# Windows 应用程序设计
## ——原理、方法和技巧

张国峰　编著

- 多窗口技术
- 选单和对话框
- 键盘和鼠标输入
- 图形输出
- 面向对象的方法
- 适用于 Windows 3. x 版本
- 七十八个优选实用程序

電子工業出版社

# Windows 应用程序设计
# ——原理、方法和技巧

张国峰　编著

电子工业出版社

(京)新登字 055 号

## 内容提要

本书讲述用 C 开发 Windows 应用程序的方法和有关技巧。本书一开始就介绍了 Windows 面向对象的特征，在后续的各章中详细地讨论了设计 Windows 应用程序用户界面的各个方面，其内容包括：窗口、控制、对话框、选单、光标、图标和位图等。本书还对图形输出、键盘和鼠标输入、计时器和动态链接库等有关程序设计技术进行了详细的讨论。书中给出了由作者编写的大量的程序设计实例，其中优选的、完整的程序实例共七十八个，有很强的实用性，可供读者借鉴。为便于读者使用本书，所有示例被集中存放在一张 5.25 英寸的软盘上。

读者通过书中完整的七十八个程序设计的实例，可学到许多程序设计的技巧。该书是 C 程序员学习 Windows 应用程序设计的一本不可多得的参考书，也可作为高等院校及有关培训班的教材。

Windows 应用程序设计
——原理、方法和技巧
龚国峰 编著
责任编辑：王昌铭
特邀编辑：王素文

*

电子工业出版社(北京市万寿路)
电子工业出版社发行　各地新华书店经销
河北省望都县印刷厂印刷

*

开本：787×1092 毫米　1/16　印张：35.5　字数：900 千字
1994 年 2 月第 1 版　1994 年 2 月第 1 次印刷
印数：8000 册　定价：25.80 元
ISBN7-5053-2258-3/TP・613

# 前　言

Microsoft Windows 是在 MS-DOS 操作系统上运行的一个基于图形的多任务和多窗口操作环境。在该环境下运行的所有应用程序都具有相似的外观和命令结构，从而使应用程序更便于用户使用，同时，用户也很容易掌握新的应用程序的使用方法。

当在 MS-DOS 上设计程序时，一旦遇到图形程序设计问题，程序的设计就变得复杂起来，并且所设计的程序极度依赖于设备，因为 MS-DOS 是一个基于字符的操作系统，它对图形仅有很有限的支持，图形程序或多或少地要和硬件直接打交道。另外，当程序员想在自己设计的程序中加入窗口和选单（又称菜单）以增强程序界面的吸引力时，程序员需要花费大量的时间来编写它们。

Microsoft Windows 的出现对 MS-DOS 程序员来说是一个激动人心的事件，它使 MS-DOS 程序员真正进入了图形应用世界。Microsoft Windows 提供了一个丰富的应用程序设计接口（API——Application Programming Interface）集合，使程序员可以很方便地编写使用窗口和选单的应用程序。更重要的是，它允许应用程序以设备无关的方式与显示器、键盘、鼠标器、打印机和系统时钟等各种输入和输出设备交互，这种方式保证了同一应用程序可以运行于各种硬件配置的计算机平台上，并且有相同的呈现形式。

Windows 应用程序的特点是友好的用户界面和多任务能力，本书介绍了 Windows 的多任务特点和与此有关的程序设计问题。有关应用程序之间数据交换的程序设计技术将在以后进行深入讨论。

与图形用户界面有关的程序设计技术是本书讨论的重点内容，这些内容包括：

- 窗口对象，包括各种控制和对话框
- 选单
- 光标、图标和位图
- 图形输出，包括打印机图形输出
- 动态链接库，包括如何在库中实现用户定制的控制

本书也对与计时器、键盘和鼠标器的输入有关的程序设计技术进行了详细的讨论，结合本书内容的需要，还对 Windows 的实现原理和内存管理作了相应的说明。

Windows 的程序模式是消息驱动的，即程序与用户的交互以及程序之间内部的相互作用都以消息的形式反映出来。Windows 的许多地方明显地受到面向对象的软件概念的影响。在进行 Windows 程序设计时，程序员是在使用非面向对象的语言（例如 C）进行面向对象的程序设计。本书的第一章介绍了这方面的内容，并在后续各章的示例程序中对此作了进一步阐述。可以说，第一章所介绍的内容是从事 Windows 程序设计必须了解和掌握的基础知识。

