

高职高专系列教材

LabVIEW8.0

入门与提高案例教程

陈宏希 主 编
尤晓玲 副主编
邹益民 主 审

311
09

中国石化出版社

[HTTP://WWW.SINOPEC-PRESS.COM](http://www.sinopec-press.com)

内容简介

高职高专系列教材

LabVIEW8.0 入门与提高案例教程

陈宏希 主 编

尤晓玲 副主编

邹益民 主 审

中国石化出版社

内 容 简 介

本书是学习 LabVIEW 的入门教材,内容分为入门篇和提高篇两部分。入门篇介绍了 LabVIEW 中的数据操作、程序结构、数组、簇、子 VI、数据的图形显示、字符串和文件操作等相关内容;提高篇主要介绍了 LabVIEW 中信号处理的基础知识及 Express VI、脚本节点的使用及应用程序的创建等内容。

本书可作为高等职业院校电子类、自动控制类、机电类、计算机等相关专业的教材或教学参考书,也可供从事相关工作的工程师及研究人员参考。

图书在版编目(CIP)数据

LabVIEW8.0 入门与提高案例教程 / 陈宏希主编.

—北京: 中国石化出版社, 2008

(高职高专系列教材)

ISBN 978-7-80229-795-1

I. L… II. 陈… III. 软件工具, LabVIEW8.0-程序设计-高等学校: 技术学院-教材 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2008) 第 192740 号

中国石化出版社出版发行

地址: 北京市东城区安定门外大街 58 号

邮编: 100011 电话: (010)84271850

读者服务部电话: (010)84289974

<http://www.sinopec-press.com>

E-mail: press@sinopec.com.cn

北京密云红光制版公司排版

河北天普润印刷厂印刷

全国各地新华书店经销

*

787×1092 毫米 16 开本 12 印张 279 千字

2009 年 1 月第 1 版 2009 年 1 月第 1 次印刷

定价: 26.00 元

前 言

LabVIEW 是专门为工程师和科学家设计的直观图形化编程语言, 相比代码编程语言有直观、易学、易用的特点, 主要用于开发测试、测量与控制系统。经过 20 多年的发展, LabVIEW 业已成为事实的工业标准编程语言。

虚拟仪器技术是基于计算机的仪器及测量技术, 它将传统仪器由硬件电路实现的数据分析处理与显示功能, 改由功能强大的计算机来执行。计算机与适当的 I/O 接口设备连接是虚拟仪器的硬件平台, 依此平台编制具备测量功能的软件, 就能构成测试仪器, 这就是“软件就是仪器”说法的由来。虚拟仪器技术主要采用 LabVIEW 来实现, LabVIEW 已成为虚拟仪器的代名词。

LabVIEW 简单、易学、实用, 必将成为一线技术人员的首选工具。目前, 国内很多著名院校, 特别是理工科院校已相继开设了 LabVIEW 课程, 它也必将成为高职高专院校的必修课。可以想象, 不久的将来, 会有越来越多的人学习和使用 LabVIEW, 基于 LabVIEW 技术的应用 and 开发将遍布各行各业。

1. 本书特点

① 坚持“够用、好用”的原则, 注重基础知识和基本技能的培养, 起点较低, 适合入门级的学习;

② 采用案例教学法, 将理论知识贯穿和融入案例之中, 按步骤逐步实现案例功能;

③ 贯彻“工学结合, 教、学、做一体化”的课程体系建设思想, 注重读者多方面能力的培养;

④ 便于自学, 学员按书中所列案例的步骤进行操作, 即可掌握本书全部内容。

2. 本书内容安排

全书分为入门篇和提高篇, 共 9 章。入门篇 6 章, 分别是基本数据操作、程序结构、数组和簇、子 VI、数据的图形显示以及字符串和文件 I/O 操作; 提高篇 3 章, 包括信号处理基础及 Express VI、脚本节点的使用及应用程序的创建。

3. 学习方法

与其他编程软件的学习方法相类似, 积极思考, 举一反三, 大胆尝试和实践是学好 LabVIEW 的一大捷径。对于一位欲从事 LabVIEW 编程设计的人来说, 只捧着一本教程看是不够的, 更多的是要动手操作, 多做实例练习, 从中总结经验与技术, 再融入自己的一些思想, 这样会进步更快。

本教材适用于高等职业技术学院电子类、自动控制类、机电类、计算机类等专业的教学, 也可供从事相关工作的工程师及研究人员学习参考。

本书第 1 章和第 5 章由尤晓玲编写, 第 6 章由梁璐编写, 其余部分由陈宏希编写, 全书由陈宏希和尤晓玲统稿。

邹益民博士在百忙之中审阅了全书, 并提出了许多宝贵的修改意见, 在此表示衷心感谢! 由于作者水平有限, 书中错误之处在所难免, 恳请读者批评指正。

作者

2008 年 12 月

入 门 篇

第 1 章 基本数据操作

1.1 LabVIEW 简介

LabVIEW 是 Laboratory Virtual Instrument Engineering Workbench（实验室虚拟仪器集成环境）的简称，是美国国家仪器公司（National Instruments，NI）研发的功能强大的仪器和分析软件应用开发工具。

因为 LabVIEW 程序具有真实物理系统或仪器的外观，所以，LabVIEW 程序被称为虚拟仪器，简称为 VI，并以 .vi 作为文件的扩展名。LabVIEW 不同于基于文本的编程语言，它是一种图形编程语言，俗称 G（Graphics）语言，其编程过程就是通过图形符号描述程序的行为。

LabVIEW 主要应用在测试、测量和自动化领域，用于数据采集和控制、数据分析和数据显示等。在汽车、通信、航空、过程控制和生物医学等领域都有一定的应用。

1.2 LabVIEW8.0 的编程环境简介

1.2.1 LabVIEW8.0 的启动

双击桌面上的 LabVIEW8.0 图标 ，或选择“开始 | 程序 | National Instruments | LabView8.0 | LabVIEW”就可启动 LabVIEW8.0。启动后进入如图 1.1 所示 LabVIEW8.0 的启动界面，启动成功后界面如图 1.2 所示。

