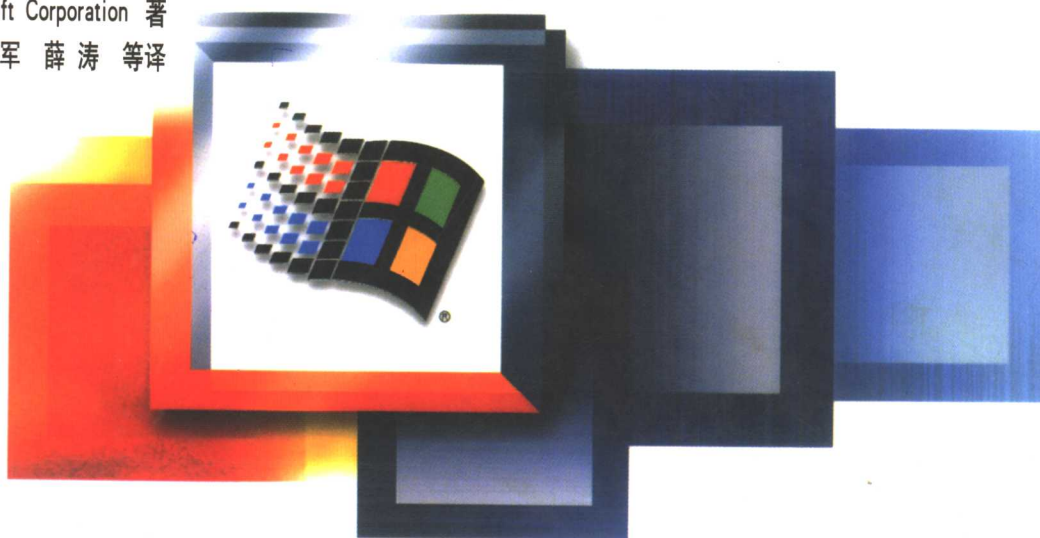


Microsoft Windows 2000 Driver Development Kit,
Volume 1: Design Guide

(美) Microsoft Corporation 著
冯博琴 朱丹军 薛涛 等译



Windows 2000

驱动程序开发大全

第1卷 设计指南

微软公司核心技术书库

Windows 2000驱动程序 开发大全

第1卷 设计指南

(美) Microsoft Corporation 著

冯博琴 朱丹军 薛 涛 等译



机械工业出版社
China Machine Press

本书为“Windows 2000驱动程序开发大全”的第1卷——设计指南。全书分五篇全面详尽地介绍了编写Windows 2000/98驱动程序所需的技术内容。包括利用Windows Driver Verifier构建、测试和调试Windows 2000驱动程序；怎样生成Windows Driver Model(WDM)驱动程序；描述了如何支持即插即用、电源管理特性及设备的设置和安装；详细介绍了编写内核模式、图形和网络驱动程序的细节。

由于篇幅所限，DDK (Driver Development Kit)文档中有少部分内容没有包含在本套书中，但随套书附赠的光盘(见套书第2卷)中包含了DDK文档的完整信息。

Microsoft Corporation: Microsoft Windows 2000 Driver Development Kit, Volume I: Design Guide.

Copyright © 2001 by Microsoft Corporation.

Original English language edition copyright © 2000 by Microsoft Corporation.

Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U.S.A. All rights reserved.

本书中文简体字版由美国微软出版社授权机械工业出版社出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

本书版权登记号：图字：01-2000-1321

图书在版编目(CIP)数据

Windows 2000驱动程序开发大全 第1卷 设计指南 / 美国微软公司著；冯博琴等译. -北京：机械工业出版社，2001.8

(微软公司核心技术书库)

书名原文：Microsoft Windows 2000 Driver Development Kit, Volume I: Design Guide
ISBN 7-111-08954-5

I. W… II. ① 美… ② 冯… III. 窗口软件，Windows 2000-程序设计-手册 IV. TP316.7-62

中国版本图书馆CIP数据核字(2001)第038018号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑：陆 鑫 刘立卿

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2001年8月第1版第1次印刷

787mm×1092mm 1/16·76.75印张

印数：0 001-4 000册

定价：155.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

译 者 序

本书是开发Windows 2000和Windows 98驱动程序的官方开发指南。全书分为五篇，分别是：驱动程序编写者指南，介绍构建、测试和调试Windows 2000驱动程序的方法；即插即用、电源管理和设置设计指南，覆盖了即插即用和电源管理这两个Windows 2000的主要特性；内核模式驱动程序设计指南，阐述了需求和系统定义的某些类型的外设驱动程序接口，包括大部分WDM驱动程序；图形驱动程序设计指南，为显示器和视频微端口驱动程序以及打印机驱动程序描述了系统定义的接口和实现需求；网络驱动程序设计指南，介绍了网络驱动程序的系统定义的接口和实现需求，并包括关于Microsoft Windows 2000中的新技术和驱动程序级接口的信息。通过本书，读者将获得编写Windows 2000/98驱动程序所需的技术。

本书由冯博琴教授组织翻译。刘路放、董文乾、薛涛、冯岚负责第一、二篇的翻译工作；刘宁、李华、王军鹏负责第三篇的翻译工作；朱丹军、肖小勃、薛涛负责第四篇的翻译工作；张险峰、张明龙、王刚、张柯、张俊武负责第五篇的翻译工作。全书由朱丹军、薛涛、刘宁、张险峰负责统稿。