本书在内容和示例程序的选择上作了精心的安排，主要介绍 Windows 3.1 的程序设计技术，在书中，如果使用术语 Windows 3.0，则这些内容也适用于 Windows 3.1；如果使用术语 Windows 3.1，则这些内容只适用于 Windows 3.1。本书的所有示例程序使用 Borland C/C++ for Windows 3.1 编译通过，并在 Microsoft Windows 3.1 上进行了测试。本书适宜于 C 程序员

学习 Windows 程序设计所用，也适合于用作 Microsoft Windows 程序设计培训班的教材。

电子科技大刘乃琦教授对该书的初稿进行了认真的审阅，并验证了部分程序，提出了一些宝贵意见。在此对他们表示衷心的感谢。

在本书编写过程中，作者得到了许多朋友的热心支持和帮助。李大伟同志为书稿的整理工作倾注了大量的精力，刘义林、喻洁、张朝辉、赵奇胜、扬辉、沈杭、徐保文和张慧芬等同志也参与了书稿的整理工作，希望公司李振格同志为作者提供了部分参考资料，在此作者对他(她)们表示衷心的感谢。作者也诚心期望广大读者对书中的错误提出批评意见。

作者

1993 年 10 月于北京航空航天大学

# 目　录

# 第一章 概 述

Microsoft Windows 是在 MS-DOS 操作系统上运行的一个操作环境。对用户而言，Windows 提供了一个基于图形的多任务、多窗口的操作环境，在这个环境下不但可以运行为 Windows 编写的应用程序，而且还可以运行一些现有的为 MS-DOS 设计的应用程序。特为 Windows 设计的应用程序都具有一致的外部特征和命令结构，因此，它比 MS-DOS 程序更易于为用户使用。对于程序开发者而言，Windows 提供了丰富的库函数，以便于开发者实现选单、会话框、滚动杠以及为组成用户友好的界面所需的其他元素。同时 Windows 基于消息的程序结构也方便了程序之间的通信，并使设计者以与设备无关的方式处理键盘、鼠标器、显示器和打印机等。Windows 还包括一个用途广泛的图形程序设计语言（称其为 GDI），很容易在应用程序中实现多种字体的排版和输出，并以一种与设备无关的方式处理图形，这保证了同一应用程序在不同的硬件配制下得到相同的结果。本章首先概要介绍 Windows 的发展历史，然后从用户角度和程序员角度分析 Windows 的特征和益处，并给出程序实例来说明 Windows 程序的结构。

## 1.1 Windows 的简要历史

Windows 最初由 Microsoft 公司在 1983 年 11 月宣布，1985 年 11 月推出了第一个公开发行版本，即 1.01 版。此后两年，1.01 版进行了几次修改以满足国际市场的要求，并增加了一些显示器和打印机驱动程序。1987 年 11 月发行的 Windows 2.0 版在用户界面上做了些改进。例如：重叠式窗口的引入，还增强了键盘和鼠标器接口，特别是增强了选单和对话框的功能。

Windows 2.0 之后不久推出的 Windows/386 使用了 80386 微处理器的虚拟 8086 模式，使 MS-DOS 程序实现了窗口化和多任务化，后来推出的 Windows 2.1 被相应地改称为 Windows/286。

Windows 3.0 是在 1990 年 5 月推出的，早期的 Windows/286 和 Windows/386 在新版本中被融为一体。Windows 3.0 最重要的变化是支持 80286/80386 微处理器的保护模式下的操作，这使 Windows 和 Windows 的应用程序可以访问多达 16MB 的物理内存。在此一年后推出的 Windows 3.1 对 Windows 3.0 作了一些改进，主要是增加了支持对象链接和嵌入技术，这种技术使用户可以方便地集成各种应用程序；另外，Windows 3.1 将多媒体技术也作为系统的一部分进行支持。