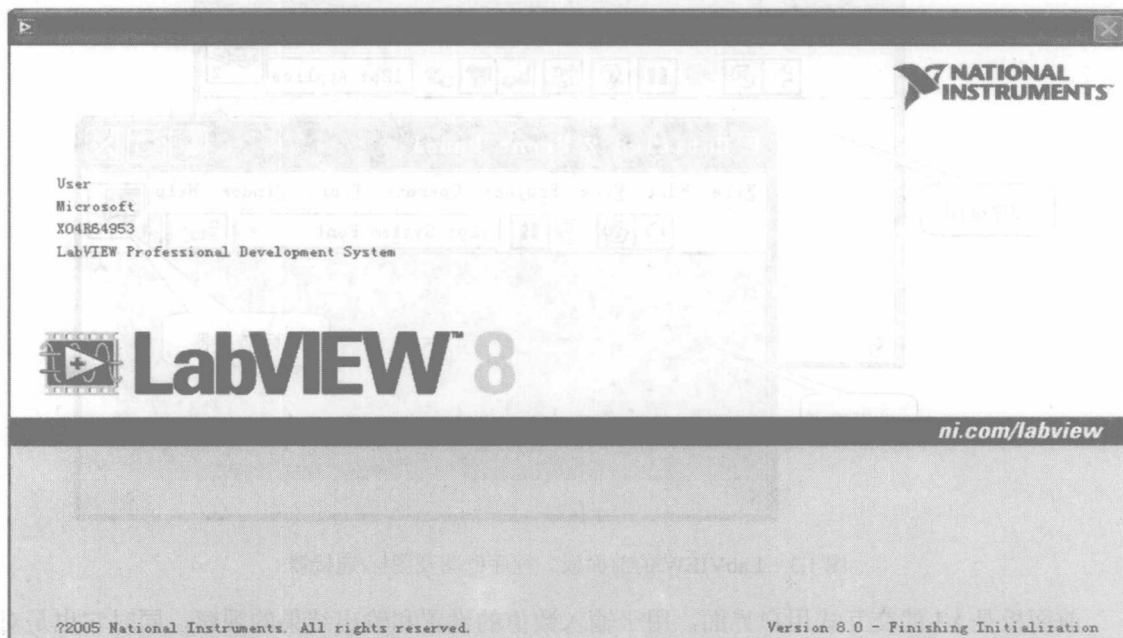


图 1.1 LabVIEW8.0 的启动界面

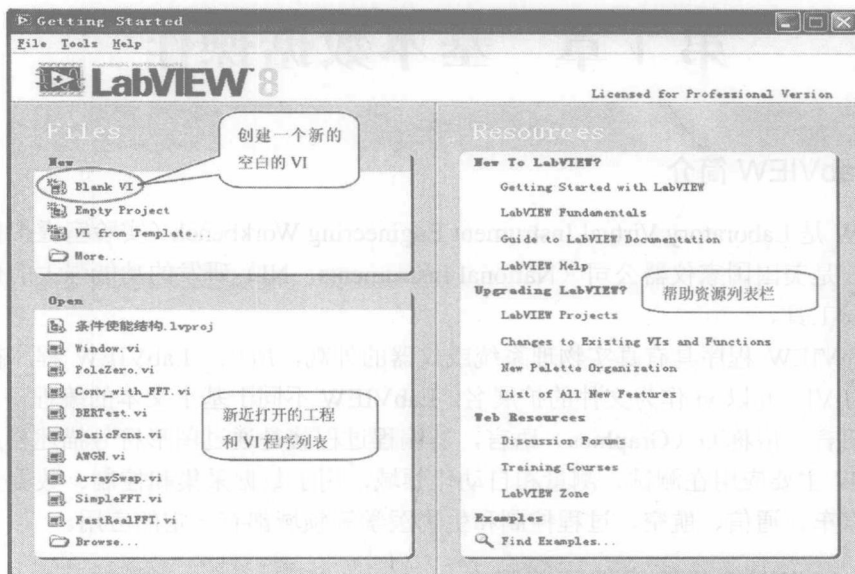


图 1.2 LabVIEW8.0 启动成功后的界面

1.2.2 新建一个空白的 VI

单击图 1.2 中 Files | New | Blank VI，就可新建一个空白的 VI。如图 1.3 所示，一个基本 VI 包含两个窗口：前面板窗口和程序框图窗口，由三部分组成：前面板、程序框图和图标/连接器。

