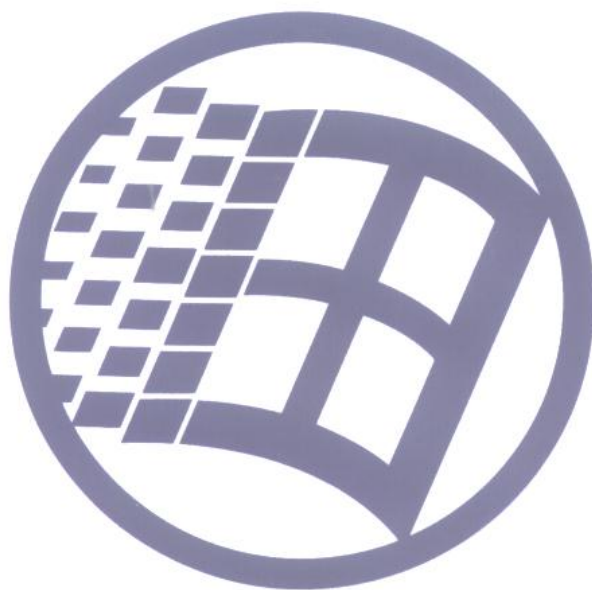


Windows CE —— 标准、权威、系统、完整的参考手册和开发工具



光盘内容:

1. 本书的英文版电子书
2. SDK for Windows CE,
Handheld PC Professional Edition 3.0
3. 范例源代码
4. 送“中文Windows 98直通车”
多媒体学习软件



Microsoft® **Windows CE** **User Interface Services Guide** 用户界面服务指南

〔美〕微软公司 著
希望图书创作室 译



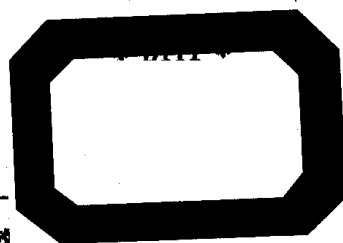
北京希望电脑公司
北京希望电子出版社
www.bhp.com.cn

Microsoft Press



美国微软出版社授权中文版系列书

HOPE Microsoft Windows CE开发人员指南系列



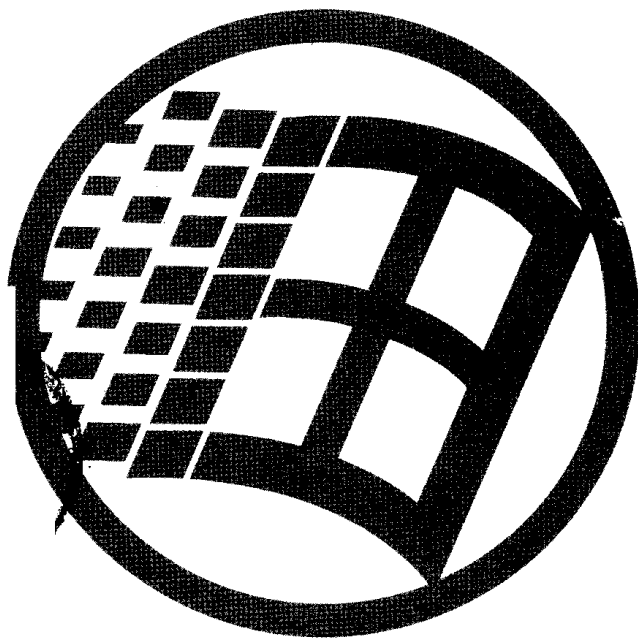
Windows CE —— 标准、权威、系统、完整

具



光盘内容:

1. 本书的英文版电子书
2. SDK for Windows CE,
Handheld PC Professional Edition 3.0
3. 范例源代码
4. 送“中文Windows 98直通车”
多媒体学习软件



Microsoft Windows CE User Interface Services Guide 用户界面服务指南

〔美〕微软公司 著
希望图书创作室 译



北京希望电脑公司
北京希望电子出版社
www.bhp.com.cn

Microsoft Press

内 容 简 介

Microsoft Windows CE 是 Microsoft 公司系统资源有限的设备开发的一种小型操作系统。因为 Windows CE 具有模块化的设计特征, 所以能够适用于多种设备, 包括从手提 PC(H/PC)到嵌入式系统。

本书是 Microsoft Windows CE 开发人员系列丛书之一。全书共分八章: Windows CE 用户界面服务概述, 使用窗口和消息, 使用资源, 创建控件, 使用图形, 使用声音, 接收用户输入, 设计 Windows CE 用户界面。书中最后的附录“窗口和控件样式”列出了 Windows CE 支持的所有窗口和控件样式。