## 1.2 面向对象的用户界面

随着 1984 年 Machintosh 计算机的出现，Apple 公司的用户界面成为计算机发展的潮流并普及起来。Apple 的用户界面起源于 70 年代 Xerox 的 Palo Alto 研究中心（PARC）所做的工作，现在人们通常将这种用户界面称为图形用户界面。图形用户界面已在许多产品中出现，例如 Apple 公司的 Macintosh、MS-DOS 上运行的 Microsoft Windows、OS/2 操作系统上运行的 Presentation Manager（PM）和 Unix 操作系统上运行的 X-Windows 等。这些图形用户界面都具有以下

特征：

- 附属的用户输入设备，通常是一种如鼠标器这样的指点器。
- 在指点器控制下在屏幕上显现或从屏幕上消失的(即弹出式的)选单。
- 以图形方式显示计算机正在完成什么任务的窗口。
- 表示文件、目录和应用程序的图标。
- 让程序员和用户告诉计算机做什么事情和的对话框、按钮、滚动杠、检取框等。

图 1-1 是 Microsoft Windows 的一种典型的图形用户界面。

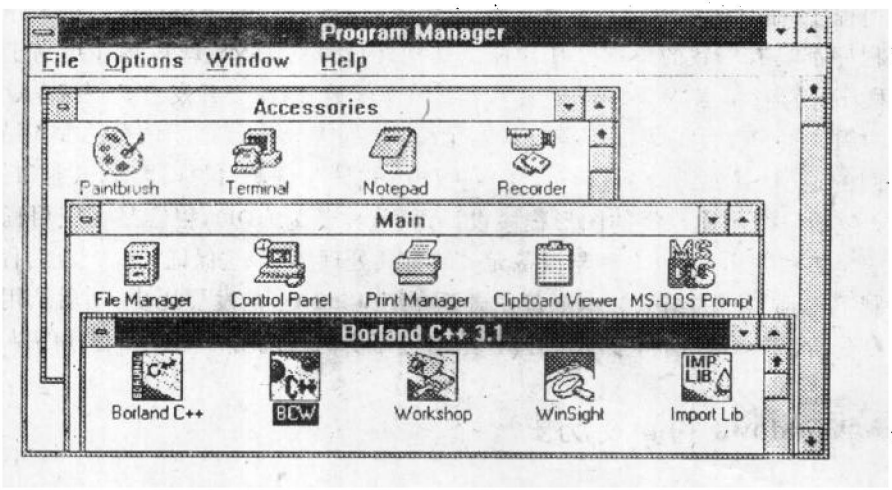


图 1-1 Microsft Windows 的 Program Manager 应用程序的一个窗口

在某种程度上，图形用户界面的用户环境和开发环境都是面向对象的。特别是在用户界面的外观上，面向对象是相当活跃的。凡是可操作的事物或实体都是对象，在图形用户界面中，对象主要指的是屏幕上显示的可视图象。

从用户角度看，屏幕上含有各种直观的可视对象，如窗口、按钮、滚动杠和图标等。用户可通过键盘或鼠标器与这些对象进行交互：选择一个对象，并通过操作该对象来操作程序。通过屏幕上的可视对象，面向对象的用户界面给用户提供了一种对程序的直接操作能力，以控制计算机的行为。

面向对象的图形界面非常适合于人，用户需要真正可操作的对象来加强应用程序的真实性和可视性，可操作对象是对用户实际处理的对象的最好的模拟方式之一。

面向对象的用户界面的最大优点就在于为用户提供了一个非常类似于真实世界而非计算机程序的易于操作的环境，从而使用户专注于任务本身，而不为所使用的工具分心。使用面向对象的图形用户界面的用户完全可以仅仅依赖于常识、经验以及日常生活中有关空间推理的能力来使用计算机，而不需要事先对计算机有专门的知识。

## 1.3 用户界面的构件

Windows 的概念和术语可分两大类：一类是 Windows 的可视特征，如选单和图标等；另一类是幕后操作，如传递消息等。为了让 Windows 应用程序的开发者相互之间进行有效的通信，所有的 Windows 特征都给出了名字和相关的用法。本节介绍与 Windows 的可视特征有关的一