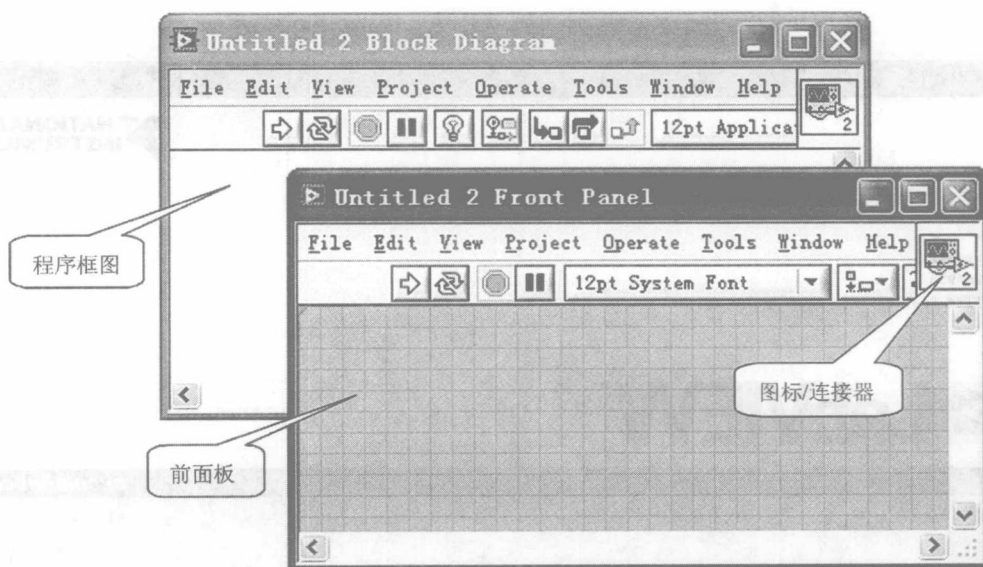


图 1.3 LabVIEW 的前面板、程序框图及图标/连接器

前面板是 VI 的交互式用户界面，用于输入数值的设置和输出结果的观察，同时它也是对真实仪器面板的模拟。图 1.4 所示是虚拟双踪示波器的前面板，其上有通道选择和时间分辨率、幅度分辨率调节等输入控制按钮或旋钮，也有图形或波形输出窗口，与真实示波器的仪器前面板基本相同。

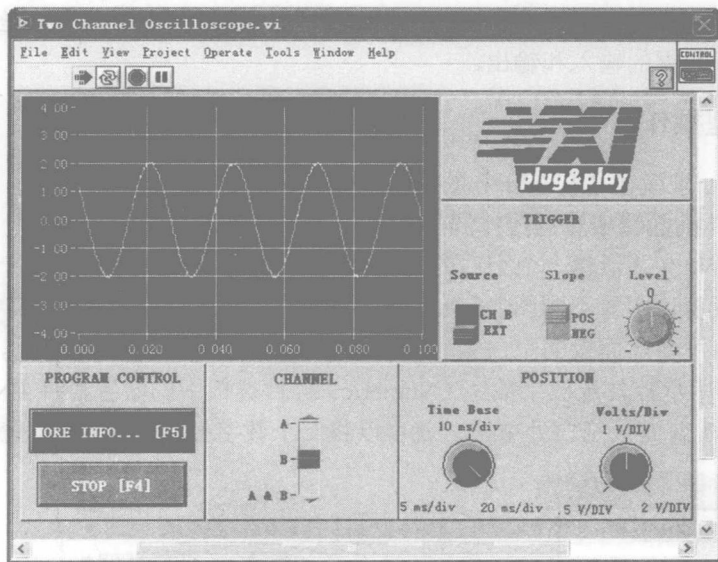


图 1.4 虚拟双踪示波器的前面板

程序框图是与前面板对应、用图形化编程语言编写的可执行代码，其上包含前面板中控件和指示器对应的图标及其连线端子，还有函数、常量、程序结构框架和连线等。图 1.4 双踪示波器前面板对应的程序框图如图 1.5 所示。

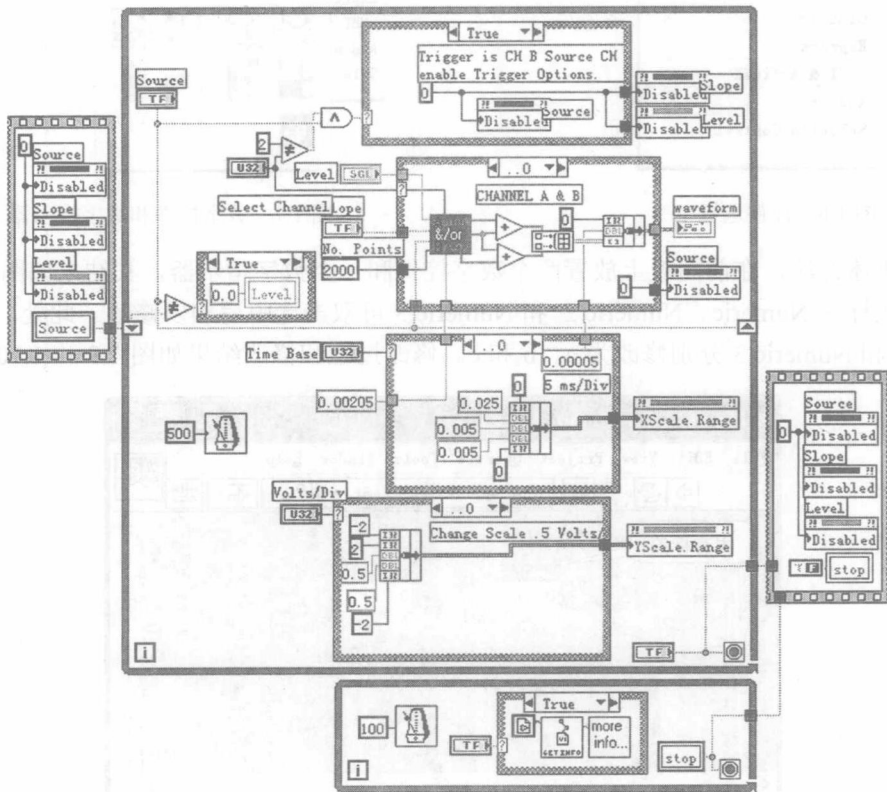


图 1.5 虚拟双踪示波器的程序框图

图标/连接器指定了数据流进流出虚拟仪器的路径。在程序框图中图标是 VI 的图形符号，而连接器则定义了数据的输入和输出。

1.3 数字型操作

【案例 1.3.1】 求两数之和 $c=a+b$ 的 VI

步骤 1：在 VI 前面板中添加控件和指示器

启动 LabVIEW8.0 并新建一个空白的 VI。

如果在前面板窗口中看不到图 1.6 所示控件选板(Controls), 可在 View 菜单中选择 Controls Palette 来打开；也可在前面板的空白处右击，动态显示图 1.6 所示控件选板。

单击如图 1.7 所示控件选板中数字 (Numeric) 控件选板中的数字控件 (Numeric Control)，然后移动鼠标到 VI 前面板空白处单击，就可以将一个数字控件放置在前面板上。

