与枚举控件的常用方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→下拉列表与枚举”即可。



图 6-9 下拉列表与枚举控件

以枚举列表框的制作为例，其具体操作是：

- ① 选取空白 VI 的前面板，选取“控件→新式→下拉列表与枚举→枚举”来添加控件。
- ② 用鼠标右键枚举控件，在弹出的快捷菜单中选取“编辑项”。
- ③ 使用“插入”和“下移”按钮来辅助枚举选项的输入，各选项的内容和运行效果分



别如图 6-10 和图 6-11 所示。

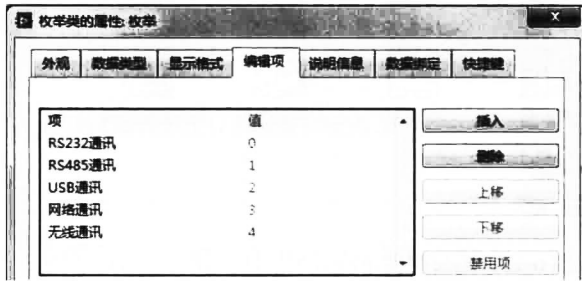


图 6-10 枚举项编辑

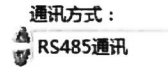


图 6-11 枚举列表

6.1.5 数组、矩阵与簇控件

数组、矩阵与簇控件包括数组、簇及矩阵等控件，其外观如图 6-12 所示。

选取数组、矩阵与簇控件的方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→数组、矩阵与簇”即可。



图 6-12 数组、矩阵与簇控件
www.PLCworld.cn

以实数矩阵值为数组为例，其具体的操作是：

- ① 选取空白 VI 的前面板，选取“控件→新式→数组、矩阵与簇→实数矩阵”来添加该控件。
- ② 切换到后面板，选取“函数→编程→数组→矩阵至数组转换”来添加此转换函数。
- ③ 为转换图标创建一个显示控件，经连线其程序框图和运行效果分别如图 6-13 和图 6-14 所示。

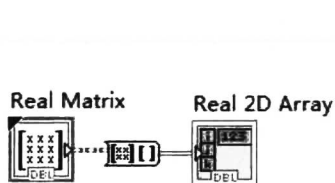


图 6-13 矩阵转换为数组程序框图

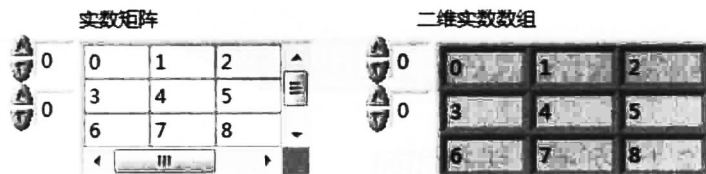


图 6-14 矩阵转换为数组运行效果

提示

在矩阵运算中，应使用矩阵型数据而非使用形似的二维数组。

6.1.6 图形控件

图形控件包括各种图表、表及二维和三维图形控件，其内容如图 6-15 所示。



图 6-15 图形控件

选取图形控件的方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→图形”即可。

以绘制按钮触发的波形图绘制为例，其具体的操作方法如下。

- ① 选取空白 VI 的前面板，选取“控件→新式→图形→波形图”来添加该控件。
- ② 再添加一个开关按钮和一个圆形指示灯。
- ③ 切换到后面板，添加一个 For 循环图标，将其循环次数设为“20”。
- ④ 为了将按钮的布尔值转换为数据值，添加一个比较图标，并设置好真假输出值。
- ⑤ 为了得到理想的观察效果，在循环中添加一个延时 200 毫秒的延时环节，经连线其程序框图和运行效果分别如图 6-16 和图 6-17 所示。

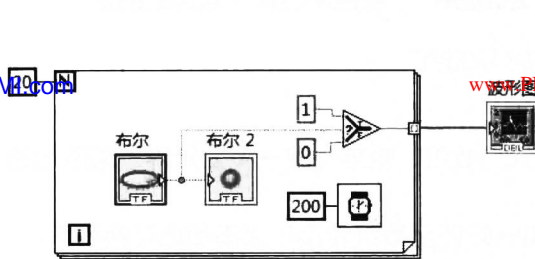


图 6-16 图形控件程序框图

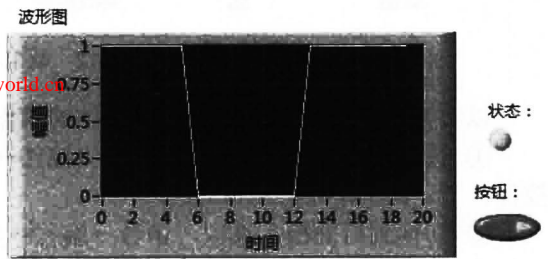


图 6-17 图形控件使用运行效果

6.2 特殊控件类型

6.2.1 列表、表格和树

列表、表格和树包括列表框、多列表框、表格、树形和 Express 表格等控件，其外观如图 6-18 所示。



图 6-18 列表、表格和树

选取列表、表格和树控件的常用方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→列表、表格和树”即可。



树形控件是以树形目录的方式来排列选项，并可展开多个层次。

以树形控件的使用为例，其具体的操作如下。

- ① 选取空白 VI 的前面板，选取“控件→新式→列表、表格和树→树形”来添加控件。
- ② 在树形控件的第一列依次输入 a、b、c，再使用快捷菜单中的“缩进项”将其按层缩进。
- ③ 选取树形的各项，并使用快捷菜单中的“项符号”来添加相应的符号。
- ④ 使用快捷菜单中的“编辑项”将各项的标识符分别改为 s1、s2 和 s3。
- ⑤ 切换到后面板，用鼠标右键单击“树形”图标，从快捷菜单中选取“创建→属性节点→活动单元格→活动列数”。
- ⑥ 选取属性节点，从激活的快捷菜单中选取“转换为写入”。
- ⑦ 选取属性图标的下边沿向下拖动，除保留“Cell String”和“ActiveItem Tag”属性之外，将其他属性删除。
- ⑧ 将“Cell String”属性修改为写入，经连线程序框图及运行效果分别如图 6-19 和图 6-20 所示。

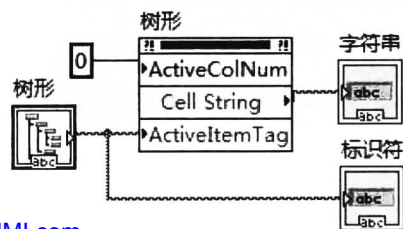


图 6-19 树形制作程序框图

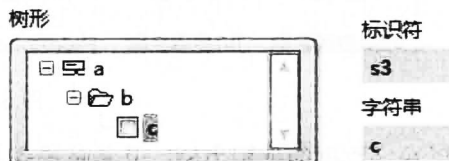


图 6-20 树形效果

6.2.2 容器控件

容器控件包括水平分隔栏、垂直分隔栏、选项卡控件、子面板及 .NET 容器和 .ActiveX 容器，其外观如图 6-21 所示。

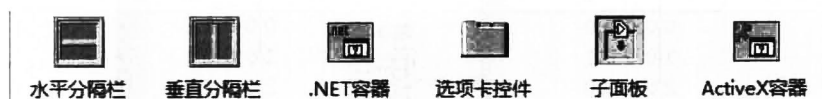


图 6-21 容器控件

选取容器控件的常用方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→容器”即可。

以选项卡的制作为例，其具体的操作如下。

- ① 选取空白 VI 的前面板，选取“控件→新式→容器→选项卡控件”来添加该控件。
- ② 用鼠标右击选项卡的边缘，用快捷菜单中的“显示项”来取消标签的显示。
- ③ 将光标移到选项卡 2 上，再次激活快捷菜单，选取“在后面添加选项卡”，则完成图 6-22 的选项卡制作。

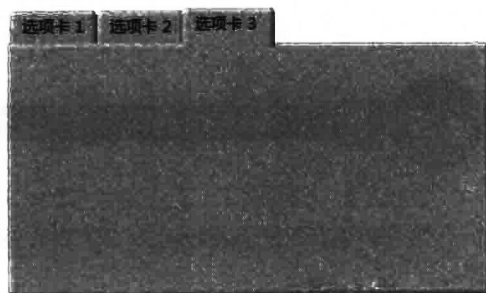


图 6-22 选项卡制作



提示

当光标移到某一选项卡时，单击鼠标就会选取该选项卡。选项卡是一种有效利用界面空间的方式，当界面内容较复杂时，可使用选项卡分类对显示内容加以管理。

6.2.3 I/O 控件

I/O 控件包括列模拟信号波形、数字信号波形、数字数据及各类下拉列表等，其外观如图 6-23 所示。

选取 I/O 控件的常用方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→I/O”即可。

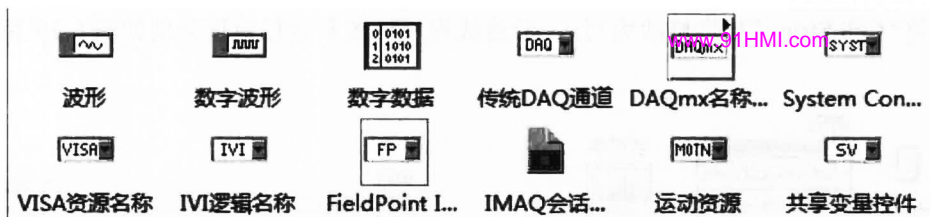


图 6-23 I/O 控件

以数字数据的使用为例，其具体操作是：

- ① 选取空白 VI 的前面板，选取“控件→新式→I/O→数字数据”来添加该控件。
- ② 切换到后面板，选取数字数据图标并单击鼠标右键，从快捷菜单中选取“转换为显示控件”。
- ③ 为该控件创建一个常量并输入相应的数据，则其程序框图和运行效果如图 6-24 所示。

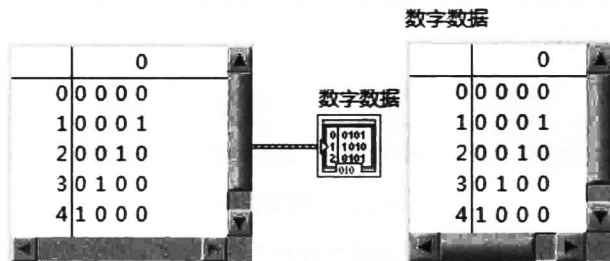


图 6-24 数字数据使用

提示

在绘制数字波形图时，是使用数字数据来向数字波形控件提供数据，该数据的特点是第一列是数据序号，第二列是以二进制形式排列的数据。

6.2.4 修饰控件

修饰控件包括方框、线条、箭头等对象，用于组合或分隔前面板的对象。修饰控件只起



修饰作用，并不参与数据处理过程，其具体形式如图 6-25 所示。

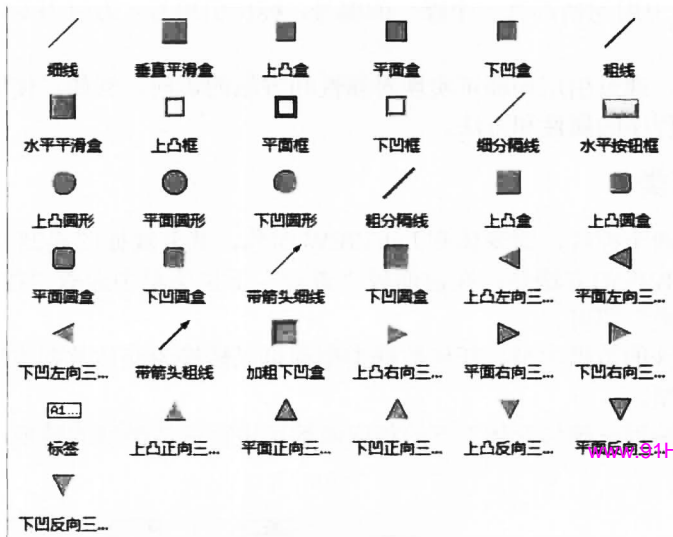


图 6-25 修饰控件

使用修饰控件的基本方法是，在前面板选取“控件→新式→修饰”子模板下的相应控件，然后拖放到前面板的空白处。

以构建一个数值输入盒为例，其具体操作如下：

- ① 在前面板放置一个“下凹盒”控件和两个数值型输入控件。
- ② 将两个数值型输入控件拖放到“下凹盒”控件内，然后调整好它们的位置及下凹盒的大小。
- ③ 框选三个控件，并选取“组”操作，则三个控件被组合成一体，其效果如图 6-26 所示。

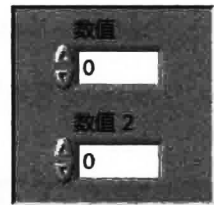


图 6-26 数值输入盒

提示

通过对修饰控件的使用，可以组合出不同的形状，这有利于装饰性图案的构造与立体效果的制作。

6.2.5 引用句柄

引用句柄包括应用程序、VI、控件、事件发生和菜单等引用句柄，其外观如图 6-27 所示。选取引用句柄的常用方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→引用句柄”即可。



图 6-27 引用句柄



引用句柄相当于指针，它的值并不重要，在使用中重点关注的是它所代表的对象。在 LabVIEW 中，每个引用句柄都有一个唯一的编号，称作引用号。为引用专门创建了数据控件即为引用控件。

在编程环境中，通过引用句柄可实现对属性和方法的访问，例如，使用引用程序引用就可以访问当前应用程序的属性和方法。

6.2.6 变体与类

变体与类只有两个控件，即变体和 LabVIEW 对象，其外观如图 6-28 所示。

选取变体与类控件的方法是：在前面板“查看”下拉菜单中选取“控件选板”，然后选取“新式→变体与类”即可。

变体是一种特殊的数据类型，任何数据类型都可以转换为变体数据类型，并在需要时再返回原来的数据类型。

使用“函数→编程→簇与变体”下的相应函数可进行变体类型的转换，其内容如图 6-29 所示。



图 6-28 变体与类

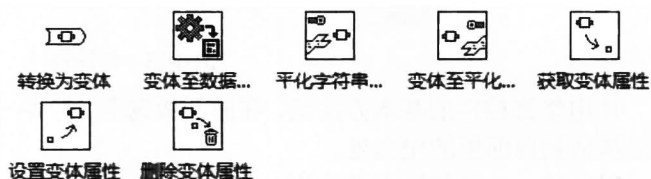


图 6-29 簇与变体函数

6.3 常用控件属性设置

6.3.1 改变控件外观

大多数控件的外观是允许按设计者意愿进行改变的，仍以温度计为例，在选取温度计控件后，通过拖动其外观的调节柄可改变其形状，调整前后的外观如图 6-30 所示。

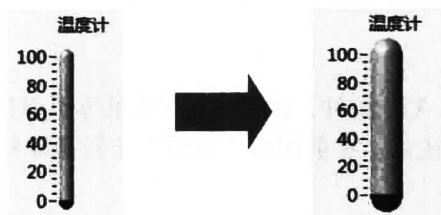


图 6-30 控件外观改变前后

6.3.2 调整数值控件上下限

数值控件基本上都具有上、下限值的属性，通过它可以限定输入值的范围。设置上、下限的方法是：选取数值型控件，然后单击鼠标右键，再从快捷菜单中选取“属性”选项。从弹出的属性对话框中，找到“最大值”和“最小值”文本框修改相应限值即可。



以设置温度计的上、下限值为例，在打开属性对话框后，其内容如图 6-31 所示。修改最大值、最小值即可完成设置调整。

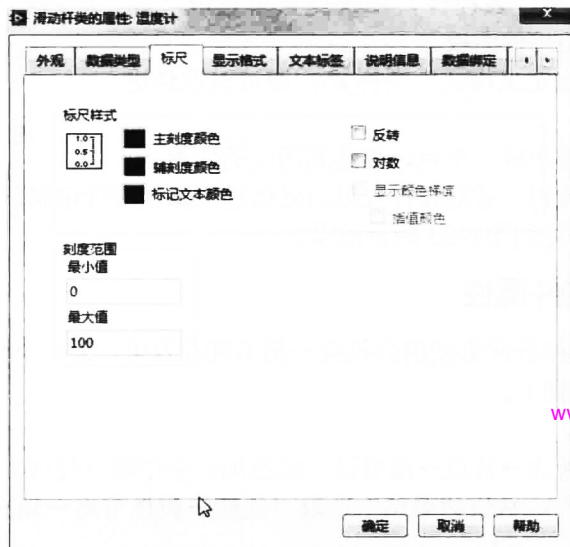


图 6-31 调整温度计限值

6.3.3 设置布尔控件机械属性

布尔控件有 6 种机械属性值，其默认值是“单击时转换”。要设置布尔控件的机械属性，其操作方法是，使用鼠标右键选取布尔型输入控件，然后由快捷菜单选取“机械动作”下的相应方式。

6 种机械属性如图 6-32 所示，可使用开关与指示灯相配合，检验各动作模式的差别。

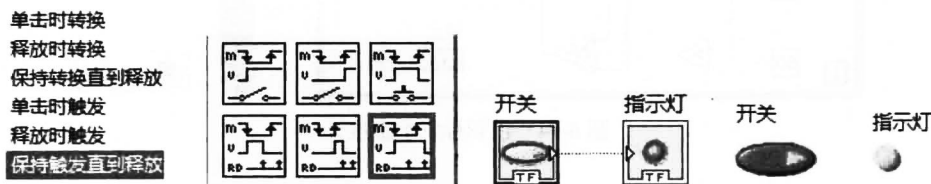


图 6-32 设置机械属性

6.4 自定义控件

控件是每个 VI 程序都必需的组成成分，控件的属性直接影响到程序的功能及显示效果。自定义控件就是将 LabVIEW 所提供的控件进行相应的修改所得到的控件，通常是修改控件的数据格式、数据类型及颜色等。

6.4.1 透明控件


要制作一个透明的仪表控件，其操作方法如下。

- ① 创建一个空白 VI。



② 选取“控件→新式→数值→仪表”来添加一个仪表控件。

③ 用鼠标右键单击仪表控件，由快捷菜单选取“高级→自定义…”，将弹出控件面板。

④ 单击“切换至自定义模式”图标，即可对控件进行自定义处理。

⑤ 将灰色的面板缩小成一个点，以达到透明的效果。

⑥ 单击“关闭”按钮，并对控件加以命名保存则退出控件编辑。

运行该程序，则见到图 6-33 所示效果。



图 6-33 透明仪表控件

6.4.2 动态更改控件属性

有时将控件的属性结合起来使用会得到一想不到的效果，以一个会按温度区间变色的温度计为例，其制作过程如下。

① 建一个空白 VI。

www.91HMI.com

② 选取“控件→新式→数值→温度计”来添加一个温度计控件。

③ 右击温度计控件，从快捷菜单中选取“创建→属性节点→填充颜色”来添加一个属性节点。

④ 切换到后面板，添加一个 While 循环，并为循环控制端添加一个停止控件。

⑤ 在循环体内，依次添加 2 个数值常量、2 个选择函数及一个大于和一个小于比较图标。

⑥ 再添加 3 个颜色盒常量，并将颜色分别设为“蓝”、“红”和“绿”。

⑦ 为循环添加一个 10 毫秒的延时环节，经连线其程序框图及效果如图 6-34 所示。

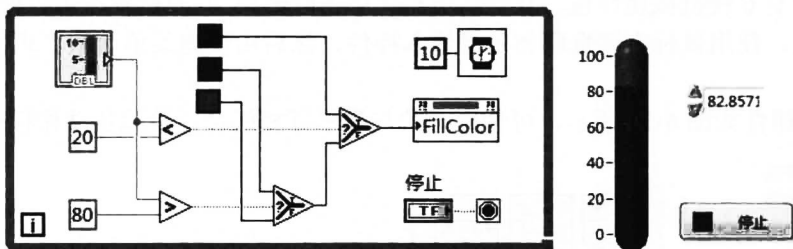


图 6-34 色彩标识型温度计

6.5 综合实例：虚拟仪表界面制作

在虚拟仪表中，其界面的设计关系到仿真性的质量。这里以一个万用表的界面制作来介绍其实现方法，具体形式如图 6-35 所示。

万用表仿真界面的操作步骤如下。

① 新建一个 VI，然后加以命名保存。

② 在前面板，选取“控件→新式→修饰→上凸盒”来添加该控件。

③ 选取工具面板的“编辑文本”工具，输入字符串“BinjiAnG”，在将字体设为“MS UI Gothic”，大小设为“24”，颜色设为“白色”，并选取“粗体”。

④ 输入字符串“DT 9906C”，在将字体设为“黑体”，大小设



图 6-35 万用表仿真界面



为“24”，颜色设为“白色”，并选取“粗体”。

⑤ 选取“控件→新式→修饰→上凸圆盒”来绘制一个小按键，然后再复制 2 个。

⑥ 选取工具面板的“设置颜色”工具，将 3 个小按键的颜色依次设为“浅灰色”、“深灰色”和“红色”。

⑦ 依次输入字符串“BLIGHT”、“KHz MHz”和“POWER”，并将颜色设为“白色”。

⑧ 使用“对齐对象”工具中的“水平对齐”，使所有按键和字符串分别处于相应的水平线上。

⑨ 选取“控件→新式→修饰→下凹圆形”来绘制一个小插孔，并将圆心染成“黑色”。

⑩ 再复制 2 个小插孔，并使之均匀地处于同一水平线上。

⑪ 输入字符串“A COM VΩHz”，并将颜色设为“白色”。

⑫ 选取“控件→新式→数值→旋钮”来添加一个旋钮，然后调整好旋钮的大小。

⑬ 切换到后面板，用鼠标右键单击旋钮图标，从快捷菜单中选取“属性”。

⑭ 在弹出的对话框中选取“文本标签”，然后使用“插入”和“下移”按钮来添加和输入文本标签。

⑮ 选取“标尺”选项卡，从中勾选“显示颜色梯度”。再选取“外观”选项卡，从中勾选“显示数字显示框”。

经细心调整，则可获得满意的制作效果。运行该程序，选取“操作值”工具旋转旋钮，则可看到显示框的信息会跟随指向的标签变化。



www.PLCworld.cn

合理地运用输入和显示控件，能制作出美观而易于操控的人机交互界面，应不断地提升制作水准。



第 7 章

LabVIEW数据表达与显示

数据处理是程序设计的主体,面对千变万化的数据,LabVIEW有丰富的表达与显示方式,如各种常见的输入、输出控件、二维图形、三维图形等,这为直观地了解和分析数据提供了极大的方便。

【本章导读】

基本数据表达与显示

- ◆ 二维图形
- ◆ 三维图形
- ◆ 其他图形显示控件等

由于 LabVIEW 是一种面向虚拟仪表系统开发的平台,因此它在数据表达方式上极具特色和优势,它推出的二维图形和三维图形表达方式,能够使开发者在弹指间完成通常需要多日才能实现的功能。

7.1 基本数据表达与显示

就基本数据类型来讲,可分为数值型数据、布尔型数据和字符型数据。由于数据类型的不同,它们在表达和显示方式上也存在较大的差异。

(1) 数值型数据的表达与显示

文本框是数值型数据的常规表达方式,随着操作对象的不同,还可采用滑杆、进度条、旋钮、仪表盘及液位计等方式。

此外,为了表达多值数据,数组也是一种有力的表达方式。

(2) 布尔型数据的表达与显示

由于布尔型数据只有两个值,因此多是以指示灯、开关或按钮的面目出现。

(3) 字符型数据的表达与显示

对于字符型数据,其表达形式最简单,只有文本框一种方式。

7.2 二维图形

在 LabVIEW 提供的二维图形显示方式中,有两类既十分相似又有所差异的显示控件,



它们是图与图表。所谓“图”是一种先将采集或检测到的各种数据进行存储，然后再集中进行显示的非实时性显示方式，而“图表”则是一种将采集或检测到的各种实时数据依次进行显示的方式，它是一种采集数据随时间而变化的趋势图。

7.2.1 波形图表

波形图表是显示一条或多条曲线的波形显示控件，主要用于显示以固定采样率得到的数据波形，可用它来模拟示波器的显示效果。

波形图表控件主要由标题、标签、图例、X 标尺和 Y 标尺、X 滚动条及数字显示等组成，具体如图 7-1 所示。

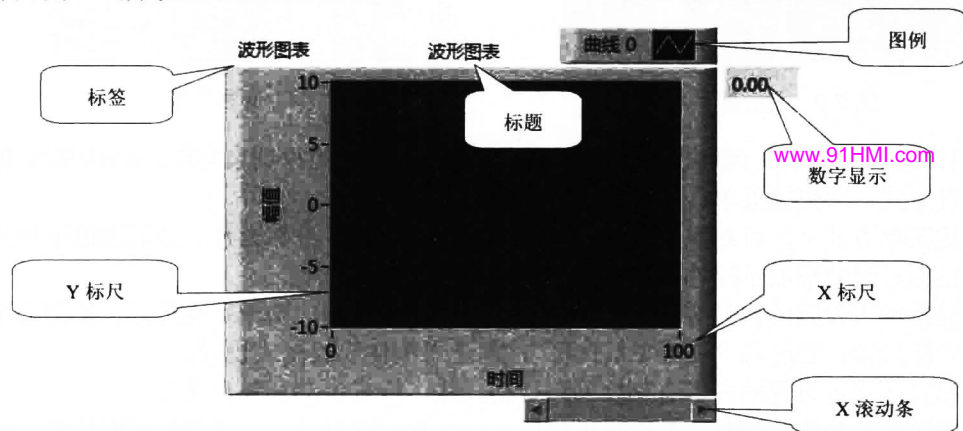


图 7-1 波形图表

在波形图表的组件中，关键部件的作用如下：

图例——用于设置波形曲线的相关属性，如线条的样式、宽度及颜色等。当显示多条曲线时，在图例的上下边框上拖动鼠标，可显示出更多的图例。

数字显示——用于显示最新一个接收到的数据。

X 滚动条——用于滚动显示图表中的数据，当图表中的数据量很大时，用滚动条来查看未显示的数据是十分必要的。

X 标尺和 Y 标尺——用于辅助识读波形数值。

(1) 波形图表的设置

① 设置图表历史长度。图表历史长度的作用是控制数据缓冲区的长度，之所以要设置缓冲区长度是因为波形图表的数据并非存放在数组中，而是实时显示的。为了能够浏览全部的数据波形，就需设置足够的缓冲区长度。否则，先前呈现的波形就会随着时间的推移而失去，从而无法再次浏览。

设置图表历史长度的方法是，用鼠标右键单击波形图表，从快捷菜单中选取“图表历史长度”，则会弹出图 7-2 所示的对话框。在改变文本框的数据后，单击“确定”按钮即可生效。在默认情况下，缓冲区的长度为 1024。

② 叠层/分格显示曲线。当波形图表用于多条曲线显示时，可以选择分格或叠层方式显示曲线。所谓“分格”显示是指将多条曲线分散到多个显示区域并以单条曲线的形式来显示；而“叠层”显示则是指在同一显示区域显示多条曲线，

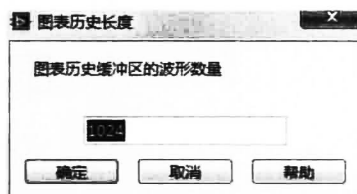


图 7-2 设置图表历史长度



具体形式如图 7-3 和图 7-4 所示。

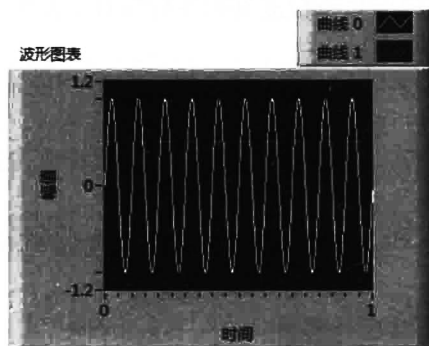


图 7-3 叠层显示

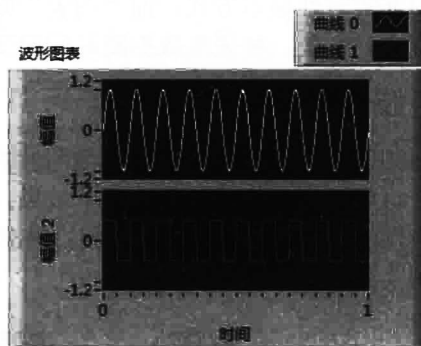


图 7-4 分格显示

③ 属性设置。为了达到理想的观测修改，可以通过属性的设置来实现。LabVIEW 提供了 3 种设置方式，即快捷菜单、控件选板和属性对话框。

在快捷菜单方式下，可对显示项、标尺、图表历史长度等进行设置。激活快捷菜单的方法是，用鼠标右键波形显示控件。

通过显示项可设置的内容有标签、标题、图例、标尺图例、图形工具选板、X 滚动条、X 标尺、Y 标尺等。若选取“图形工具选板”，则进入控件选板模式设置。

通过标尺选项可设置的内容有刻度间隔、格式化、样式自动调整标尺等。

刻度间隔——分为均匀刻度与随机刻度两种，若采用均匀方式，则刻度根据数值长度自动标注，否则只显示边界的刻度。

样式——用于选择刻度的标注风格，共有 9 种风格可供选择。

格式化——用于设置刻度的各种属性，如控件的外观、格式与精度、曲线、标尺等。当选取“格式化”选项时，会弹出属性设置对话框，内容如图 7-5 所示。此时，就进入属性对话框方式设置。



图 7-5 属性设置对话框



用属性对话框可设置的内容有波形外观、显示格式、曲线和标尺等，其中几个主要参数的作用如下。

① 时间（X 轴）的数据类型，它关系到时间轴的标尺是按数值显示还是按时间值显示，此外对于时间值还可选择以绝对时间显示还是以相对时间显示。

② 缩放因子，它涉及偏移量和缩放系数，其中的比例系数关系到标尺的刻度比例，数值越大，在单位长度上代表的值越大。而偏移量则关系到曲线距零点的距离，当偏移量大于零时，曲线向零点之上移动；当偏移量小于零时，曲线向零点之下移动。

③ 最大、最小值，用于限定波形的显示范围，对于超出显示范围的数据予以舍弃。

④ 自动调整标尺，用于在波形显示过程中动态地调整标尺，以显示出全部数据的波形。其实它就是动态地调整标尺的最大、最小值，以使波形全部落在波形显示范围内。

⑤ 反转，就是对显示的波形按指定轴进行翻转显示。

⑥ 对数，就是对显示的波形按指定轴进行对数化处理并显示。

⑦ 网格样式与颜色，它涉及是否启用主、辅网格及其色彩的设置。

⑧ 线条样式，用于改变波形曲线的类型，它分为实线、虚线及点画线等。

⑨ 线条宽度，用于改变波形曲线的线宽。

⑩ 平滑，用于改变曲线显示时的平滑算法。

⑪ 插值，用于改变曲线显示时的插值算法。

⑫ 填充至，用于设置曲线填充至基线的方式，它分为无、零、无穷大和负无穷大等方式。

⑬ 点样式，用于设置曲线显示点的样式，它分为空心、实心、圆点及方点等。

2.9 波形图表的刷新模式

www.PLCworld.cn

波形图表有三种刷新模式，分别是带状图表模式、示波器模式和扫描图模式。

① 带状图表模式：该模式所显示的波形是连续的，并以一定速度向右移动，从而便于实时观测新采集到的数据。具体的执行过程是，先从左到右显示接收到的数据，当波形曲线到达绘制区域的右边界时，先前绘制的波形自动向左移动。该模式是刷新显示的默认方式。

② 示波器模式：该模式是一种屏屏相接的显示方式，每显示完一屏，就进行一次屏幕擦除。具体的执行过程是，先从左到右显示接收到的数据，当波形曲线到达绘制区域的右边界时，便自动擦除所绘制的曲线，然后再从左向右绘制新接收的数据曲线。

③ 扫描图模式：该模式是一种无清屏的显示模式，它是通过一条移动红色扫描线来区分刷新前后的数据。具体的执行过程是，先从左到右显示接收到的数据，当波形曲线到达绘制区域的右边界时，它不对显示的曲线进行擦除，而是从左向右继续绘制新收到的数据曲线，此时在新旧数据间会有一条垂直的红线将两部分的数据分隔开。

为观察波形图表的三种刷新模式，可运行图 7-6 所示的程序框图进行观察，三种显示效果分别如图 7-7、图 7-8 和图 7-9 所示。

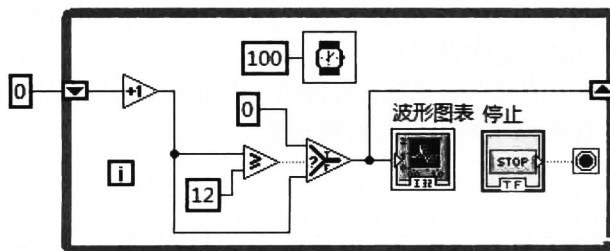


图 7-6 刷新模式验证程序框图



更改刷新模式有两种方法，一种是在运行前，通过右击波形图表控件，然后由快捷菜单的“高级→刷新模式编程”子菜单来选取；另一种是在运行中，通过右击波形图表控件，再由快捷菜单中的“刷新模式”子菜单来选取。

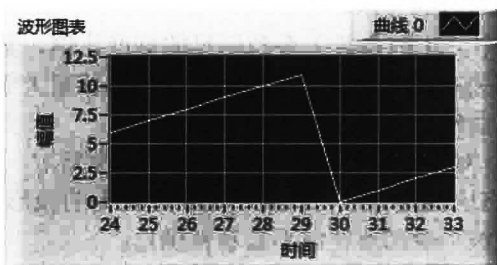


图 7-7 带状图表模式

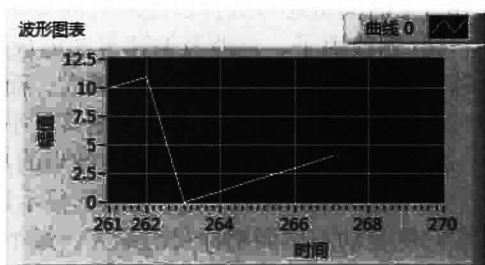


图 7-8 示波器模式

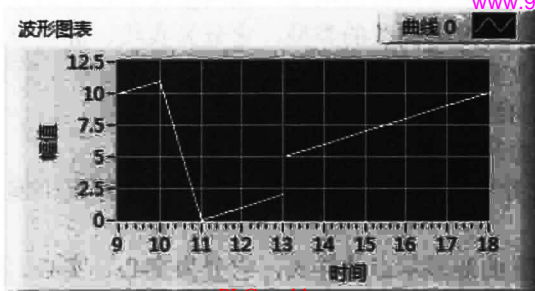


图 7-9 扫描图模式

(3) 波形图表的使用

以制作图 7-3 所示的波形图表为例，其操作步骤如下。

① 新建一个 VI，然后进行命名保存。

② 在前面板，选取“控件→新式→图形→波形图表”来添加一个控件。

③ 在后面板，选取“函数→信号处理→波形生成”子模板，从中选取方波波形和正弦波形进行添加。

④ 为使方波的幅值缩减，向后面板添加一个除函数和数值常量并赋值“2”。

⑤ 选取“函数→Express→信号操作→合并信号”并予以添加，经连线程序框图如图 7-10 所示。

⑥ 为达到理想的显示效果，打开波形图表属性对话框，在“外观”选项卡将曲线显示数改为“2”，在“显示格式”选项卡将时间（X 轴）的类型改为“十进制”，在“标尺”选项卡将时间（X 轴）设为“自动调整标尺”，然后点击“确定”完成设置。

⑦ 再次打开波形图表的属性对话框，将幅值（Y 轴）的最小值和最大值分别设为“-1.2”和“1.2”。此时，若运行该程序则会见到预期效果。

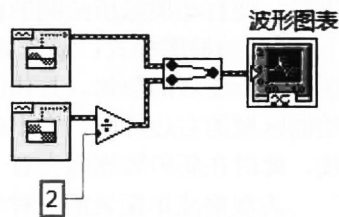


图 7-10 波形图表

7.2.2 波形图

波形图与波形图表的功能基本相同，也是显示一条或多条曲线的波形显示控件，波形图不能显示单个数值型数据，但支持数组、簇和波形数据的显示。



在显示方面，波形图不具备分格与层叠显示模式，也没有显示数据缓冲区和刷新方式。但它有一个新特色就是游标。所谓“游标”就是一种特殊数据点的标识符号，在 LabVIEW 中游标也呈多种形态，可是需要加以设置。

波形图的外观及部分的组件如图 7-11 所示。

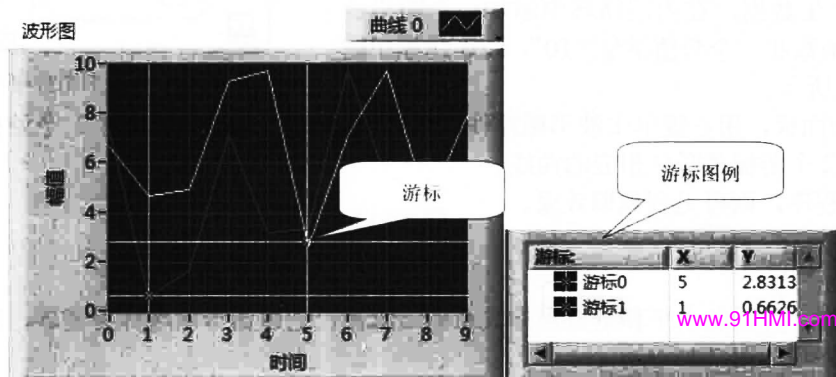


图 7-11 波形图

(1) 创建游标

创建游标的方式是，用鼠标右键选取波形图控件，然后在弹出的快捷菜单中选取“属性”选项，则弹出属性对话框。选取“游标”选项卡，单击“添加”按钮，则创建一个名称为游标0的新游标。若需要多个游标，可继续单击“添加”按钮来添加。

(2) 设置游标

通过“游标”选项卡，可对游标的属性予以设置，如游标的颜色、样式、线条宽度等。

游标有 3 种类型，即自由、单曲线和多曲线。不同类型的游标在波形图中的移动位置是不同的，自由游标可在整个显示区域任意移动；单曲线游标只在相关联的曲线上移动。用快捷菜单中的“关联至”子菜单，可使游标与指定曲线建立关联。

此外，使用游标图例可观察游标在波形显示区域内的 X、Y 坐标值。若在游标图例上激活快捷菜单，还可选取“删除游标”来清除指定游标。

(3) 设置背景色

在通常情况下，波形显示区域的背景是黑色的，要改变背景色，其方法十分简单，就是先在工具模板上设置好背景色，然后用选取“设置颜色”工具在波形显示区域单击即可。

(4) 增添标尺

当显示的波形曲线比较多时，为了能清晰地读取曲线上的数据，可为每条曲线配置一个标尺，具体的操作方法是，在波形图的标尺上单击鼠标右键，则会弹出快捷菜单，从中选取“复制标尺”，则可增添新的标尺。若选取“两侧交换”，则可互换两侧标尺的位置。

在波形图有多个标尺的情况下，快捷菜单会出现“删除标尺”选项，凭此可去除已有的标尺。

(5) 波形图的使用

以制作图 7-11 的波形图为例，其操作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→波形图”来添加一个控件。



LabVIEW 2013 完全自学手册

③ 在后面板，添加 2 个 For 循环图标，并构建成内外双层循环结构。

④ 将外层循环数设为“2”，内层循环数设为“10”。

⑤ 为产生数据，在内层循环中添加一个随机函数、一个乘函数和一个数值常量“10”，经连线程序框图如图 7-12 所示。

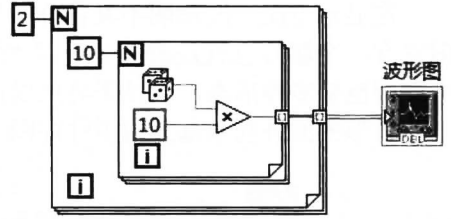


图 7-12 波形图程序框图

⑥ 在前面板，用右键单击波形图控件，从弹出的属性对话框中添加 2 个游标。再使用快捷菜单将 2 个游标关联到相应的曲线。

运行该程序，则可见到预期效果。

7.2.3 XY 图

XY 图是一种反映水平和垂直坐标关系的二维图，它以笛卡尔坐标为基础，用于绘制多值函数，其外观如图 7-13 所示。

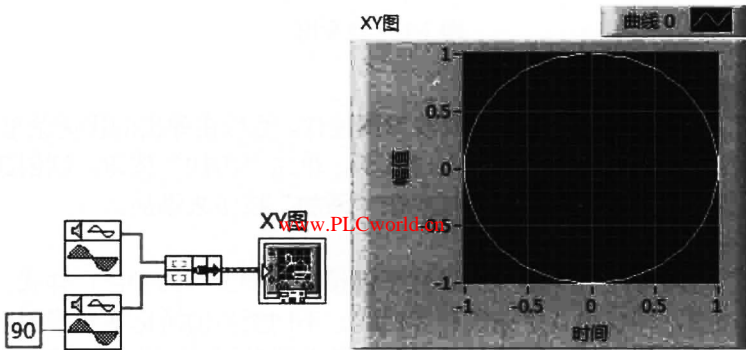


图 7-13 XY 图

以绘制一个圆为例，其操作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→XY 图”来添加一个控件。
- ③ 在后面板，选取“函数→信号处理→信号生成→正弦信号”来添加两个正弦信号图标，然后将其中一个的相位设为“90”，以形成余弦信号。
- ④ 添加一个捆绑图标，将两个信号合成到一起。经连线其程序框图如图 7-13 所示。运行该程序，则可见到预期效果。

7.2.4 Express XY 图

Express XY 图是一种更便于进行数据显示的波形图，它与 XY 图功能相近，只是它比 XY 图更便于进行数据的格式化。

以绘制一个等腰三角形为例，其操作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→Express XY 图”来添加一个控件。
- ③ 在后面板，添加 2 个数组图标并进行相应的赋值，经连线其程序框图如图 7-14 所示。运行该程序，则可见到图 7-15 所示的效果。若增添数组的元素数可增加图中的端点数和



边数，当元素数比端点数多且首尾元素相同时，会形成封闭的图形。

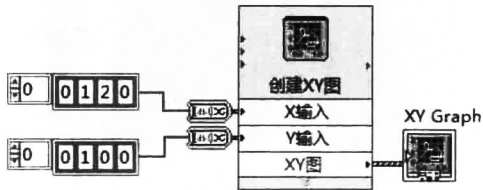


图 7-14 Express XY 图程序框图

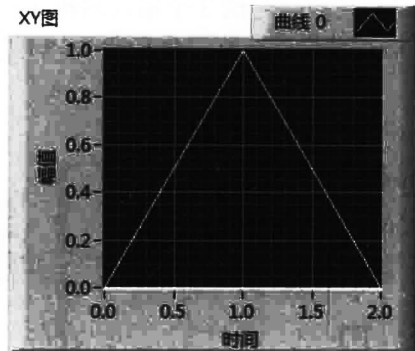


图 7-15 Express XY 图

www.91HMI.com

7.2.5 强度图

强度图是另一类图形显示方式，它是通过对二维平面放置色块的方式来显示三维数据，即用一维的数据来设置显示的颜色，用二维的数据来设置颜色的显示位置。强度图的显示效果如图 7-16 所示。

以绘制一个梯度强度图为例，其操作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→强度图”来添加一个控件。
- ③ 在后面板，添加一个 For 循环图标，并将循环数设为“100”。

④ 为了产生数据，选取“函数→信号处理→信号生成→斜坡信号”以添加该信号，将变量 i 与斜坡信号的 3 个输入端相连，最终的程序框图如图 7-17 所示。

运行该程序，则可见到图 7-16 所示的效果。

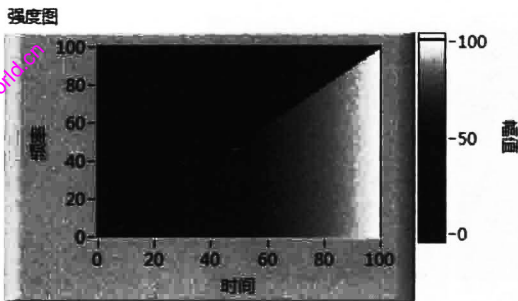


图 7-16 强度图

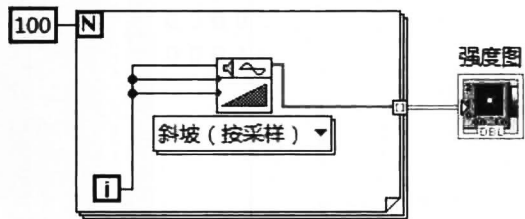


图 7-17 强度图程序框图

7.2.6 强度图表

强度图表与强度图的功能基本一致，二者的区别就是图表与表的差异。

以绘制一个随角度变化的正弦信号强度图表为例，其操作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→强度图表”以添加该控件。
- ③ 在后面板，添加一个 For 循环图标，并将循环数设为“360”。



④ 为了产生数据，选取“函数→信号处理→信号生成→正弦信号”以添加该信号，将变量 i 与正弦信号的 3 个输入端相连，最终的程序框图如图 7-18 所示。

运行该程序，则可见到图 7-19 的效果。

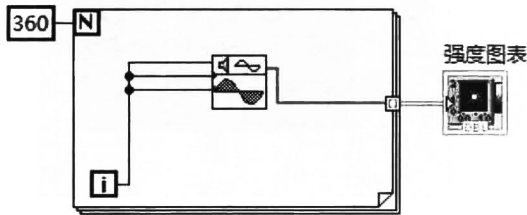


图 7-18 强度图表程序框图

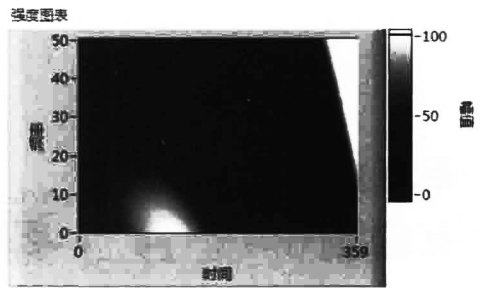


图 7-19 强度图表

www.91HMI.com

7.2.7 数字波形图

数字波形图是一种显示二进制数字的波形图，它只含有 0、1 两种状态，其输出形式有些类似逻辑分析仪，但却有着本质性的区别。逻辑分析仪是按时间顺序由一条曲线来反映一个通道的状态变化，而数字波形图则是用一系列的各行状态来反映一个数据所代表的状态变化。其中的第一列为数值 1 的波形，第二列为数值 2 的波形，其他的以此类推。而行 0 至行 3，则是二进制数由低至高的各位状态。

若用数字波形图来虚拟逻辑分析仪，需要对各通道的二进制数值进行相应的转置处理。数字波形图的外观如图 7-20 所示。此外，在数字波形图中，也可使用游标进行标识。

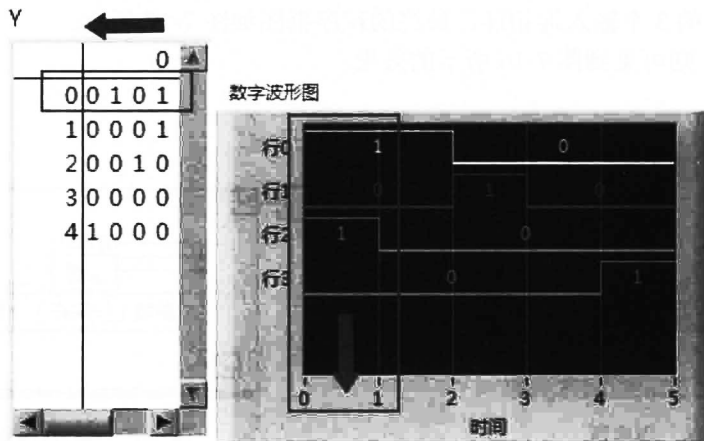


图 7-20 数字波形图

以图 7-20 为例，其数字波形图的制作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，添加一个数字波形图控件。
- ③ 在后面板，选取“函数→编程→波形→数字波形→创建波形”并予以添加。
- ④ 用鼠标右键单击“创建波形”图标，在快捷菜单中选取“添加元素”，然后再次激活快捷菜单并选取“选择项”下的“dt”选项，以更改新元素的属性。



⑤ 为创建波形图标的“dt”端口和“Y”端口分别添加常量“1”和输入控件。经连线程序框图如图 7-21 所示。

⑥ 切换到前面板，为数字输入控件 Y 输入相应的数值，然后运行程序可见到预期的效果。

除了上述方法外，还有许多种创建数字波形图的方法，感兴趣的话不妨一试，如用空数字波形、空数字数据、电子表格字符串至数字转换、数组表示数字信号集等，其程序框图如图 7-22 所示。

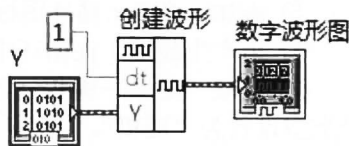


图 7-21 数字波形图程序框图

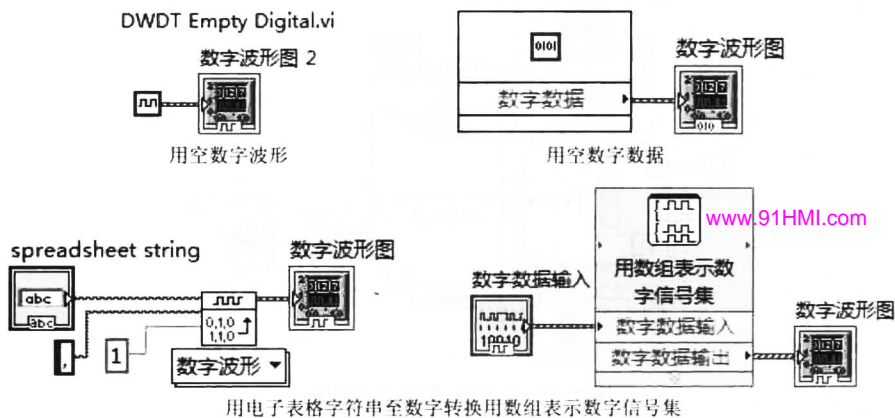


图 7-22 各种数字波形图制作方式

7.2.8 混合信号图

混合波形图是一种集成模拟信号和数字信号的波形显示模式，其显示效果如图 7-23 所示。

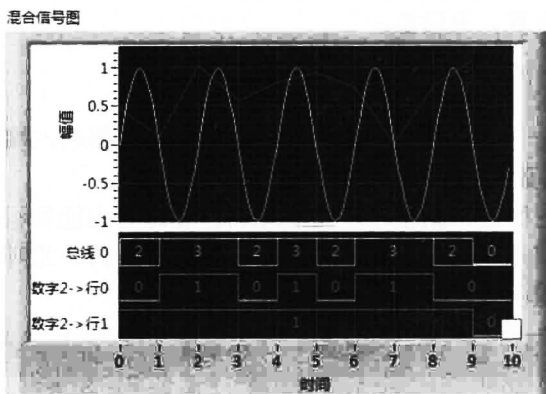


图 7-23 混合波形图

要制作图 7-23 的混合信号显示效果，其操作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→混合信号图”来添加该控件。
- ③ 在后面板，选取“函数→信号处理→信号生成→正弦信号”以添加该信号。
- ④ 在属性配置对话框，取消“自动”选项，并将正弦信号的频率、采样频率分别设为“0.5”和“10”。



- ⑤ 添加一个 For 循环图标，并将循环数设为“10”。
 - ⑥ 为产生数据，在循环中添加一个随机函数、一个乘函数和一个数值常量“1.2”。
 - ⑦ 选取“函数→编程→波形→数字波形→数字波形发生器”以添加数字信号，再将信号类型设置为“随机”。
 - ⑧ 将数字波形发生器的信号数和采样率分别设为“2”和“1”。
 - ⑨ 添加一个捆绑图标，经连线程序框图如图 7-24 所示。
- 运行该程序，则可见到预期效果。

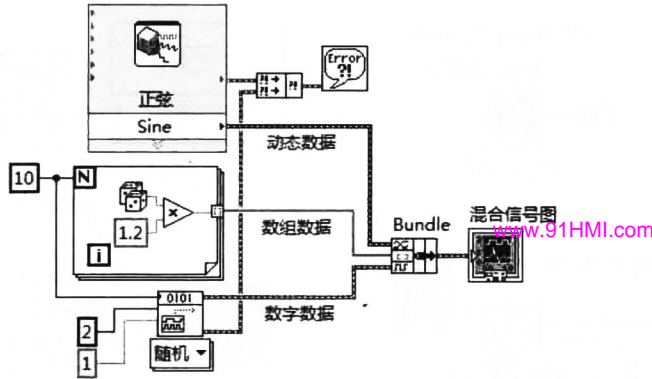


图 7-24 混合波形图程序框图

7.3 三维图形


www.PLCworld.cn

LabVIEW 不但提供了丰富的二维图形显示控件，还提供了多样的三维图形显示控件，其中包括三维线条图形控件、三维曲面图形控件及三维参数图形控件等。在许多情况下，三维图形往往比二维图形更直观、更形象。

7.3.1 三维曲线图

三维曲线图用于显示一条三维空间的曲线，它有三个数据输入端口，分别用来输入 x 向量、y 向量和 z 向量，其数据类型为一维数组。

同二维图形相比，三维图形新增了一个呈现窗，通过它也能显示三维曲线图。

当鼠标移动到图形显示区域时，光标的呈现状态会变为 ，此时按住鼠标左键进行移动可改变观察的视角，其显示效果如图 7-25 所示。

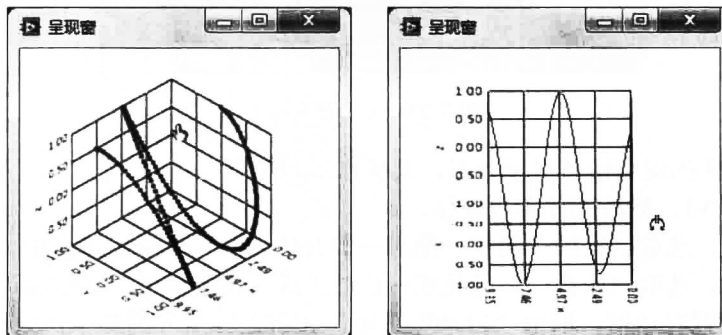


图 7-25 三维曲线效果图



以绘制一条螺旋线为例，其制作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→三维线条图形”来添加该控件。
- ③ 在后面板，添加一个 For 循环图标，并将循环数设为“200”。
- ④ 选取“函数→数字→初等与特殊函数→三角函数→正弦”来添加正弦函数。
- ⑤ 再添加一个余弦函数，并将正弦函数和余弦函数移入到循环内部。
- ⑥ 向后面板依次添加一个除函数、一个乘函数和两个数值常量并赋值，经连线程序框图如图 7-26 所示。

图如图 7-26 所示。

运行该程序，则其显示效果如图 7-25 所示。

注：该曲线的三个向量分别为， $x=i/20$ ， $y=\sin(i/20)$ ， $z=\cos(i \times 2\pi/200)$ ，其中 $i=0 \sim 200$ 。

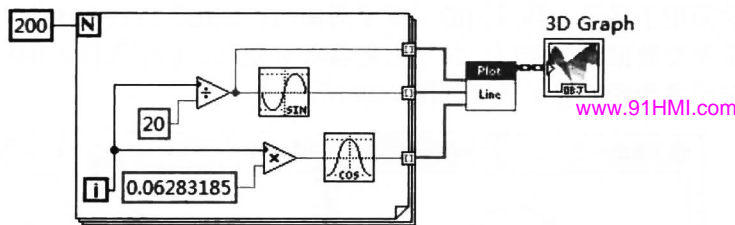


图 7-26 三维曲线程序框图

7.3.2 三维曲面图

三维曲面图用于仿真一些简单的三维空间曲面。它有三个主要数据输入端口，分别用于输入 x 坐标、y 坐标和 z 坐标，其数据类型为二维数组。三维曲面图的显示效果如图 7-27 所示。

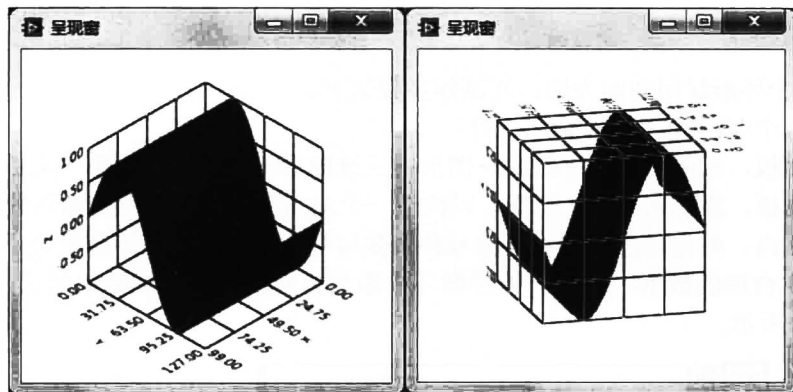


图 7-27 三维曲面效果图

以绘制一个正弦曲面为例，其制作步骤如下：

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→三维曲面图形”来添加该控件。
- ③ 在后面板，添加一个 For 循环图标，并将循环数设为“100”。
- ④ 选取“函数→信号处理→信号生成→正弦信号”来添加正弦信号，并将其与三维曲面图形的 Z 矩阵端口相连，连线后的程序框图如图 7-28 所示。



运行该程序，则其显示效果如图 7-27 所示。

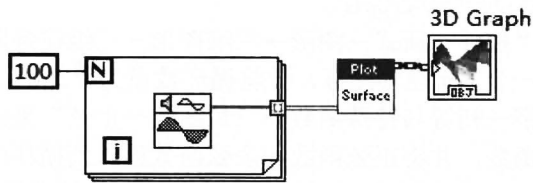


图 7-28 三维曲面程序框图

7.3.3 三维参数图

三维参数图用于显示一些封闭的三维空间曲面，这是三维曲面图形所无法实现的。三维参数图有三个主要数据输入端口，分别用来输入 x 矩阵、y 矩阵和 z 矩阵，其数据类型均为二维数组。三维参数图的显示效果如图 7-29 所示。

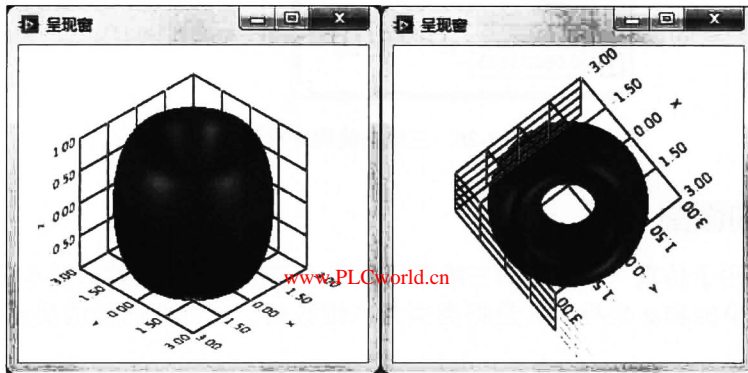


图 7-29 三维参数图

以绘制一个环形封闭曲面为例，其制作步骤如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→三维参数图形”来添加该控件。
- ③ 在后面板，添加两个 For 循环，以构造一个双重循环结构，再将循环数设为“100”。
- ④ 以此在内、外循环层放置正弦信号和余弦信号图标，以生成矩阵参数。
- ⑤ 为产生合理的数据，在内、外层循环中添加相应的乘函数和数值常量，经连线程序框图如图 7-30 所示。

框图如图 7-30 所示。

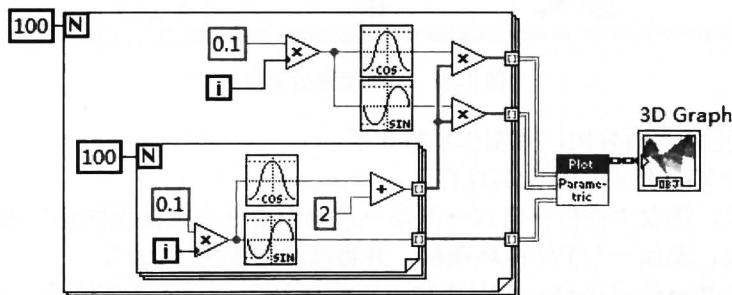


图 7-30 三维参数图程序框图



运行该程序，则其显示效果如图 7-29 所示。

7.3.4 其他三维曲面图

其他的三维曲面图包括散点图、条形图、饼图、杆图、带状图、等高线图、箭头图、彗星图、曲面图、网格图及瀑布图。此类曲面图的新特点是，再加 2 个新显示选项，即投影选板和颜色梯度选板。在一定图形方式下，颜色梯度选板会更换为颜色选板。

(1) 散点图及制作

散点图是一种通过点来反映数据特征的图，它主要有二个数据输入端口，分别用来输入 x 向量和 y 向量，其显示效果如图 7-31 所示。

散点图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→散点”来添加该控件。
- ③ 在后面板，添加两个 For 循环来构造一个双重循环结构，再设置好循环数。
- ④ 添加正弦信号、余弦信号和乘、除函数，经连线程序框图如图 7-32 所示。

运行该程序，则可见到图 7-31 的显示效果。

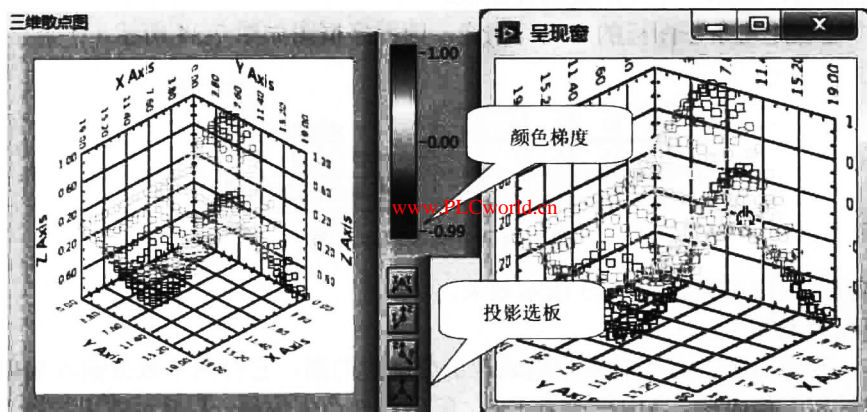


图 7-31 散点图

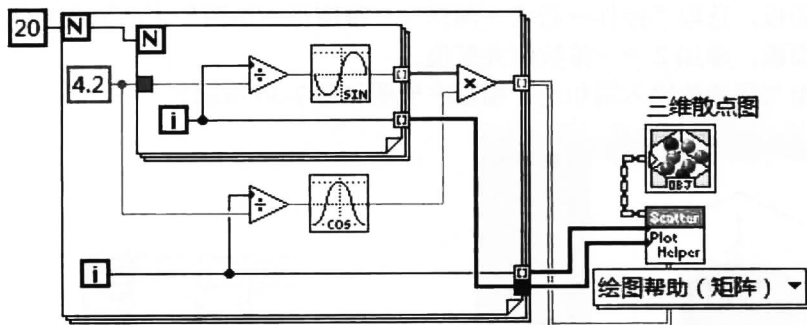


图 7-32 散点图程序框图

(2) 条形图及制作

条形图是一种以彩色条块来反映数据特征的图，它主要有两个数据输入端口，分别用来输入 y 向量和 z 矩阵，其显示效果如图 7-33 所示。

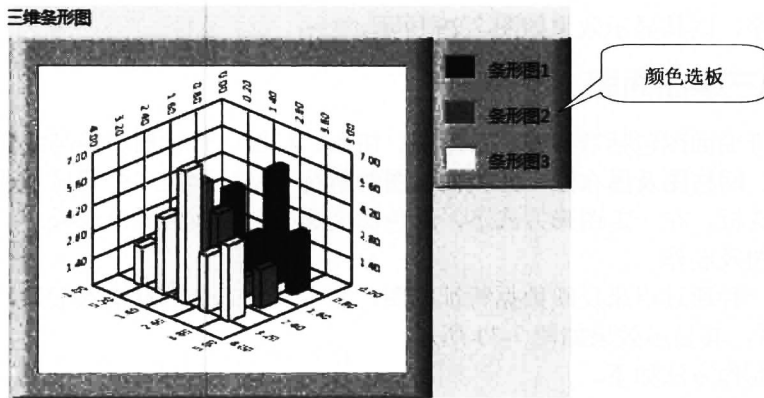


图 7-33 条形图

条形图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→条形”来添加该控件。
- ③ 在后面板，添加一个二维数组并赋值。
- ④ 将二维数组与条形图标的 z 矩阵相连，则程序框图如图 7-34 所示。

www.91HMI.com

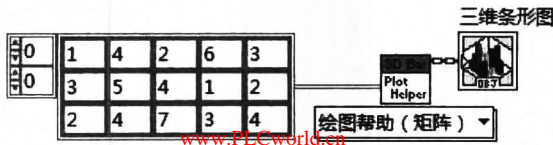


图 7-34 条形图程序框图

运行该程序，则可见到图 7-33 所示效果。

(3) 饼图及制作

饼图是一种按数据占有率将图分割成若干个扇形的图，它有两个数据输入端口，分别用来输入 x 向量和偏移向量。当偏移量为 1 时，扇形向外产生偏移，其显示效果如图 7-35 所示。饼图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→饼图”来添加该控件。
- ③ 在后面板，添加 2 个一维数组并赋值。
- ④ 将数组与饼图的输入端相连，则程序框图如图 7-36 所示。

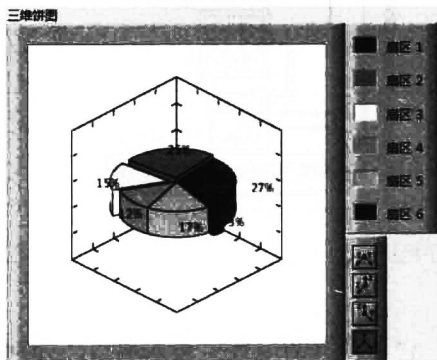


图 7-35 饼图

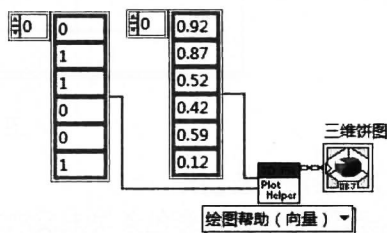


图 7-36 饼图程序框图

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



运行该程序，则可见到图 7-35 所示效果。

(4) 杆图及制作

杆图是一种以杆状线段来显示数据特征的图，它有三个主要数据输入端口，分别用来输入 x 向量、 y 向量和 z 向量，其显示效果如图 7-37 所示。

杆图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→杆图”来添加该控件。
- ③ 在后面板，添加 3 个一维数组并赋值。
- ④ 将数组与杆图的向量输入端相连，则程序框图如图 7-38 所示。

运行该程序，则可见到图 7-37 所示效果。

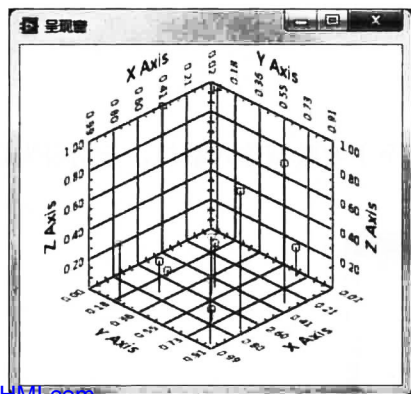


图 7-37 杆图

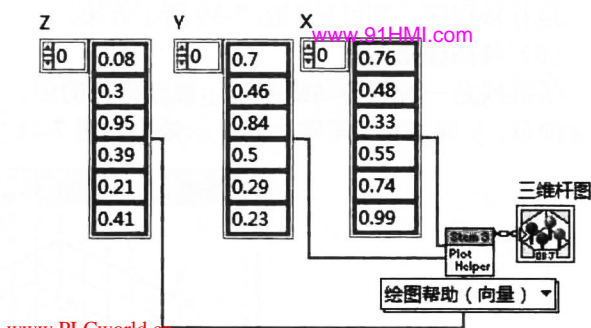


图 7-38 杆图程序框图

(5) 带状图及制作

带状图是一种以带状图形来显示数据特征的图，它有两个主要数据输入端口，分别用来输入 y 向量和 z 矩阵，其显示效果如图 7-39 所示。

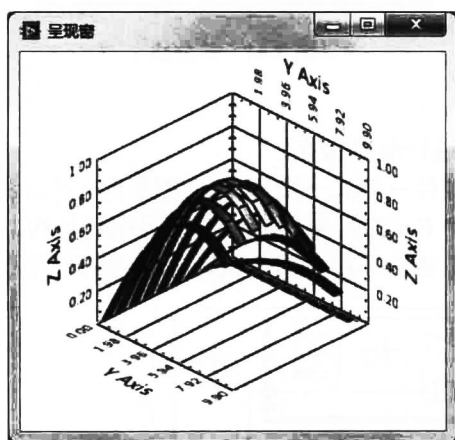


图 7-39 带状图

带状图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→带状”来添加该控件。



- ③ 在后面板，添加两个 For 循环来构造一个双重循环结构，再设置好循环数。
- ④ 添加乘、除函数、正弦信号和数值常量，连线后的程序框图如图 7-40 所示。

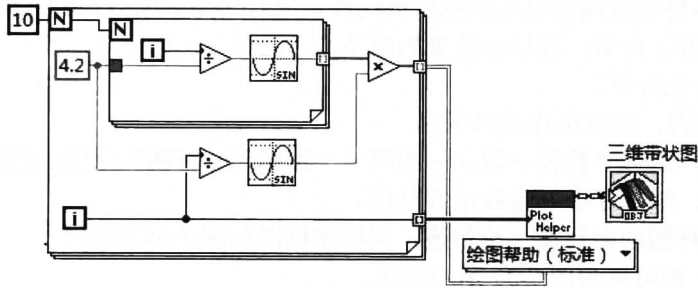


图 7-40 带状图程序框图

运行该程序，则可见到图 7-39 所示效果。

(6) 等高线及制作

www.91HMI.com

等高线是一种以等高线来显示数据特征的图，它有三个主要数据输入端口，分别用来输入 x 向量、 y 向量和 z 矩阵，其显示效果如图 7-41 所示。

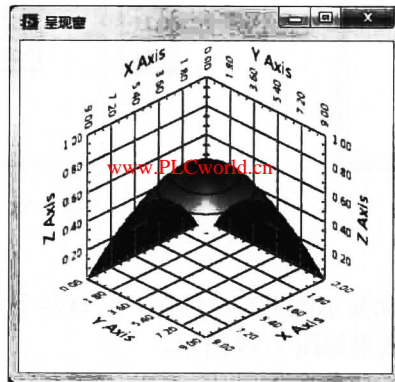


图 7-41 等高线

等高线的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→等高线”来添加该控件。
- ③ 在后面板，添加两个 For 循环来构造一个双重循环结构，再设置好循环数。
- ④ 添加乘、除函数、正弦信号和数值常量，连线后的程序框图如图 7-42 所示。

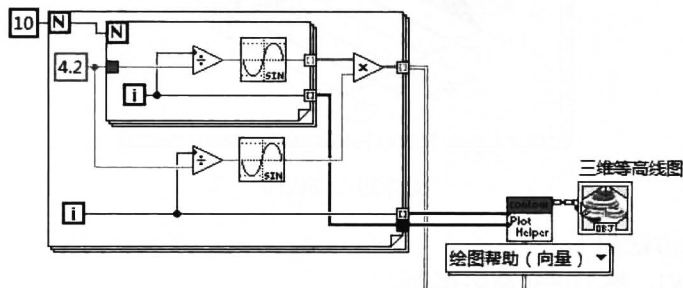


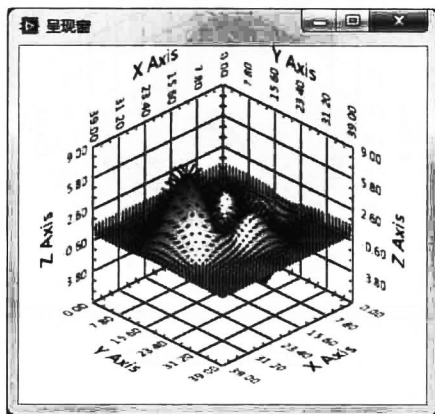
图 7-42 等高线程序框图



运行该程序，则可见到图 7-41 所示效果。

(7) 箭头图及制作

箭头图是一种以等高线来显示数据特征的图，它有 6 个主要数据输入端口，分别用来输入 x 矩阵、y 矩阵、z 矩阵、Nx 矩阵、Ny 矩阵和 Nz 矩阵，其显示效果如图 7-43 所示。



www.91HMI.com

图 7-43 箭头图

箭头图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→箭头”来添加该控件。
- ③ 在后面板，添加 6 个二维数组并赋值。
- ④ 将数值分别与箭头图标的各向量输入端相连，则程序框图如图 7-44 所示。

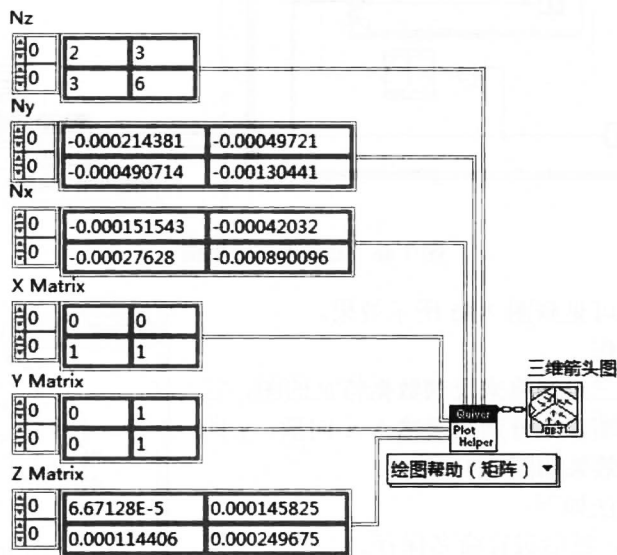


图 7-44 箭头图程序框图

运行该程序，则可见到图 7-43 所示效果。

(8) 曲面图及制作

曲面图是一种以三维曲面来反映数据特征的图，它有三个主要数据输入端口，分别用来



输入 x 向量、y 向量和 z 矩阵，其显示效果如图 7-45 所示。

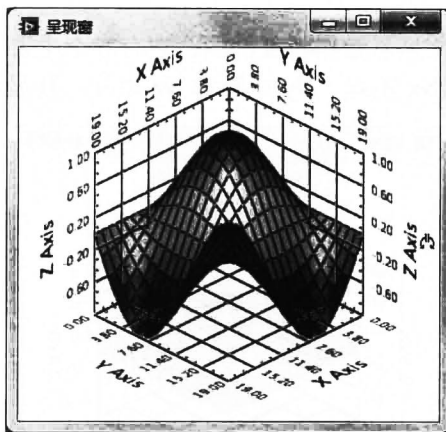


图 7-45 曲面图

www.91HMI.com

曲面图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→曲面”来添加该控件。
- ③ 在后面板，添加两个 For 循环来构造一个双重循环结构，再设置好循环数。
- ④ 添加乘、除函数、正弦信号和数值常量，连线后程序框图如图 7-46 所示。

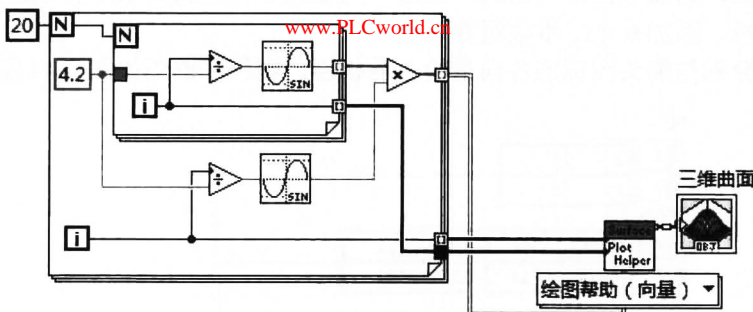


图 7-46 曲面图程序框图

运行该程序，则可见到图 7-45 所示效果。

(9) 网格图及制作

网格图是一种以三维网格来反映数据特征的图，它有三个主要数据输入端口，分别用来输入 x 向量、y 向量和 z 矩阵，其显示效果如图 7-47 所示。

网格图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→网格”来添加该控件。
- ③ 在后面板，添加两个 For 循环来构造一个双重循环结构，再设置好循环数。
- ④ 添加乘、除函数、正弦信号和数值常量，连线