些术语及其相关概念。

### 1.3.1 窗口

窗口是屏幕上与一个应用程序相关联的矩形区域，它是用户与产生该窗口的应用程序之间的可视界面。对应用程序来说，窗口是该程序控制下的屏幕上的一个矩形区域，应用程序创建并控制窗口的所有方面。当用户启动一个应用程序时，一个窗口就被创建。每当用户操作窗口中的对象时，程序就有所响应。图 1-2 给出了一个 Windows 窗口。

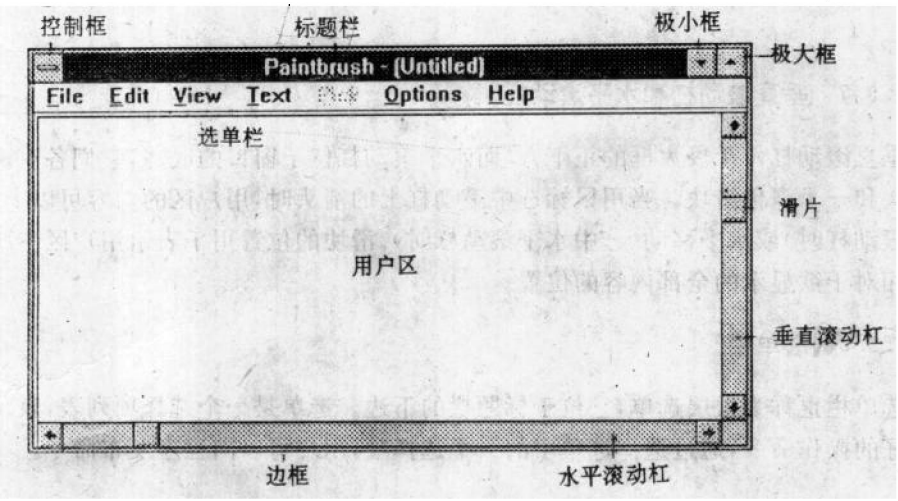


图 1-2 Windows 窗口的基本组成

### 1.3.2 边框

绝大多数窗口都有一个环绕它的边框，边框不仅作为窗口的边界，它也用来指明窗口的状态，即窗口是否是一个活动窗口。有关活动窗口的概念在 1.4.3 节介绍。当用户将光标定位在边框上，然后按下鼠标键并移动鼠标时，窗口的大小也随着发生改变。

### 1.3.3 标题栏

标题栏位于窗口的顶部，其中显示的文本信息用于标注程序，一般是应用程序的名字，这便于用户知道哪个应用程序正在运行。标题栏的颜色反映一个窗口是否是一个活动窗口。用鼠标两次点按标题栏可以使窗口充满整个屏幕。若将光标定位于标题栏上，按下鼠标键并拖动鼠标器，则可以在屏幕上移动窗口。

### 1.3.4 控制框

控制框是位于窗口左上角的一个白色小框，其中有一个暗色的平滑矩形，用鼠标点一下控制框（或按 ALT+SPACE 健）就显示出系统选单。系统选单提供标准的应用程序选项，例如：Restore（恢复窗口的原有大小），Move（使其可以通过键盘上的光标键来移动窗口的位置），Size（使用光标健来改变窗口的大小），Minimize（将窗口缩小为图标），Maximize（使窗口充满整个屏幕）和 Close（关闭窗口）。

### 1.3.5 极小框和极大框

每个窗口右上角都有两个垂直的箭头，其中向下的箭头所在的方框是极小框，用鼠标器选中该框就可以使窗口缩小为图标。向上的箭头所在的方框是极大框，当用鼠标器选中该框时，就可以使一个应用程序的窗口充满整个屏幕。

### 1.3.6 用户区

图 1-2 中最大的一块空白矩形区域称为用户区，该区域用于显示应用程序的输出。应用程序负责绘制用户区，而且只有和该窗口相关联的应用程序才能向这个窗口的用户区中输出信息。

### 1.3.7 垂直滚动杠和水平滚动杠

垂直滚动杠位于极大框的正下方，而水平滚动杠位于窗口的底部，它们各有两个方向相反的箭头和一个深色滑块。当用鼠标选中滚动杠上的箭头时，用户区的内容可以上下移动(选中垂直滚动杠时)或水平移动(选中水平滚动杠时)。滑块的位置用于表示用户区中所显示的部分内容相对于欲显示的全部内容的位置。