本书是为 Windows CE 开发人员编写的, 是 Windows CE 应用程序开发人员必备的手册, 同时也可作为广大科技人员、大专院校相关专业的师生自学、教学的参考书。

本书配套光盘内容包括: 1. 本书的英文版电子书; 2. SDK for Windows CE, Handheld PC Professional Edition; 3. 范例源代码; 4. 送“中文 Windows 98 直通车”多媒体学习软件。

版 权 声 明

本书英文版名为“Microsoft Windows CE User Interface Services Guide”, 由微软出版社出版, 版权归微软出版社所有。本书中文版由微软出版社授权出版。未经出版者书面许可, 本书的任何部分不得以任何形式或任何手段复制或传播。

- 系 列 书 : Windows CE 开发人员指南系列
书 名 : Microsoft Windows CE User Interface Services Guide 用户界面服务指南
文 本 著 者 : (美) 微软公司 著 希望图书创作室 译
审 校 : 王玉玲
C D 制 作 者 : 微软出版社 希望多媒体开发中心
C D 测 试 者 : 希望多媒体测试部
责 任 编 辑 : 陆卫民 王玉玲
出版、发行者: 北京希望电脑公司 北京希望电子出版社
地 址 : 北京海淀路 82 号 100080
网址: www.bhp.com.cn E-mail: lwm@hope.com.cn
电话: 010-62562329, 62541992, 62637101, 62637102 (图书发行, 技术支持)
010-62633308, 62633309 (多媒体发行, 技术支持)
010-62613322-215 (门市) 010-62531267 (编辑部)
经 销 : 各地新华书店、软件连锁店
排 版 : 希望图书输出中心
C D 生 产 者 : 文录激光科技有限公司
文 本 印 刷 者 : 北京双青印刷厂
规格 / 开本 : 787×1092 16 开本 12.625 印张 223 千字
版次 / 印次 : 1999 年 9 月第 1 版 1999 年 9 月第 1 次印刷
印 数 : 0001~5000 册
本 版 号 : ISBN7-900024-53-0/TP·53
定 价 : 30.00 元 (1CD, 含配套书)

说明: 凡我社光盘配套图书若有缺页、倒页、脱页、自然破损, 本社发行部负责调换。

译 者 的 话

本书翻译是希望图书创作室同仁共同努力的结果，全书由屈茂原负责翻译和统稿，陈广萍、戴文芸、黄凯、朱磊、王蒿喜、杨光、吕兰参与了部分章节的翻译。在此对付出辛勤劳动的各位表示衷心的感谢。

在翻译过程中，我们力求尽善尽美，但错误在所难免，恳请读者指正。

前 言

“Microsoft Windows CE 开发人员指南”系列丛书包含了为基于 Microsoft Windows CE 操作系统的设备开发应用程序所需的全部资料。这一系列丛书包括以下四本书：

- 《Microsoft Windows CE 程序员指南》
介绍了 Windows CE 操作系统的体系结构。
解释了创建基于 Windows CE 的应用程序的低层细节，包括处理进程和线程、管理内存和电源、访问对象库和修改注册表。
介绍了如何把基于 Windows CE 的设备与台式计算机连接，以及如何在设备与台式机之间实现同步化和传输文件。
介绍了如何使用 Unicode 和本地化 Windows CE 应用程序。
- 《Microsoft Windows CE 用户界面服务指南》
介绍了为基于 Windows CE 的设备创建用户界面（UI）的全部相关任务，包括如何创建窗口和对话框，如何处理消息，以及如何增加菜单、控件和其他 UI 资源。
讨论了如何处理不同的用户输入方法（IM），例如键盘和触摸屏。
- 《Microsoft Windows CE 通信指南》
介绍了在基于 Windows CE 的设备上实现通信支持的基本概念，包括如何处理红外连接，如何开发电话应用，如何在应用程序中实现远程访问服务（RAS），如何处理网络互连和安全问题，如何使用 Windows 套接字，以及如何建立 Internet 连接。
- 《Microsoft Windows CE 设备驱动程序开发指南》
介绍了为基于 Windows CE 的设备开发设备驱动程序的过程。
解释了如何创建简单和流接口驱动程序，以及如何实现通用串行总线（USB）和网络驱动程序接口规范（NDIS）驱动程序。
随书附带的光盘包括书的英文电子版，以及如下内容：

内容	描述
Windows CE API 参考手册	说明了接口、函数、结构、消息以及 Windows CE 应用编程接口（API）的其他内容
设备驱动程序包 API	说明了创建 Windows CE 设备驱动程序所需的接口、函数、结构、消息以及其他 API 内容
Windows CE 的 MFC 库	说明了创建完整 Windows CE 应用程序所需的类、全局函数、全局变量和宏