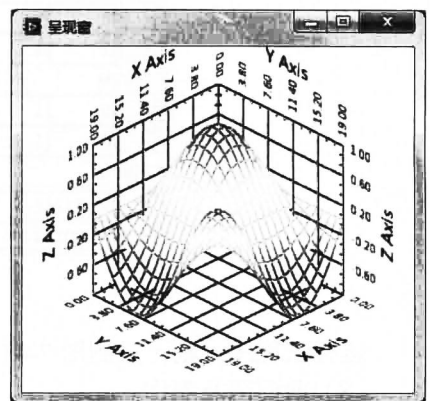


图 7-47 网格图



后程序框图如图 7-48 所示。

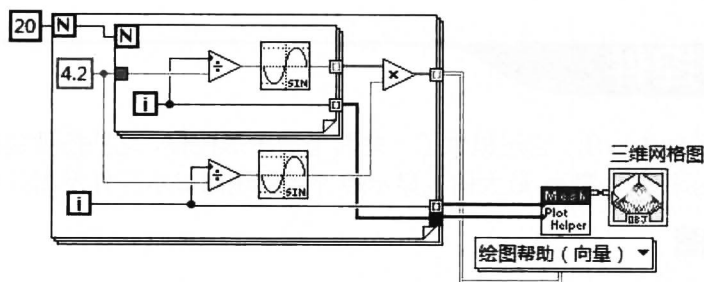


图 7-48 网格图程序框图

运行该程序，则可见到图 7-47 所示效果。

(10) 瀑布图及制作

瀑布图是一种呈瀑布状的图来反映数据特征，它有三个主要数据输入端口，分别用来输入 x 向量、y 向量和 z 矩阵，其显示效果如图 7-49 所示。

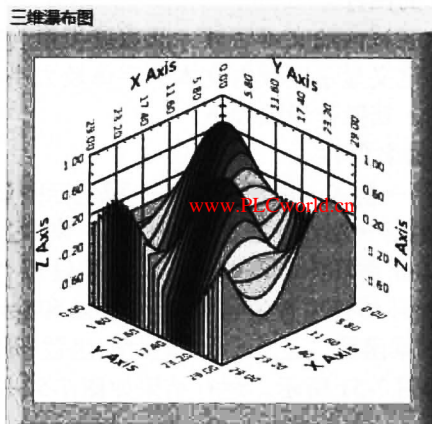


图 7-49 瀑布图

瀑布图的制作方法如下。

- ① 新建一个 VI，然后进行命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图形→瀑布”来添加该控件。
- ③ 在后面板，添加两个 For 循环来构造一个双重循环结构，再设置好循环数。
- ④ 添加乘、除函数、正弦信号和数值常量，连线后程序框图如图 7-50 所示。

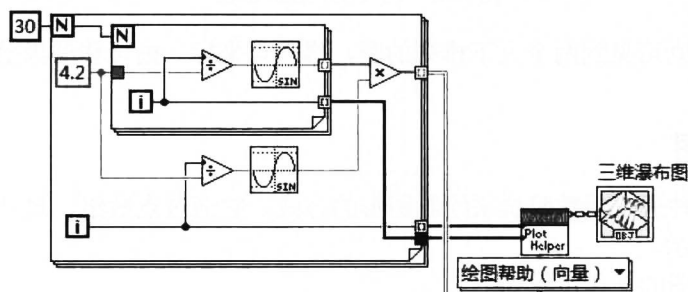


图 7-50 瀑布图程序框图



运行该程序，则可见到图 7-49 所示效果。

7.4 图形控件

LabVIEW 为了便于应用，它还提供了一些其他的图形控件，这些控件包括极坐标图显示控件、Simth 图显示控件、最小-最大曲线显示控件、雷达图显示控件及图片等。

7.4.1 极坐标图

极坐标图是一种建立在极坐标系统上的绘图方式，它通过数据数组、尺寸、坐上角点等参数来控制图形的显示。

其中：

数据数组，用于输入绘制极坐标图的数据。

尺寸，用于定义显示区域的宽度和高度。

坐上角点，用于定义显示区域坐上角点的坐标。

图片，用于输入想要添加的图片。

所需字体，用于选择显示区域内的文本字体，其数据类型为字体枚举型。

用户定义字体，用于自行定义显示区域内文本的字体及大小。

以绘制 $r=|\sin\theta|$ 曲线为例，其操作步骤如下。

① 新建一个 VI，并进行命名保存。

② 在前面板，选取“控件→新式→图形→控件→极坐标图显示控件”来添加该控件。

③ 选取“控件→新式→图形→控件→数据类型→点”来添加点设置控件。

④ 在后面板，添加一个 For 循环图标，并将循环次数设为“360”。

⑤ 为配合 For 循环，在循环体外用数值常量 2π 、360 和除函数构建一个弧度常数。

⑥ 在循环体内，依次添加乘函数、正弦函数、绝对值函数和捆绑图标，从而形成 $r=|\sin\theta|$ 的数据，经连线其程序框图如图 7-51 所示。运行结果如图 7-52 所示。

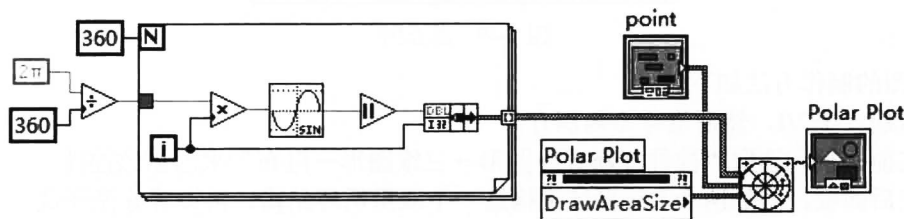


图 7-51 曲线绘制程序框图

运行该程序，则可见到两个上下排列的圆，修改点坐标，则可移动极坐标图。绘制效果如图 7-52 所示。

7.4.2 Simth 图

Simth 图是一种通过复数阻抗来绘制图形的方式，它通过点数组、尺寸、坐上角点等参数来控制图形的显示。

以绘制 Simth 图的操作步骤如下。

① 新建一个 VI，并加以命名保存。

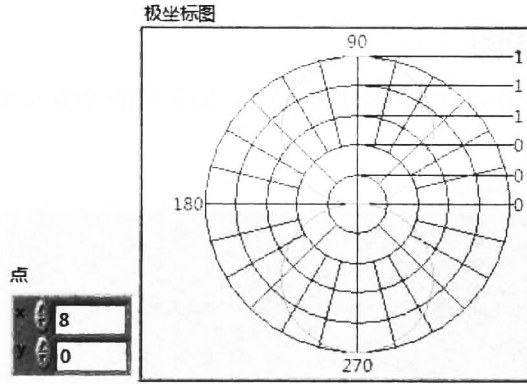


图 7-52 极坐标图

- ② 在前面板，选取“控件→新式→图形→控件→Simth 图显示控件”来添加该控件。
- ③ 选取“控件→新式→图形→控件→数据类型→点”来添加点设置控件，并为“部分 Simth 图”端口添加一个输入控件。
- ④ 在后面板，添加一个 For 循环图标，并将循环次数设为“5”。
- ⑤ 为配合 For 循环，在循环内依次添加数值常量、乘函数、平方根函数和除函数来构造复数阻抗值，经连线其程序框图如图 7-53 所示。

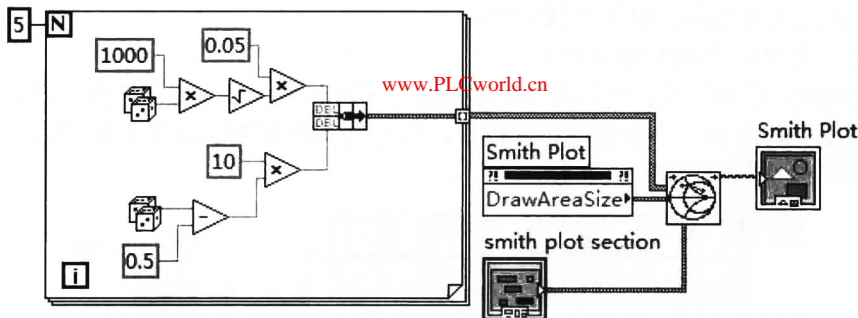


图 7-53 Simth 图程序框图

运行该程序，通过调整显示位置可见不同的观察效果，参见图 7-54 所示。

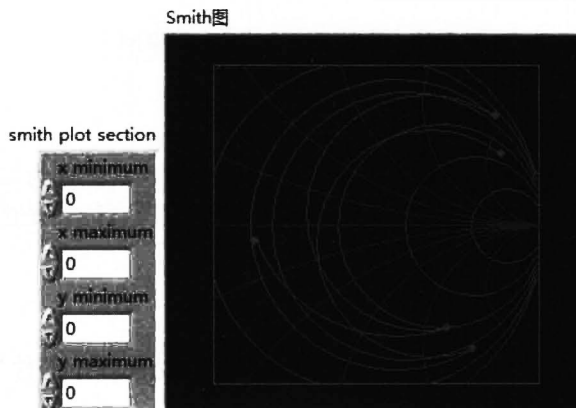


图 7-54 Simth 图



7.4.3 最小-最大曲线图

最小-最大曲线图是一种以点或线来反应最小、最大值的绘图方式，其显示效果如图 7-55 所示。

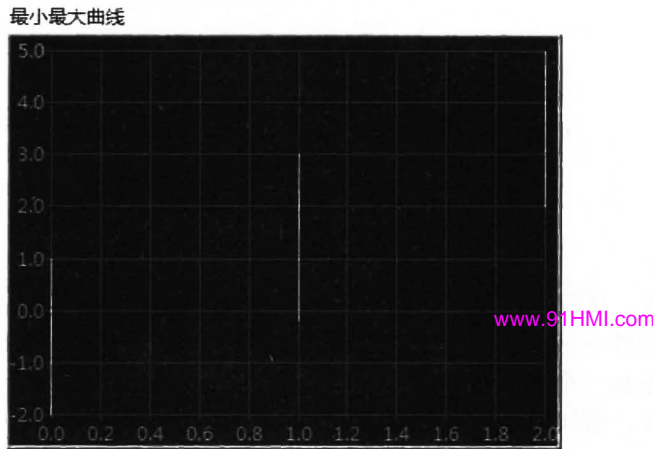


图 7-55 最小-最大曲线图

绘制最小-最大曲线图的操作步骤如下。

- ① 新建一个 VI，并加以命名保存。
 - ② 在前面板，选取“控件→新式→图形→控件→最小-最大曲线显示控件”来添加该控件。
 - ③ 为数据端口添加一个常量或输入控件，经连线其程序框图如图 7-56 所示。
- 运行该程序，则可见到预期效果。

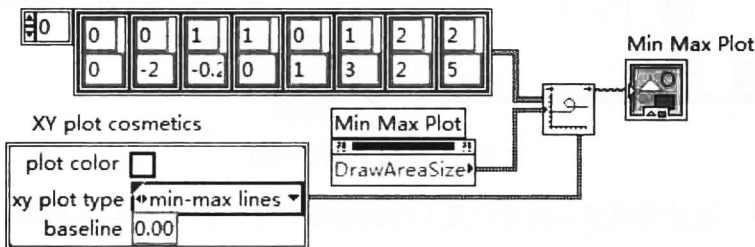


图 7-56 最小-最大曲线图程序框图

7.4.4 雷达图

雷达图是一种以雷达效果来反映数据特征的绘图方式，其数据为二维数组，每行代表一条曲线。VI 可在曲线对应的 y 轴上绘制每个索引，曲线中所有的点是由一条直线来连接，其显示效果如图 7-57 所示。

绘制雷达图的操作步骤如下。

- ① 新建一个 VI，并加以命名保存。
- ② 在前面板，选取“控件→新式→图形→控件→雷达图显示控件”来添加该控件。
- ③ 切换至后面板，添加一个 For 循环图标。
- ④ 在 For 循环内，添加一个数组最大值与最小值函数和一个数值常量“0”，以控制雷达



图的显示范围。

⑤ 切换到前面板，选取“控件→新式→列表、表格和树→表格”来添加表格控件。

⑥ 用鼠标右键单击“表格”控件，在快捷菜单中选取“显示项”下的“行首”和“列首”选项，然后按图 7-57 所示输入数据。

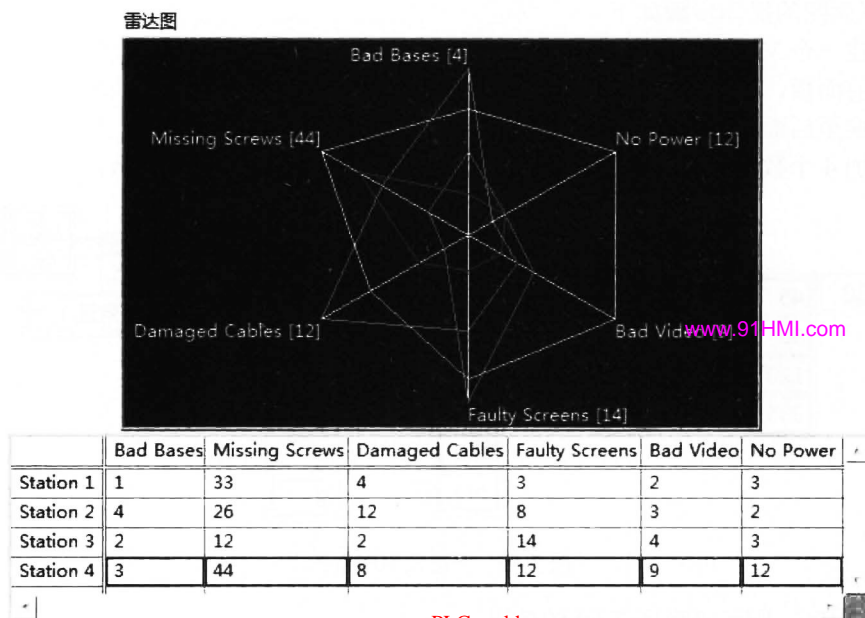


图 7-57 雷达图

⑦ 为了提供绘制的数据，切换回后面板。依次添加“十进制数字字符串至数值转换”函数和“二维数组转置”函数。

⑧ 用鼠标右键单击“表格”图标，在快捷菜单中选取“创建→属性节点→行首字符串[]”选项，经连线其程序框图如图 7-58 所示。

运行该程序，则可见到预期效果。

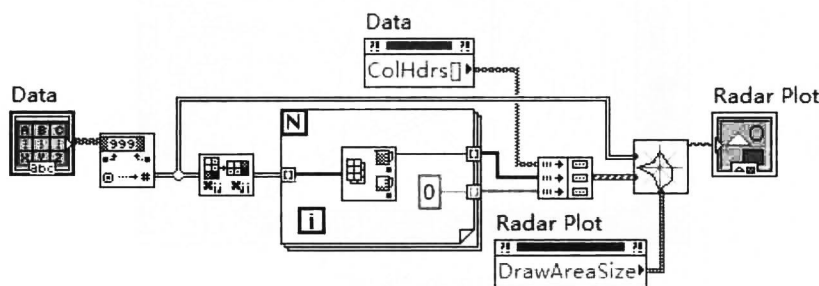


图 7-58 雷达图程序框图

7.5 其他图形展示

在其他图形显示中，主要包括罗盘图、误差线、羽状图及 XY 曲线矩阵。



7.5.1 罗盘图

罗盘图是一种由罗盘图的中心发出的向量，它主要有两个参数，即角度向量和半径向量。通过对引用的使用，可在同一图形上生成多条曲线。

绘制罗盘图的操作步骤如下。

- ① 新建一个 VI，并加以命名保存。
- ② 在前面板，选取“控件→新式→图形→罗盘图”来添加该控件。
- ③ 切换至后面板，复制一次“绘图帮助（向量）”图标。
- ④ 添加 4 个数组常量并进行赋值，经连线其程序框图如图 7-59 所示。

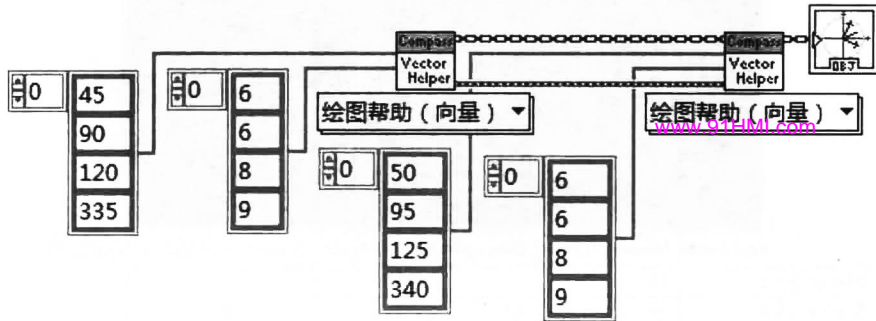


图 7-59 罗盘图程序框图

运行该程序，则可见到图 7-60 的效果。 www.PLCworld.cn

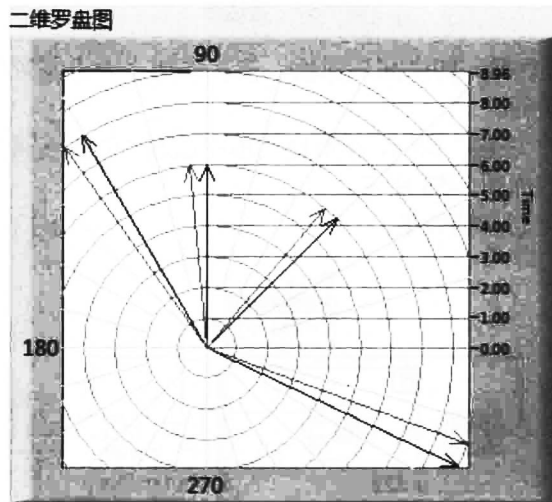


图 7-60 罗盘图

7.5.2 误差线

误差线是一种绘制线条图上各点误差百分比或向量的绘图方式，其主要参数有两个，即 x 向量和 y 向量。

绘制误差线的操作步骤如下。

- ① 新建一个 VI，并加以命名保存。



- ② 在前面板，选取“控件→新式→图形→误差线”来添加该控件。
- ③ 切换至后面板，复制一次“绘图帮助（百分比）”图标。
- ④ 添加 6 个数组常量并进行赋值，经连线其程序框图如图 7-61 所示。

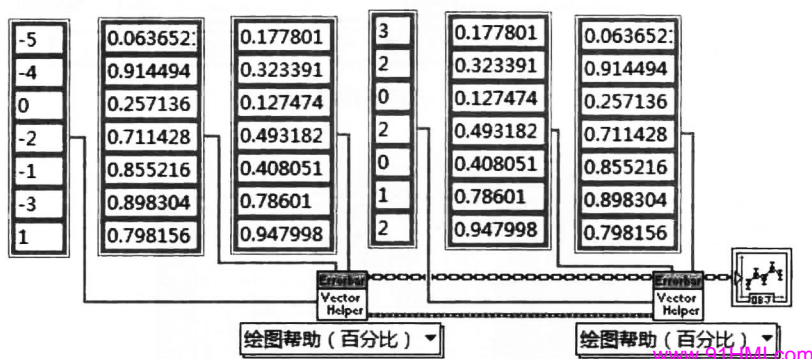


图 7-61 误差线程序框图

运行该程序，则可见到图 7-62 的效果。

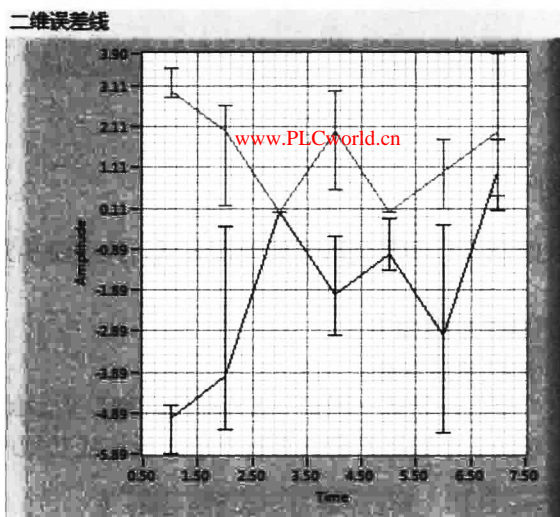


图 7-62 误差线

7.5.3 羽状图

羽状图是一种绘制由水平坐标轴上的零点所发出的均匀分布的点向量线的绘图方式，其主要参数有两个，即 x 向量和 y 向量。通过对引用的使用，可在同一图形上生成多条曲线。

绘制误差线的操作步骤如下。

- ① 新建一个 VI，并加以命名保存。
- ② 在前面板，选取“控件→新式→图形→羽状图”来添加该控件。
- ③ 切换至后面板，复制一次“绘图帮助（百分比）”图标。
- ④ 添加 4 个数组常量并进行赋值，经连线其程序框图如图 7-63 所示。

运行该程序，则可见到图 7-64 的效果。

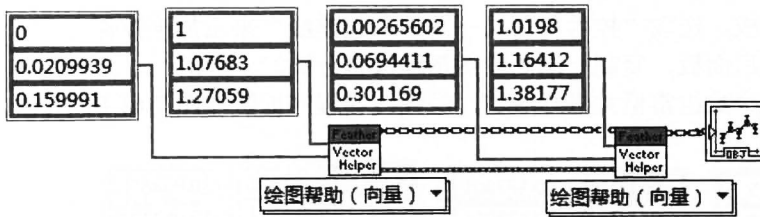


图 7-63 羽状图程序框图

羽状图

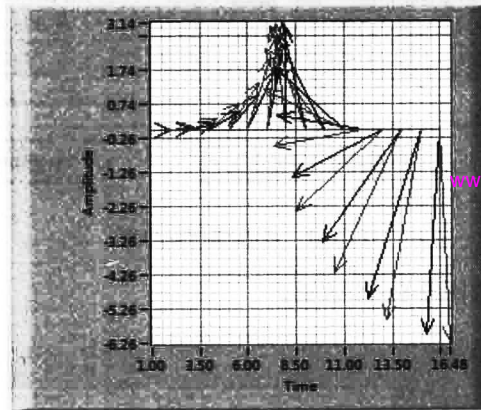


图 7-64 羽状图

www.PLCworld.cn

7.5.4 XY 曲线阵列

XY 曲线阵列是一种绘制多行和多列散点的绘图方式，其主要参数有两个，即 x 阵列和 y 阵列。

绘制 XY 曲线阵列的操作步骤如下。

- ① 新建一个 VI，并加以命名保存。
- ② 在前面板，选取“控件→新式→图形→XY 曲线阵列”来添加该控件。
- ③ 切换至后面板，添加 2 个数组常量并进行赋值，经连线其程序框图如图 7-65 所示。

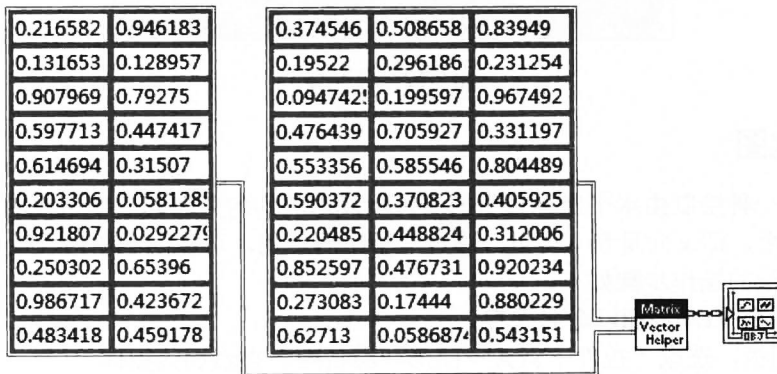


图 7-65 XY 曲线阵列图程序框图

运行该程序，则可见到图 7-66 的效果。

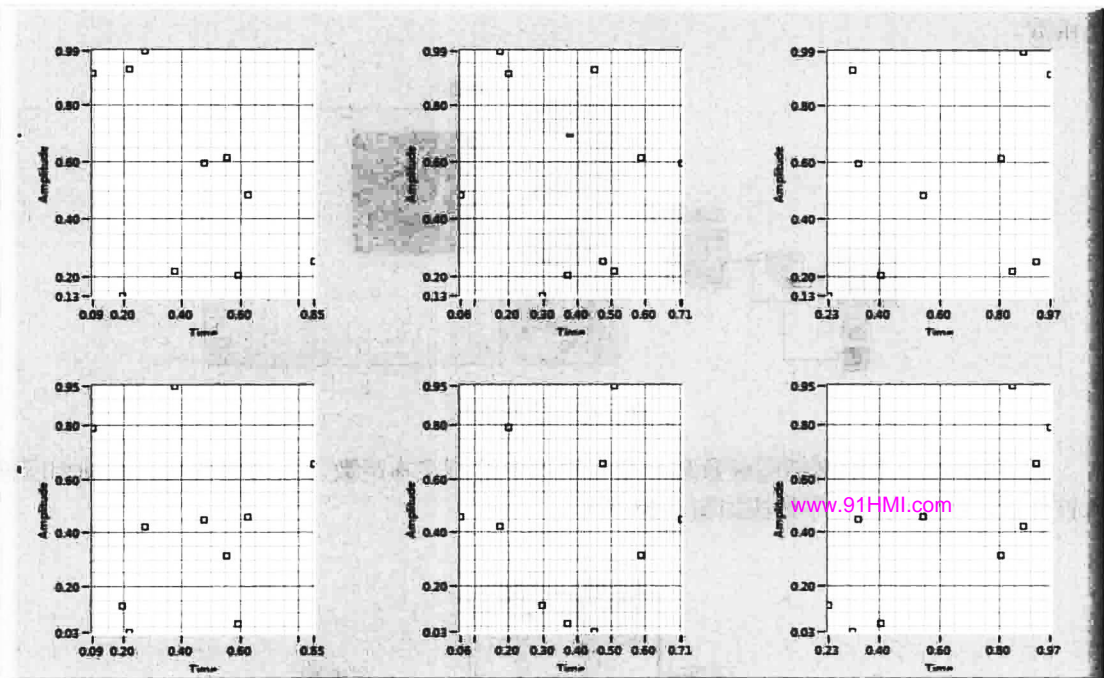


图 7-66 XY 曲线阵列图

7.6 图片控件

LabVIEW 除了提供图表控件外, 还提供有图片控件, 主要用于绘制简单的图形及显示图片。

7.6.1 二维图形绘制

二维图片函数能在二维平面内绘制点、直线、矩形、椭圆及圆弧等, 此外它还能对图形进行填充处理。

(1) 单一图形绘制

以绘制矩形为例, 其绘制方法如下。

- ① 新建一个 VI, 并命名保存。
- ② 在前面板, “控件→新式→图形→控件→二维图片” 来添加二维图片控件。
- ③ 在后面板, 选取“函数→编程→图形与声音→图片函数→绘制矩形” 来添加绘制矩形函数。

④ 为绘制矩形函数的参数端口添加相应的输入控件, 连线后程序框图如图 7-67 所示。在设置好图形参数后运行该程序, 则可见到预期效果。

(2) 多图形绘制

在绘制单一图形的基础上, 将图形进行串联即可形成多个图形并列的效果。

以绘制矩形为例, 其绘制方法如下。

- ① 新建一个 VI, 并命名保存。
- ② 在前面板, “控件→新式→图形→控件→二维图片” 来添加二维图片控件。
- ③ 在后面板, 选取“函数→编程→图形与声音→图片函数→绘制矩形” 来添加绘制矩



- ① 新建一个 VI，并命名保存。
- ② 在前面板，“控件→新式→图形→三维图片”来添加三维图片控件。
- ③ 在后面板，选取“函数→编程→图形与声音→三维图片控件→几何→创建盒”来添加该函数。

④ 选取“函数→编程→图形与声音→三维图片控件→对象→创建对象”来添加该函数，再添加相应的常量，经连线其程序框图如图 7-70 所示。

运行该程序，则可见到图 7-71 所示效果。

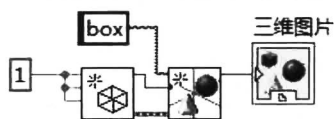


图 7-70 正方体绘制程序框图



图 7-71 正方体绘制显示效果

(2) 多形体绘制

在绘制单一图形的基础上，将图形进行串联即可形成多图形并列的效果。此外，使用变形函数也有助于形体间的位置关系。变形函数主要包括平移处理、缩放处理、旋转处理及旋转坐标轴等。

以绘制一个圆顶小屋为例，其绘制方法如下。

- ① 新建一个 VI，并命名保存。
- ② 在前面板，选取“控件→新式→图形→三维图片”来添加三维图片控件。
- ③ 在后面板，添加盒绘制函数和圆锥绘制函数。
- ④ 再添加 2 个创建对象函数，以显示这 2 个形体。
- ⑤ 为使 2 个形体不相互叠加，选取“函数→编程→图形与声音→三维图片控件→变形→设置平移”来使形体间产生位移，并为其添加一个输入控件。
- ⑥ 为使 2 个形体能同窗口显示，选取“函数→编程→图形与声音→三维图片控件→对象→添加对象”来将 2 个形体串联在一起。经连线，程序框图如图 7-72 所示。

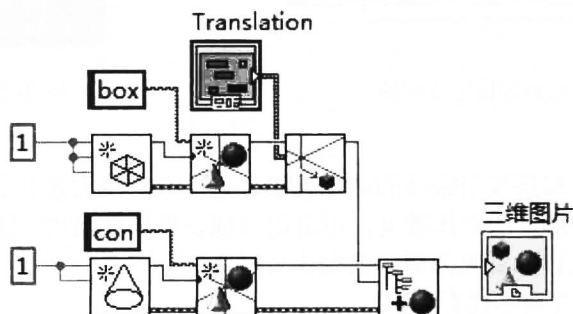


图 7-72 形体组合程序框图



在设置好平移参数后，运行该程序则可达到图 7-73 所示效果。



图 7-73 形体组合效果图

(3) 应用纹理

为了使制作的三维形体更逼真，可使用纹理函数进行贴附处理。贴附的纹理素材可以是 BMP、JPEG 和 PNG 文件。

以制作一个地球为例，其绘制方法如下。

① 新建一个 VI，并命名保存。

② 在前面板，选取“控件→新式→图形→三维图片”来添加三维图片控件。

③ 在后面板，添加一个创建球面函数。

④ 再依次添加“创建对象”函数和“设置绘图样式”函数。

⑤ 为了贴附纹理，选取“函数→编程→图形与声音→三维图片控件→助手→应用纹理”来进行纹理贴附。

⑥ 为了提供纹理，选取“函数→编程→图形与声音→图形格式→读取 JPEG 文件”并为 JPEG 文件路径赋值。经连线，其程序框图如图 7-74 所示。

运行该程序，则可见到图 7-75 所示效果。

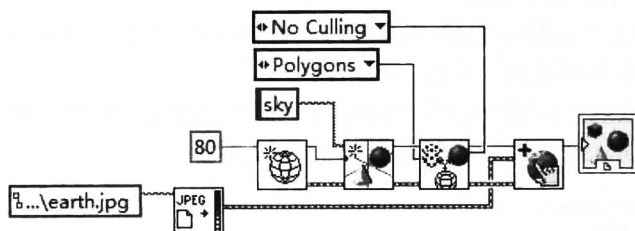


图 7-74 纹理贴附程序框图

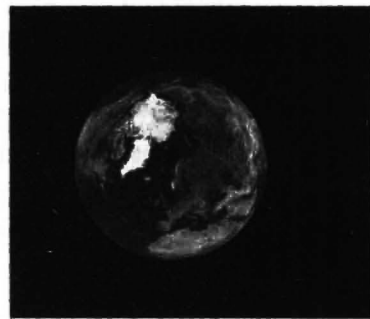


图 7-75 地球纹理贴附效果

(4) 动画制作

动画在任何软件中都是吸引眼球的热点元素，LabVIEW 也向设计者提供了三维动画的制作功能。为了简化三维形体的制作难度，可引用其他软件所建造的三维模型。

以一架 F14 战机飞行的动画为例，其制作方法如下。

① 新建一个 VI，并命名保存。

② 在前面板，选取“控件→新式→图形→三维图片”来添加三维图片控件。



③ 在后面板，选取“函数→编程→图形与声音→加载文件→加载 VRML 文件”来导入 VRML 创建的三维模型，并为其添加一个指向 F14 战机的路径常量。

④ 选取“函数→编程→图形与声音→三维图片控件→变形→设置缩放”函数，以便同乘函数相配合来调整飞机的大小。

⑤ 添加一个 For 循环，并设置好循环次数。

⑥ 在循环体内，依次添加一个设置平移图标和一个旋转 Z 轴图标以调整飞机的位置。

⑦ 为了达到动画效果，在循环体内增添一个 100 毫秒的延时函数。经连线，其程序框图如图 7-76 所示。

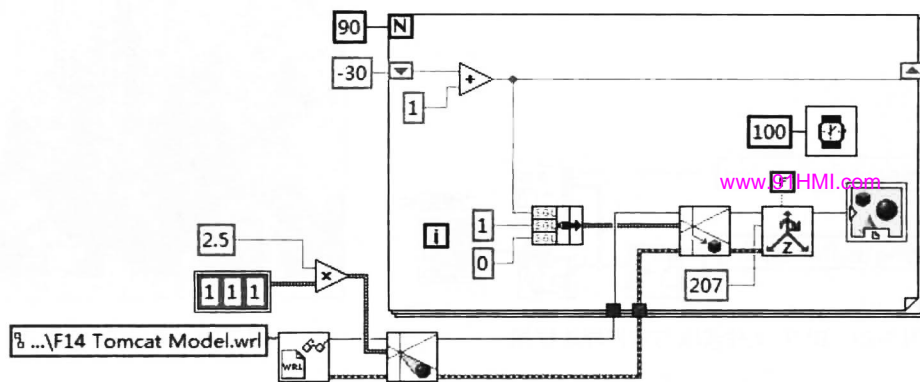


图 7-76 三维动画程序框图

运行该程序，则可见到图 7-77 的动画效果。

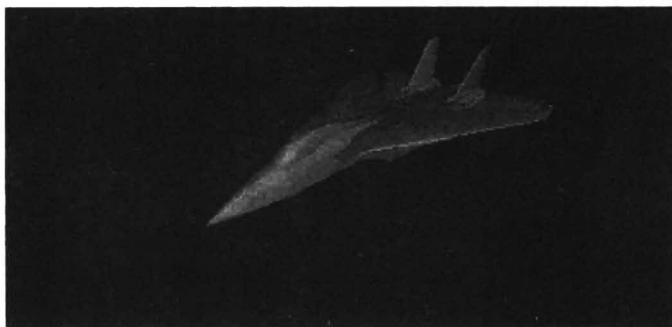


图 7-77 三维动画效果

7.6.3 二维图片读写

LabVIEW 提供了对二维图片文件读写的函数，所支持的文件格式有 BMP 格式、JPEG 格式及 PNG 格式。

(1) BMP 文件读取与转换

BMP 格式是一种非压缩的图片文件格式，其特点是在读写时无需压缩或解压处理，不足之处在于占用存储空间较大。

读取并转换 BMP 文件的操作方法如下。

① 新建一个 VI，然后命名保存。

② 在前面板，“控件→新式→图形→控件→二维图片”来添加二维图片控件。



③ 在后面板, 选取“函数→编程→图形与声音→图形格式→读取 BMP 文件”来添加读取 BMP 文件的图标, 以读取 BMP 文件。

④ 再添加一个写入 JPEG 文件图标, 以存储 JPEG 文件。

⑤ 选取“函数→编程→图形与声音→图片函数→绘制平化像素图”来添加该图标。

⑥ 添加 2 个路径常量并赋值, 连线后程序框图如图 7-78 所示。

运行该程序, 则会见到图 7-79 所示效果。



图 7-78 BMP 文件读取与转换程序框图



图 7-79 显示 BMP 图片

(2) JPEG 文件读取与转换

JPEG 格式是一种有高压缩比的图片文件格式, 其特点是占用存储空间较小, 但在读写时需进行压缩或解压处理。

读取并转换 JPEG 文件的操作方法如下。

① 新建一个 VI, 然后命名保存。

② 在前面板, “控件→新式→图形→控件→二维图片”来添加二维图片控件。

③ 在后面板, 选取“函数→编程→图形与声音→图形格式→读取 JPEG 文件”来添加读取 JPEG 文件的图标, 以读取 JPEG 文件。

④ 再添加一个写入 BMP 文件, 以存储 BMP 文件。

⑤ 选取“函数→编程→图形与声音→图片函数→绘制平化像素图”来添加该图标。

⑥ 添加 2 个路径常量并赋值, 连线后程序框图如图 7-80 所示。

运行该程序, 则会见到图 7-81 所示效果。

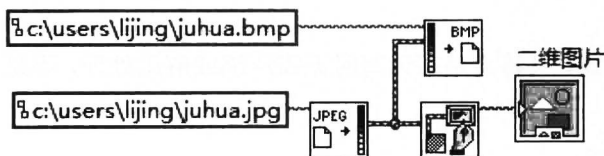


图 7-80 JPEG 文件读取与转换程序框图

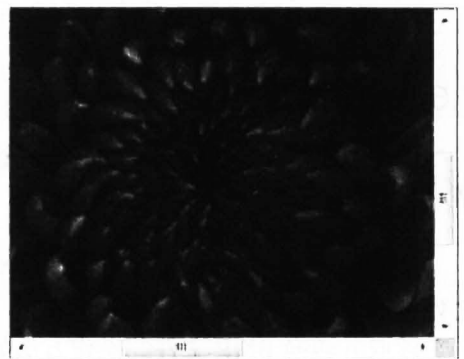


图 7-81 显示 JPEG 图片



7.7 综合实例：多曲线显示控制

在虚拟仪表中，采集的数据往往不止一个通道的，为了便于对相关曲线的在线分析，对曲线的显示控制功能是必不可缺的。解决此类问题的基本思想是：选取激活的曲线，然后对其可见性加以控制，其操作界面如图 7-82 所示。

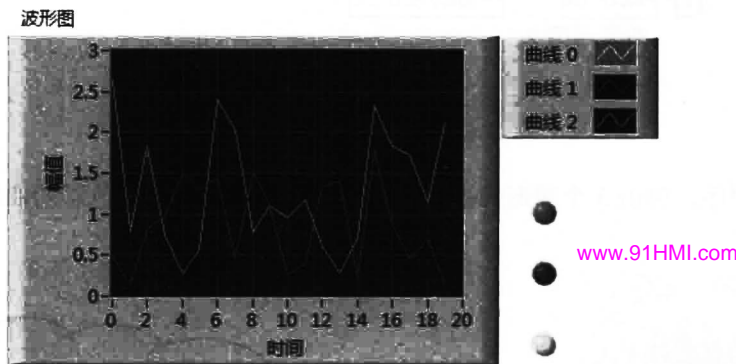


图 7-82 曲线屏蔽控制

要实现多曲线的显示控制，其操作步骤如下。

- ① 新建一个 VI，然后加以命名保存。
- ② 在前面板，选取“控件→新式→图形→波形图”来添加一个波形图控件。
- ③ 添加 3 个圆形指示灯，并将其转换为输入控件。修改 3 个控件的开关颜色，使之与曲线的颜色相一致。
- ④ 切换到后面板，添加一个 For 循环图标，并将循环数设为“20”。
- ⑤ 在循环内，添加 3 个随机函数、3 个乘函数和 3 个数值常量并赋值，以产生模拟数据。
- ⑥ 选取“函数→Express→信号操作→合并信号”来添加合并信号图标，再将其分别与波形图控件及 3 个数值信号相连。
- ⑦ 在循环内，添加一个延时图标，并设置好延时值。
- ⑧ 切换到前面板，用鼠标右键单击波形图，从快捷菜单选取“创建→属性节点→曲线→可见”来添加一个可见节点。
- ⑨ 再由快捷菜单选取“创建→属性节点→活动曲线”来添加一个属性节点。从而与曲线控件节点、控制按键及数值常量构成一个曲线屏蔽控制单元。
- ⑩ 再完成另外两组屏蔽曲线形式的单元，经连线其程序框图如图 7-83 所示。

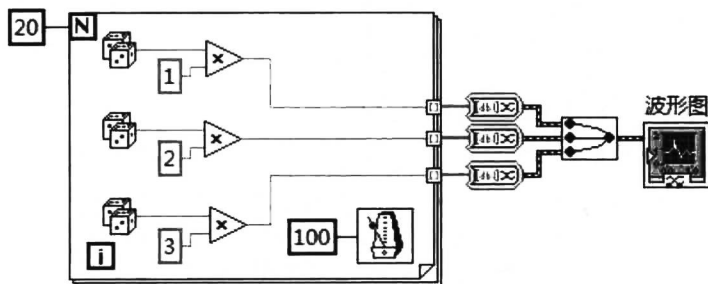


图 7-83

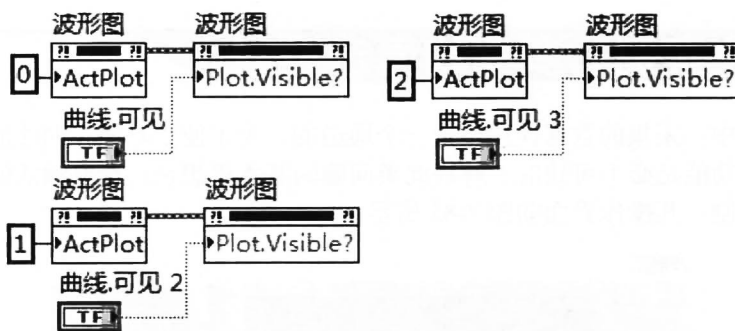


图 7-83 曲线屏蔽程序框图

运行该程序，单击 3 个圆形的控制按钮可以屏蔽或显示相应色彩的曲线。



在数据表达与显示王国，各式各样的显示方式可以帮助我们解决各类显示方面的难题。

www.PLCworld.cn

www.91HMI.com



第 2 部分

提高篇

第 8 章

LabVIEW文件I/O

www.91HMI.com

www.PLCworld.cn

在与外界进行数据交换时，需要借助文件来传递，为此 LabVIEW 提供了强大的文件 I/O 功能，主要包括读/写文本文件、读/写二进制文件、读/写电子表格文件、读/写测量文件、读/写波形文件、读/写配置文件、读/写数据记录文件等。

【本章导读】

- ◆ 常用文件操作函数
- ◆ 常用文件操作
- ◆ 特殊文件操作
- ◆ 其他文件操作
- ◆ 高级文件工具等

文件是存储在各种存储介质上数据的集合，文件的输入、输出就是对存储介质上的数据进行保存和读取。以文件形式存储的数据具有长期保存性，利用 LabVIEW 的相关函数可以使 VI 程序与文件来交换数据。

8.1 常用文件操作函数

8.1.1 文件常量与函数

文件常量主要同目录、路径等相关，其包括路径常量、当前 VI 路径、获取系统目录、



空路径常量、非法路径常量、默认目录、默认数据目录、应用程序目录及临时目录等。

使用文件常量的方法是：在后面板，选取“函数→编程→文件 I/O→文件常量”下的相应函数。

各文件常量的作用如表 8-1 所示。

表 8-1 文件常量作用表

名称	图标	说明
路径常量		用于为程序框图提供常量路径
当前 VI 路径		返回当前 VI 访问的文件路径，如 VI 尚未保存，则函数返回非法路径
空路径常量		用于在创建路径函数中作为新建路径的起点 www.91HMI.com
默认目录		用于没有指定特定保存位置时，LabVIEW 自动保存信息的目录
默认数据目录		用于保存 VI 或函数生成数据的指定目录
VI 库路径		若 VI 生成应用程序，则该函数返回包含该应用程序文件的目录的路径
应用程序目录		用于返回应用程序所在目录的路径 www.PLCworld.cn
临时目录		用于 LabVIEW 存放无需存放在默认目录中的信息
获取系统目录		用于返回当前终端的系统目录

8.1.2 获取文件路径信息

通过对文件常量的使用，可获取文件的路径信息，主要包括当前 VI 路径、系统目录等。

(1) 获取当前 VI 路径

获取当前 VI 路径的方法是，在后面板放置一个当前 VI 路径的图标，然后选取它单击鼠标右键，从快捷菜单中选取“创建→显示控件”，则程序框图如图 8-1 所示。此时，若运行程序，则会显示“非法路径”，如图 8-2 所示。对文件加以命名保存，再运行程序，则会显示出当前 VI 的路径，如图 8-3 所示。

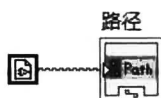


图 8-1 获取当前 VI 路径



图 8-2 非法路径显示

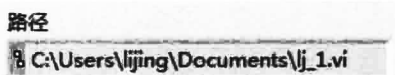


图 8-3 当前 VI 路径显示



(2) 获取其他路径

仿照获取当前 VI 路径的方法，还可获得默认目录、默认数据目录、应用程序目录、临时目录等，其程序框图如图 8-4 所示，相应的输出效果见图 8-5 所示。

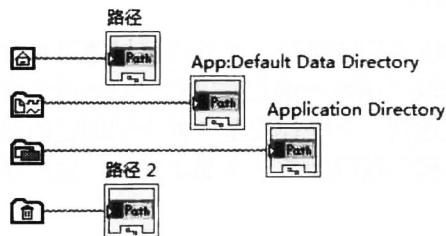


图 8-4 获取系统目录 (一)



图 8-5 显示不同目录路径 (一)

(3) 获取系统目录

获取系统目录的方法是，在后面板放置一个获取系统目录图标，然后为“系统目录”端口创建一个显示控件，为“系统目录类型”端口创建一个输入控件，则程序框图如图 8-6 所示。

运行程序，通过输入控件选取不同的选项，则可得到不同的目录信息，如图 8-7 所示。

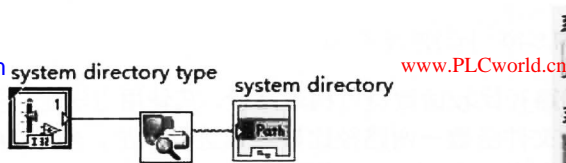


图 8-6 获取系统目录 (二)

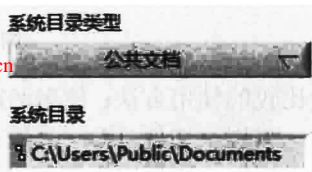


图 8-7 显示不同目录路径 (二)

8.1.3 路径设置与构造

在进行文件读写操作时，需要对操作的文件路径加以设置，其方法是，在后面板选取“函数→编程→文件 I/O→文件常量→路径常量”以进行添加，然后对该路径常量进行赋值。为显示设置效果，可为该路径常量配置一个显示控件，运行后的效果如图 8-8 所示。



图 8-8 设置文件操作路径

不仅用路径常量可以设置路径，在 LabVIEW 中，还提供了创建路径和拆分路径函数辅助构造路径。

创建路径的方法是：在后面板选取“函数→编程→文件 I/O→创建路径”以进行添加，然后为“基路径”端口添加一个常量并进行赋值。再为“名称或相对路径”端口添加一个输入控件，并为其赋值。为了显示路径值，给“添加的路径”端口配置一个显示控件，则程序框图和运行效果如图 8-9 所示。



图 8-9 创建路径

拆分路径的方法是：创建路径是采用合成的方法来构造路径，此外还可以通过拆分的方法来构造路径。拆分路径的方法是：在后面板，选取“函数→编程→文件 I/O→拆分路径”以进行添加，然后为“路径”端口添加一个常量或连到一个路径变量（如上例创建的路径）。为了显示拆分效果，分别为“拆分的路径”和“名称”端口配置一个显示控件，则程序框图和运行效果如图 8-10 所示。

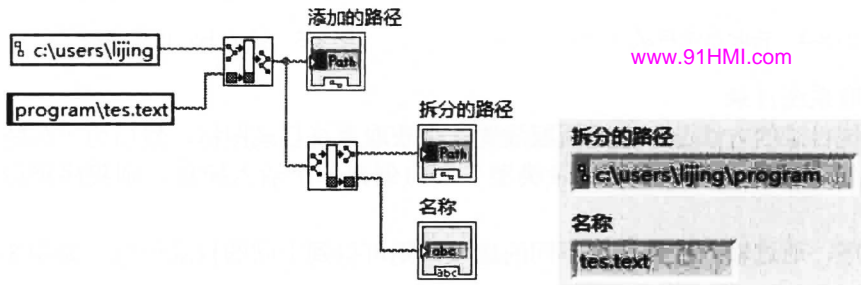


图 8-10 拆分路径（一）

两路径比较的使用方法：使用两路径比较函数也可构建路径，其使用方法是，在后面板，选取“函数→编程→文件 I/O→高级文件函数→两路径比较”以进行添加，然后为各端口添加常量或显示控件，则程序框图和运行效果如图 8-11 所示。

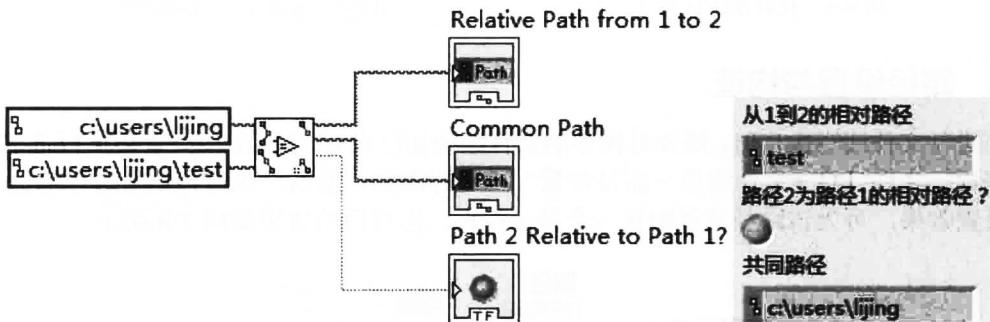


图 8-11 拆分路径（二）

8.2 常用文件操作

8.2.1 文本文件

文本文件(.txt)是一种由 ASCII 码构成的字节流文件，该格式文件的优点是通用性强，可以被记事本、写字板、Word、Excel 等程序直接读取。此外，文本文件极易进行格式转换。



其不足处在于，它不能随机地对文本文件中的指定数据进行访问。

提示

对文本文件的操作涉及文件的打开、文件的创建、文件的读/写及关闭文件等，借助 LABVIEW 提供的相关函数能够方便地实现此类操作。

(1) 读取文本文件

使用“读取文本文件”函数可以便捷地从文本文件中读取 ASCII 码字符，其使用方法是：在后面板选取“函数→编程→文件 I/O→读取文本文件”以进行添加，然后为文件端口添加一个路径常量并进行赋值（设置的路径和文件名应与读取的文件相一致）。为了显示读取的内容，可选取文本端口，并为其添加一个显示控件，其程序框图和运行效果如图 8-12 所示。

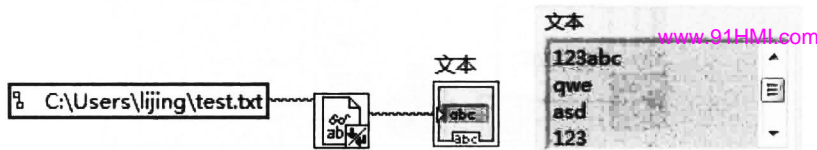


图 8-12 文本文件读取

(2) 生成文本文件

与读取文本文件相近，通过使用“写入文本文件”函数可以轻松生成文本文件，并写入相应的数据信息。生成文本文件的方法是：在后面板选取“函数→编程→文件 I/O→写入文本文件”以进行添加，然后为文件端口添加一个输入控件，以指定生成文件的路径和文件名。在文本端口添加一个输入控件，以输入文本信息。为在操作后及时关闭操作文件，再添加一个关闭文件函数与写入文本文件函数相连，完成后的程序框图和运行效果如图 8-13 所示。

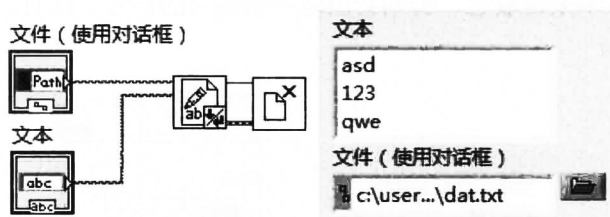


图 8-13 生成文本文件

在测试运行前，应先输入文件路径、文件名和文本信息，再运行程序。为检验操作效果，可在指定路径用记事本打开生成文件，观察写入内容是否正确，其效果如图 8-14 所示。

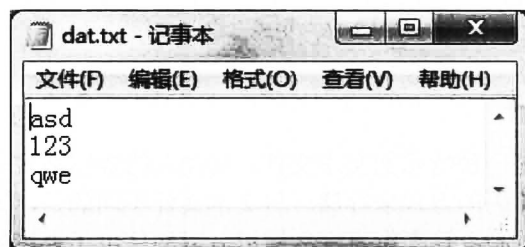


图 8-14 以记事本打开生成文件



8.2.2 二进制文件

相对于文本文件，二进制文件(.dat)是一种格式最紧凑的数据存储文件，它向文件写入的是二进制数据，因此比文本文件更节省存储空间，读写速度也更快。

(1) 读取二进制文件

使用“读取二进制文件”函数可以从二进制文件中读取相应类型的数据，具体的方法是：在后面板选取“函数→编程→文件 I/O→读取二进制文件”以进行添加，然后为文件端口添加一个路径及文件名输入控件。再为数据类型端口设置相应的类型，如定义一个簇，其包含一个 DBL 型数据和和一个字符型数据。为显示数据，为数据端口添加一个显示控件，则程序框图和运行效果如图 8-15 所示。



图 8-15 二进制文件读取

(2) 生成二进制文件

生成二进制文件的方法是：在后面板选取“函数→编程→文件 I/O→写入二进制文件”以进行添加，然后为文件端口添加一个输入控件，以指定生成文件的路径和文件名。在数据端口添加一个输入控件，以输入信息。进入前面板，向数据簇中添加一个 DBL 型输入控件和一个文本输入控件，则可向二进制文件写入相应的数值信息和文本信息。

完成后的程序框图和运行效果如图 8-16 所示。由于二进制文件所存储的不是 ASCII 码，因此不能直接用记事本来读取它，否则会出现一些离奇的符号。此时，可借助一些特定工具来读取，如 VC++ 等。

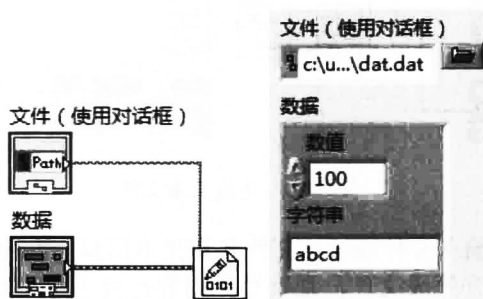


图 8-16 生成二进制文件

8.2.3 电子表格文件

电子表格文件(.xls)是一种特殊的文本文件，其输入的数据格式可以是一维或二维双精度浮点，其数据仍以 ASCII 码的方式来存储。与文本文件不同的是，该类文件对输入的数据格式作了相应的规定，如用制表符来作为段落标记，以便使电子表格处理软件 Excel 直接进行读写操作。



提示

通过对电子表格文件的读写操作，LABVIEW 可以直接与 Excel 进行数据交换，进而为虚拟仪表与电子报表处理建立交流的桥梁。

(1) 读取电子表格文件

使用“读取电子表格文件”函数可以从 Excel 文件中读取数据，具体的使用方法是：在后面板选取“函数→编程→文件 I/O→读取电子表格文件”以进行添加，然后为文件路径端口添加一个路径常量并进行赋值。为了显示读取的数据，为所有行端口添加一个显示控件，其程序框图和运行效果如图 8-17 所示。

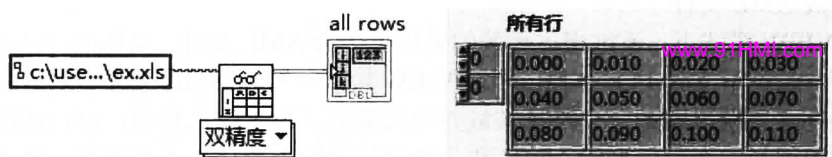


图 8-17 读取电子表格文件

(2) 生成电子表格文件

使用“写入电子表格文件”函数可以将数组中的数据存入电子表格文件，具体的方法是：在后面板选取“函数→编程→文件 I/O→写入电子表格文件”以进行添加，然后为文件路径端口添加一个路径常量并为其赋值，以指定生成文件的路径和文件名。再为二维数据端口添加一个输入控件，以输入数据，完成后的程序框图和运行效果如图 8-18 所示。

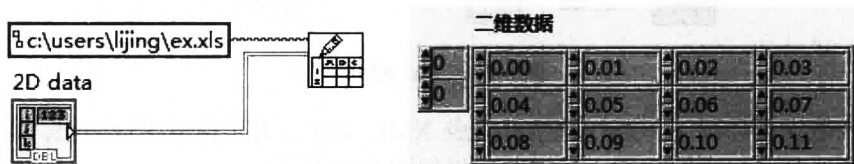


图 8-18 生成电子表格文件

在测试运行前，应先对二维数组进行赋值，再运行程序。为检验操作效果，可在指定路径由 Excel 打开生成的文件，以检验写入的内容是否正确，其效果如图 8-19 所示。



图 8-19 以 Excel 打开生成文件



8.3 特殊文件操作

LabVIEW 不仅能对常用文件进行读写操作，它还能对一些特殊文件进行操作，如 XML 文件、测量文件、数据存储文件及高速数据流文件等。

8.3.1 XML 文件

LabVIEW 处理 XML 文件有两种方式，一种是 LabVIEW 模式，另一种是 XML 解析器模式。其中，LabVIEW 模式用于操作 XML 格式的 LabVIEW 数据，而 XML 解析器模式则是通过 XML 解析器来处理 XML 文档。在 LabVIEW 中，XML 解析器是 Xerces 2.7，它需要使用文档对象模型(DOM)。

(1) 生成 XML 文件

在 LabVIEW 模式下，通过相关函数的使用可生成 XML 文件，具体的方法是：在后面板选取“函数→编程→文件 I/O→XML→LabVIEW 模式→写入 XML 文件”以进行添加，然后为 XML 文件路径端口添加一个路径常量并为其赋值。为了输入数据，再由前面板添加一个簇，并在簇中添加一个数值型输入控件和一个字符型输入控件，以作为簇文件的元素。

为了将输入数据转换为符合 XML 语法格式的文本，在后面板添加一个平化至 XML 图标和一个转换特殊字符至 XML 图标，完成连线后的程序框图和运行效果如图 8-20 所示。

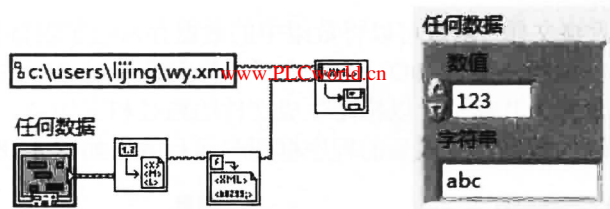


图 8-20 生成 XML 文件

在输入数据的情况下，运行程序可生成 XML 文件。为了检验生成效果，可在指定路径由网页浏览器打开生成的文件，其浏览效果如图 8-21 所示。

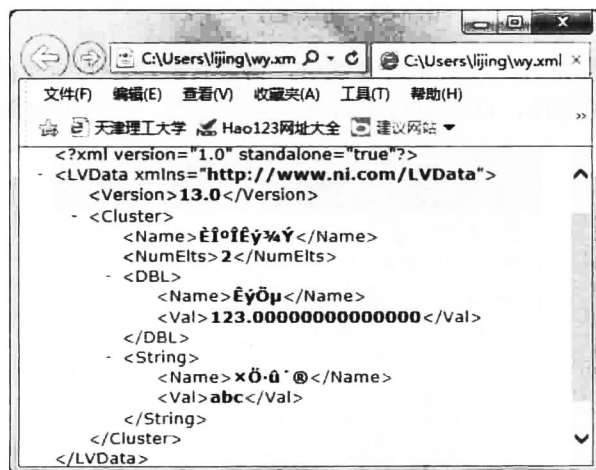


图 8-21 浏览 XML 文件



由浏览内容可以发现,页面中并非只显示出数据“123”和“abc”,而是通过标识符<Val>和</Val>标识出数据,其中数据“123”为 DBL 型数值,而数据“abc”为 String 型字符串。

(2) 读取 XML 文件

在 LabVIEW 模式下,同样可以使用相应的函数来读取 XML 文件,具体的方法是:在后面板,选取“函数→编程→文件 I/O→XML→LabVIEW 模式→读取 XML 文件”以进行添加,然后为读取 XML 文件端口添加一个路径常量并进行赋值。为了显示读取的数据,将多态 VI 选择器设置为“读取 XML 文件(字符串)”,并为 XML 元素端口添加一个显示控件(将属性设置为显示垂直滚动条),完成后的程序框图和运行效果如图 8-22 所示。

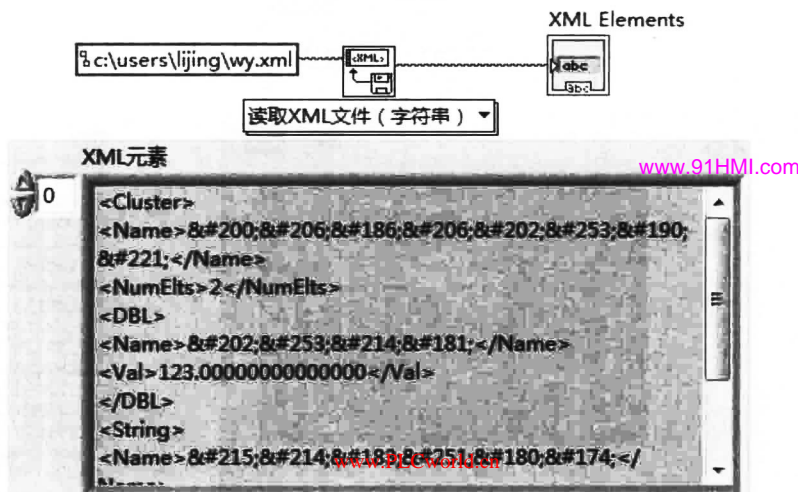


图 8-22 读取 XML 文件

提示

通过 XML 文件的读/写操作,LABVIEW 与 Web 建立了便利的数据交换通道,借此为虚拟仪表的网络化管理奠定了基础。

8.3.2 测量文件

测量文件(.lvm)是一种特殊格式的文件,它将动态类型数据以一定的格式存储在文本文件中,并只能由 LabVIEW 来读写。测量文件是在数据的前面添加一些诸如采集时间之类的信息,可以用 Excel 等文本编辑工具对其内容进行查看。

(1) 生成测量文件

生成测量文件的方法是:在后面板选取“函数→编程→文件 I/O→写入测量文件”以进行添加,然后为“文件名”端口添加一个常量,并为其赋值。

为了向测量文件提供测量数据,可利用仿真信号源,具体的操作方法是,在后面板选取“函数→Express→输入→仿真信号”。为显示仿真信号的波形,需向仿真信号图标添加一个图形显示控件。

若想显示仿真信号的数据,可选取“函数→Express→信号操作→从动态数据转换”,并为其添加一个显示控件,完成后的程序框图和运行效果分别如图 8-23 和图 8-24 所示。

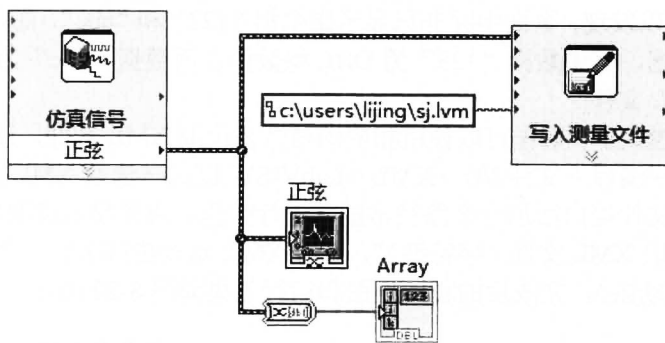


图 8-23 生成测量文件

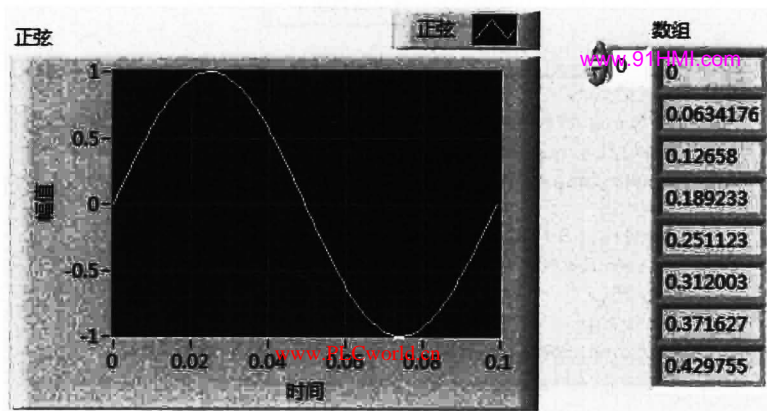


图 8-24 测量数据采集

此外，若审核测量文件的内容，在指定的测量文件存储位置，由 Excel 打开测量文件，其效果如图 8-25 所示。

	A	B	C	D	E
22	X_Value	正弦	Comment		
23			0		
24		0.063418			
25		0.12658			
26		0.189233			
27		0.251123			

图 8-25 审核测量文件

(2) 读取测量文件

读取测量文件的方法是：在后面板，选取“函数→编程→文件 I/O→读取测量文件”以进行添加，然后为“文件名”端口添加一个路径常量并进行赋值。为了显示测量文件的波形，在“信号”端口添加一个图形显示控件，完成后的程序框图和运行效果分别如图 8-26 和图 8-27 所示。

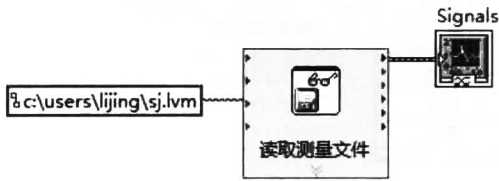


图 8-26 读取测量文件

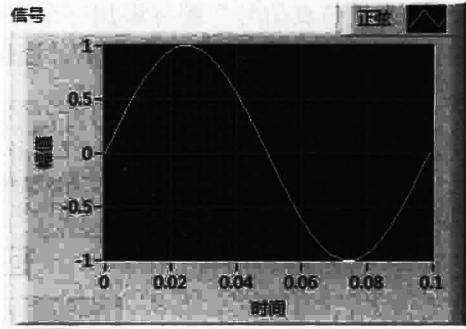


图 8-27 绘制测量文件波形

8.3.3 数据存储文件

数据存储文件(.tdm)是一种将动态类型数据存储为二进制数据的文件，它为每一个信号都添加了一些辅助信息，以便在查询时可以快速进行定位，从而便于在 NI 的软件间流畅地交换数据。

对于辅助性信息，如名称、单位和注释等，是以 XML 的格式被存储到扩展名为.tdm 文件中。而信号数据则存储到扩展名为.tdx 文件中，这两个文件在使用时是以引用的方式自动联系起来的。

(1) 生成数据存储文件

生成数据存储文件的方法是使用“存储/数据插件”子模板下的相关函数，具体的方法是：

- ① 在后面板，选取“函数→编程→文件 I/O→存储/数据插件→打开数据存储”以进行添加，并在弹出的“配置打开数据存储”对话框中，将数据插件设置为“TDM 包括 XML 头 (*.tdm)”。
- ② 为“打开数据存储”图标的“文件路径”端口添加一个常量，以设置数据存储的文件名和路径。
- ③ 在后面板添加“写入数据”、“关闭数据存储”及“简易错误处理器”图标。
- ④ 为了提供存储的数据，在后面板添加一个仿真信号图标，并将信号类型设为“正弦”，勾选“添加噪声”选项，将噪声类型选为“MLS 序列”，其设置对话框如图 8-28 所示。

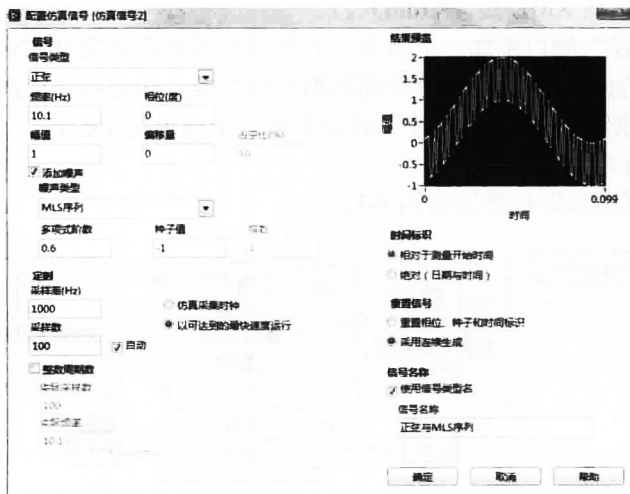


图 8-28 仿真信号设置



⑤ 为“仿真信号”图标添加一个图形显示控件，为“写入数据”图标的“名称”端口添加一个字符常量并赋值为“Data”。连接后的程序框图如图 8-29 所示。

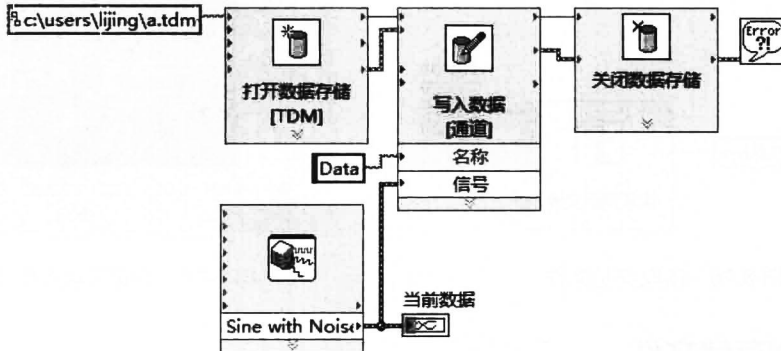


图 8-29 生成数据存储文件

www.91HMI.com

运行该程序，则界面如图 8-30 所示。

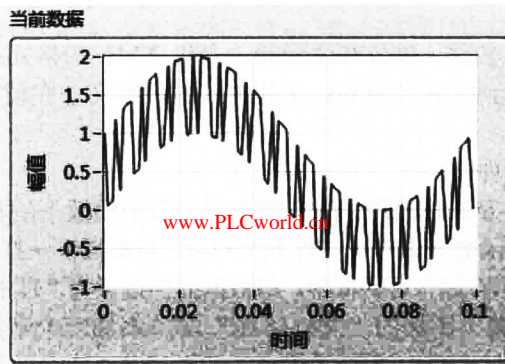


图 8-30 生成数据存储文件界面

(2) 读取数据存储文件

① 选取“函数→编程→文件 I/O→存储/数据插件→打开数据存储”以进行添加，将数据插件设置为“TDM 包括 XML 头 (*.tdm)”。

② 为“文件路径”端口添加一个常量，并设置好数据存储文件的名称和路径。

③ 在后面板添加“读取数据”、“关闭数据存储”及“简易错误处理器”图标。

④ 为“读取数据”图标的“信号”端口添加一个图形显示控件，为“名称”端口添加一个字符常量并赋值为“Data”。

完成后的程序框图和运行效果如图 8-31。

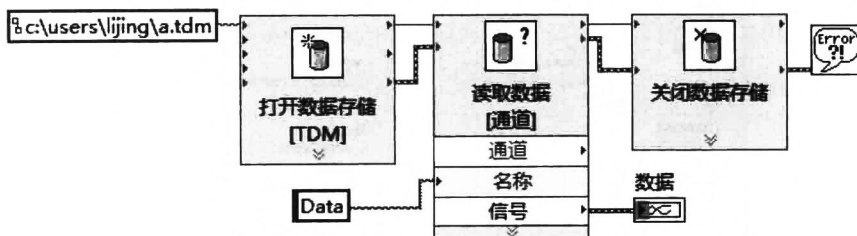


图 8-31 读取数据存储文件



为了获得和生成界面相同的波形显示效果，需将图形显示控件的时间（X 轴）标尺的最大值设置为 0.1，显示效果如图 8-32 所示。

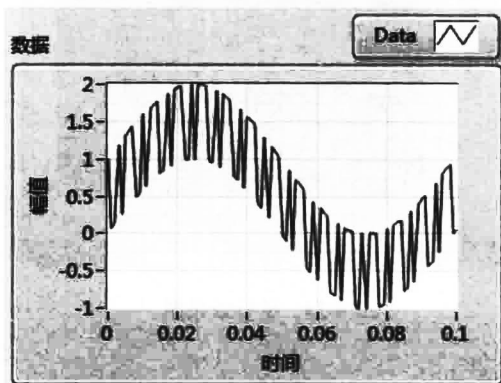


图 8-32 读取数据存储文件绘图

www.91HMI.com

8.3.4 高速数据流文件

高速数据流文件(.tdms)是在 TDM 文件的基础上进行了相应改进的一种文件格式，它比 TDM 文件的读写速度更快，属性定义的接口也更为简单。TDM 文件与 TDMS 文件可以相互转换，因此可以用 TDMS 文件来取代 TDM 文件。

尽管 TDMS 文件的操作函数用起来更为便捷，但并不比 TDM 文件的操作函数更为容易。

(1) 生成高速数据流文件

通过 TDMS 子模板的相关函数可以生成高速数据流文件，具体的方法是：

① 在后面板，选取“函数→编程→文件 I/O→TDMS→TDMS 打开”以进行添加，然后为“文件路径”端口添加一个路径常量并进行赋值，为“操作”端口添加一个常量并设置为“create or replace”。

② 再添加一个“TDMS 写入”图标和一个“TDMS 关闭”图标。

③ 为了输入数据，添加一个“仿真信号”图标，并将信号类型设为“正弦”，勾选“添加噪声”选项，将噪声类型选为“均匀白噪声”。

④ 为“仿真信号”的“正弦与均匀噪声”端口添加一个图形显示控件，经连线完成后的程序框图和运行效果如图 8-33 和图 8-34 所示。

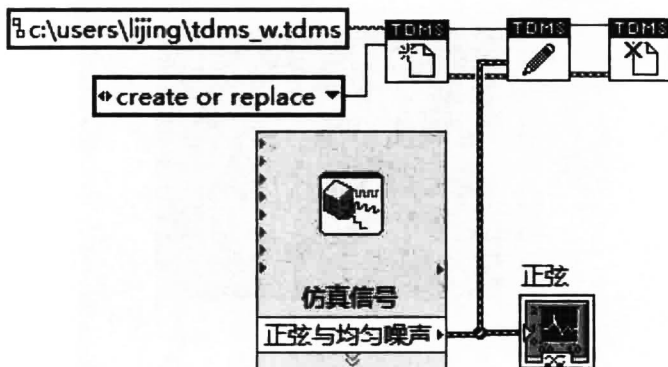


图 8-33 生成高速数据流文件

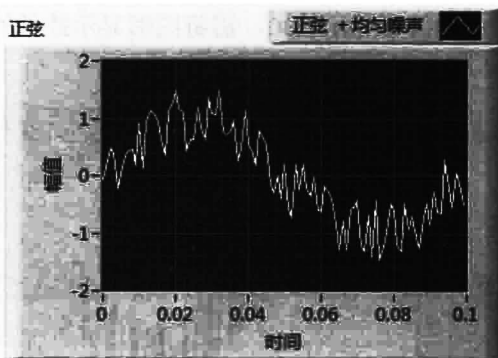


图 8-34 运行效果图

(2) 读取高速数据流文件

读取高速数据流文件与生成高速数据流文件相近，具体的方法是

- ① 在后面板，选取“函数→编程→文件 I/O→TDMS→TDMS 打开”以进行添加，然后为“文件路径”端口添加一个路径常量并进行赋值，为“操作”端口添加一个常量并设置为“open”。
- ② 再添加一个“TDMS 读取”图标和一个“TDMS 关闭”图标。
- ③ 为了显示波形，在前面板添加一个波形图表控件，经连线完成后的程序框图如图 8-35 所示。