### 1.3.8 选单栏

选单栏也称为顶层选单，它位于标题栏的下边。选单是一个选择项列表，表示程序已预先确定好的操作命令，通过选择选单中的一个选择项，用户可以向程序发布命令。

### 1.3.9 图标

对用户来说，图标为一个小的图象，当一个应用程序的主窗口缩小到最小时，它代表该应用程序，如图 1-1 所示。

### 1.3.10 光标

Windows 的光标是一个图形符号，而不像 MS-DOS 中是一条下划线。光标随着指点器(例如鼠标器)位置的变化而在屏幕上移动，它用于指示用户被鼠标操作启动的动作发生在何处。通过光标形状的变化，程序向用户反馈现在允许用户进行何种操作或程序正在执行什么操作。

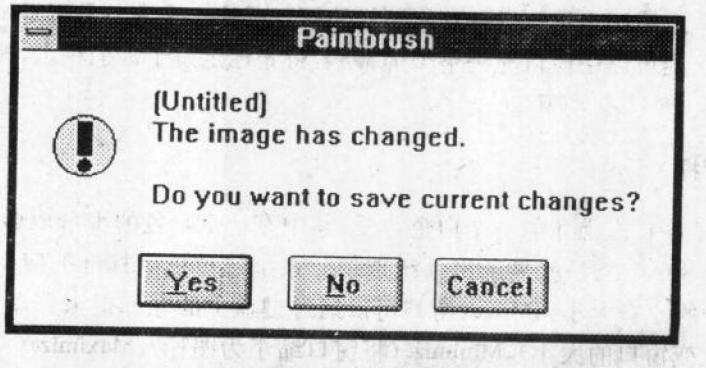


图 1-3 Windows 应用程序显示的一个信息框

### 1.3.11 信息框

信息框是一个弹出式窗口，其中包括一个标题、一个图标和一条信息。图 1-3 是应用程序显示的一个信息框。当应用程序需要从用户那里获取应答，以便决定下步的操作时，应用程序在屏幕上显示一个信息框，在接到用户的响应之后，信息框消失。在图 1-3 中，标识为 YES、NO 和 CANCEL 的图形符号被称为按钮，用户通过选择其中的一个按钮对该信息框进行响应。

### 1.3.12 对话框

对话框是一种具有特殊使用目的的窗口，对话框主要用于从用户接收输入，通过使用对话

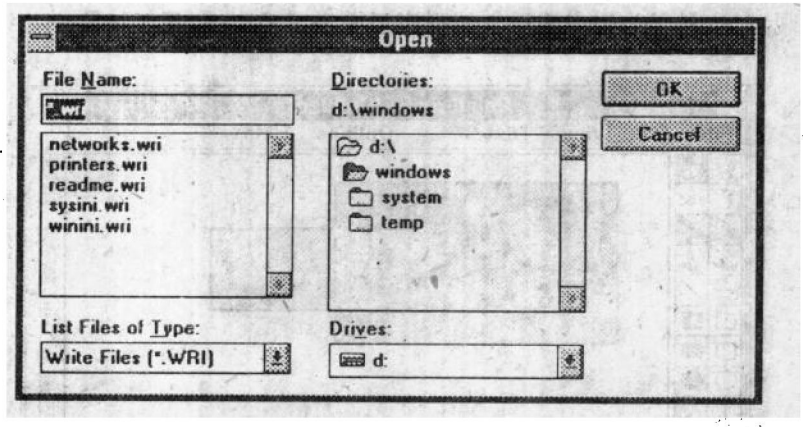


图 1-4. 打开文件用的一个对话框

框，应用程序一次可以从用户那里接收一组信息，图 1-4 是打开文件时用的对话框。

## 1.4 Windows 的功能

Windows 操作环境对用户和程序员都具有很大的优越性，其主要功能体现在标准的图形用户界面、多任务和硬件独立性。本节主要从用户的角度介绍这些功能。

### 1.4.1 标准的图形用户界面

Windows 提供的三大主要功能中，标准的图形用户界面最引人注目，而且对用户也是最重要的。从前面给出的几幅 Windows 界面的图示中可以看出，每个程序都有一个窗口或图标来代表，程序通过标题栏来标识，许多操作可以通过使用鼠标器在窗口中选择对象来进行。由于象选单、滚动杠、对话框和图标等可视对象都是通过使用 Windows 的内部例程来创建的，因此，在不同的 Windows 程序中，所有的 Windows 程序呈现给用户相同的形象。操作可视对象的方法也一样，这样，用户不再需要花费很多的时间来学习新程序的使用。