内容	描述
Windows CE 主动模板库 (ATL)	说明了开发小巧、快速的 Windows CE 平台 ActiveX 控件所需的类、宏和全局函数
移动频道	描述了如何利用主动服务器页面 (ASP) 和频道定义格式技术实现在基于 Windows CE 的设备上脱机浏览 Web 站点
开发掌上 PC 应用程序	描述了如何使用掌上 PC 的 shell 程序, 如何管理内存和电源, 如何编程访问掌上 PC 导航控件, 以及如何为掌上 PC 应用程序设计用户界面 (UI)
开发手提 PC 应用程序	描述了如何使用手提 PC 的 shell 程序, 如何管理内存和电源, 以及如何在手提 PC 与台式计算机之间同步数据
开发汽车 PC 应用程序	描述了如何实现语音, 如何控制音频系统, 如何与车辆计算机交互, 如何与全球定位系统 (GPS) 通信, 以及如何为汽车 PC 应用程序设计有效的用户界面 (UI)

本书包括如下内容:

- **Windows CE 用户界面服务概述**

这一章简要介绍了图形、窗口、事件子系统 (GWES) 模块, 特别介绍了 Windows CE 用户界面服务。这一章解释了 GWES 中的特定组件以及如何用这些组件提供窗口管理, 事件处理和电源管理功能。GWES 还支持控件、菜单、对话框、资源、用户输入和图形设备界面 (GDI)。

- **使用窗口和消息**

这一章讨论 Windows CE 的窗口和消息功能, 重点在于创建和删除窗口、改变窗口大小和位置以及发送和响应消息。

- **使用资源**

这一章介绍了各种资源, 如对话框、消息框、菜单、图标、插入符、位图和定时器。

- **创建控件**

这一章解释了如何向 UI 添加标准控件。

- **使用图形**

这一章简要介绍了 Windows CE GDI, 也解释了如何使用各种图形对象, 如笔、刷子、形状、字体和直线。

- **使用声音**

这一章解释了如何用波形 API 向应用程序添加声音。

- **接收用户输入**

这一章讨论了 Windows CE 支持的各种用户输入方法, 如键盘、鼠标、手写和语音 API (SAPI), 还介绍了如何使用虚键。

- **设计 Windows CE 用户界面**

这一章简要介绍了 UI 设计概念, 并提出了有关使用窗口和对话框、添加控件、创

建图标、使用颜色以及布局的具体建议。

- **窗口和控件样式**

这一附录列出了 Windows CE 支持的窗口和控件样式。

关于本书包含的实例程序

《Windows CE 用户界面服务指南》一书包含的实例程序是用 Microsoft Visual C++ 6.0 和面向 Visual C++ 6.0 的 Microsoft Windows CE 工具包开发的。实例程序的源代码适用于运行 Microsoft Windows CE 手提 PC 专业版的 H/PC，然而实例程序中体现的编程概念适用于所有基于 Windows CE 的平台。下表说明了本书包括的实例程序。

实例程序名	描述
CeGDI	说明了如何创建和使用 GDI 对象，例如笔、刷子、调色板和区域。也说明了如何枚举字体，如何选择字体类型，以及如何用选择的字体显示文本
CePad	说明了如何注册窗口类以及如何创建窗口、菜单、对话框和编辑控件。也说明了如何打开和保存文件，以及如何在编辑控件中查找和替换文本字符串
CmdBand	注册工具栏和 rebar 控件类，并创建一个包含两个工具栏的命令条
Cmdbar	说明了如何向命令栏中添加命令按钮，如何创建和删除命令栏以及如何向命令栏中添加位图。也说明了如何用 InsertMenu 函数创建菜单。随后，用户可以用这个菜单在命令栏上显示最多三个按钮集和两个标准位图
FileView	说明了如何注册列表视图类和标题控件类，如何创建列表视图控件，以及如何将列表视图控件的窗口样式从文件视图转变为列表视图
Listview	说明了如何注册列表视图类和标题控件类，如何创建列表视图控件，以及如何将列表视图控件的窗口样式从文件视图转变为列表视图
Rebar	说明了如何注册 rebar 和工具栏控件类，如何创建一个包含工具栏和组合框的 rebar 控件，以及如何上下移动 rebar 控件
Toolbar	说明了如何注册工具栏控件类，如何创建工具栏和如何增加工具提示