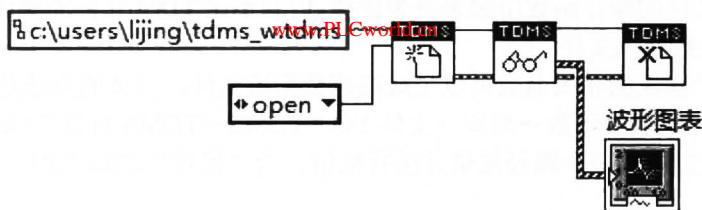


图 8-35 读取高速数据流文件

为了获得与生成界面相同的显示效果，应将波形图表的时间（X 轴）标尺设置为“自动调整标尺”，其显示效果如图 8-36 所示。

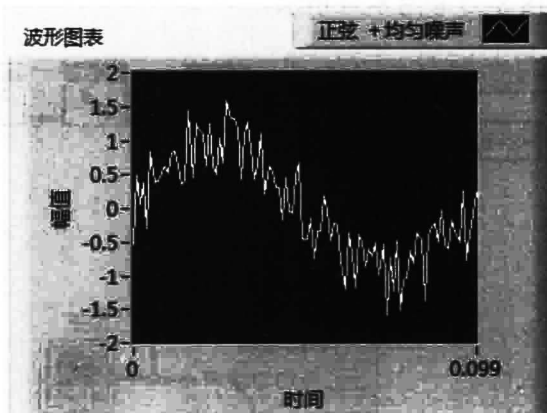


图 8-36 运行效果图



提示

对于.tdm和.tdms格式文件，还可以通过“转换至TDM或TDMS”函数来获得，这也是一条简便的获取途径。

8.4 其他文件操作

除了特殊文件外，还有其他一些文件，它们包括数据记录文件、波形文件、压缩文件(.Zip)和配置文件(.ini)。

8.4.1 数据记录文件

www.91HMI.com

数据记录文件也是一种二进制文件，但它存储数据的基本类型为记录块。在数据记录文件中，每个记录都是一个簇，可由使用者予以定义和设计。使用数据记录子模板中的函数，可以便捷地创建和读/写数据记录文件。

(1) 生成数据记录文件

要生成数据记录文件涉及创建文件、写入数据记录、关闭文件等，具体的方法是：

- ① 在后面板，选取“函数→编程→文件 I/O→高级文件函数→数据记录→打开/创建/替换数据记录文件”以进行添加，为各端口添加相应输入控件及赋值。
- ② 依次添加设置数据记录位置、写入数据记录文件和关闭文件图标。
- ③ 创建一个含有数据和字符串的簇，以提供数据记录，连线后的程序框图和运行结果如图 8-37。

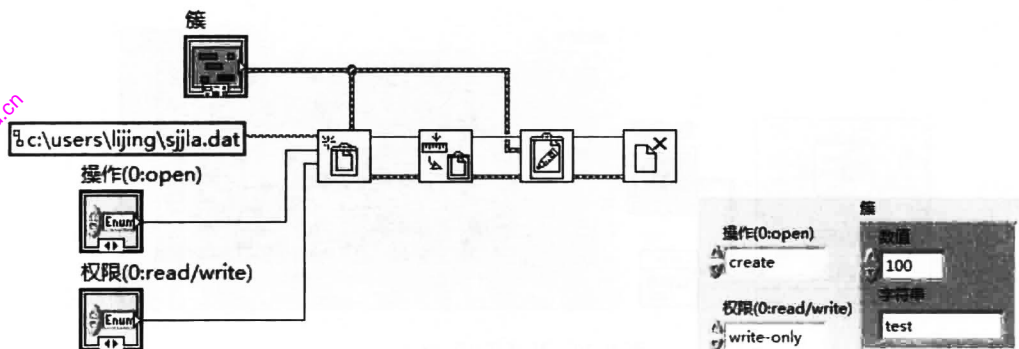


图 8-37 生成数据记录文件

(2) 读取数据记录文件

要生成数据记录文件涉及创建文件、写入数据记录、关闭文件等，具体的方法是：

- ① 在后面板，选取“函数→编程→文件 I/O→高级文件函数→数据记录→打开/创建/替换数据记录文件”以进行添加，为各端口添加相应输入控件及赋值。
- ② 依次添加获取数据记录位置、获取记录数量、读取数据记录文件和关闭文件图标。
- ③ 创建一个含有数据数据和字符串数据的簇，以读取数据记录，连线后的程序框图和运行结果如图 8-38。

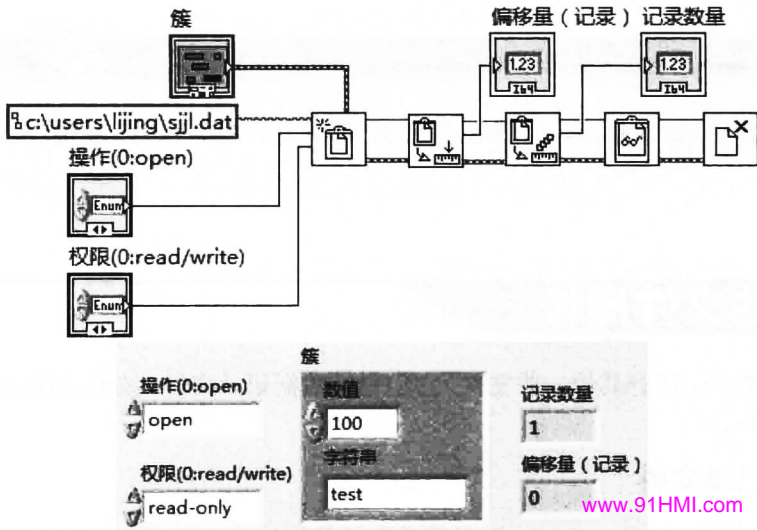


图 8-38 读取数据记录文件

8.4.2 波形文件

波形文件是一种特殊的数据记录文件，数据记录文件 I/O 也可以对波形数据进行读写操作。在保存波形数据文件时，可以使用任意扩展名（如.dat 或.txt 文件）。

(1) 生成波形文件

生成波形文件的具体方法是：在后面板，选取“函数→编程→文件 I/O→波形文件 I/O→写入波形至文件”以进行添加，然后为其添加仿真信号及显示控件，则完成连线后的程序框图和运行结果如图 8-39。

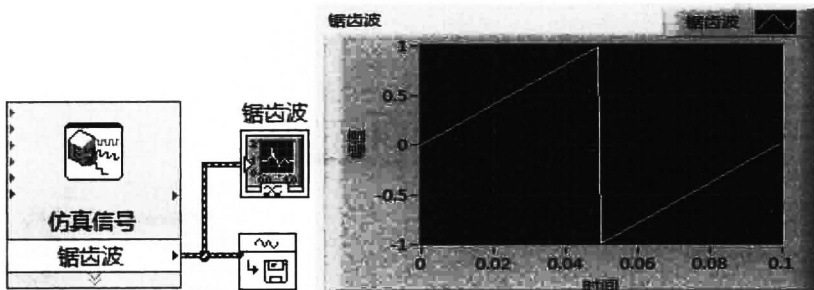


图 8-39 生成波形文件

(2) 读取波形文件

使用波形文件读取函数可读取波形文件，具体的操作方法是：在后面板，选取“函数→编程→文件 I/O→波形文件 I/O→从文件读取波形”以进行添加，然后为其添加文件路径和显示控件，则完成连线后的程序框图和运行结果如图 8-40。

为了保证显示效果与存储波形的一致性，应将波形图表的时间（X 轴）标尺设置为“自动调整标尺”。

(3) 波形文件转换

使用波形文件转换函数可将波形文件转换为电子表格文件，具体的操作方法是：

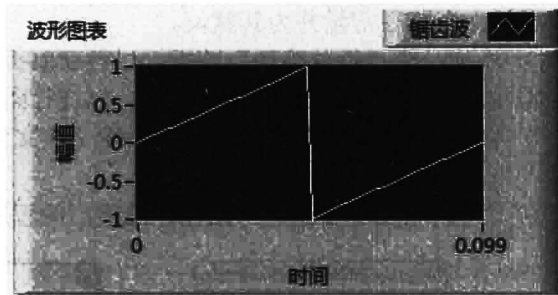
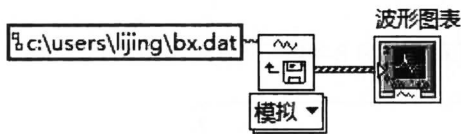


图 8-40 读取波形文件 (一)

在读取波形文件的基础上, 选取“函数→编程→文件 I/O→波形文件 I/O→导出波形至电子表格文件”以进行添加, 然后为其添加一个“新建文件路径”的显示控件和一个“文件路径”常量并予以赋值, 连线后的程序框图和运行结果分别如图 8-41 和图 8-42 所示。

使用记事本打开生成的电子表格文件, 其内容如图 8-43 所示。 www.91HMI.com

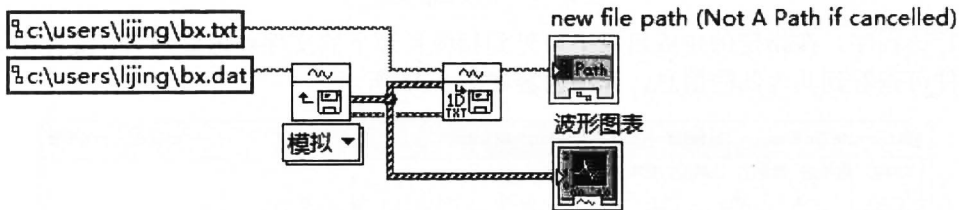


图 8-41 读取波形文件 (二)

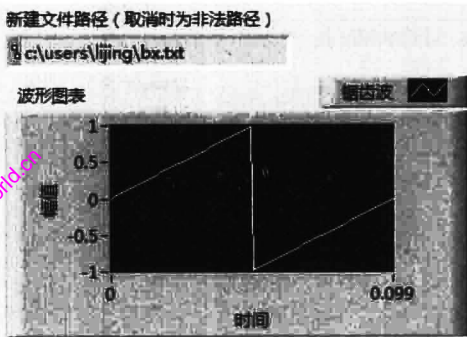


图 8-42 波形文件转换

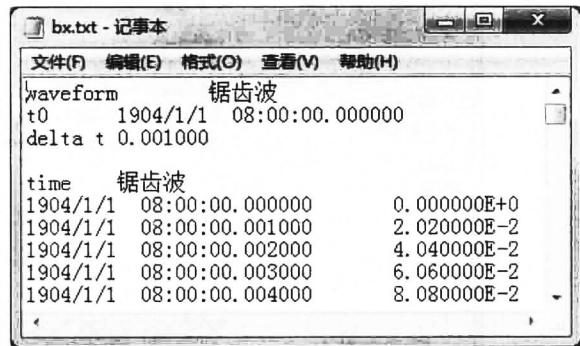


图 8-43 读取波形文件 (三)

8.4.3 压缩文件

压缩文件是一种有效节省文件存储空间占用的方式, LabVIEW 对文件压缩也提供相应的支持, 涉及的函数有新建 Zip 文件、添加文件至 Zip 文件、关闭 Zip 文件及解压缩等函数。

(1) 生成 Zip 文件

在具有压缩对象的情况下, 通过相关函数的使用可生成 Zip 文件, 具体的方法如下。

- ① 在后面板选取“函数→编程→文件 I/O→Zip→新建 Zip 文件”以进行添加, 再为“目标”端口添加一个路径常量并为其赋值。
- ② 选取“函数→编程→文件 I/O→高级文件函数→递归文件列表”, 再为“文件夹路径”



端口添加一个路径常量并为其赋值。

- ③ 添加一个“拆分路径”图标，以便获取拆分路径。
- ④ 添加一个“两路径比较”图标，以进行路径比较。
- ⑤ 在后面板，依次添加“添加文件至 Zip 文件”、“关闭 Zip 文件”和“简易错误处理器”图标。
- ⑥ 添加一个“For 循环”图标，连线后的程序框图如图 8-44 所示。

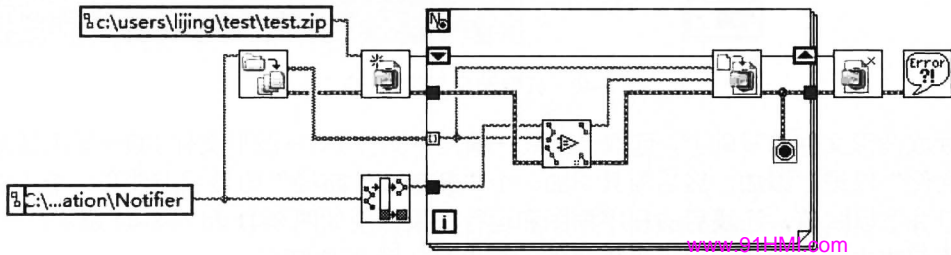


图 8-44 生成 Zip 文件

运行该程序，在指定的生成目录下可见到相应目录下的文件被压缩到 Zip 文件中。用解压缩软件可查看到其内部的信息，详细内容如图 8-45 所示。

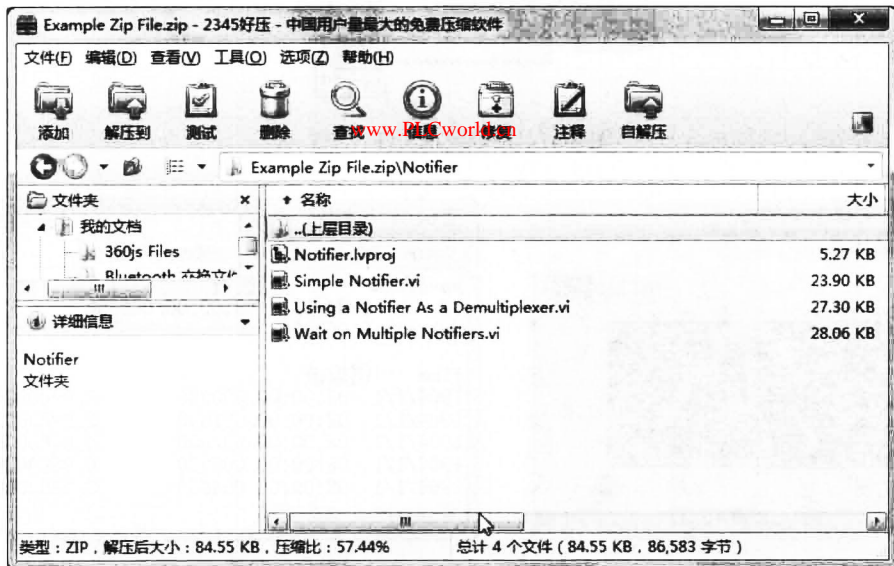


图 8-45 压缩文件内部信息

(2) 解压 Zip 文件

使用解压缩函数可对 Zip 文件进行还原，具体的方法是：在后面板，选取“函数→编程→文件 I/O→Zip→解压缩”以进行添加，然后为“Zip 文件”和“目标目录”端口添加一个路径常量并进行赋值，完成后的程序框图和运行结果分别如图 8-46 和图 8-47 所示。

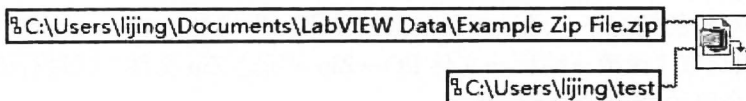


图 8-46 解压 Zip 文件程序框图



图 8-47 解压效果

www.91HMI.com

8.4.4 配置文件

配置文件是一种标准的 Windows 配置文件，并可以独立于平台进行读取和创建。配置文件由分段命名的文本文件组成，段名位于方括号内，且是唯一的。段是一些由等号 (=) 隔开的键/值对。配置文件的结构如下。

```
[Section 1]
Key1=value
Key2=value
...
[Section 2]
Key1=value
Key2=value
...
```

www.PLCworld.cn

在配置文件 VI 中，键参数的值可以是字符串、路径、布尔值、64 位双精度浮点数、32 位有符号整数及 32 位无符号整数。

提示

配置文件不仅限于配置管理，它还可以作为系统参数的记忆性文件来使用，以简化不必要的多次参数设置。

(1) 生成配置文件

在配置文件 VI 子模板下，通过相关函数的使用可生成配置文件，具体的方法如下。

① 在后面板选取“函数→编程→文件 I/O→配置文件 VI→打开配置数据”以进行添加，然后为“必要时创建文件 (T)”端口添加一个常量，再为“配置文件的路径”端口添加一个路径常量并为其赋值。

② 在后前面，依次添加两个“写入键”图标、一个“关闭配置数据”图标和一个“简易错误处理器”图标。

③ 双击“写入键”图标，将值的数据类型由逻辑型修改为 DBL 数值型，并加以保存和



返回。

④ 为“写入键”图标的段、键、值端口添加相应的常量并赋值，连线后的程序框图如图 8-48 所示。

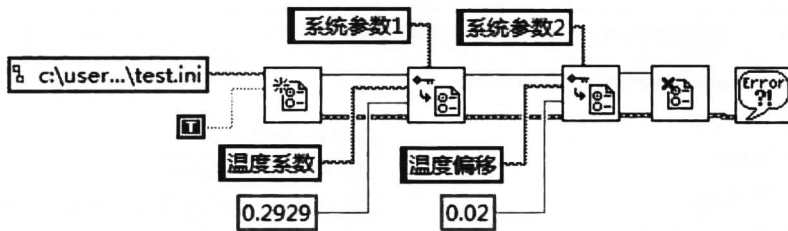


图 8-48 生成配置文件

运行该程序，在指定的生成位置用记事本打开配置文件，则内容如图 8-49 所示。

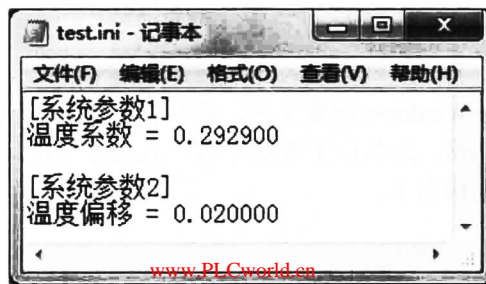


图 8-49 生成配置文件

(2) 读取配置文件

通过对读取键函数的使用，可以从配置文件中提取相关参数，具体的方法如下。

① 在后面板选取“函数→编程→文件 I/O→配置文件 VI→打开配置数据”以进行添加，然后为“必要时创建文件 (T)”端口添加一个常量，再为“配置文件的路径”端口添加一个路径常量并为其赋值。

② 在后前面，依次添加两个“读取键”图标、一个“关闭配置数据”图标和一个“简易错误处理器”图标。

③ 为“读取键”图标的段、键端口添加相应的常量并赋值，并将数据类型设为“双精度”。

④ 为“读取键”图标的“值”端口添加相应的显示控件，则连线后的程序框图和显示效果如图 8-50 所示。

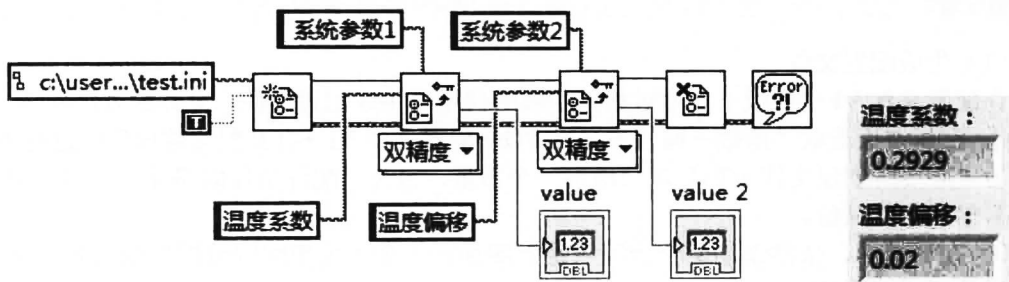


图 8-50 读取配置文件



8.5 高级文件工具

8.5.1 获取文件信息

(1) 获取卷信息

使用“获取卷信息”函数可以获得指定卷的磁盘容量（字节）、扇区大小（字节）和可用容量（字节），具体操作是：在后面板选取“函数→编程→文件 I/O→高级文件函数→获取卷信息”以进行添加，然后为各端口添加相应的常量和显示控件，则连线后的程序框图如图 8-51 所示。

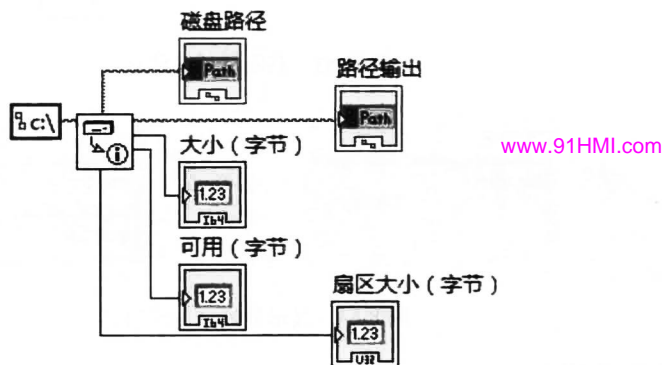


图 8-51 获取卷信息

www.PLCworld.cn

运行该程序，则可得到指定卷的信息，读取该卷的属性可予以验证，具体内容如图 8-52 所示。

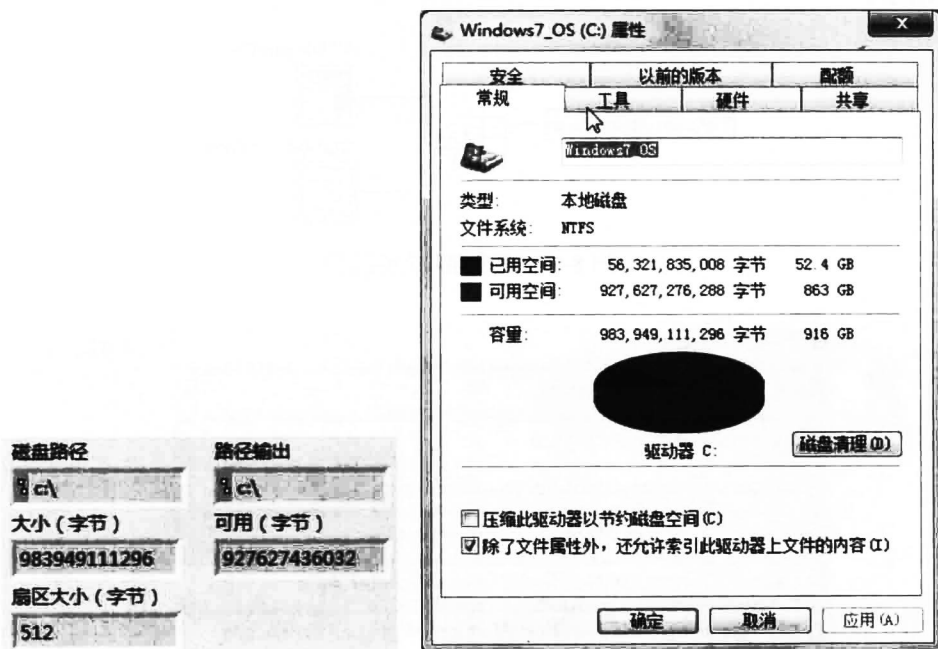


图 8-52 显示卷信息



(2) 罗列文件夹

使用“罗列文件夹”函数可以获取指定文件夹的信息，具体操作是：在后面板选取“函数→编程→文件 I/O→高级文件函数→罗列文件夹”以进行添加，然后为各端口添加相应的常量和显示控件，则连线后的程序框图和执行效果分别如图 8-53 和图 8-54 所示。

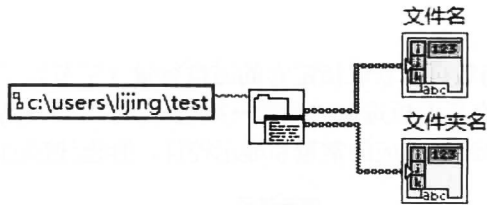


图 8-53 获取罗列文件夹

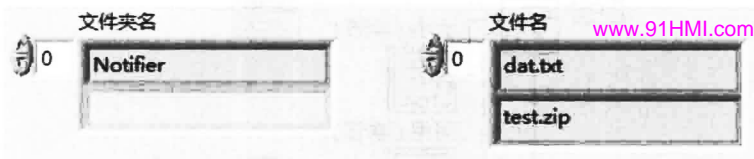


图 8-54 显示罗列文件夹信息

(3) 递归文件列表

使用“递归文件列表”函数可以获取指定文件夹下的目录、文件名、文件数等信息，具体操作是：在后面板选取“函数→编程→文件 I/O→高级文件函数→递归文件列表”以进行添加，然后为各端口添加相应的常量和显示控件，则连线后程序框图和执行效果如图 8-55 和图 8-56 所示。

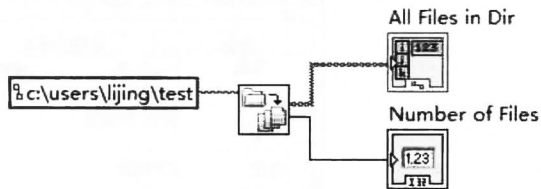


图 8-55 获取递归文件列表

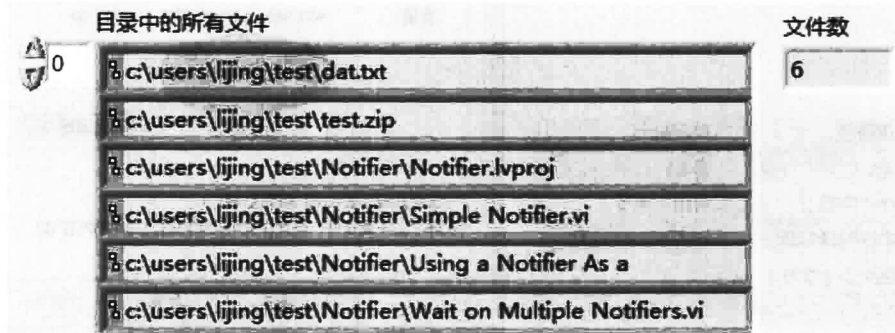


图 8-56 显示递归文件列表



(4) 获取文件扩展名

使用“获取文件扩展名”函数可以获得指定文件的名称及扩展名，具体操作是：在后面板选取“函数→编程→文件 I/O→高级文件函数→获取文件扩展名”以进行添加，然后为各端口添加相应的常量和显示控件，则连线后的程序框图和执行效果如图 8-57 所示。

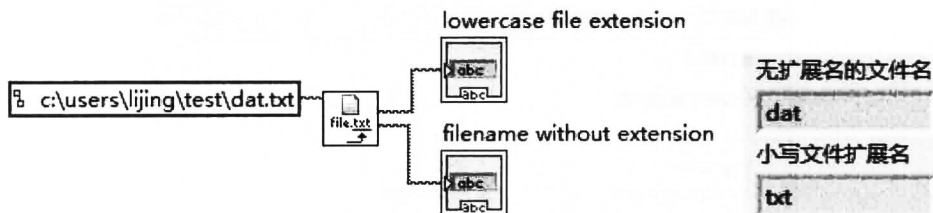


图 8-57 获取文件扩展名

(5) 获取文件大小

使用“获取文件大小”函数，可得到指定文件的大小，操作方法是：在后面板选取“函数→编程→文件 I/O→高级文件函数→获取文件大小”以进行添加，为“文件”端口添加一个路径常量并为其赋值，再为“端口”端口添加显示控件。连线后的程序框图如图 8-58 所示，用记事本可核对文件的大小，内容如图 8-59 所示。

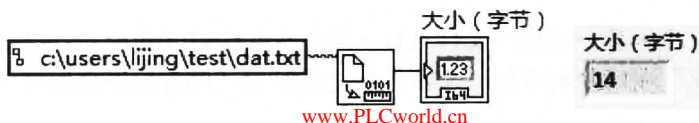


图 8-58 获取配置文件

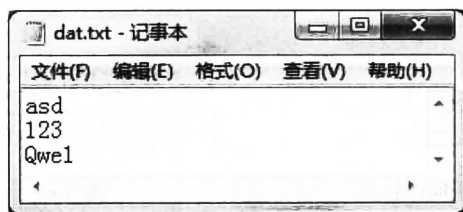


图 8-59 用记事本核对文件大小

8.5.2 文件操作与处理

对文件进行操作涉及创建文件夹及文件对话框等辅助型函数，而操作的主体则是对文件的移动、复制和删除。

(1) 创建文件夹

创建文件夹的具体操作是，在后面板选取“函数→编程→文件 I/O→高级文件函数→创建文件夹”以进行添加，然后为“路径（使用对话框）”端口添加一个输入控件，则连线后的程序框图如图 8-60 所示，进入指定目录可核实文件夹是否被创建。

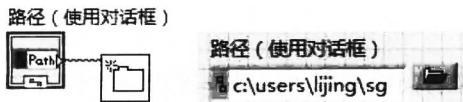


图 8-60 创建文件夹



(2) 文件对话框的使用

使用文件对话框的具体方法是，在后面板选取“函数→编程→文件 I/O→高级文件函数→文件对话框”以进行添加，在随后弹出的配置文件对话框中，可进行相应的设置，如图 8-61 所示。单击“确定”按钮，可保持设置并退出对话框。

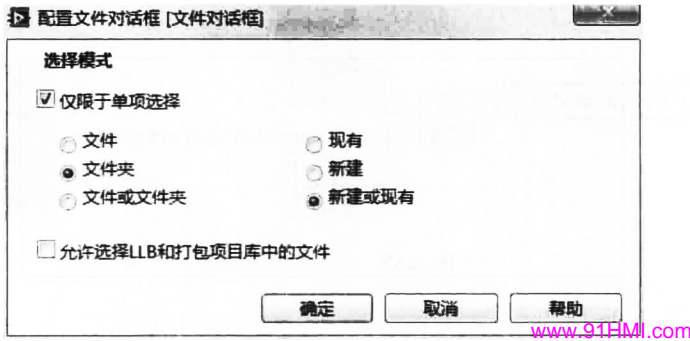


图 8-61 配置文件对话框

使用文件对话框也可辅助文件夹的创建，此时在配置对话框中应选取“文件夹”选项，则程序框图如图 8-62 所示。

(3) 移动文件

在后面板选取“函数→编程→文件 I/O→高级文件函数→移动”以进行添加，为“源路径”和“目标路径”端口分别添加一个常量并进行赋值，此时程序框图如图 8-63 所示。

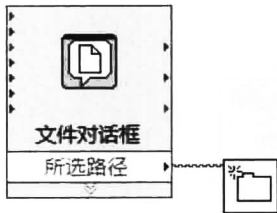


图 8-62 创建文件夹程序框图

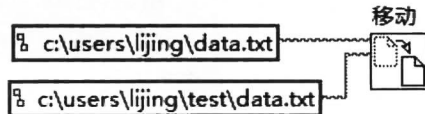


图 8-63 移动文件

(4) 复制文件

复制文件的方法是，在后面板选取“函数→编程→文件 I/O→高级文件函数→复制”以进行添加，为“源路径”和“目标路径”端口分别添加一个常量并进行赋值，此时程序框图如图 8-64 所示。



图 8-64 复制文件

(5) 删除文件

删除文件的方法是，在后面板选取“函数→编程→文件 I/O→高级文件函数→删除”以进行添加，为“路径”端口添加一个常量并进行赋值，此时程序框图如图 8-65 所示。

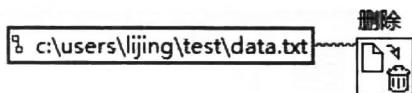


图 8-65 删除文件

8.6 综合实例：采集文件的存储与读取

在虚拟仪表设计中，对采集数据的管理是一个不可或缺的功能，因为除了对数据的在线分析，还需要对数据进行存储和再现处理。解决此类问题的基本思想是：对采集数据进行文件存储和读取，这里以数据记录文件的读、写为例来进行介绍，其操作界面如图 8-66 所示。

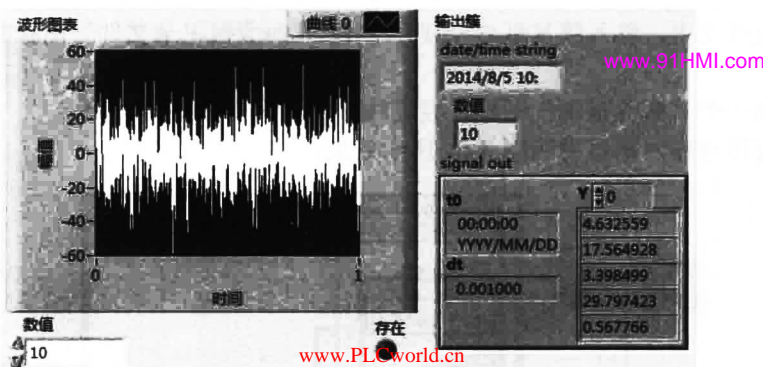


图 8-66 采集文件的存储与读取

要实现采集文件的存储与读取，其操作步骤如下。

① 新建一个 VI，并予以命名保存。

② 在前面板放置一个波形图表控件、一个数值输入控件和一个圆形指示灯控件，其中的波形图标控件用于显示采集数据的波形，数值输入控件用于数值采集数据量，指示灯控件用于指示文件的存在性。

③ 在后面板，添加一个文件对话框和一个条件结构图标。其中条件图标的“假”分支用于数据记录文件的创建与存储，而“真”分支用于数据记录文件的读取和波形的再现。

④ 将文件对话框的“存在”端口与圆形指示灯及条件图标的条件输入端相连。

⑤ 进入条件图标的“假”分支，依次放置“打开/创建/替换数据记录文件”、“写入数据记录文件”及“关闭文件”图标，并将“打开/创建/替换数据记录文件”图标的“操作”端口设为“create”。

⑥ 在该分支内，再添加“格式化日期/时间字符串”、“周期性随机噪声波形”和“捆绑”图标，并将“格式化日期/时间字符串”、“数值输入”和“周期性随机噪声波形”图标捆绑到一起。

⑦ 为捆绑图标添加一个显示控件，经连线程序框图如图 8-67 所示。

⑧ 复制一次输出簇，并将其转换为常量。

⑨ 进入条件图标的“真”分支，依次放置“打开/创建/替换数据记录文件”、“读取数据记录文件”及“关闭文件”图标，并将“打开/创建/替换数据记录文件”图标的“操作”端口设为“open”。

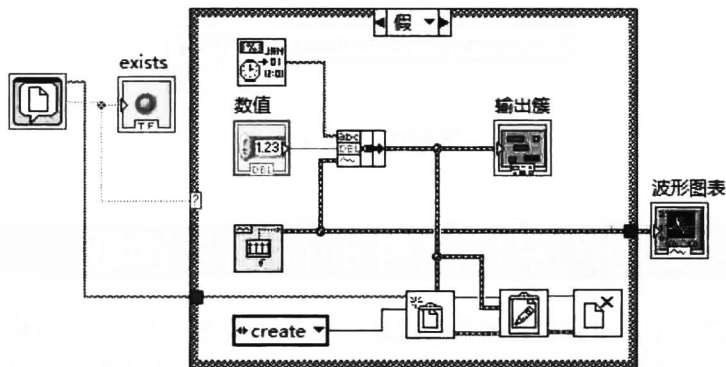


图 8-67 存储采集文件

⑩ 在该分支内，移入簇常量与“打开/创建/替换数据记录文件”图标的“记录类型”端口相连。
www.91HMI.com

- ⑪ 添加一个“解除捆绑”图标，经连线程序框图如图 8-68 所示。
- ⑫ 运行该程序，则可看到图 8-66 所示效果。

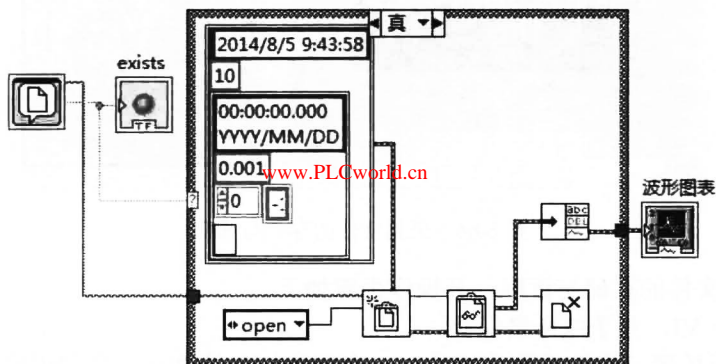


图 8-68 读取采集文件



LabVIEW 的强大文件 I/O 功能，使我们仿佛进入一个文件的王国，仅仅是它就会使大家受益无穷。



第 9 章

LabVIEW中的字符串操作

字符串是 ASCII 码的集合，它为独立于操作平台的信息和数据交换提供了有效的支持。在 LabVIEW 中，为了便于字符串的处理提供了多个字符串函数。

【本章导读】

- ◆ 常用的字符串常量
- ◆ 常用的字符串函数
- ◆ “匹配模式字符串”和“匹配正则表”函数
- ◆ “格式化字符串”和“扫描字符串”函数
- ◆ 数组与电子表格字符串
- ◆ 附加字符串函数等

www.PLCworld.cn

9.1 常用的字符串常量

在 LabVIEW 中，为了便于应用定义了一些常用的字符串常量，它们有空格常量、字符串常量、空字符串常量、回车键常量、换行符常量、行结束常量和制表符常量。

要使用这些常量，可在后面板通过选取“函数→编程→字符串”来获得，其图标如图 9-1 所示。



空格常量 字符串常量 空字符串常量 回车键常量 换行符常量 行结束常量 制表符常量

图 9-1 字符串常量

9.2 常用的字符串函数

(1) 字符串长度函数

该函数用于计算字符串的长度，并返回该字符串的字符个数。图 9-2 是求取字符串“LabVIEW2013 中文版”的长度，其结果为 17，其中的每个汉字按 2 个字符来计算。

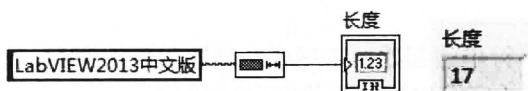


图 9-2 求取字符串长度



(2) 连接字符串函数

该函数的作用是将多个字符串连接起来构成一个新字符串，若输入的是字符串数组，则先按数组的顺序由前至后组成一个字符串，再用这个字符串参与连接。图 9-3 是将 2 个字符串常量和 1 个字符串数组连接成一个新字符串，其结果为“LabVIEW2013 版实用教程”。

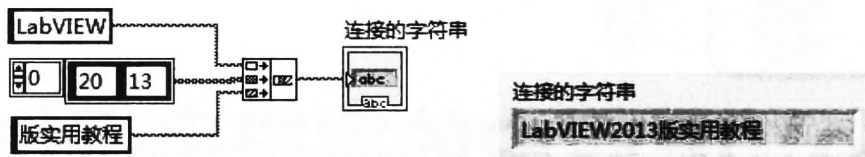


图 9-3 连接字符串

(3) 部分字符串函数

该函数的作用是从起始位置开始，截取指定长度的子字符串。图 9-4 是从字符串的第 7 个字符位开始截取 4 个字符，其结果为“2013”。

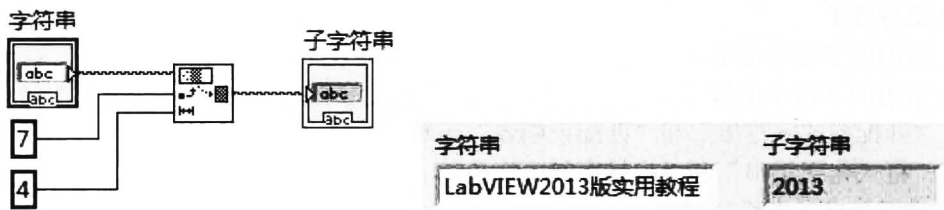


图 9-4 部分字符串

(4) 替换子字符串函数

该函数的作用是从起始位置开始，替换指定长度的子字符串。图 9-5 是从字符串的第 7 个字符位开始替换 4 个字符，其结果为“LabVIEW 英文版实用教程”。

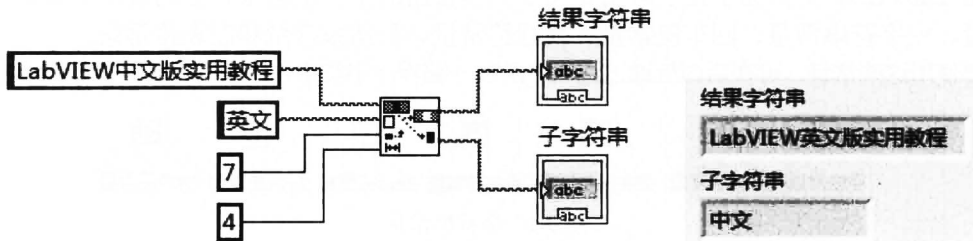


图 9-5 替换子字符串

(5) 搜索替换子字符串函数

该函数的作用是将一个或所有子字符串替换为另一个字符串。与替换子字符串不同的是，它不按位置和长度进行替换，而是搜索与子字符串一致的字符串来执行替换。图 9-6 是在字符串中搜索字符串“中文”，然后将其替换为“英文”，其结果为“LabVIEW 英文版实用教程”，并显示出替换数量和替换后的偏移量。

(6) 删除空白

该函数的作用是在字符串的起始、末尾或两端删除所有空白，该 VI 不删除双字节字符。图 9-7 是将 2 行具有空白的字符串执行删除空白处理。

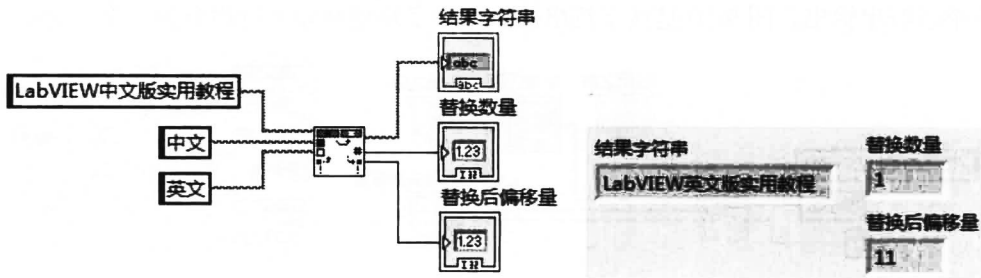


图 9-6 搜索替换子字符串

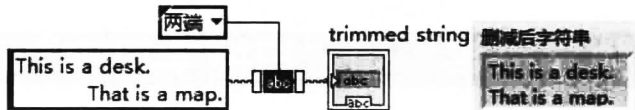


图 9-7 删除空白

www.91HMI.com

(7) 大小写转换函数

该函数的作用是将字符串的大小写进行转换。图 9-8 是将字符串“ This is a test.” 分别转换为全部大写和全部小写字母字符串。

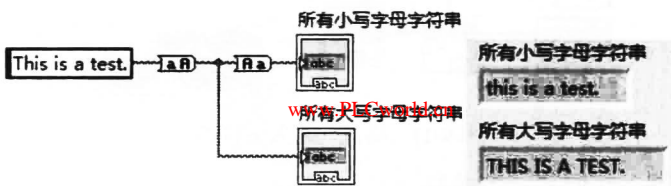


图 9-8 大小写转换

9.3 “匹配模式字符串”和“匹配正则表”函数

(1) 匹配模式字符串函数

该函数的作用是对输入的字符串按指定位置开始搜索，若发现匹配的字符串，则将输入字符串一分为三，即匹配字符串之前、匹配字符串及匹配字符串之后 3 个子字符串；否则，匹配子字符串为空。图 9-9 是从索引位置 1 开始，对字符串“自然数”进行搜索，然后输出匹配结果。

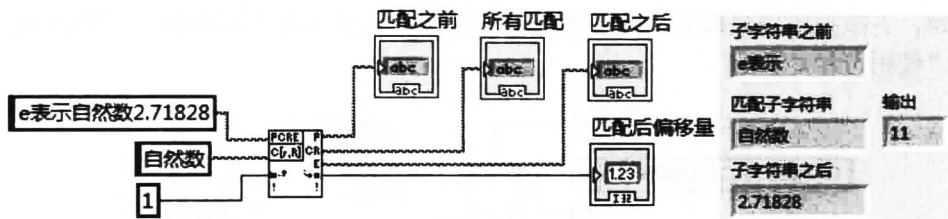


图 9-9 匹配模式字符串

(2) 匹配正则表函数

该函数的作用是从指定的位置开始搜索字符串，并以此字符串的头尾为界将输入字符串

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



分成三个字符串输出。图 9-10 是从字符串的第 1 个字符位开始字符串分为三个字符串。

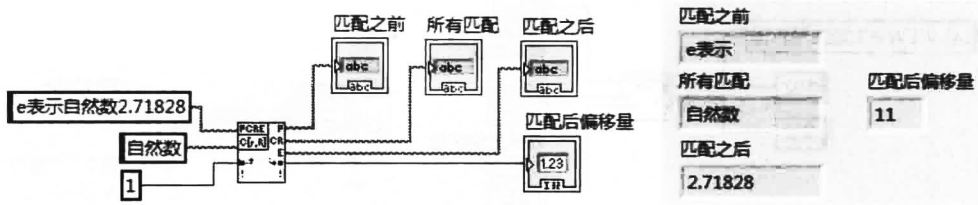


图 9-10 匹配正则表

9.4 “格式化字符串”和“扫描字符串”函数

(1) 格式化写入字符串函数

www.91HMI.com

该函数的作用是将字符串、数字或布尔值转换为文本，该文本的格式按指定的方式。图 9-11 是将字符串“自然数”和数值“2.71828”按指定的格式形成新的字符串，其结果为“自然数 2.718280”，由于是采用小数后 6 位的格式，在数位不足的情况下以 0 进行了填补。



图 9-11 格式化写入字符串

(2) 格式化日期、时间字符串函数

该函数的作用是按指定格式输出系统时间字符串，所采用的时间格式化代码为：%Y（世纪年）、%y（年）、%m（月）、%d（日）、%a（星期几）、%p（上、下午）、%H（24 小时制）、%I（12 小时制）、%M（分钟）、%S（秒）。图 9-12 是按照常规的时间形式显示时间。

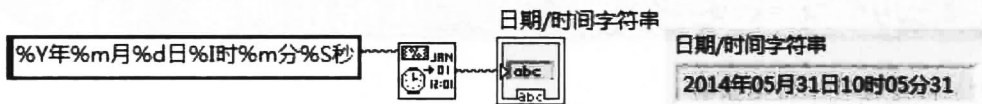


图 9-12 格式化日期、时间字符串

(3) 扫描字符串函数

该函数的作用是输入字符串，并按指定的格式输出。图 9-13 是从字符串的第 7 个字符位开始转换，并按指定格式截取 7 个字符宽度来输出，其结果为“2.71828”，截取后的剩余字符串为“使用符号 e 表示”。

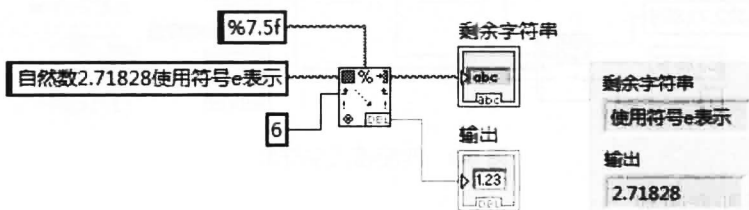


图 9-13 扫描字符串



9.5 数组与电子表格字符串

(1) 电子表格字符串至数组转换函数

该函数的作用是将电子表格字符串转换为数组，维度和表示法与数组类型一致。该函数适用于字符串数组和数值数组。图 9-14 是将电子表格字符串转换为一个 2 行 3 列的数组。

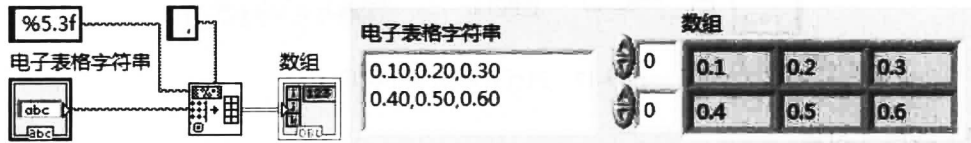


图 9-14 电子表格字符串至数组转换

(2) 数组至电子表格字符串转换函数

该函数的作用是使多维数组转换为字符串形式的表格，并可按指定分隔符来分割列元素。图 9-15 是将一个 2 行 3 列的二维数组转换为电子表格字符串，以“,”作列元素的分割符。

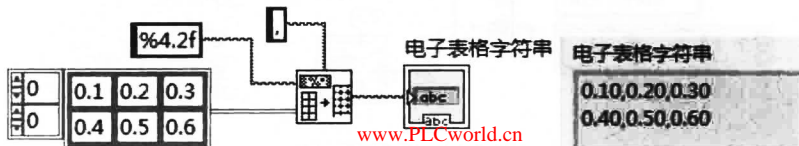


图 9-15 数组至电子表格字符串转换

9.6 附加字符串函数

附件字符串函数是一组函数，主要包括搜索/拆分字符串、选行并添加至字符串、匹配字符串、匹配真/假字符串、在字符串中搜索标记、搜索替换模式、搜索字符串数组、添加真/假字符串、字符串移位及反转字符串函数。

使用附件字符串函数的方法是，在后面板选取“函数→编程→字符串→附件字符串”，再选取对应的函数。

(1) 搜索/拆分字符串

该函数的作用是从起始位置开始，搜索匹配的字符串并将输入字符串拆分为匹配前和匹配后两部分。图 9-16 是从字符串的第 10 个字符位开始搜索字符串“V2.0”，其结果是拆分为字符串“流变分析仪”和“V2.0 版”两个字符串。

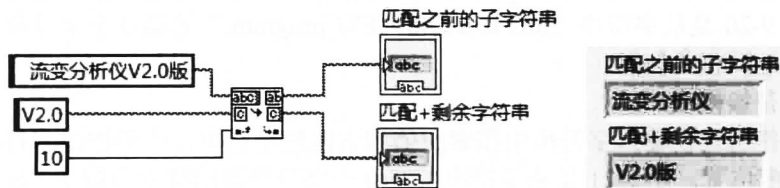


图 9-16 搜索/拆分字符串



(2) 选行并添加至字符串

该函数的作用是在多行字符串中选定一行，添加至字符串。图 9-17 是按索引值从多行字符串中选取“map.”与输入字符串“This is a ”拼接成一个语句“This is a map.”。



图 9-17 选行并添加至字符串

(3) 匹配字符串

该函数的作用是从字符串的起始位置开始与字符串数组进行比较，直至出现匹配。图 9-18 是用字符串“defxyabc”与字符串数组进行比较，由于与字符串数组中第 2 个元素相匹配，故输出字符串为“xyabc”，索引为“1”（因第 1 个元素的索引为“0”）。

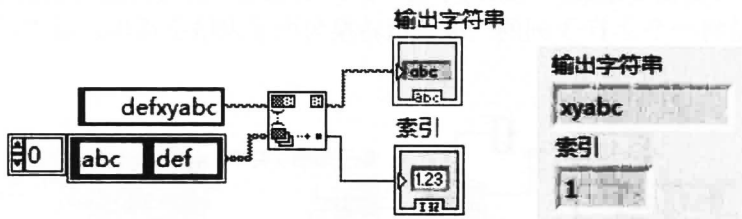


图 9-18 匹配字符串

(4) 匹配真/假字符串

该函数的作用是从头开始检查字符串，确定是否与真字符串或假字符串相匹配。图 9-19 是从头开始检查字符串“passtest”，由于同真字符串“passt”相匹配，故结果是输出字符串“test”和输出逻辑状态“真”（选择指示灯被点亮）。

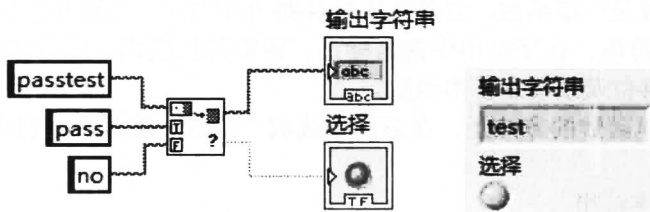


图 9-19 匹配真/假字符串

(5) 在字符串中搜索标记

该函数的作用是从偏移量位置开始扫描输入字符串，寻找标记，并通过标记字符串输出各个字段。图 9-20 是从字符串“this is a LabVIEW program.”的第 0 个字符位开始按默认标记开始搜索并返回相关信息。

(6) 搜索替换模式

该函数的作用是在输入字符串中按索引值搜索匹配字符串，再用指定字符串内容加以替换，并输出匹配位置。图 9-21 是从字符串的第 0 个字符位开始搜索字符串“end”，并用字符串“f”加以替换，其结果是“abfd”，匹配的偏移位置是“3”。

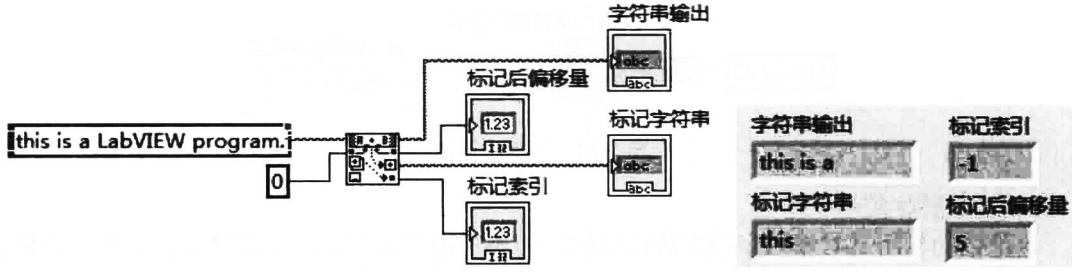


图 9-20 在字符串中搜索标记

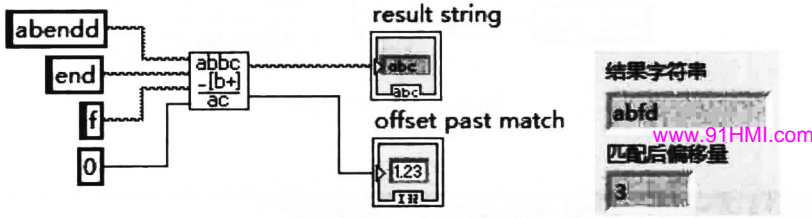


图 9-21 搜索替换模式

(7) 索引字符串数组

该函数的作用是按索引值从字符串数组中提取字符串与输入字符串进行拼接，以形成新的字符串。图 9-22 是按照索引来提取相应的年份字符串与输入字符串进行拼接，以形成新的字符串信息。

www.PLCworld.cn

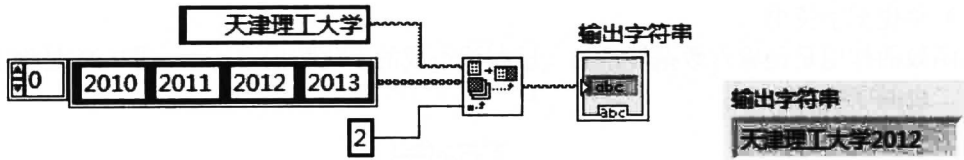


图 9-22 搜索字符串数组

(8) 添加真/假字符串

该函数的作用是按照逻辑真/假选取对应的内容与输入字符串进行拼接，从而构成一个新字符串。图 9-23 是按开关的状态来决定采集工作状态的指示。

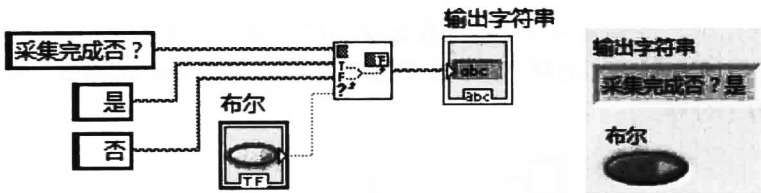


图 9-23 添加真/假字符串

(9) 字符串移位

该函数的作用是将输入字符串向左循环移动 1 位，图 9-24 是对字符串“ABCD123”进行移位，其结果是“BCD123A”。



图 9-24 字符串移位

(10) 反转字符串

该函数的作用是对输入字符串进行反转，但不适用于汉字，因为汉字要占 2 个字符位，反转后将不再是原来的汉字。图 9-25 是对字符串“ABCD123”进行反转，其结果为“321DCBA”。



图 9-25 反转字符串

www.91HMI.com

9.7 平化/还原字符串函数

平化/还原字符串函数是一组函数，主要包括平化至字符串、从字符串还原、变体至平化字符串转换、平化字符串至变体转换、平化至 XML、从 XML 还原、平化至 JSON、从 JSON 还原等函数。

使用平化/还原字符串函数的方法是，在后面板选取“函数→编程→字符串→平化/还原字符串”，再选取对应的函数。

(1) 平化至字符串

该函数的作用是使输入数据转换为二进制值组成的平化数据字符串。图 9-26 是将字符串转换为二进制字符串。

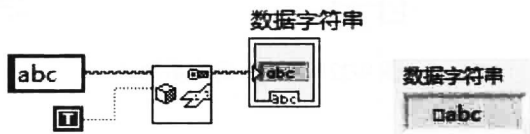


图 9-26 平化至字符串

(2) 从字符串还原

该函数的作用使二进制字符串转换为指定类型的数据。二进制字符串需包含类型的平化数据，也可使用该函数指定二进制字符串中数据的字节顺序或 endian 格式。图 9-27 是将二进制字符串还原为字符串。

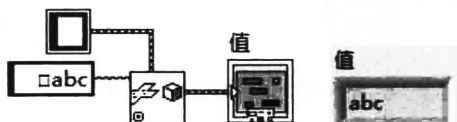


图 9-27 从字符串还原

(3) 变体至平化字符串转换

该函数的作用是将变体数据转换为平化字符串和表示数据类型的整数数组，ActiveX 变



体数据无法平化。图 9-28 是将数值转换为变体后，再执行平化字符串转换。

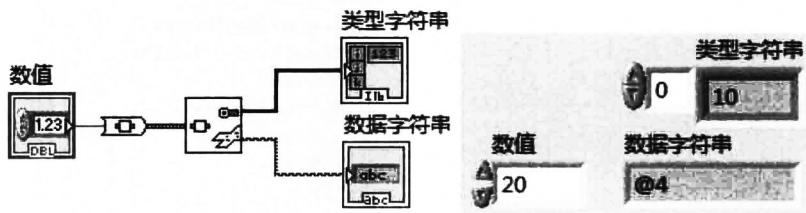


图 9-28 变体至平化字符串转换

(4) 平化字符串至变体转换

该函数的作用是使平化数据转换为变体数据。图 9-29 是将平化字符串中为变体数据。

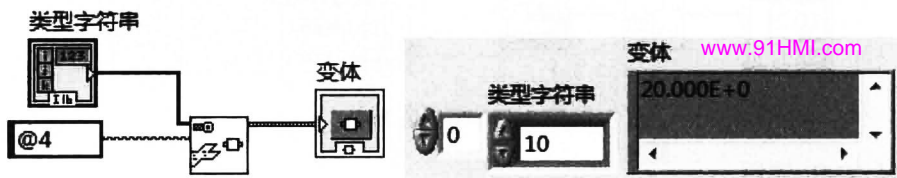


图 9-29 平化字符串至变体转换

(5) 平化至 JSON

该函数的作用是将输入数据转换为 UTF-8 编码的 JSON 字符串。图 9-30 是将数值型数据转换为 UTF-8 编码的 JSON 字符串。

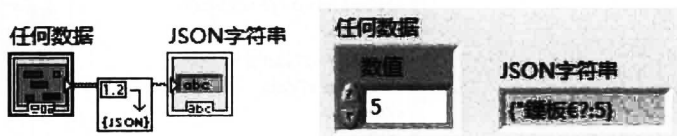


图 9-30 平化至 JSON

(6) 从 JSON 还原

该函数的作用是将 UTF-8 编码的 JSON 字符串转换为指定类型的数据。图 9-31 是将 UTF-8 编码的 JSON 字符串转换为数值型数据。

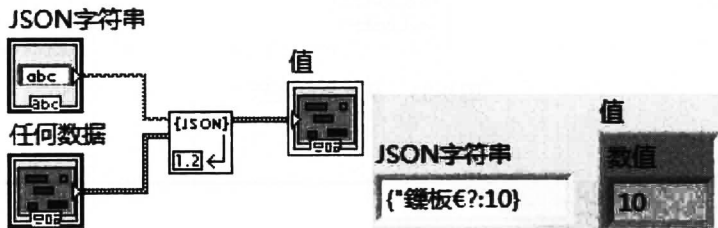


图 9-31 从 JSON 还原

(7) 平化至 XML

该函数的作用是根据 LabVIEW XML 模式转换为 XML 字符串。图 9-32 是将一个含有字符串、数值和布尔值的簇转换为 XML 字符串。

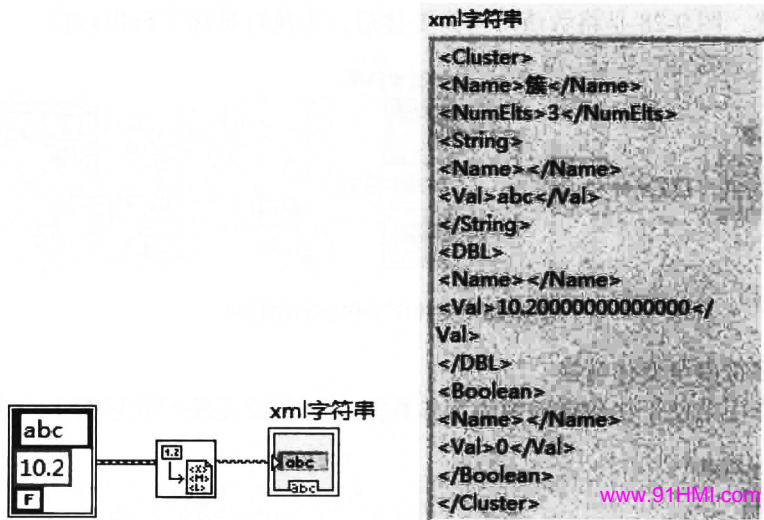


图 9-32 平化至 XML

(8) 从 XML 还原

该函数是平化至 XML 函数的逆操作，是将 XML 字符串还原为原数据类型的数据。图 9-33 是将 XML 字符串按指定的数据类型还原数据。

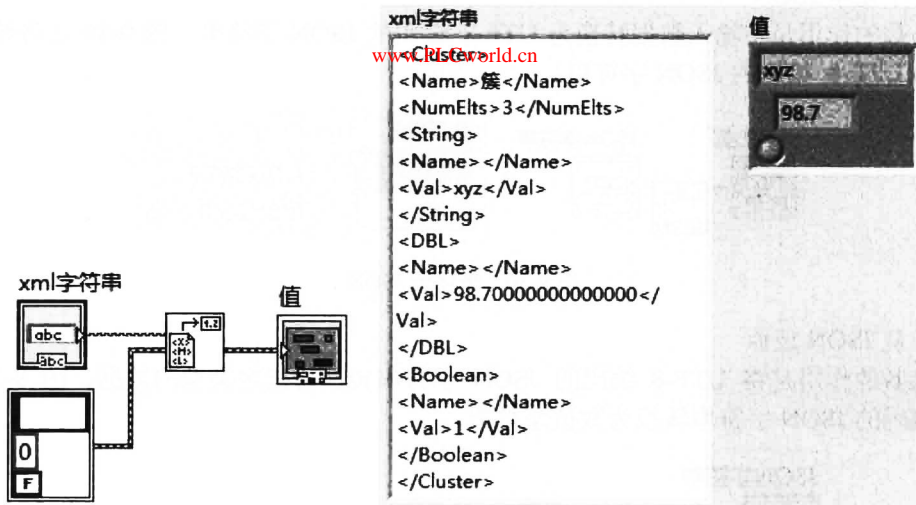


图 9-33 从 XML 还原

9.8 创建文本

该函数的作用是对文本和参数化输入进行组合来创建输出字符串。如输入的不是字符串，该 Express VI 可依据配置使其转化为字符串，图 9-34 是创建文本的对话框。

通过图 9-35 所示的程序框图，可将起始文本框中的信息与二进制文本框的二进制信息组合成新的字符串。



图 9-34 创建文本对话框

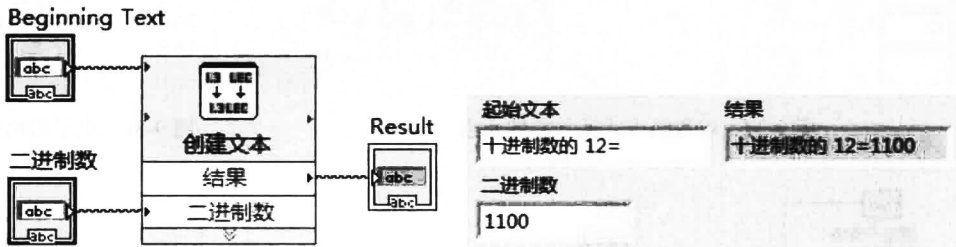


图 9-35 创建文本

9.9 综合实例：电子表格中任意数据的增删实例

电子表格是一种常用的数据格式文件，要实现电子表格中任意数据的增删不便直接解决，为此我们可以借助电子表格与数组的关系来加以解决，其思路是：

- ① 将电子表格转换为对应的数组。
- ② 利用数组的删除函数实现数组中行或列元素的删除，如图 9-36 所示。
- ③ 利用数组的插入函数实现数组中行或列元素的插入，如图 9-37 所示。
- ④ 将数组转换为对应的电子表格。

图 9-38 是将原电子表格变换为另一种结构的电子表格的界面，其程序框图如图 9-39 所示。

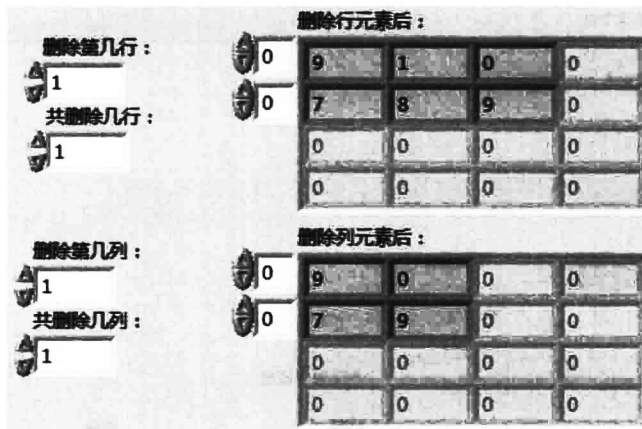


图 9-36 以数组方式删除数据元素

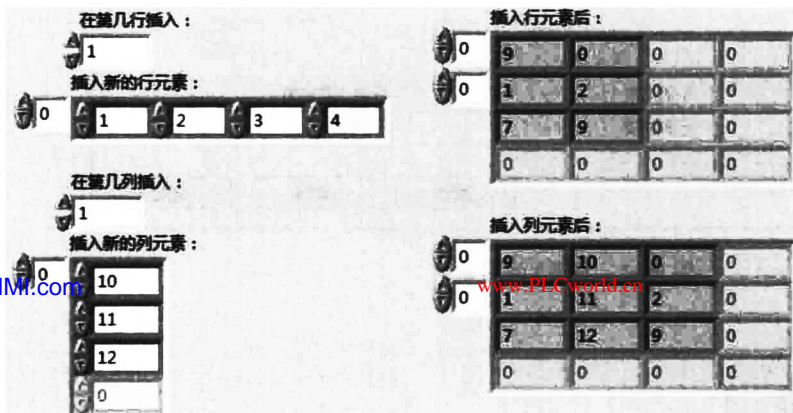


图 9-37 以数组方式插入数据元素

www.91HMI.com



图 9-38 电子表格界面

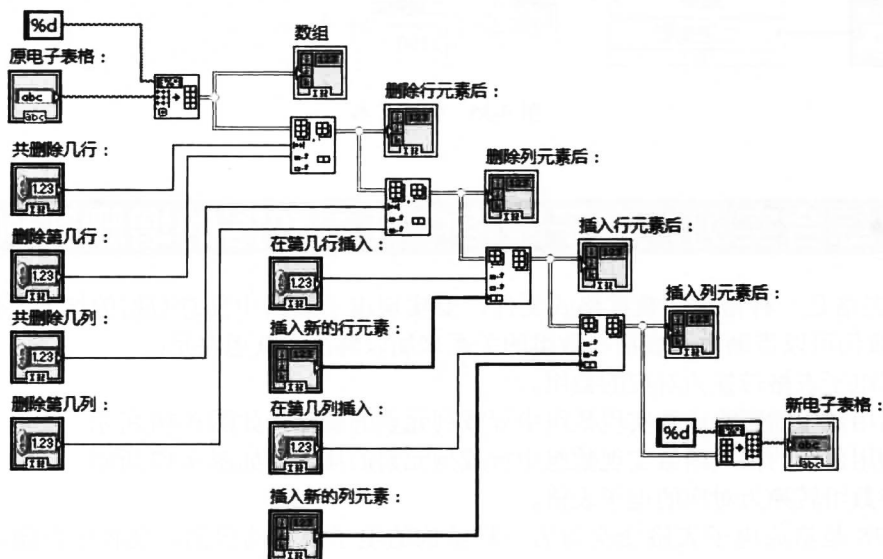


图 9-39 电子表格中增删数据的程序框图

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

www.PLCworld.cn



第 10 章

LabVIEW数学分析

在 LabVIEW 中, 为了便于数学分析配置了丰富的函数支持, 主要涉及基本数学分析、数理统计与最优化、曲线拟合与插值处理等。在数学分析中, 这些函数是最基本的, 也是最实用的函数。

【本章导读】

- ◆ 基本数学分析
- ◆ 数理统计与最优化
- ◆ 曲线拟合与插值
- ◆ 其他操作

www.PLCworld.cn

除了基本算术运算函数, LabVIEW 为了便于数据分析还提供了许多功能更为强大的函数, 通过对它们的使用, 可大幅提供解决问题的效率。

10.1 基本数学分析

10.1.1 初等与特殊函数

初等与特殊函数涉及一些较为复杂的函数, 如三角函数函数、指数函数、离散函数等, 具体内容如图 10-1 所示。

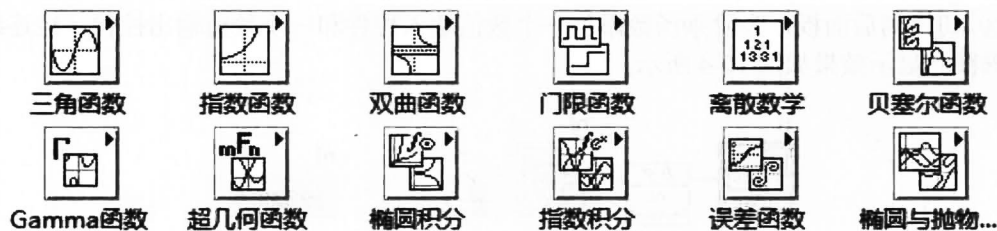


图 10-1 初等与特殊函数

(1) 三角函数

三角函数包括三角函数就是和反三角函数计算两部分, 三角函数是以角度为变量来求取函数值, 而反三角函数是通过函数值来求角度值。



提示

在三角函数及反三角函数运算中所用的角度值是以弧度为单位的，要使用角度进行计算需加以变化处理。

以计算 $\sin(x)$ 为例，其操作方法是：

- ① 在前面板添加一个数值输入控件和一个数值输出控件。
- ② 切换到后面板，选取“函数→数学→初等与特殊函数→三角函数→正弦”来添加该函数。
- ③ 依次添加数值常量、乘函数和除函数，经连线其程序框图和显示效果如图 10-2 所示。

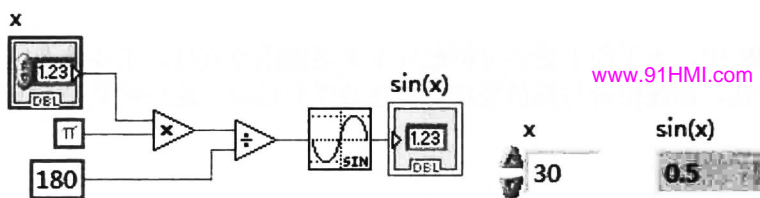


图 10-2 计算 $\sin(x)$

计算反三角函数的方法同计算三角函数基本相似，以计算反余弦函数为例，其程序框图和显示效果如图 10-3 所示。

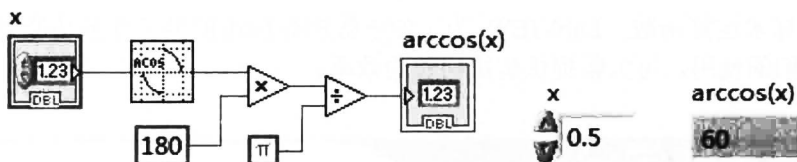


图 10-3 计算 $\arccos(x)$

(2) 离散函数

离散函数包括计算最大公倍数、最大公约数、质因数及阶乘等，其使用方法十分简单。

以计算 $n!$ 为例，其操作方法是：

- ① 在前面板，选取“函数→数学→初等与特殊函数→离散数学→阶乘”来添加该函数。
- ② 切换到后面板，为阶乘函数添加一个数值输入控件和一个数值输出控件，经连线其程序框图和显示效果如图 10-4 所示。



图 10-4 计算 $n!$

(3) 门限函数

门限函数主要用于生成门限脉冲信号，其操作方法极为简单。

以方波函数为例，其操作方法是：



- ① 在前面板，选取“函数→数学→初等与特殊函数→门限函数→方波函数”来添加该函数。
- ② 切换到后面板，为方波函数添加一个数值输入控件和一个数值输出控件，经连线其程序框图和显示效果如图 10-5 所示。

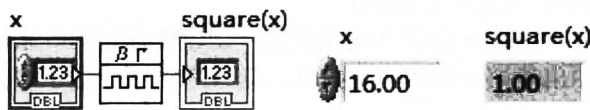


图 10-5 计算 square(x)

(4) 指数函数

指数函数主要用于计算指数和对数，其用法十分简便。

以计算 x 的 y 次方根为例，其操作方法是：

- ① 在前面板，选取“函数→数学→初等与特殊函数→指数函数→ x 的 y 次方根”来添加该函数。
- ② 切换到后面板，为该函数添加两个数值输入控件和一个数值输出控件，经连线其程序框图和显示效果如图 10-6 所示。

图 10-6 计算 x 的 y 次方根

10.1.2 线性代数

线性代数运算主要包括矩阵和行列式运算，所涉及的函数如图 10-7 所示。

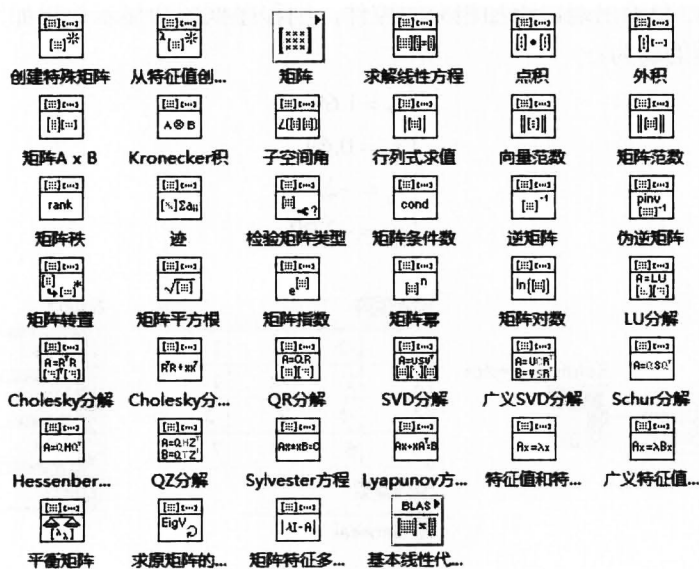


图 10-7 线性代数函数



(1) 创建矩阵

要进行矩阵运算就必须具有矩阵，为此要了解创建矩阵的方法。在矩阵中有一类特殊矩阵，如单位矩阵、对角矩阵及伴随矩阵等。

以创建特殊矩阵为例，其操作方法是：

- ① 在后面板，选取“函数→数学→线性代数→创建特殊矩阵”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-8 所示。

