

Visual Basic

Visual Basic 程序员

Windows API

编程手册

林永 编著

人民邮电出版社  
POSTS & TELECOMMUNICATIONS PRESS

Visual Basic 程序员

**Windows API**

**编程手册**

林永 编著

人民邮电出版社

## 图书在版编目 ( C I P ) 数据

Visual Basic 程序员 Windows API 编程手册 / 林永编著. —北京: 人民邮电出版社, 2002.6  
ISBN 7-115-10068-3

I. V... II. 林... III. BASIC 语言—程序设计—技术手册 IV. TP312-62

中国版本图书馆 CIP 数据核字 (2002) 第 036545 号

## 内 容 提 要

本书是一本面向 Visual Basic 程序员的 Windows API 编程手册。书中收录了 Visual Basic 编程中涉及到的标准 Windows API 函数, 涵盖窗口、硬件系统、消息、设备场景、绘图、图形、图像处理、文本、字体、打印、文件、进程和线程、网络、多媒体等各个方面, 详细介绍了 API 函数在 Visual Basic 中使用的方法。

书中大部分函数后给出了有针对性的简短示例, 大部分章节的末尾给出了函数应用的综合性实例, 帮助读者在尽可能短的时间内掌握 Windows API 编程技巧, 创建功能强大的应用程序。

本书内容丰富, 实用性强, 主要面向初、中级的 Visual Basic 程序员, 也可作为有经验的 Visual Basic 程序员编程的参考手册。

### Visual Basic 程序员 Windows API 编程手册

- 
- ◆ 编 著 林 永  
责任编辑 马 嘉  
执行编辑 董 静
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
读者热线 010-67180876  
北京汉魂图文设计有限公司制作  
北京鸿佳印刷厂印刷  
新华书店总店北京发行所经销
  - ◆ 开本: 787×1092 1/16  
印张: 50.5  
字数: 1232 千字 2002 年 6 月第 1 版  
印数: 1-4 000 册 2002 年 6 月北京第 1 次印刷

---

ISBN 7-115-10068-3/TP · 2752

定价: 78.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

# 前 言

Microsoft 公司的 Visual Basic 简单易学、功能强大且扩充性好，已成为广泛应用的编程语言之一，用户只需经过简单学习就可以开发出各种专业的应用。Visual Basic 与 Windows 的联系非常密切，Visual Basic 的事件对应于 Windows 的消息，Visual Basic 的属性通常对应于 Windows 的风格和属性，在某种意义上可以认为 Visual Basic 是 Windows 的编程外壳，所以相对于 C 或 Pascal 等语言来说，使用 Visual Basic 设计 Windows 应用程序是非常简单的。

作为一个 Visual Basic 程序员，当完成初级水平的学习和实践后，需要进一步深入 Visual Basic 编程时，将不可避免地会接触到一些 Visual Basic 不直接支持的编程需求。为了解决这个问题，可以建立使用扩展语言功能的函数库。尽管 Visual Basic 自身提供了数量众多的内部函数，但是在编程的过程中，经常需要访问操作系统的一些核心的内容，这就需要访问 Windows 操作系统中的一些系统函数，这些函数是由 Win32 应用程序接口 (API, Application Program Interface) 提供的。

目前关于 Windows API 的书籍主要都是面向 C++ 程序员的，很少有面向 Visual Basic 程序员的介绍如何在 Visual Basic 中使用 Windows API 的书籍，这使得许多 Visual Basic 学习者在提高的过程中缺乏方便简明的参考和指南。为了满足 Visual Basic 学习者的需要，我们经过长期筹划后决定编写这本面向 Visual Basic 程序员的 Windows API 的编程手册。本书主要面向初、中级的 Visual Basic 程序员 (对 Windows 操作系统有一些粗略了解，能够使用 Visual Basic 编写一些简单的小程序，但还不能够编写涉及到系统特性的程序的程序员)，也可作为有经验的 Visual Basic 程序员编程的参考手册。

本书首先介绍了在 Visual Basic 中调用 Windows API 函数的方法和相关概念，然后根据函数功能的不同分类详细介绍了窗口与消息、硬件与系统、设备场景、绘图、位图、图标和光栅运算、菜单、文本和字体、打印、文件处理、Windows 消息、网络和 Internet、进程和线程、多媒体函数等。书中大部分函数后给出了有针对性的简短示例，大部分章节的末尾给出了函数应用的综合性实例。所有的示例除特别标注以外，均在 Windows 2000 系统下经过测试，其中一些示例反映了 Windows 2000 API 的特性，相信对读者会有所帮助。