目 录

第一章 Windows CE 用户界面服务概述 . . . 1	
1.1 GWES 组件模型 1	
1.2 窗口管理和事件处理 2	
1.3 GDI 支持 2	
1.4 用户输入支持 3	
第二章 使用窗口和消息 4	
2.1 邮寄消息 5	
2.2 发送消息 6	
2.3 接收和分发消息 6	
2.4 创建窗口 9	
2.5 显示窗口 13	
2.6 改变窗口大小和位置 13	
2.7 删除窗口 15	
2.8 创建实例程序 15	
第三章 使用资源 20	
3.1 创建菜单 21	
3.2 创建键盘加速键 25	
3.3 创建对话框 28	
3.4 创建插入符 34	
3.5 创建光标 35	
3.6 创建图标, 位图, 图像和字符串 36	
3.7 创建定时器 37	
第四章 创建控件 39	
4.1 使用窗口控件 39	
4.2 使用公共控件 62	
第五章 使用图形 118	
5.1 获取设备环境的句柄 118	
5.2 创建位图 121	
5.3 使用颜色 124	
5.4 使用调色板 125	
5.5 使用笔 129	
5.6 使用刷子 132	
5.7 打印 133	
5.8 使用区域 134	
5.9 创建形状和直线 135	
5.10 创建文本和字体 139	
第六章 使用声音 144	
6.1 使用 PlaySound 函数 144	
6.2 使用波形音频接口 146	
第七章 接收用户输入 155	
7.1 接收键盘输入 155	
7.2 接收输入笔输入 161	
7.3 接收来自输入面板的输入 162	
7.4 手写体识别 166	
第八章 设计 Windows CE 用户界面 . . . 171	
8.1 设计窗口和对话框 172	
8.2 设计菜单 174	
8.3 使用命令栏 175	
8.4 选择控件 176	
8.5 使用彩色和灰度级调色板 180	
8.6 创建图标和位图 181	
8.7 用户输入设备 182	
8.8 提供用户反馈 183	
附录 A 窗口和控件样式 184	
A.1 窗口和消息框样式 184	
A.2 控件风格 186	

第一章 Windows CE 用户界面服务概述

Microsoft Windows CE 操作系统将 Microsoft Win32 应用编程接口 (API) 用户界面 (UI) 和图形设备接口 (GDI) 组合为图形窗口事件子系统 (GWES) 模块 (Gwes.exe)。GWES 是用户、应用程序和操作系统之间的接口。

GWES 支持组成 Windows CE UI 的窗口、对话框、控件、菜单和资源。这一 UI 使用户能够控制应用程序。GWES 还可以通过位图、插入符、光标、文本和图标的形式为用户提供信息。

即使是缺乏图形 UI 的 Windows CE 平台也可以使用 GWES 的基本窗口和消息处理能力以及电源管理功能。

1.1 GWES 组件模型

因为 Windows CE 是模块化的操作系统 (OS)，所以 OEM 可以针对于特定平台选择软件模块来设计操作系统。OEM 厂商还可以通过从特定模块中选择部分组件对操作系统进行进一步定制。Windows CE 提供了几种预先测试的组件配置方案，这些配置方案可以分为三类：最小配置、中等配置和完全配置。

最小配置用于创建最基本的 Windows CE 版本，包括操作系统内核 (Core.dll) 和选定的 GWES 支持组件，例如消息处理、用户输入和电源管理。最小配置不显示 UI，也不包含窗口管理功能。

Windows CE 的中等配置包括操作系统核心并支持以下 GWES 功能：

- 消息处理及用户输入
- 电源管理
- 提示性发光二极管 (LED)
- GDI，包括 Microsoft TrueType 字体和光栅字体，文本绘制，调色板和打印。
- 可定制的触摸和校准 UI
- 网络 UI 对话
- 波形 API 管理器
- 输入方法管理器 (IMM)
- 窗口和对话管理
- 可定制 UI

一些最小配置也提供控制台、通知和公共控件支持。

完全配置用于创建 Windows CE 完全版，包括全部 GWES 组件支持。有关完全配置所包含组件的完整列表，请参阅“Windows CE 库”。

1.2 窗口管理和事件处理

任何 UI 的核心特征是窗口。在具有图形显示的 Windows CE 平台上，窗口是屏幕上的一个矩形区域，应用程序通过窗口来显示输出和接收用户输入。即使应用程序运行于不具备图形显示的设备上，它也需要通过窗口从操作系统接收消息。为了帮助管理应用程序的窗口，Windows CE 支持 Win32 API UI，并针对于典型 Windows CE 设备较小的显示大小进行了改制。下表说明了 Win32 API UI 和 Windows CE 之间的相同点和不同点。

Win32 API UI	Windows CE
下属窗口	完全支持对话框
向应用程序发送非客户消息	不支持
客户区	尽量扩大客户区，以支持由应用程序控制的命令栏。
多文档界面 (MDI)	不支持
级联菜单和位图菜单	不支持
属主绘制菜单	不支持
任何大小的图标	支持

