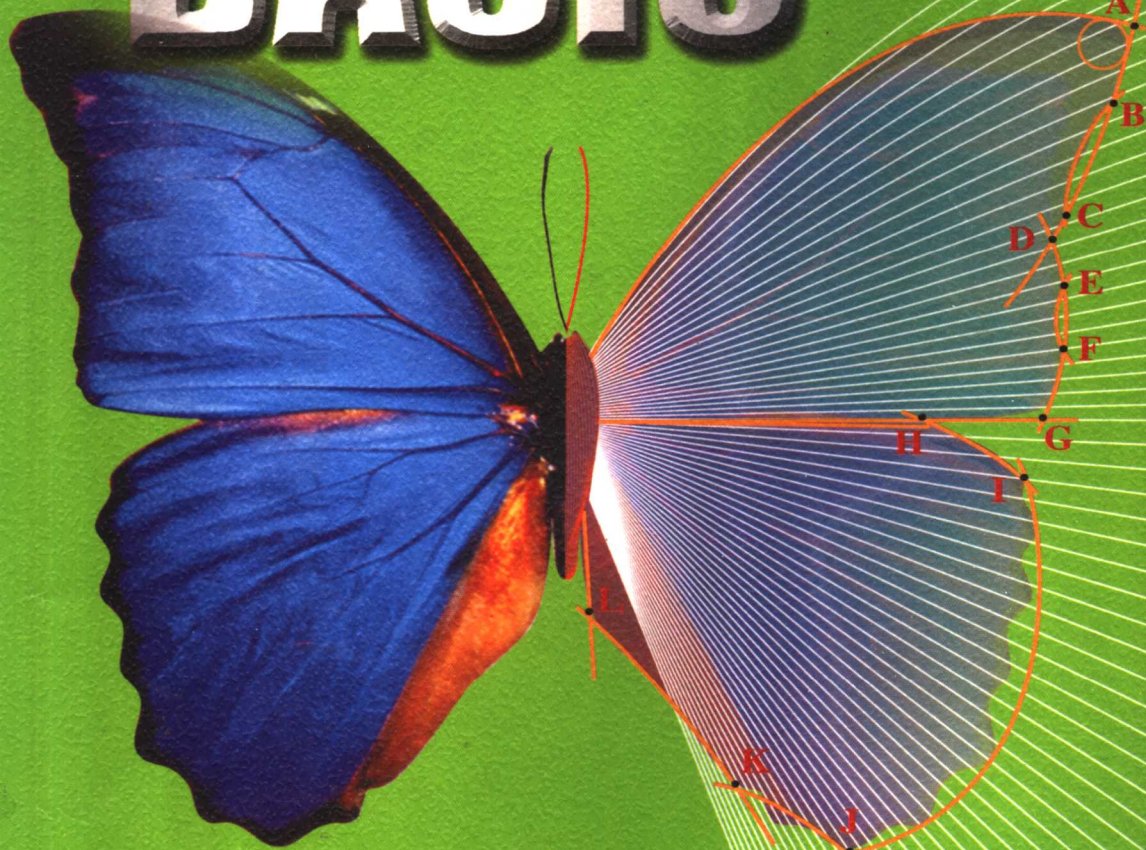


工程图学系列教材

Visual BASIC

计算机绘图
实用技术

谢步瀛 龚沛曾 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

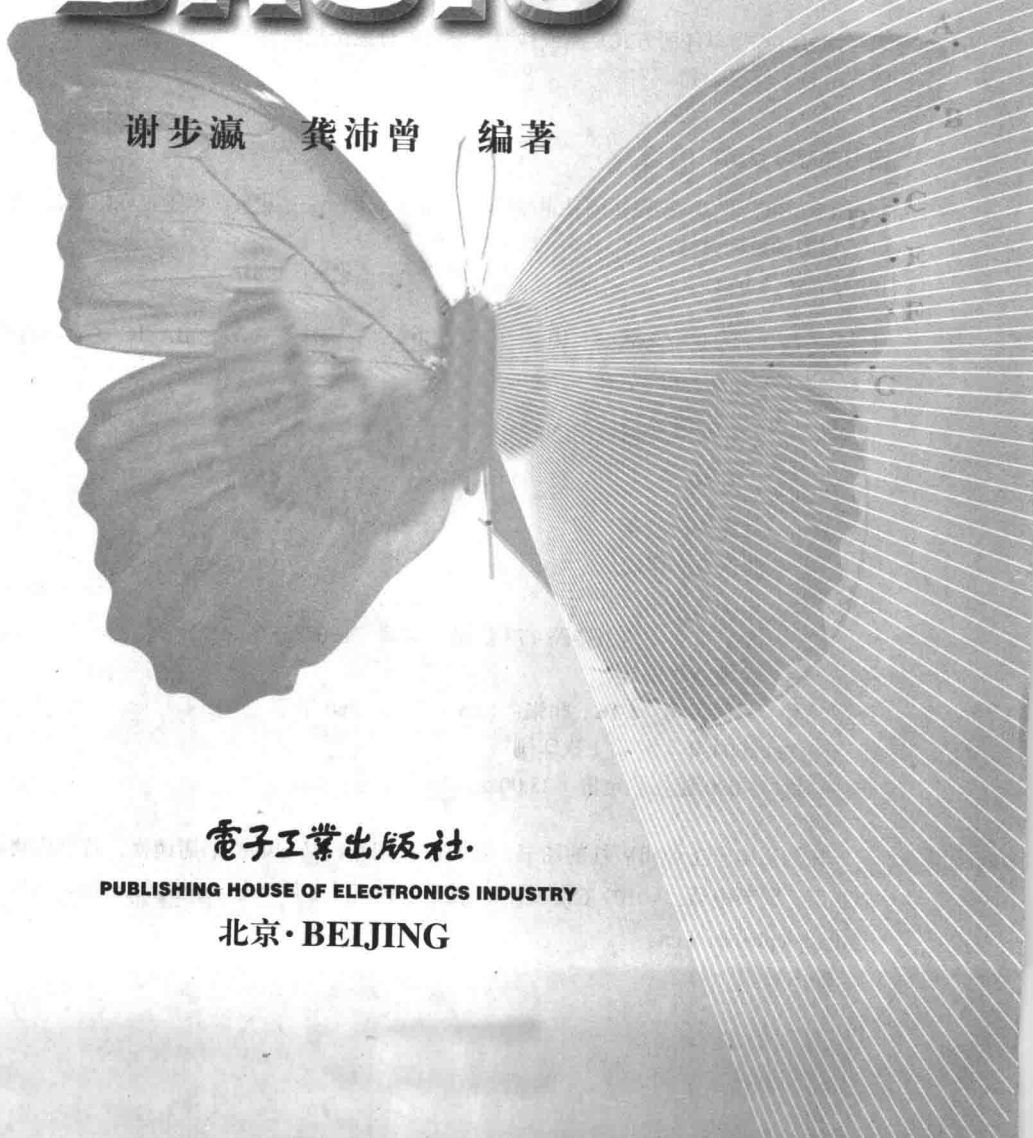
<http://www.phei.com.cn>

工程图

Visual BASIC

计算机绘图
实用技术

谢步瀛 龚沛曾 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

北京·BEIJING

内 容 简 介

本书是《工程图学系列教材》中介绍 Visual BASIC 计算机绘图实用技术的专著。

全书分教学篇和实验篇两大部分。教学篇分 5 章,第 1 章是 VB 计算机绘图技术简介,第 2 章介绍计算机绘图的数学基础,第 3 章较详细地介绍交互式绘图系统设计中的一些数据结构和实用算法,第 4 章以力学问题为例介绍 Visual BASIC 计算机绘图技术的应用,第 5 章以机械工程问题为例介绍 Visual BASIC 计算机绘图技术的应用;实验篇为教学篇配置了相应的实验内容。