由于 Visual Basic 的功能涉及范围广泛，作者水平有限，书中难免存在疏漏之处，敬请广大读者批评指正。如有疑问或其他意见，请与作者联系。

作者的 E-mail 信箱: linyong@263.net。

作 者

# 目 录

<b>第 1 章 Windows API 简介</b> .....	1
1.1 动态链接库 (DLL) .....	1
1.2 应用程序编程接口 (API) .....	3
1.2.1 Windows API 与 Visual Basic .....	3
1.2.2 主要的 DLL 库及其功能简介 .....	3
1.2.3 在 Visual Basic 中调用 Windows API .....	4
1.3 Windows API 相关概念 .....	8
1.3.1 句柄 .....	9
1.3.2 数据类型 .....	10
<b>第 2 章 窗口与消息函数</b> .....	15
2.1 获取窗口的信息函数 .....	15
2.1.1 GetClassInfo, GetClassInfoEx .....	16
2.1.2 GetClassLong .....	17
2.1.3 GetClassName .....	17
2.1.4 GetClassWord .....	18
2.1.5 SetClassLong .....	18
2.1.6 SetClassWord .....	18
2.1.7 GetDesktopWindow .....	20
2.1.8 GetWindowLong .....	21
2.1.9 GetWindowWord .....	23
2.1.10 SetWindowLong .....	24
2.1.11 SetWindowWord .....	26
2.1.12 GetWindowText .....	27
2.1.13 GetWindowTextLength .....	28
2.1.14 SetWindowText .....	29
2.1.15 IsChild .....	29
2.1.16 IsIconic .....	30
2.1.17 IsWindow .....	31
2.1.18 IsWindowEnabled .....	33
2.1.19 IsWindowUnicode .....	34
2.1.20 IsWindowVisible .....	35
2.1.21 IsZoomed .....	35
2.1.22 GetWindowContextHelpId .....	36
2.1.23 GetWindowPlacement .....	36
2.2 确定窗口相互间关系的函数 .....	36
2.2.1 EnumWindows .....	37
2.2.2 EnumChildWindows .....	38
2.2.3 EnumThreadWindows .....	38
2.2.4 FindWindow .....	40
2.2.5 FindWindowEx .....	42
2.2.6 GetWindow .....	44
2.2.7 GetActiveWindow .....	45
2.2.8 GetCapture .....	47
2.2.9 GetFocus .....	47
2.2.10 GetForegroundWindow .....	48
2.2.11 GetLastActivePopup .....	49
2.2.12 GetParent .....	49
2.2.13 GetTopWindow .....	50
2.2.14 AnyPopup .....	50
2.2.15 ChildWindowFromPoint, ChildWindowFromPointEx .....	50
2.2.16 SetActiveWindow .....	51
2.2.17 SetCapture .....	51
2.2.18 SetFocus, SetFocusAPI .....	53
2.2.19 SetForegroundWindow .....	53
2.2.20 SetParent .....	54
2.3 控制窗口外观和行为的函数 .....	55
2.3.1 ArrangeIconicWindows .....	55
2.3.2 BeginDeferWindowPos .....	55
2.3.3 BringWindowToTop .....	56
2.3.4 CascadeWindows, CascadeWindowsBynum .....	57
2.3.5 CloseWindow .....	57
2.3.6 DeferWindowPos .....	57
2.3.7 DestroyWindow .....	59
2.3.8 EnableWindow .....	59
2.3.9 EndDeferWindowPos .....	60
2.3.10 FlashWindow .....	60

2.3.11	LockWindowUpdate.....	61	3.1	键盘输入与字符集函数.....	97
2.3.12	MoveWindow.....	63	3.1.1	ActivateKeyboardLayout.....	97
2.3.13	OpenIcon.....	63	3.1.2	GetKeyboardLayout.....	97
2.3.14	ReleaseCapture.....	63	3.1.3	GetAsyncKeyState.....	97
2.3.15	SetWindowPlacement.....	64	3.1.4	GetKeyboardLayoutList.....	100
2.3.16	SetWindowPos.....	65	3.1.5	GetKeyboardLayoutName.....	101
2.3.17	ShowOwnedPopups.....	66	3.1.6	GetKeyboardState.....	102
2.3.18	ShowWindow.....	66	3.1.7	GetKeyboardType.....	105
2.3.19	ShowWindowAsync.....	67	3.1.8	GetKeyNameText.....	106
2.3.20	TileWindows.....	67	3.1.9	GetKeyState.....	106
2.3.21	UpdateWindow.....	68	3.1.10	keybd_event.....	108
2.4	操作窗口中客户区的函数.....	68	3.1.11	LoadKeyboardLayout.....	108
2.4.1	AdjustWindowRect,AdjustWindowRectEx.....	68	3.1.12	MapVirtualKey.....	109
2.4.2	ClientToScreen.....	69	3.1.13	MapVirtualKeyEx.....	110
2.4.3	CopyRect.....	71	3.1.14	OemKeyScan.....	110
2.4.4	DrawAnimatedRects.....	72	3.1.15	OemToChar, OemToCharBuff.....	110
2.4.5	EqualRect.....	73	3.1.16	SetKeyboardState.....	111
2.4.6	GetClientRect.....	73	3.1.17	ToAscii, ToAsciiEx.....	111
2.4.7	GetUpdateRect.....	74	3.1.18	ToUnicode.....	112
2.4.8	GetWindowRect.....	75	3.1.19	UnloadKeyboardLayout.....	113
2.4.9	InflateRect.....	79	3.1.20	VkKeyScan, VkKeyScanEx.....	113
2.4.10	IntersectRect.....	80	3.2	语言地区代码页相关函数.....	113
2.4.11	InvalidateRect,InvalidateRectBynum.....	80	3.2.1	CharToOem, CharToOemBuff.....	113
2.4.12	OffsetRect.....	80	3.2.2	ConvertDefaultLocale.....	114
2.4.13	PtInRect.....	81	3.2.3	EnumCalendarInfo.....	114
2.4.14	RedrawWindow.....	82	3.2.4	EnumDateFormats.....	115
2.4.15	ScrollWindow.....	83	3.2.5	EnumSystemCodePages.....	115
2.4.16	ScrollWindowEx.....	84	3.2.6	EnumDateFormats.....	116
2.4.17	SetRect.....	85	3.2.7	EnumTimeFormats.....	116
2.4.18	SetRectEmpty.....	85	3.2.8	GetACP.....	116
2.4.19	SubtractRect.....	85	3.2.9	GetCPInfo.....	117
2.4.20	ValidateRect.....	86	3.2.10	GetCurrencyFormat, GetCurrencyFormatBynum.....	117
2.5	其他与窗口相关的函数.....	87	3.2.11	GetDateFormat.....	118
2.5.1	AttachThreadInput.....	87	3.2.12	GetKBCodePage.....	119
2.5.2	ScreenToClient.....	87	3.2.13	GetOEMCP.....	119
2.5.3	SetWindowContextHelpId.....	87	3.2.14	GetLocaleInfo.....	119
2.5.4	WindowFromPoint.....	88	3.2.15	GetLocalTime.....	120
2.6	综合示例.....	88	3.2.16	GetNumberFormat, GetNumberFormatBynum.....	121
<b>第3章</b>	<b>硬件与系统函数.....</b>	<b>95</b>			