非客户区的绘制和管理由操作系统控制；Windows CE 不会把处理窗口非客户区的消息发送给应用程序。

1.2.1 事件处理

Windows CE 是事件驱动的操作系统。消息通过 MSG 结构传递。Windows CE MSG 结构包含六个成员，不支持消息钩子。下表说明了 Windows CE MSG 结构支持的成员。

成员	描述
<i>hwnd</i>	接收消息的窗口句柄
<i>message</i>	表示消息号
<i>wParam</i>	表示附加的消息数据。确切意义依赖于消息号的值
<i>lParam</i>	表示附加的消息数据。确切意义依赖于消息号的值
<i>time</i>	表示发出消息的时间
<i>pt</i>	表示在发出消息前，最后一次点击屏幕的位置的坐标

1.3 GDI 支持

GDI 是控制文本和图形显示的 GWES 子系统。GDI 用于绘制直线、曲线、闭合图形、文本和位图。

GDI 用设备环境存储在指定设备上显示文本和图形所需的数据。存储在设备环境中的图形对象包括用于画线的笔，用于描绘和填充的刷子，用于输出文本的字体，用于复制和滚屏的位图，用于定义可用颜色的调色板以及剪贴区。Windows CE 支持用于在打印机上绘图的打印机设备环境，用于在视频显示器上绘图的显示设备环境以及用于在内存中绘图的内存设备环境。

下表说明了 Windows CE 支持的 GDI 功能。

GDI 功能	描述
光栅字体和 TrueType 字体	TrueType 字体可以放大或缩小，可以旋转。ROM 中存储了七种系统光栅字体，每种字体可以有多种大小。也可以增加自己的光栅字体。Windows CE 在一个系统中只支持一类字体：光栅字体或 TrueType 字体
定制彩色调色板以及调色和非调色的彩色显示设备	支持每个像素 1 位，2 位，4 位，8 位，16 位，24 位和 32 位的颜色位深度。2-bpp（每个像素 2 位）的颜色位深度是 Windows CE 所特有的
位块传输功能和光栅操作码	支持变换和组合位图
笔和刷子	支持虚线笔、宽线笔和实线笔，以及各种形状的刷子
打印	支持图形打印
光标	支持各种光标，包括用户定义光标或等待光标
形状绘制功能	支持绘制椭圆、多边形、矩形和圆角矩形

Windows CE GDI 不支持以下功能：

- 坐标空间变换函数，如 SetMapMode，GetMapMode，SetViewportExt 和 SetWindowExt 函数。坐标空间与设备空间等价。
- 时区变换 API
- MoveTo 和 LineTo 函数
- 彩色光标
- 动画光标

1.4 用户输入支持

通过配置 Windows CE 可以满足不同平台的用户输入要求。目前，键盘、输入面板、语音、鼠标和输入笔是 Windows CE 设备支持的输入方法。

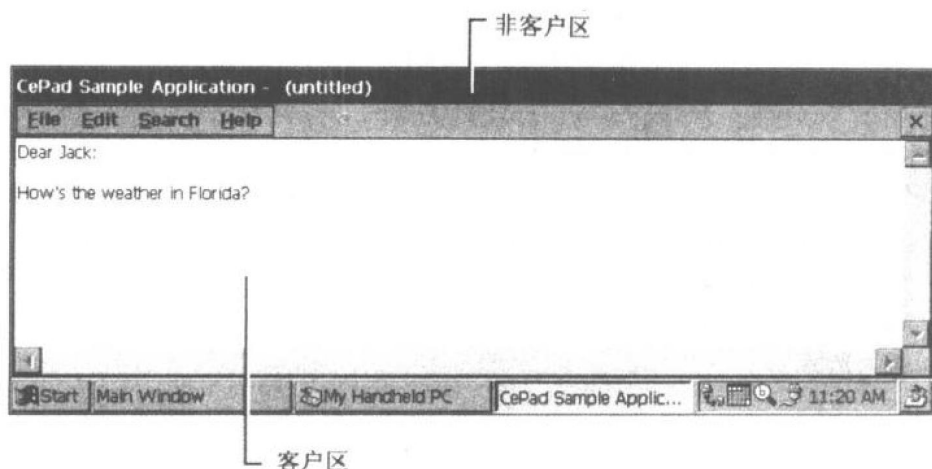
Windows CE 设备上的键盘特征与基于 Windows 的台式机平台类似。和台式机平台一样，Windows CE 支持热键。热键是将用户转到其他应用程序的击键或击键组合。通过热键，用户可以获得针对于特定目的的高优先级系统访问权，例如取消一个耗时的文件传输操作。