本书既适用于作为各大专院校本科生、研究生的教材,也可供科研院所从事计算机软件开发的专业人员作为参考资料。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

Visual BASIC 计算机绘图实用技术/谢步瀛,龚沛曾编著. —北京:电子工业出版社, 2004.4

(工程图学系列教材)

ISBN 7-5053-9747-8

I. V… II. ①谢…②龚… III. ①自动绘图—高等学校—教材②BASIC 语言—程序设计—高等学校—教材 IV. TP391.72

中国版本图书馆 CIP 数据核字(2004)第 018272 号

责任编辑:刘宪兰 李 岩

印 刷:北京大中印刷厂

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

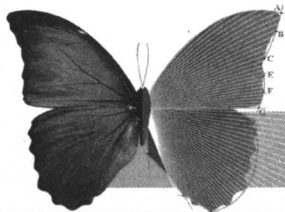
经 销:各地新华书店

开 本:787×980 1/16 印张:17.5 字数:360 千字

印 次:2004 年 4 月第 1 次印刷

印 数:5 000 册 定价:23.00 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话:(010) 68279077。质量投诉请发邮件至 zlts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。



前言

Visual BASIC 是 Windows 环境下优秀的程序设计工具,是面向对象的集成开发系统。它继承 BASIC 简单、高效、易学易用的功能,并且融入面向对象、可视化、事件驱动等先进的软件开发技术,如今有大量程序员在使用 Visual BASIC 开发各种类型的应用程序。

本书是《Visual BASIC 计算机绘图基础》的姐妹篇。在该书的基础上,本书着重介绍计算机绘图的实用技术。

全书分教学篇和实验篇两大部分。教学篇分 5 章,第 1 章是 VB 计算机绘图技术简介;第 2 章介绍计算机绘图的数学基础,鉴于《Visual BASIC 计算机绘图基础》已较详细介绍了计算机绘图的数学基础,本书只针对后续章节要用到的数学问题做了一些补充;第 3 章较详细地介绍交互式绘图系统设计中的一些数据结构和实用算法;第 4 章以力学问题为例介绍 Visual BASIC 计算机绘图技术的应用;第 5 章以机械工程问题为例介绍 Visual BASIC 计算机绘图技术的应用。

本书既适用于作为各大专院校本科生、研究生的教材,也可供科研院所从事计算机软件开发的专业人员作为参考资料。

本书由谢步瀛、龚沛曾编写。书中部分例子选自学生的作业,在采用时做了适当的修改。

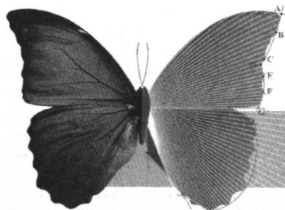
感谢有关专家、教师长期以来对我们工作的关心、支持,也感谢电子工业出版社刘宪兰、李岩编辑对“Visual BASIC 计算机绘图”系列丛书的策划、出版所做的许多工作。由于作者学识所限,缺点和错误在所难免,竭诚欢迎广大读者批评指正。

作者 (xieby@tongji.edu.cn)

2003 年 12 月

2005/6/1

10



目 录

教 学 篇

第 1 章 Visual BASIC 6.0 图形技术简介	1
1.1 坐标系统	2
1.1.1 默认坐标系	2
1.1.2 自定义坐标系	3
1.2 绘图属性	7
1.2.1 当前坐标	7
1.2.2 线宽与线型	7
1.2.3 显示模式	9
1.2.4 颜色	11
1.2.5 填充	13
1.2.6 自动重画	14
1.3 图形控件	14
1.3.1 图形框 (PictureBox) 和图像对象 (Image)	14
1.3.2 直线 (Line) 和形状 (Shape)	15
1.4 图形方法	16
1.4.1 Line 方法	16
1.4.2 Circle 方法	18
1.4.3 PSet 方法	19
1.5 综合绘图举例	20
1.6 动画	22
1.6.1 利用定时器 (Timer)	22
1.6.2 利用延时 (Delay)	28