由于本书的内容较新，译者虽然尽心，但限于水平，书中难免有不足之处，希望同行不吝赐教。

译 者

2000年10月

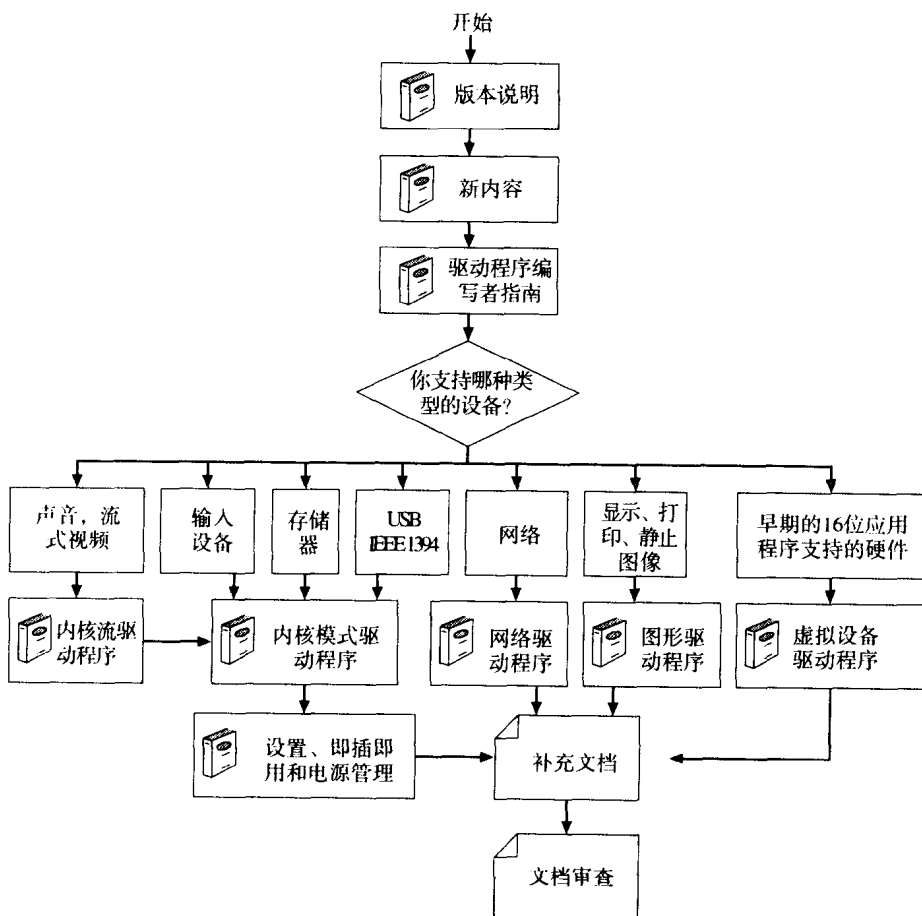
DDK文档说明

Microsoft Windows 2000驱动程序开发包（Driver Development Kit, DDK）文档概述提供了详尽的信息，这些信息对编写Windows 2000平台上任何类型的驱动程序都是必不可少的。本书也描述了Windows Driver Model（WDM）驱动程序，这些驱动程序在Windows 2000和Microsoft Windows 98平台之间是源码级兼容的。

文档构成

下图提供了本文档的构成指南。它说明了你需要读DDK的哪一部分，并基于你将要编写的驱动程序类型，以怎样的顺序发现最为有用的信息。

下面是对DDK每一部分的简单描述，在该线路的在线版本上，你可通过点击各本书的图标或者标题得到进一步的信息。



版本说明

一个独立的文件，即install.htm，包括了从附送的CD-ROM中安装在线DDK的系统需求。它也包含了一些特有的、重要的、最新的信息。该文件放在安装DDK的根目录下，也在CD-ROM安装盘的DDK根目录下。

新内容

在新内容部分中包括关于Microsoft Windows 2000里的新技术和驱动程序级接口的信息。

驱动程序编写者指南

“驱动程序编写者指南”参见(本书第一篇)告诉你怎样构建、测试和调试Windows 2000驱动程序。它包括下列信息：

- 驱动程序开发环境。
- DDK提供的BUILD实用程序。
- 测试驱动程序的Driver Verifier实用程序。
- 可用来调试驱动程序的工具和例程。

设置、即插即用和电源管理

“设置、即插即用和电源管理设计指南”(参考本书第二篇)和参考手册(参见本套书第2卷第一篇)覆盖了Windows 2000的两个主要特性：即插即用和电源管理。通过设计，WDM驱动程序支持即插即用和电源管理。如果可能的话，任何专为Windows 2000编写的其他驱动程序均应该支持即插即用和电源管理。这部分也包括了设备设置。

该文档的这一部分提供设置、即插即用和电源管理的总体描述。对驱动程序特定信息，驱动程序编写者也应该查阅本文档中特定种类的驱动程序。

“设置、即插即用和电源管理设计指南”的第一部分包括关于即插即用和电源管理的介绍。它也描述了驱动程序必须提供的技术支持类型。

“设置、即插即用和电源管理设计指南”的第二部分包括一个即插即用的概述和驱动程序必须做什么来支持即插即用。

“设置、即插即用和电源管理设计指南”的第三部分包括关于编写支持电源管理的驱动程序的信息。这些章节提供了一个从驱动程序编写者角度所看到的电源管理总览，并描述了驱动程序必须提供的支持。