第二章 使用窗口和消息

窗口的外观和特性主要由它的内部属性以及与其他窗口的关系决定。通过设置窗口样式和扩展样式，或者通过调用改变窗口属性的函数，可以指定窗口的属性。

窗口总是矩形的，它们沿着垂直于屏幕的假想直线相互堆叠在一起。堆叠在一起的窗口称为 *z 序 (z-order)*。每个窗口在 *z 序* 中具有唯一的位置。在 *z 序* 中位置靠前的窗口位于位置靠后的窗口的前面或顶部。窗口在 *z 序* 中的位置影响它的外观。一个窗口可能部分或完全地遮挡住另一个窗口，这取决于它的位置、大小和在 *z 序* 中的位置。

一个窗口被分为非客户区和客户区：非客户区包括边框、滚动条和各种控件，客户区是非客户区所包围的部分。可以在客户区中绘图，但不能在非客户区内绘图。在 Microsoft Windows CE 操作系统中，窗口的非客户区完全由窗口管理器控制。Windows CE 不会把处理非客户区的消息发送给应用程序。下图显示了窗口的非客户区和客户区。



窗口可以被显示或隐藏，这取决于它的 `WS_VISIBLE` 样式被打开还是关闭。如果 `WS_VISIBLE` 样式被关闭，则窗口不显示在屏幕上。如果 `WS_VISIBLE` 样式被打开，则窗口可能显示在屏幕上或者不显示在屏幕上，这取决于它是否被其他窗口遮挡。用一个窗口覆盖另一个窗口或者移去覆盖的窗口不影响 `WS_VISIBLE` 样式。

每个窗口具有一个称为窗口句柄的唯一标识符。创建窗口时，即获得相应的窗口句柄，然后就可以利用窗口句柄调用有关窗口的函数。句柄对于具有多个子窗口的应用程序非常有用。调用 `SetWindowLong` 函数可改变窗口句柄，调用 `GetWindowLong` 函数可获得窗口句柄。

每个应用程序至少拥有一个窗口，不管它是否拥有图形用户界面 (UI)。这是因为

Windows CE 是消息驱动型操作系统，而窗口是应用程序接收消息的途径。每个窗口必须与一个称为窗口过程或 WinProc 的特殊函数相关联。Windows CE 通过调用这个窗口过程向应用程序传递消息。

消息由消息标识符和其他可选参数组成。消息标识符是用于识别消息的命名常数。当窗口过程接收到消息时，它用消息标识符决定如何处理消息。例如：

- 当一个窗口被创建时，WM_CREATE 消息被发往这个窗口。
- 当一个窗口被删除时，WM_DESTROY 消息被发往这个窗口。
- 当窗口的客户区被改变而必须重绘时，WM_PAINT 消息被发往这个窗口。

消息参数包含窗口过程处理消息所需的数据或者数据的位置。消息参数的意义和值取决于消息标识符。消息参数可以包含整数、组合位标志、指向包含附加数据的结构指针、或者其他信息。窗口必须检查消息标识符以确定如何解释消息参数。术语“消息”既可用于表示标识符也可用于表示标识符和参数的整体。具体意义通常需要参考上下文。

通过把消息数据作为参数传递给窗口过程，系统就把消息发送给了窗口过程。随后窗口过程就可以对消息执行适当的操作；它可以检查消息标识符，并用消息参数中指定的数据处理消息。

如果窗口过程不能处理某个消息，它就应该对消息进行缺省处理。窗口过程通过调用 DefWindowProc 函数对消息进行缺省处理，这一函数执行缺省操作并返回一个消息结果。然后窗口过程必须将这一结果返回，作为它自己的消息结果。大多数窗口过程只处理少数几种消息，而将其他消息传递给 DefWindowProc 函数处理。

窗口过程可以被共享。接收消息的指定窗口的句柄可以作为窗口过程的一个参数。

除了拥有窗口过程外，每个 Windows CE 应用程序必须以 WinMain 函数作为它的进入点函数。WinMain 函数执行一系列任务，包括注册主窗口的窗口类和创建主窗口。WinMain 函数通过调用 RegisterClass 函数注册主窗口类，通过调用 CreateWindowEx 函数创建主窗口。WinMain 函数不需要自己做这些事情，它可以调用其他函数来执行这些任务之一或全部。若想获得有关 WinMain 函数的实例程序，请参阅本章后面的“创建实例应用程序”部分。

WinMain 函数必须自己执行的一项任务是创建消息循环。消息循环从线程的消息队列获取消息，并将它们发往适当的窗口过程。消息队列负责协调指定线程的消息传输。每个线程只能拥有一个消息队列。当一条消息被发往某个窗口时，它就被加入到窗口线程的消息队列中。线程负责接收和发送消息。向窗口传递消息有两种方法：邮寄消息和发送消息。

2.1 邮 寄 消 息

PostMessage 函数用于邮寄消息。PostMessage 函数将消息标识符和参数组合为一条消息，并将其加入到接收窗口的消息队列之中。最终，接收窗口的消息循环将消息从消息队