复习思考题	29
第 2 章 计算机绘图的数学基础	31
2.1 几何元素之间的关系	32
2.1.1 点到线段的距离	32
2.1.2 点到圆弧的距离	35
2.1.3 点到矩形的距离	37
2.1.4 两条线段的交点	38
2.1.5 两个圆的交点	40
2.2 不同线型的绘图	41
2.2.1 虚线元素	41
2.2.2 长画短画线元素	44
复习思考题	47
第 3 章 交互式绘图系统设计	49
3.1 交互式绘图系统的界面	50
3.1.1 窗体与控件	51
3.1.2 菜单命令	51
3.1.3 图形数据和图形数据文件	54
3.2 交互式绘图系统的文件功能	57
3.2.1 建立新图形	57
3.2.2 打开文件	58
3.2.3 保存文件	60
3.2.4 退出系统	61
3.3 交互式绘图系统的绘图功能	62
3.3.1 画直线	64
3.3.2 画圆	65
3.3.3 画圆弧	65
3.3.4 写文字	66
3.4 交互式绘图系统的修改功能	69
3.4.1 选择功能	75
3.4.2 平移	76
3.4.3 复制	78
3.4.4 旋转	80
3.4.5 对称	81
3.4.6 删除	84

3.5	交互式绘图系统的辅助功能	86
3.5.1	捕捉	86
3.5.2	缩放视图	93
3.5.3	正交	95
3.5.4	列表	96
3.5.5	系统启动	99
3.5.6	重画	99
3.5.7	画坐标轴	99
3.6	交互式绘图系统的程序源代码	100
	复习思考题	104
第4章	VB 绘图与力学	105
4.1	平面汇交力系	106
4.1.1	平面汇交力系的原理	106
4.1.2	平面汇交力系的程序模拟	107
4.1.3	平面汇交力系的界面设计	107
4.1.4	平面汇交力系的程序设计	107
4.1.5	平面汇交力系的程序源代码	108
	复习思考题	115
4.2	摩擦	115
4.2.1	摩擦的原理	115
4.2.2	摩擦的程序模拟	115
4.2.3	摩擦程序的界面设计	116
4.2.4	摩擦的程序设计	116
4.2.5	摩擦程序的源代码	117
	复习思考题	122
4.3	弹簧的自由振动	122
4.3.1	弹簧自由振动的原理	122
4.3.2	弹簧自由振动的程序模拟	123
4.3.3	弹簧自由振动程序的界面设计	123
4.3.4	弹簧自由振动的程序设计	124
4.3.5	弹簧自由振动的程序源代码	124
	复习思考题	131
4.4	弹簧滑块碰撞	131
4.4.1	弹簧滑块碰撞的原理	131

4.4.2	弹簧滑块碰撞的程序模拟	132
4.4.3	弹簧滑块碰撞程序的界面设计	132
4.4.4	弹簧滑块碰撞的程序设计	133
4.4.5	弹簧滑块碰撞程序的源代码	133
	复习思考题	137
4.5	悬臂梁受迫振动	137
4.5.1	悬臂梁受迫振动的原理	137
4.5.2	悬臂梁受迫振动的程序模拟	138
4.5.3	悬臂梁受迫振动程序的界面设计	139
4.5.4	悬臂梁受迫振动的程序设计	139
4.5.5	悬臂梁受迫振动的程序源代码	140
	复习思考题	142
4.6	简支梁加载	143
4.6.1	简支梁加载的原理	143
4.6.2	简支梁加载的程序模拟	143
4.6.3	简支梁加载程序的界面设计	144
4.6.4	简支梁加载的程序设计	145
4.6.5	简支梁加载的程序源代码	146
	复习思考题	152
4.7	单伸臂简支梁影响线	153
4.7.1	单伸臂简支梁影响线的原理	153
4.7.2	单伸臂简支梁影响线的程序模拟	154
4.7.3	单伸臂简支梁影响线程序的界面设计	154
4.7.4	单伸臂简支梁影响线的程序设计	155
4.7.5	单伸臂简支梁影响线的程序源代码	156
	复习思考题	164
第 5 章	VB 绘图与机械工程	165
5.1	圆轮摇杆	166
5.1.1	圆轮摇杆的原理	166
5.1.2	圆轮摇杆的程序模拟	167
5.1.3	圆轮摇杆的界面设计	167
5.1.4	圆轮摇杆的程序设计	167
5.1.5	圆轮摇杆的程序源代码	168
	复习思考题	173