3.2.17	GetOEMCP	122	3.4.6	GetClipCursor	153
3.2.18	GetSystemDefaultLangID	122	3.4.7	GetCursor	153
3.2.19	GetSystemDefaultLCID	122	3.4.8	GetCursorPos	154
3.2.20	GetThreadLocale	122	3.4.9	GetDoubleClickTime	154
3.2.21	GetTimeFormat	122	3.4.10	HideCaret	154
3.2.22	GetTimeZoneInformation	123	3.4.11	SetCursor	155
3.2.23	GetUserDefaultLangID	124	3.4.12	SetCursorPos	157
3.2.24	GetUserDefaultLCID	124	3.4.13	SetDoubleClickTime	157
3.2.25	IsValidCodePage	124	3.4.14	mouse_event	158
3.2.26	IsValidLocale	124	3.4.15	ShowCaret	159
3.2.27	SetLocaleInfo	124	3.4.16	ShowCursor	159
3.2.28	SetLocalTime	125	3.4.17	SwapMouseButton	160
3.2.29	SetThreadLocale	125	3.5	其他函数	161
3.2.30	SystemTimeToTzSpecificLocalTime	125	3.5.1	Beep	161
3.3	系统信息相关函数	126	3.5.2	ExitWindowsEx	161
3.3.1	GetSysColor	126	3.5.3	ExpandEnvironmentStrings	162
3.3.2	GetSystemInfo	128	3.5.4	FreeEnvironmentStrings	163
3.3.3	GetSystemMetrics	129	3.5.5	GetEnvironmentStrings	163
3.3.4	GetSystemPowerStatus	134	3.5.6	GetEnvironmentVariable	164
3.3.5	GetSystemTime	134	3.5.7	GetLastError	164
3.3.6	GetSystemTimeAdjustment	134	3.5.8	GetQueueStatus	165
3.3.7	GetTickCount	135	3.5.9	MessageBeep	166
3.3.8	GetUserName	136	3.5.10	SetEnvironmentVariable	167
3.3.9	GetVersion	137	3.6	综合示例	167
3.3.10	GetVersionEx	138	第4章	设备场景函数	183
3.3.11	GlobalMemoryStatus 函数	140	4.1	获取和释放设备场景函数	185
3.3.12	SetComputerName	141	4.1.1	CreateCompatibleDC	185
3.3.13	SetSysColors	142	4.1.2	CreateDC, CreateDCBynum	185
3.3.14	SetSystemTime	142	4.1.3	CreateIC	189
3.3.15	SetSystemTimeAdjustment	143	4.1.4	DeleteDC	189
3.3.16	SetThreadLocale	144	4.1.5	GetDC	189
3.3.17	SetTimeZoneInformation	144	4.1.6	GetDCEX	191
3.3.18	SystemParametersInfo, SystemParametersInfoByval	144	4.1.7	GetWindowDC	192
3.4	光标(插入符)与鼠标输入函数	150	4.1.8	ReleaseDC	192
3.4.1	ClipCursor, ClipCursorBynum	150	4.1.9	WindowFromDC	194
3.4.2	CreateCaret	150	4.2	坐标映射函数	195
3.4.3	DestroyCaret	152	4.2.1	CombineTransform	197
3.4.4	GetCaretBlinkTime	152	4.2.2	DPTOLP	197
3.4.5	GetCaretPos	153	4.2.3	GetDCOrgEx	197

4.2.4	GetGraphicsMode	198	4.3.19	GetRegionData	220
4.2.5	GetMapMode	198	4.3.20	GetRgnBox	221
4.2.6	GetViewportExtEx	198	4.3.21	GetUpdateRgn	223
4.2.7	GetViewportOrgEx	199	4.3.22	InvertRgn	224
4.2.8	GetWindowExtEx	199	4.3.23	ModifyWorldTransform	224
4.2.9	GetWindowOrgEx	199	4.3.24	OffsetClipRgn	224
4.2.10	GetWorldTransform	200	4.3.25	OffsetRgn	225
4.2.11	LPToDP	200	4.3.26	PaintRgn	225
4.2.12	ModifyWorldTransform	200	4.3.27	PtInRegion	226
4.2.13	OffsetViewportOrgEx	201	4.3.28	PtVisible	226
4.2.14	OffsetWindowOrgEx	201	4.3.29	RectInRegion	226
4.2.15	ScaleViewportExtEx	201	4.3.30	RectVisible	226
4.2.16	ScaleWindowExtEx	202	4.3.31	SelectClipRgn	227
4.2.17	ScrollDC	202	4.3.32	SetBoundsRect	227
4.2.18	SetGraphicsMode	203	4.3.33	SetRectRgn	228
4.2.19	SetMapMode	203	4.3.34	SetWindowRgn	228
4.2.20	SetViewportExtEx	204	4.3.35	ValidateRgn	230
4.2.21	SetViewportOrgEx	204	4.4	堆栈函数	231
4.2.22	SetWindowExtEx	205	4.4.1	RestoreDC	231
4.2.23	SetWindowOrgEx	205	4.4.2	SaveDC	231
4.2.24	SetWorldTransform	205	4.5	设备场景信息函数	232
4.3	区域和剪裁函数	206	GetDeviceCaps		232
4.3.1	CombineRgn	206	<b>第5章 绘图函数</b>		<b>240</b>
4.3.2	CreateEllipticRgn	211	5.1	GDI 绘图对象创建函数	240
4.3.3	CreateEllipticRgnIndirect	211	5.1.1	CreateBrushIndirect	240
4.3.4	CreatePolygonRgn	212	5.1.2	CreateBrushIndirect	241
4.3.5	CreatePolyPolygonRgn	214	5.1.3	CreateDIBPatternBrush, CreateDIBPatternBrushPt	241
4.3.6	CreateRectRgn	215	5.1.4	CreateEnhMetaFile	242
4.3.7	CreateRectRgnIndirect	215	5.1.5	CreateHatchBrush	243
4.3.8	CreateRoundRectRgn	216	5.1.6	CreateMetaFile	243
4.3.9	EqualRgn	216	5.1.7	CreatePatternBrush	244
4.3.10	ExcludeClipRect	216	5.1.8	CreatePen	245
4.3.11	ExcludeUpdateRgn	217	5.1.9	CreatePenIndirect	246
4.3.12	ExtCreateRegion	217	5.1.10	CreateSolidBrush	246
4.3.13	ExtSelectClipRgn	218	5.1.11	ExtCreatePen	249
4.3.14	FillRgn	218	5.1.12	GetStockObject	250
4.3.15	FrameRgn	219	5.1.13	GetSysColorBrush	251
4.3.16	GetBoundsRect	219	5.2	绘图对象选取和释放函数	251
4.3.17	GetClipBox	220			
4.3.18	GetClipRgn	220			