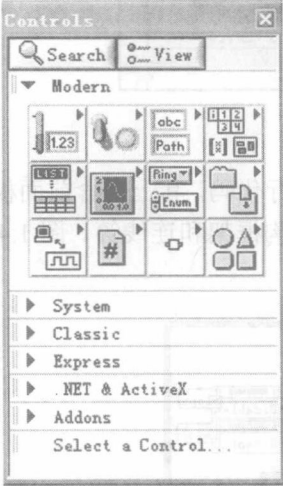


图 1.6 控件选板



图 1.7 数字控件和数字指示器

如上所述方法，在前面板上放置两个数字控件和一个数字指示器，其结果如图 1.8 所示。其中，默认标签 Numeric、Numeric 2 和 Numeric 3 可双击选中后加以修改。此处，Numeric、Numeric 2 和 Numeric 3 分别修改为 a、b 和 c。修改过程和修改结果如图 1.9 和 1.10 所示。

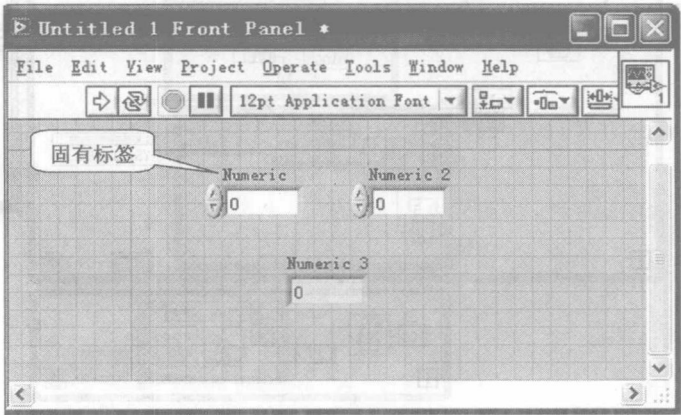


图 1.8 前面板中放置两个数字控件和一个数字指示器

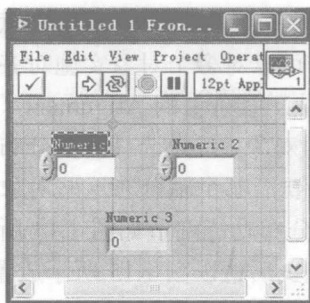


图 1.9 默认标签修改

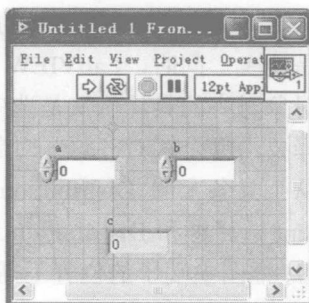


图 1.10 默认标签修改结果

说明：在 Windows 基本操作中的单击选中对象、右击打开快捷菜单、双击名字重命名、拖曳移动对象、空白区域拖曳选中多个对象、按住 Ctrl 键后再拖曳等同于复制以及 Ctrl+C 复制、Ctrl+V 粘贴、Ctrl+X 剪贴等快捷操作均在此有效。

步骤 2：编辑 VI 程序框图

打开 VI 程序框图窗口，如图 1.11 所示，可以看到前面板放置的数字控件的控件端子和数字指示器的指示器端子。

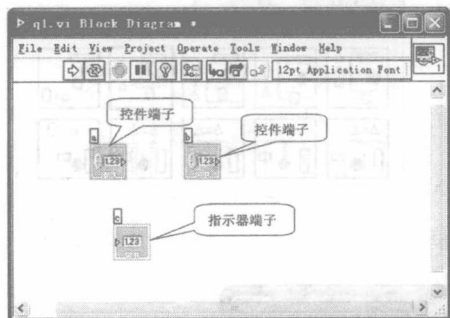


图 1.11 VI 程序框图中的控件和指示器端子

说明：从外观看，控件端子是粗边框，指示器端子是细边框；控件端子上的小箭头指向外部，表明该端子向外输出数据，而指示器端子上的小箭头指向端子的内部，表明该端子接收外部输入的数据。

提示：前面板窗口和程序框图窗口间的相互切换，除用单击标题栏或通过 Windows 菜单切换方法以外，LabVIEW 提供了快捷键 Ctrl+E，使用时简单快捷。

图 1.11 中，默认各类端子均以图标形式显示，其直观形象的特点不言而喻，但其占用空间大，在复杂问题中可能出现程序框图空间紧缺或图标放置过于密集不便于区分的情况，LabVIEW 针对端子提供了两种显示形式，一种是图标形式，另一种是非图标形式。非图标形式显示可使 VI 程序框图界面简洁、高效。具体做法是：右击 VI 程序框图中的端子，在弹出的快捷菜单中将 View As Icon 选项前的对勾去掉。对控件端子 a 的操作及结果如图 1.12 和图 1.13 所示。类似，可把其他端子也以非图标形式显示。

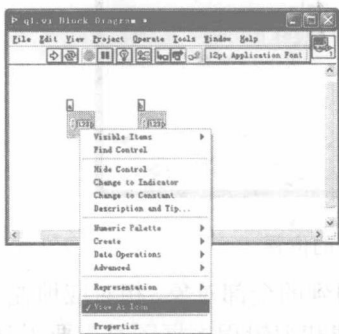


图 1.12 VI 程序框图中的控件端子 a 以非图标形式显示操作过程

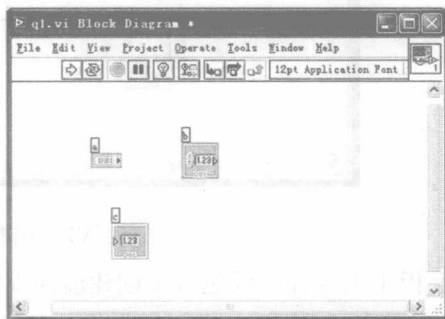


图 1.13 VI 程序框图中的控件端子 a 以非图标形式显示操作结果

另外，端子在 VI 程序框图中的位置、控件或指示器在前面板中的位置可以任意移动，具体做法就是拖曳。同时，LabVIEW 提供的对齐和分布工具可以对多个对象实施对齐和空间分布操作。对象对齐的具体做法是选中希望对齐对象，然后从工具条上的 Alignment（对齐）选择框中选择要沿哪个轴对齐这些对象。类似方法可分布一组对象，首先选择需要分布的多个对象，然后从工具条上的 Distribution（分布）选择框中选择要沿哪个轴分布这些对象。工具条上 Alignment 选择框和 Distribution 选择框的位置及对齐和分布功能细节如图 1.14 所示。本案例中 VI 程序框图中端子的位置不需要对齐和分布处理，拖曳后请以图 1.15 所示放置即可。