“设置、即插即用和电源管理设计指南”的第四部分包括设备设置的信息。

“设置、即插即用和电源管理参考手册”(参见本套书第2卷《Windows 2000驱动程序开发大全第2卷参考手册》的第一篇)的第一部分包括结构、IoXxx例程和IRP的参考信息，这些参考信息包含在即插即用支持里。

“设置、即插即用和电源管理参考手册”的第二部分包括结构、PoXxx例程和IRP的参考信息，这些参考信息包含在电源管理支持里。

“设置、即插即用和电源管理参考手册”的第三部分包括设备设置的参考信息，包括INF文件格式的参考页、设置例程、设备安装功能及结构，以及文本模式设置文件格式。

内核模式驱动程序

“内核模式驱动程序设计指南”（参考本书第三篇）和参考手册（参见本套书第2卷第二篇）覆盖了需求和系统定义的某些类型的外设驱动程序接口，包括大部分WDM驱动程序。网络驱动程序和图形驱动程序包括在DDK不同的部分里。文件系统驱动程序和过滤器包括在可安装的文件系统（IFS）包里。

“内核模式驱动程序设计指南”的第一部分给Windows 2000内核模式驱动程序编写者提供了有用的信息。它覆盖了各种驱动程序，勾画出了系统的I/O体系结构，并描述了每个驱动程序必须支持的例程。

“内核模式驱动程序设计指南”的第二至第九部分没有包含在本书中。读者可在附送的CD-ROM上发现这些信息，包括串行和并行设备、存储、输出、视窗管理手段（Windows Management Instrumentation, WMI）和系统管理总线（System Management Bus, SMB），等等。

“内核模式驱动程序参考手册”（参见本套书第2卷第二篇）已被更新，以便包括支持例程的参考内容，Windows 2000和WDM驱动程序能够调用这些例程。WDM驱动程序编写者能够通过在每个支持例程参考里检查新的“Include”部分决定一个特别的支持例程是否能够被调用，从而了解那个支持例程是否在wdm.h里输出，即在WDM驱动程序的主DDK头文件里输出。

“内核模式驱动程序参考手册”的第二部分包括对IRP的描述。不同于PnP和电源，这些例程对内核模式设备来说是总体上的描述；但对使用串行和并行端口的设备而言，则是详尽的描述。对PnP和电源IRP，参见“设置、即插即用和电源管理参考手册”（即本套书第2卷第一篇）；对设备类型的IRP细节，参看这些设备类型的单个文档。

“内核模式驱动程序参考手册”的第三部分包括SCSI端口和SCSI微端口例程、新的微类例程和相关结构的更新过的参考手册。

“内核模式驱动程序参考手册”的第四部分包括HID驱动程序和HID客户使用的系统例程和结构的参考手册。这部分也描述了I/O请求和回调例程，这些I/O请求和回调例程受键盘和鼠标设备（kbdclass, mouclass, i8042prt）主要的非HID系统驱动程序支持。

“内核模式驱动程序参考手册”的第五部分覆盖了USB驱动程序。

“内核模式驱动程序参考手册”的第六部分包括了由IEEE 1394驱动程序所使用的系统定义的例程和结构的参考手册。

“内核模式驱动程序参考手册”的第七部分包括了关于PCMCIA驱动程序的信息。

“内核模式驱动程序参考手册”的第八部分包括了关于SMB客户驱动程序的信息。

“内核模式驱动程序参考手册”的第九部分描述了WMI内核模式数据提供者。

内核流驱动程序

“内核流驱动程序设计指南”和参考手册覆盖了多媒体设备驱动程序的实现，如声卡、DVD播放器、数码相机和电视调谐卡。内核流体系结构是WDM最基本的部分，为Windows 98和

Windows 2000所支持。

“内核流驱动程序设计指南”和参考手册的总体部分包括一个内核流总览。所有的内核流驱动程序都依赖本部分里的概念。

“内核流驱动程序设计指南”和参考手册中的声频驱动程序部分描述了怎样为所有的声频设备(包括声卡或者外部声音设备)来编写驱动程序。本部分包括Microsoft Direct Music驱动程序文档。

“内核流驱动程序设计指南”和参考手册的视频捕获驱动程序部分描述了怎样为流式视频(streaming video)设备(如数码相机和电视调谐卡)编写驱动程序。

图形驱动程序

“图形驱动程序设计指南”(参见本书第四篇)和参考手册为下面的图形驱动程序描述了系统定义的接口和实现需求:

- 显示器和视频微端口驱动程序
- 打印机驱动程序

“图形驱动程序设计指南”的第一部分包括与显示和打印机驱动程序有关的信息。由于Windows 2000的推出,为了反映设备驱动程序接口(DDI)和系统提供的图形驱动程序接口(GDI)额外的和变化的信息,所有的章节已经被更新。

“图形驱动程序设计指南”的第二部分包括显示和视频微端口驱动程序的详细信息,包括GDI、DDI、Microsoft DirectDraw、Microsoft Direct3D和微端口驱动程序信息等。

“图形驱动程序设计指南”的第三部分包括打印机的详细信息。这些章节描述了打印机驱动程序和Windows 2000假脱机打印系统的体系结构。第三部分解释了怎样为新的打印机提供支持和怎样替换假脱机打印的定制组件。

“图形驱动程序设计指南”的第四部分,覆盖了微软的静止图像体系结构,它不包括在本书里,但是包括在附送的CD-ROM的DDK文档里。

网络驱动程序

“网络驱动程序设计指南”(参见本书第五篇)和参考手册描述了以下类型的网络驱动程序的系统定义的接口和实现需求:

- 微端口NIC驱动程序。
- 中间网络驱动程序,它被置于微端口NIC驱动程序和协议驱动程序之间。
- 网络传输堆栈里的协议驱动程序。

系统定义的网络驱动程序接口包括网络驱动程序接口规范(NDIS)和传输驱动程序接口(TDI)。

“网络驱动程序设计指南”的第一部分覆盖了所有Windows 2000网络驱动程序的常见主题。这些章节讨论了Windows 2000的网络驱动程序的体系结构,网络驱动程序的编程考虑,以及网络驱动程序的设置。

“网络驱动程序设计指南”的第二部分覆盖了微端口NIC驱动程序,包括WAN微端口。

“网络驱动程序设计指南”的第三部分覆盖了中间层网络驱动程序、协议驱动程序和

Windows Sockets的传输辅助程序DLL。TDI设计信息没有包括在本卷里，但是包括在完全的在线DDK文档里。

“网络驱动程序设计指南”的第四部分描述了面向连接的媒体的NDIS支持，如异步传输模式(ATM)和ISDN。

“网络驱动程序参考手册”的第一部分描述了NDIS提供的函数和宏，NDIS驱动程序使用这些函数和宏与系统及其他部分相互通信。第一部分也描述了NDIS定义的和系统定义的结构，NDIS驱动程序使用这些结构。

“网络驱动程序参考手册”的第二部分描述NDIS对象标识符(OID)，这些是能被NDIS或其他驱动程序查询和设置的驱动程序维护的信息元素。

“网络驱动程序参考手册”的第三部分描述了TDI信息。这些章覆盖了：TDI传输驱动程序例程，传输驱动程序的TDI IOCTL，TDI函数，TDIBuildXxx宏和函数，TDI客户和PnP通知，TDI结构和Windows Sockets辅助程序功能。

“网络驱动程序参考手册”的第四部分介绍网络组件的通知对象。通知对象用于显示网络组件的属性页，并在网络配置中向该组件提供程序化控制。

使用在线DDK文档

该文档包含了驱动程序开发人员所必需的信息，包括Microsoft Windows 2000/WDM驱动程序接口，设备驱动程序的完整描述，设备驱动程序及依赖于底层设备驱动程序的较高层驱动程序的相关设计要求。

对于WDM驱动程序，使用该文档编写Windows 2000兼容的驱动程序同样与Windows 98兼容。

警告 DDK头文件中未归档那些将要变化或删除的不需注意的主题。

在线DDK文档完全可交叉索引并在线查找，HTML辅助观看器的一些主要特征描述如下。

查找信息

在文档中确定信息的两个最强大的功能是关键字索引和全文查找(查询)。

• 关键字索引

关键字可帮助你确定所列标题的主题。在关键字索引中可列出“参考手册”和“指南”主题，提供了确定有关设计准则、函数、结构、消息等信息的快捷方式。单击Index标签可进行关键字查找。

• 查询

查询提供了一种全文本查找的功能，使用该功能可查找整个文档集，或查找所选的书和某个词、短语或关键概念等，查找结果列在Search标签下的窗口中。

下列DDK CD-ROM文档内容没有包含在本套书中：

• 驱动程序编写者指南4.0。

调试驱动程序。

- 设置、即插即用和电源管理第三部分，5.0。
编写电池微类驱动程序。
- 内核模式驱动程序设计指南第二至第九部分。
串行和并行驱动程序。
存储驱动程序。
输入设备驱动程序。
USB驱动程序。
IEEE 1394驱动程序。
PCMCIA驱动程序。
SMB客户驱动程序。
支持WMI。
- 图形驱动程序设计指南第四部分。
静止图像驱动程序。
- 网络驱动程序设计指南第三部分，4.0~6.0。
TDI、例程、宏和回调。
TDI操作。
Windows Sockets的传输辅助程序DLL。
- 虚拟设备驱动程序。

增补的文档

注意 在这些DDK参考手册里，“图形驱动程序参考手册”和“网络驱动程序参考手册”没有包括在本套书里，这些参考仅仅在线可用。

目 录

译者序

DDK文档说明

第一篇 驱动程序编写者指南

第1章 驱动程序开发环境	2
1.1 自由构建和检查构建	2
1.2 调试环境	3
第2章 测试驱动程序	4
2.1 Driver Verifier	4
2.2 Driver Verifier的能力	4
2.3 Driver Verifier对图形驱动程序的能力	9
2.4 激活和监视Driver Verifier	12

第二篇 即插即用、电源管理和设置设计指南

第一部分 即插即用和电源管理的要求

第1章 即插即用和电源管理的介绍	21
1.1 什么是即插即用	21
1.1.1 PnP组件	22
1.1.2 PnP的支持级别	23
1.2 什么是电源管理	23
1.2.1 电源管理的最初含义	24
1.2.2 电源管理的支持级别	24
1.2.3 全系统范围电源管理的概述	24
1.2.4 电源状态	25
1.3 设备树	26
1.4 驱动程序层次和设备对象	27
1.4.1 驱动程序种类	27
1.4.2 例子：驱动程序层次	30
1.4.3 设备对象种类	32
1.4.4 例子：设备对象	34

第2章 PnP和电源管理必需的驱动

程序支持

2.1 必需的PnP支持的概述	37
2.2 PnP和电源管理DriverEntry例程	37
2.3 PnP和电源管理AddDevice例程	38
2.4 DispatchPnP例程	41
2.5 DispatchPower例程	41
2.6 PnP和电源管理Unload例程	42