5.2.1	DeleteObject .....	251	5.3.35	PolyBezier, PolyBezierTo .....	275
5.2.2	EnumObjects.....	252	5.3.36	PolyDraw .....	275
5.2.3	GetCurrentObject.....	253	5.3.37	Polygon.....	276
5.2.4	GetObjectAPI.....	253	5.3.38	Polyline, PolyLineTo.....	276
5.2.5	GetObjectType.....	253	5.3.39	PolyPolygon .....	277
5.3	绘图属性和绘图函数 .....	255	5.3.40	PolyPolyline .....	278
5.3.1	AngleArc.....	255	5.3.41	Rectangle .....	279
5.3.2	Arc, ArcTo .....	256	5.3.42	RoundRect .....	280
5.3.3	CancelDC.....	258	5.3.43	SetArcDirection .....	280
5.3.4	Chord .....	258	5.3.44	SetBkColor .....	280
5.3.5	DrawEdge .....	259	5.3.45	SetBkMode .....	281
5.3.6	DrawEscape .....	260	5.3.46	SetBrushOrgEx .....	281
5.3.7	DrawFocusRect.....	260	5.3.47	SetMiterLimit .....	282
5.3.8	DrawFrameControl.....	260	5.3.48	SetPixel .....	282
5.3.9	DrawState.....	261	5.3.49	SetPixelV.....	282
5.3.10	Ellipse .....	263	5.3.50	SetPolyFillMode.....	283
5.3.11	ExtFloodFill.....	264	5.3.51	SetROP2 .....	283
5.3.12	FillRect.....	264	5.4	路径函数 .....	284
5.3.13	FloodFill .....	265	5.4.1	AbortPath.....	285
5.3.14	FrameRect.....	265	5.4.2	BeginPath .....	285
5.3.15	GdiFlush .....	266	5.4.3	CloseFigure.....	285
5.3.16	GdiGetBatchLimit .....	266	5.4.4	EndPath.....	286
5.3.17	GdiSetBatchLimit .....	266	5.4.5	FillPath .....	286
5.3.18	GetArcDirection .....	267	5.4.6	FlattenPath.....	286
5.3.19	GetBkColor.....	267	5.4.7	GetPath .....	286
5.3.20	GetBkMode.....	268	5.4.8	PathToRegion.....	287
5.3.21	GetBrushOrgEx .....	268	5.4.9	SelectClipPath .....	287
5.3.22	GetCurrentPositionEx.....	269	5.4.10	StrokeAndFillPath.....	288
5.3.23	GetMiterLimit.....	269	5.4.11	StrokePath.....	288
5.3.24	GetNearestColor .....	269	5.4.12	WidenPath .....	289
5.3.25	GetPixel .....	269	5.5	图元文件函数 .....	289
5.3.26	GetPolyFillMode .....	270	5.5.1	CloseEnhMetaFile.....	290
5.3.27	GetROP2 .....	270	5.5.2	CloseMetaFile.....	290
5.3.28	GetSysColorBrush .....	271	5.5.3	CopyEnhMetaFile.....	291
5.3.29	InvertRect.....	271	5.5.4	CopyMetaFile.....	291
5.3.30	LineDDA .....	273	5.5.5	CreateEnhMetaFile.....	291
5.3.31	LineTo.....	273	5.5.6	CreateMetaFile .....	292
5.3.32	MoveToEx .....	273	5.5.7	DeleteEnhMetaFile.....	293
5.3.33	PaintDesktop.....	274	5.5.8	DeleteMetaFile .....	293
5.3.34	Pie .....	274	5.5.9	EnumEnhMetaFile.....	293

5.5.10 EnumMetaFile .....	294	6.4 位图传输函数 .....	320
5.5.11 GdiComment.....	294	6.4.1 BitBlt .....	322
5.5.12 GetEnhMetaFile.....	295	6.4.2 GetStretchBltMode.....	324
5.5.13 GetEnhMetaFileBits .....	296	6.4.3 MaskBlt .....	324
5.5.14 GetEnhMetaFileDescription.....	296	6.4.4 PatBlt .....	325
5.5.15 GetEnhMetaFileHeader .....	297	6.4.5 PlgBlt.....	325
5.5.16 GetEnhMetaFilePaletteEntries .....	297	6.4.6 SetStretchBltMode .....	326
5.5.17 GetMetaFile .....	297	6.4.7 StretchBlt.....	327
5.5.18 GetMetaFileBitsEx .....	298	6.5 图标和指针函数 .....	329
5.5.19 GetWinMetaFileBits.....	298	6.5.1 CreateCursor.....	329
5.5.20 PlayEnhMetaFile .....	299	6.5.2 CreateIcon.....	331
5.5.21 PlayEnhMetaFileRecord.....	299	6.5.3 CreateIconIndirect .....	332
5.5.22 PlayMetaFile.....	299	6.5.4 DestroyCursor.....	332
5.5.23 PlayMetaFileRecord .....	300	6.5.5 DestroyIcon .....	333
5.5.24 SetEnhMetaFileBits.....	300	6.5.6 DrawIcon .....	334
5.5.25 SetMetaFileBitsEx .....	301	6.5.7 DrawIconEx.....	336
5.5.26 SetWinMetaFileBits .....	301	6.5.8 ExtractAssociatedIcon.....	337
<b>第 6 章 位图、图标和光栅运算函数 .....</b>	<b>302</b>	6.5.9 ExtractIcon .....	337
6.1 Windows 中图像的处理概述 .....	302	6.5.10 GetIconInfo.....	339
6.2 设备无关位图函数 .....	303	6.5.11 LoadCursor, LoadCursorBynum .....	339
6.2.1 CreateDIBitmap.....	306	6.5.12 LoadCursorFromFile .....	339
6.2.2 CreateDIBSection.....	306	6.5.13 LoadIcon, LoadIconBynum.....	340
6.2.3 GetDIBColorTable .....	307	6.6 调色板 .....	340
6.2.4 GetDIBits.....	308	6.6.1 计算机色彩系统 .....	340
6.2.5 SetDIBColorTable .....	308	6.6.2 调色板 .....	342
6.2.6 SetDIBits.....	309	6.7 综合示例 .....	349
6.2.7 SetDIBitsToDevice.....	309	<b>第 7 章 菜单函数 .....</b>	<b>357</b>
6.2.8 StretchDIBits .....	310	7.1 菜单函数 .....	357
6.3 设备相关位图函数 .....	311	7.1.1 AppendMenu .....	357
6.3.1 CopyImage.....	312	7.1.2 CheckMenuItem .....	359
6.3.2 CreateBitmap .....	313	7.1.3 CheckMenuRadioItem.....	360
6.3.3 CreateBitmapIndirect.....	313	7.1.4 CreateMenu .....	361
6.3.4 CreateCompatibleBitmap .....	313	7.1.5 CreatePopupMenu .....	361
6.3.5 GetBitmapBits .....	314	7.1.6 DeleteMenu .....	361
6.3.6 GetBitmapDimensionEx.....	316	7.1.7 DestroyMenu .....	361
6.3.7 LoadBitmap, LoadBitmapBynum.....	317	7.1.8 DrawMenuBar .....	362
6.3.8 LoadImage, LoadImageBynum.....	317	7.1.9 EnableMenuItem .....	363
6.3.9 SetBitmapBits .....	320	7.1.10 GetMenu .....	363
6.3.10 SetBitmapDimensionEx .....	320	7.1.11 GetMenuCheckMarkDimensions .....	365