列中删除，并将其发送给适当的窗口过程。

PostMessage 函数是一个异步函数。Windows CE 不对被邮寄消息的发送线程和接收线程进行同步。当 **PostMessage** 函数返回时，不能保证接收窗口的窗口过程对消息进行了处理。事实上，如果消息被邮寄给自己，则调用返回时，窗口进程还没有对消息进行处理。

可以在不指定窗口的情况下邮寄消息。如果在调用 **PostMessage** 函数时提供了一个 **NULL** 窗口句柄，则消息就被邮寄给与当前线程相关联的消息队列。因为没有指定窗口句柄，所以必须直接在消息循环中处理消息。这样就创建了一条适用于整个应用程序而不是特定窗口的消息。

2.2 发送消息

SendMessage 函数用于向窗口发送消息。与 **PostMessage** 函数不同，**SendMessage** 函数是一个同步函数。直到接收窗口的窗口过程对消息进行处理后，函数才返回。

当需要某个窗口过程立即执行一项任务时，通常发送一条消息。**SendMessage** 函数将消息直接发送给接收窗口的窗口过程。**SendMessage** 函数等到窗口过程完成消息处理后才返回消息结果。父窗口和子窗口常以互相发送消息的方式进行通信。例如，一个以编辑控件为子窗口的父窗口，可以通过向子窗口发送消息来设置控件的文本。控件也可以向父窗口发送消息，以通知父窗口用户对文本的改变。

如果接收线程与发送线程相同，则 **SendMessage** 函数就直接调用窗口过程。如果接收线程与发送线程不同，则两个消息队列就对消息传递进行同步。发送线程一直等到接收线程对消息进行处理后才能继续执行。如果接收线程不在执行消息循环，则它就不会对消息进行处理。因而，如果向不在执行消息循环的线程的窗口发送了一条消息，则发送线程将停止响应。

2.3 接收和分发消息

GetMessage 函数用于接收消息。当一个线程调用 **GetMessage** 函数时，Windows CE 检查该线程的消息队列里是否有新来的消息。Windows CE 按照如下顺序处理消息：

1. Windows CE 检查由 **SendMessage** 函数加入到队列中的消息。当系统从队列中删除消息后，它就在 **GetMessage** 函数内将消息分发给适当的窗口过程。这保证了发送者和接收者的消息队列能够保持同步。接收者处理发送的消息必须调用 **GetMessage** 函数。
2. 如果没有发送的消息，Windows CE 检查由 **PostMessage** 函数加入到队列中的消息。
3. 如果没有邮寄的消息，Windows CE 检查队列中用户输入系统邮寄的消息。

以较低的优先级处理用户输入消息, 可以确保系统在转到下一条输入消息之前, 已经对上一条输入消息和由它导致的邮寄消息彻底进行了处理。

4. 如果没有邮寄的输入消息, Windows CE 检查由 PostQuitMessage 函数加入到队列中的 WM_QUIT 消息。
5. 如果没有邮寄的退出消息, Windows CE 检查由窗口系统加入到队列中的 WM_PAINT 消息。
6. 如果没有绘图消息, Windows CE 检查由定时系统加入到队列中的 WM_TIMER 消息。

当 GetMessage 函数接收到任何上述消息时, 它就返回消息内容。线程必须调用 DispatchMessage 函数将消息分发给适当的窗口过程。如果接收到的消息是 WM_QUIT 消息, GetMessage 函数的返回值为 0, 这将导致线程终止其消息循环。

系统在消息循环中调用 GetMessage 函数时分发消息, 应用程序通过在消息循环中调用 DispatchMessage 函数来分发消息。

在调用 DispatchMessage 函数分发 GetMessage 函数接收到的消息之前, 可能需要处理一些消息。最常用的处理例程是 TranslateMessage, TranslateAccelerator 和 IsDialogMessage 函数。这些例程的一部分可以在内部分发消息, 因为应用程序不再需要在主消息循环中调用 DispatchMessage 函数。

通常在调用 DispatchMessage 函数之前调用 TranslateMessage 函数。TranslateMessage 函数可以确定键盘消息所携带的字符, 并将这些字符邮寄给消息队列, 以便在下次消息循环中被取走。

若想拦截键盘消息和产生菜单命令, 可以调用 TranslateAccelerator 函数。调用 IsDialogMessage 函数可以确保无模式对话框的正确操作。