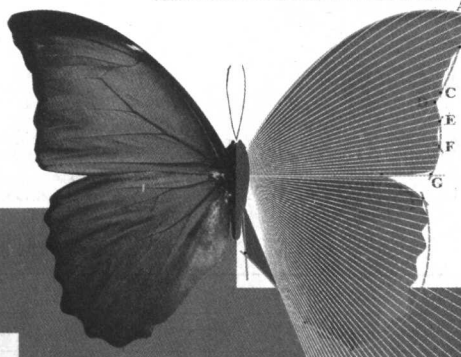
5.2	齿轮传动	173
5.2.1	齿轮传动的原理	173
5.2.2	齿轮传动的程序模拟	173
5.2.3	齿轮传动程序的界面设计	174
5.2.4	齿轮传动的程序设计	175
5.2.5	齿轮传动程序的源代码	175
	复习思考题	179
5.3	旋转式压缩机	179
5.3.1	旋转式压缩机的原理	179
5.3.2	旋转式压缩机的程序模拟	180
5.3.3	旋转式压缩机程序的界面设计	180
5.3.4	旋转式压缩机的程序设计	180
5.3.5	旋转式压缩机的程序源代码	181
	复习思考题	188
5.4	人字式起重机	189
5.4.1	人字式起重机的原理	189
5.4.2	人字式起重机的程序模拟	190
5.4.3	人字式起重机程序的界面设计	190
5.4.4	人字式起重机的程序设计	191
5.4.5	人字式起重机程序的源代码	192
	复习思考题	199
5.5	翻料机	199
5.5.1	翻料机的原理	199
5.5.2	翻料机的程序模拟	200
5.5.3	翻料机程序的界面设计	201
5.5.4	翻料机的程序设计	201
5.5.5	翻料机的程序源代码	202
	复习思考题	208
5.6	步进转轮	208
5.6.1	步进转轮的原理	208
5.6.2	步进转轮的程序模拟	209
5.6.3	步进转轮程序的界面设计	209
5.6.4	步进转轮的程序设计	210
5.6.5	步进转轮的程序源代码	210

复习思考题	215
5.7 十字节传动	216
5.7.1 十字节传动的原理	216
5.7.2 十字节传动的程序模拟	218
5.7.3 十字节传动程序的界面设计	219
5.7.4 十字节传动的程序设计	220
5.7.5 十字节传动的程序源代码	221
复习思考题	234

实 验 篇

实验 1 VB 程序设计及图形技术	235
实验 2 图形技术	245
实验 3 直线、圆和圆弧的求交点运算	249
实验 4 直线、圆和圆弧的相切运算	255
实验 5 立体几何的运算	261
参考文献	267

教学篇



第 1 章

Visual BASIC 6.0 图形技术简介



工程绘图系列教材

本章是编制计算机绘图应用程序的预备知识, 为了便于读者利用 Visual Basic 6.0 (简称 VB) 编制绘图应用程序, 主要介绍 VB 的绘图操作。对于熟悉 VB 的读者可以跳过这一章; 若要更详细地了解 VB, 请参阅有关书籍。

VB 提供了丰富的绘图属性和方法, 可方便地绘制各种图形。

1.1 坐标系

1.1.1 默认坐标系

坐标系的度量单位为标度, VB 提供了 8 个标度, 如表 1.1 所示。标度的属性 ScaleMode 的默认值为 1, 单位为 twip, 这是一个与设备无关的度量单位。