7.1.12	GetMenuContextHelpId .....	365	8.2.10	ExtTextOut .....	406
7.1.13	GetMenuDefaultItem .....	365	8.2.11	GetAspectRatioFilterEx .....	408
7.1.14	GetMenuItemCount .....	365	8.2.12	GetCharABCWidths .....	408
7.1.15	GetMenuItemID .....	366	8.2.13	GetCharABCWidthsFloat .....	409
7.1.16	GetMenuItemInfo .....	366	8.2.14	GetCharacterPlacement .....	409
7.1.17	GetMenuItemRect .....	366	8.2.15	GetCharWidth, GetCharWidth32, GetCharWidthFloat .....	410
7.1.18	GetMenuState .....	367	8.2.16	GetFontData .....	411
7.1.19	GetMenuString .....	367	8.2.17	GetFontLanguageInfo .....	411
7.1.20	GetSubMenu .....	368	8.2.18	GetGlyphOutline .....	412
7.1.21	GetSystemMenu .....	371	8.2.19	GetKerningPairs .....	412
7.1.22	HiliteMenuItem .....	371	8.2.20	GetOutlineTextMetrics .....	412
7.1.23	InsertMenu .....	372	8.2.21	GetRasterizerCaps .....	413
7.1.24	InsertMenuItem .....	372	8.2.22	GetTabbedTextExtent .....	413
7.1.25	IsMenu .....	373	8.2.23	GetTextAlign .....	414
7.1.26	LoadMenu .....	373	8.2.24	GetTextCharacterExtra .....	414
7.1.27	LoadMenuIndirect .....	373	8.2.25	GetTextCharset .....	414
7.1.28	MenuItemFromPoint .....	374	8.2.26	GetTextCharsetInfo .....	415
7.1.29	ModifyMenu, ModifyMenuBynum .....	374	8.2.27	GetTextColor .....	415
7.1.30	RemoveMenu .....	376	8.2.28	GetTextExtentExPoint .....	415
7.1.31	SetMenu .....	376	8.2.29	GetTextExtentPoint, GetTextExtentPoint32 .....	416
7.1.32	SetMenuContextHelpId .....	376	8.2.30	GetTextFace .....	419
7.1.33	SetMenuDefaultItem .....	377	8.2.31	GetTextMetrics .....	419
7.1.34	SetMenuItemBitmaps .....	377	8.2.32	GrayString, GrayStringByString .....	421
7.1.35	SetMenuItemInfo .....	378	8.2.33	PolyTextOut .....	423
7.1.36	TrackPopupMenu, TrackPopupMenuBynum .....	378	8.2.34	RemoveFontResource .....	423
7.1.37	TrackPopupMenuEx .....	379	8.2.35	SetMapperFlags .....	423
7.2	综合示例 .....	380	8.2.36	SetTextAlign .....	424
<b>第 8 章</b>	<b>文本和字体函数</b> .....	<b>388</b>	8.2.37	SetTextCharacterExtra .....	425
8.1	文本和字体简介 .....	388	8.2.38	SetTextColor .....	426
8.2	文本和字体函数 .....	389	8.2.39	SetTextJustification .....	426
8.2.1	AddFontResource .....	389	8.2.40	TabbedTextOut .....	427
8.2.2	CreateFont .....	389	8.2.41	TextOut .....	428
8.2.3	CreateFontIndirect .....	394	<b>第 9 章</b>	<b>打印函数</b> .....	<b>429</b>
8.2.4	CreateScalableFontResource .....	396	9.1	Windows 打印简介 .....	429
8.2.5	DrawText .....	396	9.2	打印函数 .....	430
8.2.6	DrawTextEx .....	400	9.2.1	AbortDoc .....	430
8.2.7	EnumFontFamilies .....	402	9.2.2	AbortPrinter .....	431
8.2.8	EnumFontFamiliesEx .....	406	9.2.3	AddForm .....	431
8.2.9	EnumFonts .....	406			