第二部分 即插即用

第3章 理解PnP	43
3.1 PnP驱动程序设计准则	43
3.2 PnP和设备树	44
3.3 PnP设备状态	45
3.4 动态地添加一个新的PnP设备	46
3.5 硬件资源	53
3.6 在驱动程序里使用GUID	55
3.6.1 定义和导出新的GUID	55
3.6.2 在驱动程序代码里包含GUID	56

第4章 处理即插即用IRP的规则	58
4.1 PnP IRP需求	58
4.2 沿设备堆栈向下传递PnP IRP	59
4.3 延迟PnP IRP的处理直到低层的驱动程序完成	60

第5章 启动、停止和删除设备	65
5.1 启动一设备	65
5.1.1 启动功能驱动程序里的一个设备	65
5.1.2 启动过滤器驱动程序里的一个设备	67
5.1.3 启动总线驱动程序里的一个设备	67
5.1.4 启动设备设计注意事项	67
5.2 为了资源重新平衡而停止一个设备	68
5.2.1 理解何时停止发布IRP	68
5.2.2 处理IRP_MN_QUERY_STOP_	

DEVICE请求	69	7.1.2 支持有不完全配置寄存器地址 的PC Card	94
5.2.3 处理IRP_MN_STOP_DEVICE请求	71	7.1.3 支持有不完全配置寄存器的PC Card	96
5.2.4 处理IRP_MN_CANCEL_STOP_ DEVICE请求	71	7.2 支持多功能PCI设备	99
5.2.5 当一个设备被暂停时, 保留进 入的IRP	72	7.3 支持其他总线上的多功能设备	100
5.3 删除一个设备	73	7.4 使用系统提供的mf.sys	100
5.3.1 理解何时发布删除IRP	73	7.5 为一个多功能设备生成资源图	103
5.3.2 处理一个IRP_MN_QUERY_REMOVE_ DEVICE请求	76		
5.3.3 处理一个IRP_MN_REMOVE_ DEVICE请求	77	第三部分 电源管理	
5.3.4 处理一个IRP_MN_CANCEL_ REMOVE_DEVICE请求	81	第8章 在驱动程序内支持电源管理	105
5.3.5 处理一个IRP_MN_SURPRISE_ REMOVAL请求	82	8.1 内核模式下的电源管理组件	105
第6章 使用PnP通知	85	8.1.1 ACPI BIOS	105
6.1 PnP通知概述	85	8.1.2 ACPI 驱动程序	105
6.2 编写PnP通知回调例程的准则	86	8.1.3 电源管理器	106
6.3 使用PnP设备接口改变通知	87	8.1.4 电源管理中驱动程序的作用	106
6.3.1 为设备接口改变通知注册	87	8.2 驱动程序的电源管理职能	107
6.3.2 处理设备接口改变事件	88	8.2.1 报告设备的电源能力	107
6.4 使用PnP目标设备改变通知	88	8.2.2 对电源管理设置设备对象标记	110
6.4.1 注册目标设备改变通知	88	8.2.3 处理电源IRP	110
6.4.2 处理一个GUID_TARGET_DEVICE_ _QUERY_REMOVE事件	89	8.2.4 设备的加电	114
6.4.3 处理一个GUID_TARGET_DEVICE_ REMOVE_COMPLETE事件	89	8.2.5 设备的掉电	114
6.4.4 处理一个GUID_TARGET_DEVICE_ REMOVE_CANCELLED事件	90	8.2.6 激活设备唤醒能力	114
6.5 使用PnP硬件配置文件改变通知	90	8.3 处理电源IRP的规则	115
6.5.1 注册硬件配置文件改变通知	90	8.3.1 使用PoCallDriver	115
6.5.2 处理硬件配置文件改变事件	91	8.3.2 传递电源IRP	115
6.6 使用PnP定制通知	91	8.3.3 设备休眠时排队I/O请求	117
第7章 支持多功能设备	93	8.3.4 处理未被支持的或者无法识别 的电源IRP	117
7.1 支持多功能PC Card设备	93	第9章 独立设备的电源管理	118
7.1.1 支持遵照多功能标准的PC Card	94	9.1 设备电源状态	118
		9.1.1 设备工作状态D0	119
		9.1.2 设备休眠状态D1、D2、和D3	119
		9.1.3 设备电源状态所需的支持	120
		9.2 管理设备电源策略	121
		9.3 为设备电源状态处理IRP_MN_SET_ _POWER	122
		9.3.1 处理设备掉电IRP	122

9.3.2 处理设备加电IRP	125
9.3.3 设备电源IRP的IoCompletion例程	126
9.4 为设备电源状态处理IRP_MN_	
QUERY_POWER	127
9.5 为设备电源状态发送IRP_MN_QUERY_	
POWER或者IRP_MN_SET_POWER	129
9.6 检测空闲的设备	130
9.6.1 用于空闲检测的电源管理器例程	131
9.6.2 执行设备特定的空闲检测	131
第10章 处理系统电源状态请求	132
10.1 系统电源状态	132
10.1.1 系统工作状态S0	133
10.1.2 系统休眠状态S1、S2、S3和S4	133
10.1.3 系统关机状态S5	134
10.1.4 系统电源动作	135
10.2 系统电源策略	135
10.3 防止系统电源状态变化	136
10.4 为系统电源状态处理IRP_MN_	
QUERY_POWER	136
10.4.1 使系统查询电源IRP失效	137
10.4.2 在设备电源策略所有者中处理	
系统查询电源IRP	137
10.4.3 在总线驱动程序中处理系统	
查询电源IRP	138
10.4.4 在过滤器驱动程序中处理系统	
查询电源IRP	139
10.5 为系统电源状态处理IRP_MN_	
SET_POWER	139
10.5.1 在设备电源策略所有者中	
处理系统设置电源IRP	140
10.5.2 在总线驱动程序中处理系统设置	
电源IRP	141
10.5.3 在过滤器驱动程序中处理系统设置	
电源IRP	141
第11章 支持具有唤醒能力的设备	143
11.1 等待/唤醒操作综述	143
11.1.1 确定设备是否能唤醒系统	144