用 GetMessage 函数可以从队列中删除一条消息。PeekMessage 函数用于检查一条消息, 但并不将消息从队列中删除。这一函数用该消息的信息填充一个 MSG 结构。使用 PeekMessage 函数时要格外小心, 因为它不会因等待消息事件而阻塞, 应用程序因此能够不考虑队列中的消息而继续处理。在基于 Windows CE 的应用程序中, 如果应用程序不因等待消息或其他事件而阻塞, 则内核就不会将 CPU 转为低功耗模式, 这种情况将很快耗尽设备的电池。

处理消息时, Windows CE 既支持系统定义的消息, 也支持应用程序定义的消息。系统定义的消息的标识符的范围为从 0 到 0x3ff。从 0x400 到 0x7fff 的消息标识符是应用程序定义的消息的可用范围。

系统定义的消息分为两类: 通用窗口消息和特殊目的消息。通用窗口消息可适用于所有窗口, 特殊目的消息适用于特定类型窗口。通用窗口消息覆盖了很广的信息和请求范围, 包括输入设备消息和键盘输入消息, 也包括窗口创建和管理消息。

消息的符号常量的前缀通常用于识别消息所属的种类。例如, 通用窗口消息都以 WM

开头，而只适用于按钮控件的消息以 **BM** 开头。

下表列出了 Windows CE 的消息类型。

消息类型	描述
BM	按钮消息
BN	按钮通知
CB	组合框消息
CBN	组合框通知
CDM	公共对话框消息
CDN	公共对话框通知
CPL	控制面板消息
DB	对象库消息
DM	对话框缺省下压按钮消息
DTM	日期和时间选择器和超文本标记语言 (HTML) 浏览器消息
DTN	日期和时间选择器通知
EM	编辑控件消息
EN	编辑控件通知
HDM	标题控件消息
HDN	标题控件通知
IMN	输入上下文消息
LB	列表框控件消息
LBN	列表框控件通知
LINE	线型设备消息
LVM	列表视图消息
LVN	列表视图通知
MCM	月历消息
MCN	月历通知
NM	多种控件发送的消息
PBM	进度条消息
PSM	属性表单消息
PSN	属性表单通知
RB	rebar 消息
RBN	rebar 通知
SB	状态栏消息
SBM	滚动条消息
STM	静态栏消息
STN	静态栏通知
TB	工具栏消息
TBM	轨迹条消息
TBN	轨迹条通知
TCM	卡片控件消息
TCN	卡片控件通知
TVM	树形视图消息
TVN	树形视图通知

(续表)

消息类型	描述
UDM	上下控件消息
UDN	上下控件通知
WM	通用窗口消息

可以定义应用程序窗口本身使用的消息。如果自己创建消息，则一定要确保接收消息的窗口过程能够正确解释和处理消息。操作系统不解释应用程序定义的消息。

在一些情况下，需要用消息与其他进程控制的窗口进行通信。在这种情况下，需要调用 RegisterWindowMessage 函数注册一个消息标识符。返回的消息号确保了在整个系统中的唯一性。通过使用这个函数，防止了由于不同的应用程序因不同的目的使用相同的消息标识符而可能引起的冲突。

Windows CE 不支持钩子消息，因为钩子所需的额外处理开销会降低 Windows CE 设备的性能。

在应用程序中处理消息时，要小心 WM_HIBERNATE 消息。Windows CE 用 WM_HIBERNATE 消息通知应用程序系统资源不足。当应用程序收到这条消息时，应释放尽可能多的资源。系统每隔 5 秒钟检查一次内存状态。每个使用适量系统资源的 Windows CE 应用程序都应实现对 WM_HIBERNATE 消息的处理。如果一个应用程序窗口不可见，则它就不能收到 WM_HIBERNATE 消息。这是因为 WM_HIBERNATE 消息只发送给在任务栏上有相应按钮的应用程序，而只有可见窗口才具有这一特征。隐藏的窗口不能收到这条消息，即使它是一个顶级重叠的窗口。

2.4 创建窗口

每个窗口都是窗口类的一个实例。窗口类是创建窗口的模板。开发应用程序时，必须注册所有创建窗口的窗口类。为了简化创建窗口的过程，Windows CE 提供了几个系统定义的窗口类。由于 Windows CE 自动注册这些窗口类，所以可以用这些类直接创建窗口。

CreateWindow 和 CreateWindowEx 函数用于创建窗口。两个函数的唯一区别是 CreateWindowEx 函数支持扩展样式参数 dwExStyle，而 CreateWindow 函数不支持。这两个函数都有许多参数，这些参数用于指定被创建的窗口的属性。在 Windows CE 中，CreateWindow 函数是通过调用 CreateWindowEx 函数的宏实现的。

Windows CE 还包含一些辅助函数，如 DialogBox，CreateDialog 和 MessageBox 函数，用于创建特殊目的窗口，如对话框和消息框。

CreateWindowEx 函数的语法说明如下：

HWND
CreateWindowEx(