9.2.4	AddJob	431	9.2.44	GetJob	457
9.2.5	AddMonitor	432	9.2.45	GetPrinter	458
9.2.6	AddPort	432	9.2.46	GetPrinterData	458
9.2.7	AddPrinter	432	9.2.47	GetPrinterDriver	459
9.2.8	AddPrinterConnection	433	9.2.48	GetPrinterDriverDirectory	459
9.2.9	AddPrinterDriver	433	9.2.49	GetPrintProcessorDirectory	460
9.2.10	AddPrintProcessor	434	9.2.50	OpenPrinter	460
9.2.11	AddPrintProvider	434	9.2.51	PrinterMessageBox	461
9.2.12	AdvancedDocumentProperties	434	9.2.52	PrinterProperties	461
9.2.13	ClosePrinter	435	9.2.53	ReadPrinter	462
9.2.14	ConfigurePort	435	9.2.54	ResetDC	462
9.2.15	ConnectToPrinterDlg	436	9.2.55	ResetPrinter	463
9.2.16	DeleteForm	436	9.2.56	ScheduleJob	463
9.2.17	DeleteMonitor	436	9.2.57	SetAbortProc	463
9.2.18	DeletePort	437	9.2.58	SetForm	464
9.2.19	DeletePrinter	437	9.2.59	SetJob	464
9.2.20	DeletePrinterConnection	437	9.2.60	SetPrinter	465
9.2.21	DeletePrinterDriver	437	9.2.61	SetPrinterData	465
9.2.22	DeletePrintProcessor	438	9.2.62	StartDoc	466
9.2.23	DeletePrintProvider	438	9.2.63	StartDocPrinter	466
9.2.24	DeviceCapabilities	439	9.2.64	StartPage	467
9.2.25	DocumentProperties	442	9.2.65	StartPagePrinter	467
9.2.26	EndDocAPI	443	9.2.66	WritePrinter	467
9.2.27	EndDocPrinter	443	<b>第 10 章 文件处理函数</b>	<b>468</b>	
9.2.28	EndPage	443	10.1 文件介绍	468	
9.2.29	EndPagePrinter	444	10.2 文件的创建和存取函数	468	
9.2.30	EnumForms	444	10.2.1 CloseHandle	468	
9.2.31	EnumJobs	444	10.2.2 CreateFile	469	
9.2.32	EnumMonitors	445	10.2.3 CreateFileMapping	471	
9.2.33	EnumPorts	445	10.2.4 FlushFileBuffers	472	
9.2.34	EnumPrinterDrivers	449	10.2.5 FlushViewOfFile	472	
9.2.35	EnumPrinters	449	10.2.6 lclose	472	
9.2.36	EnumPrintProcessorDatatypes	452	10.2.7 lcreat	472	
9.2.37	EnumPrintProcessors	453	10.2.8 llseek	473	
9.2.38	Escape	453	10.2.9 LockFile	473	
9.2.39	FindClosePrinterChangeNotification	455	10.2.10 LockFileEx	474	
9.2.40	FindFirstPrinterChangeNotification	455	10.2.11 lopen	475	
9.2.41	FindNextPrinterChangeNotification	456	10.2.12 MapViewOfFile, MapViewOfFileEx	475	
9.2.42	FreePrinterNotifyInfo	456	10.2.13 OpenFile	476	
9.2.43	GetForm	457			

10.2.14	OpenFileMapping	479	10.4.13	SearchPath	512
10.2.15	SetEndOfFile	479	10.4.14	VerFindFile	513
10.2.16	SetFilePointer	479	10.4.15	VerInstallFile	513
10.2.17	UnlockFile	480	10.4.16	WriteFile	515
10.2.18	UnlockFileEx	480	10.4.17	WriteFileEx	516
10.2.19	UnmapViewOfFile	481	10.5	压缩文件函数	516
10.3	文件的属性函数	481	10.5.1	GetCompressedFileSize	516
10.3.1	CompareFileTime	481	10.5.2	GetExpandedName	517
10.3.2	DosDateTimeToFileTime	481	10.5.3	LZClose	517
10.3.3	FileTimeToDosDateTime	482	10.5.4	LZCopy	519
10.3.4	FileTimeToLocalFileTime	482	10.5.5	LZInit	520
10.3.5	FileTimeToSystemTime	482	10.5.6	LZOpenFile	520
10.3.6	GetBinaryType	483	10.5.7	LZRead	520
10.3.7	GetFileAttributes	483	10.5.8	LZSeek	521
10.3.8	GetFileInformationByHandle	493	10.6	目录函数	521
10.3.9	GetFileSize	494	10.6.1	CreateDirectory, CreateDirectoryEx	521
10.3.10	GetFileTime	494	10.6.2	GetCurrentDirectory	522
10.3.11	GetFileType	497	10.6.3	GetSystemDirectory	522
10.3.12	GetFileVersionInfo	497	10.6.4	GetTempPath	523
10.3.13	GetFileVersionInfoSize	498	10.6.5	GetVolumeInformation	524
10.3.14	GetFullPathName	498	10.6.6	GetWindowsDirectory	525
10.3.15	GetTempFileName	499	10.6.7	RemoveDirectory	525
10.3.16	SetFileAttributes	500	10.7	驱动器函数	525
10.3.17	SetFileTime	501	10.7.1	GetDiskFreeSpace	525
10.3.18	SystemTimeToFileTime	503	10.7.2	GetDiskFreeSpaceEx	527
10.3.19	VerLanguageName	503	10.7.3	GetDriveType	528
10.3.20	VerQueryValue	503	10.7.4	GetLogicalDrives	529
10.4	文件的操作函数	504	10.7.5	GetLogicalDriveStrings	530
10.4.1	CopyFile	504	10.7.6	SetVolumeLabel	533
10.4.2	DeleteFile	505	10.8	注册表简介	536
10.4.3	FindClose	505	10.9	注册表的创建和存取函数	540
10.4.4	FindFirstFile	508	10.9.1	RegCloseKey	540
10.4.5	FindNextFile	509	10.9.2	RegConnectRegistry	540
10.4.6	hread	509	10.9.3	RegCreateKey	540
10.4.7	hwrite	509	10.9.4	RegCreateKeyEx	541
10.4.8	hread	510	10.9.5	RegOpenKey	542
10.4.9	lwrite	510	10.9.6	RegOpenKeyEx	542
10.4.10	MoveFile, MoveFileEx	510	10.9.7	RegSaveKey	543
10.4.11	ReadFile	511	10.10	注册表项目的操作	545
10.4.12	ReadFileEx	512	10.10.1	RegDeleteKey	545