表 1.1 ScaleMode 属性设置

设 置 值	定 标 单 位	说 明
0	用户定义	由程序员定义的单位
1	twip	每厘米 567 twip, 每英寸 1440 twip, 每点 20 twip
2	point (点)	通常用于字体, 每英寸 72 point
3	pixel (像素)	通常用于图像, 代表屏幕分辨率的最小单位
4	character(字符)	水平 120 twip, 垂直 240 twip
5	inch	物理意义上的英寸
6	milimeter	物理意义上的毫米
7	centimeter	物理意义上的厘米

窗体的 Left, Top 属性指示窗体在屏幕内的位置, Height, Width 属性指示窗体的大小。当新建一个窗体时, 窗体属性的默认值 Left=0, Top=0, Height=3600, Width=4800 (单位均为 twip)。

VB 的默认坐标系为左手系, 坐标原点在窗体的左上角, 水平为 x 坐标, 向右递增; 垂直为 y 坐标, 向下递增。窗体 ScaleLeft, ScaleTop, ScaleWidth, ScaleHeight 属性表示窗体的坐标系, ScaleLeft, ScaleTop 表示窗体左上角的坐标, ScaleWidth, ScaleHeight 表示窗体的宽度和高度, 由此可求出窗体右下角的坐标 ScaleLeft+ScaleHeight, ScaleTop+ScaleWidth。

新窗体采用默认坐标系, ScaleLeft=0, ScaleTop=0, ScaleHeight=3195, ScaleWidth=4680。此时, 坐标单位与窗体单位相同, 但右下角能画图的坐标是 (4672, 3187), 如图 1.1 所示, 与窗体默认值比较, 可以看出, 窗体左右两条不能画图的白条共宽 128 twip, 上端蓝条加上不能画图的白条共宽 413 twip。

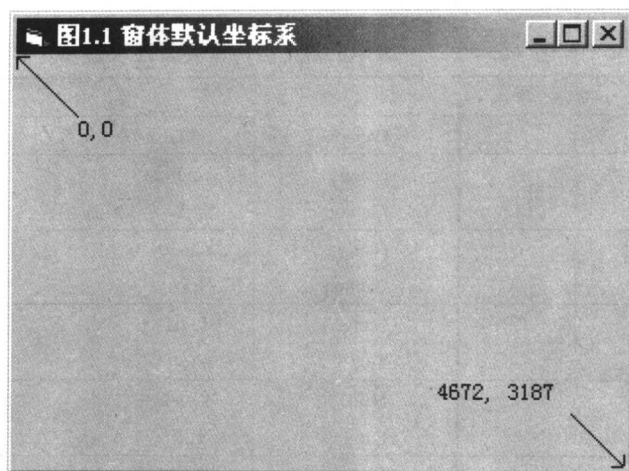


图 1.1 窗体默认坐标系

1.1.2 自定义坐标系

对象的坐标系允许用户自行定义。Scale 方法是建立用户坐标系最方便的方法，其语法如下：

[对象].Scale [(xLeft, yTop) - (xRight, yBottom)]

其中：

对象可以是窗体、图形框或打印机。(xLeft, yTop) 表示对象的左上角的坐标值，(xRight, yBottom) 为对象的右下角的坐标值，均为单精度数值。方括号[]内的内容是可选的，表示不同的用法，以下同。例如，在窗体程序中，不写[对象.]，表示定义窗体坐标系。在模块程序中，不能不写[对象.]，否则系统出错，提示“编译错误：没有适当的对象，方法无效”。VB 根据给定的坐标参数计算出 ScaleLeft, ScaleTop, ScaleWidth, ScaleHeight 的值：

ScaleLeft = xLeft

ScaleTop = yTop

ScaleWidth = xRight - xLeft

ScaleHeight = yBottom - yTop

VB 的默认坐标系为左手系，而常用的笛卡尔坐标系为右手系，可以用下面例子中的方法把坐标系设置成右手系。

【例 1.1】 在 Form_Load 事件中通过 Scale 方法定义窗体 Form1 的坐标系，如表 1.2 所示。

表 1.2 例 1.1 窗体上各个控件主要属性设置

对 象	属 性	属 性 值
窗体	Caption	例 1.1 定义坐标系
命令按钮	Name	cmdRun
	Caption	绘图
标签	Name	Label1
	Caption	下限
文本框	Name	Text1
	Text	-1
标签	Name	Label2
	Caption	上限
文本框	Name	Text2
	Text	2
图片框	Name	Picture1