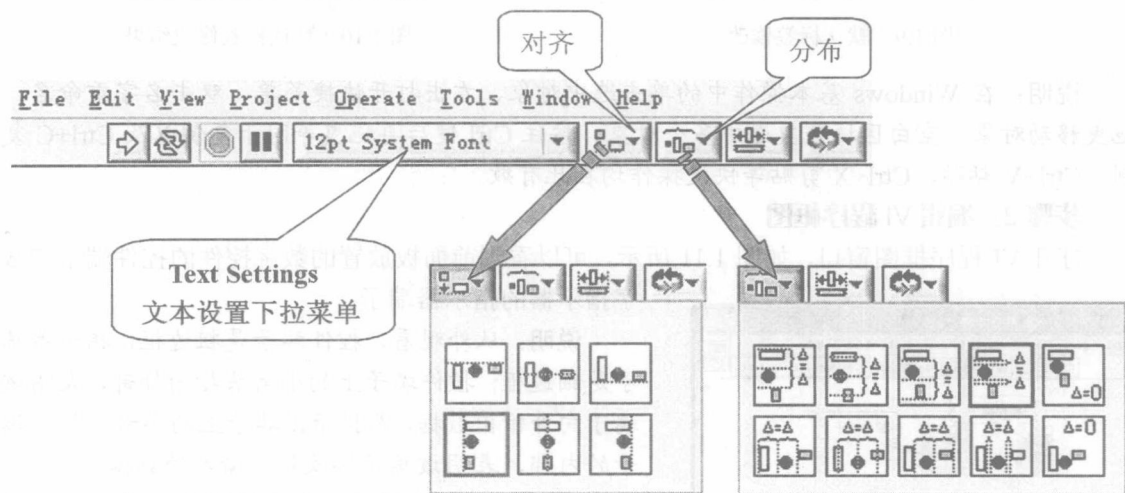


图 1.14 对齐和分布选择框

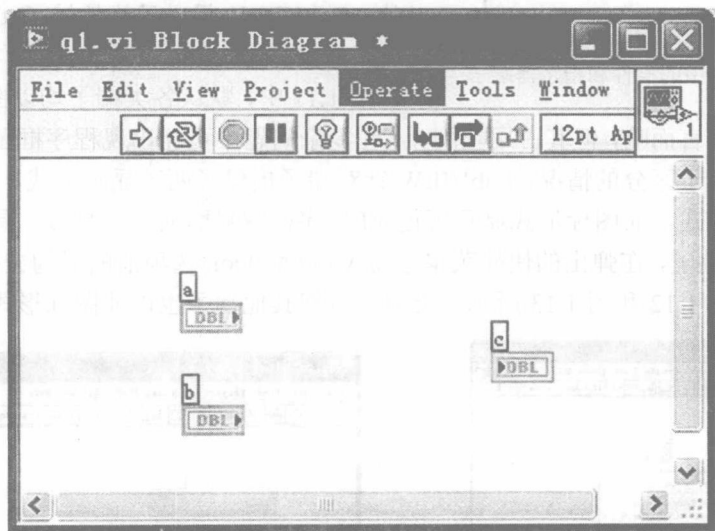


图 1.15 VI 程序框图中端子的位置

VI 前面板中的控件选板包含了创建前面板时需要用到的全部对象，相对应地在 VI 程序框图中有一个功能和函数（Functions）选板，它包含了创建和编辑程序框图时需要用到的有关结构、子 VI 和函数等对象。选择 View | Functions Palette 选项，或右击程序框图窗口空白处就可以显示如图 1.16 所示功能和函数选板。

在 VI 程序框图的功能和函数选板中,单击选中如图 1.17 所示的“数字型(Numeric)”下面的“加(Add)”函数,并放置到如图 1.18 所示位置。放置完成后,当鼠标指向该“加”函数时,其上的连线端子便显现出来,当鼠标指向连线端子时,鼠标变成线轴形状,即表示可以连线了。单击连线端子,移动鼠标,这时就有连线从线轴引出,移动鼠标直至另一个连线端子单击释放,原来的虚线变成了实线,表明两个端子已被连接。此例中,控件端子 a 与“加”函数的一个端子连接过程如图 1.19 所示。连线结果如图 1.20 所示。