### 1.4.2 直接操作特性

图形用户界面提供给用户的不仅仅是图案美丽、颜色鲜艳的屏幕显示，更主要的是，它是为应用程序所设计的现实世界在屏幕上构筑了一个虚拟世界，以模拟用户与现实世界之间的

交互。

早些时候，显示器仅用于反馈用户从键盘上输入的字符。在图形用户界面中，视频显示器本身也成为用户的输入手段之一。显示器上以图标形式显示出各种图形对象，也可以显示诸如按钮或滚动杠等虚拟输入设备。使用如鼠标之类的指点输入设备，用户可以在屏幕上直接操纵这些对象：拖动图形对象，选取按钮，使用滚动杠来滚动窗口中显示的内容等。这样，用户与程序之间的交互变得更加紧密，不再是单一的只使用键盘输入信息至屏幕显示，而是用户直接与屏幕上的对象进行交互。

图 1-5 是一个图形编辑程序所显示的窗口，窗口中显示了各种可用的绘图工具，其中有一支笔，用户使用鼠标选择这支笔，设置笔的颜色和宽度，并用鼠标拖动这支笔，这时就在用户区中绘制出特定宽度和颜色的线条，这逼真地模仿着我们实际进行手工绘图的情形，因而这样的应用程序直观自然，易于为用户使用。该程序还提供了橡皮擦，用于擦除一些线条或图形。

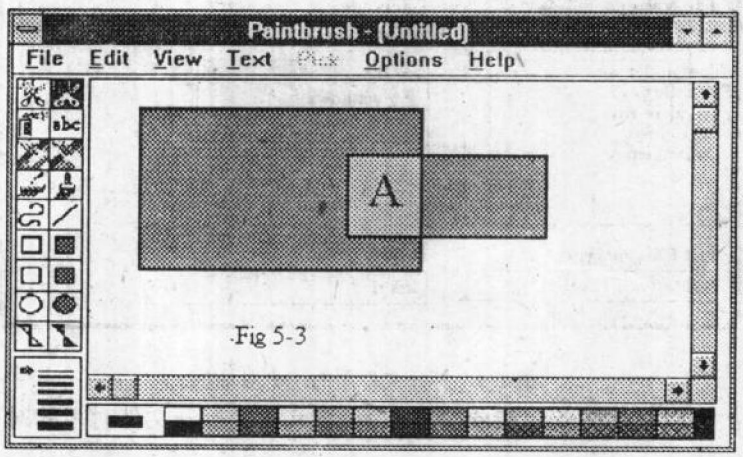


图 1-5 PaintBrush 图形编辑程序显示的一个窗口

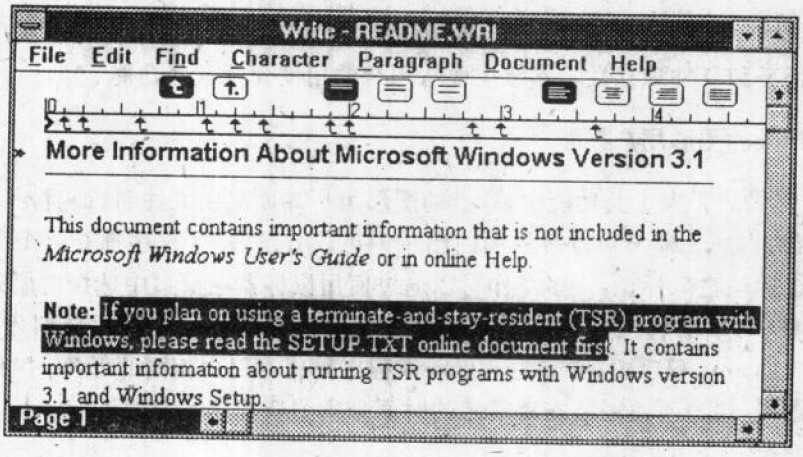


图 1-6 Write 应用程序的一个窗口

图 1-6 是 Write 应用程序的一个窗口，该窗口中显示了各种直观的排版命令，光标在不同的区域中显示不同的形状，以指示可允许的操作。当在一行上按下鼠标按键并拖动鼠标时，用