11.1.2 理解通过设备树的等待/唤醒	
IRP的路径	145
11.1.3 完成等待/唤醒IRP综述	149
11.2 处理等待/唤醒IRP的步骤	150
11.2.1 在功能驱动程序或者过滤器驱	
动程序中处理等待/唤醒IRP	150
11.2.2 在总线驱动程序中处理等待/	
唤醒IRP	151
11.2.3 等待/唤醒IRP的IoCompletion例程	152
11.3 发送等待/唤醒IRP	152
11.3.1 确定何时发送等待/唤醒IRP	153
11.3.2 等待/唤醒IRP请求	153
11.3.3 等待/唤醒的回调例程	153
11.3.4 撤消等待/唤醒IRP	154
11.4 等待/唤醒IRP的Cancel例程	155

第四部分 设 置

第12章 设备安装概述	157
12.1 设备安装组件	157
12.2 PnP设备安装示例	159
12.3 Setup如何为设备选择驱动程序	161
12.4 系统设置阶段	164
第13章 为设备提供驱动程序	166
13.1 驱动程序文件	166
13.2 注册表中的驱动程序信息	167
13.3 指定驱动程序装载顺序	168
13.4 安装过滤器驱动程序	170
13.5 为设备安装Null驱动程序	171
第14章 建立INF文件	173
14.1 INF文件的一般准则	173
14.2 为设备文件说明源和目标位置	174
14.3 建立跨平台的和/或双操作系统	
的INF文件	175
14.4 建立国际化的INF文件	176
14.5 在设备的INF文件中加强打开文件	
的安全性	177
14.6 从应用程序访问INF文件	178

14.6.1 打开和关闭INF文件	178	设计目标	211
14.6.2 从INF文件提取信息	178	1.3.1 可移植性	212
第15章 编写协作安装程序	179	1.3.2 可配置性	212
15.1 协作安装程序概述	179	1.3.3 永远抢占优先和永远中断	213
15.2 协作安装程序界面	181	1.3.4 多处理器安全	214
15.3 协作安装程序操作	182	1.3.5 基于对象	216
15.4 注册协作安装程序	185	1.3.6 带有可重用IRP的包驱动I/O	216
15.4.1 注册设备专用的协作安装程序	185	1.3.7 支持异步I/O	217
15.4.2 注册类协作安装程序	186	第2章 分层的I/O、IRP和I/O对象	218
第16章 编写类安装程序	188	2.1 Windows 2000 I/O模型概述	218
16.1 类安装程序界面	188	2.2 终端用户I/O请求和Windows 2000 文件对象	219
16.2 设备安装函数小结	189	2.3 IRP和驱动程序特定的I/O栈位置	222
16.3 注册类安装程序	190	2.4 驱动程序对象和标准驱动程序例程	227
16.4 在类安装程序的INF文件中加强 文件打开的安全性	190	2.4.1 对象的不透明性	229
第17章 编写定制设备安装应用程序	191	2.4.2 标准驱动程序对象入口点	229
17.1 安装与驱动程序共用的软件实用程序	191	2.4.3 其他标准驱动程序例程	230
17.2 定制安装应用程序准则	191	2.4.4 标准驱动程序例程的注意事项	232
第18章 提供设备属性页	193	2.5 设备配置和分层驱动程序	232
18.1 设备属性页所需的支持	193	2.5.1 样例设备和驱动程序配置	233
18.2 处理DIF_ADDPROPERTYPAGE _ADVANCED请求	194	2.5.2 添加驱动程序的注意事项	234
18.3 属性页回调函数	195	2.6 设备对象和分层驱动程序	234
18.4 处理属性页的Windows消息	195	2.6.1 针对样本配置的设备对象	235
第19章 设备安装疑难解答	198	2.6.2 设备对象的注意事项	236
19.1 使用SetupAPI记录日志	199	第3章 系统定义的对象和对驱动程 序的支持	237
19.1.1 设置SetupAPI记录日志级别	199	3.1 系统组件和分层内核模式驱动程序	237
19.1.2 解释SetupAPI日志文件示例	201	3.1.1 操纵系统定义的对象的支持例程	238
19.2 显示设备管理器中的隐藏设备	203	3.1.2 存储系统定义的对象	240
第20章 安装一个要求重新启动机器的 设备	204	3.2 设备对象和设备扩展	241
		3.2.1 定义设备扩展	242
		3.2.2 创建设备对象和设备扩展	243
		3.2.3 初始化驱动程序特定的设备 对象和设备扩展	244
		3.2.4 设置用户缓冲区的访问	245
		3.3 适配器对象和DMA	251
		3.3.1 映射寄存器	252
第三篇 内核模式驱动程序设计指南			
第1章 Windows 2000和WDM驱动程序	208		
1.1 Windows 2000组件概述	208		
1.2 Windows 2000中的驱动程序种类	209		
1.3 Windows 2000和WDM驱动程序的			