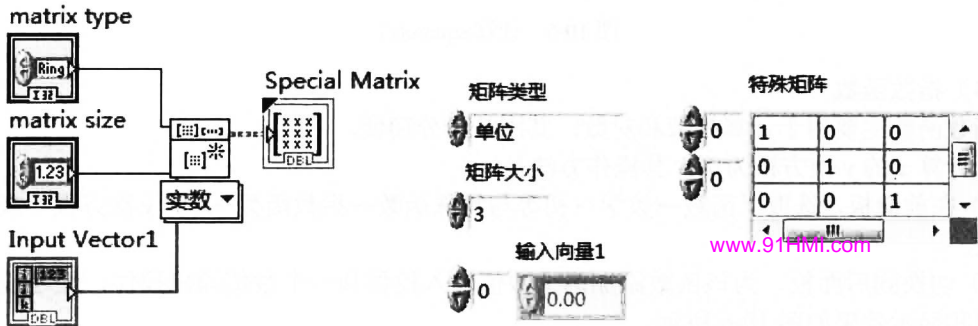


图 10-8 创建特殊矩阵

(2) 求解线性方程

利用求解线性方程函数可求解一个线性方程的根，它大大地简化了解线性方程的难度。以求解方程为例，其操作方法是：

$$\begin{cases} 2x_1 - x_2 - x_3 + x_4 = 2 \\ x_1 + x_2 - 2x_3 + x_4 = 4 \\ 4x_1 - 6x_2 + 2x_3 - 2x_4 = 4 \\ 3x_1 + 6x_2 - 9x_3 + 7x_4 = 9 \end{cases}$$

- ① 在后面板，选取“函数→数学→线性代数→求解线性方程”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-9 所示。由此可知方程的解为：

$$\begin{cases} x_1 = 1.69 \\ x_2 = 0.69 \\ x_3 = -2.33 \\ x_4 = -3.00 \end{cases}$$



图 10-9 求解线性方程



(3) 求逆矩阵

利用逆矩阵函数可求原矩阵的逆矩阵。

以求矩阵 $A = \begin{Bmatrix} 1 & 2 & 3 \\ 2 & 2 & 1 \\ 3 & 4 & 3 \end{Bmatrix}$ 的逆矩阵为例，其操作方法是：

- ① 在后面板，选取“函数→数学→线性代数→矩阵”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-10 所示。

由此可知逆矩阵 A^{-1} 为： $\begin{Bmatrix} 1 & 3 & -2 \\ -1.5 & -3 & 2.5 \\ 1 & 1 & -1 \end{Bmatrix}$

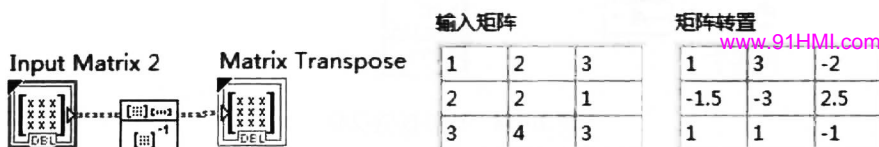


图 10-10 求逆矩阵

(4) 求矩阵乘法

在色度颜色模型中，有基于 CIE 标准的 RGB、XYZ 及 LAB 模型，将 RGB 色彩模型转

换为 XYZ 模型的公式为：
$$\begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix} = \begin{Bmatrix} 0.412 & 0.358 & 0.180 \\ 0.213 & 0.7156 & 0.072 \\ 0.019 & 0.119 & 0.950 \end{Bmatrix} \times \begin{Bmatrix} R \\ G \\ B \end{Bmatrix}$$

已知 RGB 值求 XYZ 值的操作方法是：

- ① 在后面板，选取“函数→数学→线性代数→矩阵 A*B”来添加该函数。
 - ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-11 所示。
- 由此可知，当 RGB=(200,10,15)时，XYZ=(88.68,150.83,19.24)。

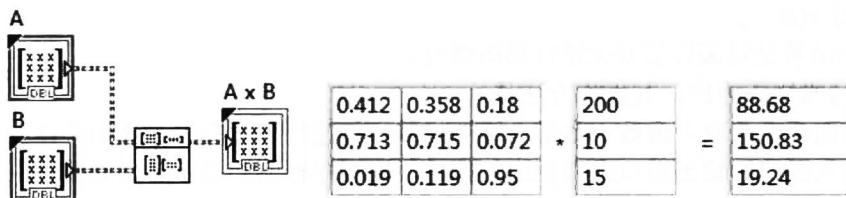


图 10-11 计算 XYZ 值

10.1.3 微分与积分

微分与积分功能提供了多种积分函数和求导函数，具体涉及的函数如图 10-12 所示。

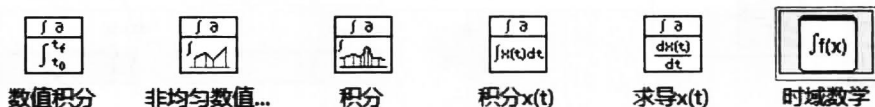


图 10-12 微分与积分函数



(1) 计算数值积分

数值积分是在 4 种常用的积分方法中选取一种来对输入数组进行数值积分运算。

计算图 10-13 所示直角梯形面积，其操作方法是：

① 在后面板，选取“函数→数学→积分与微分→数值积分”来添加该函数。

② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-14 所示。

由结果可知，该直角梯形的面积为 7.5，即 $S=(1+4)/2 \times 3$ 。

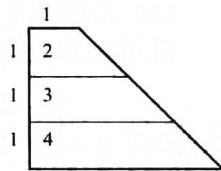


图 10-13 直角梯形

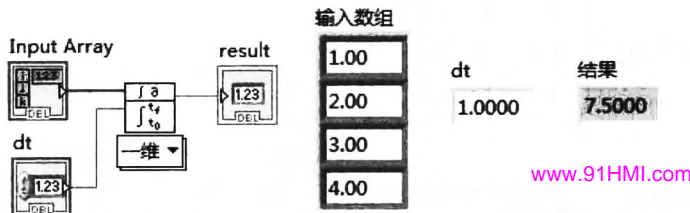


图 10-14 计算数值积分

(2) 计算积分

积分函数是对指定区间上的输入表达式进行积分运算。

以计算 $\int_0^3 3\sin(x)$ 为例，其操作方法是：

① 在后面板，选取“函数→数学→积分与微分→积分”来添加该函数。

② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-15 所示。

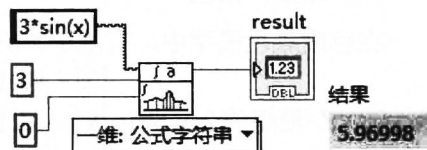


图 10-15 计算积分

由结果可知， $\int_0^3 3\sin(x) = 5.96998$ 。

(3) 求导 $x(t)$

求导 $x(t)$ 函数是对采样信号 x 进行离散微分。

以创建特殊矩阵为例，其操作方法是：

① 在后面板，选取“函数→数学→线性代数→创建特殊矩阵”来添加该函数。

② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-16 所示。

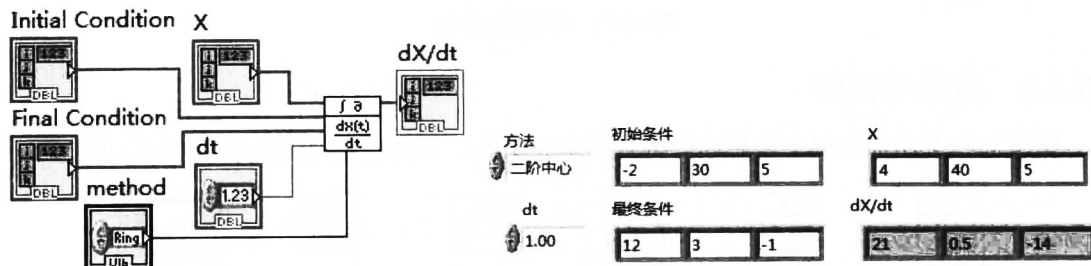


图 10-16 求导 $x(t)$



10.1.4 微分方程

微分方程函数用于求解一般微分方程，所涉及的函数如表 10-1 所示。

表 10-1 微分方程函数

函数名称	图标	函数功能
常微分方程		提供库塔法、欧拉法及卡普法求解微分方程
偏微分方程		采用 PDE 来处理偏微分方程

(1) 库塔法

ODE 库塔四阶法是一种求解带初始条件常微分方程的方法，用该法求解的操作方法是：

① 在后面板，添加一个波形图控件。

www.91HMI.com

② 切换至后面板，选取“函数→数学→微分方程→常微分方程→ODE 库塔四阶方法”

来添加该函数。

③ 为相应的输入、输出端口添加相应的控件，则程序框图和显示效果分别如图 10-17 和图 10-18 所示。

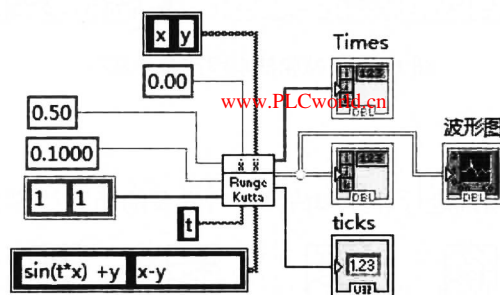


图 10-17 库塔法求解程序框图

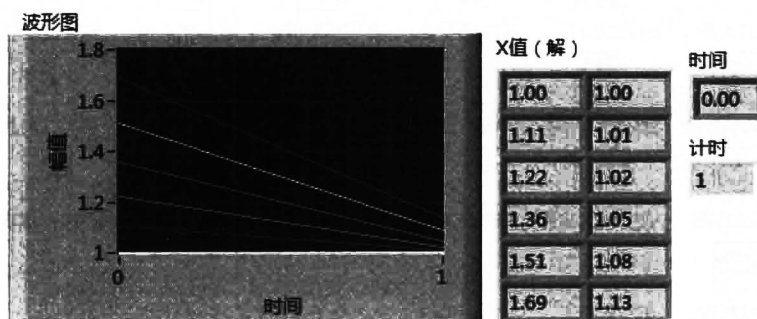


图 10-18 库塔法求解效果

(2) 欧拉法

与库塔法一样，欧拉法也是一种求解带初始条件常微分方程的方法，其求解的方法同库塔法相近。

以欧拉法求解上题的操作方法是，选取库塔法图标，然后激活快捷菜单从中选取“替



换→常微分方程选板→ODE 欧拉方法”即可，替换后的程序框图和显示效果如图 10-19 所示。经对比可发现，采用的方法不同其结果也稍有差异。

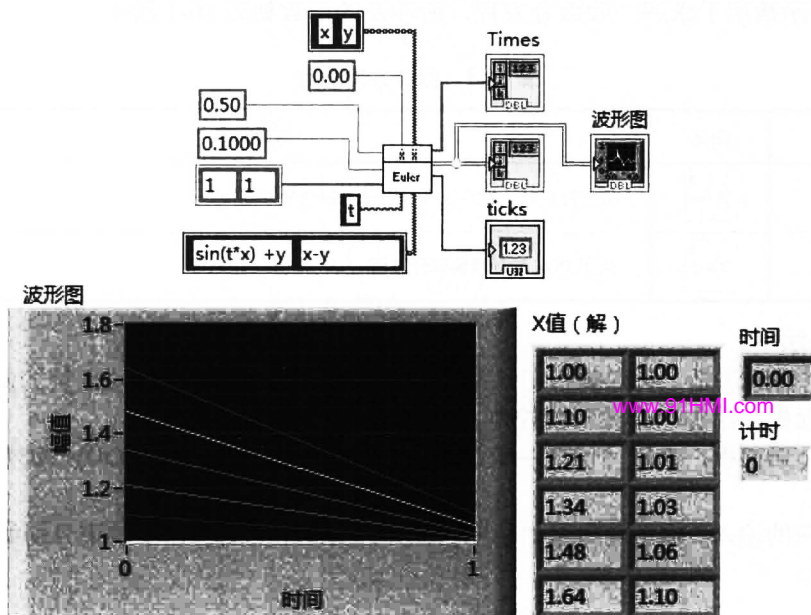


图 10-19 欧拉法求解常微分方程

10.1.5 多项式

多项式函数用于对多项式进行计算和估算，所涉及的函数如图 10-20 所示。

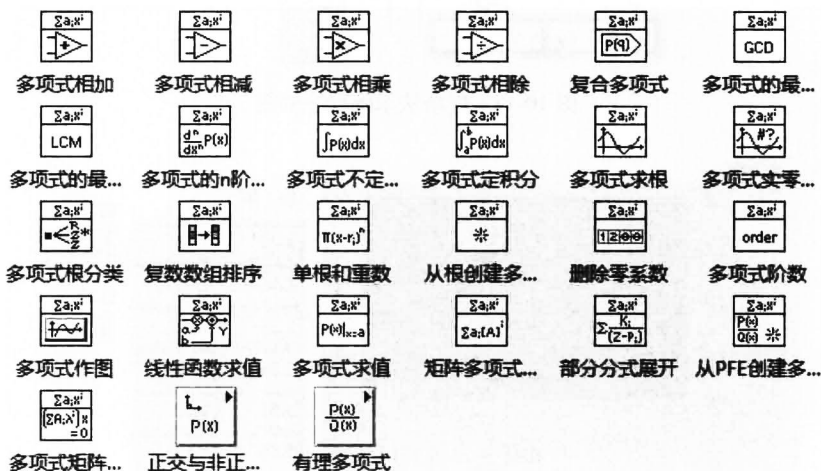


图 10-20 多项式函数

使用多项式函数可对多项式进行相应的运算，如对多项式的加减乘除运算等。

以多项式相加为例，其操作方法是：

- ① 在后面板，选取“函数→数学→多项式→多项式相加”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-21 所示。

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

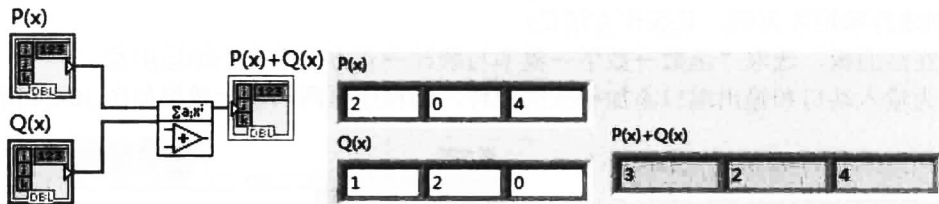


图 10-21 计算多项式

10.2 数理统计与最优化

10.2.1 概率与统计

概率与统计函数主要用于研究和揭示各种潜在在数据背后的统计规律，所涉及的函数如图 10-22 所示。



图 10-22 概率与统计函数

(1) 求均值

在统计学中，对检测的目标称为样本，并通过样本来反映整体检测目标的状况，均值就是一个反映整体状况的指标。

以对给定样本求均值为例，其操作方法是：

- ① 在后面板，选取“函数→数学→概率与统计→均值”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-23 所示。由结果可知均值为 $24.91 = (24.90 + 25 + 24.85 + 24.89) / 4$

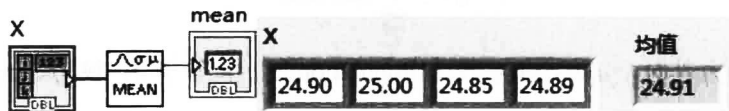


图 10-23 求均值

(2) 制作直方图

直方图是一种直观反映数据分布情况的方式，使用直方图函数可以便捷地生成直方图。



以创建特殊矩阵为例，其操作方法是：

- ① 在后面板，选取“函数→数学→概率与统计→直方图”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-24 所示。

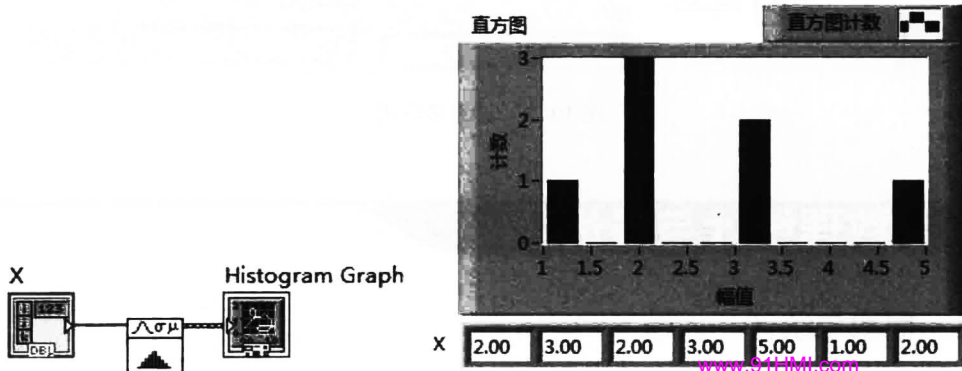


图 10-24 制作直方图

(3) 相关系数

相关系数也称积差相关系数或 Pearson 相关系数，其操作方法是：

- ① 在后面板，选取“函数→数学→概率与统计→相关系数”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-25 所示。



图 10-25 相关系数

(4) 计算标准差和方差

使用标准差和方差函数可计算序列 X 的均值、标准差和方差，其计算公式如下：

$$\mu = \sum_{i=0}^{n-1} \frac{x_i}{n}$$

式中， μ 为均值； n 是 X 中的元素数。

标准差 $= \sigma$ ，

$$\sigma^2 = \sum_{i=0}^{n-1} \frac{(x_i - \mu)^2}{w}$$

式中， σ^2 为方差时， μ 为均值；权为 Population 时， w 等于 n ；权为 Sample 时， w 等于 $(n-1)$ 。

求标准差及方差的具体操作是：

- ① 在前面板，放置一个波形图控件并将曲线显示模式设为点模式。
- ② 切换至后面板，选取“函数→数学→概率与统计→标准差和方差”来添加该函数。
- ③ 为输出端口添加相应的显示控件，为“权（采样）”端口创建一个输入控件。



④ 为生成采样样本，构建一个循环 10 次的循环，并利用随机函数和乘函数来产生 0~20 间的数值。经连线，程序框图和显示效果如图 10-26 所示。

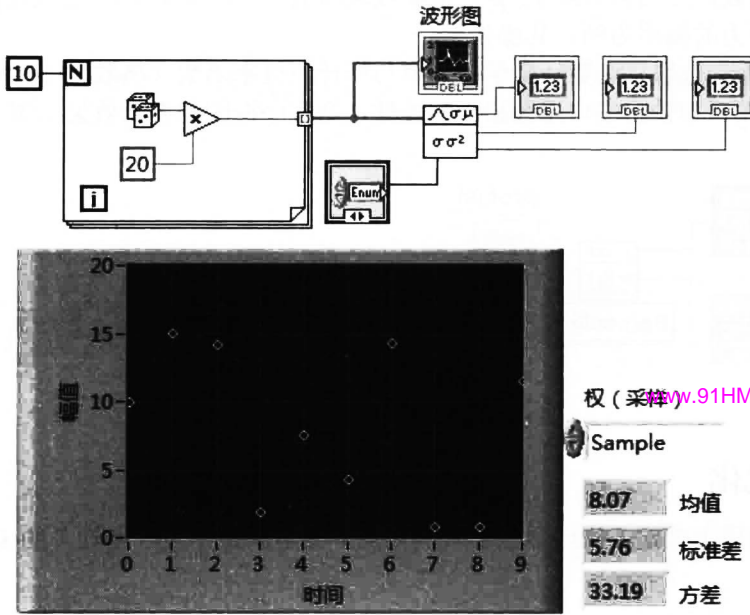


图 10-26 计算标准差和方差

(5) 统计计算

统计是一种通过数据发现事物背后规律的方式，使用 LabVIEW 提供的统计函数可以快速地计算出样本值的算术平均、中值、累加值、方差及极值等。

统计计算的基本操作是：

- ① 在后面板，选取“函数→数学→概率与统计→统计”来添加该函数。
- ② 为输出口添加相应的显示控件。
- ③ 为生成采样样本，构建一个循环 10 次的循环，并利用随机函数和乘函数来产生 0~20 间的数值。经连线，程序框图和显示效果如图 10-27 所示。

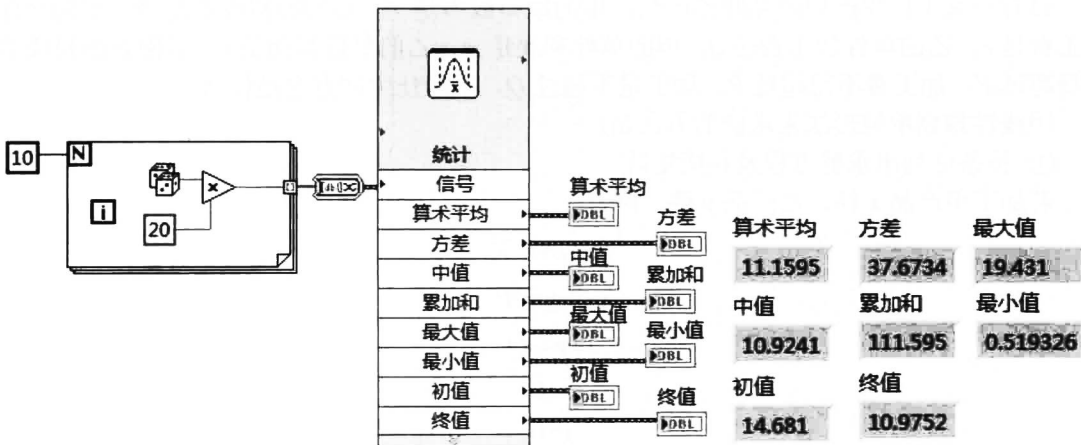


图 10-27 统计计算



(6) 概率计算

概率函数（离散）用于对随机变量 X 求概率值。对于符合 Bernoulli 分布的概率来讲， x 是试验成功的次数，必须为 0 或 1。 p 是试验成功的概率（概率 $x=1$ ），必须在 $[0,1]$ 之间。

以计算贝努力的概率为例，其操作是：

- ① 在后面板，选取“函数→数学→概率与统计→概率函数（离散）”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-28 所示。

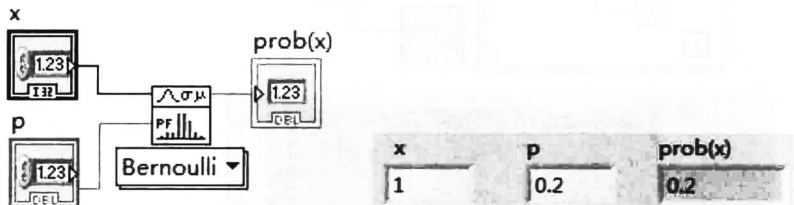


图 10-28 计算离散型概率

www.91HMI.com

10.2.2 最优化

最优化函数用于确定一维或多维实数的局部最大值和最小值，所涉及的函数如图 10-29 所示。



图 10-29 最优化函数

(1) 线性规划单纯形法

线性规划问题主要用于求解线性编程问题，如求最佳配比方案等。

设有一家工厂生产甲产品和乙产品，甲的原料成本是 a ，乙的原料成本是 b ，甲的单品加工费是 c ，乙的单品加工费是 d ，甲的单品利润是 m ，乙的单品利润是 n ，若限定原料成本不得超过 V ，加工费不得超过 P ，加工量不超过 Q ，求最佳生产方案是什么？

用线性规划单纯形法来求解的方法是：

- ① 按条件列出求解方程及相应矩阵。

若加工甲产品 x 件，乙产品 y 件，则有：

$$ax+by \leq V$$

$$cx+dy \leq P$$

$$x+y \leq Q$$

$$x > 0$$

$$y > 0$$

根据线性规划单纯形法对符号的规定，可将矩阵写为：



$$M = \begin{pmatrix} -a & -b \\ -c & -d \\ -1 & -1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \quad B = \begin{pmatrix} -V \\ -P \\ -Q \\ 0 \\ 0 \end{pmatrix} \quad C = [m \ n]$$

② 在后面板, 选取“函数→数学→最优化→线性规划单纯形法”来添加该函数, 为输入端口和输出端口添加相应的控件, 则程序框图和显示效果分别如图 10-30 和图 10-31 所示。

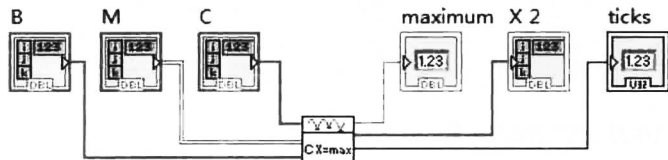


图 10-30 线性规划单纯形法程序框图

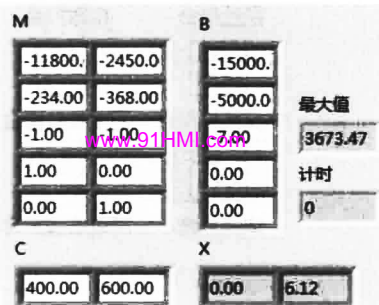


图 10-31 线性规划单纯形法界面

(2) 多元函数所有最小值

多元函数所有最小值函数用于计算 n 维函数在给定区间上的最小值。

以求函数 $-\sin 3x \cos 5y$ 的最小值为例, 其操作方法如下。

- ① 在后面板, 选取“函数→数学→最优化→多元函数所有最小值”来添加该函数。
- ② 为相应的输入和输出端口添加相应的控件。
- ③ 添加一个 while 循环, 并为停止条件添加一个输入控件。

④ 将全部图标移到循环内部, 再为循环添加一个 100 毫秒的延时环节, 经连线其程序框图和显示效果如图 10-32 所示。

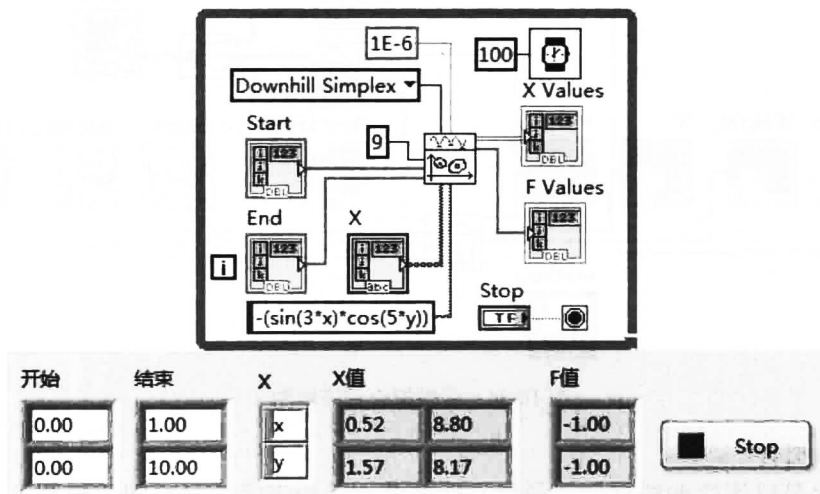


图 10-32 求多元函数所有最小值



10.3 曲线拟合与插值

10.3.1 拟合

拟合是指已知某函数的若干离散函数，通过调整该函数中若干待定系数使该函数曲线与已知点集的差别最小，它所涉及的函数如图 10-33 所示。

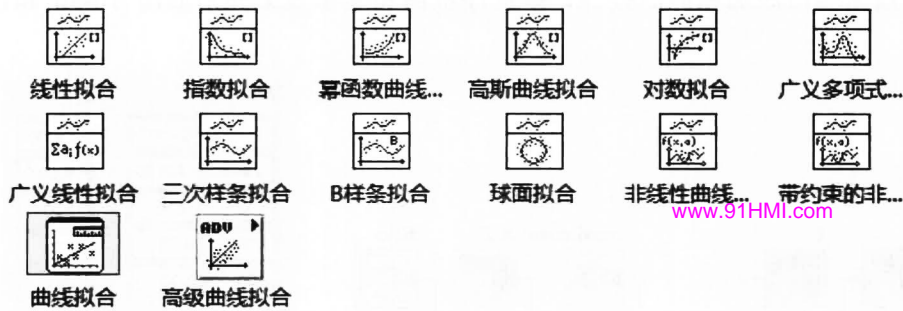


图 10-33 曲线拟合函数

(1) 线性拟合

线性拟合是以最小二乘法、最小绝对残差等方法获取数据集 (X, Y) 的方法。线性拟合的验证操作如下。

- ① 在前面板，添加一个波形图。
- ② 切换至后面板，选取“函数→数学→拟合→线性拟合”来添加该函数。
- ③ 为输入端口和输出端口添加相应的控件。
- ④ 选取“函数→Express→信号操作→合并信号”来添加该函数，经连线其程序框图和显示效果分别如图 10-34 和图 10-35 所示。

注：在波形图中，白线代表 X 值，红线代表 Y 值，绿线代表拟合值。

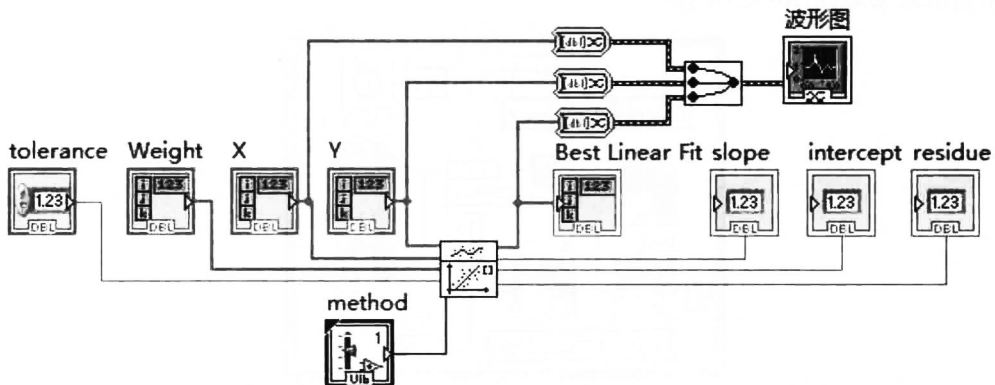
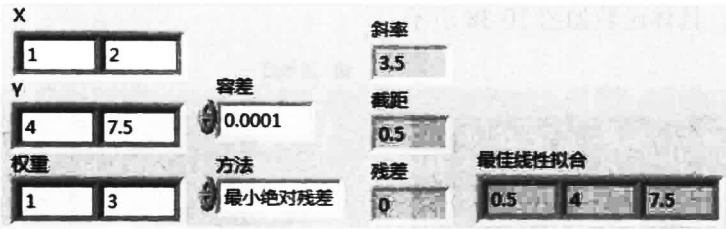


图 10-34 线性拟合程序框图

(2) 曲线拟合

曲线拟合是以连续曲线近似地替代平面离散点组所表示的坐标间的函数关系的一种数据处理方法。



波形图

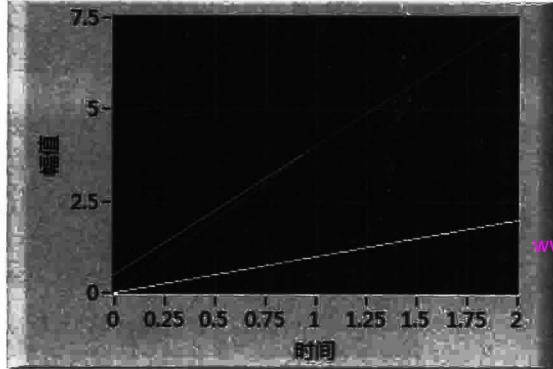


图 10-35 线性拟合效果

以正弦信号同噪声信号叠加构造模拟信号为例，其操作方法如下。

① 在前面板，添加 3 个波形图控件。

② 转换至后面板，选取“函数→数学→拟合→曲线拟合”来添加该函数。

③ 添加一个基于持续时间的信号发生器和一个均匀白噪声信号，在添加 2 个合并信号图标。

④ 为相应的输入端口和输出端口添加相应的控件，则程序框图和显示效果分别如图 10-36 和图 10-37 所示。

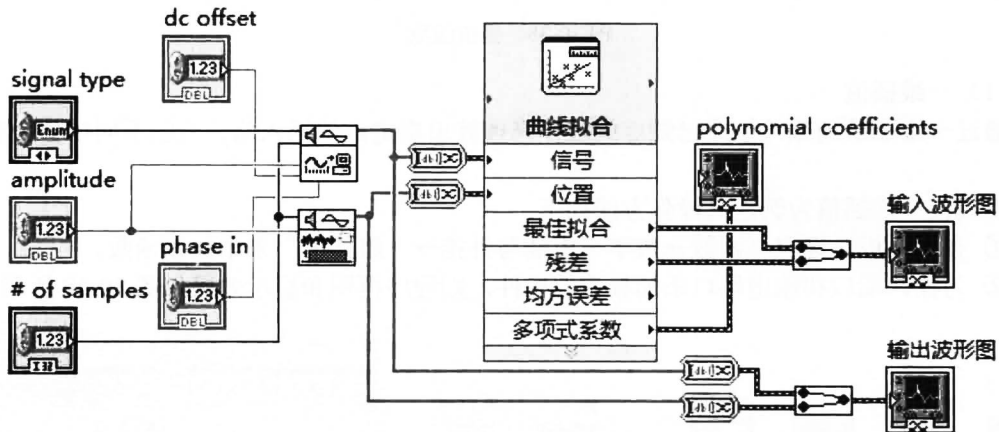


图 10-36 曲线拟合程序框图

10.3.2 插值

插值函数涉及内插和外推两类函数，主要包括行一维和二维插值、分段插值、多项式插



值及傅里叶插值，具体函数如图 10-38 所示。

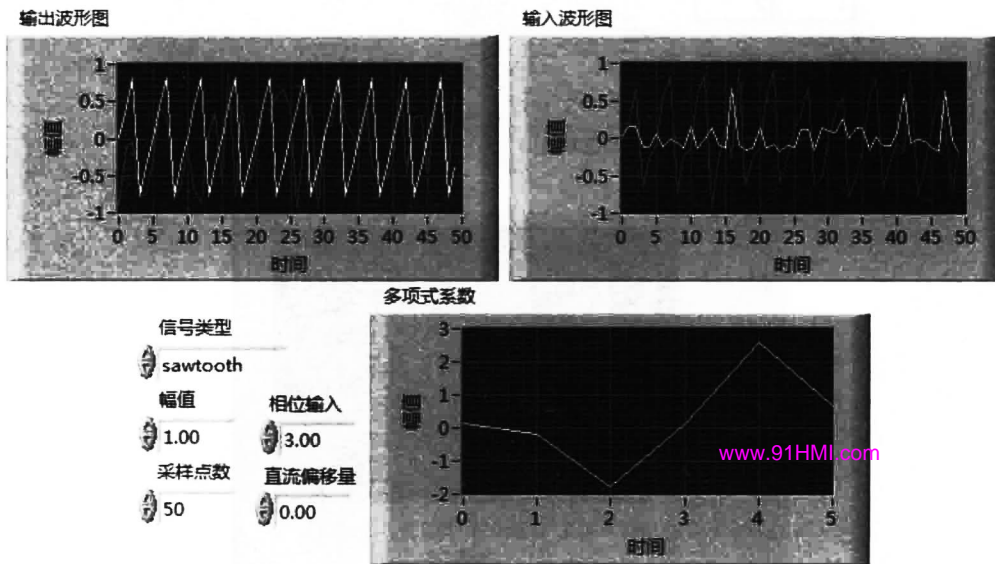


图 10-37 曲线拟合效果



图 10-38 插值函数

(1) 一维插值

通过一维插值函数可按照已知点的规律来推算出指定点的插入值，方法不同插值会有所差异。

以线性一维插值为例，其操作方法如下。

- ① 在后面板，选取“函数→数学→内插与外推→一维插值”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-39 所示。

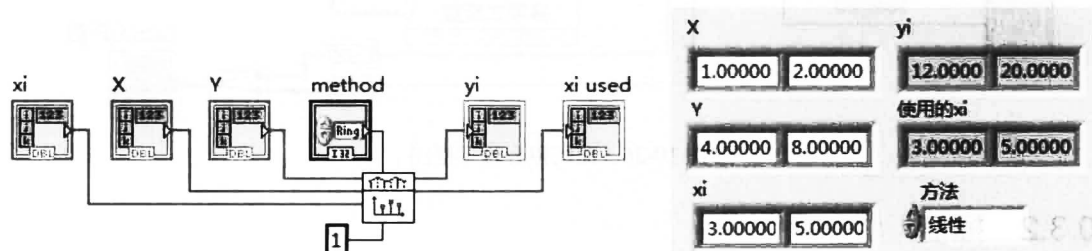


图 10-39 一维插值

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

www.91HMI.com

www.PLCworld.cn



由结果可得到表 10-2，它可反映出数据间的关系是线性化的。

表 10-2 数据关系表

X 值	1.0	2.0	3.0	5.0
Y 值	4.0	8.0	12.0	20.0

(2) 样条内插

样条内插是对给定的 $(x[i], y[i])$ ，按二阶导数来获取样条插值。

以样条内插为例，其操作方法如下。

- ① 在后面板，添加一个 XY 图控件。
- ② 切换至后面板，选取“函数→数学→内插与外推→样条内插（前）”来添加该函数。
- ③ 再添加一个样条内插（后）函数，然后为相应的端口添加输入、输出控件。经连线，其程序框图和显示效果分别如图 10-40 和图 10-41 所示。

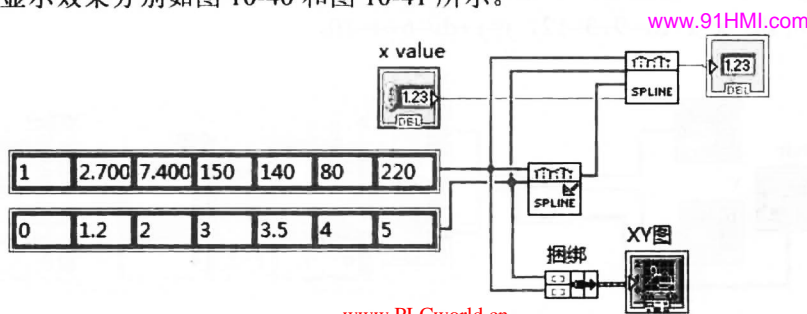


图 10-40 样条内插程序框图

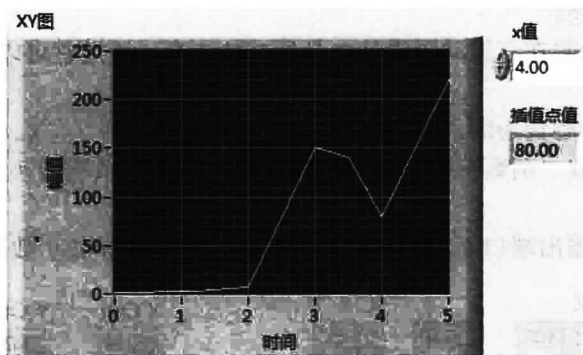


图 10-41 样条内插效果

注：通过设置 x 值，就可得到对应的插值点的值。XY 图为一维数组 X 和 Y 所定义的样条曲线。

10.4 其他操作

10.4.1 几何

几何函数用于对二维坐标和三维坐标进行运算，所涉及的函数如图 10-42 所示。

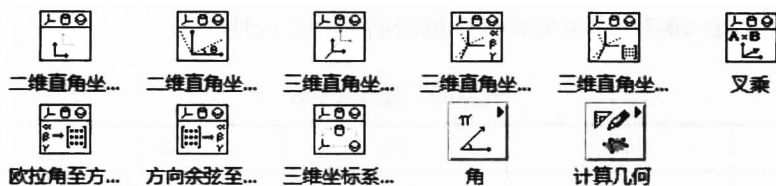


图 10-42 几何

(1) 二维直角坐标系平移

二维直角坐标系平移函数对平移处理的坐标值计算有重要作用，在制作二维动画时用它快速的确定平移后的位置。

以计算一个坐标点的平移值为例，其操作方法如下。

- ① 在后面板，选取“函数→数学→几何→二维直角坐标系平移”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-43 所示。由结果可知： $x=x+dx=9+3=12$ ， $y=y+dy=6+4=10$ 。

www.91HMI.com

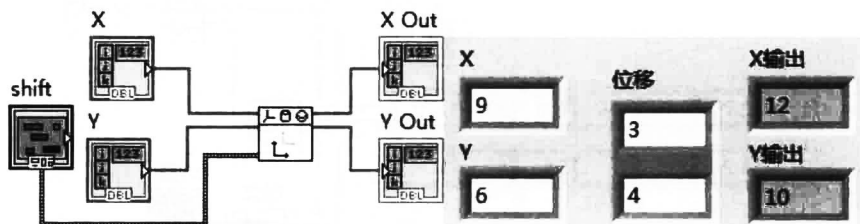


图 10-43 二维直角坐标系平移

(2) 三维坐标系旋转

三维坐标系旋转函数在三维动画设计中是便捷的工具，通过它可以迅速地求出旋转后的新坐标值。

以计算一个坐标点旋转为例，其操作方法是：

- ① 在后面板，选取“函数→数学→几何→三维直角坐标系旋转（方向余弦）”来添加该函数。
- ② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果如图 10-44 所示。

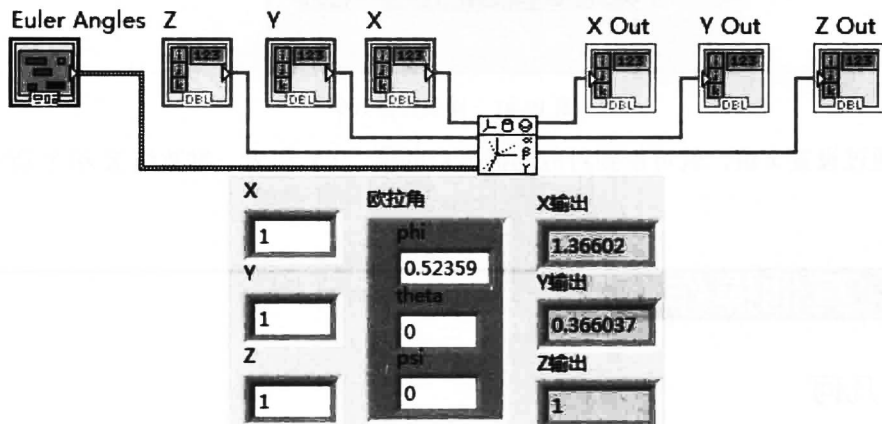


图 10-44 三维坐标系旋转

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



注：该示例是对坐标点 (1,1,1) 旋转 30 度角 (0.52359 弧度) 后的坐标点 (0.366025, 1.36603, 1)。

10.4.2 脚本与公式

脚本与公式函数主要是按指定数学公式和表达式来编写公式或程序以进行计算，所涉及的函数如图 10-45 所示。

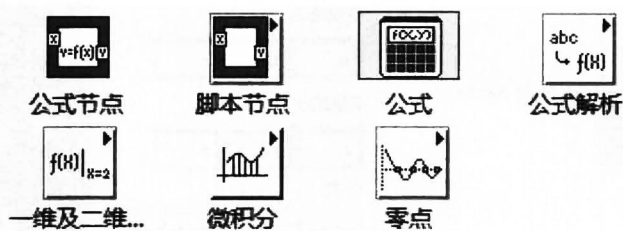


图 10-45 脚本与公式

www.91HMI.com

(1) 公式节点

公式节点用于解决复杂公式的编程问题，它可以有效地提高设计的效率。

以求解公式 $y=3x+4z$ ， $p=q^2-5$ 为例，其操作方法是：

① 在后面板，选取“函数→数学→脚本与公式→公式节点”来添加该函数。

② 在公式节点内部输入计算公式，并以“;”来分隔语句。

③ 用鼠标右击公式节点的左侧，从快捷菜单中选取“添加输入”来添加 3 个变量输入端并对其命名。再选取公式节点的右侧来激活快捷菜单，从中选取“添加输出”来添加 2 个变量输出端并对其命名。

④ 为输入端赋值，为输出端添加显示控件，则程序框图和显示效果如图 10-46 所示。

有输出结果可知： $y=11.00(y=3\times 1+4\times 2)$ ， $p=4.00(p=3\times 3-5)$ 。

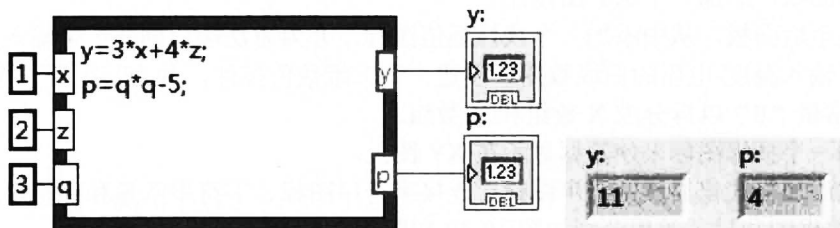


图 10-46 公式节点使用

(2) 解析公式节点

解析公式节点用于对输入值进行分析，从而输出分析公式的数量及相应公式操作的代码等。

解析公式节点的操作方法是：

① 在后面板，选取“函数→数学→脚本与公式→公式解析→解析公式节点”来添加该函数。

② 为输入端口和输出端口添加相应的控件，则程序框图和显示效果分别如图 10-47 和图 10-48 所示。

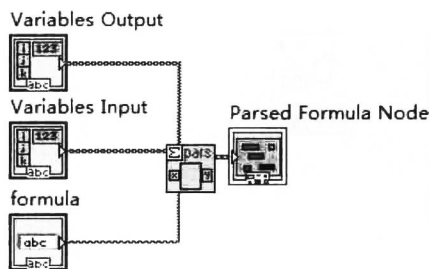


图 10-47 解析公式节点

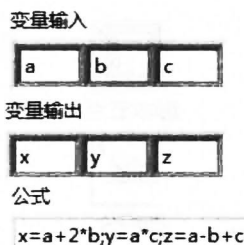


图 10-48 解析公式节点效果

解析公式节点



10.5 综合实例：基于最小二乘法的温度传感器校准处理

在温度测量仪表中，多用铂金属来作温度传感器，它可以工作在 $-200\sim 500^{\circ}\text{C}$ 环境下工作。PT100 就是众多温度传感器中的一种。为了校准该温度传感器，可用插值函数来辅助解决。具体的处理思路是经有限次采样来获取样本值，然后通过线性插值来求解出温度-电压曲线函数。

温度-电压校准界面如图 10-50 所示，其具体操作如下。

- ① 新建一个 VI，并予以命名保存。
- ② 在前面板，添加一个 XY 图控件。
- ③ 切换至后面板，从中添加一个线性插值图标，并为方法端口创建一个输入控件。
- ④ 为了输入温度-电压的曲线数据，创建一个二维数值控件，并为其配置一个索引数组图标和数值常量“0”以拆分成 X 数组和 Y 数组。
- ⑤ 添加一个捆绑图标来使数据显示在 XY 图上。
- ⑥ 依次添加格式化显然字符串图标、连接字符串图标、字符串常量和字符串显示控件，经连线其程序框图和和显示效果分别如图 10-49 和图 10-50 所示。

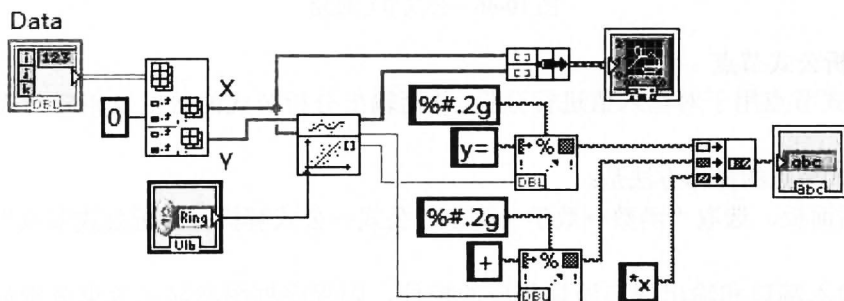
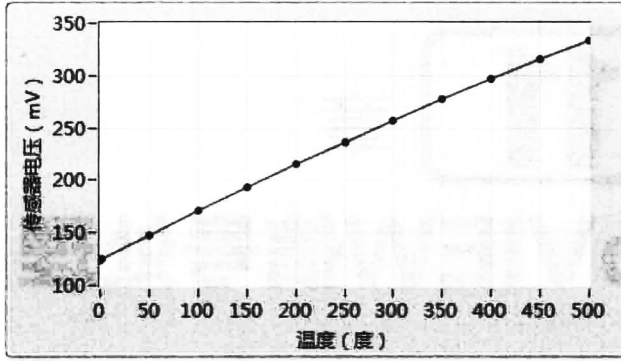


图 10-49 温度-电压校准程序框图



由波形图可知该温度传感器的温度-电压曲线是线性曲线，其方程为： $y=127.39+0.42x$

0	124.38
1	124.8
50	147.79
100	170.64
150	192.93
200	214.68
250	235.9
300	256.59
350	276.79
400	296.48
450	315.69
500	334.42



Least Square

$y=127.39+0.42x$

图 10-50 温度-电压校准界面



www.PLCworld.cn LabVIEW 提供的数学分析功能使该平台成为了解决数学问题的知识库。



第 11 章

LabVIEW信号处理

LabVIEW 在信号发生、分析和处理方面有着极强的优势，它既可以做信号发生器，也可以对信号进行采集、分析、处理及显示。本章主要对信号发生、分析和处理的相关函数及使用方法加以介绍。

【本章导读】

- ◆ 信号发生
- ◆ 信号时域分析
- ◆ 信号频域分析
- ◆ 信号调理
- ◆ 波形监测
- ◆ 逐点分析等

www.PLCworld.cn

11.1 信号发生

在 LabVIEW 中，信号是进行分析处理的主体。为此它提供了两种信号生成的方式，即信号生成方式和波形生成方式。

11.1.1 基本函数发生器

基本函数发生器是按信号规律所输出的函数关系较为简单的信号，详细内容如表 11-1 所示。

表 11-1 基本函数发生器

函数名称	图标	使用说明
正弦信号		根据指定的采样点数、幅值、相位和周期数生成一个正弦信号序列
三角波		根据指定的宽度、采样点数、幅值、延迟和不对称性产生一个三角波信号序列
脉冲信号		根据指定的采样点数、幅值、延迟周期和脉宽产生一个脉冲信号序列
坡形信号		根据指定的采样点数、初始值、结束值产生一个上升或下降斜坡形信号序列



续表

函数名称	图标	使用说明
正弦波		根据指定的采样点数、幅值、频率、初始相位产生一个正弦信号序列，并返回结束点的相位
三角波		根据指定的采样点数、幅值、频率、初始相位产生一个三角波序列，并返回结束点的相位
方波		根据指定的采样点数、幅值、频率、初始相位、占空比产生一个方波序列，并返回结束点的相位
锯齿波		根据指定的采样点数、幅值、频率、初始相位产生一个锯齿波序列，并返回结束点的相位

方波信号具有幅值、频率、采样数、占空比及初始相位等参数，其使用方法十分简单，具体的操作方法如下。

① 在前面板，添加一个波形图控件。

www.91HMI.com

② 切换至后面板，选取“函数→信号处理→信号生成→方波”来添加该信号。然后为该信号的输入端添加相应的输入控件，经连线其程序框图和显示效果如图 11-1 所示。

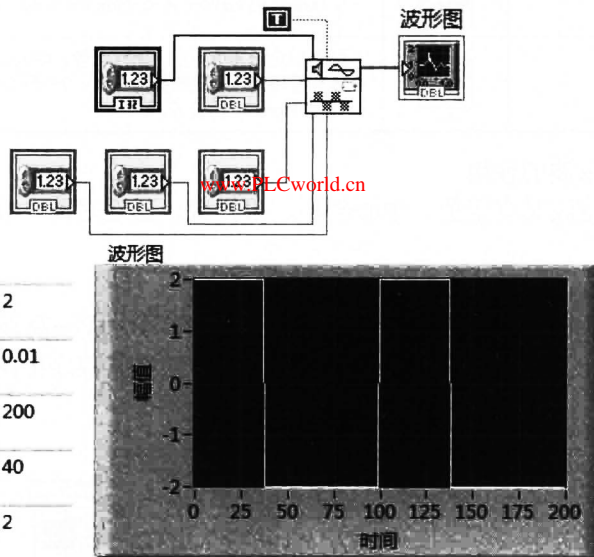


图 11-1 方波信号发生器

11.1.2 多频信号发生器

多频信号发生器要比基本函数发生器在函数关系上复杂些，其主要内容如表 11-2 所示。

表 11-2 多频信号发生器

函数名称	图标	使用说明
基于持续时间的信号发生器		根据指定的采样间隔、信号类型、采样点数、频率、幅值、直流偏置和初始相位生成一个信号序列
高斯调制正弦波		根据指定的衰减、中心频率、采样点数、幅值、延迟、时间间隔、归一化中心带宽生成一个高斯调制正弦波



续表

函数名称	图标	使用说明
高斯单脉冲		根据指定的中心频率、采样点数、幅值、时间分辨率生成一个高斯单脉冲
Sinc 信号		根据指定的采样点数、幅值、延时周期和时间间隔产生一个 Sinc 信号序列
周期 Sinc 信号		根据指定的采样点数、幅值、延时周期、阶数和时间间隔产生一个周期 Sinc 信号序列
Chrip 信号		根据指定的采样点数、幅值、上下截止频率产生一个扫频信号序列
任意波形发生器		以输入波形为周期，根据指定的采样点数、幅值、频率、初始相位、是否插值产生一个任意波形，并返回结束节点的相位
二进制 MLS		根据指定的采样点数、多选式阶数产生一个二进制最大长度序列 www.91HMI.com
冲激函数		根据指定的采样点数、幅值和延时周期产生一个冲激信号序列
准随机		生成准随机 Halton 或 Richtmeyer 序列，是差异性小的数字序列
脉冲序列		根据指定的插值方法、采样点数、时间间隔、幅值、延迟、脉冲原型产生一个脉冲号序列，选取的插值法有：0—最近插值、1—线性插值、2—样条插值、3—3 次 Hermite 插值等

高斯调制正弦波的使用

高斯调制正弦波具有幅值、中心频率、采样数、延迟时间、衰减及归一化带宽等参数，其使用方法如下。

① 在前面板，添加一个波形图控件。

② 切换至后面板，选取“函数→信号处理→信号生成→高斯调制正弦波”来添加该信号。然后为该信号的输入端添加相应的输入控件，经连线其程序框图和显示效果分别如图 11-2 和图 11-3 所示。

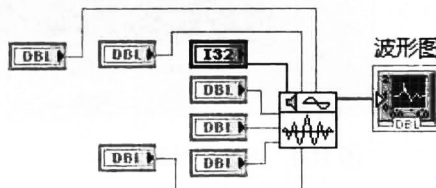


图 11-2 高斯调制正弦波程序框图

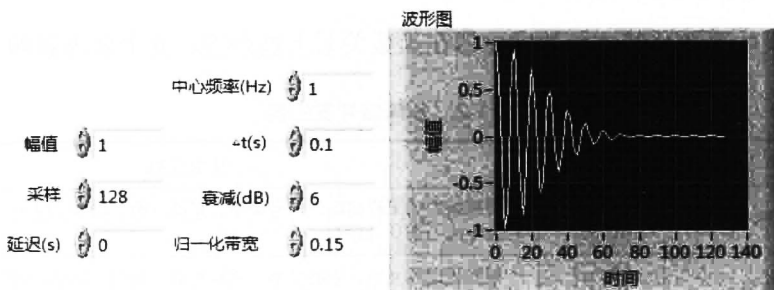


图 11-3 高斯调制正弦波界面



11.1.3 噪声信号发生器

噪声信号是模拟外部信号的有力工具，它为克服噪声的干扰提供了理论研究的模型。

表 11-3 噪声信号发生器

函数名称	图标	使用说明
混合单频与噪声		根据指定的采样点数、单频信号信息、噪声有效值、偏置、采样率生成一个信号序列
均匀白噪声		生成均匀分布的伪随机白噪声信号
高斯白噪声		生成符合高斯分布的伪随机白噪声信号
周期性随机噪声		生成周期性随机噪声信号序列 www.91HMI.com
Gamma 噪声		根据指定的采样点数、阶数生成噪声信号序列
泊松噪声		生成符合泊松分布的伪随机噪声信号序列
二项分布的噪声		生成符合二项分布的伪随机噪声信号序列
Bernoulli 噪声		生成符合贝努力分布的伪随机噪声信号序列 www.PLCworld.cn

高斯白噪声具有标准差、种子及采样数等参数，其使用方法如下。

① 在前面板，添加一个波形图控件。

② 切换至后面板，选取“函数→信号处理→信号生成→高斯白噪声”来添加该信号，

并为标准差端口添加输入控件。

③ 依次添加均值图标、分布测度图标和数据显示图标，经连线其程序框图和显示效果如图 11-4 所示。

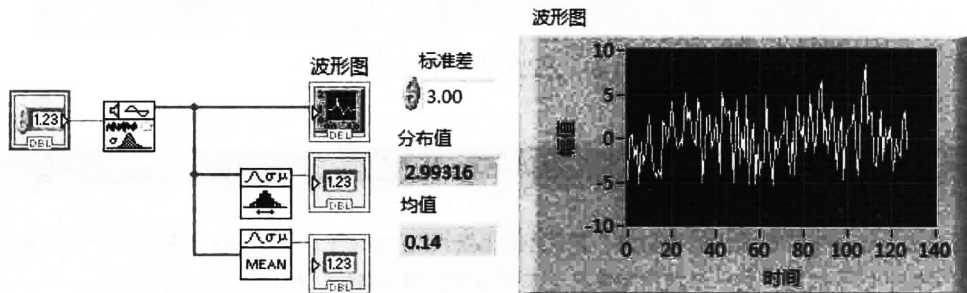


图 11-4 噪声信号发生器

由结果可知分布值与标准差是相关的，分布差越大，分布值也越大。而均值与标准差并无明显的相关性。

11.1.4 用公式节点产生信号

在 LabVIEW 中，不仅用信号发生器可输出数据信号，使用公式节点也能够产生数据信号。



以绘制李沙育图为例，使用公式节点的操作方法如下。

- ① 新建一个 VI，并在前面板放置一个波形图控件和一个 XY 图控件。
- ② 切换至后面板，选取“函数→数学→脚本与公式→公式节点”来添加该节点，然后在节点的两端分别添加两个输入和输出节点并加以命名。
- ③ 在节点内部输入相应的代码。
- ④ 为输入节点创建两个输入控件。
- ⑤ 添加一个信号合并图标和一个捆绑图标，经连线其程序框图和显示效果分别如图 11-5 和图 11-6 所示。

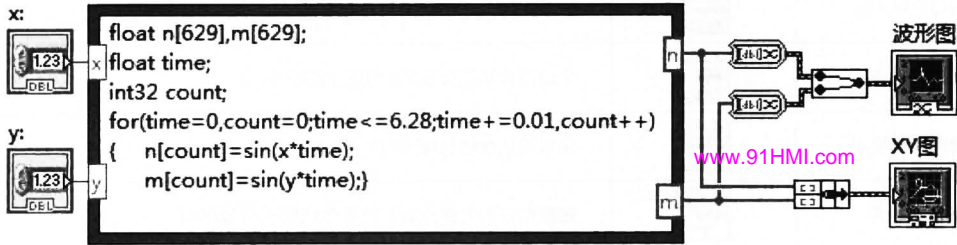


图 11-5 公式节点产生信号程序框图

运行该程序，由波形图可观察到函数 $\sin(xt)$ 和 $\sin(yt)$ 的波形，而由 XY 图则可看到二者合成后的李沙育图。调整 x 和 y ，可以形成不同的波形。

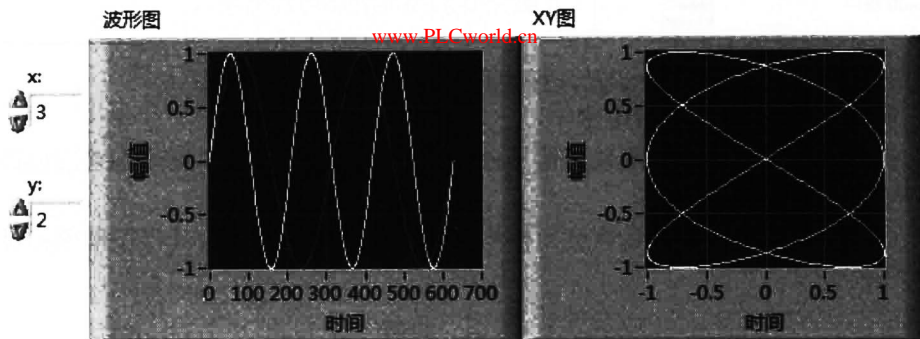


图 11-6 公式节点产生信号界面

提示

波形生成与信号生成有很大的相似性，它们都能输出相应的信号。两者的区别在于一个是用来产生数组类型的信号数据，另一个则是用来产生波形类型的信号数据，使用时应善加区别。

11.2 信号时域分析

时域分析是指在时间域内对信号进行波形变换、缩放、数值微分及积分等分析运算，并通过对不同时间段的分析，最终求出各段的最佳运行状态。时域分析函数的内容如图 11-7 所示。

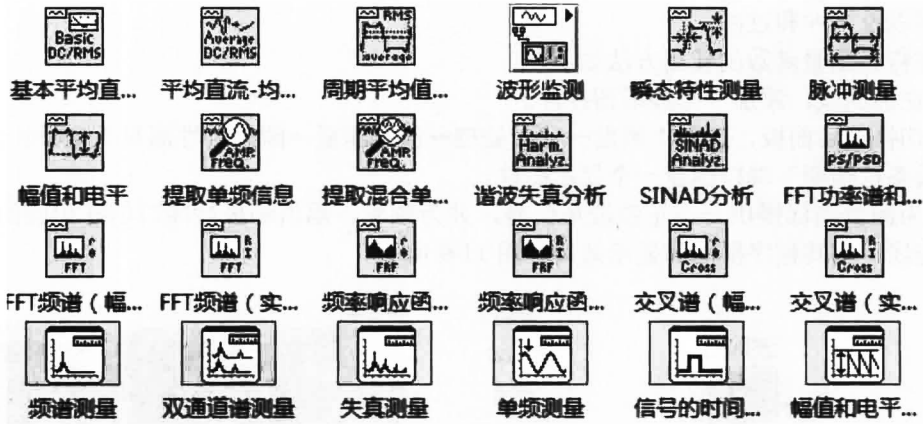


图 11-7 时域分析函数

www.91HMI.com

11.2.1 基本平均直流与均方根测量

基本平均直流-均方根函数用于测量输入波形或波形数组的直流值和均方根，其使用方法如下：

- ① 在前面板，添加一个波形图控件。
- ② 切换至后面板，选取“函数→信号处理→波形测量→基本平均直流-均方根”来添加该测量，再为其输出端口添加 2 个显示控件。
- ③ 为了提供测量的波形信号，选取“函数→信号处理→波形生成→正弦波形”来添加该波形，再为其频率和幅值端口创建输入控件。经连线，则其程序框图和显示效果如图 11-8 所示。

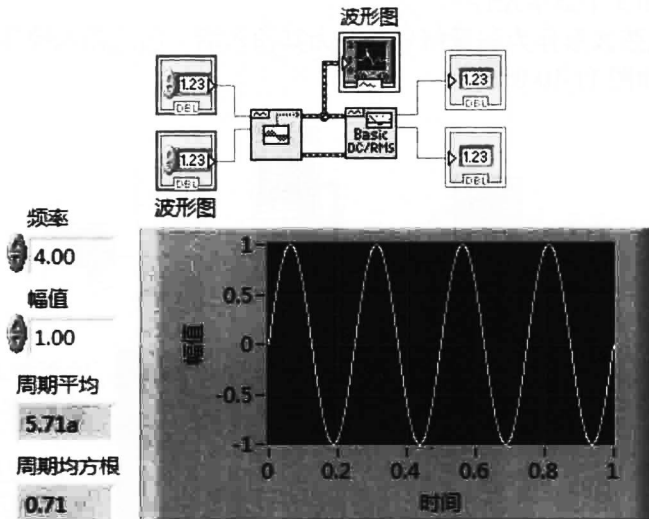


图 11-8 基本平均直流与均方根测量

11.2.2 瞬态测量

瞬态测量函数用于测量输入信号中选定正跃迁或负跃迁的瞬态持续期(上升/下降时间)、



边沿斜率以及下冲和过冲。

瞬态特性测量函数的使用方法如下。

- ① 在前面板，添加一个波形图控件。
- ② 切换至后面板，选取“函数→信号处理→波形测量→瞬态特性测量”来添加该测量，并为“瞬态持续期”端口创建一个显示控件。
- ③ 为测量图标提供一个正弦波形信号，并为频率、幅值和极性端口创建相应的输入控件。经连线，则其程序框图和显示效果如图 11-9 所示。

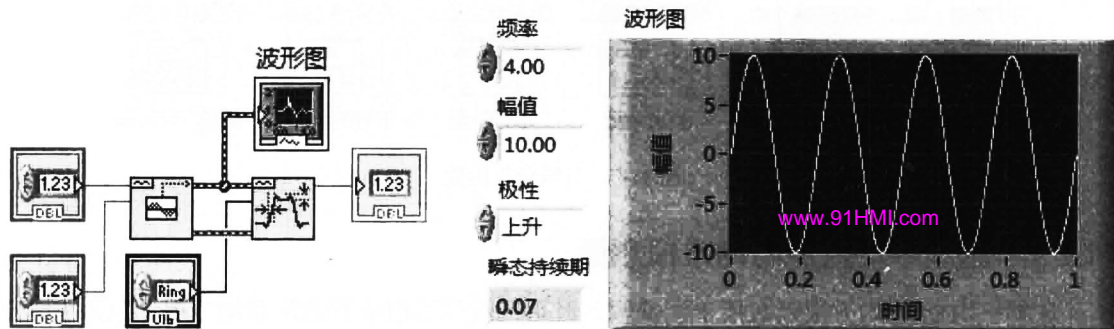


图 11-9 瞬态测量

11.2.3 提取单频信息

提取单频信息函数用于测量单频或指定频域内信号的幅值、频率和相位，其使用方法如下。

- ① 在前面板，添加一个波形图控件。
- ② 切换至后面板，选取“函数→信号处理→波形测量→提取单频信息”来添加该测量，再为其输出端口添加 3 个显示控件。
- ③ 添加一个正弦波形作为测量信号，并为其输入端口创建输入控件。经连线，则其程序框图和显示效果如图 11-10 所示。

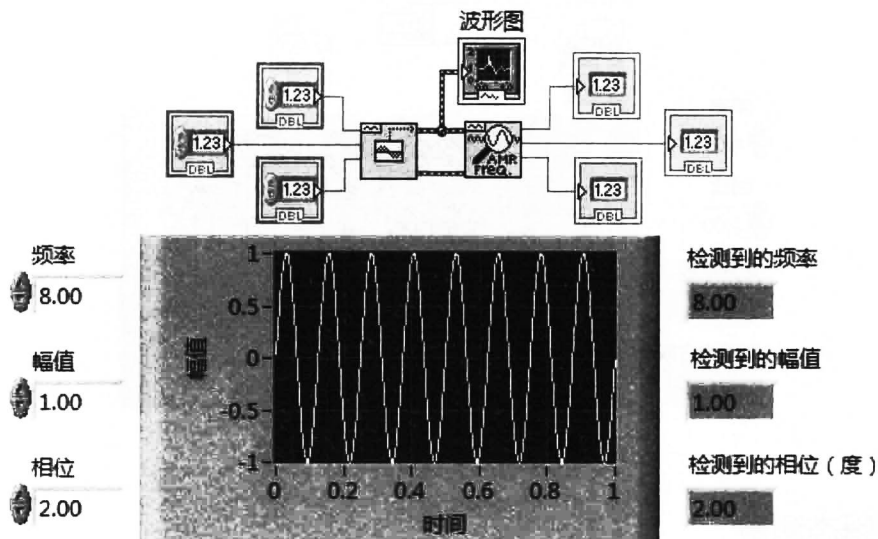


图 11-10 提取单频信息



11.2.4 幅值和电平测量

幅值和电平函数用于测量波形或波形数组的幅值、高状态电平和低状态电平。

幅值和电平函数的使用方法如下。

- ① 在前面板，添加一个波形图控件。
- ② 切换至后面板，选取“函数→信号处理→波形测量→幅值和电平”来添加该测量，再为其输出端口添加 3 个显示控件。
- ③ 添加一个正弦波形作为测量信号，并为幅值和频率端口创建输入控件。经连线，则其程序框图和显示效果如图 11-11 所示。

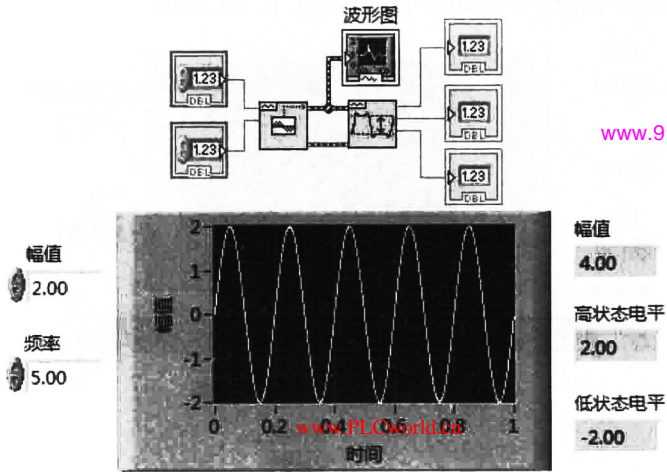


图 11-11 幅值和电平测量

11.2.5 脉冲测量

脉冲测量函数用于测量周期性波形或周期性波形数组的周期、脉冲持续期（脉冲宽度）、占空比（占空因数）及脉冲中心等。

脉冲测量函数的使用方法如下。

- ① 在前面板，添加一个波形图控件。
- ② 切换至后面板，选取“函数→信号处理→波形测量→脉冲测量”来添加该测量，再为其输出端口添加 3 个显示控件。
- ③ 为了提供测量的波形信号，选取“函数→信号处理→波形生成→正波波形”来添加该波形，再为频率和幅值端口创建输入控件。经连线，则其程序框图和显示效果如图 11-12 所示。

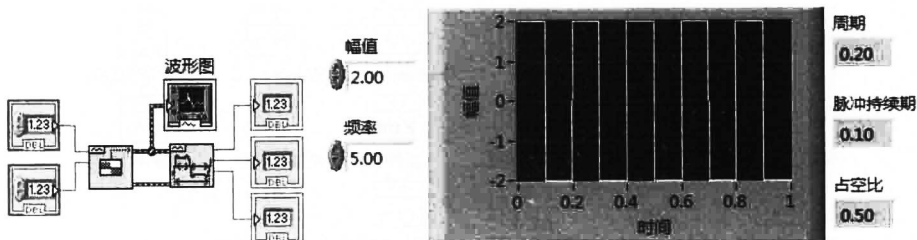


图 11-12 脉冲测量



11.2.6 谐波失真分析

谐波失真分析函数用于谐波分析，其分析内容包括测量基频、总谐波失真(THD)和所有谐波的电平幅值。

谐波失真分析函数的使用方法如下。

- ① 在前面板，添加一个波形图控件。
- ② 切换至后面板，选取“函数→信号处理→波形测量→谐波失真分析”来添加该分析，再为其输出端口添加 3 个显示控件。
- ③ 添加一个正弦波形作为测量信号，并为频率、幅值和相位端口创建输入控件。经连线，则其显示效果和程序框图分别如图 11-13 和图 11-14 所示。

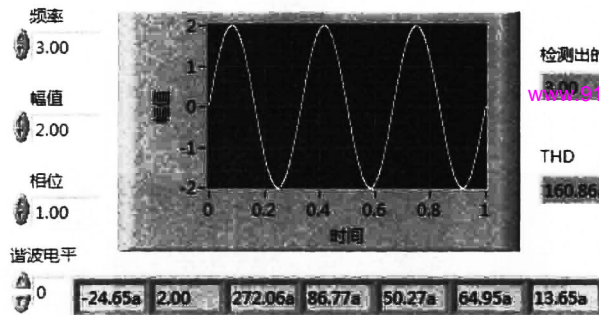


图 11-13 谐波失真分析界面

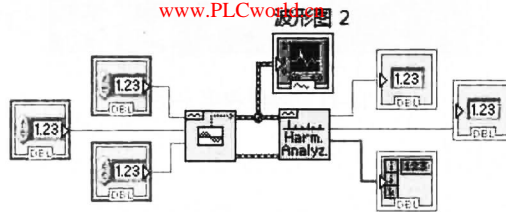


图 11-14 谐波失真分析程序框图

11.3 信号频域分析

频域分析是数字信号处理中最常用和最重要的方法，它包括变换功能和谱分析功能两部分。

变换功能主要是对数据进行相关的变换，如傅立叶变换、拉普拉斯变换和小波变换等，其具体的内容如图 11-15 所示。



图 11-15 变换函数



谱分析功能主要是对信号的频谱进行计算，通过频谱来观察和分析信号的频率成分。谱分析功能包括功率谱分析及联合时频分析等，具体内容如图 11-16 所示。

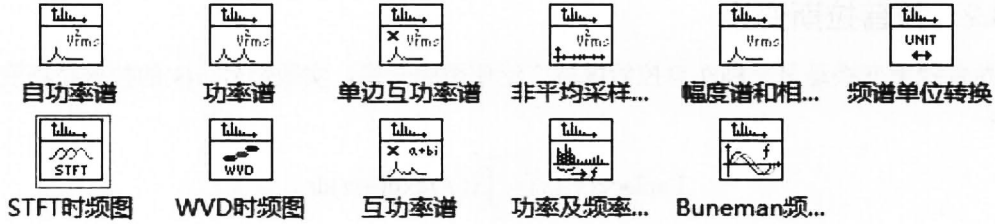


图 11-16 谱分析函数

11.3.1 傅立叶变换

傅立叶变换是一种常用的数学分析方法，它能对一定范围内的数据进行变换处理。而 FFT 变换则是一种快速傅立叶变换。

FFT 变换的使用方法如下。

- ① 在前面板，添加一个波形图控件用于显示变换前后的波形。
- ② 切换至后面板，选取“函数→信号处理→变换→FFT”来添加该变换。
- ③ 添加三个正弦波形作为信号源，为幅值和周期端口创建相应输入控件，并将采样数设为“200”。
- ④ 添加一个符合运算图标，将 3 个正弦波合并为一个信号源。
- ⑤ 选取“函数→数学→数值→复数→复数至极坐标转换”来添加该转换，以使变化后的效果更明显。
- ⑥ 依次添加除函数图标、数组大小图标和合并信号图标，经连线其程序框图和显示效果如图 11-17 所示。