户选择了一块区域，这块区域以醒目的形式被标识出来，这类似于我们用笔在书上构画出重点内容的情形。

使用直接操作技术，可以为一个程序设计各种输入方法，这些方法通常表现为现实世界中的某种控制面板的虚拟屏幕模式，用户在其上使用直接操作来调整或设置控制，被用于输入的虚拟控制面板模式为用户提供了各种能用其工作的模拟控制，如图 1-7。在设计虚拟控制之前，应仔细考虑一下实际的控制面板是如何设计的。在程序中，这些控制应按与人们真正使用的控制，例如游标滑动、刻度盘旋转以及按钮的按动同样的方式进行处理，例如，为滑动游标，用户应选择游标控制并向所希望的方向移动它；为转动刻度盘，用户应选择刻度指针并移动它，刻度盘指针应围绕该刻度盘控制的中心移动。按钮应象任何无线电设备中使用的按钮装置那样来处理。

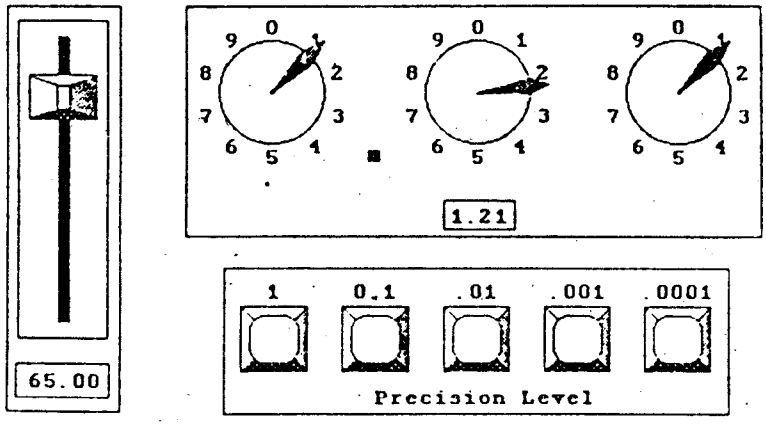


图 1-7　一个实际控制面板的屏幕模式

在设计任何控制时，都要记住一个原则，这就是要给用户反馈信息。例如，在图 1-1 中，当一个操作选中一个图标对象时，该对象被突出显示出来；在图 1-5 中，当笔被移动到用户区之外时，笔消失，而代之以普通形状的光标；在图 1-4 中，当将光标移入按钮上时，按钮对象出现一个虚框，以表示如果进行选择操作，所选择的对象是该按钮。

### 1.4.3　多任务

Windows 也是一个多任务操作环境，它允许用户同时运行多个应用程序，或在一个程序中同时做几件事情。图 1-8 说明了在 Windows 同时运行多个应用程序的情形，每个程序的窗口在屏幕上占据一块矩形区域，窗口是重叠的。用户可以在屏幕上移动这些窗口，或在不同的应用程序之间进行切换，并可以在程序之间动态地进行数据交换。

当一个应用程序被从磁盘上装入内存时，Windows 为该程序分配所需的内存空间，以存储该程序的数据和代码，并存储一些其他的管理信息以建立起该程序的一个运行环境。在内存中正运行的程序被称为(磁盘中该应用程序的)一个实例。同一个程序的多个实例可以同时在内存中运行，图 1-9 说明了这种情况。

虽然同一时刻在计算机中可以运行多个应用程序，但其中只有一个程序处于激活状态(或称为活动状态)，这可以通过与该程序相关联的窗口观察出来，它的外观和屏幕上正显示的其他窗口不一样：它具有醒目的标题栏和边框。一个处于激活状态的应用程序是指当前能够接收