3.3.2 获取适配器对象	254	4.4.1 设备对象命名和设备接口注册	313
3.3.3 拆分传输请求	255	4.4.2 为驱动程序例程选择名字	314
3.3.4 使用系统DMA	256	4.4.3 开始设计	314
3.3.5 使用总线控制器DMA	263	4.4.4 开始开发	316
3.3.6 执行发散/收集DMA	270	第5章 DriverEntry 和 Reinitialize 例程	319
3.4 控制器对象	271	5.1 DriverEntry例程需求	319
3.4.1 创建带有控制器扩展的控制器对象	272	5.1.1 DriverEntry例程语法与参数	319
3.4.2 为I/O操作分配控制器对象	273	5.1.2 DriverEntry返回值	320
3.5 中断对象	275	5.2 DriverEntry例程功能	320
3.5.1 获得系统指定的中断向量、 DIRQL和处理器掩码	276	5.2.1 DriverEntry例程必备功能	320
3.5.2 注册ISR	276	5.2.2 DriverEntry可选功能	321
3.6 DPC对象	278	5.3 Reinitialize例程需求与功能	322
3.6.1 注册和排队DpcForIsr例程	278	第6章 Dispatch例程	324
3.6.2 注册和排队CustomDpc例程	279	6.1 Dispatch例程需求	324
3.7 带有相关的DPC的定时器对象	281	6.1.1 Dispatch例程一般实现	325
3.7.1 注册和启用IoTimer例程	281	6.1.2 Dispatch例程条件实现	325
3.7.2 注册和排队CustomTimerDpc例程	282	6.2 Dispatch例程基本功能	326
3.8 设备队列对象和互锁队列	284	6.2.1 何时检查I/O栈位置	327
3.8.1 设置设备队列对象和排队IRP	285	6.2.2 如何在一个Dispatch例程中 完成一个IRP	327
3.8.2 设置一个互锁队列和排队IRP	286	6.2.3 何时在Dispatch例程中完成一个IRP	328
3.9 内核调度者对象	288	6.2.4 如何从一个Dispatch例程传送带 有效参数的IRP	329
3.9.1 定时器对象	290	6.2.5 为较低层驱动程序分配IRP	330
3.9.2 事件对象	292	6.2.6 处理异步传输请求	331
3.9.3 信号量对象	294	6.3 Dispatch例程设计	331
3.9.4 互斥体对象	296	6.3.1 DispatchCreate和DispatchClose功能	332
3.9.5 线程在调度者对象上等待的 报警和APC的处理	299	6.3.2 DispatchCleanup功能	334
3.10 回调对象	300	6.3.3 DispatchRead或DispatchWrite功能	335
3.10.1 定义回调对象	300	6.3.4 Dispatch(Internal)DeviceControl功能	339
3.10.2 使用驱动程序定义的回调对象	301	第7章 StartIo和队列管理例程	344
3.10.3 使用系统定义的回调对象	302	7.1 排队IRP	344
第4章 驱动程序基本结构	304	7.2 StartIo例程需求	345
4.1 标准驱动程序例程	304	7.2.1 设备驱动程序中的StartIo例程	345
4.2 最低层设备驱动程序的分段IRP处理	306	7.2.2 StartIo例程功能	347
4.3 中间层驱动程序的分段IRP处理	311	7.2.3 I/O操作设备编程	349
4.4 设计和开发一个驱动程序	313	7.2.4 较高层驱动程序中的StartIo例程	349

7.2.5 实现StartIo的注意事项	350	11.1 驱动程序控制例程	373
7.3 使用互锁队列或设备队列的需求	351	11.1.1 驱动程序控制例程语法和参数	373
7.3.1 驱动程序管理的队列的存储需求	351	11.1.2 从DriverControl例程返回	374
7.3.2 建立驱动程序管理的队列	352	11.1.3 设计有DriverControl例程的 驱动程序	374
7.3.3 在驱动程序管理的队列中排队IRP	352	11.1.4 存储需求	374
7.3.4 用驱动程序创建的线程管理互锁 队列	352	11.1.5 建立AdapterControl和Controller Control例程	375
7.3.5 管理补充设备队列	353	11.2 AdapterControl例程需求	376
第8章 中断服务例程	357	11.3 ControllerControl例程需求	376
8.1 ISR需求	357	第12章 Cancel例程	378
8.1.1 ISR性能	357	12.1 Cancel例程需求	378
8.1.2 其他需要的驱动程序例程	358	12.2 处理可撤消IRP	379
8.1.3 建立一个ISR	358	12.2.1 在传递IRP的驱动程序例程中 处理可撤消IRP	380
8.2 ISR基本功能	358	12.2.2 在处理请求的驱动程序例程中 处理可撤消IRP	380
8.3 重叠I/O操作的ISR功能	359	12.2.3 使用系统撤消自旋锁	381
第9章 DpcForIsr例程和CustomDpc例程	361	12.2.4 与Cancel例程一起使用驱动程序 提供的自旋锁	381
9.1 DpcForIsr和CustomDpc例程需求	361	12.3 Cancel例程功能	385
9.1.1 DpcForIsr例程与CustomDpc例程	361	12.3.1 带有StartIo例程的驱动程序中的 Cancel例程	386
9.1.2 DpcForIsr例程基本需求	362	12.3.2 不带有StartIo例程的驱动程序中 的Cancel例程	386
9.1.3 CustomDpc例程基本需求	362	12.4 处理可撤消IRP时的注意事项	387
9.1.4 其他需要的驱动程序例程	363	第13章 IoCompletion例程	389
9.1.5 建立DpcForIsr和CustomDpc例程	363	13.1 使用IoCompletion例程	389
9.2 DpcForIsr和CustomDpc功能	364	13.1.1 建立IoCompletion例程	389
9.2.1 使用缓冲I/O和满足设备控制IRP	365	13.1.2 调用IoSetCompletion Routine	390
9.2.2 直接I/O使用	365	13.1.3 为低层驱动程序分配IRP	392
9.2.3 实现一个DpcForIsr或CustomDpc 例程的注意事项	366	13.2 IoCompletion例程需要的功能	393
第10章 SynchCriticalSection例程	368	第14章 IoTimer和CustomTimerDpc例程	395
10.1 SynchCriticalSection例程需求	368	14.1 定时器例程需求	395
10.1.1 使用SynchCriticalSection例程	368	14.2 IoTimer例程功能	395
10.1.2 运行SynchCriticalSection例程	369	14.2.1 建立IoTimer例程	396
10.1.3 SynchCriticalSection例程性能需求	369	14.2.2 为IoTimer例程建立环境	396
10.2 SynchCriticalSection例程功能	369		
10.2.1 为I/O操作对设备编程	369		
10.2.2 维护有关中断驱动I/O操作的状态	370		
第11章 AdapterControl和Controller Control例程	373		