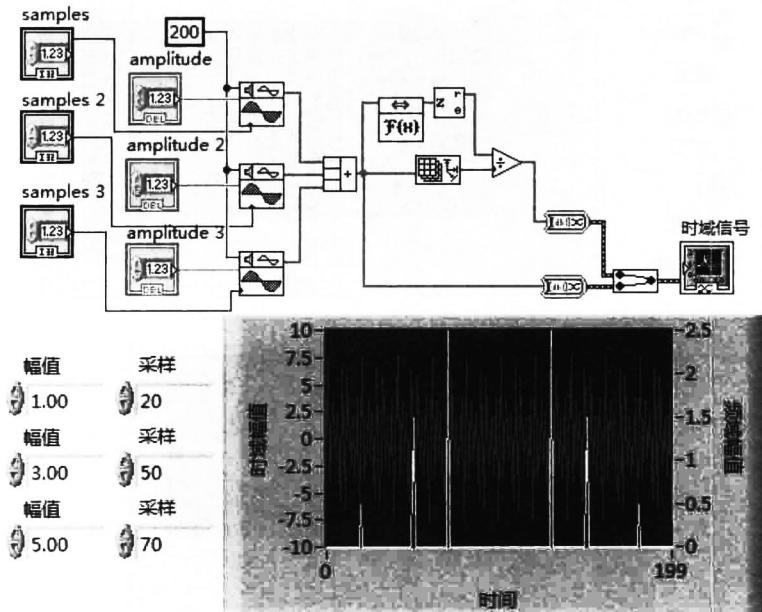


图 11-17 傅立叶变换



注：为使变换前后的波形能够最大化显示，需添加一个新标尺，并使之与变换后的曲线相对应。最后将标尺的名称分别改为“时域幅值”和“频域幅值”。

11.3.2 拉普拉斯变换

拉普拉斯变换是另一种在自控领域被广泛使用的变换，实数信号 $x(s)$ 的拉普拉斯变换定义为：

$$\text{Laplace}\{x\}(s) = \int_0^{\infty} x(f) \exp(-st) dt$$

$s \geq 0$ 且 s 为实数， $x(t)$ 定义适用于所有 $t \leq 0$ 的情况。

拉普拉斯变换的使用方法是：

- ① 在前面板，添加两个波形图控件用于显示变换前后的波形。
- ② 切换至后面板，选取“函数→信号处理→变换→实数拉普拉斯变换”来添加该变换，并为结束端口添加一个输入控件。
- ③ 添加一个基本函数发生器，并为频率、幅值和信号类型端口创建输入控件。
- ④ 添加一个乘函数图标和相应的数值常量，经连线其程序框图和显示效果如图 11-18 所示。

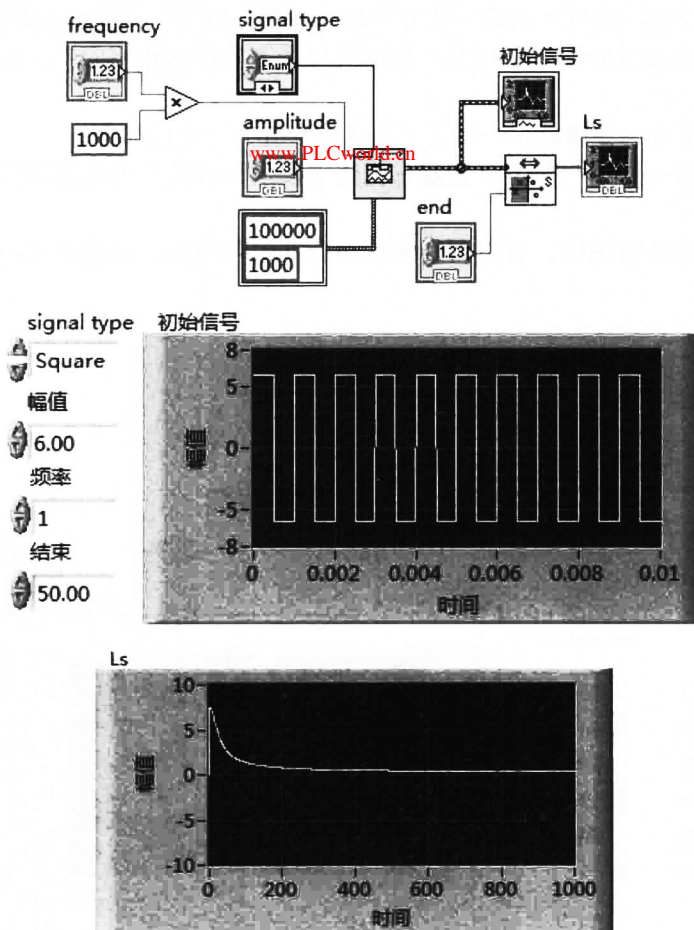


图 11-18 拉普拉斯变换



11.4 信号调理

信号调理是指对采集的数据进行数字滤波和加窗处理，以减少干扰信号的影响，进而达到提高信噪比的目的。

11.4.1 滤波器

滤波器包括巴特沃斯滤波器、等波纹滤波器及贝塞尔滤波器等，其具体内容如图 11-19 所示。



图 11-19 滤波器

中值滤波函数是一种依据阶数对输入序列 X 进行中值滤波的函数，当右秩大于零，则阶数为右秩，否则，如右秩小于零，则阶数取左秩。

中值滤波函数的使用方法如下。

- ① 在前面板，添加一个波形图控件用于显示滤波前后的波形。
- ② 切换至后面板，选取“函数→信号处理→滤波器→中值滤波器”来添加该滤波器，然后为左秩端口创建一个输入控件。
- ③ 添加一个脉冲信号图标作为信号源，并为幅值、延迟宽度端口创建对应的输入控件。
- ④ 添加一个高斯白噪声图标作为辅助信号源，并为采样、标准差和种子端口创建对应的输入控件。

添加一个加函数和一个创建数组图标，经连线其显示效果和程序框图分别如图 11-20 和图 11-21 所示。

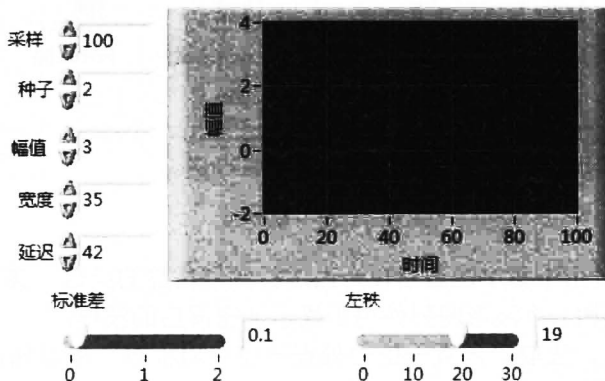


图 11-20 中值滤波界面

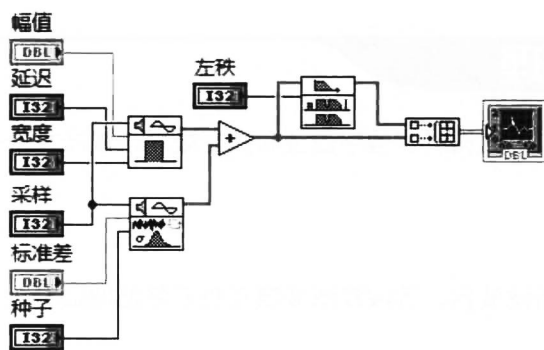


图 11-21 中值滤波程序框图

11.4.2 窗函数

窗函数具有截断信号、减少频谱泄露及分离频率相近的大幅值信号和小幅值信号等作用，在实际测量中，由于采样长度是有限的，因此在使用 FFT 分析信号频谱时，算法是将采样信号视为周期信号。即以第一个周期为采样信号，整个信号则用采样信号周期重现。

所谓“谱泄露”是指实际周期与周期信号间是不连续的，如同某一频率的能量泄漏到其他频率的现象。

解决谱泄露问题的方法之一就是无限延长采样周期，这样就使 FFT 能算出正确的频谱，然而这是不现实的，所以需要采用加窗的方法来实现，它是依据谱泄露的能量取决于周期延拓时的突变的幅度，跳跃越大，谱泄露就越大。

加窗就是将原始的采样波形乘以幅值变化平滑且边缘趋于零的有限长度的窗来减小每个周期边界处的突变。

LabVIEW 提供了多种窗函数，其内容如图 11-22 所示。

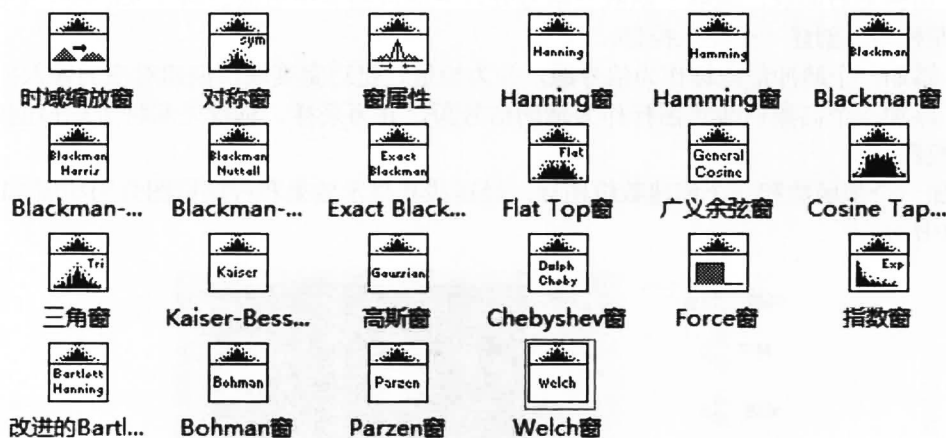


图 11-22 窗函数

对称窗函数是一种用于数字滤波器设计而进行频谱估算的函数，其使用方法如下。

- ① 在前面板，添加一个波形图控件用于显示加窗前后的波形。
- ② 切换至后面板，选取“函数→信号处理→窗→对称窗”来添加该滤波器，然后为窗口创建一个输入控件。



- ③ 添加一个正弦信号作为信号源，并为幅值、周期和相位端口创建对应的输入控件。
 - ④ 添加一个合并信号图标，经连线其程序框图和显示效果如图 11-23 所示。
- 注：图中白线为信号源波形，红线则为对称窗处理后的波形。

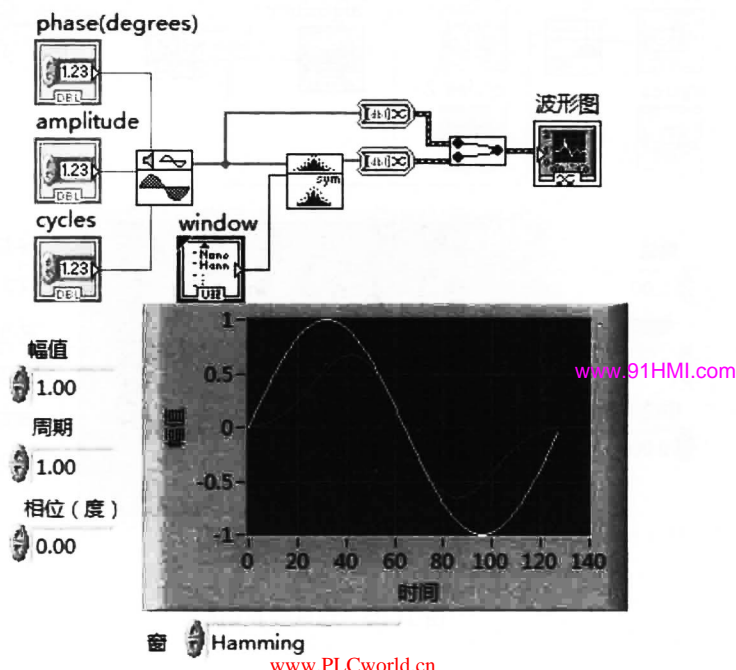


图 11-23 对称窗处理

11.4.3 波形调理

波形调理函数包括数字 FIR 滤波器、连续卷积 (FIR) 及波形对齐等函数，其具体内容如图 11-24 所示。

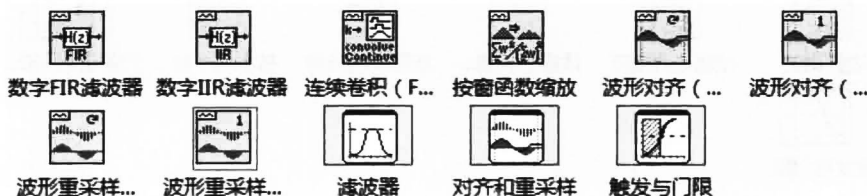


图 11-24 波形调理

连续卷积函数是一种能使单个或多个波形与单个或多个具有状态的 **kernel** 进行卷积，并使此后的调用以连续方式处理。

连续卷积函数的用法如下。

- ① 在前面板，添加一个波形图控件用于显示调理前后的波形。
- ② 切换至后面板，选取“函数→信号处理→波形调理→连续卷积 (FIR)”来添加该函数，然后为算法端口创建一个输入控件。
- ③ 添加两个正弦信号作为信号源，并为相应端口创建对应的输入控件。
- ④ 添加一个合并信号图标，经连线其程序框图和显示效果如图 11-25 所示。

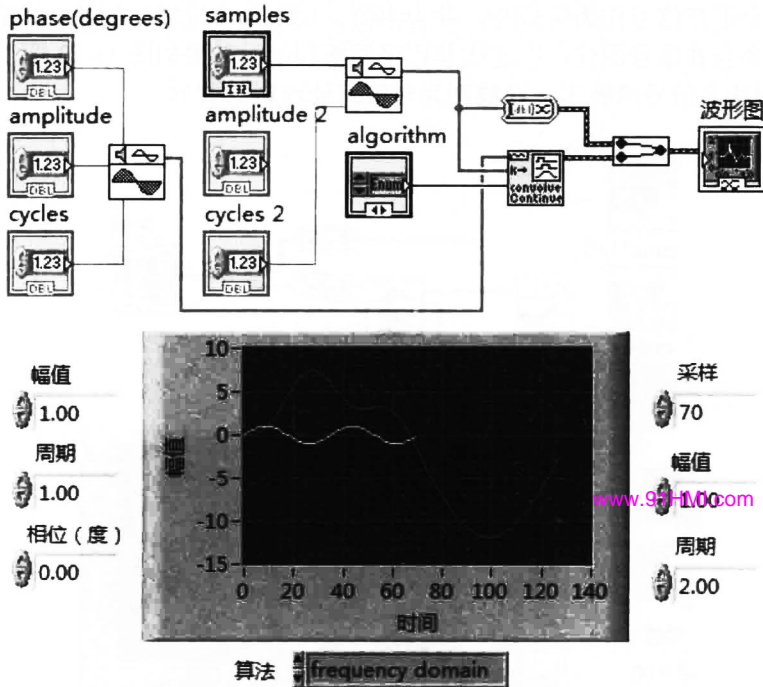


图 11-25 连续卷积调理

www.PLCworld.cn

11.5 波形监测

波形监测函数包括边界测试、波形波峰监测及触发与门限等函数，其具体内容如图 11-26 所示。



图 11-26 波形监测函数

11.5.1 边界检测

边界测试函数是一种对输入数据进行边界判别的函数，它可指示数据是否超出边界。该函数的用法如下。

- ① 在前面板，添加一个波形图控件用于显示测试波形与测试边界。
- ② 切换到后面板，选取“函数→信号处理→波形测量→波形监测→边界测试”来添加该函数。
- ③ 再添加两个创建边界规范图标及相应的配置簇。



- ④ 添加一个方波波形和一个均匀白噪声波形以模拟带干扰的方波信号。
 ⑤ 依次添加加函数和获取波形成分函数等，经连线其程序框图和显示效果如图 11-27 所示。

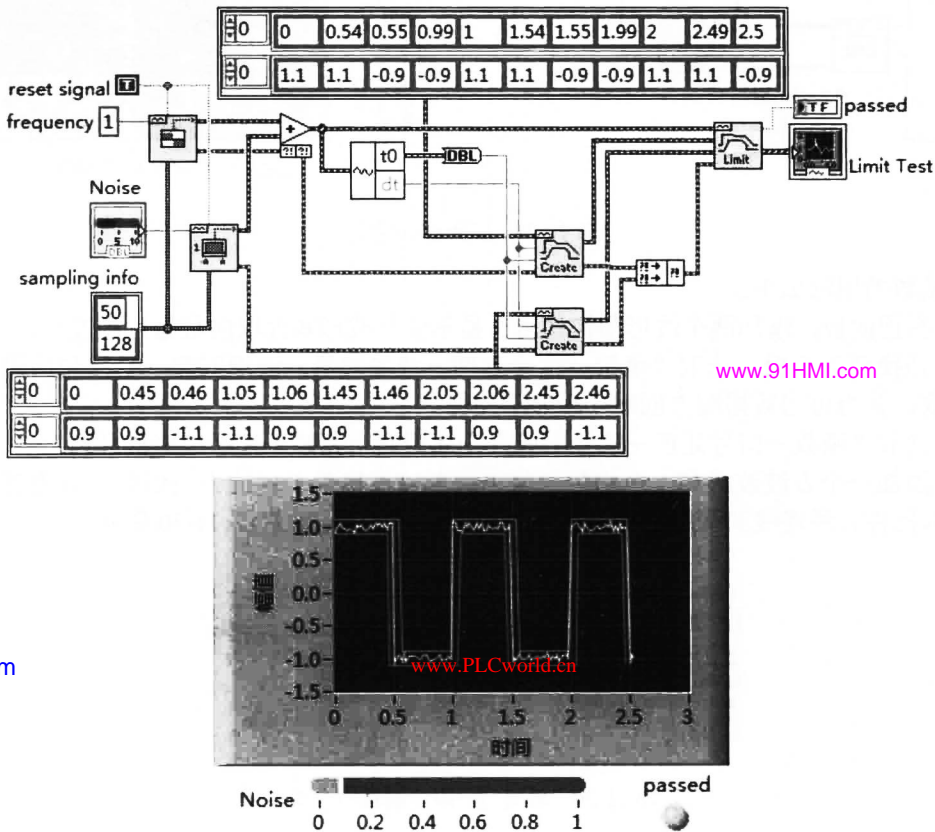


图 11-27 边界测试

运行该程序，当测试信号未超出边界范围时，指示灯点亮表明正常；否则，指示灯熄灭说明信号超界。

11.5.2 波峰检测

波形波峰检测函数的功能是在输入信号中查找峰谷或波谷的位置、振幅和二阶导数。

以检测波峰值为例，其操作方法如下。

- ① 在前面板，添加一个波形图控件用于显示检测波形。
- ② 切换至后面板，选取“函数→信号处理→波形测量→波形监测→波形波峰检测”来添加该函数，并为阈值端口创建一个输入控件。
- ③ 再添加一个公式波形图标，并为其创建相应的输入控件，经连线其程序框图和显示效果如图 11-28 所示。

11.5.3 触发与门限

触发与门限函数是一种通过触发来提取信号中可用数据片段的函数。其触发器状态可基于开启或停止触发器的阈值。

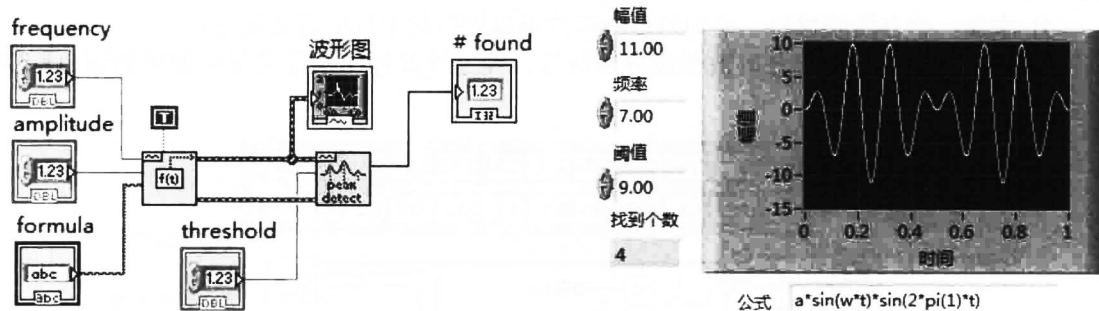


图 11-28 波形波峰检测

该函数的用法如下。

- ① 在前面板，添加两个波形图控件用于显示信号波形和提取的数据片段波形。
- ② 切换至后面板，选取“函数→信号处理→波形测量→波形监测→触发与门限”来添加该函数，并为可用数据端口创建一个显示控件。
- ③ 选取“函数→信号处理→信号生成→高斯调制正弦波”来添加该信号。
- ④ 添加一个方波波形和一个均匀白噪声波形以模拟带干扰的方波信号，并为其创建相应的输入控件。经连线其程序框图和显示效果分别如图 11-29 和图 11-30 所示。

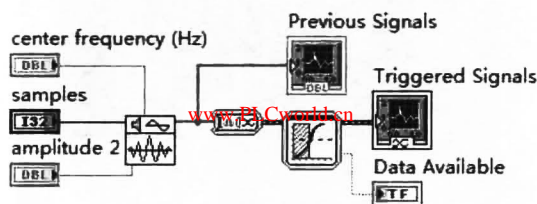


图 11-29 触发与门限检测程序框图

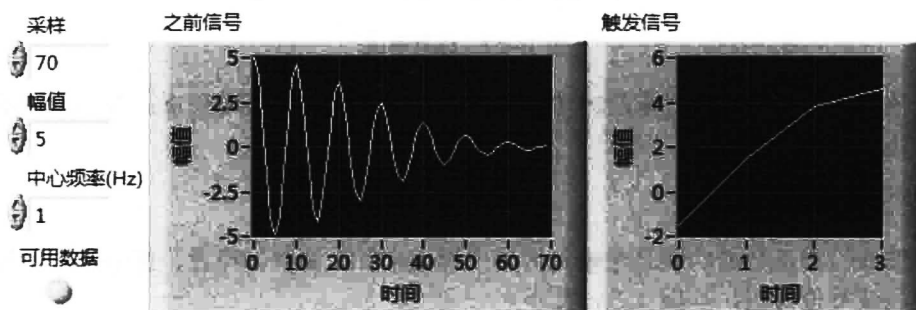


图 11-30 触发与门限检测显示效果

运行该程序，设起始方向为“下降”，起始电平为“4”，下降方向为“上升”，停止电平为“4”，则指示灯点亮说明有可用数据。

11.6 逐点分析

逐点分析是 LabVIEW 为解决实时分析处理所推出的一种处理方法，它可以在数据采集时边采集边进行数据点的分析处理，从而有效地减少了数据丢失的可能性，并使得对硬件设



备采样率的要求得以降低。逐点分析工具包括逐点信号生成、逐点信号时域处理、逐点信号频域变换及逐点信号滤波等，其具体内容如图 11-31 所示。

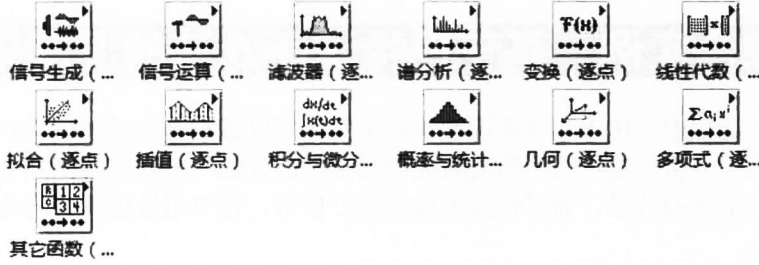


图 11-31 逐点分析

使用逐点分析功能进行实时滤波的操作如下。

- ① 在前面板，添加一个波形图表控件用于显示实时滤波处理的波形。
- ② 切换至后面板，选取“函数→信号处理→逐点→滤波器（逐点）→Butterworth 滤波器（逐点）”来添加该滤波器，将其类型设为“低通”。
- ③ 再添加一个 Butterworth 滤波器（逐点），将其类型设为“高通”。
- ④ 选取“函数→信号处理→逐点→信号生成（逐点）→正弦波（逐点）”来添加该信号，再添加一个均匀白噪声（逐点）信号。
- ⑤ 添加一个 For 循环图标，并将循环值设为“1000”。
- ⑥ 为循环设置一个 5 毫秒的延时环节，以便展示逐点分析的效果。
- ⑦ 依次添加倒数函数、乘函数、加函数、捆绑图标、数值常量及输入控件，经连线其程序框图和显示效果如图 11-32 所示。

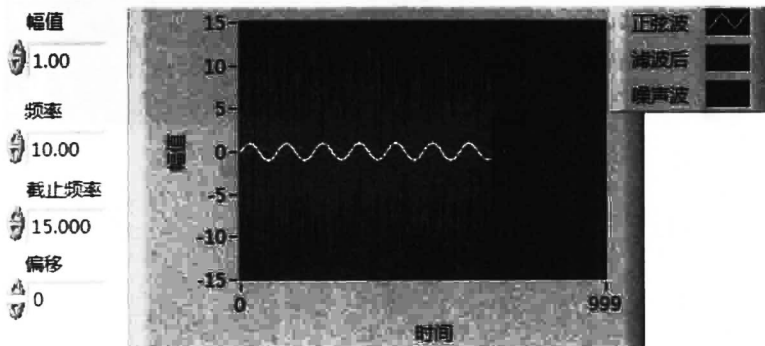
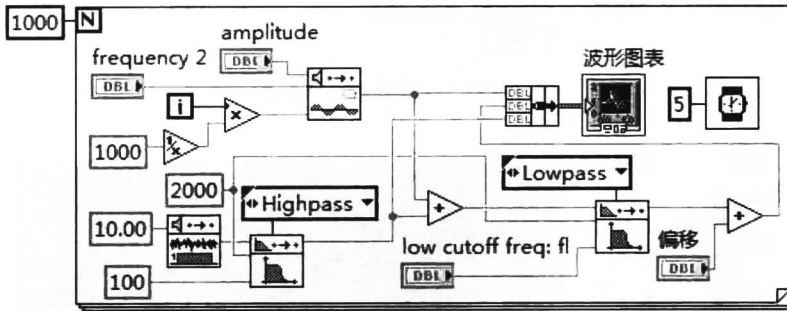


图 11-32 逐点分析



注：为了增强逐点显示效果，应将波形图表的刷新模式设为“示波器图表”。此外，通过观察可发现在滤波后，波形的幅值有所减小，相位也出现偏移。

11.7 综合实例：基于巴特沃斯滤波的温度采集

在温度采集过程中，由于环境等因素干扰采集的温度数据是呈波动变化状态的。要准确地对采集数据进行分析，进行适当的滤波处理是常用的方法。使用巴特沃斯滤波函数就可起到抑制高频信号干扰的效果，为了模拟采集的温度信号，这里用数值常量与均匀白噪声信号相叠加来获取。

以滤波方式采集温度数据的界面如图 11-34 所示，其具体操作如下。

- ① 新建一个 VI，并予以命名保存。
- ② 在前面板，添加一个波形图控件。
- ③ 切换至后面板，从中添加一个巴特沃斯滤波器，并为采样频率端口创建一个输入控件。
- ④ 为了模拟温度信号，添加一个均匀白噪声函数，并为幅值和采样端口创建对应的输入控件。
- ⑤ 添加一个信号合并图标以将滤波前后的波形共同显示在一个波形图中。经连线其程序框图和显示效果分别如图 11-33 和图 11-34 所示。

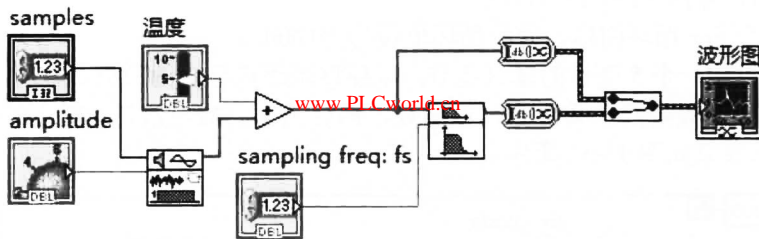


图 11-33 滤波处理程序框图

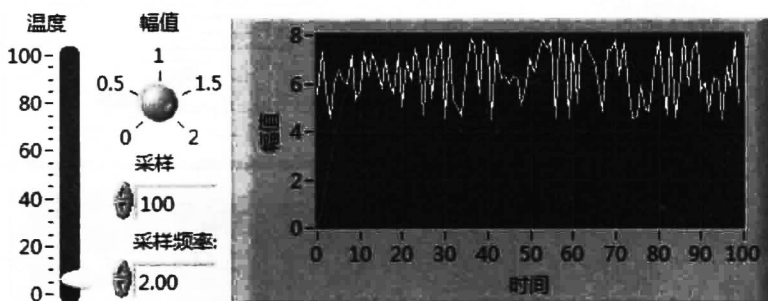


图 11-34 滤波采集温度显示效果



第 12 章

LabVIEW与其他软件通信

LabVIEW 为了与外部交换信息,提供了相应的外部互联接口,其中包括库与可执行程序、源代码控制、.NET 控制、ActiveX 技术及报表开发工具包等。

【本章导读】

- ◆ ActiveX 技术
- ◆ .NET 技术
- ◆ LabVIEW 与 Microsoft 的通信
- ◆ 动态链接库函数及调用
- ◆ 执行.exe 命令
- ◆ LabVIEW 与 MATLAB 混合编程等

在库与可执行程序中,其包括调用库函数节点和执行系统命令。利用库函数节点可实现对 Windows 平台外部共享动态链接库 DLL、Mac OS 平台上的 Frameworks 和 Linux 平台上的共享库的程序,它使 LabVIEW 具有了跨平台的强大能力。

12.1 ActiveX 技术

ActiveX 技术是微软推出的一种基于组件对象模型的技术集的统称,它可实现代码的重用并具有同开发语言的非相关性。

ActiveX 主要由 ActiveX 自动化、ActiveX 控件、ActiveX 文档及 ActiveX 脚本等组成,与 ActiveX 相关的内容有属性、方法、事件及容器等。

12.1.1 ActiveX 自动化

ActiveX 自动化是 ActiveX 中最重要的功能之一,它可以一个应用程序透过 ActiveX 来操作另一个应用程序的对象。此外,它也可以将一个应用程序中的对象及其属性和方法供另一个应用程序来调用。因此 ActiveX 的运行机制同网络所采用的客户端/服务器模式十分相近。

在 LabVIEW 中,ActiveX 既是客户端,也是服务端。当它作为客户端时,LabVIEW 可以访问已有的 ActiveX 对象来增强 LabVIEW 的功能,如访问 Word、Excel、Access 和 Web 等。当它作为服务端时,它允许其他程序访问它所提供的 ActiveX 自动化服务,如调用 VI、控制 LabVIEW 等。



ActiveX 函数位于“函数→互联接口→ActiveX”选板，其具体功能如表 12-1 所示。

表 12-1 ActiveX 函数功能表

函数名称	图标	使用说明
打开自动化		打开 ActiveX 对象并获取其引用句柄
关闭引用		关闭打开的 ActiveX 对象引用句柄
调用节点		用于调用 ActiveX 对象的方法或动作
属性节点		用于获取 ActiveX 对象的属性
转换为变体		把任意的 LabVIEW 数据转换为变体类型
变体至数据转换		用于把变体类型数据转换为 LabVIEW 数据
取消注册事件		取消与注册事件相关联的所有事件
事件回调注册		处理 ActiveX 对象提供的事件
静态 VI 引用		保持一个 VI 的静态引用

12.1.2 ActiveX 容器与事件

利用 LabVIEW 中的 ActiveX 容器，可以直接调用第三方开发的 ActiveX 控件，并能够访问其属性和方法，从而更加丰富程序的功能，并节省开发所耗费的时间。

容器和自动化的相同点是它们都可以在程序框图中以编程的方式来使用 ActiveX 控件的属性和方法，但如果要能显示所用的控件或文档，则必须使用 ActiveX 容器。使用 ActiveX 自动化是无法做到的。

添加 ActiveX 容器的方法如下。

① 在前面板，选取“控件→.NET 与 ActiveX→ActiveX 容器”。

② 要对容器添加控件内容，需右击新建的容器来激活快捷菜单。

③ 从快捷菜单中选取“插入 ActiveX 对象”，则会弹出图 12-1 所示的对话框。通过下拉列表可创建控件、创建文档及创建来自文件的对象。在选取“验证服务器”的情况下，会列出系统注册表内 LabVIEW 可以链接的全部 ActiveX 控件。否则，只会列出注册的控件。



图 12-1 选取 ActiveX 控件对话框

12.1.3 ActiveX 应用

ActiveX 控件多种多样，这里以 Windows Media Player 为例来加以介绍。



以 ActiveX 控件来调用 Windows Media Player 的操作方法如下。

- ① 新建一个 VI，并加以命名保存。
- ② 在前面板，选取“控件→.NET 与 ActiveX→ActiveX 容器”。
- ③ 右击容器控件，从控件菜单中选取“插入 ActiveX 对象”，再由弹出的对话框选取“Windows Media Player”，选取“确定”来完成选取。
- ④ 转换到后面板，选取“函数→互联接口→ActiveX→打开自动化”以进行添加。
- ⑤ 将容器图标与“打开自动化”图标连接起来，右击“自动化引用句柄”的输出端口，并由快捷菜单选取“创建→WMPLib.IWMPPlayer4 类的方法→openPlayer”。
- ⑥ 为“bstrURL”端口添加一个字符串常量，然后添加一个关闭引用图标。
- ⑦ 若想关闭 ActiveX 容器，可为添加一个关闭显示的属性节点。经连线，其界面和程序框图分别如图 12-2 和图 12-3 所示。

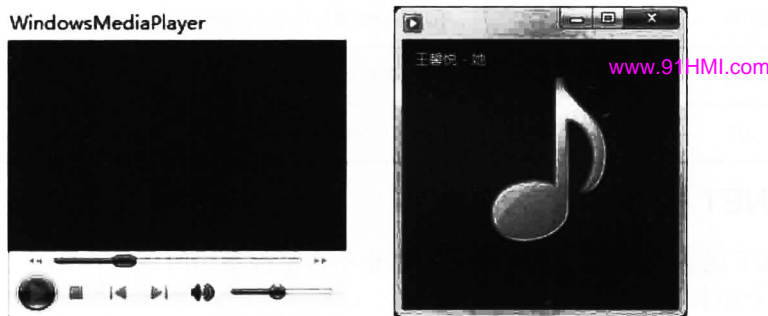


图 12-2 乐曲播放界面

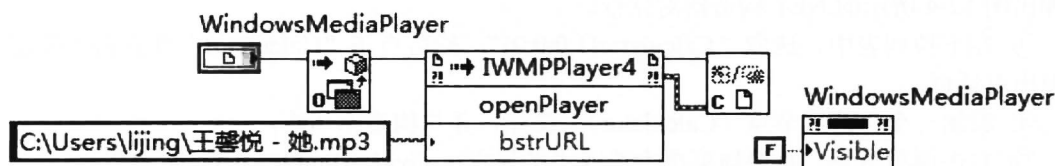


图 12-3 乐曲播放程序框图

12.2 .NET 技术

12.2.1 .NET 技术

在安装 .NET CLR 4.0 的情况下，LabVIEW 同样支持 .NET 技术。为此，它提供了创建和使用 .NET 对象的函数，具体内容如表 12-2 所示。

表 12-2 .NET 函数功能表

函数名称	图标	使用说明
构造器节点		创建 .NET 对象的实例
关闭引用		关闭打开的 .NET 对象引用句柄



续表

函数名称	图标	使用说明
调用节点 (.NET)		用于调用.NET 对象的方法或动作
属性节点 (.NET)		用于获取.NET 对象的属性
转换为通用的类		使引用强制转换为继承层次结构中更为通用的类
转换为特定的类		使引用强制转换为继承层次结构中更为具体的类
.NET 对象至变体		使.NET 对象转换为 LabVIEW 变体
至.NET 对象		使 LabVIEW 数据类型转换为.NET 对象
取消注册事件		取消与注册事件相关联的所有事件
事件回调注册		处理.NET 对象提供的事件 www.91HMI.com
静态 VI 引用		保持一个 VI 的静态引用

12.2.2 .NET 技术应用

通过.NET 函数的支持,使用.NET 技术也不再是很难的问题,下面以一个.NET 计算器的使用为例来介绍其操作方法。

① 新建一个 VI, 在前面板放置 2 个数值输入控件和 1 个数值显示控件。

② 切换至后面板, 选取“函数→互联接口→.NET→构造器节点”以进行添加。此时, 会弹出图 12-4 所示的.NET 构造器对话框。

③ 在下拉列表中, 选取“Calculator(1.0.0.0)”, 然后点击“Calculator”并点击“确定”来退出对话框。

④ 添加一个“调用节点 (Calculator)”图标, 并与构造器相连。

⑤ 右击调用节点, 从快捷菜单中选取“选择方法 divide(Double x,Double y)”。

⑥ 为调用节点的输入、输出端口添加相应的控件, 为使程序能够连续运行, 添加 while 循环并将所有图标移入循环内部。

⑦ 添加一个控制循环的停止按钮, 经连线其程序框图和界面如图 12-5 所示。

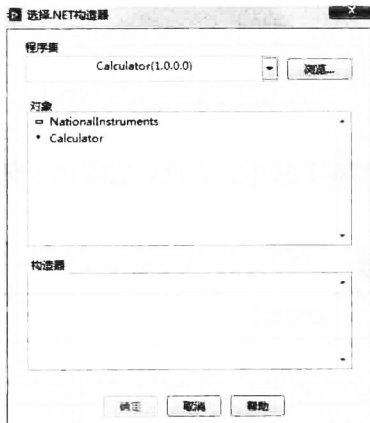


图 12-4 .NET 构造器对话框

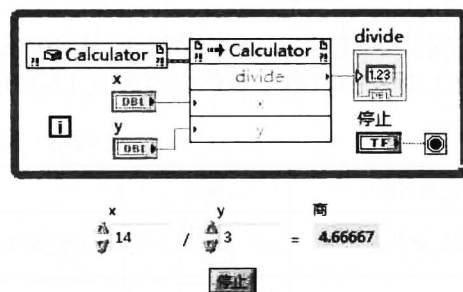


图 12-5 .NET 使用



12.3 LabVIEW 与 Microsoft 的通信

12.3.1 报表生成工具包简介

Labview Office 报表生成软件是一种突破 Office 报表开发壁垒的利器，它使普通的开发者也能制作出精美的 Office 报表。要获取 Labview Office 报表生成工具包可由 NI 的官方网站来下载。

文件是报表的载体，根据内容和特点的不同，所选择的载体形式也会有所差异，使用 Labview 生成的常见报表有两种，即 word 报表和 excel 报表。

在报表工具中，应 Office 报表生成工具所增添的内容如图 12-6 所示。



图 12-6 报表生成工具

提示

在报表的生成方法上，除了用 Office 报表生成工具包也可用其他插件来生成。

12.3.2 Word 报表

制作 Word 报表主要涉及创建新报表、添加报表信息、添加图片及添加表格等，其函数内容如图 12-7 所示。

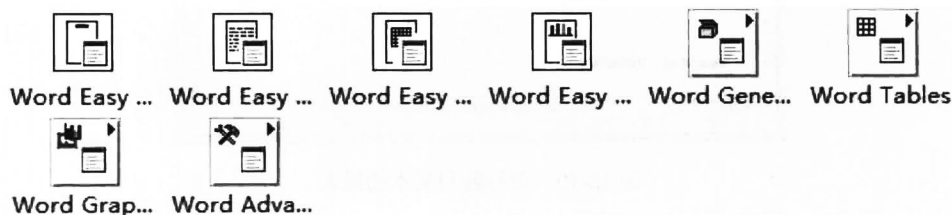


图 12-7 Word 报表生成工具



(1) 打开 Word 软件

能创建 Word 报表，必然可以操控 Word 软件。在 LabVIEW 中，用函数打开 Word 软件的操作步骤如下。

① 新建一个 VI，并切换到后面板。

② 选取“函数→编程→报表生成→新建报表”来添加该图标，并将报表类型设为“Word”，为了在程序运行时能够弹出 Word 窗口，将窗口状态选项设为“normal”，则程序框图如图 12-8 所示。

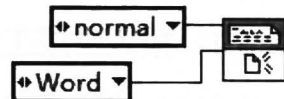


图 12-8 打开 Word 编辑界面

运行该程序，在正常情况下会弹出 Word 界面。

(2) 创建带标题及文本的 Word 报表

创建带标题及文本的 Word 报表，其操作步骤如下。

① 新建一个 VI，并切换到后面板。

② 添加一个“新建报表”图标，并将报表类型设为“Word”，为了在程序运行时不弹出 Word 窗口，将窗口状态选项设为“minimized”。

③ 选取“函数→编程→报表生成→ Word Specific→ Word Easy Title.vi”来添加该 VI，并为“title”端口创建一个字符串常量“流变分析仪测试报告”；为“font”端口创建一个常量，并将字体改为“华文新魏”，大小改为“36”。

④ 添加一个“Word Easy Text.vi”图标，为“alignment”端口创建一个常量“Centered”，为“text”端口创建一个字符串常量以显示测试时间。

⑤ 添加一个“保存报表至文件”图标，并为“报表文件路径”端口创建一个路径常量，将报表存为.doc 文档。

⑥ 添加一个“处置报表”图标，以关闭报表并释放所占用的资源。此时，程序框图如图 12-9 所示。

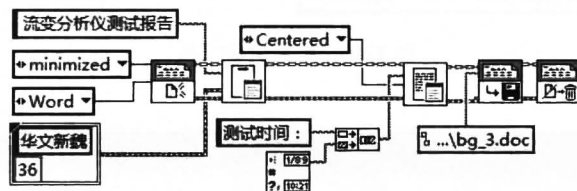


图 12-9 报表生成程序框图

运行该程序，则报表内容如图 12-10 所示。

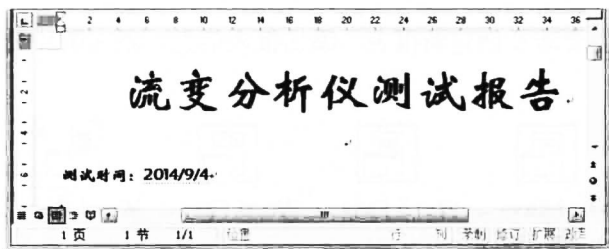


图 12-10 带标题和文本的报表

(3) 创建带图片的报表

在报表中，图片或波形图是不可缺少的元素，创建带图片报表的操作步骤如下：



- ① 新建一个 VI，并切换到后面板。
- ② 创建一个带标题的 Word 报表程序。
- ③ 在“保存报表至文件”图标前，插入一个“添加图像至报表”图标。为“图像的路径或 URL”端口创建一个路径常量以插入一幅图片，为“对齐”端口创建一个常量“MIDDLE”。经连线，其程序框图如图 12-11 所示。

运行该程序，则报表内容如图 12-12 所示。

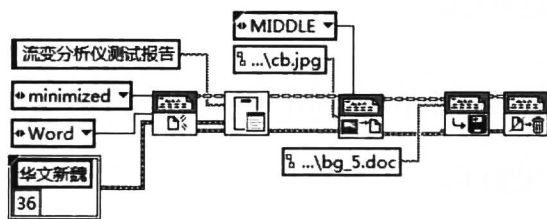


图 12-11 报表生成程序框图

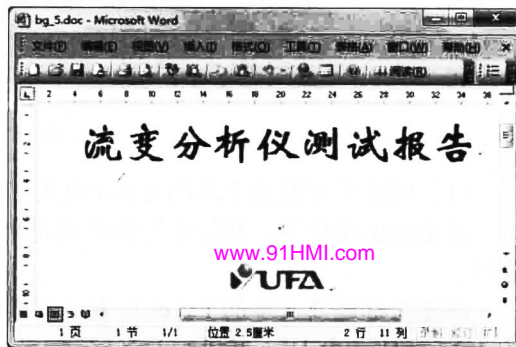


图 12-12 带标题和图片的报表

(4) 创建带表格的报表

表格同样是报表的一个重要组成部分，它可使数据更规范的排列输出。创建带表格的 Word 报表其操作步骤如下。

- ① 新建一个 VI，在前面板放置 3 个数组型输入控件并为其赋值。
- ② 创建一个带标题的 Word 报表程序。
- ③ 在“保存报表至文件”图标前，插入一个“Word Easy Table.vi”图标，并与 3 个数组相连。完成后的程序框图如图 12-13 所示。

运行该程序，则报表内容如图 12-14 所示。

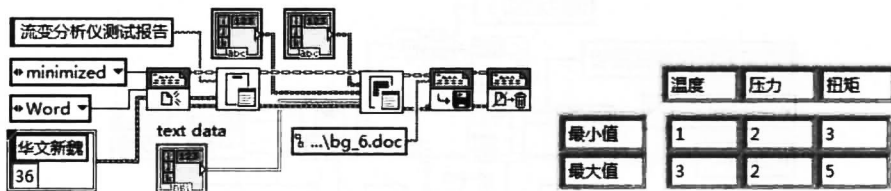


图 12-13 报表生成程序框图



图 12-14 带标题和表格的报表



12.3.3 Excel 报表

制作 Excel 报表同制作 Word 报表极为相似，其关键区别在于 Excel 属电子表格，相对 Word 来说它更易于进行定位处理。生成 Excel 报表的函数如图 12-15 所示。

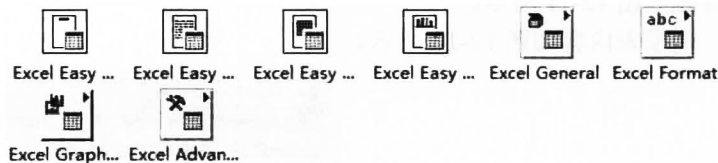


图 12-15 添加报表信息

(1) 创建带标题及文本的 Excel 报表

在 Excel 表格中，其最小工作单元是单元格，且每个单元格都有一个名字，并按序进行排列。

创建带标题及文本的 Excel 报表，其操作步骤如下。

① 新建一个 VI，并进行命名保存。

② 切换至后面板，添加一个“新建报表”图标，将报表类型设为“Excel”，为了在程序运行时不弹出 Excel 窗口，将窗口状态选项设为“minimized”。

③ 选取“函数→编程→报表生成→Excel Specific→Excel Easy Title.vi”来添加该 VI，并为“title”端口创建一个字符串常量“流变分析仪测试报告”；为“name”端口创建一个字符串常量“A1”来指定标题的位置。

④ 再添加一个“Excel Easy Title.vi”图标，并为“文本”端口创建一个文本常量以显示测试时间，其输出位置设为“A2”，字体大小为“12”。

⑤ 依次添加“报表保存至文件”和“处置报表”图标，并为“报表文件路径”端口创建一个路径常量，以保存报表。经连线，其程序框图如图 12-16 所示。

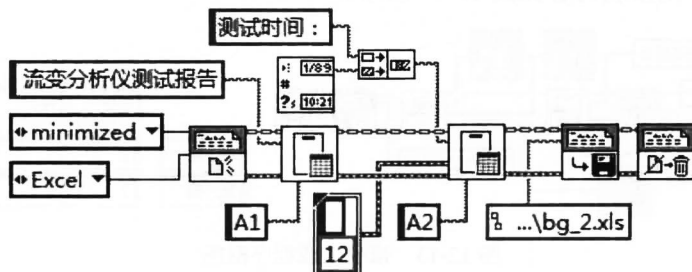


图 12-16 报表生成程序框图

运行该程序，则报表内容如图 12-17 所示。

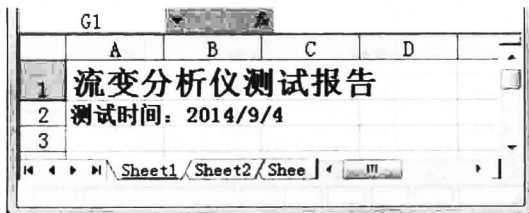


图 12-17 创建 Excel 报表



(2) 向单元格插入表格

在 Excel 的单元格中，同样也可以插入表格，其具体操作步骤如下。

① 新建一个带标题的 Excel 报表程序。

② 在“报表保存至文件”图标前，选取“函数→编程→报表生成→ Excel Specific→ Excel Easy Table.vi”来插入一个表格。将其输出位置设为“A2”，为“text data”端口添加一个只有表头的表格数据。经连线，其程序框图如图 12-18 所示。

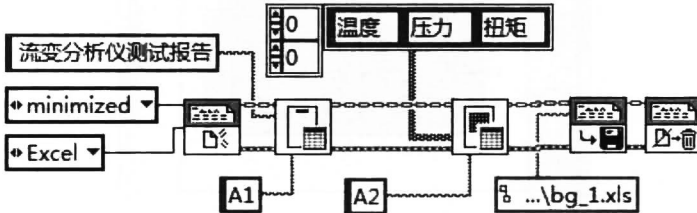


图 12-18 报表生成程序框图

www.91HMI.com

运行该程序，则报表内容如图 12-19 所示。

	A	B	C
1	流变分析仪测试报告		
2	温度	压力	扭矩
3			
4			

图 12-19 向单元格插入表格

(3) 向单元格插入图表

在产品测试方面，多采用 Excel 报表，以便于进行数据统计并绘制图表。制作带图表的报表其操作步骤如下。

① 新建一个带标题的 Excel 报表程序。

② 在“报表保存至文件”图标前，选取“函数→编程→报表生成→ Excel Specific→ Excel Graphs and pictures → Excel Insert Graph.vi”来插入一个表格。将其输出位置设为“A2”，为“text data”端口添加一个只有表头的表格数据。经连线，其程序框图如图 12-20 所示。

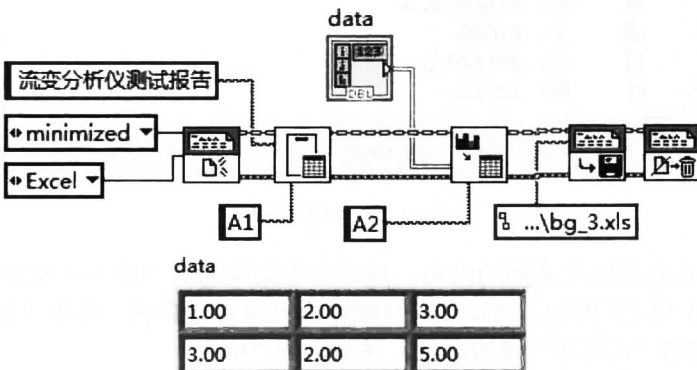


图 12-20 报表生成程序框图



运行该程序，则报表内容如图 12-21 所示。

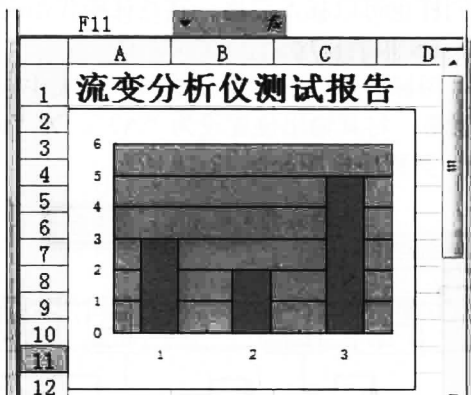


图 12-21 向单元格插入图表

www.91HMI.com

12.3.4 基于模板的报表生成

为了便于制作复杂的报表，LabVIEW 提供了一种基于模板的报表生成方式，在该模式下，即可生成 Word 报表，也可以生成 Excel 报表。

(1) 制作 Word 报表模板

制作 Word 报表模板的方法如下。

- ① 打开 Word 软件，然后按图 12-22 所示制作一个 Word 文档。

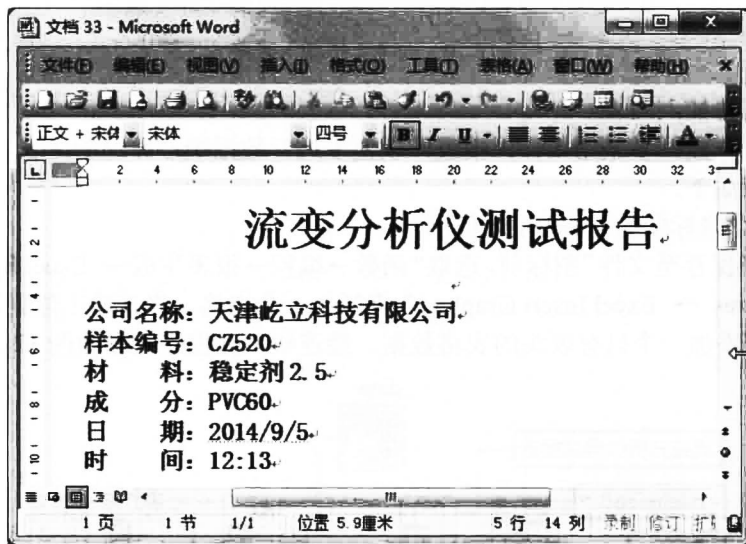


图 12-22 报表模板

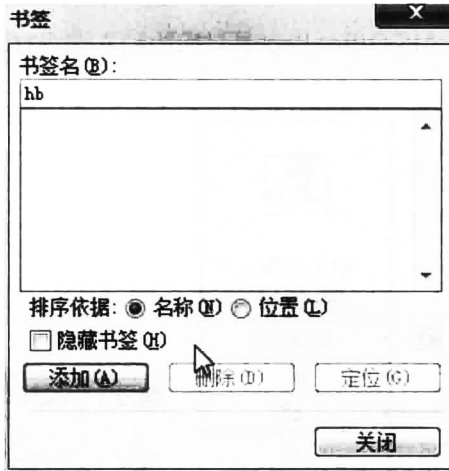
② 在文档中选取易发生变动的内容，然后由主菜单选取“插入→书签”，则会弹出书签设置对话框，如图 12-23 所示。通过该对话框可为书签进行命名。单击“添加”按钮后，可退出设置。此处可将“CZ520”设为书签，并命名为“bh”。

③ 为其他易产生变动的内容分别设置相应的书签。

④ 由主菜单选取“文件→保存”，在弹出的对话框中由“保存类型”下拉列表选取“文



档模板”，然后命名保存。至此，则完成报表模板的制作。



www.91HMI.com

图 12-23 书签设置对话框

(2) 生成 Word 报表

在制作好报表模板后，即可开始 Word 报表的生成，其操作方法如下。

- ① 新建一个 VI，在前面板添加 3 个字符串输入控件并对标签加以修改。
- ② 切换至后面板，选取“函数→编程→报表生成→MS Office Report”来添加该图标。

此时将弹出设置模板对话框，其内容如图 12-24 所示。

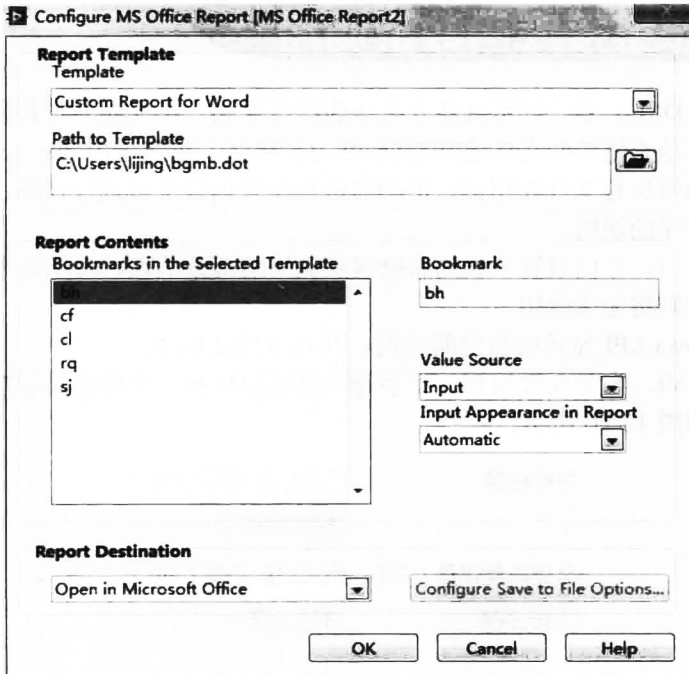


图 12-24 设置模板对话框

- ③ 在模板对话框中，由 Template 下拉列表选取“Custom Report for Word”，在 Path to



Template 文本框选取相应的报表模板文件, 此时会呈现出书签的名称。单击“OK”按钮, 退出设置。

④ 将字符串输入图标与 MS Office Report 对应的输入端进行连接, 则程序框图如图 12-25 所示。

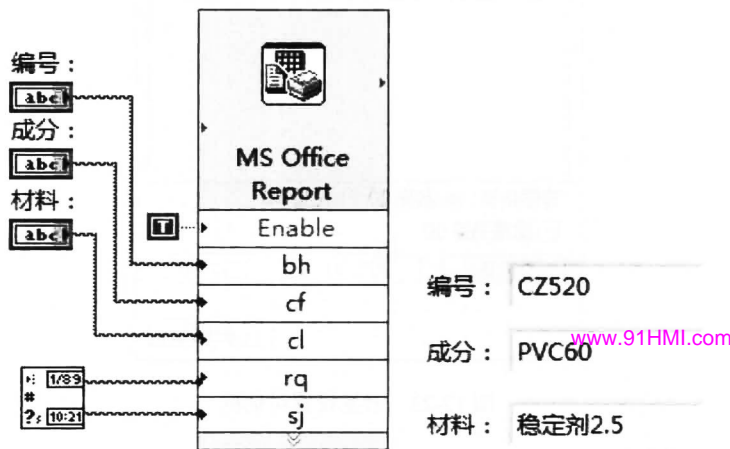


图 12-25 用模板生成报表

修改与标签相对应的内容并运行程序, 则按报表模板的格式生成新的报表。单击工具栏中的“保存”图标, 则生成新的报表文档。单击“关闭”按钮, 以退出 Word 软件。

www.91HMI.com

www.PLCworld.cn

12.4 动态链接库函数及调用

动态链接库 (DLL) 是一种可供多方共享的程序模块, 在其内部对共享的例程和资源进行了相应的封装。动态链接库文件通常是以 .dll 为扩展名, 也会采用 .drv、.sys 或 .fon。

动态链接库与可执行文件较相似, 但动态链接库只包含了可执行代码, 不能单独执行, 它必须借助应用程序的调用。

在 LabVIEW 中, 专门设置了供动态链接库调用的函数节点, 对于该节点的使用必须进行相应的配置, 否则将无法使用。

以调用 Windows API 显示磁盘空间为例, 其基本用法如下。

① 新建一个 VI, 在前面板放置一个字符串显示控件和三个数值显示控件并对标签进行相应设置, 最终如图 12-26 所示。



图 12-26 磁盘空间显示界面

② 切换至后面板, 选取“函数→互联接口→库与可执行程序→调用库函数节点”。

③ 双击添加的节点, 则弹出“调用库函数”对话框。



④ 在“函数”选项卡，并按图 12-27 加以设置。



图 12-27 函数设置

⑤ 切换到“参数”选项卡，然后按图 12-28 进行设置。最后单击“确定”按钮，返回后面板。

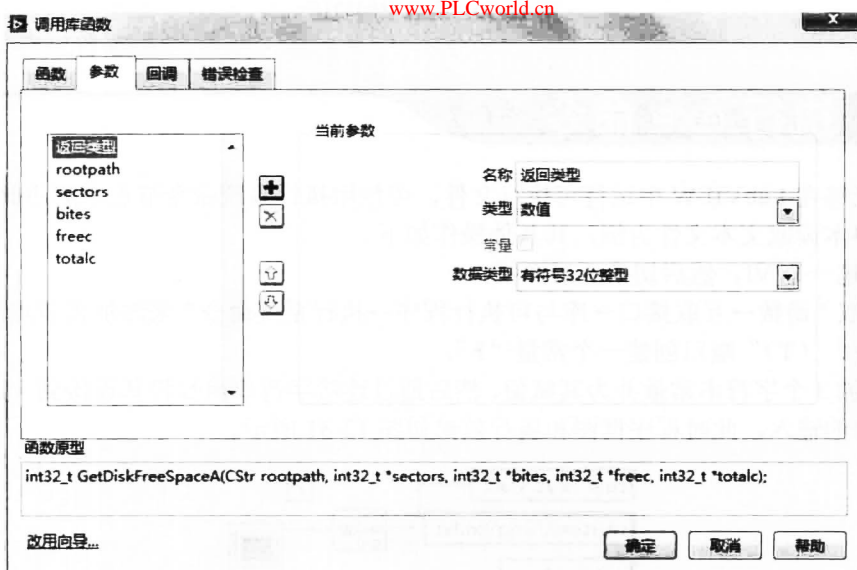


图 12-28 参数设置

⑥ 在后面板，依次添加复合、乘、除、减函数及常量“1024”，经连线，则程序框图如图 12-29 所示。

指定当前磁盘符并运行程序，则可获取磁盘空间信息。为进行验证，选取指定的磁盘并单击右键，由快捷菜单选取“属性”，则可见到图 12-30 所示的内容。经对比，证明显示结果完全正确。

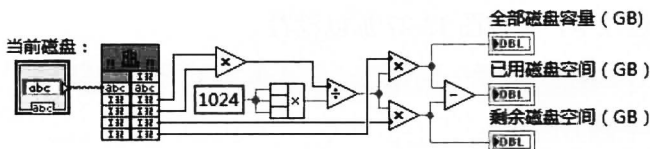


图 12-29 调用库函数程序框图

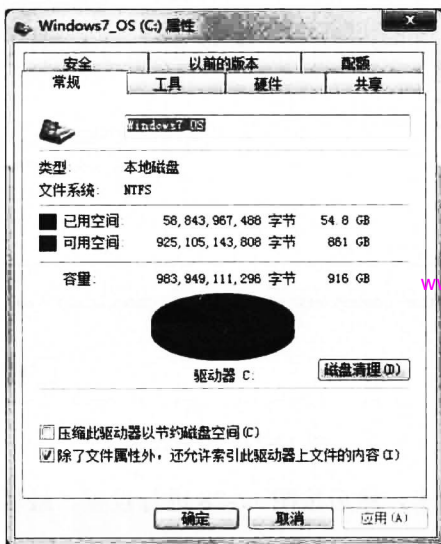


图 12-30 C 盘空间验证

12.5 执行.exe 命令

为了能够在 LabVIEW 中运行可执行文件，可使用执行系统命令节点来解决此类问题。以记事本读取文本文件为例，其具体操作如下。

- ① 新建一个 VI，然后切换至后面板。
- ② 选取“函数→互联接口→库与可执行程序→执行系统命令”来添加该节点，并为“等待直到结束？(T)”端口创建一个常量“F”。
- ③ 添加 3 个字符串常量并为其赋值，然后通过连接字符串函数将其连接到一起作为“命令行”端口的输入，此时程序框图和运行效果如图 12-31 所示。

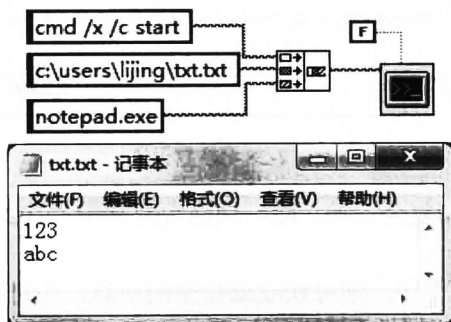


图 12-31 执行系统命令

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



12.6 LabVIEW 与 MATLAB 混合编程

MATLAB 是 MathWorks 公司推出的一种面向数学的文本化编程软件，其程序文件是符合 m 脚本语法的文件。为了使面向图形化数据流编程的 LabVIEW 同 MATLAB 结合起来发挥更大的优势，LabVIEW 常用 Math Script 来向 MATLAB 扩展，大大增强数学运算的灵活性。

由于 LabVIEW 是通过脚本节点来调用 MATLAB 语言编写的脚本，因此需要使用 MATLAB 脚本服务器来辅助执行，在安装了具有许可证的 MATLAB 后，才能使用 MATLAB 脚本节点。

此外，由于 LabVIEW 是使用 ActiveX 技术来执行 MATLAB 脚本节点，所以 MATLAB 脚本节点仅用于 Windows 平台。

Math Script 是通过 MATLAB 脚本节点来执行的，其操作方法如下。

- ① 新建一个 VI，并添加一个波形图表控件以显示生成的波形。
- ② 再添加一个旋钮控件，以设置波形的幅值。
- ③ 切换到后面板，选取“函数→数学→脚本与公式→脚本节点→MATLAB 脚本”来添加该节点，并在节点内部输入相应的脚本代码。
- ④ 在节点的左右两侧，依次添加输入和输出端口，并命名为“A”和“b”，经连线其程序框图和显示效果分别如图 12-32 和图 12-34 所示。图 12-33 是其等效程序框图。

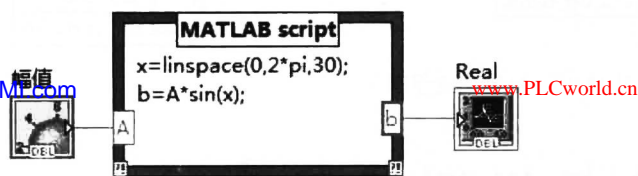


图 12-32 MATLAB 脚本使用程序框图

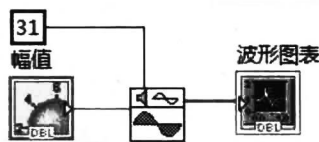


图 12-33 等效程序框图

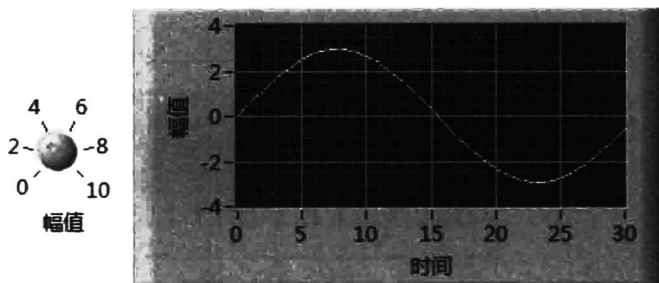


图 12-34 MATLAB 脚本运行效果

注：语句 $x=\text{linspace}(0,2\pi,30)$ 是从 0 到 2π 间均匀取 30 个值并赋给变量 x 。

语句 $b=A*\sin(x)$ 是用变量 x 的值来求正弦函数值，并与读取的幅值 A 相成形成变量 b ，最终绘制出一条正弦曲线。

12.7 综合实例：利用报表工具制作报表

在数据采集分析系统中，报表是一个综合呈现分析结果的方式。常规的报表多具有标题、



第 13 章

LabVIEW时间相关操作


时间在数据采集分析系统中是一个注意的参数，因此 LabVIEW 中配置了一定的与时间相关的函数，通过它们可以精确地了解当前时间、已用时间以及控制时间延迟。

【本章导读】

- ◆ 已用时间
- ◆ 时间延迟
- ◆ 当前时间等

13.1 已用时间

www.PLCworld.cn

已用时间函数的作用是从指定起始时间起，输出已经用去的时间。使用该函数的方法是：在后面板，选取“函数→编程→定时→已用时间”函数，其图标为 。在使用该函数时，会弹出图 13-1 所示的配置已用时间对话框，通过该对话框可设置已用时间值及选择是否启用超过目标时间后自动重置。

该函数所涉及的输入、输出参数含义如下：

输入参数：

设置起始时间(s)：使用当前时间，或相对于 1904 年 1 月 1 日星期五 12:00 a.m 以来无时区影响的时间偏移量，作为起始时间。

该值对于 0，则 Express VI 以当前时间作为起始时间。

该值大于 1，则 Express VI 可测量自 1904 年 1 月 1 日星期五 12:00:01a.m 以来无时区影响的秒数。

目标时间(s)：指定在结束布尔端的值变为 TRUE 之前必须经历的时间，默认值为 1。

重置：控制 VI 内部状态的初始化，默认值为 FALSE。

自动重置：Express VI 达到目标时间(s)时，重置起始时间为当前日期和时间(s)的值。

错误输入：描述该节点运行前发生的错误。

输出参数：

当前日期和时间(s)：显示自 1904 年 1 月 1 日星期五 12:00 a.m 以来无时区影响的秒数。

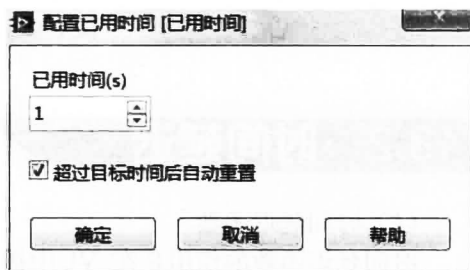


图 13-1 配置已用时间对话框



当前日期和时间文本：以文本格式显示当前日期和时间。

已用时间(s)：显示从指定的起始时间或当前日期和时间(s)起已经历过的时间，以秒为单位。

结束：表明已用时间(s)是否大于起始时间与目标时间(s)的和。

已用时间文本：显示从起始时间起已经历的时间，以秒为单位。

错误输出：输出提供标准错误输出。

起始时间文本：如超过目标时间后自动重置的值为 FALSE，则以文本格式显示 VI 初次运行的日期和时间，或连线至设置起始时间(s)输入端的时间。否则，显示最后一次重置的日期和时间。

获取起始时间(s)：如超过目标时间后自动重置的值为 FALSE，可显示 VI 初次运行的日期和时间，或连线至设置起始时间(s)输入端的时间。否则，显示最后一次重置的日期和时间。

图 13-2 是已用时间函数的使用界面。

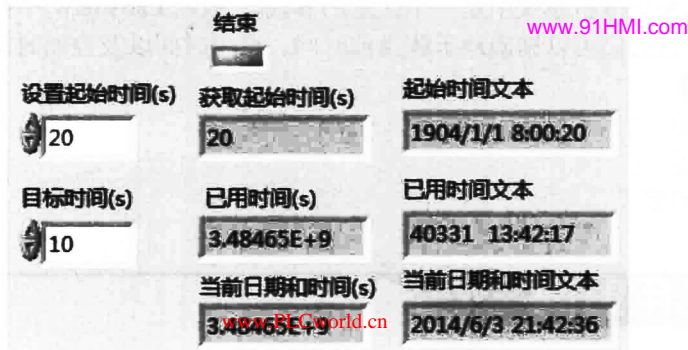



图 13-2 已用时间函数的使用界面

13.2 时间延迟

(1) 时间延迟函数

时间延迟函数的作用是在 VI 中插入时间延迟，使用该函数的方法是：在后面板，选取“函数→编程→定时→时间延迟”函数，其图标为 。

在使用该函数时，会弹出图 13-3 所示的配置已用时间对话框，通过该对话框可设置延迟时间值。

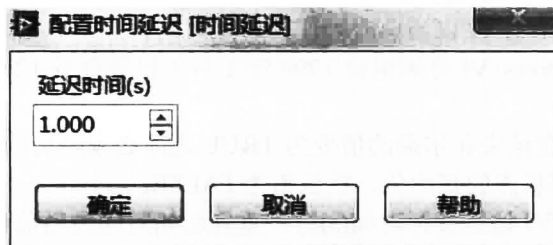


图 13-3 配置时间延迟对话框

图 13-4 是时间延迟函数的验证方法。

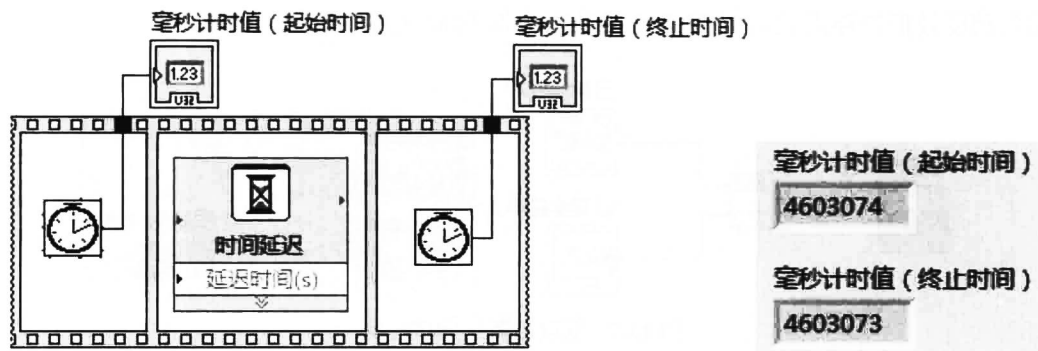


图 13-4 时间延迟函数的使用

(2) 等待 (ms)

等待 (ms) 函数的作用是等待指定毫秒数，并返回毫秒计时器的值。

图 13-5 是使用该函数演示 1 秒，并显示毫秒计时值。



图 13-5 等待 (ms) 函数的使用

(3) 等待下一个整数倍毫秒

等待下一个整数倍毫秒函数的作用是等待毫秒计时器的值为毫秒数的倍数，图 13-6 是该函数的使用方法。



图 13-6 等待下一个整数倍毫秒函数的使用

13.3 当前时间

在程序运行期间，当前时间是一个重要的参数，通过定时函数可方便地获取该参数，获取当前时间的函数有获取日期/时间字符串和获取日期/时间（秒）。

(1) 获取日期/时间字符串

该函数的功能是使时间标识的值或数值转换为计算机配置时区的日期和时间字符串。该函数是时间标识和数值解析为自 1904 年 1 月 1 日星期五 12:00 a.m（通用时间[01-01-1904 00:00:00]）以来无时区影响的秒数，并以字符串的形式输出日期和时间。图 13-7 是通过开关来选择是否需要显示秒的值。

(2) 获取日期/时间（秒）

该函数的功能是返回当前时间的时间标识。LabVIEW 通过自 1904 年 1 月 1 日星期五 12:00 a.m（通用时间）以来的秒数计算时间标识。通过转换为双精度浮点数函数，使时间标识的值



转换为精度较低的浮点数。图 13-8 是对获取日期/时间（秒）的使用。

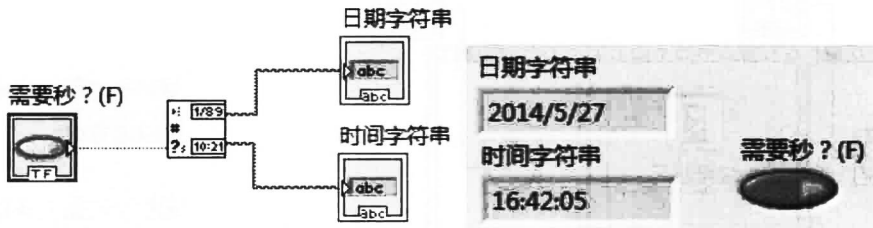


图 13-7 获取日期/时间字符串

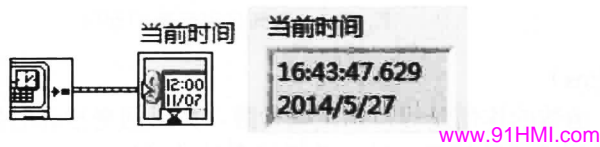


图 13-8 获取日期/时间（秒）

13.4 时间格式转换函数

(1) 转换为时间标识

该函数的功能是将数字转换为时间标识，其使用方法如图 13-9 所示。



图 13-9 数字转换为时间标识

(2) 日期/时间至秒转换

该函数的功能是将基于时间的簇转换为时间标识，其使用方法如图 13-10 所示。

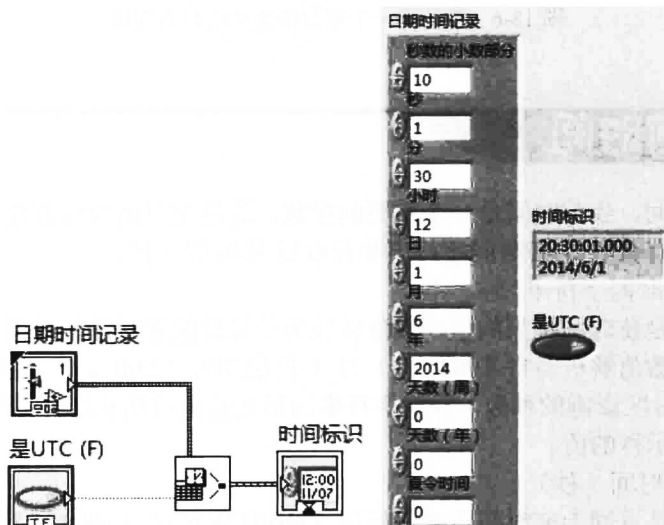


图 13-10 日期/时间至秒转换



(3) 秒至日期/时间转换

该函数的功能是将时间标识值或数值转换为时间簇，其使用方法如图 13-11 所示。

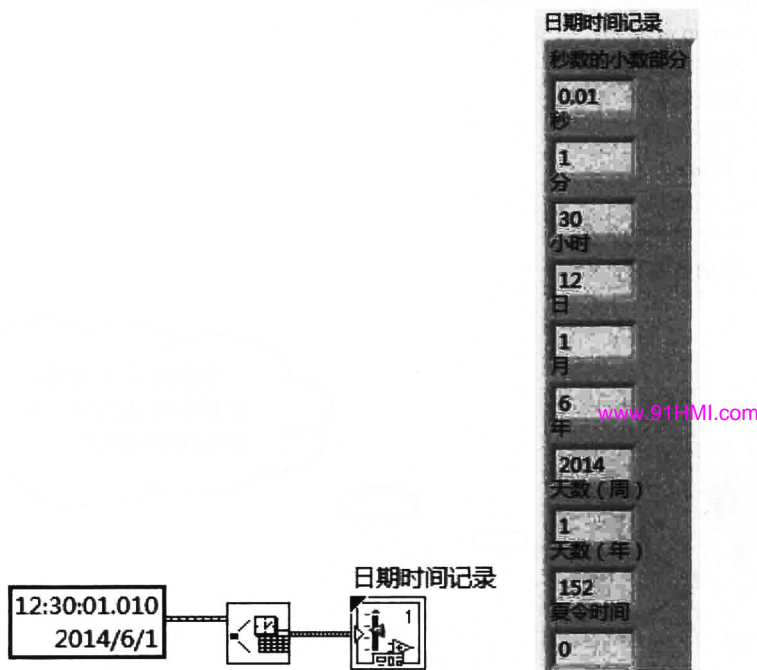


图 13-11 秒至日期/时间转换

(4) 时间标识常量

该函数的功能是用时间标识常量为程序框图传递时间和日期，其使用方法如图 13-12 所示。

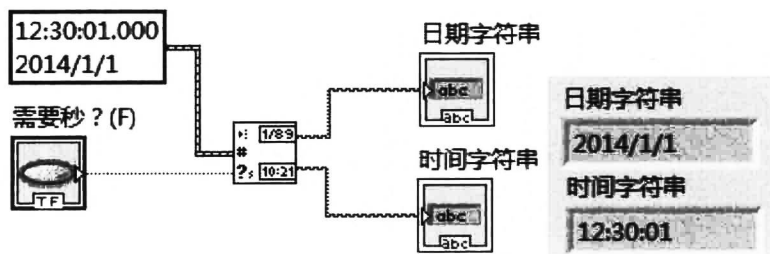


图 13-12 时间标识常量使用

(5) 格式化日期/时间字符串

该函数的功能是通过时间格式代码指定格式，并按照该格式使时间标识的值或数值显示为时间。

常用的时间格式代码如下：

- %c, 可显示依据地域语言设定的日期/时间;
- %X, 指定地域的时间;
- %H, 小时, 24 小时;
- %I, 小时, 12 小时;