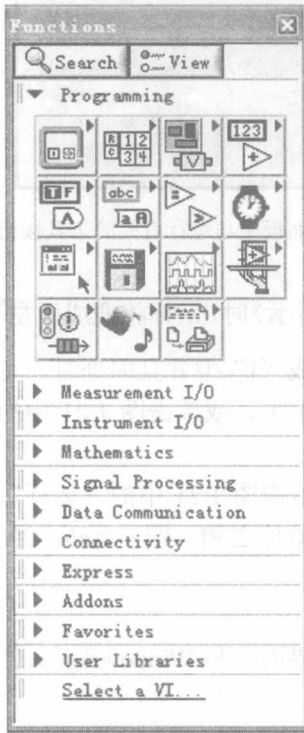


图 1.16 功能和函数选板

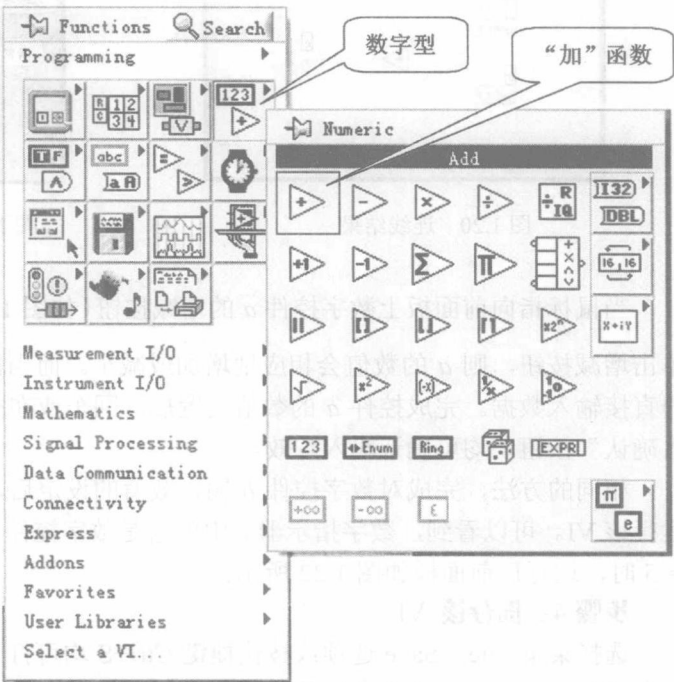


图 1.17 “加”函数的选择

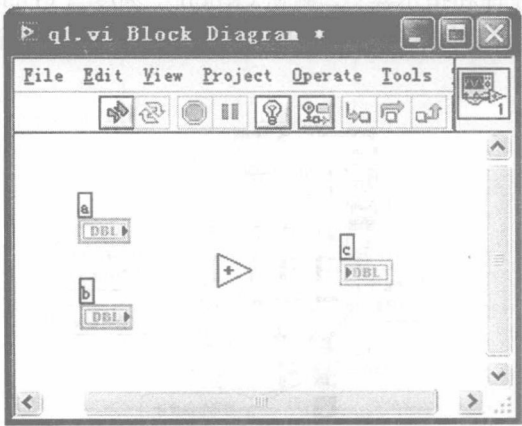


图 1.18 “加”函数的放置

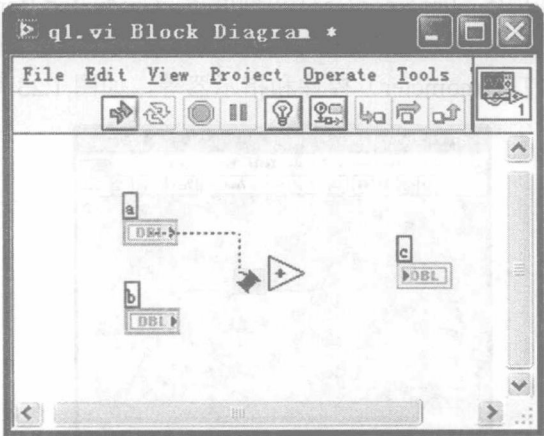


图 1.19 连线过程

提示：如果需要删除错误连线，则单击准备要删除的连线后按 Delete 键即可。当然，LabVIEW 提供了删除程序框图中的所有无效连线的快捷键 Ctrl+B，使用时很方便。

步骤 3：运行 VI

返回到 VI 前面板窗口，在前面板的工具条中，与运行有关的工具按钮如图 1.21 所示。

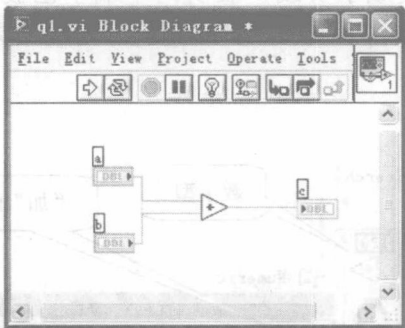


图 1.20 连线结果

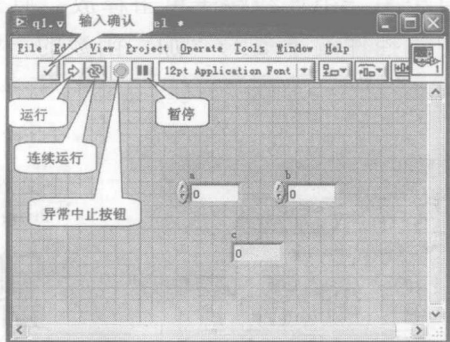


图 1.21 前面板中与运行有关的工具按钮

当鼠标指向前面板上数字控件 a 的增减按钮 (如图 1.21 中所示) 时，鼠标将变成手型 ，单击增减按钮，则 a 的数值会相应地增加或减小。而当鼠标变成 “I” 型并且闪烁时，可向其内直接输入数据。完成控件 a 的数值设置后，回车或在空白处单击，或单击图 1.21 中的 “输入确认” 按钮，均可确认输入有效。

相同的方法，完成对数字控件 b 输入数值的设定后，就可单击图 1.21 中的 “运行” 按钮运行该 VI。可以看到，数字指示器 c 中的值是数字控件 a 和 b 数值之和，即 $c=a+b$ 。当 $a=3$ ， $b=5$ 时，运行后前面板如图 1.22 所示。

步骤 4：保存该 VI

选择菜单 File | Save 选项或按快捷键 Ctrl+S 均可打开保存界面，与 Windows 下文件的保存方式基本一致，需要注意的是，文件扩展名为 .vi。

至此，第一个 VI，求两数之和 $c=a+b$ 的虚拟仪器制作完成。

在前面板中的数字控件选板中，除数字控件和数字指示器以外，还有：①滑动条 (Slide)、进程条 (Progress Bar) 和刻度条 (Graduated Bar) 控件和指示器；②旋钮 (Knob)、刻度盘 (Dial) 控件和指示器；③仪表 (Meter)、表头 (Gauge) 控件和指示器；④容器 (Tank)、温度计 (Thermometer) 控件和指示器等，如图 1.23 所示。