14.2.3	使用IoTimer例程	396
14.3	CustomTimerDpc例程功能	397
14.3.1	设置CustomTimerDpc例程	398
14.3.2	为CustomTimerDpc例程设置环境	398
14.3.3	使用CustomTimerDpc例程	399
第15章	Unload例程	401
15.1	Unload例程需求	401
15.2	Unload例程环境	401
15.3	Unload例程功能	402
15.3.1	PnP驱动程序的Unload功能	402
15.3.2	非PnP驱动程序的Unload功能	402
第16章	常见的驱动程序设计问题	405
16.1	管理硬件优先级	405
16.2	使用自旋锁	407
16.2.1	为自旋锁和被保护数据提供 存储空间	408
16.2.2	初始化自旋锁	409
16.2.3	调用使用了自旋锁的支持例程	409
16.2.4	快速释放自旋锁	410
16.2.5	使用自旋锁时防止错误或死锁 的出现	412
16.3	轮询设备	413
16.4	管理内存的使用	414
16.4.1	使用系统内存	415
16.4.2	使用内核栈	419
16.4.3	使用后备列表	420
16.4.4	只读内存保护	421
16.5	为DMA和PIO维护高速缓存的一致性	422
16.5.1	在DMA操作期间刷新高速 缓存数据	422
16.5.2	在PIO操作期间刷新高速缓存数据	423
16.6	错误记录和NTSTATUS值	423
16.6.1	调用IoAllocateErrorLogEntry	424
16.6.2	填充错误记录包	424
16.6.3	设置错误记录包中的NTSTATUS值	426
16.6.4	调用IoWriteErrorLogEntry	427
16.6.5	定义新的IO_ERR_XXX	427
16.6.6	定义私有NTSTATUS常量	427
16.7	处理可删除介质	428
16.7.1	响应来自文件系统的验证请求	428
16.7.2	通知文件系统可能的介质改变	429
16.7.3	检查设备对象中的标志	430
16.7.4	在中间层驱动程序中建立IRP	430
16.8	使设备对应用程序和驱动程序可用	431
16.8.1	注册设备接口	431
16.8.2	激活和禁止设备接口	432
16.8.3	使用设备接口	432
16.9	可分页代码和数据	433
16.9.1	使驱动程序代码可分页	433
16.9.2	锁住可分页代码或数据	435
16.9.3	对整个驱动程序分页	437
16.10	常见的驱动程序可靠性问题	438
16.10.1	缓冲I/O中的错误	439
16.10.2	引用用户空间地址时的错误	442
16.10.3	直接I/O中的错误	443
16.10.4	调用者输入和设备状态中的错误	444
16.10.5	Dispatch例程中的错误	446
16.10.6	多处理器环境中的错误	447
16.10.7	处理IRP时的错误	448
 第四篇 图形驱动程序设计指南 		
第一部分 图形驱动程序 		
第1章	图形系统概述	451
第2章	对图形驱动程序的GDI支持	452
2.1	从驱动程序的观点看GDI	452
2.1.1	作为应用程序图形语言的GDI	453
2.1.2	作为绘制引擎的GDI	453
2.2	GDI/驱动程序的分工	454
2.2.1	GDI和驱动程序的通信	454
2.2.2	PDEV协商	455
2.2.3	表面协商	456
2.3	GDI支持的服务	460
2.3.1	GDI对表面的支持	460

第四篇 图形驱动程序设计指南

第一部分 图形驱动程序

第1章 图形系统概述	451
第2章 对图形驱动程序的GDI支持	452
2.1 从驱动程序的观点看GDI	452
2.1.1 作为应用程序图形语言的GDI	453
2.1.2 作为绘制引擎的GDI	453
2.2 GDI驱动程序的分工	454
2.2.1 GDI和驱动程序的通信	454
2.2.2 PDEV协商	455
2.2.3 表面协商	456
2.3 GDI支持的服务	460
2.3.1 GDI对表面的支持	460