%M, 分钟;
%S, 秒;
%<digit>u, <digit>精度, u 分数秒;
%p, a.m./p.m.标识;
%x, 指定地域日期;
%y, 两位年份;
%Y, 四位年份;
%m, 月份;
%b, 月名缩写;
%d, 一个月中的天值;
%a, 星期名缩写;



单单的一个时间，一样
有着丰富的内容，对它完
全不能等闲视之。



第 14 章

LabVIEW应用程序控制

应用程序控制主要涉及程序在运行期间如何动态控制控件的外观特征,如何获取 CPU 的信息参数及如何处理程序错误等。

【本章导读】

- ◆ 属性节点与调用节点
- ◆ 信息获取与提示
- ◆ 程序错误处理与控制等

通常的程序设计是用程序来解决某些问题。往往是在运行前通过处理数据,经运行处理再观察执行结果。在运行过程中,通常对界面的形式是不进行干预或控制的。要想在运行过程中进行程序控制,就必须掌握一些新的控制手段。

14.1 属性节点与调用节点

14.1.1 属性节点

尽管 LabVIEW 提供了丰富的输入、输出控件且风格多样,但在实际应用中面对动态修改控件外观的需求还是有些力不从心。为了增强这方面的能力,LabVIEW 引入了属性节点这一概念。在程序运行期间,通过动态修改控件对象的属性即可达到目的。

(1) 创建属性节点

创建属性节点要比创建公式节点简单,只需用鼠标右击相应的控件,然后从快捷菜单中选取“创建→属性节点”下的相应选项即可。

若想修改数据流的方向,只需从快捷菜单中选取相反的属性,例如某控件的当前属性是输入控件,要变更方向将其变为输出控件即可。

(2) 修改属性节点

一个控件往往有多个属性,如可见性、可用行、外观形态等。使用属性节点就可以轻易地控制它们的形态。

① 控制可见性。为了验证可见性的控制功能,在前面板依次选取液罐、温度计和开关按钮。分别为液罐和温度计添加“可见”属性节点,以控制其可见性,再将节点转换为写入状态。为了实现排斥性显示,向后面板添加一个“非”逻辑图标,经连线其程序框图和界面



如图 14-1 所示。

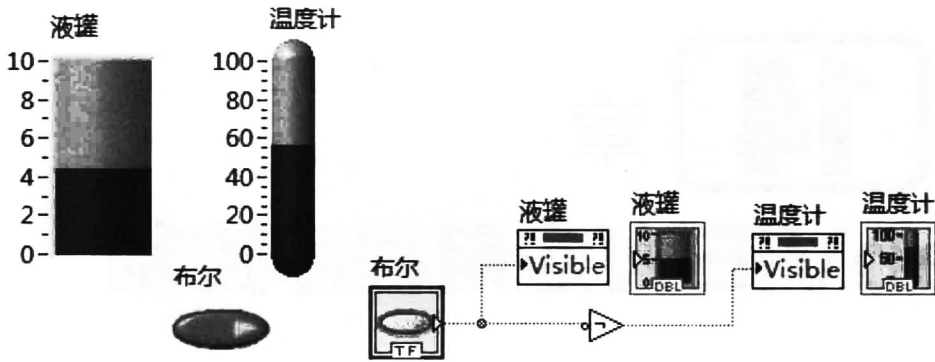


图 14-1 可见性控制

运行该程序可发现，在按钮的控制下界面中只会呈现一个控件。

② 控制可用性。在过程控制中，按键的操作往往是有条件性的不能随意的启动。以某流变分析仪的操作为例，它有预热和启动两个按钮，只有在预热得到一定温度的情况下，启动按钮才可使用，否则无效，该控制可概括成下面的控制来实现。

首先在前面板上添加两个开关按钮，并标签更名为“预热”和“启动”。为启动按钮创建一个“禁用”属性节点，由于该节点的控制参数为枚举类型，因此需要添加两个枚举常量和一个“选择”比较图标，经连线其程序框图和界面如图 14-2 所示。

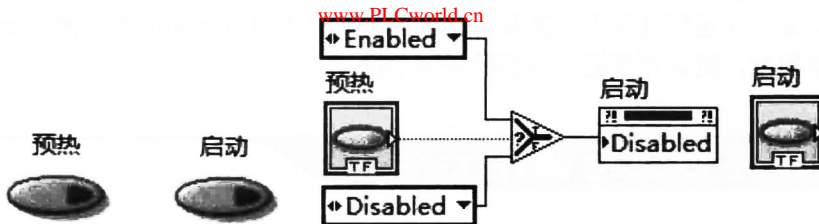


图 14-2 可用性控制

运行该程序可发现，只有在预热按钮工作的状态下启动按钮才可使用，否则是不受控的。

③ 控制闪烁性。在常规的控制中，当出现异常情况时，往往是借助声音、颜色或闪烁来提出警示。使用“闪烁”属性节点就可以实现这种制作效果。

在前面板依次放置滑动杆和温度计，然后为温度计创建一个“闪烁”属性节点，为了在超过指定温度（40 度）时产生闪烁，在后面板添加一个数值常量“40”和一个“对于”比较图标，经连线其程序框图和界面如图 14-3 所示。

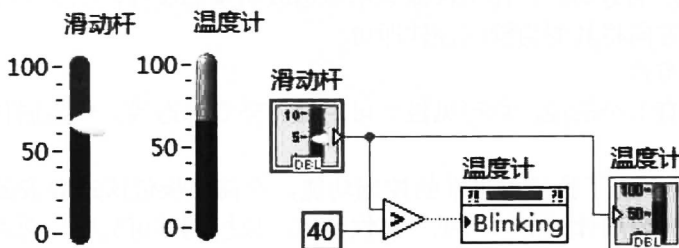


图 14-3 闪烁性控制



运行该程序可发现，当滑动杆所给定的温度小于 40 度时，温度计正常变化。一旦温度超过 40 度，温度计将会出现闪烁来给出警示。

14.1.2 方法节点

方法节点和属性节点较为相似，它的作用是执行特定的动作，有时也会配置相应的参数或反馈一定的数据。

在报表制作中，往往离不开波形图的使用，那如何来获取波形图？下面就通过方法节点的使用来获取波形图，其具体的实现方法是：

- ① 在前面板，放置一个波形图表控件以显示波形。
- ② 切换至后面板，添加相应的 For 循环、乘函数等构造一个幅值在 15 以内的 10 个随机数，并绘制出波形图。
- ③ 为获取该波形图，右键波形图表图标，由快捷菜单选取“创建→调用节点→导出图像”。
- ④ 为相应的端口创建输入控件，经连线其程序框图和显示效果如图 14.4 所示。

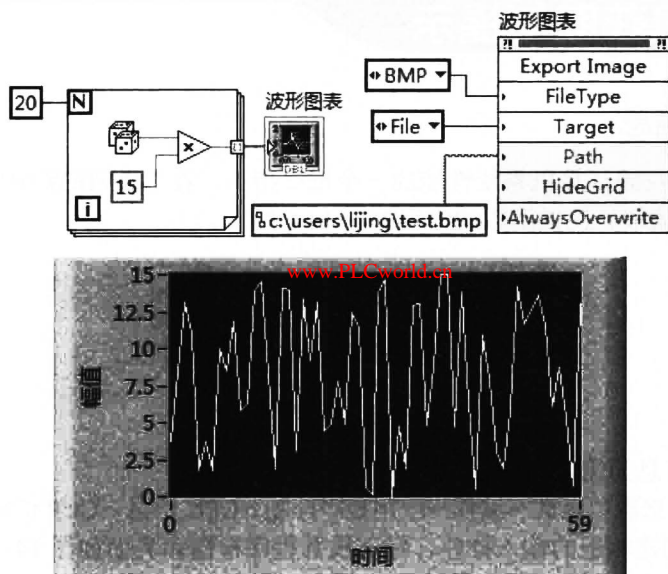


图 14-4 获取波形图文件

- ⑤ 运行该程序，则可在指定的路径处见到生成的波形图文件，如图 14-5 所示。

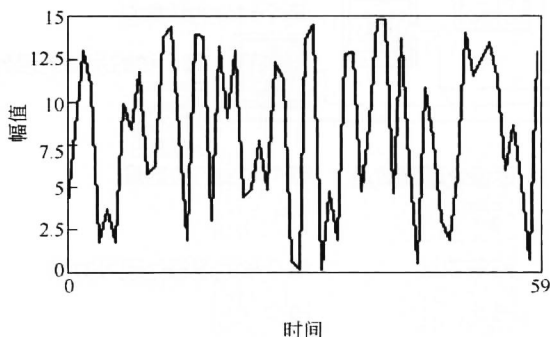


图 14-5 波形图文件内容



14.1.3 节点使用的方式

在不同的情况下，对方法节点和属性节点的使用会有不同的方式，其基本可分为3种。

① 使用的属性节点或方法节点与其控件同在一个程序中，此时可直接调用对于的属性节点或方法节点。该用法最简单也用的最多。

② 使用的属性节点或方法节点与其控件不在同一程序中，此时需要通过引用句柄来传递信息。这种将属性设置放在子VI的方式可为属性设置提供更大的应用空间。

③ 不能直接从控件上得到引用，而是先获得前面板所有的引用，进而再查找到引用句柄的引用，并设置属性或方法节点来设置行为。此时，会涉及到类的调用和转换，因此必须对它们之间的层次关系要十分清楚，否则就会产生失误。

在以上3种方法中，其共同点是对属性的操作都是通过节点来完成，不同处则在于所采用的引用句柄的方式有所差异。

www.91HMI.com

14.2 信息获取与提示

14.2.1 CPU 信息

CPU 的信息是反映计算机系统性能的一个重要指标，在 LabVIEW 中提供相应的函数，以供程序设计人员读取。此类函数的内容如图 14-6 所示。



图 14-6 CPU 信息函数

以读取 CPU 信息为例，其操作方法如下。

- ① 在后面板，选取“函数→编程→应用程序控制→CPU 信息→CUP 信息”来添加该函数。
- ② 为各端口创建相应的输入控件，经连线其程序框图和界面如图 14-7 所示。

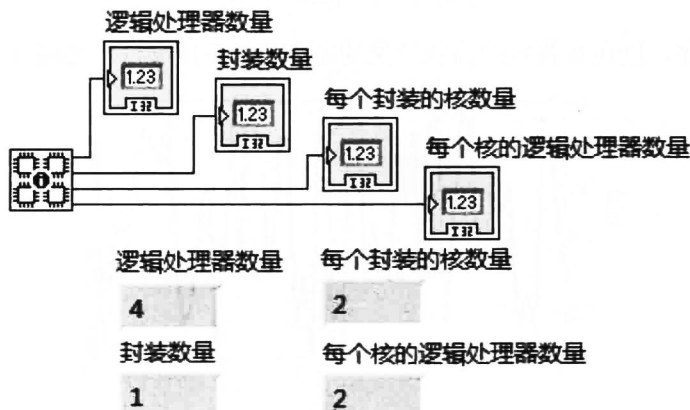


图 14-7 获取 CPU 信息



14.2.2 获取拖放数据

拖放功能是一种便捷的操作方式，在 LabVIEW 中为该功能提供了相应的支持。通过拖放，不仅可以传递数据，也能传递波形等。

拖放是一种处于拖曳对象和放置对象间的操作，该功能的编程模式是固定的，并建立在事件结构的基础上。拖放过程要经过 3 个步骤，从源对象拖曳信息、找到目标对象及将相应数据写入目标对象。

以列表选项的拖曳传递为例，其具体的操作步骤如下。

- ① 在前面板，放置 2 个列表框并将标签分别改为“候选信号源：”和“输出信号源：”。
- ② 切换至后面板，添加一个 while 循环图标并为条件端口创建一个输入控件。
- ③ 为进行拖放处理，在循环内放置一个事件结构。
- ④ 右击“选择器标签”，从弹出的快捷菜单中选取“添加事件分支”，则打开“编辑事件”对话框，如图 14-8 所示。

www.91HMI.com

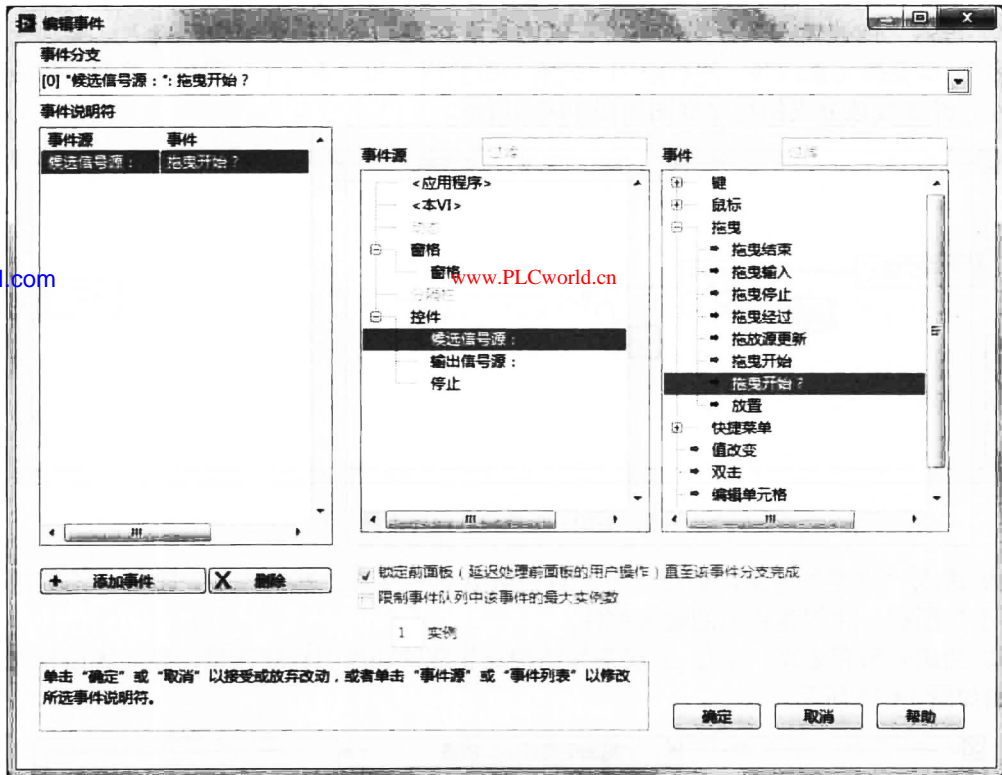


图 14-8 编辑事件分支

⑤ 在“事件源”栏选取控件中的“候选信号源：”，在“事件”栏选取拖曳中的“拖拽开始？”，然后点击“确定”退出。

⑥ 再添加 2 个分支，以进行拖拽输入和放置处理。

⑦ 将“候选信号源：”图标放入“拖拽开始？”分支，然后右击该图标，由快捷菜单选取“创建→属性节点→项名”。

⑧ 依次在分支内添加索引数组、转换为变体、捆绑及创建数组图标，再创建相应的输



入控件，经连线该分支的程序框图如图 14-9 所示。

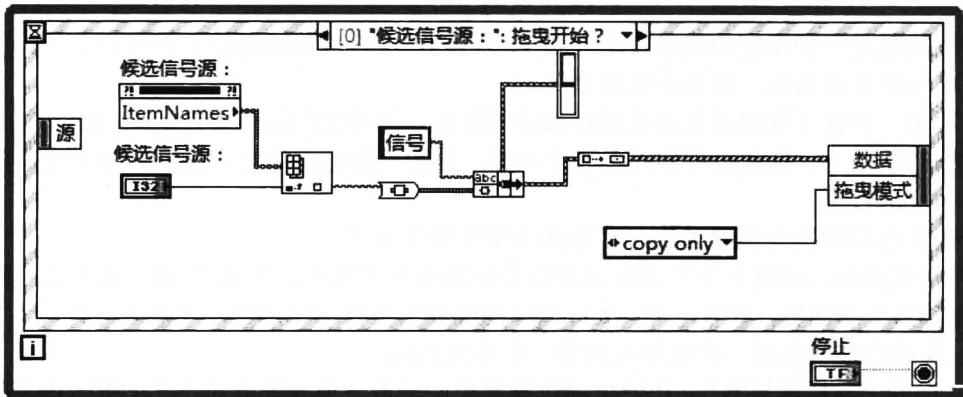


图 14-9 拖拽开始程序框图

www.91HMI.com

⑨ 进入“拖拽输入”分支，右击左侧的元素标签，从快捷菜单中选取“选择项→可用数据名”，以更换元素内容。然后添加“搜索一维数组”和“不等于”图标，并创建相应的输入控件。经连线该分支的程序框图如图 14-10 所示。

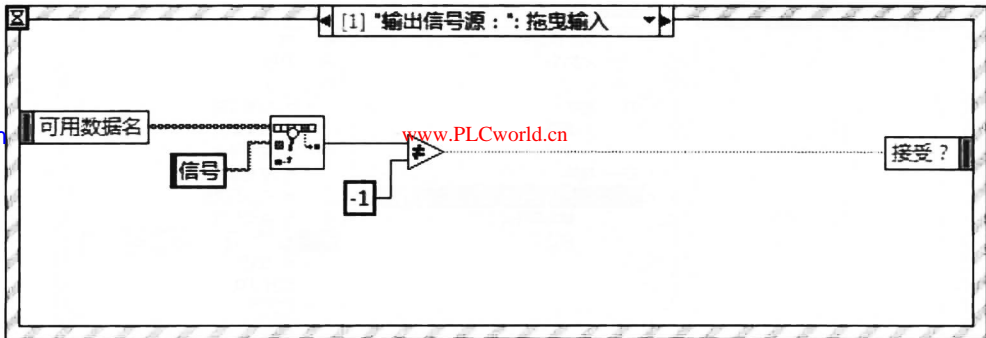


图 14-10 拖拽输入程序框图

⑩ 进入“放置”分支，调出一个“可用数据名”元素。然后添加“搜索一维数组”和“不等于”图标，并创建相应的输入控件。

⑪ 为进行放置处理，添加 3 层条件结构及相应的功能图标和常量，经连线该分支的程序框图如图 14-11 所示。

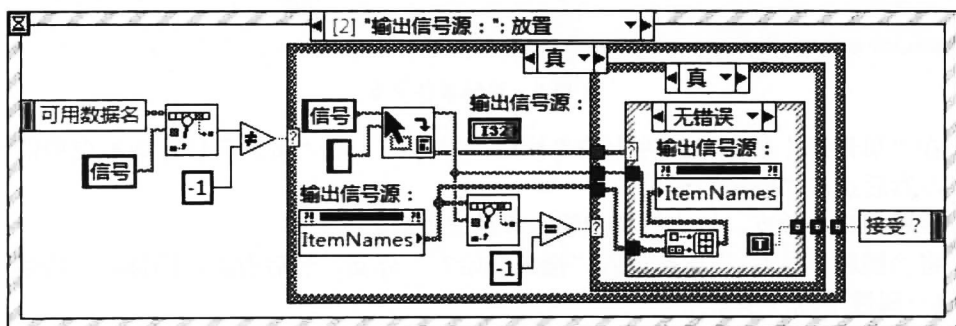


图 14-11 放置处理程序框图



运行该程序，从源列表框选取某选项，然后按住鼠标左键进行拖曳，当到达目标列表框后再松开鼠标左键，则该选项将被放入目标列表框。效果如图 14-12 所示。单击停止按钮，则可退出运行。

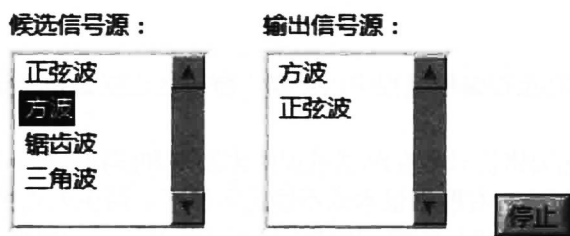


图 14-12 拖放操作界面

14.2.3 对话框

www.91HMI.com

对话框是一种便捷的人机对话方式，LabVIEW 为了便于程序开发，提供了 3 种对话框方式函数，分别为单按钮对话框、双按钮对话框及三按钮对话框。

以双按钮对话框的使用为例，其使用的方法如下。

① 在前面板，放置一个字符串显示控件，并将标签改为“运行方式”。

② 切换至后面板，选取“函数→编程→对话框与用户界面→双按钮对话框”来添加该对话框。为“信息”端口创建一个字符串常量“未检测到变频器是否继续运行？”，为“T 按钮？”端口创建一个显示控件，并将其标签改为“变频器”。

③ 添加一个“条件结构”图标，依次在真、假条件分支放置字符串常量“未检测到变频器，只能手动运行”和“未检测到变频器，退出测试”。经连线，则程序框图和界面分别如图 14-13 和图 14-14 所示。

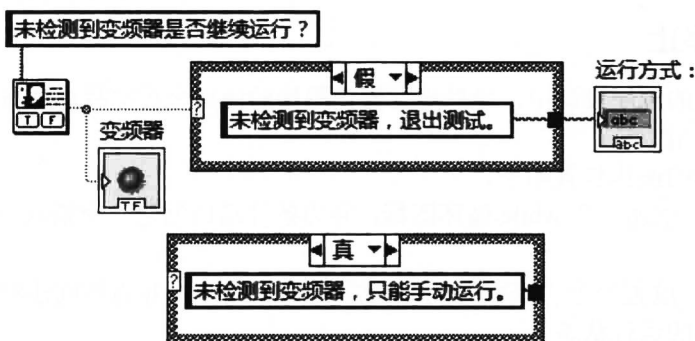


图 14-13 对话框控制程序框图



图 14-14 对话框控制界面





14.3 程序错误处理与控制

14.3.1 错误处理

再好的编程高手也难免在编程过程中出差错，解决之道就是发现错误和解决错误。常规的错误可分为3类。

① 逻辑性错误，此类错误只能在测试过程中来发现和解决。以 while 循环的控制为例，图 14-15 的左图为正确用法，右图则根本就不能进入循环。两图差别很小，较难看出。

② 语法和语句类错误，包括关键词的拼写和使用问题、语法规定的符号使用问题等。对于此类错误，LabVIEW 提供了错误列表工具，通过它可以检查和发现错误。如果 VI 的运行按钮呈断裂状态，则说明在当前程序框图中有错误，是不能正常运行的。此时，单击“显示错误”按钮可细查错误的原因。经修改当运行按钮呈箭头状时，方可运行程序。

③ I/O 错误，此类错误的定义较宽泛，会出现在各类软硬件间的交互过程中，如出现在数据通信、数据采集、仪器 I/O、文件 I/O 等过程。对于此类错误，需借助 LabVIEW 所提供的简易错误处理器和通用错误处理器。作为简易错误处理器，它是通用错误处理器经封装所构成的。选取“函数→编程→对话框与用户界面”子模板可添加相应的错误处理器。

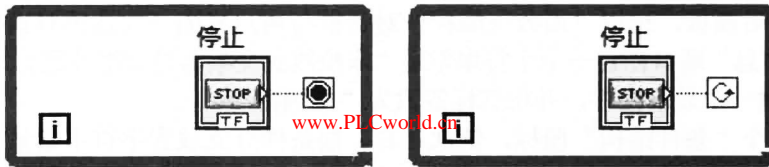


图 14-15 循环控制

14.3.2 程序停止

在 LabVIEW 的程序设计中，可以通过停止图标的使用来控制程序执行。下面就通过一个示例来说明其使用方法。

使用停止图标的操作步骤如下。

① 在后面板，添加一个 while 循环图标，并为条件端口创建一个输入控件，以控制循环的结束。

② 在循环内，放置一个“获取日期/时间字符串”图标，并为其创建相应的输入、输出控件，以表明程序的运行状态。

③ 选取“函数→编程→应用程序控制→停止”来添加该图标，并为其创建一个输入控件，为此该程序就有了 2 个退出循环而停止运行的途径。经连线其程序框图和界面如图 14-16 所示。

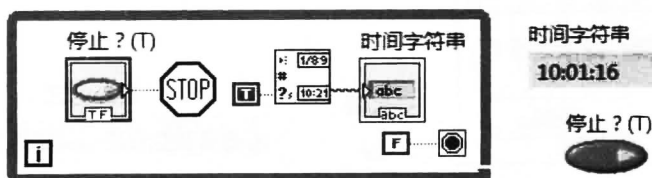


图 14-16 程序停止



14.3.3 程序退出

使用停止图标只能终止程序的运行，但不能退出相应的执行环境，要彻底退出 LabVIEW 执行环境，就必须使用退出 LabVIEW 图标。

使用退出 LabVIEW 图标的方法是，选取“函数→编程→应用程序控制→退出 LabVIEW”来添加该图标，再为其创建一个输入控件，经连线其程序框图和界面如图 14-17 所示。



图 14-17 程序退出

运行该程序，当按下退出按钮时，程序将彻底退出 LabVIEW。

www.91HMI.com

14.4 综合实例：文件的任意位置存储

在数据采集过程中，数据不仅能存入数组或文件，而且还能够存入剪贴板供粘贴使用。下面以一个实例来加以说明，其设计的思路是由随机函数生成波形数据，并显示波形图。然后使用方法节点将波形数据放入剪贴板。

实现波形数据放入剪贴板的具体操作如下。

- ① 新建一个 VI，并予以命名保存。
- ② 在前面板，添加一个波形图控件。
- ③ 切换至后面板，以 For 循环和随机函数构造一个双通道的数据信号，并通过合并信号图标连到波形图图标。
- ④ 右击波形图图标，从快捷菜单中选取“创建→调用节点→导出曲线数据至剪贴板”，并为“曲线”端口创建一个输入控件。此时，程序框图和运行效果如图 14-18 所示。

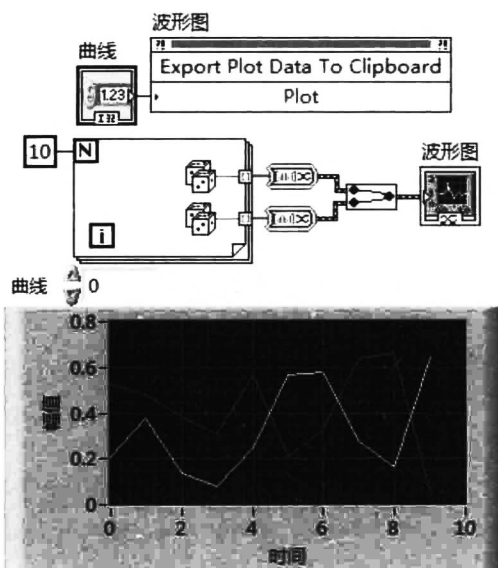
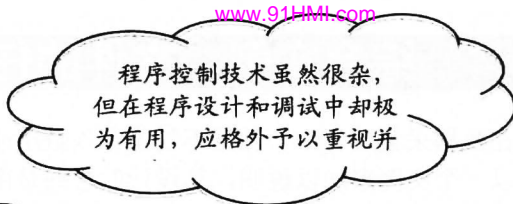


图 14-18 波形数据放入剪贴板



⑤ 若进行粘贴，则可见到如下的剪贴板内容：

时间 - 曲线 0	幅值 - 曲线 0
0	0.207134
1	0.378657
2	0.135376
3	0.0797384
4	0.244649
5	0.565847
6	0.579245
7	0.274056
8	0.162278
9	0.644946



程序控制技术虽然很杂，
但在程序设计和调试中却极
为有用，应格外予以重视并

www.PLCworld.cn



第 **15** 章

LabVIEW网络编程技术

在网络通信过程中，必须共同遵守一些规则，这些规则在计算机相互通信时便被称为网络通信协议。在 LabVIEW 中，它所支持的通信协议包括 TCP/IP 协议、串行通信协议、无线网络协议及邮件传输协议。利用这些协议的节点，就可方便地编写出网络通信方面的程序。

【本章导读】

- ◆ TCP/IP 网络通信
- ◆ DataSocket 通信
- ◆ 远程面板

在网络通信协议中，TCP/IP 协议是互联网界使用最基本的协议。在其内部是分层进行管理的，它们是链路层、网络层、传输层及应用层。TCP 和 UDP 都属于 TCP/IP 的传输层，因此在 TCP/IP 协议下又细分为 TCP 协议和 UDP 协议。此外，DataSocket 技术也是面向测控领域的网络通信技术。有它们的支持，网络编程已不仅是编程高手的事。




15.1 TCP/IP 网络通信

15.1.1 TCP 协议通信

TCP 协议是一种面向连接的传输层协议，所谓“面向连接”是指在传输数据之前，在通信两端必须建立可靠的连接，否则无法执行通信的技术。采用 TCP 协议传输数据其遵循的流程是，先由发送端发出连接请求，当接收端侦听到请求回复后则建立起连接。随后将进行数据的传输，一旦完成传输则关闭连接并结束通信。

TCP 函数的各节点功能如表 15-1 所示。

表 15-1 TCP 函数各节点功能表

函数名称	图标	使用说明
TCP 侦听		在服务器端创建一个侦听，并在指定的端口上等待关闭 TCP 连接请求
打开 TCP 连接		在接收端根据指定的计算机名和端口打开一个 TCP 连接
读取 TCP 数据		从指定的 TCP 连接读取数据



续表

函数名称	图标	使用说明
写入 TCP 数据		将数据写入指定的 TCP 连接
关闭 TCP 连接		关闭指定的 TCP 连接
IP 地址至字符串转换		将 IP 地址转换成对应的计算机名, 默认输出为本地计算机名
字符串至 IP 地址转换		将计算机名转换成对应的 IP 地址, 默认输出是本地计算机 IP 地址
解释机器别名		解释指定主机的网络 IP 地址, 如果解释不成功则返回输入的主机名
创建 TCP 侦听器		创建一个 TCP 端口的侦听
等待 TCP 侦听器		在指定的侦听端口等待连接请求 www.91HMI.com

要进行 TCP 通信, 需要收发双方及相应程序的支持。

(1) 创建服务器端

服务器端的主要任务是生成一个正弦波形数据, 用于波形显示和 TCP 传输。而 TCP 的任务则是建立连接、发送波形数据及关闭 TCP。

创建服务器端的步骤如下。

① 在前面板, 放置一个波形图控件和一个停止按钮。

② 切换到后面板, 放置一个 For 循环图标, 并置循环次数为“100”。

③ 在循环内, 放置一个正弦函数和一个乘函数。

④ 在循环外, 添加除函数、数值常量“ 2π ”和“5”, 以形成正弦函数的步进角。

⑤ 将步进角与变量 i 相乘, 并以其为正弦函数的角度值, 通过连接波形图就可观察信号源的状况。

⑥ 为使信号源可受控运行, 在信号源的外部再添加一层 while 循环, 并将循环条件端口与停止按钮相连。至此, 就完成了信号源的设计。

⑦ 在循环外层, 选取“函数→数据通信→协议→TCP→TCP 侦听”来添加该节点。并为“端口”创建一个输入控件。

⑧ 再添加一个“等待 TCP 侦听器”节点, 再与“TCP 侦听”节点相连, 这样就建立起一个 TCP 侦听环节。

⑨ 在外层循环内, 依次添加 2 个“写入 TCP 数据”节点。

⑩ 为了发送波形数据及数据的多少, 选取“函数→编程→数值→数据操作→强制类型转换”来添加该转换。

⑪ 添加“强制类型转换”、“数组大小”和“关闭 TCP 连接”图标, 经连线其程序框图和显示效果如图 15-1 所示。

(2) 创建客户端

客户端的主要任务是打开 TCP 连接、接收波形数据及关闭 TCP。其次就是将接收的数据重新还原成正弦波。

创建客户端的步骤如下。

① 在前面板, 放置一个波形图控件和一个停止按钮。

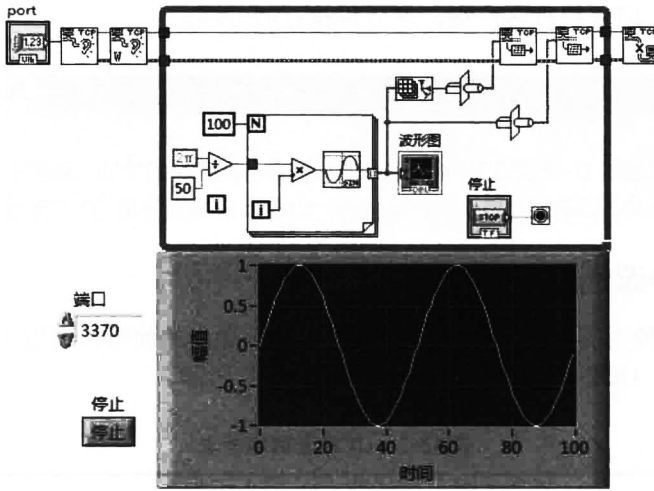


图 15-1 TCP 服务器端

www.91HMI.com

- ② 切换到后面板, 添加一个 while 循环, 并将循环条件端口与停止按钮相连。
- ③ 在循环外, 添加一个“打开 TCP 连接”节点, 并为“地址”和“远程端口或服务名称”端口添加相应的输入控件。这样就打开了指定的 TCP。
- ④ 添加 2 个“读取 TCP 数据”节点和 2 个“强制类型转换”图标, 以读取数据长度和波形数据。
- ⑤ 添加数值常量“2”和一维数组, 以指定读取数据的类型。
- ⑥ 波形图图标移入循环内, 并与接收数据相连以显示数据的波形。
- ⑦ 在循环外, 添加一个“关闭 TCP 连接”图标。经连线程序框图和显示效果如图 15-2 所示。

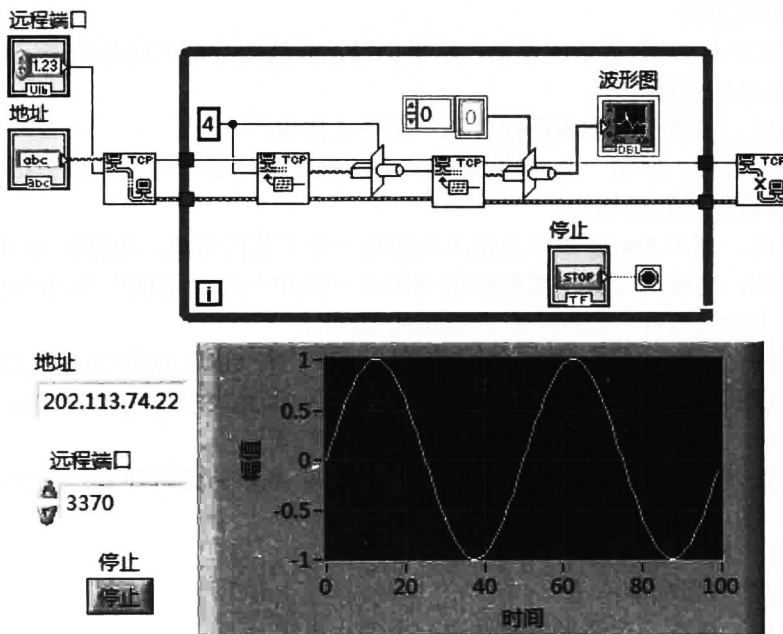


图 15-2 TCP 客户端



注：在运行时，应先执行客户端，然后再执行服务端。





提示

通信端口的地址范围时 0~65535，不同的通信服务要占用不同的地址，通常不要使用 1000 以下的端口号，因为 1000 以下端口号是系统保留的，如 http 的端口号是 80，而 ftp 的端口号是 21。

15.1.2 UDP 协议通信

UDP 通信是一种无需连接的通信，也不需要进行端口的侦听，所以 UDP 协议通信要比 TCP 协议更为简单。UDP 函数的各节点功能如表 15-2 所示。

表 15-2 UDP 函数功能表

函数名称	图标	使用说明
打开 UDP		在指定的端口上打开一个传送 UDP socket
打开 UDP 多点传输		在指定的接收端口上打开多点传送 UDP socket
读取 UDP 数据		根据指定的 UDP socket 读取数据，并返回发送端计算机的地址和端口
写入 UDP 数据		根据指定的计算机地址、端口和 UDP socket 读取数据，并返回发送端计算机地址和端口
关闭 UDP		关闭指定的 UDP socket

要进行 TCP 通信，需要收发双方及相应程序的支持。

(1) 创建服务器端

为了对比 TCP 与 UDP 通信的差别，这里仍以正弦波的数据传送为例来进行介绍。

创建服务器端的步骤如下。

- ① 在前面板，放置一个波形图控件和一个停止按钮。
- ② 切换到后面板，添加一个 while 循环，并将循环条件端口与停止按钮相连。这样就构成一个可控的持续发送数据结构。
- ③ 在循环内，使用 For 循环与正弦函数构造一个正弦信号源，数据量为 100。
- ④ 在循环外，选取“函数→数据通信→协议→UDP→打开 UDP”来添加该函数，并为“端口”创建一个输入控件，以此来打开指定的 UDP。
- ⑤ 在循环内，添加一个“强制类型转换”图标和一个“读取 UDP 数据”图标，并为“地址”和“端口或服务名称”端口创建相应的输入控件，以此来发送正弦波数据。
- ⑥ 在循环外，添加一个“关闭 UDP”图标，来结束通信。
- ⑦ 为了减缓传送的频度，在循环内添加一个 100 毫秒的延时环节。经连线，其程序框图和显示效果如图 15-3 所示。

(2) 创建客户端

创建客户端的步骤如下：

- ① 在前面板，放置一个波形图控件和一个停止按钮。
- ② 切换到后面板，添加一个 while 循环，并将循环条件端口与停止按钮相连。

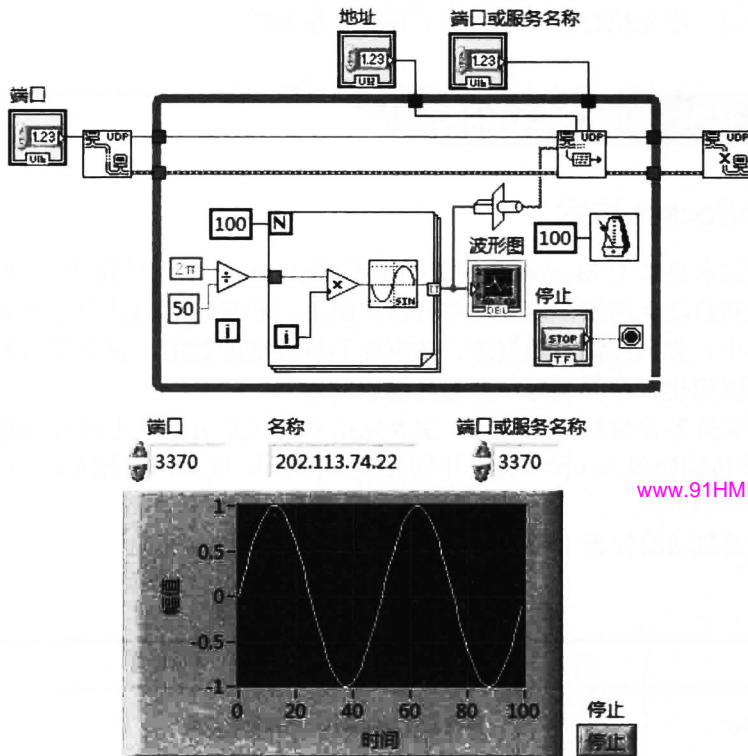


图 15-3 创建 UDP 服务器端

③ 在循环外，添加一个“打开 UDP”图标，并为“端口”创建一个输入控件，以此来打开指定的 UDP。

④ 在循环内，添加一个“读取 UDP 数据”图标和一个“强制类型转换”图标，并指定读取数据量和数据的类型，以此来接收正弦波数据。

⑤ 在循环外，添加一个“关闭 TCP 连接”图标，以此来结束通信。经连线，其程序框图和显示效果如图 15-4 所示。

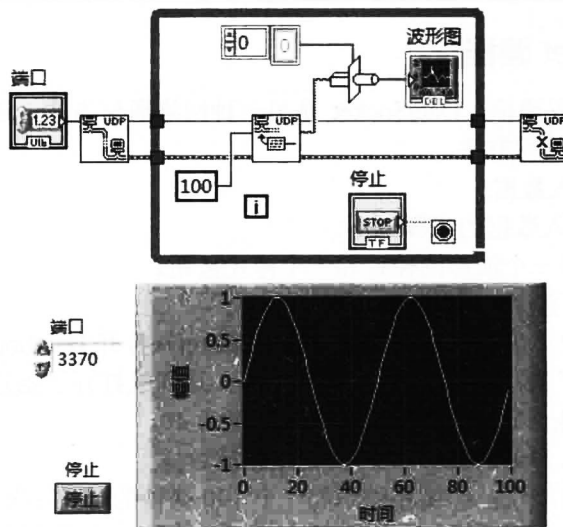


图 15-4 UDP 客户端



注：在运行时，应先执行客户端，然后再执行服务端。

15.2 DataSocket 通信

15.2.1 DataSocket 技术






DataSocket 技术是基于 Microsoft 的 COM 和 ActiveX 技术，是对 TCP/IP 协议的高度封装。它面向测量和自动化应用，能实现夸机器、跨语言和跨进程地共享实时数据。

在应用过程中，你只需面对数据源、数据库和待传送的数据，完全可以无视具体的实现细节。因而它可以简化程序的编写，并提高编程的效率。

DataSocket 支持多种数据传送协议，主要包括 **dstp**（专用通信协议）、**http**（超文本传输协议）、**ftp**（文件传输协议）、**opc**（操作计划与控制协议）、**logos**（数据记录与监控技术协议）及 **file** 协议。
www.91HMI.com

DataSocket 函数功能如表 15-3 所示。

表 15-3 DataSocket 函数功能表

函数名称	图标	使用说明
读取 DataSocket		从指定的 DataSocket 连接读取数据
写入 DataSocket		将数据写入到指定的 DataSocket 连接
DataSocket 选择 URL		弹出 URL 选择对话框来选择 URL，并返回选中的 URL
打开 DataSocket		根据指定的 URL 打开一个 DataSocket 连接
关闭 DataSocket		关闭指定的 DataSocket 连接

15.2.2 DataSocket 编程

有别于 TCP/IP 协议通信，DataSocket 是另一种网络通信方式，还以发送正弦波数据为例来介绍 DataSocket 的编程方法。

(1) DataSocket 写入数据

实现 DataSocket 写入数据的步骤如下。

- ① 在前面板，放置一个波形图控件和一个停止按钮。
- ② 切换到后面板，添加一个 **while** 循环。
- ③ 在循环外，选取“函数→数据通信→DataSocket→打开 DataSocket”来添加该函数，并为“URL”和“模式”端口添加相应的输入控件。这样就打开了指定的 DataSocket。
- ④ 在循环内，构建一个正弦信号源，其数据量为 40。
- ⑤ 添加一个“写入 DataSocket”，以发送正弦波数据。
- ⑥ 为了减缓传送的频度，在循环内添加一个 100 毫秒的延时环节。
- ⑦ 添加一个“或”运算图标，并通过停止按钮和读取的结束循环来控制循环的退出。



⑧ 在循环外，添加一个“关闭 DataSocket”图标。经连线，其程序框图和显示效果分别如图 15-5 和图 15-6 所示。

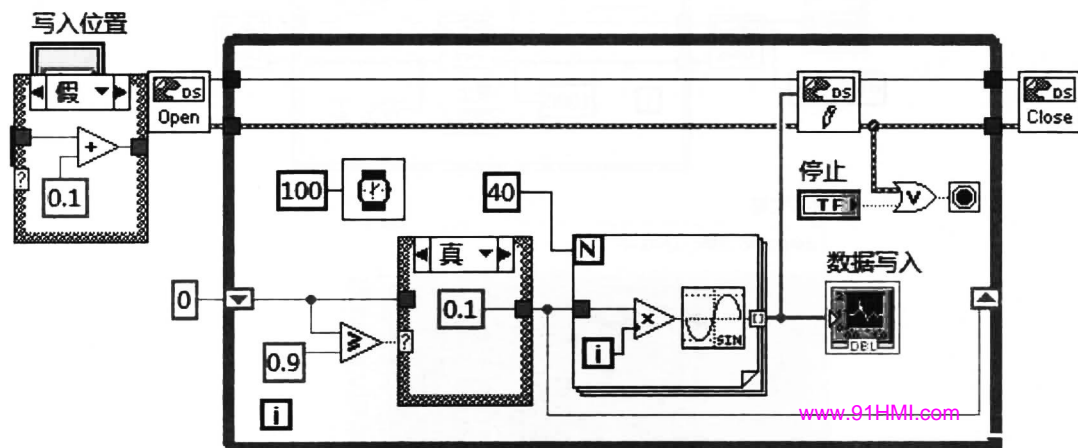


图 15-5 DataSocket 写入数据程序框图

写入位置

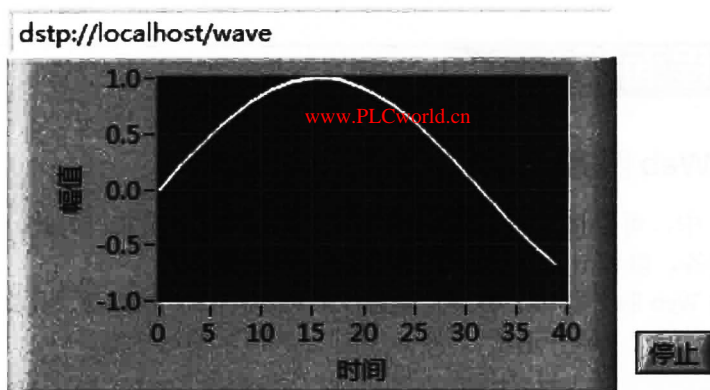


图 15-6 DataSocket 写入数据界面

(2) DataSocket 读取数据

实现 DataSocket 读取数据的步骤如下。

- ① 在前面板，放置一个波形图控件和一个停止按钮。
- ② 切换到后面板，添加一个 while 循环。
- ③ 在循环外，添加一个“打开 DataSocket”图标，并为“URL”和“模式”端口添加相应的输入控件。这样就打开了指定的 DataSocket。
- ④ 在循环内添加一个“读取 DataSocket”图标，并指定读取的数据量和数据类型，以读取接收的数据。
- ⑤ 添加一个“或”运算图标，并通过停止按钮和读取的结束循环来控制循环的退出。
- ⑥ 在循环外，添加一个“关闭 DataSocket”图标。经连线，其程序框图和显示效果如图 15-7 所示。

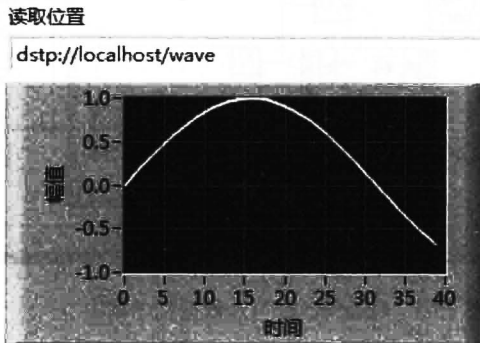
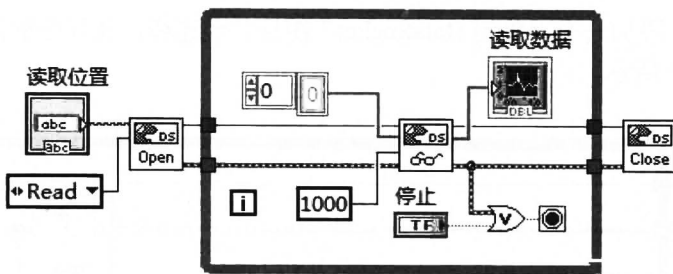


图 15-7 DataSocket 读取数据

15.3 远程面板

15.3.1 配置 Web 服务器

在 LabVIEW 中，可以将 VI 作为服务器端的 Web 服务来发布。为此，必须在网络上创建和部署 Web 服务，以使 HTTP 客户端可以访问此类应用。

若将 VI 作为 Web 服务发布，必须先激活 LabVIEW Web 服务器，其具体的操作是：

- ① 选取菜单栏的“工具→选项”，则弹出图 15-8 所示的对话框。

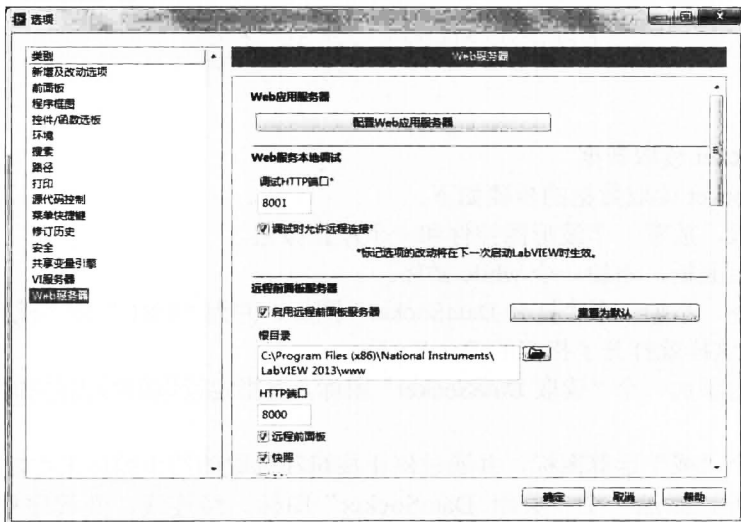


图 15-8 Web 服务器设置

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

www.91HMI.com

www.PLCworld.cn

www.PLCworld.cn

www.91HMI.com



② 选择“Web 服务器”选项，在“Web 服务本地调试”部分勾选“启用远程前面板服务器”。

15.3.2 操作远程面板

使用远程面板 Remote Panels 实现网络通信，同样涉及服务器端和客户端。

(1) 创建服务器端

创建远程面板服务器端的步骤如下。

① 在后面板，放置一个“获取日期/时间字符串”图标，为了显示完整的时间值，分别为“需要秒？(F)”端口创建一个逻辑常量“T”，为“时间字符串”端口创建一个显示控件。这里的时间值将是数据传送的对象。

② 为了能持续的发送数据，添加一个 while 循环，并为循环的条件端口创建一个输入控件。

③ 为了减缓传送的工作量，在循环内部添加一个 500 毫秒的延时环节。经连线其程序框图如图 15-9 所示。

④ 如需查看前面板各控件的顺序，可在前面的状态下选取菜单栏的“编辑→设定 Tab 键顺序”，则见到图 15-10 所示的界面。在顺序号上单击鼠标，则可关闭指定控件的顺序值，单击“确认”工具按钮，可保存设置并退出。否则，设置失效。

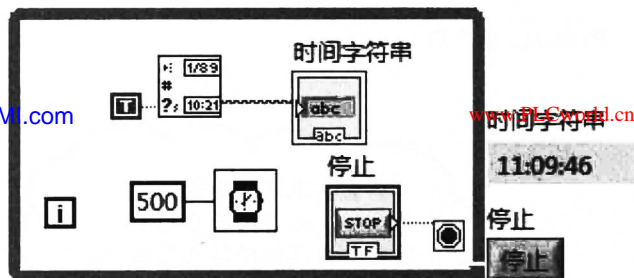


图 15-9 创建服务器端



图 15-10 设置控件顺序

(2) 创建客户端

创建远程面板客户端的步骤如下。

① 在后面板，选取“函数→编程→应用程序控制→打开应用程序引用”来添加该函数，再为“机器名”端口创建一个输入控件，以获取服务器上应用程序的引用。

② 添加一个“调用节点”图标，并与“打开应用程序引用”图标相连接。

③ 右键“调用节点”图标，从快捷菜单中选取“选择方法→连接响应→信息”，以检测连接的状态。

④ 添加一个“打开 VI 引用”图标，并为“VI 路径”端口添加一个字符串常量“server.vi”。

⑤ 为“VI 引用”端口创建一个前面板属性节点。

⑥ 为“前面板”端口创建一个“控件[]”属性节点，进而得到所有控件的引用。

⑦ 为“控件[]”端口添加一个“索引数组”图标，以此来获得前面板的控件个数。

⑧ 添加一个“转换为特定的类”图标，并为“目标类”端口添加一个类说明符常量。

⑨ 右击类说明符常量，从快捷菜单中选取“通用→图形对象→控件→字符串→字符串”。

⑩ 添加一个 while 循环，并为条件端口创建一个输入控件，以便不断地读取服务器程序所发出的时间值字符串。



- ⑪ 在循环内，添加一个属性节点，并与“转换为特定的类”图标相连。
- ⑫ 右击属性节点，从快捷菜单中选取“选择属性→值”，并为“值”端口创建一个显示控件。
- ⑬ 在循环外，添加一个“关闭引用”图标，经连线路程序框图和显示效果如图 15-11 所示。

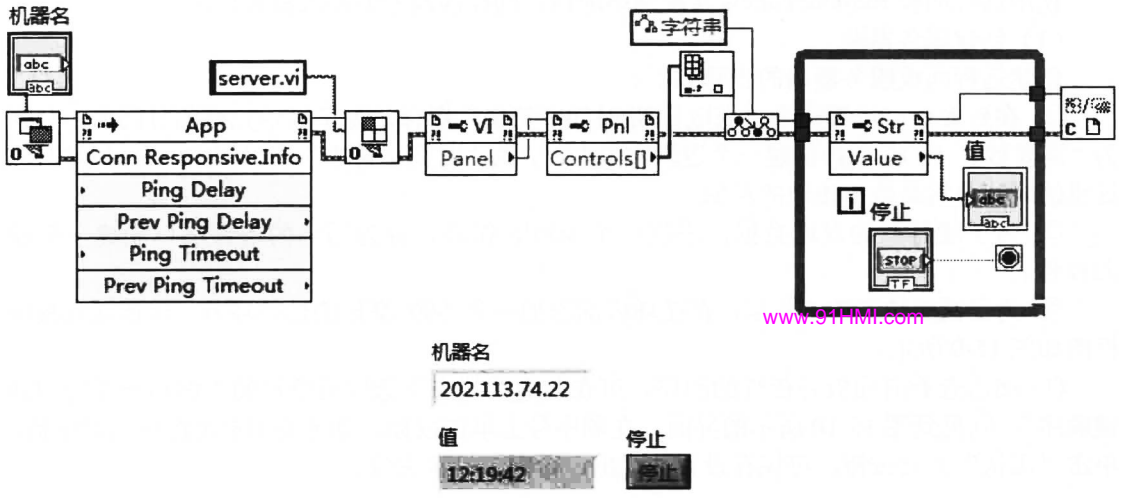


图 15-11 客户端



尽管实现网络通信的方法很多，但在实际应用中还是要视具体的特点做出取舍。



第 16 章

LabVIEW总线技术与仪器控制

利用 LabVIEW 设计的虚拟仪器必须有实际硬件的支持才能达到目的，而总线技术及仪器控制技术就是连接虚拟仪器和实际硬件的桥梁。本章介绍的总线技术主要应用于仪器控制领域，并由 LabVIEW 实现仪器的控制。

【本章导读】

- ◆ LabVIEW 2013 支持的总线
- ◆ GPIB 总线
- ◆ VXI、PXI 和 LXI 总线
- ◆ 串行总线
- ◆ VISA 的使用

◆ 用 NI MAX 查找仪器

www.PLCworld.cn

16.1 LabVIEW 2013 支持的总线

16.1.1 仪器总线技术简介

总线是连接一个或多个部件的信号线总称，它可以应用于不同的领域，如计算机、测试与测量、自动控制等，它们都与总线技术密不可分。

总线的特点在于它有良好的公用性和兼容性，能同时挂接多个功能部件，并可以互换使用。总线技术发展的很快，这与其自身的优越性不无关系。总线概念的提出，不仅简化了电路设计及系统结构，同时还大大地加快了计算机及相关领域的发展速度。

按总线的功能，可将总线归为三类：即数据总线、地址总线和控制总线。其中：数据总线用于传输数据，多采用双向三态逻辑。数据总线是并行的，即数据有一定的宽度，它反映了总线的性能；地址总线用于传送地址信息，并采用单向三态逻辑；控制总线用于传递控制信号和状态，其既可以是单向的，也可以是双向的，这主要取决于使用的条件。

对于仪器总线，可分为两类，即外部总线和内部总线。所谓“外部总线”也称独立总线，主要供仪器设备与上位机（PC 或工控机）进行连接，主要包括 GPIB、RS232、USB、LAN/LXI 和 IEEE1394 等。而“内部总线”通常被称为模块化仪器总线，是指把接口总线合并到仪器中，通常包括 VXI、PCI/PXI、PCI/PXI Express 等。

衡量总线的性能主要涉及下列指标：

- ① 总线宽度：指数据总线宽度，单位是位（bit）。如 8 位/16 位 GPIB 总线，32 位 PXI 总线。



- ② 寻址能力：主要指地址总线的位数及所能直接寻址的存储器空间大小。
- ③ 总线时钟频率：处理器完成一步操作的最小时间的倒数。是影响总线传输速率的重要因素之一，总线频率越高，传输速率越快。如 PCI 总线的工作频率为 33MHz。
- ④ 数据传输速率：总线上每秒传输的最大字节数，计算公式为：

$$Q = \frac{\text{总线宽度}}{8} \times \text{总线频率 (MBps)}$$

- ⑤ 同步方式：有同步和异步两种方式。
- ⑥ 负载能力：总线上能挂接的器件个数。

16.1.2 仪器控制及 LabVIEW 2013 支持的总线类型

仪器控制是指上位机（PC 或工控机）通过总线控制一台或多台仪器设备。通常上位机和仪器设备是靠某种总线来进行连接，并由上位机发出指令，仪器设备则予以响应并反馈特定参数来协同工作。它借助上位机的数据处理、分析、显示及超强存储力，可以极大地增强仪器设备的功能，并能有效地拓展系统规模。

一个完整的仪器控制系统，除了上位机和仪器设备外，还需要底层仪器驱动程序、上层应用控制程序以及相应的传输总线。所谓仪器驱动程序实为一个指令集，是将底层的通信命令或寄存器配置等封装起来。使用时，用户只需调用封装好的函数库就可实现仪器的轻松操作。通常一个指令集可以控制一种仪器或一类仪器，一般是由仪器开发商来提供。

对于 LabVIEW 使用者，在一般情况下他无需了解底层的驱动程序，只需下载仪器所对应的 LabVIEW 驱动程序即可。

提示

LabVIEW 所支持的第三方仪器驱动程序可以到 NI 官网下载。
<http://www.ni.com/downloads/drivers/>

LabVIEW 2013 所支持的总线类型几乎涵盖了各类流行的总线，图 16-1 展示了基于 LabVIEW 的仪器控制系统框架，所涉及的总线有 GPIB，RS232，USB，LAN 等。

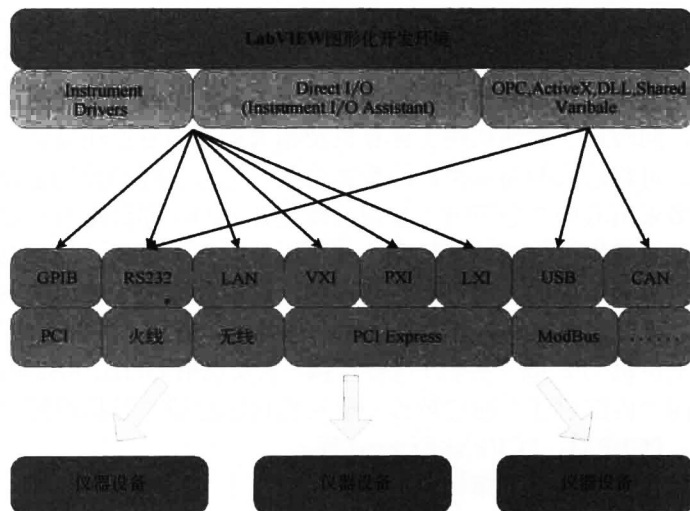
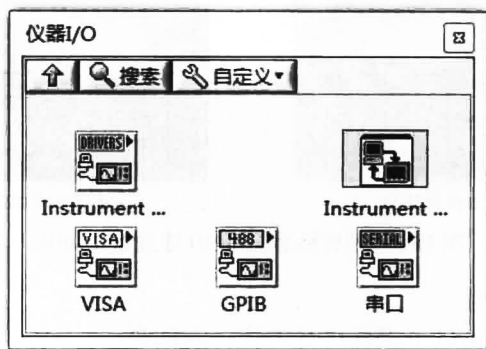


图 16-1 基于 LabVIEW 的仪器控制系统框架



16.1.3 LabVIEW 2013 中的仪器 I/O 选板

在 LabVIEW 2013 中, 要使用某种总线, 可由仪器 I/O 选板来实现, 其操作顺序是“函数→仪器 I/O”, 其选板的内容如图 16-2 所示。



www.91HMI.com

图 16-2 仪器 I/O 选板

在仪器 I/O 选板中, 涉及以下功能。

- ① Instrument Drivers: 包含各种已在 LabVIEW 下开发好的仪器设备应用 vi。
- ② VISA: 包含 VISA 模块子 vi。
- ③ GPIB: 包含 GPIB 模块子 vi。
- ④ 串口: 包含串口模块子 vi。
- ⑤ Instrument I/O Assist: Express vi 可以通过简单的定义, 辅助连接 VISA 资源仪器设备。后面会依据不同的总线, 对各功能模块加以具体介绍。

16.2 GPIB 总线

16.2.1 GPIB 总线简介

GPIB 是标准通用接口总线的缩写 (General Purpose Interface Bus), 是通用仪器接口标准。在智能仪器上, 几乎都配有 GPIB 标准接口。

在工程中, GPIB 总线包括 HP-IB、GPIB、IEEE Bus、ASCII Bus、IEC-IB、IEC 625-Bus、ANSI MC1.1 Bus、IEEE488.1 和 IEEE488.2, 它们都是以 IEEE488 所规定的标准规范作为基础。在美国, GPIB 总线称为 IEEE488.2 或 HP-IB, 欧洲和日本称为 GPIB 或 IEC 625。在中国, GPIB 的专业标准等同于 IEC 625, 并在 1986 年作为国家标准开始实施。

(1) GPIB 总线信号结构

GPIB 总线是一种具有 24 (IEEE488.2 标准) 或 25 (IEC625 标准) 针引脚的并行总线。其中包括 8 根双向数据线、3 根数据传输控制线、16 根电平信号传输线、5 根接口管理线和 8 根或 9 根地线/屏蔽线。GPIB 使用 8 位并行、字节串行、异步通信方式, 所有字节通过总线顺序传送。

(2) GPIB 总线机械结构

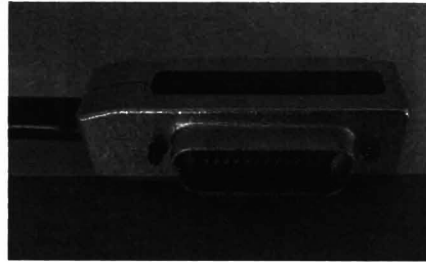
GPIB 总线的机械结构主要是指电缆和连接器的形式, 通常电缆连接器为双面结构接插件, 从而便于设备互联; 接插件的形式则会依据地域的不同有所差异, IEEE488 美国标准为



24 芯弹簧片连接器, IEC625 国际标准为 25 芯针形连接器。我国是采用 IEC 国际标准。图 16-3 是两种连接器的实物图。



(a) 国际标准 IEC625 连接器



(b) 美国标准 IEEE488.2 连接器

图 16-3 两种标准的 GPIB 连接器实物图

(3) GPIB 总线基本特性

GPIB 总线的特性主要由以下指标进行考评, 它们是:

www.91HMI.com

- ① 数据传输速率: 通常 GPIB 总线传输速率为 10000~250000B/s, 最高可达 1MB/s。其传输速率与总线长度和接口的发送器有关。
- ② 地址容量: 单字节最大 31 个, 双字节最大 961 个。
- ③ 连接仪器数量: 最多 15 台。
- ④ 最大传输距离: 20m。若要长距离传输, 可利用扩展器进行距离扩展。
- ⑤ 电平: 逻辑 1 小于等于+0.8V, 逻辑 0 大于等于+2.0V。

(4) GPIB 仪器分类

www.PLCworld.cn

在计算机同各类仪器组成的 GPIB 总线系统中, 按照仪器的作用, 可分为三类:

- ① 讲者 (Talker): 在数据传输过程中, 产生并向总线发送消息的装置。GPIB 系统工作时, 只允许有一台仪器充当讲者。若没有讲者, 就没有数据传输。
- ② 听者 (Listener): 在数据传输过程中, 通过总线来接收讲者所发来消息的装置。一个 GPIB 系统中, 可以同时具有多个听者。
- ③ 控者 (Controller): 是数据传输的组织者和控制者。控者负责指定数据传输过程中的讲者和听者, 在 GPIB 系统中只能有一个控者, 且多数由计算机来担当。

16.2.2 GPIB 子面板

使用 GPIB 子面板能对 GPIB 仪器进行操作, 进入的路径为“函数→仪器 I/O→GPIB”, 其面板如图 16-4 所示。

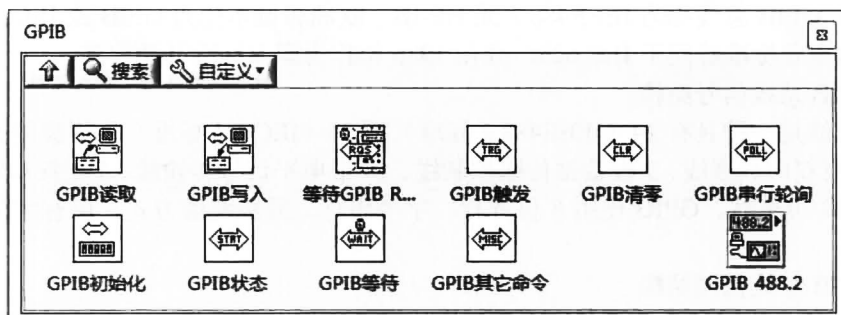


图 16-4 GPIB 子面板



(1) 相关函数

在 GPIB 子面板下, 提供了多个 GPIB 函数及 GPIB488.2 函数。对于初学者, 只要掌握主要函数的使用就可实现设备的开发工作。

- ① GPIB 初始化: 与 GPIB 设备进行连接, 实现仪器初始化。
- ② GPIB 读取: 从指定地址的 GPIB 设备读取数据。
- ③ GPIB 写入: 向指定地址的 GPIB 设备发送指令或数据。
- ④ GPIB 清零: 发送 SDC 选中设备清零或 DCL 设备清零指令。

(2) 初始化设备

调用 GPIB 初始化函数, 可对 GPIB 设备进行初始化, 其函数如图 16-5 所示。

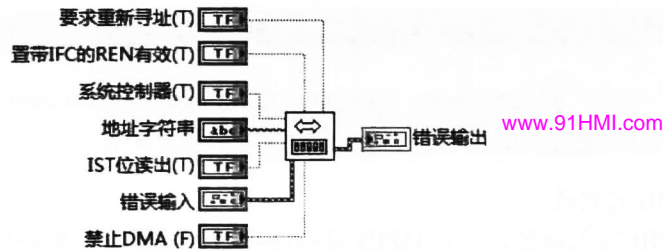


图 16-5 GPIB 初始化函数

其输入控件如下。

- ① 地址字符串: GPIB 设备的地址号, 通常需要查询 GPIB 设备手册确定。
- ② 要求重新寻址: 若值为真, 则该函数可在每次读取或写入前寻址设备。若值为假, 设备须使保留寻址至下一次读取或写入。默认值为真。
- ③ 置带 IFC 的 REN 有效: 若值为真, 并且控制器为系统控制器时, 则 GPIB 设备发送一个远程控制信号。默认值为真。
- ④ 系统控制器: 若值为真, 控制器为系统控制器。默认值为真。
- ⑤ IST 位读出: 若值为真, 设备的个别状态位用 TURE 响应并行轮询; 如值为假, 设备的个别状态位用 FALSE 响应并行轮询。默认值为真。
- ⑥ 禁止 DMA: 若值为真, 设备使用程序控制 I/O 传输数据。默认值为真。
- ⑦ 错误输入: 错误代码输入控件。可接受前一级 vi 的错误代码输入。

输出控件为:

错误输出: 错误代码输出控件。显示错误状态及错误代码。

(3) GPIB 读操作

调用 GPIB 读取函数, 可对 GPIB 设备进行读取操作, 其函数如图 16-6 所示。

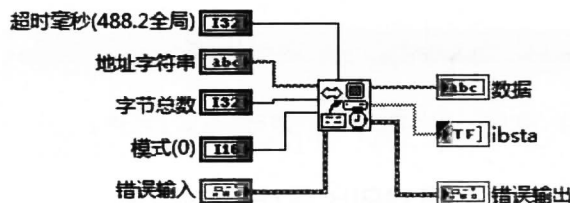


图 16-6 GPIB 读取函数



其输入控件有：

- ① 地址字符串：GPIB 设备的地址号，通常需要查询 GPIB 设备手册确定。
- ② 字节总数：需要从 GPIB 设备读取数据的字节总数。
- ③ 超时毫秒：函数在超时前等待的时间。若函数在设置的时间内未完成，则输出错误。

单位为毫秒。设置数值 0 可禁用超时，设置数值-1 可使用 488.2 全局超时。默认值为-1。

- ④ 模式：指定在没有达到字节总数时终止读取的条件。默认值为 0。

其输出控件有：

- ① 数据：函数读出的数据，为字符串类型。
- ② ibsta：GPIB 状态。每一位都表示 GPIB 控制器的一个状态。

提示

显示控件“数据”为字符串格式，若使用该数据，可利用“编程→字符串”选板中的相应函数来实现字符串到其他数据类型的转换。
www.91HMI.com

(4) GPIB 写操作

调用 GPIB 写入函数，可对 GPIB 设备进行写入操作，其函数如图 16-7 所示。

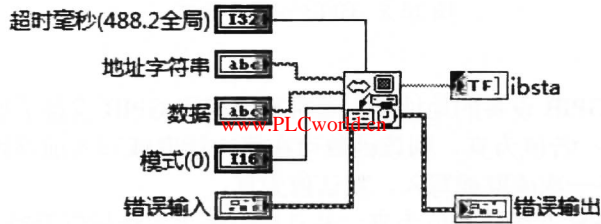


图 16-7 GPIB 写入函数

其输入控件有：

- ① 地址字符串：GPIB 设备的地址号，通常需要查询 GPIB 设备手册确定。
- ② 数据：向 GPIB 设备写入的数据或指令，数据类型为字符串。
- ③ 超时毫秒：函数在超时前等待的时间。若函数在设置的时间内未完成，则输出错误。单位为毫秒。设置数值 0 可禁用超时，设置数值-1 可使用 488.2 全局超时。默认值为-1。

- ④ 模式：指定在没有达到字节总数时终止读取的条件。默认值为 0。

其输出控件为：

ibsta：GPIB 状态。每一位都表示 GPIB 控制器的一个状态。

提示

也要注意其中显示控件“数据”为字符串格式，使用方法同上所述。

16.2.3 在 LabVIEW 中实现 GPIB 总线控制

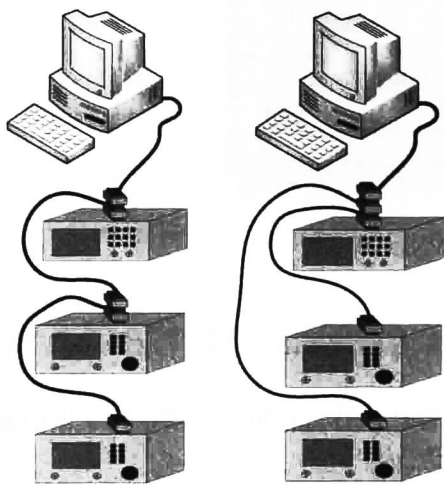
GPIB 总线控制要靠 GPIB 硬件支持，通常 GPIB 仪器系统是由计算机、GPIB 接口硬件



及若干 GPIB 设备构成。

GPIB 系统有两种连接方式，分别是线型连接和星型连接，如图 16-8 所示。此外，两种连接方式还可以混合使用。GPIB 设备地址为 0~30，通常将 GPIB 接口卡地址设为 0，而将其他仪器设备地址设为 1~30。

在应用中，向计算机的 PCI 插槽上插入一块 GPIB 板卡即可控制 GPIB 仪器。随着 USB 接口的普及，GPIB 仪器的连接方式也在 USB 接口化。图 16-9 是 NI 公司的 GPIB-USB 转换器，该连接器采用美国标准。



(a) 线型连接

(b) 星型连接 www.PLCworld.cn

图 16-8 GPIB 系统的连接方式



图 16-9 NI 公司的 GPIB-USB 转换器

提示

对于采用国际标准 IEC625 的仪器（25 针接口），其转换器需要配备一条 24-25 针的转换线。

16.3 VXI、PXI 和 LXI 总线

在仪器、仪表领域，应用较为广泛的三种总线，即 VXI 总线、PXI 总线和 LXI 总线。

16.3.1 VXI 总线

VXI 是 VME Extensions for Instrumentation 的缩写，即“VME 总线在仪器领域的扩展”的简称。它是由 HP、Colorado Data Sy、Racal Dane、Tektronix 和 Wavetek 这五家著名的仪器公司于 1987 年联合推出，是一种开放的新一代自动测试系统工业总线标准。

VXI 总线是计算机技术、数字接口技术与电子测量技术相结合的产物，它结合了工业标准 VME 总线高速通信和 GPIB 接口易于组合、程控简单的特点，实现了模块化仪器结构。与传统军用测试系统采用机架式结构相比，VXI 系统的设计充分考虑了抗震、冷却、抗干扰等可靠性指标，适于在运动现场的环境下高可靠性的工作，图 16-10 是典型的 VXI 总线系统。