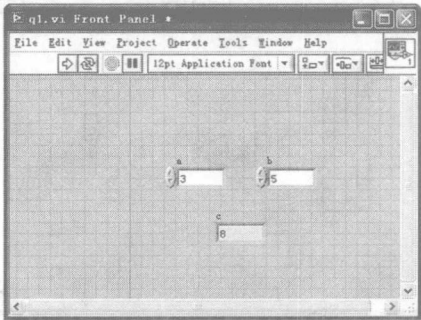


图 1.22 $c=a+b$ 前面板运行结果

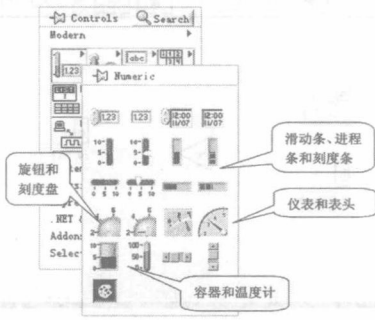


图 1.23 前面板中的其他数字控件和指示器

提示：对于已经放置在前面板上的控件以及在程序框图中与之对应的控件端子，可以在其上右击，在弹出的快捷菜单中选择 **Change to Indicator**，使其变成指示器；相反，对于已经放置在前面板上的指示器以及在程序框图中与之对应的指示器端子，可以在其上右击，在弹出的快捷菜单中选择 **Change to Control**，使其变成控件。这对于控件和指示器放置错误后的修改提供了方便。另外，在前面板的对象上右击，在弹出的快捷菜单中选择 **Replace** 菜单，为对象放置错误后的替换提供了极大的灵活性。

【案例 1.3.2】 旋钮和刻度盘、仪表和表头、容器和温度计的简单运用

(1) 旋钮和刻度盘的简单运用

旋钮和刻度盘一般被用作控件，用于对连续量的输出，例如，阀门的开度等。在前面板中放置好旋钮或刻度盘后，可以看到，旋钮或刻度盘默认的最大指示数值为 10，双击指示数值 10，则指示数值 10 变成反色显示，处于编辑状态。此时改变其值为具体问题需要输入的最大值，则其他指示数值也随即成比例地变化。这样就可以修改旋钮和刻度盘的输入值范围，旋钮数值范围调整示例如图 1.24 所示。

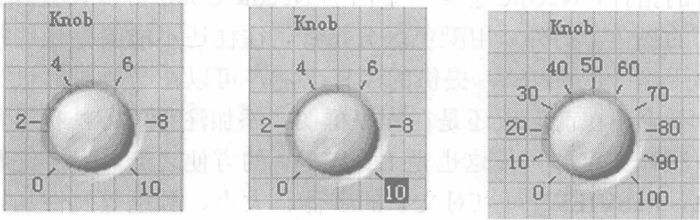


图 1.24 调整旋钮数值范围示例

旋钮或刻度盘在前面板中的大小也可以改变。当鼠标指向旋钮或刻度盘时，其上的四个大小调节句柄可见，拖曳大小调节句柄到适当大小，松开鼠标，即可调整其大小。旋钮大小调整示例如图 1.25 所示。

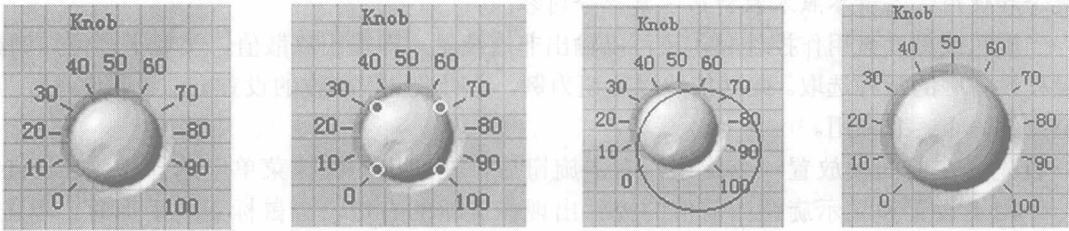


图 1.25 调整变旋钮大小示例

通过拖曳旋钮或刻度盘的指针，可以改变它们作为控件时的输出值。但是在输出一个特定的具体数值时，拖曳改变数值的方法一般难于精确定位。此时，作为控件，数值编辑输入的方法就比较方便。在前面板中的旋钮或刻度盘上右击，在弹出的快捷菜单中选择 **Visible Items | Digital Display**，则一个数字控件出现在旋钮或刻度盘的旁边，此时直接输入具体数值，就可以精确设置旋钮或刻度盘作为控件时的输出值，同时旋钮或刻度盘的指针也指向该数值位置。对旋钮选择数字显示并设置数值为 80 的示例如图 1.26 所示。