程序 1.1 自定义坐标系

```

Private Sub Form_Load ( )
    Cls
    Form1.Scale (-300,250) - (300, -150)
    Line (-300, 0) - (300, 0)           ' 画 X 轴
    Line (0, 250) - (0,-150)           ' 画 Y 轴
    CurrentX = 0 : CurrentY = 0: Print 0 ' 标记坐标原点
    CurrentX = 280 : CurrentY = 20 : Print "X" ' 标记 X 轴
    CurrentX = 10 : CurrentY = 240 : Print "Y" ' 标记 Y 轴
End Sub

```

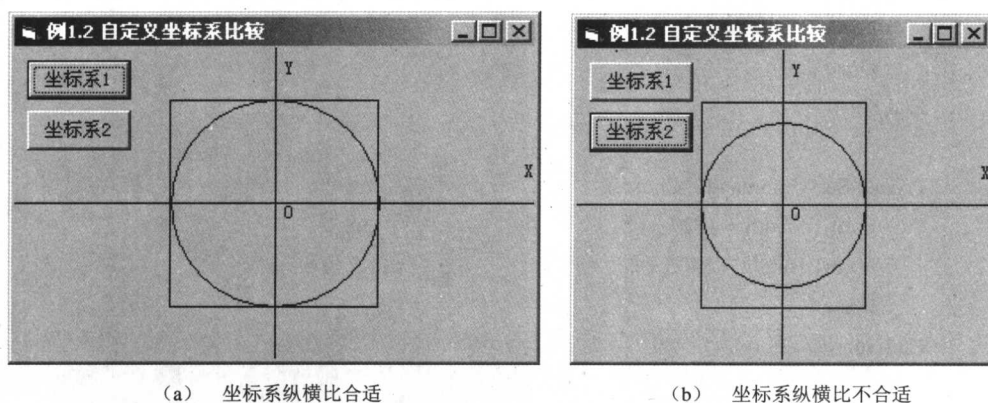
窗体 Form1 的自定义坐标系如图 1.2 所示。

任何时候在程序代码中使用 Scale 方法都能有效地改变坐标系统。当 Scale 方法不带参数时，则取消用户自定义的坐标系，而采用默认坐标系。

使用自定义的坐标系时要注意圆的处理。在 VB 中，不管采用何种标度，画出的圆都是正圆。如果自定义坐标系的宽度和高度不合适，会使圆与其他元素比例失调。下面的程序中，在不同大小的窗体中设置同样的坐标系，用同样的绘图语句画图，采用第一个坐标系绘图时，直线与圆都相切；采用第二个坐标系绘图时，垂直线与圆相切，水平线与圆不相切。对比图 1.3 (a) 和 (b) 可以得出，圆的水平半径与水平标度一致，而垂直半径与垂直标度不一致。



图 1.2 自定义坐标系



(a) 坐标系纵横比合适

(b) 坐标系纵横比不合适

图 1.3 自定义坐标系比较

由于窗体周围一小圈边中不能画图，还要扣除上端一条蓝色条幅，累计水平宽度为 120 twip，上下高度为 405 twip。如果要使圆与其他元素的关系协调，自定义坐标系时要在窗体尺寸上减去上述尺寸。按剩下的尺寸乘以同一个系数，作为注意自定义坐标系的 ScaleWidth, ScaleHeight，在这样的坐标系中画图，圆与其他元素的关系是绝对协调的。