10.10.2	RegDeleteValue	545	PostMessageBystring	591
10.10.3	RegEnumKey	545	11.1.5 PostThreadMessage	602
10.10.4	RegEnumKeyEx	546	11.1.6 RegisterWindowMessage	603
10.10.5	RegEnumValue	548	11.1.7 ReplyMessage	603
10.10.6	RegFlushKey	548	11.1.8 SendMessage, SendMessageBynum,	
10.10.7	RegGetKeySecurity	549	SendMessageByString	603
10.10.8	RegLoadKey	549	11.1.9 SendMessageCallback	605
10.10.9	RegNotifyChangeKeyValue	550	11.1.10 SendMessageTimeout	605
10.10.10	RegQueryInfoKey	551	11.1.11 SendNotifyMessage	606
10.10.11	RegQueryValue	552	11.2 综合示例	607
10.10.12	RegQueryValueEx	552	<b>第 12 章 网络和 Internet 函数</b>	614
10.10.13	RegReplaceKey	553	12.1 网络函数	614
10.10.14	RegRestoreKey	554	12.1.1 WNetAddConnection	614
10.10.15	RegSetKeySecurity	554	12.1.2 WNetAddConnection2	616
10.10.16	RegSetValue	554	12.1.3 WNetAddConnection3	622
10.10.17	RegSetValueEx	555	12.1.4 WNetCancelConnection	622
10.10.18	RegUnLoadKey	555	12.1.5 WNetCancelConnection2	623
10.11	INI 文件函数	560	12.1.6 WNetCloseEnum	623
10.11.1	GetPrivateProfileInt	561	12.1.7 WNetConnectionDialog	626
10.11.2	GetPrivateProfileSection	562	12.1.8 WNetDisconnectDialog	626
10.11.3	GetPrivateProfileString	562	12.1.9 WNetEnumResource	626
10.11.4	GetProfileInt	563	12.1.10 WNetGetConnection	627
10.11.5	GetProfileSection	564	12.1.11 WNetGetLastError	627
10.11.6	GetProfileString	564	12.1.12 WNetGetUniversalName	628
10.11.7	WritePrivateProfileSection	565	12.1.13 WNetGetUser	629
10.11.8	WritePrivateProfileString	566	12.1.14 WNetOpenEnum	629
10.11.9	WriteProfileSection	566	12.2 Internet 函数	630
10.11.10	WriteProfileString	567	12.2.1 Internet 的相关概念	630
10.12	设备控制函数	567	12.2.2 Internet 函数简介	633
10.12.1	DeviceIoControl	567	12.3 Winsock 函数	635
10.12.2	QueryDosDevice	571	12.3.1 Windows Sockets 规范简介	635
10.12.3	SetHandleCount	572	12.3.2 Winsock 函数简介	637
10.13	综合示例	572	12.4 综合示例	639
<b>第 11 章 Windows 消息函数</b>		589	<b>第 13 章 进程和线程函数</b>	680
11.1	消息函数	589	13.1 进程和线程简介	680
11.1.1	BroadcastSystemMessage	589	13.1.1 多任务、进程与线程	680
11.1.2	GetMessagePos	591	13.1.2 线程的同步	681
11.1.3	GetMessageTime	591	13.2 进程和线程操作函数	682
11.1.4	PostMessage, PostMessageBynum,		13.2.1 CreatProcess	682

13.2.2	ExitProcess .....	684	14.2.2	AVI 数字视频的特点 .....	737
13.2.3	FindExecutable .....	685	14.2.3	AVI 采用的压缩算法 .....	738
13.2.4	FreeLibrary .....	686	14.3	MCI 命令字符串接口 .....	738
13.2.5	GetCurrentProcess .....	686	14.4	多媒体程序实例 .....	744
13.2.6	GetCurrentProcessId .....	691	<b>附录</b>	.....	771
13.2.7	GetCurrentThread .....	692	附录 A	虚拟键码 .....	771
13.2.8	GetExitCodeProcess .....	692	附录 B	扬声器发音频率 .....	775
13.2.9	GetExitCodeThread .....	693	附录 C	可用代码页列表 .....	776
13.2.10	GetModuleFileName .....	693	附录 D	结构定义 .....	776
13.2.11	GetModuleHandle .....	694	D.1	BITMAP .....	776
13.2.12	GetPriorityClass .....	695	D.2	CPINFO .....	777
13.2.13	GetProcessShutdownParameters .....	695	D.3	DEVMODE .....	777
13.2.14	GetProcessTimes .....	695	D.4	FILETIME .....	778
13.2.15	GetProcessWorkingSetSize .....	696	D.5	FORM_INFO_1 .....	779
13.2.16	GetStartupInfo .....	696	D.6	LOGBRUSH .....	779
13.2.17	GetThreadPriority .....	696	D.7	LOGPEN .....	780
13.2.18	GetTheardTimes .....	697	D.8	MENUITEMINFO .....	780
13.2.19	GetWindowThreadProcessId .....	697	D.9	MONITOR_INFO_2 .....	782
13.2.20	LoadLibrary .....	699	D.10	NETRESOURCE .....	783
13.2.21	LoadLibraryEx .....	701	D.11	OSVERSIONINFO .....	784
13.2.22	LoadModule .....	702	D.12	POINTAPI .....	784
13.2.23	SetPriorityClass .....	702	D.13	PROVIDOR_INFO_1 .....	785
13.2.24	SetProcessShutdownParameters .....	703	D.14	RECT .....	785
13.2.25	SetProcessWorkingSetSize .....	703	D.15	RGNDATA .....	785
13.2.26	SetThreadPriority .....	703	D.16	SYSTEM_INFO .....	786
13.2.27	ShellExecute .....	704	D.17	SYSTEMTIME .....	787
13.2.28	TerminateProcess .....	704	D.18	TIME_ZONE_INFORMATION .....	788
13.2.29	WinExec .....	705	D.19	WINDOWPLACEMENT .....	789
13.3	同步函数 .....	705	D.20	XFORM .....	789
13.3.1	DuplicateHandle .....	705			
13.3.2	GetHandleFormation .....	706			
13.3.3	MsgWaitForMultipleObjects .....	707			
13.4	综合示例 .....	708			
<b>第 14 章</b>	<b>多媒体函数 .....</b>	<b>734</b>			
14.1	音频函数 .....	734			
14.1.1	PlaySound .....	734			
14.1.2	sndPlaySound .....	735			
14.2	AVI 文件 .....	736			
14.2.1	AVI 数字视频的格式 .....	736			