默认的旋钮或刻度盘只有一个指针，作为控件时只能输出一个数值。若想使一个旋钮或刻度盘上有两个或更多的指针，可以一次性地输出多个数值，而且输出的每个数值由各自的指针

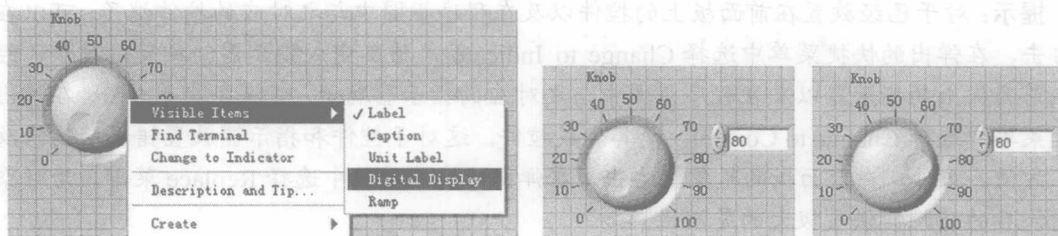


图 1.26 改变旋钮大小示例

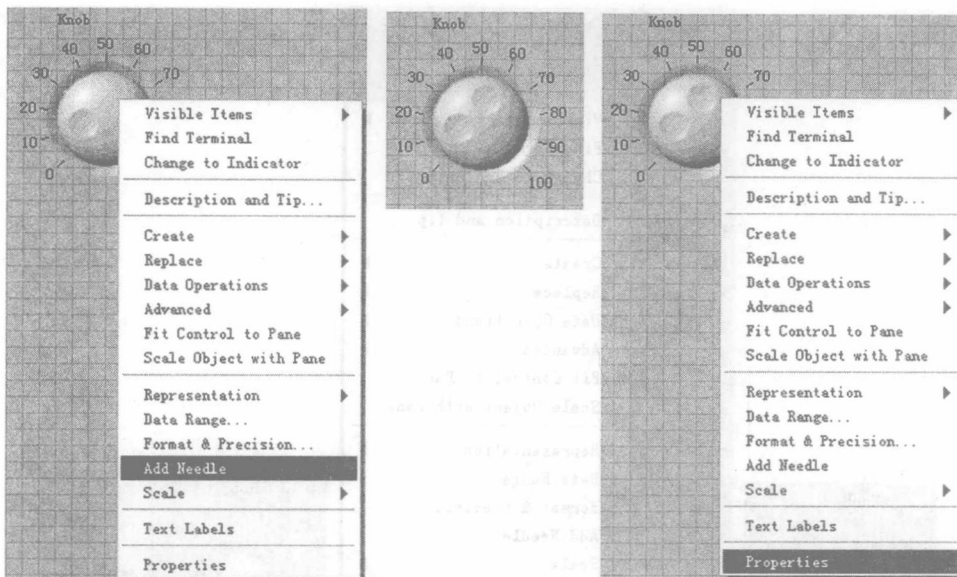
独立控制，具体做法是：右击旋钮或刻度盘，在弹出的快捷菜单中选择 **Add Needle**，就可添加一个指针。相反，右击该指针，在弹出的快捷菜单中选择 **Remove Needle**，就可删除一个已有的指针。由于新添加的指针与默认指针一模一样，不便于区分，为此 LabVIEW 提供了用不同颜色区分不同指针的方法。具体做法是：右击旋钮或刻度盘，在弹出的快捷菜单中选择 **Properties** 选项，在打开的属性设置对话框中，选择 **Appearance** 标签页，其中的 **Needle** 选项卡列出了所有的指针，**Needle** 选项卡下面的 **Needle Color** 可以设置颜色。这样可以为不同的指针设置不同的颜色。另外，用颜色区分指针，往往还不能清楚说明指针的指示量并区分各个指针的具体用途，LabVIEW 提供的注释功能，可以在必要时通过文字注释对有关事项进行详细说明。不论是在前面板还是在程序框图，添加注释的方法极其简单，在需要注释的地方双击，输入注释文字即可，这也是 LabVIEW 的方便之处。对于注释、控件和指示器的标签等文字对象，LabVIEW 可以对文字的字体、大小、颜色等属性加以设置。具体做法是：首先选择被设置的文字，然后在如图 1.14 所示的工具栏中打开文本设置 (**Text Settings**) 下拉菜单，选择其中的 **Size** 用于设置字号，选择其中的 **Style** 用于设置字体，选择其中的 **Color** 用于设置字体颜色。此一系列操作以旋钮为例，相关操作及结果如图 1.27 (a) 至图 1.27 (e) 所示。

说明：有多个指针的旋钮作为控件时，在程序框图中的端子输出的数据类型是簇，有关簇的概念和使用方法请参照本教材第 3 章部分内容。

当旋钮或刻度盘用作控件时，还可以输出非连续量，即输出离散值，类似于档位，例如，多量程设备中的量程选取。此处依然以旋钮为例，说明多量程档位的设置方法。假设设置四个档位：关、I、II 和 III。

首先，在前面板放置一个旋钮，右击旋钮后，在弹出的快捷菜单中选择 **Text Labels**，则一个用于设置和显示旋钮当前值的控件出现在旋钮的右边。当鼠标变成手形时，单击这个设置和显示旋钮当前值的控件，发现当前控件的值只能选取 **min** 或 **max** 其中之一，其他值不能选取，类似于 C 语言中的枚举型数据类型（本章后面也要讲解 LabVIEW 的枚举型数据）。拖曳旋钮的指针依然可以选取其他值，但目前只能显示 **min** 或 **max** 其中之一。如图 1.28 所示。

接着，右击旋钮，在弹出的快捷菜单中选择属性设置 **Properties** 选项，打开属性设置对话框。在 **Text Labels** 选项卡中，在 **Text Labels** 栏中编辑输入如图 1.29 所示的：关、I、II 和 III 四个档位。在 **Data Range** 选项卡中，首先设置数据格式，即单击 **Representation** 后选取无符号 8 位数 U8，然后去除 **Use Default Range** 选项前的对勾，自行设置最大、最小值以及单步增量值。如图 1.30 所示，分别为 0、3 和 1。即最小值为 0，最大值为 3，单步增量为 1。完成后单击 **OK** 确认返回前面板。

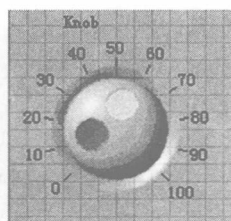


(a) 给旋钮添加一个指针及其结果

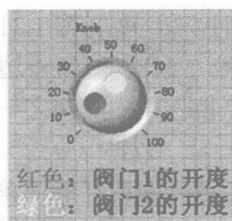
(b) 打开旋钮属性设置对话框



(c) 改变旋钮指针的颜色



(d) 改变颜色后的旋钮



(e) 给旋钮添加注释并对注释格式进行设置

图 1.27 旋钮指针的添加及属性设置、注释文本的添加及格式设置