【例 1.2】 在不同大小的窗体中定义相同的坐标系，见表 1.3。

表 1.3 例 1.2 窗体上各个控件主要属性设置

对 象	属 性	属 性 值
窗体	Caption	例 1.2 自定义坐标系比较
命令按钮	Name	Command1
	Caption	坐标系 1

续表

对 象	属 性	属 性 值
命令按钮	Name	Command2
	Caption	坐标系 2

坐标系 1 采用的窗体宽度、高度分别为 5120 twip、3405 twip，坐标系 2 采用的窗体宽度、高度分别为 4120 twip、3405 twip，窗体左上角坐标都定义为 (-250, 150)，右下角坐标都定义为 (250, -150)，坐标系的宽高比是 5:3，坐标系 1 采用的窗体扣除非作图尺寸后，宽度:高度=5:3，而坐标系 2 采用的窗体扣除非作图尺寸后，宽度:高度=4:3。

程序 1.2 坐标系比较

```
Private Sub Command1_Click( )
```

```
    Form1.Width = 5120
```

```
    Form1.Height = 3405
```

```
    Draw
```

```
End Sub
```

```
Private Sub Command2_Click( )
```

```
    Form1.Width = 4120
```

```
    Form1.Height = 3405
```

```
    Draw
```

```
End Sub
```

```
Private Sub Draw( )
```

```
    Form1.Scale (-250, 150) - (250, -150)
```

```
    Cls
```

```
    Line (-250, 0) - (250, 0)
```

```
' 画 X 轴
```

```
    Line (0, 150) - (0, -150)
```

```
' 画 Y 轴
```

```
    CurrentX = 0: CurrentY = 0: Print 0
```

```
' 标记坐标原点
```

```
    CurrentX = 240: CurrentY = 40: Print "X"
```

```
' 标记 X 轴
```

```
    CurrentX = 10: CurrentY = 140: Print "Y"
```

```
' 标记 Y 轴
```

```
    Line (-100, -100) - (100, 100), B
```

```
' 画矩形
```

```
    Circle (0, 0), 100
```

```
' 画圆
```

```
End Sub
```

1.2 绘图属性

1.2.1 当前坐标

窗体、图形框或打印机的 `CurrentX`, `CurrentY` 属性给出这些对象在绘图时的当前坐标。这两个属性在设计阶段不能使用。当坐标系确定后, 坐标值 (x, y) 表示对象上的绝对坐标位置。如果坐标值前加上关键字 `Step`, 则坐标值 (x, y) 表示对象上的相对坐标位置, 即从当前坐标分别平移 x, y 个单位, 其绝对坐标值为 $(\text{CurrentX}+x, \text{CurrentY}+y)$ 。

在执行 `Print` 方法后, `CurrentX` 保持不变, `CurrentY` 移到下一行的坐标; 在执行 `Line` 方法后, `CurrentX`, `CurrentY` 改为直线终点的坐标; 在执行 `Circle` 方法后, `CurrentX`, `CurrentY` 改为圆的圆心坐标。

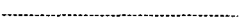
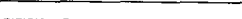
在例 1.2 中, 要在指定位置打印字母, 可设置 `CurrentX`, `CurrentY` 的属性值, 再用 `Print` 方法就可以在 `CurrentX`, `CurrentY` 位置显示该字母。在 `Print` 方法后, 当使用 `Cls` 方法时, `CurrentX`, `CurrentY` 属性值为 0。

1.2.2 线宽与线型

窗体、图形框或打印机的 `DrawWidth` 属性给出在这些对象上所画线的宽度或点的大小。`DrawWidth` 属性以像素为单位来度量, 为整形量, 最小值为 1, 具体属性如表 1.4 所示。

所画线型仅当 `DrawWidth` 属性值为 1 时才有效。当 `DrawWidth` 的值大于 1, `DrawStyle` 属性值为 1~4 时, 都只能产生实线效果; 当 `DrawWidth` 的值大于 1, 而 `DrawStyle` 属性值为 6 时, 所画的内实线仅当是封闭线时起作用。

表 1.4 DrawStyle 属性设置

设 置 值	线 型	图 示
0	实线 (默认)	
1	虚线	
2	点线	
3	点画线	
4	双点画线	
5	透明线	
6	内实线	