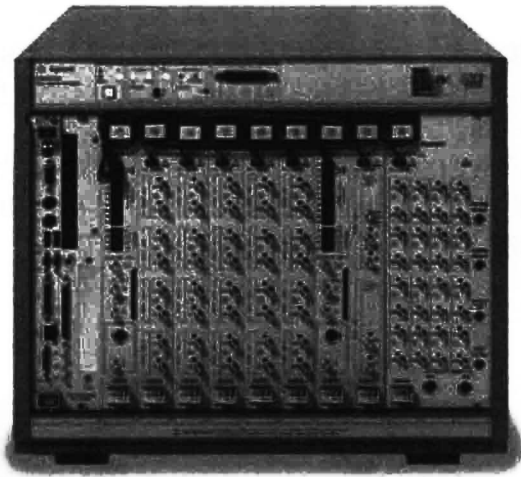


图 16-10 典型的 VXI 总线系统

www.91HMI.com

(1) 特点

① 系统搭建灵活: VIX 系统使用统一的机械与电气规范, 不同厂商的同类产品可以相互替换。

② 测试系统小型化: VXI 系统采用模块化设计, 具有体积小、重量轻、机构紧凑和易于携带等特点, 相比于 GPIB 系统, 体积缩小了近 3/4。

③ 高速数据传输: VXI 是一种 32 位并行总线, 传输速率理论上可达到 40MB/s。

④ 高可靠性及维护性: VXI 系统平均无故障时间达到几万甚至十几万小时, 可用于军事模块化仪器的测试设备。

(2) 系统结构

物理结构。VXI 总线系统是由提供背板的主机箱和模块卡组成, 如图 16-11 所示。一个 19 英寸宽的标准机箱最多可提供 21 个插槽位置。模块卡有 4 种标准尺寸, 分别为:

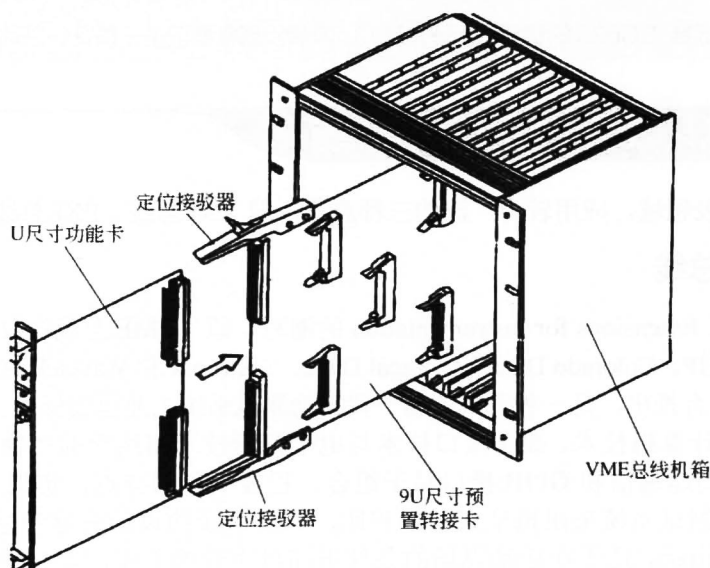


图 16-11 VXI 系统的物理结构



A 尺寸: $3U \times 160\text{mm}$

B 尺寸: $6U \times 160\text{mm}$

C 尺寸: $6U \times 340\text{mm}$

D 尺寸: $9U \times 340\text{mm}$

其中 $1U=1.75$ 英寸 (44.45mm)。

(3) 器件与线路

VXI 总线支持 32 位 VME 计算机总线。VME 总线包括数据传输总线 (Data Transfer Bus, DTB)、优先级中断总线 (Priority Interrupt Bus)、DTB 仲裁总线 (DTB Arbitration Bus)、公用总线 (Utility Bus)。除此之外, VXI 总线还增加了用于模拟供电和 ECL 供电的额外电源线、用于测量同步和触发的仪器总线、星状总线、模拟相加总线以及用于模块之间通信的本地总线。图 16-12 是时钟总线和星状总线示意图。VXI 总线规范定义了 3 个 96 针的 DIN 连接器 P1、P2 和 P3。其中 P1 连接器是必备的, 而 P2 和 P3 是两个可选连接器。

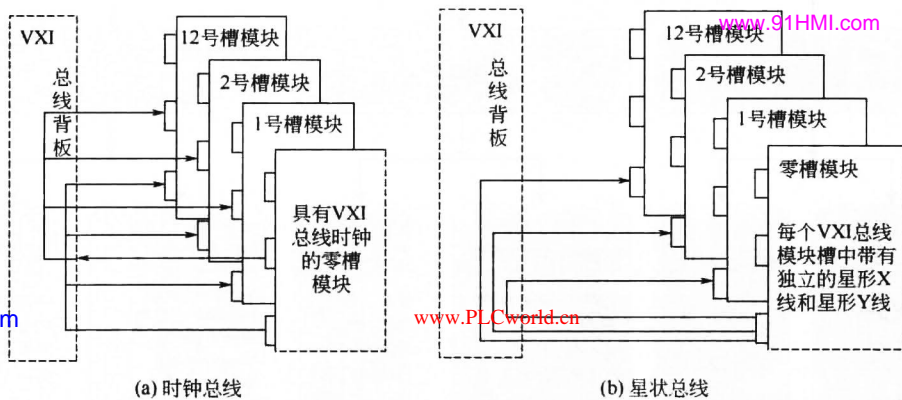


图 16-12 时钟总线和星状总线示意图

VXI 总线还规定了系统传导及辐射 EMC (电磁兼容) 产生和敏感度的上限值。EMC 的限制保证了包含敏感电路的模块能够完成所期望的操作, 而不受到系统中其他模块的干扰。

16.3.2 PXI 总线

PXI (PCI Extensions for Instrumentation) 是 NI 公司在 1997 年推出的一种全新的开放式、模块化的仪器总线规范。它将 CompactPCI 规范定义的外部组件互联总线 PCI (Peripheral Component Interconnect) 技术拓展为适合于试验、测量与数据采集场合的机械、电气和软件规范, 形成了新的虚拟仪器体系结构。同年, PXI 系统联盟 PXISA (PXI System Alliance) 成立, 这进一步加快了 PXI 与 CompactPCI 在自动化测量等领域的应用, 并促进了 PXI 规范维护和版本控制, 从而保障了多厂商系统在机械、电气和软件方面的兼容性。

PXI 总线与 PCI 总线高度的兼容性使得二者保持了较好的互换性, 使得 PXI 系统用户能够轻松地使用各种熟悉的 PC 软件工具和开发环境, 包括 PC 机的操作系统, 底层的硬件驱动、高级的仪器驱动和图形化 API 等。同时, PXI 实现了 VISA 规范, 不仅能够控制 PXI 模块, 也能控制 VXI、GPIB 及串行接口器件。这使得各类应用程序在不同总线系统之间的移植成为可能。因此, 在许多测试测量领域, PXI 系统是一个理想的选择。

(1) 特点

① PXI 总线是 PCI 总线的增强与扩展, 它兼容现有工业标准 CompactPCI。



② 一个标准 PXI 机箱可插入 1 个系统模块和 7 个仪器模块，通过扩展接口可连接多个 PXI 机箱。

③ PXI 总线提供 33/66MHz 的系统时钟和 10MHz 的 TTL 参考时钟。

④ PXI 总线提供 32~64 位数据传输。

⑤ PXI 总线提供 132MB/s 到 528MB/s 峰值数据传输速率。

⑥ 支持即插即用。

(2) 系统结构

PXI 总线系统结构包括三方面内容，即机械结构、电气总线结构和软件结构，如图 16-13 所示。

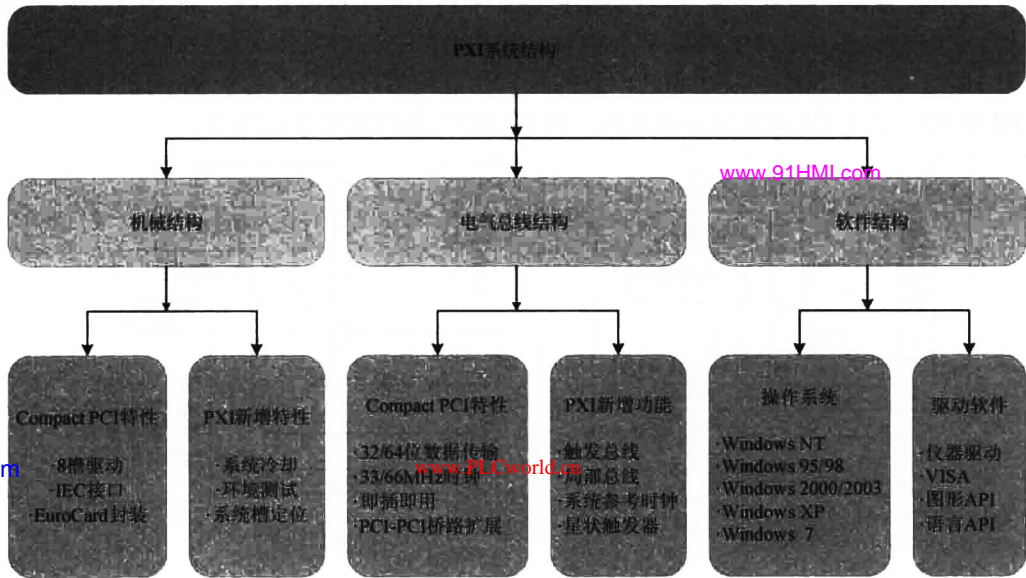


图 16-13 PXI 总线系统结构

① 机械结构。PXI 系统主要由三部分构成，即机箱、系统控制模块和仪器外设模块。一个 PXI 机箱可提供 1 个系统控制模块插槽和 7 个外设仪器模块插槽，其中的系统控制模块插槽必须位于最左端。图 16-14 所示为 NI 的 PXIe-8820 RT 控制器，它有 2.2 GHz dual-core Intel Celeron 1020E 双核处理器；2 GB 1333 MHz DDR3L RAM，最大可扩展到 8 GB；集成 1 个高性能以太网控制器，4 个高速 USB 接口，串行接口等，可安装标准的 Windows 操作系统。

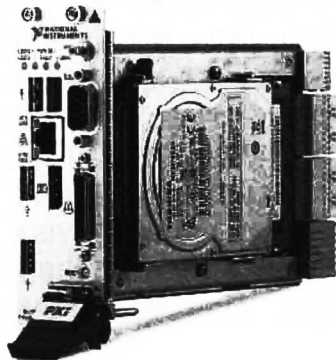


图 16-14 NI 的 PXIe-8820RT 控制器



图 16-15 是插入 PXIe-8820 控制器的 PXI 机箱，内嵌的 7 个外设模块包括高速以太网接口，信号发生器，数据采集卡，GPIB 接口卡等。

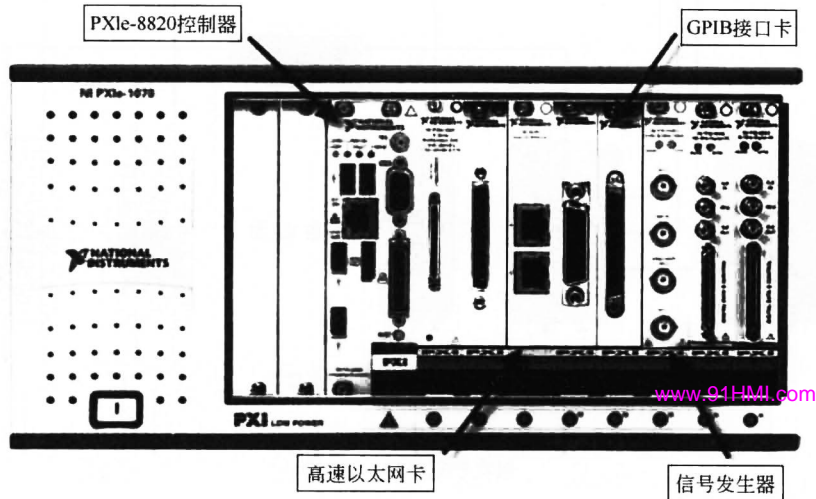


图 16-15 插入 PXIe-8820 控制器的 PXI 机箱



读者可到 NI 官方网站自行配置 PXI 系统。

<http://ohm.ni.com/advisors/pxi/pages/common/intro.xhtml>

PXI 系统支持两种模块规格，分别对应 VXI 系统的 A 尺寸（3U）和 B 尺寸（6U）。PXI 仪器模块的两种主要连接器为 J1 和 J2。在 J1 连接器上定义了标准的 32 位 PCI 总线，所有的 PXI 总线性能则定义在 J2 连接器上。PXI 采用与 CompactPCI 相同的高密度、屏蔽型、针孔式连接器，连接器引脚间距为 2mm，符合 IEC 1076 国际标准。而所有 CompactPCI 规范中定义的物理规范均适用于 PXI 3U 和 6U 模块。

② 电气总线结构。PXI 系统的总线标准是根据 PCI 的总线标准发展而来，它具有 PCI 总线全部的性能和特点，包括 32 位/64 位数据传输能力以及高达 132MB/s（32 位）和 264MB/s（64 位）的数据传输速度。此外，它还支持 PCI-PCI 桥路扩展、即插即用以及 3.3V 系统电压。

在保持 PCI 总线优点的前提下，它增加了专门的系统参考时钟、触发总线、星形触发线和模块间的局部总线，以此来满足高精度的定时、同步与数据通信要求。所有总线均置于 PXI 系统背板，在最左侧的是系统控制器，其右边相邻的是星状触发控制器，它与另外 6 个插槽之间配置了唯一的触发线，图 16-16 为 PXI 系统的电气结构图。

PXI 系统所集成的总线如下。

本地总线：本地总线是星状触发器与外设仪器模块以及外设模块之间的链状总线，共定义了 13 根信号线，用户可对其自行加以定义和使用，如选择传输模拟信号或者高速 TTL 信号等。但在 13 根信号线中，不包含系统控制器总线及星状触发线，既系统控制线和星状触发线不占用本地总线带宽。

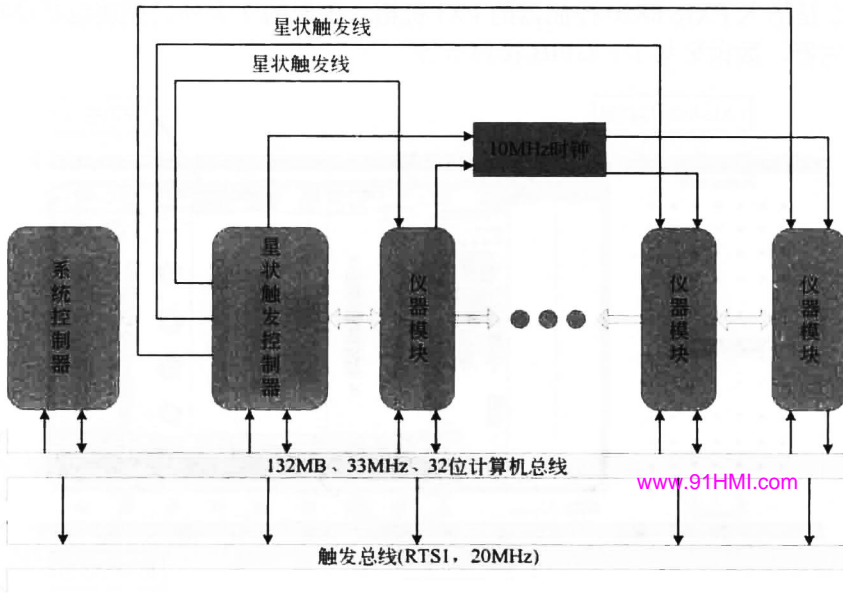


图 16-16 PXI 系统电气结构

触发总线：PXI 系统总线为 TTL 参考时钟定义了 8 根 TTL 触发线，且 PXI 规范定义了两种触发协议，即 PXI 异步触发协议和 PXI 同步触发协议。

星状触发线：PXI 星状触发线提供了更高性能的同步功能。星状触发控制器处于紧邻系统控制器右侧的第一个外设仪器模块插槽上，它与其他 6 个外设仪器插槽之间各配备了一根且唯一的触发线。若 PXI 系统插入一块星状触发控制器，就可以向另外的 6 个外设仪器提供高精度的触发信号。若不使用星状触发控制器，PXI 系统仍可正常工作。PXI 系统的星状触发控制器有两个特点：一是 PXI 规范规定，每个 PXI 机箱中至多有 1 个星状触发控制器，以使复杂的系统可以消除一根触发线上组合多个模块功能的限制；二是不同插槽间的星状触发信号传输延迟小，规范要求其不能大于 1ns。星状触发插槽至各外设仪器插槽的星状触发信号延迟不能大于 5ns，这样就保证了系统中每个模块间的非常精准的触发关系。

参考时钟：PXI 规范定义了将 10MHz 参考时钟分布到系统中所有外设仪器模块的方法，该参考时钟可用于同一测量或控制系统中的多卡同步信号。由于 PXI 系统严格定义了背板总线上的参考时钟，且参考时钟所具有的低延时性能可使各触发总线的时钟边缘更适于复杂的触发协议。

16.3.3 LXI 总线

在 HP 公司推出 GPIB，并成为第一代仪器总线 IEEE488 标准后，GPIB 总线几乎出现在每一台可编程控制仪器中，它的通用性大大地促进了测试及测量领域的发展。在 2004 年 9 月，安捷伦（Agilent）公司和 VXI 科技公司联合推出了新一代基于 LAN 的模块化平台标准 LXI（LAN-based extensions for Instrumentation）总线。

(1) LXI 系统与其他主流总线系统的比较

LXI 系统总线与其他主流总线的性能比较如表 16-1 所示，可见，LXI 系统在几何尺寸及系统成本方面相对其他总线系统有明显优势，同时，LXI 总线系统的模块本身配有处理器、LAN 连接器、电源供应和触发输入，因此它可作为仪器单独使用，这又是有别于其他总线系



统的一个突出优点。

表 16-1 主流仪器总线性能比较

参数	GPIB	VXI	LXI	PXI
定时和同步	无	有	有	有
几何尺寸	大	中	中/小	中/小
模块化	否	是	是	是
EMI 防护	可选	有	视具体模块而定	视具体模块而定
系统成本	高	中高	低中	低中

图 16-17 为 Agilent 公司基于 LXI 总线系统的数据采集器 L453X，它既可以集成于 LXI 系统，又可以单独使用。



图 16-17 基于 LXI 总线系统的数据采集器 L453X

(2) 仪器分类

根据仪器的功能和触发精度，可将 LXI 仪器分为 3 个等级，如表 16-2 所示。

表 16-2 LXI 仪器等级划分

等级	特征
A 类	触发总线硬件触发机制； IEEE1588 精确时间协议同步； 网络功能性（辨识、浏览界面）
B 类	IEEE1588 精确时间协议同步； 网络功能性（辨识、浏览界面）
C 类	网络功能性（辨识、浏览界面）

C 类是基本类型，使用网线触发，没有基于时间的触发能力。

B 类在 C 类的基础上增加了 IEEE1588，它允许测试系统内的不同设备自主地进行复杂的时间序列，而无需系统控制器的干预。

A 类在 B 类的基础上还加上 8 路的 LVDS 高速触发总线。

在 3 类方式中，其同步精度是依次递增的，这种划分完全是基于仪器的功能，与物理尺寸无关，故称为 LXI 功能类。不同需求的用户可选用不同的功能类，不同功能类的仪器间可



以协同工作。

(3) 关键技术

LXI 是一种功能接口标准,它定义了基于 802.3 标准的接口技术来确保仪器间的互通性。它由一个嵌入式的 IEEE1588 协议来提供必需的同步能力。此外,它还指定了相关的有线触发总路来为重要或关键的应用程序提供改进的同步能力。

LXI 的推出得益于以下关键技术:

① 定时与同步技术: LXI 提供精度由低到高的三种触发机制:基于 NTP 的触发方式、基于 IEEE1588 (PTP) 的触发方式和基于 LXI 触发总线 (LXI Trigger Bus) 的硬件触发。

② 基于 NTP 的触发方式:网络时间协议 (Network Time Protocol, NTP) 是互联网的时间同步标准协议,其用途是把计算机的时间同步到某些时间标准。

③ 基于 IEEE1588 (PTP) 的触发方式:精确同步时钟协议 (Precision Time Protocol, PTP) 又称为 IEEE1588,该协议定义了一个在测量和控制网络中,与网络交流、本地计算和分配对象有关的精确同步时钟的协议,它的设计目的在于针对更稳定和更安全的网络环境,尤其是用于工业自动化和测量环境。

④ 基于 LXI 触发总线 (LXI Trigger Bus) 的硬件触发: LXI 的硬件触发系统基于高速有线 LVDS (低压差分信号) 触发总线,该总线将各个模块通过硬件接口连接在一起,其连接方式与 VXI 和 PXI 的背板总线比较相似。

⑤ 千兆位以太网技术的发展:提高网络实时性最有效的方法是提高网络带宽,网络速度在过去的十多年里提高了三个数量级。千兆位以太网技术及网关、交换机、路由器、嵌入式以太网技术的发展使高速以太网可以满足仪器测量的需要。

⑥ 面向信号的 IVI-COM 技术:IVI-MSS (Measure & Stimulus Subsystem) 方案是其中的一种,该方案的关键是在位于测试应用层和仪器层中间的 MSS 测量和激励子系统层。MSS 包括 MSS 服务器和角色控制组件。通过 IVI-Signal 接口标准可用一系列方法执行信号操作,如初始化、建立、更改等。IVI 信号组件控制一台或多台仪器产生客户需要的信号,完成客户的测试需要。它对仪器的控制是通过 VISA、IVI 驱动器、SCPI 命令等实现的。IVI-Signal 接口通过 COM 组件的方式来实现,每一个信号组件都可以用带有一系列方法和属性的 COM 组件来描述。

16.4 串行总线

16.4.1 串行总线简介

串口是一种非常通用的设备通信协议,在绝大多数的计算机上都含有 1 个以上的基于 RS232 串口 (COM 接口)。虽然在笔记本电脑上串口已经被 USB 取代,但大多数仪器仪表仍采用串口进行通信,很多 GPIB 设备也兼有 RS-232 串口。此外,串口通信协议也可用于获取远程采集设备的数据。

提示

串口总线不要与通用串行总线 USB 混淆。

(1) 串口信号结构

串口按位 (bit) 发送和接收字节。尽管比按字节 (byte) 的并行通信速度稍慢,但是串



口采用异步通信方式，所以利用 1 根发送数据线、1 根接收数据线和 1 根地线，就能在同一时间完成数据的双向传输。传输的数据一般使用 ASCII 码形式。图 16-18 为串口的连线示意图，表 16-3 给出了串口各引脚功能。

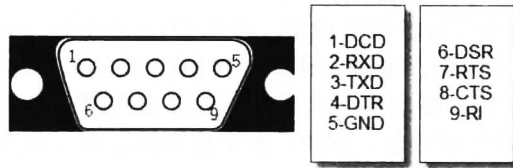


图 16-18 串口连线示意图

表 16-3 串口各引脚功能

名称	功能	引脚	www.91HMI.com 类型
TXD	串口数据输出	3	数据信号
RXD	串口数据输入	2	数据信号
GND	地线	5	地线
RTS	发送数据请求	7	握手信号
CTS	清除发送	8	握手信号
DSR	数据发送就绪	6	握手信号
DCD	数据载波检测	1	握手信号
DTR	数据终端就绪	4	握手信号
RI	铃声提示	5	其他

串口容易实现长距离通信。比如 IEEE488 协议规定 GPIB 设备总线不得超过 20 米，并且任意 2 台设备间的线缆长度不得超过 2m。相比之下，串口的总线长度可以达到 1200 米。

(2) 串口机械结构

串口的连接器多数选用 DB-9 针（公头）或 DB-9 孔（母头），少数 PLC 设备使用 PS2 连接器。一般计算机提供的 RS-232 串口为 DB-9 针。图 16-19 分别是 DB-9 孔和 DB-9 针的实物图。

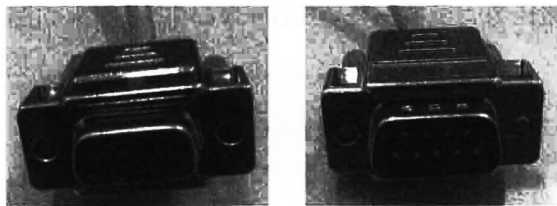


图 16-19 串口实物图

(3) 基本参数

- ① 波特率：表征通信速度的参数，为每秒钟传送 bit 的个数。与通信距离成反比。
- ② 数据位：表征通信中实际数据位的参数。取决于通信协议的选择。



- ③ 停止位：表征信息包中的最后一位。
 - ④ 奇偶校检位：一种简单的检错方式，可以设置校检位，也可以不设置。
- (4) 串口的分类

串行接口按电气标准及协议来分，包括 RS-232-C、RS-422、RS485 等。RS-232-C、RS-422 与 RS-485 标准只对接口的电气特性做出规定，并不涉及接插件、电缆或协议。

① RS-232C：也称标准串口，是最常用的一种串行通信接口。在 1970 年由美国电子工业协会（EIA）联合贝尔系统、调制解调器厂家及计算机终端生产厂家共同制定，是串行通信的标准。其全称是“数据终端设备（DTE）和数据通信设备（DCE）间串行二进制数据交换接口技术标准”。传统的 RS-232-C 接口标准有 22 根线，采用标准的 25 芯 D 型插头座（DB25），后来多使用简化的 9 芯 D 型插座（DB9），现在 25 芯的插头座已很少采用。

RS-232 采取不平衡传输方式，即所谓单端通信。由于其发送电平与接收电平的差值仅为 2V 至 3V 左右，所以其共模抑制能力弱，再加上双绞线上的分布电容，其最大传送距离约为 15 米，最高速率为 20Kb/s。RS-232 是为点对点（即只用一对收、发设备）通信而设计的，其驱动器负载为 3~7kΩ。所以 RS-232 适合本地设备之间的通信。

② RS-422：标准全称是“平衡电压数字接口电路的电气特性”，它定义了接口电路的特性。典型的 RS-422 是四线接口。实际上还有一根信号地线，共 5 根线。由于接收器采用高输入阻抗和发送驱动器比 RS232 更强的驱动能力，故允许在相同传输线上连接多个接收节点，最多可接 10 个节点。即一个主设备（Master），其余为从设备（Slave），从设备之间不能通信，所以 RS-422 支持点对多的双向通信。接收器输入阻抗为 4k，故发端最大负载能力是 $(10 \times 4)k + 100\Omega$ （终接电阻）。RS-422 四线接口由于采用单独的发送和接收通道，因此不必控制数据方向，各装置之间任何的信号交换均可以软件方式（XON/XOFF 握手）或硬件方式（一对单独的双绞线）来实现。

RS-422 的最大传输距离为 1219m，最大传输速率为 10Mb/s。其平衡双绞线的长度与传输速率成反比，在 100Kb/s 速率以下，才可能达到最大传输距离。只有在很短的距离下才能获得最高速率传输。一般 100m 长的双绞线上所能获得的最大传输速率仅为 1Mb/s。

③ RS-485：是基于 RS-422 发展而来，所以 RS-485 的许多电气规定与 RS-422 相仿。如都采用平衡传输方式、都需要在传输线上接终接电阻等。RS-485 可以采用二线与四线方式，二线制可实现真正的多点双向通信，而采用四线连接时，与 RS-422 一样只能实现点对多的通信，即只能有一个主（Master）设备，其余为从设备，但它比 RS-422 有改进，无论四线还是二线连接方式总线上均可接至 32 个设备。

16.4.2 串口子面板

进入串口子面板的路径为“函数→仪器 I/O→串口”，其面板如图 16-20 所示，共有 8 个串口函数。

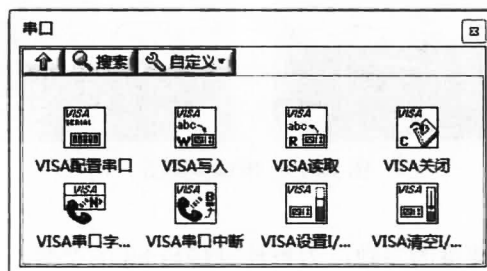


图 16-20 串口子面板



对于初学者，只要掌握下面 4 个函数就可对串口设备进行开发。

VISA 配置串口：设置串口通信的基本参数，如波特率、数据位、校检位和停止位等。

VISA 写入：写入指令或数据到指定的串口设备。

VISA 读取：从指定的串口设备读取数据。

VISA 关闭：关闭与指定设备的对话。

(1) 配置串口

执行“函数→仪器 I/O→串口→VISA 配置串口.vi”，即可配置串口，该函数如图 16-21 所示。

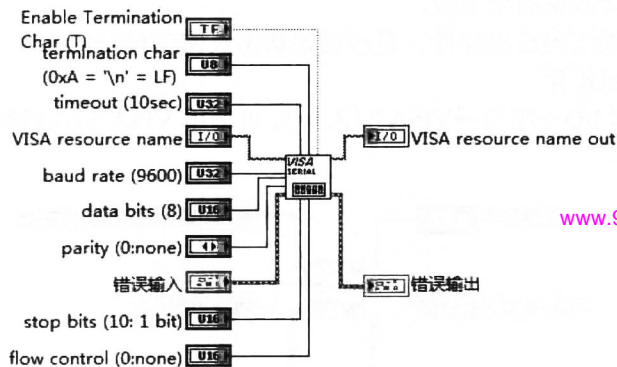


图 16-21 VISA 配置串口函数

① 其输入控件如下。

VISA resource name：根据 VISA 语法写入串口设备的地址。

baud rate：波特率设置。默认值为 9600。

data bits：数据位设置，数值范围 5~8。默认值为 8。

parity：奇偶校检位设置。默认值为 0，无校检。

stop bits：停止位设置。默认值为 10，表示 1 个停止位。

Enable Termination：启用终止符。使设备做好识别终止符的准备。默认值为真。

termination char：终止符。通过调用终止读取操作。

flow control：流控制。设置传输机制使用的控制类型。默认值为 0。

错误输入：错误代码输入控件。可接受前一级 vi 的错误代码输入。

② 其输出控件如下。

VISA resource name out：输出串口设备的地址字符串，可供下一级 vi 使用。

错误输出：错误代码输出控件。显示错误状态及错误代码。

(2) 对串口进行读操作

执行“函数→仪器 I/O→串口→VISA 读取.vi”，可调用 VISA 读取函数，该函数如图 16-22 所示。

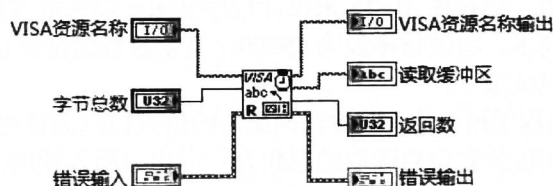


图 16-22 VISA 读取函数



① 其输入控件如下。

VISA 资源名称：根据 VISA 语法写入串口设备的地址。

字节总数：需要从串口设备读取数据的字节总数。

错误输入：错误代码输入控件。可接受前一级 vi 的错误代码输入。

② 其输出控件如下。

VISA 资源名称输出：输出串口设备的地址字符串，可供下一级 vi 使用。

读取缓冲区：根据指定的字节总数，读取串口缓冲区的字符串数据。

返回数：实际读取数据的字节数。

错误输出：错误代码输出控件。显示错误状态及错误代码。

(3) 对串口进行写操作

执行“函数→仪器 I/O→串口→VISA 写入.vi”，可调用 VISA 写入函数，该函数如图 16-23 所示。

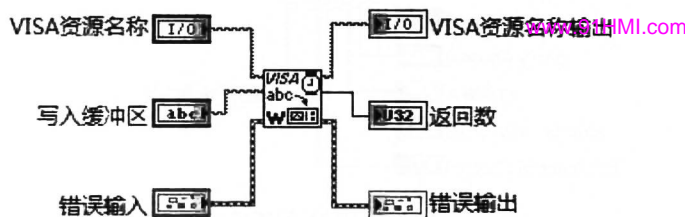


图 16-23 VISA 写入函数

① 其输入控件如下。

VISA 资源名称：根据 VISA 语法写入串口设备的地址。

写入缓冲区：向串口发送的指令或数据。

错误输入：错误代码输入控件。可接受前一级 vi 的错误代码输入。

② 其输出控件如下。

VISA 资源名称输出：输出串口设备的地址字符串，可供下一级 vi 使用。

返回数：实际读取数据的字节数。

错误输出：错误代码输出控件。显示错误状态及错误代码。

16.4.3 利用串口总线控制伺服驱动器

Emol 伺服驱动器是一种可以带动电机进行提升、下降动作的装置，它提供有 RS232 接口功能，用户可由串口总线来实现控制任务。图 16-24 是采用 LabVIEW 为其设计的前面板，通过它可以设定电机的工作方式、运行速度等，并可显示提升杆位置、实现限位报警功能等。

为了实现前面板的设计功能，需解决以下几个关键问题。

(1) 串口基本参数设置

要设置串口基本参数，需按图 16-25 来设计程序框图，以选取 VISA 资源名称。此外，根据伺服器应用手册的要求，将波特率设为 19200，其余参数保持默认值。

(2) 提升杆位置信息读取

根据 Emol 伺服器编程手册要求，要读取伺服器的位置信息需使用“PX;”指令，图 16-26 是该功能的程序框图。实现位置信息读取的思想是：向串口写入读取命令以读取信息，然后由串口接收返回的字符串。由于在接收的字符串中含有其他信息，因此要用 LabVIEW 的字符串截取模块来提取位置信息，并送入字符串显示控件来显示。

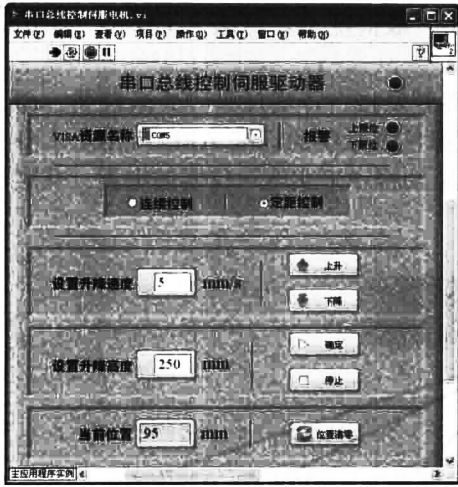


图 16-24 串口总线控制伺服驱动器前面板

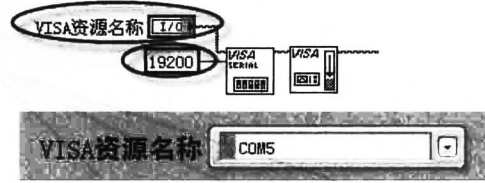


图 16-25 串口基本参数设置程序框图

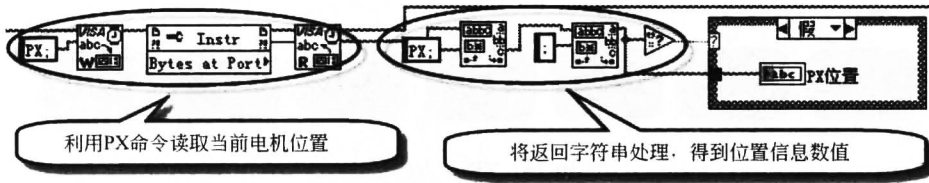


图 16-26 提升杆位置信息读取程序框图

(3) 提升杆定距控制

提升杆定距控制涉及位置与速度的设定及电机的开启，图 16-27 是该功能的程序框图。

实现定距控制的思想是：向伺服器发送控制指令，并在指令间含有位置和速度等参数。

由于提升杆的手动控制与定距控制基本相同，故此不再详述。

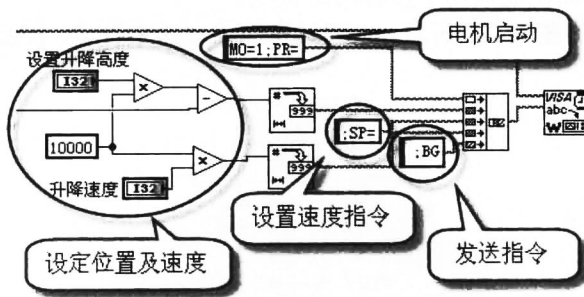


图 16-27 提升杆定距控制程序框图

16.5 VISA 的使用

16.5.1 VISA 简介

VISA 是 Virtual Instrument Software Architecture 的缩写，即虚拟仪器软件体系结构，



其实质就是一个 I/O 接口软件库机器规范的总称。在一般情况下，将这个 I/O 接口库称为 VISA 库。

API (Application Program Interface) 是应用软件开发接口，实质是一组函数集，通过它可以直接访问计算机的硬件设备。VISA 是用于虚拟仪器系统的标准 API。VISA 本身并不具备编程能力，它是一个高层 API，通过调用底层驱动程序来实现对仪器的编程。其层次图如图 16-28 所示。

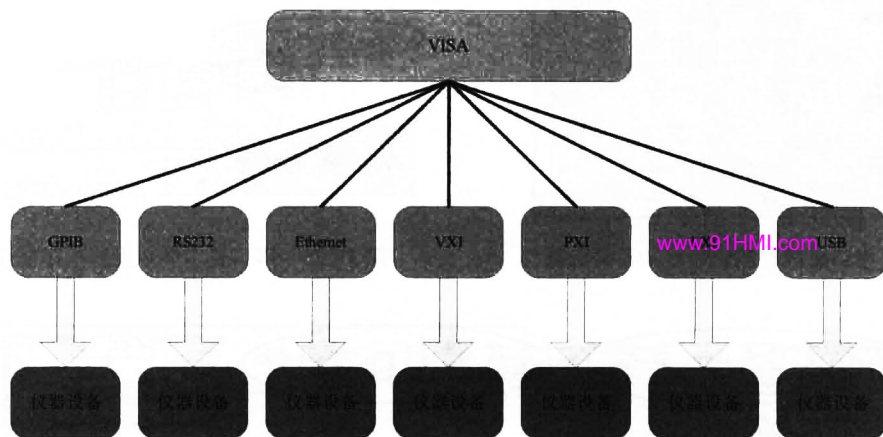


图 16-28 VISA 内部层次

与其他 I/O 接口软件相比，VISA 有以下三个特点。

① VISA 的 I/O 控制功能适用于各种仪器类型，既包含了 VXI 仪器、GPIB 仪器及串口仪器等各类仪器的控制操作，也包含了消息基器件、寄存器器件、存储器器件等仪器的操作，具有形式上的统一。

② VISA 的 I/O 控制功能适用于各种仪器硬件接口类型。

③ VISA 的 I/O 控制功能适用于多种网络机制。

VISA 的内部结构如图 16-29 所示，它通过资源管理器来管理、控制与分配 VISA 资源的



图 16-29 VISA 内部结构示意图



操作, 其各类操作包括: 资源寻址、资源创建与删除、资源属性的读取与修改、操作激活、事件报告、并行与存取控制、默认值设置等。

在 VISA 的结构中, 仪器类型的不同体现在资源名称的不同, 对 VISA 的使用者来说, 不同类型仪器的使用在形式上和方法上都是一样的。

16.5.2 VISA 子面板

进入 GPIB 子面板的路径为“函数→仪器 I/O→VISA”, 其面板如图 16-30 所示。

在 VISA 子面板中, 有 5 个 VISA 基本函数和 1 个 VISA 高级函数类, 掌握其中的几个基本函数就可进行开发。

在基本函数中, VISA 写入函数、VISA 读取函数都与串口的相应函数相同, 此处不再赘述。其他的函数功能如下:

VISA 设备清零: 对设备的输入和输出缓冲区进行清零。

VISA 读取 STB: 从 VISA 资源名称指定的基于消息的设备中读取服务请求状态字节。

VISA 打开: 打开 VISA 资源名称指定设备的会话句柄并返回会话句柄标识符, 该标识符可用于调用该设备的其他操作。

VISA 关闭: 关闭 VISA 资源名称指定的设备会话句柄或事件对象。

VISA 查找资源: 查询系统, 定位与指定接口相关的设备。

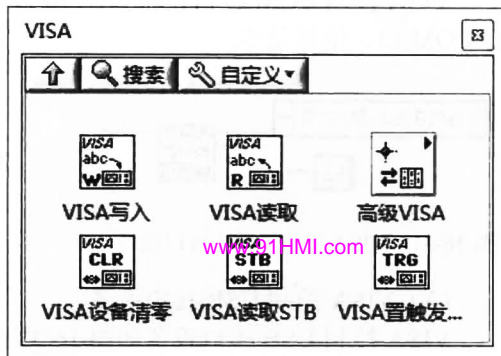


图 16-30 VISA 子面板

提示

VISA 查找资源函数可以用来列举出当前计算机所连接的设备及 VISA 资源名称, 这为提高程序的自动化程度提供了可能。

16.5.3 在 LabVIEW 中使用 VISA 控制各种接口设备

使用 VISA 最方便的地方在于能以统一的形式控制各种接口设备, 通过 VISA 资源字符串, 即可通知 VISA 仪器地址信息, 表 16-4 列出了 VISA 资源字符串语法表。

表 16-4 VISA 资源字符串语法表

物理接口	VISA 语法
GPIB	GPIB[板卡]::主地址[::GPIB 次地址][::INSTR]
VXI	VXI[板卡]::VXI 逻辑地址[::INSTR]
PXI	PXI[总线]::设备[::函数][::INSTR]
串口	ASRL[板卡][::INSTR]
TCP-IP	TCPIP[板卡]::主机地址[::LAN 设备名][::INSTR]
USB	USB[板卡]::制造商 ID::型号编码::序列号[::USB 接口号][::INSTR]



在仪器仪表方面，最常出现的接口有 4 种，分别是 GPIB 接口、串行（COM）接口、USB 接口和 LAN 接口，它们几乎涵盖了大部分的智能仪器仪表。

(1) VISA 控制 GPIB 接口设备

VISA 控制 GPIB 接口设备如图 16-31 所示，该 VISA 地址的含义是：该 GPIB 仪器设备位于 GPIB 接口 0，主地址为 4。

(2) VISA 控制 COM 接口设备

VISA 控制 COM 接口设备如图 16-32 所示，该 VISA 地址的含义是：该仪器设备位于串行 COM 口，地址是 5。



图 16-31 VISA 控制 GPIB 接口设备



图 16-32 VISA 控制 COM 接口设备

(3) VISA 控制 USB 接口设备

VISA 控制 USB 接口设备如图 16-33 所示，该 VISA 地址的含义是：该仪器设备位于 USB 板卡 0 上，其制造商 ID 为 0x0957（Agilent），型号编码为 0x17A4，仪器的序列号为 MY51135727。

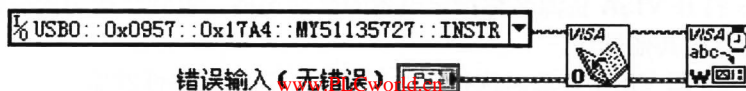


图 16-33 VISA 控制 USB 接口设备

(4) VISA 控制 LAN 接口设备

VISA 控制 LAN 接口设备如图 16-34 所示，该 VISA 地址的含义是：访问 IP 地址为 169.254.45.32 的仪器设备，并使用 inst0 的默认 LAN 设备名。

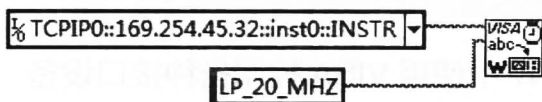


图 16-34 VISA 控制 LAN 接口设备

16.6 用 NI MAX 查找仪器

16.6.1 NI MAX 简介

MAX 也是 NI 公司的设备管理器，类似于 Windows 的设备管理器，是 NI 开发的一款帮助用户快速掌握和使用其产品的软件。它功能强大，可以配置 NI 硬件和软件，备份或复制配置数据，创建和编辑通道、任务、接口、换算和虚拟仪器，若进行系统诊断，还可以查看与系统连接的设备和



图 16-35 MAX 桌面快捷方式



仪器。依照用户安装的 NI 产品，MAX 可提供某些特定工具，以对系统进行配置、诊断和测试。一般情况下，安装好 LabVIEW 就会在桌面上生成 MAX 的快捷方式，如图 16-35 所示。

16.6.2 NI MAX 软件操作界面

双击 MAX 图标，即可出现图 16-36 所示的程序界面。

MAX 的主要功能有 6 类，由左侧的配置栏列出，即数据邻居、设备与接口、换算、软件、IVI Drivers 和远程系统。

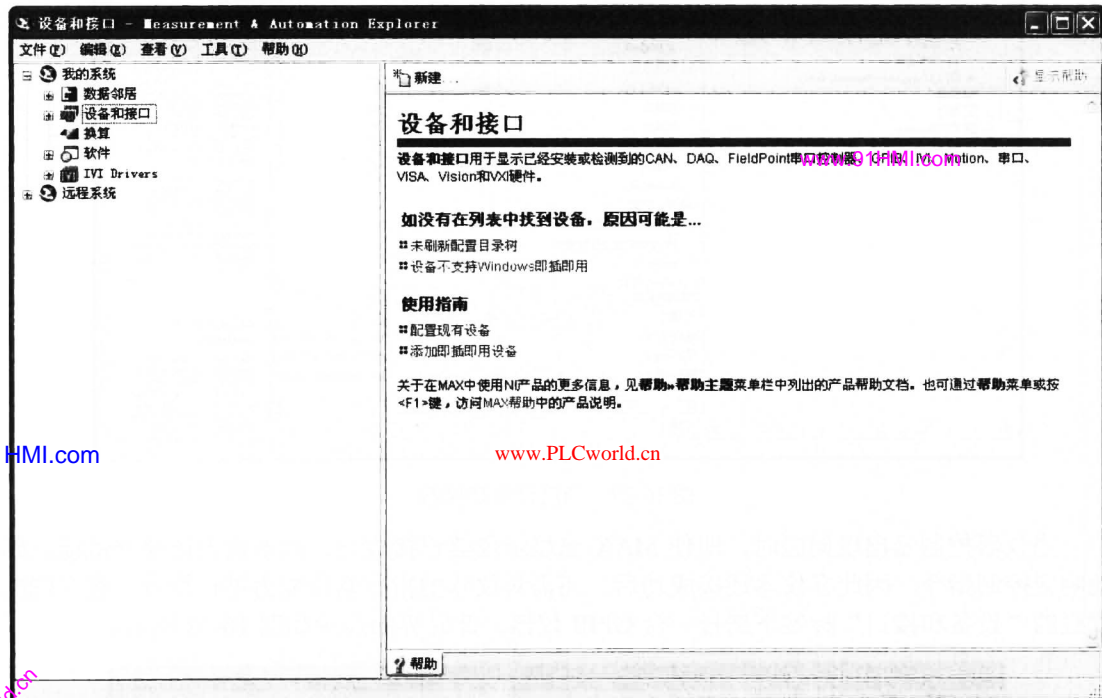


图 16-36 Measurement & Automation Explorer 主界面

① 数据邻居是通过带有描述性名称的快捷方式来配置系统的物理通道，其快捷方式包括 CAN 信息、NI-DAQ 虚拟通道和任务，以及 FieldPoint 项。

② 设备和接口用于显示已经安装或检测到的 CAN、DAQ、FieldPoint 串口控制器、GPIB、IVI、Motion、串口、VISA、Vision 和 VXI 硬件。

③ 换算包含所有已配置的换算，并对采集到的数据进行运算。

④ 软件用于查看、运行和更新已经安装的 NI 软件。

⑤ IVI 的意思是可互换虚拟仪器 (Interchangeable Virtual Instrument)，是一种新的驱动程序架构。IVI Drivers 用于配置 IVI 仪器或 IVI 驱动程序。

⑥ 远程系统是通过网络管理和配置的实时终端，与不运行实时操作系统的网络设备有所区别。远程设备不同于网络设备。网络设备是通过以太网或无线连接的设备。网络设备不运行实时操作系统，可同时被多台计算机访问。通过远程系统可查看和配置连接在以太网上的设备和系统。



16.6.3 在 NI MAX 中查找仪器

在连接好仪器后，打开 MAX，并点击左侧配置栏中“设备和接口”项，就可查找到当前已安装的仪器，如图 16-37 所示。由图可见已有 3 台 GPIB 设备和 1 台 USB 设备被连接，若未经 MAX 自动刷新，可点击“扫描仪器”来进行即插即用仪器的查找与连接。



图 16-37 当前已安装仪器

当仪器控制器出现问题时，即使 MAX 已显示仪器连接成功，也不代表该仪器就能正确地响应控制指令。因此在仪器连接成功后，还需对仪器的指令响应能力进行检查。在左侧配置栏的“设备和接口”标签下选择一台 GPIB 仪器，此时界面显示如图 16-38 所示。

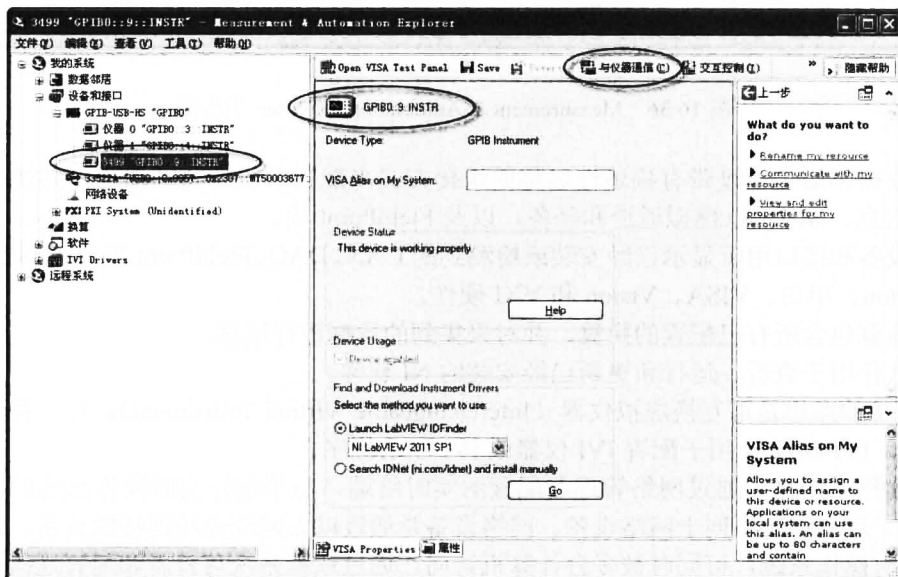


图 16-38 某台仪器的详细信息



使用“*IDN?”指令，可测试仪器对指令的响应能力，点击图 16-38 的“与仪器通信”，会出现图 16-39 所示的“NI-488.2 通讯器”界面，点击“写(W)”发送指令，再点击“读(R)”接收返回字符串，则显示在界面下方。按返回的字符串信息，说明该设备是安捷伦公司生产的型号为 3499 的仪器设备，设备唯一编码为“MY42001890”，版本号是 4.2。

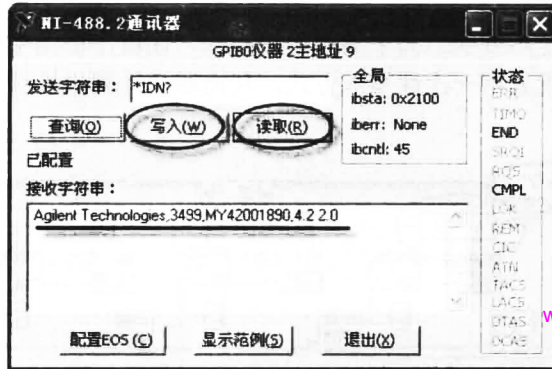


图 16-39 NI-488.2 通讯器界面

16.7 综合实例：用 GPIB 总线控制可编程滤波器

双通道可编程滤波器 3627 是日本 NF 公司的产品，具有 2 个滤波通道，支持多种滤波模式。其频带范围为 1HZ~1.59MHZ，通道增益可达 25 倍。由于 NF3627 具备 GPIB 接口，因此在 LabVIEW 环境下，可利用 GPIB 总线实现上位机对它的编程控制。

在功能设计方面，是要实现两台可编程滤波器 NF3627 的控制，具体涉及的有：多种滤波方式选择、增益及通道使用方式切换等。

根据使用需求，该可编程滤波器的远程控制界面如图 16-40 所示。

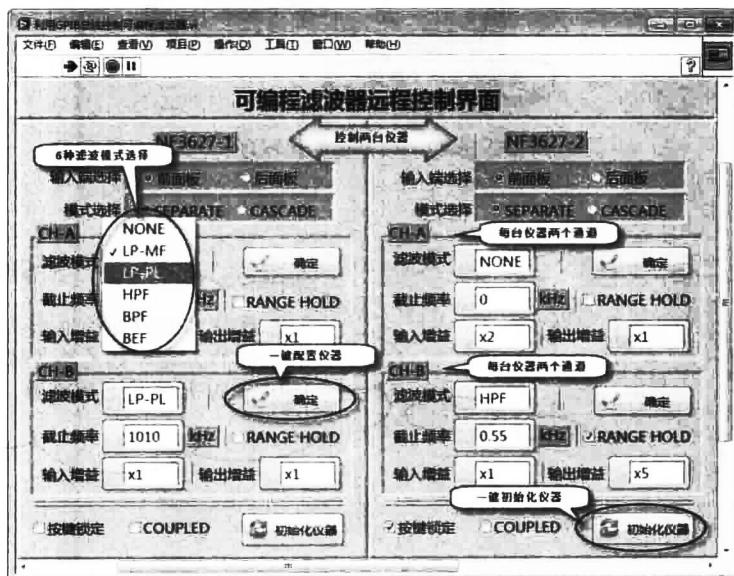


图 16-40 可编程滤波器远程控制界面



为了能顺利地完成任务，事先应仔细阅读仪器编程手册，并从中获取具体的功能指令，然后再用 MAX 查询来确定所用仪器的正确地址。

在实现仪器连接的情况下，就可进行控制程序的开发了，图 16-41 是一台可编程滤波器的 CH-A 控制模块（CH-B 控制模块仿此），它集成了频率修改、滤波模式选择、增益选择及单位锁定功能。

由于仪器地址是固定不变的，为了简化用户操作，在程序框图中用常量设置。此外，基于可靠性考虑，参数设置采用一次性写入方式，从而避免计算机连续与 GPIB 仪器通信而产生的异常情况。

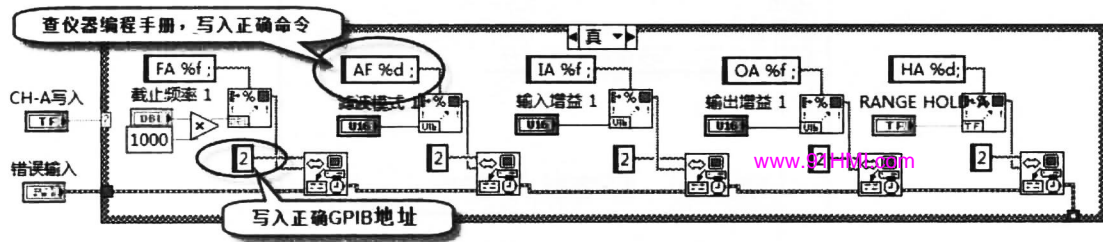


图 16-41 CH-A 模块控制程序

仪器的初始化命令可以将仪器置于初始状态，由于远程控制的初始化指令不能重置“仪器模式选择”和“COUPLED”两个状态，所以在编程时若要达到所有参数初始化的效果，需要单独将两个参数重置，图 16-42 为仪器初始化程序。

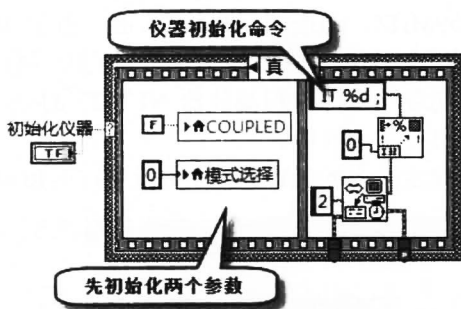


图 16-42 仪器初始化程序



总线技术不但是构造虚拟仪表系统的关键，也是单片机与 LabVIEW 传递信息的桥梁。



第 17 章

LabVIEW数据库的访问

在测量和控制领域，常常会面对大量的测控数据，因而如何高效地管理这些数据是个重要的任务。采用数据库技术来进行管理，已在实际应用中证明是可行的且是有效的。它可以方便地实现数据的保存、修改、删除和查询等关键的操作。

【本章导读】

- ◆ 数据库连接工具包及其使用
- ◆ 综合实例

数据库是一种基于表格的数据管理技术，在每个数据库中，允许有多个数据表存在，而每个数据表又是由行和列构成。数据表的一行被称为一条记录，它的一列被称为一个字段。在众多的商用数据库管理系统中，常用有 Oracle、Visual Foxpro、MS SQL Server 和 Access 等。

17.1 数据库连接工具包及其使用

17.1.1 数据库连接工具包

LabVIEW 数据库连接工具包是一种基于 ODBC 技术的应用工具，它涵盖多个数据库操作的函数，具体内容如图 17-1 所示。利用该工具包，可以轻松地实现数据库表格的创建、删除、添加记录、查询记录等操作，并可执行一些 SQL 语句。

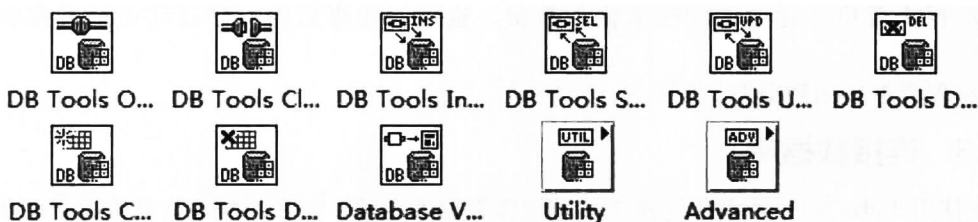


图 17-1 数据库连接工具函数

使用数据库连接工具之前，必须事先安装好该工具，否则在函数模板下是没有其子模板的。该子模板位于“函数→互联接口→Database”下。



17.1.2 建立 Access 数据库

在使用数据库连接工具包之前，必须先建立数据源。创建数据库的工具活动，Access 就是其中的一种。Access 从属于 Microsoft Office 办公组件，用它可以创建基于关系型的数据库。由于 Office 办公软件十分常见，因此选取 Access 构建数据库具有广泛的普遍性。

下面以一个采集数据管理库为例来介绍其创建过程，该数据库含有 2 个数据表，其中数据表 A 涉及实验日期、材料、转速、时长和实验名称；数据表 B 涉及温度、压力和扭矩。

具体的创建过程如下。

- ① 打开 Access 软件，创建一个名为 sygl 的数据库。
- ② 在 sygl 数据库下新建 2 个数据表，分别为“sjba”和“sjbb”，其数据库构成如图 17-2 所示。

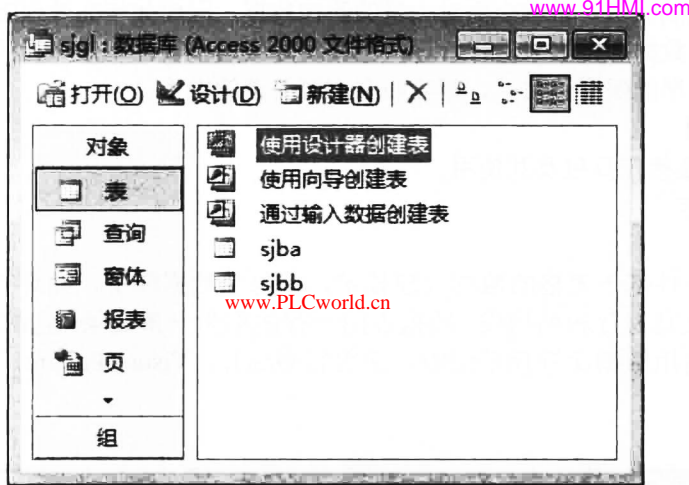


图 17-2 Access 数据库结构

③ 打开数据表 sjba，并创建实验日期、材料、转速、时长和实验名称等字段，将其数据类型设为“文本”，字段大小设为“50”。

④ 打开数据表 sjbb，并创建温度、压力和扭矩字段，将其数据类型设为“数字”，字段大小设为“单精度型”，格式设为“常规数字”，小数位数设为“4”。完成后的数据库结构分别如图 17-3 和图 17-4 所示。

⑤ 视需要可在建库的同时来输入数据，也可以在事后由程序自动记录或读取采集数据。

⑥ 保存并关闭数据库。

17.1.3 连接数据库

在使用 LabVIEW 数据库连接工具包操作数据库之前，需要事先连接数据库，就如同对操作文件一样，在使用前必须先打开该文件，否则无法执行操作。

连接数据库主要有两种方式，即使用 DSN 连接数据库和使用 udl 文件来连接数据库。

(1) 使用 DSN 连接数据库

在具有数据库的情况下，创建 DSN 的方法如下。

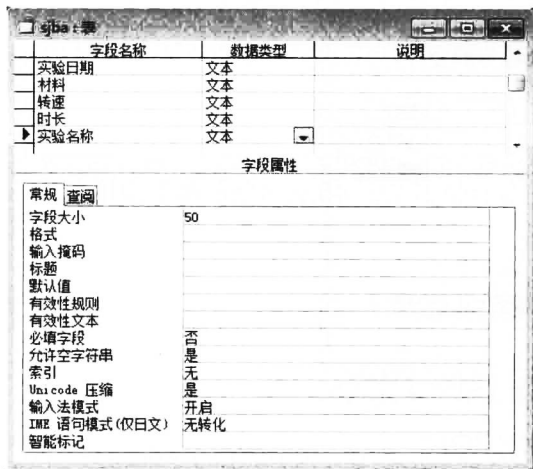


图 17-3 创建数据表 sjba

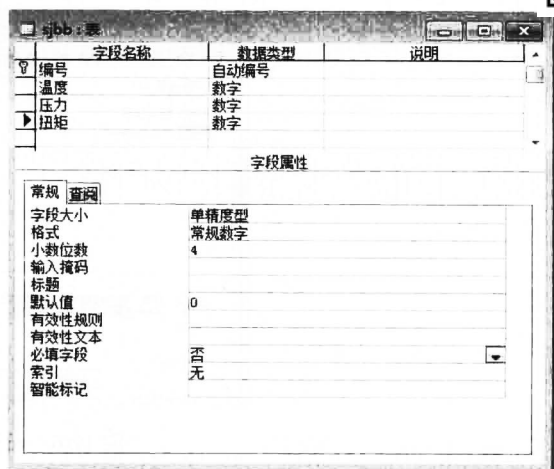


图 17-4 创建数据表 sjbb

① 在 Windows 的开始菜单，选取“控制面板→所有控制面板项→管理工具→数据源 (ODBC)”来打开数据源管理器，其界面如图 17-5 所示。

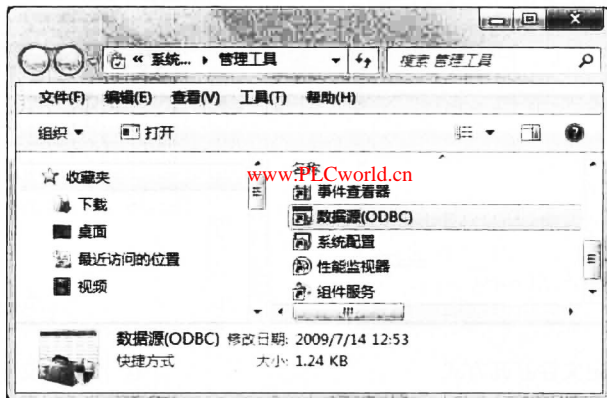


图 17-5 打开数据源管理器

② 选取“用户 DSN”选项卡，单击“添加”按钮，在弹出的对话框中选取“SQL Server”启动并单击“完成”。

③ 在“创建到 SQL Server 的新数据源”对话框，由“名称”文本框输入数据源的名称，若单击“完成”，则进入图 17-6 所示界面，可单击“测试数据源”进行测试，若配置成功，会弹出测试成功的对话框。

④ 连续单击“确定”按钮，直至退出设置。

(2) 使用 udl 文件连接数据库

Microsoft 制定的 ODBC 标准只适用于关系型数据库，为了支持非关系型数据库，Microsoft 又提供了另一种称为 ADO (Active Data Objects) 的技术。ADO 与数据库的连接是通过 UDL (通用数据连接)，该 UDL 则是由 udl 文件来实现的。

创建并配置 udl 文件的方法如下。

① 使用记事本建立一个文本文件，然后以 udl 为后缀加以保存。

② 右击此文件，由快捷菜单选取“打开方法”。则弹出的对话框如图 17-7 所示。从中选取“OLE DB Core Services”并单击“确定”。

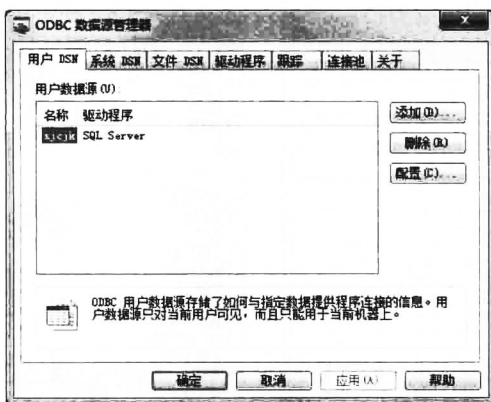


图 17-6 设置用户 DSN

- ③ 在对话框中，按图 17-8 所示选取“Microsoft Jet 4.0 OLE DB Provider”，然后单击“下一步”。

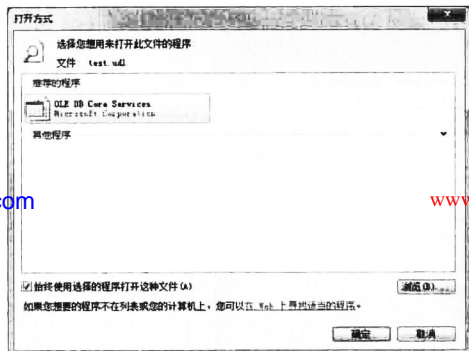


图 17-7 指定文件打开方式

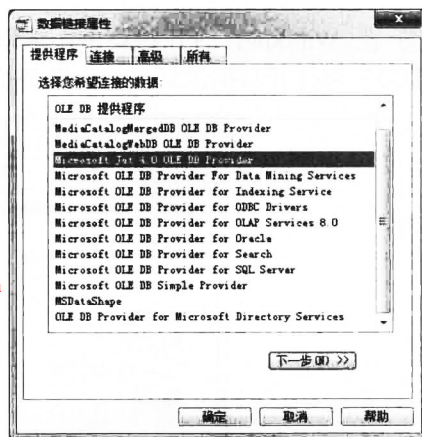


图 17-8 设置驱动程序

- ④ 在“连接”选项卡，按图 17-9 所示选取相应的连接数据库。
⑤ 单击“测试连接”按钮，可验证连接是否成功。若成功，则如图 17-10 所示。

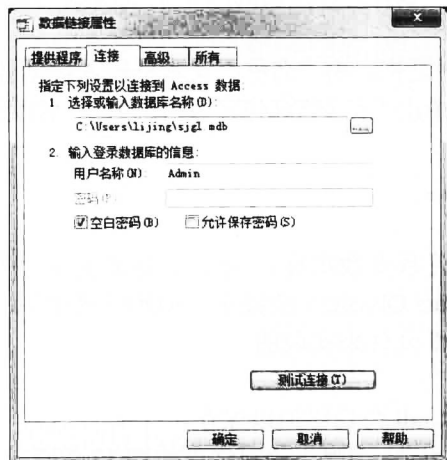


图 17-9 设置连接数据库

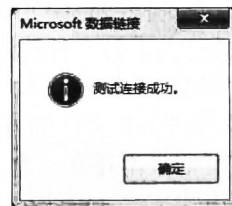


图 17-10 测试成功界面



⑥ 单击“确定”按钮，完成配置处理。

17.1.4 常规数据库操作

常规的数据库操作包括创建表格、删除表格、添加记录及查询记录等，为了便于介绍这里统一采用 udl 文件来连接数据库，数据库则是基于 ACCESS 数据库。

(1) 创建数据库表格

添加数据记录是在打开数据库连接的前提下，通过指定表格名称及表格结构来实现的，具体的操作步骤如下。

① 在后面板，选取“函数→互联接口→Database→DB Tools Open Connection.vi”来添加连接数据库函数。

② 再依次添加“DB Tools Create Table.vi”和“DB Tools Close Connection.vi”图标，并为相应的端口创建输入控件，经连线其程序框图和显示效果分别如图 17-11 和图 17-12 所示。

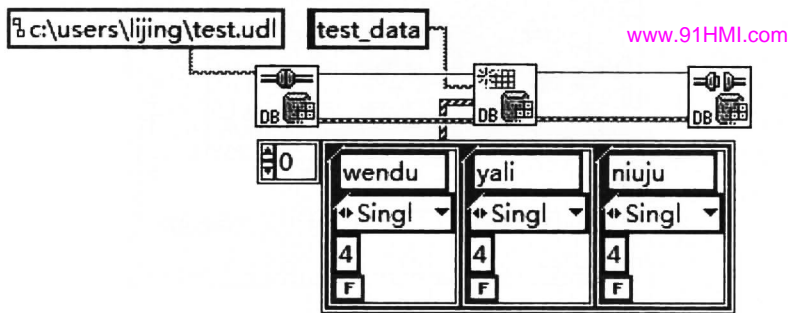


图 17-11 创建数据库表格程序框图

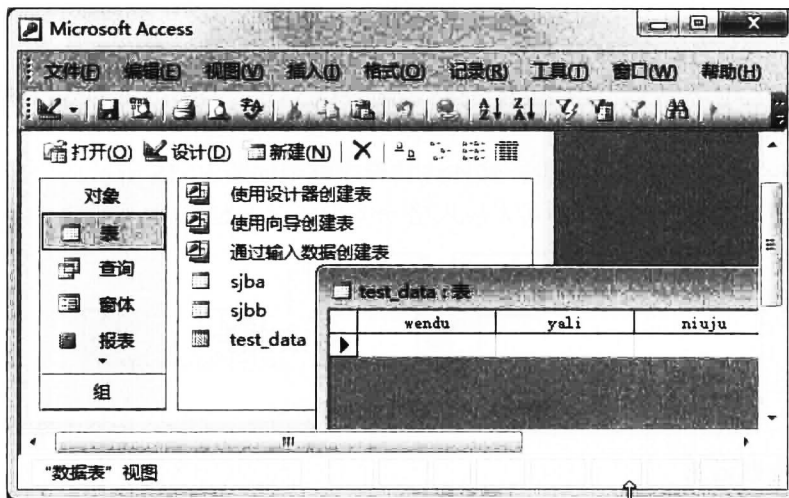


图 17-12 新建数据库表格效果

(2) 添加数据记录

添加数据记录是在打开数据库连接的前提下，通过指定表格名称及提供插入记录值来实现的，具体的操作步骤如下。

① 在后面板，放置一个“DB Tools Open Connection.vi”图标和一个“DB Tools Close



Connection.vi”图标。

② 选取“函数→互联接口→Database→Utility→DB Tools List Columns .vi”来添加该函数，再添加一个“DB Tools Insert Data .vi”图标。为相应的端口创建输入控件，经连线其程序框图如图 17-13 所示。

运行该程序，可发现新的记录数据已被插入表格，其结果如图 17-14 所示。

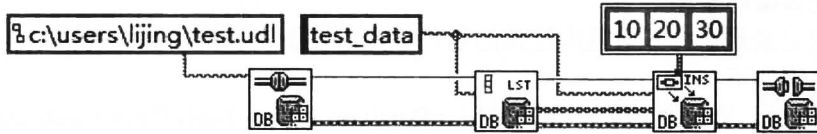


图 17-13 添加数据记录程序框图

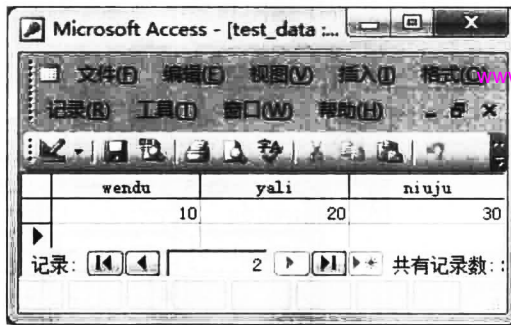


图 17-14 添加数据记录结果

(3) 删除数据库表格

删除数据库记录是在打开数据库连接的前提下，通过指定表格名称来实现的，具体的操作步骤如下。

① 在后面板，依次放置“DB Tools Open Connection.vi”、“DB Tools Drop Table.vi”及“DB Tools Close Connection.vi”。

② 为相应的端口创建输入控件，经连线其程序框图如图 17-15 所示。

运行该程序，在打开数据库后可发现表格 test_data 已被删除。

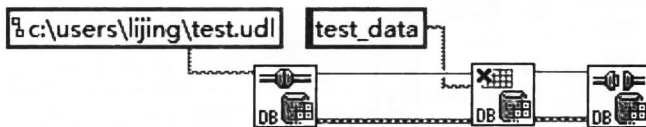


图 17-15 删除数据库表格

(4) 查询数据记录

查询数据记录是在打开数据库连接的前提下，通过指定表格名称、显示字段名或查询添加来实现。

按字段进行查询，其操作步骤如下。

① 在后面板，依次放置“DB Tools Open Connection.vi”、“DB Tools Select Data.vi”及“DB Tools Close Connection.vi”。

② 为相应的端口创建输入、输出控件，经连线其程序框图和显示效果如图 17-16 所示。

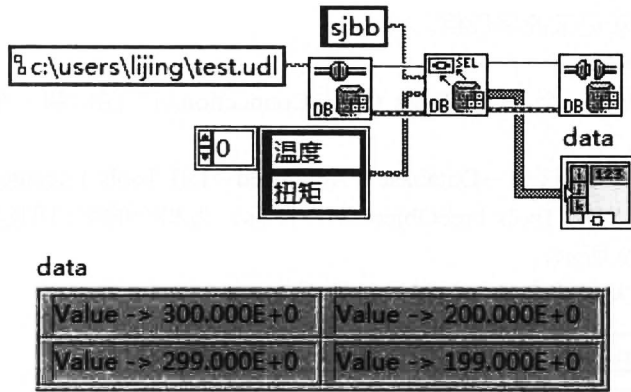


图 17-16 字段查询

按条件进行查询，其操作步骤如下。

- ① 在后面板，依次放置“DB Tools Open Connection.vi”、“DB Tools Select Data.vi”及“DB Tools Close Connection.vi”。
- ② 为相应的端口创建输入、输出控件，经连线其程序框图和显示效果分别如图 17-17 所示。

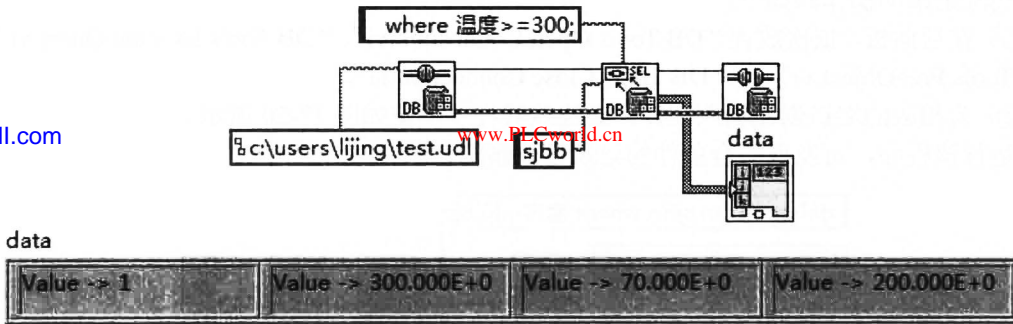


图 17-17 条件查询

17.1.5 SQL 语句的使用

除了表格的创建、删除、查询操作外，记录的删除、更新等也是常用的操作，对于此类操作需借助 LabVIEW 数据库连接工具中的高级操作，其具体内容如图 17-18 所示。

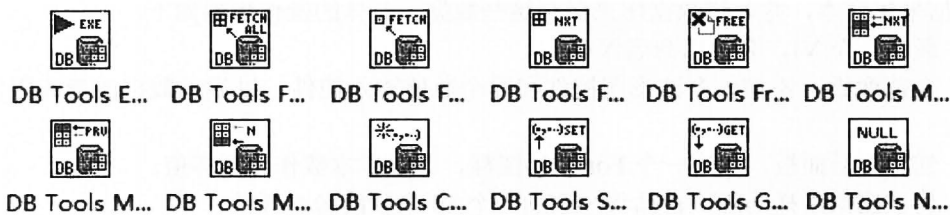


图 17-18 高级操作函数

(1) 更改数据记录

更改数据记录是在打开数据库连接的前提下，通过指定的条件查找出符合要求的记录，

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



然后按给定操作来更新记录的字段值。

更新记录的操作步骤如下。

① 在后面板，放置一个“DB Tools Open Connection.vi”图标和一个“DB Tools Close Connection.vi”图标。

② 选取“函数→互连接口→Database→Advanced→DB Tools Execute Query.vi”来添加该函数，再添加一个“DB Tools FreeObject.vi”图标。为相应的端口创建输入控件，经连线其程序框图如图 17-19 所示。