# 第 1 章 Windows API 简介

## 1.1 动态链接库 (DLL)

程序员在编写各种用途的应用程序时经常需要使用一些函数库，所谓函数库就是一些目标代码模块经过组合形成的代码群。应用程序从函数库中调用函数实际上就是通过链接使应用程序能够从函数库中存取和使用目标代码，链接是把外部函数结合到一个应用程序中的进程。

与函数库链接有两种方法：静态链接和动态链接。静态链接发生在程序创建时，动态链接则在程序运行时发生。动态链接库 (Dynamic Link Library, DLL) 是用于动态链接的库，是 Windows 中的基础程序单元。DLL 使当前正在运行的几个程序能够共享一组函数的一个拷贝。大多数 Windows 程序结合使用静态链接和动态链接。

### 1. 静态链接

为了增强编程语言设计程序的能力，通常有两种扩展方法，首先大多数编程语言都提供了一些直接访问操作系统底层的能力；其次大多数语言允许创建或使用第三方的函数库，这些函数库中的函数可以直接结合进程中。

在静态链接这个过程中，包含函数的程序模块被预先编译成目标文件 (.obj)。在创建最终应用程序的可执行文件时，链接器扫描这些目标文件，将所有需求的目标代码从函数库中拷贝到可执行文件中去。当可执行文件被创建后，所有的函数调用在链接时就已经完全确定，换言之，程序访问库函数所需的全部寻址信息都是固定的，并且当程序运行时这些信息也保持不变。

静态链接的应用程序的可执行文件可以单独运行，因为在可执行文件中已经包含有应用程序所需要的所有函数的模块。

当多个使用同一个静态链接的应用程序的实例同时运行时，因为每个实例都拥有相同的代码，这导致重复占用宝贵的内存空间，并且当这些应用程序存储时也会因为重复的函数代码部分而浪费磁盘空间。

### 2. 动态链接

动态链接过程与静态链接过程的不同之处在于链接过程，尽管包含函数的程序模块仍然被预先编译成目标文件，但它不再拷贝到最终的可执行文件中，而是被链接成一种特殊的 Windows 可执行文件——动态链接库 (DLL)。当应用程序运行时，Windows 操作系统检查可执行文件，如果需要不包含在可执行文件自身中的函数，Windows 就自动装入指定的 DLL 文件，使得 DLL 中的所有函数都能被应用程序访问，仅到这个时候，Windows 才能确定每个



函数的地址并且将其动态地链接进应用程序。DLL 是在运行时被装载的，并且所有使用这个 DLL 的应用程序都可以在运行时共享这个 DLL 在内存中的同一个拷贝。

Microsoft Windows 提供动态链接库 (DLL) 的目的是让应用程序共享代码和资源。DLL 是一个包括了若干函数的可执行模块，为其他应用程序提供服务。Windows 利用 DLL 来提供所有 Windows 应用程序均可使用的代码和资源。此外，程序员也可以创建自己的 DLL，以便在自己的各个应用程序之间共享代码和资源。

DLL 是一种过程库，应用程序可以在运行时链接并使用它，这种链接不同于编译时的静态链接。这意味着 DLL 库的更新可以独立地进行，而且许多个应用程序可以共享同一个 DLL。Microsoft Windows 本身就是由若干个 DLL 组成的，其他应用程序可以调用这些库中的过程，完成窗口与图形的显示、内存管理或其他任务。这些过程有时被称为 Windows API 或应用程序接口。

使用 DLL 主要有如下目的：

- (1) 应用程序之间共享代码和资源；
- (2) 基于系统范围的消息过滤；
- (3) 创建设备驱动程序；
- (4) 提供开发复杂应用程序的库。

有两种方法可以用来装载和使用一个 DLL 文件，隐含的（或称为装载时的）动态链接，及明确的（或称为运行时的）动态链接。我们通常使用后者。

### 3. Visual Basic 和 DLL

Visual Basic 创建的可执行文件并非真正的 Windows 可执行文件，它们实际上包含了一种称为伪码或 P 码的特有中间代码，这种代码由一个特殊的 Visual Basic DLL 来解释，在 Visual Basic 以前的版本中这个 DLL 文件就是 VBRUN???.DLL (??? 对应于不同的版本号)。在 Visual Basic 6.0 中，这个文件已经更名为 MSVBVM60.DLL。

P-code，又称为伪代码，是介于 Basic 程序中的高级指令和计算机处理器执行的低级本机代码之间的一种中间步骤。在运行时刻，Visual Basic 将每一句伪代码转换成本机代码。如果将程序直接编译成本机代码，则取消了伪代码这一中间步骤。

可以使用类似于在 Visual C++ 中提供的调试环境的标准的本机代码调试工具来调试编译后的本机代码。调试过程中也可以使用在 Visual C++ 的语言环境中提供的优化和调试本机代码的选项，例如可以根据代码运行速度和大小优化代码。

按照缺省规定，Visual Basic 将应用程序编译成可运行的解释型或 P-code 程序。运行时，动态链接库 DLL 将翻译或解释执行程序中的指令。

这意味着 Visual Basic 不是按照正常的链接次序描述外部函数的，因此 Visual Basic 程序员使用 Declare 语句告诉 Windows 是哪个 DLL 包含了应用程序所需要的函数以及传递给该函数所需要的参数，以便 Windows 调用。

Visual Basic 使用 P-code 也带来一个副作用，就是 Visual Basic 本身不能用来创建动态链接库。由于 Windows 自身已经提供了数百个成熟的 API 函数，并且有很多附加的 DLL 可供 Visual Basic 调用，所以对于 Visual Basic 程序员来说，通常不需要考虑创建自己的 DLL 文件。