运行该程序，可发现符合条件的记录已被更改。

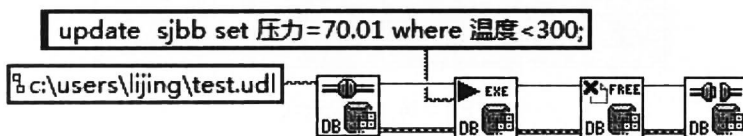


图 17-19 更改数据记录

www.91HMI.com

(2) 删除数据记录

删除数据记录是在打开数据库连接的前提下，通过指定的条件查找出符合要求的记录，然后予以删除。

删除记录的操作步骤如下。

① 在后面板，依次放置“DB Tools Open Connection.vi”、“DB Tools Execute Query.vi”、“DB Tools FreeObject.vi”及“DB Tools Close Connection.vi”。

② 为相应的端口创建输入控件，经连线其程序框图如图 17-20 所示。

运行该程序，可发现符合条件的记录已被删除。

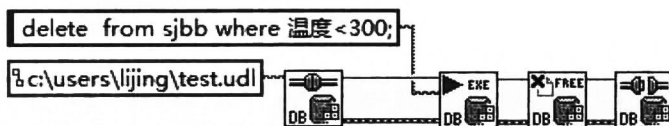


图 17-20 删除数据记录

17.2 综合实例：采集数据的数据库管理

将采集的数据纳入数据库进行管理是一项具有实际意义的工作，这里以 3 个随机函数来模拟数据采集样本，并通过数据库来保存这些数据。具体的操作步骤如下。

① 新建一个 VI，并加以命名保存。

② 在前面板，添加一个波形图控件和一个数值输入控件，以显示数据波形和限定采用样本数。

③ 切换至后面板，放置一个 For 循环图标，并以样本数作为循环值。

④ 为了模拟采样间隔，在循环内添加一个 10 毫秒的延时环节。

⑤ 依次放置“DB Tools Open Connection.vi”、“DB Tools List Columns.vi”、“DB Tools Insert Data.vi”及“DB Tools Close Connection.vi”图标。

⑥ 创建一个含有 3 个数据输入值的簇，并为每个数据创建“值”属性节点。

⑦ 为相应的输入端创建对应的输入控件，经连线程序框图和显示效果分别如图 17-21 和



图 17-22 所示。

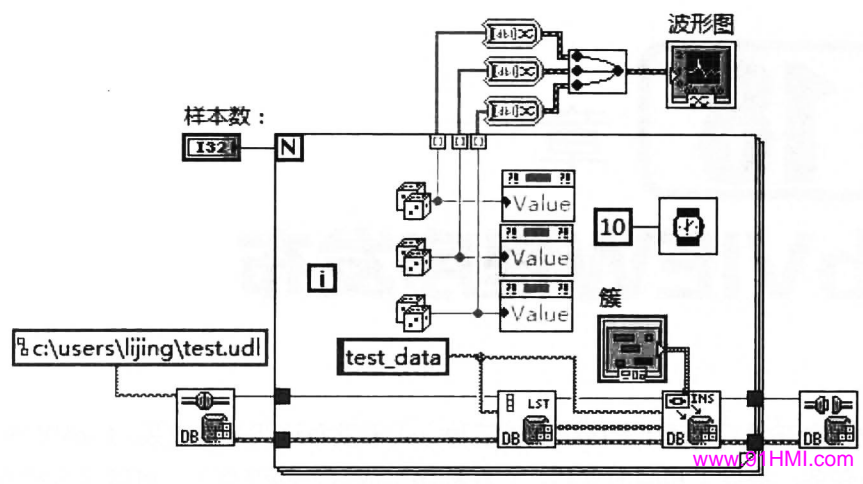
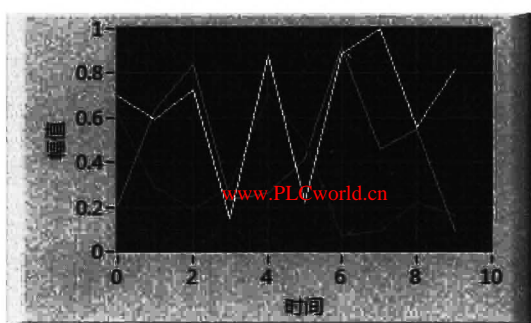


图 17-21 采集数据的数据库存储程序框图



样本数: 10 0.817557 0.162171 0.0942624

图 17-22 采集数据的显示界面

通过运行程序，可发现在数据库中已添加了 10 个数据记录，内容如图 17-23 所示。

test_data: 表			
	wendu	yali	niuju
	.9852673	.492962	.2201633
	.6969988	.6136185	.2122927
	.5924755	.2866289	.632319
	.7241741	.1931337	.8350009
	.1497322	.2902756	.2332196
	.8815034	.7036093	.2592
	.2210292	.4942468	.4135261
	.8902788	.07123235	.9186772
	.9992334	.09430502	.4599939
	.5575107	.2208702	.5496536

记录: 9 共有记录数: 10

图 17-23 采集数据的数据库存储



第 18 章

LabVIEW程序发布

在完成程序开发后，设计者往往要以某种方式来发布这些开发成果。LabVIEW 为了便于处理此类任务，提供了相应的支持。它在程序生成规范的配合下，可以按多种方式进行程序发布。

【本章导读】

- ◆ 创建项目
- ◆ 程序生成规范
- ◆ 发布前的准备
- ◆ 源代码发布
- ◆ 创建应用程序
- ◆ 创建安装程序

按照 LabVIEW 的程序生成规范，它所支持的发布包括源代码发布、共享库、Zip 压缩文件、Windows 安装程序、.NET 互操作程序集等服务。

18.1 创建项目

在面对 LabVIEW 项目开发时，设计者的着眼点不应单纯停留在 VI 的层面上，而是要将整个项目作为一个整体去通盘进行研究。为此，LabVIEW 推出了相应的项目管理器来帮助设计者高效地实施管理。

在打开 LabVIEW 时，首先呈现的就是图 18-1 的界面。此时，作为一个项目的起点，应从“创建项目”开始。

单击“创建项目”后，将弹出图 18-2 的界面。在该界面下提供了相应的生成模板及项目范例。单击“项目模板”，则可创建一个空白的项目并弹出浏览器窗口，浏览器的界面如图 18-3 所示。此时，可通过“保存项目”工具图标对新建项目进行命名存储。

透过项目浏览器，可在项目列表的底层见到“程序生成规范”项，通过它可执行相应的发布。

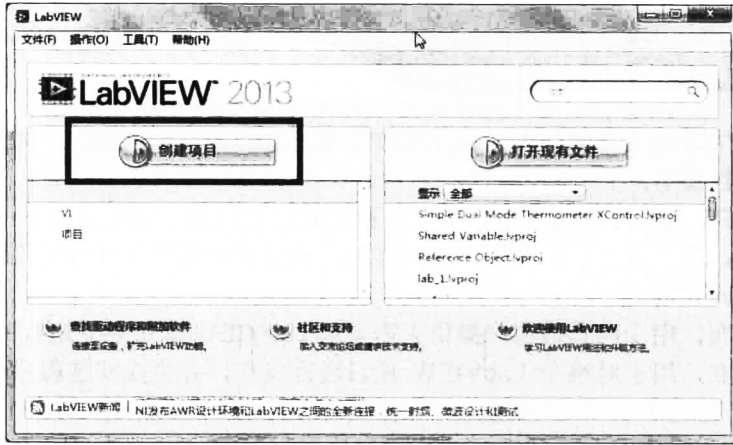


图 18-1 起始界面

www.91HMI.com

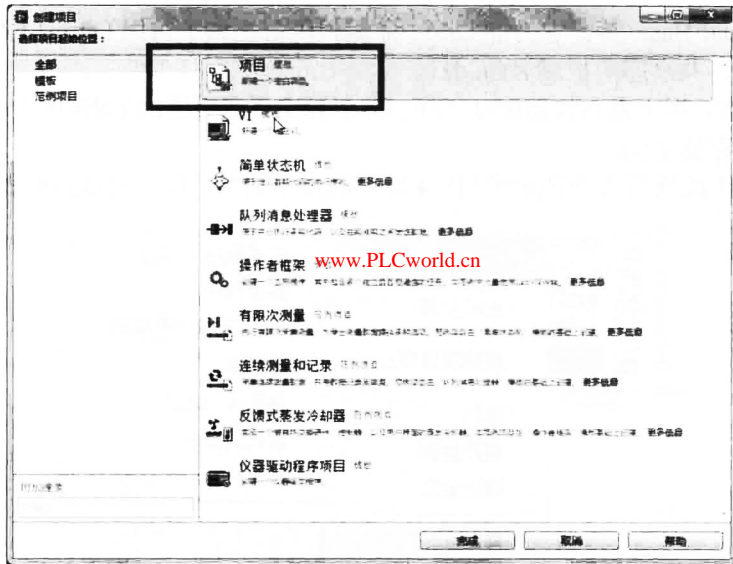


图 18-2 创建项目界面

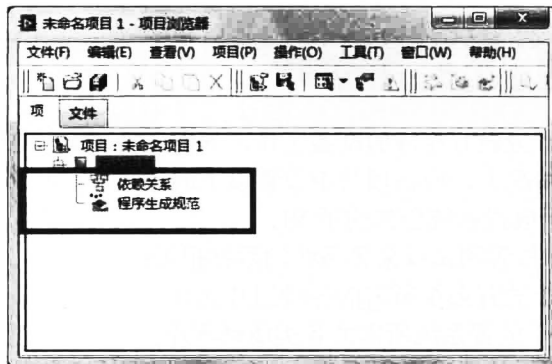


图 18-3 项目浏览器界面

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

www.91HMI.com

www.PLCworld.cn



18.2 程序生成规范

所谓“程序生成规范”就是为了进行某种发布所需的各种设置操作，如需要包含哪些文件、要创建哪些目录等。

按照 LabVIEW 的程序生成规范，可以多种方式进行发布，其作用如下：

源代码发布，用于对一系列源代码进行打包，通过源代码发布可将源代码发送给其他的开发人员进行协同开发。

应用程序发布，用于向其他用户提供无需安装 LabVIEW 即可独立运行的应用程序。

安装程序发布，用于对整个 LabVIEW 项目进行发布，并生成安装程序包，是一种最常见的发布方式。

.NET 互操作程序集发布，是将 VI 用于 Microsoft .NET Framework，在使用该发布前应安装 Microsoft .NET Framework 2.0 或更高版本。

www.91HMI.com

共享库发布，用于在文本编程的环境中调用 VI，如 LabWindows/CVI、Microsoft Visual C++ 及 Microsoft Visual Basic 等。它为非 LabVIEW 编程语言提供了访问 LabVIEW 代码的方式，在 Windows 系统下，共享库的扩展名为 .dll。

Zip 文件发布，用于发布仪器驱动程序、多个源文件或完整的 LabVIEW 项目，以形成压缩文件包来节省存储空间。

右击“程序生成规范”，则弹出图 18-4 所示的快捷菜单，从中可以选取相应的发布方式。

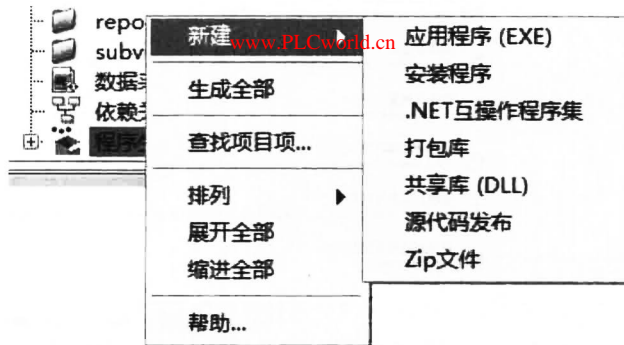


图 18-4 程序生成规范菜单

18.3 发布前的准备

在发布应用程序前，应做好相应的准备工作，如创建便于文档管理的文件夹及相应的支持性文件等。在常规的情况下，应在项目中设置以下的文件夹：

subVI 文件夹，用于放置系统全部的子 VI；

report 文件夹，用于放置测试或采集系统的数据报表；

picture 文件夹，用于放置系统所需的各种图片文件；

Drivers 文件夹，用于放置系统所需的各类驱动程序；

database 文件夹，用于放置数据库文件，以便于数据的检索和对比分析；

data 文件夹，用于放置原始的采集数据，以便对数据做进一步加工处理。



在项目中，创建文件夹的方法如下。

- ① 右击“我的电脑”，由快捷菜单选取“新建→虚拟文件夹”，如图 18-5 所示。
- ② 对新建文件夹进行重新命名，以便于分类存储。在完成全部文件夹创建后，其存储结构如图 18-6 所示。

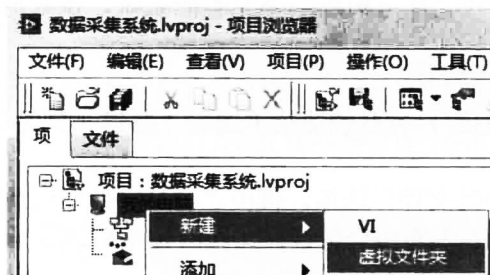


图 18-5 创建虚拟文件夹

www.91HMI.com

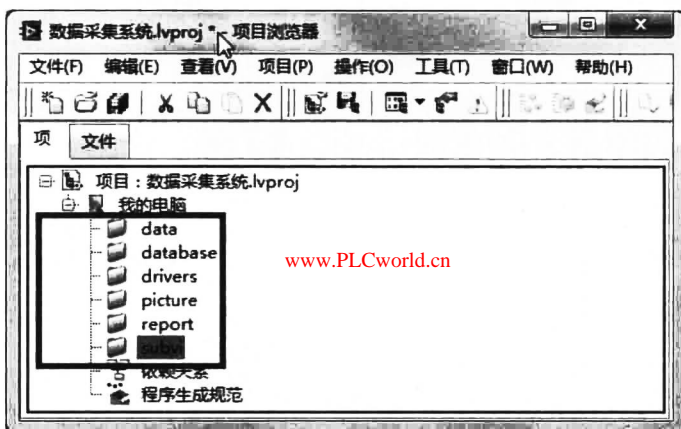


图 18-6 文件夹存储结构

- ③ 将准备好的支持性文件放置到对于的文件夹内，以便于引用。
- ④ 通过快捷菜单来创建一个 VI，并以此来作为整个系统的主程序，从而通过它来实现相关子 VI 的调用。

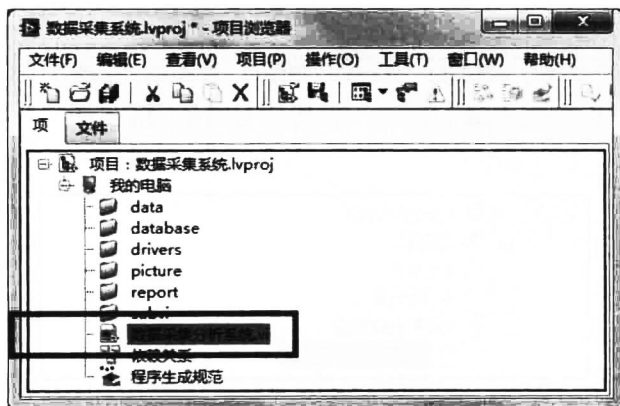


图 18-7 创建 VI 主程序



18.4 源代码发布

源代码发布主要作用是将一系列的源文件进行整体打包，其中包含 LabVIEW 安装目录中的库文件等，通过这些文件可构成一个完整的系统，供其他的开发人员在 LabVIEW 中使用。

源代码发布的操作方法如下：

① 右击“程序生成规范”，并从快捷菜单选取“新建→源代码发布”，则会弹出源代码发布对话框。

② 在对话框中选取“源文件”选项，再由“项目文件”栏选取待发布的源代码文件，单击“右移”按钮 \rightarrow ，则在发布中会包含该文件，其界面如图 18-8 所示。

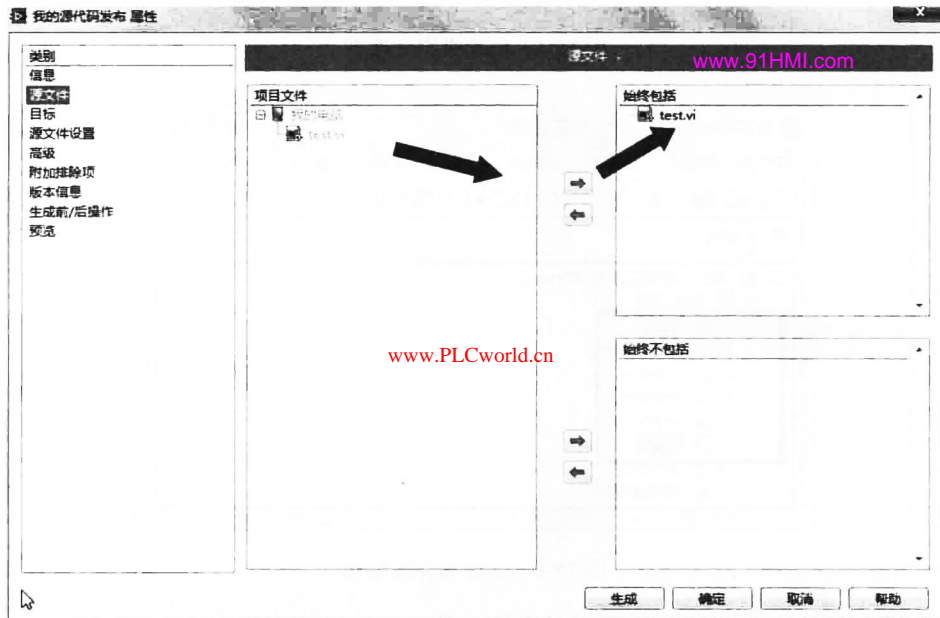


图 18-8 源代码发布对话框

③ 单击“生成”按钮，则执行源代码发布，完成后的界面如图 18-9 所示。

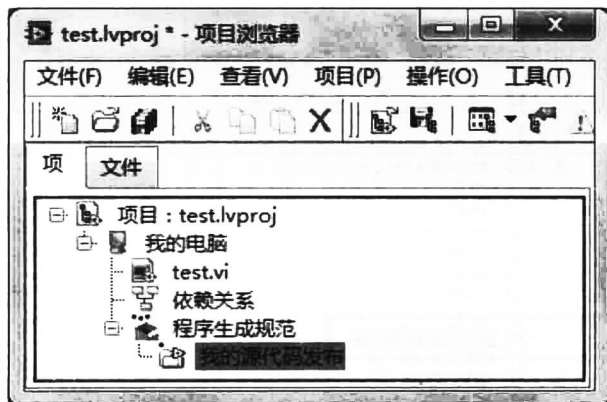


图 18-9 源代码发布



18.5 创建应用程序

所谓“应用程序”此处是指在未安装 LabVIEW 开发系统的环境下能独立运行的程序，在 Windows 系统下，其应用程序为.exe 后缀；在 Mac OS 系统下，其应用程序为.app 后缀。

创建应用程序的操作如下。

- ① 右击“程序生成规范”，并由快捷菜单选取“新建→应用程序（EXE）”。
- ② 在弹出的应用程序发布对话框中，对相应的选项进行设置或修改。
- ③ 单击“生成”按钮，则可执行应用程序发布，完成后的界面如图 18-10 所示。

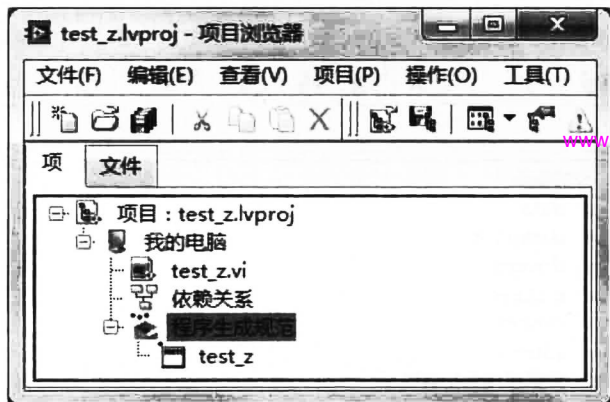


图 18-10 应用程序发布

18.6 创建 Zip 文件

所谓“Zip 文件”是将指定的文件进行压缩存储的文件，创建 Zip 文件有利于进行文件的传递和交流，其文件的后缀为.zip。

创建 Zip 文件的操作如下。

- ① 右击“程序生成规范”，并由快捷菜单选取“新建→Zip 文件”。
- ② 在弹出的应用程序发布对话框中，对相应的选项进行设置或修改。
- ③ 单击“生成”按钮，则可执行 Zip 文件发布，完成后的界面如图 18-11 所示。

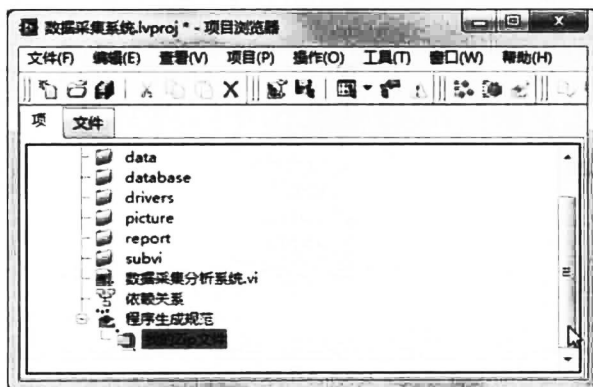


图 18-11 Zip 文件发布



18.7 创建安装程序

创建安装程序可对应用程序、共享库及源代码发布等生成安装程序。
创建安装程序的操作如下。

- ① 右击“程序生成规范”，并由快捷菜单选取“新建→安装程序”。
- ② 在应用程序发布对话框中，对相应的选项进行设置或修改。
- ③ 单击“生成”按钮，则可执行安装程序发布，完成后的界面如图 18-12 所示。



图 18-12 安装程序发布

通过运行程序，可获得 setup.exe 安装文件，其结果如图 18-13 所示。双击该文件，则会呈现图 18-14 的安装界面，通过单击“下一步”，持续进行安装处理，直至出现图 18-15 的界面完成安装。

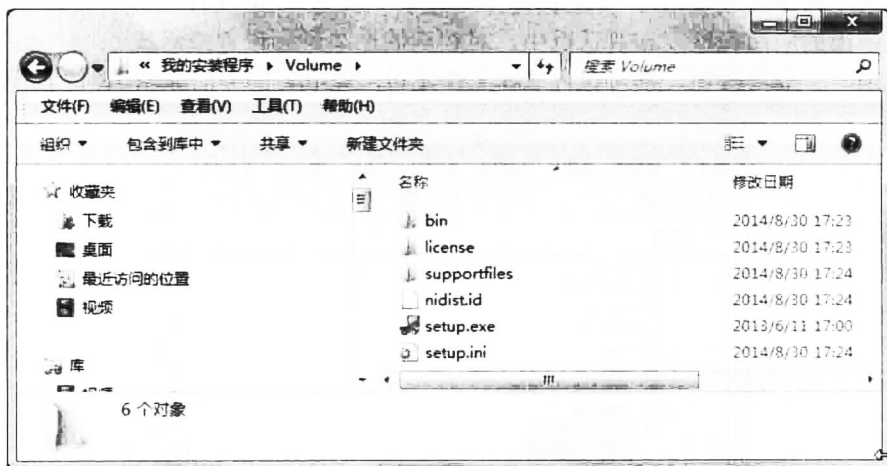


图 18-13 生成 setup.exe 安装文件

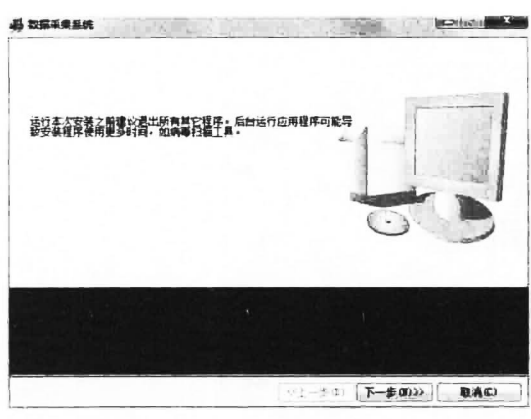


图 18-14 执行程序安装

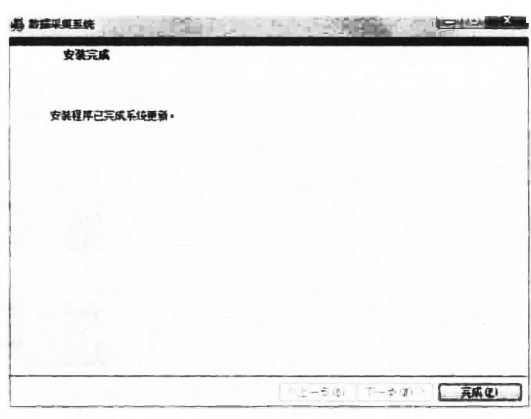


图 18-15 完成程序安装



通过对 LabVIEW 的学习，相信大家已经能够从编辑程序开始，直至发布自己的设计成果了。

www.PLCworld.cn

www.91HMI.com

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

www.PLCworld.cn



第 3 部分

应用篇

第 19 章

多通道温度检测记录系统

www.91HMI.comwww.PLCworld.cn

在工业生产过程中，温度总是扮演着主要的角色。因此对温度的管控就成为一件非常普遍的工作。多通道温度检查记录系统就是基于这一背景而设计，由于具体的检测要涉及 RS232/485 串行通信转换及多台温度测控仪表，因此这里以虚拟的方式来介绍其系统的设计，并不涉及具体硬件的购置和组建。

【本章导读】

- ◆ 虚拟硬件环境
- ◆ 程序框架设计
- ◆ 功能模块设计

利用 LabVIEW 与各种测控仪表相结合是构造虚拟仪器的主要方式，通常多是由串口来连接仪表和计算机，这为虚拟设计与测试提供了极大的便利。它使设计工作可以分成硬件设计和软件设计两部分，并可独立进行。当都通过测试后，两部分搭接在一起即可执行联合测试。采用该设计方法是提高开发效率的一条主要途径。

19.1 虚拟硬件环境

温度检测记录系统的原型是用智能型调节仪表与计算机构造一个测控系统，并采用 LabVIEW 来开发软件。为了简化开发的环境要求，这里采用虚拟的方式来组织设计，它可增



强设计的独立性，并避免硬件方面影响。

系统连接仪表的途径是串行通信接口，由于计算机的串行接口属 RS232 通信是点对点通信，因此在进行多通道测控时需通过 RS232/485 转换来进行 RS485 通信，其虚拟的硬件环境如图 19-1 所示。

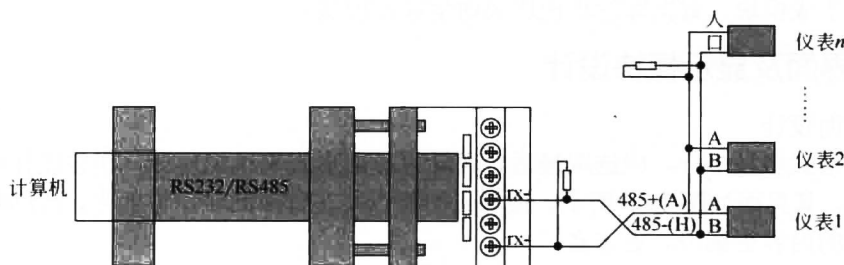


图 19-1 虚拟的硬件环境

在实际应用中，是由系统发出指定通道的采集指令，再由指定的仪表反馈回采集数据。作为虚拟系统，它无需发送采集指令，只需提供相应的模拟数据即可。故虚拟测试系统的成本更低、效率更高。

www.91HMI.com

19.2 程序框架设计

多通道温度检测记录系统在功能上主要分为检测管理和再现管理两部分，因此在整体构架设计时选用了选项卡控件来进行切分，其程序框架如图 19-2 所示。

检测功能由 4 个环节组成，并受启动按钮的控制。由于 4 个环节有顺序性，因而采用顺序结构组织设计。

再现功能有 3 个环节组成，其中的再现处理受再现按钮的控制。在使用时，通常是先从数据库中查到指定条件的数据文件名，然后再打开此文件，以实现数据曲线的再现。

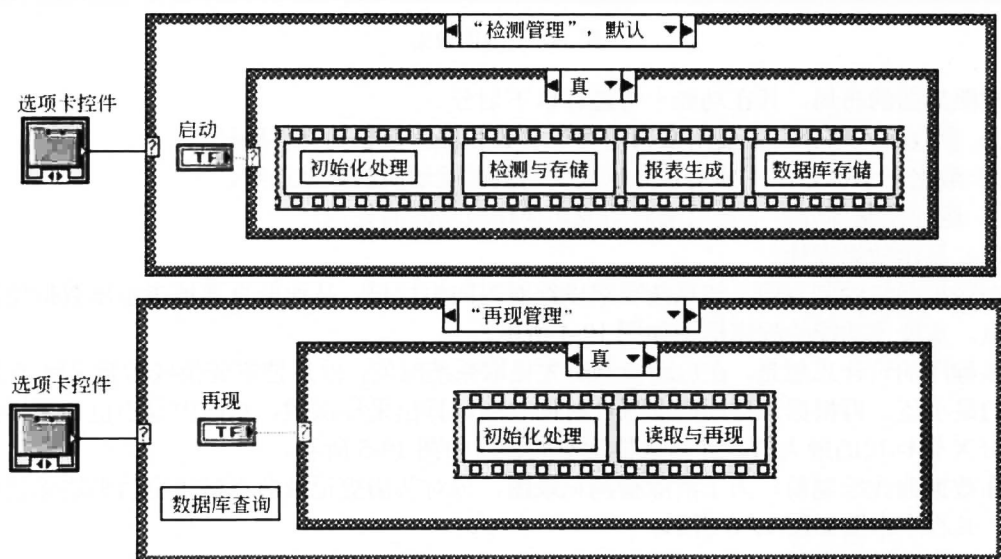


图 19-2 程序框架



19.3 功能模块设计

在功能实现方面,这个系统可划分4大模块,其分别是界面及显示模块、数据文件读写模块、报表生成模块、数据库管理模块及通信收发模块。

19.3.1 界面及显示模块设计

(1) 界面设计

界面的形式多种多样,应选取最适合设计目标的形式来组织设计,在该设计中是采用选项卡的形式,其界面如图19-3所示。该形式的特点是在有限的显示界面内,借助功能的划分可以使界面的内容更集中、更丰富。



图 19-3 界面效果

按照界面的布局,其在功能上可进行以下划分。

- ① 参数输入部分,主要用来指定运行参数和保存数据的文件路径。
- ② 数据显示部分,主要用来显示采集的数据及绘制的数据曲线。
- ③ 运行控制部分,主要用来启动或切换相应功能的处理。

(2) 显示效果设计

在数据曲线绘制方面,该系统要求以绝对时间为标尺,从而能直观地指示出数据变化的时间点。实现该功能的程序框图如图19-4所示。

该程序的设计思想是,在启动检测时先提取系统时间,然后把它转换成秒数来赋予X轴标尺的最小值。再根据采样间隔和采样时间长度计算出采样次数,从而由最小值加采样次数而求出X轴标尺的最大值。计算采样次数的程序如图19-5所示。

在数据曲线绘制前,为了清除检测记录图,可对其历史记录节点赋予空值来清除已有的数据,其程序框图如图19-6所示。

(3) 生成波形图片

在完成检测后,通过程序可自动获取数据波形图来生成图片,以便独立存档或在报表中



引用，其程序如图 19-7 所示。

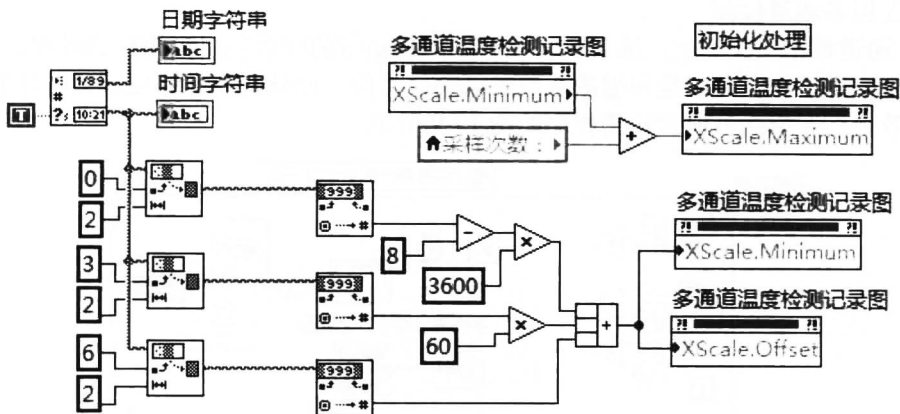


图 19-4 时间标尺设置

www.91HMI.com

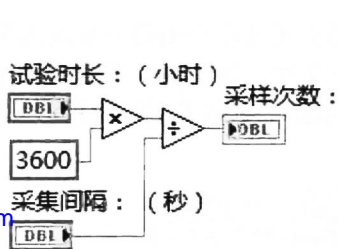


图 19-5 计算采样次数的程序

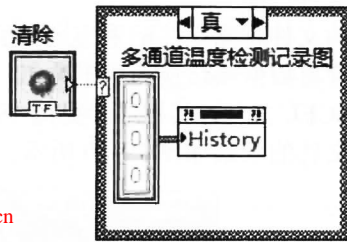


图 19-6 清除检测数据

(4) 控制处理

在程序控制控制方面，主要有两类控制，一个是启动控制，另一个是退出处理。

由于各功能模块是通过按钮来激活的，因此各功能按钮的动作行为应设为“释放时触发”，而非“单击时转换”。

由于界面是采用选项卡的形式，因此在进行退出处理时是按图 19-8 的方式。

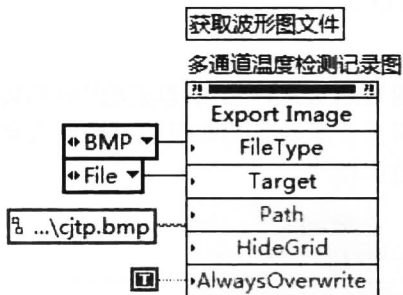


图 19-7 生成图片文件程序

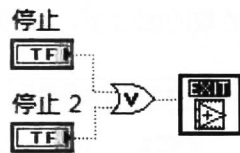


图 19-8 生成图片文件界面

19.3.2 数据文件读写模块设计

数据文件的读写模块由两部分组成，并分处在两个不同位置。为了构造系统运行的虚拟



环境，需设计相应的虚拟多通道检测环境。

(1) 虚拟多通道检测

以 3 通道数据采集为例，通过循环结构和随机函数的配合就可虚拟检测过程。为了增强逼真性，可通过增设偏移量和缩放系数来调整信号值。此外，添加相应的延时环节，也可增添逼真效果。虚拟的多通道检测程序如图 19-9 所示。

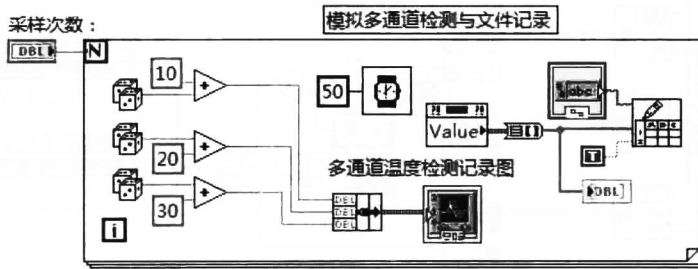


图 19-9 虚拟多通道检测及数据存储

www.91HMI.com

(2) 文件存储

数据的文件存储是伴随采集过程的，它是在采集循环内通过波形图的值节点来获取的，经转换处理将数据簇转换为数据数组，再插入指定的数据文件。为了便于审核，此处是将数据存入 EXCEL 文件，其程序参见图 19-9。其中的数组显示控件是为了实时地显示采集数据。数据文件的形式如图 19-10 所示。

	A	B	C	D	E	F	G
86	10.326	20.904	30.738				
87	10.294	20.983	30.211				
88	10.083	20.183	30.781				
89	10.476	20.472	30.911				
90	10.674	20.308	30.449				
91							

图 19-10 数据文件形式

(3) 文件读取

文件读取功能被放置在再现处理过程中，通过指定文件名来读取文件中的数据，其程序如图 19-11 所示。为了浏览读取的数据，通过带垂直滚动条的数组显示控件来实现。此外，在相关数组函数的配合下，可通过循环来再现数据曲线。

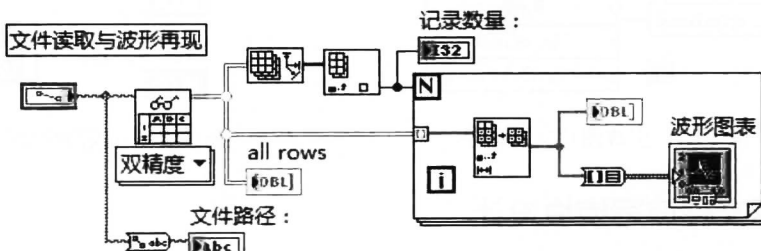


图 19-11 数据文件读取与曲线再现



值得指出的是，在曲线再现过程中其 X 轴标尺不再是绝对时间，而是采用数据序号。

19.3.3 报表生成模块设计

为适应长时间检测的需要，报表的版式要求采用横向报表。由于报表生成器所生成的报表属纵向报表，因此需要借助模板来生成横向报表，报表的模板形式如图 19-12 所示。

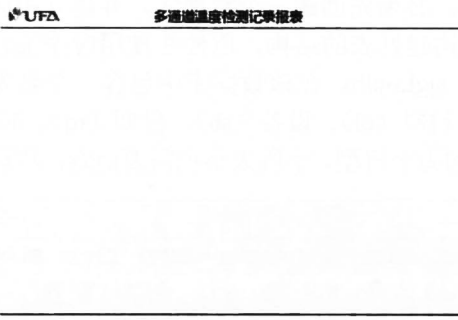


图 19-12 报表模板

利用报表模板所生成的报表如图 19-13 所示，其生成程序如图 19-14 所示。

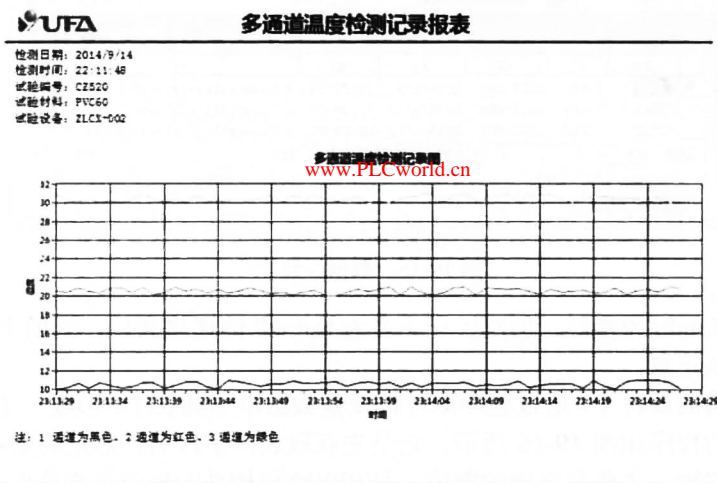


图 19-13 报表格式

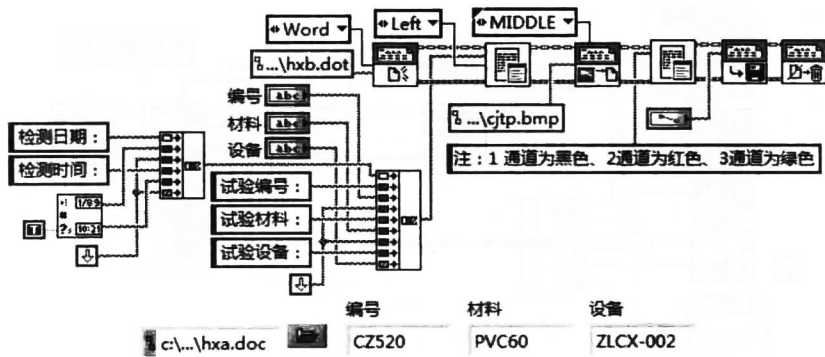


图 19-14 报表生成程序



19.3.4 数据库管理模块设计

数据库管理模块在功能上包括 3 部分，即构建数据库及连接、添加数据记录及查询数据记录。

(1) 构建数据库及连接

在进行数据库管理之前，必须先创建一个数据库，并建立相应的连接。此外，若不采用程序创建表，应在前期一并构建好表的结构，以便在使用时添加记录。

将数据库的名称命名为 sjgl.mdb，在该数据库中包含一个名为 test_a 的表，该表具有 6 个字段，分别是编号 (bh)、材料 (cl)、设备 (sb)、日期 (rq)、时间 (sj) 及文件路径 (wj)。为了便于处理，这 6 个字段均为字符型，字段大小视需要而定，其数据库结构如图 19-15 所示。

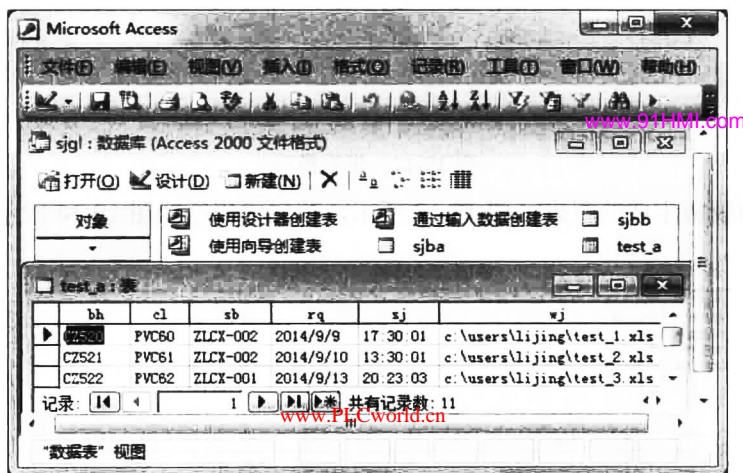


图 19-15 数据库结构

以数据库 sjgl.mdb 为基础，创建了一个名为 test.udl 的连接文件，以便于数据库管理。

(2) 添加数据记录

在具有表结构的条件下，通过程序即可把试验数据添加到表中，形成一条条的数据记录。添加数据库记录的程序如图 19-16 所示，它是先获取表的字段名，然后将数据信息拼接成一个字符串来实现添加。文件名字段的取值，是利用路径控件的值节点来获取的，经转换为字符串，以便执行字符串拼接。

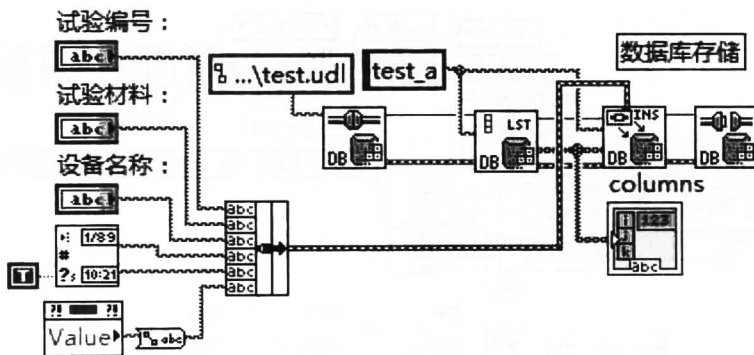


图 19-16 添加记录程序



(3) 查询数据记录

记录查询的主要用途是从数据库中获取相应的信息，以执行相关的处理。在该系统，其查询功能是为了按试验信息获得对应的数据文件名，以便读取数据实现曲线再现，其程序如图 19-17 所示。

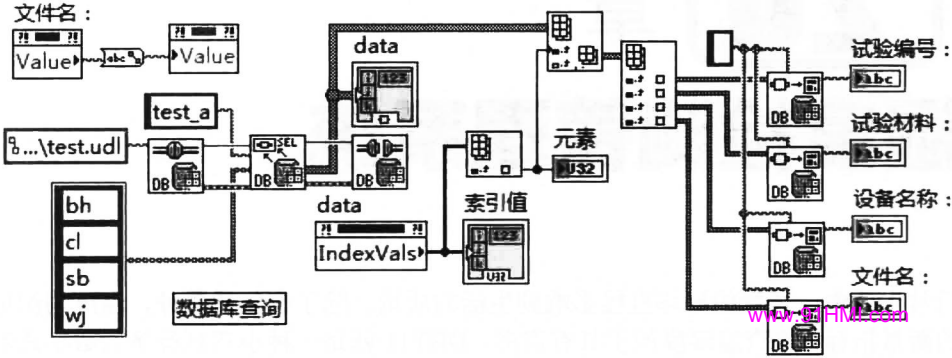


图 19-17 查询记录程序

按照实际的需要，在显示记录信息时只显示了部分需要的字段，未进行全字段的显示。此外，为了便于数据处理，通过数组函数的使用，将表的记录信息转换成了相应的字符串信息。

www.PLCworld.cn



面对检测系统的开发，可先从虚拟系统的设计着手，再逐步过渡到实际系统的设计，这是一条有效的开发途径。



第 **20** 章

温湿度检测管理系统

在日常生活中，环境的好坏直接影响到生活的质量。除了空气质量外，温度与湿度应是最基本的衡量指标。检查温湿度的工具有许多，DHT11 就是一种小巧且经济的数字式温湿度传感器。用单片机与 DHT11 相配合，是一种合理的温湿度采集系统设计方案。

【本章导读】

- ◆ 硬件设计
- ◆ 软件设计

在虚拟仪器设计中，采用专业仪表来组织研发往往具有效率高的特点，但在灵活性上往往会受定型仪表的制约，例如某些仪表的通信指令是不支持字符串指令的，而 LabVIEW 的串行接口则是采用字符串指令，因此不能用串行通信来连接此类仪表以构建虚拟仪器。此时，采用单片机技术或采用单片机与仪表相结合的方式，就成为一种必要且很好的补充形式。

20.1 硬件设计

20.1.1 硬件构成及工作原理

硬件环节在该系统是一个重要的组成部分，缺少它的配合，系统就无从感知外部环境的变化。为此硬件环境主要有 STC89C52 单片机、DHT11 传感器及 MAX232 串行接口等构成，其线路原理图如图 20-1 所示。其中，K1 是启动按键，L1 是启动指示灯。

DHT11 是一款数字式湿温度传感器，该传感器由电阻式测湿元件、NTC 测温元件及内部的高性能单片机构成，其外观如图 20-2 所示。

DHT11 具有功耗低、易于使用的特点，它可以实时地采集湿度和温度信息，并以单总线的方式来传送数据。该传感器的湿度测量范围为 20%~90%RH，温度的测量范围为 0~50℃。

DHT11 的引脚用途如下：

VDD：3.5~5.5V 电源。

DATA：单总线的串行数据传送引脚。

NC：空引脚。

GND：电源地。

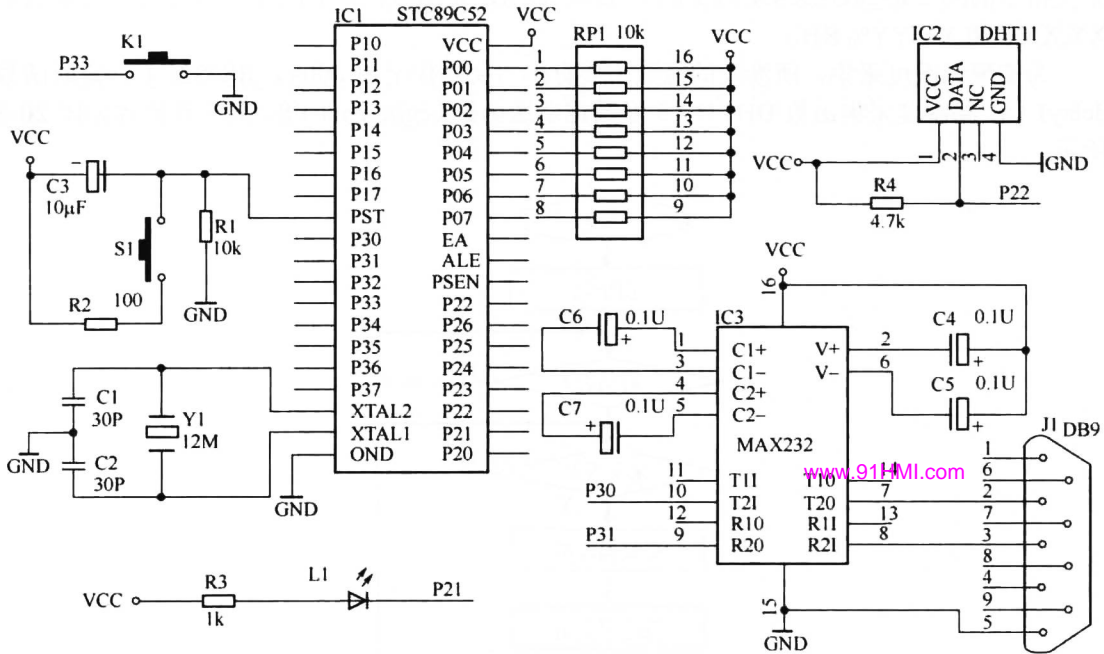


图 20-1 硬件线路原理图

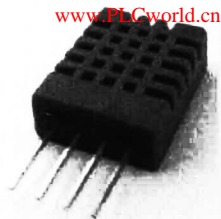


图 20-2 湿温度传感器 DHT11

DHT11 的数据包由 5 个字节构成，共 40 位，其数据结构如下：

byte4	byte3	byte2	byte1	byte0
00101101	00000000	00011100	00000000	01001001
湿度整数	湿度小数	温度整数	温度小数	校验和

温湿度的识读方法为：

湿度 = byte4 . byte3 = 45.0 (%RH)。

温度 = byte2 . byte1 = 28.0 (°C)。

校验和 = byte4 + byte3 + byte2 + byte1 = 73(49H)。

20.1.2 检测功能的实现

硬件环节要实现温湿度的采集，其处理流程为单片机接收采集指令后，先读取 DHT11 再经串口发送采集数据，以供 LabVIEW 进行后续处理。

串行通信的波特率是 2400bps，串行通信所支持的指令有 2 个，即串口检测指令@05T 和数据采集指令@05R。在串口通信正常的情况下，串口检测指令的回文是@05TOK。数据

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



采集指令的回文是@05RXXXXYYYY，其中 XXXX 是温度值，YYYY 是湿度值，分别表示 XX.XX°C 和 YY.YY% RH。

为实现温湿度采集，所涉及的函数主要有串口初始化函数 `initiate_RS232()`、定时函数 `delay1()`、温湿度采集函数 `DHT11()` 及数据发送函数 `beginSend()` 等，其工作流程如图 20-3 所示。

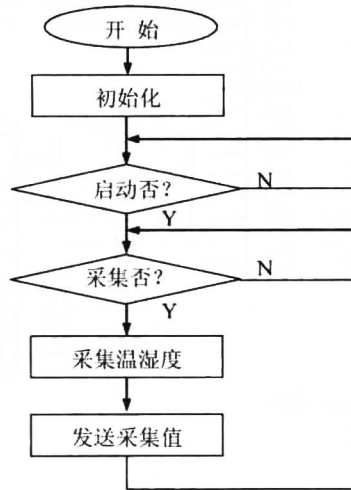


图 20-3 程序流程图

实现温湿度采集的单片机程序如下：

```

#include<reg51.h>
#include<absacc.h>
#include<intrins.h>
#define uchar unsigned char
#define uint unsigned int
uchar ch=0, zc=0, zj=0;
uchar sendCount; //发送字节个数
uchar receCount; //接收到的字节个数
uchar sendPosi; //发送位置
uchar receBuf[16];
uchar sendBuf[16];
sbit dht=P2^2; //定义 DHT11 引脚
uchar bdata dht_data; //存放 DHT11 字节数据
uchar dht_tab[5]; //存放 DHT11 数据
sbit KEY=P3^3; //启动键
sbit LED=P2^1; //启动指示
void delay1(uint time) //延时函数大约 9.9μs
{
while(time--);
}
void initiate_RS232(void)
{
SCON=0x58; //选择串口工作方式，打开接收允许
  
```



```

TMOD=0x20; //定时器 1 工作在方式 2
TH1=0xf3; //实现波特率 2400（系统时钟 12MHz）
TL1=0xf3;
TR1=1; //启动定时器 T1
ET1=0;
ES=1; //允许串行口中断
PS=1; //设计串行口中断优先级
EA=1; //单片机中断允许
}
void comm() interrupt 4 //串口中断处理函数
{
uchar zf;
if(TI)
{
TI = 0;
if(sendPosi < sendCount)
{
SBUF = sendBuf[sendPosi];
sendPosi++;
}
}
else if(RI)
{
zf = SBUF;
if(SBUF=='@') zj=0;
if(zj<12)
{
receBuf[zj] = zf;
zj++;
}
if((receBuf[zj-2]==0x0d)&&(receBuf[zj-1]==0x0a)) zc=zj;
RI = 0;
}
}
void beginSend(void)
{
sendPosi = 0;
TI=1;
}
void DHT11() //DHT11 温度湿度采集
{
uchar i,j;
EA=0; //关闭总中断
dht=0; //单总线引脚，设为 P2.2，主机发出开始信号
dht_data=0; //存放字节数据
for(i=0;i<20;i++)
delay1(102); //延时约 1ms

```

www.91HMI.com

www.PLCworld.cn



```

dht=1; //拉高并延时
while(dht); //等待主机发出的开始信号结束
while(!dht); //输出响应信号
while(dht); //输出响应信号延时
for(j=0;j<5;j++) //五个字节，共四十位
{
for(i=0;i<8;i++) //每字节八位数据
{
dht_data=dht_data<<1; //左移一位，以接收 DHT11 发来的数据
while(!dht); //等待 50μs 低电平结束
delay1(4); //延时约 40μs
if(dht==1)
{ //判断总线信号是否为逻辑"1"
dht_data=dht_data|0x01; //接收一位发来的数据 www.91HMI.com
while(dht); //等待低电平
}
else
{ //总线信号为低电平
dht_data=dht_data|0x00; //接收一位发来的数据
}
}
dht_tab[j]=dht_data; //存储数据
}
delay1(6);
EA=1;
void begintest(void) //检查单片机通信
{
sendCount=8;
sendBuf[3]='T';
sendBuf[4]='O';
sendBuf[5]='K';
sendBuf[6]=0x0d;
sendBuf[7]=0x0a;
}
void beginnumber(void) //发送采集数据
{
uchar n=0;
sendCount=14;
sendBuf[3]='R';
DHT11();
sendBuf[4]= dht_tab[0]/10+'0'; //湿度整数部分
sendBuf[5]= dht_tab[0]%10+'0'; //湿度整数部分
sendBuf[6]= dht_tab[1] /10+'0'; //湿度小数部分
sendBuf[7]= dht_tab[1] %10+'0'; //湿度小数部分
sendBuf[8]= dht_tab[2] /10+'0'; //温度整数部分
sendBuf[9]= dht_tab[2] %10+'0'; //温度整数部分

```




```

sendBuf[10]= dht_tab[3]/10+'0'; //温度小数部分
sendBuf[11]= dht_tab[3]%10+'0'; //温度小数部分
sendBuf[12]=0x0d;
sendBuf[13]=0x0a;
}
void main()
{
LED=1;
while(KEY==1); //判别是否启动
LED=0; //指示已启动
initiate_RS232(); //串行通讯初始化
sendBuf[0]='@';
sendBuf[1]=0x30;
sendBuf[2]=0x33;
receCount=0;
sendCount=0;
zc=15;
while(1)
{
if(zc==6)
{
if((receBuf[0]=='@')&&(receBuf[1]=='0')&&(receBuf[2]=='5')&&(receBuf[4]==0x0d)&&(receBuf
[5]==0x0a))
{
if(receBuf[3]=='T') //T 通信测试
{
begintest();
beginSend();
}
if(receBuf[3]=='R') //R 数据采集
{
beginnumber ();
beginSend();
}
else //本机错误指令
{
zc=15; //非本机指令
}
}
}
}
}

```

www.91HMI.com

www.PLCworld.cn

20.2 软件设计

20.2.1 软件程序的结构设计

温湿度检测系统在软件设计方面可划分为 4 大模块，分别是界面显示模块、数据文件存



储模块、报表生成模块及通信收发模块。

该系统的处理流程为：

设置运行参数→检测通信状态→启动检测→数据处理（曲线绘制、数据存储、实时数据分析）→结束检测→报表生成

20.2.2 界面显示模块设计

根据设计要求，温湿度检测系统的界面如图 20-4 所示。

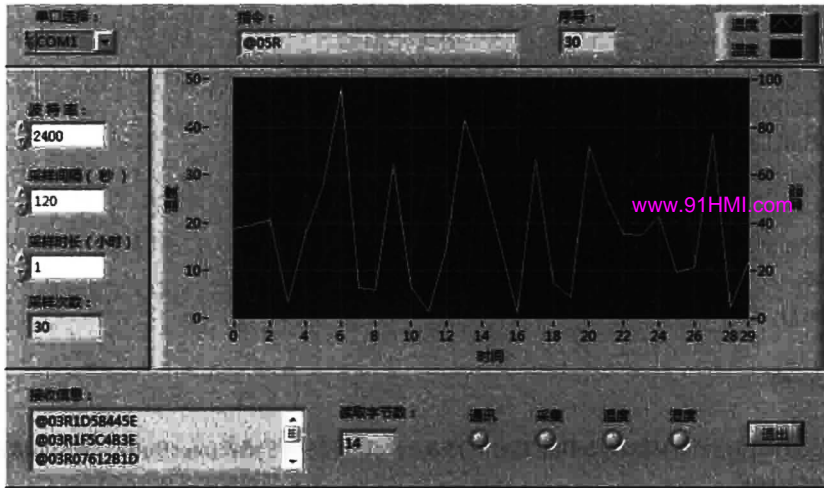


图 20-4 界面效果

按照界面的布局，在功能上可进行以下划分：

- ① 参数输入部分，主要用来指定串行通信参数和检测参数。
- ② 数据显示部分，主要用来显示发送的指令、接收的数据及绘制的数据曲线。
- ③ 控制处理部分，主要用来切换指令、控制曲线的可见性及结束运行。

(1) 显示效果设计

在数据曲线绘制方面，该系统采用了双数据标尺，以便直观地识读检测数据。此外，由于检测数据的变化范围时固定的，故温度标尺的范围设为 0~50，而湿度的标尺范围设为 0~100。对于时间标尺，则以自动调整标尺方式工作。

在绘制数据曲线前，应通过历史记录节点来清除已有的数据，使波形图表达达到擦除效果。此外，为了能清晰地识读曲线信息，应采用屏蔽技术来控制曲线的可见性，其设计思想是，先通过激活曲线节点来指定曲线，再通过开关按钮来控制该曲线是否可见，其程序如图 20-5 所示。

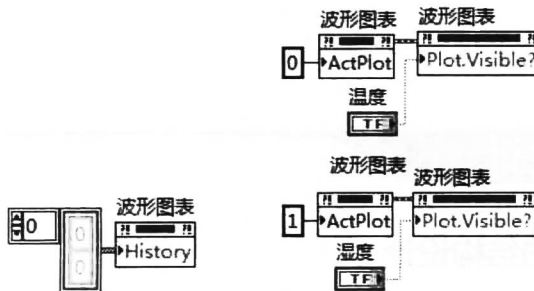


图 20-5 曲线屏蔽控制

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



(2) 虚拟检测处理

在软件设计过程中，为了强化设计目标，可采用虚拟检测来获取模拟数据，并通过模拟数据来构建功能程序。

虚拟检测效果的常用方法是用随机函数来构建，在增设偏移量和缩放系数的基础上能调整信号的变换范围。此外，增添延时环节，也利于增强逼真性。虚拟温湿度检测程序如图 20-6 所示。

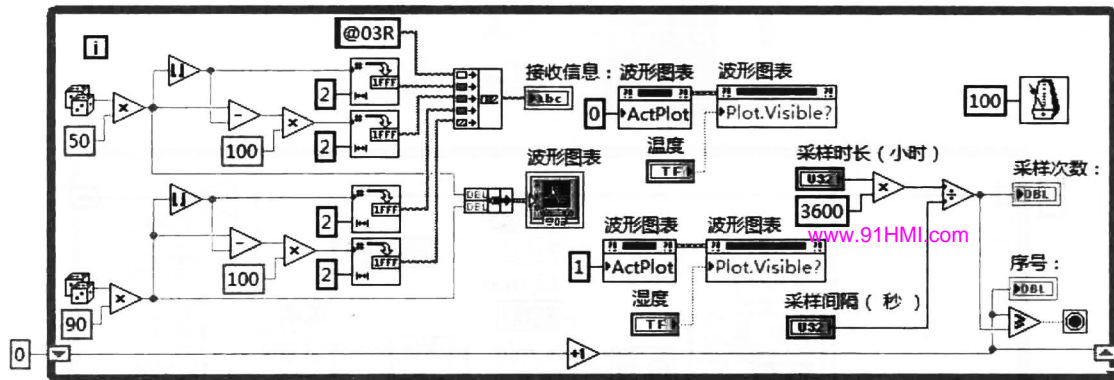


图 20-6 虚拟检测程序

(3) 运行控制

在运行控制方面，主要是按照输入的运行参数解算出采样的次数，再通过与运行计数器的比较来控制运行。此外，也可通过退出按钮来退出 LabVIEW，其程序如图 20-7 所示。

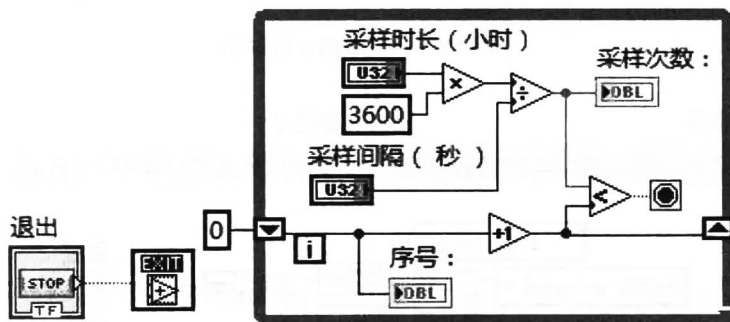


图 20-7 运行控制程序

(4) 实时数据分析

对于实时采集到的数据，要进行分析处理需要一定的设计技巧。在温湿度检测系统中，它所关心的是温湿度的极值和极值点，为此需要将检测的数据值拼接成一个数组，然后再利用数组的最大最小值函数来获取分析值，其程序如图 20-8 所示。

20.2.3 数据文件存储模块设计

数据文件的存储涉及两项内容，一个是自动命名文件，另一个是将分散的采集数据添加到文件中加以保存。

(1) 自动命名文件

为了便于数据文件的识别，按采集时间来命名文件是一种可取的方式。要实现文件自动



命名，需将路径、日期值及文件后缀拼接成一个字符串，然后再将其转换成路径值，其程序如图 20-9 所示。

温度	w_max	t1_max
0 43.5946	46.3177	16
40.9381	w_min	t1_min
17.7711	0.455597	23
湿度	s_max	t2_max
0 21.8524	88.9101	25
35.8689	s_min	t2_min
49.3181	1.19069	19

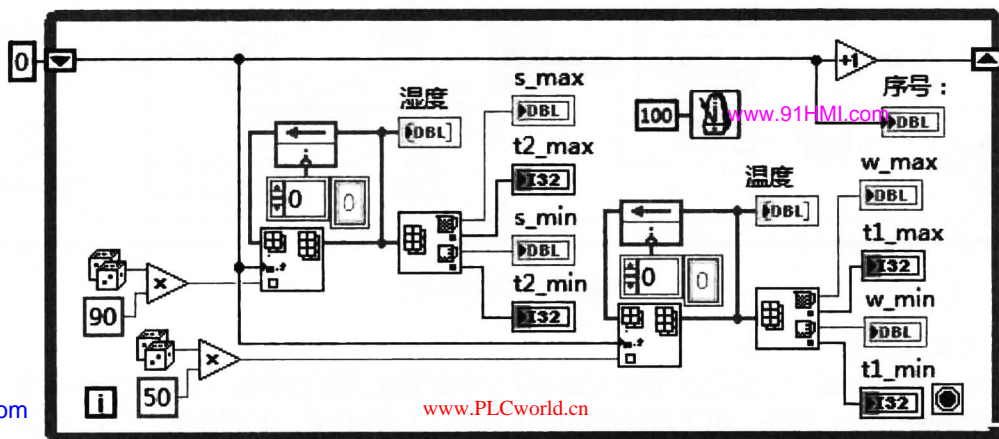


图 20-8 实时数据分析程序

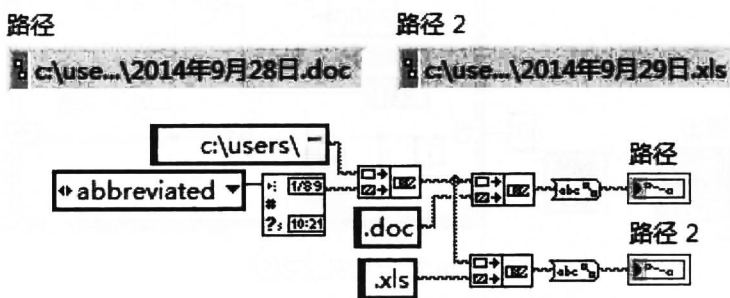


图 20-9 文件自动命名程序

(2) 数据文件存储

在 LabVIEW 中，大多数的文件存储是以数组的形式进行存储的，对于分散的数据是无法直接进行保存的。为此，可利用电子表格文件可执行追加存储的特点来保存采集数据，其程序如图 20-10 所示。它是将捆绑的绘图数据加以转换来形成数组数据，来追加到指定的文件中。

20.2.4 报表生成模块设计

报表的形式多种多样，为了能在报表中反映出分析结果，在报表中引入数据表格是一种

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050

www.PLCworld.cn

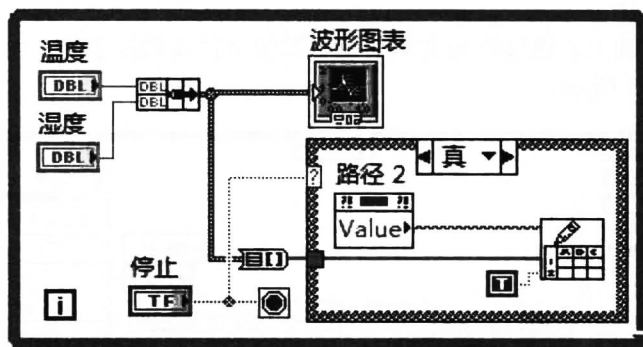


图 20-10 采集数据的文件存储

较好的方式。在该系统，为生成美观的报表，采用了报表模板，其形式如图 20-11 所示。生成的报表如图 20-12 所示。

www.91HMI.com

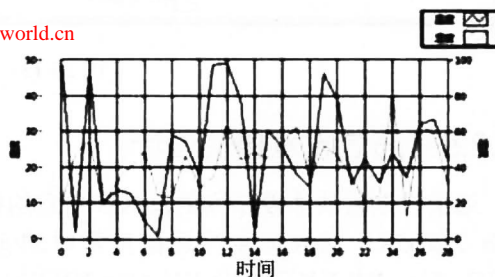
环境温湿度检测报告

环境温湿度检测报告

检测日期：2014/9/27

检测时间：21:48

www.PLCworld.cn



检测数据波形图

	温度	时间点	温度	时间点
最大值	49.6	0.0	81.4	25.0
最小值	0.7	8.0	12.1	26.0

图 20-11 报表模板

图 20-12 报表格式

(1) 生成波形图片

在报表生成过程中，要用到数据波形图所生成图片，其生成的方法是利用导出图像节点来生成.bmp 位图文件。

(2) 生成数据分析表

在报表中，由于要生成数据分析表，需要将零散的分析结果拼接成相应的一维数组，再通过二次拼接来构成二维的数据分析表。



(3) 生成报表文件

在报表模板的基础上，通过添加文本信息、图像文件及数据表格，可生成规范的检测表格，其程序如图 20-13 所示。

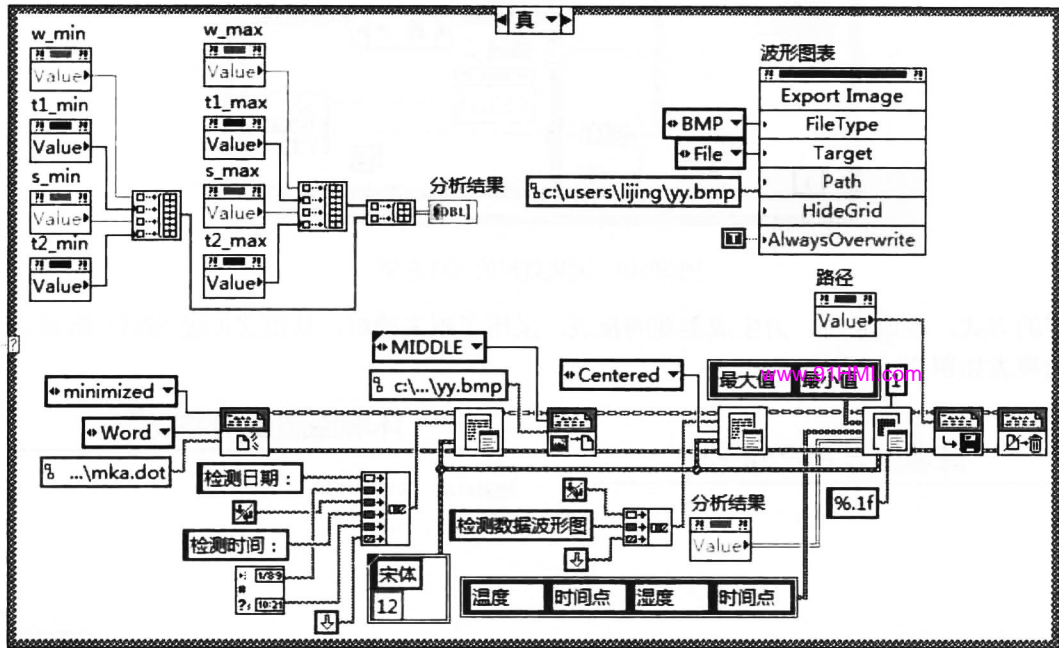
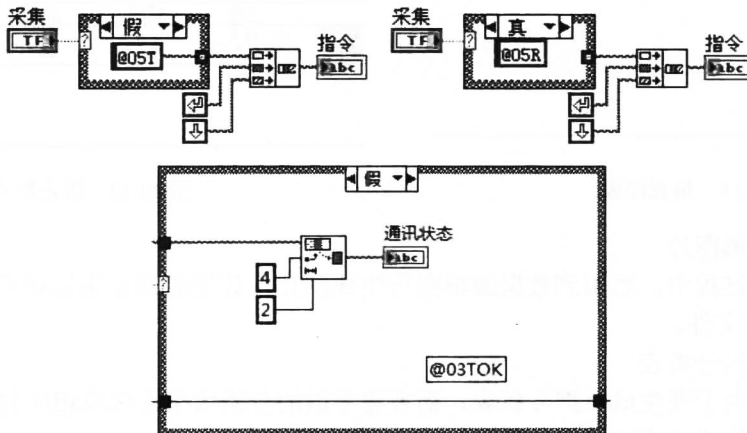


图 20-13 报表生成程序

20.2.5 通信收发模块设计

通信控制模块涉及指令的发送和信息的接收，发送指令为通信测试指令@05T 和数据检测指令@05R。接收信息为通信正常时的@05TOK 和采集数据@05RXXXXYYYY，其中 XXXX 表示 XX.XX°C，YYYY 表示 YY.YY% RH。

通过对接收内容的分析，可判别出是通信指令的回文或是采集指令的回文。再通过字符串的截取及转换处理，最终可提取检测数据。通信收发处理的程序如图 20-14 所示。



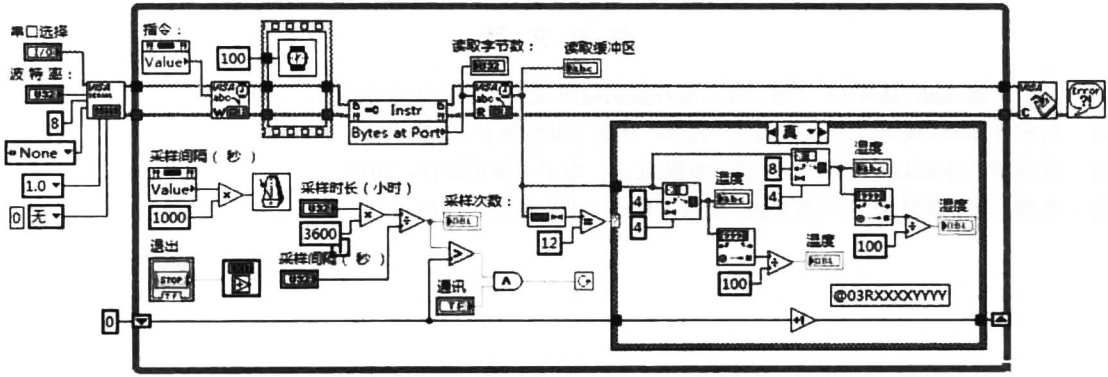


图 20-14 通信收发处理程序

在程序运行过程中，是通过通信按钮来启动，并由采集按钮来切换发送的指令。在采集按钮按下前，发送的指令为通信检测指令；当按下采集按钮后，发送的指令为采集指令。

在采集数据时，每采集一次数据计数器就进行一次计数值，直到达到采集次数时就退出采集。若单击退出按钮，则可退出运行程序。



将单片机技术同 LabVIEW 结合起来，不但可以增强开发的自主性，也有利于提升设计的性价比。

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



参 考 文 献

- [1] 徐晓东等. LabVIEW8.5 常用功能与编程实例精讲.北京: 电子工业出版社, 2009.
- [2] 付家才. LabVIEW 工程实践技术. 北京: 化学工业出版社, 2010.
- [3] 江建军, 孙彪. LabVIEW 程序设计教程.北京: 电子工业出版社, 2012.
- [4] 李静. 轻松学通 51 系列单片机. 北京: 化学工业出版社, 2014.

www.91HMI.com

www.PLCworld.cn

www.91HMI.com

www.PLCworld.cn

台湾永宏 PLC、四方变频器 : 15021230050

台湾永宏 PLC、四方变频器 : 15021230050



本书帮助读者从实用角度学习LabVIEW, 内容分为三篇。入门篇包括LabVIEW基本操作、基本数据类型及其相互转换、程序结构、数据运算函数、控件及其用法、数据显示与表达等; 提高篇包括LabVIEW文件输入/输出、字符串操作、数学分析、信号处理、软件间通信、时间相关操作、应用程序控制、网络编程技术、总线技术与仪器控制、数据库访问及程序发布等; 应用篇分别以多通道温度检测记录系统和温湿度检测管理系统为实例介绍如何运用LabVIEW解决实际问题。

本书以实例为依托, 将LabVIEW知识点的讲解化为直观的显示效果和具体的操作方法, 因而容易使初学者轻松入门并建立起学习兴趣。

本书适于从事测试与计量、电子开发等行业的工程技术人员阅读, 也可用作计算机、电子信息、通信工程、自动化等专业的本科生及研究生的教学参考。

LabVIEW

2013

完全自学手册

ISBN 978-7-122-22792-8



9 787122 227928 >

◀ 销售分类建议: 自控软件/计算机

定价: 68.00元

PLC 变频器 触摸屏 伺服 软启动 电气工程 15021230050